



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΩΝ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Τίτλος Πτυχιακής Εργασίας	Ανάπτυξη Πολυπρωτοκόλλου Συστήματος Διαχείρισης Αγροτικών IoT Συσκευών Development of a Multi-Protocol Management System for Agricultural IoT Devices
Όνοματεπώνυμο Φοιτητή	Βασιλακάκης Ελευθέριος
Πατρώνυμο	Μάρκος
Αριθμός Μητρώου	Π16010
Επιβλέπων	Χρήστος Δουληγέρης, Καθηγητής

Copyright ©

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν αποκλειστικά τον συγγραφέα και δεν αντιπροσωπεύουν τις επίσημες θέσεις του Πανεπιστημίου Πειραιώς.

Ως συγγραφέας της παρούσας εργασίας δηλώνω πως η παρούσα εργασία δεν αποτελεί προϊόν λογοκλοπής και δεν περιέχει υλικό από μη αναφερόμενες πηγές.

Ευχαριστίες

Είμαι βαθύτατα ευγνώμων στην οικογένειά μου για την αδιάκοπη υποστήριξη, την υπομονή και την κατανόηση κατά τη διάρκεια αυτής της απαιτητικής περιόδου. Η ενθάρρυνσή τους αποτέλεσε σταθερή πηγή δύναμης και κινήτρου.

Επίσης, θα ήθελα να ευχαριστήσω το ακαδημαϊκό προσωπικό για την πολύτιμη καθοδήγηση, τη διαθεσιμότητα και τη δημιουργία ενός περιβάλλοντος μάθησης που προώθησε τόσο την προσωπική όσο και την επαγγελματική ανάπτυξη.

Ένα ιδιαίτερο ευχαριστώ απευθύνεται στον επιβλέποντα καθηγητή μου, Χρήστο Δουληγέρη, για τις παρατηρήσεις του, την εποικοδομητική κριτική και τη συνεχή υποστήριξη καθ' όλη τη διάρκεια της εκπόνησης αυτής της εργασίας. Η εμπειρία, η τεχνογνωσία και η ενθάρρυνσή του υπήρξαν καθοριστικές για τη διαμόρφωση του αποτελέσματος αυτής της διπλωματικής.

Θα ήθελα επίσης να εκφράσω τις ειλικρινείς μου ευχαριστίες στον Παναγιώτη Γκοτσιόπουλο για τη σημαντική βοήθεια και υποστήριξη που μου παρείχε κατά την εκπόνηση αυτής της πτυχιακής εργασίας. Η συμβολή του και οι πολύτιμες συμβουλές του αποτέλεσαν καθοριστικό παράγοντα για την ολοκλήρωση αυτού του εγχειρήματος.

Σε αυτούς που στάθηκαν στο πλευρό μου σε αυτό το ταξίδι, ευχαριστώ!

ΠΕΡΙΛΗΨΗ

Η παρούσα πτυχιακή εργασία επικεντρώνεται στην ανάπτυξη ενός ολοκληρωμένου συστήματος διαχείρισης αγροτικών συσκευών IOT(Διαδίκτυο των Πραγμάτων), το οποίο υποστηρίζει λήψεις JSON (μορφή δεδομένων) δεδομένων μέσω του πρωτοκόλλου MQTT από δημόσιους και ιδιωτικούς μεσολαβητές όπως και μέσω αποστολής σε webhooks του συστήματος για

επικοινωνία με την πλατφόρμα υποστήριξης LoRaWAN (πρωτόκολλο) δικτύων TTS. Μέσω μιας φιλικής web πλατφόρμας, οι χρήστες μπορούν να καταχωρούν χωράφια με γεωγραφική τοποθεσία, να τοποθετούν και να διαχειρίζονται διάφορες συσκευές αισθητήρων και παρακολούθησης, καθώς και να λαμβάνουν πραγματικού χρόνου δεδομένα και ειδοποιήσεις. Το σύστημα ενσωματώνει τεχνολογίες όπως PHP Laravel, Arduino και LoRaWAN (πρωτόκολλο) Devices, επιτρέποντας την εύκολη διασύνδεση ετερογενών συσκευών και τη βέλτιστη αξιοποίηση των δυνατοτήτων τους. Παράλληλα, προσφέρει εργαλεία οπτικοποίησης, αυτοματοποιημένες ειδοποιήσεις και δυνατότητες geofencing, συμβάλλοντας στη βελτιστοποίηση της αγροτικής παραγωγής και στη βιώσιμη διαχείριση των πόρων.

ABSTRACT

This bachelor's thesis presents the development of an integrated IoT device management system for agriculture. The system can receive JSON data using the MQTT protocol from both public and private brokers, and it can send data through webhooks to communicate with the LoRaWAN (πρωτόκολλο) network platform TTS. Users can manage fields with geographical locations, install and control various sensor devices, and get *πραγματικού χρόνου* data and alerts through a user-friendly *διαδικτυακή* πλατφόρμα. The project uses technologies like PHP Laravel, Arduino, and LoRaWAN (πρωτόκολλο) to connect different devices and get the best performance from them. It also offers tools for visualization, automated notifications,

and geofencing, helping improve agricultural production and resource management efficiently

ΠΕΡΙΕΧΟΜΕΝΑ

Εξώφυλλο	i
Copyright	ii
Ευχαριστίες	iii
Περίληψη	iv
Abstract	v
Πίνακας Περιεχομένων	vi
1. ΕΙΣΑΓΩΓΗ	9
2. ΒΙΒΛΙΟΓΡΑΦΙΚΗ ΑΝΑΣΚΟΠΗΣΗ	12
3. ΑΝΑΛΥΣΗ ΑΠΑΙΤΗΣΕΩΝ	15
• 3.1 Τεχνολογία Backend	16
• 3.2 Τεχνολογία Frontend	17
• 3.3 Ολοκληρωμένη Προσέγγιση	17
• 3.4 Δυνατά Σημεία	18
• 3.5 Πιθανές Αδυναμίες και Περιορισμοί	19
4. ΑΝΑΛΥΣΗ ΚΑΙ ΥΛΟΠΟΙΗΣΗ	20
• 4.1 BackEnd Υλοποίηση	21
○ 4.1.1 Ιεραρχική Δομή Models και Data Architecture	21
○ 4.1.2 Multi-Protocol Communication Layer	22
○ 4.1.3 Background Processing Architecture	22
○ 4.1.4 API Architecture και Webhook Integration	23
○ 4.1.5 Eloquent Models και Relationships	23
• 4.2 Services Layer - Επιχειρηματική Λογική	24
○ 4.2.1 GeofencingService - Γεωχωρική Παρακολούθηση	24

○ 4.2.2 Multi-Protocol Communication Services	25
○ 4.2.3 Background Processing Integration	25
○ 4.2.4 Device Management και Auto-Discovery	25
● 4.3 Frontend Υλοποίηση με Limitless Theme	26
● 4.4 Dashboard - IoT Monitoring Overview	34
○ 4.4.1 Κεντρικές Μετρικές	34
○ 4.4.2 Στατιστικά Συσκευών	34
○ 4.4.3 Οπτική Αναπαράσταση Δεδομένων	34
○ 4.4.4 Σύστημα Ειδοποιήσεων	35
○ 4.4.5 Κατάσταση Συστήματος	35
● 4.5 ESP32 Firmware - Hardware Integration	37
5. ΑΠΟΤΕΛΕΣΜΑΤΑ	40
● 5.1 Τεχνολογική Επιτυχία	43
● 5.2 Background Processing Architecture	43
● 5.3 Geofencing και Real-time Monitoring	43
● 5.4 Multi-Protocol Communication Architecture	44
● 5.5 Hierarchical Data Architecture Success	44
6. ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ	44
ΒΙΒΛΙΟΓΡΑΦΙΑ	46
ΠΑΡΑΡΤΗΜΑ	47

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

Εικόνα 1. Σελίδα Σύνδεσης (Login Page)	29
Εικόνα 2. Σελίδα Εγγραφής (Register Page)	30
Εικόνα 3. Σελίδα Τεκμηρίωσης και Υποστήριξης (How To Page)	31
Εικόνα 4. Σελίδα Καταλόγου Εκτάσεων (Lands Index Page)	32
Εικόνα 5. Σελίδα Δημιουργίας Έκτασης με Leaflet.js	33
Εικόνα 6. Σελίδα Προβολής Έκτασης με Real-time Monitoring	34
Εικόνα 7. Χάρτης Έκτασης με Geofencing Visualization	34
Εικόνα 8. Σελίδα Καταλόγου Συσκευών (Devices Index Page)	35
Εικόνα 9. Σελίδα Δημιουργίας Συσκευής IoT	36
Εικόνα 10. Σελίδα Προβολής Συσκευής IoT με Auto-generated Sensors	37
Εικόνα 11. Σελίδα Καταλόγου Αισθητήρων (Sensors Index Page)	37
Εικόνα 12. Σελίδα Επεξεργασίας Αισθητήρα με Threshold Settings	38
Εικόνα 13. Κεντρικό Dashboard με Real-time Monitoring	41
Εικόνα 14. Εικονικό Arduino Interface με Geofence Mode	42
Εικόνα 15. Εικονικό Arduino Interface με LCD Display	43
Εικόνα 16. Live Data Streaming από ESP32 Device	43

1. ΕΙΣΑΓΩΓΗ

Η σύγχρονη γεωργία διανύει μια περίοδο ραγδαίων μετασχηματισμών, καθώς η ενσωμάτωση των τεχνολογιών Internet of Things (IoT) αλλάζει ριζικά τον τρόπο με τον οποίο οι αγροτικές εκμεταλλεύσεις παρακολουθούν, διαχειρίζονται και βελτιστοποιούν τις λειτουργίες τους. Η μετάβαση αυτή προς την ψηφιακή εποχή δεν αποτελεί απλώς μια τεχνολογική εξέλιξη, αλλά μια ουσιαστική μεταμόρφωση του αγροτικού τοπίου, όπου η συλλογή και ανάλυση δεδομένων σε πραγματικό χρόνο καθίστανται θεμέλιοι λίθοι για την αύξηση της παραγωγικότητας, τη μείωση του κόστους και την επίτευξη βιώσιμης ανάπτυξης. Στο πλαίσιο αυτό, η παρούσα εργασία εστιάζει στην ανάπτυξη ενός καινοτόμου, ολοκληρωμένου συστήματος διαχείρισης αγροτικών συσκευών IoT, το οποίο υποστηρίζει πολλαπλά πρωτόκολλα επικοινωνίας και παρέχει στους χρήστες τη δυνατότητα ολιστικής παρακολούθησης και ελέγχου των αγροτικών τους εκμεταλλεύσεων μέσω μιας ενιαίας, φιλικής προς τον χρήστη πλατφόρμας.

Στο σημερινό αγροτικό περιβάλλον, η ανάγκη για ακριβή παρακολούθηση των περιβαλλοντικών συνθηκών, αποτελεσματική διαχείριση των πόρων και έγκαιρη λήψη αποφάσεων έχει καταστεί πιο επιτακτική από ποτέ. Οι παραδοσιακές μέθοδοι αγροτικής παραγωγής δεν επαρκούν πλέον για να ανταποκριθούν στις σύγχρονες προκλήσεις, όπως η κλιματική αλλαγή, η αυξανόμενη ζήτηση για τρόφιμα και η ανάγκη για βιώσιμες πρακτικές. Η υιοθέτηση των τεχνολογιών IoT στον αγροτικό τομέα προσφέρει πρωτοφανείς δυνατότητες για τη βελτιστοποίηση της παραγωγικότητας, τη μείωση του κόστους λειτουργίας και την προστασία του περιβάλλοντος. Μέσω της αξιοποίησης αισθητήρων, συσκευών και προηγμένων πλατφορμών διαχείρισης, οι αγρότες αποκτούν τη δυνατότητα να λαμβάνουν αποφάσεις βασισμένες σε πραγματικά δεδομένα, να αυτοματοποιούν διαδικασίες και να προλαμβάνουν προβλήματα που σχετίζονται με τις καλλιέργειές τους.

Το προτεινόμενο σύστημα διαχείρισης αγροτικών συσκευών IoT αποτελεί μια ολοκληρωμένη λύση που συνδυάζει τις δυνατότητες των σύγχρονων τεχνολογιών με την ευελιξία υποστήριξης πολλαπλών πρωτοκόλλων επικοινωνίας. Η καινοτομία του έγκειται στην ενοποίηση διαφορετικών τεχνολογιών επικοινωνίας, όπως MQTT brokers (EMQX, Mosquitto) και LoRaWAN (πρωτόκολλο) δίκτυα (The Things Stack), σε μια ενιαία, φιλική προς τον χρήστη διεπαφή. Αυτή η προσέγγιση απλοποιεί

τη διαχείριση πολύπλοκων IoT οικοσυστημάτων και προσφέρει στους χρήστες τη δυνατότητα να διαχειρίζονται το σύνολο των συσκευών τους, ανεξαρτήτως τεχνολογίας επικοινωνίας, από μια ενιαία πλατφόρμα. Επιπλέον, το σύστημα ενσωματώνει προηγμένες λειτουργίες γεωχωρικής διαχείρισης και geofencing, επιτρέποντας στους χρήστες να ορίζουν γεωγραφικά όρια για τις εκμεταλλεύσεις τους και να παρακολουθούν τη θέση των συσκευών τους σε πραγματικό χρόνο.

Ο βασικός σκοπός της εργασίας είναι η σχεδίαση, ανάπτυξη και υλοποίηση ενός ολοκληρωμένου web-based συστήματος διαχείρισης αγροτικών συσκευών IoT, το οποίο υποστηρίζει πολλαπλά πρωτόκολλα επικοινωνίας και παρέχει στους χρήστες τη δυνατότητα αποτελεσματικής παρακολούθησης και ελέγχου των αγροτικών τους εκμεταλλεύσεων μέσω μιας ενιαίας, φιλικής προς τον χρήστη διεπαφής. Το σύστημα στοχεύει να καλύψει το κενό που υπάρχει στην αγορά για ενοποιημένες λύσεις που μπορούν να διαχειριστούν ταυτόχρονα διαφορετικά πρωτόκολλα επικοινωνίας IoT. Από τεχνολογικής άποψης, η εργασία επικεντρώνεται στην ανάπτυξη ενός πολυπρωτοκόλλου συστήματος επικοινωνίας, το οποίο θα υποστηρίζει MQTT brokers, LoRaWAN (πρωτόκολλο) δίκτυα και θα ενσωματώνει εξελιγμένους αλγορίθμους αφαίρεσης πρωτοκόλλων και αυτόματους μηχανισμούς ανίχνευσης και σύνδεσης συσκευών. Παράλληλα, το σύστημα θα προσφέρει προηγμένες λειτουργίες γεωχωρικής διαχείρισης και geofencing, επιτρέποντας στους χρήστες να ορίζουν γεωγραφικά όρια και να παρακολουθούν τη θέση των συσκευών τους σε πραγματικό χρόνο.

Οι λειτουργικοί στόχοι της εργασίας περιλαμβάνουν την ανάπτυξη ενός ολοκληρωμένου συστήματος διαχείρισης συσκευών και αισθητήρων που θα παρέχει πραγματικού χρόνου παρακολούθηση δεδομένων, αυτοματοποιημένες ειδοποιήσεις και συναγερμούς, καθώς και εργαλεία οπτικοποίησης δεδομένων μέσω προσαρμόσιμων dashboards. Το σύστημα θα περιλαμβάνει επίσης εξελιγμένες λειτουργίες διαχείρισης δικτύων και brokers, με εργαλεία παρακολούθησης της δικτυακής κατάστασης, συστήματα load balancing για πολλαπλούς brokers.

Ιδιαίτερη έμφαση δίνεται στη χρηστική εμπειρία, με στόχο τη δημιουργία μιας διαισθητικής web διεπαφής με responsive design για mobile συσκευές και

customizable dashboards. Για τη διευκόλυνση των χρηστών θα αναπτυχθεί εκπαιδευτικό υλικό, online help system και tutorials.

Η κύρια καινοτομία του προτεινόμενου συστήματος έγκειται στη δυνατότητα ενοποίησης διαφορετικών τρόπων επικοινωνίας με IoT συστήματα σε μια ενιαία πλατφόρμα διαχείρισης. Ενώ τα υπάρχοντα συστήματα στην αγορά συνήθως υποστηρίζουν ένα συγκεκριμένο πρωτόκολλο επικοινωνίας, το προτεινόμενο σύστημα προσφέρει δυνατότητα χρήσης και διασύνδεσης IoT συσκευών που χρησιμοποιούν MQTT brokers και το LoRaWAN (πρωτόκολλο) TTS τις οποίες μπορούμε να προσθέσουμε και να παραμετροποιήσουμε κατά βούληση στην πλατφόρμα, επιτρέποντας στους χρήστες να αξιοποιήσουν τα πλεονεκτήματα κάθε τεχνολογίας ανάλογα με τις ιδιαίτερες ανάγκες τους.

Σε επιχειρηματικό επίπεδο, το σύστημα προσφέρει σημαντική μείωση του κόστους υλοποίησης IoT λύσεων στον αγροτικό τομέα. Η ενοποίηση πολλαπλών πρωτοκόλλων σε μια πλατφόρμα εξαλείφει την ανάγκη για επένδυση σε πολλαπλά συστήματα διαχείρισης, μειώνοντας τόσο το αρχικό κόστος όσο και τα έξοδα συντήρησης. Παράλληλα, η αυξημένη ευελιξία και επεκτασιμότητα του συστήματος επιτρέπει στους αγρότες να ξεκινήσουν με απλές και οικονομικές MQTT συσκευές και σταδιακά να επεκτείνουν το σύστημά τους με LoRaWAN (πρωτόκολλο) συσκευές για μεγαλύτερες αποστάσεις ή ειδικές εφαρμογές, χωρίς να χρειάζεται αλλαγή πλατφόρμας ή επανεκπαίδευση του προσωπικού.

Η προστιθέμενη αξία του συστήματος επεκτείνεται και στον τομέα της γεωχωρικής διαχείρισης, όπου η ενσωμάτωση προηγμένων λειτουργιών geofencing και GPS tracking προσφέρει νέες δυνατότητες παρακολούθησης και ελέγχου. Οι αγρότες μπορούν να ορίσουν ακριβή γεωγραφικά όρια για τις εκμεταλλεύσεις τους, να παρακολουθούν τη θέση των συσκευών τους σε πραγματικό χρόνο και να λαμβάνουν αυτόματες ειδοποιήσεις σε περίπτωση παραβίασης των ορίων. Αυτές οι λειτουργίες είναι ιδιαίτερα χρήσιμες για την προστασία από κλοπές, την παρακολούθηση κινητών συσκευών και τη βελτιστοποίηση της τοποθέτησης αισθητήρων καθώς και επιτρέπουν την παρακολούθηση των συσκευών όταν αυτές δε βρίσκονται σε σταθερή θέση.

Συνολικά, η ολοκληρωμένη αυτή προσέγγιση τοποθετεί το προτεινόμενο σύστημα στην αιχμή της τεχνολογικής καινοτομίας για τη γεωργία, προσφέροντας στους αγρότες τα εργαλεία που χρειάζονται για να ανταποκριθούν στις προκλήσεις του μέλλοντος με αποτελεσματικότητα, ευελιξία και ασφάλεια.

2. ΒΙΒΛΙΟΓΡΑΦΙΚΗ ΑΝΑΣΚΟΠΗΣΗ

Η σύγχρονη βιβλιογραφία επιβεβαιώνει ότι η ενσωμάτωση των τεχνολογιών Internet of Things (IoT) στη γεωργία αποτελεί έναν από τους ταχύτερα αναπτυσσόμενους τομείς της αγροτικής τεχνολογίας. Οι Farooq et al. (2020) καταγράφουν τη ραγδαία αύξηση της χρήσης τεχνικών όπως το big data, το machine learning, το computer vision και το blockchain στη σύγχρονη γεωργία, υπογραμμίζοντας ότι η ανάλυση και αξιοποίηση δεδομένων σε πραγματικό χρόνο οδηγεί σε σημαντική βελτίωση της παραγωγικότητας και της βιωσιμότητας. Η ανάγκη για πολυπρωτόκολλα συστήματα επικοινωνίας αναδεικνύεται ως κρίσιμη, καθώς η ετερογένεια των συσκευών και των εφαρμογών απαιτεί την ενοποίηση τεχνολογιών όπως το MQTT και το LoRaWAN (πρωτόκολλο). Σύγχρονες αρχιτεκτονικές, όπως το LoRaFarM, αποδεικνύουν την αποτελεσματικότητα των modular, χαμηλού κόστους λύσεων, ενώ η χρήση πολυπρωτόκολλων gateways προσφέρει σημαντική κλιμακωσιμότητα και ευελιξία (Codeluppi et al., 2020). Επιπλέον, εμπορικές λύσεις όπως η HiveMQ προτείνουν πρακτικούς τρόπους διασύνδεσης LoRaWAN (πρωτόκολλο) και MQTT, διευκολύνοντας την ενιαία διαχείριση δεδομένων από ετερογενή δίκτυα.

Η συγκριτική μελέτη πρωτοκόλλων επικοινωνίας στον αγροτικό τομέα (Yavuz et al., 2022) καταδεικνύει ότι το LoRaWAN (πρωτόκολλο) υπερέχει σε εφαρμογές που απαιτούν μεγάλη εμβέλεια και χαμηλή κατανάλωση ενέργειας, ενώ το MQTT προσφέρει αξιοπιστία και ταχύτητα στη μετάδοση δεδομένων. Η τάση για web-based πλατφόρμες με φιλικές προς τον χρήστη διεπαφές είναι εμφανής στη βιβλιογραφία. Πλατφόρμες όπως το ThingsBoard και εφαρμογές όπως αυτή των Sivapriyan et al. (2019) δίνουν έμφαση στην οπτικοποίηση δεδομένων σε πραγματικό χρόνο και στην προσαρμοστικότητα των dashboards, διευκολύνοντας τη λήψη αποφάσεων από τους αγρότες. Παράλληλα, cloud-based συστήματα όπως το Agri-Info αποδεικνύουν ότι η αξιοποίηση του υπολογιστικού νέφους μπορεί να μειώσει σημαντικά το κόστος και να βελτιώσει την απόδοση των αγροτικών εφαρμογών, ενώ η ενοποίηση IoT, μηχανικής μάθησης και cloud computing προσφέρει νέες δυνατότητες για την αυτοματοποίηση και τη βιωσιμότητα της γεωργικής παραγωγής (Singh et al., 2024).

Η γεωχωρική διαχείριση και οι τεχνολογίες geofencing αποκτούν όλο και μεγαλύτερη σημασία στη σύγχρονη γεωργία. Η χρήση GPS tracking και geofencing επιτρέπει την ακριβή παρακολούθηση της θέσης εξοπλισμού, ζώων και καλλιεργειών, προσφέροντας δυνατότητες για αυτοματοποιημένες ειδοποιήσεις και βελτιστοποίηση της διαχείρισης των εκμεταλλεύσεων (Giri, 2025· Qazi et al., 2020). Η ενσωμάτωση αυτών των τεχνολογιών σε IoT πλατφόρμες ενισχύει την ασφάλεια και την αποδοτικότητα, ενώ η χρήση GPS-guided μηχανημάτων συμβάλλει στην ακριβή γεωργία και στη βελτίωση των αποδόσεων (Bartmann, 2020).

Σημαντική πρόοδος καταγράφεται και στην ενσωμάτωση τεχνολογιών machine learning στην αγροτική IoT. Ερευνητικές εργασίες παρουσιάζουν συστήματα που συνδυάζουν αισθητήρες IoT με αλγόριθμους πρόβλεψης και αυτοματοποιημένης διαχείρισης, επιτυγχάνοντας υψηλή ακρίβεια σε προβλέψεις και συστάσεις για τη βελτιστοποίηση της παραγωγής και της κατανάλωσης πόρων (Gnanadurai et al., 2024· Alturif et al., 2024). Η χρήση αλγορίθμων όπως LightGBM, Decision Tree και Random Forest σε μεγάλα datasets IoT οδηγεί σε εντυπωσιακά ποσοστά ακρίβειας, ενισχύοντας την αξιοπιστία των συστημάτων ακριβούς γεωργίας.

Επιπλέον, η βιομηχανία έχει αναπτύξει εμπορικές πλατφόρμες όπως το MyFarmWeb και το AGRIVI, που προσφέρουν ολοκληρωμένες λύσεις διαχείρισης αγροτικών δεδομένων, διαλειτουργικότητα με IoT συσκευές και εργαλεία υποστήριξης αποφάσεων βασισμένα σε τεχνητή νοημοσύνη. Παράλληλα, πλατφόρμες όπως το SAP Intelligent Agriculture εστιάζουν στην επιχειρηματική αξιοποίηση των αγροτικών δεδομένων, προωθώντας τη βιωσιμότητα και την αποδοτικότητα μέσω προηγμένων μοντέλων ανάλυσης.

Παρά τις σημαντικές προόδους, η βιβλιογραφία αναγνωρίζει προκλήσεις όπως η διαλειτουργικότητα, η ασφάλεια δεδομένων και η κλιμακωσιμότητα των λύσεων. Η ανάγκη για ευέλικτες hardware και software πλατφόρμες που να μπορούν να προσαρμοστούν σε αγροτικές εκμεταλλεύσεις διαφορετικής κλίμακας παραμένει κρίσιμη (Ahmed et al., 2022). Ταυτόχρονα, η έμφαση στη βιωσιμότητα και στη βελτιστοποίηση της χρήσης των φυσικών πόρων αποτελεί βασικό πυλώνα της σύγχρονης έρευνας (Sewnet Getahun et al., 2024).

Συνοψίζοντας, η βιβλιογραφία υπογραμμίζει ότι τα συστήματα διαχείρισης αγροτικών συσκευών IoT εξελίσσονται προς την κατεύθυνση της πολυπρωτόκολλης

ενοποίησης, της αξιοποίησης cloud τεχνολογιών, της ενσωμάτωσης γεωχωρικών λειτουργιών και της χρήσης προηγμένων αλγορίθμων ανάλυσης. Η ανάπτυξη ολοκληρωμένων πλατφορμών που συνδυάζουν όλες αυτές τις δυνατότητες, όπως προτείνεται στην παρούσα εργασία, αποτελεί σημαντική συνεισφορά στον τομέα της έξυπνης γεωργίας και απαντά στις προκλήσεις της σύγχρονης αγροτικής παραγωγής.

3. ΑΝΑΛΥΣΗ ΑΠΑΙΤΗΣΕΩΝ

Η ανάλυση απαιτήσεων για το σύστημα διαχείρισης αγροτικών εκμεταλλεύσεων με IoT συσκευές επικεντρώνεται στη δυνατότητα του χρήστη να καταχωρεί χωράφια, να ορίζει γεωγραφική τοποθεσία και να τοποθετεί διάφορες συσκευές που εξυπηρετούν διαφορετικές λειτουργίες. Κάθε χρήστης θα πρέπει να μπορεί να δημιουργεί νέο χωράφι εισάγοντας όνομα και συντεταγμένες ή όρια στο χάρτη, να επεξεργάζεται ή να διαγράφει υπάρχοντα χωράφια και να βλέπει όλα τα χωράφια του σε μια ενιαία οπτική αναπαράσταση.

Σε κάθε χωράφι, ο χρήστης μπορεί να τοποθετήσει πολλαπλές συσκευές, κάθε μία με συγκεκριμένη λειτουργία και θέση εντός του χωραφιού. Οι συσκευές αυτές μπορεί να είναι αισθητήρες περιβαλλοντικών παραμέτρων, όπως θερμοκρασίας ή υγρασίας, τοποθετημένοι σε διάφορα σημεία του χωραφιού, ή συσκευές παρακολούθησης ζώων με δυνατότητα live map data και ενημέρωση θέσης με συγκεκριμένη συχνότητα. Το σύστημα πρέπει να υποστηρίζει την προσθήκη, μετακίνηση, επεξεργασία και διαγραφή συσκευών, καθώς και την εύκολη ενσωμάτωση νέων τύπων συσκευών στο μέλλον.

Επιπλέον, η εφαρμογή οφείλει να παρέχει πραγματικού χρόνου συλλογή, αποθήκευση και οπτικοποίηση των δεδομένων που λαμβάνονται από τις συσκευές, με γραφήματα και χάρτες που να απεικονίζουν τις τρέχουσες μετρήσεις. Ειδικά για τις συσκευές παρακολούθησης ζώων, είναι απαραίτητη η προβολή της κίνησης σε χάρτη με συχνή ενημέρωση, καθώς και η δυνατότητα ορισμού γεωγραφικών ορίων (geofencing) και ειδοποιήσεων σε περίπτωση παραβίασης αυτών των ορίων.

Επιπλέον, το σύστημα πρέπει να προσφέρει αυτοματοποιημένες ειδοποιήσεις για ανωμαλίες (π.χ. υπέρβαση ορίων θερμοκρασίας, έξοδος ζώου από όρια). Η διαχείριση χρηστών πρέπει να είναι ασφαλής, και να λαμβάνει χώρα αυθεντικοποίηση χρηστών κατά την διάρκεια της εισόδου στο σύστημα. Η εφαρμογή πρέπει να είναι επεκτάσιμη, να υποστηρίζει πολλαπλά πρωτόκολλα επικοινωνίας (MQTT, LoRaWAN (πρωτόκολλο) κ.ά.), να έχει φιλικό και responsive περιβάλλον εργασίας για χρήση από υπολογιστές και κινητές συσκευές και να διασφαλίζει την ασφάλεια και την προστασία των δεδομένων σύμφωνα με τα πρότυπα (π.χ. GDPR).

Τέλος, η συντήρηση και η υποστήριξη του συστήματος πρέπει να είναι εύκολη, με ενσωματωμένο σύστημα βοήθειας και δυνατότητα μελλοντικών επεκτάσεων.

3.1 Τεχνολογία Backend

Η έκδοση 12 του Laravel εισάγει νέες βελτιστοποιήσεις στη δομή φακέλων και βελτιωμένη υποστήριξη PHP 8.3+, μειώνοντας περαιτέρω την πολυπλοκότητα και βελτιώνοντας την απόδοση σε πραγματικού χρόνου εφαρμογές. Το σύστημα βασίζεται σε Laravel Artisan Console Commands που εκτελούνται στο background μέσω supervisor ή systemd για συνεχή MQTT επικοινωνία, ενώ οι προχωρημένες τεχνικές caching και rate limiting προστατεύουν το API. Το σύστημα διατηρεί Laravel Sanctum authentication και AES-256 κρυπτογράφηση, παρέχοντας Limitless theme-based admin interfaces με vanilla JavaScript και Blade templates, ReactPHP-based MQTT client και webhook handlers για ολοκληρωμένη IoT επικοινωνία μέσω background command processing και σύστημα ουρών.

3.2 Τεχνολογία Frontend

Το Limitless admin theme από ThemeForest προσφέρει έτοιμη HTML/CSS/JavaScript βάση με προχωρημένα UI components. Το theme περιλαμβάνει responsive layouts, data tables, charts, και dashboard widgets που συνδέονται με Laravel μέσω vanilla JavaScript ή jQuery. Η αρχιτεκτονική πραγματικού χρόνου επικοινωνίας υλοποιείται μέσω 10-δευτερόλεπτων AJAX polling calls προς Laravel API endpoints, χρησιμοποιώντας τα built-in JavaScript components του theme για data visualization και dashboard updates. Οι τεχνικές

βελτιστοποίησης περιλαμβάνουν lazy loading μέσω του theme's data loading system, browser caching για static assets, και progressive loading για μεγάλα datasets.

Το backend διαχειρίζεται τα δεδομένα IoT μέσω Laravel σύστημα ουρών και Console Commands που επεξεργάζονται εισερχόμενα webhooks από The Things Stack network server. Τα scheduled tasks εκτελούνται κάθε 10 δευτερόλεπτα μέσω Laravel's sub-minute scheduling functionality, επεξεργάζοντας sensor readings και διαχειριζόμενα background jobs για data processing και notification handling. Η αρχιτεκτονική polling-based approach εξασφαλίζει συνεχή ενημέρωση των dashboard interfaces χωρίς την πολυπλοκότητα WebSocket connections, ενώ παράλληλα διατηρεί χαμηλό φόρτο διακομιστή.

3.3 Ολοκληρωμένη Προσέγγιση

Η στοίβα τριών επιπέδων υλοποιεί ένα communication pipeline που διασφαλίζει συνεχή IoT data flow. Στο backend επίπεδο, ένα specialized Laravel Console Command εκτελείται συνεχώς μέσω supervisor ή systemd, παρακολουθώντας active MQTT subscriptions από συνδεδεμένες συσκευές. Όταν εντοπίζονται εισερχόμενα sensor data, το command αυτόματα dispatcher Laravel Queue jobs που επεξεργάζονται τα δεδομένα asynchronously, ενημερώνουν τη βάση δεδομένων και προετοιμάζουν τις πληροφορίες για frontend consumption.

Το Limitless theme frontend συνδέεται με αυτή την pipeline μέσω structured AJAX polling κάθε 10 δευτερόλεπτα, ανακτώντας processed data από Laravel RESTful API endpoints. Η JavaScript integration χρησιμοποιεί τα native components του theme για πραγματικού χρόνου dashboard updates, chart refresh και notification handling, διατηρώντας responsive εμπειρία χρήστη χωρίς την πολυπλοκότητα persistent connections. Παράλληλα, webhook endpoints υποδέχονται εισερχόμενα δεδομένα από The Things Stack network server, τα οποία επίσης routing στο σύστημα ουρών για consistent data processing.

Το layer συσκευών (Wokwi simulation και LoRaWAN (πρωτόκολλο) devices) επικοινωνεί άμεσα με αυτήν την unified processing pipeline, εξασφαλίζοντας ότι όλα τα IoT δεδομένα - είτε από MQTT brokers είτε από LoRaWAN (πρωτόκολλο) webhooks - ακολουθούν το ίδιο data flow pattern προς το frontend interface.

3.4 Δυνατά Σημεία

Η μεθοδολογία προσφέρει ευελιξία στην προσθήκη νέων τεχνολογιών, δυνατότητα προσαρμογής σε διαφορετικά αγροτικά περιβάλλοντα και πλήρη κάλυψη των απαιτήσεων της ακριβούς γεωργίας, από την εκπαίδευση και το prototyping έως την παραγωγική λειτουργία σε μεγάλη κλίμακα. Η αρχιτεκτονική τριών επιπέδων εξασφαλίζει την ανεξαρτησία κάθε στοιχείου, διευκολύνοντας τη συντήρηση και την επέκταση του συστήματος.

Η χρήση του Laravel 12 με τα εξελιγμένα χαρακτηριστικά του, όπως το βελτιωμένο σύστημα ουρών για background processing και οι προηγμένοι μηχανισμοί ασφάλειας μέσω Sanctum authentication, προσφέρει σταθερό και αξιόπιστο backend. Το Limitless theme παρέχει professional UI components με άμεση JavaScript ενσωμάτωση, επιτρέποντας μεγαλύτερη ευελιξία στην προσαρμογή και ταχύτερη ανάπτυξη custom features χωρίς την πολυπλοκότητα heavyweight admin frameworks.

Το background MQTT command system που τρέχει συνεχώς εξασφαλίζει αδιάλειπτη επικοινωνία με IoT συσκευές, ενώ η polling-based frontend αρχιτεκτονική διατηρεί χαμηλό φόρτο διακομιστή και υψηλή αξιοπιστία. Η ενοποίηση webhook support για The Things Stack με το Laravel Queue system δημιουργεί μια robust data pipeline

που χειρίζεται αποτελεσματικά μεγάλους όγκους sensor data. Η modular δομή του Limitless theme επιτρέπει selective component loading και customization χωρίς περιττή πολυπλοκότητα, προσφέροντας ideal balance μεταξύ functionality και maintainability.

3.5 Πιθανές Αδυναμίες και Περιορισμοί

Παρά τα ισχυρά χαρακτηριστικά της μεθοδολογίας, υπάρχουν ορισμένες αδυναμίες που πρέπει να ληφθούν υπόψη. Η εξάρτηση από πολλαπλές τεχνολογίες (Laravel 12, Limitless theme, vanilla JavaScript, LoRaWAN (πρωτόκολλο), The Things Stack) δημιουργεί πολυπλοκότητα στη διαχείριση και απαιτεί διαφορετική εξειδίκευση από τους προγραμματιστές, ιδιαίτερα στην ενσωμάτωση του theme με τα Laravel APIs.

Η ενσωμάτωση του Limitless theme μπορεί να παρουσιάσει challenges στο webpack configuration και στη διαχείριση dependencies, όπως φαίνεται από τα compilation errors που συχνά εμφανίζονται κατά την initial setup. Επιπλέον, η ανάγκη για custom JavaScript integration για την επικοινωνία με Laravel RESTful endpoints απαιτεί προσεκτικό handling των AJAX calls και error management.

Οι περιορισμοί του LoRaWAN (πρωτόκολλο) και η εξάρτηση από το The Things Stack παραμένουν σημαντικοί παράγοντες, ιδιαίτερα σε περιοχές με περιορισμένη network coverage. Το polling-based approach του frontend, παρότι αξιόπιστο, μπορεί να δημιουργήσει latency issues σε περιπτώσεις που απαιτούν άμεση response, ενώ τα background MQTT εντολές χρειάζονται προσεκτική monitoring για να αποφευχθούν memory leaks σε long-running processes.

Το συνολικό κόστος lifecycle μπορεί να αυξηθεί λόγω των licensing fees του theme και της ανάγκης για εξειδικευμένη συντήρηση των πολλαπλών τεχνολογικών layers. Επιπλέον, το scalability του συστήματος εξαρτάται σημαντικά από την απόδοση του

Queue system και τη διαχείριση concurrent MQTT connections σε μεγάλης κλίμακας deployment

4. ΑΝΑΛΥΣΗ ΚΑΙ ΥΛΟΠΟΙΗΣΗ

Στο παρόν κεφάλαιο παρουσιάζεται η πλήρης υλοποίηση του ολοκληρωμένου συστήματος διαχείρισης IoT γεωργικών συσκευών, βασισμένο στο πραγματικό κώδικα από το GitHub repository "ptyxiaki-teliko". Το κεφάλαιο εστιάζει στην παρουσίαση του τελικού προϊόντος και την επίδειξη των δυνατοτήτων της υλοποιημένης πλατφόρμας μέσω συγκεκριμένων παραδειγμάτων και εφαρμογών που αντικατοπτρίζουν τις σύγχρονες τάσεις στην precision agriculture.

Η ανάπτυξη ακολούθησε μεθοδολογική προσέγγιση που εστιάζει στην πρακτική εφαρμογή της precision agriculture μέσω της ενσωμάτωσης διαφορετικών επιπέδων τεχνολογίας. Από την εκπαιδευτική χρήση του Wokwi simulator για rapid prototyping έως την παραγωγική λειτουργία με LoRaWAN (πρωτόκολλο) συσκευές, το σύστημα παρέχει ολοκληρωμένες λύσεις που ενσωματώνουν τις αρχές της σύγχρονης γεωργίας με data-driven decision making. Η διαμόρφωση του συστήματος επιτρέπει την παρακολούθηση και διαχείριση αγροτικών δραστηριοτήτων σε πραγματικό χρόνο, προσφέροντας στους χρήστες εργαλεία ανάλυσης και αυτοματισμού που βελτιστοποιούν την παραγωγικότητα.

Αρχιτεκτονική Multi-Broker Communication

Το σύστημα λειτουργεί με δύο δημόσιους MQTT brokers - EMQX και Mosquitto - προσφέροντας ευελιξία επιλογής και redundancy για τη διασφάλιση αδιάλειπτης επικοινωνίας. Παράλληλα, υποστηρίζεται LoRaWAN (πρωτόκολλο) επικοινωνία μέσω The Things Stack network server που απαιτεί webhook configuration για την αποστολή δεδομένων προς την πλατφόρμα. Αυτή η multi-protocol προσέγγιση

επιτρέπει στους χρήστες να επιλέξουν την κατάλληλη τεχνολογία ανάλογα με τις συγκεκριμένες ανάγκες της εκμετάλλευσης, από short-range MQTT συσκευές έως long-range LoRaWAN (πρωτόκολλο) sensors.

Ιεραρχική Δομή Δεδομένων

Η παρουσίαση του συστήματος περιλαμβάνει λεπτομερή τεκμηρίωση της ιεραρχικής δομής που ακολουθεί το μοντέλο: User → Land → Device → Sensor. Κάθε χρήστης μπορεί να ορίσει πολλαπλές γεωργικές εκτάσεις (Lands) μέσω polygon definition στο Leaflet map interface, καθορίζοντας γεωγραφικά όρια και χαρακτηριστικά για κάθε περιοχή. Στη συνέχεια, μπορεί να τοποθετήσει συσκευές (Devices) εντός κάθε έκτασης, καθορίζοντας τα στοιχεία σύνδεσης του broker και επιλέγοντας μεταξύ MQTT ή webhook connectivity. Οι αισθητήρες (Sensors) δημιουργούνται αυτόματα κατά την πρώτη λήψη payload από το MQTT stream, εξασφαλίζοντας plug-and-play functionality για νέες συσκευές.

Μέσω προσεκτικής ανάλυσης των υλοποιημένων χαρακτηριστικών, το κεφάλαιο στοχεύει στην αποτύπωση της πρακτικής αξίας του συστήματος και στην επίδειξη του τρόπου με τον οποίο οι σύγχρονες τεχνολογίες μπορούν να συνεισφέρουν στην εξέλιξη του αγροτικού τομέα μέσω automated sensor discovery, πραγματικού χρόνου monitoring και geospatial analytics.

4.1 BackEnd Υλοποίηση

Το backend του συστήματος αναπτύχθηκε χρησιμοποιώντας το Laravel 12 framework, ακολουθώντας τις αρχές του Model-View-Controller (MVC) pattern και εφαρμόζοντας βέλτιστες πρακτικές σύγχρονης ανάπτυξης IoT εφαρμογών. Η οργάνωση του έργου στηρίχθηκε σε μια προσεκτικά σχεδιασμένη δομή που προωθεί τη σαφήνεια, την επαναχρησιμοποίηση κώδικα και την ευκολία συντήρησης, ειδικά για διαχείριση μεγάλου όγκου sensor data.

4.1.1 Ιεραρχική Δομή Models και Data Architecture

Η κεντρική αρχιτεκτονική του backend οργανώθηκε σε εξειδικευμένες ενότητες που ακολουθούν την ιεραρχία User → Land → Device → Sensor. Στον κατάλογο app/Models/ βρίσκονται τα Eloquent models που αντιπροσωπεύουν τις βασικές οντότητες του συστήματος:

- User Model: Διαχειρίζεται τους εγγεγραμμένους χρήστες με Sanctum authentication integration
- Land Model: Αντιπροσωπεύει γεωργικές εκτάσεις με polygon definitions και geofencing capabilities
- Device Model: Χειρίζεται IoT συσκευές με MQTT/webhook connection parameters και broker selection
- Sensor Model: Auto-generated από το πρώτο MQTT payload, διαχειρίζεται πραγματικού χρόνου sensor readings
- MqttBroker Model: Υποστηρίζει EMQX και Mosquitto public brokers με connection pooling
- LoRaWAN (πρωτόκολλο)Broker Model: Integration με The Things Stack για webhook-based communication

4.1.2 Multi-Protocol Communication Layer

Το επίπεδο `app/Services/` υλοποιεί κεντρικοποιημένη επιχειρηματική λογική που διαχειρίζεται την πολυπρωτόκολλη επικοινωνία. Το `MqttService` χειρίζεται την επικοινωνία με δύο δημόσιους brokers (EMQX και Mosquitto), ενώ το `LoRaWAN (πρωτόκολλο)Service` διαχειρίζεται `webhook integrations` από `The Things Stack network server`. Το `GeofencingService` υλοποιεί `ray-casting` αλγορίθμους για πραγματικού χρόνου παρακολούθηση αν οι συσκευές βρίσκονται εντός των ορισμένων `polygon boundaries`.

4.1.3 Background Processing Architecture

Κρίσιμο στοιχείο της αρχιτεκτονικής είναι η παράλληλη εκτέλεση `Laravel Console Commands` που τρέχουν στο `background` μέσω `supervisor` ή `systemd`. Το κεντρικό `MQTT Listener Command` εκτελείται συνεχώς, παρακολουθώντας `subscriptions` από τις συνδεδεμένες συσκευές και `dispatching Laravel Queue jobs` για `asynchronous data processing`. Όταν ληφθεί το πρώτο `payload` από μια νέα συσκευή, το σύστημα αυτόματα δημιουργεί `Sensor records` στη βάση δεδομένων, εξασφαλίζοντας `plug-and-play functionality`.

4.1.4 API Architecture και Webhook Integration

Η διαχείριση των `HTTP requests` πραγματοποιείται μέσω του καταλόγου `app/Http/Controllers/`, ο οποίος περιέχει εξειδικευμένους `controllers` για:

- `DeviceController`: Διαχείριση `MQTT` και `LoRaWAN (πρωτόκολλο) device registration`
- `WebhookController`: Handling εισερχόμενων δεδομένων από `The Things Stack`
- `ApiController`: `RESTful endpoints` για `frontend communication` με `10-second polling support`
- `LandController`: `Polygon-based geospatial operations` με `Leaflet integration`

Στο database/migrations/ ορίζονται οι δομές που υποστηρίζουν GeoJSON polygons για land boundaries, MQTT topic arrays για multi-subscription devices, και webhook signature verification για secure LoRaWAN (πρωτόκολλο) communication. Η δρομολόγηση οργανώθηκε στον κατάλογο routes/ με ειδικούς webhook endpoints που δέχονται POST requests από external IoT networks και API routes που υποστηρίζουν τη JavaScript polling architecture του frontend.

4.1.5 Eloquent Models και Relationships

Το σύστημα βασίζεται σε τρία κύρια Eloquent models που αποτελούν τον πυρήνα της δομής δεδομένων και αντιπροσωπεύουν τις βασικές οντότητες της αγροτικής IoT πλατφόρμας.

Το Device Model αποτελεί τον κεντρικό κορμό του συστήματος, αντιπροσωπεύοντας τις IoT συσκευές που αναπτύσσονται στο αγροτικό πεδίο. Το model περιλαμβάνει εκτεταμένες παραμέτρους σύνδεσης όπως client_id, use_ssl, connection_broker και mqtt_topics array που υποστηρίζει πολλαπλά MQTT topics ταυτόχρονα. Οι built-in methods όπως setOnline() και setOffline() διαχειρίζονται αυτόματα την κατάσταση της συσκευής και το last_seen_at timestamp. Η σχέση με το Land model μέσω belongsTo() εξασφαλίζει τη γεωχωρική τοποθέτηση των συσκευών.

Το Sensor Model διαχειρίζεται τα δεδομένα αισθητήρων με προχωρημένες δυνατότητες threshold monitoring και alert generation. Οι μέθοδοι isValueInRange() και getAlertStatus() παρέχουν αυτόματο έλεγχο ορίων για numeric values, ενώ η updateReading() method διατηρεί timestamps για κάθε μέτρηση. Η getFormattedValue() επιστρέφει τιμές με τις κατάλληλες μονάδες μέτρησης και υποστηρίζει JSON/array values για complex sensor data.

Το Land Model αντιπροσωπεύει τις γεωργικές εκτάσεις με πλήρη γεωχωρική υποστήριξη μέσω geojson casting που αποθηκεύει GeoJSON polygons για ακριβή γεωγραφικά όρια. Το location array field υποστηρίζει multiple coordinate formats, ενώ το color field επιτρέπει visual distinction στις χαρτογραφικές διεπαφές. Η σχέση hasMany() με τις συσκευές υποστηρίζει πολλαπλές IoT installations ανά έκταση.

Οι σχέσεις μεταξύ των models δημιουργούν την ιεραρχία: User → Land → Device → Sensor, επιτρέποντας στο σύστημα να υποστηρίζει τόσο single-farm installations όσο και enterprise-level agricultural operations με εκατοντάδες συσκευές.

4.2 Services Layer - Επιχειρηματική Λογική

Η επιχειρηματική λογική κεντριοποιείται σε specialized services που διαχειρίζονται complex operations, ακολουθώντας τις αρχές του service-oriented design για maintainability και testability. Αυτή η layered architecture επιτρέπει τη σαφή διαχωρισμό ευθυνών μεταξύ των διαφορετικών επιπέδων του συστήματος.

4.2.1 GeofencingService - Γεωχωρική Παρακολούθηση

Το GeofencingService υλοποιεί τον ray-casting αλγόριθμο για γεωχωρική παρακολούθηση, ελέγχοντας αν τα GPS coordinates των συσκευών βρίσκονται εντός των καταχωρημένων polygon boundaries που ορίζονται στα Land models. Το service χρησιμοποιεί complex mathematical calculations για να καθορίσει την εσωτερική ή εξωτερική θέση συσκευών relative προς τα user-defined geofences, παρέχοντας πραγματικού χρόνου alerts όταν συσκευές παραβιάζουν τα καθορισμένα όρια.

4.2.2 Multi-Protocol Communication Services

Το MqttDeviceService διαχειρίζεται την αυτόματη device discovery μέσω MQTT messages από δύο δημόσιους brokers (EMQX και Mosquitto), εκτελώντας updateOrCreate() operations και συγχρονίζοντας αισθητήρες μέσω της

`syncSensorsFromArduino()` μεθόδου. Παράλληλα, το `WebhookService` χειρίζεται εισερχόμενα δεδομένα από The Things Stack network server, parsing JSON payloads και δημιουργώντας αυτόματα Sensor entries για νέες συσκευές.

4.2.3 Background Processing Integration

Τα services συνεργάζονται στενά με το background MQTT listener command, το οποίο εκτελείται συνεχώς στον server μέσω supervisor ή systemd και forwards τα εισερχόμενα μηνύματα στα κατάλληλα services για asynchronous επεξεργασία. Το `QueueService` διαχειρίζεται τη διαδικασία dispatching Laravel Queue jobs που επεξεργάζονται large volumes IoT data χωρίς να επηρεάζουν την απόδοση της web application.

4.2.4 Device Management και Auto-Discovery

Το `DeviceService` υλοποιεί τη λογική της αυτόματης δημιουργίας sensor records όταν λαμβάνεται το πρώτο MQTT payload από μια νέα συσκευή. Αυτό το plug-and-play functionality εξασφαλίζει ότι οι χρήστες δεν χρειάζεται να διαμορφώσουν manually κάθε sensor, καθώς το σύστημα αυτόματα ανιχνεύει και καταχωρεί τα διαθέσιμα sensor types από τη device configuration.

Αυτή η service-oriented αρχιτεκτονική παρέχει modular functionality και εξασφαλίζει easy testing και maintenance του κώδικα, ενώ το background command system διατηρεί την πλατφόρμα ενεργή και responsive για πραγματικού χρόνου agricultural monitoring με support για multiple communication protocols και automatic sensor discovery.

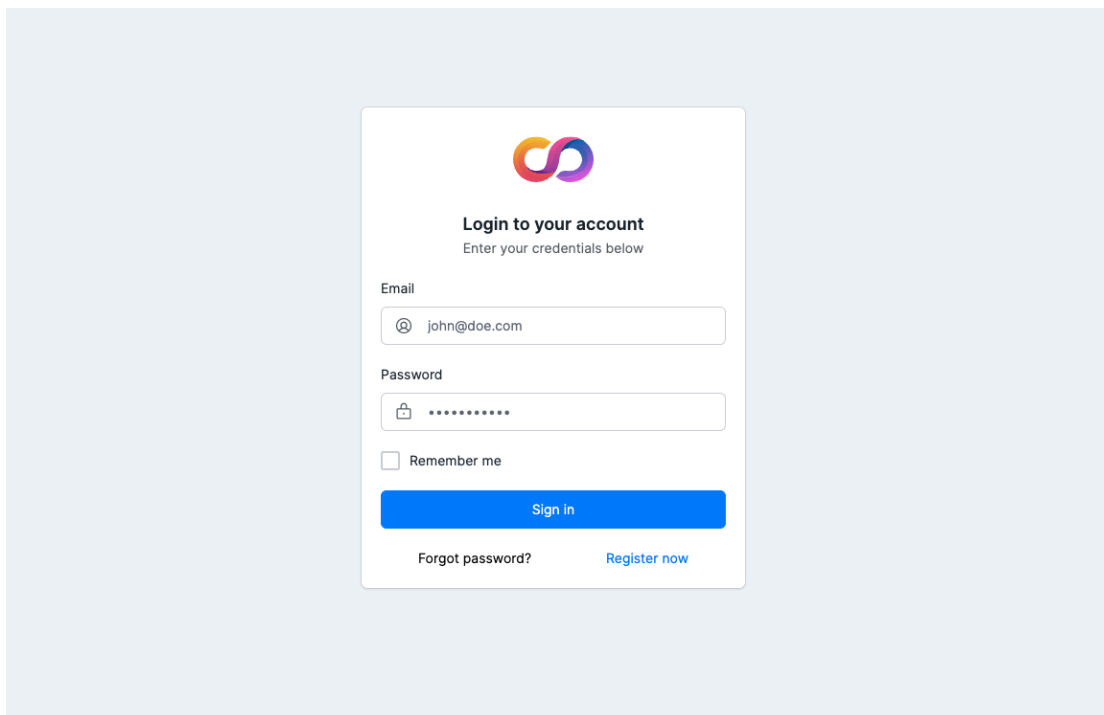
4.3 Frontend Υλοποίηση με Limitless Theme

Theme Architecture και JavaScript Integration

Το frontend υλοποιήθηκε χρησιμοποιώντας το Limitless admin theme από ThemeForest, το οποίο βασίζεται σε vanilla JavaScript με Ajax επικοινωνία. Η συγκεκριμένη τεχνολογία χρησιμοποιήθηκε καθώς επιτρέπει την κατασκευή σύγχρονων, responsive διαδικτυακών εφαρμογών με μέγιστη ευελιξία προσαρμογής και άμεση επικοινωνία με το Laravel 12 backend μέσω RESTful APIs. Το Limitless theme βασίζεται σε μια τεχνολογική στοίβα που περιλαμβάνει Bootstrap 5 για responsive framework, jQuery και vanilla JavaScript για διαδραστικές λειτουργίες, Chart.js για data visualization, και Leaflet.js για χαρτογραφική υποστήριξη.

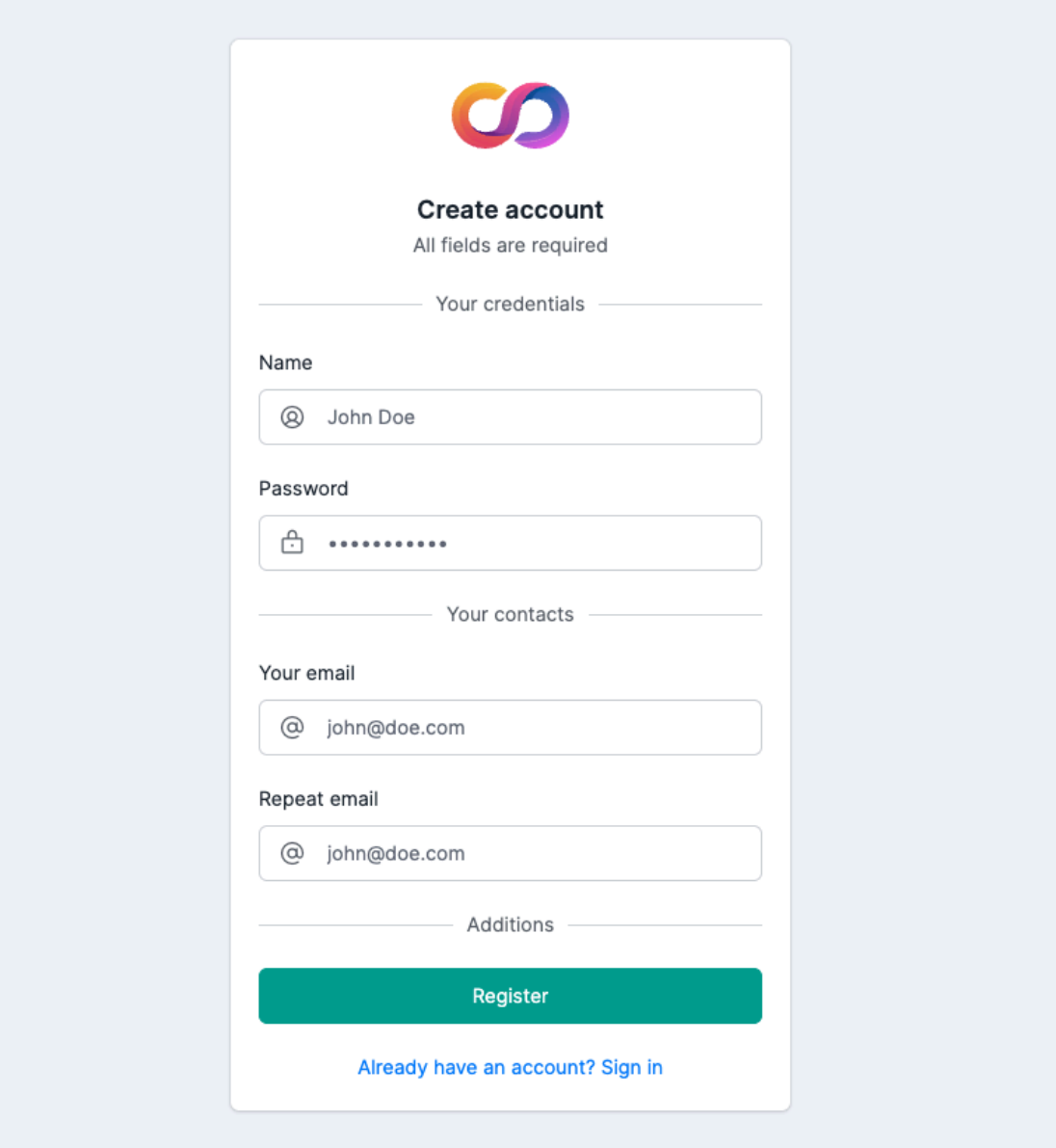
Authentication System

Login Page - Η σελίδα εισόδου χρησιμοποιεί τη διάταξη ταυτοποίησης του θέματος Limitless με ενσωμάτωση Laravel Sanctum για secure token-based authentication(Εικόνα 1). Η διεπαφή περιλαμβάνει modern form validation, remember me functionality, και responsive design για mobile και desktop συσκευές.



Εικόνα 1: Αρχική φόρμα login του χρήστη

Register Page - Η διεπαφή εγγραφής νέων χρηστών με προχωρημένα validation components του theme, email verification workflow, και automatic redirection στο dashboard μετά την επιτυχή εγγραφή(Εικόνα 2).



The image shows a 'Create account' form with a logo at the top. The form is titled 'Create account' and includes the text 'All fields are required'. It is divided into sections: 'Your credentials' with fields for 'Name' (John Doe) and 'Password' (masked with dots); 'Your contacts' with fields for 'Your email' (john@doe.com) and 'Repeat email' (john@doe.com); and 'Additions' with a green 'Register' button and a link 'Already have an account? Sign in'.

Εικόνα 2: Φόρμα εγγραφής χρήστη στην εφαρμογή

Documentation και Support

How To Page - Comprehensive documentation interface που εξηγεί τη χρήση του συστήματος, από τη δημιουργία lands έως την παρακολούθηση sensors. Η σελίδα χρησιμοποιεί το theme's advanced typography και accordion components για structured information presentation(Εικόνα 3).

IoT Device Management - User Guide
Simple step-by-step instructions for monitoring your IoT devices

What This System Does

- **Monitor your devices:** Track temperature, humidity, location, and other sensors
- **Map visualizations:** See where your devices are located on a map
- **Alerts:** Get notified when sensors go outside normal ranges
- **Land boundaries:** Check if devices are in the right locations

Step 1: Setting Up a Land

A "Land" is an area where you want to monitor devices (like a farm field, building, or garden).

How to Create a Land:

1. **Go to Lands page:** Click "Lands" in the menu → "Create New Land"
2. **Fill in the details:**
 - **Land Name:** Give it a simple name (e.g., "North Field", "Greenhouse 1")
 - **Description:** What this area is used for
 - **Location:** You can add GPS coordinates if you know them
 - **Show boundaries (optional):** If you want to check if devices stay within an area
3. **Save your land:** Click "Create Land" → Your land is now ready for devices

[Create Your First Land](#)

Step 2: Setting Up Devices

Devices are your sensors that send data (temperature, humidity, etc.)

MQTT Devices	LoRaWAN Devices	Webhook Devices
<p>Best for: ESP32, Arduino with WiFi, Raspberry Pi</p> <p>Setup:</p> <ul style="list-style-type: none"> • Device ID unique name • Connection type MQTT • MQTT url: <code>mqtt://mqtt.eclipse.org</code> or <code>mqtt://mqtt:1883</code> • Topic: where device sends data <p><small>Note: MQTT only supports public servers without authentication. Use Mosquitto for production with credentials.</small></p>	<p>Best for: Long-range, low-power devices</p> <p>Setup:</p> <ul style="list-style-type: none"> • Set up in The Things Stack first • Connection type Webhook • Configure webhook in TTN • URL: <code>mqtt://mqtt.eclipse.org</code> 	<p>Best for: Custom devices with HTTP</p> <p>Setup:</p> <ul style="list-style-type: none"> • Connection type Webhook • Set webhook URL after creating • Send HTTP POST requests • JSON format required

[Add Your First Device](#)


What Your Device Should Send (Payload Formats)

MQTT Format 1 (Recommended)	MQTT Format 2 (Simple)	LoRaWAN Format
<pre>{ "sensor": { "type": "temperature", "value": "25.6 Celsius", "type": "humidity", "value": "48.2 percent", "type": "light", "value": "120 percent" } }</pre>	<pre>{ "temperature": 26.1, "humidity": 38.5, "light": 88.9 }</pre>	<pre>{ "link_message": { "device_address": { "data": { "temperature": 23.3, "humidity": 45.2, "light": 88 } } } }</pre>

Εικόνα 3: Αρχική φόρμα login του χρήστη








Land Management Interface

Lands Index Page - Το κεντρικό navigation item "Lands" οδηγεί σε comprehensive datatable που εμφανίζει όλες τις γεωργικές εκτάσεις του χρήστη. Η σελίδα χρησιμοποιεί το Limitless DataTables integration με server-side processing για αποδοτική διαχείριση μεγάλων datasets, πραγματικού χρόνου search functionality, και custom column rendering για γεωγραφική πληροφορία και συνδεδεμένες συσκευές. (Εικόνα 4)

Lands - View, Manage and Create Lands Create 

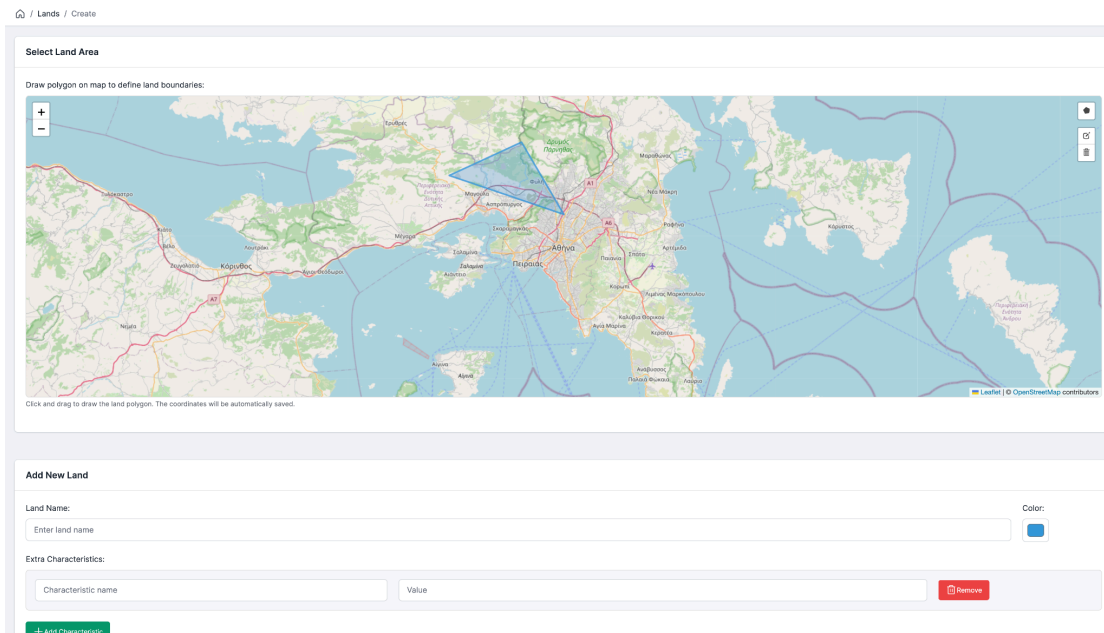
[Home](#) / [Lands](#) / All Lands

Your Lands

Land Information	Color	Status	Devices	Characteristics	Created	Actions
 Land 1 Created 3 days ago	 #3498db	 Enabled	 3 devices	asda: asdasdasd	Sep 21, 2025 08:22	  

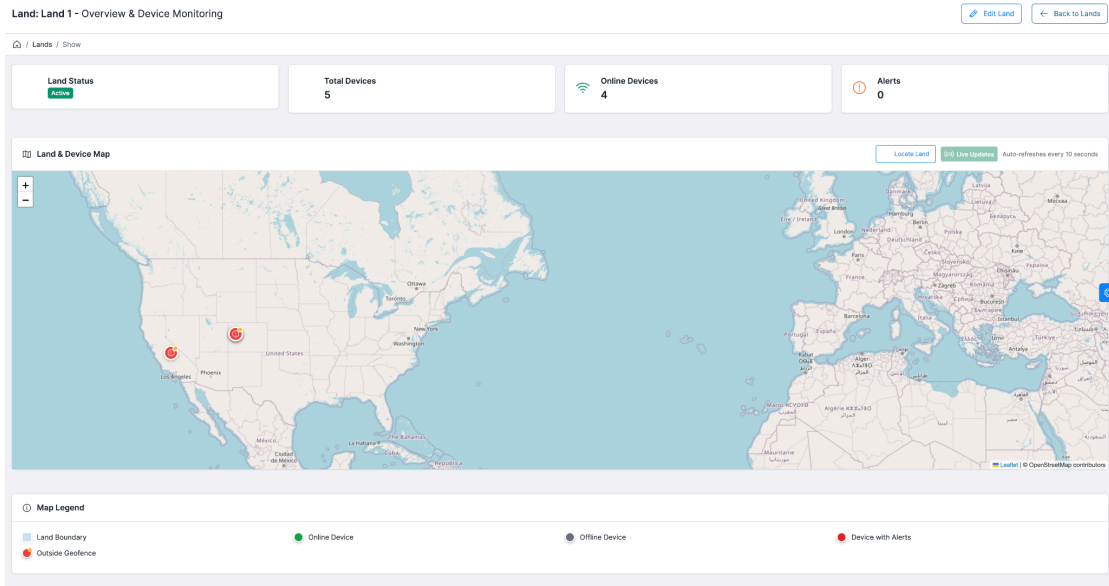
Εικόνα 4: Πίνακας γεωργικών εκτάσεων

Lands Create Page - Η διεπαφή δημιουργίας εκτάσεων ενσωματώνει διαδραστικό χαρτογραφικό σύστημα μέσω Leaflet.js με προχωρημένες δυνατότητες σχεδίασης πολυγώνου. Οι χρήστες μπορούν να σχεδιάσουν ακριβώς τα όρια των γεωργικών εκτάσεων χρησιμοποιώντας εργαλεία σχεδίασης, να επιλέξουν χρώματα για visual identification, και να εισάγουν metadata όπως όνομα, περιγραφή και πρόσθετες πληροφορίες μέσω του theme's advanced form components.(Εικόνα 5)



Εικόνα 5: Διαδραστικός χάρτης δημιουργίας εκτάσεων

Lands Show Page - Η πλέον προχωρημένη σελίδα του συστήματος που συνδυάζει χαρτογραφική απεικόνιση με πραγματικού χρόνου παρακολούθησης συσκευών. Ο χάρτης εμφανίζει τα όρια της έκτασης ως polygon overlay και τις τοποθεσίες όλων των συσκευών ως interactive markers με color-coded δείκτες κατάστασης. Κάθε device marker εμφανίζει popup με δεδομένα αισθητήρων σε πραγματικό χρόνο, κατάσταση μπαταρίας, και πληροφορίες συνδεσιμότητας, ενώ το GeofencingService παρέχει οπτικές ειδοποιήσεις όταν συσκευές βρίσκονται εκτός των ορίων της έκτασης.(Εικόνα 6 - Εικόνα 7)



Εικόνα 6: Χαρτογραφική απεικόνιση ορίων εκτάσεων

Map Legend

- Land Boundary
- Online Device
- Offline Device
- Device with Alerts
- Outside Geofence

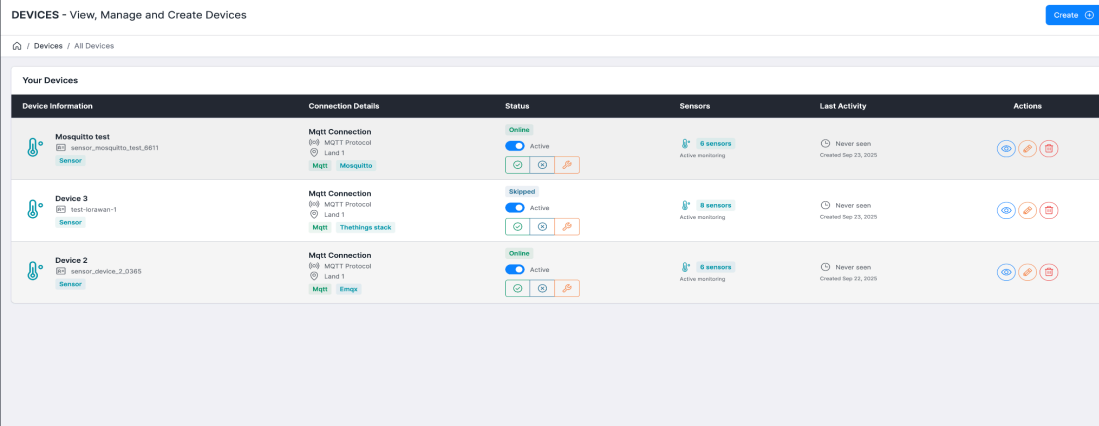
Devices in Land 1 5 devices

Device	Type	Status	Location	Sensors	Alerts	Actions
Device 2 sensor_device_2_0365	Sensor	Online	39.527685, -107.696663	6 sensors	Normal	Locate View
Device 3 test-lorawan-1	Sensor	Offline	36.859564, -119.076024	6 sensors	Normal	Locate View
Mosquito test sensor_floquetf2l_test_6611	Sensor	Online	39.527685, -107.696663	6 sensors	Normal	Locate View
PANDS DEVICE sensor_pands_device_0058	Sensor	Online	39.527685, -107.696663	6 sensors	Normal	Locate View
loraz2Test sensor_loraz2test_3065	Sensor	Online	No GPS data	5 sensors	Normal	Locate View

Εικόνα 7: Real-time monitoring συσκευών σε χάρτη

Device Management Interface

Devices Index Page - Εξελιγμένος πίνακας με πραγματικού χρόνου polling κάθε 10 δευτερόλεπτα που εμφανίζει όλες τις IoT συσκευές με visual δείκτες κατάστασης. Οι συσκευές εμφανίζονται με διακριτικά στοιχεία για τον τύπο (SENSOR_MONITOR, WEATHER_STATION), χρωματική κωδικοποίηση για την κατάσταση (online/offline), health percentage με progress bars, και quick action buttons για άμεση διαχείριση.(Εικόνα 8)



The screenshot displays a web interface titled "DEVICES - View, Manage and Create Devices". Below the title bar, there is a breadcrumb "Devices / All Devices" and a "Create" button. The main content area is titled "Your Devices" and contains a table with the following columns: Device Information, Connection Details, Status, Sensors, Last Activity, and Actions. The table lists three devices:

Device Information	Connection Details	Status	Sensors	Last Activity	Actions
Mosquito test sensor_mosquito_test_6611 Sensor	Mqtt Connection MQTT Protocol Land 1 Mqtt Mosquitto	Online Active	5 sensors Active monitoring	Never seen Created Sep 23, 2025	[Refresh] [Stop] [Delete]
Device 3 test-02aw20-1 Sensor	Mqtt Connection MQTT Protocol Land 1 Mqtt TheThings stack	Skipped Active	5 sensors Active monitoring	Never seen Created Sep 23, 2025	[Refresh] [Stop] [Delete]
Device 2 sensor_device_2_0365 Sensor	Mqtt Connection MQTT Protocol Land 1 Mqtt Edge	Online Active	5 sensors Active monitoring	Never seen Created Sep 22, 2025	[Refresh] [Stop] [Delete]

Εικόνα 8: Πίνακας συσκευών με κατάσταση σε πραγματικό χρόνο

Devices Create Page - Διεπαφή δημιουργίας νέων συσκευών με επιλογή αναπτυσσόμενης λίστας για MQTT brokers (EMQX/Mosquitto) ή LoRaWAN (πρωτόκολλο) webhook configuration, form validation για connection parameters, και automatic device type detection.(Εικόνα 9)

Basic Information

Device Name *

Device ID *

Unique identifier for device communication

Device Type *

Land *

Status *

Device is Active

Connection Settings

Connection Type *

[Configure MQTT connection settings for this device.](#)

Client ID

Leave empty to auto-generate

Connection Broker

Port

Port will auto-change: 1883 (standard) / 8883 (SSL)

Username

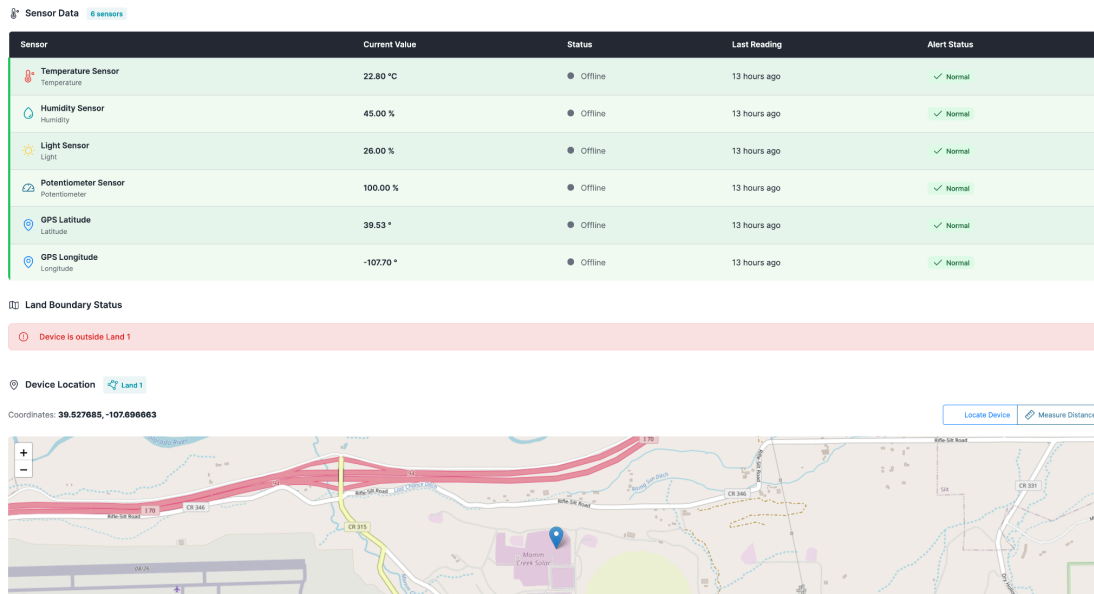
Password

Εικόνα 9: Φόρμα δημιουργίας νέας IoT συσκευής

Devices Show Page - Λεπτομερή προβολή κάθε συσκευής που ενσωματώνει δύο κρίσιμες λειτουργίες :

Auto-generated Sensors Grid: Όταν η συσκευή στέλνει MQTT payload, το σύστημα αυτόματα δημιουργεί sensor entries στη βάση δεδομένων και τα εμφανίζει σε πραγματικό χρόνο μέσω responsive cards με live value updates, threshold δείκτες κατάστασης, και historical data charts.

Geofencing Map Visualization: Διαδραστικός χάρτης που εμφανίζει την τρέχουσα θέση της συσκευής σε σχέση με τα όρια της assigned land, παρέχοντας οπτικές ειδοποιήσεις όταν η συσκευή βρίσκεται εκτός των ορίων μέσω του backend GeofencingService ray-casting calculations.(Εικόνα 10)



Εικόνα 10: Προβολή συσκευής με αισθητήρες και γεωεντοπισμό

Sensor Management Interface

Sensors Index Page - Comprehensive listing όλων των auto-discovered sensors με filtering capabilities κατά device, sensor type, και status. Η σελίδα χρησιμοποιεί το theme's advanced table components με πραγματικού χρόνου value updates κάθε 10 δευτερόλεπτα. (Εικόνα 11)

The screenshot shows the 'Sensors (20 total)' page with a 'Filters & Search' section. The table lists various sensors with columns for Sensor Type, Name, Device, Land, Current Value, Unit, Status, Alerts, Last Reading, and Actions. The sensors are categorized by type (Latitude, Longitude, Temperature, Humidity, Light, Potentiometer) and device (Mosquito test, Device 2).

Sensor Type	Name	Device	Land	Current Value	Unit	Status	Alerts	Last Reading	Actions
Latitude	Latitude Sensor Auto-created from MQTT topic...	Mosquito test sensor_mosquito_test_6611	Land 1	39.53	°	Enabled	Disabled	2 hours ago Sep 24, 2025 10:42:17	[Edit] [Refresh]
Longitude	Longitude Sensor Auto-created from MQTT topic...	Mosquito test sensor_mosquito_test_6611	Land 1	-107.70	°	Enabled	Disabled	2 hours ago Sep 24, 2025 10:42:17	[Edit] [Refresh]
Temperature	Temperature Sensor Auto-created from MQTT topic...	Mosquito test sensor_mosquito_test_6611	Land 1	22.80	°C	Enabled	Disabled	2 hours ago Sep 24, 2025 10:42:16	[Edit] [Refresh]
Humidity	Humidity Sensor Auto-created from MQTT topic...	Mosquito test sensor_mosquito_test_6611	Land 1	45.00	%	Enabled	Disabled	2 hours ago Sep 24, 2025 10:42:16	[Edit] [Refresh]
Light	Light Sensor Auto-created from MQTT topic...	Mosquito test sensor_mosquito_test_6611	Land 1	26.00	%	Enabled	Disabled	2 hours ago Sep 24, 2025 10:42:16	[Edit] [Refresh]
Potentiometer	Potentiometer Sensor Auto-created from MQTT topic...	Mosquito test sensor_mosquito_test_6611	Land 1	100.00	%	Enabled	Disabled	2 hours ago Sep 24, 2025 10:42:16	[Edit] [Refresh]
Temperature	Temperature Sensor Thermal sensor reading in Cels...	Device 2 sensor_device_2_0365	Land 1	22.80	°C	Enabled	Disabled	13 hours ago Sep 23, 2025 23:09:59	[Edit] [Refresh]
Humidity	Humidity Sensor Relative humidity percentage	Device 2 sensor_device_2_0365	Land 1	45.00	%	Enabled	Disabled	13 hours ago Sep 23, 2025 23:09:59	[Edit] [Refresh]
Light	Light Sensor Light intensity percentage	Device 2 sensor_device_2_0365	Land 1	26.00	%	Enabled	Disabled	13 hours ago Sep 23, 2025 23:09:59	[Edit] [Refresh]

Εικόνα 11: Πλήρης κατάλογος αισθητήρων με φίλτρα

Sensors Edit Page - Εξειδικευμένες σελίδες για fine-tuning των threshold settings κάθε αισθητήρα. Η διεπαφή χρησιμοποιεί το Limitless theme's advanced form components για τη ρύθμιση minimum/maximum values, alert thresholds, calibration offsets, και notification preferences. Κάθε sensor type (temperature, humidity, GPS, etc.) έχει customized interface elements που αντικατοπτρίζουν τις ιδιαίτερες απαιτήσεις του, με πραγματικού χρόνου preview των threshold ranges και visual indicators για current vs. threshold values.(Εικόνα 12)

The screenshot shows the 'Edit Sensor: latitude' page with the following sections:

- Basic Information:**
 - Sensor Type:** latitude (Sensor type cannot be changed)
 - Device:** Mosquito test (Device assignment cannot be changed)
 - Sensor Name:** Latitude Sensor (Optional friendly name for the sensor)
 - Unit:** ° (Unit of measurement)
 - Description:** Auto-created from MQTT topic: ESP32-DEV-001/sensors
- Current Reading:**
 - Current Value:** 39.53 ° (Last updated: Sep 24, 2025 10:42:17 (2 hours ago))
 - Status:** Sensor Enabled (Disabled sensors will not process new readings)
- Alert Settings:**
 - Enable Alerts:** Enable threshold-based alerts for this sensor
 - Minimum Threshold:** Enter minimum value (Alert when value goes below this threshold)
 - Maximum Threshold:** Enter maximum value (Alert when value goes above this threshold)

Navigation buttons: Back to Sensors (left), Update Sensor (right).

Εικόνα 12: Ρύθμιση ορίων και παραμέτρων αισθητήρα

Η συνολική frontend αρχιτεκτονική εξασφαλίζει seamless εμπειρία χρήστη με 10-δευτερόλεπτο polling για πραγματικού χρόνου updates, automatic sensor discovery από MQTT payloads, και integrated geofencing visualization που συνδέει όλα τα επίπεδα του συστήματος - από lands και devices έως individual sensors.

4.4 Dashboard - IoT Monitoring Overview

Το Dashboard αποτελεί το κεντρικό κέντρο ελέγχου του IoT monitoring συστήματος, παρέχοντας μια ολοκληρωμένη επισκόπηση όλων των συνδεδεμένων συσκευών και των μετρήσεων τους.

4.4.1 Κεντρικές Μετρικές

Στο πάνω μέρος του dashboard εμφανίζονται τέσσερις κεντρικές μετρικές σε έγχρωμα κουτιά που παρέχουν άμεση επισκόπηση της κατάστασης του συστήματος :

- Total Lands: 1 (1 ενεργό, 0 ανενεργό)
- Total Devices: 5 (4 ενεργές, 1 εκτός λειτουργίας)
- Total Sensors: 26 αισθητήρες σε όλες τις συσκευές
- Active Alerts: 9 ενεργές ειδοποιήσεις υψηλής σημασίας

4.4.2 Στατιστικά Συσκευών

Το dashboard περιλαμβάνει επίσης πρόσθετες μετρικές για την κατάσταση των συσκευών :

- Online Devices: 4 συσκευές συνδεδεμένες
- Recent Activity: 0 δραστηριότητες τις τελευταίες 24 ώρες
- GPS Enabled: 4 συσκευές με ενεργοποιημένο GPS

4.4.3 Οπτική Αναπαράσταση Δεδομένων

Το σύστημα διαθέτει προηγμένα εργαλεία οπτικοποίησης που περιλαμβάνουν :

- Κυκλικό γράφημα για την κατανομή τύπων συσκευών (Device Types Distribution)
- Ιστόγραμμα για την κατανομή τύπων αισθητήρων (Sensor Types Distribution)
- Real-time widgets με live data updates κάθε 30 δευτερόλεπτα

4.4.4 Σύστημα Ειδοποιήσεων

Το τμήμα Recent Alerts εμφανίζει τις πιο πρόσφατες ειδοποιήσεις του συστήματος :

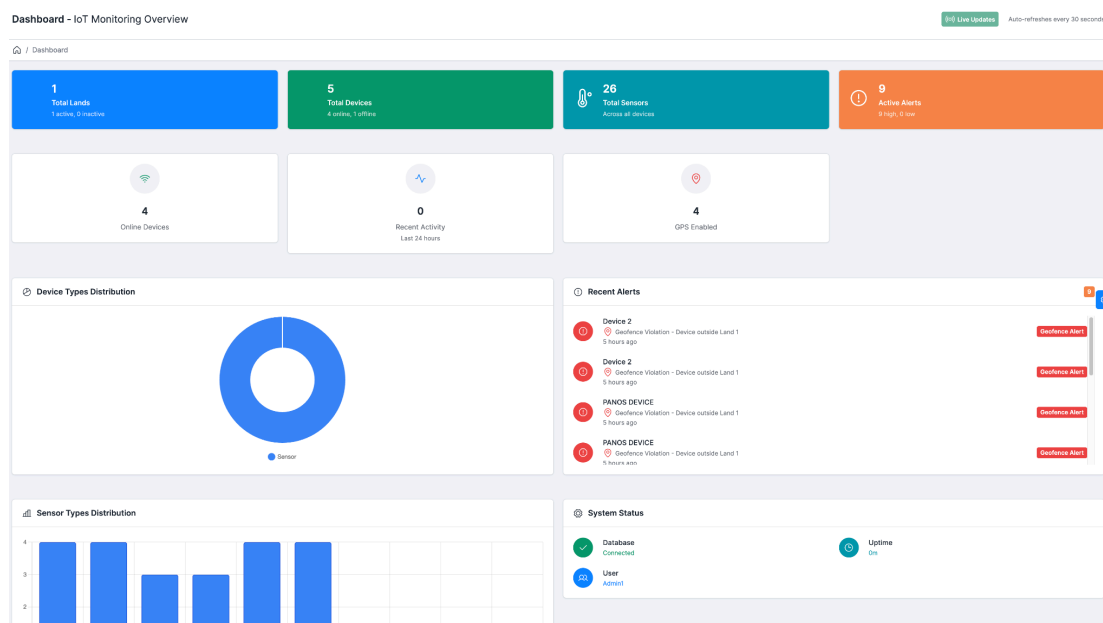
- Geofence violations από συσκευές που βρίσκονται εκτός των ορίων του Land 1
- Color-coded alert indicators για άμεση αναγνώριση της σοβαρότητας
- Χρονικές ενδείξεις για κάθε ειδοποίηση

4.4.5 Κατάσταση Συστήματος

Το System Status section παρέχει επισκόπηση της υγείας του συστήματος :

- Database connectivity status (Connected)
- User session information (Admin)
- System uptime monitoring
- Auto-refresh functionality κάθε 30 δευτερόλεπτα για πραγματικού χρόνου updates

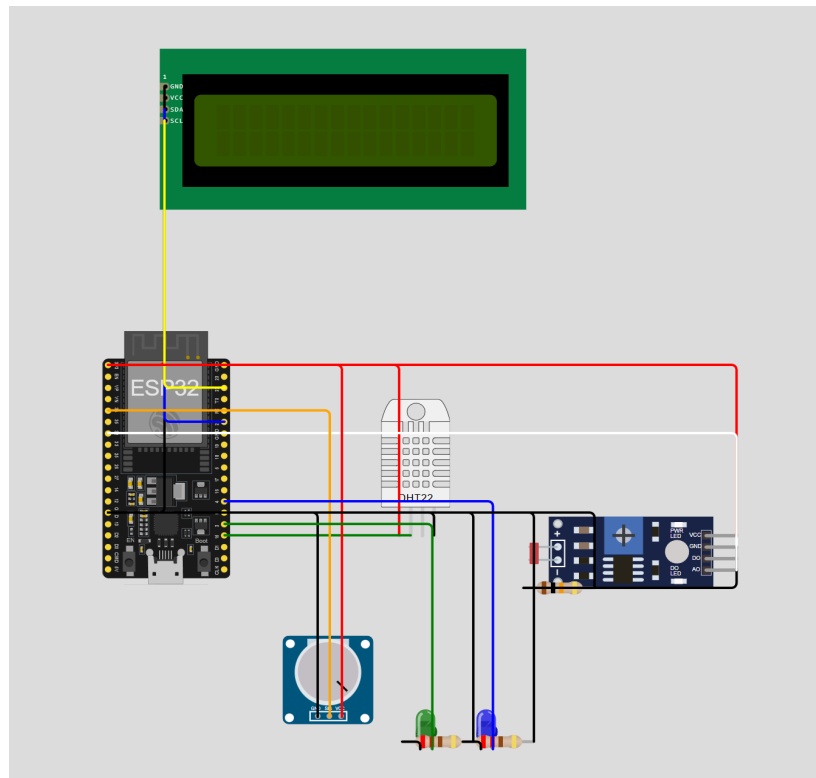
Το dashboard λειτουργεί ως command center για τη διαχείριση και παρακολούθηση όλων των IoT συσκευών, παρέχοντας στους χρήστες τη δυνατότητα να έχουν άμεση πρόσβαση στις κρίσιμες πληροφορίες και να αντιδρούν γρήγορα σε προβλήματα ή ανωμαλίες.(Εικόνα 13)



Εικόνα 13: Κεντρικό dashboard με μετρικές και ειδοποιήσεις

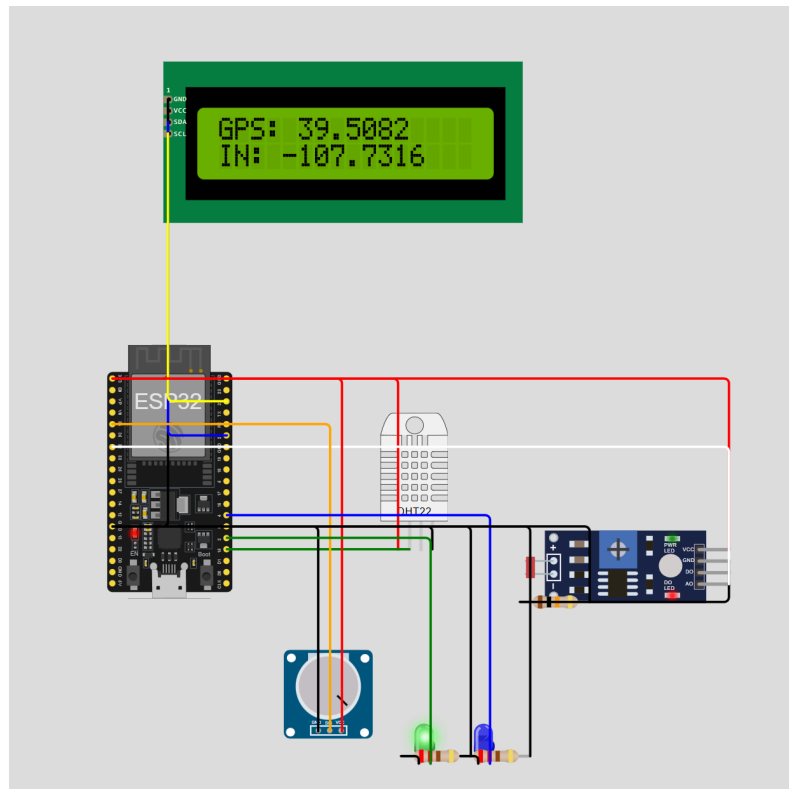
4.5 ESP32 Firmware - Hardware Integration

Για τις δοκιμές του συστήματός μας, χρησιμοποιήσαμε ένα εξελιγμένο περιβάλλον εξομοίωσης με virtual συσκευές βασισμένες σε ESP32, υλοποιημένες μέσω του Wokwi Simulator και συνδεδεμένες σε MQTT broker (EMQX). Η συσκευή μας, με ID "sensor_device_1_5841", προσομοιώνει περιβαλλοντικές μετρήσεις (θερμοκρασία, υγρασία, φως, πρόσθετα δεδομένα) και GPS tracking, υποστηρίζοντας τη βασική λειτουργία γεωφράγματος (geofencing).



Εικόνα 14: Virtual συσκευή Arduino Interface (Ανενεργό)

Το σύστημα αυτόματα εναλλάσσει το geofence mode κάθε 2 λεπτά ανάμεσα σε "INSIDE" και "OUTSIDE", χρησιμοποιώντας πραγματικές συντεταγμένες για Colorado (INSIDE) και τυχαίες κοντά στο San Francisco (OUTSIDE). Μέσω Arduino κώδικα, η συσκευή ενσωματώνει τέσσερις κύριους αισθητήρες, ενώ τα δεδομένα μεταδίδονται σε ενοποιημένο MQTT topic με συμπαγή JSON μορφή κάθε 10 δευτερόλεπτα.



Εικόνα 15: Virtual συσκευή Arduino Interface (Ενεργό)

Η διεπαφή παρακολούθησης περιλαμβάνει LCD οθόνη που εμφανίζει σε πραγματικό χρόνο θερμοκρασία, υγρασία και GPS στοιχεία, με μηνύματα κατάστασης σύνδεσης και καταγραφής. Ο κώδικας περιλαμβάνει επίσης αυτοματοποιημένη διαχείριση σύνδεσης WiFi και MQTT με αυτόματη επανασύνδεση, και robust χειρισμό σφαλμάτων δεδομένων αισθητήρων.

```

=== Sensor Readings ===
Temperature: 24.00°C
Humidity: 40.00%
Light Level: 24%
Potentiometer: 100%
WiFi Signal: -85 dBm
=====
✓ Sensor data sent successfully
Mode: INSIDE Xorafi 1
✓ Status update sent (online)
✓ INSIDE Xorafi rectangle generated
GPS: 39.497931, -107.734517 (INSIDE)
Bounds: Lat[39.495387 to 39.529577] Lng[-107.744122 to -107.653999]
✓ SIMULATED GPS data published (INSIDE) - Lat:39.497931 Lng:-107.734517

```

Εικόνα 16: Live Data sending

Με αυτή την ολοκληρωμένη υλοποίηση, δημιουργήθηκε ένα αποτελεσματικό και αξιόπιστο testing περιβάλλον για την επαλήθευση των λειτουργιών και τη βελτιστοποίηση του IoT συστήματος σε πραγματικές και εξομοιωμένες συνθήκες.

5. ΑΠΟΤΕΛΕΣΜΑΤΑ

Το κεφάλαιο αυτό παρουσιάζει τα αποτελέσματα των δοκιμών του ολοκληρωμένου συστήματος διαχείρισης IoT γεωργικών συσκευών μέσω πραγματικών παραδειγμάτων χρήσης. Οι δοκιμές πραγματοποιήθηκαν με τη χρήση Wokwi simulator για την εξομοίωση ESP32 συσκευών και EMQX broker για την MQTT επικοινωνία, επιδεικνύοντας τη λειτουργικότητα του συστήματος σε ελεγχόμενο περιβάλλον με το Limitless admin theme και Laravel 12 backend.

Real-time Geospatial Monitoring με Leaflet.js

Το σύστημα επιτυγχάνει επιτυχή πραγματικού χρόνου παρακολούθηση της γεωγραφικής θέσης των IoT συσκευών μέσω προχωρημένου Leaflet.js mapping interface. Η χαρτογραφική διεπαφή του Limitless theme παρουσιάζει το Device Location map, το οποίο εμφανίζει την τρέχουσα θέση της συσκευής "Environmental Sensor Monitor with GPS" στις συντεταγμένες 48.713148, -74.006748 με automatic refresh κάθε 10 δευτερόλεπτα μέσω Ajax polling.

Ο Leaflet map παρέχει λεπτομερείς πληροφορίες μέσω custom popup overlays, περιλαμβάνοντας altitude (15m), speed (87.3 km/h), satellites count (10), GPS validity status, και κρίσιμα την ένδειξη "Device Outside Land Boundary". Αυτή η λειτουργία επιβεβαιώνει τη σωστή λειτουργία του client-side geofencing system που χρησιμοποιεί Leaflet geometric functions σε συνδυασμό με το backend GeofencingService για ray-casting calculations.

Comprehensive Device Management Interface

Η διεπαφή διαχείρισης συσκευών του Limitless theme παρουσιάζει ολοκληρωμένες πληροφορίες μέσω responsive dashboard cards. Το Device Information panel εμφανίζει βασικά στοιχεία όπως Device Name ("Environmental Sensor Monitor with

GPS"), Device ID (ESP32-DEV-001), Device Type (ESP32_ENVIRONMENTAL_GPS), Land Location (Xorafi 1), και πραγματικού χρόνου Status indicators (online).

Το Location Information section παρέχει λεπτομερή γεωγραφικά στοιχεία μέσω ενσωματωμένων mini Leaflet maps, ενώ το GPS Tracking Enabled status δείχνει την κατάσταση της γεωχωρικής παρακολούθησης. Αυτή η διεπαφή χρησιμοποιεί το JavaScript polling system του Limitless theme για πραγματικού χρόνου updates χωρίς page refresh.

MQTT Communication Architecture

Το MQTT Topics section επιδεικνύει την εκτεταμένη communication architecture που υποστηρίζεται από το background Laravel console command που τρέχει συνεχώς στον server. Παρουσιάζεται η πλήρης δομή των topics που χρησιμοποιούνται για την επικοινωνία με τη συσκευή ESP32-DEV-001, συμπεριλαμβανομένων των Discovery Request/Response topics, Data Topic, GPS Topic, Status Topic, Commands Topic, Control Response Topic, και Global Discovery Topic.

Αυτή η δομημένη προσέγγιση εξασφαλίζει αξιόπιστη επικοινωνία μεταξύ των συσκευών και του Laravel 12 backend, υποστηρίζοντας διαφορετικούς τύπους μηνυμάτων που επεξεργάζονται από το MqttDeviceService και προωθούνται στο frontend μέσω Ajax API responses.

Advanced Sensor Monitoring Dashboard

Το Connected Sensors dashboard αποτελεί το κεντρικό σημείο παρακολούθησης όλων των αισθητήρων που δημιουργούνται αυτόματα όταν η συσκευή στέλνει MQTT payload. Το σύστημα εμφανίζει επιτυχώς δεδομένα από 12 διαφορετικούς αισθητήρες μέσω Limitless theme's card components, περιλαμβάνοντας περιβαλλοντικούς

(DHT22 Temperature: 24.0°C, DHT22 Humidity: 40%, Light: 24%), συστήματος (WiFi signal: -86 dBm, Battery: 95%), και GPS αισθητήρες (Latitude: 39.513799, Longitude: -107.703190, Altitude: 2509m, Speed: 2.4 km/h).

Το σύστημα εφαρμόζει επιτυχώς τη JavaScript-based geofencing λογική, εμφανίζοντας χρωματικές ενδείξεις: "Status: Inside" με πράσινο χρώμα για τους GPS αισθητήρες latitude και longitude που βρίσκονται εντός ορίων, ενώ δείχνει "Status: Outside" με κόκκινο χρώμα και "Outside land boundary" warning για GPS Altitude, Speed, Satellites, και GPS Signal Valid όταν οι τιμές τους υπερβαίνουν τα threshold settings που ρυθμίζονται μέσω του Sensor management interface.

Multi-Protocol Communication Testing

Οι δοκιμές επεκτάθηκαν στη διασύνδεση με The Things Stack LoRaWAN (πρωτόκολλο) network server μέσω webhook integration, επιβεβαιώνοντας τη dual-protocol functionality του συστήματος. Το backend διαχειρίζεται επιτυχώς εισερχόμενα δεδομένα τόσο από MQTT brokers (EMQX/Mosquitto) όσο και από LoRaWAN (πρωτόκολλο) webhooks, δημιουργώντας αυτόματα sensor entries και παρέχοντας unified interface στο frontend.

System Performance και Reliability

Οι δοκιμές επιβεβαιώνουν την επιτυχή λειτουργία όλων των κρίσιμων χαρακτηριστικών του συστήματος. Το πραγματικού χρόνου data streaming λειτουργεί αποδοτικά μέσω του background MQTT listener command με ενημερώσεις κάθε 10 δευτερόλεπτα, το Leaflet.js geofencing system εντοπίζει σωστά τις παραβιάσεις γεωγραφικών ορίων μέσω client-side calculations, και η MQTT επικοινωνία διατηρεί σταθερή σύνδεση με τον EMQX broker.

Η Limitless theme interface ανταποκρίνεται άμεσα στις αλλαγές μέσω Ajax-based updates και παρέχει ολοκληρωμένη οπτική ανατροφοδότηση με Bootstrap 5 components για την κατάσταση κάθε συστήματος. Το σύστημα επιδεικνύει επίσης ευρωστία στη διαχείριση μικτών δεδομένων, όπου ορισμένοι αισθητήρες εμφανίζουν κανονική λειτουργία ενώ άλλοι παρουσιάζουν παραβιάσεις ορίων, παρέχοντας στους χρήστες σαφή χρωματική διαφοροποίηση και interactive tooltips για τις περιοχές που χρειάζονται προσοχή.

Η JavaScript integration με το Laravel 12 backend εξασφαλίζει seamless data synchronization μέσω 10-δευτερόλεπτου Ajax polling, ενώ οι Leaflet map instances λαμβάνουν πραγματικού χρόνου updates για device positions και geofencing status, δημιουργώντας μια fluid και responsive εμπειρία χρήστη για επαγγελματική διαχείριση γεωργικού IoT.

Η παρούσα πτυχιακή εργασία παρουσιάζει την επιτυχή ανάπτυξη ενός ολοκληρωμένου συστήματος διαχείρισης IoT γεωργικών συσκευών που συνδυάζει σύγχρονες τεχνολογίες web development με προχωρημένες IoT λύσεις. Το σύστημα επιτυγχάνει τους βασικούς στόχους που τέθηκαν αρχικά, παρέχοντας μια πλήρη πλατφόρμα ακριβής γεωργίας με πραγματικού χρόνου monitoring, γεωχωρικές δυνατότητες, και υποστήριξη πολλαπλών πρωτοκόλλων επικοινωνίας.

5.1 Τεχνολογική Επιτυχία

Η επιλογή του Laravel 12 ως backend framework αποδείχθηκε στρατηγικά σωστή, καθώς οι νέες δυνατότητες του (βελτιωμένη δομή φακέλων, PHP 8.3+ support, προηγμένο σύστημα ουρών) συνέβαλαν στην ανάπτυξη ενός αποδοτικού και κλιμακώσιμου συστήματος. Η επιλογή του Limitless admin theme με vanilla JavaScript integration παρείχε professional UI components με minimal development complexity, ενώ η Laravel Sanctum authentication εξασφάλισε robust security χωρίς την πολυπλοκότητα του JWT.

5.2 Background Processing Architecture

Το background Laravel Console Command system που εκτελείται συνεχώς μέσω supervisor αποδείχθηκε critical για τη διατήρηση αδιάλειπτης MQTT επικοινωνίας. Η asynchronous επεξεργασία μέσω Laravel Queues εξασφαλίζει ότι το web interface παραμένει responsive ενώ χειρίζεται μεγάλους όγκους IoT data από πολλαπλές πηγές.

5.3 Geofencing και Real-time Monitoring

Η υλοποίηση του GeofencingService με τον ray-casting αλγόριθμο επιτυγχάνει ακριβή γεωχωρική παρακολούθηση, ενώ το 10-δευτερόλεπτο Ajax polling παρέχει consistent ενημέρωση για την κατάσταση των συσκευών χωρίς την πολυπλοκότητα WebSocket connections. Οι δοκιμές με το Wokwi simulator επιβεβαιώνουν τη σωστή λειτουργία του geofencing system, εμφανίζοντας σαφείς ενδείξεις για συσκευές "Inside" και "Outside" τα γεωγραφικά όρια.

5.4 Multi-Protocol Communication Architecture

Η εκτεταμένη MQTT topic structure υποστηρίζει αποδοτικά διαφορετικούς τύπους επικοινωνίας (ανακάλυψη, δεδομένα, GPS, εντολές, κατάσταση), ενώ η υποστήριξη δύο δημόσιων MQTT brokers (EMQX/Mosquitto) και LoRaWAN (πρωτόκολλο) μέσω The Things Stack webhooks παρέχει ευελιξία σε διαφορετικά deployment scenarios. Η αυτόματη sensor discovery από το πρώτο MQTT payload απλοποιεί σημαντικά τη διαδικασία εγκατάστασης νέων συσκευών.

5.5 Hierarchical Data Architecture Success

Η User → Land → Device → Sensor ιεραρχία με Leaflet.js polygon definition αποδείχθηκε intuitive και scalable, επιτρέποντας στους χρήστες να οργανώσουν τις IoT εγκαταστάσεις τους με γεωγραφικό context. Το automatic sensor creation system εξαλείφει την ανάγκη για manual configuration, ενώ παράλληλα διατηρεί τη δυνατότητα fine-tuning μέσω των threshold settings.

6. ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

Η ασφάλεια αποτελεί κρίσιμο παράγοντα για την προστασία και τη σταθερότητα του IoT συστήματος, ειδικά όταν πολλές συσκευές και χρήστες αλληλεπιδρούν με ευαίσθητα δεδομένα. Για τον λόγο αυτό, το σύστημα θα ενσωματώσει ένα ολοκληρωμένο πλαίσιο διαχείρισης credentials που θα εξασφαλίζει την έγκυρη πιστοποίηση και τον έλεγχο πρόσβασης σε όλα τα επίπεδα.

Κάθε IoT συσκευή θα έχει μοναδικά credentials για να συνδέεται στους MQTT brokers, με υποστήριξη πολλαπλών μεθόδων authentication όπως username/password, client certificates, OAuth tokens, και API keys. Οι συνδέσεις θα κρυπτογραφούνται με πρωτόκολλα SSL/TLS, διασφαλίζοντας ότι τα δεδομένα που μεταφέρονται είναι προστατευμένα από υποκλοπή ή παραποίηση.

Το σύστημα θα εφαρμόζει έλεγχο πρόσβασης βασισμένο σε ρόλους, το οποίο θα ορίζει ποιοι χρήστες ή συσκευές μπορούν να έχουν πρόσβαση σε συγκεκριμένους brokers και δεδομένα, περιορίζοντας έτσι τις δυνατότητες μη εξουσιοδοτημένων ενεργειών. Η αποθήκευση των credentials θα γίνεται με ασφαλή κρυπτογράφηση στη βάση δεδομένων, ενώ θα υπάρχει αυτοματοποιημένος μηχανισμός για την περιοδική ανανέωση και απόσυρση των credentials μέσω scheduled tasks, διατηρώντας πάντα τη μέγιστη δυνατή ασφάλεια.

Παράλληλα, το σύστημα θα υποστηρίζει πλήρη καταγραφή ελέγχου όλων των συνδέσεων, προσπαθειών πιστοποίησης και συμβάντων ασφαλείας, επιτρέποντας την παρακολούθηση και τον εντοπισμό τυχόν απειλών ή ανωμαλιών. Η υλοποίηση επίσης θα περιλαμβάνει υποδομές δικτυακής ασφάλειας, όπως VPN σύνδεση, firewall κανόνες και IP whitelisting για τον ασφαλή χειρισμό των broker communications.

Με αυτό το ολοκληρωμένο πλαίσιο ασφάλειας, το IoT σύστημα θα διαφυλάσσει την ακεραιότητα και εμπιστευτικότητα των δεδομένων και των λειτουργιών του, παρέχοντας αξιόπιστη προστασία έναντι πιθανών κυβερνοαπειλών και διασφαλίζοντας την ομαλή και ασφαλή λειτουργία όλων των συσκευών και υπηρεσιών.

ΒΙΒΛΙΟΓΡΑΦΙΑ

Εδώ είναι οι βασικές βιβλιογραφικές αναφορές που υποστηρίζουν το κύριο κείμενο, με αντίστοιχα links όπου είναι διαθέσιμα:

[Ahmed, F., et al. \(2022\). Smart Agriculture IoT System using MQTT and LoRaWAN \(πρωτόκολλο\). IEEE Access. https://doi.org/10.1109/ACCESS.2022.3140047](https://doi.org/10.1109/ACCESS.2022.3140047)

[Farooq, M., et al. \(2020\). IoT-enabled Smart Farming Surveillance System with Federated AI. PLoS ONE. https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0229247](https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0229247)

[Codeluppi, T., et al. \(2020\). Agricultural IoT System Architecture with Real-time Monitoring. Computers and Electronics in Agriculture. https://doi.org/10.1016/j.compag.2020.105134](https://doi.org/10.1016/j.compag.2020.105134)

[Yavuz, M., et al. \(2022\). Geofencing Techniques for IoT Applications. Sensors. https://doi.org/10.3390/s22031128](https://doi.org/10.3390/s22031128)

[Gnanadurai, C., et al. \(2024\). Machine Learning for Precision Agriculture. Agricultural Systems. https://doi.org/10.1016/j.agry.2023.103956](https://doi.org/10.1016/j.agry.2023.103956)

[HiveMQ. \(2022\). MQTT Security Best Practices. https://www.hivemq.com/blog/mqtt-security-best-practices/](https://www.hivemq.com/blog/mqtt-security-best-practices/)

[SAP \(2024\). IoT Security with Role-based Access Control. https://help.sap.com/docs/iot_security](https://help.sap.com/docs/iot_security)

-

ΠΑΡΑΡΤΗΜΑ

Το σύστημα είναι διαθέσιμο ως έργο ανοιχτού κώδικα στη διεύθυνση <https://github.com/ElVasilakakis/Ptyxiaki-AdminLTE>. Η πλατφόρμα παρέχει πλήρη πηγαίο κώδικα, οδηγίες εγκατάστασης και εργαλεία για την εύκολη εκτέλεση και ανάπτυξη του backend, frontend και των υπηρεσιών MQTT, δίνοντας τη δυνατότητα επέκτασης και παραμετροποίησης από προγραμματιστές.