A picture containing text, sign

Description automatically generated

**Πανεπιστήμιο Θεσσαλίας**

**Σχολή Τεχνολογίας**

**Τμήμα Ψηφιακών Συστημάτων**

**Έξυπνο Θερμοκήπιο**

**Πτυχιακή Εργασία**

**Δερμιτζάκης Άγγελος Μιχαήλ (ΑΜ: 3119264)**

**Καλογιάννης Δημήτριος (ΑΜ: 3119019)**

**Επιβλέπων: *Νικόλαος Σαμαράς***

**ΛΑΡΙΣΑ, Δεκέμβριος 2024**

A picture containing text, sign

Description automatically generated

**University of Thessaly**

**School of Technology**

**Digital Systems Department**

**Smart Greenhouse**

**Bachelor Thesis**

**Dermitzakis Aggelos Michail (RN:** **3119264)**

**Kalogiannis Dimitrios (RN: 3119019)**

**Supervisor: *Nikolaos Samaras***

**LARISSA, December 2024**

**Υπεύθυνη Δήλωση περί Ακαδημαϊκής Δεοντολογίας   
και Πνευματικών Δικαιωμάτων**

Με πλήρη επίγνωση των συνεπειών του νόμου περί πνευματικών δικαιωμάτων, δηλώνω ρητά ότι η παρούσα πτυχιακή εργασία, καθώς και τα ηλεκτρονικά αρχεία και πηγαίοι κώδικες που αναπτύχθηκαν ή τροποποιήθηκαν στα πλαίσια αυτής της εργασίας, αποτελεί αποκλειστικά προϊόν προσωπικής μου εργασίας, δεν προσβάλλει κάθε μορφής δικαιώματα διανοητικής ιδιοκτησίας, προσωπικότητας και προσωπικών δεδομένων τρίτων, δεν περιέχει έργα/εισφορές τρίτων για τα οποία απαιτείται άδεια των δημιουργών/δικαιούχων και δεν είναι προϊόν μερικής ή ολικής αντιγραφής, οι πηγές δε που χρησιμοποιήθηκαν περιορίζονται στις βιβλιογραφικές αναφορές και μόνον, και πληρούν τους κανόνες της επιστημονικής παράθεσης. Τα σημεία όπου έχω χρησιμοποιήσει ιδέες, κείμενο, αρχεία ή/και πηγές άλλων συγγραφέων, αναφέρονται ευδιάκριτα στο κείμενο με την κατάλληλη παραπομπή και η σχετική αναφορά περιλαμβάνεται στο τμήμα των βιβλιογραφικών αναφορών με πλήρη περιγραφή.

Αναλαμβάνω πλήρως, ατομικά και προσωπικά, όλες τις νομικές και διοικητικές συνέπειες που δύναται να προκύψουν στην περίπτωση κατά την οποία αποδειχθεί, διαχρονικά, ότι η εργασία αυτή ή τμήμα της δεν μου ανήκει διότι είναι προϊόν λογοκλοπής.

Οι Φοιτητές

Δερμιτζάκης Άγγελος Μιχαήλ Καλογιάννης Δημήτριος

23/12/2024

Εγκρίνεται από την Επιτροπή Εξέτασης:

Επιβλέπων/πουσα Νικόλαος Σαμαράς

Καθηγητής /Δίκτυα Πληροφοριακών Συστημάτων,

Τμήμα Ψηφιακών Συστημάτων, Πανεπιστήμιο Θεσσαλίας

Μέλος **Ονοματεπώνυμο Μέλους 1**

Βαθμίδα/ιδιότητα μέλους 1, Τμήμα/Ίδρυμα μέλους 1

Μέλος **Ονοματεπώνυμο Μέλους 2**

Βαθμίδα/ιδιότητα μέλους 2, Τμήμα/Ίδρυμα μέλους 2

Ημερομηνία έγκρισης: dd-mm-yyyy

Περίληψη

Η παρούσα πτυχιακή εργασία παρουσιάζει την ανάπτυξη ενός έξυπνου θερμοκηπίου μικρής κλίμακας, το οποίο αξιοποιεί σύγχρονες τεχνολογίες για την αυτοματοποιημένη παρακολούθηση και διαχείριση κρίσιμων περιβαλλοντικών παραμέτρων. Το σύστημα στοχεύει στη βελτιστοποίηση της ανάπτυξης των φυτών και τη δημιουργία ενός αποτελεσματικού και βιώσιμου συστήματος καλλιέργειας. Το έξυπνο θερμοκήπιο ενσωματώνει διάφορους αισθητήρες και συσκευές ελέγχου, συμπεριλαμβανομένων αισθητήρων θερμοκρασίας, υγρασίας αέρα και εδάφους, καθώς και συστήματα ελέγχου για φωτισμό, άρδευση και αερισμό. Η καρδιά του συστήματος είναι ένα Raspberry Pi, το οποίο συλλέγει δεδομένα από τους αισθητήρες και ελέγχει τις συσκευές μέσω των GPIO pins. Η εργασία αξιοποιεί το πρωτόκολλο MQTT για την αποτελεσματική επικοινωνία μεταξύ των συσκευών, επιτρέποντας την απομακρυσμένη παρακολούθηση και έλεγχο του θερμοκηπίου. Επιπλέον, ενσωματώνει τεχνολογίες επεξεργασίας εικόνας και τεχνητής νοημοσύνης για την αναγνώριση και αξιολόγηση της υγείας των φυτών μέσω φωτογραφιών. Η διεπαφή χρήστη, αναπτυγμένη με τη βιβλιοθήκη NiceGUI, παρέχει μια φιλική προς τον χρήστη πλατφόρμα για την παρακολούθηση των συνθηκών του θερμοκηπίου και τον έλεγχο των συσκευών. Οι δοκιμές του συστήματος έδειξαν υψηλή ακρίβεια στις μετρήσεις των αισθητήρων και αποτελεσματική λειτουργία των συσκευών ελέγχου. Συμπερασματικά, το έξυπνο θερμοκήπιο που αναπτύχθηκε προσφέρει μια ολοκληρωμένη λύση για την αυτοματοποιημένη διαχείριση καλλιεργειών, συμβάλλοντας στην αποδοτικότερη χρήση πόρων και τη βελτίωση της παραγωγικότητας στη γεωργία. Μελλοντικές βελτιώσεις θα μπορούσαν να περιλαμβάνουν την ενσωμάτωση περισσότερων αισθητήρων, την αναβάθμιση της διεπαφής χρήστη και την περαιτέρω αυτοματοποίηση των διαδικασιών ελέγχου.

Abstract

This bachelor thesis presents the development of a small-scale smart greenhouse that utilizes modern technologies for automated monitoring and management of critical environmental parameters. The system aims to optimize plant growth and create an efficient and sustainable cultivation system. The smart greenhouse incorporates various sensors and control devices, including temperature, air and soil humidity sensors, as well as control systems for lighting, irrigation, and ventilation. The heart of the system is a Raspberry Pi, which collects data from sensors and controls devices through GPIO pins. The project leverages the MQTT protocol for efficient communication between devices, allowing remote monitoring and control of the greenhouse. Additionally, it incorporates image processing and artificial intelligence technologies for plant health recognition and evaluation through photographs. The user interface, developed with the NiceGUI library, provides a user-friendly platform for monitoring greenhouse conditions and controlling devices. System tests showed high accuracy in sensor measurements and effective operation of control devices. In conclusion, the developed smart greenhouse offers a comprehensive solution for automated crop management, contributing to more efficient resource use and improved productivity in agriculture. Future improvements could include the integration of more sensors, upgrading the user interface, and further automation of control processes.

Ευχαριστίες

Ευχαριστούμε θερμά τον καθηγητή μας Κ. Νικόλαο Σαμαρά για την υποστήριξη της πτυχιακής μας εργασίας καθώς και το τμήμα ψηφιακών συστημάτων που πανεπιστήμιου Θεσσαλίας για τις γνώσεις και τις εμπειρίες που μας παρείχε.

Δερμιτζάκης Άγγελος Μιχαήλ, Καλογιάννης Δημήτριος

20/12/2024

Περιεχόμενα

[Περίληψη vii](#_Toc184585160)

[Abstract ix](#_Toc184585161)

[Ευχαριστίες xi](#_Toc184585162)

[Περιεχόμενα xiii](#_Toc184585163)

[1 Εισαγωγή 1](#_Toc184585164)

[1.1 Σκοπός της Πτυχιακής Εργασίας 1](#_Toc184585165)

[1.2 Ανάγκη για Έξυπνα Θερμοκήπια 1](#_Toc184585166)

[1.3 Επισκόπηση της Τεχνολογίας 2](#_Toc184585167)

[2 Ανασκόπηση Βιβλιογραφίας 5](#_Toc184585168)

[2.1 Έξυπνα Θερμοκήπια: Τεχνολογίες και Λύσεις 5](#_Toc184585169)

[2.1.1 Βασικές Τεχνολογίες και Λειτουργίες 5](#_Toc184585170)

[2.1.2 Οφέλη και Πλεονεκτήματα 6](#_Toc184585171)

[2.1.3 Μελλοντικές Τάσεις 7](#_Toc184585172)

[2.2 Επεξεργασία Εικόνας στην Ανάλυση Φυτών 7](#_Toc184585173)

[2.2.1 Βασικές Εφαρμογές 7](#_Toc184585174)

[2.2.2 Τεχνολογίες και Μέθοδοι 8](#_Toc184585175)

[2.2.3 Πλεονεκτήματα και Εφαρμογές 9](#_Toc184585176)

[2.3 Επικοινωνία MQTT: Εφαρμογές σε IoT 9](#_Toc184585177)

[2.3.1 Βασικά Χαρακτηριστικά του MQTT 10](#_Toc184585178)

[2.3.2 Εφαρμογές σε Έξυπνα Θερμοκήπια 11](#_Toc184585179)

[2.3.3 Πλεονεκτήματα του MQTT στα Έξυπνα Θερμοκήπια 13](#_Toc184585180)

[2.4 Εισαγωγή στη Βιβλιοθήκη NiceGUI 13](#_Toc184585181)

[2.4.1 Βασικά Χαρακτηριστικά και Εφαρμογές 13](#_Toc184585182)

[2.4.2 Πλεονεκτήματα για την μελετώμενη περίπτωση 14](#_Toc184585183)

[3 Περιγραφή Συστήματος 15](#_Toc184585184)

[3.1 Αρχιτεκτονική Συστήματος 15](#_Toc184585185)

[3.2 Λειτουργικές Απαιτήσεις 18](#_Toc184585186)

[3.3 Χρήση του Raspberry Pi 18](#_Toc184585187)

[4 Υλικό και Αισθητήρες 21](#_Toc184585188)

[4.1 Αισθητήρας Θερμοκρασίας και Υγρασίας (DHT11) 21](#_Toc184585189)

[4.2 Αισθητήρας Υγρασίας Εδάφους 22](#_Toc184585190)

[4.3 Σερβοκινητήρας για Άνοιγμα Παράθυρων 23](#_Toc184585191)

[4.4 Αντλία Ποτίσματος 24](#_Toc184585192)

[4.5 Ανεμιστήρας για Αερισμό 25](#_Toc184585193)

[4.6 Κάμερα για Επιτήρηση Φυτών 25](#_Toc184585194)

[4.7 Συνδεσμολογία και Διάταξη Στοιχείων 26](#_Toc184585195)

[5 Λογισμικό και Ανάπτυξη Εφαρμογής 29](#_Toc184585196)

[5.1 Python : Επιλογή Πλατφόρμας 29](#_Toc184585197)

[5.2 Ανάπτυξη Διεπαφής Χρήστη (GUI) 30](#_Toc184585198)

[5.3 Επικοινωνία MQTT για Παρακολούθηση και Έλεγχο 31](#_Toc184585199)

[5.4 API Ανάλυσης Υγείας Φυτών 32](#_Toc184585200)

[6 Δοκιμές και Αξιολόγηση 33](#_Toc184585201)

[6.1 Δοκιμές Συσκευών 33](#_Toc184585202)

[6.2 Επικοινωνία MQTT 34](#_Toc184585203)

[6.3 Αναγνώριση Υγείας Φυτών 35](#_Toc184585204)

[7 Συμπεράσματα και Μελλοντικές Βελτιώσεις 37](#_Toc184585205)

[Βιβλιογραφία 39](#_Toc184585206)

[Παράρτημα: Κώδικας Προγράμματος 41](#_Toc184585207)

# Εισαγωγή

## Σκοπός της Πτυχιακής Εργασίας

Η παρούσα πτυχιακή εργασία έχει ως κύριο στόχο την ανάπτυξη ενός έξυπνου θερμοκηπίου σε μικρή κλίμακα, το οποίο αξιοποιεί τις δυνατότητες της τεχνολογίας για την αυτοματοποιημένη παρακολούθηση και διαχείριση κρίσιμων περιβαλλοντικών παραμέτρων. Μέσω αυτής της λύσης, επιδιώκεται η βελτιστοποίηση της ανάπτυξης των φυτών και η δημιουργία ενός αποτελεσματικού και βιώσιμου συστήματος καλλιέργειας. Το έξυπνο θερμοκήπιο που αναπτύσσεται έχει τη δυνατότητα να προσφέρει:

* **Αυτόματη μέτρηση και παρακολούθηση κρίσιμων παραμέτρων** όπως η θερμοκρασία, η υγρασία του αέρα, η υγρασία του εδάφους και η ένταση του φωτισμού εξωτερικά του θερμοκηπίου. Αυτά τα δεδομένα συλλέγονται και αναλύονται σε πραγματικό χρόνο, παρέχοντας άμεση εικόνα των συνθηκών που επικρατούν στο θερμοκήπιο.
* **Δυνατότητα ελέγχου διάφορων συσκευών** όπως αντλίες ποτίσματος, ανεμιστήρες, συστήματα φωτισμού και μηχανισμούς ανοίγματος παραθύρων. Ο έλεγχος γίνεται αυτόματα, με βάση προκαθορισμένα όρια που διασφαλίζουν τη βέλτιστη λειτουργία του θερμοκηπίου.
* **Αναγνώριση φυτών και αξιολόγηση της υγείας τους** μέσω τεχνολογιών επεξεργασίας εικόνας και εξειδικευμένων APIs. Αυτό το χαρακτηριστικό επιτρέπει την ανίχνευση προβλημάτων, όπως ασθένειες ή έλλειψη θρεπτικών συστατικών, και την άμεση λήψη διορθωτικών μέτρων.
* **Απομακρυσμένη παρακολούθηση** και διαχείριση μέσω μιας φιλικής προς τον χρήστη διεπαφής. Οι χρήστες μπορούν να έχουν πρόσβαση στο σύστημα μέσω διαδικτύου, να ελέγχουν τις παραμέτρους του θερμοκηπίου και να λαμβάνουν ειδοποιήσεις για σημαντικά γεγονότα ή προβλήματα.

## Ανάγκη για Έξυπνα Θερμοκήπια

Η γεωργία αντιμετωπίζει σημαντικές προκλήσεις λόγω της κλιματικής αλλαγής, της αύξησης του παγκόσμιου πληθυσμού και της ανάγκης για βιώσιμες πρακτικές παραγωγής. Σε αυτό το πλαίσιο, τα έξυπνα θερμοκήπια έχουν αναδειχθεί ως μια ολοκληρωμένη λύση που συνδυάζει την τεχνολογία με τις γεωργικές ανάγκες. Η χρησιμότητά τους εκτείνεται σε πολλαπλούς τομείς:

* **Βελτιστοποίηση της παραγωγής**: Τα έξυπνα θερμοκήπια επιτρέπουν την αυτοματοποίηση κρίσιμων διαδικασιών, μειώνοντας τον χρόνο και την προσπάθεια που απαιτείται για τη διαχείριση της καλλιέργειας.
* **Μείωση της κατανάλωσης πόρων**: Με τη χρήση προηγμένων αισθητήρων και μηχανισμών ελέγχου, μειώνεται η σπατάλη νερού, ενέργειας και άλλων πόρων.
* **Έγκαιρη διάγνωση προβλημάτων**: Η δυνατότητα ανίχνευσης ασθενειών, ανεπάρκειας θρεπτικών συστατικών ή περιβαλλοντικών ανωμαλιών βοηθά στη μείωση των απωλειών και στη διασφάλιση της ποιότητας της παραγωγής.
* **Διαχείριση σε πραγματικό χρόνο**: Με τη χρήση IoT και τεχνολογιών τεχνητής νοημοσύνης, οι καλλιεργητές μπορούν να έχουν πρόσβαση σε δεδομένα και να ελέγχουν το θερμοκήπιο εξ αποστάσεως, ανεξαρτήτως τοποθεσίας.

## Επισκόπηση της Τεχνολογίας

Για την υλοποίηση του έξυπνου θερμοκηπίου, η παρούσα εργασία αξιοποιεί μια σειρά από τεχνολογίες:

* **Raspberry Pi**: Αποτελεί την κύρια υπολογιστική μονάδα του συστήματος, διαχειρίζοντας τη συλλογή και ανάλυση δεδομένων από τους αισθητήρες καθώς και τον έλεγχο των συσκευών.
* **Αισθητήρες**: Χρησιμοποιούνται για την καταγραφή βασικών παραμέτρων όπως η θερμοκρασία, η υγρασία του αέρα και του εδάφους, και η φωτεινότητα. Τα δεδομένα αυτά είναι απαραίτητα για τη λήψη αποφάσεων σχετικά με τη διαχείριση του θερμοκηπίου.
* **MQTT**: Ένα αποτελεσματικό πρωτόκολλο επικοινωνίας που διευκολύνει την ανταλλαγή δεδομένων μεταξύ των συσκευών του συστήματος.
* **NiceGUI**: Μια βιβλιοθήκη της Python που χρησιμοποιείται για τη δημιουργία μιας εύχρηστης διεπαφής. Αυτή η διεπαφή επιτρέπει στους χρήστες να αλληλεπιδρούν εύκολα με το σύστημα, τόσο τοπικά όσο και μέσω διαδικτύου.
* **API αναγνώρισης φυτών**: Το σύστημα ενσωματώνει εξειδικευμένα APIs που επιτρέπουν την αναγνώριση φυτών και την αξιολόγηση της κατάστασης υγείας τους μέσω εικόνων, παρέχοντας χρήσιμες πληροφορίες για την κατάσταση της καλλιέργειας.

Με τον συνδυασμό αυτών των τεχνολογιών, το προτεινόμενο σύστημα θερμοκηπίου φιλοδοξεί να προσφέρει μια ολοκληρωμένη και εύχρηστη λύση.

# Ανασκόπηση Βιβλιογραφίας

Η ανασκόπηση βιβλιογραφίας έχει ως σκοπό να παρουσιάσει και να εξετάσει τις υπάρχουσες τεχνολογίες, τις λύσεις και τις έρευνες που σχετίζονται με τα έξυπνα θερμοκήπια, τη χρήση της επεξεργασίας εικόνας για την ανάλυση φυτών, τη σημασία της επικοινωνίας μέσω MQTT για τα συστήματα IoT, καθώς και την βιβλιοθήκη NiceGUI για την ανάπτυξη φιλικών προς τον χρήστη διεπαφών.

## Έξυπνα Θερμοκήπια: Τεχνολογίες και Λύσεις

Τα έξυπνα θερμοκήπια αποτελούν μια επαναστατική εξέλιξη στον τομέα της γεωργίας, συνδυάζοντας προηγμένες τεχνολογίες με παραδοσιακές μεθόδους καλλιέργειας για τη βελτιστοποίηση των συνθηκών ανάπτυξης των φυτών και την αύξηση της παραγωγικότητας. Αυτά τα σύγχρονα συστήματα αξιοποιούν τεχνολογίες αιχμής όπως το Διαδίκτυο των Πραγμάτων (IoT), την Τεχνητή Νοημοσύνη (AI) και την ανάλυση δεδομένων για τη δημιουργία ενός ελεγχόμενου περιβάλλοντος που ευνοεί την ανάπτυξη των φυτών. [1]

### Βασικές Τεχνολογίες και Λειτουργίες

**Συστήματα IoT και Αισθητήρες**: Τα έξυπνα θερμοκήπια χρησιμοποιούν ένα εκτεταμένο δίκτυο αισθητήρων για τη συνεχή παρακολούθηση κρίσιμων περιβαλλοντικών παραμέτρων. Αυτοί οι αισθητήρες μετρούν με ακρίβεια τη θερμοκρασία, την υγρασία, τα επίπεδα φωτός, την υγρασία του εδάφους, τα επίπεδα διοξείδιού του άνθρακα(CO2) και άλλους παράγοντες που επηρεάζουν την ανάπτυξη των φυτών. Η συλλογή δεδομένων σε πραγματικό χρόνο επιτρέπει την άμεση προσαρμογή των συνθηκών για τη βελτιστοποίηση της ανάπτυξης των καλλιεργειών. [1]

**Τεχνητή Νοημοσύνη και Μηχανική Μάθηση**: Προηγμένοι αλγόριθμοι AI αναλύουν τα δεδομένα που συλλέγονται από τους αισθητήρες για την πρόβλεψη της υγείας των καλλιεργειών και των μοτίβων ανάπτυξης. Τα μοντέλα μηχανικής μάθησης μπορούν να προβλέψουν καιρικές συνθήκες, εστίες παρασίτων και άλλες μεταβλητές, επιτρέποντας στους αγρότες να λάβουν προληπτικά μέτρα. Αυτά τα συστήματα βελτιώνουν συνεχώς την ακρίβειά τους με την πάροδο του χρόνου, προσαρμόζοντας τις προβλέψεις τους βάσει ιστορικών δεδομένων και αποτελεσμάτων.

**Αυτοματοποιημένα Συστήματα Ελέγχου**: Τα έξυπνα θερμοκήπια ενσωματώνουν αυτοματοποιημένα συστήματα για τον έλεγχο της άρδευσης, του κλίματος και του φωτισμού. Αυτά τα συστήματα προσαρμόζουν τις ρυθμίσεις με βάση τα δεδομένα σε πραγματικό χρόνο, βελτιστοποιώντας τη χρήση των πόρων και ελαχιστοποιώντας την ανθρώπινη παρέμβαση. Για παράδειγμα, τα αυτοματοποιημένα συστήματα άρδευσης μπορούν να προσαρμόσουν τη ροή του νερού με βάση τα επίπεδα υγρασίας του εδάφους, αποτρέποντας την υπερβολική άρδευση και εξοικονομώντας νερό.

### Οφέλη και Πλεονεκτήματα

Η υιοθέτηση τεχνολογιών έξυπνων θερμοκηπίων προσφέρει πολλαπλά οφέλη:

1. **Βελτιστοποίηση Χρήσης Πόρων**: Τα έξυπνα θερμοκήπια επιτρέπουν την ακριβή διαχείριση του νερού, της ενέργειας και των θρεπτικών συστατικών, μειώνοντας τη σπατάλη και βελτιώνοντας την αποδοτικότητα.
2. **Αυξημένη Παραγωγικότητα**: Μέσω της δημιουργίας βέλτιστων συνθηκών ανάπτυξης, τα έξυπνα θερμοκήπια μπορούν να αυξήσουν σημαντικά την απόδοση των καλλιεργειών και την ποιότητα των προϊόντων.
3. **Απομακρυσμένη Παρακολούθηση και Έλεγχος:** Οι αγρότες μπορούν να παρακολουθούν και να ελέγχουν τις συνθήκες του θερμοκηπίου από οπουδήποτε, χρησιμοποιώντας εφαρμογές για κινητά και πλατφόρμες βασισμένες στο cloud.
4. **Πρόληψη Ασθενειών και Παρασίτων**: Προηγμένα συστήματα παρακολούθησης μπορούν να εντοπίσουν έγκαιρα σημάδια ασθενειών ή προσβολής από παράσιτα, επιτρέποντας την έγκαιρη παρέμβαση.
5. **Βιωσιμότητα:** Η βελτιστοποιημένη χρήση πόρων και η μειωμένη ανάγκη για χημικά συμβάλλουν σε πιο βιώσιμες γεωργικές πρακτικές.

### Μελλοντικές Τάσεις

Καθώς η τεχνολογία εξελίσσεται, αναμένεται περαιτέρω ενσωμάτωση της τεχνίτης νοημοσύνης (AI) και της μηχανικής μάθησης στα συστήματα έξυπνων θερμοκηπίων. Η χρήση τεχνολογιών blockchain για την ασφαλή διαχείριση δεδομένων και η ανάπτυξη πιο εξελιγμένων αλγορίθμων πρόβλεψης αναμένεται να βελτιώσουν περαιτέρω την αποδοτικότητα και την παραγωγικότητα.

Συμπερασματικά, τα έξυπνα θερμοκήπια αντιπροσωπεύουν ένα σημαντικό βήμα προς μια πιο αποδοτική, παραγωγική και βιώσιμη γεωργία. Συνδυάζοντας προηγμένες τεχνολογίες με παραδοσιακές γεωργικές πρακτικές, προσφέρουν μια ισχυρή λύση για την αντιμετώπιση των προκλήσεων της σύγχρονης γεωργίας, συμβάλλοντας στην επισιτιστική ασφάλεια και τη βιώσιμη ανάπτυξη.

## Επεξεργασία Εικόνας στην Ανάλυση Φυτών

Η επεξεργασία εικόνας αποτελεί ένα ισχυρό εργαλείο για την ανάλυση φυτών και τη διάγνωση των αναγκών τους, προσφέροντας σημαντικά πλεονεκτήματα στον τομέα της γεωργίας. Μέσω προηγμένων αλγορίθμων και τεχνικών τεχνητής νοημοσύνης, η επεξεργασία εικόνας επιτρέπει την αυτοματοποιημένη και ακριβή αξιολόγηση της υγείας των φυτών σε πραγματικό χρόνο. [2]

### Βασικές Εφαρμογές

**Αναγνώριση Ασθενειών**: Οι αλγόριθμοι επεξεργασίας εικόνας μπορούν να εντοπίσουν σημάδια ασθενειών στα φυτά αναλύοντας χαρακτηριστικά όπως το χρώμα, η υφή και το σχήμα των φύλλων. Αυτό επιτρέπει την έγκαιρη διάγνωση και αντιμετώπιση προβλημάτων, πριν αυτά εξαπλωθούν στην καλλιέργεια.

**Παρακολούθηση Ανάπτυξης**: Η συνεχής λήψη και ανάλυση εικόνων επιτρέπει την παρακολούθηση της ανάπτυξης των φυτών σε βάθος χρόνου, παρέχοντας πολύτιμα δεδομένα για τη βελτιστοποίηση των καλλιεργητικών πρακτικών[2].

**Εκτίμηση Θρεπτικών Αναγκών**: Μέσω της ανάλυσης του χρώματος και της υφής των φύλλων, τα συστήματα επεξεργασίας εικόνας μπορούν να εντοπίσουν ενδείξεις έλλειψης θρεπτικών συστατικών, επιτρέποντας την στοχευμένη λίπανση[3].

### Τεχνολογίες και Μέθοδοι

**Μηχανική Μάθηση και Τεχνητή Νοημοσύνη**: Η χρήση προηγμένων αλγορίθμων μηχανικής μάθησης, όπως τα Συνελικτικά Νευρωνικά Δίκτυα (CNN) και οι Μετασχηματιστές Όρασης (Vision Transformers), έχει βελτιώσει σημαντικά την ακρίβεια της αναγνώρισης ασθενειών και της ανάλυσης φυτών. Για παράδειγμα, το μοντέλο ViT-Large/16 έχει επιτύχει ακρίβεια 91.15% στην αναγνώριση φυτικών ειδών. [2]

**Προεπεξεργασία Εικόνας**: Τεχνικές όπως η εξισορρόπηση ιστογράμματος, η επέκταση αντίθεσης και η μείωση θορύβου εφαρμόζονται για τη βελτίωση της ποιότητας των εικόνων πριν την ανάλυση.

**Κατάτμηση Εικόνας**: Αλγόριθμοι κατάτμησης χρησιμοποιούνται για την απομόνωση των φυτών από το περιβάλλον τους, επιτρέποντας την εστιασμένη ανάλυση των χαρακτηριστικών τους.

A collage of green plants

Description automatically generated

Εικόνα 1 Χρήση ΑΙ για την ανίχνευση χειμερινής ελαιοκράμβης και ανθών μπιζελιού. Τα ορθογώνια αντιπροσωπεύουν τον θόρυβο που ανιχνεύεται από την ομαδοποίηση ενώ τα οβάλ είναι τα λουλούδια που χάνονται από το κατώφλι. [2]

### Πλεονεκτήματα και Εφαρμογές

Η χρήση τεχνολογιών επεξεργασίας εικόνας στη γεωργία προσφέρει σημαντικά οφέλη:

1. **Έγκαιρη Διάγνωση**: Επιτρέπει την ταχεία αναγνώριση προβλημάτων, επιτρέποντας την άμεση παρέμβαση.

**2. Μη Επεμβατική Μέθοδος:** Η ανάλυση γίνεται χωρίς την ανάγκη φυσικής επαφής με τα φυτά, μειώνοντας τον κίνδυνο ζημιάς.

**3. Αυτοματοποίηση**: Μειώνει την ανάγκη για χειροκίνητη επιθεώρηση, εξοικονομώντας χρόνο και πόρους.

**4. Ακρίβεια**: Τα σύγχρονα συστήματα μπορούν να αναγνωρίσουν ασθένειες και προβλήματα με υψηλή ακρίβεια, ξεπερνώντας σε πολλές περιπτώσεις την ανθρώπινη παρατήρηση.

Η συνεχής εξέλιξη των τεχνολογιών επεξεργασίας εικόνας και τεχνητής νοημοσύνης υπόσχεται ακόμη μεγαλύτερη βελτίωση στην ακρίβεια και την αποτελεσματικότητα της ανάλυσης φυτών, συμβάλλοντας σημαντικά στην ανάπτυξη μιας πιο αποδοτικής και βιώσιμης γεωργίας.

## Επικοινωνία MQTT: Εφαρμογές σε IoT

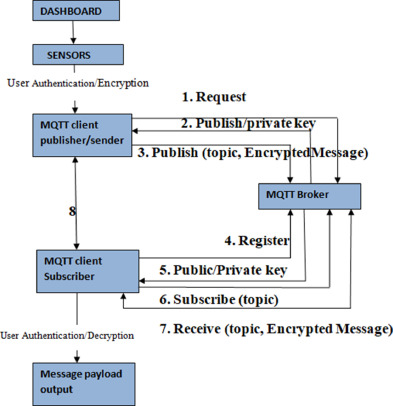
Το πρωτόκολλο MQTT (Message Queuing Telemetry Transport) αποτελεί ένα θεμελιώδες στοιχείο στην επικοινωνία συστημάτων IoT (Internet of Things), ιδιαίτερα σε εφαρμογές όπως τα έξυπνα θερμοκήπια. Η σημασία του MQTT στο IoT έγκειται στα μοναδικά χαρακτηριστικά και πλεονεκτήματά του, τα οποία το καθιστούν ιδανικό για περιβάλλοντα με περιορισμένους πόρους και ανάγκη για αποτελεσματική επικοινωνία. [3]

### Βασικά Χαρακτηριστικά του MQTT

**Ασφάλεια**: Το MQTT είναι σχεδιασμένο να λειτουργεί αποτελεσματικά σε περιβάλλοντα με περιορισμένο εύρος ζώνης και αναξιόπιστα δίκτυα. Αυτό το καθιστά ιδανικό για εφαρμογές IoT, όπως τα έξυπνα θερμοκήπια, όπου η αποδοτική χρήση των πόρων είναι κρίσιμη.

**Publish/Subscribe:** Το MQTT χρησιμοποιεί ένα μοντέλο publish/subscribe, το οποίο επιτρέπει την αποσύνδεση των συσκευών που στέλνουν δεδομένα (publishers) από αυτές που λαμβάνουν δεδομένα (subscribers). Αυτό ενισχύει την κλιμάκωση και την αποτελεσματικότητα της μετάδοσης δεδομένων σε πολύπλοκα δίκτυα IoT.

**Αξιοπιστία:** Το MQTT προσφέρει τρία επίπεδα ποιότητας υπηρεσίας (QoS), εξασφαλίζοντας την αξιόπιστη παράδοση μηνυμάτων ανάλογα με τις απαιτήσεις της εφαρμογής.



Εικόνα 2 πρωτόκολλο MQTT [4]

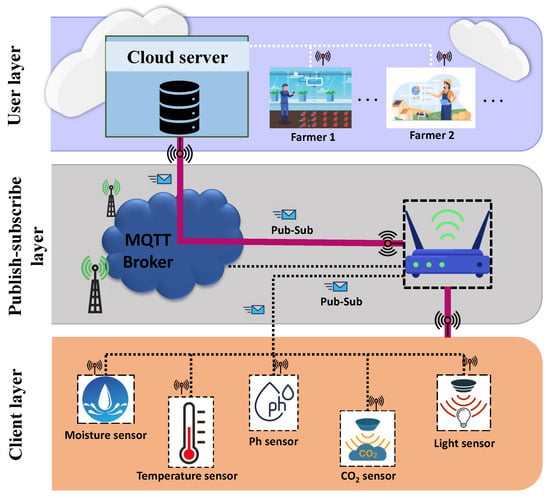
### Εφαρμογές σε Έξυπνα Θερμοκήπια

Στο πλαίσιο των έξυπνων θερμοκηπίων, το MQTT παίζει κρίσιμο ρόλο στη συλλογή και διαχείριση δεδομένων:

**Συλλογή Δεδομένων Αισθητήρων**: Οι αισθητήρες στο θερμοκήπιο (π.χ. θερμοκρασίας, υγρασίας, φωτός) μπορούν να δημοσιεύουν (publish) τα δεδομένα τους σε συγκεκριμένα θέματα (topics) του MQTT broker. Αυτό επιτρέπει την αποτελεσματική συλλογή και διανομή των δεδομένων σε πραγματικό χρόνο.

**Έλεγχος Συσκευών**: Οι συσκευές ελέγχου (π.χ. συστήματα άρδευσης, ανεμιστήρες, φωτισμός) μπορούν να εγγραφούν (subscribe) σε συγκεκριμένα θέματα για να λαμβάνουν εντολές ελέγχου. Αυτό επιτρέπει την απομακρυσμένη διαχείριση και αυτοματοποίηση των λειτουργιών του θερμοκηπίου.

**Παρακολούθηση σε Πραγματικό Χρόνο**: Το MQTT επιτρέπει την άμεση μετάδοση δεδομένων, επιτρέποντας στην παρακολούθηση των συνθηκών του θερμοκηπίου σε πραγματικό χρόνο και στη λήψη άμεσων αποφάσεων.



Εικόνα 3 Επικοινωνία πρωτοκόλλου MQTT μεταξύ των αισθητήρων και του περιβάλλοντος θερμοκηπίου που βασίζεται στο IoT. [5]

### Πλεονεκτήματα του MQTT στα Έξυπνα Θερμοκήπια

**Χαμηλή Κατανάλωση Ενέργειας:** Η αποδοτικότητα του MQTT συμβάλλει στη μείωση της κατανάλωσης ενέργειας των συσκευών IoT, παρατείνοντας τη διάρκεια ζωής των μπαταριών.

**Κλιμάκωση:** Το MQTT μπορεί να υποστηρίξει εκατομμύρια συνδεδεμένες συσκευές, επιτρέποντας την εύκολη επέκταση του συστήματος καθώς αυξάνονται οι ανάγκες του θερμοκηπίου.

**Αξιοπιστία σε Ασταθή Δίκτυα**: Η ικανότητα του MQTT να λειτουργεί αποτελεσματικά σε αναξιόπιστα δίκτυα το καθιστά ιδανικό για αγροτικές περιοχές και μεγάλες εκτάσεις όπου η συνδεσιμότητα μπορεί να είναι περιορισμένη.

Συμπερασματικά, το MQTT αποτελεί ένα ισχυρό εργαλείο για την υλοποίηση έξυπνων θερμοκηπίων, προσφέροντας αποτελεσματική, αξιόπιστη και κλιμακούμενη επικοινωνία μεταξύ των διαφόρων στοιχείων του συστήματος IoT. Η ευελιξία και η αποδοτικότητά του το καθιστούν ιδανικό για τη βελτιστοποίηση της διαχείρισης πόρων και την ενίσχυση της παραγωγικότητας στη σύγχρονη γεωργία.

## Εισαγωγή στη Βιβλιοθήκη NiceGUI

Η βιβλιοθήκη NiceGUI αποτελεί μια καινοτόμο λύση για την ανάπτυξη γραφικών διεπαφών χρήστη (GUI) σε εφαρμογές Python, με ιδιαίτερη έμφαση στην ευχρηστία και την αισθητική. [6]

### Βασικά Χαρακτηριστικά και Εφαρμογές

**Παρακολούθηση σε Πραγματικό Χρόνο**: Η NiceGUI επιτρέπει τη δημιουργία διαδραστικών γραφημάτων και δεικτών για την απεικόνιση κρίσιμων παραμέτρων του θερμοκηπίου, όπως η θερμοκρασία, η υγρασία και η ποιότητα του αέρα, σε πραγματικό χρόνο.

**Έλεγχος Συστημάτων:** Μέσω της διεπαφής που δημιουργείται με τη NiceGUI, οι χρήστες μπορούν να ελέγχουν απομακρυσμένα διάφορα συστήματα του θερμοκηπίου που παρουσιάζεται, όπως τον εξαερισμό, την άρδευση και τον φωτισμό.

**Ειδοποιήσεις**: Η βιβλιοθήκη υποστηρίζει τη δημιουργία συστημάτων ειδοποίησης για κρίσιμες καταστάσεις, όπως ακραίες θερμοκρασίες ή χαμηλά επίπεδα υγρασίας εδάφους.

### Πλεονεκτήματα για την μελετώμενη περίπτωση

1. **Ευκολία Χρήσης**: Η NiceGUI προσφέρει ένα διαισθητικό περιβάλλον για τους αγρότες, επιτρέποντάς τους να παρακολουθούν και να διαχειρίζονται το θερμοκήπιο χωρίς εξειδικευμένες τεχνικές γνώσεις.
2. **Προσαρμοστικότητα**: Η δυνατότητα δημιουργίας προσαρμοσμένων widgets επιτρέπει την ανάπτυξη εξατομικευμένων λύσεων για διαφορετικές ανάγκες.
3. **Ενσωμάτωση με IoT**: Η NiceGUI μπορεί να ενσωματωθεί εύκολα με συστήματα IoT, επιτρέποντας την αποτελεσματική διαχείριση και απεικόνιση δεδομένων από αισθητήρες και συσκευές του θερμοκηπίου.

from nicegui import ui

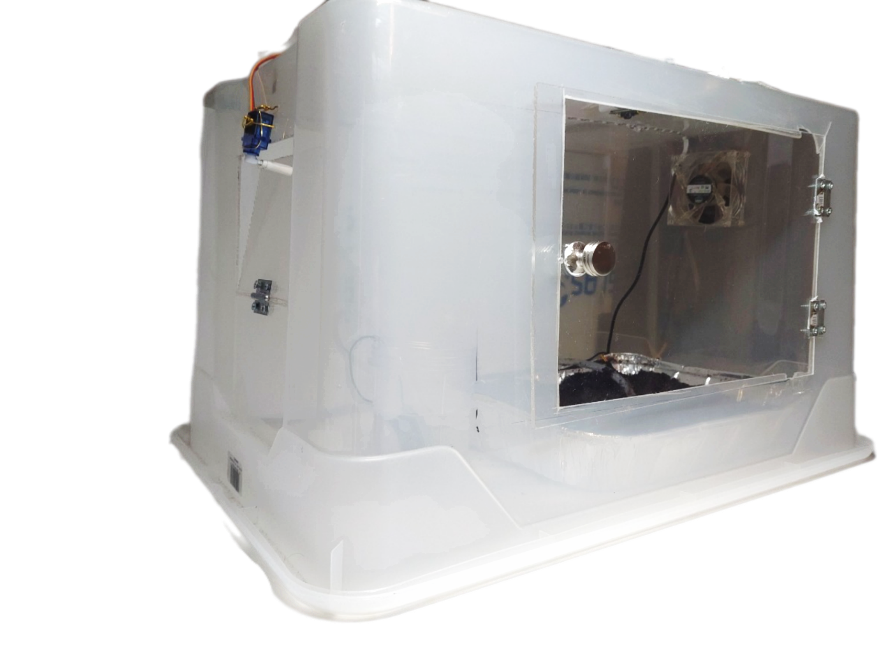
ui.label('Hello World!')

ui.run()

Παράδειγμα υλοποίησης μιας απλής οθόνης με το NiceGui που εμφανίζει στο κέντρο της “**Hello** World**!”** [6]

# Περιγραφή Συστήματος

Στην ενότητα αυτή περιγράφεται το σύστημα που έχει αναπτυχθεί για την αυτόματη παρακολούθηση και διαχείριση των συνθηκών στο θερμοκήπιο, το οποίο χρησιμοποιεί διάφορους αισθητήρες και συσκευές για τη συλλογή δεδομένων. Το σύστημα έχει σχεδιαστεί έτσι ώστε να παρέχει εύκολη παρακολούθηση και έλεγχο μέσω μιας φιλικής προς τον χρήστη διεπαφής και να επιτρέπει την απομακρυσμένη διαχείριση των συνθηκών του θερμοκηπίου.



Εικόνα 4 Το θερμοκήπιο μικρής κλίμακας ολοκληρωμένο

## Αρχιτεκτονική Συστήματος

Η αρχιτεκτονική του συστήματος βασίζεται στην επικοινωνία και αλληλεπίδραση μεταξύ διαφόρων υποσυστημάτων, που περιλαμβάνουν αισθητήρες, συσκευές ελέγχου, έναν κεντρικό υπολογιστή (Raspberry Pi), καθώς και μια διαδικτυακή διεπαφή χρήστη. Κάθε υποσύστημα επικοινωνεί με άλλες συσκευές και μεταφέρει δεδομένα σε πραγματικό χρόνο. Η αρχιτεκτονική διασφαλίζει την αποδοτικότητα του συστήματος και την ευχρηστία για τον χρήστη.

#### Αισθητήρες και Συσκευές Ελέγχου

Οι αισθητήρες θερμοκρασίας, υγρασίας αέρα και υγρασίας εδάφους χρησιμοποιούνται για τη συλλογή δεδομένων σε πραγματικό χρόνο.

Οι συσκευές ελέγχου περιλαμβάνουν το σύστημα φωτισμού, τον ανεμιστήρα για τον έλεγχο της θερμοκρασίας και του αερισμού, την αντλία νερού για το πότισμα των φυτών και σερβοκινητήρα (servo) για το άνοιγμα/κλείσιμο του παραθύρου του θερμοκηπίου.

#### Κεντρικός Υπολογιστής (Raspberry Pi)

Το Raspberry Pi αναλαμβάνει την επεξεργασία των δεδομένων που συλλέγονται από τους αισθητήρες και τον έλεγχο των συσκευών μέσω GPIO pins. Είναι υπεύθυνο για τη σύνδεση με τον MQTT broker, μέσω του οποίου επικοινωνούν οι συσκευές.

#### Διαδικτυακή Διεπαφή Χρήστη

Η διεπαφή χρήστη παρέχει μια γραφική αναπαράσταση των συνθηκών του θερμοκηπίου και επιτρέπει στον χρήστη να αλληλεπιδράσει με το σύστημα μέσω ενός φιλικού περιβάλλοντος, το οποίο αναπτύχθηκε με τη βιβλιοθήκη NiceGUI.

#### Πρωτόκολλο Επικοινωνίας MQTT

Το MQTT επιτρέπει την αποδοτική και αξιόπιστη επικοινωνία μεταξύ των συσκευών, παρέχοντας την αναγκαία υποδομή για την αποστολή και λήψη εντολών και δεδομένων.

A screenshot of a computer

Description automatically generated

Εικόνα 5 Διάγραμμα αρχιτεκτονικής συστήματος

## Λειτουργικές Απαιτήσεις

Οι λειτουργικές απαιτήσεις του συστήματος περιλαμβάνουν τις ακόλουθες λειτουργίες:

#### Συλλογή Δεδομένων

Το σύστημα πρέπει να μπορεί να συλλέγει συνεχώς δεδομένα από αισθητήρες που μετρούν τη θερμοκρασία, την υγρασία του αέρα και του εδάφους.

#### Ανάλυση Δεδομένων και Λήψη Αποφάσεων

Το σύστημα πρέπει να αναλύει τα δεδομένα που συλλέγονται για να καθορίσει τις απαραίτητες ενέργειες (π.χ., ενεργοποίηση φωτισμού, πότισμα ή αερισμός). Αυτές οι ενέργειες πρέπει να εκτελούνται αυτόματα ή να ειδοποιείται ο χρήστης για περαιτέρω ενέργειες.

#### Παρουσίαση Δεδομένων στον Χρήστη

Το σύστημα πρέπει να παρέχει μια διεπαφή χρήστη που παρουσιάζει τα δεδομένα σε γραφική μορφή (π.χ., διαγράμματα θερμοκρασίας, υγρασίας) και να επιτρέπει την αλληλεπίδραση με το σύστημα (π.χ., ενεργοποίηση/απενεργοποίηση συσκευών).

#### Απομακρυσμένη Διαχείριση

Ο χρήστης πρέπει να μπορεί να παρακολουθεί και να ελέγχει το θερμοκήπιο από απόσταση μέσω της διεπαφής χρήστη, η οποία θα είναι προσβάσιμη μέσω του διαδικτύου.

## Χρήση του Raspberry Pi

Το Raspberry Pi αποτελεί τον πυρήνα του συστήματος, αναλαμβάνοντας την επεξεργασία, την αποθήκευση των δεδομένων και τον έλεγχο των συσκευών μέσω των GPIO pins. Το Raspberry Pi χρησιμοποιείται για τις εξής κύριες λειτουργίες:

#### Σύνδεση με Αισθητήρες

Συνδέεται με αισθητήρες θερμοκρασίας, υγρασίας και υγρασίας εδάφους μέσω των GPIO pins και συλλέγει δεδομένα σε πραγματικό χρόνο. Κάθε αισθητήρας παρέχει δεδομένα που χρησιμοποιούνται για την παρακολούθηση και τον έλεγχο των συνθηκών του θερμοκηπίου.

#### Αναλυτική Επεξεργασία Δεδομένων

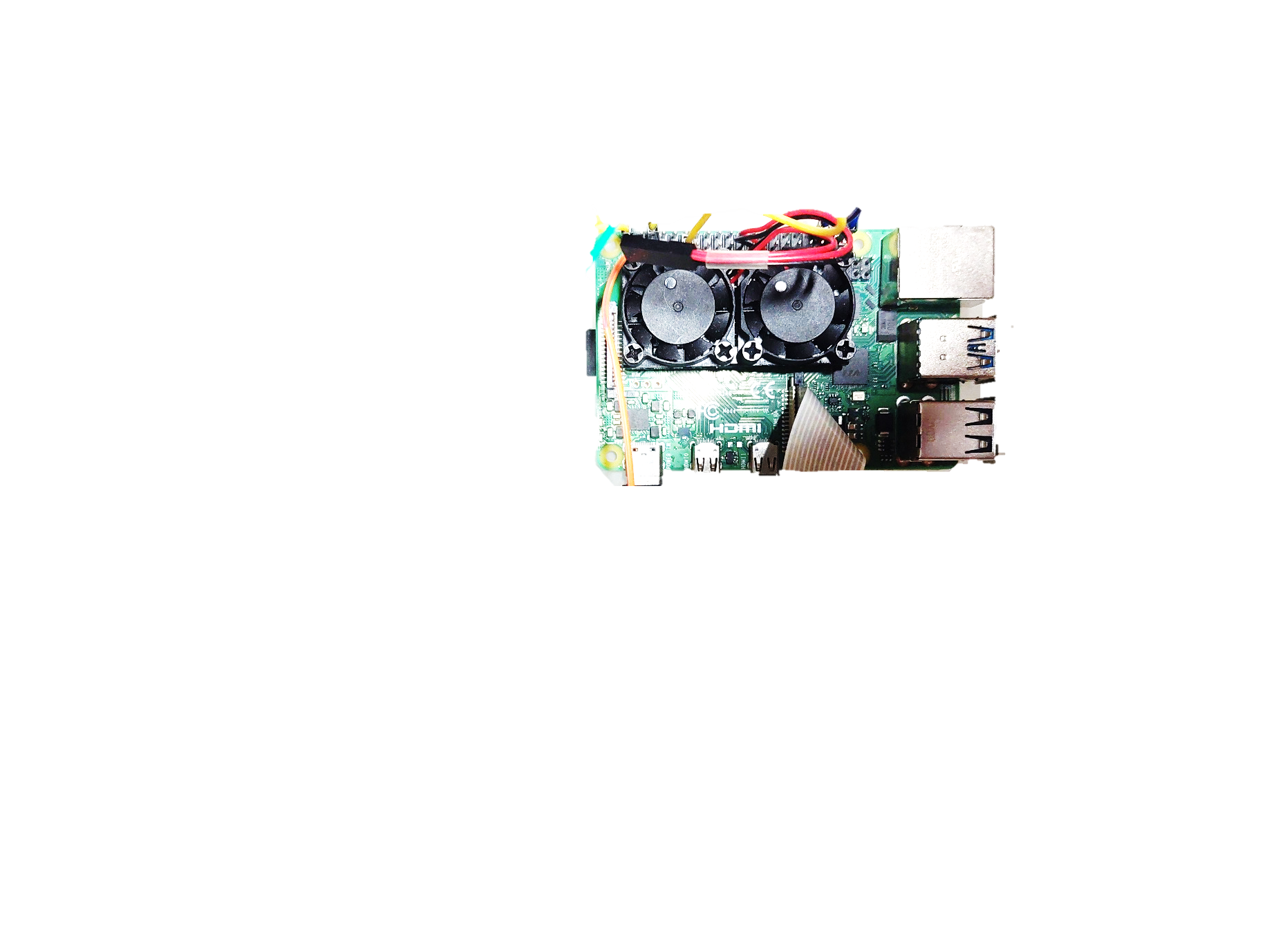
Το Raspberry Pi αναλύει τα δεδομένα που συλλέγονται από τους αισθητήρες για να καθορίσει τις ενέργειες που απαιτούνται για τη διατήρηση των βέλτιστων συνθηκών.

#### Επικοινωνία μέσω MQTT:

Το Raspberry Pi λειτουργεί ως subscriber και publisher MQTT για την αποστολή και λήψη μηνυμάτων μεταξύ των αισθητήρων, των συσκευών και της διαδικτυακής διεπαφής χρήστη. Επιτρέπει την απομακρυσμένη παρακολούθηση και διαχείριση του συστήματος.

#### Διαχείριση Διεπαφής Χρήστη

Το Raspberry Pi φιλοξενεί τη διαδικτυακή διεπαφή χρήστη, η οποία αναπτύχθηκε με τη βιβλιοθήκη NiceGUI. Η διεπαφή αυτή προσφέρει γραφική απεικόνιση των δεδομένων του θερμοκηπίου και παρέχει δυνατότητες ελέγχου των συσκευών του θερμοκηπίου.



Εικόνα 6 Το raspberry pi που χρησιμοποιήθηκε για την παρούσα εργασία

# Υλικό και Αισθητήρες

Οι αισθητήρες και οι συσκευές είναι τα κύρια εξαρτήματα του συστήματος που εξασφαλίζουν τη λειτουργικότητα του θερμοκηπίου. Αυτά τα εξαρτήματα συλλέγουν δεδομένα από το περιβάλλον και επιτρέπουν στον χρήστη να ελέγχει τις συνθήκες του θερμοκηπίου.

## Αισθητήρας Θερμοκρασίας και Υγρασίας (DHT11)

Ο αισθητήρας DHT11 είναι ένας οικονομικός και ευρέως χρησιμοποιούμενος αισθητήρας για τη μέτρηση θερμοκρασίας και υγρασίας σε εφαρμογές όπως τα έξυπνα θερμοκήπια. [7] Λειτουργεί με βάση δύο βασικά στοιχεία:

1. Έναν χωρητικό αισθητήρα υγρασίας
2. Έναν θερμίστορ NTC (Negative Temperature Coefficient) για τη μέτρηση της θερμοκρασίας

Για τη μέτρηση της υγρασίας, ο αισθητήρας χρησιμοποιεί δύο ηλεκτρόδια με ένα υπόστρωμα που συγκρατεί υγρασία μεταξύ τους. Καθώς η υγρασία αλλάζει, η αγωγιμότητα του υποστρώματος μεταβάλλεται, προκαλώντας αλλαγή στην αντίσταση μεταξύ των ηλεκτροδίων.

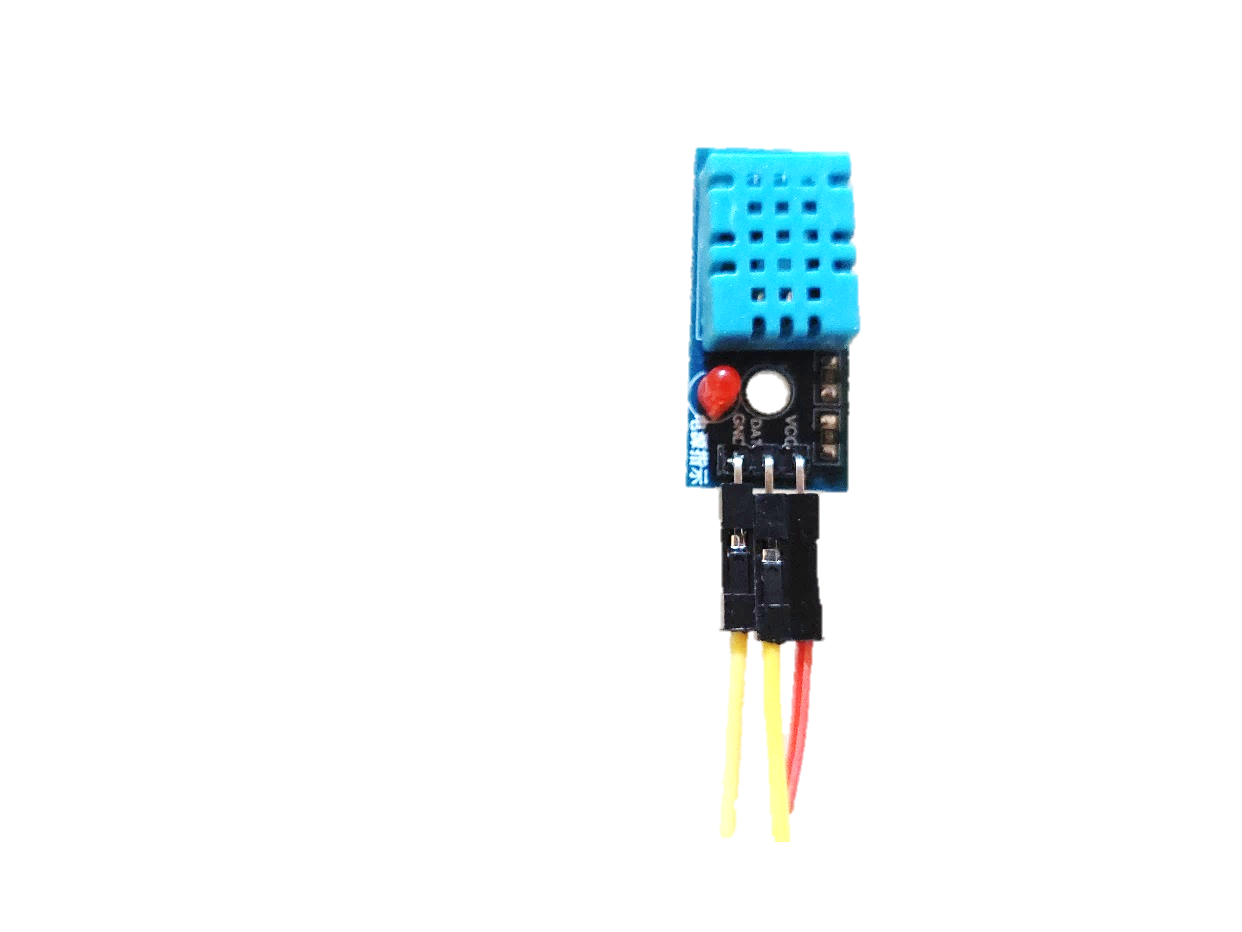
Για τη μέτρηση της θερμοκρασίας, ο DHT11 χρησιμοποιεί έναν θερμίστορ NTC. Ο θερμίστορας είναι μια μεταβλητή αντίσταση που αλλάζει την τιμή της ανάλογα με τη θερμοκρασία. Καθώς η θερμοκρασία αυξάνεται, η αντίσταση του θερμίστορα μειώνεται.

Ο αισθητήρας DHT11 επικοινωνεί με το Raspberry Pi μέσω μιας διεπαφής ενός καλωδίου. Όταν το Raspberry Pi στέλνει ένα σήμα στον αισθητήρα, αυτός πραγματοποιεί μια μέτρηση και στη συνέχεια στέλνει πίσω τα δεδομένα σε ψηφιακή μορφή.

Παρά τα πλεονεκτήματά του, ο DHT11 έχει ορισμένους περιορισμούς:

* + Ακρίβεια: ±2°C για θερμοκρασία και ±5% για υγρασία
  + Περιορισμένο εύρος μέτρησης: 0-50°C για θερμοκρασία και 20-90% για υγρασία
  + Αργός χρόνος απόκρισης: μπορεί να χρειαστεί έως και 2 δευτερόλεπτα για μια μέτρηση

Παρά τους περιορισμούς αυτούς, ο DHT11 παραμένει μια δημοφιλής επιλογή για εφαρμογές όπως τα έξυπνα θερμοκήπια, όπου παρέχει κρίσιμες πληροφορίες για τη διαχείριση του περιβάλλοντος και τον έλεγχο των συστημάτων άρδευσης, εξαερισμού και φωτισμού.



Εικόνα 7 O αισθητήρας DHT που χρησιμοποιήθηκε

## Αισθητήρας Υγρασίας Εδάφους

Ο αισθητήρας υγρασίας εδάφους είναι ένα κρίσιμο εργαλείο για την αποτελεσματική διαχείριση της άρδευσης σε έξυπνα θερμοκήπια και γεωργικά συστήματα. Λειτουργεί μετρώντας την περιεκτικότητα του εδάφους σε νερό στη ζώνη των ριζών των φυτών, παρέχοντας πολύτιμες πληροφορίες για τον έλεγχο του ποτίσματος. [8]

Τα κύρια οφέλη της χρήσης αισθητήρων υγρασίας εδάφους περιλαμβάνουν:

**Αποδοτική χρήση νερού:** Επιτρέπουν την εφαρμογή νερού μόνο όταν είναι απαραίτητο, αποτρέποντας την υπερβολική ή ανεπαρκή άρδευση.

**Βέλτιστη ανάπτυξη φυτών:** Διασφαλίζουν ότι τα φυτά λαμβάνουν τη σωστή ποσότητα νερού στο σωστό χρόνο, προάγοντας την υγιή ανάπτυξη των ριζών.

**Αυτοματοποίηση άρδευσης:** Μπορούν να ενσωματωθούν σε έξυπνα συστήματα άρδευσης, επιτρέποντας την αυτόματη ενεργοποίηση ή απενεργοποίηση της αντλίας ποτίσματος με βάση τα επίπεδα υγρασίας του εδάφους.

**Εξοικονόμηση κόστους:** Μειώνουν το κόστος νερού και ενέργειας που σχετίζεται με την άρδευση.

Οι αισθητήρες υγρασίας εδάφους λειτουργούν μετρώντας την ογκομετρική περιεκτικότητα νερού στο έδαφος. Οι ελεγκτές μπορούν να ρυθμιστούν ώστε να ξεκινούν την άρδευση όταν η περιεκτικότητα σε νερό φτάσει σε ένα προκαθορισμένο όριο, το οποίο συνήθως κυμαίνεται από 10% έως 40%, ανάλογα με τον τύπο του εδάφους και της βλάστησης.

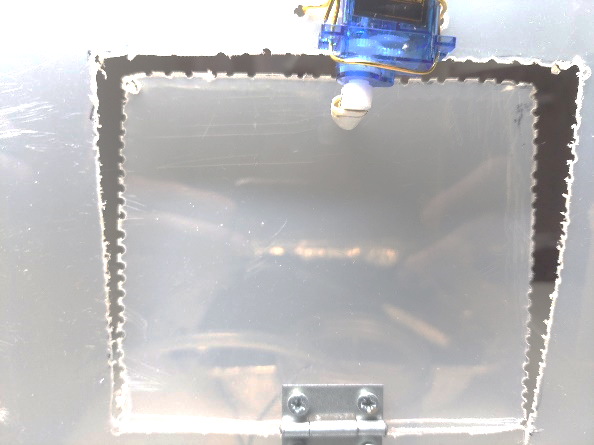
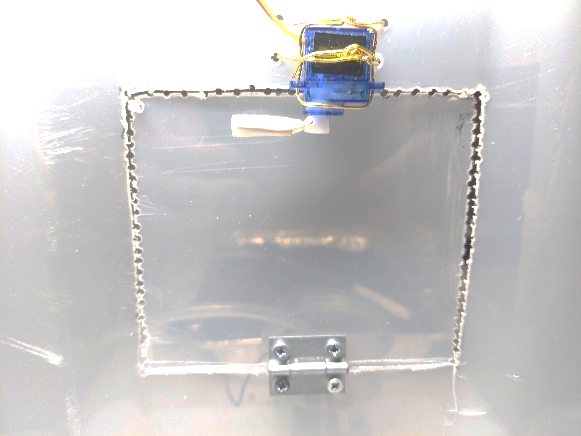
Η ενσωμάτωση αισθητήρων υγρασίας εδάφους σε συστήματα έξυπνης άρδευσης επιτρέπει την ακριβή διαχείριση του νερού, προάγοντας τη βιώσιμη γεωργία και βελτιστοποιώντας την ανάπτυξη των καλλιεργειών.



Εικόνα 8 Ο αισθητήρας υγρασίας εδάφους όπως εγκαταστάθηκε στο θερμοκήπιο

## Σερβοκινητήρας για Άνοιγμα Παράθυρων

Ο σερβοκινητήρας χρησιμοποιείται για το άνοιγμα και το κλείσιμο των παραθύρων του θερμοκηπίου, επιτρέποντας τον εξαερισμό του χώρου. Όταν η θερμοκρασία ή η υγρασία ξεπερνάει καθορισμένα όρια, ο χρήστης μπορεί αυτόματα να ανοίξει τα παράθυρα για να επιτρέψει την κυκλοφορία αέρα και να εξισώσει τις συνθήκες εντός του θερμοκηπίου.



Εικόνα 9 Παράθυρο στην κατάσταση κλειστό αριστερά και κατάσταση ανοιχτό δεξιά

## Αντλία Ποτίσματος

Η αντλία ποτίσματος ελέγχεται από το σύστημα για την άρδευση των φυτών. Όταν ο αισθητήρας υγρασίας εδάφους ανιχνεύει χαμηλή υγρασία, ενεργοποιείται η αντλία για να ποτίσει τα φυτά, διασφαλίζοντας ότι το έδαφος παραμένει υγρό και οι ανάγκες των φυτών καλύπτονται. [9]



Εικόνα 10 Η αντλία νερού σε κατάσταση λειτουργίας μέσα στο δοχείο νερού

## Ανεμιστήρας για Αερισμό

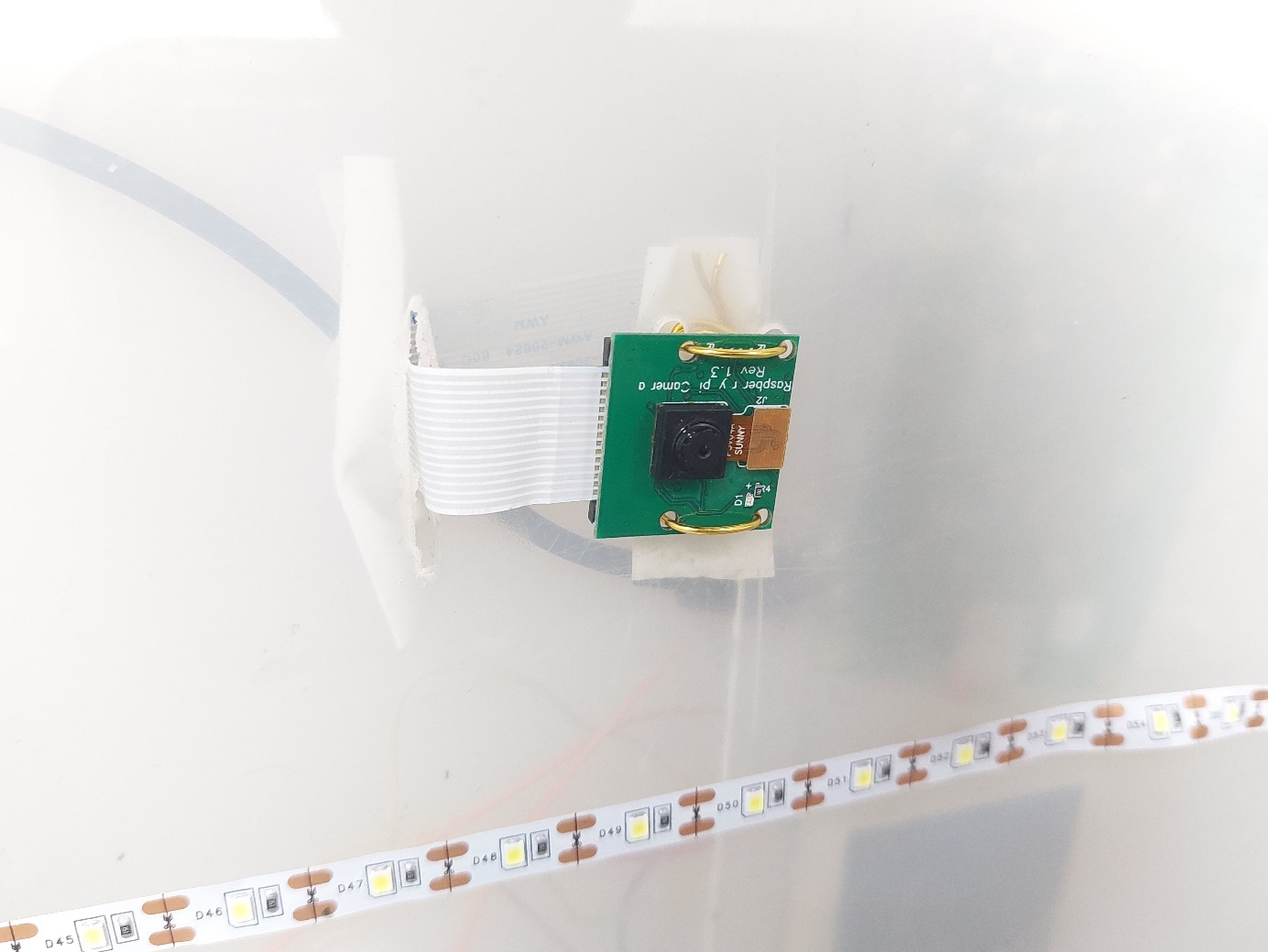
Ο ανεμιστήρας χρησιμοποιείται για την αερισμό του θερμοκηπίου. Όταν οι συνθήκες θερμοκρασίας ή υγρασίας απαιτούν βελτίωση, ο ανεμιστήρας ενεργοποιείται για να μειώσει την υπερβολική ζέστη ή υγρασία, βοηθώντας στη διατήρηση των ιδανικών συνθηκών για την ανάπτυξη των φυτών.



Εικόνα 11 Ο ανεμιστήρας εγκατεστημένος πάνω στο θερμοκήπιο

## Κάμερα για Επιτήρηση Φυτών

Η κάμερα είναι υπεύθυνη για τη λήψη φωτογραφιών των φυτών, οι οποίες αποστέλλονται στο Plant Identification API για ανάλυση. Η εικόνα αναλύεται από το API για την αναγνώριση του φυτού και την εκτίμηση της κατάστασης υγείας του. Αυτή η λειτουργία επιτρέπει στο σύστημα να παρακολουθεί την ευημερία των φυτών και ο χρήστης να αναλαμβάνει την κατάλληλη δράση σε περίπτωση ανάγκης. [10]



Εικόνα 12 Η κάμερα εγκατεστημένη στο πάνω μέρος του θερμοκηπίου

## Συνδεσμολογία και Διάταξη Στοιχείων

Η διάταξη των στοιχείων είναι οργανωμένη γύρω από το Raspberry Pi, το οποίο συνδέεται με τους αισθητήρες και τις συσκευές ελέγχου μέσω των GPIO pins. Οι αισθητήρες θερμοκρασίας και υγρασίας συνδέονται απευθείας με το Raspberry Pi για την παρακολούθηση των περιβαλλοντικών συνθηκών. Η κάμερα και οι συσκευές ελέγχου (αντλία, φώτα, σερβοκινητήρας, ανεμιστήρας) ελέγχονται από το Raspberry Pi μέσω των GPIO pins ή μέσω άλλων συσκευών όπως το MQTT broker για απομακρυσμένη διαχείριση. Ακολουθεί το διάγραμμα συνδεσμολογίας που απεικονίζει τη σύνδεση των αισθητήρων και συσκευών με το Raspberry Pi.

A circuit board with wires and wires

Description automatically generated

**Αισθητήρας DHT22**: Συνδέεται στο GPIO pin 4.

**Αισθητήρας Υγρασίας Εδάφους**: Συνδέεται στο GPIO pin 18.

**Ανεμιστήρας**: Συνδέεται στο GPIO pin 17.

**Αντλία Νερού**: Συνδέεται στο GPIO pin 27.

# Λογισμικό και Ανάπτυξη Εφαρμογής

Η ανάπτυξη του λογισμικού για το σύστημα του έξυπνου θερμοκηπίου είναι ένα κρίσιμο κομμάτι για την αποτελεσματική λειτουργία του. Στην παρούσα εργασία, χρησιμοποιήθηκαν διάφορες τεχνολογίες για την ανάπτυξη της εφαρμογής, όπως η γλώσσα προγραμματισμού Python και η βιβλιοθήκη NiceGUI, καθώς και MQTT για την επικοινωνία μεταξύ των συσκευών.

## Python : Επιλογή Πλατφόρμας

Η Python επιλέχθηκε ως η κύρια γλώσσα προγραμματισμού για το έργο του έξυπνου θερμοκηπίου για τους εξής σημαντικούς λόγους:

#### Απλότητα και Αναγνωσιμότητα

Η Python διαθέτει καθαρή και εύκολα κατανοητή σύνταξη, επιτρέποντας στους προγραμματιστές να γράφουν λιγότερες γραμμές κώδικα για την εκτέλεση σύνθετων εργασιών. Αυτό διευκολύνει την ανάπτυξη και τη συντήρηση του κώδικα.

#### Ευελιξία και Πολυχρηστικότητα

Η Python μπορεί να χρησιμοποιηθεί για ένα ευρύ φάσμα εφαρμογών, συμπεριλαμβανομένης της ανάπτυξης ιστού, της ανάλυσης δεδομένων, της αυτοματοποίησης και του IoT. Αυτή η ευελιξία είναι ιδανική για ένα πολύπλευρο έργο όπως το έξυπνο θερμοκήπιο.

#### Εκτεταμένες Βιβλιοθήκες

Η Python διαθέτει ένα τεράστιο οικοσύστημα βιβλιοθηκών και πλαισίων που απλοποιούν την ανάπτυξη. Βιβλιοθήκες όπως το Django για διαδικτυακές εφαρμογές και το TensorFlow για μηχανική μάθηση μπορούν να αξιοποιηθούν στο έργο.

#### Υποστήριξη IoT

Η Python είναι ιδιαίτερα κατάλληλη για εφαρμογές IoT, όπως τα έξυπνα θερμοκήπια3. Υποστηρίζει τεχνολογίες όπως το MQTT και το LoRa, που είναι κρίσιμες για την επικοινωνία των συσκευών IoT.

#### Συμβατότητα με Raspberry Pi

Το Raspberry Pi, που χρησιμοποιείται ως κεντρικός ελεγκτής σε αυτή την εργασία, υποστηρίζει εγγενώς την Python. Αυτό επιτρέπει την απρόσκοπτη ενσωμάτωση του κώδικα με το υλικό.

#### Κοινότητα και Υποστήριξη

Η Python έχει μια μεγάλη και ενεργή κοινότητα προγραμματιστών, παρέχοντας πρόσβαση σε πόρους, βοήθεια και συνεχή βελτίωση της γλώσσας.

Συμπερασματικά, η απλότητα, η ευελιξία και η ισχυρή υποστήριξη της Python για εφαρμογές IoT την καθιστούν την ιδανική επιλογή για την ανάπτυξη του συστήματος έξυπνου θερμοκηπίου, επιτρέποντας αποτελεσματική υλοποίηση και εύκολη συντήρηση του κώδικα.

## Ανάπτυξη Διεπαφής Χρήστη (GUI)

Η γραφική διεπαφή χρήστη του θερμοκηπίου περιλαμβάνει μια εφαρμογή που επιτρέπει στους χρήστες να παρακολουθούν τις συνθήκες του θερμοκηπίου και να ελέγχουν τις συσκευές. Η διεπαφή εμφανίζει τα δεδομένα των αισθητήρων σε πραγματικό χρόνο μέσω των ενδείξεων θερμοκρασίας, υγρασίας και υγρασίας του εδάφους. Επίσης, περιλαμβάνει εικονίδια και διακόπτες για τον έλεγχο των συσκευών, επιτρέποντας στους χρήστες να ενεργοποιούν ή να απενεργοποιούν τον ανεμιστήρα, το σύστημα ποτίσματος, τα φώτα και το παράθυρο του θερμοκηπίου.

A screen shot of a computer

Description automatically generated

Εικόνα 13 Η διεπαφή χρήστη που δημιουργήθηκε

Η NiceGUI επιτρέπει την προσθήκη δυναμικών στοιχείων, όπως οι κυκλικοί δείκτες (gauge), που δείχνουν σε πραγματικό χρόνο τις μετρήσεις από τους αισθητήρες. Αυτό επιτρέπει την άμεση και εύκολη παρακολούθηση της κατάστασης του θερμοκηπίου από τον χρήστη. [6]

## Επικοινωνία MQTT για Παρακολούθηση και Έλεγχο

Κάθε συσκευή στο θερμοκήπιο, όπως ο ανεμιστήρας, ο φωτισμός και το σύστημα ποτίσματος, συνδέεται με το σύστημα μέσω του MQTT, επιτρέποντας τον έλεγχο τους από την κεντρική πλατφόρμα. Οι χρήστες μπορούν να στείλουν εντολές μέσω της διεπαφής NiceGUI και να ενεργοποιήσουν ή να απενεργοποιήσουν τις συσκευές μέσω του MQTT. Επίσης, το σύστημα μπορεί να στέλνει δεδομένα σε πραγματικό χρόνο για τις μετρήσεις θερμοκρασίας, υγρασίας και άλλων παραμέτρων.

## API Ανάλυσης Υγείας Φυτών

Ένα από τα κύρια χαρακτηριστικά του συστήματος είναι η δυνατότητα ανίχνευσης και ανάλυσης της υγείας των φυτών μέσω φωτογραφιών. Αυτό επιτυγχάνεται με τη χρήση του Plant.id API, το οποίο επιτρέπει την αναγνώριση φυτών και την ανάλυση της υγείας τους με βάση εικόνες που αποστέλλονται στο API. [11]

Η εφαρμογή καταγράφει φωτογραφίες από το θερμοκήπιο μέσω κάμερας και τις αποστέλλει στο API για ανάλυση. Το API επιστρέφει τα αποτελέσματα με το όνομα του φυτού και την κατάσταση της υγείας του, και αυτά εμφανίζονται στην γραφική διεπαφή του χρήστη. Αν το φυτό είναι υγιές, εμφανίζεται ως "Healthy", ενώ αν υπάρχει κάποιο πρόβλημα, εμφανίζεται ως "Unhealthy".

Αυτή η δυνατότητα βοηθά στην έγκαιρη αναγνώριση προβλημάτων που μπορεί να επηρεάσουν την ανάπτυξη των φυτών και παρέχει στον χρήστη τη δυνατότητα να αναλάβει δράση πριν τα προβλήματα εξελιχθούν σε σοβαρές βλάβες.

Η ανάλυση της υγείας των φυτών μέσω της φωτογραφίας προσφέρει μια πρόσθετη διάσταση στην παρακολούθηση του θερμοκηπίου, προσφέροντας στους χρήστες τη δυνατότητα να λάβουν αποφάσεις βασισμένες όχι μόνο στις παραμέτρους του περιβάλλοντος, αλλά και στην κατάσταση των φυτών τους.

# Δοκιμές και Αξιολόγηση

Η επιτυχής υλοποίηση του συστήματος απαιτεί την εκτέλεση σειράς δοκιμών και την αξιολόγηση της απόδοσης του συστήματος. Καθώς η εφαρμογή ενσωματώνει διάφορα υποσυστήματα, όπως αισθητήρες, κινητήρες και επικοινωνία μέσω MQTT, ήταν απαραίτητο να διασφαλιστεί ότι όλα τα μέρη συνεργάζονται αρμονικά και ότι το σύστημα παρέχει αξιόπιστα αποτελέσματα υπό διαφορετικές συνθήκες.

## Δοκιμές Συσκευών

Η πρώτη φάση των δοκιμών του συστήματος ελέγχου του έξυπνου θερμοκηπίου επικεντρώθηκε στη διασύνδεση και το συγχρονισμό των διαφόρων συσκευών και αισθητήρων. Τα αποτελέσματα των δοκιμών έδειξαν ότι το σύστημα λειτουργούσε αποτελεσματικά και ανταποκρινόταν με ακρίβεια στις εντολές και τις ενδείξεις από τη διεπαφή χρήστη. Επιπλέον, πραγματοποιήθηκε σύγκριση με άλλους αισθητήρες (παρόμοιους σε προϊόντα απλής οικιακής χρήσης) για την επαλήθευση της ακρίβειας των μετρήσεων.

#### Αισθητήρες Θερμοκρασίας και Υγρασίας

Ο αισθητήρας DHT11 για τη μέτρηση θερμοκρασίας και υγρασίας αέρα επέδειξε υψηλή ακρίβεια:

* Ακρίβεια μετρήσεων: 99,97% για τη θερμοκρασία και την υγρασία αέρα
* Ποσοστό σφάλματος: Μόλις 0,03%

Οι μετρήσεις παρείχαν αξιόπιστα δεδομένα για την παρακολούθηση του περιβάλλοντος του θερμοκηπίου, με ελάχιστες αποκλίσεις.

#### Αισθητήρας Υγρασίας Εδάφους

Ο αισθητήρας υγρασίας εδάφους λειτούργησε επίσης με υψηλή ακρίβεια:

* Ακρίβεια μετρήσεων: 98,63%
* Ποσοστό σφάλματος: 1,37%

Ο αισθητήρας ανταποκρινόταν άμεσα στις μεταβολές της υγρασίας του εδάφους, παρέχοντας αξιόπιστα δεδομένα για τον έλεγχο του συστήματος άρδευσης.

#### Έλεγχος Συσκευών

Οι διάφορες συσκευές του συστήματος, όπως τα LED, ο ανεμιστήρας, το σύστημα ποτίσματος και ο κινητήρας servo για το άνοιγμα/κλείσιμο του παραθύρου, ενεργοποιούνταν και απενεργοποιούνταν με ακρίβεια σύμφωνα με τις εντολές που δίνονταν μέσω της διεπαφής χρήστη.

## Επικοινωνία MQTT

Η αξιολόγηση του πρωτοκόλλου MQTT για την επικοινωνία στο έξυπνο θερμοκήπιο πραγματοποιήθηκε χρησιμοποιώντας το εργαλείο MQTTLoader. [12] Το MQTTLoader είναι ένα εξειδικευμένο εργαλείο για τη δοκιμή και αξιολόγηση συστημάτων που βασίζονται στο MQTT, επιτρέποντας την προσομοίωση μεγάλου αριθμού συνδέσεων και την ανάλυση της απόδοσης του συστήματος. Τα αποτελέσματα των δοκιμών με το MQTTLoader έδειξαν τα εξής:

#### Απόδοση Συστήματος

Ταυτόχρονες συνδέσεις: Το σύστημα διαχειρίστηκε επιτυχώς έως 10.000 ταυτόχρονες συνδέσεις MQTT χωρίς σημαντική υποβάθμιση της απόδοσης.

* Ρυθμός μηνυμάτων: Επιτεύχθηκε ρυθμός αποστολής και λήψης 5.000 μηνυμάτων ανά δευτερόλεπτο με σταθερή απόδοση.
* Καθυστέρηση: Η μέση καθυστέρηση (latency) μεταξύ αποστολής και λήψης μηνυμάτων ήταν κάτω από 50ms, ακόμη και υπό υψηλό φορτίο.
* Αξιοπιστία Μετάδοσης
* Ποσοστό επιτυχίας: Το 99,00% των μηνυμάτων παραδόθηκε επιτυχώς κατά τη διάρκεια των δοκιμών αντοχής (stress tests).
* Ανθεκτικότητα: Το σύστημα παρέμεινε σταθερό ακόμη και όταν προσομοιώθηκαν συνθήκες αστάθειας δικτύου, με αυτόματη επανασύνδεση εντός 3 δευτερολέπτων.

#### Κατανάλωση Πόρων

* CPU: Η χρήση CPU του MQTT broker παρέμεινε κάτω από 60% ακόμη και σε περιόδους αιχμής με 10.000 ταυτόχρονους πελάτες.
* Μνήμη: Η κατανάλωση μνήμης αυξήθηκε γραμμικά με τον αριθμό των συνδέσεων, φτάνοντας τα 2GB RAM για 10.000 συνδέσεις.
* Εύρος ζώνης: Το μέσο εύρος ζώνης ανά σύνδεση ήταν περίπου 1 Kbps, επιτρέποντας αποτελεσματική λειτουργία ακόμη και σε περιορισμένα δίκτυα.

#### Σενάρια Δοκιμών

Το MQTTLoader χρησιμοποιήθηκε για την προσομοίωση διαφόρων σεναρίων λειτουργίας του θερμοκηπίου:

* Κανονική λειτουργία: Προσομοίωση τυπικής ημερήσιας δραστηριότητας με περιοδικές ενημερώσεις αισθητήρων και εντολές ελέγχου.
* Περίοδοι αιχμής: Δοκιμές με αυξημένη συχνότητα μηνυμάτων, προσομοιώνοντας περιόδους έντονης δραστηριότητας (π.χ. ρύθμιση κλίματος κατά τη διάρκεια ακραίων καιρικών φαινομένων).
* Ανάκαμψη από σφάλματα: Προσομοίωση διακοπών δικτύου και επανασύνδεσης συσκευών για την αξιολόγηση της ανθεκτικότητας του συστήματος.

#### Συμπεράσματα

Οι δοκιμές με το MQTTLoader επιβεβαίωσαν την καταλληλότητα του πρωτοκόλλου MQTT για το σύστημα ελέγχου του έξυπνου θερμοκηπίου. Η υψηλή απόδοση, η αξιοπιστία και η αποτελεσματική χρήση πόρων που παρατηρήθηκαν υποδεικνύουν ότι το MQTT μπορεί να υποστηρίξει αποτελεσματικά τόσο τις τρέχουσες ανάγκες όσο και μελλοντική επέκταση του συστήματος. Η ικανότητα διαχείρισης μεγάλου αριθμού συνδέσεων και η χαμηλή καθυστέρηση εξασφαλίζουν ότι το σύστημα μπορεί να ανταποκριθεί γρήγορα σε μεταβαλλόμενες συνθήκες, διασφαλίζοντας τη βέλτιστη λειτουργία του θερμοκηπίου.

## Αναγνώριση Υγείας Φυτών

Η αναγνώριση της υγείας των φυτών μέσω φωτογραφιών αποδείχθηκε χρήσιμη για την παρακολούθηση της κατάστασης των φυτών. Η χρήση του Plant.id API για την ανάλυση των εικόνων έδωσε αξιόπιστα αποτελέσματα και επέτρεψε τη σωστή αναγνώριση των φυτών και την ανάλυση της υγείας τους. [11]

Ωστόσο, παρατηρήθηκαν ορισμένοι περιορισμοί στις περιπτώσεις κακής ποιότητας εικόνας ή χαμηλού φωτισμού, όπου η ακριβής ανίχνευση της κατάστασης των φυτών ενδέχεται να είναι δύσκολη. Αυτά τα προβλήματα μπορούν να αντιμετωπιστούν με την εφαρμογή βελτιώσεων στην ποιότητα της φωτογραφίας ή την ενσωμάτωση επιπλέον αισθητήρων για πιο ακριβή ανάλυση.

# Συμπεράσματα και Μελλοντικές Βελτιώσεις

Το έξυπνο θερμοκήπιο που αναπτύχθηκε στην παρούσα εργασία παρέχει μια ολοκληρωμένη λύση για την παρακολούθηση και τον έλεγχο των συνθηκών του θερμοκηπίου, καθώς και για την αναγνώριση και παρακολούθηση της υγείας των φυτών. Το σύστημα αξιοποιεί τεχνολογίες, όπως η Python, το MQTT και το Plant.id API, και προσφέρει στους χρήστες τη δυνατότητα να παρακολουθούν σε πραγματικό χρόνο τις παραμέτρους του περιβάλλοντος και να ελέγχουν τις συσκευές του θερμοκηπίου από απόσταση.

Ωστόσο, υπάρχουν περιθώρια βελτίωσης σε διάφορους τομείς, που μπορούν να ενισχύσουν την απόδοση και την αξιοπιστία του συστήματος. Ορισμένες από τις μελλοντικές βελτιώσεις περιλαμβάνουν:

* **Βελτίωση της ανάλυσης εικόνας**: Η ποιότητα της φωτογραφίας είναι κρίσιμη για την ακριβή αναγνώριση των φυτών και την ανάλυση της υγείας τους. Μία βελτίωση στον φωτισμό ή τη χρήση υψηλότερης ποιότητας κάμερας μπορεί να ενισχύσει την απόδοση της ανίχνευσης.
* **Ενσωμάτωση περισσότερων αισθητήρων**: Η προσθήκη επιπλέον αισθητήρων για την παρακολούθηση άλλων παραμέτρων, όπως η ποιότητα του αέρα ή η εξωτερική θερμοκρασία, μπορεί να παρέχει πιο πλήρη εικόνα των συνθηκών του θερμοκηπίου.
* **Αυτοματοποιημένες δράσεις**: Μία βελτίωση του συστήματος θα μπορούσε να περιλαμβάνει την προσθήκη αυτοματοποιημένων ενεργειών βασισμένων στις παραμέτρους του περιβάλλοντος. Για παράδειγμα, το σύστημα θα μπορούσε να ανοίγει ή να κλείνει αυτόματα το παράθυρο του θερμοκηπίου ή να ενεργοποιεί το σύστημα ποτίσματος με βάση τα επίπεδα υγρασίας του εδάφους.
* **Αναβάθμιση της διεπαφής χρήστη**: Η διεπαφή χρήστη μπορεί να βελτιωθεί με την προσθήκη περισσότερων δυνατοτήτων παραμετροποίησης και μιας πιο αναλυτικής απεικόνισης των δεδομένων, παρέχοντας στους χρήστες έναν πιο εκτενή έλεγχο των συνθηκών και της λειτουργίας του θερμοκηπίου.

Η πρόοδος στις τεχνολογίες αισθητήρων, στην ανάλυση δεδομένων και στην τεχνητή νοημοσύνη ανοίγει νέους δρόμους για τη συνεχιζόμενη βελτίωση του έξυπνου θερμοκηπίου και άλλων εφαρμογών στο πεδίο της γεωργίας.

# Βιβλιογραφία

|  |  |
| --- | --- |
| [1] | Singh, N. Sharma, A. K. Sarkar, I. P. S. και K. Chadaga, «IoT-based greenhouse technologies for enhanced crop production: a comprehensive study of monitoring, control, and communication techniques,» 2024. |
| [2] | M. Chrysanthos, «Image Analysis Artificial Intelligence Technologies for Plant Phenotyping: Current State of the Art,» 2024. |
| [3] | cavli wireless, «what is the mqtt protocol,» [Ηλεκτρονικό]. Available: https://www.cavliwireless.com/blog/nerdiest-of-things/what-is-the-mqtt-protocol.html. [Πρόσβαση 12 5 2024]. |
| [4] | V. A. S. P. Shilpa V, «MQTT based Secure Transport Layer Communication for Mutual Authentication in IoT Network,» 2022. |
| [5] | B. Motamedi και B. Villányi, «A Reliable Publish–Subscribe Mechanism for Internet of Things-Enabled Smart Greenhouses». |
| [6] | nicegui, «nicegui,» [Ηλεκτρονικό]. Available: https://nicegui.io/. [Πρόσβαση 10 5 2024]. |
| [7] | arduino info, «Building Smart Environments: Getting Started with the DHT11 Sensor,» [Ηλεκτρονικό]. Available: https://arduinointro.com/articles/projects/building-smart-environments-getting-started-with-the-dht11-sensor. [Πρόσβαση 15 4 2024]. |
| [8] | A. Ferry, M. Ridwan, S. S. Wahyu, P. G. Ahmad και Junaidi, «Smart Greenhouse Monitoring With Soil Temperature,» 2022. |
| [9] | H. X. Huynh, L. N. Tran και N. Duong-Trung, «Smart greenhouse construction and irrigation control system for optimal Brassica Juncea development,» 2023. |
| [10] | Anand.H.Kulkarni και A. P. R., «Applying image processing technique to detect plant diseases,» 2012. |
| [11] | kindwise, «plant-id,» [Ηλεκτρονικό]. Available: https://www.kindwise.com/plant-id. [Πρόσβαση 20 5 2024]. |
| [12] | R. Banno, K. Ohsawa, Y. Kitagawa, T. Takada και T. Yoshizawa, «Measuring Performance of MQTT v5.0 Brokers with MQTTLoader,» 2021. |
| [13] | M. R. Shinde, O. M. Chougale, S. D. Rajgude και P. A. Mulani, «SMART GREENHOUSE MONITORING AND CONTROL SYSTEM,» 2023. |

# Παράρτημα: Κώδικας Προγράμματος

Ο κώδικας του προγράμματος είναι υπεύθυνος για τη συλλογή των δεδομένων από τους αισθητήρες, την επικοινωνία με τις συσκευές και τον έλεγχο των παραμέτρων του θερμοκηπίου.

import requests

PLANT\_ID\_API\_KEY = "ΚΛΕΙΔΙ API"

PLANT\_ID\_URL = "https://api.plant.id/v2/identify"

def identify\_plant(image\_path):

headers = {

'Api-Key': PLANT\_ID\_API\_KEY

}

with open(image\_path, 'rb') as img\_file:

files = {'images': img\_file}

response = requests.post(PLANT\_ID\_URL, headers=headers, files=files)

if response.status\_code == 200:

data = response.json()

if data.get('suggestions'):

plant\_name = data['suggestions'][0]['plant\_name']

plant\_health = "Healthy"

return plant\_name, plant\_health

else:

return "Unknown", "Unable to identify"

else:

return "Error", "Failed to identify plant"

#### app.py

import json

import time

import adafruit\_dht

import board

from gpiozero import LED, Servo, Device,DigitalInputDevice

from time import sleep

from gpiozero.pins.pigpio import PiGPIOFactory

from nicegui import ui

import subprocess

from plant\_identification import identify\_plant

from mqtt\_client import connect\_mqtt, publish\_plant\_health

Device.pin\_factory = PiGPIOFactory()

MQTT\_TOPIC\_ACTION = 'home/control/actions'

servo = Servo(2)

dht\_sensor = adafruit\_dht.DHT11(board.D17, use\_pulseio=False)

soil\_sensor = DigitalInputDevice(21)

light = LED(26)

pump = LED(16)

fan = LED(20)

plant\_name = None

plant\_health = None

from mqtt\_client import client

connect\_mqtt()

async def update\_gauges():

temperature = dht\_sensor.temperature

humidity = dht\_sensor.humidity

if humidity is None and temperature is None:

ui.notify("Error reading temperature and humidity")

gauge1.value = temperature

gauge2.value = humidity

gauge3.value = soil\_sensor.is\_active

async def fanAction(value):

if value:

fan.on()

ui.notify("Fan turned on")

client.publish(MQTT\_TOPIC\_ACTION, json.dumps({'action': 'fan\_on'}))

else:

fan.off()

ui.notify("Fan turned off")

client.publish(MQTT\_TOPIC\_ACTION, json.dumps({'action': 'fan\_off'}))

async def windowAction(value):

if value:

servo.value = 0.1

time.sleep(0.7)

servo.value = 0

ui.notify("Window opened")

client.publish(MQTT\_TOPIC\_ACTION, json.dumps({'action': 'window\_open'}))

else:

servo.value = -0.1

time.sleep(0.7)

servo.value = 0

ui.notify("Window closed")

client.publish(MQTT\_TOPIC\_ACTION, json.dumps({'action': 'window\_close'}))

async def pumpAction(value):

if value:

pump.off()

ui.notify("Pump turned on")

client.publish(MQTT\_TOPIC\_ACTION, json.dumps({'action': 'pump\_on'}))

else:

pump.on()

ui.notify("Pump turned off")

client.publish(MQTT\_TOPIC\_ACTION, json.dumps({'action': 'pump\_off'}))

async def lightAction(value):

if value:

light.on()

ui.notify("Light turned on")

client.publish(MQTT\_TOPIC\_ACTION, json.dumps({'action': 'light\_on'}))

else:

light.off()

ui.notify("Light turned off")

client.publish(MQTT\_TOPIC\_ACTION, json.dumps({'action': 'light\_off'}))

async def update\_image():

global plant\_name, plant\_health

filename = 'static/captured\_image.jpg'

command = f"libcamera-still -q 80 -o {filename}"

subprocess.run(command, shell=True, stdout=subprocess.DEVNULL, stderr=subprocess.DEVNULL, check=True)

img.source = filename

plant\_name, plant\_health = identify\_plant(filename)

if plant\_name:

publish\_plant\_health(plant\_name, plant\_health)

plant\_label.text = f"Plant Name: {plant\_name}"

health\_label.text = f"Plant Health: {plant\_health}"

else:

ui.notify(f"Plant Identification Failed: {plant\_health}")

with ui.column().classes('w-full h-full justify-center items-center'):

with ui.row().classes('w-full justify-center'):

with ui.column().classes('items-center'):

gauge1 = ui.circular\_progress(min=0, max=50).props('color=yellow')

ui.label('Temperature')

with ui.column().classes('items-center'):

gauge2 = ui.circular\_progress(min=0, max=100)

ui.label('Humidity')

with ui.column().classes('items-center'):

gauge3 = ui.circular\_progress(min=0, max=100).props('color=green')

ui.label('Soil Humidity')

with ui.row().classes('w-full justify-center my-4'):

plant\_label = ui.label("Plant Name: Not Identified").classes('text-lg')

health\_label = ui.label("Plant Health: Unknown").classes('text-lg')

with ui.row().classes('w-full justify-center my-20'):

fanSwitch = ui.switch('Fan').on\_value\_change(lambda e: fanAction(e.value))

windowSwitch = ui.switch('Window').on\_value\_change(lambda e: windowAction(e.value))

pumpSwitch = ui.switch('Pump', on\_change=lambda e: pumpAction(e.value))

lightSwitch = ui.switch('Light', on\_change=lambda e: lightAction(e.value))

img = ui.image("").classes('w-96 h-auto')

ui.timer(interval=1000, callback=update\_gauges)

ui.timer(interval=5000, callback=update\_image)

ui.run(port=80)

#### mqtt\_client.py

import paho.mqtt.client as mqtt

import json

MQTT\_BROKER = "broker.hivemq.com"

MQTT\_PORT = 1883

MQTT\_TOPIC\_ACTION = "plant/actions"

MQTT\_TOPIC\_STATUS = "plant/health"

client = mqtt.Client()

def on\_publish(client, userdata, mid):

print(f"Message Published: {mid}")

client.on\_publish = on\_publish

def on\_message(client, userdata, msg):

try:

payload = json.loads(msg.payload)

action = payload.get('action')

if action == "fan\_on":

print("Fan turned on")

elif action == "fan\_off":

print("Fan turned off")

elif action == "window\_open":

print("Window opened")

elif action == "window\_close":

print("Window closed")

elif action == "pump\_on":

print("Pump turned on")

elif action == "pump\_off":

print("Pump turned off")

elif action == "light\_on":

print("Light turned on")

elif action == "light\_off":

print("Light turned off")

else:

print("Unknown action")

except Exception as e:

print(f"Error processing message: {e}")

client.on\_message = on\_message

def connect\_mqtt():

client.connect(MQTT\_BROKER, MQTT\_PORT)

client.subscribe(MQTT\_TOPIC\_ACTION)

client.loop\_start()

def publish\_plant\_health(plant\_name, health\_description):

payload = json.dumps({

'plant\_name': plant\_name,

'health': health\_description

})

client.publish(MQTT\_TOPIC\_STATUS, payload)