

Παραδοτέα

Έργο ΛΑΕΡΤΗΣ Υποέργο 2



ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ ΙΟΝΙΑ ΝΗΣΙΑ 2014-2020
Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

**Τρέχουσες Μετεωρολογικές Συνθήκες, Κλιματική
Μεταβλητότητα,
και Εκτίμηση Κινδύνου Δασικής Πυρκαγιάς στα Επτάνησα**

Συγγραφέας Δρ.Χαλβατζάρας Δημήτρης

Περιεχόμενα

1. Πρόλογος.....	6
2. Παραδοτέα.....	8
3. Επιχειρησιακή διάγνωση Μετεωρολογικών συνθηκών σε πραγματικό χρόνο.....	12
4. Υλοποίηση κεντρικού εξυπηρετητή του Επιχειρησιακού Δικτύου Μετεωρολογικών Σταθμών Ιονίου.....	15
4.1. Αρχιτεκτονική Server (κεντρικού εξυπηρετητή).....	17
4.2. Τεχνολογίες ανάπτυξης.....	20
4.2.1 Πλατφόρμα λογισμικού ανοιχτού κώδικα Docker.....	20
4.2.2 Web Server NGINX.....	25
4.2.3 Μονάδα ασφαλών συνδέσεων.....	27
4.2.4 Laravel.....	32
4.2.4.1. Διαδρομές.....	33
4.2.4.2. Ελεγκτές.....	33
4.2.4.3. Πρότυπα Blade.....	34
4.2.4.4. Eloquent	35
4.2.4.5. Εγκατάσταση Laravel Server.....	37
4.2.5 MySql.....	38
4.2.6 MODx.....	40
4.2.7 Node.js.....	42

4.2.8 VueJs.....	43
4.3. Αρχιτεκτονική web server Λαέρτης.....	46
4.3.1 Πακέτο <i>Models</i>	48
4.3.1.1. WeatherData.....	48
4.3.1.2. Channel.....	48
4.3.1.3. Station.....	49
4.3.1.4. StationChannel.....	49
4.3.2 Πακέτο <i>Services</i>	50
4.3.2.1. ExportFile.....	50
4.3.2.2. ParseFile.....	50
4.3.2.3. ProcessFile.....	50
4.3.3 Πακέτο <i>API</i>	52
4.3.3.1. WeatherDataController.....	52
4.3.3.2. StationController.....	53
4.3.3.3. ChannelController.....	53
4.3.3.4. RemoteSourceController.....	54
4.3.3.5. FileController.....	54
4.3.3.6. ModxController.....	54
4.3.4 Μοντέλο δεδομένων.....	55
4.3.4.1. iw_stations.....	56
4.3.4.2. iw_station_channel.....	56
4.3.4.3. iw_weatherdata.....	56
4.3.4.4. iw_files.....	56

4.3.4.5. iw_jobs.....	57
4.3.4.6. iw_failed_jobs.....	57
4.3.5 Πακέτο Policies.....	58
4.4. Διαχειριστικό σύστημα διαδικτυακού τόπου ionianweather.gr.....	59
4.4.1 Διαχείριση σελίδων.....	59
4.4.2 Διαχείριση Χρηστών.....	61
4.5. Διαχείριση μετεωρολογικών δεδομένων.....	63
4.5.1 Διαχείριση μετεωρολογικών σταθμών.....	63
4.5.2 Διαχείριση κλιματικών αρχείων.....	64
5. Σύστημα ανασκόπησης-προβολής αρχειακών δεδομένων.....	65
5.1. Προδιαγραφές συστήματος συνεχούς προβολής μετεωρολογικών αρχειακών δεδομένων.....	66
6. Υπολογισμός κλιματικών δεικτών – κλιματικών ανωμαλιών.....	68
6.1. Προδιαγραφές μονάδος.....	68
7. Αυτόματη ενσωμάτωση δεδομένων από εξωτερικές πηγές.....	70
7.1. Προδιαγραφές μονάδος.....	71
8. Βελτίωση προβολής “συνθηκών τώρα”	73
8.1. Υλοποίηση μονάδας διαχείρισης συνθηκών τώρα.....	74
9. Μονάδα ανίχνευσης δυσλειτουργιών.....	76
9.1. Προδιαγραφές μονάδος.....	77
10. Υπηρεσίες υποστήριξης παραδοτέων.....	80
10.1. Άμεση τεχνική υποστήριξη σε περίπτωση δυσλειτουργίας ή βλάβης του λογισμικού.....	80

10.2. Επιτόπια τεχνική υποστήριξη σε περίπτωση δυσλειτουργίας ή βλάβης του λογισμικού.....	82
10.3. Εκπαίδευση προσωπικού.....	84
11. Οδηγίες εγκατάστασης.....	85

1. Πρόλογος

Στα πλαίσια των Πακέτων Εργασίας του υποέργο 2 αναπτύχθηκαν ποικιλία μονάδων και συστημάτων τα οποία προσφέρουν υπηρεσίες οι οποίες εξυπηρετούν τα εξής:

- I. Την συνεχή τροφοδότηση (ανά λεπτό) των Μετεωρολογικών συνθηκών που επικρατούν κατά μήκος των Επτανήσων από ένα δίκτυο Μετεωρολογικών σταθμών, προκειμένου να γίνονται γνωστές σε πραγματικό χρόνο κυρίως οι συνθήκες βροχόπτωσης, ανέμου, θερμοκρασίας, υγρασίας, που προ-απαιτούνται για την διάγνωση και αντιμετώπιση πυρκαγιών και πλημμυρικών φαινομένων.

Πρέπει να σημειωθεί ότι το δίκτυο διαθέτει 14 υψηλής τεχνολογίας σταθμούς, τύπου μόνιμου ανεμομετρικού ιστού ύψους 10 m (σύμφωνα προς το πρότυπο του Διεθνούς Μετεωρολογικού Οργανισμού WMO), το οποίο έχει 5-πλασιάσει την πυκνότητα σταθμών εδάφους στα Επτάνησα. Όταν το δίκτυο λειτουργεί σε Επιχειρησιακή μορφή (operational mode) τροφοδοτεί διαρκώς μέσω ασύρματων επικοινωνιών του δικτύου GSM συνολικά 104 μετρούμενες τιμές ανά λεπτό, τις οποίες μπορεί να διαθέσει σε πραγματικό χρόνο, μέσω μιας αλυσίδας εμπλεκόμενων λογισμικών.

Με τις παρεμβάσεις του συγκεκριμένου έργου η Επιχειρησιακή λειτουργία του δικτύου βελτιστοποιήθηκε η διαρκή και κατάλληλου επιπέδου ετοιμότητα και λειτουργικότητα όλου του επιστημονικού εξοπλισμού καθώς και των περιφερειακών υποστηρικτικών διατάξεων.

- II. Την διάγνωση και μελέτη της στατιστικής συμπεριφοράς των βροχοπτώσεων, των πλημμυρικών βροχοπτώσεων, των χαρακτηριστικών του ανέμου, των θερμοκρασιών, και των ξηρών περιόδων στα Επτάνησα, όπως αυτή προκύπτει από τις λεπτομερείς καταγραφές του δικτύου σταθμών (περίπου 140.000 τιμές ανά ημέρα).

Επίσης, την αναγνώριση της εκδήλωσης κλιματικών ανωμαλιών στα Επτάνησα και την διάγνωση της έντασης τους.

Επίσης παρέχονται λειτουργίες για την πρόβλεψη των κλιματικών μεταβολών που θα πρέπει να αναμένονται στα Επτάνησα, μέσω εξειδικευμένων Μαθηματικών προσομοιώσεων (μοντέλων) κλιματικής μεταβλητότητας στον ευρύτερο χώρο του Ιονίου.

III. Εκτίμηση του κινδύνου πυρκαγιών, βασισμένη στα Μετεωρολογικά δεδομένα πραγματικού χρόνου.

Σε περιπτώσεις που υπάρχουν πυρκαγιές σε εξέλιξη, προσφέρονται λειτουργίες για την εκτίμηση της ταχύτητας και κατεύθυνσης διαδοσης τους.

Η βασική μεθοδολογία υποστήριξης αυτών των λειτουργιών βασίζεται σε ανάλυση από τα καταγεγραμμένα στοιχεία διαχρονικών πυρκαγιών στα Επτάνησα και την αντίστοιχη Στατιστική τους, και στην συνέχεια εντοπίζει και καταγράφει την θέση και την ποσότητα της καύσιμης ύλης και της δασικής βιομάζας στα Επτάνησα. Με βάση την παραπάνω πληροφορία και σε συνδυασμό με τα Κλιματικά δεδομένα, το έργο παράγει σενάρια (μοντέλα) στα οποία εντοπίζονται κατά την θερμή περίοδο του έτους οι περιοχές υψηλού κινδύνου και εκτιμάται ο τρόπος εξάπλωσης πιθανών πυρκαγιών σε αυτές τις περιοχές (Μοντέλα Καύσιμης Ύλης).

2. Παραδοτέα

Στα πλαίσια αυτού του έργου σχεδιαστήκαν και αναπτύχθηκαν διάφορες μονάδες λογισμικού οι οποίες καλύπτουν τις προδιαγραφές του έργου.

Όλος ο κώδικας ο οποίος παράχθηκε στα πλαίσια αυτού του έργου παραδίδεται σε συμπιεσμένο αρχείο τύπου ZIP ονομαζόμενο:

LAERTIS-2023-SERVER.zip.

Στις παρακάτω παραγράφους αναφέρονται οι διαδρομές των αρχείων όπως αυτές είναι διαμορφωμένες μέσα στο παραδοτέο zip αρχείο.

Τα πακέτα των λογισμικών που αναπτύχθηκαν (βάση του πίνακα συμμόρφωσης) αποτελούνται από διάφορες μορφές αρχείων τα οποία παραδίδονται και είναι τα εξής:

1. Πακέτο κώδικα πυρήνα server ο οποίος αποτελείται απο διαφορετικές μονάδες κώδικα (που εξυπηρετούν την διάσειση και διαχείριση δεδομένων) και καλύπτουν τις προδιαγραφές του πίνακα συμμόρφωσης όπως αυτές αναφέρονται στον αριθμό 1 (Αναβάθμιση κώδικα επικοινωνίας).

Επίσης καλύπτει μερικός (server side) τις προδιαγραφές που αναφέρονται (πίνακα συμμόρφωσης) στους αριθμούς 2,3,4,5,6

Αναλυτική περιγραφεί του παραδοτέου παρέχεται στο κεφάλαιο 4.

Η διαδρομές των βασικών αρχείων κώδικα που αναπτύχθηκε σε αυτό το έργο είναι οι εξής:

- /app/Actions
- /app/Console/Commands
- /app/Http/Controllers/Api

- /app/Http/Middleware
- /app/Http/Requests
- /app/Http/Resources
- /app/Jobs
- /app/Models
- /app/Policies
- /app/Services
- /config
- /routes

2. Πακέτο κώδικα ο οποίος καλύπτει τις προδιαγραφές του πίνακα συμμόρφωσης όπως αυτές αναφέρονται στον αριθμό 2 (Ανάπτυξη κώδικα ανασκόπησης).

Η διαδρομές των βασικών αρχείων κώδικα που αναπτύχθηκαν για την κάλυψη των συγκεκριμένων προδιαγραφών είναι οι εξής:

- /modx/assets/laertis/stations
- /modx/assets/laertis/stations/station.html
- /app/Http/Controllers/Api/WeatherDataController.php
- /app/Http/Controllers/ModxController.php

Αναλυτική περιγραφή του παραδοτέου παρέχεται στο κεφάλαιο 5.

3. Πακέτο κώδικα ο οποίος καλύπτει τις προδιαγραφές του πίνακα συμμόρφωσης όπως αυτές αναφέρονται στον αριθμό 3 (Ανάπτυξη κώδικα αυτόματης ενσωμάτωσης κλιματικών δεικτών).

Η διαδρομές των βασικών αρχείων κώδικα που αναπτύχθηκαν για την κάλυψη των συγκεκριμένων προδιαγραφών είναι οι εξής:

- /app/Jobs/ProcessFiles.php
- /app/Models/File.php
- /app/Models/WeatherData.php
- /app/Http/Controllers/Api/FileController.php
- /app/Services/ExportData.php
- /app/Services/ParseFile.php

Αναλυτική περιγραφή του παραδοτέου παρέχεται στο κεφάλαιο 6.

4. Πακέτο κώδικα ο οποίος καλύπτει τις προδιαγραφές του πίνακα συμμόρφωσης όπως αυτές αναφέρονται στον αριθμό 4 (Ανάπτυξη κώδικα αυτόματης ενσωμάτωσης προϊόντων παρατηρησιακών πλεγματικών δεδομένων πραγματικού χρόνου).

Η διαδρομές των βασικών αρχείων κώδικα που αναπτύχθηκαν για την κάλυψη των συγκεκριμένων προδιαγραφών είναι οι εξής:

- /app/Http/Controllers/Api/RemoteSourceController.php
- /app/Jobs/ProcessRemoteSource.php

Αναλυτική περιγραφή του παραδοτέου παρέχεται στο κεφάλαιο 7.

5. Πακέτο κώδικα ο οποίος καλύπτει τις προδιαγραφές του πίνακα συμμόρφωσης όπως αυτές αναφέρονται στον αριθμό 5 (Ανάπτυξη κώδικα εμφάνισης τιμών των μετρούμενων παραμέτρων σε δυο διακριτές χρονικές βάσεις).

Η διαδρομές των βασικών αρχείων κώδικα που αναπτύχθηκαν για την κάλυψη των συγκεκριμένων προδιαγραφών είναι οι εξής:

- /modx/assets/laertis/stations
- /modx/assets/laertis/stations/station.html

- /app/Http/Controllers/Api/WeatherDataController.php
- /app/Http/Controllers/ModxController.php

Αναλυτική περιγραφή του παραδοτέου παρέχεται στο κεφάλαιο 8.

6. Πακέτο κώδικα ο οποίος καλύπτει τις προδιαγραφές του πίνακα συμμόρφωσης όπως αυτές αναφέρονται στον αριθμό 6 (Ανάπτυξη κώδικα ανίχνευσης των ακόλουθων δυσλειτουργιών στο Επιχειρησιακού Δικτύου Μετεωρολογικών Σταθμών Ιονίου σε πραγματικό χρόνο).

Η διαδρομές των βασικών αρχείων κώδικα που αναπτύχθηκαν για την κάλυψη των συγκεκριμένων προδιαγραφών είναι οι εξής:

- /app/Policies/WeatherDataPolicy.php
- /app/Providers/BroadcastServiceProvider.php

Αναλυτική περιγραφή του παραδοτέου παρέχεται στο κεφάλαιο 9.

7. SQL αρχείο το οποίο χρησιμοποιείται για την αρχικοποίηση της βάσης δεδομένων. Η διαδρομή (και το όνομα) του αρχείου είναι:

- /laertis.sql

8. Αρχείο Docker configuration το οποίο χρησιμοποιείται για την παραγωγή docker image. Η διαδρομή (και το όνομα) του αρχείου είναι:

- /docker-compose.yml

9. Όσον αφορά τις προδιαγραφές που αφορούν τα είδη (του πίνακα συμμόρφωσης) 7,8 και 9 αποτελούν υπηρεσίες που δεσμεύουν και θα προφέρει ο ανάδοχος στα πλαίσια αυτού του έργου.

3. Επιχειρησιακή διάγνωση Μετεωρολογικών συνθηκών σε πραγματικό χρόνο

Ο βασικός σκοπός αυτού του έργου είναι η Επιχειρησιακή διάγνωση Μετεωρολογικών συνθηκών σε πραγματικό χρόνο. Για να επιτευχθεί αυτός ο σκοπός, στα πλαίσια αυτού του έργου αναπτύχθηκαν μονάδες λογισμικού οι οποίες καλύπτουν τους εξής στόχους:

1. Επιχειρησιακή λειτουργία και παραγωγή μετρήσεων από το δίκτυο των Μετεωρολογικών Σταθμών του Ιονίου Πανεπιστημίου (αρχιτεκτονικής Μετεωρολογικών Ιστών με ενεργειακή και τηλεπικοινωνιακή αυτονομία, για την διάθεση σε πραγματικό χρόνο (ανά λεπτό) των επικρατούντων Μετεωρολογικών συνθηκών στον χώρο των Επτανήσων, και ειδικότερα των εξής καθοριστικών για τις πλημμυρικές βροχοπτώσεις και την εκδήλωση πυρκαγιών, παραγόντων:
 - ✓ Άνεμος στα 10 m (μέτρο ταχύτητας και κατεύθυνση),
 - ✓ Ριπή ανέμου στα 10 m (μέτρο ριπής και κατεύθυνση),
 - ✓ Ρυθμός βροχόπτωσης ανά λεπτό,
 - ✓ Ύψος βροχόπτωσης ανά ώρα και ανά ημέρα,
 - ✓ Ισχύς Ηλιακής ακτινοβολίας (στο οπτικό και στα UV-A-B),
 - ✓ Θερμοκρασία (συμπεριλαμβανομένης της ημερήσιας μέγιστης και ελάχιστης τιμής),

- ✓ Υγρασία (σχετική και απόλυτη),
- ✓ Πίεση (και συνεπώς, πυκνότητα αέρα)

Οι θέσεις των σταθμών οι υποστηρίζοντε μέσα απο το δικτύου είναι οι εξής:

- ✓ Κέρκυρα: 4 σταθμοί (Αυλιώτες, Τεμπλόνη, Λίμνη Κορισίων, πόλη Κέρκυρας)
 - ✓ Παξοί: 1 σταθμός (Αγ. Ισαυρος),
 - ✓ Λευκάδα: 1 σταθμός (Λιμνοθάλασσα Λευκάδας),
 - ✓ Κεφαλλονιά: 3 σταθμοί (Αντυπάτα Ερισσού, Κηπούρια Παλικής, Σκάλα Πόρου)
 - ✓ Ζάκυνθος: 4 σταθμοί (Ακρωτήριο Σκινάρι, Καλαμάκι, Αγαλάς, και πόλη Ζακύνθου)
 - ✓ Ηλεία: 1 σταθμός (Ακρωτήριο Κατάκολο)
2. Συλλογή, οργάνωση, ψηφιακή αρχειοθέτηση και Διαδικτυακή διάθεση σε πραγματικό χρόνο των μετρούμενων Μετεωρολογικών συνθηκών, καθώς και χαρτών με τις επικρατούσες συνθήκες ανέμου, βροχόπτωσης, θερμοκρασίας, υγρασίας, και Ηλιοφάνειας, με ρυθμό ανανέωσης ανά λεπτό.
3. Παραγωγή Μηνιαίων Μετεωρολογικών συνόψεων για τον άνεμο και τα χαρακτηριστικά της μεταβλητότητάς του, την βροχόπτωση, την θερμοκρασία, την υγρασία, και την Ηλιοφάνεια στα Επτάνησα, που προαπαιτούνται για την ποσοτική γνώση και τον επιχειρησιακό σχεδιασμό αντιμετώπισης των πλημμυρικών φαινομένων και των επεισοδίων πυρκαγιών στα Επτάνησα.

Για να επιτευχθούν οι παραπάνω στόχοι πραγματοποιήθηκαν οι εξής κύκλοι εργασιών:

1. Επιτόπιες τεχνικές εργασίες επιχειρησιακής λειτουργίας και παραγωγής μετρήσεων από τους σταθμούς του δικτύου (εφαρμοζόμενες στους 104 Μετεωρολογικούς αισθητήρες, τους ψηφιακούς καταγραφείς, τις φωτοβολταϊκές γεννήτριες, τις διατάξεις τηλεπικοινωνιών, και τις περιφερειακές βοηθητικές διατάξεις κάθε σταθμού).
2. Υλοποίηση μονάδων επιπρόσθετου μονάδων λογισμικού για την πλατφόρμα λογισμικού διαχείρισης, επεξεργασίας, απεικόνισης, και διαδικτυακής διάθεσης δεδομένων πραγματικού χρόνου υπό μορφή ανεξάρτητων μονάδων κώδικα.
3. Προμήθεια τηλεπικοινωνιακών υπηρεσιών μετάδοσης αριθμητικών δεδομένων.
4. Τεχνική υποστήριξη επιχειρησιακής λειτουργίας: (α) συστήματος ροής δεδομένων προς τον κεντρικό server του δικτύου, (β) του κέντρου ελέγχου του δικτύου, (γ) του συστήματος διαδικτυακής διάθεσης σε πραγματικό χρόνο των επικρατούντων Μετεωρολογικών συνθηκών.
5. Μαθηματική επεξεργασία και ανάλυση των Μετεωρολογικών δεδομένων με έμφαση στα τοπικά χαρακτηριστικά του ανέμου, της βροχόπτωσης, και της θερμοκρασίας.

Στα επόμενα κεφάλαια παρουσιάζονται οι εργασίες και οι μονάδες λογισμικού που αναπτύχθηκαν για την επίτευξη των στόχων.

4. Υλοποίηση κεντρικού εξυπηρετητή του Επιχειρησιακού Δικτύου Μετεωρολογικών Σταθμών Ιονίου

Το συγκεκριμένο κεφάλαιο αφορά τις εργασίες και τα παραδοτέα που αντιστοιχούν στην τεχνική προδιαγραφή του πίνακα συμμόρφωσης νούμερο1

Στα πλαίσια αυτού του έργου αναπτύχθηκε κεντρικός εξυπηρετητής του Επιχειρησιακού Δικτύου Μετεωρολογικών Σταθμών Ιονίου με τους επί μέρους σταθμούς υπαίθρου (14 κατ' ελάχιστο). Η ανάπτυξη του κώδικα πραγματοποιήθηκε σε περιβάλλον Linux / Docker Container και γλώσσα PHP. Ο συγκεκριμένος εξυπηρετητής καλύπτει τις εξής ανάγκες:

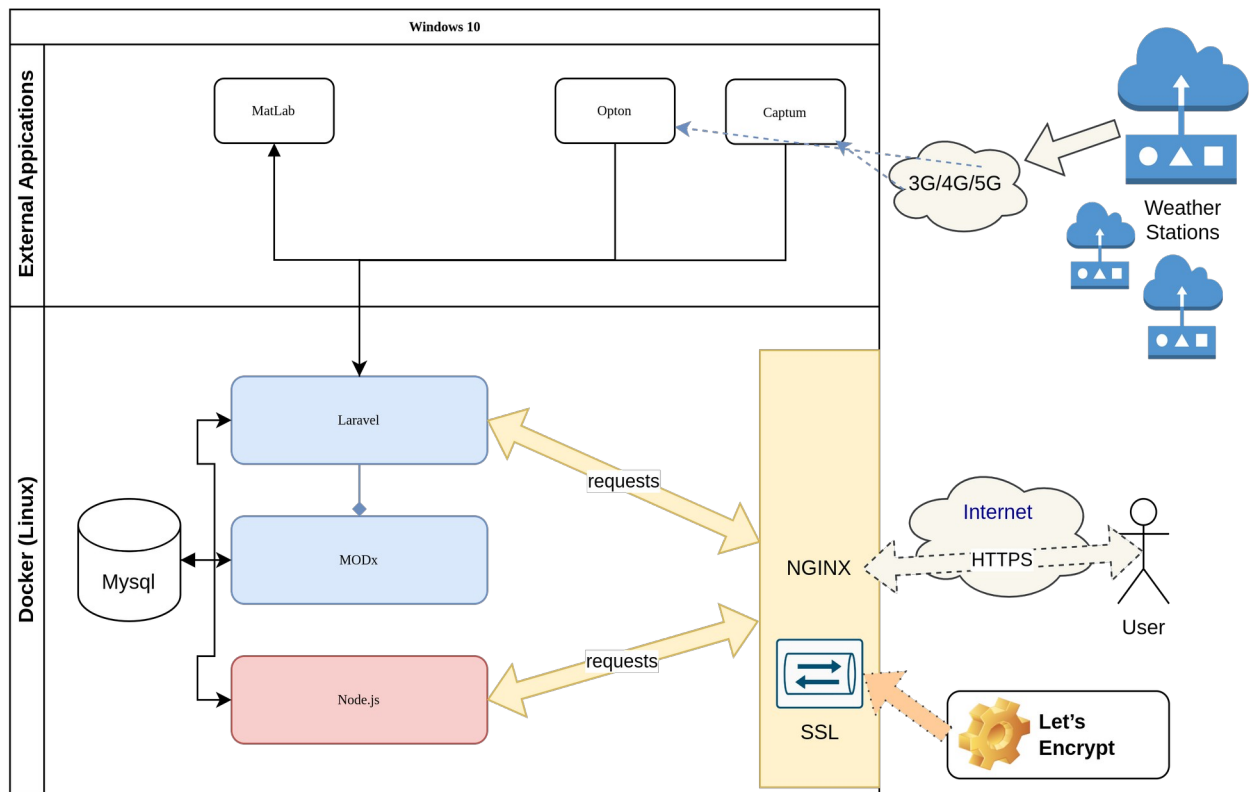
- Συνεχή αμφίδρομη επικοινωνία του server με τους ψηφιακούς καταγραφείς των σταθμών μέσω του συστήματος GSM υπό μορφή τοπικού δικτύου (LAN).
- Συνεχή παραλαβή και αρχειοθέτηση σε πραγματικό χρόνο των τιμών των μετρούμενων παραμέτρων από το σύνολο αισθητήρων του δικτύου (κατ' ελάχιστον 104 αισθητήρων) με ρυθμό παραλαβής μιας τιμής ανά λεπτό και μετρούμενη παράμετρο συμπεριλαμβανομένων στατιστικών δεικτών που παράγονται επιτόπια στους ψηφιακούς καταγραφείς για κάθε παράμετρο.
- Δυνατότητα παραμετροποίηση από τον διαχειριστή (μέσα από φιλικό διαδικτυακό γραφικό περιβάλλον UX) του συστήματος όσον αφορά:
 - Την ενεργοποίηση ή απενεργοποίηση εγγραφής στην βάση δεδομένων ανά σταθμό.

- Την ενεργοποίηση ή απενεργοποίηση εγγραφής στην βάση δεδομένων ανα κανάλι δεδομένων
- Δυνατότητα παραμετροποίηση από τον διαχειριστή (μέσα από φιλικό διαδικτυακό γραφικό περιβάλλον UX) της διάθεσης των ανά σταθμό και παράμετρο μετρούμενων τιμών στην ιστοσελίδα του δικτύου (επί του χάρτη αναφερόμενων τιμών πραγματικού χρόνου).
- Δυνατότητα συμπλήρωσης ελλειπουσών τιμών απο αρχεία που έχουν ανακτηθεί απο τα data loggers ασύγχρονα.
- Εξασφαλίζει την συμβατότητα με το περιβάλλον των νέων (τρεχουσών) εκδόσεων των λογισμικών Diamesson, Captum, και Opton της Symmetron και Windows 10 της Microsoft.

4.1. Αρχιτεκτονική Server (κεντρικού εξυπηρετητή)

Η αρχιτεκτονική και η τοπολογία του συνολικού συστήματός που αποτελεί τον Application Server του συγκεκριμένου έργου αναπτύχθηκε λαμβάνοντας υπόψη τις εξής προδιαγραφές:

- Cross Platform περιβάλλον λειτουργίας. Ο Server έχει αναπτυχθεί μέσα σε περιβάλλον container έτσι ώστε να είναι δυνατόν να έχει την δυνατότητα να εγκατασταθεί σε οποιοδήποτε λειτουργικό σύστημα συμπεριλαμβανομένων τα Windows της Microsoft.
- Ασφάλεια δεδομένων και αποτροπή κακόβουλων επιθέσεων.
- Δυνατότητα γεφύρωσης με άλλα λογισμικά όπως Matlab, Captum κ.λ.π.
- Δυνατότητα φιλοξενίας διαδικτυακών εφαρμογών αναπτυγμένων σε PHP και Node.js



Διάγραμμα 1 Τοπολογία Συστήματος Server

Η βασική τοπολογία των πακέτων απο τα οποία αποτελείται ο server είναι η εξής:

1. **Docker Linux** Container (βλέπε παράγραφο 4.2.1). Το Docker Linux Container αποτελεί το περιβάλλον μέσα απο το οποίο εκτελούνται και παρέχονται υπηρεσίες απο τα εξής πακέτα
 - I. **Laravel** Framework (βλέπε παράγραφο 4.2.4) το οποίο αποτελεί την υλοποίηση της λογικής του κεντρικού εξυπηρετητή.
 - II. **MODx** αποτελεί το πακέτο υλοποίησης του διαδικτυακού τόπου
 - III. **Node.js** αποτελεί εργαλείο εκτέλεσης ετεροχρονισμένων και διεργασιών του server (π.χ. εκτέλεση διαδικασιών backup)
2. **External Applications / Εξωτερικές εφαρμογές.** Η σχεδιάσει του server επιτρέπει την συνεργασία με εξωτερικές εφαρμογές (οι οποίες δεν αποτελούν παραδοτέο κομάτι του έργου) κάνοντας χρήση εξειδικευμένων

services και controllers. Στην συγκεκριμένη υλοποίηση η διασυνδεδεμένες εφαρμογές είναι οι εξής:

- I. MatLab (βλέπε παράγραφο 4.3.2.3)
- II. Captum
- III. Optum

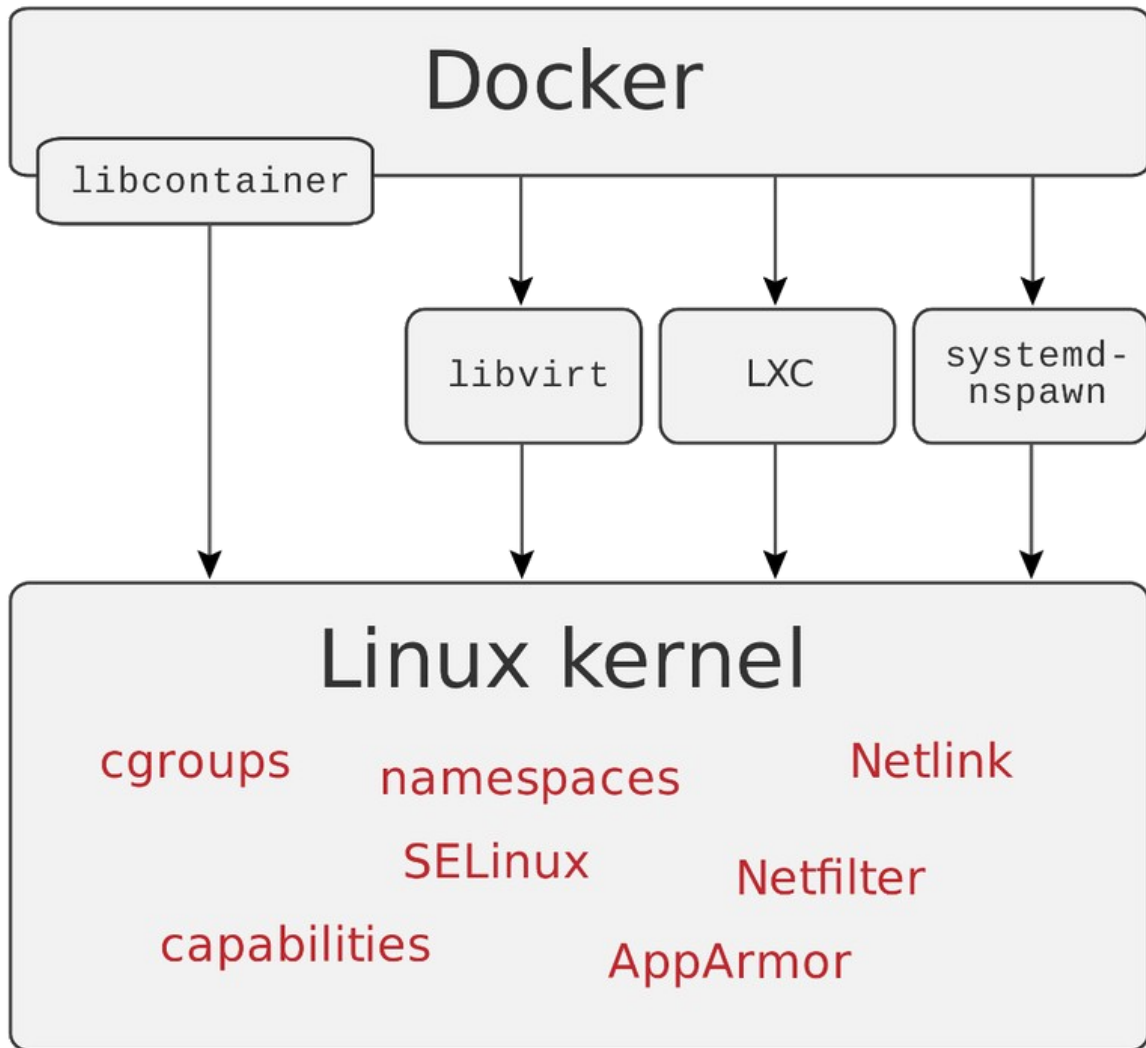
4.2. Τεχνολογίες ανάπτυξης

Στα πλαίσια αυτού του έργου και μετά απο έρευνα των τελευταίων τάσεων στην ανάπτυξη διαδικτυακών υπηρεσιών, επιλέχθηκαν οι καταλληλότερες τεχνολογίες και Frameworks με στόχο την βέλτιστη ανάπτυξη των συστημάτων που παρήχθησαν. Οι τεχνολογίες στις οποίες βασίστηκε οι ανάπτυξη των βασικών δομικών στοιχείων του συστήματος Server περιγράφονται στις παρακάτω παραγράφους.

4.2.1 Πλατφόρμα λογισμικού ανοιχτού κώδικα *Docker*

Το Docker είναι μια πλατφόρμα λογισμικού ανοιχτού κώδικα η οποία προσφέρει ένα κλειστό περιβάλλον λειτουργικού συστήματος, μέσα στο οποίο εγκαταστάθηκε και βρίσκεται σε λειτουργία το Ε.Δ.Μ.Σ.Ι. πραγματοποιήθηκε στο παρόν έργο.

Στο απομονωμένο περιβάλλον που προσφέρει το Docker έχει εγκατασταθεί μόνο ο απαιτούμενος κώδικας του Ε.Δ.Μ.Σ.Ι. και τα απαραίτητα παρελκόμενα αναγκαία συστήματα.

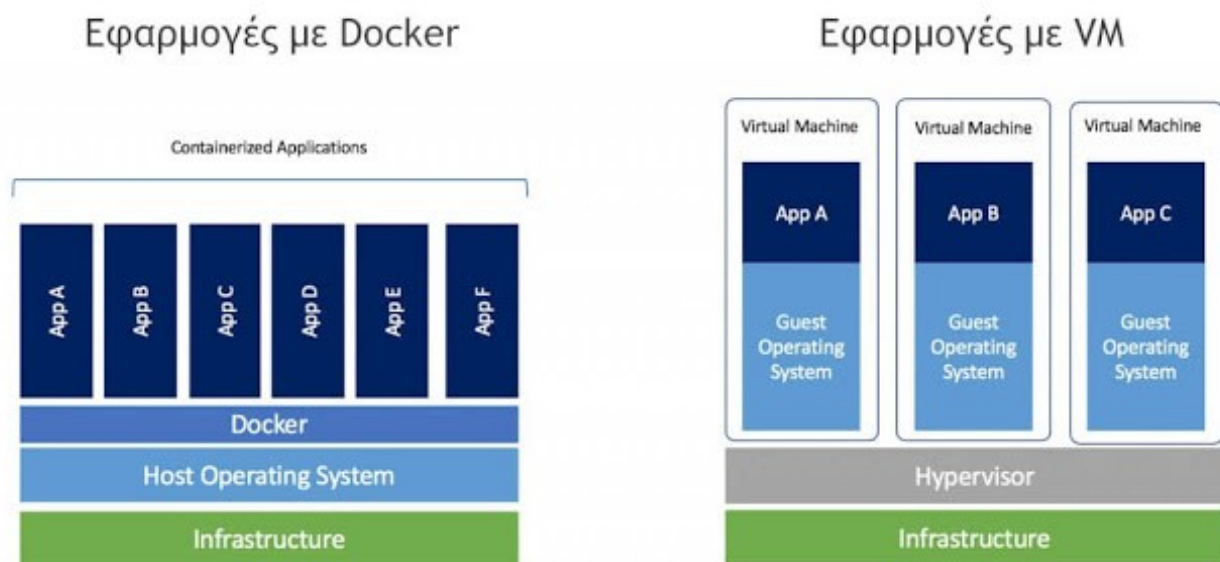


Εικόνα 1: Διάγραμμα πακέτων docker

Ενα επιπλέον πλεονέκτημα αυτής της αρχιτεκτονικής είναι η βέλτιστη χρήση της CPU, και της RAM του κεντρικού server του δικτύου σταθμών, το οποίο προσφέρει βελτιωμένη απόκριση του Ε.Δ.Μ.Σ.Ι. ιδιαίτερα κατά την στατιστική ανάλυση μεγάλων ομάδων μετρήσεων.

Κύριος στόχος του Docker είναι να αυτοματοποιήσει την διανομή εφαρμογών μέσα σε ασφαλής τομής λογισμικού (containers) καθώς και την αυτοματοποίηση του Virtualization των μικρο-υπηρεσιών του λειτουργικού συστήματος στο Linux.

Το Docker χρησιμοποιείται, σε αυτό το έργο, ως υπηρεσία με την οποία σε περιβάλλον Windows υλοποιήθηκαν και εκτελούνται διαδικασίες και κώδικες περιβάλλοντος Linux, ή γενικά εκτελούνται ακόμα και άγνωστης προέλευσης διαδικασίες και εφαρμογές χωρίς να κινδυνεύσει η ευστάθεια, τα επιμέρους τμήματα, και το λειτουργικό σύστημα του server του δικτύου σταθμών.



Εικόνα 2: Σύγκριση Docker με VM

Στο παρόν έργο, το Ε.Δ.Μ.Σ.Ι. καθώς και οι παρελκόμενες εφαρμογές οι εφαρμογές αυτές τρέχουν σε έναν ασφαλή τομέα (container), τα

περιεχόμενα του οποίου είναι απομονωμένα από το υπόλοιπο περιβάλλον του κεντρικού server του δικτύου σταθμών.

Τα οφέλη που προέκυψαν από την εφαρμογή της παραπάνω τεχνικής με χρήση του Docker σε σχέση με ένα "συμβατικό" server είναι ότι:

- το λειτουργικό σύστημα του server του διαδικτυακού συστήματος διαχείρισης των δεδομένων που παραλαμβάνονται από τους Μετεωρολογικούς Σταθμούς υπαίθρου είναι ασφαλές,
- Οποιαδήποτε λάθος ή μη-επιθυμητή διαδικασία σε κάποιο container δεν επηρεάζονται τα υπόλοιπα μέρη του συστήματος ούτε το λειτουργικό σύστημα του server,
- η δημιουργία και η διαγραφή του κάθε container καθίσταται απλή διαδικασία
- Μελλοντική ανάγκη "μετακόμισης" του Ε.Δ.Μ.Σ.Ι. σε άλλο server θα είναι πλέον εφικτή χωρίς να υπάρχει η ανάγκη εγκατάστασης άλλων λογισμικών παρά μόνο το Ε.Δ.Μ.Σ.Ι. image του docker το οποίο συμπεριλαμβάνει τον κώδικα του Ε.Δ.Μ.Σ.Ι. και όλες τα απαραίτητα συστήματα.

Επιπλέον , ένα container docker -σε αντίθεση με μια εικονική μηχανή και το LXC δεν απαιτεί ούτε περιλαμβάνει ξεχωριστό λειτουργικό σύστημα, αλλά αντίθετα, βασίζεται στη λειτουργικότητα του πυρήνα του Linux και χρησιμοποιεί απομόνωση πόρων. Τα container docker δημιουργούνται από στιγμιότυπα (images) docker που στην πράξη αντιπροσωπεύουν και λειτουργούν ως ζωντανή κατάσταση μιας διαδικτυακής εφαρμογής που

τρέχει από ένα αρχείο ISO που όμως είναι το ισοδύναμο του στιγμιότυπου του docker, και περιέχει μόνο την εφαρμογή και τις εξαρτήσεις της.

4.2.2 *Web Server NGINX*

Ο Nginx είναι (ανοικτού κώδικα), υψηλής απόδοσης web εξυπηρετητής. Ο Nginx είναι γνωστός για την υψηλή απόδοσή του, τη σταθερότητα, το πλούσιο σύνολο λειτουργιών, την απλή διαμόρφωση και την χαμηλή κατανάλωση πόρων. Αναπτύχθηκε για την αντιμετώπιση ζητημάτων απόδοσης και επεκτασιμότητας που σχετίζονται με τον Apache.

Χρησιμοποιείται για την υλοποίηση του κατανεμητή φόρτου εργασίας (Load Balancer) στο Cloud και εμπεριέχεται στο πακέτο LEMP stack. Σκοπός του είναι να κατανέμει το φόρτο των αιτημάτων που προέρχονται από τους κινητούς χρήστες της εφαρμογής.

Ο Nginx Server υποστηρίζει εγκατάσταση σε πολλά από τα γνωστά λειτουργικά συστήματα διακομιστών όπως τα Linux Server, Windows Server και Mac Server.

Κάποια από τα πιο σημαντικά χαρακτηριστικά του διακομιστή Nginx είναι:

- Υποστήριξη ασφαλών συνδέσεων SSL.
- Υποστήριξη CGI.
- Συμβατότητα με διευθύνσεις IPv6.
- Streaming αρχείων βίντεο MP4, FLV.
- Server Side Includes.
- Παρακολούθηση συμπεριφοράς χρηστών.

- Υποστήριξη εγκατάστασης πρόσθετων λειτουργιών με χρήση modules, όπως ένα firewall.
- Λειτουργία ως Mail Server για SMTP, POP και IMAP.

Ο διακομιστής Nginx σε σχέση με τον διακομιστή Apache κερδίζει σε αποδοτικότητα καθώς καταναλώνει λιγότερη μνήμη και μπορεί να χειριστεί 4 φορές περισσότερα αιτήματα εξυπηρέτησης, χάνοντας όμως σε ευελιξία αλλαγής ρυθμίσεων, όπως τα αρχεία .htaccess του Apache.

4.2.3 Μονάδα ασφαλών συνδέσεων

Η συγκεκριμένη μονάδα βρίσκεται ενσωματωμένη στον NGINX και υλοποιεί το πρωτόκολλο SSL .

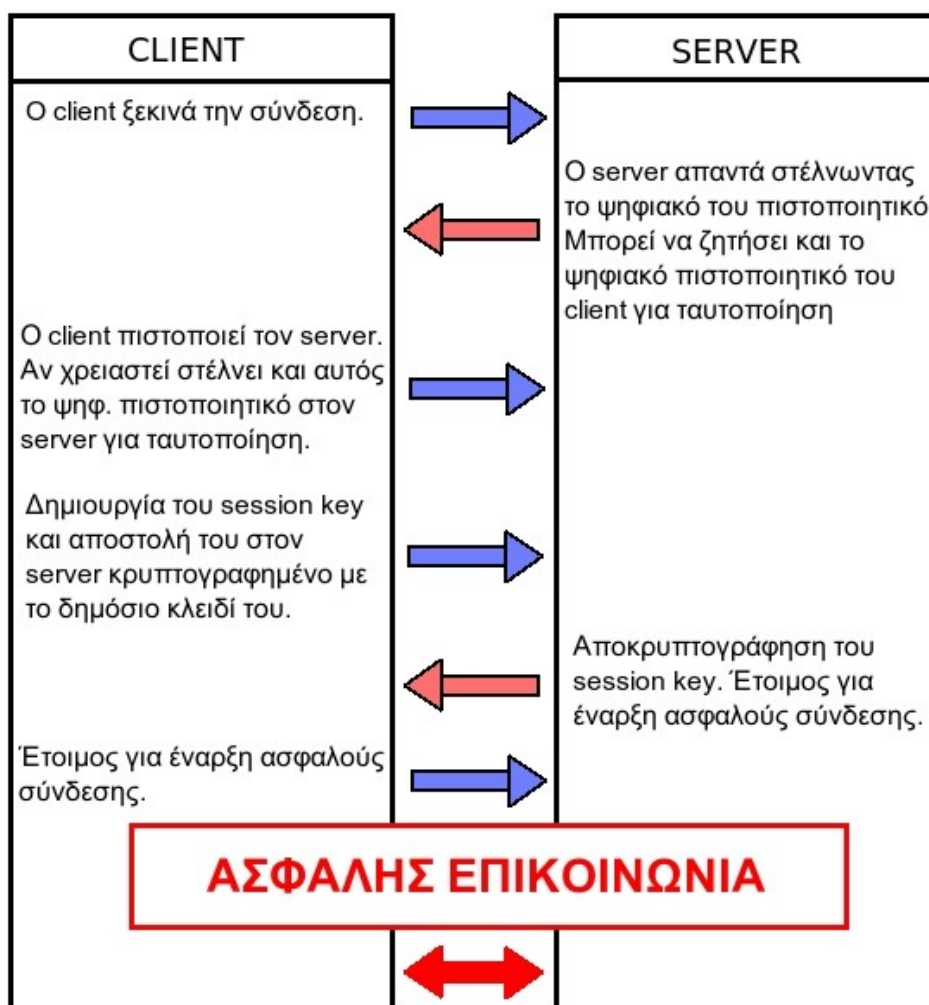
Το πρωτόκολλο SSL (Secure Sockets Layer) αναπτύχθηκε από την εταιρεία Netscape και σχεδιάστηκε για να παρέχει ασφάλεια κατά την μετάδοση ευαίσθητων δεδομένων στο διαδίκτυο. Το πρωτόκολλο χρησιμοποιείται ευρέως για ηλεκτρονικές αγορές και χρηματικές συναλλαγές, καθώς και όχι μόνον. Το SSL χρησιμοποιεί μεθόδους κρυπτογράφησης των δεδομένων που ανταλλάσσονται μεταξύ δύο συσκευών (συνηθέστερα Ηλεκτρονικών Υπολογιστών) εγκαθιδρύοντας μία ασφαλή σύνδεση μεταξύ τους μέσω του διαδικτύου. Το πρωτόκολλο αυτό χρησιμοποιεί το TCP/IP για τη μεταφορά των δεδομένων και είναι ανεξάρτητο από την εφαρμογή που χρησιμοποιεί ο τελικός χρήστης. Για τον λόγο αυτό μπορεί να παρέχει υπηρεσίες ασφαλούς μετάδοσης πληροφοριών σε πρωτόκολλα ανώτερου επιπέδου όπως για παράδειγμα το HTTP, το FTP, το telnet κοκ.

Η μετάδοση πληροφοριών μέσω του διαδικτύου γίνεται ως επί το πλείστον χρησιμοποιώντας τα πρωτόκολλα TCP/IP (Transfer Control Protocol / Internet Protocol). Το SSL λειτουργεί μετά το TCP/IP και πριν τις εφαρμογές υψηλού επιπέδου, όπως είναι για παράδειγμα το HTTP (προβολή ιστοσελίδων), το FTP (μεταφορά αρχείων) και το IMAP (email). Άρα το SSL συλλέγει τις πληροφορίες από τις εφαρμογές υψηλότερων επιπέδων, τις κρυπτογραφεί και στην συνέχεια τις μεταδίδει στο Internet προς τον Η/Υ πελάτη .

Τρόπος λειτουργίας

Το πρωτόκολλο SSL χρησιμοποιεί έναν συνδυασμό της κρυπτογράφησης δημοσίου και συμμετρικού κλειδιού. Η κρυπτογράφηση συμμετρικού κλειδιού είναι πολύ πιο γρήγορη και αποδοτική σε σχέση με την κρυπτογράφηση δημοσίου

κλειδιού, παρ' όλα αυτά όμως η δεύτερη προσφέρει καλύτερες τεχνικές πιστοποίησης. Κάθε σύνδεση SSL ξεκινά πάντα με την ανταλλαγή μηνυμάτων από τον server και τον client έως ότου επιτευχθεί η ασφαλής σύνδεση, πράγμα που ονομάζεται χειραψία (handshake). Η χειραψία επιτρέπει στον server να αποδείξει την ταυτότητά του στον client χρησιμοποιώντας τεχνικές κρυπτογράφησης δημοσίου κλειδιού και στην συνέχεια επιτρέπει στον client και τον server να συνεργαστούν για την δημιουργία ενός συμμετρικού κλειδιού που θα χρησιμοποιηθεί στην γρήγορη κρυπτογράφηση και αποκρυπτογράφηση των δεδομένων που ανταλλάσσονται μεταξύ τους.



Εικόνα 3: Διαδικασία αδειοδότησης

Η αδειοδότηση και έκδοση των πιστοποιητικών αυτών απαιτεί διεξοδικές

διαδικασίες, μέσα από τις οποίες οι Αρχές Πιστοποίησης ελέγχουν σχεδόν όλες τις πτυχές μιας εταιρίας και της ιστοσελίδας (εικόνα 3)

Certificate Authorities

Το SSL είναι μια ηλεκτρονική βεβαίωση που εκδίδεται (κατόπιν ελέγχων) από ανεξάρτητους και φερέγγυους οργανισμούς οι οποίοι ονομάζονται Αρχές Πιστοποιητικών (Certificate Authorities). Οι εταιρείες αυτές αναλαμβάνουν την ταυτοποίηση των στοιχείων της ιστοσελίδας καθώς και την ασφαλή μεταφορά δεδομένων μεταξύ των ιστοσελίδων αυτών και των χρηστών τους.

Στο συγκεκριμένο έργο η υπηρεσία η οποία χρησιμοποιήθηκε είναι η Let's Encrypt η οποία αποτελεί μια αυτοματοποιημένη και ανοιχτή αρχή έκδοσης πιστοποιητικών (CA), που λειτουργεί προς όφελος του κοινού. Είναι μια υπηρεσία που παρέχεται από το Internet Security Research Group (ISRG). Δίνει τα ψηφιακά πιστοποιητικά που χρειάζονται για να ενεργοποιηθεί το HTTPS (SSL / TLS) για ιστότοπους, με τον πιο φιλικό προς τον χρήστη τρόπο.

Οι λόγοι που επιλέχθηκε το Let's Encrypt είναι:

- Κόστος συντήρησης: Όποιος έχει ένα domain name μπορεί να χρησιμοποιήσει το Let's Encrypt για να αποκτήσει ένα αξιόπιστο πιστοποιητικό χωρίς να απαιτείται επαναλαμβανόμενη χρέωση για την έκδοση του.
- Αυτοματισμός: Το λογισμικό που εκτελείται σε έναν web server μπορεί να αλληλεπιδράσει με το Let's Encrypt για να αποκτήσει ένα πιστοποιητικό, διαμορφωμένο με ασφάλεια για χρήση και να αναλάβει αυτόματα την ανανέωσή του.
- Ασφάλεια: Η Let's Encrypt λειτουργεί ως πλατφόρμα για την προώθηση των βέλτιστων πρακτικών ασφάλειας TLS, τόσο από την πλευρά της CA όσο

και βοηθώντας τους χειριστές ιστότοπων να ασφαλίσουν σωστά τους διακομιστές τους.

- Διαφάνεια: Όλα τα πιστοποιητικά που εκδίδονται ή ανακαλούνται καταγράφονται δημόσια και είναι διαθέσιμα προς ανοικτή επιθεώρηση.
- Ανοικτό πρότυπο: Το πρωτόκολλο αυτόματης έκδοσης και ανανέωσης δημοσιεύτηκε ως ανοικτό πρότυπο.

Επικύρωση Τομέα (Domain)

Το Let's Encrypt προσδιορίζει τον διαχειριστή του διακομιστή με δημόσιο κλειδί. Την πρώτη φορά που το λογισμικό του πράκτορα αλληλεπιδρά με το Let's Encrypt, δημιουργεί ένα νέο ζεύγος κλειδιών και αποδεικνύει στο Let's Encrypt CA ότι ο διακομιστής ελέγχει έναν ή περισσότερους τομείς (domain). Αυτό είναι παρόμοιο με την παραδοσιακή διαδικασία CA για τη δημιουργία λογαριασμού και την προσθήκη τομέων σε αυτόν τον λογαριασμό.

Για να ξεκινήσει η διαδικασία, ο πράκτορας ρωτά το Let's Encrypt CA τι πρέπει να κάνει για να αποδείξει ότι ελέγχει το example.com. Το Let's Encrypt CA εξετάζει το ζητούμενο όνομα τομέα και εκδίδει ένα ή περισσότερα σύνολα προκλήσεων. Αυτοί είναι διαφορετικοί τρόποι με τους οποίους ο πράκτορας μπορεί να αποδείξει τον έλεγχο του τομέα. Για παράδειγμα, η CA ενδέχεται να δώσει στον πράκτορα την επιλογή ενός από τα εξής:

- Παροχή εγγραφής DNS στο ionianweather. ή
- Παροχή πόρου HTTP κάτω από ένα γνωστό URI στο <http://ionianweather.gr/>

Μαζί με τις προκλήσεις, το Let's Encrypt CA παρέχει επίσης την δυνατότητα στον πράκτορα να υπογράψει με το ζεύγος ιδιωτικών κλειδιών του για να αποδείξει ότι ελέγχει το ζεύγος κλειδιών .

Το λογισμικό του πράκτορα ολοκληρώνει ένα από τα παρεχόμενα σύνολα προκλήσεων. Αν είναι σε θέση να ολοκληρώσει τη δεύτερη εργασία τότε δημιουργεί ένα αρχείο σε μια καθορισμένη διαδρομή στον ιστότοπο ionianweather.gr. Ο πράκτορας υπογράφει επίσης την παρεχόμενη nonce (έναν αυθαίρετο αριθμό που μπορεί να χρησιμοποιηθεί μόνο μία φορά σε μια κρυπτογραφική επικοινωνία) με το ιδιωτικό κλειδί του. Μόλις ο πράκτορας ολοκληρώσει αυτά τα βήματα, ειδοποιεί την CA ότι είναι έτοιμη να ολοκληρώσει την επικύρωση.

Στη συνέχεια, είναι καθήκον της CA να ελέγχει εάν οι προκλήσεις έχουν ικανοποιηθεί. Η CA επαληθεύει την υπογραφή στο nonce και προσπαθεί να κατεβάσει το αρχείο από τον διακομιστή ιστού και να βεβαιωθεί ότι έχει το αναμενόμενο περιεχόμενο.

Εάν η υπογραφή μέσω του nonce είναι έγκυρη και οι προκλήσεις ελεγχθούν, τότε ο πράκτορας που προσδιορίζεται από το δημόσιο κλειδί είναι εξουσιοδοτημένος να κάνει διαχείριση πιστοποιητικών για το ionianweather.gr. Καλούμε το ζεύγος κλειδιών που ο πράκτορας χρησιμοποίησε ένα "ζεύγος εξουσιοδοτημένων κλειδιών" (authorized key pair) για το ionianweather.gr

4.2.4 *Laravel*

Το Laravel αποτελεί ένα από τα σημαντικότερα προγραμματιστικά framework PHP ανοιχτού κώδικα.

Μερικά από τα πιο σημαντικά χαρακτηριστικά είναι:

- ✓ Όργανωμένη, και ευκολο-ανάγνωστη σύνταξη
- ✓ Eloquent ORM (για εύκολη πρόσβαση στην βάση δεδομένων Mysql)
- ✓ Service Container / Providers
- ✓ Χρησιμοποιεί πολύ καλά και δοκιμασμένα τμήματα του Symphony framework
- ✓ Drivers για Caching / Queues / Mailers
- ✓ Δικό του Templating Engine (Blade)
- ✓ Queues (Background tasks)
- ✓ Easy Event Driven logic
- ✓ Filesystem / Cloud Storage
- ✓ Φιλικότητα Testing
- ✓ Helpers (DB Seeders)
- ✓ Elixir (Front end css/js minify-concatenation)

Οι βασικές έννοιες του Laravel αναλύονται παρακάτω:

4.2.4.1. Διαδρομές

Οι διαδρομές (routes) είναι αυτές που δείχνουν στο Laravel τι να εκτελέσει όταν πραγματοποιείται μια αίτηση σε ένα συγκεκριμένο URL. Οι διαδρομές ορίζονται στο αρχείο ./app/Http/routes.php. Οι διαδρομές ορίζονται με τον παρακάτω τρόπο: (Otwell, x.η.-e)

Παράδειγμα διαδρομών

```
Route::get('/', function () {  
    return 'Hello, World! (από GET)';  
});  
  
Route::post('/', function () {  
    return 'Hello, World! (από POST)';  
});
```

4.2.4.2. Ελεγκτές

Αντί να γράφεται ολόκληρη η λογική της ιστοσελίδας στο αρχείο των διαδρομών, μπορούν να δημιουργηθούν ελεγκτές (controllers) που θα εκτελούν τις ίδιες διαδικασίες. Οι ελεγκτές αποθηκεύονται στον φάκελο ./app/Http/Controllers. Η απλούστερη μορφή ενός ελεγκτή είναι η παρακάτω: (Otwell, x.η.-d)

Παράδειγμα ελεγκτή

```
<?php  
  
namespace App\Http\Controllers;  
  
use App\Http\Controllers\Controller;  
  
class SayHelloController extends Controller
```

```
{  
    public function sayHello($name)  
    {  
        return 'Γεια σου ' . $name;  
    }  
}
```

Για να χρησιμοποιηθεί ο ελεγκτής, πρέπει να οριστεί στο αρχείο routes.php ως εξής:

Χρήση ελεγκτή σε διαδρομή

```
Route::get('name/{name}', 'SayHelloController@sayHello');
```

4.2.4.3. Πρότυπα Blade

Η μηχανή προτύπων Blade παρέχει έναν εύκολο και εύχρηστο τρόπο διαχωρισμού του HTML κώδικα από τον PHP κώδικα (Otwell, κ.η.-b). Επίσης, επιτρέπει την επαναχρησιμοποίηση κάποιου master layout. Έτσι, το τμήμα κώδικα :

HTML μέσα σε PHP

```
Route::get('name/{name}', function ($name) {  
    $output = '  
        <!doctype html>  
        <html lang="el">  
            <head>  
                <meta charset="utf-8">  
                <title>Η ιστοσελίδα μου</title>  
            </head>  
            <body>
```

```
        Γεια σου ';\n$output .= $name;\n$output .= '</body>\n        </html>';\n\n    return $output;\n});
```

4.2.4.4. Eloquent

Το Eloquent είναι ένα σύστημα ORM (Object-relational mapping — Αντικείμενο σχεσιακή αντιστοίχιση) που κάνει εύκολη την διαχείριση της βάσης δεδομένων της ιστοσελίδας. Κάθε πίνακας της βάσης έχει ένα αντίστοιχο μοντέλο, που αλληλεπιδρά με τον συγκεκριμένο πίνακα. Το Eloquent κάνει εύκολη την άντληση και εισαγωγή δεδομένων από και προς τη βάση δεδομένων (Otwell, x.η.-c).

Σε περίπτωση που υπάρχει μελλοντικά η ανάγκη επέκτασης του συστήματος με νέ μοντέλα δεδομένων μπορεί να χρησιμοποιηθεί η εντολή *php artisan make:model*

```
$ php artisan make:model WeatherStation\nModel created successfully.
```

Ο κώδικας του μοντέλου που δημιουργείται είναι ο εξής:

/app/model/WeatherStation.php

```
<?php\n\nnamespace App;
```

```
use Illuminate\Database\Eloquent\Model;
```

```
class WeatherStation extends Model
```

```
{
```

```
    //
```

```
}
```

Η σύμβαση προδιαγράφει ως όνομα του μοντέλου τον ενικό αριθμό του ονόματος του πίνακα (π.χ. αν το όνομα του πίνακα είναι WeatherStations, το όνομα του μοντέλου θα πρέπει να είναι WeatherStation). Αυτό είναι δυνατόν να αλλάξει μέσω της μεταβλητής \$table στην κλάση του μοντέλου.

Χωρίς να έχει γραφτεί καθόλου κώδικας στην κλάση του μοντέλου, είναι δυνατόν να επεξεργαστούν τα στοιχεία του αντίστοιχου πίνακα στη βάση:

Παράδειγμα χρήσης Eloquent

```
$WeatherStations = WeatherStation::all();
```

```
// η μεταβλητή $WeatherStations περιέχει τώρα όλους τους
```

```
φοιτητές $newWeatherStation = new WeatherStation();
```

```
$newWeatherStation->first_name = "ΚΩΝΣΤΑΝΤΙΝΟΣ";
```

```
$newWeatherStation->last_name = "ΠΑΠΑΔΟΠΟΥΛΟΣ";
```

```
// ...
```

```
$newWeatherStation->save();
```

```
// έχει εισαχθεί μια καινούργια γραμμή στον πίνακα
```

WeatherStations // με τα στοιχεία του \$newWeatherStation

4.2.4.5. Εγκατάσταση Laravel Server

Αρχικά πρέπει να παραμετροποιηθούν αναγκαίες ρυθμίσεις που αφορούν το περιβάλλον εγκατάστασης.

Στον φάκελο στο οποίο θα εγκατασταθεί ο server του έργου θα υπάρχει ένα αρχείο με το όνομα .env. Το αρχείο αυτό περιέχει μερικές βασικές ρυθμίσεις.

```
APP_ENV=local
APP_DEBUG=true
APP_KEY=pyx2OXcKMPPhQB5McES7b7FoTO9w4UIS3

DB_HOST=localhost
DB_DATABASE= <database name>
DB_USERNAME= <database user name>
DB_PASSWORD= <database password>

CACHE_DRIVER=file
SESSION_DRIVER=file
QUEUE_DRIVER=sync

MAIL_DRIVER=smtp
MAIL_HOST=mailtrap.io
MAIL_PORT=2525
MAIL_USERNAME=null
MAIL_PASSWORD=null
MAIL_ENCRYPTION=null
```

4.2.5 *MySql*

MySQL είναι μια ανοικτού κώδικα σχεσιακή βάση δεδομένων που έχει αναπτυχθεί από την εταιρεία Oracle Corporation. Πρόκειται για μια από τις πιο δημοφιλείς βάσεις δεδομένων στον κόσμο και χρησιμοποιείται ευρέως από προγραμματιστές και οργανισμούς για την αποθήκευση και διαχείριση δεδομένων.

Αρχικά αναπτύχθηκε από τους Michael Widenius και David Axmark στη Σουηδία και κυκλοφόρησε για πρώτη φορά το 1995. Σήμερα, η MySQL υποστηρίζεται και συντηρείται από την Oracle Corporation, ενώ υπάρχουν και πολλά παράγωγα προϊόντα και διανομές που βασίζονται σε αυτήν.

Ορισμένα χαρακτηριστικά της MySQL περιλαμβάνουν:

1. Ανοιχτός κώδικας: Η MySQL είναι διαθέσιμη ως ελεύθερο λογισμικό με βάση τη General Public License (GPL), η οποία επιτρέπει στους χρήστες να την χρησιμοποιούν, να την τροποποιούν και να την διανέμουν χωρίς περιορισμούς.

2. Σχεσιακή δομή: Η MySQL υλοποιεί μια σχεσιακή δομή βάσης δεδομένων, όπου τα δεδομένα οργανώνονται σε πίνακες που περιέχουν σειρές και στήλες.

3. Απλότητα: Είναι σχετικά εύκολο στη χρήση και τη διαχείριση, καθιστώντας την κατάλληλη για μικρομεσαίες επιχειρήσεις και ανάπτυξη εφαρμογών.

4. Αποτελεσματική απόδοση: Η MySQL είναι γνωστή για την καλή της απόδοση, ιδιαίτερα όταν χρησιμοποιείται σε μικρές και μεσαίες βάσεις δεδομένων.

5. Υποστήριξη διαφόρων πλατφορμών: Η MySQL είναι πολυπλατφορμική και υποστηρίζει διάφορα λειτουργικά συστήματα, όπως Windows, macOS και πολλές διανομές Linux.

6. Ευελιξία: Μπορεί να χρησιμοποιηθεί για διάφορες εφαρμογές, από μικρές ιστοσελίδες έως μεγάλες επιχειρηματικές εφαρμογές.

Η MySQL είναι ένα ισχυρό εργαλείο για τη διαχείριση και αλληλεπίδραση με δεδομένα, και υποστηρίζεται από πολλές γλώσσες προγραμματισμού (όπως PHP, Python, Java κ.λπ.) για να ενσωματώσετε τη λειτουργικότητά της στις εφαρμογές σας. Επίσης, η MySQL παρέχει προηγμένες λειτουργίες όπως ασφάλεια, αντιγραφή ασφαλείας, διαχείριση δικαιωμάτων για να καλύψει τις ανάγκες πολύπλοκων εφαρμογών.

4.2.6 MODX

MODX είναι μια ανοικτού κώδικα πλατφόρμα διαχείρισης περιεχομένου (Content Management System - CMS) που χρησιμοποιείται για την δημιουργία και την διαχείριση ιστοσελίδων. Το MODX παρέχει ένα ευέλικτο περιβάλλον για την ανάπτυξη ιστοσελίδων και εφαρμογών, συνδυάζοντας την ευκολία χρήσης με προηγμένες δυνατότητες προσαρμογής.

Το MODX βασίζεται σε μια αρχιτεκτονική που διαχωρίζει σαφώς το περιεχόμενο (content) από την παρουσίαση (presentation) και την λογική της εφαρμογής (application logic).

Ο πυρήνας του MODX παρέχει ένα πλήθος λειτουργιών, όπως δυνατότητες διαχείρισης περιεχομένου, διαχείριση χρηστών και αδειών, αναζήτηση, ευέλικτα συστήματα δρομολόγησης (routing), προσαρμογή των μεταδεδομένων (metadata), διαχείριση αρχείων και πολλά άλλα. Επιπλέον, το MODX είναι επεκτάσιμο μέσω πρόσθετων (add-ons) που παρέχουν επιπλέον λειτουργικότητα και δυνατότητες.

Η πλατφόρμα MODX έχει σχεδιαστεί για να είναι ευέλικτη και προσαρμόσιμη, επιτρέποντας στους προγραμματιστές να δημιουργήσουν προηγμένες ιστοσελίδες και εφαρμογές .

Το MODX αποτελεί ένα framework τύπου CMS. Τα πλεονεκτήματα του είναι τα εξής:

- Σύστημα διαχείρισης χρηστών το οποίο συμπεριλαμβάνει μονάδα διαχείρισης δικαιωμάτων ανα χρήστη ή ομάδα χρηστών.

Η διαχείριση των χρηστών (τών ρόλων και των δικαιωμάτων) προσφέρεται μέσα από προηγμένο UX (παραθυρικό περιβάλλον)

- Σύστημα διαχείρισης παραμέτρων συστήματος μέσα από προηγμένο UX (παραθυρικό περιβάλλον)
- Σύστημα διαχείρισης επιλογών πλοήγησης (π.χ. μενού) από προηγμένο UX (παραθυρικό περιβάλλον)
- Σύστημα διαχείρισης διαδικτυακού περιεχομένου το οποίο χρησιμοποιείται από τον διαδικτυακό χώρο του έργου.

Το περιεχόμενο είναι εύκολα προσβάσιμο για διαχείριση μέσα από προηγμένο UX (παραθυρικό περιβάλλον). Οι δυνατότητες που προσφέρονται είναι οι εξής:

- Διάρθρωση δενδροειδής δομής περιεχομένου
- Διαχείριση κειμένων
- Διαχείριση αρχείων
- Διαχείριση δικαιωμάτων προσβασιμότητας
- Διαχείριση web components π.χ. χάρτης

Συνοψίζοντας, το MODX είναι μια πλατφόρμα CMS που συνδυάζει ευκολία χρήσης με προηγμένες δυνατότητες προσαρμογής. Αναπτύσσεται από μια ενεργή κοινότητα προγραμματιστών και παρέχει ένα πλήρες περιβάλλον για την δημιουργία ιστοσελίδων και εφαρμογών.

Το MODx στο συγκεκριμένο έργο βρίσκεται εγκαταστημένο σαν μονάδα (Module) στο Laravel και υποστηρίζεται από εξειδικευμένο controller (παράγραφο 4.3.3.6)

Η συγκεκριμένη αρχιτεκτονική προσφέρει αυξημένο επίπεδο ασφάλειας καθώς το Lagavel μεσολαβεί ως Firewall και προλαμβάνει κακόβουλες επιθέσεις.

4.2.7 *Node.js*

Node.js είναι μια ανοικτού κώδικα, ελεύθερης χρήσης, διασύνδεσης προγραμματισμού εκτέλεσης πλευράς διακομιστή, η οποία βασίζεται στη μηχανή JavaScript V8 της Google. Αποτελεί ένα περιβάλλον εκτέλεσης που επιτρέπει την εκτέλεση κώδικα JavaScript έξω από τον περιηγητή (browser) και στον διακομιστή (server-side). Αυτό καθιστά το Node.js ένα ισχυρό εργαλείο για την ανάπτυξη διακομιστών, δικτυακών εφαρμογών και backend λογισμικού.

Οι βασικές χαρακτηριστικές πτυχές του Node.js περιλαμβάνουν:

- Χωρίς εμπλοκή, μη-σύγχρονος φυλλομετρητής (non-blocking, asynchronous): Ο Node.js χρησιμοποιεί μια αρχιτεκτονική μη-εμπλοκής ειδικά σχεδιασμένη για να υποστηρίξει μη-σύγχρονες λειτουργίες εισόδου/εξόδου (I/O). Αυτό σημαίνει ότι ο κώδικας δεν περιμένει μπλοκαρισμένος κατά την εκτέλεση εντολών εισόδου/εξόδου, αλλά μπορεί να συνεχίσει να εκτελεί άλλες λειτουργίες κατά τη διάρκεια αυτής της περιόδου.
- Ανταπόκριση σε γεγονότα (Event-driven): Ο Node.js βασίζεται στο μοντέλο ανταπόκρισης σε γεγονότα, όπου καλείται να ανταποκριθεί σε γεγονότα και εισερχόμενα αιτήματα. Χρησιμοποιείται ένα

ενεργητικό σύστημα ειδοποίησης (event loop) που αναμένει και διαχειρίζεται αυτά τα γεγονότα.

- **Επεκτασιμότητα:** Το Node.js υποστηρίζει μια πληθώρα επεκτάσεων (modules) που μπορούν να εισαχθούν και να χρησιμοποιηθούν για να προσθέσουν λειτουργικότητα στον κώδικα της εφαρμογής.
- **Διαχείριση πακέτων (Package management):** Το Node.js χρησιμοποιεί το NPM (Node Package Manager) για τη διαχείριση εξαρτήσεων και πακέτων που χρησιμοποιούνται σε ένα έργο.

Χάρη στην αποτελεσματικότητα του στη μη-μπλοκαριστή και μη-σύγχρονη αρχιτεκτονική του, το Node.js είναι εξαιρετικά κατάλληλο για δικτυακές εφαρμογές που απαιτούν υψηλές αποκρίσεις σε πραγματικό χρόνο, όπως συζήτηση σε πραγματικό χρόνο, συνεργατικά περιβάλλοντα, παιχνίδια και πολλά άλλα. Επιπλέον, με τη βοήθεια του Node.js, οι προγραμματιστές μπορούν να αναπτύξουν ολοκληρωμένες εφαρμογές διακομιστή χρησιμοποιώντας μόνο τη γλώσσα JavaScript, η οποία συμβάλλει στη μείωση του χρόνου ανάπτυξης και της πολυπλοκότητας του έργου.

4.2.8 *VueJs*

Στα πλαίσια αυτού του έργου το VueJs έχει χρησιμοποιηθεί για την ανάπτυξη μονάδων (κυρίως web components) οι οποίες απαρτίζουν το περιβάλλον εργασίας του χρήστη (client-side).

Το Vue.js είναι ένα προοδευτικό JavaScript πλαίσιο εφαρμογών που χρησιμοποιείται για την ανάπτυξη διεπαφών χρήστη. Έχει σχεδιαστεί για να είναι ελαφρύ, ευέλικτο και εύκολο στη χρήση, και επιτρέπει την ομαλή και αποδοτική δημιουργία μονοσέλιδων εφαρμογών.

Ο στόχος του Vue.js είναι να επιτρέψει τη δημιουργία επαναχρησιμοποιήσιμων συνιστωσών, τις οποίες ονομάζει "components". Οι συνιστώσες είναι αυτόνομα, ανεξάρτητα μέρη της διεπαφής χρήστη που μπορούν να συνδυαστούν για να δημιουργήσουν πλήρεις εφαρμογές. Κάθε συνιστώσα μπορεί να έχει τη δική της λογική, την κατάσταση της και την προβολή της.

Ο Vue.js χρησιμοποιεί μια σύνταξη προτύπου HTML, που ονομάζεται "template syntax", για να παρέχει δυνατότητες δυναμικής δέσμευσης δεδομένων. Αυτό σημαίνει ότι μπορείτε να δηλώσετε το πώς τα δεδομένα θα προβληθούν στη διεπαφή χρήστη και πώς θα αντιδράσει η διεπαφή όταν τα δεδομένα αλλάξουν.

Ένα ακόμη σημαντικό χαρακτηριστικό του Vue.js είναι η δυνατότητα διαχείρισης κατάστασης (state management). Ο Vue.js παρέχει ένα κατάσταση κεντρικού αποθετηρίου, το οποίο μπορεί να χρησιμοποιηθεί για την αποθήκευση και την ανανέωση των δεδομένων που χρησιμοποιούνται από τις συνιστώσες.

Επιπλέον, το Vue.js έχει μια ισχυρή κοινότητα υποστήριξης και πολλά παραδείγματα κώδικα και επέκτασης που μπορείτε να χρησιμοποιήσετε για την ανάπτυξη των εφαρμογών σας.

Το Vue.js είναι ένα ευέλικτο και αποδοτικό πλαίσιο εφαρμογών JavaScript που σας επιτρέπει να δημιουργήσετε επαναχρησιμοποιήσιμα συνιστώσες, να διαχειρίζεστε την κατάσταση των δεδομένων και να αναπτύσσεται πολύπλοκες μονοσέλιδες εφαρμογές με ευκολία.

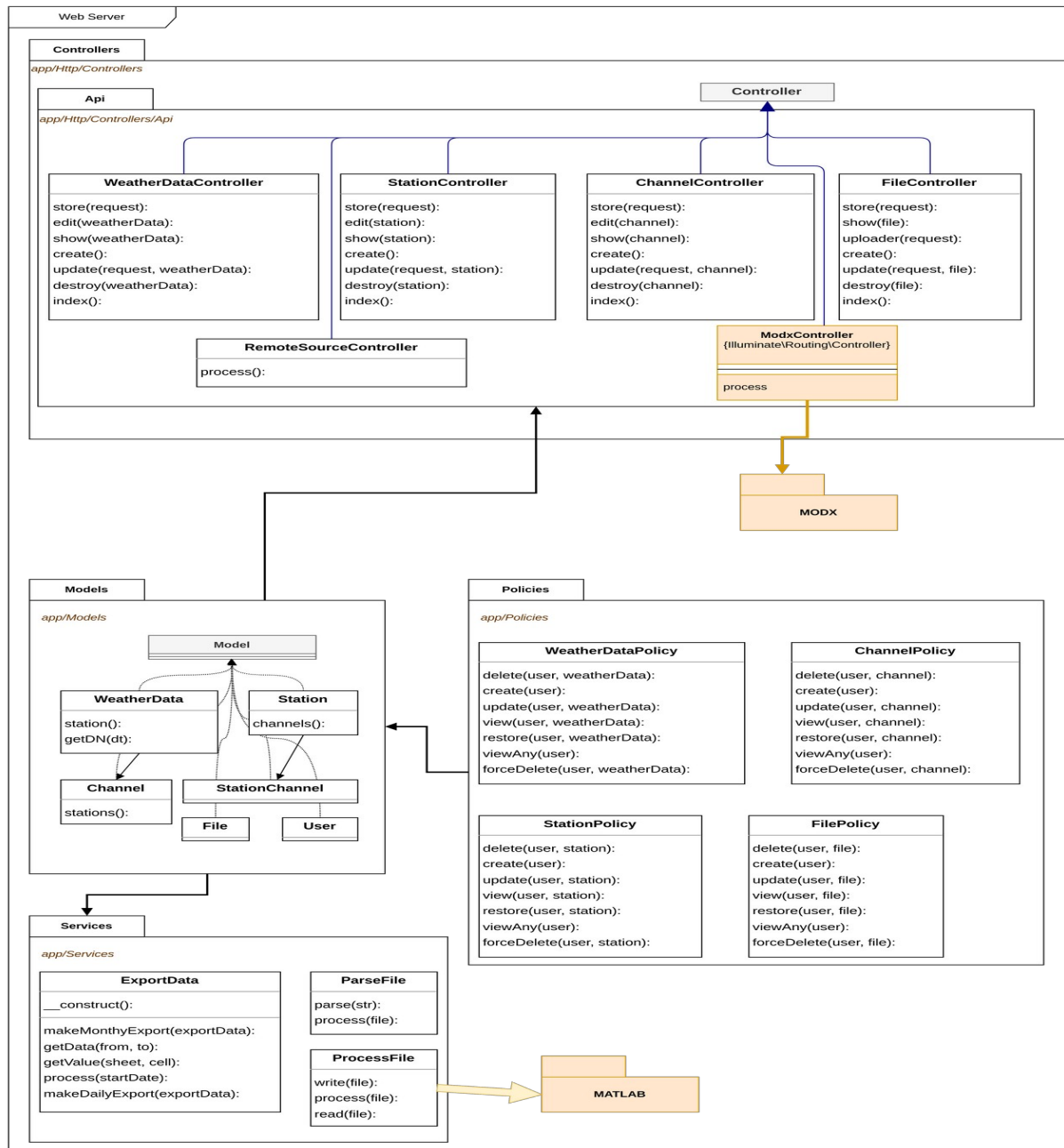
Στο συγκεκριμένο έργο το Vue.js επιλέχθηκε καθότι κατατάσσεται στα πλαίσια αντιδραστικού προγραμματισμού (Reactive Framework)

Όταν δημιουργείτε μια εφαρμογή με σύνθετους υπολογισμούς, τα Reactive Framework είναι μια εξαιρετική επιλογή.

Η διαχείριση μεγάλου όγκου δεδομένων, με διαφορετικές υπηρεσίες και λειτουργίες αποτελούν το κυριότερο τοπίο εφαρμογής των reactive framework. Οι αντιδραστικές βιβλιοθήκες παρέχουν βοηθητικές μεθόδους που διευκολύνουν τη διαχείριση αυτής της ροής συμβάντων. Μπορεί επίσης να χειριστεί ασύγχρονα συμβάντα

Ο αντιδραστικός προγραμματισμός λειτουργεί με βάση ροές γεγονότων ή «γεγονότων». Αυτά τα γεγονότα συμβαίνουν σε διάφορες χρονικές στιγμές. Εάν συμβεί κάποιο συμβάν, ένας συνδρομητής θα ειδοποιηθεί. Υπάρχουν διαφορετικές υλοποιήσεις αντιδραστικών συστημάτων, αλλά όλα έχουν έναν εκπομπό συμβάντων με επίκεντρο τα δεδομένα. Το Reactive javascript προσφέρει δύο κύριους τύπους αντιδραστικότητας, οι οποίοι εξελίχθηκαν ως απάντηση σε διαφορετικά προβλήματα. Ενώ έχουν τις ίδιες βασικές ιδιότητες, διαφέρουν ως προς τον τρόπο μοντελοποίησης τους.

4.3. Αρχιτεκτονική web server λαέρτης



Διάγραμμα 2: UML αρχιτεκτονικής

Η εφαρμογή web server έχει αναπτυχθεί ακολουθώντας την δομή του Framework Laravel.

Στις επόμενες παραγράφους παρουσιάζονται τα πακέτα που απαρτίζουν τον web server μαζί με τις βασικότερες οντότητες κλάσεων που αναπτύχθηκαν.

4.3.1 Πακέτο *Models*

Το συγκεκριμένο πακέτο αποτελείται από την υλοποίηση κλάσεων (οι οποίες επεκτείνουν την κλάση *Illuminate\Database\Eloquent\Model* του Laravel) που αφορούν την διαχείριση δεδομένων και καλύπτουν τις εξής βασικές ανάγκες:

- Αποθήκευση/Ανάγνωση/Διαγραφή στην βάση δεδομένων (CRUD διεργασίες)
- Επεξεργασία δεδομένων

Οι βασικότερες κλάσεις που έχουν υλοποιηθεί είναι οι εξής:

4.3.1.1. WeatherData

Η συγκεκριμένη κλάση διαχειρίζεται τα μετα επεξεργασμένα δεδομένα τα οποία παρέχονται στον τελικό χρήστη μέσα απο την διαδικτυακή σελίδα ionianweather.gr.

Το μοντέλο WeatherData συμπεριλαμβάνει μετα δεδομένα της περιοχής που άπτονται των κλιματικών δεδομένων που αντιπροσωπεύει.

Κάθε οντότητα WeatherData αποτελείται απο συλλογή μοντέλων Channel

4.3.1.2. Channel

Η συγκεκριμένη κλάση διαχειρίζεται τα μετα επεξεργασμένα δεδομένα για κάθε κλιματικό " κανάλι", δηλαδή για κάθε διαφορετικό τύπο κλιματικών δεδομένων π.χ. θερμοκρασία, ταχύτητα ανέμου κ.α. Τα δεδομένα του κάθε καναλιού παρέχονται στον τελικό χρήστη μέσα απο την διαδικτυακή σελίδα ionianweather.gr.

Το μοντέλο Channel ανήκει σε κάποια υλοποίηση του WeatherData μοντέλου.

4.3.1.3. Station

Η συγκεκριμένη κλάση διαχειρίζεται τα (raw) δεδομένα όπως αυτά συλλέγονται από τους μετεωρολογικούς σταθμούς.

Το μοντέλο Station συμπεριλαμβάνει μετα δεδομένα της περιοχής που στην οποία βρίσκεται εγκατεστημένος ο μετεωρολογικός σταθμός.

Κάθε οντότητα Station αποτελείται από συλλογή μοντέλων StationChannel

4.3.1.4. StationChannel

Η συγκεκριμένη κλάση διαχειρίζεται τα (raw) δεδομένα για κάθε κλιματικό “ κανάλι” όπως αυτά παρέχονται από τον εκάστοτε αισθητήρα του κάθε μετεωρολογικού σταθμού, π.χ. θερμομέτρο, ανεμοδείκτης κ.α.

Τα δεδομένα του κάθε καναλιού προσφέρονται για μεταεπεξεργασία από το μοντέλο Channel.

Το μοντέλο StationChannel ανήκει σε κάποια υλοποίηση του Station μοντέλου.

4.3.2 Πακέτο Services

Το συγκεκριμένο πακέτο αποτελείται από μονάδες κώδικα προσβάσιμες από όλες τις άλλες μονάδες του Server και προσφέρουν υπηρεσίες επεξεργασίας/εισαγωγής/εξαγωγής δεδομένων.

Οι παρούσες υλοποιήσεις είναι οι εξής:

4.3.2.1. ExportFile

Το συγκεκριμένο service αναλαμβάνει την εξαγωγή δεδομένων σε μορφή αρχείων διαφορετικών τύπων όπως csv, excel κ.α.

Η εξαγωγή των δεδομένων χρησιμοποιείται για δυο κύριους λόγους:

1. Εξαγωγή δεδομένων για μεταεπεξεργασία από άλλα λογισμικά
2. Εξαγωγή δεδομένων για backup

4.3.2.2. ParseFile

Το συγκεκριμένο service αναλαμβάνει την “ανάγνωση” και εισαγωγή δεδομένων (στην κλιματική βάση δεδομένων) από διαφορετικούς τύπους αρχείων όπως csv, excel κ.α.

Η εισαγωγή δεδομένων από αρχείο επιτρέπει την εύκολη γεφύρωση από άλλα λογισμικά που χρησιμοποιούνται για μεταεπεξεργασία

4.3.2.3. ProcessFile

Το συγκεκριμένο service αναλαμβάνει την μεταεπεξεργασία δεδομένων τύπου csv χρησιμοποιώντας εξωτερικές εφαρμογές.

Στην παρούσα υλοποίηση χρησιμοποιείται η εφαρμογή MATLAB για την παραγωγή επιπλέον κλιματικών δεικτών.

4.3.3 Πακέτο API

Το πακέτο API είναι ένα από τα πλέον νευραλγικά τμήματα του ευρύτερου συστήματος καθώς αποτελεί την υλοποίηση της πύλης WSDL η οποία προσφέρει την δυνατότητα ζεύξης με άλλα συστήματα.

Το συγκεκριμένο πακέτο αποτελείται από την υλοποίηση κλάσεων (οι οποίες επεκτείνουν την κλάση *Illuminate\Routing\Controller\BaseController* του Laravel) που αφορούν την διαχείριση κλήσεων (requests) από λογισμικά τύπου client.

Οι υπηρεσίες που προσφέρονται από το πακέτο API χρησιμοποιούνται κυρίως:

- Από τον διαδικτυακό τόπο (ionianweather.gr)
- Από την διαχειριστική διαδικτυακή εφαρμογή
- Από μετεωρολογικούς οργανισμούς π.χ. Ε.Μ.Υ.

Οι βασικοί controllers οι οποίοι υλοποιούν το WSDL είναι οι εξής:

4.3.3.1. WeatherDataController

Ο συγκεκριμένος controller παρέχει τις γενικές CRUD λειτουργίες όσον αφορά την διαχείριση των επεξεργασμένων κλιματικών δεδομένων.

Η πιο σημαντική λειτουργία που παρέχεται είναι η δυνατότητα αναζήτησης και ανάκλησης των επεξεργασμένων κλιματικών δεδομένων.

Η λειτουργία επιτρέπει τον προσδιορισμό του σταθμού, καναλιού και το χρονικό εύρος των δεδομένων που αναζητάμε. Για την ορθή διαχείριση των αποτελεσμάτων (καθότι ο όγκος των αποτελεσμάτων συχνά είναι υπερβολικά μεγάλος) η εφαρμογή που κάνει χρήση του

WeatherDataController μπορεί να προσδιορίσει το πλήθος των αποτελεσμάτων που θα επιστρέφονται ανα "page".

Page θεωρείται ένα τμήμα των αποτελεσμάτων *κάποιας αναζήτησης*. Π.χ. αν μια αναζήτηση παρέχει 1000 αποτελέσματα και το πλήθος των αποτελεσμάτων είναι 10/σελίδα η σελίδα 2 θα περιέχει τα αποτελέσματα απο 10-20.

Η κληση API που παρέχει τα δεδομένα (μέσο του WSDL) περιγράφεται ως:

Route: api/weather_data

Request Params:

- station (το όνομα του σταθμού απο τον οποίο θέλουμε δεδομένα)
- channel (το όνομα του καναλιού, π.χ. temperature, απο το οποίο θέλουμε δεδομένα)
- date_from (κάτω χρονικό όριο δεδομένων)
- date_to (άνω χρονικό όριο δεδομένων)
- limit (πλήθος δεδομένων ανα σελίδα)
- page (το νούμερο της σελίδας αποτελεσμάτων)

4.3.3.2. StationController

Ο συγκεκριμένος controller παρέχει όλες τις CRUD λειτουργίες όσον αφορά την διαχείριση πληροφοριών που αφορούν τον μετεωρολογικό σταθμό.

Οι λειτουργίες που παρέχονται απο τον controller επιτρέπουν την δήλωση/αντιστοίχιση των καναλιών που ενσωματώνει ο κάθε σταθμός

4.3.3.3. ChannelController

Ο συγκεκριμένος controller παρέχει όλες τις CRUD λειτουργίες όσον αφορά την διαχείριση πληροφοριών που αφορούν τα κανάλια πληροφοριών που παρέχει ο κάθε μετεωρολογικός σταθμός.

Το κάθε κανάλι παρέχει τα δεδομένα που συλλέγονται απο τον εκάστοτε αισθητήρα ο οποίος είναι ενσωματωμένος στο κάθε σταθμό.

4.3.3.4. RemoteSourceController

Ο συγκεκριμένος controller χρησιμοποιείται για την εκτέλεση κλήσεων (requests) σε εξωτερικές διαδικτυακές υπηρεσίες για την συλλογή επιπλέον κλιματικών δεδομένων.

Η εκτέλεση των κλήσεων εφαρμόζεται σε τακτά χρονικά διαστήματα και με παραμέτρους όπως αυτές έχουν οριστεί από τον διαχειριστή (βλέπε παράγραφο 6.1)

4.3.3.5. FileController

Ο συγκεκριμένος controller παρέχει όλες τις CRUD λειτουργίες όσον αφορά την διαχείριση αρχείων.

Επιπλέον υλοποιεί την μέθοδο ανοφόρτωσης και αποθήκευσης αρχείων στον server.

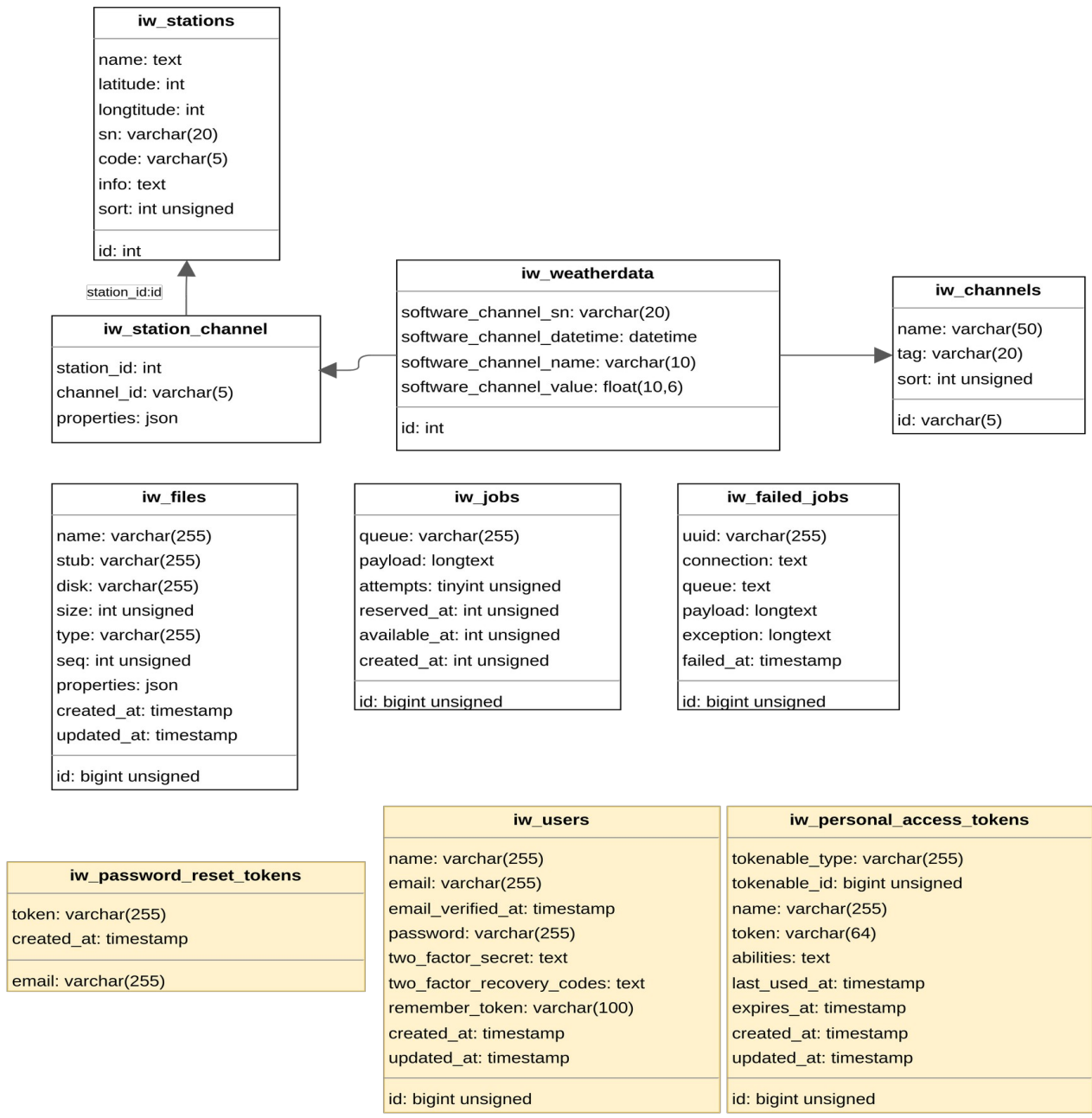
4.3.3.6. ModxController

Ο συγκεκριμένος controller δεν αποτελεί μέρος του API. Αποτελεί μια web πύλη η οποία εκτελεί δυο σκοπούς:

1. Προβολή και διαχείριση των κλήσεων που αφορούν τις διαδικτυακές σελίδες του διαδικτυακού τόπου ionianweather.gr
2. Αποτροπή κακόβουλων επιθέσεων. Η αποτροπή των επιθέσεων επιτυγχάνονται με αξιολόγηση των στοιχείων κλήσεως (request) των σελίδων.

Π.χ. σε περίπτωση που τα requests από συγκεκριμένη διεύθυνση εκτελούνται με υπερβολική πυκνότητα (πολύ γρήγορα) τότε οι κλίσεις αυτής της διεύθυνσης ακυρώνονται καθότι τέτοιου είδους συμπεριφορά συνιστά DoS attack

4.3.4 Μοντέλο δεδομένων



Διάγραμμα 3: Σχήμα βάσης δεδομένων ERD

Το βασικό μοντέλο δεδομένων το οποίο υλοποιήθηκε στα πλαίσια του συγκεκριμένου έργου παρουσιάζεται στο επόμενο διάγραμμα (Διάγραμμα 3)

Οι πίνακες που παρουσιάζονται στο μοντέλο δεδομένων είναι οι εξής:

4.3.4.1. iw_stations

Περιέχει πληροφορίες σχετικές με τον επιλεγόμενο σταθμό όπως:

- όνομα σταθμού
- γεωγραφικές συντεταγμένες
- σειριακό αριθμό
- κ.α.

4.3.4.2. iw_station_channel

Περιέχει πληροφορίες σχετικές με το κάθε κανάλι (αισθητήρα) του κάθε σταθμού.

Η παράμετρος `properties` αποτελεί ένα πεδίο τύπου JSON έτσι ώστε να μπορεί να διαχειριστή ποικίλες πληροφορίες ακόμα και για μελλοντικούς αισθητήρες που μπορούν να προστεθούν ή να αλλάξουν.

4.3.4.3. iw_weatherdata

Περιέχει τα επεξεργασμένα κλιματικά δεδομένα

4.3.4.4. iw_files

Περιέχει πληροφορίες σχετικές με τα αρχεία που έχουν εισαχθεί/εξαχθεί στο σύστημα.

4.3.4.5. iw_jobs

Περιέχει δεδομένα σχετικά με την παραμετροποίηση των προγραμματισμένων εργασιών π.χ. καθημερινό backup

4.3.4.6. iw_failed_jobs

Περιέχει δεδομένα σχετικά με προγραμματισμένες εργασίες οι οποίες απέτυχαν είτε να εκτελεστούν είτε να ολοκληρωθούν.

4.3.5 *Πακέτο Policies*

Το πακέτο Policies αποτελείται από κλάσεις οι οποίες υλοποιούν αλγορίθμους αξιολόγησης και επικύρωσης των εισερχομένων δεδομένων προς το σύστημα. Οι λειτουργίες του συγκεκριμένου πακέτου διασφαλίζουν τα εξής:

- Αποφυγή κακόβουλων επιθέσεων με στόχο την εισαγωγή ανεπιθύμητων δεδομένων στην βάση
- Διασφάλιση της πρόσβασης δεδομένων ανάλογα με την διαβάθμιση του χρήστη
- Εκτέλεση κανόνων αξιολόγησης/επικύρωσης κλιματικών δεδομένων (βλέπε κεφάλαιο 9) όπως αυτοί έχουν στοιχειοθετηθεί από τον διαχειριστή του συστήματος.

4.4. Διαχειριστικό σύστημα διαδικτυακού τόπου *ionianweather.gr*

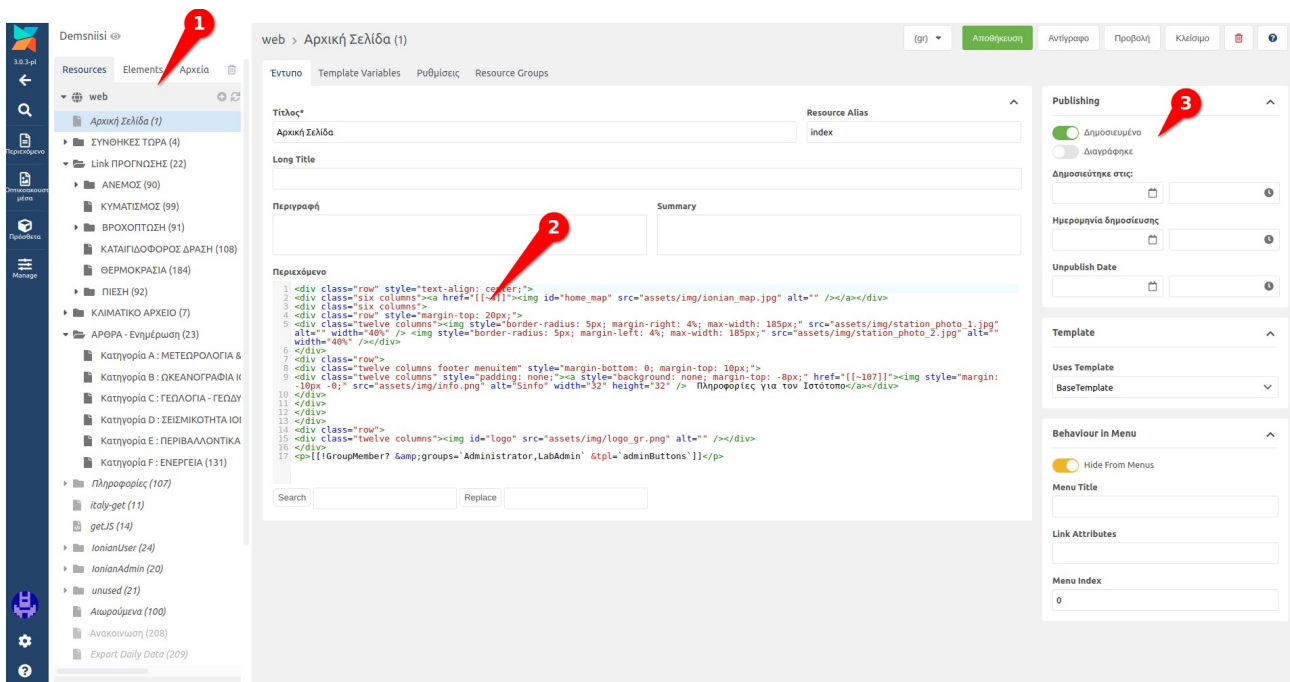
Για την διαχείριση του διαδικτυακού τόπου *ionianweather.gr* ενσωματώθηκε στο ευρύτερο σύστημα το Framework MODx 3.0.3.

Το MODx προσφέρει ένα προηγμένο περιβάλλον διαχείρισης διαδικτυακών σελίδων, χρηστών και δικαιωμάτων πρόσβασης.

4.4.1 Διαχείριση σελίδων

Η διαχείριση των σελίδων προσφέρεται μέσα απο ένα φιλικό περιβάλλον (UX).

Ο διαχειριστής μπορεί εύκολα να προσθέσει σελίδες και να ορίσει τα δικαιώματα πρόσβασης σε αυτές.

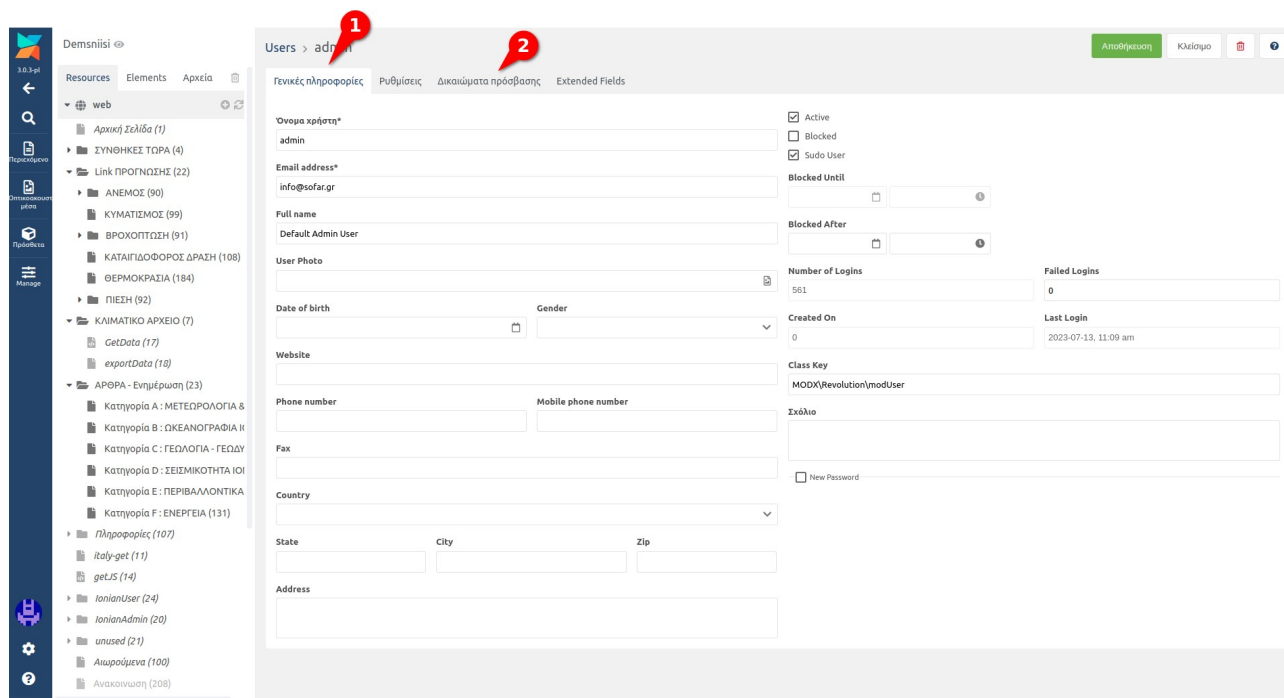


Εικόνα 4: Παράδειγμα διαχείρισης διαδικτυακής σελίδας

Οι βασικές δυνατότητες που έχει όσον αφορά το περιεχόμενο των σελίδων είναι οι εξής (όπως αριθμούνται στην Εικόνα 4):

1. Κατηγοριοποίηση των σελίδων (και άλλων δεδομένων) σε δένδρομή. Το MODx παρέχει εύκολη διαχείριση πολυγλωσσικότητας χρησιμοποιώντας **τομές** οι οποίες αποτελούν τις “κορυφές” της δένδρομής. Για την κάθε γλώσσα έχει δημιουργηθεί ένας ξεχωριστός τομέας. Στην συγκεκριμένη υλοποίηση έχουν ενεργοποιηθεί τρεις γλώσσες (αγγλικά/ελληνικά/ιταλικά)
2. Επεξεργασία HTML περιεχομένου διαμέσου εξειδικευμένου editor
3. Διαχείριση παραμέτρων σελίδας (π.χ. δημοσίευση/απο-δημοσίευση)

4.4.2 Διαχείριση Χρηστών

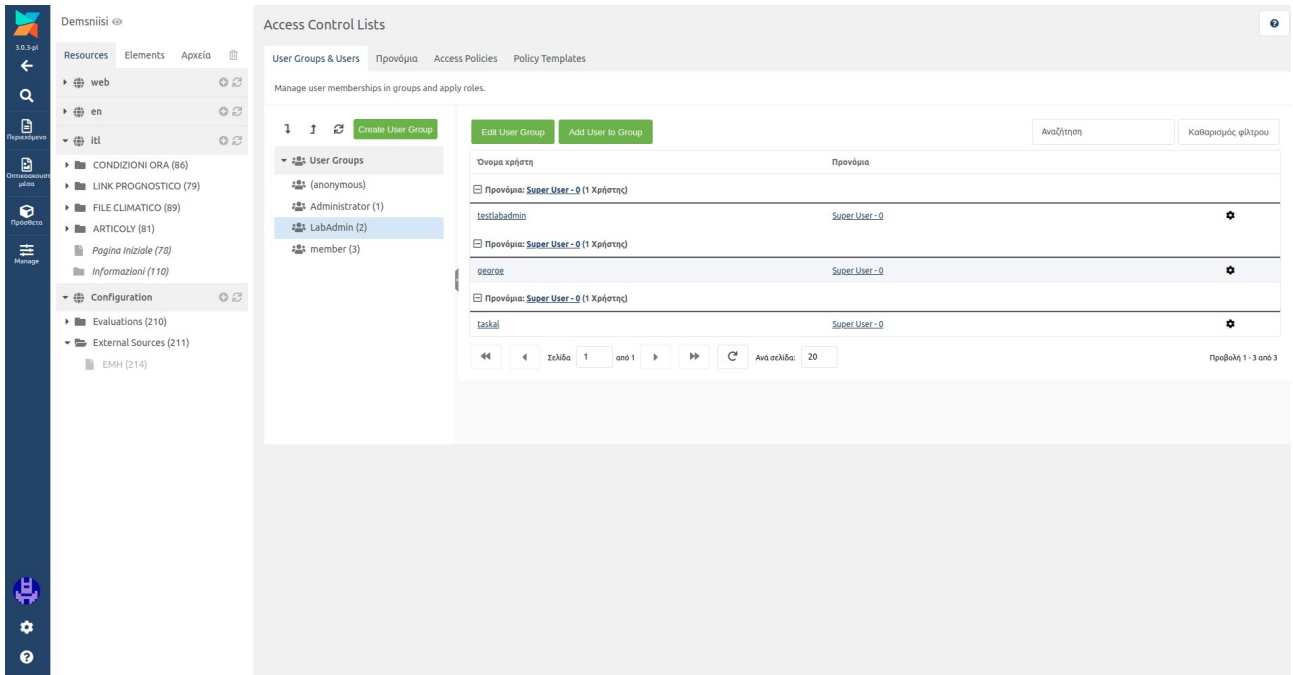


Εικόνα 5: Παράδειγμα καρτέλας χρηστών

Το MODx προσφέρει ένα προηγμένο σύστημα διαχείρισης χρηστών (Εικόνα 5) και δικαιωμάτων (Εικόνα 6)

Η διαχείριση δικαιωμάτων επιτρέπει τον ορισμό ομάδων χρήστη.

Η κάθε σελίδα ή μονάδα του MODx μπορεί να δεσμευθεί για συγκεκριμένες ομάδες χρηστών. Με αυτό τον τρόπο είναι εύκολη η διαχείριση της προσβασιμότητας της κάθε ομάδας χρηστών σε διαφορετικούς μονάδες και σελίδες.



Εικόνα 6: Παράδειγμα ομάδων χρήστη

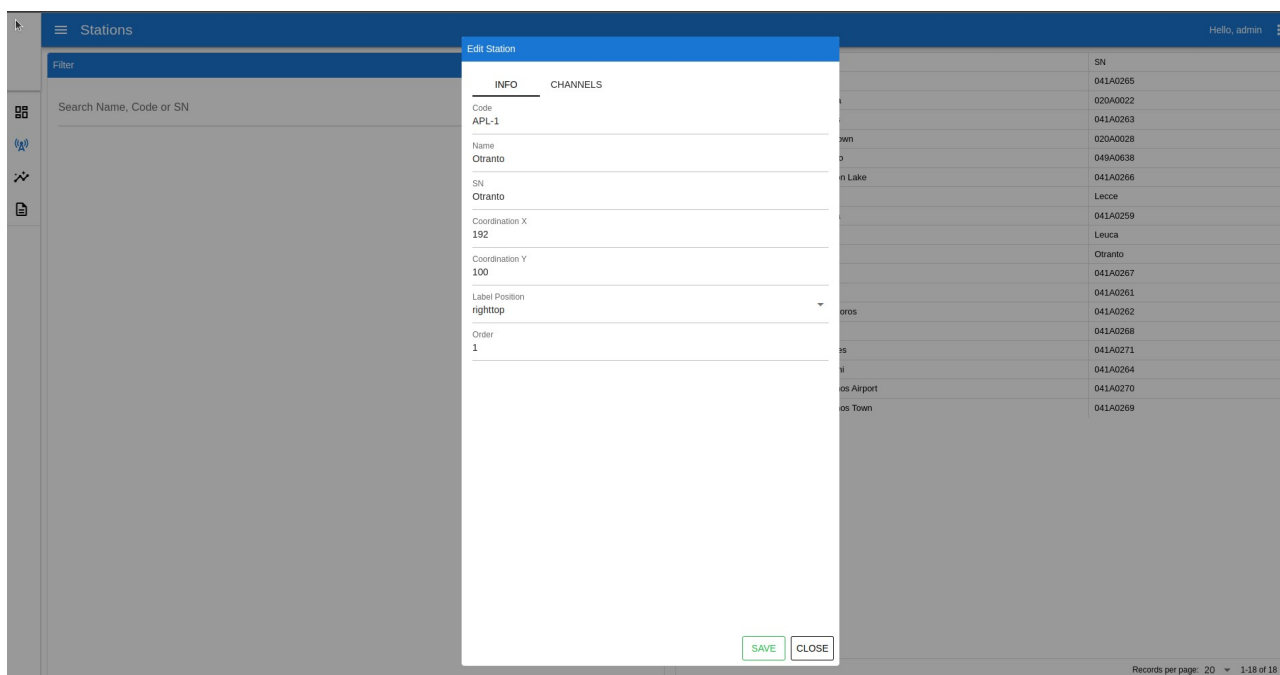
4.5. Διαχείριση μετεωρολογικών δεδομένων

Στα πλαίσια αυτού του έργου αναπτύχθηκε διαχειριστικό σύστημα για την παραμετροποίηση και διαχείριση κλιματικών πηγών δεδομένων.

Η πρόσβαση στο διαχειριστικό σύστημα προσφέρεται μέσω της διεύθυνσης ionianweather.safecontrol.gr/admin/

4.5.1 Διαχείριση μετεωρολογικών σταθμών

Το διαχειριστικό σύστημα επιτρέπει την δήλωση μετεωρολογικών σταθμών που προσφέρουν κλιματικά δεδομένα στο σύστημα.



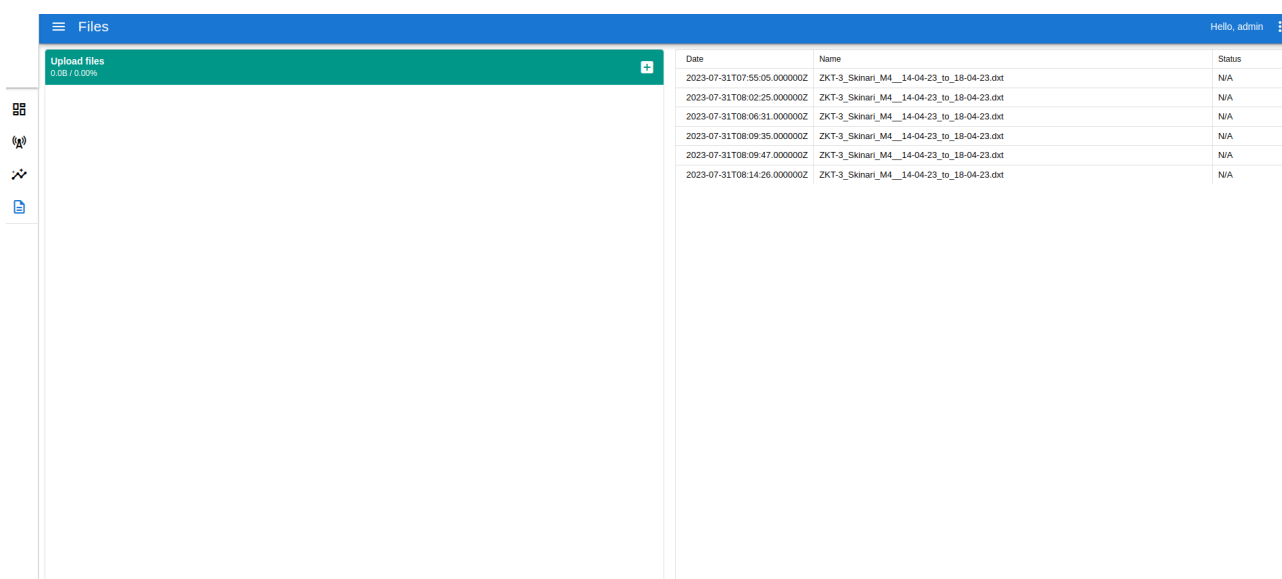
Εικόνα 7: Παράδειγμα διαχείρισης σταθμών

Για τον κάθε σταθμό παρέχεται η δυνατότητα αποτύπωσης των καναλιών που παρέχουν κλιματικά δεδομένα από τους υπάρχοντες αισθητήρες.

4.5.2 Διαχείριση κλιματικών αρχείων

Το διαχειριστικό σύστημα επιτρέπει την εισαγωγή κλιματικών δεδομένων απο κλιματικά αρχεία.

Ο διαχειριστής έχει την δυνατότητα ανω-φόρτωσης αρχείων τα οποία σε συνέχεια και μετα απο επεξεργασία, τα δεδομένα τους, ενσωματώνονται στην κλιματική βάση δεδομένων.



The screenshot shows a web interface for file management. At the top, there is a blue header with the word 'Files' and a user greeting 'Hello, admin'. Below the header, there is a green bar indicating the upload progress: 'Upload files 0.0B / 0.00%'. To the left of the main content area, there is a sidebar with several icons. The main content area is divided into two sections. The top section is a table with three columns: 'Date', 'Name', and 'Status'. The bottom section is a large empty white area, likely for displaying the contents of the uploaded files.

Date	Name	Status
2023-07-31T07:55:05.000000Z	ZKT-3_Skinari_M4_14-04-23_to_18-04-23.dxt	N/A
2023-07-31T08:02:25.000000Z	ZKT-3_Skinari_M4_14-04-23_to_18-04-23.dxt	N/A
2023-07-31T08:06:31.000000Z	ZKT-3_Skinari_M4_14-04-23_to_18-04-23.dxt	N/A
2023-07-31T08:09:35.000000Z	ZKT-3_Skinari_M4_14-04-23_to_18-04-23.dxt	N/A
2023-07-31T08:09:47.000000Z	ZKT-3_Skinari_M4_14-04-23_to_18-04-23.dxt	N/A
2023-07-31T08:14:26.000000Z	ZKT-3_Skinari_M4_14-04-23_to_18-04-23.dxt	N/A

Εικόνα 8: Παράδειγμα διαχείρισης κλιματικών αρχείων

5. Σύστημα ανασκόπησης-προβολής αρχειακών δεδομένων

Το συγκεκριμένο κεφάλαιο αφορά τις εργασίες και τα παραδοτέα που αντιστοιχούν στην τεχνική προδιαγραφή του πίνακα συμμόρφωσης νούμερο2

Στα πλαίσια αυτού του έργου αναπτύχθηκε web component το οποίο σε συνδυασμό μονάδας τύπου server-side controler παρέχει την δυνατότητα προβολής και ανασκόπησης των ανά-λεπτό καταγεγραμμένων τιμών των μετρούμενων παραμέτρων από το κλιματικό αρχείο του server.

Η προβολή των μετρήσεων παρουσιάζεται μέσα απο την διαδικτυακή σελίδα υπό μορφή βίντεο (animation). Ο χρήστης έχει την δυνατότητα να επιλέξει χρονική περίοδο, μετεωρολογικό σταθμό και κανάλι δεδομένων (μετρούμενων παραμέτρων π.χ. θερμοκρασία), στη συνέχεια η ιστοσελίδα θα του παρουσιάσει με animation την μεταβολή των τιμών για την συγκεκριμένη χρονική περίοδο.

Όλες οι ρυθμιζόμενες από τον χρήστη παράμετροι επιλέγονται εύκολα απο τον χρήστη μέσα απο εξειδικευμένο UX.

Ο κώδικας θα αναπτύχθηκε σε προγραμματιστικό περιβάλλον Linux / Docker Container και γλώσσα προγραμματισμού Node.js PHP και το web component σε Javascript VueJs.

5.1. Προδιαγραφές συστήματος συνεχούς προβολής μετεωρολογικών αρχειακών δεδομένων

Το εν λόγω σύστημα αποτελεί μονάδα προβολής δεδομένων (visualisation) προς τον χρήστη. Δεδομένου ότι πρέπει να παράγει την απεικόνιση δεδομένων σε πραγματικό χρόνο και με συνεχή ροή το σύστημα υλοποιήθηκε ως client-side web component.

Η παρούσα υλοποίηση καλύπτει τις εξής προδιαγραφές:

- Παραμετροποίηση της συλλογής των δεδομένων που θα απεικονίσει. Στην παρούσα υλοποίηση ο χρήστης έχει την δυνατότητα να επιλέξει δεδομένα εφαρμόζοντας τα εξής φίλτρα αναζήτησης:
 - Χρονική περίοδο
 - Μετεωρολογικό Σταθμό
 - Κανάλι δεδομένων (π.χ. θερμοκρασία)
- Παραμετροποίηση της συχνότητας προβολής δεδομένων σε αντιστοίχιση με τον πραγματικό χρόνο. Η συγκεκριμένη παραμετροποίηση επιτρέπει να ρυθμιστεί το ισοδύναμο του πραγματικού χρόνου καταγραφής των δεδομένων σε διαφορετική μονάδα χρόνου προβολής δεδομένων. Π.χ ο χρήστης έχει την δυνατότητα ρύθμισης εναλλαγής δεδομένων (κατα την διαδικασία προβολής) του 1νος λεπτού ανα ένα δευτερόλεπτο.

Στην υλοποίηση του συγκεκριμένου παραδοτέου ο χρήστης έχει την δυνατότητα να επιλέξει την προβολή του κλιματικού αρχείου για τα εξής χρονικά διαστήματα:

1. Τελευταίας μιας ώρας
2. Τελευταίων 6 ωρών

- Το web component έχει αναπτυχθεί ως αυτόνομη μονάδα έτσι ώστε να είναι δυνατή η εύκολη αναβάθμιση και μεταφορά του σε κάποια νέα μετεωρολογική πλατφόρμα.

6. Υπολογισμός κλιματικών δεικτών – κλιματικών ανωμαλιών

Το συγκεκριμένο κεφάλαιο αφορά τις εργασίες και τα παραδοτέα που αντιστοιχούν στην τεχνική προδιαγραφή του πίνακα συμμόρφωσης νούμερο3

Στα πλαίσια αυτού του έργου αναπτύχθηκε μονάδα λογισμικού η οποία αναλαμβάνει αυτόματη ενσωμάτωσης κλιματικών δεικτών παραγόμενων σε ημερήσια βάση από λογισμικό MatLab (που παρέχονται έτοιμα), για κάθε σταθμό του δικτύου και για κάθε μετρούμενη παράμετρο.

Ο κώδικας της μονάδας λογισμικού αναπτύχθηκε σε προγραμματιστικό περιβάλλον Linux/Docker Container και γλώσσα προγραμματισμού Node.js.

Η μονάδα που αναπτύχθηκε προσφέρει τα εξής:

- Αυτόματη αρχειοθέτηση στην Κλιματική Βάση του συστήματος σε ημερήσια αρχεία csv.
- Προσφέρει επιλογή στο κεντρικό menu επιλογών της ιστοσελίδας του δικτύου ως “Κλιματική Βάση” η οποία παρέχει πρόσβαση στον χρήστη στα συγκεκριμένα δεδομένα. Ο Διαχειριστής του συστήματος έχει εύκολο έλεγχο των κλιματικών παραμέτρων που θα προβάλλονται μέσα απο φιλικό UX διαχειριστικό περιβάλλον.

6.1. Προδιαγραφές μονάδος

Η συγκεκριμένη μονάδα αναλαμβάνει το ρόλο μεταεπεξεργασίας δεδομένων με στόχο την παραγωγή νέων κλιματικών παραμέτρων. Οι προδιαγραφές που υλοποιήθηκαν είναι οι εξής:

- Παραμετροποίηση των καναλιών/μετεωρολογικό σταθμό τα οποία θα αποθηκεύονται σε format csv ικανό για μεταεπεξεργασία απο MatLab. Η παραμετροποίηση θα επιτρέπει των καθορισμό των εξής παραμέτρων:
 - Μονοπάτι στην αποθηκευτική μονάδα (σκληρό δίσκο) που θα αποθηκεύονται
 - Ημερήσια ώρα που θα εξάγονται τα δεδομένα
- Παραμετροποίηση ρουτίνας (εντολή κονσόλας) που θα καλείτε στο MatLab για την μεταεπεξεργασία των δεδομένων
- Ημερήσια ώρα που θα εισάγονται τα μετα-επεξεργασμένα δεδομένα στην κλιματική βάση των δεδομένων.

7. Αυτόματη ενσωμάτωση δεδομένων από εξωτερικές πηγές

Το συγκεκριμένο κεφάλαιο αφορά τις εργασίες και τα παραδοτέα που αντιστοιχούν στην τεχνική προδιαγραφή του πίνακα συμμόρφωσης νούμερο⁴

Στα πλαίσια αυτού του έργου αναπτύχθηκε μονάδα λογισμικού η οποία αναλαμβάνει την αυτόματη ενσωμάτωση προϊόντων παρατηρησιακών πλεγματικών δεδομένων πραγματικού χρόνου (όπως ηλεκτρική ατμοσφαιρική δραστηριότητα ή νέφωση) από τρίτες ελεύθερες διαδικτυακές πηγές, ανάλογα με την διαθεσιμότητά τους. Τα πλεγματικά δεδομένα απεικονίζονται στην περιοχή κάλυψης του χάρτη που διατίθεται στην ιστοσελίδα του Επιχειρησιακού Δικτύου Μετεωρολογικών Σταθμών Ιονίου (όπως δημοσιεύτηκε στον ιστότοπο ionianweather.gr).

Ο Διαχειριστής του συστήματος μπορεί εύκολα τις τριτογενής πηγές μέσα απο φιλικό UX διαχειριστικό περιβάλλον.

Ο κώδικας της μονάδας λογισμικού αναπτύχθηκε σε προγραμματιστικό περιβάλλον Linux/Docker Container και γλώσσα προγραμματισμού Node.js.

7.1. Προδιαγραφές μονάδος

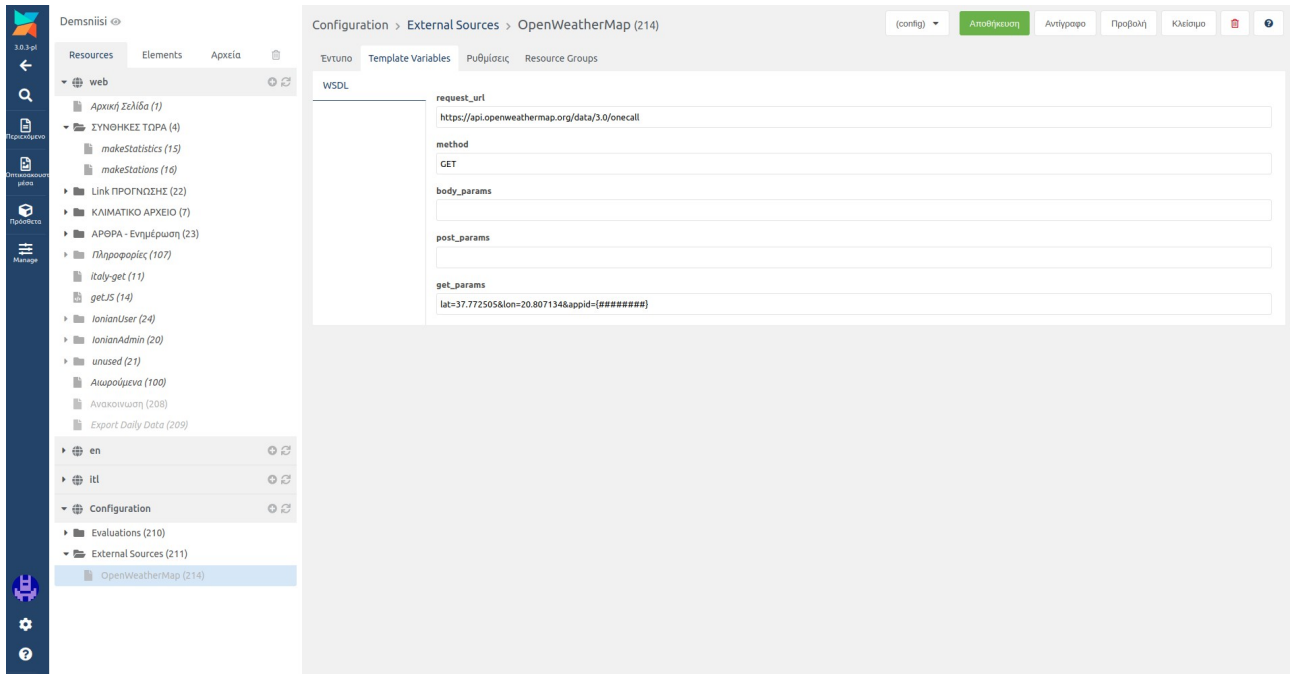
Η συγκεκριμένη μονάδα αναλαμβάνει το συλλογής απο εξωτερικές πηγές κλιματικών δεδομένων οι οποίες παρέχουν τα δεδομένα διαμέσου διαδικτυακών υπηρεσιών τύπου RESTful Services – WSDL(www.w3.org/TR/2004/NOTE-ws-arch-20040211/#relwwwrest).

Η υλοποίηση της ενσωμάτωσης κλιματικών δεδομένων απο εξωτερικές πηγές έχει υλοποιηθεί με τον controller *RemoteSourceController* (παράγραφος 4.3.3.4).

Οι προδιαγραφές που υλοποιήθηκαν είναι οι εξής:

- Δυνατότητα παραμετροποίησης πολλαπλών πηγών WSDL
- Παραμετροποίηση όλων των αναγκαίων χαρακτηριστικών κλίσεως (request) του WSDL API που απαιτούνται για την διασύνδεση και συλλογή δεδομένων απο την κάθε εξωτερική πηγή. Τα χαρακτηριστικά τα οποία είναι παραμετροποιήσιμα, είναι τα εξής:
 - Request Url
 - Method (POST/GET)
 - Body Params
 - Post Params
 - Get Params
- Παραμετροποίηση χρονικής επανάληψης (π.χ. κάθε 6 ώρες) κλήσεως προς κάθε εξωτερική πηγή έτσι ώστε η κλιματική βάση αρχείο του συστήματος να βρίσκεται πάντα ενημερωμένη με τις νέες τιμές κλιματικών δεδομένων που παρέχονται απο άλλες πηγές WSDL.

- Παραμετροποίηση της ενσωμάτωσης των συλλεγέντων δεδομένων στο κλιματικό αρχείο του συστήματος. Η παραμετροποίηση αφορά την αντιστοιχία των συλλεγέντων δεδομένων με κάποιο κλιματικό κανάλι δεδομένων το οποίο θα προβάλλεται απο το σύστημα.



Εικόνα 9: Παράδειγμα παραμετροποίησης στην υπηρεσία WSDL OpenWeatherMap

8. Βελτίωση προβολής “συνθηκών τώρα”

Το συγκεκριμένο κεφάλαιο αφορά τις εργασίες και τα παραδοτέα που αντιστοιχούν στην τεχνική προδιαγραφή του πίνακα συμμόρφωσης νοούμερο5

Στα πλαίσια αυτού του έργου αναπτύχθηκε web component το οποίο επιτρέπει την εμφάνιση τιμών των μετρούμενων παραμέτρων σε δυο διακριτές χρονικές βάσεις:

- ενός (1) λεπτού
- δέκα (10) λεπτών

Η επιλογή παρέχεται προς του χρήστες στο menu “Συνθήκες Τώρα” της ιστοσελίδας του δικτύου.

Ο κώδικας της μονάδας λογισμικού αναπτύχθηκε σε προγραμματιστικό περιβάλλον Linux/Docker Container και γλώσσα προγραμματισμού Node.js.

8.1. Υλοποίηση μονάδας διαχείρισης συνθηκών τώρα

Η συγκεκριμένη μονάδα αποτελείται από δυο τμήματα:

1. Τμήμα client το οποίο αποτελείται από UX component επιλογής χρονικού εύρους. Το component έχει τοποθετηθεί στην διαδικτυακή διεύθυνση ionianweather.gr/stations
2. Τμήμα server το οποίο αποτελεί ρουτίνα επεξεργασίας στο WeatherDataController (παράγραφο 4.3.3.1). Η ρουτίνα αναλαμβάνει (βάση της επιλογής του χρήστη) να παρέχει τον μέσο όρο της εκάστοτε κλιματικής μεταβλητής για το επιλεγμένο χρονικό εύρος.

Η λογική που ακολουθήθηκε είναι η παραγωγή τιμών από τον controller (ανά σταθμό/κανάλι) βάση του μέσου όρου του χρονικού εύρους που έχει επιλέξει ο χρήστης.

Στο συγκεκριμένο έργο προσφέροντε στον χρήστη 2 επιλογές μέσω ωρών:

1. Μέσος όρος λεπτού
2. Μέσος όρος δεκαλέπτου

Log In

Δίκτυο Μετεωρολογικών Σταθμών ΙΟΝΙΟΥ

ΣΥΝΘΗΚΕΣ ΤΩΡΑ
Link ΠΡΟΓΝΩΣΗΣ
ΚΛΙΜΑΤΙΚΟ ΑΡΧΕΙΟ
ΑΡΘΡΑ - Ενημέρωση

Ιόνιο Πανεπιστήμιο
 Σχολή Περιβάλλοντος
 Τμήμα Περιβάλλοντος
ΕΡΓΑΣΤΗΡΙΟ
 Φυσικής Περιβάλλοντος, Ενέργειας και
 Περιβαλλοντικής Βιολογίας

Τελευταία
 Ενημέρωση **13:58**

Μέση τιμή ανά:

λεπτό δεκάλεπτο

ΑΝΕΜΟΣ

m/s Km/h Knt Beaufort

ΑΝΕΜΟΣ (ριπή)

m/s Km/h Knt Beaufort

ΒΡΟΧΟΠΤΩΣΗ

Ανα Λεπτό Ωριαία
 Ημερήσια Μηνιαία

ΘΕΡΜΟΚΡΑΣΙΑ

Celsius Fahrenheit
 Min/Max

ΥΓΡΑΣΙΑ

Σχετική Απόλυτη

ΠΙΕΣΗ

ΙΣΧΥΣ ΗΛΙΑΚΗΣ ΑΚΤΙΝΟΒΟΛΙΑΣ

ΥΠΕΡΙΩΔΗΣ ΑΚΤΙΝΟΒΟΛΙΑ

UV-A UV-B

ΣΥΓΚΕΝΤΡΩΣΗ Aerosols

PM1 PM2.5 PM4
 PM10 Total

ΟΛΑ

1h ▶
6h ▶
■

Πληροφορίες για τον Ιστότοπο

Created by SoFar.Gr

Εικόνα 10: Σελίδα επισκόπησης κλιματικών συνθηκών σε πραγματικό χρόνο

9. Μονάδα ανίχνευσης δυσλειτουργιών

Το συγκεκριμένο κεφάλαιο αφορά τις εργασίες και τα παραδοτέα που αντιστοιχούν στην τεχνική προδιαγραφή του πίνακα συμμόρφωσης νούμερο6

Στα πλαίσια αυτού του έργου αναπτύχθηκε μονάδα λογισμικού η οποία αναλαμβάνει την ανίχνευση δυσλειτουργιών στο Επιχειρησιακού Δικτύου Μετεωρολογικών Σταθμών Ιονίου σε πραγματικό χρόνο και την άμεση ενημέρωση του διαχειριστή όταν παρουσιάζεται κάποια δυσλειτουργία. Οι προβλεπόμενες δυσλειτουργίες είναι:

- Ανίχνευση απώλειας διαδικτυακής διάθεσης της ιστοσελίδας (ionianweather.gr).
- Ανίχνευση διακοπής ροής ή μη διαθεσιμότητας μετρούμενων τιμών από οποιονδήποτε αισθητήρα σταθμού του δικτύου προς τον server.
- Ανίχνευση εσφαλμένων τιμών για κάθε μια από τις μετρούμενες παραμέτρους σε κάθε σταθμό του δικτύου.

Η άμεση ενημέρωση του διαχειριστή επιτυγχάνεται με την αποστολή μηνύματος τύπου SMS.

Ο κώδικας της μονάδας λογισμικού αναπτύχθηκε σε προγραμματιστικό περιβάλλον Linux/Docker Container και γλώσσα προγραμματισμού PHP.

9.1. Προδιαγραφές μονάδος

Η συγκεκριμένη μονάδα αναλαμβάνει την ενημέρωση του διαχειριστή για τυχόν δυσλειτουργίες του συστήματος. Οι προδιαγραφές που υλοποιήθηκαν είναι οι εξής:

- Υλοποίηση υπηρεσίας παροχής καναλιού μετάδοσης μηνυμάτων διαμέσου διαδικτυακών υπηρεσιών μηνυμάτων. Η υπηρεσία μπορεί να χρησιμοποιηθεί από διαφορετικές μονάδες του συστήματος για να στείλει τα ανάλογα μηνύματα.

Στην παρούσα υλοποίηση χρησιμοποιήθηκε η υπηρεσία **Telegram** (telegram.org). Η συγκεκριμένη υπηρεσία επιλέχθηκε (μετά απο έρευνα που εκπονήθηκε) διότι:

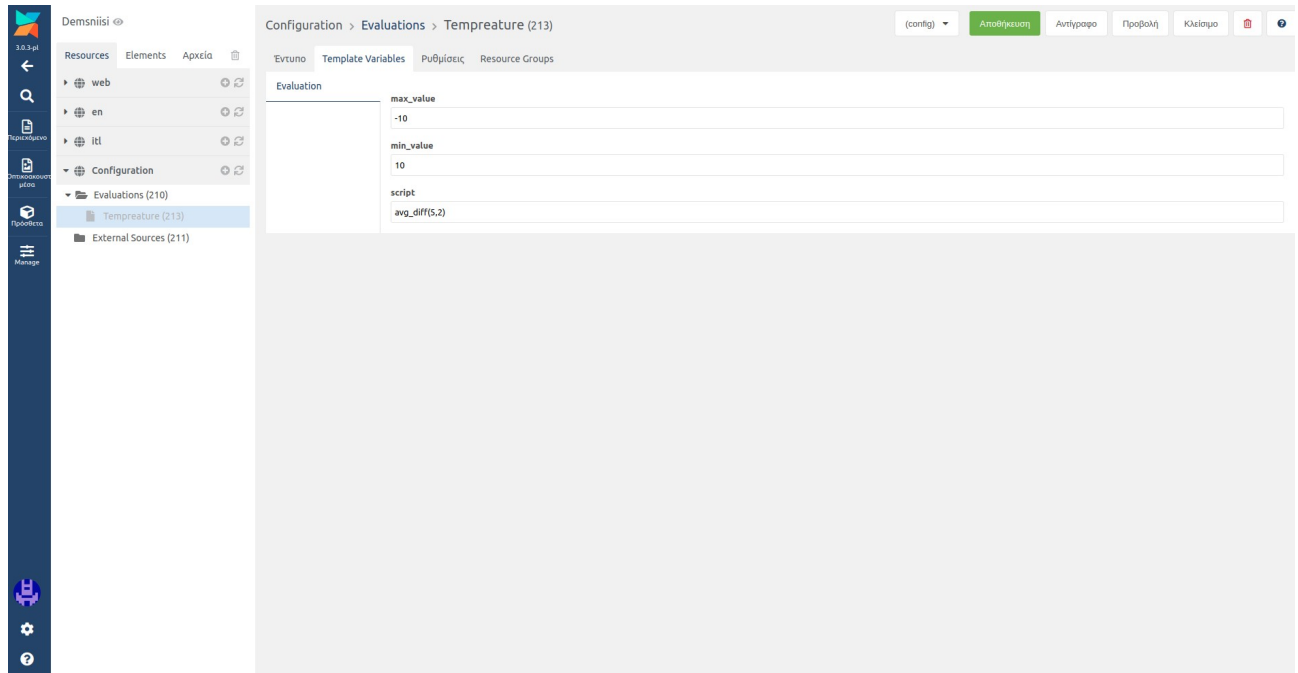
- Προσφέρει τον πλέον αξιόπιστο και άμεσο μηχανισμό αποστολής μηνυμάτων.
- Παρέχεται εφαρμογή για όλα τα λειτουργικά συστήματα
- Έλεγχος καλής λειτουργίας του διαδικτυακού τόπου με `node.js`. Σε περίπτωση ανίχνευσης απώλειας διαδικτυακής διάθεσης της ιστοσελίδας (ionianweather.gr) ενεργοποιείται και αποστέλλεται ειδοποίηση στον διαχειριστή.
- Επικύρωση/Αξιολόγηση ορθότητας τιμών ανα κλιματική μεταβλητή (κανάλι). Σε περίπτωση στην οποία ο κανόνας αξιολόγησης δεν επικύρωση την ορθότητα των τιμών η καταγραφή της τιμής ακυρώνεται και ο διαχειριστής ενημερώνεται με ανάλογο μήνυμα το οποίο περιέχει:
 - Το σταθμό

- Το κανάλι
- Την εσφαλμένη τιμή
- Τον κανόνα αξιολόγησης που εντόπισε την εσφαλμένη τιμή
- Την χρονική στιγμή που προέκυψε

Η υλοποίηση της λογικής αξιολόγησης πραγματοποιείται μεσο της κλάσης *WeatherDataPolicy* (παράγραφο 4.3.5).

Στην παρούσα υλοποίηση οι ανίχνευση ορθότητας παραμέτρων γίνεται με τις εξής μεταβλητές:

- *max_value*: Μέγιστη προβλεπόμενη τιμή
- *min_value*: Ελάχιστη προβλεπόμενη τιμή
- *script*: Η συγκεκριμένη μεταβλητή έχει την δυνατότητα να καλέσει ρουτίνες αξιολόγησης βάση στατιστικών. Στην παρούσα υλοποίηση έχει ενσωματωθεί η ρουτίνα:
 - *avg_diff(time_range,error_limit)*: Η ρουτίνα προβλέπει 2 παραμέτρους εισαγωγής:
 - *time_range*: Το χρονικό διάστημα απο το παρόν προς το παρελθόν κατα το οποίο θα υπολογιστεί ο μέσος όρος τιμών
 - *error_limit*: Τα όρια σφάλματος τα οποία θα θεωρούνται παραδεκτές οι τιμές



Εικόνα 11: Παράδειγμα κανόνα αξιολόγησης

Η διαχείριση των κανόνων αξιολόγησης παρέχεται μέσα από το διαχειριστικό τμήμα **Configurations**. Η πρόσβαση επιτυγχάνεται μέσα από την διεύθυνση <https://ionianweather.gr/manager/>.

Στην εικόνα (Εικόνα 11) βλέπουμε το παράδειγμα κωδικοποίησης κανόνα ορθότητας για την θερμοκρασία.

10. Υπηρεσίες υποστήριξης παραδοτέων

Στα πλαίσια αυτού του έργου ο ανάδοχος θα παρέχει υπηρεσίες υποστήριξης

10.1. Άμεση τεχνική υποστήριξη σε περίπτωση δυσλειτουργίας ή βλάβης του λογισμικού

Το συγκεκριμένο κεφάλαιο αφορά τις εργασίες και τα παραδοτέα που αντιστοιχούν στην τεχνική προδιαγραφή του πίνακα συμμόρφωσης νούμερο⁷

Σε περίπτωση δυσλειτουργίας ή βλάβης του λογισμικού ή του κεντρικού εξυπηρετητή (server) του Επιχειρησιακού Δικτύου Μετεωρολογικών Σταθμών Ιονίου, ο ανάδοχος (κατόπιν τηλεφωνικής ενημέρωσης από τον Διαχειριστή του δικτύου) αναλαμβάνει να ανταπεξέλθει στα εξής:

- Ενέργειες άμεσης διάγνωσης της δυσλειτουργίας ή βλάβης λογισμικού ή server μέσω τηλεφωνικής επικοινωνίας με τον Διαχειριστή του συστήματος και αρχικά μέσω απομακρυσμένης εξέτασης της κατάστασης λειτουργίας όλων των μερών του δικτύου.
- Απόκριση μετά την ανακοίνωση βλάβης ή δυσλειτουργίας από τον Διαχειριστή του συστήματος, σε χρονικό διάστημα μικρότερο της ½ ώρας κατά τις εργάσιμες ημέρες ή ανεξαρτήτως ημέρας σε περιπτώσεις εκδήλωσης έντονων καιρικών φαινομένων (ανάπτυξης Μεσογειακών τυφώνων στην περιοχή Ιονίου, εμφάνισης ανεμοστροβίλων, εκδήλωσης

πλημμυρικών βροχοπτώσεων, θυελλωδών ανέμων, συνθηκών καύσωνα) κατά τις ώρες 08:00 – 20:00.

- Απόκριση μετά την ανακοίνωση βλάβης ή δυσλειτουργίας από τον Διαχειριστή του συστήματος σε μικρότερο των 6 ωρών σε κάθε άλλη περίπτωση και ημέρα.

10.2. Επιτόπια τεχνική υποστήριξη σε περίπτωση δυσλειτουργίας ή βλάβης του λογισμικού

Το συγκεκριμένο κεφάλαιο αφορά τις εργασίες και τα παραδοτέα που αντιστοιχούν στην τεχνική προδιαγραφή του πίνακα συμμόρφωσης νούμερο8

Σε περίπτωση όπου η δυσλειτουργία ή βλάβης του λογισμικού ή του κεντρικού εξυπηρετητή (server) δεν μπορεί να αντιμετωπιστεί απομακρυσμένα η εταιρεία μας αναλαμβάνει την επιτόπια αντιμετώπιση της κατάστασης στον φυσικό χώρο του μηχανήματος φιλοξενίας των λογισμικών μονάδων το οποίο βρίσκεται στο Τμήμα Περιβάλλοντος του Ιονίου Πανεπιστημίου (οδός Μινώτου Γιαννοπούλου 26, περιοχή Παναγούλα, Ζακύνθου)

Η όροι επιτόπιας τεχνικής υποστήριξη θα καλύπτουν τα εξής:

- Παροχή εξειδικευμένου προσωπικού για την αντιμετώπιση του προβλήματος.
- Η ανταπόκριση για αντιμετώπισης βλάβης, θα παρέχεται σε διάστημα μικρότερο της μίας (1) ώρας κατά τις εργάσιμες ημέρες ή σε περιπτώσεις εκδήλωσης έντονων καιρικών φαινομένων (ανάπτυξης Μεσογειακών τυφώνων στην περιοχή Ιονίου, εμφάνισης ανεμοστροβίλων, εκδήλωσης πλημμυρικών βροχοπτώσεων, θυελλωδών ανέμων, συνθηκών καύσωνα) μετά την ανακοίνωση βλάβης ή δυσλειτουργίας από τον Διαχειριστή του συστήματος, στις ώρες 08:00 – 18:00.

- Η ανταπόκριση για αντιμετώπισης βλάβης, θα παρέχεται σε χρόνο μικρότερο των οκτώ (8) ωρών σε κάθε άλλη περίπτωση μετά την ανακοίνωση βλάβης ή δυσλειτουργίας από τον Διαχειριστή του συστήματος.
- Σε περίπτωση μη επιλύσιμης βλάβης του κύριου υπολογιστή έχει προβλεφθεί και προετοιμαστεί εφεδρικός υπολογιστής (παρεχόμενος από τον Διαχειριστή του συστήματος) ο οποίος μπορεί να αναλάβει τον ρόλο κεντρικού εξυπηρετητή ο πολύ δώδεκα (12) ώρες μετά την σχετική ειδοποίηση του αναδόχου από τον Διαχειριστή του συστήματος.

10.3. Εκπαίδευση προσωπικού

Το συγκεκριμένο κεφάλαιο αφορά τις εργασίες και τα παραδοτέα που αντιστοιχούν στην τεχνική προδιαγραφή του πίνακα συμμόρφωσης νούμερο9

Ο ανάδοχος θα παρέχει εκπαίδευση ενός χρήστη του συστήματος οριζόμενου από τον Διαχειριστή του Επιχειρησιακού Δικτύου Μετεωρολογικών Σταθμών Ιονίου, κατ' ελάχιστο ανερχόμενη σε δέκα (20) ώρες, συμπεριλαμβανομένης υπηρεσίας συμβουλευτικής υποστήριξης (help desk) κατά τις εργάσιμες ημέρες και ώρες,

τουλάχιστον για όσο διάστημα διαρκεί η ολοκλήρωση του έργου.

Ο προγραμματισμός της περιόδου εκπαίδευσης θα συμφωνηθεί μέσω email μεταξύ του αναδόχου και του προσωπικού του Επιχειρησιακού Δικτύου Μετεωρολογικών Σταθμών Ιονίου.

11. Οδηγίες εγκατάστασης

Για να εγκατασταθούν οι μονάδες που αναπτύχθηκαν στα πλαίσια αυτού του έργου, το μηχάνημα στο οποίο θα εγκατασταθεί πρέπει να έχει εγκατασταθεί αρχικά το λογισμικό ανοικτού κώδικα **docker** (<https://github.com/docker>).

Η εγκατάσταση των μονάδων ακολουθεί τα εξής βήματα:

1. Αποσυμπίεση του αρχείου zip σε επιθυμητή τοποθεσία του δίσκου η οποία πρέπει να έχει δικαιώματα εγγραφής/ανάγνωσης/εκτέλεσης για τον χρήστη

2. Δημιουργία του docker container εκτελώντας την εντολή:

```
docker-compose build mysql phpmyadmin nginx php-fpm workspace
```

3. Ενεργοποίηση του docker εκτελώντας την εντολή:

```
docker-compose up -d mysql phpmyadmin nginx php-fpm workspace
```

4. Είσοδος στο docker container εκτελώντας την εντολή:

```
docker-compose exec --user laradock workspace bash
```

5. Αρχικοποίηση των μονάδων εκτελώντας τις εντολές:

```
docker-compose install
```

```
npm install
```

6. Εγκατάσταση τις αρχικοποιημένης βάσης δεδομένων εκτελώντας την εντολή:

```
mysql -u user_name database_name < laertis.sql
```

7. Τοποθέτηση των στοιχείων εξουσιοδότησης της βάσης δεδομένων στο αρχείο .env