



**ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ
ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ**

Διπλωματική Εργασία

Εφαρμογή για την διαχείριση ενός Ασύρματου Δικτύου Αισθητήρων, βασισμένο στο Arduino για Online ύδρευση

A Web of Things Arduino Wireless, Sensor-Based Automatic Plant
Watering System Application

Σμυρναίος Δημήτριος

**ΕΞΕΤΑΣΤΙΚΗ ΕΠΙΤΡΟΠΗ
Καθηγητής Καλαϊτζάκης Κωνσταντίνος (Επιβλέπων)
Αναπλ. Καθηγητής Κουτρούλης Ευτύχης
Αναπλ. Καθηγητής Δεληγιαννάκης Αντώνιος**

Χανιά Ιανουάριος 2020

Πρόλογος

Η παρούσα διπλωματική εργασία αποτελεί τη δεύτερη σε σειρά και αποτελεί τη συνέχεια της εργασίας με τίτλο «Σύστημα αυτόματου ποτίσματος βασισμένο σε ένα ασύρματο δίκτυο αισθητήρων» του Γεώργιου Βρατσάνου. Δεν θα μπορούσε να πραγματοποιηθεί χωρίς τη συνεργασία και τη συνεχή επικοινωνία μας. Επίσης σημαντική ως προς την εκπόνηση της εργασίας ήταν η καθοδήγηση του επιβλέποντα καθηγητή Κωνσταντίνου Καλαϊτζάκη.

Περίληψη

Αντικείμενο της παρούσας διπλωματικής εργασίας αποτελεί η σχεδίαση και υλοποίηση ενός συστήματος αυτόματης άρδευσης αγροτεμαχίου. Το σύστημα αποτελείται από ένα ασύρματο δίκτυο αισθητήρων και ένα ενσωματωμένο σύστημα (embedded system) για την καταγραφή των φυσικών φαινομένων (θερμοκρασιών, υγρασίας ατμόσφαιρας και εδάφους), ένα σύστημα μεταφόρτωσης των δεδομένων μέσω ασύρματου δικτύου 4G σε ένα web server χρησιμοποιώντας web services και την επακόλουθη αποθήκευση τους σε μία σχεσιακή βάση δεδομένων (postgres). Στη συνέχεια, περιλαμβάνει την κατασκευή RESTful webservices για τη διαχείριση των δεδομένων, καθώς και την τροποποίηση των παραμέτρων μέσω των οποίων ο αλγόριθμος επιλέγει αν θα ξεκινήσει ή όχι την άρδευση του αγροτεμαχίου. Η διαχείριση των δεδομένων των web services γίνεται μέσω μιας φιλικής προς το χρήστη διεπαφής έτσι ώστε να είναι προσιτό σε όσο το δυνατόν μεγαλύτερο κοινό. Ακόμη έχει δημιουργηθεί μια εφαρμογή (Android app) και τα αντίστοιχα restful webservices για το Android app.

Στα επόμενα κεφάλαια γίνεται ανάλυση του κάθε υποσυστήματος. Η σχεσιακή βάση δεδομένων σχεδιάστηκε ακολουθώντας τα πρότυπα για τη καταγραφή «μετρήσεων» αισθητήρων όπως αυτά έχουν οριστεί από την Open Geospatial Consortium (OGC). Για την κατασκευή των διεπαφών της διαδικτυακής εφαρμογής (web application), χρησιμοποιήθηκε η γλώσσα προγραμματισμού Java και το spring boot framework, σε συνδυασμό με το Javascript AngularJS framework καθώς και άλλων ανοιχτών βιβλιοθηκών. Τα webservices για την Android εφαρμογή δημιουργήθηκαν χρησιμοποιώντας dotnetcore framework και τη γλώσσα προγραμματισμού C#. Ενώ για την ίδια την εφαρμογή Android χρησιμοποιήθηκε η πλατφόρμα Xamarin. Για το διαμοιρασμό της web εφαρμογής χρησιμοποιήθηκε ένας διακομιστής σε virtual machine. Η σχεσιακή βάση δεδομένων (postgres), η εφαρμογή Java springboot καθώς και η εφαρμογή dotnetcore λειτουργούν σε 3 dockers που επικοινωνούν μεταξύ τους απομονωμένοι από το υπόλοιπο λειτουργικό σύστημα.

Περιεχόμενα

1. ΕΙΣΑΓΩΓΗ	10
1. Γενικά.....	10
2. Περιγραφή τεχνολογιών που χρησιμοποιήθηκαν	13
2.1 Postgresql.....	13
2.2 To Spring Boot Framework.....	13
2.3 Hibernate Framework	14
2.4 AngularJS	15
2.5 Xamarin	17
2.6 Docker	18
3. Λειτουργικές προδιαγραφές	21
3.1 Stakeholders και Actors	21
3.2 Τεχνικές προδιαγραφές	22
3.2.1 Μοντέλο Βάσης	22
3.2.2 Backend (Web & Android)	22
3.2.3 Web Application.....	23
3.2.4 Android Application.....	24
3.3 Use cases	24
4. Μοντέλο Σχεσιακής Βάσης Δεδομένων	42
4.1 Προσδιορισμός	42
4.2 Παραδείγματα χρήσης μοντέλων	47
5. Αρχιτεκτονική συστημάτων	50
5.1 Server Side	50
5.1.1 Service Layer	50
5.1.2 Business Logic Layer	51
5.1.3 Data Layer	52
5.2 Client Side	52
5.2.1 Application Manager.....	52
5.2.2 Model	52
5.2.3 View	52
5.2.4 Controller	52
5.2.5 Components	53
5.2.6 Router	53
6. Βασική Υλοποίηση	54
6.1 Server Side	54
6.1.1 Αποθήκευση δεδομένων.....	54
6.1.2 RESTful Web Services	54
6.1.3 Ασφάλεια	58
6.2 Client Side – Web Application.....	58
6.2.1 Αρχιτεκτονικές	59
6.2.2 Router	61
6.2.3 User Interface.....	62
6.2.4 Controller	63
6.2.5 Services	64
6.2.6 Χάρτες, γραφήματα και πίνακες δεδομένων	65
7. Γραφικό περιβάλλον και περιβάλλον χρήστη.....	66
7.1 Prototypes	66
7.2 Διεπαφή χρήστη	73
7.2.1 Web εφαρμογή διαχείρισης.....	73
7.2.2 Mobile εφαρμογή θέασης.....	88

3. ΣΥΜΠΕΡΑΣΜΑΤΑ	92
4. ΒΙΒΛΙΟΓΡΑΦΙΑ	94

Κατάλογος Σχημάτων

ΣΧΗΜΑ 1.1 ΔΙΑΓΡΑΜΜΑ ΑΝΑΠΑΡΑΣΤΑΣΗΣ ΛΕΙΤΟΥΡΓΙΑΣ ΤΟΥ ΑΥΤΟΜΑΤΟΠΟΙΗΜΕΝΟΥ ΣΥΣΤΗΜΑΤΟΣ ΥΔΡΕΥΣΗΣ	12
ΣΧΗΜΑ 2.1 ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΜΙΑΣ SPRING WEB ΕΦΑΡΜΟΓΗΣ	14
ΣΧΗΜΑ 2.2 ANGULARJS ARCHITECTURE	17
ΣΧΗΜΑ 2.3: XAMARIN - MONO ARCHITECTURE	18
ΣΧΗΜΑ 2.4: DOCKER ARCHITECTURE	19
ΣΧΗΜΑ 3.1: ΔΙΑΧΕΙΡΙΣΗ ΛΕΙΤΟΥΡΓΙΩΝ ΠΡΟΦΙΛ ΧΡΗΣΤΗ	25
ΣΧΗΜΑ 3.2: ΔΙΑΧΕΙΡΙΣΗ ΜΕΤΡΗΣΕΩΝ ΑΙΣΘΗΤΗΡΩΝ	25
ΣΧΗΜΑ 3.3: ΑΥΤΟΜΑΤΟΣ ΑΛΓΟΡΙΘΜΟΣ ΑΡΔΕΥΣΗΣ	26
ΣΧΗΜΑ 4.1: ΜΟΝΤΕΛΟ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ	ERROR! BOOKMARK NOT DEFINED.
ΣΧΗΜΑ 5.1: ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΣΥΣΤΗΜΑΤΟΣ WEB ΕΦΑΡΜΟΓΗΣ	51
ΣΧΗΜΑ 6.1 EMBEDDED ΚΑΙ EXTRACT WEB SERVICES	56
ΣΧΗΜΑ 6.2 HOME CONTROLLER WEB SERVICES	57
ΣΧΗΜΑ 6.3 USER, AUTHENTICATE WEB SERVICES	57
ΣΧΗΜΑ 6.4 WEB SERVICES ΓΙΑ ANDROID APPLICATION	58
ΣΧΗΜΑ 6.5 TWO WAY BINDING ΑΡΧΙΤΕΚΤΟΝΙΚΗ	60
ΣΧΗΜΑ 6.6 ΑΡΧΙΤΕΚΤΟΝΙΚΗ MODEL VIEW VIEW-MODEL	61
ΣΧΗΜΑ 6.7 ΤΑ ΤΡΙΑ ΜΕΡΗ ΤΗΣ WEB ΕΦΑΡΜΟΓΗΣ	62
ΣΧΗΜΑ 6.8 ΜΕΡΗ ANDROID ΕΦΑΡΜΟΓΗΣ	63
ΣΧΗΜΑ 7.1: ΠΡΩΤΟΤΥΠΟ ΑΡΧΙΚΗΣ ΣΕΛΙΔΑΣ ΕΦΑΡΜΟΓΗΣ. ΑΡΙΣΤΕΡΑ ΔΙΑΚΡΙΝΕΤΑΙ ΤΟ ΜΕΝΟΥ ΔΥΝΑΤΟΤΗΤΩΝ ΚΑΙ ΣΤΟ ΠΑΝΩ ΚΟΜΜΑΤΙ ΟΙ ΕΠΙΛΟΓΕΣ ΣΕ ΕΠΙΠΕΔΟ ΧΡΗΣΤΗ	66
ΣΧΗΜΑ 7.2: ΠΡΩΤΟΤΥΠΟ ΑΠΕΙΚΟΝΙΣΗΣ ΜΕΤΡΗΣΕΩΝ ΓΙΑ ΕΠΙΛΕΓΜΕΝΟ ΜΕΤΡΗΣΙΜΟ ΜΕΓΕΘΟΣ, ΕΠΙΛΕΓΜΕΝΗ ΣΥΣΚΕΥΗ ΜΕΤΡΗΣΕΩΝ ΚΑΙ ΕΠΙΛΕΓΜΕΝΟ ΔΙΑΣΤΗΜΑ ΗΜΕΡΟΜΗΝΙΩΝ.	67
ΣΧΗΜΑ 7.3: ΠΡΩΤΟΤΥΠΟ ΑΠΕΙΚΟΝΙΣΗΣ ΑΡΔΕΥΣΗΣ ΕΠΙΛΕΓΜΕΝΗΣ ΣΥΣΚΕΥΗΣ ΚΑΙ ΕΠΙΛΕΓΜΕΝΟΥ ΔΙΑΣΤΗΜΑΤΟΣ ΗΜΕΡΟΜΗΝΙΩΝ.	67
ΣΧΗΜΑ 7.4: ΜΕΓΙΣΤΕΣ ΚΑΙ ΕΛΑΧΙΣΤΕΣ ΤΙΜΕΣ ΓΙΑ ΤΗΝ ΕΝΑΡΞΗ ΤΟΥ ΠΟΤΙΣΜΑΤΟΣ.	68
ΣΧΗΜΑ 7.5: ΠΡΟΓΝΩΣΗ ΚΑΙΡΟΥ ΓΙΑ ΔΙΑΣΤΗΜΑ 7 ΗΜΕΡΩΝ	68
ΣΧΗΜΑ 7.6: ΠΙΝΑΚΑΣ ΕΛΕΓΧΟΥ ΚΑΙ ΔΙΑΧΕΙΡΙΣΗΣ ΣΥΣΚΕΥΩΝ	69
ΣΧΗΜΑ 7.7: ΕΝΕΡΓΕΙΑ «ΝΕΕΣ ΜΕΤΡΗΣΕΙΣ ΓΙΑ ΟΛΕΣ ΤΙΣ ΤΕΛΙΚΕΣ ΣΥΣΚΕΥΕΣ»	69
ΣΧΗΜΑ 7.8: ΈΝΑΡΞΗ ΑΡΔΕΥΣΗΣ ΓΙΑ ΕΠΙΛΕΓΜΕΝΗ ΣΥΣΚΕΥΗ ΚΑΙ ΕΠΙΛΕΓΜΕΝΟ ΔΙΑΣΤΗΜΑ ΗΜΕΡΟΜΗΝΙΩΝ.	70
ΣΧΗΜΑ 7.9: ΑΛΛΑΓΗ ΤΙΜΩΝ ΕΚΤΕΛΕΣΗΣ ΑΛΓΟΡΙΘΜΟΥ ΑΥΤΟΜΑΤΗΣ ΑΡΔΕΥΣΗΣ (ΔΙΑΣΤΗΜΑ ΗΜΕΡΟΜΗΝΙΩΝ, ΧΩΡΗΤΙΚΟΤΗΤΑ ΗΛΕΚΤΡΟΒΑΝΑΣ)	70
ΣΧΗΜΑ 7.10: ΕΠΕΞΕΡΓΑΣΙΑ ΚΑΙ ΘΕΑΣΗ ΠΛΗΡΟΦΟΡΙΩΝ ΓΙΑ ΤΙΣ ΣΥΣΚΕΥΕΣ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ ΑΥΤΟΜΑΤΗΣ ΥΔΡΟΔΟΤΗΣΗΣ.	71
ΣΧΗΜΑ 7.11: ΕΠΙΛΟΓΗ ΣΥΝΤΕΤΑΓΜΕΝΩΝ ΣΗΜΕΙΟΥ ΤΟΠΟΘΕΤΗΣΗΣ ΤΟΥ ΣΤΑΘΜΟΥ.	71
ΣΧΗΜΑ 7.12: ΕΠΕΞΕΡΓΑΣΙΑ ΠΡΟΣΩΠΙΚΩΝ ΣΤΟΙΧΕΙΩΝ ΤΟΥ ΧΡΗΣΤΗ.	72
ΣΧΗΜΑ 7.13: ΑΠΟΣΥΝΔΕΣΗ ΧΡΗΣΤΗ ΑΠΟ ΤΗΝ ΕΦΑΡΜΟΓΗ	72
ΣΧΗΜΑ 7.14: WEB ΕΦΑΡΜΟΓΗ - ΟΘΟΝΗ ΣΥΝΔΕΣΗΣ	73
ΣΧΗΜΑ 7.15: ΑΡΧΙΚΗ ΟΘΟΝΗ	75
ΣΧΗΜΑ 7.16: ΟΘΟΝΗ ΘΕΑΣΗΣ ΚΑΙ ΔΙΑΧΕΙΡΙΣΗΣ ΤΙΜΩΝ ΕΙΔΟΥΣ ΑΤΜΟΣΦΑΙΡΙΚΗΣ ΘΕΡΜΟΚΡΑΣΙΑΣ	76
ΣΧΗΜΑ 7.17: ΟΘΟΝΗ ΘΕΑΣΗΣ ΚΑΙ ΔΙΑΧΕΙΡΙΣΗΣ ΤΙΜΩΝ ΘΕΡΜΟΚΡΑΣΙΑΣ ΣΥΣΚΕΥΗΣ	77
ΣΧΗΜΑ 7.18: ΟΘΟΝΗ ΘΕΑΣΗΣ ΚΑΙ ΔΙΑΧΕΙΡΙΣΗΣ ΤΙΜΩΝ ΥΓΡΑΣΙΑΣ ΑΤΜΟΣΦΑΙΡΑΣ	77
ΣΧΗΜΑ 7.19: ΟΘΟΝΗ ΘΕΑΣΗΣ ΚΑΙ ΔΙΑΧΕΙΡΙΣΗΣ ΤΙΜΩΝ ΥΓΡΑΣΙΑΣ ΕΔΑΦΟΥΣ	78
ΣΧΗΜΑ 7.20: ΑΝΑΦΟΡΑ ΤΙΜΩΝ ΣΥΣΚΕΥΗΣ ΣΕ ΜΟΡΦΗ PDF	78
ΣΧΗΜΑ 7.21: ΑΝΑΦΟΡΑ ΤΙΜΩΝ ΣΥΣΚΕΥΗΣ ΣΕ ΜΟΡΦΗ XLS	79
ΣΧΗΜΑ 7.22: ΟΘΟΝΗ ΘΕΑΣΗΣ ΚΑΙ ΔΙΑΧΕΙΡΙΣΗΣ ΤΙΜΩΝ ΔΙΑΣΤΗΜΑΤΩΝ ΑΡΔΕΥΣΗΣ	80
ΣΧΗΜΑ 7.23: ΟΘΟΝΗ ΔΙΑΧΕΙΡΙΣΗΣ ΑΝΩΦΛΙΩΝ ΚΑΤΩΦΛΙΩΝ ΑΥΤΟΜΑΤΗΣ ΑΡΔΕΥΣΗΣ	81
ΣΧΗΜΑ 7.24: ΣΕΛΙΔΑ ΠΡΟΓΝΩΣΗΣ ΚΑΙΡΟΥ ΓΙΑ ΔΙΑΣΤΗΜΑ 5 ΗΜΕΡΩΝ	82
ΣΧΗΜΑ 7.25: ΠΙΝΑΚΑΣ ΠΡΟΓΝΩΣΗΣ ΚΑΙΡΟΥ ΣΥΓΚΕΚΡΙΜΕΝΗΣ ΗΜΕΡΟΜΗΝΙΑΣ	82
ΣΧΗΜΑ 7.26: ΟΘΟΝΗ «ΠΙΝΑΚΑΣ ΕΛΕΓΧΟΥ» ΔΙΑΧΕΙΡΙΣΗΣ ΕΝΕΡΓΕΙΩΝ ΣΥΣΚΕΥΗΣ ΑΥΤΟΜΑΤΗΣ ΥΔΡΟΔΟΤΗΣΗΣ	83

ΣΧΗΜΑ 7.27: ΕΠΙΒΕΒΑΙΩΣΗ ΝΕΑΣ ΜΕΤΡΗΣΗΣ	84
ΣΧΗΜΑ 7.28: ΈΝΑΡΞΗ ΦΑΙΝΟΜΕΝΟΥ ΑΡΔΕΥΣΗΣ ΣΥΓΚΕΚΡΙΜΕΝΟΥ ΧΡΟΝΙΚΟΥ ΔΙΑΣΤΗΜΑΤΟΣ ..	84
ΣΧΗΜΑ 7.29: ΈΓΚΡΙΣΗ ΑΠΟΘΗΚΕΥΣΗΣ ΔΕΔΟΜΕΝΩΝ	84
ΣΧΗΜΑ 7.30: ΘΕΑΣΗ ΚΑΙ ΔΙΑΧΕΙΡΙΣΗ ΠΛΗΡΟΦΟΡΙΩΝ ΣΥΣΚΕΥΩΝ	85
ΣΧΗΜΑ 7.31: ΑΝΑΔΥΟΜΕΝΟ ΠΑΡΑΘΥΡΟ ΟΡΙΣΜΟΥ ΣΥΝΤΕΤΑΓΜΕΝΩΝ ΣΤΑΘΜΟΥ	86
ΣΧΗΜΑ 7.32: ΕΙΔΟΠΟΙΗΣΕΙΣ ΧΡΗΣΤΗ	87
ΣΧΗΜΑ 7.33: ΟΘΟΝΗ ΕΠΕΞΕΡΓΑΣΙΑΣ ΠΡΟΣΩΠΙΚΩΝ ΔΕΔΟΜΕΝΩΝ ΧΡΗΣΤΗ	87
ΣΧΗΜΑ 7.34: ΟΘΟΝΗ ΑΠΟΣΥΝΔΕΣΗΣ	88
ΣΧΗΜΑ 7.35: ΟΘΟΝΗ ΣΥΝΔΕΣΗΣ ΧΡΗΣΤΗ (MOBILE).....	89
ΣΧΗΜΑ 7.36: ΟΘΟΝΗ ΜΕΝΟΥ - ΕΠΙΛΟΓΗΣ ΚΑΡΤΕΛΑΣ ΘΕΑΣΗΣ (MOBILE).....	90
ΣΧΗΜΑ 7.37: ΘΕΑΣΗ ΔΕΔΟΜΕΝΩΝ ΘΕΡΜΟΚΡΑΣΙΑΣ – ΥΓΡΑΣΙΑΣ ΕΔΑΦΟΥΣ ΑΝΑ ΣΥΣΚΕΥΗ (MOBILE)	91
ΣΧΗΜΑ 7.38: ΘΕΑΣΗ ΔΕΔΟΜΕΝΩΝ ΘΕΡΜΟΚΡΑΣΙΑΣ – ΥΓΡΑΣΙΑΣ ΑΝΑ ΣΥΣΚΕΥΗ (MOBILE)	91

Κατάλογος πινάκων

ΠΙΝΑΚΑΣ 3.1: USE CASE 1: ΕΙΣΟΔΟΣ ΣΤΗΝ ΕΦΑΡΜΟΓΗ	26
ΠΙΝΑΚΑΣ 3.2: USE CASE 2: ΈΞΟΔΟΣ ΑΠΟ ΤΗΝ ΕΦΑΡΜΟΓΗ	27
ΠΙΝΑΚΑΣ 3.3: USE CASE 3: ΕΠΕΞΕΡΓΑΣΙΑ ΠΛΗΡΟΦΟΡΙΩΝ ΧΡΗΣΤΗ.....	27
ΠΙΝΑΚΑΣ 3.4: USE CASE 4: ΘΕΑΣΗ ΕΙΔΟΠΟΙΗΣΕΩΝ.....	28
ΠΙΝΑΚΑΣ 3.5: USE CASE 5: ΔΙΑΓΡΑΦΗ ΕΙΔΟΠΟΙΗΣΕΩΝ	28
ΠΙΝΑΚΑΣ 3.6: USE CASE 6: ΘΕΑΣΗ ΓΡΑΦΗΜΑΤΟΣ ΔΕΔΟΜΕΝΩΝ ΤΕΛΕΥΤΙΑΣ ΜΕΡΑΣ.....	29
ΠΙΝΑΚΑΣ 3.7: USE CASE 7: ΘΕΑΣΗ ΔΕΔΟΜΕΝΩΝ ΤΥΠΟΥ ΜΕΤΡΗΣΗΣ ΣΥΓΚΕΚΡΙΜΕΝΗΣ ΠΕΡΙΟΔΟΥ	30
ΠΙΝΑΚΑΣ 3.8: USE CASE 8: ΚΑΤΕΒΑΣΜΑ ΑΝΑΦΟΡΑΣ ΔΕΔΟΜΕΝΩΝ ΤΥΠΟΥ ΜΕΤΡΗΣΗΣ ΣΥΓΚΕΚΡΙΜΕΝΗΣ ΠΕΡΙΟΔΟΥ	31
ΠΙΝΑΚΑΣ 3.9: USE CASE 9: ΘΕΑΣΗ ΔΕΔΟΜΕΝΩΝ ΑΡΔΕΥΣΗΣ ΣΥΓΚΕΚΡΙΜΕΝΗΣ ΠΕΡΙΟΔΟΥ	31
ΠΙΝΑΚΑΣ 3.10: USE CASE 10: ΚΑΤΕΒΑΣΜΑ ΑΝΑΦΟΡΑΣ ΔΕΔΟΜΕΝΩΝ ΑΡΔΕΥΣΗΣ ΣΥΓΚΕΚΡΙΜΕΝΗΣ ΠΕΡΙΟΔΟΥ	32
ΠΙΝΑΚΑΣ 3.11: USE CASE 11: ΚΑΤΕΒΑΣΜΑ ΑΝΑΦΟΡΑΣ ΔΕΔΟΜΕΝΩΝ ΑΡΔΕΥΣΗΣ ΣΥΓΚΕΚΡΙΜΕΝΗΣ ΠΕΡΙΟΔΟΥ	33
ΠΙΝΑΚΑΣ 3.12: USE CASE 12: ΜΕΤΑΦΟΡΤΩΣΗ ΜΕΤΡΗΣΕΩΝ ΑΙΣΘΗΤΗΡΩΝ.....	34
ΠΙΝΑΚΑΣ 3.13: USE CASE 13: ΜΕΤΑΦΟΡΤΩΣΗ ΜΕΤΡΗΣΗΣ ΑΡΔΕΥΣΗΣ.....	35
ΠΙΝΑΚΑΣ 3.14: USE CASE 14: ΔΗΜΙΟΥΡΓΙΑ ΣΥΜΒΑΝΤΩΝ ΑΡΔΕΥΣΗΣ ΚΑΙ ΜΕΤΡΗΣΕΩΝ ΑΙΣΘΗΤΗΡΩΝ.....	35
ΠΙΝΑΚΑΣ 3.15: USE CASE 15: ΕΠΕΞΕΡΓΑΣΙΑ ΜΕΓΙΣΤΩΝ ΚΑΙ ΕΛΑΧΙΣΤΩΝ ΤΙΜΩΝ ΑΥΤΟΜΑΤΟΥ ΑΛΓΟΡΙΘΜΟΥ	36
ΠΙΝΑΚΑΣ 3.16: USE CASE 16: ΝΕΑ ΜΕΤΡΗΣΗ.....	37
ΠΙΝΑΚΑΣ 3.17: USE CASE 17: ΑΡΔΕΥΣΗ ΣΥΓΚΕΚΡΙΜΕΝΟΥ ΔΙΑΣΤΗΜΑΤΟΣ.....	37
ΠΙΝΑΚΑΣ 3.18: USE CASE 18: ΕΠΕΞΕΡΓΑΣΙΑ ΔΙΑΣΤΗΜΑΤΟΣ ΛΕΙΤΟΥΡΓΙΑΣ ΑΛΓΟΡΙΘΜΟΥ ΑΥΤΟΜΑΤΗΣ ΑΡΔΕΥΣΗΣ	38
ΠΙΝΑΚΑΣ 3.19: USE CASE 19: ΔΙΑΧΕΙΡΙΣΗ ΟΝΟΜΑΤΩΝ ΠΕΔΙΩΝ ΕΝΔΙΑΦΕΡΟΝΤΟΣ.....	39
ΠΙΝΑΚΑΣ 3.20: USE CASE 20: ΘΕΑΣΗ ΚΑΙ ΔΙΑΧΕΙΡΙΣΗ ΓΕΩΓΡΑΦΙΚΟΥ ΣΗΜΕΙΟΥ ΘΕΣΗΣ ΤΟΥ ΣΤΑΘΜΟΥ	39
ΠΙΝΑΚΑΣ 3.21: USE CASE 20: ΘΕΑΣΗ ΤΙΜΩΝ ΕΒΔΟΜΑΔΙΑΙΑΣ ΠΡΟΓΝΩΣΗΣ ΚΑΙΡΟΥ	40
ΠΙΝΑΚΑΣ 3.21: USE CASE 21: ΚΑΤΕΒΑΣΜΑ ΜΕΓΙΣΤΩΝ, ΕΛΑΧΙΣΤΩΝ ΤΙΜΩΝ, ΣΥΝΘΗΚΩΝ ΜΕΤΡΗΣΗΣ ΚΑΙ ΑΡΔΕΥΣΗΣ, ΔΙΑΣΤΗΜΑΤΟΣ ΑΥΤΟΜΑΤΟΥ ΑΛΓΟΡΙΘΜΟΥ	40
ΠΙΝΑΚΑΣ 3.22: USE CASE 22: ΠΑΡΟΧΗ ΔΕΔΟΜΕΝΩΝ ΠΡΟΓΝΩΣΗΣ ΚΑΙΡΟΥ	41

Κατάλογος Συμβόλων

B.Δ	Βάση Δεδομένων
OGC	Open Geospatial Consortium
IoT	Internet of Things
RDBMS	Relational Database Management Systems
ORM	Object Relation Mapping
MVC	Model View Controller
IoC	Inversion of Control
JDK	Java Development Kit
POJO	Plain Old Java Object
HQL	Hibernate Query Language
OOP	Object Oriented Programming
MVVM	Model View View Model
DOM	Document Object Models
DI	Dependency Injection
CLR	Common Language Runtime
JNI	Java Native Interface
CLI	Common language interface
SOS	Sensor Observation Services
RIA	Rich Internet Application
URL	Uniform Resource Locators
SOAP	Simple Object Access Protocol
XAML	Extensible Application Markup Language
RIA	Rich Internet Applications
UI	User Interface
SPA	Single Page Applications
AMQP	Advanced Message Queuing Protocol
RIA	Rich Internet Application
CSV	Comma Separated Values
PDF	Portable Document Format
XAML	Extensible Application Markup Language

1.

ΕΙΣΑΓΩΓΗ

1. Γενικά

Στις μέρες μας η συνεχής αύξηση του πληθυσμού και η βελτίωση της ποιότητας ζωής καθιστά τη μείωση της κατανάλωσης ενεργειακών και φυσικών πόρων, ως ένα μείζον και κομβικό ζήτημα [1]. Η ανάγκη για την παραγωγή όλο και περισσότερων γεωργικών προϊόντων θέτει σε πρώτο πλάνο την ανάγκη εξοικονόμησης νερού και τη βέλτιστη αξιοποίηση του [2]. Η ανάγκη για χρήση νέων τεχνολογιών με σκοπό τη βελτιστοποίηση των διαδικασιών της γεωργικής παραγωγής καθώς και η λεπτομερής καταγραφή των δεδομένων-συνθηκών (θερμοκρασίας, υγρασίας) που αυτή επιτυγχάνεται γίνεται όλο και πιο επιτακτική.

Η δεκαετία που διανύουμε έχει χαρακτηριστεί από πολλούς σαν δεκαετία της πληροφορίας. Η ανάπτυξη όλο και περισσότερων εφαρμογών Internet Of Things (IoT) καθώς και η ενσωμάτωση των ασύρματων δικτύων αισθητήρων σε πολλούς τομείς της παραγωγής δίνει τη δυνατότητα καταγραφής και ανάλυσης μεγαλύτερου όγκου πληροφοριών προς καλύτερη κατανόηση και εξαγωγή ασφαλέστερων συμπερασμάτων [3]. Στον τομέα της γεωργίας, τα τελευταία χρόνια έχουν κάνει την εμφάνισή τους πληθώρα «έξυπνων» συσκευών και εφαρμογών, ακολουθώντας έτσι τομείς όπου η χρήση τέτοιων τεχνολογιών ήταν αναπόσπαστο κομμάτι της παραγωγής εδώ και χρόνια (βλ. βιομηχανική παραγωγή, ναυσιπλοΐα, ναυπηγική, αεροναυπηγική κ.α.). Ακόμα και στην οικιακή χρήση παρατηρείται η ανάπτυξη μιας πληθώρας συσκευών embedded και εφαρμογών για όλες τις πτυχές της καθημερινής μας ζωής. Για όλα τα παραπάνω σημαντικός παράγοντας είναι η βελτίωση της ταχύτητας των ασυρμάτων δικτύων καθώς και των πρωτοκόλλων πάνω στα οποία στηρίζονται (WiFi, Bluetooth, ZigBee, Lora...).

Σκοπός της συλλογής, της αξιολόγησης και της σύγκρισης όλων αυτών των δεδομένων είναι η λήψη πιο σωστών αποφάσεων για τη βέλτιστη χρήση των φυσικών πόρων. Πιο συγκεκριμένα, στον τομέα της γεωργίας, η όσο το δυνατόν καλύτερη αξιοποίηση του νερού σε σχέση με τις περιβαλλοντικές συνθήκες, για την παραγωγή αγροτικών προϊόντων βοηθάει σε μεγάλο βαθμό κυρίως τις περιοχές που «υποφέρουν» από έλλειψη νερού. Η χρήση απλών και εύχρηστων για το ευρύ κοινό εργαλείων-εφαρμογών (εφαρμογές φυλλομετρητή, εφαρμογές «έξυπνων τηλεφώνων»), προς την επίτευξη του σκοπού αυτού, καθώς και η συλλογή δεδομένων

για την παραγωγή ασφαλών συμπερασμάτων ως προς τις καλλιέργειες, ανά περιοχή αποτελεί τον κορμό της παρούσας διπλωματικής.

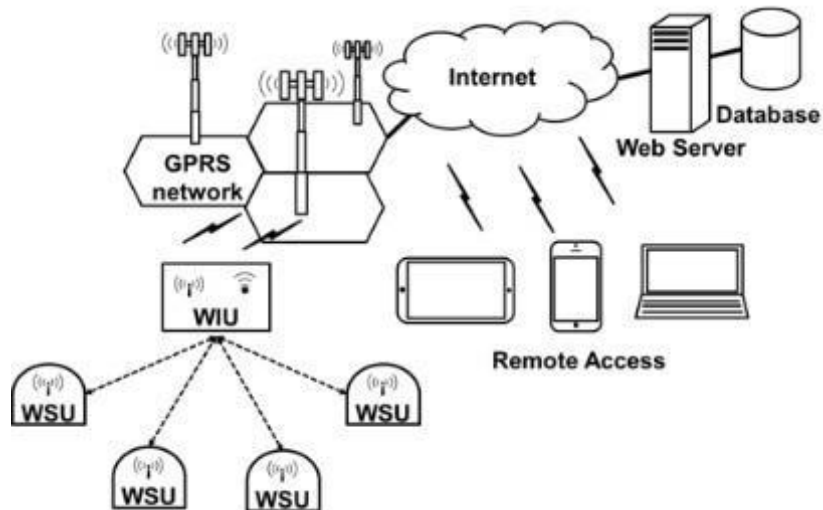
Το πεδίο στο οποίο κινήθηκε η υλοποίηση του ενσωματωμένου συστήματος είναι η καλλιέργεια και η γεωργική παραγωγή όπως αυτή λαμβάνει χώρα στην ελληνική ύπαιθρο. Ενώ η σχεδίαση της διεπαφής του χρήστη στις εφαρμογές (web & mobile) πραγματοποιήθηκε με γνώμονα, να μπορεί να εξυπηρετήσει και να βοηθήσει άμεσα όσους εργάζονται σε αγροκτήματα. Δηλαδή χωρίς να κρίνεται απαραίτητη η εγκατάσταση κάποιου ειδικού λογισμικού ή κάποιου είδους εκπαίδευση. Έτσι οι εφαρμογές δημιουργήθηκαν με τέτοιο τρόπο ώστε να είναι φιλικές και παραμετροποιήσιμες από τους ίδιους τους χρήστες. Η επιλογή της αρχιτεκτονικής του συστήματος έγινε ώστε να μπορεί να λειτουργεί χωρίς τη χρήση modem για τη σύνδεση στο WiFi, γι' αυτό προτιμήθηκε η χρήση GPRS δικτύου καθώς και του πρωτόκολλου επικοινωνίας ZigBee μεταξύ των αισθητήρων και του σταθμού. Η σχεσιακή βάση δεδομένων και τα webservices, κατασκευάστηκαν σύμφωνα με τα Sensor Observation Services (SOS) πρότυπα, έτσι όπως έχουν οριστεί από την OGC. Η λειτουργία των services επιτρέπει πρόσβαση στα χαρακτηριστικά των αισθητήρων (περιγραφή, όνομα και αναγνωριστικός αριθμός) καθώς και στις μετρήσεις που αυτοί συλλέγουν. Είναι δυνατή η πρόσθεση νέων αισθητήρων καθώς και την κατάργηση-απενεργοποίηση άλλων. Δίνεται επίσης στους χρήστες η δυνατότητα σύνδεσης με τύπο φαινομένων και μονάδα μετρήσεων για διαφορετικού είδους μετρήσεις (π.χ. αριθμητικά, γεωγραφικά, αλφαριθμητικά κ.α.).

Η παρούσα διπλωματική εργασία αναλύει τις μεθόδους, καθώς και τα εργαλεία που χρησιμοποιήθηκαν για την ανάπτυξη εφαρμογών, συμπληρωματικών ως προς την κατασκευή του ενσωματωμένου συστήματος (hardware) για να μπορούν οι χρήστες να αλληλεπιδρούν με αυτό. Το hardware το οποίο σχεδιάστηκε και αναπτύχθηκε με μορφή ενσωματωμένου συστήματος, καθώς και ο προγραμματισμός αυτού αποτελούν κομμάτια μιας συμπληρωματικής διπλωματικής εργασίας. Στο Σχήμα 1.1 Διάγραμμα αναπαράστασης λειτουργίας του αυτοματοποιημένου συστήματος ύδρευσης Σχήμα 1.1 φαίνονται τα κομμάτια (hardware & software) που συνεργάζονται για την παραγωγή της τελικής λύσης.

Στα επόμενα κεφάλαια παρουσιάζονται τα εργαλεία που χρησιμοποιήθηκαν, καθώς και η αρχιτεκτονική που επιλέχθηκε για τη δημιουργία των εφαρμογών. Η αρχιτεκτονική αυτή αναπτύσσεται από τα πιο χαμηλά επίπεδα στα πιο υψηλά ξεκινώντας από τον διακομιστή (server), τα εικονικά λειτουργικά (virtual machines), τους dockers που «φιλοξενούν» τις εφαρμογές καθώς και τα API (Application Programming Interface). Στη συνέχεια παρουσιάζεται η σχεσιακή βάση δεδομένων, το σχήμα και τα υπόλοιπα χαρακτηριστικά (constraints, foreign keys κ.α.). Μετέπειτα τα webservices που δημιουργήθηκαν για τις δύο εφαρμογές (web, mobile). Τέλος παρουσιάζονται τα mockups, use interfaces που αναπτύχθηκαν για την επικοινωνία των συστημάτων με τους χρήστες. Ειδικότερα τα κεφάλαια που απαρτίζουν τη συγκεκριμένη διπλωματική εργασία είναι τα εξής:

- **Κεφάλαιο 2:** Παρουσιάζει τα βασικά χαρακτηριστικά και τα πλεονεκτήματα των τεχνολογιών που χρησιμοποιήθηκαν.
- **Κεφάλαιο 3:** Περιγράφει τις λειτουργικές προδιαγραφές, τα είδη χρηστών και τα use cases που αναπτύχθηκαν για τις εφαρμογές.
- **Κεφάλαιο 4:** Παρουσιάζει το μοντέλο της σχεσιακής βάσης δεδομένων που αναπτύχθηκε για την υποστήριξη του συστήματος και περιγράφει αναλυτικά τους επιμέρους πίνακες.
- **Κεφάλαιο 5:** Περιγράφει την αρχιτεκτονική που χρησιμοποιήθηκε για τις εφαρμογές τόσο στον server όσο και στον client και αναλύει τα επιμέρους στοιχεία που τη συνθέτουν και τον τρόπο που αυτά επικοινωνούν μεταξύ τους.

- **Κεφάλαιο 6:** Παρουσιάζει τα κομμάτια που συνθέτουν τη βασική υλοποίηση των εφαρμογών που αναπτύχθηκαν. Αυτών που εξυπηρετούν τις κλήσεις στο διακομιστή, όσο και αυτές που εξυπηρετούν τις εφαρμογές web & mobile.
- **Κεφάλαιο 7:** Παρουσιάζει τα πρωτότυπα της σχεδίασης των εφαρμογών. Περιγράφει τον τρόπο λειτουργίας τους και τις δυνατότητες που δίνουν στο χρήστη καθώς και παραδείγματα λειτουργίας του μέσω εικόνων.



Σχήμα 1.1 Διάγραμμα αναπαράστασης λειτουργίας του αυτοματοποιημένου συστήματος ύδρευσης

2. Περιγραφή τεχνολογιών που χρησιμοποιήθηκαν

Στο παρόν κεφάλαιο παρουσιάζονται τα εργαλεία, οι εφαρμογές που είναι σχετικές με την παρούσα διπλωματική εργασία. Για καθένα από αυτά παρατίθενται μια σύντομη αναφορά για τις δυνατότητες τους καθώς και για το πως χρησιμοποιήθηκαν στο project.

2.1 Postgresql

Η postgres είναι ένα σύστημα διαχείρισης σχεσιακής βάσης δεδομένων με έμφαση στην επεκτασιμότητα και τη συμμόρφωση με τα πρότυπα [4]. Η ανάπτυξη της ξεκίνησε από το Πανεπιστήμιο του Berkeley και αποτελεί αναβάθμιση του παλιού project της ingres. Είναι ένα project ανοικτού κώδικα και χρησιμοποιείται σε πληθώρα εφαρμογών, ενδεικτικό είναι ότι παρέχει σύνδεση για το σύνολο σχεδόν των γλωσσών προγραμματισμού. Έχει όλες τις δυνατότητες μια σχεσιακής βάσης δεδομένων, όπως complex queries, foreign keys, triggers, updatable views, transactional integrity, multiversion concurrency control κ.α. παρέχει τη δυνατότητα παραμετροποίησης στους χρήστες της καθώς μπορούν να φτιάξουν data types, functions, operators, aggregate functions, index methods κ.α.

Σε σύγκριση με άλλες σχεσιακές βάσεις δεδομένων η postgres μπορεί να εγκατασταθεί σε πληθώρα λειτουργικών συστημάτων. Παρέχει τον όγκο βάσης δεδομένων στο σύστημα αρχείων του λειτουργικού, μέγεθος πίνακα 32TB, μέγιστο μέγεθος γραμμής 2gb, μέγιστο αριθμό στηλών ανά γραμμή: 1536, μέγιστο Blob/Clob μέγεθος: 2GB κ.α. Δίνει επίσης τη δυνατότητα χρήσης «Temporary tables» και «Materialized views». Μπορεί να κάνει χρήση των ακόλουθων Indices: R-/R+ tree, Hash, Expression, Partial, Reverse, Bitmap, GiST, GIN, Full-text, Spatial. Χρησιμοποιεί όλους τους τύπους δεδομένων καθώς και πιο περίπλοκους χρησιμοποιώντας ορισμένες επεκτάσεις όπως το postgis. Το postgis είναι μια από τις πιο γνωστές επεκτάσεις για την Postgres, είναι ελεύθερο και ανοικτού κώδικα λογισμικό. Χρησιμοποιείται για εφαρμογές που σχετίζονται με GIS καθώς περιέχει τύπους δεδομένων όπως γεωγραφικά δεδομένα (πολύγωνα, γραμμές, σημεία) καθώς και τοπολογίες, raster κ.α. spatial indices για αναζήτηση σε γεωγραφικά δεδομένα όπως είναι τα R-tree-over-GIST. Χρησιμοποιείται σαν τρόπος αποθήκευσης από πολλές GIS εφαρμογές ArcGIS (via GISquirrel, ST-Links SpatialKit, ZigGIS, ArcSDE and other third-party connectors), Cadcorp SIS, CartoDB, CitySurf Globe, GeoMedia (via third-party connectors), GeoServer, GeoNetwork, GRASS GIS (GPL), gvSIG (GPL) κ.α.

2.2 To Spring Boot Framework

Το spring framework αποτελεί μια «ελαφριά» υλοποίηση εναλλακτική στις εκδόσεις JEE και J2EE της Java [5]. Ο πυρήνας του framework μπορεί να χρησιμοποιηθεί από οποιαδήποτε εφαρμογή που χρησιμοποιεί τη γλώσσα προγραμματισμού Java. Το spring framework περιέχει ενότητες (modules) που παρέχουν πολλές δυνατότητες, οι οποίες περιγράφονται λεπτομερώς στη συνέχεια.

- **Conversion and Configuration:**

Παραμετροποίηση της αρχιτεκτονικής, έτσι ώστε να μπορεί να χρησιμοποιηθεί για την ανάπτυξη διαφορετικών εφαρμογών.

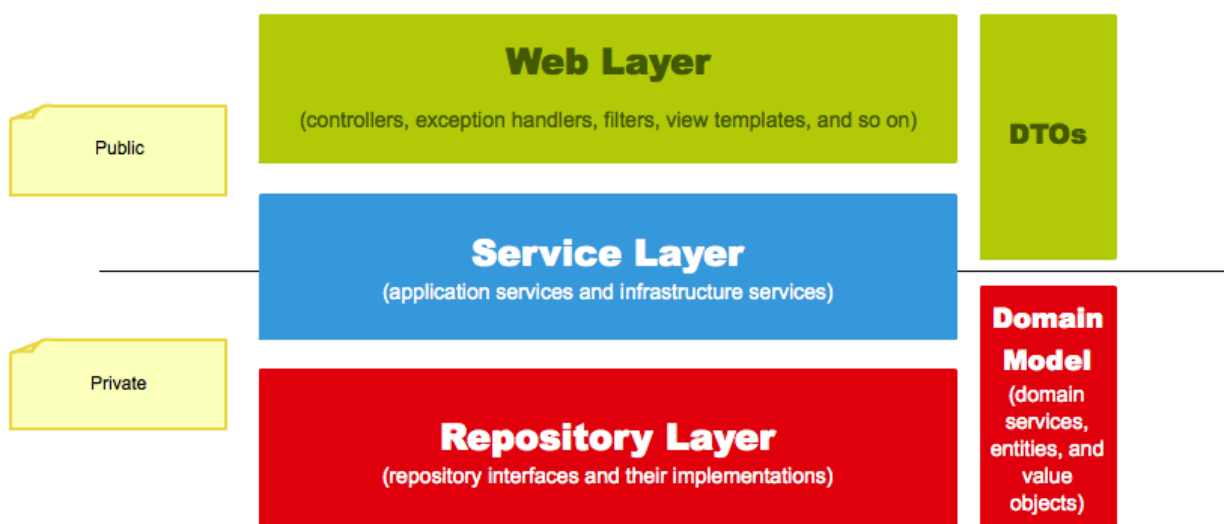
- **Data access:**

Παρέχει τη δυνατότητα σύνδεσης του framework με διάφορες βάσεις δεδομένων είτε σχεσιακές (RDBMS) είτε NOSQL. Το JDBC παρέχει έναν καλό τρόπο για την διασύνδεση αλλά

παρόλα αυτά υπάρχει η δυσκολία χρήσης του API που υποστηρίζει. Παρέχεται όμως η δυνατότητα χρήσης της πλειονότητας των access και frameworks ORM (Object Relation Mapping) εργαλείων (JDBC, Hibernate κ.α.) Αυτό σημαίνει ότι το framework παρέχει υποστήριξη όπως (α) Resource management: Μπορούν να χειριστούν τη θέση και τη διαμόρφωση των Hibernate Session Factories, JDBC data sources. Αυτό καθιστά δυνατό την παραμετροποίηση τιμών. (β) Transaction management. Διαχείριση transactions στο επίπεδο χρήσης του ORM εργαλείου με τη δυνατότητα επαναφοράς, ή ανακατεύθυνσης. (γ) Η αρχιτεκτονική δίνει τη δυνατότητα εύρεσης και εντοπισμού των πόρων που σχετίζονται με τα επίπεδα του ORM.

- **WEB**

Περιέχει ένα δυναμικό και παραμετροποιήσιμο Model View Controller (MVC) framework. Παρέχεται ένας καθαρός διαχωρισμός μεταξύ controllers, models και views. Είναι ευέλικτο ως προς την παραμετροποίηση του εφόσον στηρίζεται σε διεπαφές (interfaces) οι οποίες είναι παραμετροποιήσιμες και μπορούν να κατασκευαστούν νέες για τις ανάγκες της κάθε εφαρμογής. Οι κλάσεις του χρησιμοποιούν την σχεδίαση Inversion of Control (IoC). Είναι μια δυνατότητα όπου μπορεί αντικείμενα μπορούν να διαχειριστούν και να ρυθμιστούν χρησιμοποιώντας reflection.



Σχήμα 2.1 Αρχιτεκτονική μιας Spring web εφαρμογής

2.3 Hibernate Framework

Το Hibernate [6] είναι ένα λογισμικό ανοιχτού κώδικα που χρησιμοποιείται για τη διασύνδεση των πινάκων μιας σχεσιακής βάσης δεδομένων με τα αντικείμενα (κλάσεις) που δημιουργούνται χρησιμοποιώντας την αντικειμενοστραφή γλώσσα προγραμματισμού (Java). Για να καταστεί δυνατή αυτή η σύνδεση χρησιμοποιούνται metadata σε ξεχωριστά αρχεία. Το hibernate κατασκευάστηκε και λειτουργεί σύμφωνα με τα πρότυπα τις J2EE. Στις τελευταίες εκδόσεις έχουν ενσωματωθεί αρκετά χαρακτηριστικά από τις τελευταίες εκδόσεις Java Development Kit (JDK), όπως τα annotations.

Είναι λογισμικό τύπου ORM. Τα λογισμικά αυτά χρησιμοποιούνται για τη διασύνδεση μέσω διεπαφών, σχεσιακών και άλλων βάσεων δεδομένων με σκοπό τη χρησιμοποίησή τους σαν αντικειμενοστραφείς κλάσεις. Η διασύνδεση επιτυγχάνεται είτε μέσω αρχείων xml, είτε μέσω κλάσεων Java με annotations. Παρέχει πολλά πλεονεκτήματα όπως τη σύνδεση των

αντικειμένων μέσω σχέσεων (one-to-many, one-to-one, many-to-many) με άλλα instances αντικειμένων. Για κάθε μέλος (property) μιας κλάσης που αντιπροσωπεύει ένα πίνακα παρέχεται η δυνατότητα της επανεγγραφής του τύπου όπως αυτό είναι στη βάση, τη διασύνδεση με enumerations και τη διασύνδεση ενός property σε πολλές στήλες ενός πίνακα. Τα αντικείμενα που δημιουργούνται ονομάζονται Plain Old Java Object (POJO). Το framework παρέχει επίσης μια γλώσσα φιλική προς προγραμματιστές που ονομάζεται HQL (Hibernate Query Language). Μια γλώσσα που μας επιτρέπει τη χρήση των POJO κλάσεων για τη δημιουργία SQL-like queries, τα οποία με τη σειρά τους μεταφράζονται σε sql queries.

Τα πλεονεκτήματα που προσφέρει το συγκεκριμένο framework είναι πολλά. Κάποια από αυτά είναι (α) Η αύξηση της παραγωγικότητας, αφού δύναται να ξεπεραστεί το κομμάτι της κατασκευής διεπαφής με τη βάση δεδομένων για τη δημιουργία των queries. (β) Η δυνατότητα συντήρησης ενός project αφού αποφεύγονται τα μεγάλα queries. Ο κώδικας της εφαρμογής έτσι είναι πιο περιεκτικός και καλογραμμένος ακολουθώντας τους κανόνες που ορίζει το OOP (Object Oriented Programming) πρότυπο. (γ) Δυνατότητα χρήσης των ίδιων αντικειμένων για πολλαπλές βάσεις δεδομένων αλλάζοντας ή προσθέτοντας μόνο τα κομμάτια της διασύνδεσης.

Τέλος οι ενότητες λογισμικού που χρησιμοποιεί το λογισμικό αυτό είναι:

- Το βασικό λογισμικό για το ORM.
- Hibernate annotations: Μεταδεδομένα που χρησιμοποιούνται για τη μετατροπή μεταξύ των αντικειμενοστραφών αντικειμένων και των μοντέλων της σχεσιακής βάσης δεδομένων.
- Hibernate Entity Manager: Το API αυτό χρησιμοποιείται για την πρόσβαση σε μια βάση δεδομένων. Χρησιμοποιείται για δημιουργία ή αφαίρεση οντοτήτων, εύρεση τους μέσω πρωτευόντων κλειδίων καθώς και αναζήτηση σε όλες τις οντότητες.
- Hibernate Events: Έλεγχος και δημιουργία εκδόσεων των κλάσεων.

2.4 AngularJS

Το AngularJS framework [7] είναι ένα ανοιχτού κώδικα front-end framework, το οποίο βασίζεται στη γλώσσα προγραμματισμού Javascript. Κατασκευάστηκε και συντηρείται κατά κύριο λόγο από την google. Στοιχεύει να απλοποιήσει τις διαδικασίες ανάπτυξης και δοκιμής των εφαρμογών προσφέροντας ένα framework για client-side MVC και MVVM (model view view model) αρχιτεκτονικές. Στο Σχήμα 2.2 φαίνεται σχηματικά η αρχιτεκτονική μιας angular εφαρμογής.

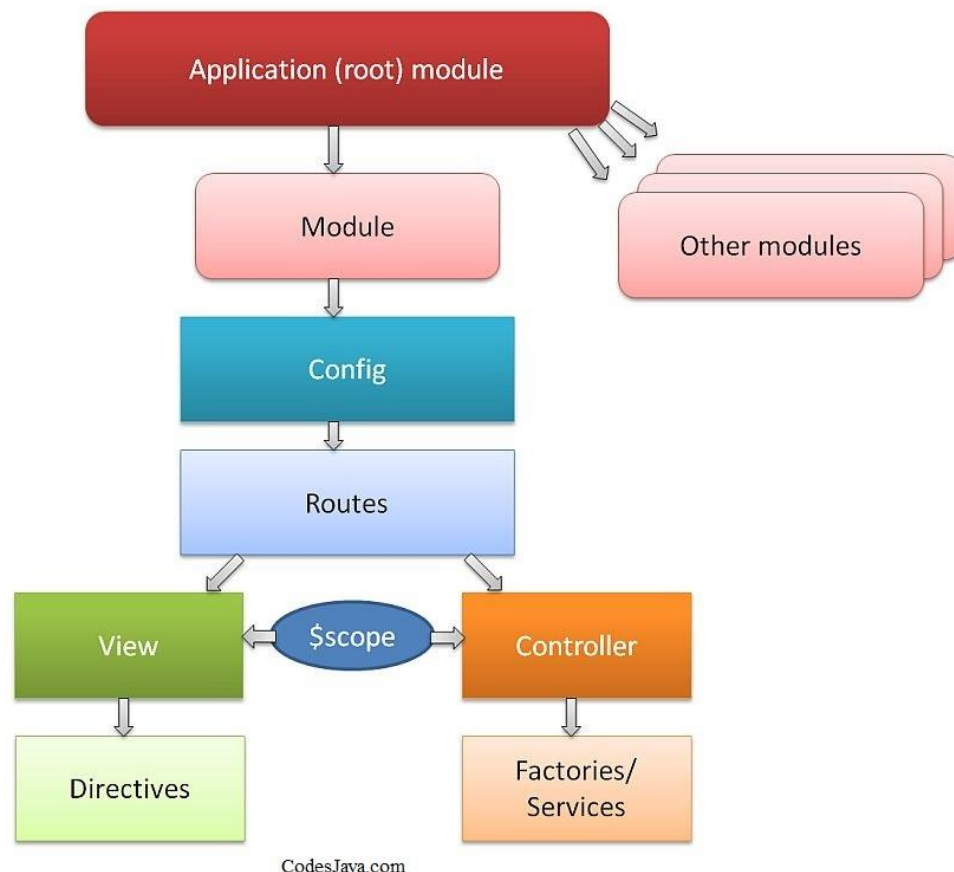
- **Modules:** Τα βασικά στοιχεία δόμησης μιας angular εφαρμογής είναι τα modules, τα οποία προσφέρουν ένα περιβάλλον σύνδεσης για τα στοιχεία. Τα modules χρησιμοποιούνται για να συγκεντρώσουν σχετιζόμενα κομμάτια κώδικα σε λειτουργικά συστατικά. Μια εφαρμογή angular αποτελείται από τουλάχιστον ένα ή περισσότερα modules. Οι βασικοί στόχοι της αρχιτεκτονικής αυτής είναι οι εξής:
 - Να αποσυνδέσει το χειρισμό DOM (Document Object Models) αντικειμένων από τη λογική της κάθε εφαρμογής.
 - Να αποσυνδέσει το client side μιας εφαρμογής από το server side. Το συγκεκριμένο επιτρέπει να δουλεύονται τα δύο αυτά κομμάτια παράλληλα.
 - Να δημιουργηθεί η δομή για να γίνει η ανάπτυξη μιας εφαρμογής από το σχεδιασμό του UI (User interface) στην υλοποίηση του business logic και στη δημιουργία αυτοματοποιημένων δοκιμών.

- DI (Dependency Injection)

Η AngularJS υλοποιεί το MVC πρότυπο σχεδίασης. Χρησιμοποιώντας DI (Dependency Injection) μας δίνεται η δυνατότητα χρήσης των παραδοσιακών server-side υπηρεσιών σε client-

side εφαρμογές. Πιο συγκεκριμένα το DI είναι ένα πρότυπο σχεδίασης που ασχολείται με το πως components μένουν με τα dependencies τους.

- **Scope:** Η angular χρησιμοποιεί τον όρο scope σαν ένα συγκεκριμένο είδος αντικειμένου. Αυτό σημαίνει ότι μπορεί να είναι είτε εκτός είτε εντός πεδίου εφαρμογής, οπουδήποτε στο εκτελέσιμο πρόγραμμα. Αναλαμβάνει να συνδέσει την πλευρά του «View» με αυτή του «Controller». Συγκεκριμένα το scope είναι αυτό που δίνει μορφή στα «models» και όλες οι μεταβλητές που υπάρχουν στο view μπορούν να γίνουν προσβάσιμες από την πλευρά του controller.
- **Bootstrapping:** Είναι η ενέργεια που εκτελείται από τον bootstrapper της AngularJS μετά τη φόρτωση του DOM. Οι ενέργειες που λαμβάνουν χώρα είναι οι εξής:
 - Δημιουργία νέου injector.
 - Συλλογή όλων των directives που κοσμούν το DOM.
 - Σύνδεση όλων των directives με το scope.
- **Directives:** Ένα directive μπορεί να αλλάξει τη συμπεριφορά και την εμφάνιση ενός αντικειμένου DOM. Η angular διαθέτει αρκετά. Επίσης δίνει τη δυνατότητα κατασκευής directive τα οποία χωρίζονται σε structural και attribute. Τα πρώτα μπορούν να προσθέσουν ή να αφαιρέσουν στοιχεία DOM. Τα δεύτερα χρησιμοποιούνται για την αλλαγή της συμπεριφοράς ενός στοιχείου, ενός component ή κάποιου άλλου directive.
- **Controllers:** Οι controllers αποτελούν ένα σημαντικό στοιχείο της εφαρμογής, είναι Javascript objects που καθορίζονται από μια μέθοδο κονστράκτορα έτσι ώστε να γίνουν inject από κάποιο άλλο κομμάτι της εφαρμογής. Προσθέτουν όλη τη λειτουργικότητα στο scope καθώς είναι το σημείο που λαμβάνει χώρα όλο το business logic της εφαρμογής.
- **Services:** Τα services μπορούν να κάνουν http κλήσεις προς ένα απομακρυσμένο διαδικτυακό server. Αποτελούνται από Javascript objects τα οποία υλοποιούν http υπηρεσίες. Η angular δίνει τη δυνατότητα δημιουργίας και χρήσης προσαρμοσμένων services πέρα των ενσωματωμένων που διαθέτει.
- **Filters:** Τα φίλτρα μπορούν να γίνουν injected και να χρησιμοποιηθούν σε διάφορα σημεία της αρχιτεκτονικής. Μπορούν να χρησιμοποιηθούν μέσα στα services, στους controllers καθώς και στα html templates. Ο ρόλος τους είναι να μορφοποιούν τα δεδομένα.



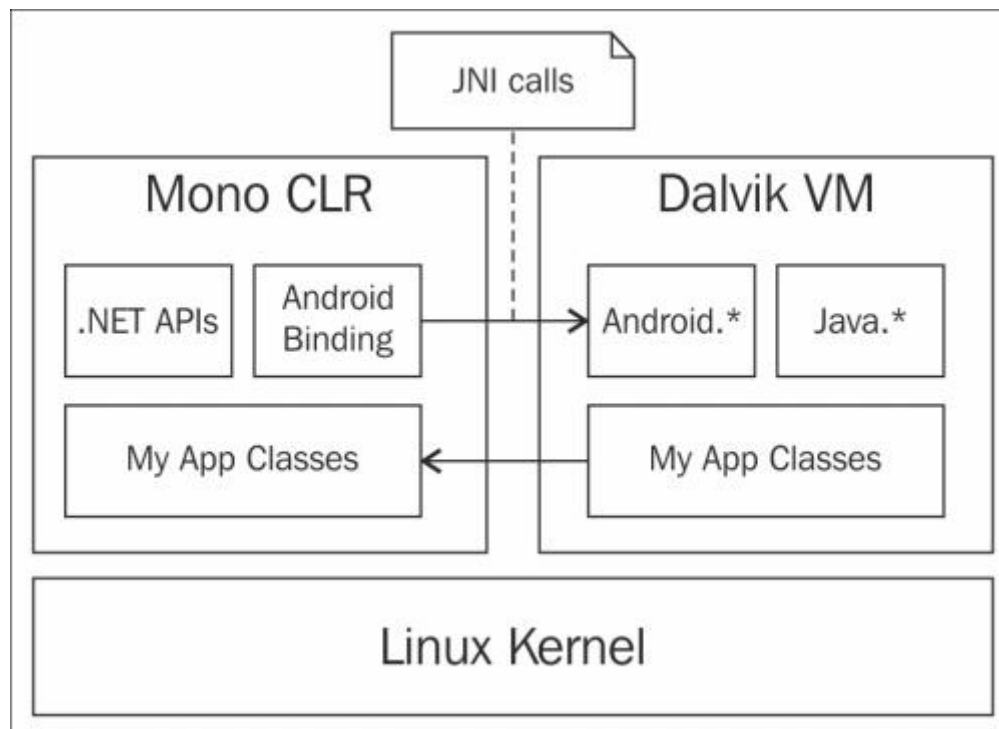
Σχήμα 2.2 AngularJS Architecture

2.5 Xamarin

Το Xamarin [8] είναι μια εταιρεία που υπάγεται στη Microsoft και αναπτύσσει μια πλατφόρμα ανάπτυξης λογισμικού για εφαρμογές iOS, OS X και Android χρησιμοποιώντας τη γλώσσα προγραμματισμού c# και το .NET framework. Η ανάπτυξη έχει στηριχτεί στο project ελεύθερου λογισμικού Mono. Το mono είναι ένα μια υλοποίηση για πολλαπλά Λ.Σ. του C# μεταγλωττιστή και του CLR που είναι συμβατό με το Microsoft .net. Το mono CLR μπορεί να μεταφερθεί σε πολλές πλατφόρμες όπως Android, Linux Distribution κ.α.

Μια εφαρμογή Xamarin Android μπορεί να τρέξει χρησιμοποιώντας το mono CLR (Common Language Runtime) καθώς και το Dalvik VM. Η συνεργασία αυτών των δυο τμημάτων επιτυγχάνεται μέσω ενός framework που ονομάζεται JNI (Java Native Interface) και των peer objects. Το JNI επιτρέπει σε ένα κομμάτι κώδικα διαφορετικό από Java να καλέσει ή να καλεστεί μέσω ενός Java κομματιού κώδικα το οποίο υλοποιείται μέσα σε ένα JVM. Τα Peer Objects αντιστοιχούν σε ένα σύνολο αντικειμένων που «κατοικούν» είτε μέσα στο χώρο του mono CLR είτε στο χώρο του Dalvik VM. Το Xamarin.Android παραδίδεται με ένα σετ από βιβλιοθήκες. Οι κλάσεις στις βιβλιοθήκες αυτές αντιστοιχούν στις Java κλάσεις στο Android application framework και οι μέθοδοι λειτουργούν σαν wrappers για να καλέσουν άλλες μεθόδους στις Java κλάσεις. Έτσι κάθε φορά που δημιουργείται μια C# κλάση, που κληρονομεί από τις κλάσεις δεσμού, μια νέα proxy Java κλάση δημιουργείται. Μια proxy κλάση είναι συνήθως μία διεπαφή (interface) ή ένας mediator. Η proxy κλάση περιέχει μια παραγόμενη επανεγγραφή για όλες τις επανεγγράψιμες μεθόδους στη C# κλάση.

Στο Σχήμα 2.3 φαίνεται η αρχιτεκτονική μιας εφαρμογής χτισμένης στην πλατφόρμα του Xamarin.



Σχήμα 2.3: Xamarin - Mono Architecture

2.6 Docker

Το Docker [9] είναι μια πλατφόρμα λογισμικού ανοικτού κώδικα που υλοποιεί εικονικοποίηση (virtualization) σε επίπεδο λειτουργικού συστήματος (Λ.Σ). Ο όρος αυτός χρησιμοποιείται για τα λειτουργικά συστήματα των οποίων ο πυρήνας τους επιτρέπει την ύπαρξη πολλαπλών απομονωμένων (isolated) υποδειγμάτων χώρων χρήστη. Αυτά τα υποδείγματα ονομάζονται containers. Αρχικά η υλοποίηση ξεκίνησε για το λειτουργικό σύστημα Linux χρησιμοποιώντας τα «απομονωμένα χαρακτηριστικά» του πυρήνα του Λ.Σ για να μπορούν οι containers να υλοποιούνται ανεξάρτητα μέσα σε ένα Linux instance αποφεύγοντας έτσι την επιβάρυνση για την έναρξη και τη διατήρηση που προκύπτει από τη χρήση ενός VM.

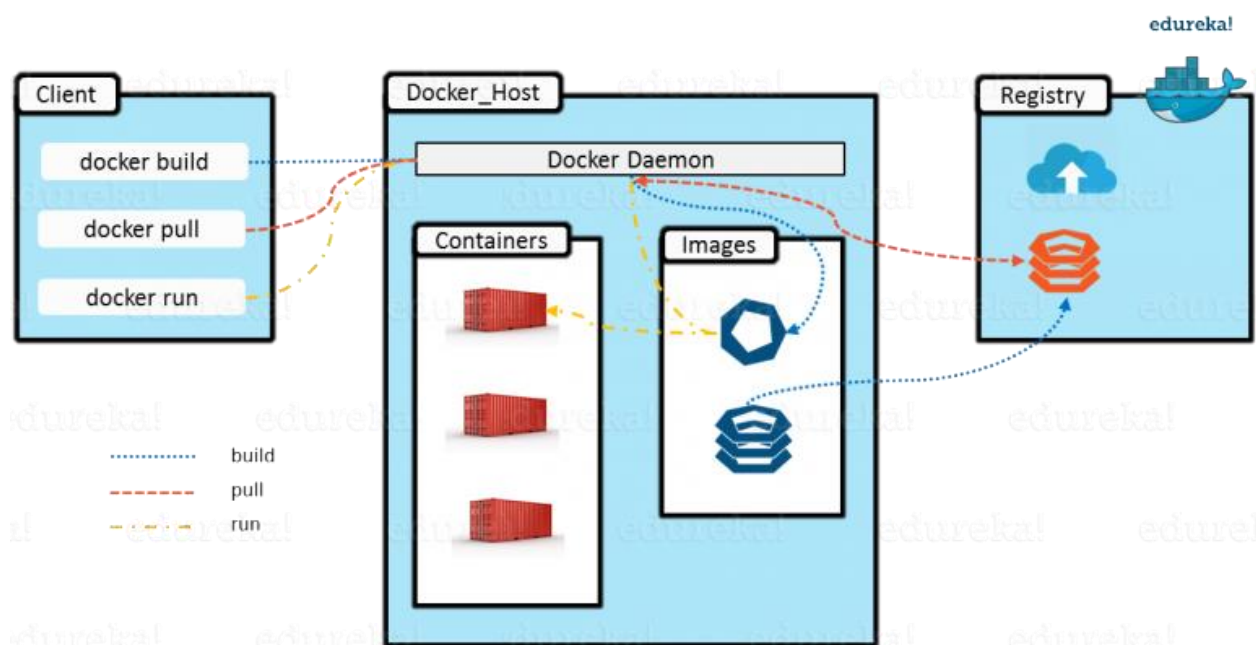
Το βασικό πλεονέκτημα ενός docker container είναι ότι δεν χρειάζεται όλους τους πόρους ενός Λ.Σ για να τρέξει. Βασίζεται στη λειτουργικότητα του πυρήνα και χρησιμοποιεί απομονωμένους πόρους για τη CPU, τη μνήμη ram, και διαφορετικά namespaces για να «απομονώσει» την εικόνα της εφαρμογής από το υπόλοιπο λειτουργικό σύστημα. Επειδή τα Docker containers είναι ελαφριές εφαρμογές ένα virtual machine ή ένας διακομιστής μπορεί να τρέξει πολλά docker containers ταυτόχρονα. Ένας docker μπορεί να συνεργαστεί με διάφορα δημοφιλή εργαλεία, όπως amazon services κ.α. Επίσης έχει αναπτυχθεί τα τελευταία χρόνια και σε windows servers.

Το λογισμικό του docker παρέχει διάφορα συστατικά και εργαλεία.

- «dockerd», είναι μια διαδικασία τύπου daemon (daemon process) για τη διαχείριση των docker containers, και docker objects. Η διαδικασία δέχεται κλήσεις από το API του

docker engine. Ο client του docker δίνει τη δυνατότητα επικοινωνίας μέσω CLI (Common language interface). Χρησιμοποιεί το docker REST API για να επικοινωνήσει με μια ή περισσότερες daemon διαδικασίες.

- **Objects:** Τα docker objects αναφέρονται σε διαφορετικές οντότητες που χρησιμοποιούνται για να μετατρέψουν μια εφαρμογή σε docker. Τα βασικά objects αποτελούνται από images, containers και services.
 - Ένας docker container είναι μια ένα τυποποιημένο, ενθυλακωμένο περιβάλλον που εκτελεί εφαρμογές. Η διαχείριση ενός container πραγματοποιείται μέσω του cli ή του docker API.
 - Ένα docker image είναι ένα πρότυπο που χρησιμοποιείται για την κατασκευή containers. Ενώ τα images χρησιμοποιούνται για την αποθήκευση και την πλοήγηση εφαρμογών.
 - Ένα docker service επιτρέπει στους containers να διαμοιράζονται μέσω πολλαπλών daemon διαδικασιών. Το αποτέλεσμα αυτό ονομάζεται swarm.
- **Registries:** Το docker registry είναι μια «αποθήκη» (repository) για εικόνες docker. Τα docker clients συνδέονται στα registries για να αποθηκεύσουν τα images ή να χρησιμοποιήσουν και να μεταφορτώσουν τα images που έχουν ήδη κατασκευαστεί. Τα registries μπορεί να είναι δημόσια ή ιδιωτικά. Δύο βασικά δημόσια registries είναι το Docker Hub και το Docker Cloud. Ενώ το Docker Hub είναι η default registry όπου γίνεται η αναζήτηση για τα images. Στο παρακάτω σχήμα [2.3] φαίνεται η αρχιτεκτονική που μόλις περιεγράφηκε



Σχήμα 2.4: Docker Architecture

Γνωστά εργαλεία για dockers είναι:

- **Docker Compose:** ένα εργαλείο για τον καθορισμό και για την έναρξη πολλαπλών container εφαρμογών. Χρησιμοποιεί **YAML** αρχεία για τη διαμόρφωση των υπηρεσιών

της εφαρμογής και για την εκτέλεση της διαδικασίας δημιουργίας και εκκίνησης όλων των containers με τη χρήση εντολής από ένα σετ εντολών.

- **Docker Swarm:** παρέχει clustering λειτουργία για τους Docker containers. Το οποίο μετατρέπει μια ομάδα από dockers σε έναν απλό docker engine.

3. Λειτουργικές προδιαγραφές

Στο παρόν κεφάλαιο περιγράφονται οι λειτουργικές προδιαγραφές της εφαρμογής και των εργαλείων που έχουν σχεδιαστεί και δημιουργηθεί για την διαχείριση των αισθητήρων και των τιμών.

3.1 Stakeholders και Actors

Τα «ενδιαφερόμενα μέρη» **Stakeholders** σε μια εφαρμογή είναι οι οντότητες που έχουν συνεισφέρει στην εφαρμογή με οποιονδήποτε τρόπο. Κάποια παραδείγματα που συνήθως έχουν τον ρόλο των ενδιαφερόμενων μερών είναι: τελικοί χρήστες, πελάτες, συνεργαζόμενες εταιρείες, προγραμματιστές κ.α. [10]

Οι «χρήστες της εφαρμογής» **actors** αποτελούν τις εξωτερικές οντότητες που αλληλεπιδρούν με τις εφαρμογές. Αντιπροσωπεύουν το ρόλο τον οποίο ο χρήστης μπορεί να διαδραματίσει στην εφαρμογή. Συγκεκριμένα οι ρόλοι αυτοί δεν είναι αναγκαίο να είναι φυσικά πρόσωπα, αλλά οτιδήποτε αλληλεπιδρά με την εφαρμογή και έχει ένα συγκεκριμένο ρόλο. Η σχέση actors και stakeholders δεν είναι αμφίδρομη, δηλαδή ένας actor είναι ταυτόχρονα stakeholder, ενώ το αντίστροφο δεν ισχύει. Στα use cases της εφαρμογής εμφανίζονται μόνο actors. Συγκεκριμένα στις εφαρμογές (web και Android) ο κύριος ρόλος είναι αυτός του αγρότη, ο οποίος ενημερώνεται σχετικά με τις μετρήσεις που έχουν συγκεντρώσει οι αισθητήρες, μπορεί να αλλάζει τις παραμέτρους του αλγορίθμου αυτόματης άρδευσης, τα προσωπικά του στοιχεία κ.α. Για τα παραπάνω σενάρια τα ενδιαφερόμενα μέρη του συστήματος είναι:

- Ο **γεωργός** ενός αγροκτήματος που χρησιμοποιεί την εφαρμογή για να ενημερωθεί σχετικά με τις συνθήκες θερμοκρασίας και υγρασίας που υπάρχουν στο χωράφι του για παρελθοντικές στιγμές και να μεταφορτώσει αναφορές σε πολλές μορφές (pdf, xls, csv). Ακόμα και σε πραγματικό χρόνο, εκτελώντας νέα μέτρηση από τους αισθητήρες των συσκευών, την έναρξη της άρδευσης του αγροκτήματος για οποιαδήποτε μελλοντικό χρονικό διάστημα, την αλλαγή των ωρών του διαστήματος, των κατωφλίων και ανωφλίων εκτέλεσης του αλγορίθμου αυτόματης υδροδότησης. Επίσης μπορεί να αλλάξει τα ονόματα των συσκευών αισθητήρων που έχουν τοποθετηθεί σε συγκεκριμένα μέρη του αγροκτήματος. Είναι σε θέση οποιαδήποτε στιγμή να ενημερωθεί για τις υπάρχουσες καιρικές συνθήκες της περιοχής (άνεμος, ατμοσφαιρική πίεση, ακριβής ώρα ξημερώματος κ.α.) σε διάστημα πρόγνωσης πέντε (5) ημερών.
- Οι **αισθητήρες** παρέχουν τις μετρήσεις ατμοσφαιρικής υγρασίας, υγρασίας εδάφους, θερμοκρασίας συσκευής και θερμοκρασίας ατμόσφαιρας για το αγρόκτημα.
- Ο **σταθμός** που εκτελεί τον αυτόματο αλγόριθμο υδροδότησης για συγκεκριμένο διάστημα ημερησίως και αποφασίζει αν θα ξεκινήσει ή όχι η άρδευση του αγροκτήματος. Επίσης είναι υπεύθυνος για τη διαχείριση των κλήσεων που δέχεται από την εφαρμογή καθώς και για την αποστολή των μετρήσεων μέσω του δικτύου 4G στο διακομιστή στα κατάλληλα endpoints.
- Η **υπηρεσία πρόγνωσης καιρού** λειτουργεί μέσα στην εφαρμογή στέλνοντας δεδομένα παίρνοντας σαν παραμέτρους την γεωγραφική θέση σε μορφή τετμημένης, τεταγμένης της θέσης του σταθμού του αγροκτήματος.

3.2 Τεχνικές προδιαγραφές

Το κομμάτι αυτό εξηγεί τις τεχνικές προδιαγραφές του μοντέλου της βάσης καθώς και των δύο εφαρμογών (web & Android) και τελικά των δυο υπηρεσιών backend που έχουν δημιουργηθεί για την εξυπηρέτηση των δύο εφαρμογών.

3.2.1 Μοντέλο Βάσης

Στο μοντέλο της βάσης διακρίνονται κάποιες βασικές οντότητες που διαδραματίζουν σημαντικό ρόλο στις υπηρεσίες του backend και στις εφαρμογές μας.

- **Feature of Interest (Πεδίο ενδιαφέροντος):** Είναι η οντότητα στην οποία αποθηκεύονται οι πληροφορίες για τα φαινόμενα των μετρήσεων καθώς και της άρδευσης. Ένα πεδίο ενδιαφέροντος μπορεί να είναι τριών (3) διαφορετικών τύπων: καλλιέργεια, σταθμός ή μια τελική συσκευή δηλαδή ένα σύστημα αισθητήρων. Εκεί αποθηκεύονται οι χρόνοι ποτίσματος για κάθε τελική συσκευή καθώς και η συνθήκη (αληθής - ψευδής) για την πραγματοποίηση νέας μέτρησης ή νέας άρδευσης του αγροκτήματος. Ο χρήστης της εφαρμογής μπορεί να αλλάξει τις τιμές του αυτόματου αλγορίθμου υδροδότησης καθώς και να ορίσει το διάστημα της άρδευσης.
- **Procedure:** Τα ονόματα των αισθητήρων καθώς και τις επίσημες ονομασίες τους
- **Observable Property (παρατηρήσιμο μέγεθος):** Το είδος της μέτρησης που πραγματοποιείται στο σημείο ενδιαφέροντος π.χ. Θερμοκρασία, υγρασία κ.α. Ένας χρήστης μπορεί να διαβάσει τις τιμές αυτές ενώ ο διαχειριστής της εφαρμογής μπορεί να εισάγει νέες τιμές ή να αλλάξει την περιγραφή και την ονομασία των ήδη υπαρχουσών.
- **Serie:** Είναι η οντότητα στην οποία αποθηκεύεται ο συνδυασμός πεδίου ενδιαφέροντος, αισθητήρα και είδους μέτρησης. Μπορεί να δημιουργηθεί και να μεταβληθεί μόνο από τους διαχειριστές του συστήματος, ο απλός χρήστης μπορεί απλά να παρατηρήσει τις τιμές των δεδομένων.
- **Observable (Μέτρηση):** Εκεί αποθηκεύονται οι μετρήσεις που καταφθάνουν από τους αισθητήρες, αποθηκεύεται η αντίστοιχη σειρά (Serie), η μονάδα μέτρησης, το διάστημα ή η στιγμή που αυτή πραγματοποιείται καθώς και η στιγμή που αποθηκεύεται στη Β.Δ του διακομιστή. Ένα μοναδικό αναγνωριστικό που αντιστοιχεί στη στιγμή αποθήκευσης της μέτρησης σε συνδυασμό με ένα κλειδί GUID. Για τις μετρήσεις που δεν αφορούν διάστημα αλλά αποτελούν στιγμιαίο φαινόμενο οι χρόνοι του διαστήματος (από, και έως) συμπίπτουν. Ο χρήστης μπορεί να διαβάσει σε μορφή πινάκων διανυσμάτων καθώς και να «μεταφορτώσει» τις μετρήσεις αυτές σε μορφή αναφορών στους τύπους αρχείων pdf, excel και plain text.
- **User:** Ο χρήστης όπου περιέχει τα στοιχεία που χρειάζεται κάποιος για να γίνει διαπιστευμένος χρήστης συστήματος, δηλαδή να πραγματοποιήσει επιτυχημένα είσοδο στο σύστημα.

3.2.2 Backend (Web & Android)

Οι απαιτήσεις που έχουν οριστεί για τις δύο εφαρμογές (web και Android) εξυπηρετούν τις απαιτήσεις της κάθε εφαρμογής. Συγκεκριμένα για την web εφαρμογή οι υπηρεσίες που δημιουργήθηκαν χωρίζονται σε αυτές που υποστηρίζουν την επικοινωνία με τους σταθμούς, δηλαδή τα ενσωματωμένα συστήματα που βρίσκονται σε κάθε καλλιέργεια και σε αυτές που χρησιμοποιούνται για την ίδια την εφαρμογή. Αναλυτικότερα:

- Υποστηρίζει τη μεταφόρτωση των μετρήσεων των αισθητήρων.

- Υποστηρίζει τη μεταφόρτωση δεδομένων ικανά να αλληλεπιδράσουν με το ενσωματωμένο σύστημα της αυτόματης άρδευσης.
- Για το κατέβασμα των χαρακτηριστικών του κάθε σταθμού, όπως οι χρόνοι του αυτόματου αλγορίθμου υδροδότησης και τις διευθύνσεις του ZigBee για κάθε συσκευή.
- Κατά τη βασική λειτουργία του ενσωματωμένου συστήματος και εκτός διαστήματος λειτουργίας του αλγορίθμου, επικοινωνεί μέσω των υπηρεσιών με τη βάση δεδομένων του διακομιστή για να πραγματοποιήσει μεταφόρτωση δεδομένων που έχει αλλάξει ο χρήστης για κάθε μια συσκευή αισθητήρων.

Η βασική εφαρμογή:

- Είναι σύμφωνη με πρότυπα σχεδίασης.
- Δημιουργεί restful web services για να μπορούν κι άλλες εφαρμογές να τα χρησιμοποιήσουν χρησιμοποιώντας το απαραίτητο authentication.
- Υποστηρίζει τη δημιουργία και διαμοιρασμό σε αρχεία αναφορών για τα δεδομένα της άρδευσης και των αισθητήρων.
- Υποστηρίζει την επεκτασιμότητα δεδομένων.
- Διαμοιράζει τα δεδομένα για χρήση σε βιβλιοθήκες γραφημάτων.
- Υποστηρίζει τη σειριοποίηση και αποσειριοποίηση δεδομένων.
- Υποστηρίζει τη χρήση σύνθετων δεδομένων π.χ. γεωμετρίες.
- Παρέχει authentication μέσω της βάσης δεδομένων μόνο σε διαπιστευμένους χρήστες.

3.2.3 Web Application

Η web εφαρμογή και οι υπηρεσίες που σερβίρει αποτελεί το βασικό στοιχείο του συστήματος. Πιο συγκεκριμένα οι υπηρεσίες που έχουν παραχθεί είναι υπεύθυνες για την επικοινωνία με το ενσωματωμένο σύστημα καθώς και με τη web εφαρμογή. Η εφαρμογή επιλέχθηκε να παρέχει ένα εύχρηστο και απλό περιβάλλον χρήστη για να είναι σε θέση ο καθένας να δει τις μετρήσεις των αγροκτημάτων του καθώς και να πραγματοποιήσει ενέργειες. Για την δημιουργία της εφαρμογής επιλέχθηκε η χρήση εργαλείων όπως angularJS, bootstrap CSS framework καθώς και άλλων βιβλιοθηκών από το διαδίκτυο για τη θέαση δεδομένων σε μορφή γραφημάτων, σε μορφή πινάκων και τη χρήση χαρτών.

- Είναι σύμφωνη με πρότυπα σχεδίασης.
- Να έχει αποκρινόμενο περιβάλλον χρήστη για χρήση σε διαφορετικού μεγέθους οθόνες και συσκευές.
- Να μπορεί να χρησιμοποιηθεί από απλούς χρήστες με ελάχιστη εμπειρία, καθώς το βασικό κοινό απεύθυνσης είναι οι αγρότες.
- Υποστηρίζει την εξατομίκευση για κάθε χρήστη έτσι ώστε να μπορεί να διαμορφώνει το περιβάλλον και τα δεδομένα σε όποια μορφή θέλει.
- Υποστηρίζει την παραγωγή γραφημάτων με σκοπό την «οπτικοποίηση» των μετρήσεων άρδευσης και των παρατηρήσιμων μεγεθών των αισθητήρων.
- Παρέχει τη δυνατότητα αποθήκευσης και αλλαγής της θέσης της καλλιέργειας για τη θέαση του και την παροχή πληροφοριών από εξωτερικές πηγές για την πρόγνωση καιρού σύμφωνα με τη γεωγραφική αυτή θέση.
- Παρέχει πληροφορίες για πρόγνωση του καιρού σε διάστημα πέντε ημερών.
- Υποστηρίζει την αλλαγή των δεδομένων σε πραγματικό χρόνο στο επίπεδο του ενσωματωμένου συστήματος μέσω της επικοινωνίας με το σταθμό της κάθε καλλιέργειας.

- Υποστήριξη προώθησης κοινοποιήσεων για τα γεγονότα που λαμβάνουν χώρα στη καλλιέργεια.
- Δυνατότητα θέασης και επεξεργασίας των προσωπικών στοιχείων του χρήστη.

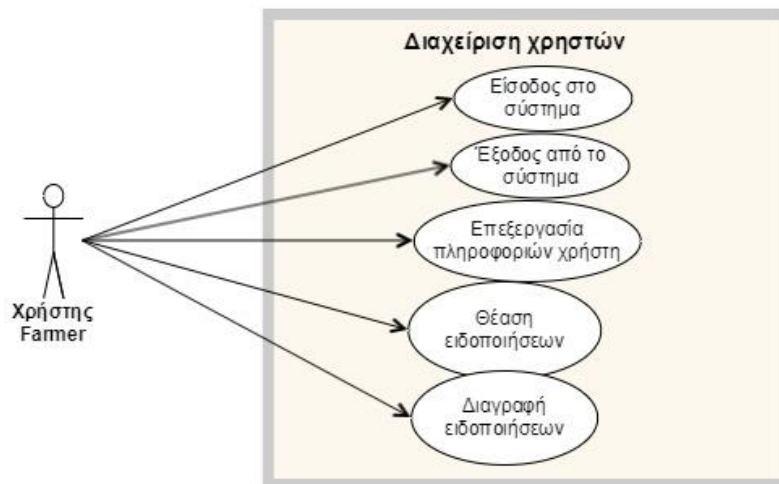
3.2.4 Android Application

Η Android εφαρμογή δημιουργήθηκε για να παρέχει ακόμα ένα τρόπο θέασης και επεξεργασίας των δεδομένων της εφαρμογής. Εφόσον θα μπορεί να κάνει είσοδο της εφαρμογής σε απομακρυσμένες περιοχές χωρίς να υπάρχει η ανάγκη ύπαρξης προσωπικού υπολογιστή. Πιο συγκεκριμένα:

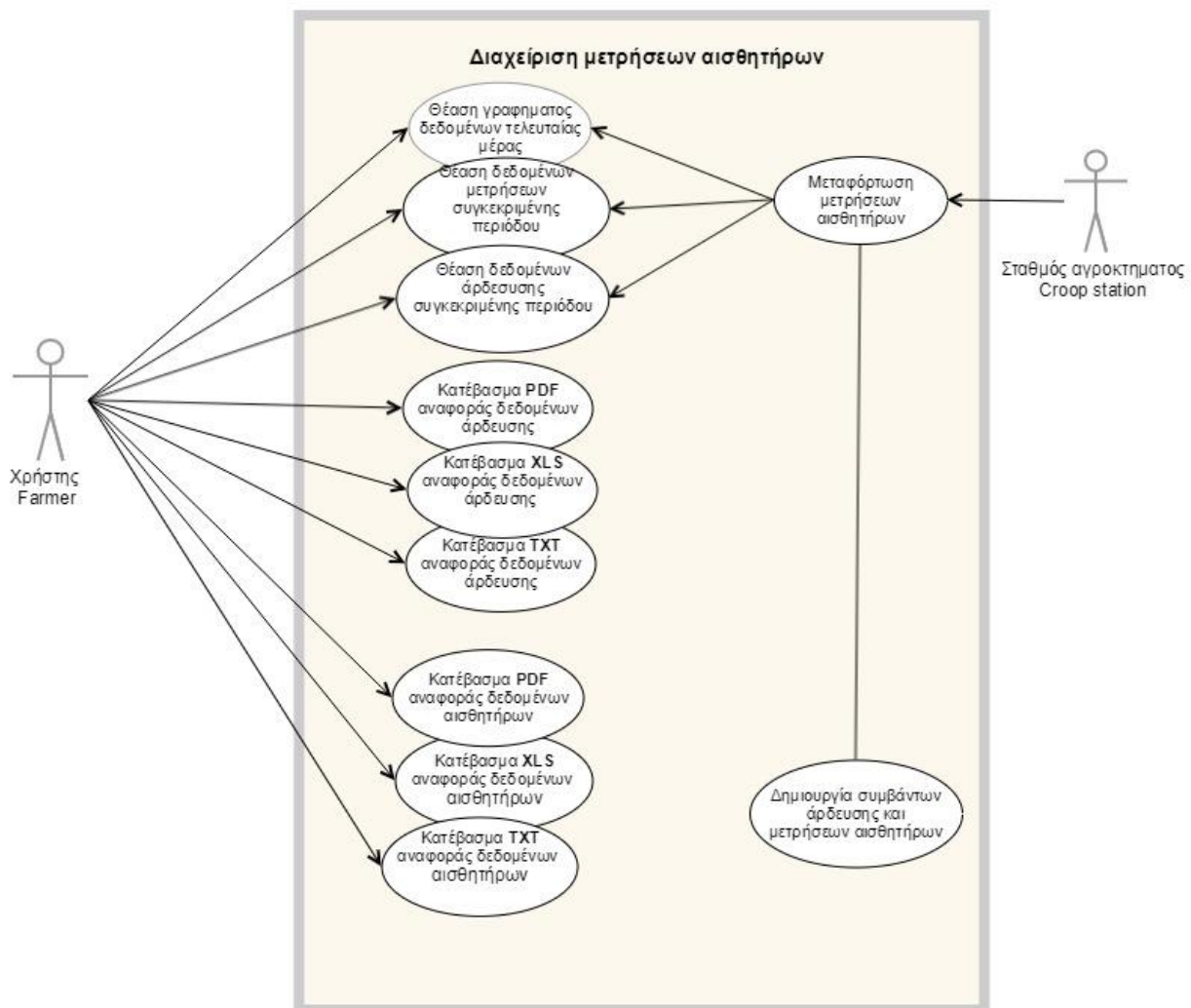
- Υποστηρίζει την είσοδο διαπιστευμένων χρηστών.
- Παρέχει ένα όμορφο και εύχρηστο περιβάλλον χρήστη.
- Θέαση τιμών σε μορφή γραφημάτων και σε μορφή λιστών.
- Δημιουργία νέων συμβάντων για την έναρξη της άρδευσης και της λήψης τιμών από τους αισθητήρες σε πραγματικό χρόνο.
- Δυνατότητα θέασης προσωπικών στοιχείων του χρήστη.

3.3 Use cases

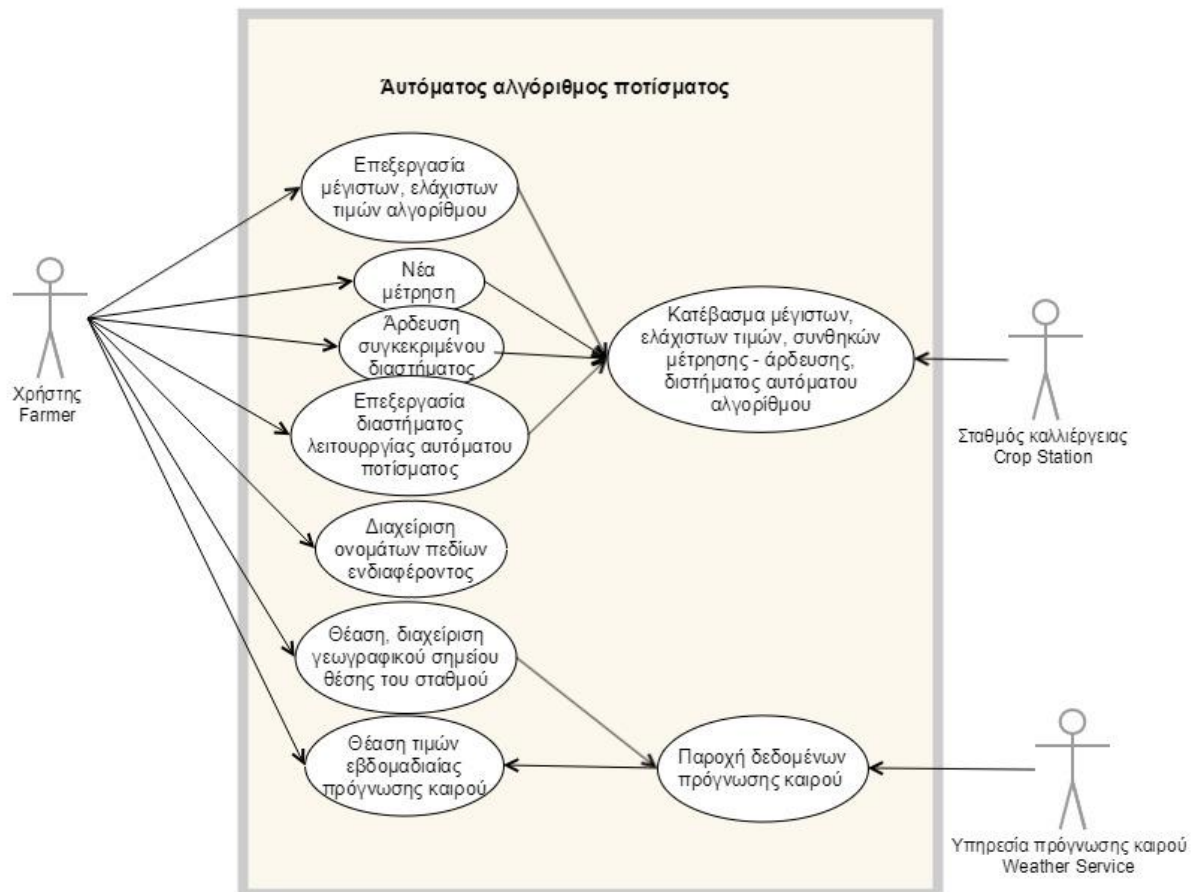
Τα use cases περιγράφουν τη συμπεριφορά του συστήματος που σχεδιάστηκε ανταποκρινόμενο σε κάποια κλήση από τους ενδιαφερόμενους, οι ρόλοι των οποίων παρουσιάστηκαν παραπάνω. Στο κεφάλαιο αυτό περιγράφεται η συμπεριφορά των εφαρμογών κάτω από ορισμένες συνθήκες και τα ερεθίσματα που δέχονται είτε από το χρήστη των εφαρμογών είτε από άλλους παράγοντες όπως είναι η επικοινωνία με το σταθμό της καλλιέργειας. Τα σχήματα Σχήμα 3.1, Σχήμα 3.2, Σχήμα 3.3 καθώς και οι πίνακες που ακολουθούν περιγράφουν τα use cases της web εφαρμογής. Συγκεκριμένα στο Σχήμα 3.1 παρουσιάζονται οι δυνατότητες παραμετροποίησης του προφίλ του κάθε χρήστη. Το Σχήμα 3.2 παρουσιάζει τις επιλογές θέασης δεδομένων που προέρχονται από τους αισθητήρες των συσκευών. Ενώ το Σχήμα 3.3 παρουσιάζει τις ενέργειες που έχει στη διάθεση του ο χρήστης για να αλληλεπιδράσει με το σταθμό της καλλιέργειας για τη ρύθμιση των παραμέτρων του αλγορίθμου αυτόματης άρδευσης.



Σχήμα 3.1: Διαχείριση λειτουργιών προφίλ χρήστη



Σχήμα 3.2: Διαχείριση μετρήσεων αισθητήρων.



Σχήμα 3.3: Αυτόματος Αλγόριθμος άρδευσης

Πίνακας 3.1: Use Case 1: Είσοδος στην εφαρμογή

Πλαίσιο χρήσης	Ο χρήστης επιθυμεί να κάνει είσοδο στην web εφαρμογή για να χρησιμοποιήσει την λειτουργικότητα της.	
Πεδίο εφαρμογής	Web εφαρμογή (Agric Cloud)	
Επίπεδο	Δευτερεύουσα λειτουργία	
Προαπαιτούμενα	Να υπάρχει ο χρήστης στη βάση δεδομένων	
Συνθήκες επιτυχίας	Ο χρήστης κάνει είσοδο στην εφαρμογή.	
Συνθήκες αποτυχίας	Ο χρήστης δεν μπορεί να κάνει εισαγωγή στην εφαρμογή.	
Πρωτεύον, Δευτερεύον Actors	User (αγρότης) Web εφαρμογή (Agric Cloud)	
Trigger (Εναυσμα)		
Περιγραφή	Βήμα	Ενέργεια
	1	Ο χρήστης βλέπει την οθόνη εισαγωγής.
	2	Ο χρήστης εισάγει τα διαπιστευτήρια του στα αντίστοιχα πεδία (όνομα, κωδικός).

	3	Η εφαρμογή επικυρώνει τα στοιχεία από τις υποβαλλόμενες τιμές, ελέγχει αν τα διαπιστευτήρια χρήστη είναι σωστά.
	4	Η εφαρμογή απεικονίζει την πρώτη οθόνη στον χρήστη.
Επεκτάσεις	Βήμα	Ενέργεια
	1	Οι υποβαλλόμενες τιμές είναι λανθασμένες.
	2	Η εφαρμογή παραμένει στην οθόνη εισαγωγής και απεικονίζει τα λάθη του χρήστη.

<i>Πίνακας 3.2: Use Case 2: Έξοδος από την εφαρμογή</i>		
Πλαίσιο χρήσης	Ο χρήστης επιθυμεί να κάνει έξοδο από την web εφαρμογή.	
Πεδίο εφαρμογής	Web Application, Android App	
Επίπεδο	Υπολειτουργία	
Προαπαιτούμενα	Ο χρήστης πρέπει να είναι διαπιστευμένος και να έχει κάνει είσοδο στην εφαρμογή	
Συνθήκες επιτυχίας	Ο χρήστης κάνει έξοδο από την εφαρμογή.	
Συνθήκες αποτυχίας	Ο χρήστης δεν μπορεί να αποσυνδεθεί από την εφαρμογή.	
Πρωτεύον, Δευτερεύον Actors	User (αγρότης) Web εφαρμογή (Agric Cloud, Agric App)	
Trigger (Εναυσμα)	Εμφανίζεται ένα αναδυόμενο παράθυρο, για να επιβεβαιώσει ο χρήστης την έξοδο του από την εφαρμογή.	
Περιγραφή	Βήμα	Ενέργεια
	1	Ο χρήστης επιλέγει αποσύνδεση στο αναδυόμενο παράθυρο για να επιβεβαιώσει την έξοδο του από την εφαρμογή.
	2	Η εφαρμογή αποσυνδέει το χρήστη.
	3	Ο χρήστης μεταφέρεται στην αρχική οθόνη σύνδεσης
Επεκτάσεις	Βήμα	Ενέργεια
	1	Ο χρήστης επιλέγει άκυρο στο αναδυόμενο παράθυρο κι έτσι η εφαρμογή δεν τον αποσυνδέει.

<i>Πίνακας 3.3: Use Case 3: Επεξεργασία πληροφοριών χρήστη</i>		
Πλαίσιο χρήσης	Ο χρήστης επιθυμεί να επεξεργαστεί τα προσωπικά του στοιχεία στην εφαρμογή.	
Πεδίο εφαρμογής	Web Application, Android App	
Επίπεδο	Υπολειτουργία	
Προαπαιτούμενα	Ο χρήστης πρέπει να έχει κάνει είσοδο στην εφαρμογή, και η εφαρμογή να επικυρώσει τα στοιχεία στη φόρμα επεξεργασίας.	
Συνθήκες επιτυχίας	Ο χρήστης επεξεργάζεται τα στοιχεία του και προχωράει στην αποθήκευση. Ένα μήνυμα επιτυχίας σε μορφή toaster εμφανίζεται στο χρήστη.	

Συνθήκες αποτυχίας	Ο χρήστης δεν μπορεί να αποθηκεύσει τα στοιχεία που έχει επεξεργαστεί. Ένα μήνυμα αποτυχίας σε μορφή toaster εμφανίζεται στο χρήστη.	
Πρωτεύον, Δευτερεύον Actors	User (αγρότης) Web εφαρμογή (Agric Cloud)	
Trigger (Εναυσμα)	Η εφαρμογή επικυρώνει αυτόματα τα στοιχεία του χρήστη.	
Περιγραφή	Βήμα	Ενέργεια
	1	Ο χρήστης επιλέγει την είσοδο του στη φόρμα επικοινωνίας από το μενού της εφαρμογής.
	2	Ο χρήστης επεξεργάζεται τα στοιχεία του .
	3	Η εφαρμογή επικυρώνει τα στοιχεία που έχει επεξεργαστεί ο χρήστης.
	4	Ο χρήστης επιλέγει αποθήκευση.
	5	Ένα παράθυρο επιτυχίας εμφανίζεται στο χρήστη.
Επεκτάσεις	Βήμα	Ενέργεια
	1	Ο χρήστης επιλέγει εκκαθάριση, η εφαρμογή σβήνει όλες τις τιμές.

Πίνακας 3.4: Use Case 4: Θέαση ειδοποιήσεων

Πλαίσιο χρήσης	Ο χρήστης επιθυμεί να δει τις ειδοποιήσεις που παράγονται αυτόματα από την εφαρμογή.	
Πεδίο εφαρμογής	Web Application	
Επίπεδο	Υπολειτουργία	
Προαπαιτούμενα	Ο χρήστης πρέπει να έχει κάνει είσοδο στην εφαρμογή.	
Συνθήκες επιτυχίας	Ο χρήστης βλέπει σε μορφή λίστας τις ειδοποιήσεις που παράγονται από την εφαρμογή.	
Συνθήκες αποτυχίας	Ο χρήστης δεν μπορεί να αποθηκεύσει τα στοιχεία που έχει επεξεργαστεί. Ένα μήνυμα αποτυχίας σε μορφή toaster εμφανίζεται στο χρήστη.	
Πρωτεύον, Δευτερεύον Actors	User (αγρότης) Web εφαρμογή (Agric Cloud)	
Trigger (Εναυσμα)		
Περιγραφή	Βήμα	Ενέργεια
	1	Ο χρήστης επιλέγει τη θέαση των αυτόματων ειδοποιήσεων σε μορφή λίστας.
	2	Ο χρήστης βλέπει τις ειδοποιήσεις ανά αύξουσα ημερομηνία.
Επεκτάσεις	Βήμα	Ενέργεια

Πίνακας 3.5: Use Case 5: Διαγραφή ειδοποιήσεων

Πλαίσιο χρήσης	Ο χρήστης επιθυμεί να διαγράψει τις ειδοποιήσεις που παράγονται αυτόματα από την εφαρμογή.
-----------------------	--

Πεδίο εφαρμογής	Web application	
Επίπεδο	Υπολειτουργία	
Προαπαιτούμενα	Ο χρήστης πρέπει να έχει κάνει είσοδο στην εφαρμογή και να υπάρχουν ειδοποιήσεις.	
Συνθήκες επιτυχίας	Ο χρήστης διαγράφει μια ή όλες τις ειδοποιήσεις.	
Συνθήκες αποτυχίας	Ο χρήστης αποτυγχάνει να σβήσει μία ή παραπάνω ειδοποιήσεις	
Πρωτεύον, Δευτερεύον Actors	User (αγρότης) Web εφαρμογή (Agric Cloud)	
Trigger (Εναυσμα)		
Περιγραφή	Βήμα	Ενέργεια
	1	Ο χρήστης επιλέγει τη θέαση των αυτόματων ειδοποιήσεων σε μορφή λίστας.
	2	Ο χρήστης επιλέγει να διαγράψει μια ειδοποίησή.
	3	Ένα μήνυμα επιτυχίας εμφανίζεται στην οθόνη του χρήστη.
Επεκτάσεις	Βήμα	Ενέργεια
	1	Ο χρήστης επιλέγει τη διαγραφή όλων των ειδοποιήσεων.
	2	Ένα μήνυμα επιτυχίας εμφανίζεται στην οθόνη του χρήστη.

Πίνακας 3.6: Use case 6: Θέαση γραφήματος δεδομένων τελευταίας μέρας

Πλαίσιο χρήσης	Ο χρήστης επιλέγει να δει συγκεντρωμένες τις μετρήσεις από όλους τους αισθητήρες σε διάστημα μιας μέρας στην πρώτη οθόνη της εφαρμογής	
Πεδίο εφαρμογής	Web application	
Επίπεδο	Υπολειτουργία	
Προαπαιτούμενα	Ο χρήστης πρέπει να έχει κάνει είσοδο στην εφαρμογή και να υπάρχουν δεδομένα στη βάση δεδομένων.	
Συνθήκες επιτυχίας	Ο χρήστης βλέπει τις μετρήσεις της τελευταίας μέρας σε μορφή διαγράμματος.	
Συνθήκες αποτυχίας	Ο χρήστης βλέπει ένα άδειο διάγραμμα με τη σημείωση «NO DATA»	
Πρωτεύον, Δευτερεύον Actors	User (αγρότης) Web εφαρμογή (Agric Cloud)	
Trigger (Εναυσμα)		
Περιγραφή	Βήμα	Ενέργεια
	1	Ο χρήστης επιλέγει να ανακατευθυνθεί στην βασική οθόνη της εφαρμογής.
	2	Ο χρήστης επιλέγει μέσω της λεζάντας του γραφήματος ποιο τύπο παρατήρησης θέλει να δει.
Επεκτάσεις	Βήμα	Ενέργεια

Πίνακας 3.7: Use case 7: Θέαση δεδομένων τύπου μέτρησης συγκεκριμένης περιόδου		
Πλαίσιο χρήσης	Ο χρήστης επιλέγει να δει τα δεδομένα μετρήσεων ανάλογα με τον τύπο μέτρησης, τη συσκευή αισθητήρων και το χρονικό διάστημα	
Πεδίο εφαρμογής	Web application	
Επίπεδο	Υπολειτουργία	
Προαπαιτούμενα	Ο χρήστης πρέπει να έχει κάνει είσοδο στην εφαρμογή και να υπάρχουν δεδομένα στη βάση δεδομένων.	
Συνθήκες επιτυχίας	Ο χρήστης βλέπει τις μετρήσεις του τύπου μέτρησης, της συσκευής αισθητήρων και του διαστήματος που έχει επιλέξει σε μορφή διαγράμματος και πίνακα.	
Συνθήκες αποτυχίας	Ο χρήστης βλέπει ένα άδειο διάγραμμα και έναν άδειο πίνακα με τη σημείωση «NO DATA»	
Πρωτεύον, Δευτερεύον Actors	User (αγρότης) Web εφαρμογή (Agric Cloud)	
Trigger (Εναυσμα)		
Περιγραφή	Βήμα	Ενέργεια
	1	Ο χρήστης επιλέγει να ανακατευθυνθεί στην οθόνη θέασης δεδομένων ανάλογα με το είδος παρατήρησης που θέλει. Συγκεκριμένα μπορεί να δει δεδομένα από: θερμοκρασία περιβάλλοντος, θερμοκρασία συσκευής, υγρασία περιβάλλοντος, υγρασία εδάφους
	2	Ο χρήστης επιλέγει μια συσκευή αισθητήρων στον οποίο θέλει να πάρει μετρήσεις.
	3	Ο χρήστης επιλέγει μια ημερομηνία «από» η οποία πρέπει να είναι παρελθοντική
	4	Ο χρήστης επιλέγει μια ημερομηνία «έως» η οποία πρέπει να είναι παρελθοντική και μεγαλύτερη από την ημερομηνία έως.
	5	Ο χρήστης βλέπει τις τιμές των μετρήσεων σε μορφή διαγράμματος καθώς και σε μορφή πίνακα
Επεκτάσεις	Βήμα	Ενέργεια
	1	Ο χρήστης πρέπει να επιλέξει στα δεδομένα σε μορφή πίνακα το πλήθος των δεδομένων ανά σελίδα.
	2	Ο χρήστης είναι σε θέση να ανακατευθυνθεί στις καρτέλες των δεδομένων του πίνακα.
	3	Ο χρήστης μπορεί κάνοντας hover στο διάγραμμα να δει τη συγκεκριμένη χρονική στιγμή και τη μέτρηση τη στιγμή αυτή.

Πίνακας 3.8: Use case 8: Κατέβασμα αναφοράς δεδομένων τύπου μέτρησης συγκεκριμένης περιόδου

Πλαίσιο χρήσης	Ο χρήστης επιλέγει να κατεβάσει και να δει τα δεδομένα μετρήσεων ανάλογα με τον τύπο μέτρησης, τη συσκευή αισθητήρων και το χρονικό διάστημα.	
Πεδίο εφαρμογής	Web application	
Επίπεδο	Υπολειτουργία	
Προαπαιτούμενα	Ο χρήστης πρέπει να έχει κάνει είσοδο στην εφαρμογή, να έχει επιλέξει συσκευή αισθητήρων καθώς και σωστό διάστημα	
Συνθήκες επιτυχίας	Ο χρήστης κατεβάζει και ανοίγει μια αναφορά με τις τιμές των μετρήσεων που φαίνονται στο διάγραμμα και στον πίνακα σε μορφές (pds, xls, csv)	
Συνθήκες αποτυχίας	Ο χρήστης αποτυγχάνει να κατεβάσει την αναφορά και ένα αναδυόμενο μήνυμα αποτυχίας πρέπει να εμφανιστεί.	
Πρωτεύον, Δευτερεύον	User (αγρότης) Web εφαρμογή (Agric Cloud)	
Trigger (Εναυσμα)	Το σύστημα κατασκευάζει την αναφορά και τη σερβίρει στο χρήστη για κατέβασμα.	
Περιγραφή	Βήμα	Ενέργεια
	1	Ο χρήστης στην οθόνη θέασης δεδομένων επιλέγει τον τύπο αρχείου της αναφοράς.
	2	Ο χρήστης επιλέγει μεταξύ των τύπων αρχείων (PDF, XLS, CSV) για κατέβασμα
	3	Ο χρήστης επιλέγει να δει το αρχείο που έχει κατεβάσει
Επεκτάσεις	Βήμα	Ενέργεια
	1	Στο αρχείο αναφοράς σε μορφή PDF ο χρήστης μπορεί να δει μεταξύ των τιμών σε μορφή πίνακα τη μέγιστη, ελάχιστη καθώς και τη μέση τιμή
	2	Ο χρήστης είναι σε θέση να ανακατευθυνθεί στις καρτέλες των δεδομένων του πίνακα.
	3	Ο χρήστης μπορεί κάνοντας hover στο διάγραμμα να δει τη συγκεκριμένη χρονική στιγμή και τη μέτρηση τη στιγμή αυτή.

Πίνακας 3.9: Use case 9: Θέαση δεδομένων άρδευσης συγκεκριμένης περιόδου

Πλαίσιο χρήσης	Ο χρήστης επιλέγει να δει τα δεδομένα άρδευσης μετρήσεων ανάλογα τη συσκευή αισθητήρων και το χρονικό διάστημα
Πεδίο εφαρμογής	Web application
Επίπεδο	Υπολειτουργία
Προαπαιτούμενα	Ο χρήστης πρέπει να έχει κάνει είσοδο στην εφαρμογή και να υπάρχουν δεδομένα άρδευσης στη βάση δεδομένων.
Συνθήκες επιτυχίας	Ο χρήστης βλέπει τις μετρήσεις ανάλογα της συσκευής αισθητήρων και του διαστήματος που έχει επιλέξει σε μορφή διαγράμματος και πίνακα.

Συνθήκες αποτυχίας	Ο χρήστης βλέπει ένα άδειο διάγραμμα και έναν άδειο πίνακα με τη σημείωση «NO DATA»	
Πρωτεύον, Δευτερεύον	User (αγρότης) Web εφαρμογή (Agric Cloud)	
Trigger (Εναυσμα)	Ελέγχονται οι ημερομηνίες διαστήματος από-έως,	
Περιγραφή	Βήμα	Ενέργεια
	1	Ο χρήστης επιλέγει να ανακατευθυνθεί στην οθόνη θέασης δεδομένων άρδευσης.
	2	Ο χρήστης επιλέγει μια συσκευή αισθητήρων στον οποίο θέλει να πάρει μετρήσεις.
	3	Ο χρήστης επιλέγει μια ημερομηνία «από» η οποία πρέπει να είναι παρελθοντική.
	4	Ο χρήστης επιλέγει μια ημερομηνία «έως» η οποία πρέπει να είναι παρελθοντική και μεγαλύτερη από την ημερομηνία έως.
	5	Ο χρήστης βλέπει τις τιμές των μετρήσεων σε μορφή διαγράμματος καθώς και σε μορφή πίνακα
Επεκτάσεις	Βήμα	Ενέργεια
	1	Ο χρήστης πρέπει να επιλέξει στα δεδομένα σε μορφή πίνακα το πλήθος των δεδομένων ανά σελίδα.
	2	Ο χρήστης είναι σε θέση να ανακατευθυνθεί στις καρτέλες των δεδομένων του πίνακα.
	3	Ο χρήστης μπορεί κάνοντας hover στο διάγραμμα να δει τη συγκεκριμένη χρονική στιγμή και τη μέτρηση τη στιγμή αυτή.

Πίνακας 3.10: Use case 10: Κατέβασμα αναφοράς δεδομένων άρδευσης συγκεκριμένης περιόδου

Πλαίσιο χρήσης	Ο χρήστης επιλέγει να κατεβάσει και να δει τα δεδομένα μετρήσεων άρδευσης ανάλογα με τη συσκευή αισθητήρων και το χρονικό διάστημα.
Πεδίο εφαρμογής	Web application
Επίπεδο	Υπολειτουργία
Προαπαιτούμενα	Ο χρήστης πρέπει να έχει κάνει είσοδο στην εφαρμογή, να έχει επιλέξει συσκευή αισθητήρων καθώς και σωστό διάστημα
Συνθήκες επιτυχίας	Ο χρήστης κατεβάζει και ανοίγει μια αναφορά με τις τιμές των μετρήσεων που φαίνονται στο διάγραμμα και στον πίνακα σε μορφές (pdf, xls, csv)
Συνθήκες αποτυχίας	Ο χρήστης αποτυγχάνει να κατεβάσει την αναφορά και ένα αναδυόμενο μήνυμα αποτυχίας πρέπει να εμφανιστεί.

Πρωτεύον, Δευτερεύον	User (αγρότης) Web εφαρμογή (Agric Cloud)	
Trigger (Εναυσμα)	Το σύστημα κατασκευάζει την αναφορά και τη σερβίρει στο χρήστη για κατέβασμα.	
Περιγραφή	Βήμα	Ενέργεια
	1	Ο χρήστης στην οθόνη θέασης δεδομένων άρδευσης επιλέγει τον τύπο αρχείου της αναφοράς.
	2	Ο χρήστης επιλέγει μεταξύ των τύπων αρχείων (PDF, XLS, CSV) για κατέβασμα
	3	Ο χρήστης επιλέγει να δει το αρχείο που έχει κατεβάσει
Επεκτάσεις	Βήμα	Ενέργεια
	1	Στο αρχείο αναφοράς σε μορφή PDF ο χρήστης μπορεί να δει μεταξύ των τιμών σε μορφή πίνακα τη μέγιστη, ελάχιστη καθώς και τη μέση τιμή της κατανάλωσης σε λίτρα.
	2	Ο χρήστης είναι σε θέση να ανακατευθυνθεί στις καρτέλες των δεδομένων του πίνακα.
	3	Ο χρήστης μπορεί κάνοντας hover στο διάγραμμα να δει τη συγκεκριμένη χρονική στιγμή και τη μέτρηση τη στιγμή αυτή.

Πίνακας 3.21: Use case 11: Κατέβασμα αναφοράς δεδομένων άρδευσης συγκεκριμένης περιόδου

Πλαίσιο χρήσης	Ο χρήστης επιλέγει να κατεβάσει και να δει τα δεδομένα μετρήσεων άρδευσης ανάλογα με τη συσκευή αισθητήρων και το χρονικό διάστημα.	
Πεδίο εφαρμογής	Web application	
Επίπεδο	Υπολειτουργία	
Προαπαιτούμενα	Ο χρήστης πρέπει να έχει κάνει είσοδο στην εφαρμογή, να έχει επιλέξει συσκευή αισθητήρων καθώς και σωστό διάστημα	
Συνθήκες επιτυχίας	Ο χρήστης κατεβάζει και ανοίγει μια αναφορά με τις τιμές των μετρήσεων που φαίνονται στο διάγραμμα και στον πίνακα σε μορφές (pdf, xls, csv)	
Συνθήκες αποτυχίας	Ο χρήστης αποτυγχάνει να κατεβάσει την αναφορά και ένα αναδυόμενο μήνυμα αποτυχίας πρέπει να εμφανιστεί.	
Πρωτεύον, Δευτερεύον Actors	User (αγρότης) Web εφαρμογή (Agric Cloud)	
Trigger (Εναυσμα)	Το σύστημα κατασκευάζει την αναφορά και τη σερβίρει στο χρήστη για κατέβασμα.	
Περιγραφή	Βήμα	Ενέργεια
	1	Ο χρήστης στην οθόνη θέασης δεδομένων άρδευσης επιλέγει

		τον τύπο αρχείου της αναφοράς.
	2	Ο χρήστης επιλέγει μεταξύ των τύπων αρχείων (PDF, XLS, CSV) για κατέβασμα
	3	Ο χρήστης επιλέγει να δει το αρχείο που έχει κατεβάσει
Επεκτάσεις	Βήμα	Ενέργεια
	1	Στο αρχείο αναφοράς σε μορφή PDF ο χρήστης μπορεί να δει μεταξύ των τιμών σε μορφή πίνακα τη μέγιστη, ελάχιστη καθώς και τη μέση τιμή της κατανάλωσης σε λίτρα.
	2	Ο χρήστης είναι σε θέση να ανακατευθυνθεί στις καρτέλες των δεδομένων του πίνακα.
	3	Ο χρήστης μπορεί κάνοντας hover στο διάγραμμα να δει τη συγκεκριμένη χρονική στιγμή και τη μέτρηση τη στιγμή αυτή.

Πίνακας 3.32: Use case 12: Μεταφόρτωση μετρήσεων αισθητήρων

Πλαίσιο χρήσης	Ο σταθμός του αγροκτήματος επικοινωνεί με την εφαρμογή για να αποστείλει μέσω του δικτύου 3g τα δεδομένα των μετρήσεων των αισθητήρων.	
Πεδίο εφαρμογής	Web application	
Επίπεδο	Υπολειτουργία	
Προαπαιτούμενα	Ο σταθμός πρέπει να αποστείλει τα δεδομένα με τη σωστή κωδικοποίηση σε συγκεκριμένο endpoint.	
Συνθήκες επιτυχίας	Η εφαρμογή αποκωδικοποιεί τις μετρήσεις και αποθηκεύει τις τιμές στη βάση δεδομένων του διακομιστή.	
Συνθήκες αποτυχίας	Η εφαρμογή δεν αποκωδικοποιεί τα δεδομένα των μετρήσεων. Η αποτυχία καταγράφεται σε ένα plain text αρχείο στο διακομιστή.	
Πρωτεύον, Δευτερεύον Actors	Station (σταθμός αγροκτήματος) Web εφαρμογή (Agric Cloud)	
Trigger (Εναυσμα)	Το σύστημα αποκωδικοποιεί και σώζει τις τιμές των μετρήσεων στη Β.Δ..	
Περιγραφή	Βήμα	Ενέργεια
	1	Ο σταθμός αγροκτήματος αποστέλλει τα δεδομένα σε μορφή JSON σε συγκεκριμένη διεύθυνση υπηρεσίας (endpoint) .
	2	Το σύστημα αποκωδικοποιεί τις τιμές.
	3	Οι ακριβής μέρα και ώρα των μετρήσεων μετατρέπεται σε Coordinated Universal Time (UTC)
	4	Το σύστημα αποθηκεύει τις μετρήσεις στους αντίστοιχους πίνακες της Β.Δ.

Επεκτάσεις	Βήμα	Ενέργεια
------------	------	----------

Πίνακας 3.43: Use case 13: Μεταφόρτωση μέτρησης άρδευσης

Πλαίσιο χρήσης	Ο σταθμός του αγροκτήματος επικοινωνεί με την εφαρμογή για να αποστείλει μέσω του δικτύου 3g τα δεδομένα της άρδευσης.	
Πεδίο εφαρμογής	Web application	
Επίπεδο	Υπολειτουργία	
Προαπαιτούμενα	Ο σταθμός πρέπει να αποστείλει τα δεδομένα με τη σωστή κωδικοποίηση σε συγκριμένο endpoint.	
Συνθήκες επιτυχίας	Η εφαρμογή αποκωδικοποιεί τις μετρήσεις και αποθηκεύει τις τιμές στη βάση δεδομένων του διακομιστή.	
Συνθήκες αποτυχίας	Η εφαρμογή δεν αποκωδικοποιεί τα δεδομένα των μετρήσεων. Η αποτυχία καταγράφεται σε ένα plain text αρχείο στο διακομιστή.	
Πρωτεύον, Δευτερεύον Actors	Station (σταθμός αγροκτήματος) Web εφαρμογή (Agric Cloud)	
Trigger (Εναυσμα)	Το σύστημα αποκωδικοποιεί και σώζει τις τιμές της μέτρησης στη Β.Δ..	
Περιγραφή	Βήμα	Ενέργεια
	1	Ο σταθμός αγροκτήματος αποστέλλει τα δεδομένα σε μορφή JSON σε συγκεκριμένη διεύθυνση υπηρεσίας (endpoint) .
	2	Το σύστημα αποκωδικοποιεί τις τιμές.
	3	Οι ακριβής μέρα και ώρα τους διαστήματος της μέτρησης μετατρέπεται σε UTC
	4	Το σύστημα αποθηκεύει τη μέτρηση στους αντίστοιχους πίνακες της Β.Δ.
Επεκτάσεις	Βήμα	Ενέργεια

Πίνακας 3.54: Use case 14: Δημιουργία συμβάντων άρδευσης και μετρήσεων αισθητήρων

Πλαίσιο χρήσης	Η εφαρμογή δημιουργεί συμβάντα για την άρδευση και τις μετρήσεις των αισθητήρων
Πεδίο εφαρμογής	Web application
Επίπεδο	Υπολειτουργία
Προαπαιτούμενα	Ο σταθμός πρέπει να αποστείλει τα δεδομένα με τη σωστή κωδικοποίηση σε συγκριμένο endpoint, και να έχουν σωθεί στους κατάλληλους πίνακες της Β.Δ.
Συνθήκες επιτυχίας	Το σύστημα δημιουργεί νέα συμβάντα δημιουργώντας νέες εγγραφές σε πίνακα της Β.Δ.

Συνθήκες αποτυχίας	Η εφαρμογή δεν καταφέρνει να δημιουργήσει τις νέες εγγραφές στον πίνακα της Β.Δ. Η αποτυχία καταγράφεται σε ένα plain text αρχείο στο διακομιστή.	
Πρωτεύον, Δευτερεύον Actors	Web εφαρμογή (Agric Cloud) Web εφαρμογή (Agric Cloud)	
Trigger (Εναυσμα)	Το σύστημα αποθηκεύει τη στιγμή (ημερομηνία και ώρα) σε χρόνο utc	
Περιγραφή	Βήμα	Ενέργεια
	1	Το σύστημα δημιουργεί νέες εγγραφές
	2	Πραγματοποιείται η καταχώρηση τους στη Β.Δ σε χρόνο utc.
Επεκτάσεις	Βήμα	Ενέργεια

<i>Πίνακας 3.65: Use case 15: Επεξεργασία μέγιστων και ελάχιστων τιμών αυτόματου αλγορίθμου</i>		
Πλαίσιο χρήσης	Ο χρήστης επεξεργάζεται τις μέγιστες και ελάχιστες τιμές.	
Πεδίο εφαρμογής	Web application	
Επίπεδο	Υπολειτουργία	
Προαπαιτούμενα	Ο χρήστης πρέπει να έχει κάνει είσοδο στην εφαρμογή.	
Συνθήκες επιτυχίας	Ο χρήστης επεξεργάζεται τα στοιχεία μέγιστων και ελαχίστων ανά τύπο παρατήρησης και αποθηκεύει τις τιμές αυτές.	
Συνθήκες αποτυχίας	Η	
Πρωτεύον, Δευτερεύον Actors	Χρήστης (Agric Cloud) Web εφαρμογή (Agric Cloud)	
Trigger (Εναυσμα)	Το σύστημα ελέγχει τις μέγιστες τιμές, πρέπει να είναι μικρότερες από τις ελάχιστες και αντίστοιχα οι ελάχιστες να είναι μεγαλύτερες από τις μέγιστες.	
Περιγραφή	Βήμα	Ενέργεια
	1	Ο χρήστης κατευθύνεται στην αντίστοιχη οθόνη μέσω του μενού
	2	Επεξεργάζεται τις τιμές μέγιστων και ελαχίστων ανά συσκευή αισθητήρων.
	3	Σώζει τις επεξεργασμένες τιμές.
	4	Ένα μήνυμα επιτυχίας σε μορφή αναδυόμενου παραθύρου εμφανίζεται στον χρήστη.
Επεκτάσεις	Βήμα	Ενέργεια
	1	Ο χρήστης μπορεί να καθαρίσει τα πεδία μέγιστων και ελαχίστων τιμών

<i>Πίνακας 3.76: Use case 16: Νέα μέτρηση</i>		
Πλαίσιο χρήσης	Ο χρήστης επιλέγει νέα μέτρηση.	
Πεδίο εφαρμογής	Web application	
Επίπεδο	Υπολειτουργία	
Προαπαιτούμενα	Ο χρήστης πρέπει να έχει κάνει είσοδο στην εφαρμογή.	
Συνθήκες επιτυχίας	Ένα μήνυμα επιτυχίας εμφανίζεται στην οθόνη του χρήστη.	
Συνθήκες αποτυχίας	Ένα μήνυμα αποτυχίας εμφανίζεται στην οθόνη του χρήστη.	
Πρωτεύον, Δευτερεύον Actors	Χρήστης (Agric Cloud) Web εφαρμογή (Agric Cloud)	
Trigger (Εναυσμα)	Το σύστημα βρίσκει όλες τις τελικές συσκευές του αγροκτήματος και αλλάζει τη συνθήκη νέας μέτρησης κάθε συσκευής.	
Περιγραφή	Βήμα	Ενέργεια
	1	Ο χρήστης κατευθύνεται στην αντίστοιχη οθόνη μέσω του μενού
	2	Επιλέγει νέα μέτρηση για όλους τους αισθητήρες του αγροκτήματος
	3	Ένα μήνυμα επιτυχίας σε μορφή αναδυόμενου παραθύρου εμφανίζεται στον χρήστη.
Επεκτάσεις	Βήμα	Ενέργεια
	1	Ο χρήστης μπορεί να δει τις τιμές των τελευταίων μετρήσεων ανά συσκευή αισθητήρων για όλες τις συσκευές.

<i>Πίνακας 3.87: Use case 17: Άρδευση συγκεκριμένου διαστήματος</i>		
Πλαίσιο χρήσης	Ο χρήστης επιλέγει νέα άρδευση.	
Πεδίο εφαρμογής	Web application	
Επίπεδο	Υπολειτουργία	
Προαπαιτούμενα	Ο χρήστης πρέπει να έχει κάνει είσοδο στην εφαρμογή, το διάστημα χρόνων άρδευσης να έχει μελλοντικές στιγμές και το διάστημα να είναι έγκυρο.	
Συνθήκες επιτυχίας	Αποθηκεύονται οι χρόνοι του διαστήματος στη Β.Δ. Ένα μήνυμα επιτυχίας εμφανίζεται στην οθόνη του χρήστη.	
Συνθήκες αποτυχίας	Ένα μήνυμα αποτυχίας εμφανίζεται στην οθόνη του χρήστη.	
Πρωτεύον, δευτερεύον Actors	Χρήστης (Agric Cloud) Web εφαρμογή (Agric Cloud)	
Trigger (Εναυσμα)	Το σύστημα ελέγχει τις αν οι μέρες και οι ώρες του διαστήματος είναι έγκυρες και αν είναι εκτός διαστήματος αυτόματου αλγορίθμου.	

Περιγραφή	Βήμα	Ενέργεια
	1	Ο χρήστης κατευθύνεται στην αντίστοιχη οθόνη μέσω του μενού
	2	Επιλέγει νέο συμβάν άρδευσης, ένα αναδυόμενο παράθυρο εμφανίζεται για την επιλεγμένη συσκευή άρδευσης.
	3	Ο χρήστης επιλέγει το διάστημα μελλοντικών ημερομηνιών από και έως.
	4	Στη συνέχεια επιλέγει να ξεκινήσει η άρδευση.
	5	Ένα μήνυμα επιτυχίας εμφανίζεται στο χρήστη.
Επεκτάσεις	Βήμα	Ενέργεια
	1	Ο χρήστης βλέπει το διάστημα άρδευσης που έχει επιλέξει καθώς και τη συνολική κατανάλωση του διαστήματος αυτού.

<i>Πίνακας 3.98: Use case 18: Επεξεργασία διαστήματος λειτουργίας αλγορίθμου αυτόματης άρδευσης</i>		
Πλαίσιο χρήσης	Ο χρήστης επιλέγει επεξεργάζεται τις ώρες του διαστήματος από και έως του αυτόματου αλγορίθμου άρδευσης.	
Πεδίο εφαρμογής	Web application	
Επίπεδο	Υπολειτουργία	
Προαπαιτούμενα	Ο χρήστης πρέπει να έχει κάνει είσοδο στην εφαρμογή, το διάστημα χρόνων αυτόματου ποτίσματος άρδευσης να είναι έγκυρο	
Συνθήκες επιτυχίας	Αποθηκεύονται οι χρόνοι του διαστήματος στη Β.Δ. Ένα μήνυμα επιτυχίας εμφανίζεται στην οθόνη του χρήστη.	
Συνθήκες αποτυχίας	Ένα μήνυμα αποτυχίας εμφανίζεται στην οθόνη του χρήστη.	
Πρωτεύον, δευτερεύον Actors	Χρήστης (Agric Cloud) Web εφαρμογή (Agric Cloud)	
Trigger (Εναυσμα)	Το σύστημα ελέγχει αν οι ώρες του διαστήματος είναι έγκυρες	
Περιγραφή	Βήμα	Ενέργεια
	1	Ο χρήστης κατευθύνεται στην αντίστοιχη οθόνη μέσω του μενού
	2	Επεξεργάζεται τις στιγμές από και έως καθώς και την κατανάλωση της συγκεκριμένης ηλεκτρικής βάνας που χρησιμοποιεί.
	3	Ένα αναδυόμενο παράθυρο εμφανίζεται για να εγκρίνει ο χρήστης την αποθήκευση στη Β.Δ.
	4	Ένα μήνυμα επιτυχίας εμφανίζεται στο χρήστη.
Επεκτάσεις	Βήμα	Ενέργεια

	1	Ο χρήστης βλέπει το διάστημα άρδευσης για τον αυτόματο αλγόριθμο που έχει επιλέξει σε μορφή ώρες και λεπτά.
--	---	---

<i>Πίνακας 3.109: Use case 19: Διαχείριση ονομάτων πεδίων ενδιαφέροντος</i>		
Πλαίσιο χρήσης	Ο χρήστης επεξεργάζεται τις περιγραφές των πεδίων ενδιαφέροντος.	
Πεδίο εφαρμογής	Web application	
Επίπεδο	Υπολειτουργία	
Προαπαιτούμενα	Ο χρήστης πρέπει να έχει κάνει είσοδο στην εφαρμογή.	
Συνθήκες επιτυχίας	Αποθηκεύονται οι περιγραφές των σημείων ενδιαφέροντος στη Β.Δ. Ένα μήνυμα επιτυχίας εμφανίζεται στην οθόνη του χρήστη.	
Συνθήκες αποτυχίας	Ένα μήνυμα αποτυχίας εμφανίζεται στην οθόνη του χρήστη.	
Πρωτεύον, δευτερεύον Actors	Χρήστης (Agric Cloud) Web εφαρμογή (Agric Cloud)	
Trigger (Εναυσμα)	Το σύστημα ελέγχει αν τα πεδία των περιγραφών έχουν τιμή.	
Περιγραφή	Βήμα	Ενέργεια
	1	Ο χρήστης κατευθύνεται στην αντίστοιχη οθόνη μέσω του μενού
	2	Επεξεργάζεται τις περιγραφές σημείων ενδιαφέροντος
	3	Ο χρήστης επιλέγει αποθήκευση με σκοπό να στείλει τις τιμές των περιγραφών στη Β.Δ.
	4	Ένα μήνυμα επιτυχίας εμφανίζεται στο χρήστη.
Επεκτάσεις	Βήμα	Ενέργεια

<i>Πίνακας 3.20: Use case 20: Θέαση και διαχείριση γεωγραφικού σημείου θέσης του σταθμού</i>		
Πλαίσιο χρήσης	Ο χρήστης επεξεργάζεται τη θέση του σημείου του σταθμού του αγροκτήματος.	
Πεδίο εφαρμογής	Web application	
Επίπεδο	Υπολειτουργία	
Προαπαιτούμενα	Ο χρήστης πρέπει να έχει κάνει είσοδο στην εφαρμογή.	
Συνθήκες επιτυχίας	Αποθηκεύονται η θέση του σημείου ενδιαφέροντος του σταθμού.	
Συνθήκες αποτυχίας	Ένα μήνυμα αποτυχίας εμφανίζεται στην οθόνη του χρήστη.	
Πρωτεύον, δευτερεύον Actors	Χρήστης (Agric Cloud) Web εφαρμογή (Agric Cloud)	
Trigger (Εναυσμα)	Το σύστημα μεταφράζει το σημείο σε συντεταγμένες.	
Περιγραφή	Βήμα	Ενέργεια

	1	Ο χρήστης κατευθύνεται στην αντίστοιχη οθόνη μέσω του μενού
	2	Επιλέγει να δημιουργήσει ή να αλλάξει το σημείο του σταθμού.
Επεκτάσεις	Βήμα	Ενέργεια

<i>Πίνακας 3.21: Use case 20: Θέαση τιμών εβδομαδιαίας πρόγνωσης καιρού</i>		
Πλαίσιο χρήσης	Ο χρήστης επεξεργάζεται τη θέση του σημείου του σταθμού στο αγρόκτημα.	
Πεδίο εφαρμογής	Web application	
Επίπεδο	Υπολειτουργία	
Προαπαιτούμενα	Ο χρήστης πρέπει να έχει κάνει είσοδο στην εφαρμογή, και να υπάρχει η πληροφορία της γεωγραφικής θέσης του σημείου ενδιαφέροντος.	
Συνθήκες επιτυχίας	Ο χρήστης μπορεί να δει τα δεδομένα πρόγνωσης σε διάστημα επτά (7) ημερών.	
Συνθήκες αποτυχίας	Ο χρήστης δεν μπορεί να δει τα δεδομένα πρόγνωσης σε διάστημα επτά (7) ημερών. Ένα μήνυμα αποτυχίας εμφανίζεται στην οθόνη του χρήστη.	
Πρωτεύον, δευτερεύον Actors	Χρήστης (User) Υπηρεσία πρόγνωσης καιρού (Weather service)	
Trigger (Εναυσμα)		
Περιγραφή	Βήμα	Ενέργεια
	1	Ο χρήστης κατευθύνεται στην αντίστοιχη οθόνη μέσω του μενού
	2	Ο χρήστης παρατηρεί σε μορφή διαγράμματος και πίνακα τις τιμές για την πρόγνωση καιρού, καθώς και δεδομένα όπως ατμοσφαιρική πίεση, υγρασία κ.α.
Επεκτάσεις	Βήμα	Ενέργεια
	1	Ο χρήστης μπορεί να επιλέξει συγκεκριμένη ημέρα της εβδομαδιαίας πρόγνωσης για να δει τις καιρικές συνθήκες σε διάστημα ωρών

<i>Πίνακας 3.21: Use case 21: Κατέβασμα μέγιστων, ελάχιστων τιμών, συνθηκών μέτρησης και άρδευσης, διαστήματος αυτόματου αλγορίθμου</i>	
Πλαίσιο χρήσης	Ο χρήστης επεξεργάζεται τη θέση του σημείου του σταθμού στο αγρόκτημα.
Πεδίο εφαρμογής	Web application

Επίπεδο	Υπολειτουργία	
Προαπαιτούμενα	Ο σταθμός του αγροκτήματος πρέπει να επικοινωνήσει με την κατάλληλη υπηρεσία, και να υπάρχουν ορισμένες οι πληροφορίες μέγιστων ελάχιστων τιμών καθώς και των χρόνων του αλγορίθμου στη Β.Δ.	
Συνθήκες επιτυχίας	Ο σταθμός αλλάζει τις τιμές σύμφωνα με αυτές που λαμβάνει από την εφαρμογή	
Συνθήκες αποτυχίας	Το ενσωματωμένο σύστημα του σταθμού δεν λαμβάνει τα κατάλληλα δεδομένα και εισέρχεται σε κατάσταση σφάλματος.	
Πρωτεύον, δευτερεύον Actors	Σταθμός αγροκτήματος (Crop Station) Web εφαρμογή (Agric Cloud)	
Trigger (Εναυσμα)		
Περιγραφή	Βήμα	Ενέργεια
	1	Ο σταθμός αιτείται μέσω των υπηρεσιών του συστήματος τα κατάλληλα δεδομένα.
	2	Ο σταθμός λαμβάνει τα μορφοποιημένα δεδομένα και αλλάζει τις τιμές του βασικού αλγορίθμου του.
Επεκτάσεις	Βήμα	Ενέργεια

Πίνακας 3.22: Use case 22: Παροχή δεδομένων πρόγνωσης καιρού

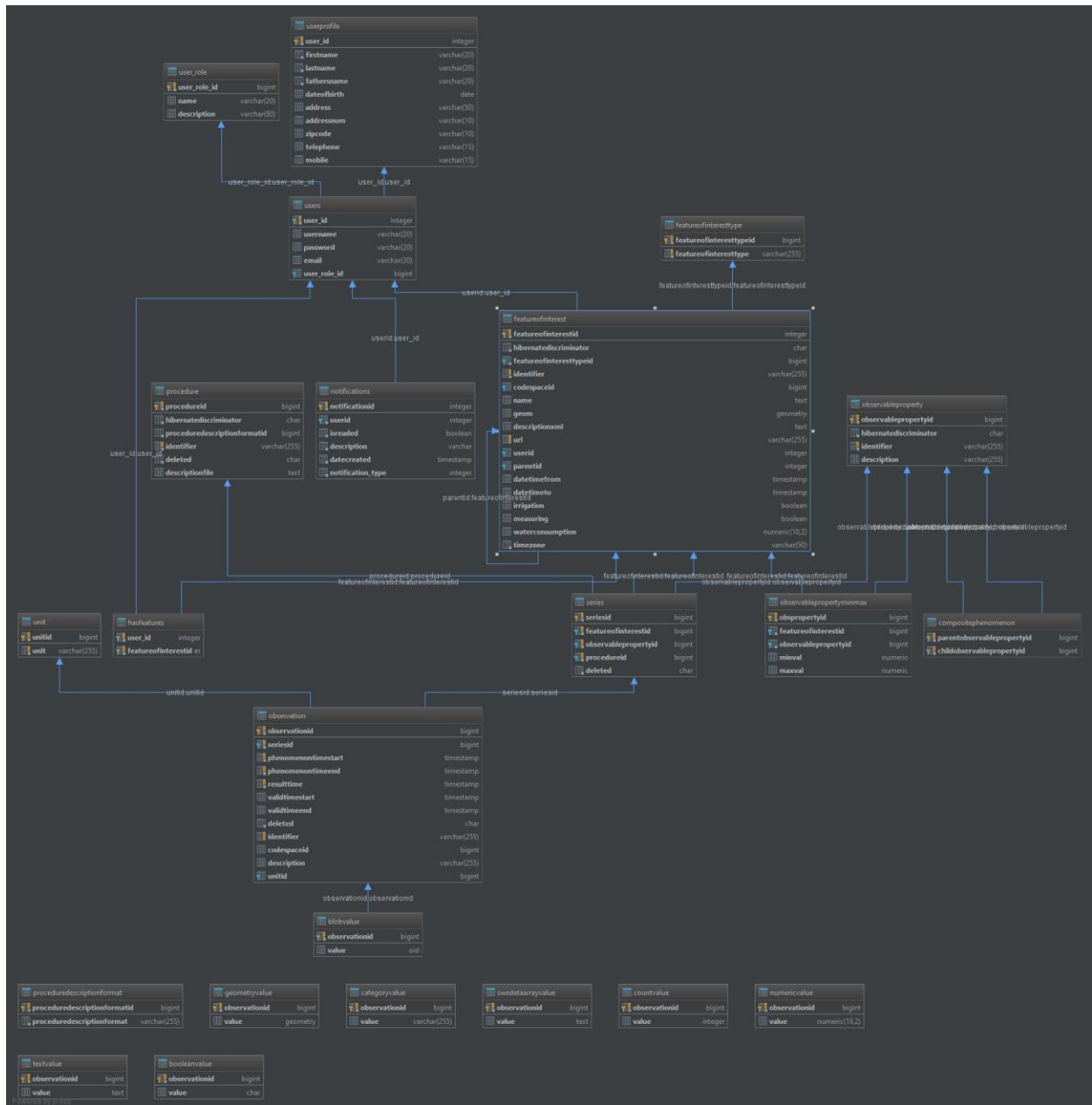
Πλαίσιο χρήσης	Η υπηρεσία πρόγνωσης καιρού στέλνει στο σύστημα δεδομένα σε μορφή json για την πρόγνωση καιρού των επόμενων 7 ημερών	
Πεδίο εφαρμογής	Web application	
Επίπεδο	Υπολειτουργία	
Προαπαιτούμενα	Να υπάρχουν η γεωγραφική θέση του σημείου στη Β.Δ.	
Συνθήκες επιτυχίας	Δεδομένα με την ημερήσια και με την εβδομαδιαία πρόγνωση του καιρού αποστέλλονται στο χρήστη.	
Συνθήκες αποτυχίας	Η υπηρεσία αποτυγχάνει να αποστείλει δεδομένα στο σύστημα	
Πρωτεύον, δευτερεύον Actors	Υπηρεσία πρόγνωσης καιρού (Weather service) Web εφαρμογή (Agric Cloud)	
Trigger (Εναυσμα)	Αποστέλλει τις συντεταγμένες του σταθμού του αγροκτήματος	
Περιγραφή	Βήμα	Ενέργεια
	1	Η εφαρμογή αιτείται από την υπηρεσία πρόγνωσης καιρού τα δεδομένα πρόγνωσης σύμφωνα με το γεωγραφικό σημείο.
	2	Η εφαρμογή λαμβάνει τα στοιχεία αυτά και τα παρουσιάζει σε μορφή γραφήματος και πίνακα ανά ημέρα πρόγνωσης
Επεκτάσεις	Βήμα	Ενέργεια

4. Μοντέλο Σχεσιακής Βάσης Δεδομένων

Στο κεφάλαιο αυτό περιγράφεται το μοντέλο της βάσης δεδομένων Β.Δ καθώς και βασικές λειτουργίες της εφαρμογής και ο τρόπος με τον οποίο μεταφράζονται στο επίπεδο αυτό. Το κεφάλαιο 4.1 περιγράφει τον προσδιορισμό των οντοτήτων (πινάκων, πρωτεύων κλειδιών, μοναδικών αναγνωριστικών) που χρησιμοποιεί η σχεσιακή Β.Δ. Στο κεφάλαιο 4.2 παρουσιάζονται παραδείγματα χρήσης των μοντέλων της βάσης για την δημιουργία εγγραφών στη Β.Δ.

4.1 Προσδιορισμός

Το μοντέλο σχεδιάστηκε για να καλύψει τις ανάγκες των εφαρμογών του συστήματος. Σχεδιάστηκε σύμφωνα με τα πρότυπα του SOS [10], τα οποία θεσπίστηκαν από την OGC. Το μοντέλο αυτό υποστηρίζει την επεκτασιμότητα των συστημάτων αισθητήρων καθώς και ένα πλήθος τύπου παρατηρήσεων, είδος τιμών και διαφορετικές μονάδες μετρήσεων. Υποστηρίζει επίσης διαβάθμιση στο είδος των παρατηρήσεων του κάθε αισθητήρα. Το μοντέλο επεκτάθηκε έτσι ώστε να καλύψει τις ιδιαιτερότητες των εφαρμογών, όπως την υποστήριξη διαπιστευμένων χρηστών και την επικοινωνία με τα ενσωματωμένα συστήματα των σταθμών των αγροκτημάτων. Παρακάτω αναλύεται η χρήση του μοντέλου για κάθε μια από τις βασικές ενέργειες που λαμβάνουν χώρα στις εφαρμογές (web και Android) με τους χρήστες τους καθώς και με την επικοινωνία των σταθμών των αγροκτημάτων με τις υπηρεσίες που έχουν παραχθεί στο επίπεδο της παραγωγής restful υπηρεσιών. Στο **Error! Reference source not found.** διακρίνονται οι πίνακες που εμπλέκονται στο μοντέλο βάσης μας. Για κάθε ένα πίνακα εξηγείται ο ρόλος του, αναλύονται τα χαρακτηριστικά του όπως στήλες, πρωτεύον κλειδιά, ξένα κλειδιά, μοναδικά χαρακτηριστικά κ.α.



Σχήμα 4.1: Μοντέλο Βάσης Δεδομένων

User (Χρήστης)

- user_id: Το πρωτεύον κλειδί και μοναδικό αναγνωριστικό του πίνακα.
- username: Το κωδικό όνομα του χρήστη.
- password: Ο κωδικός του χρήστη
- email: Το email του χρήστη, αποτελεί επίσης μοναδικό αναγνωριστικό
- user_role_id: Ο ρόλος του χρήστη. Δευτερεύον κλειδί συνδέει τους πίνακες user με τον πίνακα user_role.

user_role (Ρόλος χρήστη)

- user_role_id: Το πρωτεύον κλειδί και μοναδικό αναγνωριστικό του πίνακα.

- name: Η σύντομη ονομασία του ρόλου (π.χ. User, Admin)
- description: Η περιγραφή του ρόλου.

userprofile (Προφίλ χρήστη), αποθηκεύονται τα προσωπικά δεδομένα του χρήστη για επικοινωνία.

- user_id: Το πρωτεύον κλειδί και μοναδικό αναγνωριστικό του πίνακα. Αποτελεί επίσης ξένο κλειδί, καθώς συνδέει με τον πίνακα user με σχέση 1 προς 1.
- firstname: Το όνομα του χρήστη.
- lastname: Το επώνυμο του χρήστη.
- fathersname: Το όνομα πατέρα του χρήστη.
- dateofbirth: Η ημερομηνία γέννησης τους χρήστη. Αποθηκεύεται στη βάση σε μορφή date.
- address: Η διεύθυνση του χρήστη.
- addressnum: Ο αριθμός της διεύθυνσης.
- zipcode: Ο ταχυδρομικός κώδικα της διεύθυνσης
- telephone: Ο αριθμός του σταθερού τηλεφώνου του χρήστη.
- mobile: Ο αριθμός του κινητού τηλεφώνου του χρήστη.

Notifications (Ειδοποιήσεις)

- notificationid: Το πρωτεύον κλειδί και μοναδικό αναγνωριστικό του πίνακα
- userid: Ο χρήστης που λαμβάνει την ειδοποίηση. Ξένο κλειδί συνδέει τους πίνακες notifications και user.
- isreaded: Μεταβλητή true/false αν το μήνυμα είναι διαβασμένο από το χρήστη.
- description: Το κείμενο του μηνύματος. Παράγεται αυτόματα από την εφαρμογή.
- datecreated: Ημερομηνία δημιουργίας της ειδοποίησης στην εφαρμογή. Η χρονική στιγμή είναι σύμφωνα με το ρολόι του χρήστη.
- notification_type: Είδος ειδοποίησης, αν πρόκειται για νέα μέτρηση ή νέα άρδευση. Πιθανές τιμές 0: Νέα μέτρηση, 1: Νέα άρδευση

featureofinterest (Πεδίο ενδιαφέροντος): Ένας από τους βασικούς πίνακες της εφαρμογής.

Αποτελούν τις οντότητες πάνω στις οποίες εφαρμόζονται οι ενέργειες των νέων μετρήσεων και της άρδευσης.

- featureofinterestid: Πρωτεύον κλειδί μοναδικό αναγνωριστικό του πίνακα.
- featureofinteresttypeid: Ξένο κλειδί συνδέει τον πίνακα με τον πίνακα είδος σημείου ενδιαφέροντος (featureofinteresttype).
- identifier: Μοναδικό αναγνωριστικό. Ανάλογα με το είδος του πεδίου ενδιαφέροντος για τις τελικές συσκευές και τους σταθμούς αποτελείται από τη διεύθυνση του κάθε ZigBee αντίστοιχα, ενώ για τα αγροτεμάχια αποτελείται από μια τυχαία συμβολοσειρά (GUID).
- name: Το όνομα του σημείο ενδιαφέροντος, ο χρήστης έχει τη δυνατότητα να το παραμετροποιεί.
- geom: Το γεωγραφικό σημείο για το κάθε σημείο ενδιαφέροντος. Είναι τύπου (GEOMETRY), μπορεί να αναπαραστήσει διάφορους τύπους γεωμετριών. Για το

σταθμό και τις τελικές συσκευές με τους αισθητήρες χρησιμοποιείται το σημείο (point) με συντεταγμένες (X,Y) τετμημένη τεταγμένη, ενώ για το αγροτεμάχιο αποτελείται από γεωμετρία τύπου πολυγώνου.

- **userid:** Ξένο κλειδί που συνδέει με τον πίνακα user μέσω του πρωτεύοντος κλειδιού του πίνακα user. Δηλώνει το «κάτοχο» του κάθε πεδίου ενδιαφέροντος.
- **parentid.** Ξένο κλειδί συνδέει τον πίνακα με τον εαυτό του, αυτοσυσχέτιση. Το κάθε πεδίο ενδιαφέροντος «ανήκει» σε κάποιο άλλο. Έτσι ιεραρχούνται τα πεδία ενδιαφέροντος σε διαφορετικά επίπεδα. Ξεκινώντας από τις τελικές συσκευές με τους αισθητήρες, οι οποίοι ανήκουν – συνδέονται με ένα σταθμό. Ο σταθμός ανήκει σε ένα αγροτεμάχιο.
- **measuring:** Συνθήκη για δημιουργία νέας μέτρησης ανά τελική συσκευή αισθητήρων. Τύπος boolean true/false.
- **irrigation:** Συνθήκη για τη δημιουργία νέας άρδευσης ανά τελική συσκευή αισθητήρων. Τύπος boolean true/false.
- **datetimefrom:** Αποτελεί την ημερομηνία έναρξης της άρδευσης όταν ο τύπος του πεδίου ενδιαφέροντος είναι τελική συσκευή. Ενώ αν πρόκειται για σταθμό αποτελεί τη χρονική στιγμή έναρξης του αυτόματου αλγόριθμου ποτίσματος του σταθμού. Στην περίπτωση του αγροτεμαχίου το πεδίο μένει κενό.
- **datetimeto:** Αποτελεί την ημερομηνία λήξης της άρδευσης όταν ο τύπος του πεδίου ενδιαφέροντος είναι τελική συσκευή. Ενώ αν πρόκειται για σταθμό αποτελεί τη χρονική στιγμή λήξης του αυτόματου αλγόριθμου ποτίσματος του σταθμού. Στην περίπτωση του αγροτεμαχίου το πεδίο μένει κενό.
- **waterconsumption:** Είναι η κατανάλωση της βάνας που χρησιμοποιείται για την άρδευση. Αναφέρεται μόνο στον τύπο που το πεδίο ενδιαφέροντος είναι σταθμός, στις άλλες περιπτώσεις το πεδίο μένει κενό.
- **timezone:** Αναφέρεται στη ζώνη ώρας που βρίσκεται το κάθε αγροτεμάχιο για την αλλαγή των ημερομηνιών όταν αυτές στέλνονται σαν πληροφορίες στον χρήστη του συστήματος.

featureofinterestype (Τύπος πεδίου ενδιαφέροντος):

- **featureofinterestypeid:** Πρωτεύον κλειδί μοναδικό αναγνωριστικό πίνακα.
- **description:** Περιγραφή του τύπου ενδιαφέροντος (π.χ. Σταθμός, Αγροτεμάχιο, Τελική συσκευή αισθητήρων)

observableproperty (Είδος μέτρησης):

- **observablepropertyid:** Πρωτεύον κλειδί και μοναδικό αναγνωριστικό πίνακα.
- **identifier:** Κωδική ονομασία τύπου μέτρησης.
- **description:** Περιγραφή είδους μέτρησης.

observablepropertyminmax (Μέγιστες ελάχιστες τιμές τύπου μέτρησης):

- **obspropertyid:** Πρωτεύον κλειδί μοναδικό αναγνωριστικό πίνακα.
- **featureofinterestid:** Ξένο κλειδί προς τον πίνακα featureofinterest,
- **observablepropertyid:** Ξένο κλειδί προς τον πίνακα observableproperty
- **minval:** Ελάχιστη τιμή (κατώφλι) για το πεδίο ενδιαφέροντος του είδους μέτρησης.

- `maxval`: Ανώτατη τιμή (ανώφλι) για το πεδίο ενδιαφέροντος του είδους μέτρησης.

procedure (Διαδικασία): Αποθηκεύει τις πληροφορίες για όλα τα όργανα του ενσωματωμένου.

- `procedureid`: Πρωτεύον κλειδί μοναδικό αναγνωριστικό πίνακα.
- `proceduredescriptionformatid`: Ξένο κλειδί σύνδεσης με τον πίνακα `proceduredescriptionformat`.
- `identifier`: Μοναδικό αναγνωριστικό, κωδική ονομασία του κάθε αισθητήρα ή κάποιου άλλου οργάνου που χρησιμοποιεί το ενσωματωμένο σύστημα.
- `deleted`: Υποδηλώνει αν κάποιο όργανο έχει αφαιρεθεί από το ενσωματωμένο.
- `descriptionfile`: Περιγραφή του οργάνου και των διαδικασιών του.

proceduredescriptionformat (Μορφή διαδικασιών):

- `proceduredescriptionformatid`: Πρωτεύον κλειδί μοναδικό αναγνωριστικό πίνακα.
- `description`: Περιγραφή της μορφής διαδικασίας. Στην περίπτωση της εφαρμογής διαχωρίζονται τα δεδομένα σε: (α) προς θέαση για το χρήστη (β) εσωτερικά της εφαρμογής.

series (σειρά - διεργασία): Χρησιμοποιείται για να περιγράψει την ενέργεια της μέτρησης ενός συγκεκριμένου φαινομένου σε ένα πεδίο ενδιαφέροντος, χρησιμοποιώντας κάποιο όργανο.

- `seriesid`: Πρωτεύον κλειδί μοναδικό αναγνωριστικό πίνακα.
- `featureofinterestid`: Ξένο κλειδί σύνδεσης με τον πίνακα πεδίου ενδιαφέροντος (`featureofinterest`).
- `observablepropertyid`: Ξένο κλειδί σύνδεσης με τον πίνακα είδους μέτρησης (`observableproperty`).
- `procedureid`: Ξένο κλειδί σύνδεσης με τον πίνακα των οργάνων (`procedure`).
- `deleted`: Υποδηλώνει αν κάποιο όργανο έχει αφαιρεθεί από το ενσωματωμένο.

Ο πίνακας αυτός περιέχει σαν μοναδικό αναγνωριστικό το συνδυασμό των ξένων κλειδιών καθιστώντας με αυτό τον τρόπο μοναδική την κάθε ενέργεια μέτρησης.

Observation (Μέτρηση): Αποτελεί το βασικό πίνακα στον οποίο αποθηκεύονται οι τιμές των μετρήσεων από τους αισθητήρες και τα υπόλοιπα όργανα των καταναμεμένων.

- `observationid`: Πρωτεύον κλειδί μοναδικό αναγνωριστικό πίνακα.
- `seriesid`: Ξένο κλειδί σύνδεσης με τον πίνακα πεδίου ενδιαφέροντος (`series`).
- `phenomenontimestart`: Χρονική στιγμή έναρξης του φαινομένου μέτρησης.
- `phenomenontimeend`: Χρονική στιγμή λήξης του φαινομένου μέτρησης. Αν το φαινόμενο είναι στιγμιαίο τότε η στιγμή λήξης συμπίπτει με τη στιγμή έναρξης.
- `resulttime`: Χρονική στιγμή αποθήκευσης του φαινομένου στη βάση.
- `validtimestart`: Χρονική στιγμή έγκυρης έναρξης φαινομένου (δεν χρησιμοποιείται).
- `validtimeend`: Χρονική στιγμή έγκυρης λήξης φαινομένου (δεν χρησιμοποιείται).
- `deleted`: Υποδηλώνει αν κάποια μέτρηση έχει αφαιρεθεί από το σύστημα.
- `identifier`: Μοναδικό αναγνωριστικό του πίνακα, δημιουργείται αυτόματα από το σύστημα και έχει την μορφή (YYYYMMDDHHmmss-GUID). Όπου YYYY είναι το έτος, MM ο

μήνας σε μορφή (01-12), DD οι μέρες σε μορφή (01-31), HH οι ώρες σε μορφή (01-24), mm τα λεπτά σε μορφή (01-60) και SS τα δευτερόλεπτα στη μορφή (01-60) της χρονικής στιγμής έναρξης του φαινομένου μέτρησης. Ενώ guid είναι μια τυχαία συμβολοσειρά μεγέθους 32 χαρακτήρων που παράγεται από το σύστημα.

- description: Περιγραφή της κάθε μέτρησης (δεν χρησιμοποιείται)
- unitid: Ξένο κλειδί συνδέει με τον πίνακα unit.

Ο πίνακας αυτός έχει σαν μοναδικό αναγνωριστικό το συνδυασμό των στηλών seriesid, phenomenontimestart, phenomenontimeend, έτσι εξασφαλίζεται η μοναδικότητα της κάθε μέτρησης ανά διεργασία.

unit (Μονάδα μέτρησης):

- unitid: Πρωτεύον κλειδί μοναδικό αναγνωριστικό πίνακα.
- unit: Περιγραφή μονάδας μέτρησης.

numericvalue (Αριθμητικές τιμές μέτρησης):

- observationid: Πρωτεύον κλειδί μοναδικό αναγνωριστικό πίνακα. Ξένο κλειδί σύνδεσης με τον πίνακα μέτρησης (observation) με σχέση ένα προς ένα.
- value: Αριθμητική τιμή μέτρησης.

Εκτός από τους παραπάνω πίνακες - οντότητες που χρησιμοποιούνται στο σύστημα το οποίο σχεδιάστηκε, οι πίνακες blobvalue, booleanvalue, geometryvalue, categoryvalue, swedataarrayvalue, countvalue, textvalue στους οποίους αποθηκεύονται μετρήσεις τιμών τύπου boolean, text, string, γεωμετρία, ακέραιος αριθμός, blob και clob. Επίσης ο πίνακας compositerphenomenon όπου χρησιμοποιείται για να περιγράψει σύνθετα φαινόμενα που περιλαμβάνουν σχέσεις μεταξύ των ειδών μέτρησης. Οι πίνακες αυτοί υπάρχουν στη βάση δεδομένων για την επέκταση του συστήματος με την προσθήκη νέων αισθητήρων καθώς και νέων τύπων μετρήσεων.

4.2 Παραδείγματα χρήσης μοντέλων

Όπως προαναφέρθηκε το σχήμα της βάσης δεδομένων έχει σχεδιαστεί ώστε να υποστηρίζει την αποθήκευση και τη θέαση διαφορετικών τύπων μετρήσεων με διαφορετικές μονάδες μέτρησης, καθώς και τύπους χρονικής διάρκειας (στιγμιαία, χρονικό διάστημα). Επίσης υποστηρίζει την επικοινωνία με το ενσωματωμένο σύστημα του σταθμού στο αγροτεμάχιο για τη δημιουργία εναυσμάτων νέας μέτρησης, νέας άρδευσης μεταξύ συγκεκριμένου χρονικού διαστήματος και παραμετροποίησης της χρονικής διάρκειας του αυτόματου αλγορίθμου άρδευσης. Την παραμετροποίηση των ονομάτων των συσκευών παρατήρησης που συλλέγουν τις μετρήσεις από τους αισθητήρες για τη μετέπειτα χρήση τους στο επίπεδο των δύο εφαρμογών. Τη θέαση της αναλυτικής πρόγνωσης των καιρικών συνθηκών για το αγροτεμάχιο σε διάστημα μιας εβδομάδας. Παρακάτω παρουσιάζονται παραδείγματα χρήσης της εφαρμογής.

Θέαση δεδομένων μετρήσεων αισθητήρων

Η θέαση των δεδομένων υποστηρίζεται με διάφορους τρόπους. Είτε μέσω επιλογής του τύπου μέτρησης, είτε της θέασης στη μορφή γραφικής παράστασης σε ζεύγος ανά μονάδα μέτρησης (π.χ. υγρασία εδάφους, υγρασία ατμόσφαιρας). Δίνεται στο χρήστη η δυνατότητα θέασης των τιμών των παρατηρήσεων ανάλογα με το είδος της μέτρησης, την τελική συσκευή αισθητήρων και ένα διάστημα ημερομηνιών το οποίο επιλέγεται είτε σε ζεύγος (από-έως) είτε μεμονωμένα (από ή έως). Η μορφοποίηση της θέασης όπως προαναφέρθηκε περιλαμβάνει γραφήματα, πίνακες, φυσικά αρχεία σε μορφή αναφορών τύπου pdf (portable data file), excel, csv (comma separated values). Η θέαση των τιμών περιλαμβάνει στοιχεία από πολλούς πίνακες όπως αναφέρθηκε παραπάνω, είναι συνδυασμός πεδίου ενδιαφέροντος, τύπου μέτρησης, μονάδα μέτρησης, χρόνων μέτρησης και τιμής μέτρησης.

Παραμετροποίηση περιγραφών συσκευών αισθητήρων

Ο χρήστης έχει τη δυνατότητα παραμετροποίησης των περιγραφών των συσκευών. Επίσης μπορεί να δει από τι στοιχεία αποτελείται η κάθε συσκευή και τι είδους μέτρησης δύναται να λάβει το ενσωματωμένο σύστημα. Είναι δυνατή η επεξεργασία του γεωγραφικού σημείου όπου βρίσκεται ο σταθμός με το ενσωματωμένο σύστημα επιλέγοντας ένα σημείο σε υπόβαθρο χάρτη. Ο χρήστης στις ενέργειες αυτές παραμετροποιεί τη στήλη description (περιγραφή) από τον πίνακα featureofinterest (Πεδίο ενδιαφέροντος). Επίσης μπορεί να ορίσει την τιμή για τη ζώνη ώρας ώστε να γίνεται σωστά η αλλαγή των χρονικών στιγμών όπως αυτές έρχονται από τη βάση δεδομένων του συστήματος.

Δημιουργία συμβάντων και επικοινωνία με το σταθμό σε πραγματικό χρόνο.

Ο χρήστης στις περιπτώσεις αυτές μπορεί να δει τις τελευταίες μετρήσεις, τα τελευταία περιστατικά άρδευσης ανά συσκευή αισθητήρων καθώς και το διάστημα λειτουργίας του αυτόματου αλγορίθμου άρδευσης. Οι ενέργειες που μπορεί να πραγματοποιήσει ο χρήστης είναι η δημιουργία νέων συμβάντων μέτρησης και άρδευσης ανά τελική συσκευή. Το περιστατικό άρδευσης δημιουργείται για ένα χρονικό διάστημα εκτός του διαστήματος που λειτουργεί ο αυτόματος αλγόριθμος άρδευσης. Η ενέργεια νέας μέτρησης αλλάζει την τιμή measuring στον πίνακα featureofinterest για την αλλαγή της συνθήκης νέας μέτρησης σε αληθής. Η ενέργεια νέας άρδευσης αλλάζει την τιμή irrigation καθώς και τις τιμές datetimefrom και datetimeto. Οι τιμές measuring και irrigation αφορούν μόνο πεδία ενδιαφέροντος τύπου τελικής συσκευής, ενώ οι τιμές αυτές αγνοούνται για τα πεδία ενδιαφέροντος τύπου σταθμός και καλλιέργεια. Οι τιμές datetimefrom και datetimeto για τα πεδία ενδιαφέροντος τύπου τελική συσκευή αφορούν το χρονικό διάστημα όπου πραγματοποιείται το πότισμα εκτός αυτομάτου αλγορίθμου άρδευσης. Ενώ για το πεδίο ενδιαφέροντος τύπου σταθμού αφορά το διάστημα εντός αλγορίθμου έτσι αποθηκεύεται μόνο η ώρα και τα λεπτά της ημερομηνίας. Η τιμή για το πεδίο ενδιαφέροντος τύπου αγροτεμάχιο αγνοείται από το σύστημα. Τέλος η τιμή waterconsumption αφορά την κατανάλωση που έχει η κάθε βάνα άρδευσης για κάθε σταθμό ενώ αγνοείται για τους τύπους τελική συσκευή και καλλιέργεια.

Επεξεργασία μεγίστων και ελαχίστων τιμών.

Ο χρήστης μπορεί να αλλάξει τις τιμές μεγίστων και ελαχίστων που λειτουργούν σαν απαραίτητη συνθήκη για να ξεκινήσει η άρδευση ανά τελική συσκευή κατά τη διάρκεια λειτουργίας του αυτόματου αλγορίθμου άρδευσης. Ο πίνακας στο μοντέλο της βάσης δεδομένων που αντιπροσωπεύει αυτές τις τιμές είναι ο observablepropertyminmax και

αποτελείται όπως αναφέρθηκε από ξένα κλειδιά προς τους πίνακες «πεδίο ενδιαφέροντος» και «τύπος μέτρησης», καθώς και με τις δύο τιμές μια ελάχιστη και μια μέγιστη. Έτσι ο χρήστης είναι σε θέση να παραμετροποιήσει τις τιμές αυτές ανά τελική συσκευή μετρήσεων και τύπο μέτρησης όπως κρίνει αυτός ανάλογα με τις αντικειμενικές συνθήκες όπως είναι οι καιρικές συνθήκες της περιοχής, το είδος του υπεδάφους, η χημική του σύσταση κ.α.

5. Αρχιτεκτονική συστημάτων

Το κεφάλαιο αυτό περιγράφει την αρχιτεκτονική των εφαρμογών αναλύει τα βασικά τους συστατικά και τις επιμέρους λειτουργίες τους. Η εφαρμογή σχεδιάστηκε και υλοποιήθηκε ακολουθώντας τα πρότυπα και τις αρχές του Rich Internet Application (R.I.A) οι οποίες υπαγορεύουν μια WEB εφαρμογή να έχει πολλά χαρακτηριστικά μιας Desktop εφαρμογής. Αυτό επιτυγχάνεται κάνοντας χρήση του περιβάλλοντος ενός browser της γλώσσας Javascript. Το πρότυπο για την ανάπτυξη τέτοιων εφαρμογών στις μέρες μας είναι το HTML5 το οποίο την τελευταία δεκαετία έχει υιοθετηθεί από το σύνολο των browsers.

Για τη σχεδίαση των εφαρμογών ακολουθήθηκαν συγκεκριμένα design patterns τα οποία βοήθησαν τη γρηγορότερη ανάπτυξη της εφαρμογής με τη χρήση επαναχρησιμοποιούμενου κώδικα όπου αυτό ήταν δυνατό. Η χρήση προτύπων σχεδίασης διευκολύνει επίσης την κατανόηση και την ανάπτυξη κώδικα από τρίτους για μελλοντική χρήση και αναβάθμιση. Συγκεκριμένα για την client εφαρμογή χρησιμοποιείται το Model View Controller (MVC) πρότυπο σχεδίασης. Ενώ για το backend της εφαρμογής χρησιμοποιείται αρχιτεκτονική τύπου multi-tier δηλαδή τα κομμάτια που συνθέτουν την πολυστρωματική (multilayer) αυτή απεικόνιση είναι ανεξάρτητα μεταξύ τους.

Όπως έχει αναφερθεί παραπάνω η εφαρμογή χωρίζεται σε δύο μέρη το server side και το client side όπως φαίνεται στο Σχήμα 5.1], στη συνέχεια αναλύονται τα δύο αυτά μέρη και τα κομμάτια που τα αποτελούν.

5.1 Server Side

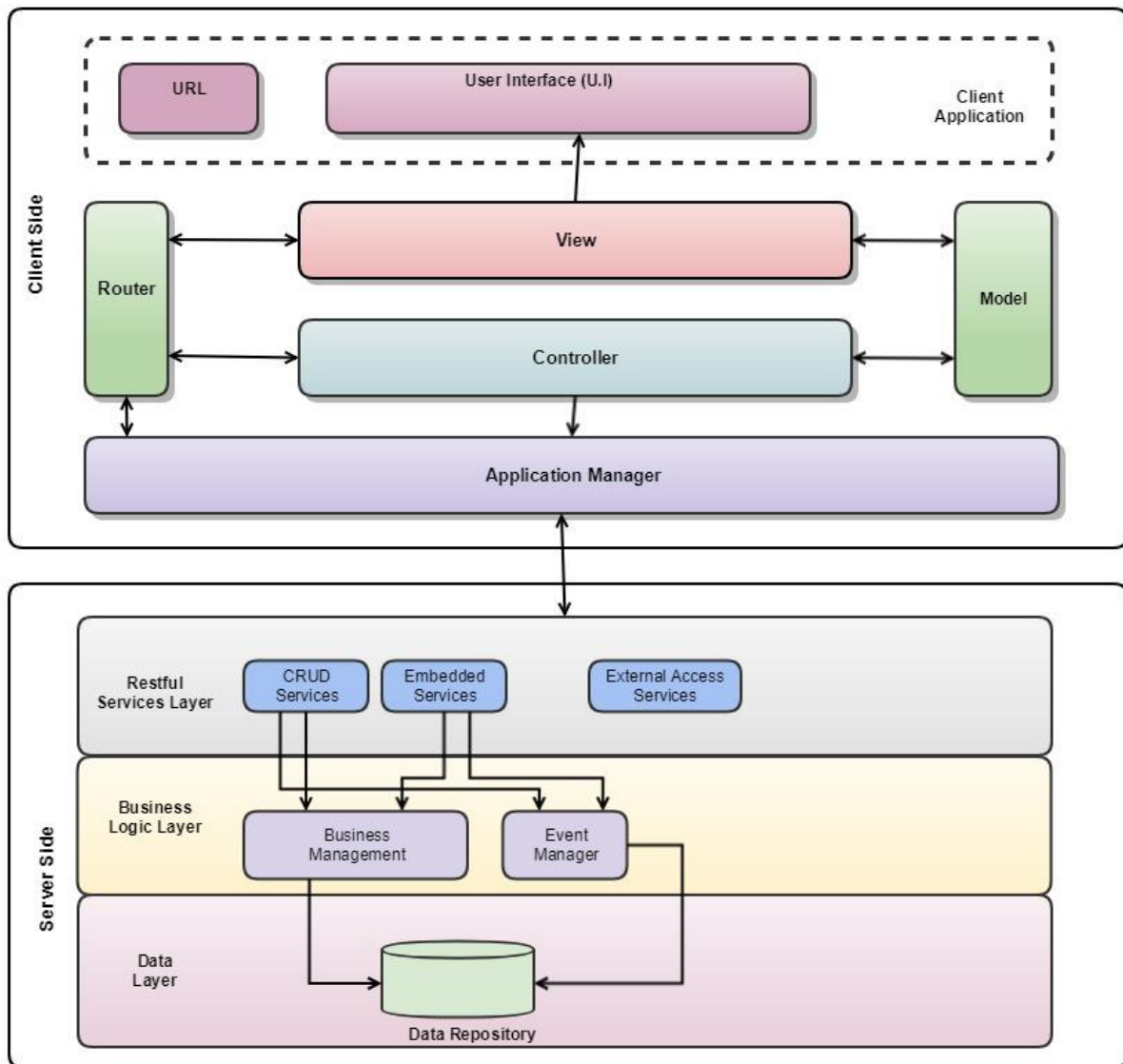
Το server side κομμάτι της εφαρμογής όπως περιεγράφηκε ακολουθεί πολυστρωματική αρχιτεκτονική. Τα κομμάτια που την αποτελούν είναι τα εξής **service layer**, **business logic layer**, **data repository layer**, **data layer**. Τα κομμάτια αυτά επικοινωνούν μεταξύ τους κάθετα προς μια κατεύθυνση δηλαδή χρησιμοποιώντας την ιεραρχία όπως φαίνεται στο σχήμα. Παρακάτω αναλύονται τα επιμέρους κομμάτια.

5.1.1 Service Layer

Το service layer είναι υπεύθυνο για την επικοινωνία μεταξύ των κομματιών server-side και client-side. Δημιουργεί και προσφέρει μια ομάδα υπηρεσιών με σκοπό τη χρήση τους από τα διάφορα κομμάτια της client-side εφαρμογής [11]. Στην ουσία αποτελεί τον ενδιάμεσο σταθμό μεταξύ των δύο κομματιών server και client, αφού χρησιμοποιεί συναρτήσεις από το business logic κομμάτι της εφαρμογής με σκοπό τη δημιουργία RESTful υπηρεσιών [12,13] για το διαμοιρασμό των δεδομένων σε μορφή JSON.

- **CRUD Services:** Χρησιμοποιούνται από τις εφαρμογές καθώς και από τα ενσωματωμένα συστήματα των σταθμών των καλλιεργειών με στόχο τη δημιουργία νέων μετρήσεων, τη μορφοποίηση των τιμών των μετρήσεων, την επεξεργασία και τη διαγραφή συγκεκριμένων οντοτήτων του σχήματος βάσης.
- **Embedded Services:** Χρησιμοποιούνται από τα ενσωματωμένα συστήματα των σταθμών των καλλιεργειών με στόχο τη δημιουργία νέων μετρήσεων ανά σταθμό και για την ενημέρωση τους για την αλλαγή των συνθηκών νέας μέτρησης και νέας άρδευσης.
- **Report Services:** Δημιουργούν και εκθέτουν τα δεδομένα της εφαρμογής σε μορφή αναφορών.

- **External Access Services:** Χρησιμοποιούνται από την εφαρμογή για την επικοινωνία με υπηρεσίες με στόχο την ανάκτηση δεδομένων από εξωτερικές πηγές (π.χ. πρόγνωση καιρού).



Σχήμα 5.1: Αρχιτεκτονική Συστήματος web εφαρμογής

5.1.2 Business Logic Layer

Αποτελεί τη ραχοκοκαλιά της εφαρμογής αφού περιέχει όλες τις βασικές λειτουργίες. Συνδέεται με τις λειτουργικότητες των υπηρεσιών δεδομένων.

- **Business Management:** Είναι υπεύθυνο για τη διαχείριση όλων των δεδομένων και όλων των κλήσεων από και προς τη βάση δεδομένων.
- **Event Management:** Είναι υπεύθυνο για τη διαχείριση όλων των εναυσμάτων που δημιουργούνται από την ίδια την εφαρμογή και έχουν σχέση με το χρήστη.

5.1.3 Data Layer

Το data layer κομμάτι της εφαρμογής χρησιμοποιείται για την επικοινωνία με τη βάση δεδομένων. Το κομμάτι αυτό χρησιμοποιεί διαφορετικές κλάσεις ανά μοντέλο της βάσης δεδομένων είτε για την ανάκτηση δεδομένων είτε για την αποθήκευση νέων ή ανανέωση των ήδη υπαρχόντων. Είναι το μόνο κομμάτι της εφαρμογής που επικοινωνεί με τη βάση δεδομένων του συστήματος.

5.2 Client Side

Το Client Side κομμάτι της εφαρμογής είναι εξ ολοκλήρου υπεύθυνο για την επικοινωνία του χρήστη με την εφαρμογή καθώς και για την επικοινωνία με το server side κομμάτι της εφαρμογής. Για να επιτευχθεί η επικοινωνία των επιμέρους κομματιών της αρχιτεκτονικής μεταξύ τους υιοθετήθηκαν σαν τρόποι σχεδίασης τα MVC και component based πρότυπα σχεδίασης [14,15]. Η χρήση του έχει πλεονεκτήματα ως προς τη δημιουργία αυτοματοποιημένων δοκιμών, την επαναχρησιμοποίηση μεγάλων κομματιών κώδικα μεταξύ διαφορετικών σελίδων της εφαρμογής, το διαχωρισμό του business logic με το user interface. Παρακάτω αναλύεται το κάθε μέρος της αρχιτεκτονικής ξεχωριστά.

5.2.1 Application Manager

Το κομμάτι αυτό είναι υπεύθυνο για την επικοινωνία με το διακομιστή καθώς χρησιμοποιεί τις υπηρεσίες που εκθέτονται στο κομμάτι του Service Layer για τη μεταφορά δεδομένων από τη βάση δεδομένων ή άλλες πηγές.

5.2.2 Model

Τα Μοντέλα αποτελούν τα αντικείμενα (business objects) της εφαρμογής. Αυτά που διαχειρίζεται ο controller και αυτά που βλέπει ο χρήστης από το User Interface (U.I) της εφαρμογής. Έτσι όταν η εφαρμογή χρειάζεται να παρουσιάσει κάποια στοιχεία από το διακομιστή, αιτείται τα κατάλληλα δεδομένα μέσω των κατάλληλων υπηρεσιών και στη συνέχεια τα εκθέτει στο χρήστη μέσω του κατάλληλου U.I. Αντίστοιχα όταν ο χρήστης θέλει να επεξεργαστεί ή να δημιουργήσει νέα δεδομένα το μοντέλο μεταφέρεται στο διακομιστή, μεταφράζεται στο αντίστοιχο αντικείμενο και στη συνέχεια χρησιμοποιεί τις συναρτήσεις του business logic κομματιού.

5.2.3 View

Είναι υπεύθυνο για τη παρουσίαση των πληροφοριών στην πλευρά του user interface. Κάθε view ελέγχει έναν ορισμένο αριθμό από HTML templates και handlers που είναι υπεύθυνα για να μεταφράζουν τις ενέργειες του χρήστη καθώς και να ορίζουν την παρουσίαση των widgets.

5.2.4 Controller

Οι controllers είναι τα κομμάτια εκείνα που αλληλεπιδρούν με το view και το model και διαχειρίζονται τις κλήσεις του χρήστη. Αναλυτικότερα είναι το κομμάτι που διαχειρίζεται όλο το business logic της εφαρμογής. Συνδέεται με το αντίστοιχο view με σχέση 1 προς 1 και είναι

υπεύθυνο για την αλλαγή των στοιχείων του μοντέλου. Υπάρχουν περιπτώσεις που ένας controller αλληλεπιδρά με άλλους δημιουργώντας πιο περίπλοκα widget. Συνήθως η σχέση αυτή αποτυπώνεται σαν παιδί-πατέρας, έχοντας μη αμφίδρομη σχέση.

5.2.5 Components

Η χρήση components έχει σαν στόχο την καλύτερη οργάνωση του κώδικα καθώς και την επαναχρησιμοποίηση τους όπου αυτό κρίνεται απαραίτητο. Γίνεται πιο εύκολα μια εφαρμογή παραμετροποιήσιμη πραγματοποιώντας αλλαγές όπου κρίνεται απαραίτητο και όχι στο σύνολο της εφαρμογής. Έτσι τα components που αφορούν στο μενού του χρήστη συμμετέχουν σε κάθε σελίδα της εφαρμογής.

5.2.6 Router

Το router είναι υπεύθυνο για να «ζευγαρώσει» τα αντίστοιχα Uniform Resource Locators (URLs) με τα αντίστοιχα components της εφαρμογής. Αποτελεί ένα από τα πιο σημαντικά στοιχεία των Single Page Application (SPA) καθώς είναι υπεύθυνα για τη διατήρηση της ίδιας σελίδας και τη μη ανανέωση του browser. Όταν ένα URL αλλάζει το router αναλαμβάνει να αναλύσει το καινούργιο μονοπάτι και χειρίζεται τη μετάβαση στο νέο view.

6. Βασική Υλοποίηση

Η λειτουργικότητα και η αρχιτεκτονική του συστήματος των εφαρμογών έχει περιγραφεί στο προηγούμενο κεφάλαιο. Στο κεφάλαιο αυτό αναλύονται σε περισσότερο βάθος και παρουσιάζονται κάποια παραδείγματα χρήσης κώδικα. Για να επιτευχθεί αυτό το παρόν κεφάλαιο χωρίζεται σε δύο μεγάλες ενότητες, την ενότητα που περιγράφει τη λειτουργικότητα του server και αυτή που περιγράφει το client. Για την ανάπτυξη κώδικα με τη χρήση ενός (version control) συστήματος χρησιμοποιήθηκε το GitHub ¹.

6.1 Server Side

Το server κομμάτι της εφαρμογής έχει υλοποιηθεί ακολουθώντας τους κανόνες του spring framework.

6.1.1 Αποθήκευση δεδομένων

Για την αποθήκευση των δεδομένων έχει χρησιμοποιηθεί η σχεσιακή βάση δεδομένων Postgres. Όπως αναφέρθηκε παραπάνω τα πλεονεκτήματα από τη χρήση αυτή είναι πολλαπλά· καθώς παρέχει όλα τα απαραίτητα εργαλεία (indices, unique constraints κ.α.) για τη δημιουργία ενός σχήματος. Το σχήμα κατασκευάστηκε ακολουθούμενο βασικούς κανόνες σχεδίασης και δίνοντας βάρος στην ευελιξία. Το hibernate ORM framework παρέχει τη δυνατότητα χρήσης του συνόλου των εντολών SQL μέσω της γλώσσας HQL, που είναι πιο εύκολη στη χρήση. Έτσι αποφεύγονται τα μεγάλα και δυσνόητα queries. Επίσης οι πίνακες στη βάση αντιστοιχίζονται με κλάσεις δίνοντας τη δυνατότητα χρήσης των δυνατοτήτων της αντικειμενοστραφούς γλώσσας.

Η επέκταση Postgis [16] σε επίπεδο βάσης δεδομένων παρέχει τη δυνατότητα αποθήκευσης και χρήσης πολύπλοκων δομών δεδομένων. Επίσης παρέχεται η δυνατότητα για γεωγραφικά ερωτήματα στη βάση όσον αφορά τα σημεία ενδιαφέροντος και τις καλλιέργειες π.χ. τα σημεία που βρίσκονται στο εσωτερικό ενός πολυγώνου ή ενός κύκλου, καλλιέργειες που ανήκουν σε μια γεωγραφική ή διοικητική διαίρεση (π.χ. δήμος, κοινότητα, περιφέρεια). Οι γεωγραφικοί δείκτες (spatial indices) κάνουν την ανάκτηση, διαχείριση και αποθήκευση τέτοιων γεωγραφικών δεδομένων γρήγορη και πιο ελαφριά. Στο σύστημα που αναλύεται η βάση δεδομένων λειτουργεί σε ένα docker προσφέροντας πιο μεγάλη ασφάλεια και πιο μεγάλη ευελιξία και ταχύτητα αφού λειτουργεί απομονωμένη από τις υπόλοιπες λειτουργίες του εικονικού μηχανήματος (vm).

6.1.2 RESTful Web Services

Για να μπορούν οι εφαρμογές που υλοποιήθηκαν και άλλες εξωτερικές να ανακτήσουν τα δεδομένα από το διακομιστή υλοποιήθηκε ένα API το οποίο χρησιμοποιεί RESTful υπηρεσίες [17]. Κάποια από τα πλεονεκτήματα χρήσης αυτού του πρωτοκόλλου έναντι άλλων όπως είναι το SOAP (Simple Object Access Protocol) είναι τα εξής:

- Χρήση των πρότυπων HTTP κλήσεων.
- Πιο εύκολο στη χρήση του αφού απαιτείται μικρότερο bandwidth για τα data που μεταφέρονται από και προς το διακομιστή. Στην περίπτωση του συστήματος που υλοποιήθηκε, επειδή το ενσωματωμένο χρησιμοποιεί 4g σύνδεση για την

¹ <https://github.com/>

επικοινωνία με το διακομιστή και το bandwidth που δίνεται από τους παρόχους κινητής τηλεφωνίας είναι περιορισμένο, η εξοικονόμηση δεδομένων αποτέλεσε κύρια μέριμνα κατά την υλοποίηση του συστήματος.

- Το REST πρωτόκολλο επιτρέπει τη χρήση πολλών τύπων δεδομένων ενώ το SOAP επιτρέπει τη χρήση μόνο xml.
- Κάνει την επικοινωνία με τους browsers πιο εύκολη λόγω της χρήσης JSON.
- Υλοποίηση CRUD διαδικασιών χωρίς τη χρήση της κατάστασης τους (stateless) .
- Οι υπηρεσίες REST μπορούν να χρησιμοποιήσουν cache.
- Οι RESTful υπηρεσίες καλούνται μέσω το client side της εφαρμογής και όχι από τον ίδιο το χρήστη. Αυτό σημαίνει ότι ο χρήστης δεν βλέπει απευθείας το endpoint στο οποίο κάνει την κλίση.

Βέβαια η χρήση της αρχιτεκτονικής REST απέναντι στη χρήση SOAP δεν είναι πάντα η βέλτιστη για όλους τους τύπους εφαρμογών. Αυτό εξαρτάται από πολλές παραμέτρους όπως είναι η ασφάλεια και η χρήση που θέλει κάποιος να έχει μια εφαρμογή που αναπτύσσει. Για τη χρήση και την αλληλεπίδραση με τα API χρησιμοποιήθηκε το swagger UI. Πρόκειται για ένα λογισμικό ανοιχτού κώδικα που παράγεται απευθείας από τα services και δίνει τη δυνατότητα σε ομάδες να αλληλεπιδρούν με τα webservices χωρίς να έρχονται σε επαφή με τη λογική των επιμέρους κομματιών κώδικα. Για τη web εφαρμογή το UI για την αλληλοεπίδραση με τα web services που καταναλώνονται βρίσκονται στο URL: <http://78.46.70.93:8086/swagger-ui.html>. Ενώ τα αντίστοιχα services που καταναλώνονται στη mobile εφαρμογή <http://78.46.70.93:5000/swagger> [18]. Στο Σχήμα 6.1 φαίνονται τα web services που χρησιμοποιήθηκαν για τον embedded-controller ο οποίος είναι υπεύθυνος για την επικοινωνία με το σταθμό της καλλιέργειας, ενώ ο extract-controller περιέχει τα endpoints που παράγουν τα αρχεία αναφοράς μετρήσεων φυσικών φαινομένων και άρδευσης. Στο Σχήμα 6.2 παρουσιάζονται τα web services που χρησιμοποιούνται για όλη την εφαρμογή ενώ στο Σχήμα 6.3 φαίνονται αυτά που χρησιμοποιούνται για πιστοποίηση χρηστών, και για τα υπόλοιπα δεδομένα των χρηστών. Στο Σχήμα 6.4 αναπαρίστανται τα endpoints του API που καταναλώνει η Android εφαρμογή.

embedded-controller : Embedded Controller			Show/Hide	List Operations	Expand Operations
GET	/embedded/endDevicesTime	getAllDevicesWateringTime			
POST	/embedded/irrigation	saveAutomaticWateringObservation			
POST	/embedded/measures	postSensor			
GET	/embedded/station/{coordinator}/setup	getSetup			
GET	/embedded/{coordinator}/enddevices	getAllDevicesAddress			
GET	/embedded/{coordinator}/irrigation	getMeasuringIrrigation			
extract-controller : Extract Controller			Show/Hide	List Operations	Expand Operations
POST	/extract/{mydevice}/watering/csv	getCsv			
POST	/extract/{mydevice}/watering/pdf	getPDF			
POST	/extract/{mydevice}/watering/xls	getXlsx			
POST	/extract/{mydevice}/{observablepropertyid}/csv	getCsv			
POST	/extract/{mydevice}/{observablepropertyid}/pdf	getPDF			
POST	/extract/{mydevice}/{observablepropertyid}/xls	getXlsx			

Σχήμα 6.1 embedded και extract web services

home-controller : Home Controller

Show/Hide | List Operations | Expand Operations

POST	/automaticwater/dates	saveAutomaticWatering
GET	/charthome/{id}	getChartByDevice
GET	/enddevices	getFeatureEndDevices
GET	/features/profile	getProfile
POST	/features/profile	postSensor
GET	/measures/lastdate	getMeasuresByObsProperty
GET	/observableproperties	getObsProperties
GET	/station/coords	getStationCoords
GET	/wateringprofile/minmax	getStationMinMaxValues
POST	/wateringprofile/minmax	saveWateringProfile
GET	/{{coordinator}}/automaticwater/dates	getAutomaticWaterTimes
GET	/{{coordinator}}/measures	startMeasuring
POST	/{{mydevice}}/irrigation/times	setIrrigationDates
GET	/{{mydevice}}/measures/last	getMeasuresByObsProperty
GET	/{{mydevice}}/watering/last	getLastWateringMeasures
GET	/{{mydevice}}/watering/measures	getMeasuresByObsProperty
GET	/{{mydevice}}/{observablePropertyId}/measures	getMeasuresByObsProperty
GET	/{{mydevice}}/{observablePropertyId}/measures/counter	getTotalMeasuresCounter

Σχήμα 6.2 Home controller web services

index-controller : Index Controller

Show/Hide | List Operations | Expand Operations

GET

/

getIndexPage

POST

/login

validate

POST

/logout

signout

GET

/topbaruser

getObsProperties

user-controller : User Controller

Show/Hide | List Operations | Expand Operations

GET

/user/notifications

getCounterNotifications

PUT

/user/notifications/all/read

saveUserProfile

PUT

/user/notifications/{id}/read

saveUserProfile

GET

/user/profile

getUserProfile

POST

/user/profile

saveUserProfile

Σχήμα 6.3 User, Authenticate web services

Automatic irrigation WebApi

Device		Show/Hide	List Operations	Expand Operations
GET	/api/devices			
GET	/api/devices/{identifier}/measuredata			
GET	/api/devices/{identifier}/{observableproperty}/measuredata			
GET	/api/devices/{identifier}/lastmeasuredata			
GET	/api/devices/minmaxvalues			
GET	/api/devices/{identifier}/lastvalues			
POST	/api/devices/{identifier}/measuring			
POST	/api/devices/{identifier}/watering			
Identity		Show/Hide	List Operations	Expand Operations
GET	/identity/login			
GET	/identity/profile/{user_id}			

Σχήμα 6.4 Web services για Android application

6.1.3 Ασφάλεια

Η ασφάλεια είναι σημαντικό στοιχείο για τις web εφαρμογές. Πολλές σύγχρονες εφαρμογές δίνουν αρκετό βάρος στο τομέα της ασφάλειας. Με την υλοποίηση και μετέπειτα χρήση όλο και περισσότερων APIs, κρίνεται επιτακτική ανάγκη η προσθήκη ασφάλειας στην κατανάλωση των web services. Στη παρούσα εφαρμογή χρησιμοποιήθηκε η αρχιτεκτονική ασφαλείας που βασίζεται στη χρήση cookie. Το cookie αυτό δημιουργείται χρησιμοποιώντας το αντίστοιχο endpoint για login χρήστη χρησιμοποιώντας username και κωδικό. Ο κύκλος ζωής του ολοκληρώνεται όταν ο χρήστης αποσυνδεθεί από την εφαρμογή χρησιμοποιώντας δηλαδή το αντίστοιχο endpoint αποσύνδεσης. Όταν ο χρήστης δεν έχει ενεργοποιημένο cookie δεν μπορεί να χρησιμοποιήσει τα αντίστοιχα endpoints της εφαρμογής. Αν ο χρήστης δεν είναι διαπιστευμένος τότε επιστρέφεται στο χρήστη HTTP Status με κωδικό 401 που μεταφράζεται σε Unauthorized και ο κώδικας δεν προχωράει περαιτέρω στην εκτέλεση του. Η εφαρμογή διαθέτει επίσης μηχανισμό αυτόματου αποκλεισμού μετά από κάποια συγκεκριμένη χρονική διάρκεια. Έτσι καταστρέφεται αυτόματα το cookie. Όσον αφορά την περίπτωση των web services που καταναλώνονται από το σταθμό της καλλιέργειας αυτά δεν χρησιμοποιούν το αντίστοιχο cookie αλλά τις διευθύνσεις των συσκευών ZigBee του σταθμού της καλλιέργειας. Αν η οκταδική συμβολοσειρά, που αντιστοιχεί στη διεύθυνση μνήμης του ZigBee, ισούται με το κατάλληλο πεδίο στον πίνακα featureofinterest τότε το κομμάτι κώδικα του endpoint εκτελείται κανονικά αλλιώς επιστρέφεται HTTP Status Unauthorized.

6.2 Client Side – Web Application

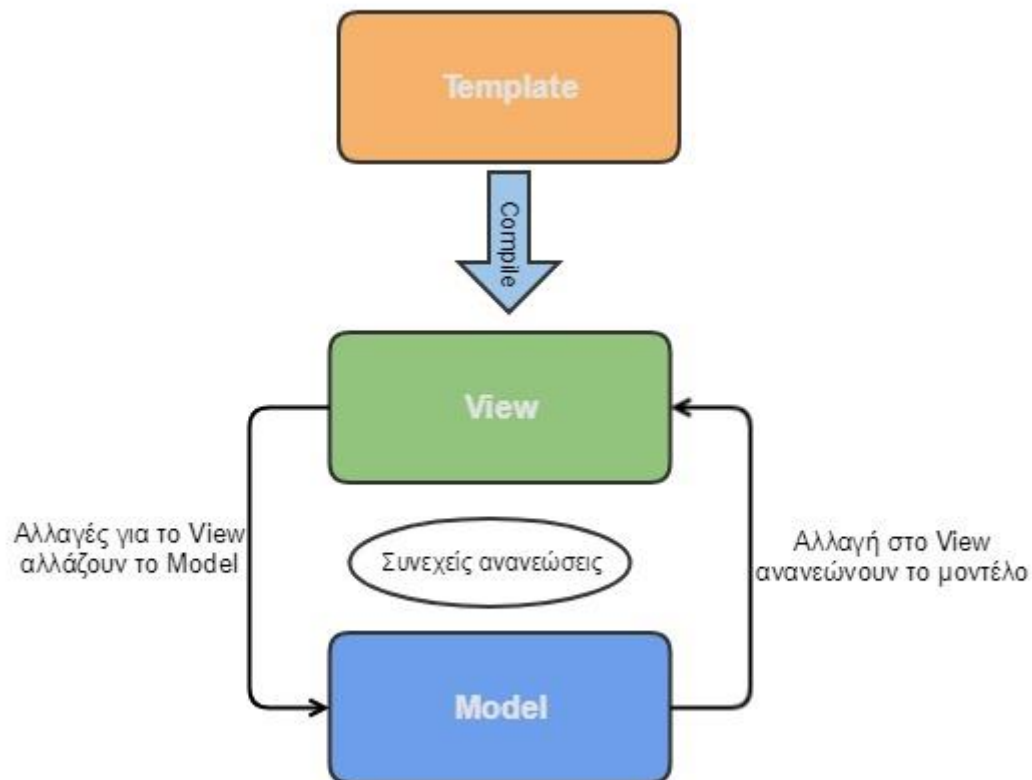
Οι δύο εφαρμογές που χρησιμοποιούν client side έχουν χτιστεί σύμφωνα με πρότυπα σχεδίασης και ανάπτυξη κώδικα. Στη web εφαρμογή έχει γίνει χρήση της γλώσσας Javascript

και συγκεκριμένα του AngularJS framework. Η υλοποίηση βασίζεται στο MVC πρότυπο ανάπτυξης ακολουθώντας τους κανόνες που αυτό επιτάσσει. Για το UI χρησιμοποιήθηκε HTML5 και CSS3 σε συνδυασμό με τα templates και τα directives της AngularJS. Επίσης έγινε χρήση ανοιχτών βιβλιοθηκών για τη θέαση και την επεξεργασία δεδομένων. Στα παρακάτω κεφάλαια περιγράφονται οι βιβλιοθήκες και οι χρήσεις τους.

6.2.1 Αρχιτεκτονικές

Web Application MVC

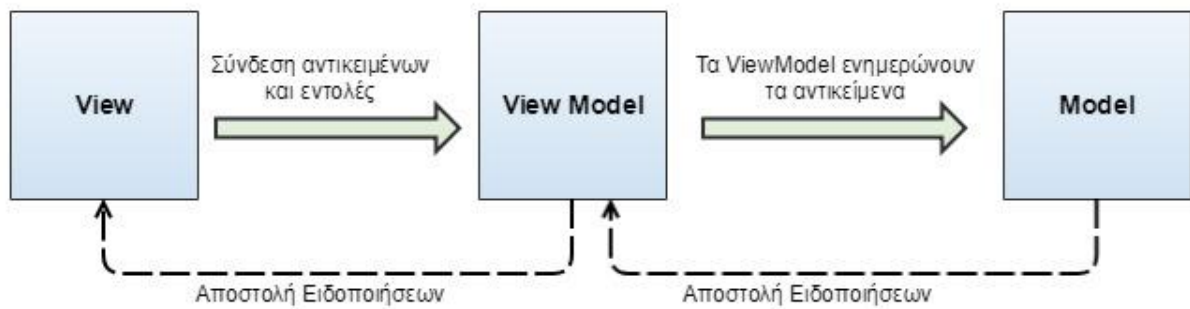
Όπως αναφέρθηκε η υλοποίηση βασίζεται στο MVC πρότυπο. Τα τελευταία χρόνια με την άνοδο των web applications έχουν δημιουργηθεί πολλά Javascript MVC frameworks. Η κύρια ανάγκη που οδήγησε σε αυτή τη δημιουργία είναι η προσπάθεια για πιο αυστηρές ορισμούς στη χρήση της γλώσσας. Στην περίπτωση της συγκεκριμένης εφαρμογής χρησιμοποιήθηκε το AngularJS Javascript framework. Το συγκεκριμένο framework είναι αρκετά γρήγορο σε σχέση με άλλα παρόμοιας αρχιτεκτονικής και ευέλικτο αφού ο προγραμματιστής μπορεί να παραμετροποιήσει ή να δημιουργήσει καινούρια βασικά συστατικά (directives, templates, pipes, events κ.α.) σύμφωνα με τις ανάγκες της κάθε εφαρμογής. Ο τρόπος λειτουργίας είναι ο εξής ο controller της εφαρμογής αρχικοποιεί το view και αλλάζει το Model. Με τη σειρά του το View παρακολουθεί τις αλλαγές στο UI και ενημερώνει το Model. Η λειτουργικότητα αυτή ονομάζεται two-way-binding και χρησιμοποιείται κατά κόρον στις CRUD εφαρμογές. Το View επίσης αναλαμβάνει την ανανέωση των DOM Objects [19] όποτε αυτό είναι απαραίτητο όταν δηλαδή αλλάζουν οι τιμές του model ή ενεργοποιούνται κάποια εναύσματα που έχουν οριστεί από το χρήστη της εφαρμογής.



Σχήμα 6.5 Two way binding αρχιτεκτονική

Android App MVVM

Η αρχιτεκτονική που προτιμήθηκε για την Android εφαρμογή είναι αυτή του MVVM [20]. Αποσκοπεί στο διαχωρισμό της ανάπτυξης κώδικα για το user interface με την ανάπτυξη για το business logic. Για το UI συνήθως χρησιμοποιείται κάποια markup γλώσσα προγραμματισμού (XML, XAML, YAML κ.α.). Στην περίπτωση του Xamarin που χρησιμοποιήθηκε στην παρούσα διπλωματική η γλώσσα που έχει ενσωματώσει για την ανάπτυξη του UI είναι η Extensible Application Markup Language (XAML). Το «View Model» κομμάτι της αρχιτεκτονικής είναι υπεύθυνο για την μετατροπή των κλάσεων μοντέλων έτσι ώστε να είναι εύκολα στη διαχείριση. Έτσι το περισσότερο κομμάτι της business λογικής ενός project βρίσκεται στο κομμάτι αυτό. Το «View Model» δρα περισσότερο σαν View και λιγότερο σαν Model. Η σχέση των αντικειμένων που υπάρχουν στο model με αυτά που υπάρχουν στο view υποστηρίζουν τη σχέση two-way-binding. Έτσι όπως συμβαίνει και στην MVC αρχιτεκτονική το «View Model» είναι υπεύθυνο για την αρχικοποίηση και τη διαχείριση των models. Το view αλληλεπιδρά με το χρήστη μέσω του UI και αλλάζει τις τιμές του δεσμευμένου «Bound» μοντέλου, ενώ οι αλλαγές που προέρχονται μέσα από εναύσματα αλλάζουν τη συμπεριφορά του View άρα και του UI. Στο Σχήμα 6.6 φαίνεται η αρχιτεκτονική MVVM.



Σχήμα 6.6 Αρχιτεκτονική Model View View-Model

6.2.2 Router

Για την κατασκευή σύγχρονων web εφαρμογών στα πρότυπα των Rich Internet Applications (RIA) ώστε να προσομοιάζουν τις desktop εφαρμογές και τη λειτουργικότητα, έχουν αναπτυχθεί τα τελευταία χρόνια SPA (Single Page Applications) εφαρμογές. Έτσι η περιήγηση του χρήστη σε διαφορετικούς διευθύνσεις μέσα στην εφαρμογή δεν προϋποθέτει την ανανέωση της σελίδας (View) από το server. Η εφαρμογή χρησιμοποιεί μια μόνο σελίδα όπου οι τιμές των περιεχομένων αλλάζουν ανάλογα με τις κλήσεις στο server. Η υλοποίηση αυτή βοηθά πολύ στο user experience των εφαρμογών ώστε να μην αποσπάται ο χρήστης από την ίδια την εφαρμογή. Οι αλλαγές στα δεδομένα των περιεχομένων γίνεται επικοινωνώντας με το server side κομμάτι των εφαρμογών ή με εξωτερικά APIs μέσα στο ίδιο view. Στη περίπτωση της AngularJS όπως έχει περιγραφεί παραπάνω, χρησιμοποιεί two way binding για την ενημέρωση του model και του view (επικοινωνούν μεταξύ τους και ενημερώνουν το ένα το άλλο) μέσω του controller. Κατά το compilation δημιουργείται html κώδικας, και αποδίδεται στο μοναδικό view. Έτσι γίνεται δυνατή η υλοποίηση νέων σελίδων χωρίς τη χρήση server side κώδικα.

Στη web εφαρμογή χρησιμοποιείται το ngRoute της AngularJS που σαν σκοπό έχει να συνδέσει τα URLs της εφαρμογής με τους αντίστοιχους controllers και views. Παρακάτω γίνεται παράθεση του κομματιού κώδικα που υλοποιεί την παραπάνω διαδικασία μέσω του ngRoute.

```

module.component("cropApp", {
  templateUrl: '/js/components/crop-app/crop-app.component.html',
  $routeConfig: [
    {path: "/myprofile", component: "myProfile", name: "Profile"},
    {path: "/userprofile", component: "userProfile", name: "UserProfile"},
    {path: "/observableProperty/:id/:description", component: "observableProperty", name: "ObservableProperty"},
    {path: "/wateringProperty", component: "wateringProperty", name: "WateringProperty"},
    {path: "/getweather", component: "weatherApi", name: "WeatherApi"},
    {path: "/controlpanel", component: "controlPanel", name: "ControlPanel"},
    {path: "/wateringprofile", component: "wateringProfile", name: "WateringProfile"},
    {path: "**", component: "firstPage", name: "Firstpage"}
  ]
})

```

6.2.3 User Interface

Το UI των εφαρμογών επιλέχθηκε σύμφωνα με τις συσκευές που μπορεί να τις φιλοξενήσουν.

Web Application

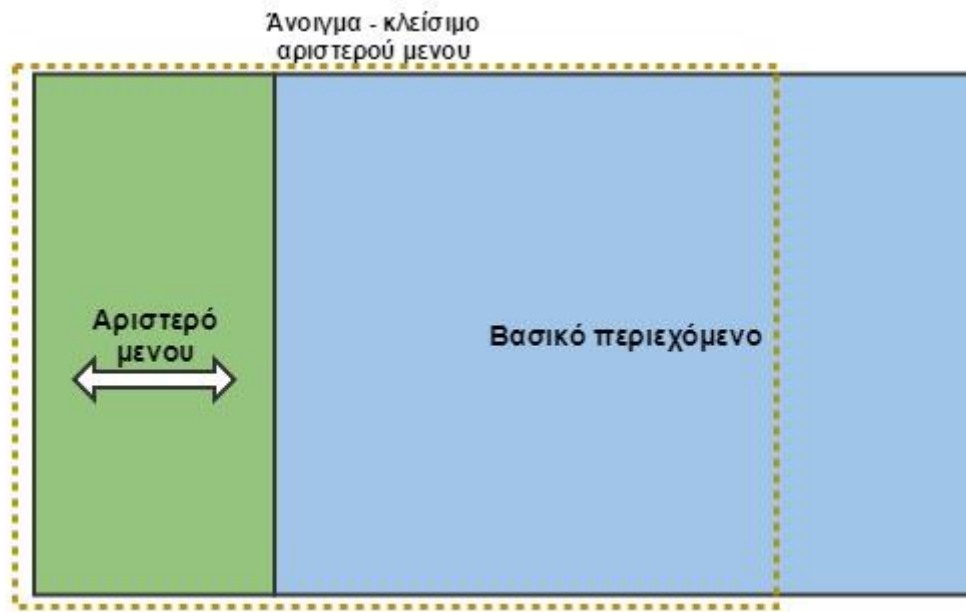
Για τη δημιουργία της web εφαρμογής επιλέχθηκε ο διαχωρισμός της οθόνης σε τρία διαφορετικά μέρη όπως φαίνεται στο Σχήμα 6.7. Το αριστερό μενού το οποίο περιέχει τις επιλογές για τη περιήγηση του χρήστη στις διαφορετικές οθόνες της εφαρμογής. Το βασικό περιεχόμενο όπου εκεί φορτώνονται τα δεδομένα και τα επιμέρους περιεχόμενα των οθονών. Αποτελεί το μεγαλύτερο κομμάτι του UI του χρήστη καθώς εκεί ο χρήστης αλληλεπιδρά με τα δεδομένα και το business logic της εφαρμογής. Ενώ η πάνω μπάρα που εκτείνεται σε όλο το μήκος της οθόνης αποτελεί το μέρος του μενού όπου υπάρχουν οι προσωποποιημένες επιλογές για το χρήστη. Συγκεκριμένα είναι το σημείο που εμφανίζονται οι ειδοποιήσεις ενώ μπορεί ο χρήστης να αλλάξει τα στοιχεία του προφίλ του. Η εφαρμογή μπορεί να φιλοξενηθεί σε μεγάλες οθόνες σταθερών υπολογιστών, φορητών υπολογιστών καθώς και σε ταμπλέτες.



Σχήμα 6.7 Τα τρία μέρη της web εφαρμογής

Android Application

Για το μέγεθος των κινητών έχει δημιουργηθεί ξεχωριστή εφαρμογή. Το UI που χρησιμοποιήθηκε και τα μέρη που αποτελείται φαίνεται στο Σχήμα 6.8. Το αριστερό μενού όπως φαίνεται μπορεί να αυξομειώνεται προς το κομμάτι του βασικού περιεχομένου.



Σχήμα 6.8 Μέρη Android εφαρμογής

6.2.4 Controller

Όπως έχει αναφερθεί οι controllers περιέχουν όλη τη λειτουργικότητα μια εφαρμογής. Είναι Javascript objects που καθορίζονται από μια μέθοδο κονστράκτορα. Στην εφαρμογή οι controllers εισαγάγουν τιμές που λαμβάνουν μέσω των services στα αντίστοιχα μοντέλα και επικοινωνούν με το view μέσω συναρτήσεων και μέσω αντικειμένων που υλοποιούν two-way-binding σχέση. Στη παρούσα εφαρμογή γίνεται χρήση controllers μέσα σε components.

```
controller: function (WateringProfileService, $log, $q, $scope, toastr) {
    var model = this;
    model.obspropminmax = {};

    model.$onInit = function () {
        var defer = $q.defer();

        WateringProfileService.getMinMaxValues().then(
            function (responsedata) {
                model.obspropminmax = responsedata;
                defer.resolve(model.obspropminmax);
            }
        );

        toastr.info("You can set the minimum and maximum values of your
        irrigation system. The irrigation will start when the" +
            "observable property is bigger than max value or smallen than
        less value.", "Information");
    }

    model.saveAllData = function () {
        // console.log(angular.toJson(model.obspropminmax));
        WateringProfileService.saveWateringProfile(model.obspropminmax).then(
```



```
function (response) {
    if (response === true || response == "true") {
        toastr.success('Watering profile saved succesfully',
'Success');
    } else {
        toastr.error('Error on save watering profile data',
'Error');
    }
},
function (errResponse) {
    toastr.error(errResponse, 'Error');
}
);
};
}
```

6.2.5 Services

Τα services χρησιμοποιούνται στην εφαρμογή για να συνδέσουν το server κομμάτι με το client. Χρησιμοποιούνται για να μεταφέρουν δεδομένα από και προς το server του συστήματος καθώς και σε άλλες εξωτερικές πηγές. Η σύνδεση αυτή γίνεται μέσω HTTP κλήσεων. Παρακάτω φαίνεται ένα παράδειγμα χρήσης service.

```
getMinMaxValues: function () {
    return $http.get('wateringprofile/minmax').then(
        function (response) {
            return response.data;
        },
        function (errResponse) {
            console.error('Error while fetchingService user
profile!!!');
            return $q.reject(errResponse);
        }
    );
},
saveWateringProfile: function (wateringprofiledata) {
    console.log(angular.toJson(wateringprofiledata));
    return $http({
        method: 'POST',
        url: 'wateringprofile/minmax',
        data: angular.toJson(wateringprofiledata),
        headers: { 'Content-Type': 'application/json' }
    }).then(function (response) {
        return true;
    },
        function (errResponse) {
            console.error('Error while saving minimum and maximum
values for watering profile!');
            return $q.reject(errResponse);
        }
    );
}
```

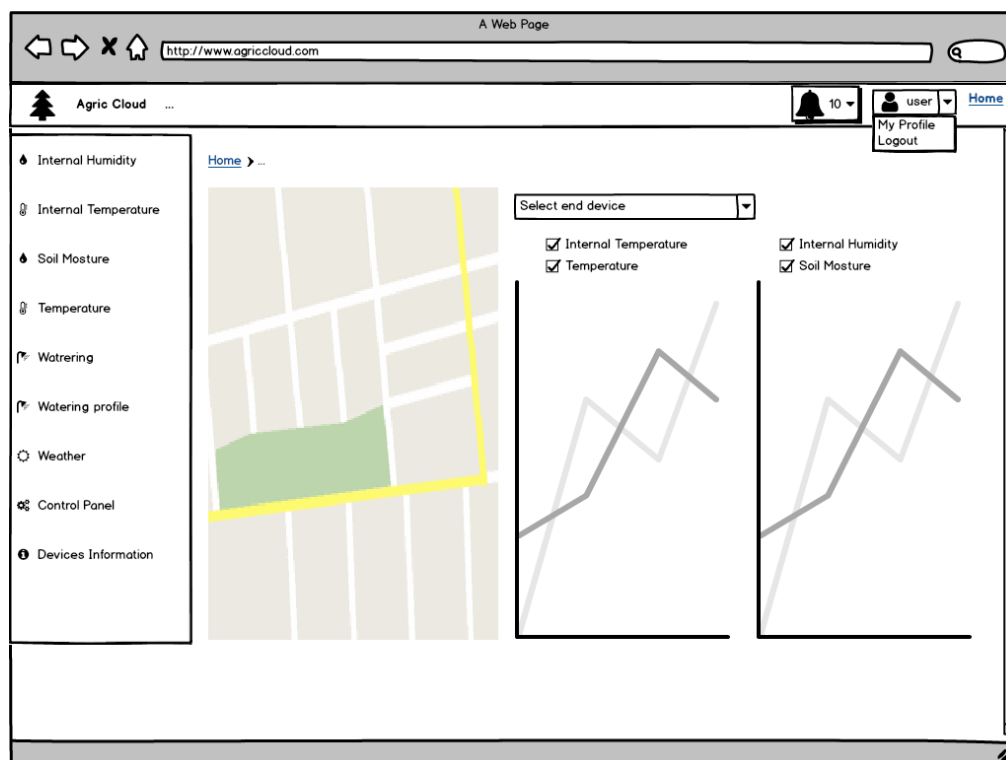

6.2.6 Χάρτες, γραφήματα και πίνακες δεδομένων

Για τη δημιουργία γραφικών παραστάσεων για τη θέαση των δεδομένων στον άξονα του χρόνου χρησιμοποιήθηκε μια επέκταση της Javascript βιβλιοθήκης D3JS [21]. Η επέκταση έχει αναπτυχθεί ώστε να υποστηρίζει τη χρήση του AngularJS framework. Για τη θέαση των δεδομένων σε μορφή πινάκων, τη σελιδοποίηση και την ταξινόμηση τους χρησιμοποιήθηκε η βιβλιοθήκη ngTable [22]. Τέλος για τη θέαση και την επεξεργασία χωρικών δεδομένων σε υπόβαθρα χαρτών χρησιμοποιήθηκε η βιβλιοθήκη angular-OpenLayers-directive [23] που στηρίζεται στο AngularJS framework OpenLayers [24]. Συγκεκριμένα δημιουργεί directives που χρησιμοποιούνται για τα υπόβαθρα χάρτη, με σκοπό την αναπαράσταση και επεξεργασία γεωγραφικών οντοτήτων (σημείο, πολύγωνο κ.α.).

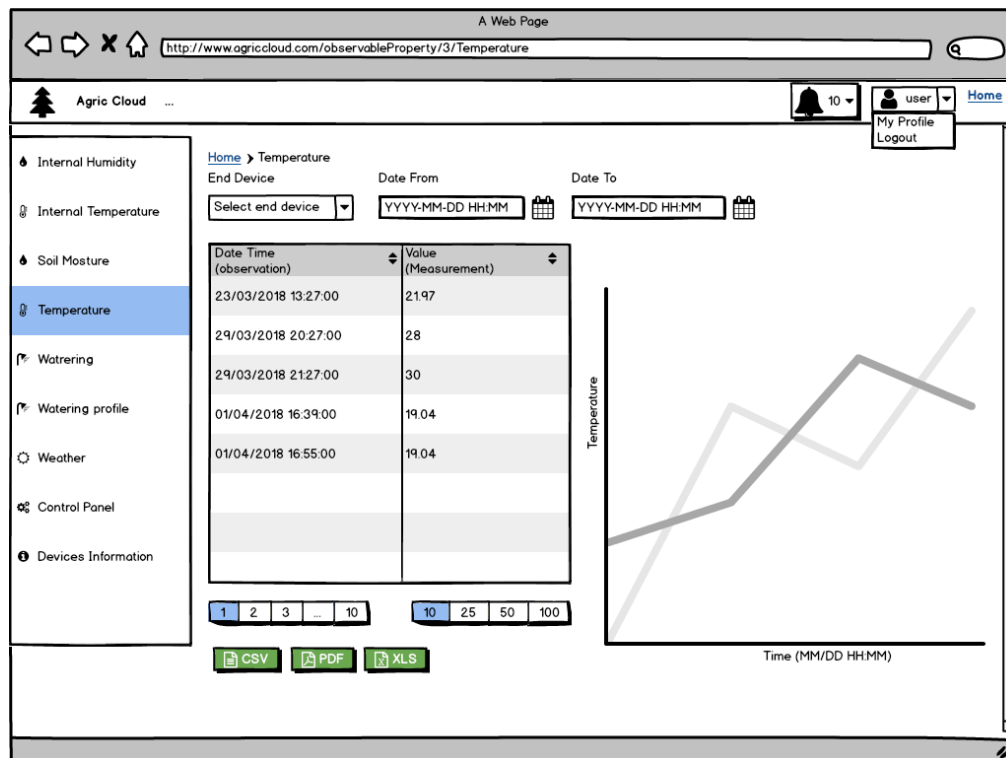
7. Γραφικό περιβάλλον και περιβάλλον χρήστη

Το παρόν κεφάλαιο περιγράφει τη μεθοδολογία που ακολουθήθηκε για τη σχεδίαση και τη δημιουργία του User Interface των εφαρμογών. Αποτελείται από τα σχήματα (storyboards) που σχεδιάστηκαν στα πρώιμα στάδια της υλοποίησης της διεπαφής χρήστη. Κάθε σχήμα αντιπροσωπεύει μια ενέργεια για τον χρήστη των client εφαρμογών. Η δημιουργία των prototypes πραγματοποιήθηκε με τη χρήση του λογισμικού σχεδίασης Balsamiq.

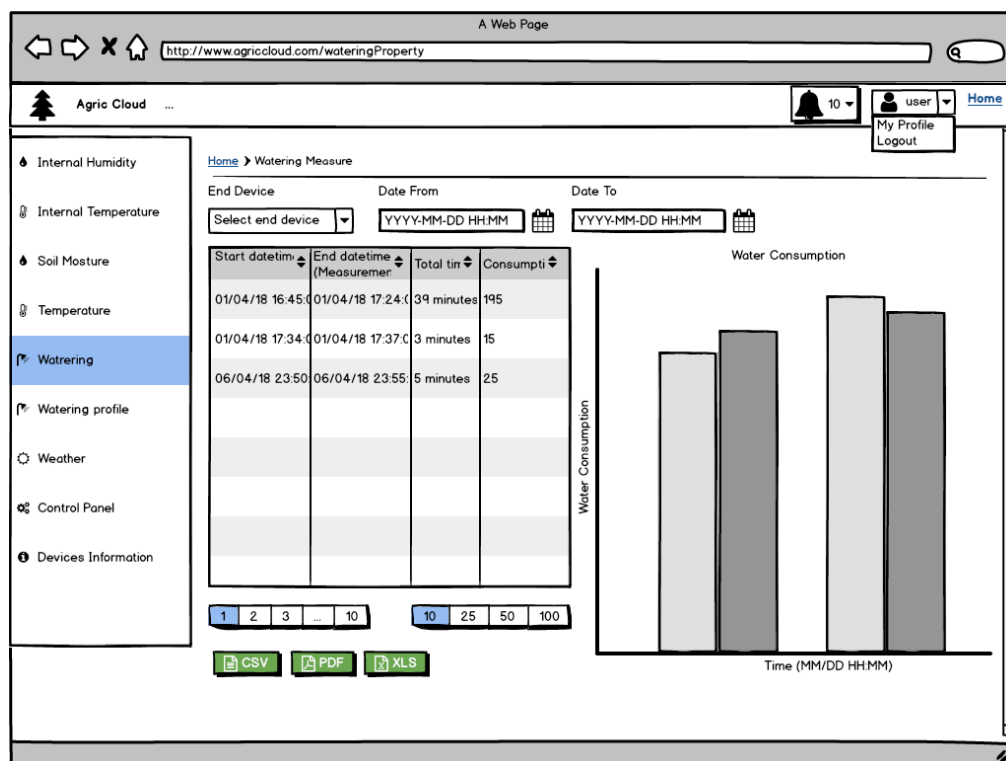
7.1 Prototypes



Σχήμα 7.1: Πρωτότυπο αρχικής σελίδας εφαρμογής. Αριστερά διακρίνεται το μενού δυνατοτήτων και στο πάνω κομμάτι οι επιλογές σε επίπεδο χρήστη



Σχήμα 7.2: Πρωτότυπο απεικόνισης μετρήσεων για επιλεγμένο μετρήσιμο μέγεθος, επιλεγμένη συσκευή μετρήσεων και επιλεγμένο διάστημα ημερομηνιών.



Σχήμα 7.3: Πρωτότυπο απεικόνισης άρδευσης επιλεγμένης συσκευής και επιλεγμένου διαστήματος ημερομηνιών.

Watering Profile

Watering Device 1: <Identifier> - <Name>

Soil Moisture

Minimum Value Maximum Value

Temperature

Minimum Value Maximum Value

Watering Device N: <Identifier> - <Name>

Soil Moisture

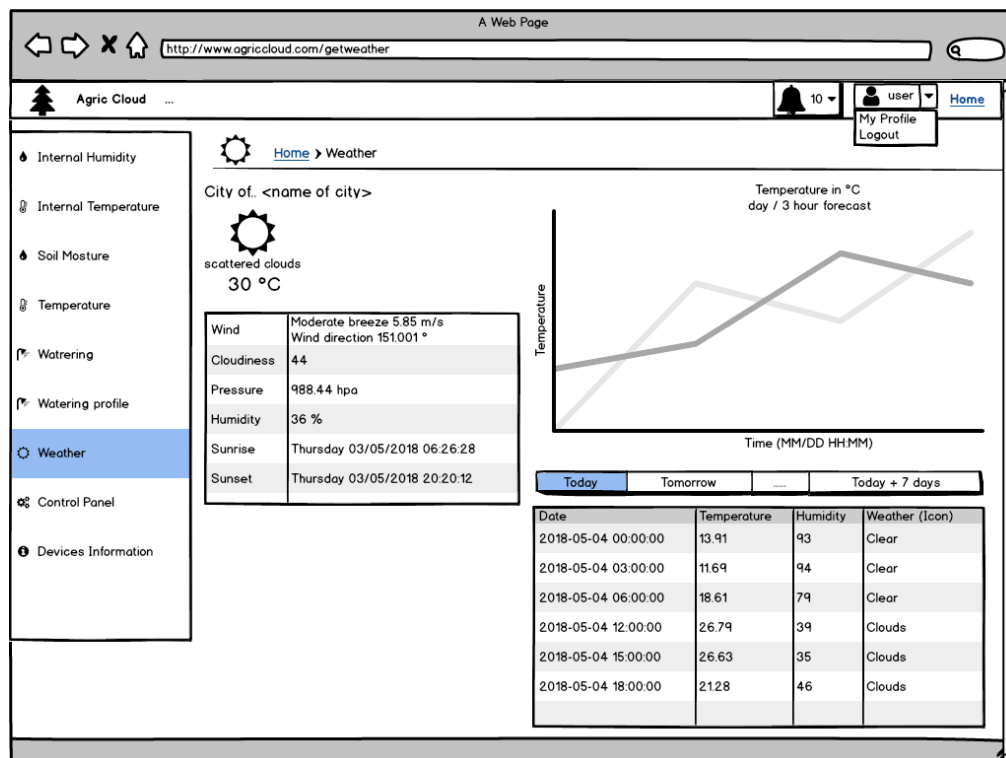
Minimum Value Maximum Value

Temperature

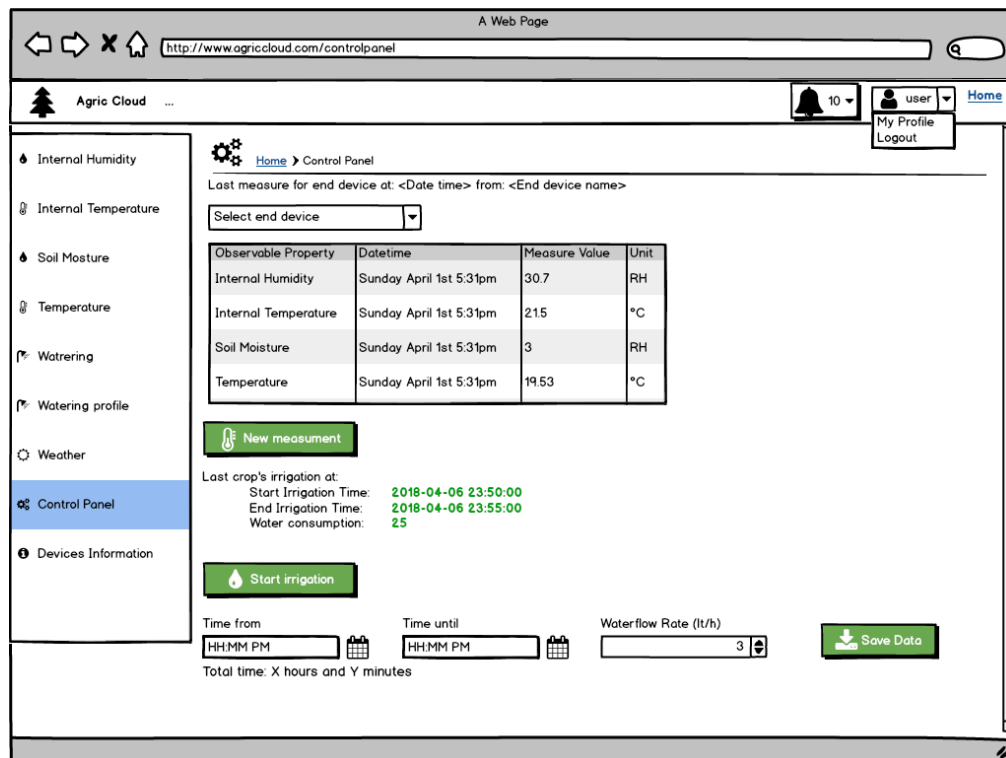
Minimum Value Maximum Value

Save Data

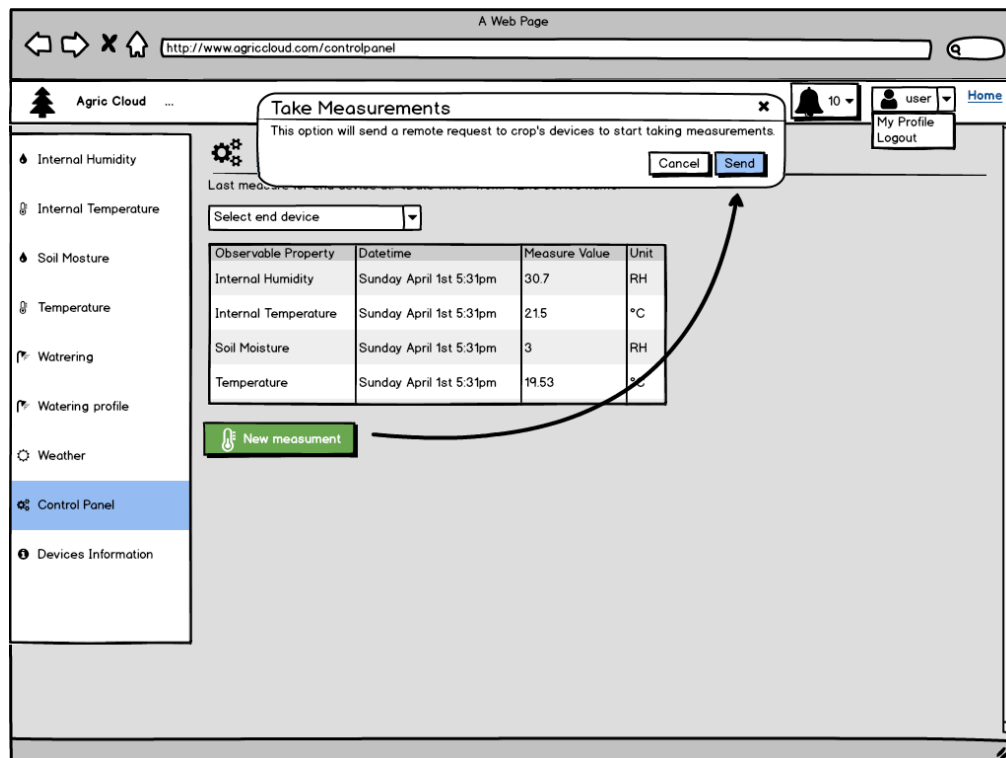
Σχήμα 7.4: Μέγιστες και ελάχιστες τιμές για την έναρξη του ποτίσματος.



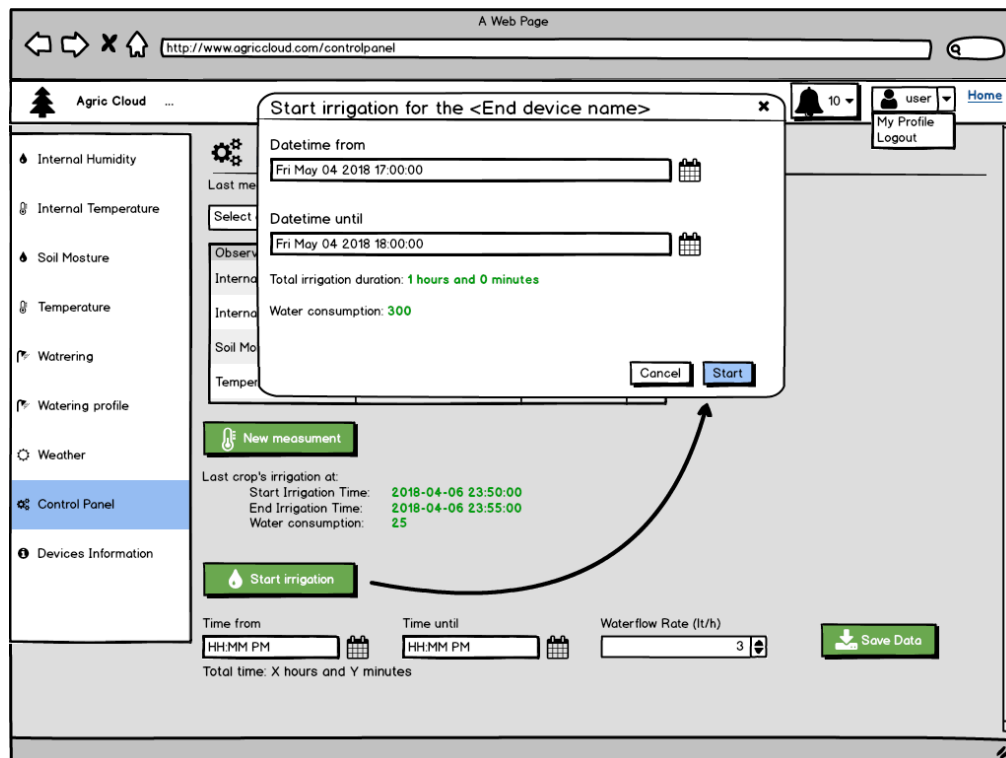
Σχήμα 7.5: Πρόγνωση καιρού για διάστημα 7 ημερών



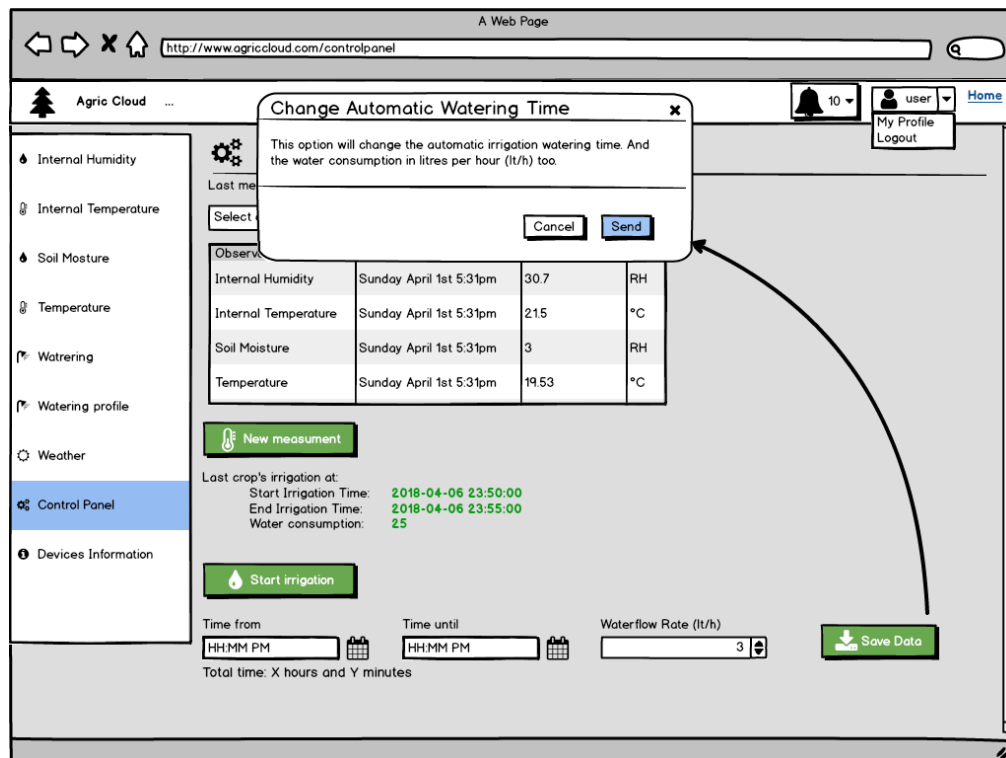
Σχήμα 7.6: Πίνακας ελέγχου και διαχείρισης συσκευών



Σχήμα 7.7: Ενέργεια «Νέες μετρήσεις για όλες τις τελικές συσκευές»



Σχήμα 7.8: Έναρξη άρδευσης για επιλεγμένη συσκευή και επιλεγμένο διάστημα ημερομηνιών.




Σχήμα 7.9: Αλλαγή τιμών εκτέλεσης αλγορίθμου αυτόματης άρδευσης (Διάστημα ημερομηνιών, χωρητικότητα ηλεκτροβάνας)

Settings

Crop

Crop's Identifier: Description:

Station

Name: Description:  add point coordinates for station

End Device 1

Identifier: Description:

Sensors


Name	Measurement Type
LM35	Analog Temperature
DHT22	Digital Measurement
DHT22	Digital Measurement
LM393	Soil Moisture
Hunter PGV 100 GB DC	DC Latching Solenoid

End Device N

Identifier: Description:

Sensors

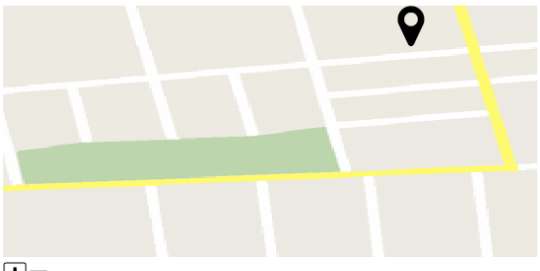
Name	Measurement Type
LM35	Analog Temperature
DHT22	Digital Measurement
DHT22	Digital Measurement
LM393	Soil Moisture
Hunter PGV 100 GB DC	DC Latching Solenoid



Σχήμα 7.10: Επεξεργασία και θέαση πληροφοριών για τις συσκευές του συστήματος αυτόματης υδροδότησης.

Set coordinates

You can set the coordinators of the station



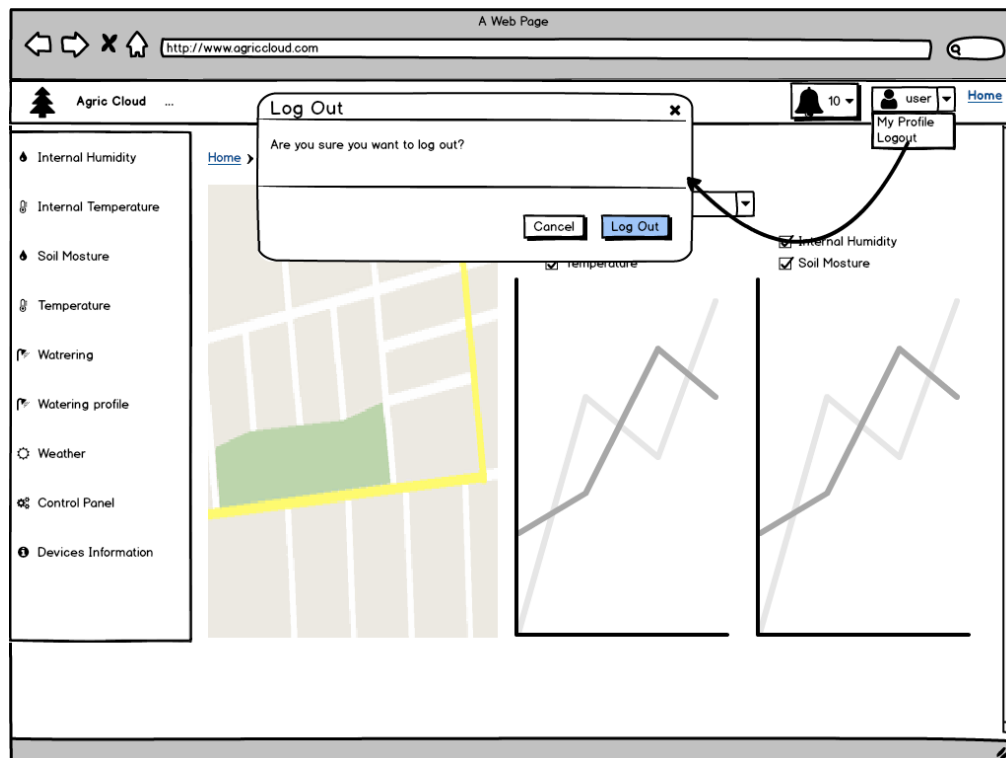
add point coordinates for station

Identifier: Description:

Σχήμα 7.11: Επιλογή συντεταγμένων σημείου τοποθέτησης του σταθμού.

The screenshot shows a web browser window titled 'A Web Page' with the URL 'http://www.agriccloud.com/userprofile'. The page header includes the 'Agric Cloud' logo, a notification bell with '10', a user profile dropdown, and a 'Home' link. A left sidebar contains a list of menu items: Internal Humidity, Internal Temperature, Soil Moisture, Temperature, Watering, Watering profile, Weather, Control Panel, and Devices Information. The main content area is titled 'Home > My Profile' and contains a form for user information. The form fields are: first name, last name, father's name, Street (optional), Number (optional), Zipcode (optional), Telephone (optional), Mobile (optional), and Date of birth (with a calendar icon). At the bottom right of the form are 'Cancel' and 'Save' buttons.

Σχήμα 7.12: Επεξεργασία προσωπικών στοιχείων του χρήστη.



Σχήμα 7.13: Αποσύνδεση χρήστη από την εφαρμογή

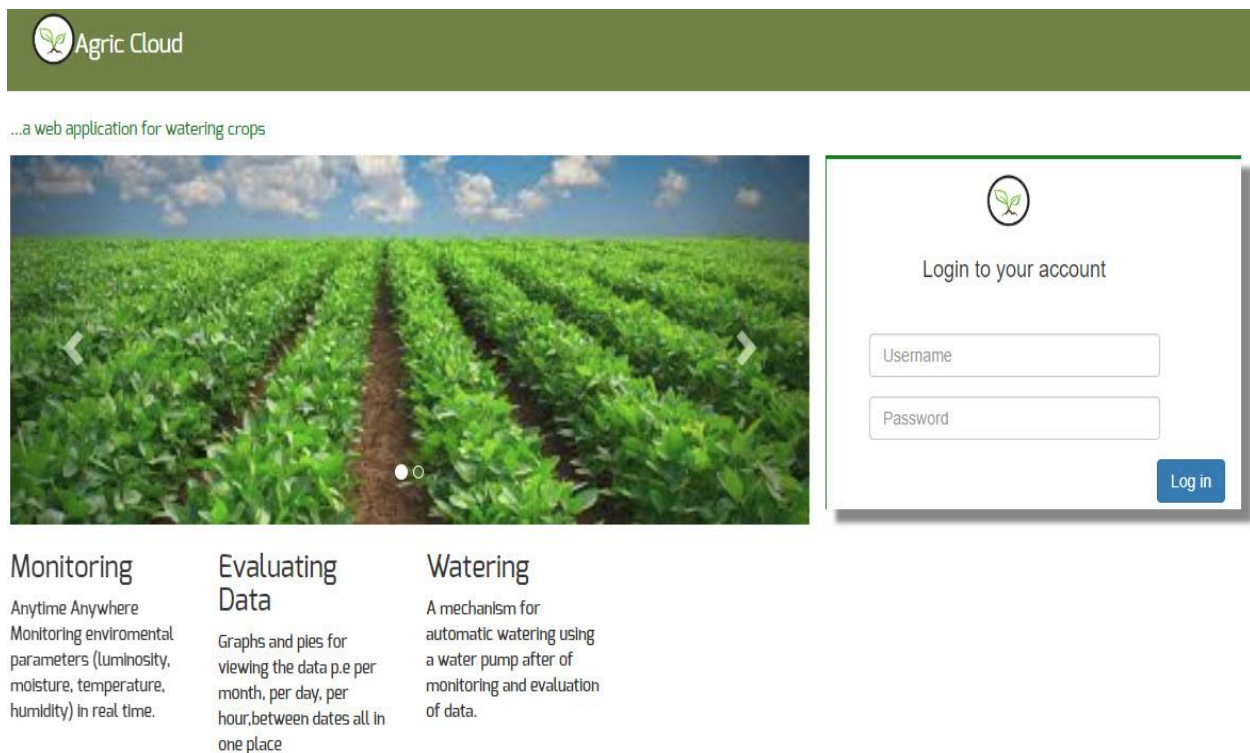
7.2 Διεπαφή χρήστη

Στο κομμάτι αυτό παρουσιάζεται η διεπαφή χρήστη για τις client εφαρμογές που έχουν αναπτυχθεί. Στο υποκεφάλαιο [7.2.1](#) παρουσιάζονται οι οθόνες της διεπαφής του χρήστη για τη διαδικτυακή εφαρμογή διαχείρισης και θέασης των δεδομένων. Ενώ στο υποκεφάλαιο [7.2.2](#) παρουσιάζονται οι αντίστοιχες οθόνες για τη διεπαφή χρήστη της mobile εφαρμογής.

7.2.1 Web εφαρμογή διαχείρισης

Οθόνη σύνδεσης (Login page):

Στην οθόνη εισαγωγής (σχήμα δίνεται η δυνατότητα στο χρήστη να εισάγει το username καθώς και τον κωδικό του για να πραγματοποιήσει την είσοδο του στην εφαρμογή.



Σχήμα 7.14: Web εφαρμογή - Οθόνη σύνδεσης

Αρχική οθόνη (Home page):

Αφού εισαχθούν σωστά το username και ο κωδικός, ο χρήστης οδηγείται στην αρχική οθόνη (Σχήμα 7.15), η οποία αποτελείται από τρία μέρη: την μπάρα που καλύπτει το επάνω μέρος της οθόνης (top bar), την μπάρα που καλύπτει το αριστερό μέρος της οθόνης (left bar) και την κύρια οθόνη. Οι δυο μπάρες που τοποθετήθηκαν επάνω και αριστερά χρησιμοποιούνται για την περιήγηση και την πραγματοποίηση ενεργειών του χρήστη. Ενώ το κύριο μέρος είναι αυτό που φιλοξενεί τις διαφορετικές σελίδες που φορτώνονται στην εφαρμογή.

Επάνω μπάρα (Top bar):

- *Home button:* Επιλογή περιήγησης στην αρχική σελίδα της εφαρμογής. Πατώντας την ένδειξη με τον τίτλο (Home) ο χρήστης επιστρέφει στην αρχική οθόνη.
- *Ένδειξη μηνυμάτων:* Πατώντας την ένδειξη με την καμπάνα ο χρήστης μπορεί να δει τις ειδοποιήσεις που δημιουργούνται αυτόματα από τις ενέργειες άρδευσης και νέων μετρήσεων. Στη συνέχεια ο χρήστης μπορεί να διαγράψει όλες ή κάποια συγκεκριμένη ειδοποίηση. Οι πληροφορίες που αναγράφονται στην οθόνη κάτω από την ειδοποίηση αφορούν την ακριβή ημερομηνία που πραγματοποιήθηκε το συμβάν, το είδος του συμβάντος (νέα μέτρηση, νέα άρδευση) καθώς και οι αναγνωριστικοί αριθμοί των συσκευών που πήραν μέρος στην κάθε ενέργεια.
- *Ένδειξη προσωπικών πληροφοριών χρήστη:* Πατώντας στην ένδειξη με το κωδικό όνομα του, ο χρήστης μπορεί να περιηγηθεί στην αντίστοιχη οθόνη και να επεξεργαστεί τις προσωπικές του πληροφορίες.
- *Ένδειξη αποσύνδεσης:* Ο χρήστης μπορεί να αποσυνδεθεί από την εφαρμογή πατώντας στην αντίστοιχη ένδειξη αποσύνδεσης (Logout). Θα εμφανιστεί ένα αναδυόμενο παράθυρο, όπου ο χρήστης θα μπορεί να οριστικοποιήσει ή να ακυρώσει την επιλογή του.

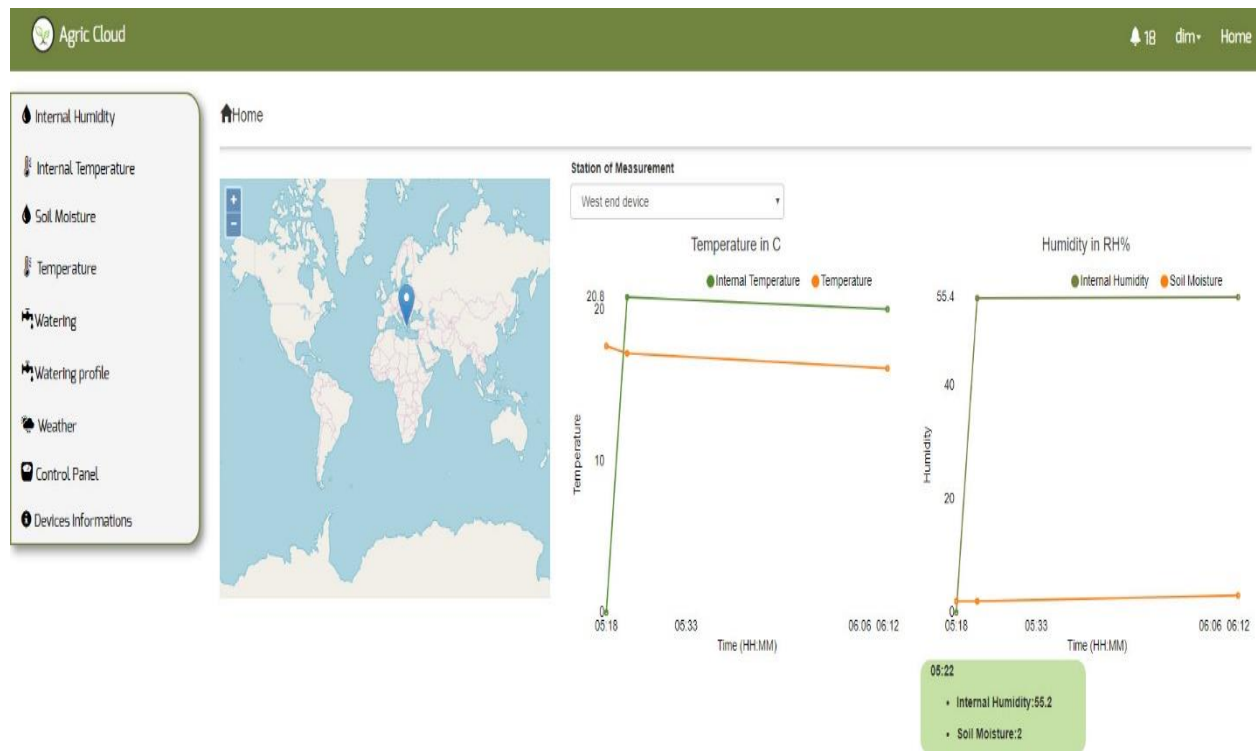
Αριστερή μπάρα (Left bar):

- *Επιλογή είδους μετρήσεων:* Ο χρήστης επιλέγει το είδος μετρήσεων για το οποίο θέλει να δει τις τιμές που έχουν συλλέξει οι αισθητήρες της εφαρμογής. Οι επιλογές που δίνονται στο χρήστη είναι οι εξής: Υγρασία συσκευής (Internal Humidity), Θερμοκρασία συσκευής (Internal Temperature), Υγρασία εδάφους (Soil Moisture), Θερμοκρασία ατμόσφαιρας (Temperature).
- *Επιλογή θέασης μετρήσεων άρδευσης:* Ο χρήστης επιλέγει τη θέαση των δεδομένων άρδευσης για κάθε συσκευή.
- *Επιλογή διαχείρισης ανωφλίων κατωφλίων:* Ο χρήστης έχει την επιλογή να επεξεργαστεί τις τιμές μεγίστων ελαχίστων για κάθε συσκευή έτσι ώστε να πραγματοποιηθεί η αυτόματη άρδευση.
- *Επιλογή πρόγνωσης καιρού:* Ο χρήστης έχει την επιλογή να δει με σχετική ακρίβεια την πρόγνωση του καιρού όσον αφορά την καλλιέργεια του για τις επόμενες 5 ημέρες.
- *Πίνακας ελέγχου (Control panel):* Επιλογή βασικού μενού για την πραγματοποίηση ενεργειών (νέα μέτρηση, νέα άρδευση) στις συσκευές της καλλιέργειας. Ο χρήστης επίσης μπορεί να διαχειριστεί τις ώρες που υλοποιείται ο αλγόριθμος αυτόματου ποτίσματος, καθώς επίσης και τη ροή του νερού από την ηλεκτροβάννα άρδευσης.
- *Πληροφορίες συσκευών (Devices' information):* Ο χρήστης μεταφέρεται στην οθόνη όπου μπορεί να επεξεργαστεί τις περιγραφές των συσκευών.

Κυρίως περιεχόμενο (Main content):

Αποτελεί το κομμάτι της οθόνης που φιλοξενεί τις διαφορετικές σελίδες, στις οποίες μπορεί να περιηγηθεί ο χρήστης μέσω των επιλογών που αναπτύχθηκαν παραπάνω. Στην αρχική οθόνη απεικονίζεται ο χάρτης με το σημείο στο οποίο βρίσκονται οι συντεταγμένες του σταθμού, καθώς και δυο γραφικές που απεικονίζουν τα ζεύγη των τελευταίων τιμών σε διάστημα μιας ημέρας (υγρασία εδάφους – υγρασία ατμόσφαιρας, θερμοκρασία ατμόσφαιρας – θερμοκρασία συσκευής). Δίνεται η δυνατότητα επιλογής της τελικής συσκευής, της ακριβούς ημερομηνίας και του ζεύγους τιμών μετρήσεων κάνοντας επιλογή στα κομμάτια των γραφικών

παραστάσεων. Τα αποτελέσματα φαίνονται σε αναδυόμενο παράθυρο ακριβώς κάτω από τις γραφικές παραστάσεις.



Σχήμα 7.15: Αρχική οθόνη

Οθόνη τιμών είδους μέτρησης:

Ο χρήστης μπορεί να περιηγηθεί μέσω του αριστερού μενού στις οθόνες που αντιστοιχούν στα είδη μετρήσεων των συσκευών. Στις εικόνες **Σχήμα 7.16**, **Σχήμα 7.17**, **Σχήμα 7.18**, **Σχήμα 7.19** φαίνονται οι τιμές των αισθητήρων στα είδη μετρήσεων (ατμοσφαιρική θερμοκρασία, θερμοκρασία συσκευής, υγρασία ατμόσφαιρας, υγρασία εδάφους). Επιλέγοντας τη συσκευή και το διάστημα των ημερομηνιών για τις οποίες θα πραγματοποιηθεί η θέαση των τιμών των μετρήσεων. Οι τιμές αυτές απεικονίζονται σε μορφή πίνακα καθώς και σε μορφή γραφικής παράστασης. Ο πίνακας υποστηρίζει σελιδοποίηση και διάταξη με βάση την τιμή μέτρησης και την ημερομηνία. Δίνεται τέλος η δυνατότητα μεταφόρτωσης των δεδομένων με τη μορφή αναφοράς σε τρεις διαφορετικούς τύπους: pdf, excel, csv (plain text). Η επιλογή του τύπου αρχείου μεταφόρτωσης πραγματοποιείται ανάλογα με τη χρήση του αντίστοιχου κουμπιού. Τα παραδείγματα των αναφορών σε μορφή pdf, xls φαίνονται στις εικόνες **Σχήμα 7.20**, **Σχήμα 7.21** αντίστοιχα. Η αναφορά σε μορφή pdf περιέχει το μοναδικό αναγνωριστικό συσκευής, το είδος της μέτρησης, το διάστημα ημερομηνιών αναζήτησης, το μέσο όρο των τιμών καθώς και τη μέγιστη και ελάχιστη τιμή των μετρήσεων. Οι τιμές σε μορφή απλού κειμένου φαίνονται παρακάτω:

Measure date & time;Measure value;Measure unit

05/05/2018 15:36:00;26.20;°C
 05/05/2018 15:40:00;26.00;°C
 05/05/2018 15:42:00;26.10;°C
 05/05/2018 16:08:00;25.90;°C
 27/10/2018 23:39:00;22.70;°C
 04/11/2018 13:10:00;22.20;°C
 04/11/2018 13:10:00;22.20;°C
 04/11/2018 13:37:00;22.30;°C
 04/11/2018 13:43:00;22.30;°C
 04/11/2018 13:49:00;22.30;°C
 18/11/2018 03:12:00;20.00;°C
 18/11/2018 03:18:00;20.10;°C
 18/11/2018 03:23:00;20.00;°C
 18/11/2018 04:12:00;19.80;°C

Temperature

End Device

West end device

Date from

Thu Mar 01 2018 00:00:00

Date To

2018-06-03 15:09:42

Units of measurement:°C

Date	Value
23/03/2018 13:27:00	21.97
29/03/2018 20:27:00	28
29/03/2018 21:27:00	30
01/04/2018 16:39:00	19.04
01/04/2018 16:55:00	19.04
01/04/2018 17:23:00	19.53
01/04/2018 17:31:00	19.53
05/05/2018 16:36:00	23.93
05/05/2018 16:40:00	24.41
05/05/2018 16:42:00	24.41

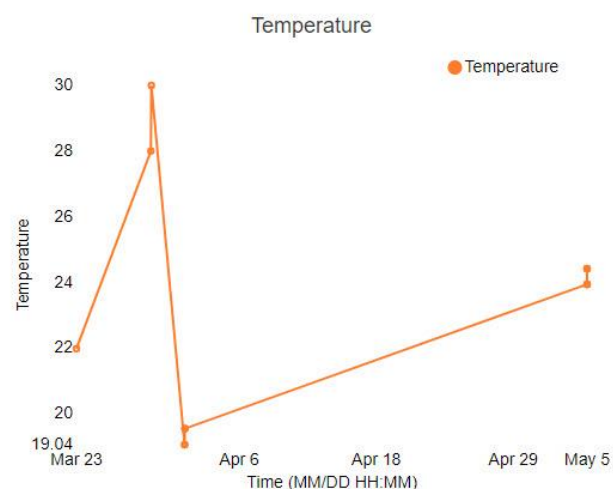
« 1 2 »

10 25 50 100

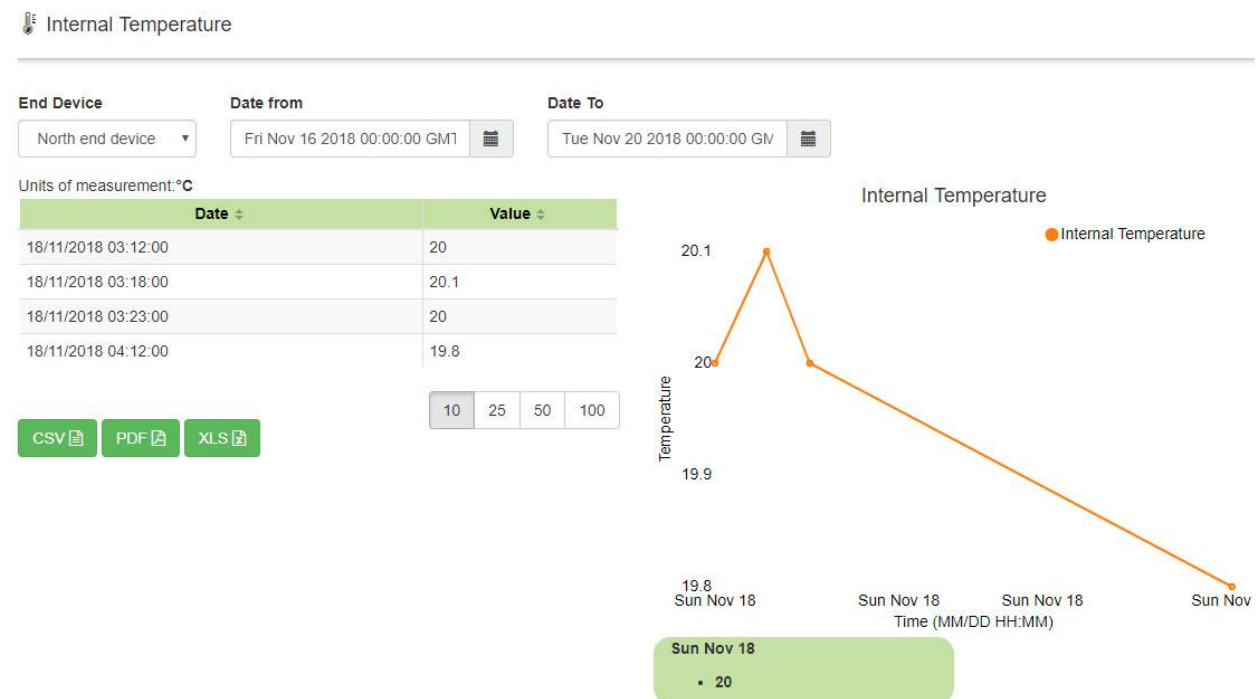
CSV

PDF

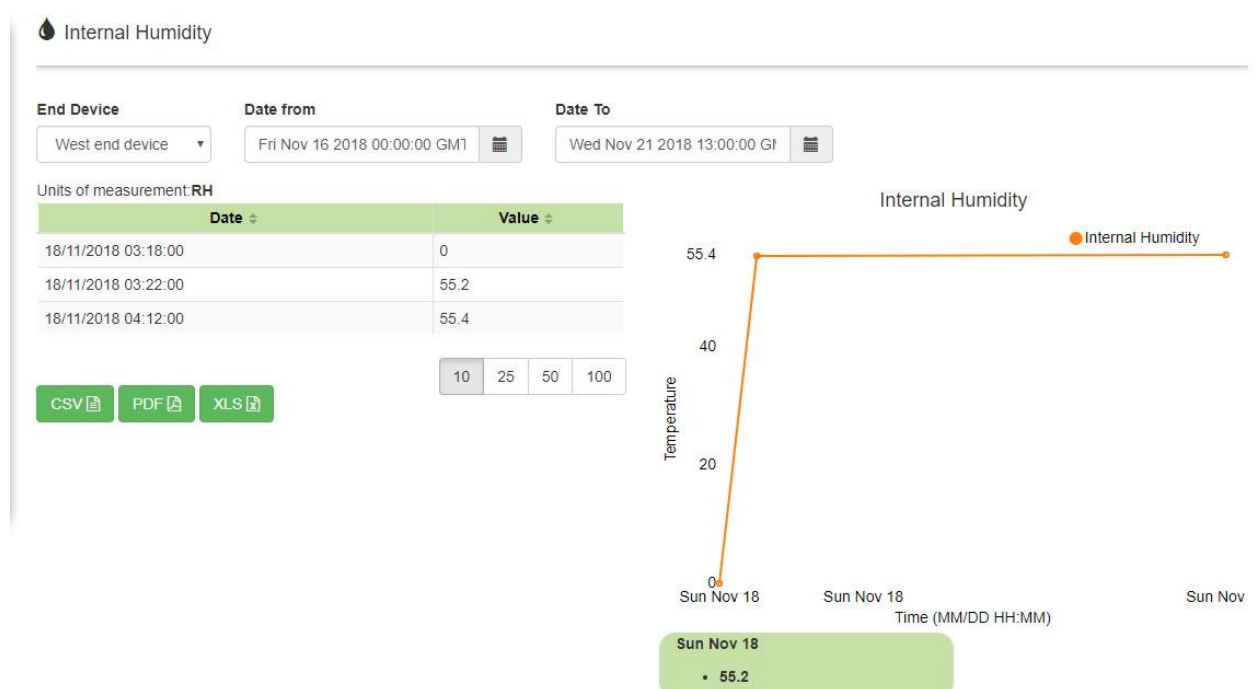
XLS



Σχήμα 7.16: Οθόνη θέασης και διαχείρισης τιμών είδους ατμοσφαιρικής θερμοκρασίας

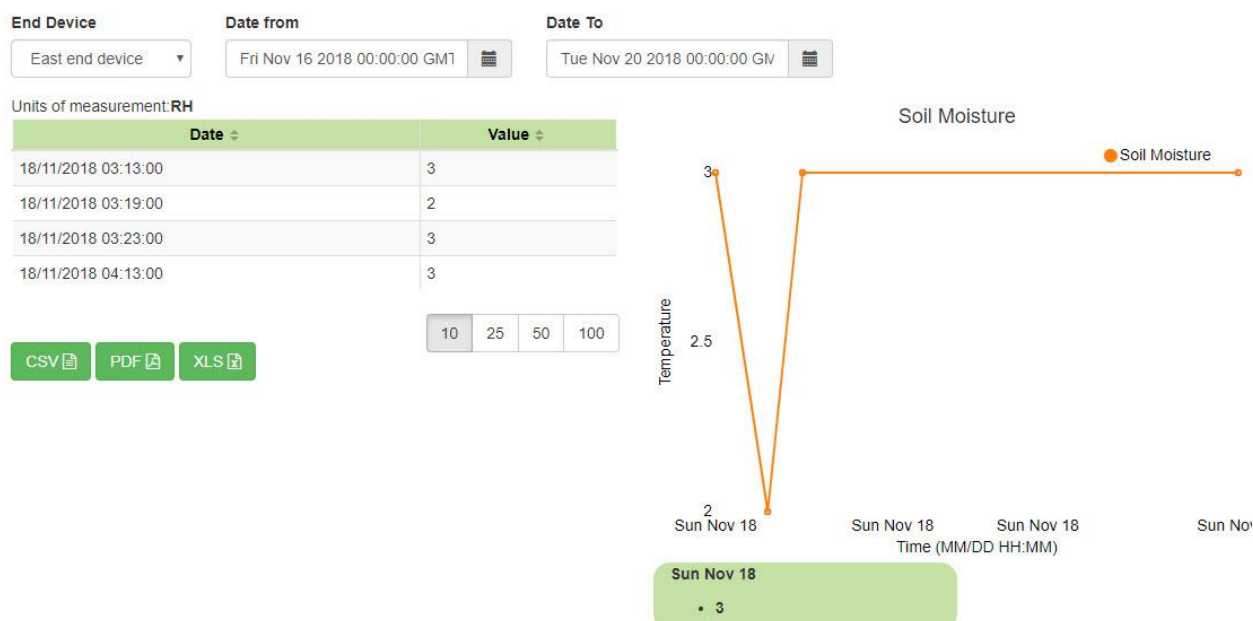


Σχήμα 7.17: Οθόνη θέασης και διαχείρισης τιμών Θερμοκρασίας συσκευής



Σχήμα 7.18: Οθόνη θέασης και διαχείρισης τιμών Υγρασίας ατμόσφαιρας

Soil Moisture



Σχήμα 7.19: Οθόνη θέασης και διαχείρισης τιμών υγρασίας εδάφους

Measures for Internal Temperature (°C) from end device with identifier 40D6A2CF
from 2018-05-01 00:00:00 until 2018-11-20 00:00:00

Measure date & time	Measure value
05/05/2018 15:36:00	26.20 °C
05/05/2018 15:40:00	26.00 °C
05/05/2018 15:42:00	26.10 °C
05/05/2018 16:08:00	25.90 °C
27/10/2018 23:39:00	22.70 °C
04/11/2018 13:10:00	22.20 °C
04/11/2018 13:10:00	22.20 °C
04/11/2018 13:37:00	22.30 °C
04/11/2018 13:43:00	22.30 °C
04/11/2018 13:49:00	22.30 °C
18/11/2018 03:12:00	20.00 °C
18/11/2018 03:18:00	20.10 °C
18/11/2018 03:23:00	20.00 °C
18/11/2018 04:12:00	19.80 °C

minimum Internal Temperature : 19.80 °C

maximum Internal Temperature : 26.20 °C

average Internal Temperature : 22.73 °C

Σχήμα 7.20: Αναφορά τιμών συσκευής σε μορφή pdf

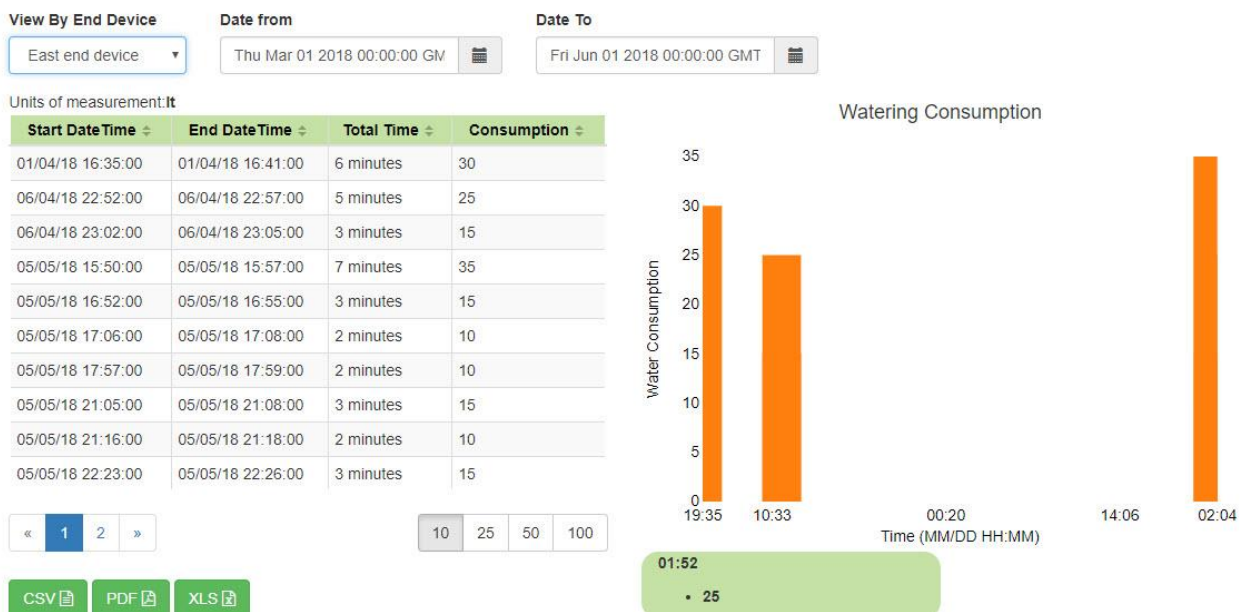
	A	B	C	D
1	Measure date & time	Measure value	Measure unit	
2	05/05/2018 15:36:00	26.20	°C	
3	05/05/2018 15:40:00	26.00	°C	
4	05/05/2018 15:42:00	26.10	°C	
5	05/05/2018 16:08:00	25.90	°C	
6	27/10/2018 23:39:00	22.70	°C	
7	04/11/2018 13:10:00	22.20	°C	
8	04/11/2018 13:10:00	22.20	°C	
9	04/11/2018 13:37:00	22.30	°C	
10	04/11/2018 13:43:00	22.30	°C	
11	04/11/2018 13:49:00	22.30	°C	
12	18/11/2018 03:12:00	20.00	°C	
13	18/11/2018 03:18:00	20.10	°C	
14	18/11/2018 03:23:00	20.00	°C	
15	18/11/2018 04:12:00	19.80	°C	
16				

Σχήμα 7.21: Αναφορά τιμών συσκευής σε μορφή xls

Οθόνη τιμών άρδευσης:

Ο χρήστης μπορεί να περιηγηθεί μέσω του αριστερού μενού στην οθόνη που αντιστοιχεί στα δεδομένα άρδευσης των συσκευών. Στην εικόνα **Σχήμα 7.22** αποτυπώνεται το χρονικό διάστημα (έναρξη - λήξη) του φαινομένου της άρδευσης, η χρονική διάρκεια και η κατανάλωση σε μορφή λίτρων (lt) της κάθε συσκευής άρδευσης. Επιλέγοντας τη συσκευή και το διάστημα των ημερομηνιών για τις οποίες θα πραγματοποιηθεί η θέαση των τιμών των μετρήσεων. Οι τιμές αυτές απεικονίζονται σε μορφή πίνακα καθώς και σε μορφή γραφικής παράστασης όπως και στις μετρήσεις των αισθητήρων. Το γράφημα είναι σε μορφή μπάρας για να απεικονίζονται τα χρονικά διαστήματα άρδευσης. Όπως και στις σελίδες θέασης και διαχείρισης τιμών μετρήσεων έτσι κι εδώ ο χρήστης έχει τη δυνατότητα να μεταφορτώσει αναφορές άρδευσης σε τρεις διαφορετικούς τύπους αρχείων pdf, excel, csv.

Watering Measure



Σχήμα 7.22: Οθόνη θέασης και διαχείρισης τιμών διαστημάτων άρδευσης

Οθόνη διαχείρισης ανωφλίων κατωφλίων αυτόματης άρδευσης:

Στη σελίδα αυτή ο χρήστης έχει τη δυνατότητα διαχείρισης μέγιστων και ελάχιστων τιμών για κάθε μια από τις συσκευές άρδευσης. Οι τιμές αυτές χρησιμοποιούνται από τον αλγόριθμο αυτόματου ποτίσματος της συσκευής για την έναρξη ή τη λήξη του φαινομένου της άρδευσης. Η έναρξη λαμβάνει χώρα όταν ικανοποιηθούν οι συνθήκες των ανωφλίων και κατωφλίων που έχουν οριστεί. Τα μετρήσιμα μεγέθη που ελέγχονται από τον αλγόριθμο της αυτόματης άρδευσης είναι η υγρασία εδάφους και η ατμοσφαιρική θερμοκρασία. Οι δύο αυτοί παράγοντες παίζουν μεγάλο ρόλο στην ανάπτυξη των φυτών. Με το διαχωρισμό των συσκευών δίνεται η επιλογή στο χρήστη του συστήματος να ρυθμίζει τις μέγιστες και ελάχιστες τιμές ανάλογα με το είδος της καλλιέργειας που μπορεί να υπάρχει σε ένα αγρόκτημα. Οι τιμές σώζονται στη βάση κάνοντας χρήση του κουμπιού με την ένδειξη «save» στο κάτω μέρος της οθόνης. Στο Σχήμα 7.23 φαίνεται η σελίδα για το παράδειγμα του αγροκτήματος που έχει δημιουργηθεί στη Β.Δ με σκοπό την πραγματοποίηση δοκιμών στην εφαρμογή. Παρατηρείται ότι υπάρχουν τιμές ανωφλίων κατωφλίων για κάθε μια από τις συσκευές με επικεφαλίδες το αναγνωριστικό της κάθε συσκευής, καθώς και την περιγραφή που έχει επιλέξει ο χρήστης.

Watering Profile

Watering Device: 40E7CC39 - West end device

Soil Moisture
Minimum value: 65 ✓ Maximum value: 80 ✓

Temperature
Minimum value: 0 ✓ Maximum value: 25 ✓

Watering Device: 40D6A2C9 - East end device

Soil Moisture
Minimum value: 70 ✓ Maximum value: 80 ✓

Temperature
Minimum value: 0 ✓ Maximum value: 30 ✓

Watering Device: 40D6A2CF - North end device

Soil Moisture
Minimum value: 70 ✓ Maximum value: 80 ✓

Temperature
Minimum value: 0 ✓ Maximum value: 30 ✓

Save Data

Information ⓘ You can set the minimum and maximum values of your irrigation system. The irrigation will start when the observable property is bigger than max value or smaller than less value.

Σχήμα 7.23: Οθόνη διαχείρισης ανωφλίων κατωφλίων αυτόματης άρδευσης

Οθόνη θέασης πρόγνωσης καιρού 5 ημερών:

Προς διευκόλυνση της απόφασης του χρήστη για την πραγματοποίηση άρδευσης υπάρχει η σελίδα πρόγνωσης καιρού για το διάστημα των 5 προσεχών ημερών. Η σελίδα όπως φαίνεται στο Σχήμα 7.24 χωρίζεται σε δύο μέρη.

Στο αριστερό μέρος της οθόνης υπάρχουν πληροφορίες σχετικά με τις συνθήκες που επικρατούν στο αγρόκτημα. Απεικονίζεται η θερμοκρασία, το είδος της ατμόσφαιρας (συννεφιασμένη, ηλιόλουστη..), η ταχύτητα και η διεύθυνση του ανέμου, το ποσοστό συννεφιάς, η ατμοσφαιρική πίεση σε «hPa», το ποσοστό υγρασίας που υπάρχει στην ατμόσφαιρα, καθώς και την ακριβή στιγμή ανατολής και δύσης του ήλιου.

Στο δεξί μέρος της οθόνης παρατηρείται το γράφημα της πρόγνωσης του καιρού σε σχέση με την ημερομηνία των επόμενων 5 ημερών σε διάστημα τριών ωρών μεταξύ των μετρήσεων, καθώς επίσης η θερμοκρασία, η υγρασία και το είδος των καιρικών συνθηκών ανά ημέρα. Όπως παρατηρείται στο Σχήμα 7.25 ο χρήστης μπορεί να επιλέξει συγκεκριμένη μέρα πρόγνωσης.

Weather

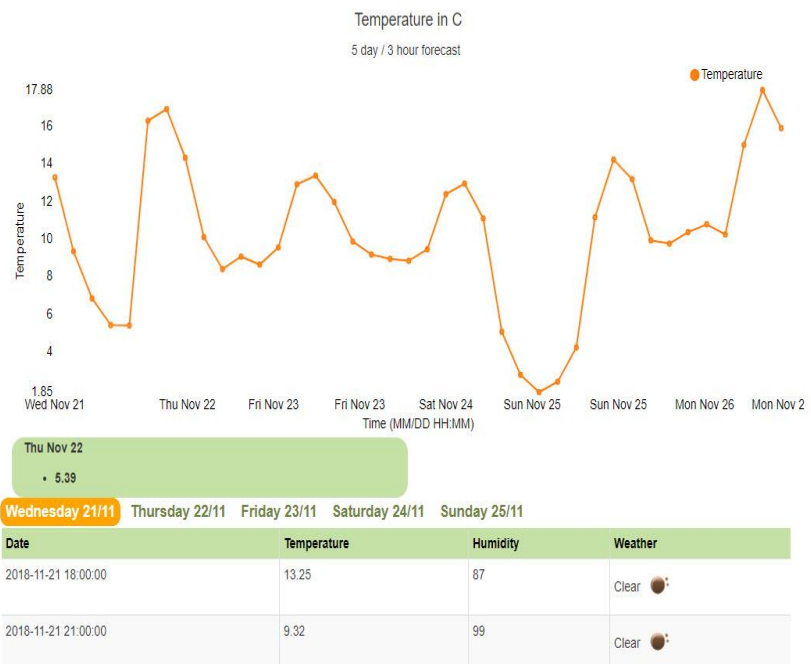
City of Loukisia



18.00 °C

scattered clouds

Wind	Moderate breeze 3.6 m/s Wind direction 300 °
Cloudiness	40
Pressure	1011 hpa
Humidity	77 %
Sunrise	Wednesday 21/11/2018 07:14:49
Sunset	Wednesday 21/11/2018 17:09:13



Σχήμα 7.24: Σελίδα πρόγνωσης καιρού για διάστημα 5 ημερών

Wednesday 21/11	Thursday 22/11	Friday 23/11	Saturday 24/11	Sunday 25/11
Date	Temperature	Humidity	Weather	
2018-11-23 00:00:00	9.04	100	Rain 🌧️	
2018-11-23 03:00:00	8.62	100	Rain 🌧️	
2018-11-23 06:00:00	9.53	100	Rain 🌧️	
2018-11-23 09:00:00	12.88	88	Rain 🌧️	
2018-11-23 12:00:00	13.34	81	Rain 🌧️	
2018-11-23 15:00:00	11.94	84	Rain 🌧️	
2018-11-23 18:00:00	9.84	96	Clouds ☁️	
2018-11-23 21:00:00	9.15	98	Rain 🌧️	

Σχήμα 7.25: Πίνακας πρόγνωσης καιρού συγκεκριμένης ημερομηνίας

Οθόνη πίνακα ελέγχου:

Η σελίδα αυτή (Σχήμα 7.26) αποτελεί τη ραχοκοκαλιά της εφαρμογής με την οποία ο χρήστης επικοινωνεί με τις συσκευές και το σταθμό στο αγρόκτημα. Αναγράφονται οι πληροφορίες για τις τελευταίες μετρήσεις ανά συσκευή και το τελευταίο συμβάν άρδευσης που συνέβη στο αγρόκτημα. Η σελίδα χωρίζεται σε τρία μέρη. Στο πρώτο μέρος αναγράφονται οι μετρήσεις σε μορφή πίνακα. Οι στήλες του πίνακα αποτελούνται από το είδος μέτρησης, την ημερομηνία που πραγματοποιήθηκε, την τιμή και τη μονάδα μέτρησης. Με την επιλογή του κουμπιού «Νέα μέτρηση (New measurement)», ανοίγει ένα αναδυόμενο παράθυρο (...) για να επιβεβαιώσει ο χρήστης την αποστολή νέας μέτρησης.

Στο δεύτερο μέρος αναγράφονται οι πληροφορίες για τα γεγονότα άρδευσης, οι χρονικές στιγμές έναρξης και λήξης του φαινομένου, καθώς και η κατανάλωση σε λίτρα. Επιλέγοντας το κουμπί με την ένδειξη «Έναρξη άρδευσης (New Irrigation)» εμφανίζεται το αναδυόμενο παράθυρο για την έναρξη του φαινομένου. Επιλέγοντας το χρονικό διάστημα έναρξης και λήξης του φαινομένου, υπολογίζεται αυτόματα η κατανάλωση. Στη συνέχεια ο χρήστης επιλέγοντας το κουμπί με την ένδειξη «Έναρξη (Start)» αποστέλλεται στο σταθμό του αγροκτήματος οι χρονικές στιγμές έναρξης και λήξης καθώς και η συσκευή υδροδότησης.

Στο τελευταίο κομμάτι της σελίδας δίνεται στο χρήστη η δυνατότητα επεξεργασίας του χρονικού διαστήματος έναρξης και λήξης λειτουργίας του αλγορίθμου αυτόματης υδροδότησης και ο ρυθμός ροής νερού για τις ηλεκτροβάνες του συστήματος. Με την επιλογή «Save Data» εμφανίζεται το αναδυόμενο παράθυρο, όπου ζητείται από το χρήστη η έγκριση της αλλαγής των δεδομένων. Κατά τις ώρες λειτουργίας του αλγορίθμου αυτόματης υδροδότησης ο χρήστης δεν μπορεί να πραγματοποιήσει νέα μέτρηση ή έναρξη νέου φαινομένου άρδευσης.

Control panel

Last measure for end device at Sunday, November 18th 4:12am from East end device

East end device

Observable Property	Datetime	Measure Value	Unit
Internal Humidity	Sunday, November 18th 4:13am	52.3	RH
Internal Temperature	Sunday, November 18th 4:13am	19.9	°C
Soil Moisture	Sunday, November 18th 4:13am	3	RH
Temperature	Sunday, November 18th 4:13am	19.53	°C

New Measurement

Last crop's Irrigation at

- Start Irrigation Time: Sunday, November 18th, 2018 6:38am
- End Irrigation Time: Sunday, November 18th, 2018 6:40am
- Water Consumption (lt): 10

Start Irrigation

Time From

06:00 PM

Time To

08:30 PM

Waterflow Rate (lt/h)

300

Save Data

Total time: 2 hours and 30 minutes

Σχήμα 7.26: Οθόνη «Πίνακας ελέγχου» διαχείρισης ενεργειών συσκευής αυτόματης άρδευσης

Take Measurements

This option will send a remote request to crop's devices to start taking measurements.

Cancel

Send

Σχήμα 7.27: Επιβεβαίωση νέας μέτρησης

Start irrigation for the: **East end device**

Date From

Wed Nov 21 2018 19:00:00 GMT+0200

Date To

Wed Nov 21 2018 20:20:00 GMT+0200

Total irrigation duration: **1 hours and 20 minutes**

Water consumption: **400**

Cancel

Start

Σχήμα 7.28: Έναρξη φαινομένου άρδευσης συγκεκριμένου χρονικού διαστήματος

Change Automatic Watering Time

This option will change the automatic irrigation watering time. And the water consumption in litres per hour (lt/h) too.

Cancel

Send

Σχήμα 7.29: Έγκριση αποθήκευσης δεδομένων

Για την αποφυγή λαθών πραγματοποιούνται έλεγχοι στα χρονικά διαστήματα σε όλα τα επίπεδα της αρχιτεκτονικής των εφαρμογών. Επίσης μετά την επιτυχή αποθήκευση των τιμών στο επίπεδο της εφαρμογής εμφανίζονται μηνύματα επιτυχίας ή αποτυχίας για να μπορεί ο χρήστης να κατανοήσει την εξέλιξη των διαφορών κλήσεων στο διακομιστή.

Οθόνη πληροφοριών συσκευών:

Ο χρήστης έχει τη δυνατότητα να επεξεργαστεί τις περιγραφές των συσκευών, να διαβάσει το είδος και τις ιδιότητες των αισθητήρων που απαρτίζουν τις συσκευές καθώς και να προχωρήσει σε καθορισμό των συντεταγμένων του σημείου όπου βρίσκεται ο σταθμός του αγροκτήματος. Τα είδη των συσκευών διαιρούνται ανάλογα με τον τύπο τους σε αγρόκτημα, σταθμό και τελικές συσκευές. Τα αναγνωριστικά των τελικών συσκευών καθώς και του σταθμού αποτελούνται από τις οκταδικές συμβολοσειρές των ZigBee που απαρτίζουν την κάθε συσκευή του αγροκτήματος (Σχήμα 7.30). Πατώντας πάνω στο κουμπί με την ένδειξη πινέζας ο χρήστης μπορεί να εισάγει το σημείο όπου βρίσκεται ο σταθμός εντός αγροκτήματος στο αναδυόμενο παράθυρο που εμφανίζεται αμέσως μετά (Σχήμα 7.31). Με την επιλογή του κουμπιού «Send» ο χρήστης προχωράει στην αποθήκευση των συντεταγμένων του σημείου.

⚙️ Settings

Crop ↕

Crop Name: 15c4d035-bdf5-449f-8005-336f103b6941

Description: Olives crop ✓

Station

Name: 40E7CC41

Description: Base station ✓

EndDevice 1 📶

Identifier: 40D6A2CF

Description: North end device ✓

Sensors

Name	Measurement Type
LM35	Analog Temperature
DHT22	Digital Measurement
DHT22	Digital Measurement
LM393	Soil Moisture
Hunter PGV 100 GB DC	DC Latching Solenoid

EndDevice 2 📶

Identifier: 40E7CC39

Description: West end device ✓

Sensors

Name	Measurement Type
LM35	Analog Temperature
DHT22	Digital Measurement
DHT22	Digital Measurement
LM393	Soil Moisture
Hunter PGV 100 GB DC	DC Latching Solenoid

EndDevice 3 📶

Identifier: 40D6A2C9

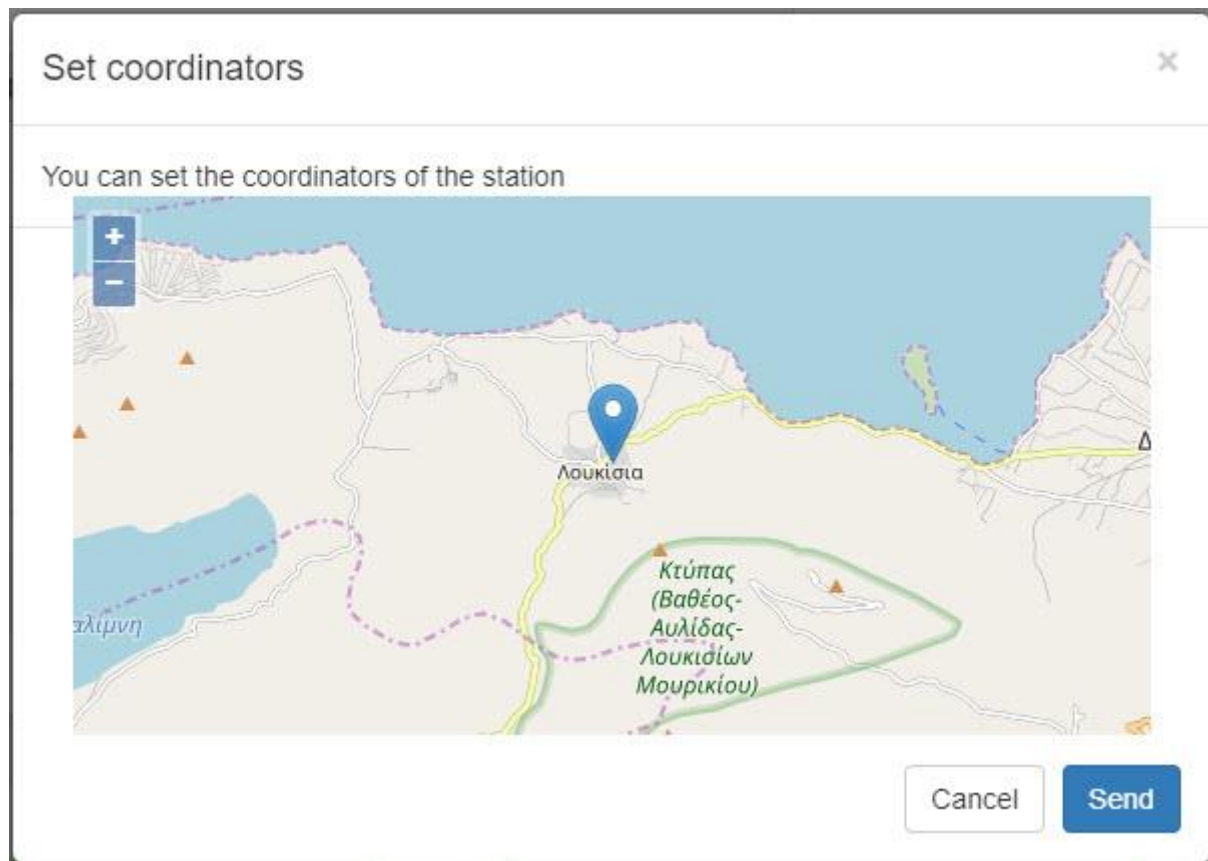
Description: East end device ✓

Sensors

Name	Measurement Type
LM35	Analog Temperature
DHT22	Digital Measurement
DHT22	Digital Measurement
LM393	Soil Moisture
Hunter PGV 100 GB DC	DC Latching Solenoid

Save Data 📥

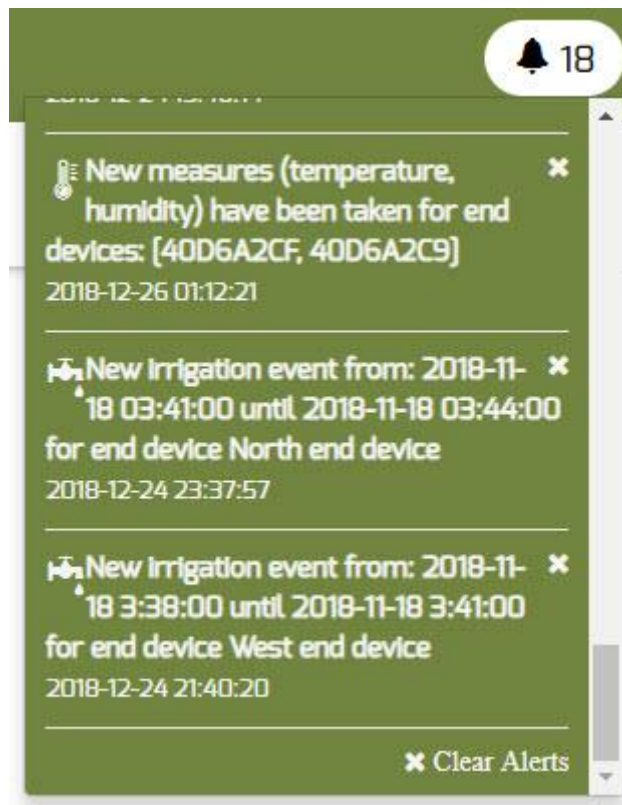
Σχήμα 7.30: Θέαση και διαχείριση πληροφοριών συσκευών



Σχήμα 7.31: Αναδυόμενο παράθυρο ορισμού συντεταγμένων σταθμού

Οθόνη προσωπικών στοιχείων και ειδοποιήσεων:

Ο χρήστης επιλέγοντας το κουμπί ειδοποιήσεων μπορεί να δει τις ειδοποιήσεις από τις ενέργειες που πραγματοποιούνται στο αγρόκτημα. Το είδος των ειδοποιήσεων διαχωρίζονται σε «Νέα μέτρηση» και «Νέα άρδευση». Αποτελείται από τις πληροφορίες των στιγμών έναρξης και λήξης των φαινομένων, καθώς και τις συσκευές του αγροκτήματος κατά τις οποίες έλαβαν μέρος (Σχήμα 7.32). Δίνεται επίσης η δυνατότητα διαγραφής ενός ή περισσότερων ειδοποιήσεων από το χρήστη. Η ένδειξη «My Profile» φορτώνει τη σελίδα των προσωπικών δεδομένων χρήστη, όπου εκεί ο χρήστης μπορεί να επεξεργαστεί τα προσωπικά και τα στοιχεία επικοινωνίας του. Στοιχεία όπως είναι ονοματεπώνυμο διεύθυνση και τρόποι επικοινωνίας (Σχήμα 7.33). Τέλος με την επιλογή «Logout (Αποσύνδεση)» και με την επιβεβαίωση στο αναδυόμενο παράθυρο () ο χρήστης αποσυνδέεται από την εφαρμογή και επιστρέφει στην οθόνη σύνδεσης (Σχήμα 7.14)

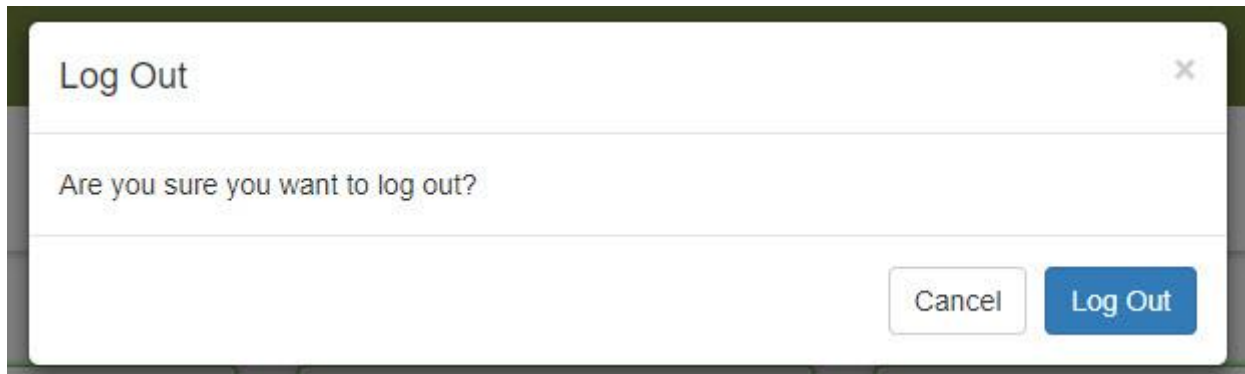


Σχήμα 7.32: Ειδοποιήσεις χρήστη

My Profile

firstname dim	lastname smirn	fatherstname alex
Street (optional) feidiou	Number (optional) 45	Zipcode (optional) 17642
Telephone (optional) 5678954	Mobile (optional) 456445	Date of birth 01/03/1989
		<input type="button" value="Clear"/> <input type="button" value="Save"/>

Σχήμα 7.33: Οθόνη επεξεργασίας προσωπικών δεδομένων χρήστη

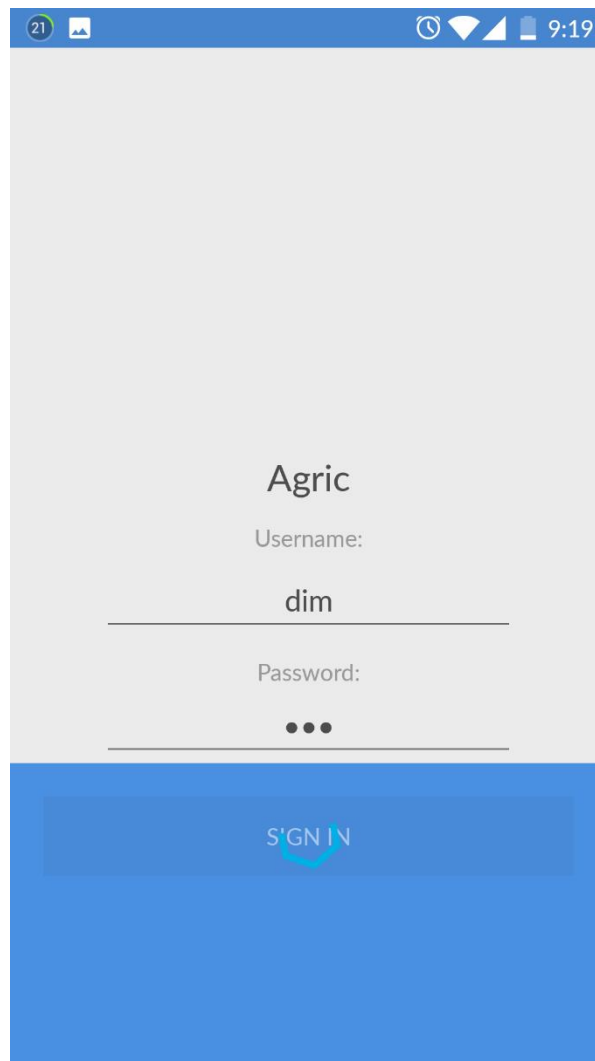


Σχήμα 7.34: Οθόνη αποσύνδεσης

7.2.2 Mobile εφαρμογή θέασης

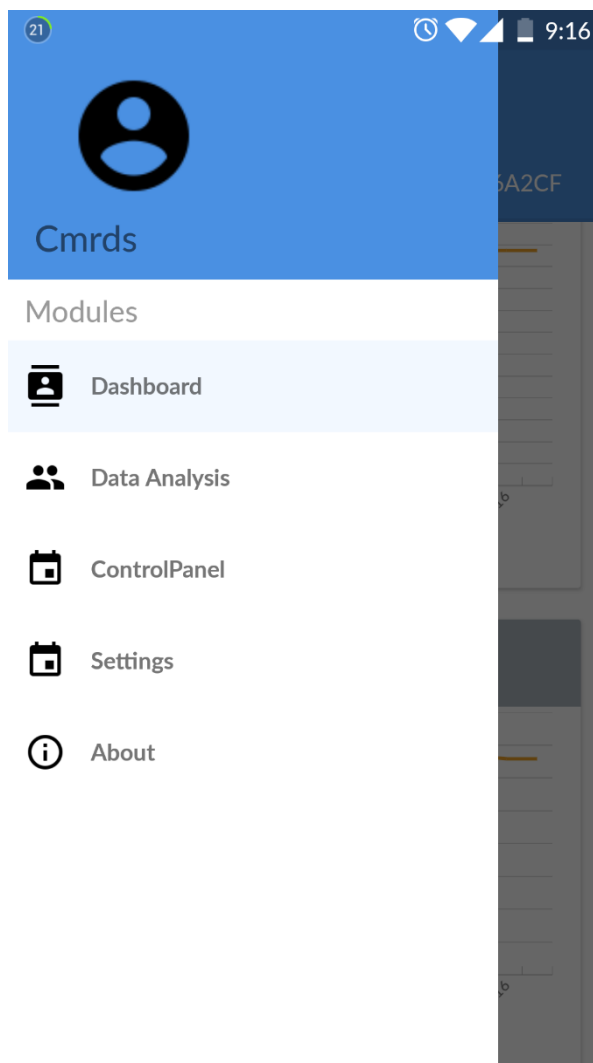
Οθόνη σύνδεσης (Login page):

Στην οθόνη εισαγωγής (σχήμα) δίνεται η δυνατότητα στο χρήστη να εισάγει το username καθώς και τον κωδικό του για να πραγματοποιήσει την είσοδο του στην εφαρμογή. Στη συνέχεια η εφαρμογή απομνημονεύει τα διαπιστευτήρια του χρήστη για τη μελλοντική αυτόματη είσοδο του σε αυτή.



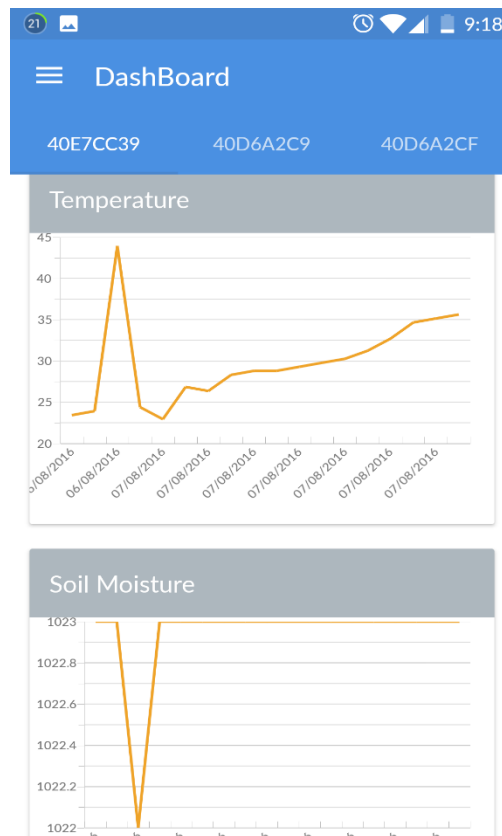
Σχήμα 7.35: Οθόνη σύνδεσης χρήστη (Mobile)

Στη συνέχεια μέσω του μενού ο χρήστης έχει τη δυνατότητα επιλογής της απαραίτητης καρτέλας. Οι διαθέσιμες καρτέλες είναι οι εξής: (α) Dashboard (β) Data analysis (γ) Control Panel (δ) Settings (ε) About.

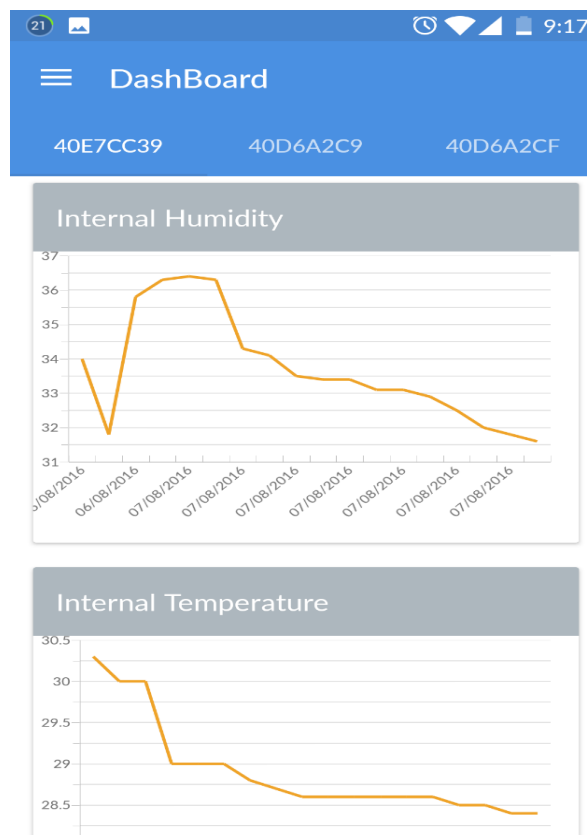


Σχήμα 7.36: Οθόνη μενού - επιλογής καρτέλας θέασης (Mobile)

Επιλέγοντας την καρτέλα dashboard ο χρήστης είναι σε θέση να δει τις τιμές των αισθητήρων ανά συσκευή στη μορφή γραφικής παράστασης όπως φαίνεται στο παρακάτω σχήμα.



Σχήμα 7.37: Θέαση δεδομένων θερμοκρασίας – υγρασίας εδάφους ανά συσκευή (Mobile)



Σχήμα 7.38: Θέαση δεδομένων θερμοκρασίας – υγρασίας ανά συσκευή (Mobile)

3.

ΣΥΜΠΕΡΑΣΜΑΤΑ

Η παρούσα διπλωματική περιγράφει τη σχεδίαση και την υλοποίηση εφαρμογών για τη διαχείριση αυτοματοποιημένων και μη εργασιών, καθώς και τη συλλογή δεδομένων στα αγροκτήματα. Η σχεδίαση των συστημάτων αυτών πραγματοποιήθηκε παράλληλα και σε συνάρτηση με την ανάπτυξη των κατανεμημένων συστημάτων που εγκαθίσταται στα αγροκτήματα (σταθμός και τελικές συσκευές). Η υλοποίηση διαχωρίζεται στα κομμάτια της σχεδίασης της Β.Δ, των διαδικτυακών υπηρεσιών (web services) για την αμφίδρομη επικοινωνία μεταξύ των εφαρμογών και των κατανεμημένων συστημάτων, τη web εφαρμογή διαχείρισης και την Android εφαρμογή θέασης και παρακολούθησης της καλλιέργειας.

Πιο συγκεκριμένα ένας χρήστης του συστήματος μπορεί να προχωρήσει στη θέαση των δεδομένων ανά διάστημα ημερομηνιών και ανά είδος μέτρησης που συλλέγονται από τους αισθητήρες των τελικών συσκευών, τη μεταφόρτωση των τιμών αυτών σε μορφή αναφορών, καθώς και την επιλογή συγκεκριμένης συσκευής αισθητήρων και χρονικού διαστήματος. Παρόμοιες πληροφορίες λαμβάνονται για την υδροδότηση της καλλιέργειας. Οι πληροφορίες αυτές μπορούν να γίνουν ορατές και από τις δύο εφαρμογές web & Android.

Η επικοινωνία του χρήστη με το ενσωματωμένο σύστημα γίνεται ασύγχρονα, καθώς όλες οι εντολές που αποστέλλονται από το χρήστη του συστήματος μέσω των εφαρμογών εκτελούνται με μια καθυστέρηση η οποία οφείλεται στο διάστημα που μένει ανενεργό το ενσωματωμένο σύστημα για λόγους εξοικονόμησης ενέργειας. Ο χρήστης έχει τη δυνατότητα να αλλάξει τους χρόνους αυτόματου ποτίσματος του ενσωματωμένου καθώς και να δημιουργήσει καινούριο συμβάν ποτίσματος μόνο για διαστήματα εκτός του αλγορίθμου αυτόματου ποτίσματος. Για κάθε νέο συμβάν στέλνεται ειδοποίηση στην εφαρμογή του χρήστη. Ακόμη δίνεται η δυνατότητα στο χρήστη για αλλαγή των ανώτερων και κατώτερων τιμών θερμοκρασίας και υγρασίας (ατμόσφαιρας και εδάφους) για τις οποίες ρυθμίζεται ο αλγόριθμος αυτόματης υδροδότησης του ενσωματωμένου. Οι στιγμές που περιγράφονται τα συμβάντα αποθηκεύονται στη Β.Δ σε ζώνη ώρας UTC ενώ στη συνέχεια μετατρέπονται στο ενδιάμεσο κομμάτι των υπηρεσιών (REST API) στη σωστή ημερομηνία και ώρα ανάλογα με τη γεωγραφική θέση στην οποία βρίσκεται τοποθετημένο το ενσωματωμένο. Ο χρήστης έχει τη δυνατότητα επίσης να διαβάσει τις προγνώσεις για τον καιρό μέσω της εφαρμογής, καθώς και να ζητήσει από το ενσωματωμένο νέες μετρήσεις θερμοκρασίας και υγρασίας, οπότε ανάλογα με τις καιρικές συνθήκες που επικρατούν στο αγρόκτημα τη συγκεκριμένη χρονική στιγμή αλλά και

τις ατμοσφαιρικές συνθήκες της περιοχής να προγραμματίσει είτε τον αλγόριθμο ποτίσματος για συγκεκριμένο διάστημα, είτε να ορίσει διάστημα για ύδρευση εκτός του αυτόματου αλγορίθμου.

Ο αλγόριθμος αυτόματης ύδρευσης καθώς και οι εφαρμογές που προσφέρουν τη δυνατότητα παραμετροποίησης των τιμών ανωφλίων και κατωφλίων προσφέρεται τόσο για απομακρυσμένη διαχείριση καλλιέργειών όσο και για την ασφαλή εξαγωγή συμπερασμάτων όσο αφορά τις καλλιέργειες. Έτσι κάποιος παραγωγός μπορεί να αποφασίσει ποια είδη είναι προς το συμφέρον του για παραγωγή. Η σύνδεση και ο διαμοιρασμός των δεδομένων αυτών σε διάφορους οργανισμούς (π.χ. οργανισμοί ύδρευσης) βοηθάει στην εξαγωγή ασφαλών συμπερασμάτων όσον αφορά τις εποχιακές καλλιέργειες και τα είδη που μπορούν να καλλιεργηθούν.

Οι συγκεκριμένες εφαρμογές μπορούν να επεκταθούν μελλοντικά από όλες τις σκοπιές, είτε τεχνικά, προσθέτοντας επιμέρους στρώματα (layers) στην υπάρχουσα αρχιτεκτονική συστήματος είτε σε επίπεδο διασύνδεσης και εμπλουτισμού των δεδομένων που παράγονται. Σε επίπεδο σχεδιασμού μπορεί να προστεθεί μηχανισμός που να υλοποιεί το πρωτόκολλο AMQP (Advanced Message Queuing Protocol) ανάμεσα στα web services και στις εφαρμογές. Έτσι θα μπορούν οι εντολές που δίνουν οι χρήστες να διαμοιράζονται σωστότερα προς τα αντίστοιχα ολοκληρωμένα συστήματα. Μπορεί επίσης να χρησιμοποιηθεί στην παραγωγή των αρχείων καταγραφής (logs) που παράγονται από τις εφαρμογές. Η μετατροπή των web services σε microservices για την πιο γρήγορη παραγωγή και διαμοιρασμό τους στις εφαρμογές. Ο εμπλουτισμός των δεδομένων εισόδου για τις καλλιέργειες με περισσότερες πληροφορίες όπως PH εδάφους, χημική σύσταση, μορφολογία κ.α. για την εξαγωγή ασφαλέστερων συμπερασμάτων για τα είδη καλλιέργειας. Προσθήκη αυτοματοποιημένων εργασιών στην πλευρά των εφαρμογών του χρήστη (π.χ. αυτόματη αλλαγή των τιμών του αλγορίθμου αυτόματης άρδευσης) σε σχέση με τα ιστορικά δεδομένα που έχουν συλλεχθεί. Αυτόματη αναβάθμιση λογισμικού για τα ενσωματωμένα συστήματα μέσω web services.

4.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] Enrique Playan, Luciano Mateos. "Modernization and optimization of irrigation systems to increase water productivity", 2005
- [2] Shri Prakash R. Kolhe, Shri Pradip P Kolhe, M.H Tharkar. "Role of information technology in modern irrigation sector", 2014
- [3] Rose Mary Seymour, Sheryl Wells. "Using Water Wisely with Automated Irrigation Systems", The University of Georgia 2012
- [4] <https://www.postgresql.org/docs/9.4/index.html>
- [5] <https://spring.io/projects/spring-boot>
- [6] <https://hibernate.org/orm/documentation/5.4/>
- [7] <https://docs.angularjs.org/guide>
- [8] Charles Petzold: "Creating Mobile Apps with Xamarin.Forms, Cross-platform C# programming for iOS, Android, and Windows"
- [9] <https://docs.docker.com/v17.12/>
- [10] Open Geospatial Consortium: "OGC® Sensor Observation Service Interface Standard" 2012
- [11] G. Alonso. Web Services: Concepts, Architectures and Applications. Springer, 2004.
- [12] C. Pautasso, O. Zimmermann, and F. Leymann. RESTful Web Services vs. "Big" Web Services: Making the Right Architectural Decision. In WWW '08: Proceeding of the 17th international conference on World Wide Web, Proceedings of the 17th international conference on World Wide Web, pages 805–814, New York, 2008. ACM.
- [13] R. T. Fielding and R. N. Taylor. Principled Design of the Modern Web Architecture. ACM Transactions on Internet Technology (TOIT), 2(2):115–150, 2002.
- [14] T. Reenskaug. Models - Views - Controllers. Technical report, Technical Note, Xerox Parc, 1979.
- [15] T. Reenskaug. The Model-View-Controller (MVC) Its Past and Present, 2003.
- [16] <https://postgis.net/stuff/postgis-2.4.pdf>
- [17] Spring RESTful Services. <http://spring.io/guides/gs/rest-service/>.
- [18] <https://swagger.io/>

- [19] A. L. Hors, P. L. Hégaret, L. Wood, G. Nicol, J. Robie, M. Champion, and S. Byrve. Document Object Object Model (DOM) Level 3 Core Specification. W3C Recommendation, April 2004.
- [20] The Model-View-ViewModel Pattern: <https://docs.microsoft.com/en-us/xamarin/xamarin-forms/enterprise-application-patterns/mvvm>
- [21] <https://d3js.org>
- [22] <https://github.com/esvit/ng-table>
- [23] <https://github.com/tombatossals/angular-OpenLayers-directive>
- [24] <https://OpenLayers.org/en/latest/APIdoc/>
- [25] D. Alur, D. Malks, and J. Crupi. Core J2EE Patterns: Best Practices and Design Strategies (2nd Edition). Prentice Hall, 2 edition, 2003.
- [26] Md.Khaled Hossain, S.M.Ariful Haque, Sumit Bhattacharyya: “GSM based Wireless Sensor Network to Measure Global Warming, Humidity and CO2”, August 2013
- [27] Syed Mubarak, Shoukhi Khan, Sahana.N, Megha B, S.Sujatha: “AUTOMATED IRRIGATION SYSTEM USING WIRELESS SENSOR NETWORKS AND GSM MODULE”, May 2015
- [28] Joaquín Gutiérrez, Juan Francisco Villa-Medina, Alejandra Nieto-Garibay, and Miguel Ángel Porta-Gándara: “Automated Irrigation System Using a Wireless Sensor Network and GPRS Module”
- [29] Tom Igoe: “Making Things Talk Second Edition” O’Reilly, 2011
- [30] A. Cockburn. Writing Effective Use Cases. Addison-Wesley Professional, 2001.
- [31] ECMA International. Standard ECMA-262 - ECMAScript Language Specification. 5.1 edition, June 2011.
- [32] R. T. Fielding. Architectural Styles and the Design of Network-based Software Architectures. Phd thesis, University of California, 2000.
- [33] R. T. Fielding and R. N. Taylor. Principled Design of the Modern Web Architecture. ACM Transactions on Internet Technology (TOIT), 2(2):115–150, 2002.
- [34] E. Gamma, R. Helm, R. Johnson, and J. Vlissides. Design Patterns, Elements of Reusable Object-Oriented Software. Addison-Wesley, 1995.