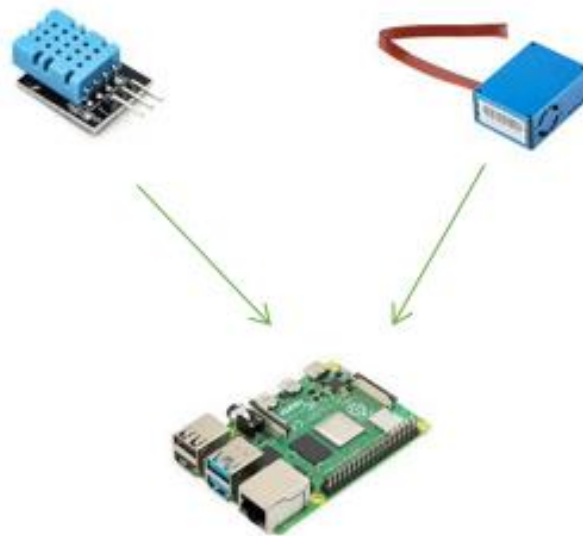


ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ  
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ  
ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

«Μέτρηση ποιότητας αέρα σε πραγματικό χρόνο σε κλειστό  
χώρο»



**Φοιτητής**

ΑΝΤΩΝΙΟΣ ΚΥΡΙΑΚΟΥ

513098

**Επιβλέπων**

Δρ. Κυριάκος Τσιακμάκης

## Φεβρουάριος 2024

Μέτρηση ποιότητας αέρα σε πραγματικό χρόνο σε κλειστό χώρο

Κωδικός: 23202

Φοιτητής: Αντώνιος Κυριακού

Εισηγητής: Δρ Κυριάκος Τσιακμάκης

Ημερομηνία ανάληψης Π.Ε. 30-03-2023

Ημερομηνία περάτωσης Π.Ε. 10-09-2023

*Βεβαιώνω ότι είμαι ο συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω καταγράψει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών, εικόνων και κειμένου, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επιπλέον, βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά, ειδικά ως πτυχιακή εργασία, στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε.*

*Η παρούσα εργασία αποτελεί πνευματική ιδιοκτησία του φοιτητή Αντώνιου Κυριακού που την εκπόνησε. Στο πλαίσιο της πολιτικής ανοικτής πρόσβασης, ο συγγραφέας/δημιουργός εκχωρεί στο Διεθνές Πανεπιστήμιο της Ελλάδος άδεια χρήσης του δικαιώματος αναπαραγωγής, δανεισμού, παρουσίασης στο κοινό και ψηφιακής διάχυσης της εργασίας διεθνώς, σε ηλεκτρονική μορφή και σε οποιοδήποτε μέσο, για διδακτικούς και ερευνητικούς σκοπούς, άνευ ανταλλάγματος. Η ανοικτή πρόσβαση στο πλήρες κείμενο της εργασίας, δεν σημαίνει καθ' οιονδήποτε τρόπο παραχώρηση δικαιωμάτων διανοητικής ιδιοκτησίας του συγγραφέα/δημιουργού, ούτε επιτρέπει την αναπαραγωγή, αναδημοσίευση, αντιγραφή, πώληση, εμπορική χρήση, διανομή, έκδοση, μεταφόρτωση (downloading), ανάρτηση (uploading), μετάφραση, τροποποίηση με οποιονδήποτε τρόπο, τμηματικά ή περιληπτικά της εργασίας, χωρίς τη ρητή προηγούμενη έγγραφη συναίνεση του συγγραφέα/δημιουργού.*

Η έγκριση της πτυχιακής εργασίας από το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνούς Πανεπιστημίου της Ελλάδος, δεν υποδηλώνει απαραίτητως και αποδοχή των απόψεων του συγγραφέα, εκ μέρους του Τμήματος.



## Περίληψη

Η εργασία αυτή αφορά σύστημα μέτρησης ποιότητας αέρα εσωτερικού χώρου σε πραγματικό χρόνο. Κεντρικό στοιχείο αυτής της εργασίας είναι το Raspberry Pi 3 και οι αισθητήρες όπως οι PMS5003 και DHT11, οι οποίοι μαζί επιτρέπουν ακριβή παρακολούθηση και ανάλυση της ποιότητας του αέρα, της θερμοκρασίας και της υγρασίας. Αυτό το σύστημα είναι σχολαστικά σχεδιασμένο για να διευκολύνει την απρόσκοπτη απόκτηση, μετάδοση και αποθήκευση δεδομένων, διασφαλίζοντας ότι οι πληροφορίες δεν είναι μόνο εύκολα ανακτήσιμες αλλά και εύκολα εμφανίσιμες σε μια διαδραστική διαδικτυακή πλατφόρμα. Αυτή η προσβασιμότητα καθιστά το σύστημα απαραίτητο στοιχείο τόσο για εξουσιοδοτημένους όσο και για μη εξουσιοδοτημένους χρήστες, καλύπτοντας ένα ευρύ φάσμα αναγκών παρακολούθησης.

## « Real-time indoor air quality measurement »

### **Abstract**

This work is about a real-time indoor air quality measurement system. The main of this work is the Raspberry Pi 3 and sensors such as the PMS5003 and DHT11, which together allow accurate monitoring and analysis of air quality, temperature and humidity. This system is designed to facilitate seamless acquisition, transmission and storage of data, ensuring that information is not only easily retrievable but also easily viewable on an interactive online platform. This accessibility makes the system a must-have for both authorized and unauthorized users, covering a wide range of monitoring needs.

## **Ευχαριστίες**

Εκφράζω τις θερμές μου ευχαριστίες στους γονείς μου για την αμέριστη υποστήριξή τους και στον επιβλέποντα μου κ. Τσιακμάκη Κυριάκο για τη συνεπή εκπαιδευτική και την οξυδερκή επιστημονική καθοδήγηση και την πολύτιμη συμβολή του στον κώδικα της εφαρμογής.

# Περιεχόμενα

Περίληψη .....	iv
Abstract .....	v
Ευχαριστίες .....	vi
Περιεχόμενα.....	vii
Κατάλογος Σχημάτων .....	viii
Κεφάλαιο 1ο: Εισαγωγή.....	9
1.1 Εισαγωγή.....	9
1.2 Δομή της εργασίας .....	12
Κεφάλαιο 2ο: Συσκευές και αισθητήρες.....	13
2.1 Raspberry Pi .....	13
2.2 Αισθητήρας θερμοκρασίας και υγρασίας DHT11.....	16
2.3 Αισθητήρας μέτρησης σωματιδίων PMS5003 .....	19
2.4 Αποστολή δεδομένων σε server .....	24
Κεφάλαιο 3ο: Τεχνολογία και προγραμματισμός .....	27
3.1 Python .....	27
3.2 Flask .....	30
3.3 MySQL.....	32
3.4 Javascript.....	35
3.4.1 Chart.js .....	37
Κεφάλαιο 4ο: Το σύστημα μέτρησης ποιότητας αέρα σε πραγματικό χρόνο σε κλειστό χώρο .41	
4.1 Εισαγωγή – Η διαδικασία .....	41
4.2 Η εφαρμογή - ιστοσελίδα.....	47
4.3 Η Βάση.....	50
4.4 Ασφάλεια πρόσβασης στο σύστημα και στα δεδομένα .....	52
Κεφάλαιο 5ο: Συμπεράσματα και προτάσεις βελτίωσης .....	53
ΒΙΒΛΙΟΓΡΑΦΙΑ .....	54
ΠΑΡΑΡΤΗΜΑ Α.....	55

## Κατάλογος Σχημάτων

Εικόνα 2.1: Raspberry pi 3 – Αναπτυξιακό και pins [8] .....	14
Εικόνα 2.2: Αισθητήρας DTH11 [9] .....	17
Εικόνα 2.3: Αισθητήρας PMS5003 [10] .....	20
Εικόνα 3.1: Python χρήσεις-εφαρμογές [11] .....	30
Εικόνα 3.2: Παράδειγμα Chart.js [16] .....	38
Εικόνα 4.1: Σύνδεση Raspberry με τους αισθητήρες.....	41
Εικόνα 4.2: Το σύστημα .....	43
Εικόνα 4.3: Το σύστημα – μέτρηση και αποστολή .....	44
Εικόνα 4.4: Το σύστημα – προβολή .....	45
Εικόνα 4.5: Η σελίδα για της σύνδεση .....	47
Εικόνα 4.6: Προβολή στοιχείων χρήστη και δωματίου .....	47
Εικόνα 4.7: Γράφημα με μετρήσεις Θερμοκρασίας.....	48
Εικόνα 4.8: Γράφημα με μετρήσεις Υγρασίας.....	48
Εικόνα 4.9: Γράφημα με μετρήσεις σωματιδίων PM1.0.....	49
Εικόνα 4.10: Γράφημα με μετρήσεις σωματιδίων PM2.5.....	49
Εικόνα 4.11: Γράφημα με μετρήσεις σωματιδίων PM10.....	50
Εικόνα 4.12: Η βάση monitorairdb με τους πίνακες.....	50
Εικόνα 4.13: Η δομή του πίνακα User.....	51
Εικόνα 4.14: Τα περιεχόμενα του πίνακα User.....	51
Εικόνα 4.15: Η δομή του πίνακα Sensor .....	51
Εικόνα 4.16: Τα περιεχόμενα του πίνακα Sensor .....	51
Εικόνα 4.17: Η δομή του πίνακα Room .....	52
Εικόνα 4.18: Τα περιεχόμενα του πίνακα Room .....	52

# Κεφάλαιο 1ο: Εισαγωγή

## 1.1 Εισαγωγή

Το Διαδίκτυο των Πραγμάτων (IoT) έχει φέρει επανάσταση στον τρόπο με τον οποίο αντιλαμβανόμαστε και αλληλεπιδρούμε με το περιβάλλον μας, ιδιαίτερα μέσω της μέτρησης, της αποθήκευσης και της εμφάνισης δεδομένων που λαμβάνονται από διάφορους αισθητήρες. Στην καρδιά αυτού του τεχνολογικού άλματος βρίσκεται η ικανότητα παρακολούθησης και ελέγχου συσκευών εξ αποστάσεως, οδηγώντας σε σημαντικές προόδους σε τομείς όπως η μέτρηση της ποιότητας του αέρα εσωτερικών χώρων σε πραγματικό χρόνο [1]. Οι αισθητήρες εξοπλισμένοι με δυνατότητες IoT μπορούν να συλλέγουν συνεχώς δεδομένα για διάφορες περιβαλλοντικές παραμέτρους, διασφαλίζοντας ότι η ποιότητα του αέρα στους εσωτερικούς χώρους διατηρείται στα βέλτιστα επίπεδα. Αυτοί οι αισθητήρες είναι ικανοί να ανιχνεύουν ένα ευρύ φάσμα ρύπων, συμπεριλαμβανομένων των πτητικών οργανικών ενώσεων, του μονοξειδίου του άνθρακα (CO) και άλλων επιβλαβών ουσιών. Τα δεδομένα που συλλέγονται δεν είναι μόνο ενδεικτικά της ποιότητας του αέρα, αλλά, όταν αναλύονται με την πάροδο του χρόνου, μπορούν να αποκαλύψουν τάσεις και μοτίβα που είναι καθοριστικά για τη λήψη τεκμηριωμένων αποφάσεων σχετικά με τις στρατηγικές αερισμού και καθαρισμού του αέρα. Επιπλέον, η όψη σε πραγματικό χρόνο διασφαλίζει την άμεση απόκριση σε οποιαδήποτε διαπιστωθείσα υποβάθμιση της ποιότητας του αέρα, διασφαλίζοντας έτσι την υγεία και την ευημερία των επιβατών.

Μια άλλη κρίσιμη πτυχή του IoT στην περιβαλλοντική παρακολούθηση είναι η μέτρηση των σωματιδίων (PM), ιδιαίτερα των PM<sub>2,5</sub> και PM<sub>10</sub>. Αυτά τα λεπτά σωματίδια, που συχνά προέρχονται από διάφορες πηγές, όπως βιομηχανικές εκπομπές, καυσαέρια οχημάτων ή ακόμα και οικιακές δραστηριότητες, ενέχουν σημαντικούς κινδύνους για την υγεία. Οι αισθητήρες με δυνατότητα IoT μπορούν να παρακολουθούν συνεχώς τη συγκέντρωση αυτών των σωματιδίων, παρέχοντας μια σαφή εικόνα της ποιότητας του αέρα και των πιθανών κινδύνων για την υγεία που συνδέονται με αυτήν. Τα δεδομένα που συλλέγονται από αυτούς τους αισθητήρες δεν είναι μόνο ζωτικής σημασίας για άμεσες προειδοποιήσεις για την υγεία, αλλά χρησιμεύουν επίσης ως κρίσιμη πηγή για την έρευνα και τη χάραξη πολιτικής, συμβάλλοντας στη διαμόρφωση στρατηγικών με στόχο τον έλεγχο της ρύπανσης και την προστασία του περιβάλλοντος. Η ικανότητα αυτών των συστημάτων να προσφέρουν δεδομένα σε πραγματικό χρόνο διασφαλίζει ότι κάθε ξαφνική αύξηση των σωματιδίων μπορεί να αντιμετωπιστεί γρήγορα, με μέτρα μετριασμού που εφαρμόζονται για την προστασία της υγείας των ατόμων, ιδιαίτερα σε αστικές και βιομηχανικές συνθήκες.

Η μέτρηση θερμοκρασίας και υγρασίας είναι ένας άλλος τομέας όπου οι αισθητήρες IoT διαδραματίζουν κεντρικό ρόλο. Αυτές οι παράμετροι δεν είναι απλώς παράγοντες άνεσης, αλλά επηρεάζουν σημαντικά την ποιότητα του εσωτερικού αέρα, την κατανάλωση ενέργειας και τη μακροζωία του εξοπλισμού και των κατασκευών μέσα σε ένα κτίριο. Οι αισθητήρες IoT παρέχουν ακριβή και συνεχή παρακολούθηση των επιπέδων θερμοκρασίας και υγρασίας, επιτρέποντας στα αυτοματοποιημένα συστήματα ελέγχου του κλίματος να διατηρούν τις βέλτιστες συνθήκες. Αυτό όχι μόνο ενισχύει την άνεση και την παραγωγικότητα των επιβατών, αλλά συμβάλλει επίσης στην ενεργειακή απόδοση διασφαλίζοντας ότι τα συστήματα θέρμανσης, εξαερισμού και κλιματισμού (HVAC) λειτουργούν μόνο όταν είναι απαραίτητο και σε βέλτιστα επίπεδα. Επιπλέον, τα δεδομένα που συλλέγονται μπορούν να χρησιμοποιηθούν για προγνωστική συντήρηση, εντοπίζοντας πιθανά προβλήματα στα συστήματα HVAC προτού κλιμακωθούν σε δαπανηρές επισκευές ή αντικαταστάσεις.

Η αποθήκευση και η επεξεργασία των τεράστιων ποσοτήτων δεδομένων που συλλέγονται από αισθητήρες IoT είναι ζωτικής σημασίας για την απόκτηση πρακτικών πληροφοριών. Οι προηγμένες αναλύσεις δεδομένων και οι αλγόριθμοι μηχανικής μάθησης μπορούν να επεξεργαστούν αυτά τα δεδομένα για τον εντοπισμό μοτίβων, την πρόβλεψη τάσεων και ακόμη και την αυτοματοποίηση των διαδικασιών λήψης αποφάσεων. Για παράδειγμα, ένα σύστημα παρακολούθησης της ποιότητας του αέρα με δυνατότητα IoT μπορεί να αναλύσει ιστορικά δεδομένα και δεδομένα σε πραγματικό χρόνο για να προβλέψει πότε θα επιδεινωθεί η ποιότητα του αέρα και να προσαρμόσει αυτόματα τα συστήματα εξαερισμού για την προληπτική αντιμετώπιση του ζητήματος. Επιπλέον, η αποθήκευση αυτών των δεδομένων, συχνά σε πλατφόρμες που βασίζονται σε σύννεφο, διασφαλίζει ότι είναι προσβάσιμα από οπουδήποτε, διευκολύνοντας την απομακρυσμένη παρακολούθηση και έλεγχο. Αυτό όχι μόνο ενισχύει την αποτελεσματικότητα των συστημάτων διαχείρισης κτιρίων, αλλά επιτρέπει επίσης την ενσωμάτωση δεδομένων από πολλαπλές πηγές, οδηγώντας σε μια πιο ολοκληρωμένη κατανόηση των περιβαλλοντικών συνθηκών.

Η οπτικοποίηση και η εμφάνιση των δεδομένων που συλλέγονται από αισθητήρες IoT είναι ζωτικής σημασίας για την αλληλεπίδραση και τη λήψη αποφάσεων με τους χρήστες. Οι φιλικόι προς τον χρήστη πίνακες εργαλείων και οι εφαρμογές για κινητά που βρίσκονται στα άτομα να παρακολουθούν τις περιβαλλοντικές συνθήκες του περιβάλλοντός τους σε πραγματικό χρόνο. Αυτές οι διεπαφές παρέχουν συχνά ειδοποιήσεις και συστάσεις, δίνοντας τη δυνατότητα στα άτομα να αναλάβουν άμεση δράση όταν είναι απαραίτητο. Για τις επιχειρήσεις και τους οργανισμούς, αυτό το επίπεδο αλληλεπίδρασης διευκολύνει τη διαχείριση μεγάλων εγκαταστάσεων, διασφαλίζοντας ότι οι περιβαλλοντικές συνθήκες δεν διατηρούνται μόνο για άνεση και για συμμόρφωση με τους κανονισμούς υγείας και ασφάλειας. Επιπλέον, η οπτικοποίηση αυτών των δεδομένων σε μια εύκολα κατανοητή μορφή είναι το κλειδί για τη συμμετοχή των ενδιαφερομένων, την προώθηση της ευαισθητοποίησης και την ώθηση της αλλαγής συμπεριφοράς προς τις πιο βιώσιμες και ευαισθητοποιημένες για την υγεία πρακτικές.

Η ενσωμάτωση του IoT στη μέτρηση, την αποθήκευση και την εμφάνιση δεδομένων από περιβαλλοντικούς αισθητήρες αντιπροσωπεύει ένα σημαντικό βήμα προς τη δημιουργία εξυπνότερων, πιο υγιεινών και πιο βιώσιμων περιβαλλόντων διαβίωσης και εργασίας. Από την παρακολούθηση της ποιότητας του αέρα σε πραγματικό χρόνο έως την αποτελεσματική διαχείριση του εσωτερικού κλίματος, το IoT βρίσκεται στην πρώτη γραμμή της περιβαλλοντικής διαχείρισης και της τεχνολογικής καινοτομίας, ανοίγοντας το δρόμο για ένα μέλλον όπου η αλληλοεπίδρασή μας με το περιβάλλον είναι ενημερωμένη, ανταποκρινόμενη και υπεύθυνη.

Το σύστημα μέτρησης ποιότητας αέρα εσωτερικού χώρου σε πραγματικό χρόνο αποτελεί σημαντική τεχνολογική καινοτομία, αξιοποιώντας τις δυνατότητες του Raspberry Pi 3 μαζί με εξελιγμένους αισθητήρες όπως ο PMS5003 και ο DHT11. Αυτό το σύστημα είναι σχολαστικά κατασκευασμένο για να παρέχει ακριβή παρακολούθηση και αναλυτικές πληροφορίες για την ποιότητα του αέρα, μαζί με κρίσιμες περιβαλλοντικές παραμέτρους όπως η θερμοκρασία και η υγρασία. Αυτό όχι μόνο χρησιμεύει ως εργαλείο για τη διατήρηση μιας υγιεινής και ευχάριστης ατμόσφαιρας, αλλά επίσης ενσωματώνει απρόσκοπτα τη συλλογή, τη μετάδοση και την αποθήκευση δεδομένων, όπως απεικονίζεται στο σχηματικό σχήμα του συστήματος. Η επιδεξιότητά του στη διευκόλυνση της εύκολης πρόσβασης και προβολής δεδομένων μέσω μιας διαδικτυακής πλατφόρμας το καθιστά εξαιρετικά ωφέλιμο, εξυπηρετώντας τόσο τις εξουσιοδοτημένες όσο και τις μη εξουσιοδοτημένες ομάδες χρηστών [2-6].

Κεντρική θέση σε αυτό το σύστημα είναι η τεχνολογία αιχμής των αισθητήρων και η προσαρμοσμένη φύση της επεξεργασίας δεδομένων και του αναλυτικού πλαισίου. Η συγχώνευση του Raspberry Pi 3 με αισθητήρες όπως ο PMS5003 και ο DHT11 δίνει τη δυνατότητα στο σύστημα να εκτελεί ακριβείς μετρήσεις και να διατηρεί ένα σταθερό αρχείο βασικών περιβαλλοντικών μετρήσεων. Αυτοί οι αισθητήρες είναι στρατηγικά τοποθετημένοι για να διασφαλίζουν τη συνεχή παρακολούθηση και να διασφαλίζουν τη συγκέντρωση περιεκτικών και ακριβών συνόλων δεδομένων. Μετά τη συλλογή δεδομένων, οι πληροφορίες μεταδίδονται με ασφάλεια σε έναν κεντρικό διακομιστή. Εδώ, υφίσταται μια σειρά από βήματα επεξεργασίας, καθαρίζοντας και δομώντας τα ακατέργαστα δεδομένα σε μια μορφή που είναι κατάλληλη για ανάλυση και ερμηνεία. Η υποδομή του συστήματος ενσωματώνει επίσης αυστηρούς μηχανισμούς ελέγχου πρόσβασης, διασφαλίζοντας την προσβασιμότητα των δεδομένων αποκλειστικά σε εξουσιοδοτημένο προσωπικό και τηρώντας τα μέγιστα πρότυπα ασφάλειας και εμπιστευτικότητας δεδομένων.

Η διεπαφή χρήστη χρησιμεύει ως η κορυφή της αλυσίδας διεργασιών του συστήματος, όπου τα εκλεπτυσμένα δεδομένα παρουσιάζονται με σαφήνεια και ακρίβεια. Αυτή η διεπαφή, που υλοποιείται μέσω μιας σχολαστικά σχεδιασμένης διαδικτυακής εφαρμογής, καθιστά τα δεδομένα όχι μόνο προσβάσιμα αλλά και εύκολα στην ερμηνεία, επιτρέποντας στους χρήστες να αναλύουν τις τάσεις, να εντοπίζουν ανωμαλίες και να λαμβάνουν τεκμηριωμένες αποφάσεις. Ενισχυμένη με διαδραστικά γραφήματα και πίνακες ελέγχου, η διεπαφή αντικατοπτρίζει ζωντανά τις συνθήκες περιβάλλοντος και ιστορικού περιβάλλοντος σε πραγματικό χρόνο. Οι προσαρμοσμένες ρυθμίσεις και οι ειδοποιήσεις

προσαρμόζουν περαιτέρω την εμπειρία χρήστη. Αυτή η συνένωση βαθιάς ανάλυσης δεδομένων και απεικόνισης με επίκεντρο τον χρήστη καταλήγει σε μια ισχυρή και αποτελεσματική λύση. Ως εκ τούτου, το σύστημα μέτρησης της ποιότητας του αέρα όχι μόνο αποτελεί την επιτομή της τεχνολογικής πολυπλοκότητας, αλλά προσφέρει επίσης απτή προστιθέμενη αξία στους τελικούς χρήστες του, διασφαλίζοντας μια πλήρως επωφελής αλληλεπίδραση με τον χρήστη.

## **1.2 Δομή της εργασίας**

Στο πρώτο κεφάλαιο παρουσιάζεται η εισαγωγή της εργασίας.

Στο δεύτερο κεφάλαιο παρουσιάζονται οι συσκευές και οι αισθητήρες που χρησιμοποιήθηκαν.

Στο τρίτο κεφάλαιο περιγράφεται η τεχνολογία και οι γλώσσες προγραμματισμού που χρησιμοποιήθηκαν και τα βοηθητικά εργαλεία που ενσωματώνονται σε αυτό.

Στο τέταρτο κεφάλαιο αναλύεται το σύστημα μέτρησης ποιότητας αέρα σε πραγματικό χρόνο σε κλειστό χώρο.

Στο πέμπτο κεφάλαιο παρουσιάζονται τα συμπεράσματα της εργασίας και θέματα για βελτίως.

Στο τέλος της εργασίας παρατίθεται το παράρτημα με κάποιους από τους ρυθμον κώδικες που χρησιμοποιήθηκαν στην εργασία.

## Κεφάλαιο 2ο: Συσκευές και αισθητήρες

### 2.1 Raspberry Pi

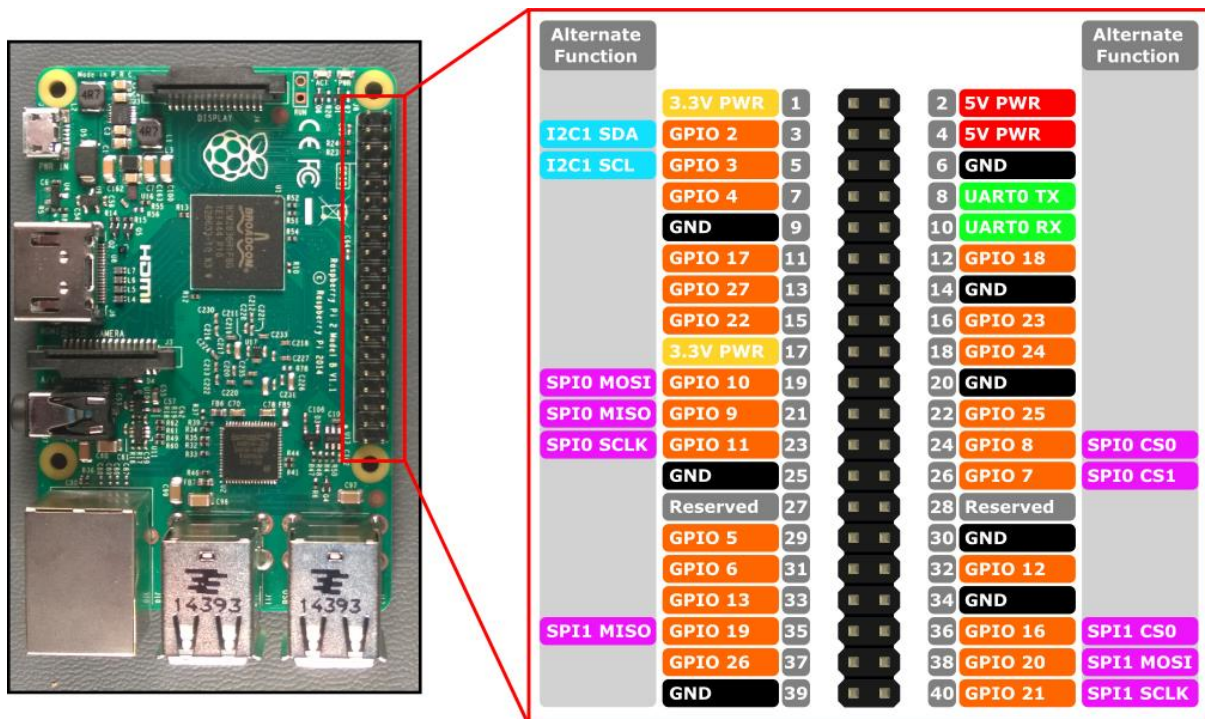
Το Raspberry Pi 3 είναι μια αξιοσημείωτη απόδειξη της εξέλιξης των συμπαγών υπολογιστών, παρουσιάζοντας τον εαυτό του ως μια ισχυρή, ευέλικτη και οικονομικά αποδοτική λύση για μυριάδες εφαρμογές [7]. Από εκπαιδευτικούς και χομπίστες έως επαγγελματίες προγραμματιστές, το Raspberry Pi 3 απευθύνεται σε ένα ευρύ κοινό, προσφέροντας μια πλατφόρμα για μάθηση, πειραματισμό και ανάπτυξη διαφόρων έργων. Αποτελεί την τρίτη γενιά της σειράς Raspberry Pi, εισάγοντας βελτιωμένη επεξεργαστική ισχύ, βελτιωμένες επιλογές συνδεσιμότητας και μεγαλύτερη συμβατότητα με περιφερειακά σε σύγκριση με τους προκατόχους του. Αυτή η μικροσκοπική αλλά πανίσχυρη συσκευή έχει συμβάλει σημαντικά στην ευρεία χρήση των υπολογιστών, καθιστώντας την τεχνολογία πιο προσιτή και ενισχύοντας την καινοτομία σε πολλούς τομείς.

Στον πυρήνα του Raspberry Pi 3 βρίσκεται η ισχυρή μονάδα επεξεργασίας του. Εξοπλισμένο με τετραπύρηνο επεξεργαστή ARM Cortex-A53 1,2 GHz 64 bit, το Raspberry Pi 3 προσφέρει σημαντική ενίσχυση της απόδοσης, διασφαλίζοντας την ομαλή και αποτελεσματική λειτουργία των εφαρμογών. Αυτό το άλμα στην επεξεργαστική ισχύ ανοίγει νέες δυνατότητες για πιο σύνθετα και απαιτητικά έργα, που κυμαίνονται από κέντρα πολυμέσων και κονσόλες παιχνιδιών έως εξελιγμένες συσκευές οικιακού αυτοματισμού και Internet of Things (IoT). Η αυξημένη υπολογιστική ικανότητα επιτρέπει στους χρήστες να εμβαθύνουν σε προηγμένα έργα προγραμματισμού, επεξεργασίας δεδομένων, ακόμη και μηχανικής μάθησης, καθιστώντας το ιδανικό εκπαιδευτικό εργαλείο για μαθητές και λάτρεις που θέλουν να εξερευνήσουν τη σφαίρα της επιστήμης των υπολογιστών και των ηλεκτρονικών.

Η συνδεσιμότητα είναι ένα άλλο χαρακτηριστικό του Raspberry Pi 3, που το ξεχωρίζει ως μια εξαιρετικά ευέλικτη και φιλική προς το χρήστη συσκευή. Με ενσωματωμένη υποστήριξη WiFi και Bluetooth, το Raspberry Pi 3 απλοποιεί τις ασύρματες συνδέσεις, επιτρέποντας την απρόσκοπτη ενσωμάτωση σε διάφορες ρυθμίσεις δικτύου και την εύκολη επικοινωνία με άλλες συσκευές. Αυτή η δυνατότητα εξαλείφει την ανάγκη για πρόσθετα dongles ή αξεσουάρ, μειώνοντας την ακαταστασία και απλοποιώντας την ανάπτυξη του έργου. Οι βελτιωμένες επιλογές συνδεσιμότητας επιτρέπουν στους χρήστες να δημιουργούν συνδεδεμένες συσκευές χωρίς κόπο, από διακομιστές πολυμέσων ροής έως gadgets IoT που μπορούν να αλληλεπιδράσουν με άλλες έξυπνες συσκευές, εμπλουτίζοντας το οικοσύστημα των εφαρμογών οικιακού αυτοματισμού και απομακρυσμένης παρακολούθησης.

Το Raspberry Pi 3 διαθέτει επίσης μια σειρά επιλογών εισόδου/εξόδου, προσφέροντας αξιοσημείωτη ευελιξία στη σύνδεση μιας μεγάλης σειράς περιφερειακών και αξεσουάρ. Με μια θύρα HDMI πλήρους μεγέθους, τέσσερις θύρες USB, μια κεφαλίδα GPIO 40 ακίδων και πολλά άλλα, οι χρήστες μπορούν εύκολα να συνδέσουν οθόνες, συσκευές αποθήκευσης, αισθητήρες και άλλα στοιχεία υλικού. Αυτή η ευελιξία διευκολύνει τη δημιουργία διαφορετικών έργων, είτε πρόκειται για την κατασκευή ενός

προσωπικού υπολογιστή, τη δημιουργία ενός μετεωρολογικού σταθμού ή τη δημιουργία μιας διαδραστικής καλλιτεχνικής εγκατάστασης. Οι άφθονες επιλογές I/O ενθαρρύνουν επίσης τον πειραματισμό και τη μάθηση, επιτρέποντας στους χρήστες να διασυνδέονται με διάφορα ηλεκτρονικά εξαρτήματα και να αποκτούν πρακτική εμπειρία στην ενοποίηση υλικού και λογισμικού.



Εικόνα 2.1: Raspberry pi 3 – Αναπτυξιακό και pins [8]

Παρά το συμπαγές του μέγεθος, το Raspberry Pi 3 δεν θέτει σε κίνδυνο τη μνήμη και την αποθήκευση. Με 1 GB μνήμης RAM και υποστήριξη για κάρτες microSD για αποθήκευση, χειρίζεται αποτελεσματικά λειτουργικά συστήματα, εφαρμογές και δεδομένα. Αυτό το καθιστά κατάλληλο για εφαρμογές που απαιτούν αξιοπρεπή ποσότητα μνήμης και χωρητικότητα αποθήκευσης, όπως διακομιστές πολυμέσων, διακομιστές αρχείων και ελαφροί διακομιστές ιστού. Η δυνατότητα χρήσης καρτών microSD προσφέρει επίσης ευελιξία όσον αφορά τη χωρητικότητα αποθήκευσης και παρέχει έναν απλό τρόπο ανταλλαγής λειτουργικών συστημάτων ή δημιουργίας αντιγράφων ασφαλείας δεδομένων.

Η δέσμευση του Raspberry Pi 3 στην εκπαίδευση και την προσβασιμότητα είναι εμφανής στον σχεδιασμό και την υποστήριξη της κοινότητας. Το Raspberry Pi Foundation, ο οργανισμός πίσω από αυτήν τη συσκευή, τονίζει την αποστολή του να βάλει τη δύναμη της ψηφιακής κατασκευής στα χέρια των ανθρώπων σε όλο τον κόσμο. Παρέχουν εκτενείς εκπαιδευτικούς πόρους, σεμινάρια και φόρουμ κοινότητας, ενισχύοντας ένα περιβάλλον όπου οι χρήστες όλων των ηλικιών και επιπέδων δεξιοτήτων

μπορούν να μάθουν, να μοιραστούν και να αναπτυχθούν. Τα σχολεία και τα εκπαιδευτικά ιδρύματα υιοθετούν ευρέως το Raspberry Pi 3, αξιοποιώντας τις δυνατότητές του για τη διδασκαλία προγραμματισμού, βασικών αρχών της επιστήμης των υπολογιστών και ψηφιακής κατασκευής.

Η επιρροή του Raspberry Pi 3 εκτείνεται πέρα από την εκπαίδευση στη σφαίρα των επαγγελματικών και βιομηχανικών εφαρμογών. Η αξιοπιστία του, σε συνδυασμό με τη χαμηλή κατανάλωση ενέργειας και ένα υποστηρικτικό οικοσύστημα, το καθιστά ελκυστική πλατφόρμα για ανάπτυξη εμπορικών προϊόντων και βιομηχανικό αυτοματισμό. Εταιρείες και νεοφυείς επιχειρήσεις χρησιμοποιούν το Raspberry Pi 3 για να δημιουργήσουν πρωτότυπα και ακόμη και να αναπτύξουν εμπορικά προϊόντα, επωφελούμενοι από τη σχέση κόστους-αποτελεσματικότητας και την πλούσια γκάμα συμβατού λογισμικού και υλικού. Η ισχυρή κοινότητα και η εκτεταμένη τεκμηρίωσή του μειώνουν επίσης τον χρόνο και το κόστος ανάπτυξης, καθιστώντας την τεχνολογία πιο προσιτή και την καινοτομία πιο εφικτή για επιχειρήσεις όλων των μεγεθών. Η ισχυρή του απόδοση, οι επιλογές συνδεσιμότητας, η ευελιξία στη διασύνδεση με περιφερειακά και η ισχυρή υποστήριξη για εκπαίδευση και επαγγελματική χρήση το καθιστούν ένα ανεκτίμητο εργαλείο στη σύγχρονη ψηφιακή εποχή. Είτε πρόκειται για διδασκαλία, είτε για ανάπτυξη εφαρμογών αιχμής, το Raspberry Pi 3 συνεχίζει να εμπνέει και να ενδυναμώνει δημιουργούς, μαθητές και καινοτόμους σε όλο τον κόσμο.

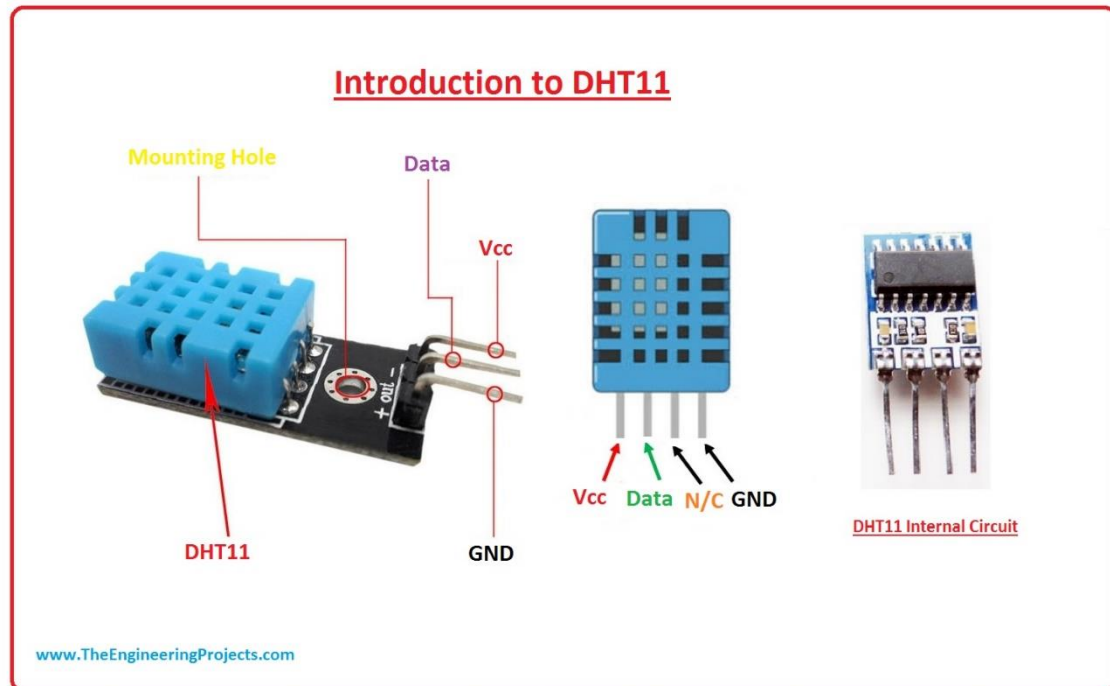
Το Raspberry Pi 3, γνωστό για το συμπαγές του μέγεθος και τις ισχυρές υπολογιστικές του δυνατότητες, αποτελεί ιδανική πλατφόρμα για ενσωμάτωση με αισθητήρες σε ένα ευρύ φάσμα εφαρμογών, από την παρακολούθηση του περιβάλλοντος έως τα έξυπνα οικιακά συστήματα. Η εγγενής ευελιξία του, σε συνδυασμό με τη σημαντική επεξεργαστική ισχύ, το καθιστά μια προτιμώμενη επιλογή τόσο για λάτρεις όσο και για επαγγελματίες που επιδιώκουν να αναπτύξουν καινοτόμες λύσεις IoT (Internet of Things). Το Raspberry Pi 3 είναι εξοπλισμένο με τετραπύρηνο επεξεργαστή και διαθέτει βελτιωμένες επιλογές συνδεσιμότητας, όπως Wi-Fi και Bluetooth, καθιστώντας το μια στιβαρή αυτόνομη συσκευή ικανή να χειρίζεται περίπλοκες υπολογιστικές εργασίες και να διευκολύνει την ασύρματη επικοινωνία. Αυτή η συνδεσιμότητα είναι ιδιαίτερα πλεονεκτική όταν οι αισθητήρες απαιτούν μετάδοση δεδομένων σε πραγματικό χρόνο ή όταν το σύστημα πρέπει να ενσωματωθεί σε μεγαλύτερα δίκτυα. Επιπλέον, οι εκτεταμένες ακίδες GPIO (Είσοδος/Εξόδος Γενικής Χρήσης) του Raspberry Pi παρέχουν μια άμεση διεπαφή για μια πληθώρα αισθητήρων, επιτρέποντας τη συλλογή και ανάλυση διαφόρων τύπων δεδομένων, όπως θερμοκρασία, υγρασία, κίνηση και ακόμη πιο περίπλοκες περιβαλλοντικές μεταβλητές.

Εκτός από την τεχνική του ικανότητα, το Raspberry Pi 3 είναι αναγνωρισμένο για τη μεγάλη κοινότητα και το πλούσιο οικοσύστημά του, το οποίο περιλαμβάνει μια πληθώρα από βιβλιοθήκες, σεμινάρια και φόρουμ. Αυτό το δίκτυο υποστήριξης μειώνει σημαντικά το εμπόδιο εισόδου για την υλοποίηση έργων που βασίζονται σε αισθητήρες, επιτρέποντας τόσο στους αρχάριους όσο και στους έμπειρους προγραμματιστές να πραγματοποιήσουν τις ιδέες τους γρήγορα. Η συμβατότητα του Raspberry Pi με μια σειρά γλωσσών προγραμματισμού, κυρίως με την Python, απλοποιεί περαιτέρω τη διαδικασία

ανάπτυξης. Η Python, με την απλή σύνταξη και τις εκτεταμένες βιβλιοθήκες της, είναι ιδιαίτερα κατάλληλη για το χειρισμό δεδομένων αισθητήρων, την εκτέλεση ανάλυσης δεδομένων και την ενοποίηση με υπηρεσίες web. Αυτή η συμβίωση μεταξύ του Raspberry Pi 3 και των αισθητήρων ανοίγει μια πληθώρα δυνατοτήτων, από τη δημιουργία έξυπνων συστημάτων περιβαλλοντικής παρακολούθησης που μπορούν να προβλέψουν τα καιρικά μοτίβα μέχρι την ανάπτυξη εξελιγμένων συστημάτων οικιακού αυτοματισμού που ενισχύουν την ενεργειακή απόδοση και την ασφάλεια. Κατά συνέπεια, ο συνδυασμός των υπολογιστικών δυνατοτήτων του Raspberry Pi 3, των επιλογών συνδεσιμότητας και της υποστήριξης μιας τεράστιας κοινότητας το καθιστά απaráμιλλη επιλογή για έργα που βασίζονται σε αισθητήρες, ανοίγοντας το δρόμο για καινοτόμες λύσεις σε έναν όλο και πιο διασυνδεδεμένο κόσμο.

## **2.2 Αισθητήρας θερμοκρασίας και υγρασίας DHT11**

Ο αισθητήρας DHT11, μια συμπαγής και ευέλικτη συσκευή, είναι ευρέως αναγνωρισμένος στη σφαίρα του IoT για την ικανότητά του να παρέχει ακριβείς μετρήσεις θερμοκρασίας και υγρασίας. Ο σχεδιασμός αυτού του αισθητήρα είναι ιδιαίτερα φιλικός προς το χρήστη, καθιστώντας τον ιδανική επιλογή τόσο για χομπίστες όσο και για επαγγελματίες που μπαίνουν στον κόσμο της περιβαλλοντικής παρακολούθησης και ελέγχου. Στο μικρό του πλαίσιο είναι ενσωματωμένος ένας χωρητικός αισθητήρας υγρασίας και ένα θερμίστορ, που λειτουργούν από κοινού για τη μέτρηση των συνθηκών του περιβάλλοντος αέρα. Το DHT11 ξεχωρίζει για την αξιοπιστία και την οικονομική του απόδοση, προσφέροντας μια πρακτική λύση για ένα ευρύ φάσμα εφαρμογών, που κυμαίνονται από απλά οικιακά έργα έως πιο σύνθετα βιομηχανικά συστήματα περιβαλλοντικής παρακολούθησης. Η ψηφιακή του έξοδος, η οποία στέλνει εύκολα δεδομένα σε σειριακή μορφή, καθιστά εξαιρετικά εύκολη τη διασύνδεση με μικροελεγκτές και υπολογιστές. Επιπλέον, το DHT11 λειτουργεί σε ένα μέτριο εύρος τροφοδοσίας, ενισχύοντας περαιτέρω την ελκυστικότητά του για έργα με μπαταρία ή χαμηλής κατανάλωσης.



Εικόνα 2.2: Αισθητήρας DHT11 [9]

Η λειτουργική ικανότητα του αισθητήρα DHT11 δεν περιορίζεται μόνο στις δυνατότητες μέτρησης του. λάμπει επίσης μέσω των πτυχών χειρισμού δεδομένων και ενσωμάτωσης. Όταν ενσωματώνεται σε ένα σύστημα IoT, το DHT11 μπορεί να μεταδώσει δεδομένα θερμοκρασίας και υγρασίας σε μια κεντρική μονάδα επεξεργασίας ή μια πλατφόρμα cloud, όπου μπορούν να αποθηκευτούν, να αναλυθούν και να οπτικοποιηθούν. Αυτή η απρόσκοπτη ενοποίηση επιτρέπει την παρακολούθηση και τη λήψη αποφάσεων σε πραγματικό χρόνο. Για παράδειγμα, σε μια εγκατάσταση έξυπνου σπιτιού, τα δεδομένα από το DHT11 μπορούν να χρησιμοποιηθούν για τον έλεγχο των συστημάτων HVAC, διασφαλίζοντας βέλτιστα επίπεδα άνεσης, ενώ παράλληλα μεγιστοποιούν την ενεργειακή απόδοση. Τα δεδομένα του αισθητήρα μπορούν επίσης να διαδραματίσουν κρίσιμο ρόλο σε γεωργικές ρυθμίσεις, όπου η διατήρηση των σωστών επιπέδων θερμοκρασίας και υγρασίας είναι ζωτικής σημασίας για την υγεία και την απόδοση των καλλιεργειών. Παρά τη σχετικά χαμηλή του ανάλυση και τον πιο αργό χρόνο απόκρισης σε σύγκριση με πιο προηγμένους αισθητήρες, το DHT11 παρέχει μια ισορροπημένη λύση για έργα όπου το κόστος, η ευκολία χρήσης και η μέτρια ακρίβεια αποτελούν βασικά ζητήματα. Η στιβαρότητα και η απλή διασύνδεσή του, σε συνδυασμό με την ικανότητα λειτουργίας κάτω από ένα ευρύ φάσμα περιβαλλοντικών συνθηκών, καθιστούν το DHT11 βασικό στοιχείο στην εργαλειοθήκη όσων χτίζουν τα διασυνδεδεμένα, ανταποκρινόμενα περιβάλλοντα του μέλλοντος.

Ο αισθητήρας DHT11, σε συνδυασμό με το Raspberry Pi, αποτελούν ένα ισχυρό δίδυμο για την παρακολούθηση των περιβαλλοντικών συνθηκών όπως η θερμοκρασία και η υγρασία. Ο DHT11 είναι ένας βασικός, εξαιρετικά χαμηλού κόστους ψηφιακός αισθητήρας θερμοκρασίας και υγρασίας.

Χρησιμοποιεί έναν χωρητικό αισθητήρα υγρασίας και ένα θερμίστορ για τη μέτρηση του περιβάλλοντος αέρα και εξάγει ένα ψηφιακό σήμα στην ακίδα δεδομένων. Η απλότητα και η στιβαρότητά του το καθιστούν κατάλληλο για χομπίστες και απλές εφαρμογές. Όταν διασυνδέεται με το Raspberry Pi, έναν συμπαγή αλλά ισχυρό υπολογιστή ικανό να εκτελεί πλήρεις διανομές Linux, το DHT11 μπορεί να χρησιμοποιηθεί για τη συλλογή περιβαλλοντικών δεδομένων, τα οποία στη συνέχεια μπορούν να υποβληθούν σε επεξεργασία, ανάλυση και ακόμη και αποστολή μέσω Διαδικτύου.

Για την χρήση του θα χρειαστείτε έναν αισθητήρα DHT11, ένα Raspberry Pi και μερικά καλώδια jumper. Το πρώτο βήμα περιλαμβάνει τη σύνδεση του αισθητήρα DHT11 με το Raspberry Pi. Συνήθως, ο αισθητήρας έχει τρεις ή τέσσερις ακίδες: VCC (ισχύς), Δεδομένα (επικοινωνία δεδομένων) και GND (γείωση). Εάν έχει μια τέταρτη ακίδα, συνήθως μένει ασύνδετη καθώς είναι μια εναλλακτική καρφίτσα δεδομένων. Συνδέστε την ακίδα VCC σε μια ακίδα 5 V ή 3,3 V στο Raspberry Pi, την ακίδα GND σε μια ακίδα γείωσης στο Raspberry Pi και την καρφίτσα δεδομένων σε οποιαδήποτε ακίδα GPIO (είσοδος/έξοδος γενικής χρήσης) στο Raspberry Pi.

Στη συνέχεια, μπορείτε να χρησιμοποιήσετε μια βιβλιοθήκη Python όπως το Adafruit\_DHT για διασύνδεση με τον αισθητήρα. Ακολουθεί ένα παράδειγμα αποσπάσματος κώδικα που διαβάζει τη θερμοκρασία και την υγρασία από τον αισθητήρα DHT11 και εκτυπώνει τα αποτελέσματα στην κονσόλα:

```
import Adafruit_DHT

import time

# Sensor type - DHT11
sensor = Adafruit_DHT.DHT11

# The GPIO pin number where Data is connected
pin = 4

while True:

    # Reading from the DHT11 sensor
    humidity, temperature = Adafruit_DHT.read_retry(sensor, pin)
```

```
if humidity is not None and temperature is not None:

    # Printing the results

    print(f"Temperature={temperature}°C, Humidity={humidity}%")

else:

    print('Failed to get reading from the sensor. Try again!')

# Wait for 2 seconds before the next read

time.sleep(2)
```

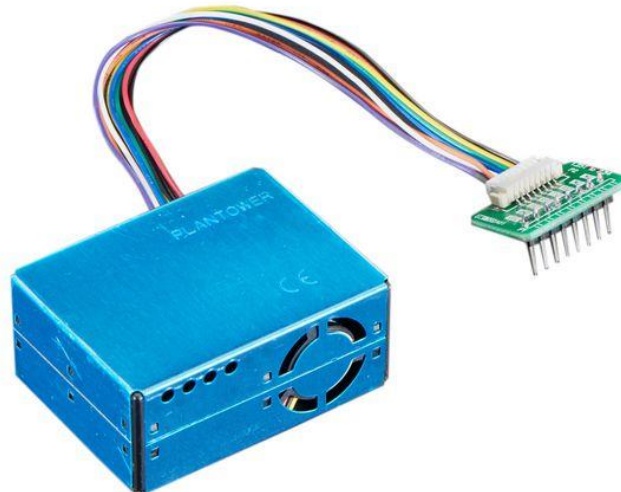
Αρχικοποιεί τον αισθητήρα DHT11 και διαβάζει τη θερμοκρασία και την υγρασία κάθε δύο δευτερόλεπτα, εκτυπώνοντας τα αποτελέσματα στην κονσόλα. Η μέθοδος Adafruit\_DHT.read\_retry είναι ιδιαίτερα χρήσιμη καθώς επιχειρεί να λάβει μετρήσεις αισθητήρα, επαναλαμβάνοντας έως και 15 φορές εάν αποτύχει. Αυτό διασφαλίζει ότι οι περιστασιακές αποτυχημένες αναγνώσεις δεν διαταράσσουν τη ροή του προγράμματός.

Ο συνδυασμός του αισθητήρα DHT11 και του Raspberry Pi προσφέρει μια πρακτική και οικονομικά αποδοτική λύση για την παρακολούθηση του περιβάλλοντος σε ένα ευρύ φάσμα ρυθμίσεων. Με λίγες μόνο γραμμές κώδικα Python, μπορείτε να αρχίσετε να συλλέγετε πολύτιμα δεδομένα για τη θερμοκρασία και την υγρασία, ανοίγοντας το δρόμο για πιο σύνθετα έργα όπως συστήματα οικιακού αυτοματισμού, μετεωρολογικούς σταθμούς ή έξυπνες εφαρμογές γεωργίας. Η προσαρμοστικότητα και η ευκολία χρήσης τόσο του DHT11 όσο και του Raspberry Pi διασφαλίζουν ότι τόσο οι λάτρεις όσο και οι επαγγελματίες μπορούν να υλοποιήσουν και να κλιμακώσουν τα έργα τους με σχετική ευκολία.

### 2.3 Αισθητήρας μέτρησης σωματιδίων PMS5003

Ο PMS5003 είναι ένας εξελιγμένος αισθητήρας σωματιδίων που έχει συγκεντρώσει σημαντική προσοχή στον τομέα της παρακολούθησης της ποιότητας του αέρα λόγω της εξαιρετικής ακρίβειας και αξιοπιστίας του. Σχεδιασμένο για να ανιχνεύει ένα ευρύ φάσμα μεγεθών σωματιδίων, συμπεριλαμβανομένων των PM1.0, PM2.5 και PM10, το PMS5003 ξεχωρίζει για την ικανότητά του να παρέχει πληροφορίες σε πραγματικό χρόνο σχετικά με τη συγκέντρωση σωματιδίων στον αέρα. Αυτό το χαρακτηριστικό είναι ιδιαίτερα κρίσιμο στο σημερινό περιβάλλον, όπου η ποιότητα του αέρα αποτελεί αυξανόμενη ανησυχία λόγω των βιομηχανικών δραστηριοτήτων και της αστικής ρύπανσης. Ο αισθητήρας λειτουργεί χρησιμοποιώντας τη θεωρία σκέδασης λέιζερ, η οποία περιλαμβάνει τη μέτρηση της ανάκλασης του φωτός λέιζερ από τα αιωρούμενα σωματίδια. Αυτή η μέθοδος προσφέρει

υψηλή ακρίβεια και απόκριση σε πραγματικό χρόνο, καθιστώντας το PMS5003 απαραίτητο εργαλείο για συστήματα περιβαλλοντικής παρακολούθησης, καθαριστές αέρα και συστήματα HVAC (Θέρμανση, εξαερισμός και κλιματισμός). Ο συμπαγής σχεδιασμός και η χαμηλή κατανάλωση ενέργειας του PMS5003 ενισχύουν περαιτέρω την ελκυστικότητά του, επιτρέποντάς του να ενσωματωθεί σε φορητές συσκευές και συσκευές που τροφοδοτούνται με μπαταρία, διευρύνοντας έτσι το πεδίο εφαρμογής του.



Εικόνα 2.3: Αισθητήρας PMS5003 [10]

Πέρα από τις τεχνικές του δυνατότητες, το PMS5003 επαινείται για τη φιλική προς τον χρήστη διεπαφή και την ευκολία ενσωμάτωσης με δημοφιλείς μικροελεγκτές και υπολογιστές με μία πλακέτα, όπως οι πλατφόρμες Raspberry Pi και Arduino. Η διαθεσιμότητα ολοκληρωμένης τεκμηρίωσης και μια πληθώρα βιβλιοθηκών ανοιχτού κώδικα απλοποιεί τη διαδικασία διασύνδεσης του PMS5003 με αυτά τα συστήματα, επιτρέποντας στους προγραμματιστές και τους χομπίστες να ενσωματώνουν εξελιγμένα χαρακτηριστικά παρακολούθησης της ποιότητας του αέρα στα έργα τους με ελάχιστη ταλαιπωρία. Επιπλέον, η ικανότητα του PMS5003 να λειτουργεί σε ένα ευρύ φάσμα θερμοκρασιών και επιπέδων υγρασίας εξασφαλίζει στιβαρή απόδοση σε διαφορετικές περιβαλλοντικές συνθήκες. Αυτή η ανθεκτικότητα, σε συνδυασμό με τις δυνατότητες ανίχνευσης λεπτών σωματιδίων του αισθητήρα, δίνει τη δυνατότητα σε άτομα και οργανισμούς να παρακολουθούν με ακρίβεια την ποιότητα του αέρα, ενισχύοντας τη λήψη τεκμηριωμένων αποφάσεων για την περιβαλλοντική υγεία και ασφάλεια. Καθώς η συνειδητοποίηση της σημασίας της ποιότητας του αέρα συνεχίζει να αυξάνεται, το PMS5003 πρόκειται να διαδραματίσει καθοριστικό ρόλο στην ενδυνάμωση των κοινοτήτων, των επιχειρήσεων και των ατόμων να κατανοήσουν και να βελτιώσουν το περιβάλλον στο οποίο ζουν και εργάζονται.

Ο PMS5003 είναι ένας εξελιγμένος αισθητήρας σωματιδίων που ξεχωρίζει για την ακρίβεια και την αξιοπιστία του στην παρακολούθηση της ποιότητας του αέρα, ιδιαίτερα στη μέτρηση λεπτών σωματιδίων, βασικό δείκτη περιβαλλοντικής υγείας. Στο επίκεντρο της λειτουργίας του βρίσκεται η αρχή της σκέδασης με λέιζερ, μια μέθοδος γνωστή για την ακρίβειά της στον εντοπισμό και τον ποσοτικό προσδιορισμό των αιωρούμενων σωματιδίων στον αέρα. Ο αισθητήρας στεγάζει μια δίοδο λέιζερ που εκπέμπει μια δέσμη φωτός και καθώς ο αέρας περνά μέσα από τον θάλαμο ανίχνευσης, τα σωματίδια μέσα στον αέρα διασκορπίζουν το φως σε διαφορετικές κατευθύνσεις. Αυτό το σκεδαζόμενο φως συλλαμβάνεται στη συνέχεια από μια φωτοδίοδο, η οποία μετατρέπει το φως σε σήμα που είναι ανάλογο με τον αριθμό των σωματιδίων. Αυτό που κάνει το PMS5003 εξαιρετικά έμπειρο είναι η ικανότητά του να διαφοροποιεί μεγέθη σωματιδίων, συμπεριλαμβανομένων των PM1.0, PM2.5 και PM10, που αντιπροσωπεύουν σωματίδια με διαμέτρους 1.0, 2.5 και 10 μικρόμετρα αντίστοιχα. Αυτή η ικανότητα είναι ζωτικής σημασίας, καθώς τα διαφορετικά μεγέθη σωματιδίων έχουν ποικίλες επιπτώσεις στην ποιότητα της υγείας και του περιβάλλοντος. Ο εξελιγμένος αλγόριθμος του αισθητήρα επεξεργάζεται τα σήματα από τη φωτοδίοδο, υπολογίζει τη συγκέντρωση των σωματιδίων και εξάγει τα δεδομένα μέσω μιας ψηφιακής διεπαφής, συνήθως σε μικρογραμμάρια ανά κυβικό μέτρο, παρέχοντας ένα σαφές και μετρήσιμο μέτρο της ποιότητας του αέρα.

Η ενσωμάτωση του PMS5003 με πλατφόρμες όπως το Raspberry Pi 3 ξεκλειδώνει πλήρως τις δυνατότητές του, επιτρέποντας τη συλλογή δεδομένων σε πραγματικό χρόνο, την ανάλυση και την οπτικοποίηση. Ο αισθητήρας συνήθως επικοινωνεί μέσω σειριακής διεπαφής, προσφέροντας απλότητα και αξιοπιστία στη μετάδοση δεδομένων. Οι προγραμματιστές μπορούν να αξιοποιήσουν αυτή τη δυνατότητα για να δημιουργήσουν ολοκληρωμένα συστήματα παρακολούθησης της ποιότητας του αέρα. Για παράδειγμα, συνδέοντας το PMS5003 σε ένα Raspberry Pi, μπορεί κανείς να αποθηκεύσει τις μετρήσεις των σωματιδίων σε μια βάση δεδομένων, επιτρέποντας μακροπρόθεσμη ανάλυση δεδομένων και παρατήρηση τάσεων. Επιπλέον, αυτή η ρύθμιση μπορεί να ενσωματωθεί σε μεγαλύτερα συστήματα, όπως έξυπνα οικιακά δίκτυα ή δίκτυα παρακολούθησης της ποιότητας του αέρα στα αστικά κέντρα, παρέχοντας χρήσιμες πληροφορίες και ειδοποιήσεις με βάση τις μετρήσεις της ποιότητας του αέρα. Η ευελιξία του PMS5003, σε συνδυασμό με την ευκολία ενσωμάτωσής του, το καθιστά ένα ανεκτίμητο εργαλείο για μια ευρεία γκάμα εφαρμογών, από τη βελτίωση της προσωπικής ευημερίας και της ποιότητας του αέρα σε εσωτερικούς χώρους έως τη συμβολή σε πρωτοβουλίες περιβαλλοντικής παρακολούθησης μεγάλης κλίμακας. Οι ακριβείς μετρήσεις και ο στιβαρός σχεδιασμός του διασφαλίζουν ότι ανταποκρίνεται στις απαιτήσεις τόσο των χομπίστων όσο και των επαγγελματιών με στόχο να κατανοήσουν και να βελτιώσουν την ποιότητα του αέρα στο περιβάλλον τους.

Ο PMS5003 είναι ένας εξελιγμένος αισθητήρας ικανός να παρέχει λεπτομερείς πληροφορίες σχετικά με τις συγκεντρώσεις σωματιδίων (PM) στον αέρα, που χρησιμοποιείται συχνά σε συνδυασμό με ένα Raspberry Pi 3 για συλλογή και ανάλυση δεδομένων. Αυτή η ρύθμιση είναι ιδιαίτερα επωφελής για έργα περιβαλλοντικής παρακολούθησης, επιτρέποντας στους χρήστες να συλλέγουν δεδομένα σε

πραγματικό χρόνο για τα επίπεδα PM1.0, PM2.5 και PM10. Το Raspberry Pi 3, με τις ισχυρές του δυνατότητες επεξεργασίας, τις ακίδες GPIO (General Purpose Input Output) και τη συνδεσιμότητα Wi-Fi, χρησιμεύει ως ιδανική πλατφόρμα για λήψη, επεξεργασία και μετάδοση δεδομένων από το PMS5003. Η ενσωμάτωση του αισθητήρα PMS5003 με το Raspberry Pi 3 όχι μόνο επιτρέπει την παρακολούθηση της ποιότητας του αέρα σε πραγματικό χρόνο, αλλά επιτρέπει επίσης την αποθήκευση, την ανάλυση και ακόμη και την κοινή χρήση των δεδομένων μέσω του Διαδικτύου. Οι χρήστες μπορούν να αναπτύξουν προσαρμοσμένες εφαρμογές, πίνακες εργαλείων ή να ενσωματωθούν με πλατφόρμες IoT για να οπτικοποιήσουν τα δεδομένα και να λάβουν τεκμηριωμένες αποφάσεις με βάση την ποιότητα του αέρα στο άμεσο περιβάλλον τους.

Για να ξεκινήσουμε, θα συνδέσουμε τον αισθητήρα PMS5003 στο Raspberry Pi 3 χρησιμοποιώντας μια σειριακή σύνδεση. Το παρακάτω είναι ένα παράδειγμα κώδικα Python που δείχνει πώς να διαβάζουμε δεδομένα από τον αισθητήρα PMS5003. Αυτός ο κώδικας χρησιμοποιεί τη βιβλιοθήκη pyserial για σειριακή επικοινωνία και υποθέτει ότι ο αισθητήρας είναι συνδεδεμένος στη σειριακή θύρα του Raspberry Pi. Πριν εκτελέσετε τον κώδικα, βεβαιωθείτε ότι η σειριακή διεπαφή είναι ενεργοποιημένη στο Raspberry Pi μέσω του βοηθητικού προγράμματος raspi-config και ότι η βιβλιοθήκη pyserial είναι εγκατεστημένη στο περιβάλλον Python.

```
import serial

import struct

from datetime import datetime

# Setup serial connection

serial_port = serial.Serial(

    port="/dev/ttyS0",

    baudrate=9600,

    timeout=2

)

# Function to read a frame from PMS5003

def read_pms5003_frame():

    frame = serial_port.read(32) # Read 32-byte frame from PMS5003
```

```

if len(frame) < 32:
    return None

# Verify frame header and checksum
if frame[0:2] != b'\x42\x4d' or sum(frame[0:30]) % 256 != frame[30]:
    return None

# Unpack frame data
frame_data = struct.unpack('>HHHHHHHHHHHHHHHH', frame[4:30])
return frame_data

try:
    while True:
        data = read_pms5003_frame()

        if data is not None:
            pm1_0, pm2_5, pm10 = data[0], data[1], data[2]
            print(f"{datetime.now()}: PM1.0={pm1_0}, PM2.5={pm2_5}, PM10={pm10}")
except KeyboardInterrupt:
    print("Terminated by user")
finally:
    serial_port.close()

```

Σε αυτό το παράδειγμα, η συνάρτηση `read_pms5003_frame` είναι υπεύθυνη για την ανάγνωση πλαισίων δεδομένων από τον αισθητήρα PMS5003. Διαβάζει ένα πλαίσιο 32 byte, επικυρώνει την κεφαλίδα του πλαισίου και το άθροισμα ελέγχου και, στη συνέχεια, αποσυσκευάζει τις τιμές συγκέντρωσης PM. Ο κύριος βρόχος διαβάζει συνεχώς δεδομένα από τον αισθητήρα και εκτυπώνει τις τιμές PM1.0, PM2.5 και PM10 μαζί με την τρέχουσα χρονική σήμανση. Αυτός ο κώδικας είναι μια βασική επίδειξη και μπορεί να επεκταθεί με πρόσθετο χειρισμό σφαλμάτων, καταγραφή δεδομένων ή ενοποίηση με

υπηρεσίες web για απομακρυσμένη παρακολούθηση και οπτικοποίηση δεδομένων. Η ευελιξία του Raspberry Pi 3, σε συνδυασμό με την ακρίβεια του PMS5003, ανοίγει μια τεράστια ποικιλία δυνατοτήτων για χομπίστες, ερευνητές και λάτρεις του περιβάλλοντος.

## 2.4 Αποστολή δεδομένων σε server

Η δημιουργία ενός πλήρους συστήματος που περιλαμβάνει την ανάγνωση δεδομένων αισθητήρα από το PMS5003 και το DHT11 και, στη συνέχεια, την αποστολή αυτών των δεδομένων σε έναν διακομιστή Flask μέσω Διαδικτύου περιλαμβάνει πολλά στοιχεία.

Πρώτα, πρέπει να ρυθμίσουμε το Raspberry Pi με τους αισθητήρες PMS5003 και DHT11. Θα χρειαστούμε βιβλιοθήκες για τη διασύνδεση με αυτούς τους αισθητήρες. Για το DHT11, μπορούμε να χρησιμοποιήσουμε τη βιβλιοθήκη Adafruit\_DHT και για το PMS5003, μπορούμε να χρησιμοποιήσουμε μια βιβλιοθήκη όπως το pms5003 ή να χειριστούμε την επικοινωνία χρησιμοποιώντας pyserial.

Ακολουθεί ένα δείγμα σεναρίου Python για την ανάγνωση δεδομένων από τους αισθητήρες:

```
import serial

import Adafruit_DHT

import time

import json

import requests

# Setup for DHT11

DHT_SENSOR = Adafruit_DHT.DHT11

DHT_PIN = 4 # GPIO pin connected to the DHT11 data pin

# Setup for PMS5003

PMS5003_SERIAL_PORT = '/dev/ttyS0' # The serial port to which PMS5003 is connected

ser = serial.Serial(PMS5003_SERIAL_PORT, baudrate=9600, timeout=2)

# Function to read from PMS5003
```

```

def read_pms5003():
    # Implement the reading mechanism based on your specific library or method
    # Here's a placeholder return statement
    return {
        "pm1_0": 12,
        "pm2_5": 15,
        "pm10": 20
    }

# Function to read from DHT11
def read_dht11():
    humidity, temperature = Adafruit_DHT.read(DHT_SENSOR, DHT_PIN)
    if humidity is not None and temperature is not None:
        return {"temperature": temperature, "humidity": humidity}
    else:
        return {"temperature": -1, "humidity": -1}

def send_data_to_server(data):
    SERVER_URL = "http://your_flask_server_address/data"
    headers = {'Content-Type': 'application/json'}
    response = requests.post(SERVER_URL, headers=headers, data=json.dumps(data))

while True:
    # Read sensor data
    pms_data = read_pms5003()
    dht_data = read_dht11()

    # Combine data
    combined_data = {**pms_data, **dht_data}

```

```
# Send data to the server
send_data_to_server(combined_data)
time.sleep(60) # Sleep for 1 minute
```

Ρύθμιση διακομιστή Flask

Στην πλευρά του διακομιστή Flask, θα χρειαστούμε ένα τελικό σημείο για να λάβουμε τα δεδομένα που αποστέλλονται από το Raspberry Pi. Εδώ είναι ένα απλό παράδειγμα:

```
from flask import Flask, request, jsonify
app = Flask(__name__)
@app.route('/data', methods=['POST'])
def receive_data():
    data = request.json
    print(data)
    return jsonify({"message": "Data received successfully"}), 200
if __name__ == '__main__':
    app.run(debug=True, host='0.0.0.0', port=5000)
```

Αυτός ο διακομιστής Flask θα ακούσει για αιτήματα POST στο τελικό σημείο /data και θα εκτυπώσει τα λαμβανόμενα δεδομένα.

## Κεφάλαιο 3ο: Τεχνολογία και προγραμματισμός

Στο κεφάλαιο αυτό θα περιγράψει η τεχνολογία και οι γλώσσες προγραμματισμού που χρησιμοποιήθηκαν για την υλοποίηση του συστήματος.

### 3.1 Python

Η Python ξεχωρίζει στον κόσμο του προγραμματισμού για την έμφαση που δίνει στην αναγνωσιμότητα, την απλότητα και την ευελιξία. Δημιουργήθηκε από τον Guido van Rossum και κυκλοφόρησε για πρώτη φορά το 1991, η σχεδιαστική φιλοσοφία της Python περιστρέφεται γύρω από την αναγνωσιμότητα κώδικα και τη χρήση σημαντικού κενού χώρου. Αυτή η εστίαση κάνει την Python μια εξαιρετική επιλογή τόσο για αρχάριους όσο και για έμπειρους προγραμματιστές, καθώς τους δίνει τη δυνατότητα να εκφράζουν έννοιες σε λιγότερες γραμμές κώδικα σε σύγκριση με άλλες γλώσσες προγραμματισμού. Η Python είναι μια πλήρης γλώσσα προγραμματισμού που μπορεί να χρησιμοποιηθεί για τα πάντα, από την ανάπτυξη απλών σεναρίων έως την κατασκευή πολύπλοκων αλγορίθμων μηχανικής εκμάθησης. Η εκτεταμένη τυπική βιβλιοθήκη της Python, είναι ένα από τα πιο ελκυστικά στοιχεία της, παρέχοντας εργαλεία κατάλληλα για πολλές εργασίες.

Η ευελιξία της Python είναι εμφανής στο τεράστιο εύρος των εφαρμογών που υποστηρίζει. Η ανάπτυξη ιστού, η ανάλυση δεδομένων, η τεχνητή νοημοσύνη, η επιστημονική πληροφορική και η ανάπτυξη λογισμικού είναι μόνο μερικοί τομείς όπου η Python υπερέχει. Τα πλαίσια και οι βιβλιοθήκες όπως το Django και το Flask κάνουν την ανάπτυξη ιστού παιχνιδάκι, ενώ η επιστήμη δεδομένων και η μηχανική εκμάθηση χρησιμοποιούν εκτεταμένες βιβλιοθήκες όπως οι NumPy, Pandas και Scikit-learn. Αυτή η προσαρμοστικότητα είναι ένας σημαντικός λόγος για τον οποίο η Python έχει αγκαλιαστεί τόσο από τον ακαδημαϊκό κόσμο όσο και από τη βιομηχανία, επιτρέποντας στους επαγγελματίες σε διάφορους τομείς να αξιοποιήσουν τον προγραμματισμό για τις συγκεκριμένες ανάγκες τους.

Η φύση ανοιχτού κώδικα της Python είναι ένας άλλος κρίσιμος παράγοντας για την ευρεία υιοθέτηση και επιτυχία της. Η κοινότητα της Python έχει συνεισφέρει αμέτρητες ενότητες και βιβλιοθήκες, καθιστώντας την ένα από τα πλουσιότερα οικοσυστήματα προγραμματισμού. Αυτή η κοινοτική υποστήριξη μεταφράζεται σε πληθώρα πόρων, όπως τεκμηρίωση, φόρουμ και σεμινάρια, που διευκολύνουν τη διαδικασία εκμάθησης και παρέχουν ουσιαστική υποστήριξη για αναπτυξιακά έργα. Είτε ένας προγραμματιστής αντιμετωπίζει ένα κοινό πρόβλημα είτε αντιμετωπίζει μια μοναδική

πρόκληση, υπάρχει μεγάλη πιθανότητα να υπάρχουν διαθέσιμοι πόροι ή συμβουλές από την κοινότητα για βοήθεια.

Ένα από τα καθοριστικά χαρακτηριστικά της Python είναι η χρήση μιας γλώσσας διερμηνείας και όχι μεταγλωττισμένης. Αυτό σημαίνει ότι ο κώδικας της Python εκτελείται γραμμή προς γραμμή, γεγονός που μπορεί να κάνει τον εντοπισμό σφαλμάτων ευκολότερο και πιο διαισθητικό, καθώς τα σφάλματα αναφέρονται αμέσως μόλις συμβούν. Ενώ οι γλώσσες διερμηνείας είναι γενικά πιο αργές από τις μεταγλωττισμένες γλώσσες, η αναγνωσιμότητα και τα οφέλη παραγωγικότητας της Python συχνά υπερτερούν αυτού του μειονεκτήματος. Επιπλέον, για εργασίες που απαιτούν υψηλή απόδοση, η Python μπορεί να ενσωματωθεί με επεκτάσεις γραμμένες σε γλώσσες όπως η C ή η C++, επιτρέποντας τη βελτιστοποίηση κρίσιμων τμημάτων μιας εφαρμογής.

Η δέσμευση της Python στην απλότητα και οι ισχυρές δυνατότητές της επεκτείνονται στον τομέα του ασύγχρονου προγραμματισμού. Με λειτουργίες όπως το `async` και το `await`, η Python κάνει τη σύνταξη ασύγχρονου κώδικα απλή και ευανάγνωστη. Αυτό είναι ιδιαίτερα ωφέλιμο όταν ασχολείστε με δομημένο κώδικα δικτύου υψηλού επιπέδου και δεσμευμένο σε IO. Ο ασύγχρονος προγραμματισμός επιτρέπει στην Python να χειρίζεται μεγάλο αριθμό ταυτόχρονων συνδέσεων, καθιστώντας την εξαιρετική επιλογή για την ανάπτυξη εφαρμογών όπως διακομιστές ιστού, πύλες IoT και άλλα συστήματα που απαιτούν χειρισμό πολλών ταυτόχρονων, ασύγχρονων λειτουργιών.

Στον τομέα της τεχνητής νοημοσύνης και της μηχανικής μάθησης, η Python είναι αναμφισβήτητα η πιο δημοφιλής γλώσσα. Η απλότητά του επιτρέπει τον εύκολο πειραματισμό και την επανάληψη, ενώ το ευρύ φάσμα των διαθέσιμων βιβλιοθηκών και πλαισίων καθιστά δυνατή τη δημιουργία εξελιγμένων μοντέλων και συστημάτων. Οι ερευνητές και οι επαγγελματίες μπορούν να επικεντρωθούν στην επίλυση του προβλήματος αντί να κολλήσουν στην πολυπλοκότητα της γλώσσας προγραμματισμού. Το ζωντανό οικοσύστημα γύρω από την Python σε τεχνητή νοημοσύνη και ML, συμπεριλαμβανομένων βιβλιοθηκών όπως οι TensorFlow, PyTorch και Keras, συνεχίζει να οδηγεί την καινοτομία και την πρόοδο στον τομέα.

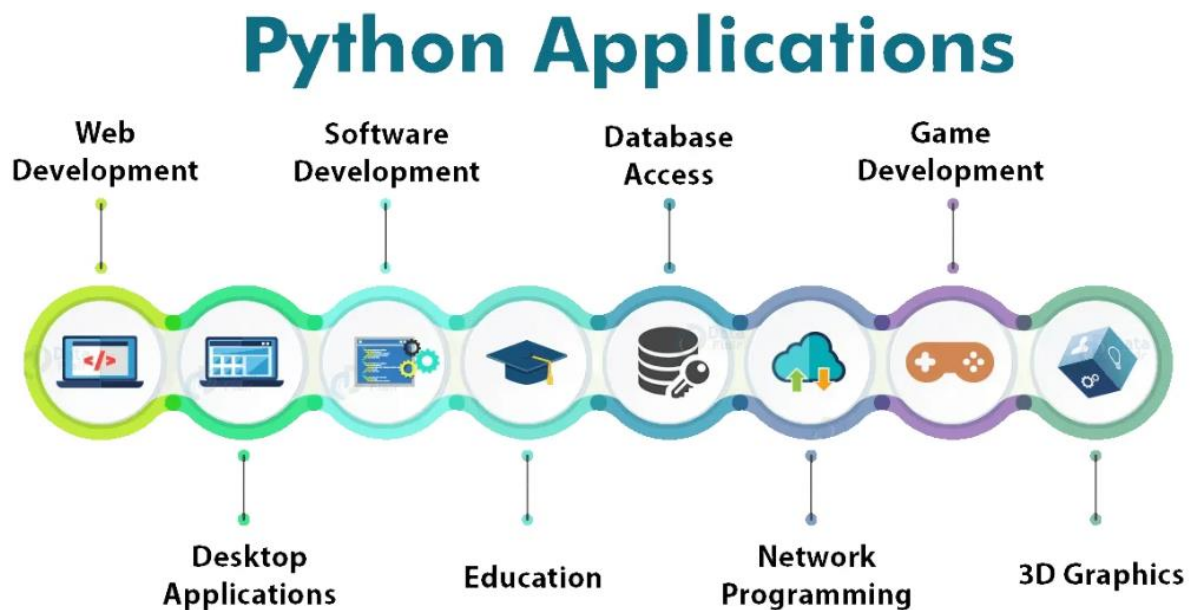
Ο συνδυασμός απλότητας, ευελιξίας και ισχυρής κοινότητας της Python την καθιστά μια γλώσσα που μπορεί να καλύψει ένα ευρύ φάσμα αναγκών. Από απλά σενάρια έως πολύπλοκα συστήματα, οι αρχές σχεδιασμού της Python και το εκτεταμένο οικοσύστημα παρέχουν μια σταθερή βάση για κάθε προγραμματιστική εργασία. Ο ρόλος της στην προώθηση της καινοτομίας σε τομείς όπως η ανάπτυξη ιστού, η επιστήμη δεδομένων και η τεχνητή νοημοσύνη εδραιώνουν περαιτέρω τη θέση της ως ακρογωνιαίο λίθο του σύγχρονου προγραμματισμού. Είτε για έναν αρχάριο που κάνει τα πρώτα του βήματα στην κωδικοποίηση είτε για έναν έμπειρο προγραμματιστή που εργάζεται σε τεχνολογίες αιχμής, η Python προσφέρει τα εργαλεία, την ευελιξία και την υποστήριξη που απαιτούνται για να μετατρέψουν τις ιδέες σε πραγματικότητα.

Η Python αποτελεί μια εξαιρετικά ευέλικτη και προσιτή γλώσσα προγραμματισμού, καθιστώντας την εξαιρετική επιλογή για την ανάπτυξη εφαρμογών στο Raspberry Pi 3. Η σαφής σύνταξη και οι ισχυρές βιβλιοθήκες της επιτρέπουν τόσο σε αρχάριους όσο και σε έμπειρους προγραμματιστές να δημιουργούν ένα ευρύ φάσμα εφαρμογών, από απλά σενάρια έως πολύπλοκα , προγράμματα πολλαπλών νημάτων. Η εκτεταμένη υποστήριξη της Python για διασύνδεση υλικού, επεξεργασία δεδομένων και επικοινωνία δικτύου την καθιστά ιδιαίτερα κατάλληλη για έργα που περιλαμβάνουν το Raspberry Pi. Βιβλιοθήκες όπως το RPi.GPIO επιτρέπουν τον άμεσο έλεγχο των ακίδων GPIO, επιτρέποντας στα σενάρια Python να αλληλεπιδρούν με ένα ευρύ φάσμα εξωτερικών αισθητήρων, κινητήρων και άλλου υλικού. Επιπλέον, η τεράστια συλλογή βιβλιοθηκών της Python, όπως η NumPy για αριθμητική επεξεργασία, το Matplotlib για οπτικοποίηση δεδομένων και το Flask για την ανάπτυξη εφαρμογών ιστού, παρέχει μια ισχυρή πλατφόρμα για τη μετατροπή του Raspberry Pi σε ένα ισχυρό εργαλείο για τα πάντα, από τον οικιακό αυτοματισμό έως την προηγμένη ανάλυση δεδομένων.

Στο πλαίσιο των έργων IoT και οικιακού αυτοματισμού, η Python στο Raspberry Pi 3 ανοίγει ατελείωτες δυνατότητες. Με την ενσωματωμένη συνδεσιμότητα Wi-Fi και Bluetooth, το Raspberry Pi 3 μπορεί να χρησιμεύσει ως κεντρικός κόμβος για ένα έξυπνο οικοσύστημα στο σπίτι. Τα σενάρια Python μπορούν να συλλέγουν δεδομένα από αισθητήρες, να επεξεργάζονται τις πληροφορίες και να λαμβάνουν αποφάσεις με βάση προκαθορισμένα κριτήρια. Για παράδειγμα, ένα πρόγραμμα Python θα μπορούσε να παρακολουθεί τα επίπεδα θερμοκρασίας και υγρασίας και να ελέγχει ένα σύστημα θέρμανσης, εξαερισμού και κλιματισμού για να διατηρεί ένα άνετο και ενεργειακά αποδοτικό περιβάλλον. Επιπλέον, η ενοποίηση με υπηρεσίες web ή πρωτόκολλα ανταλλαγής μηνυμάτων όπως το MQTT επιτρέπει στο Raspberry Pi να στέλνει ειδοποιήσεις ή δεδομένα σε άλλες συσκευές, επιτρέποντας στους χρήστες να παρακολουθούν και να ελέγχουν το περιβάλλον του σπιτιού τους από απόσταση. Η δυνατότητα διασύνδεσης με ένα ευρύ φάσμα αισθητήρων και ενεργοποιητών, επεξεργασίας των δεδομένων τοπικά και επικοινωνίας με άλλες συσκευές ή υπηρεσίες μέσω του Διαδικτύου, καθιστά το Raspberry Pi 3 ιδανική πλατφόρμα για την ανάπτυξη λύσεων έξυπνου σπιτιού με την Python.

Επιπλέον, η εκπαιδευτική αξία της Python στο Raspberry Pi 3 δεν μπορεί να υπερεκτιμηθεί. Ο συνδυασμός παρέχει μια εξαιρετική πλατφόρμα για τη διδασκαλία αρχών προγραμματισμού και επιστήμης υπολογιστών. Η απλότητα της Python επιτρέπει στους μαθητές να κατανοήσουν γρήγορα τα βασικά του προγραμματισμού, ενώ οι δυνατότητες υλικού του Raspberry Pi επιτρέπουν στους μαθητές να εφαρμόσουν αμέσως τις γνώσεις τους σε πρακτικά σενάρια πραγματικού κόσμου. Αυτή η πρακτική εμπειρία είναι ανεκτίμητη για την κατανόηση αφηρημένων εννοιών και για την ανάφλεξη του ενδιαφέροντος στα πεδία STEM. Εκπαιδευτικές πρωτοβουλίες όπως το Raspberry Pi Foundation προσφέρουν πόρους και έργα ειδικά σχεδιασμένα για να κάνουν την εκμάθηση προγραμματισμού και ηλεκτρονικών ειδών διασκεδαστική και ελκυστική. Είτε πρόκειται για τη δημιουργία απλών σεναρίων για τον έλεγχο των LED, την ανάπτυξη ενός μετεωρολογικού σταθμού για τη συλλογή και ανάλυση

περιβαλλοντικών δεδομένων ή την κατασκευή ενός ρομπότ που μπορεί να πλοηγηθεί σε έναν λαβύρινθο, η Python και το Raspberry Pi 3 μαζί προσφέρουν ένα γόνιμο έδαφος για καινοτομία, μάθηση και δημιουργικότητα.



Εικόνα 3.1: Python χρήσεις-εφαρμογές [11]

## 3.2 Flask

Το Flask είναι ένα ελαφρύ και ισχυρό πλαίσιο web για την Python, γνωστό για την απλότητα και την ευελιξία του. Είναι ένα μικροπλαίσιο, που σημαίνει ότι παρέχει τα βασικά εργαλεία και τις δυνατότητες που απαιτούνται για τη δημιουργία εφαρμογών ιστού, αλλά δεν γνωρίζει πώς πρέπει να δομήσετε την εφαρμογή σας ή ποια στοιχεία πρέπει να χρησιμοποιήσετε. Αυτή η λιτή προσέγγιση κάνει το Flask απίστευτα ευέλικτο, κατάλληλο για μικρά έργα με απλές απαιτήσεις, αλλά αρκετά στιβαρό ώστε να επεκτείνεται για πιο σύνθετες εφαρμογές. Το Flask είναι χτισμένο σε δύο βασικά στοιχεία, το Werkzeug και το Jinja2. Το Werkzeug είναι μια βιβλιοθήκη βοηθητικού προγράμματος WSGI, η οποία παρέχει τα εργαλεία για τη δημιουργία μιας διαδικτυακής εφαρμογής που επικοινωνεί με έναν διακομιστή ιστού. Το Jinja2 είναι μια μηχανή προτύπων, που επιτρέπει στους προγραμματιστές να δημιουργούν δυναμικά πρότυπα HTML για την εφαρμογή τους. Μαζί, αυτά τα στοιχεία αποτελούν τη ραχοκοκαλιά του Flask, επιτρέποντας στους προγραμματιστές να δημιουργούν εφαρμογές web με ευκολία και αποτελεσματικότητα [12].

Ένα από τα ξεχωριστά χαρακτηριστικά του Flask είναι ο ενσωματωμένος διακομιστής ανάπτυξης και ο εντοπισμός σφαλμάτων. Όταν αναπτύσσετε μια εφαρμογή Flask, μπορείτε να εκτελέσετε έναν τοπικό διακομιστή για να δοκιμάσετε την εφαρμογή σας σε πραγματικό χρόνο. Εάν υπάρχουν σφάλματα στον κώδικά σας, το Flask θα εμφανίσει μια διαδραστική ανίχνευση στο πρόγραμμα περιήγησης, επιτρέποντάς σας να επιθεωρήσετε και να επιλύσετε ζητήματα όταν προκύπτουν. Αυτή η δυνατότητα

επιταχύνει σημαντικά τη διαδικασία ανάπτυξης, καθώς οι προγραμματιστές μπορούν να δουν αμέσως τον αντίκτυπο των αλλαγών τους και να διορθώσουν τα προβλήματα αποτελεσματικά. Επιπλέον, το Flask υποστηρίζει διάφορες επεκτάσεις που μπορούν να προσθέσουν νέες δυνατότητες στην εφαρμογή σας. Αυτές οι επεκτάσεις κυμαίνονται από τον έλεγχο ταυτότητας χρήστη και την εξουσιοδότηση έως το χειρισμό των υποβολών φορμών και την αλληλεπίδραση με βάσεις δεδομένων. Αυτή η επεκτασιμότητα σημαίνει ότι ενώ το ίδιο το Flask παραμένει απλό και ελαφρύ, μπορεί εύκολα να επεκταθεί για να καλύψει τις αυξανόμενες ανάγκες του έργου σας.

Το σύστημα δρομολόγησης του Flask είναι μια άλλη ισχυρή πτυχή του πλαισίου. Επιτρέπει στους προγραμματιστές να αντιστοιχίζουν διευθύνσεις URL σε συναρτήσεις Python, καθιστώντας εύκολο τον σχεδιασμό καθαρών, ευανάγνωστων διευθύνσεων URL που είναι φιλικές προς το SEO και είναι εύκολο να θυμούνται. Αυτό το σύστημα είναι εξαιρετικά ευέλικτο, υποστηρίζει μεταβλητά μέρη σε μια διεύθυνση URL και μετατροπείς για να καθορίσετε τον τύπο της μεταβλητής, όπως συμβολοσειρές, ακέραιους αριθμούς ή κινητήρες. Αυτή η ευελιξία καθιστά εύκολη τη λήψη τιμών από τη διεύθυνση URL και τη μετάβασή τους ως ορίσματα στις λειτουργίες προβολής, απλοποιώντας τη διαδικασία δημιουργίας δυναμικών, φιλικών προς το χρήστη εφαρμογών ιστού. Επιπλέον, το Flask παρέχει υποστήριξη και για τις μεθόδους αιτήματος GET και POST, επιτρέποντας στους προγραμματιστές να χειρίζονται διαφορετικούς τύπους αιτημάτων HTTP εντός της εφαρμογής τους.

Η ασφάλεια αποτελεί πρωταρχικό μέλημα στην ανάπτυξη ιστού και το Flask είναι καλά εξοπλισμένο με εργαλεία για τη δημιουργία ασφαλών εφαρμογών. Προσφέρει ενσωματωμένη προστασία έναντι μιας σειράς κοινών επιθέσεων, συμπεριλαμβανομένης της δέσμης ενεργειών μεταξύ τοποθεσιών (XSS), της πλαστογράφησης αιτημάτων μεταξύ τοποθεσιών (CSRF) και της έγχυσης SQL. Το Flask προωθεί επίσης ασφαλείς πρακτικές από προεπιλογή, όπως η ασφαλής αποθήκευση δεδομένων συνεδρίας χρήστη και η ενθάρρυνση της χρήσης HTTPS. Ενώ το Flask παρέχει μια σταθερή βάση για την ασφάλεια, έχει σχεδιαστεί για να παρέχει στους προγραμματιστές την ευελιξία να εφαρμόζουν πρόσθετα μέτρα ασφαλείας όπως απαιτείται, διασφαλίζοντας ότι οι εφαρμογές μπορούν να προσαρμοστούν ώστε να πληρούν συγκεκριμένες απαιτήσεις ασφαλείας.

Το Flask είναι επίσης γνωστό για την εκτενή τεκμηρίωση και την υποστηρικτική του κοινότητα. Η επίσημη τεκμηρίωση είναι εμπειριστατομένη και καλά οργανωμένη, προσφέροντας ένα εξαιρετικό σημείο εκκίνησης για τους νέους προγραμματιστές του Flask και μια πολύτιμη πηγή για πιο έμπειρους χρήστες. Πέρα από την επίσημη τεκμηρίωση, υπάρχει πληθώρα εκπαιδευτικών προγραμμάτων, οδηγιών και παραδειγμάτων που έχουν δημιουργηθεί από την κοινότητα, που διευκολύνουν την εύρεση βοήθειας και έμπνευσης για τα έργα σας στο Flask. Η κοινότητα του Flask είναι ενεργή και φιλόξενη, με πολλά φόρουμ, δωμάτια συνομιλίας και λίστες αλληλογραφίας όπου οι προγραμματιστές μπορούν να συνδεθούν, να κάνουν ερωτήσεις και να μοιραστούν τις γνώσεις τους. Ξεχωρίζει ως πλαίσιο που εξισορροπεί την απλότητα με την ισχύ. Ο ελαφρύς πυρήνας του το καθιστά προσβάσιμο για αρχάριους, ενώ η ευελιξία και η επεκτασιμότητα του το καθιστούν κατάλληλο για πολύπλοκες εφαρμογές μεγάλης

κλίμακας. Είτε δημιουργείτε ένα μικρό προσωπικό έργο είτε μια εφαρμογή web υψηλής επισκεψιμότητας, το Flask παρέχει τα εργαλεία που χρειάζεστε για να δημιουργήσετε, να αναπτύξετε και να κλιμακώσετε αποτελεσματικά την εφαρμογή Ιστού σας. Με τα ενσωματωμένα εργαλεία ανάπτυξης, το ισχυρό σύστημα δρομολόγησης, τα ισχυρά χαρακτηριστικά ασφαλείας και την υποστηρικτική κοινότητα, το Flask είναι μια σταθερή επιλογή για ανάπτυξη ιστού στην Python.

### 3.3 MySQL

Η MySQL στέκεται ως φάρος στον κόσμο της διαχείρισης βάσεων δεδομένων, γνωστή για την στιβαρότητα, την αξιοπιστία και την απρόσκοπτη απόδοσή της [13]. Ως σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων ανοιχτού κώδικα (RDBMS), η θεμελιώδης αρχιτεκτονική του βασίζεται σε SQL (Structured Query Language), μια τυποποιημένη γλώσσα που χρησιμοποιείται για τη διαχείριση και το χειρισμό σχεσιακών βάσεων δεδομένων. Η εξέχουσα θέση της MySQL δεν είναι μη κερδισμένη. προσφέρει έναν βέλτιστο συνδυασμό απλότητας και ισχύος, καθιστώντας το ιδανική επιλογή τόσο για εφαρμογές μικρής κλίμακας όσο και για πολύπλοκες, μεγάλης κλίμακας επιχειρηματικές λύσεις. Η ευελιξία του υπογραμμίζεται περαιτέρω από τη συμβατότητά του σε πολλαπλές πλατφόρμες, επιτρέποντάς του να λειτουργεί σε διάφορα λειτουργικά συστήματα, όπως Linux, Windows και macOS. Αυτή η καθολική προσαρμοστικότητα, σε συνδυασμό με το πλούσιο σύνολο χαρακτηριστικών και την ευκολία ενσωμάτωσης με πολλές γλώσσες προγραμματισμού και εργαλεία, εδραιώνει τη θέση της MySQL ως ακρογωνιαίο λίθο στην ανάπτυξη δυναμικών εφαρμογών που βασίζονται σε δεδομένα.

Στο επίκεντρο της απήχησης της MySQL είναι η δέσμευσή της για απόδοση και αξιοπιστία. Το σύστημα έχει σχεδιαστεί για να χειρίζεται ένα σημαντικό φορτίο, διαχειριζόμενο μεγάλο όγκο δεδομένων με αξιοσημείωτη αποτελεσματικότητα. Αυτή η απόδοση αποδίδεται σε μεγάλο βαθμό στην ισχυρή μηχανή αναζήτησης που βελτιστοποιεί την ανάκτηση δεδομένων, επιτρέποντας γρήγορο και ακριβή χειρισμό δεδομένων. Επιπλέον, η MySQL υποστηρίζει μια σειρά από μηχανές αποθήκευσης, καθεμία προσαρμοσμένη σε συγκεκριμένες περιπτώσεις χρήσης. Αυτή η ευελιξία επιτρέπει στους προγραμματιστές να ρυθμίζουν με ακρίβεια τη βάση δεδομένων σύμφωνα με τις μοναδικές απαιτήσεις της εφαρμογής τους, είτε δίνουν προτεραιότητα στην ταχύτητα, την αξιοπιστία ή την ικανότητα χειρισμού σύνθετων συναλλαγών. Συγκεκριμένα, η InnoDB, μία από τις μηχανές αποθήκευσης της MySQL, προσφέρει πλήρη συμμόρφωση με ACID (Atomicity, Consistency, Isolation, Durability), διασφαλίζοντας την αξιοπιστία και την ακεραιότητα των λειτουργιών που βασίζονται σε συναλλαγές—ένα κρίσιμο χαρακτηριστικό για εφαρμογές όπου η ακρίβεια δεδομένων είναι πρωταρχικής σημασίας.

Η ασφάλεια εντός της MySQL λαμβάνεται με μεγάλη σοβαρότητα, αντικατοπτρίζοντας την ευρεία χρήση της σε εφαρμογές όπου η ευαισθησία δεδομένων αποτελεί κρίσιμο πρόβλημα. Το σύστημα παρέχει ολοκληρωμένους μηχανισμούς προστασίας δεδομένων, συμπεριλαμβανομένων ισχυρών προτύπων κρυπτογράφησης για την προστασία των δεδομένων τόσο σε κατάσταση ηρεμίας όσο και κατά τη μεταφορά. Ο έλεγχος πρόσβασης χρήστη είναι σχολαστικά λεπτομερής, επιτρέποντας στους διαχειριστές της βάσης δεδομένων να ορίζουν τα δικαιώματα χρήστη με ακριβή ακρίβεια, διασφαλίζοντας ότι τα άτομα μπορούν να αλληλεπιδρούν με τη βάση δεδομένων κατά τρόπο ευθυγραμμισμένο με τους ρόλους και τις ευθύνες τους. Επιπλέον, τα πρωτόκολλα ασφαλείας της MySQL ενημερώνονται διαρκώς, ενισχύοντας το σύστημα έναντι των αναδυόμενων απειλών και τρωτών σημείων, διατηρώντας έτσι την ιερότητα και την εμπιστευτικότητα των δεδομένων που διατηρεί.

Η πτυχή της επεκτασιμότητας στη MySQL μιλάει πολύ για τη σχεδιαστική της φιλοσοφία. Κατανοώντας την εξελισσόμενη φύση των επιχειρήσεων και των εφαρμογών, η MySQL έχει κατασκευαστεί σε κλίμακα. Είτε κλιμακώνεται κατακόρυφα αυξάνοντας τους πόρους του διακομιστή είτε γίνεται οριζόντια με την προσθήκη περισσότερων διακομιστών, η MySQL προσαρμόζεται απρόσκοπτα. Αυτή η επεκτασιμότητα διασφαλίζει ότι η βάση δεδομένων αναπτύσσεται παράλληλα με την εφαρμογή που υποστηρίζει, αντιμετωπίζοντας αποτελεσματικά αυξημένα φορτία δεδομένων και ταυτόχρονες συνδέσεις χρηστών χωρίς συμβιβασμούς στην απόδοση. Αυτό καθιστά τη MySQL μια συνετή επιλογή όχι μόνο για νεοφυείς επιχειρήσεις αλλά και για μεγάλες επιχειρήσεις που είναι έτοιμες για ανάπτυξη.

Η κοινοτική υποστήριξη και τεκμηρίωση αποτελούν τη ραχοκοκαλιά της διαρκούς δημοτικότητας της MySQL. Η ζωντανή κοινότητα των προγραμματιστών, των διαχειριστών βάσεων δεδομένων και των ενθουσιωδών συμβάλλει σε ένα πλούσιο οικοσύστημα πόρων, συμπεριλαμβανομένων λεπτομερών τεκμηρίωσης, φόρουμ, ιστολογίων και σεμιναρίων. Αυτός ο πλούτος γνώσεων διευκολύνει μια ομαλή καμπύλη μάθησης για τους νεοφερμένους και παρέχει στους έμπειρους επαγγελματίες πληροφορίες για προηγμένες τεχνικές βελτιστοποίησης και βέλτιστες πρακτικές. Η ενεργή κοινότητα οδηγεί επίσης την εξέλιξη της MySQL, με τους συνεργάτες ανοιχτού κώδικα να ενισχύουν συνεχώς τις δυνατότητές της, διασφαλίζοντας ότι η βάση δεδομένων παραμένει στην αιχμή της τεχνολογίας, ικανοποιώντας και προβλέποντας τις ανάγκες των σύγχρονων εφαρμογών.

Ο συνδυασμός απόδοσης, αξιοπιστίας, ασφάλειας και επεκτασιμότητας της MySQL, σε συνδυασμό με την ισχυρή υποστήριξη της κοινότητας, την καθιστά μια αξιόπιστη και ευέλικτη λύση στον τομέα της διαχείρισης βάσεων δεδομένων. Από την τροφοδοσία μικρών ιστότοπων έως τη ραχοκοκαλιά μεγάλων εταιρικών εφαρμογών, η ικανότητα της MySQL να προσαρμόζεται και να υπερέχει υπό ποικίλους φόρτους εργασίας και απαιτήσεις υπογραμμίζει την κατάστασή της ως προτιμώμενο RDBMS. Η συνεχής ανάπτυξή της και η δέσμευση της κοινότητάς της υπόσχονται να διατηρήσουν τη MySQL σχετική και ισχυρή, καλύπτοντας το συνεχώς εξελισσόμενο τοπίο των αναγκών διαχείρισης δεδομένων.

Για να συνδεθείτε σε μια βάση δεδομένων MySQL από την Python, μπορείτε να χρησιμοποιήσετε το πακέτο `mysql-connector-python`, το οποίο είναι ένα πρόγραμμα οδήγησης που παρέχεται από τη MySQL για σύνδεση σε διακομιστή βάσης δεδομένων MySQL. Ακολουθεί ένα παράδειγμα που δείχνει πώς να συνδεθείτε σε μια βάση δεδομένων MySQL.

Πρώτα, πρέπει να εγκαταστήσετε το πακέτο `mysql-connector-python`. Μπορείτε να το εγκαταστήσετε χρησιμοποιώντας το `pip`:

```
pip install mysql-connector-python
```

Αφού εγκαταστήσετε το πακέτο, μπορείτε να χρησιμοποιήσετε το ακόλουθο script Python ως παράδειγμα για να συνδεθείτε στη βάση δεδομένων MySQL, να εκτελέσετε ένα ερώτημα και να λάβετε τα αποτελέσματα:

```
import mysql.connector

from mysql.connector import Error

def connect_fetch():

    """ Connect to MySQL database and fetch data """

    conn = None

    try:

        conn = mysql.connector.connect(host='localhost',

                                      database='your_database',

                                      user='your_username',

                                      password='your_password')

    if conn.is_connected():

        print('Connected to MySQL database')

        cursor = conn.cursor()

        cursor.execute("SELECT * FROM your_table")
```

```

rows = cursor.fetchall()

print("Total Row(s):", cursor.rowcount)

for row in rows:

    print(row)

except Error as e:

    print(e)

finally:

    if conn is not connected() and conn is not None:

        conn.close()

if __name__ == '__main__':

    connect_fetch()

```

### 3.4 Javascript

Η JavaScript είναι μια βασική και ευέλικτη γλώσσα δέσμης ενεργειών που αποτελεί τη ραχοκοκαλιά του διαδραστικού και δυναμικού περιεχομένου Ιστού, διαδραματίζοντας κεντρικό ρόλο στη σύγχρονη ανάπτυξη Ιστού παράλληλα με την HTML και το CSS [14]. Αρχικά σχεδιασμένο για τη δημιουργία μικρών σεναρίων από την πλευρά του πελάτη, η JavaScript έχει εξελιχθεί σημαντικά με την πάροδο των ετών, καθιστώντας μια πλήρως ανεπτυγμένη, αντικειμενοστραφή γλώσσα προγραμματισμού ικανή να χειρίζεται πολύπλοκες εφαρμογές. Λειτουργεί εντός του προγράμματος περιήγησης ιστού του χρήστη χωρίς την ανάγκη επεξεργασίας από την πλευρά του διακομιστή, καθιστώντας το ένα θεμελιώδες εργαλείο για τη βελτίωση των εμπειριών των χρηστών με ενημερώσεις σε πραγματικό χρόνο, διαδραστικούς χάρτες, κινούμενα γραφικά 2D/3D, κύλιση βίντεο jukebox και πολλά άλλα. Η πραγματική δύναμη της JavaScript έγκειται στην ικανότητά της να χειρίζεται το Μοντέλο Αντικειμένου Εγγράφου (DOM), επιτρέποντας στους προγραμματιστές να αλλάζουν δυναμικά το περιεχόμενο, τη

δομή και το στυλ μιας ιστοσελίδας, αντιδρώντας στις ενέργειες των χρηστών χωρίς να χρειάζεται να φορτώσουν ξανά ολόκληρη τη σελίδα. Αυτή η διαδραστική δυνατότητα είναι αναπόσπαστη για τη δημιουργία σύγχρονων, αποκρινόμενων εφαρμογών web.

Το οικοσύστημα που περιβάλλει τη JavaScript έχει αναπτυχθεί εκθετικά, με μια πληθώρα βιβλιοθηκών και πλαισίων που έχουν σχεδιαστεί για να εξορθολογίσουν την ανάπτυξη και να επεκτείνουν τις δυνατότητες της γλώσσας. Βιβλιοθήκες όπως το jQuery απλοποιούν τον χειρισμό DOM και τον χειρισμό συμβάντων, αφαιρώντας τις ασυνέπειες του προγράμματος περιήγησης και μειώνοντας τον όγκο του κώδικα που πρέπει να γράψουν οι προγραμματιστές. Εν τω μεταξύ, πλαίσια όπως το Angular, το React και το Vue.js προσφέρουν πιο δομημένα περιβάλλοντα για τη δημιουργία επεκτάσιμων και διατηρήσιμων εφαρμογών από την πλευρά του πελάτη. Αυτά τα εργαλεία επιβάλλουν μια αρθρωτή προσέγγιση στο σχεδιασμό εφαρμογών, προωθώντας επαναχρησιμοποιήσιμα στοιχεία και δηλωτικά παραδείγματα προγραμματισμού που κάνουν τον κώδικα πιο διαισθητικό και ισχυρό. Επιπλέον, η εμφάνιση του Node.js σηματοδότησε ένα σημαντικό ορόσημο για το JavaScript, επεκτείνοντας την εμβέλειά του από το πρόγραμμα περιήγησης στην πλευρά του διακομιστή. Το Node.js είναι χτισμένο στη μηχανή JavaScript V8 του Chrome, επιτρέποντας εφαρμογές υψηλής απόδοσης από την πλευρά του διακομιστή και τη δυνατότητα χειρισμού πολυάριθμων ταυτόχρονων συνδέσεων με υψηλή απόδοση, καθιστώντας την JavaScript μια γλώσσα «πλήρους στοίβας» κατάλληλη για ανάπτυξη τόσο από τον πελάτη όσο και από τον διακομιστή.

Μία από τις πιο συναρπαστικές πτυχές της JavaScript είναι το μοντέλο εισόδου/εξόδου που βασίζεται σε συμβάντα, χωρίς αποκλεισμό, ιδιαίτερα σε περιβάλλον Node.js. Αυτό το μοντέλο επιτρέπει στην JavaScript να χειρίζεται πολλές λειτουργίες ταυτόχρονα χωρίς να περιμένει να ολοκληρωθεί καμία μεμονωμένη λειτουργία, μια δυνατότητα που είναι ιδιαίτερα ωφέλιμη για την ανάπτυξη εφαρμογών υψηλής κλιμάκωσης. Είτε χειρίζεται χιλιάδες ταυτόχρονα αιτήματα σε έναν διακομιστή ιστού είτε αλληλεπιδρά με βάσεις δεδομένων και συστήματα αρχείων, η ασύγχρονη φύση της JavaScript διασφαλίζει ότι οι εφαρμογές παραμένουν ανταποκρινόμενες και αποτελεσματικές υπό συνθήκες μεγάλου φόρτου. Σε συνδυασμό με τον βρόχο συμβάντων και τις υποσχέσεις, η ασύγχρονη JavaScript παρέχει ένα ισχυρό παράδειγμα για το χειρισμό λειτουργιών που διαφορετικά θα ήταν χρονοβόρες ή εντατικές με πόρους.

Παρά τα αμέτρητα πλεονεκτήματά της, η δυναμική και χαλαρή φύση της JavaScript συχνά οδηγεί σε κριτική, ιδιαίτερα όσον αφορά τη διατήρηση και την αξιοπιστία του κώδικα σε εφαρμογές μεγάλης κλίμακας. Σε απάντηση, το TypeScript, ένα υπερσύνολο JavaScript που αναπτύχθηκε από τη Microsoft, προσφέρει προαιρετική στατική πληκτρολόγηση μαζί με τις πιο πρόσφατες δυνατότητες ECMAScript. Το σύστημα τύπων του TypeScript επιτρέπει σαφείς δηλώσεις τύπου, επιτρέποντας στους προγραμματιστές να γράφουν πιο προβλέψιμο και ανθεκτικό σε σφάλματα κώδικα. Η γλώσσα μεταφράζεται σε απλή JavaScript, διασφαλίζοντας τη συμβατότητα με οποιοδήποτε πρόγραμμα περιήγησης ή μηχανή JavaScript. Αυτός ο συνδυασμός ευελιξίας και ευρωστίας καθιστά το TypeScript

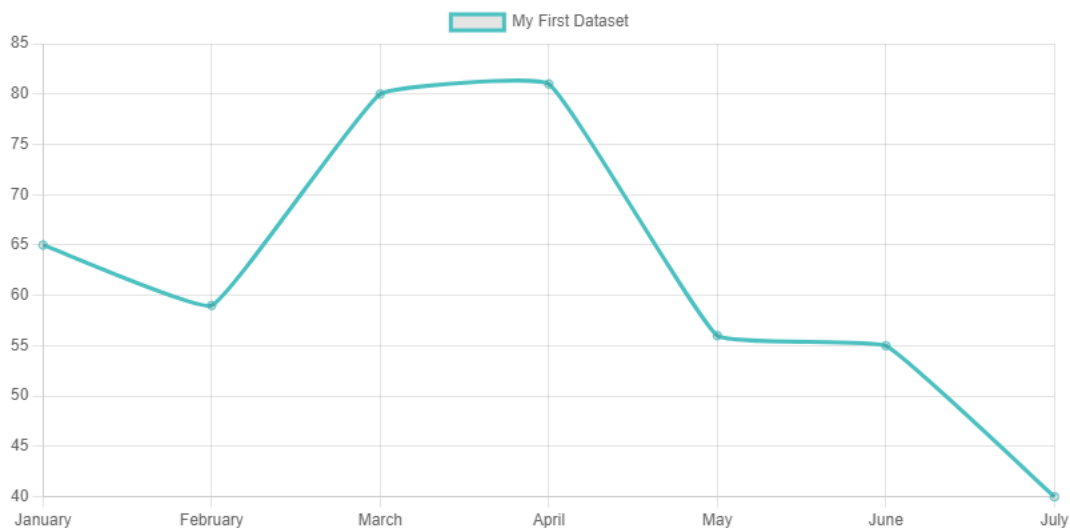
μια ολοένα και πιο δημοφιλή επιλογή για εφαρμογές σε εταιρικό επίπεδο, προσφέροντας ό,τι καλύτερο και από τους δύο κόσμους: την ευελιξία της JavaScript και την αξιοπιστία μιας στατικά πληκτρολογημένης γλώσσας.

Η ευελιξία της JavaScript εκτείνεται πέρα από τα προγράμματα περιήγησης και τους διακομιστές ιστού στη σφαίρα της ανάπτυξης εφαρμογών για κινητά. Πλαίσια όπως το React Native και το NativeScript επιτρέπουν στους προγραμματιστές να δημιουργούν εφαρμογές για κινητά χρησιμοποιώντας JavaScript και επαναχρησιμοποιήσιμο κώδικα, ο οποίος μπορεί να μεταγλωττιστεί σε εγγενή κώδικα εφαρμογής για Android και iOS. Αυτή η προσέγγιση επιτρέπει τη δημιουργία εφαρμογών για κινητές συσκευές υψηλής απόδοσης, πολλαπλών πλατφορμών, χωρίς να χρειάζεται οι προγραμματιστές να μάθουν Swift ή Kotlin για κάθε πλατφόρμα. Η δυνατότητα κοινής χρήσης κώδικα μεταξύ πλατφορμών ιστού και κινητών όχι μόνο επιταχύνει τη διαδικασία ανάπτυξης, αλλά διασφαλίζει επίσης τη συνέπεια στην εμπειρία χρήστη σε διαφορετικές συσκευές. Η γλώσσα εξελίσσεται συνεχώς, με το πρότυπο ECMAScript να εισάγει τακτικά νέες δυνατότητες και βελτιώσεις. Οι καινοτομίες στην τεχνολογία web, όπως το WebAssembly, ανοίγουν ακόμη περισσότερες δυνατότητες για εφαρμογές web υψηλής απόδοσης, επεκτείνοντας πιθανώς τη χρήση της JavaScript σε περιοχές όπου παραδοσιακά κυριαρχούν οι εφαρμογές επιτραπέζιου υπολογιστή. Σε συνδυασμό με σύγχρονα εργαλεία ανάπτυξης, βελτιστοποιήσεις απόδοσης και πλαίσια που βασίζονται στην κοινότητα, η JavaScript αποτελεί τον ακρογωνιαίο λίθο της ανάπτυξης ιστού, οδηγώντας την καινοτομία και διαμορφώνοντας το μέλλον του τρόπου με τον οποίο οι χρήστες αλληλεπιδρούν με τον Ιστό. Καθώς συνεχίζει να εξελίσσεται, η JavaScript υπόσχεται να παραμείνει στην πρώτη γραμμή του ψηφιακού τοπίου, προσφέροντας ένα δυναμικό, ευέλικτο και ισχυρό σύνολο εργαλείων για προγραμματιστές σε όλο τον κόσμο.

### 3.4.1 Chart.js

Το Chart.js είναι μια ισχυρή και ευέλικτη βιβλιοθήκη JavaScript ανοιχτού κώδικα που εξουσιοδοτεί τους προγραμματιστές να δημιουργήσουν ένα ευρύ φάσμα διαδραστικών και αποκρινόμενων γραφημάτων εντός εφαρμογών ιστού [15]. Η απλότητα και η ευελιξία του εκτιμώνται ιδιαίτερα στην κοινότητα προγραμματιστών, καθιστώντας την μια από τις πιο δημοφιλείς βιβλιοθήκες χαρτογράφησης. Το Chart.js παρέχει ένα πλούσιο σύνολο τύπων γραφημάτων, όπως γραμμή, ράβδος, ραντάρ, ντόνατ, πίτα, πολική περιοχή, φουσαλίδα και διασπορά, καλύπτοντας ένα ευρύ φάσμα αναγκών οπτικοποίησης δεδομένων. Ο ανταποκρινόμενος σχεδιασμός της βιβλιοθήκης διασφαλίζει ότι τα γραφήματα όχι μόνο φαίνονται υπέροχα σε διαφορετικές συσκευές, αλλά και προσαρμόζονται απρόσκοπτα στις αλλαγές μεγέθους οθόνης, διατηρώντας την αναγνωσιμότητα και τη λειτουργικότητα σε όλες τις πλατφόρμες. Το Chart.js αξιοποιεί το στοιχείο καμβά HTML5 για απόδοση, προσφέροντας ευκρινείς και καλά βελτιστοποιημένες απεικονίσεις. Η εκτεταμένη τεκμηρίωση και η υποστήριξη της

κοινότητας το καθιστούν προσβάσιμο τόσο για αρχάριους όσο και για έμπειρους προγραμματιστές, επιτρέποντας τη γρήγορη ενσωμάτωση και προσαρμογή των γραφημάτων. Η δυνατότητα κίνησης γραφημάτων, προσθήκης προσαρμοσμένων προσθηκών και ο ελαφρύς χαρακτήρας της βιβλιοθήκης συμβάλλουν περαιτέρω στη δημοτικότητά της, καθιστώντας την ιδανική λύση για προγραμματιστές που θέλουν να εμπλουτίσουν τις διαδικτυακές τους εφαρμογές με οπτικά ελκυστικές και ενημερωτικές αναπαραστάσεις δεδομένων.



Εικόνα 3.2: Παράδειγμα Chart.js [16]

Η πρακτική εφαρμογή του Chart.js σε έργα ιστού είναι απλή, χάρη στο διαισθητικό API και τις ολοκληρωμένες επιλογές διαμόρφωσης. Οι προγραμματιστές μπορούν να ορίσουν σύνολα δεδομένων, να προσαρμόσουν άξονες, να ορίσουν μύθους, συμβουλές εργαλείων και πολλά άλλα, προσαρμόζοντας τα γραφήματα στις συγκεκριμένες απαιτήσεις των έργων τους. Οι δυνατότητες διαχείρισης συμβάντων της βιβλιοθήκης επιτρέπουν τη δημιουργία διαδραστικών γραφημάτων όπου οι χρήστες μπορούν να κάνουν κλικ ή να τοποθετούν το δείκτη του ποντικιού πάνω σε στοιχεία γραφήματος για να αποκαλύψουν πρόσθετες πληροφορίες ή να ενεργοποιήσουν ενέργειες, βελτιώνοντας την εμπειρία και την αφοσίωση του χρήστη. Επιπλέον, το Chart.js ενσωματώνεται καλά με διάφορα πλαίσια και βιβλιοθήκες, όπως το React, το Vue.js και το Angular, καθιστώντας το μια ευέλικτη επιλογή σε διαφορετικά περιβάλλοντα ανάπτυξης. Η δέσμευσή του στην απόδοση και την αποτελεσματικότητά του διασφαλίζει ότι ακόμη και όταν έχουμε να κάνουμε με μεγάλα σύνολα δεδομένων ή πολύπλοκες απεικονίσεις, τα γραφήματα παραμένουν αποδοτικά και ανταποκρίνονται. Ουσιαστικά, το Chart.js είναι ένας ισχυρός σύμμαχος στον τομέα της οπτικοποίησης δεδομένων, παρέχοντας στους προγραμματιστές τα εργαλεία για τη μετάφραση δεδομένων σε ουσιαστικές, ελκυστικές και αισθητικά ευχάριστες οπτικές ιστορίες, επιτρέποντας έτσι στους χρήστες να κατανοούν και να αλληλεπιδρούν με τα δεδομένα πιο αποτελεσματικά.

Το Chart.js είναι μια ισχυρή και ευέλικτη βιβλιοθήκη για τη δημιουργία διαδραστικών γραφημάτων σε εφαρμογές web. Παρακάτω είναι ένα παράδειγμα για το πώς μπορείτε να χρησιμοποιήσετε το Chart.js για να δημιουργήσετε ένα απλό γραμμικό γράφημα.

Αρχικά, πρέπει να συμπεριλάβετε το Chart.js στο αρχείο HTML σας. Μπορείτε να κάνετε λήψη του Chart.js και να το φιλοξενήσετε τοπικά ή να χρησιμοποιήσετε ένα CDN (Δίκτυο παράδοσης περιεχομένου) για να το συμπεριλάβετε απευθείας. Εδώ, θα χρησιμοποιήσω το CDN:

```
<!DOCTYPE html>
<html>
<head>
  <title>Chart.js Example</title>
  <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>
</head>
<body>
  <div style="width:60%;">
    <canvas id="myChart"></canvas>
  </div>

  <script>
    // Setup block
    const data = {
      labels: ['January', 'February', 'March', 'April', 'May', 'June', 'July'],
      datasets: [{
        label: 'Monthly Sales Data',
        backgroundColor: 'rgb(255, 99, 132)',
        borderColor: 'rgb(255, 99, 132)',
        data: [0, 10, 5, 2, 20, 30, 45],
      }]
    };

    // Configuration block
    const config = {
      type: 'line',
      data: data,
```

```
    options: {}  
  };  
  
  // Render block  
  const myChart = new Chart(  
    document.getElementById('myChart'),  
    config  
  );  
</script>  
</body>  
</html>
```

Το στοιχείο `<canvas>` με `id="myChart"` είναι το σημείο όπου θα αποδοθεί το γράφημα.

Οι ετικέτες `<script>` περιλαμβάνουν τη βιβλιοθήκη `Chart.js` από το CDN και το προσαρμοσμένο σενάριο για τη δημιουργία γραφήματος.

Το αντικείμενο δεδομένων περιέχει τα δεδομένα και τη διαμόρφωση του γραφήματος σας.

Οι ετικέτες είναι οι κατηγορίες που εμφανίζονται στον άξονα x.

Τα σύνολα δεδομένων είναι μια σειρά από σύνολα δεδομένων. Κάθε σύνολο δεδομένων έχει μια ετικέτα (όνομα υπομνήματος), `backgroundColor` και `borderColor` για το στυλ και έναν πίνακα δεδομένων που περιέχει τις τιμές για κάθε ετικέτα.

Το αντικείμενο διαμόρφωσης περιέχει τη διαμόρφωση του γραφήματος. Ο τύπος καθορίζει τον τύπο του γραφήματος (π.χ. "γραμμή", "ράβδος" κ.λπ.). Στην ιδιότητα δεδομένων εκχωρείται το αντικείμενο δεδομένων από το μπλοκ εγκατάστασης.

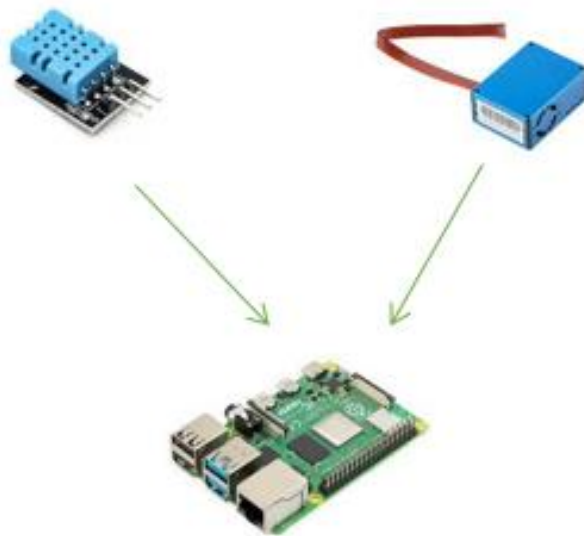
Το γράφημα αποδίδεται στο στοιχείο καμβά χρησιμοποιώντας το νέο `γράφημα()`. Χρειάζεται δύο παραμέτρους: το περιβάλλον (ανακτήθηκε από το `document.getElementById('myChart')`) και το αντικείμενο διαμόρφωσης.

Όταν εκτελείτε αυτόν τον κώδικα σε ένα πρόγραμμα περιήγησης, θα εμφανίζει ένα γραμμικό γράφημα που αντιπροσωπεύει τα "Δεδομένα μηνιαίων πωλήσεων". Μπορείτε να προσαρμόσετε το γράφημα αλλάζοντας τον τύπο του (π.χ. σε "bar", "pie" κ.λπ.), προσθέτοντας περισσότερα σύνολα δεδομένων ή διαμορφώνοντας επιλογές για να φαίνεται και να συμπεριφέρεται όπως θέλετε. Η τεκμηρίωση του `Chart.js` παρέχει μια ολοκληρωμένη λίστα με διαμορφώσεις και παραδείγματα που θα σας βοηθήσουν να προσαρμόσετε τα γραφήματα σας στις ανάγκες σας.

## Κεφάλαιο 4ο: Το σύστημα μέτρησης ποιότητας αέρα σε πραγματικό χρόνο σε κλειστό χώρο

### 4.1 Εισαγωγή – Η διαδικασία

Το σύστημα μέτρησης ποιότητας αέρα σε πραγματικό χρόνο σε κλειστό χώρο αποτελείται από το Raspberry Pi 3 και τους δύο αισθητήρες PMS5003 για μέτρηση σωματιδίων στον αέρα και το DHT11 για μέτρηση θερμοκρασίας και υγρασίας στο χώρο, όπως φαίνεται στην Εικόνα 4.1.



Εικόνα 4.1: Σύνδεση Raspberry με τους αισθητήρες

Στην Εικόνα 4.2 φαίνεται το διάγραμμα του συστήματος το οποίο περιλαμβάνει την αποστολή των δεδομένων-μετρήσεων από τους αισθητήρες στον server και την αποθήκευση τους στην βάση και την προβολή τους σε ιστοσελίδα στους τελικούς εξουσιοδοτημένους ή μη χρήστες.

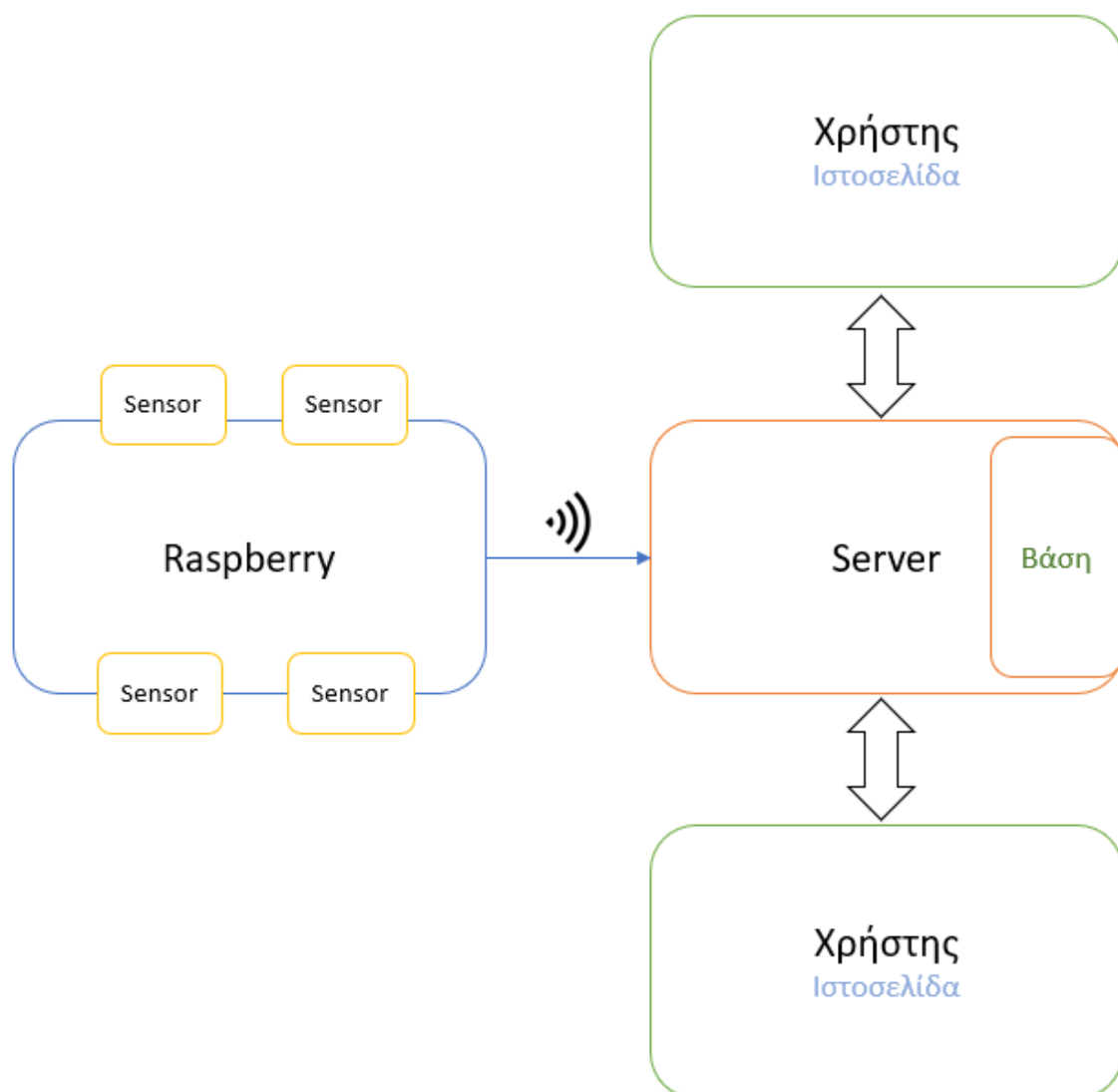
Το εν λόγω σύστημα αντιπροσωπεύει μια ολοκληρωμένη λύση για τη λήψη, αποθήκευση και οπτικοποίηση δεδομένων αισθητήρων, γεφυρώνοντας το χάσμα μεταξύ της συλλογής ακατέργαστων δεδομένων και της παρουσίασης πληροφοριών με επίκεντρο τον χρήστη. Στον πυρήνα του, το σύστημα ενσωματώνει τεχνολογία αισθητήρων, διαχείριση δεδομένων από την πλευρά του διακομιστή και οπτικοποίηση μέσω ιστού για να εξυπηρετεί τόσο εξουσιοδοτημένους όσο και μη εξουσιοδοτημένους χρήστες. Η αρχική φάση του συστήματος περιλαμβάνει την ανάπτυξη αισθητήρων ικανών να μετρούν διάφορες περιβαλλοντικές παραμέτρους. Αυτοί οι αισθητήρες είναι στρατηγικά τοποθετημένοι για να παρακολουθούν συνθήκες όπως η θερμοκρασία, η υγρασία, η ποιότητα του αέρα ή άλλες σχετικές

μετρήσεις, ανάλογα με τη συγκεκριμένη εφαρμογή και τις απαιτήσεις του χρήστη. Οι αισθητήρες έχουν σχεδιαστεί για να συλλέγουν δεδομένα σε προκαθορισμένα χρονικά διαστήματα, διασφαλίζοντας ένα συνεπές και πλούσιο σύνολο δεδομένων. Αυτά τα δεδομένα στη συνέχεια μεταδίδονται σε έναν κεντρικό διακομιστή, συνήθως μέσω μιας ασφαλούς και αξιόπιστης σύνδεσης δικτύου. Το πρωτόκολλο και η συχνότητα μετάδοσης επιλέγονται προσεκτικά για να εξισορροπούν τις έγκαιρες ενημερώσεις δεδομένων με τη διατήρηση των πόρων του δικτύου, διασφαλίζοντας ένα αποτελεσματικό και ανταποκρινόμενο σύστημα.

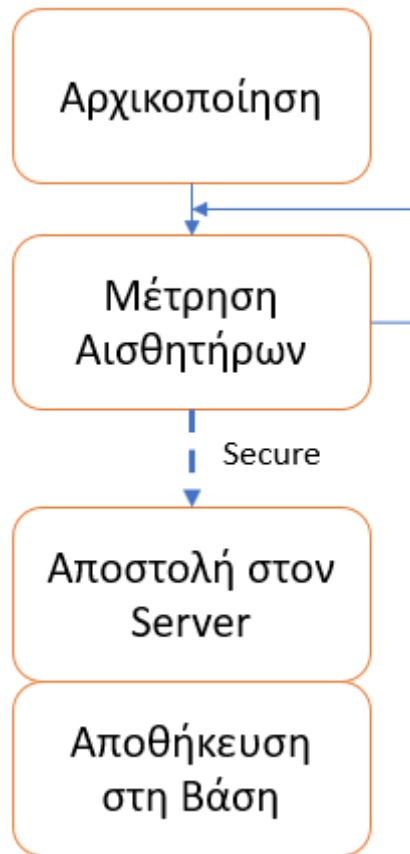
Μόλις τα δεδομένα φτάσουν στον διακομιστή, υποβάλλονται σε μια σειρά βημάτων επεξεργασίας για να διασφαλιστεί η ακεραιότητα και η χρησιμότητά τους. Ο διακομιστής, εξοπλισμένος με ισχυρά συστήματα διαχείρισης βάσεων δεδομένων, όπως η MySQL, αποθηκεύει με ασφάλεια τα εισερχόμενα δεδομένα, δημιουργώντας ένα ιστορικό αρχείο αναγνώσεων αισθητήρων. Αυτή η βάση δεδομένων χρησιμεύει ως η ραχοκοκαλιά του συστήματος, υποστηρίζοντας όχι μόνο την αποθήκευση δεδομένων αλλά και εξελιγμένες δυνατότητες ανάλυσης και ερωτημάτων. Η επεξεργασία από την πλευρά του διακομιστή μπορεί να περιλαμβάνει επικύρωση, καθαρισμό και συγκέντρωση δεδομένων, μετατρέποντας τις ακατέργαστες αναγνώσεις του αισθητήρα σε δομημένα, αξιόπιστα σύνολα δεδομένων. Επιπλέον, ο διακομιστής είναι υπεύθυνος για τη διαχείριση της πρόσβασης των χρηστών, διασφαλίζοντας ότι τα δεδομένα είναι προσβάσιμα σε εξουσιοδοτημένους χρήστες, διατηρώντας παράλληλα τα πρότυπα απορρήτου και ασφάλειας. Προηγμένες τεχνικές επεξεργασίας δεδομένων, όπως αλγόριθμοι μηχανικής μάθησης ή στατιστικά μοντέλα, μπορούν να εφαρμοστούν για την εξαγωγή πληροφοριών, τον εντοπισμό ανωμαλιών ή την πρόβλεψη τάσεων, ενισχύοντας περαιτέρω την αξία των συλλεγόμενων δεδομένων.

Το τελευταίο συστατικό του συστήματος είναι η διεπαφή χρήστη, συνήθως μια διαδικτυακή εφαρμογή, σχεδιασμένη να παρουσιάζει τα επεξεργασμένα δεδομένα με διαισθητικό και προσιτό τρόπο. Αυτή η διεπαφή ιστού έχει σχεδιαστεί προσεκτικά για να καλύψει τις ανάγκες τόσο των επιδοτούμενων όσο και των μη επιδοτούμενων χρηστών, προσφέροντας προσαρμοσμένες απόψεις και λειτουργίες. Οι χρήστες μπορούν να αλληλεπιδράσουν με το σύστημα μέσω πινάκων εργαλείων, γραφημάτων και αναφορών, οι οποίες αντικατοπτρίζουν δυναμικά τις τρέχουσες και ιστορικές καταστάσεις του περιβάλλοντος που παρακολουθείται. Λειτουργίες όπως η ροή δεδομένων σε πραγματικό χρόνο, οι προσαρμόσιμες ειδοποιήσεις και τα λεπτομερή αναλυτικά εργαλεία δίνουν στους χρήστες τη δυνατότητα να κατανοούν και να αντιδρούν αποτελεσματικά στα δεδομένα. Για επιδοτούμενους χρήστες, η διεπαφή ενδέχεται να περιλαμβάνει πρόσθετες λειτουργίες ή πληροφορίες δεδομένων, παρέχοντας βελτιωμένες δυνατότητες σύμφωνα με την υποστήριξη ή τους στόχους της οντότητας που επιδοτεί. Τα ζητήματα προσβασιμότητας, ο αποκριτικός σχεδιασμός και οι σαφείς απεικονίσεις είναι κεντρικές για τη σχεδίαση της διεπαφής, διασφαλίζοντας ότι οι χρήστες μπορούν να πλοηγηθούν εύκολα στο σύστημα και να αντλήσουν ουσιαστικές πληροφορίες από τα δεδομένα του αισθητήρα, ανεξάρτητα από την τεχνική τους εμπειρία ή τη συσκευή επιλογής.

Στην ουσία, το σύστημα προσφέρει μια ολιστική προσέγγιση στη διαχείριση δεδομένων αισθητήρων, μετατρέποντας τις ακατέργαστες μετρήσεις σε χρήσιμες πληροφορίες για μια διαφορετική βάση χρηστών. Μέσω της προσεκτικής ενσωμάτωσης της τεχνολογίας αισθητήρων, της επεξεργασίας από την πλευρά του διακομιστή και της οπτικοποίησης εστιασμένης στον χρήστη, το σύστημα διασφαλίζει ότι τα πολύτιμα δεδομένα όχι μόνο συλλέγονται και αποθηκεύονται, αλλά και καθίστανται προσβάσιμα και χρήσιμα, υποστηρίζοντας τη λήψη τεκμηριωμένων αποφάσεων και ενισχύοντας τη βαθύτερη κατανόηση των παρακολουθούμενων περιβάλλοντα.



Εικόνα 4.2: Το σύστημα



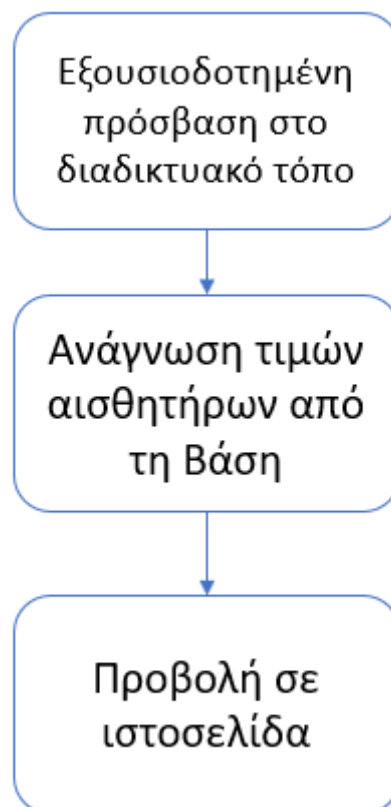
Εικόνα 4.3: Το σύστημα – μέτρηση και αποστολή

Η διαδικασία διαχείρισης δεδομένων αισθητήρα περιλαμβάνει πολλά βασικά στάδια, ξεκινώντας από την προετοιμασία του αισθητήρα έως την ασφαλή μετάδοση και αποθήκευση των δεδομένων που συλλέγει, όπως φαίνεται στην Εικόνα 4.3. Αυτή η διαδικασία είναι κρίσιμης σημασίας σε διάφορους τομείς, συμπεριλαμβανομένης της περιβαλλοντικής παρακολούθησης, της υγειονομικής περίθαλψης, του βιομηχανικού αυτοματισμού και της διαχείρισης έξυπνων πόλεων, όπου τα ακριβή και αξιόπιστα δεδομένα είναι πρωταρχικής σημασίας για τη λήψη αποφάσεων και τη λειτουργική αποτελεσματικότητα.

Ξεκινάμε με την προετοιμασία του Raspberry, μια κρίσιμη φάση όπου ο Raspberry είναι έτοιμος να ξεκινήσει τη λειτουργία του. Αυτό περιλαμβάνει τη βαθμονόμηση του αισθητήρα, τη ρύθμιση των αρχικών παραμέτρων και τη διασφάλιση ότι είναι σε κατάσταση ετοιμότητας για να ξεκινήσει η συλλογή δεδομένων. Η σωστή αρχικοποίηση είναι ζωτικής σημασίας καθώς επηρεάζει άμεσα την ακρίβεια και την αξιοπιστία των μετρήσεων. Μόλις αρχικοποιηθεί, ο αισθητήρας εισέρχεται στη φάση μέτρησης, όπου αρχίζει να συλλέγει δεδομένα από το περιβάλλον του. Αυτά τα δεδομένα μπορεί να είναι οτιδήποτε, από ενδείξεις θερμοκρασίας, επίπεδα υγρασίας, δείκτες ποιότητας αέρα. Η συχνότητα, η ανάλυση και το εύρος αυτών των μετρήσεων καθορίζονται με βάση τις δυνατότητες του αισθητήρα

και τις απαιτήσεις της εφαρμογής που εξυπηρετεί. Είναι επιτακτική ανάγκη αυτό το στάδιο να χαρακτηρίζεται από ακρίβεια και συνέπεια, καθώς η ποιότητα των δεδομένων που συλλέγονται εδώ καθορίζει τον τόνο για όλες τις επόμενες ενέργειες.

Μετά τη συλλογή των δεδομένων, τα επόμενα βασικά στάδια είναι η ασφαλής μετάδοση των δεδομένων σε έναν διακομιστή και η επακόλουθη αποθήκευση τους σε μια βάση δεδομένων. Η φάση μετάδοσης περιλαμβάνει την αποστολή των δεδομένων από τον αισθητήρα σε έναν κεντρικό διακομιστή ή υποδομή cloud. Αυτό το βήμα πρέπει να αντιμετωπίζεται με ασφάλεια για την προστασία των δεδομένων από μη εξουσιοδοτημένη πρόσβαση ή παραποίηση, ειδικά εάν τα δεδομένα είναι ευαίσθητα ή κρίσιμης φύσης. Τεχνικές όπως κρυπτογράφηση δεδομένων, ασφαλή πρωτόκολλα όπως HTTPS ή MQTT μέσω SSL και ισχυροί μηχανισμοί ελέγχου ταυτότητας χρησιμοποιούνται για τη διασφάλιση της εμπιστευτικότητας, της ακεραιότητας και της αυθεντικότητας των δεδομένων κατά τη μεταφορά. Μόλις τα δεδομένα φτάσουν στον προορισμό τους, αποθηκεύονται σε μια βάση δεδομένων.



Εικόνα 4.4: Το σύστημα – προβολή

Η επιλογή της βάσης δεδομένων—είτε είναι SQL, NoSQL, χρονοσειρές ή οποιοσδήποτε άλλος τύπος—εξαρτάται από τη φύση και τη δομή των δεδομένων, καθώς και από τις απαιτήσεις απόδοσης του συστήματος. Η φάση αποθήκευσης δεν αφορά μόνο τη διατήρηση των δεδομένων, αλλά και τη διασφάλιση ότι είναι οργανωμένα, προσβάσιμα και ευρετηριασμένα για αποτελεσματική ανάκτηση και ανάλυση. Οι κατάλληλες πρακτικές διαχείρισης βάσεων δεδομένων, τα τακτικά αντίγραφα ασφαλείας

και η συμμόρφωση με τους κανονισμούς προστασίας δεδομένων αποτελούν αναπόσπαστο κομμάτι αυτής της φάσης, διασφαλίζοντας ότι τα δεδομένα δεν αποθηκεύονται απλώς αλλά διατηρούνται με τρόπο που υποστηρίζει την αποτελεσματική τους χρήση στην ανάλυση, τη λήψη αποφάσεων και άλλες μεταγενέστερες διαδικασίες.

Η εφαρμογή εξουσιοδοτημένης πρόσβασης σε έναν ιστότοπο που διαβάζει τιμές αισθητήρων από μια βάση δεδομένων και τις εμφανίζει διασφαλίζει ότι τα ευαίσθητα δεδομένα προστατεύονται και ότι οι πληροφορίες παρουσιάζονται με τρόπο φιλικό προς το χρήστη. Σε μια τέτοια ρύθμιση, ο ιστότοπος χρησιμεύει ως διεπαφή μεταξύ του τελικού χρήστη και των υποκείμενων δεδομένων αισθητήρων, απαιτώντας ισχυρά μέτρα ασφαλείας για την αποτροπή μη εξουσιοδοτημένης πρόσβασης και παραβιάσεων δεδομένων. Ο έλεγχος ταυτότητας χρήστη είναι ο ακρογωνιαίος λίθος αυτού του πλαισίου ασφαλείας. Συνήθως, αυτό περιλαμβάνει τη δημιουργία ενός συστήματος σύνδεσης όπου οι χρήστες παρέχουν διαπιστευτήρια, τα οποία στη συνέχεια επαληθεύονται έναντι των αποθηκευμένων δεδομένων σε μια ασφαλή βάση δεδομένων. Οι σύγχρονες διαδικτυακές εφαρμογές χρησιμοποιούν συχνά αλγόριθμους κατακερματισμού για την ασφαλή αποθήκευση των κωδικών πρόσβασης και μπορούν επίσης να εφαρμόσουν πρόσθετα επίπεδα ασφαλείας, όπως ο έλεγχος ταυτότητας πολλαπλών παραγόντων (MFA) για περαιτέρω προστασία της πρόσβασης. Μόλις συνδεθούν, οι χρήστες λαμβάνουν δικαιώματα βάσει του ρόλου τους, διασφαλίζοντας ότι μπορούν να έχουν πρόσβαση μόνο στα δεδομένα και τις λειτουργίες που σχετίζονται με τις ανάγκες τους.

Όσον αφορά την παρουσίαση δεδομένων, η ανάγνωση των τιμών των αισθητήρων από μια βάση δεδομένων και η εμφάνισή τους στον ιστότοπο απαιτεί απρόσκοπτη και αποτελεσματική ροή δεδομένων από backend-to-frontend. Το backend, που συνήθως κατασκευάζεται με μια γλώσσα διακομιστή όπως Python, Node.js ή PHP, είναι υπεύθυνο για την αναζήτηση στη βάση δεδομένων, την επεξεργασία των δεδομένων του αισθητήρα και την προβολή τους στο frontend. Το frontend, κατασκευασμένο με HTML, CSS και JavaScript, ή το flask όπως το χρησιμοποιήσαμε εμείς, χρησιμοποιώντας ίσως βιβλιοθήκες όπως το Chart.js για οπτικοποίηση δεδομένων, στη συνέχεια παρουσιάζει αυτά τα δεδομένα σε μια διαισθητική και διαδραστική μορφή. Για την εμφάνιση δεδομένων σε πραγματικό χρόνο, τεχνολογίες όπως WebSockets ή συμβάντα που αποστέλλονται από διακομιστή μπορούν να χρησιμοποιηθούν για τη δημιουργία μιας μόνιμης σύνδεσης μεταξύ του διακομιστή και του πελάτη, επιτρέποντας στον διακομιστή να προωθήσει ενημερώσεις στον πελάτη μόλις είναι διαθέσιμα νέα δεδομένα αισθητήρα. Αυτό διασφαλίζει ότι ο χρήστης προβάλλει πάντα τις πιο ενημερωμένες πληροφορίες χωρίς να χρειάζεται να ανανεώσει τη σελίδα με μη αυτόματο τρόπο. Ο συνδυασμός ενός ασφαλούς συστήματος πρόσβασης για συγκεκριμένο χρήστη και ενός ισχυρού επιπέδου παρουσίασης δεδομένων σε πραγματικό χρόνο οδηγεί σε ένα ισχυρό εργαλείο για την παρακολούθηση και την ανάλυση δεδομένων αισθητήρων, παρέχοντας στους χρήστες πολύτιμες πληροφορίες και τη δυνατότητα να ανταποκρίνονται άμεσα σε οποιεσδήποτε αλλαγές στα δεδομένα .

## 4.2 Η εφαρμογή - ιστοσελίδα

Ο χρήστης πρέπει να συνδεθεί στο σύστημα αφού πατήσει το κουμπί σύνδεση όπως φαίνεται στο παρακάτω Εικόνα 4.5. Στο υποκεφάλαιο αυτό θα περιγραφεί η εφαρμογή-ιστοσελίδα του συστήματος.

# Σύνδεση για Μέτρηση ποιότητας αέρα

Username:

Password:

Εικόνα 4.5: Η σελίδα για της σύνδεση

Ο χρήστης που έχει συνδεθεί μπορεί να δει τη σελίδα με τις μετρήσεις, με το username και το δωμάτιο του γραμμένο, όπως φαίνεται στην Εικόνα 4.6.

**User:dimitris**

**Room:Εργαστήριο Δ1**

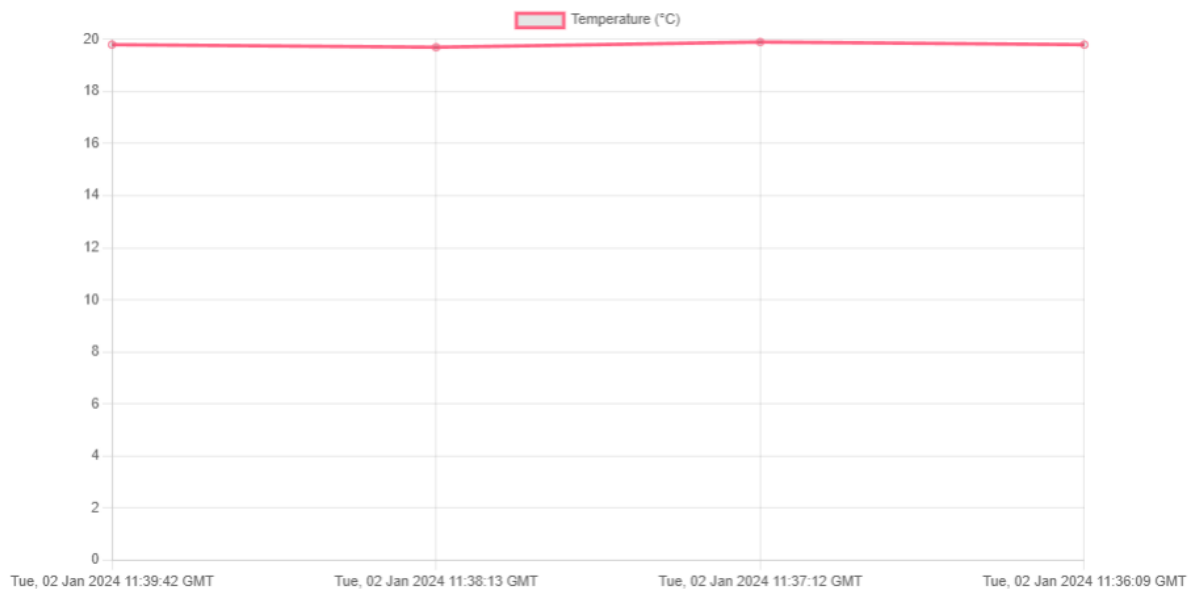
[Logout](#)

Εικόνα 4.6: Προβολή στοιχείων χρήστη και δωματίου

Στις επόμενες Εικόνες φαίνεται η σελίδα με τα γραφήματα από τις μετρήσεις.

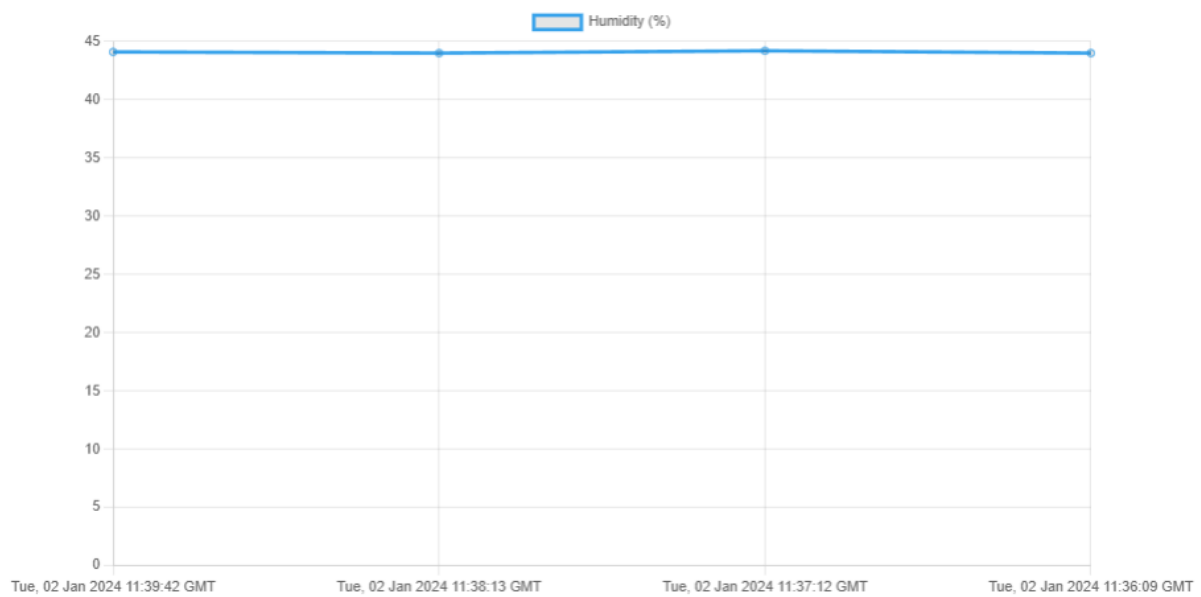
Η σελίδα έχει υλοποιηθεί με Flask και τα γραφήματα με javascript μέσω Chart.js.

## Temperature Data



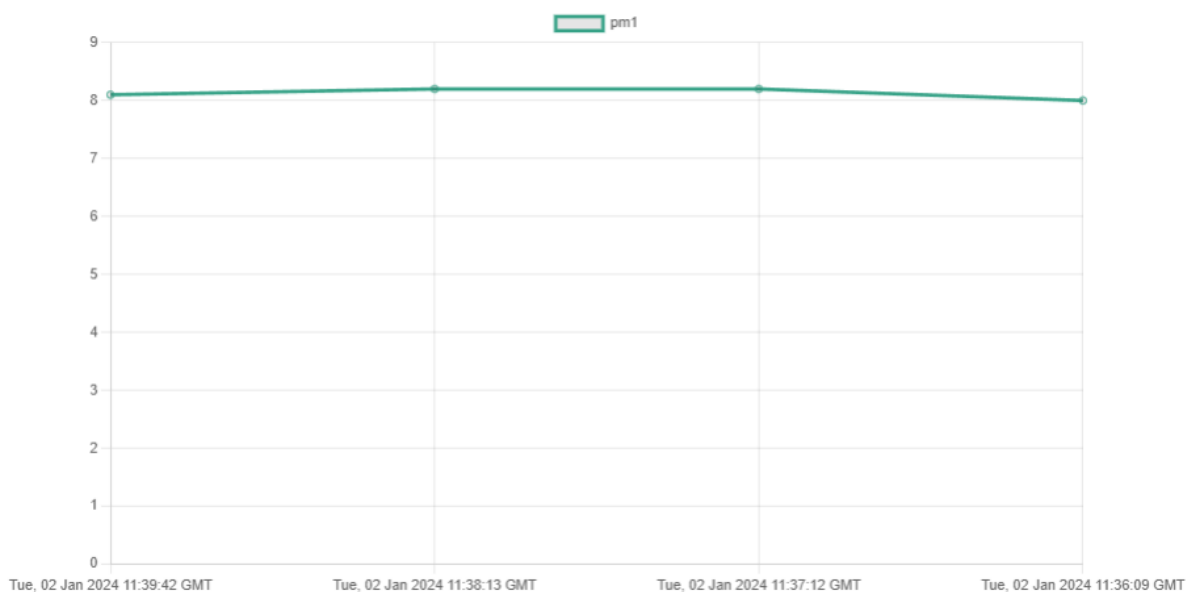
Εικόνα 4.7: Γράφημα με μετρήσεις Θερμοκρασίας

## Humidity Data



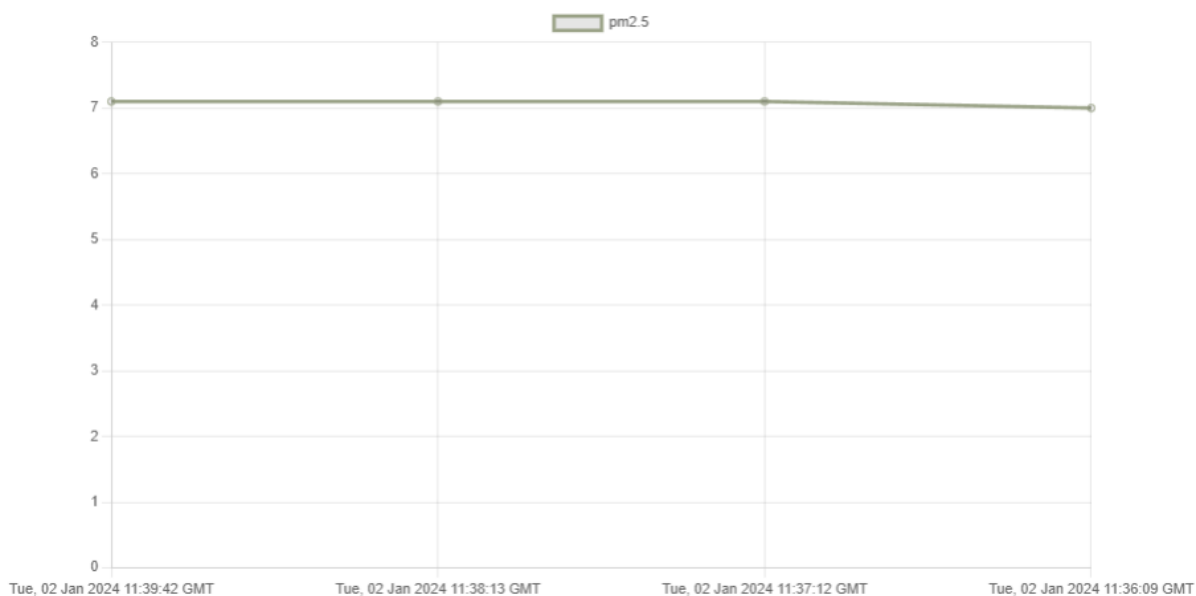
Εικόνα 4.8: Γράφημα με μετρήσεις Υγρασίας

## PM1.0 Data



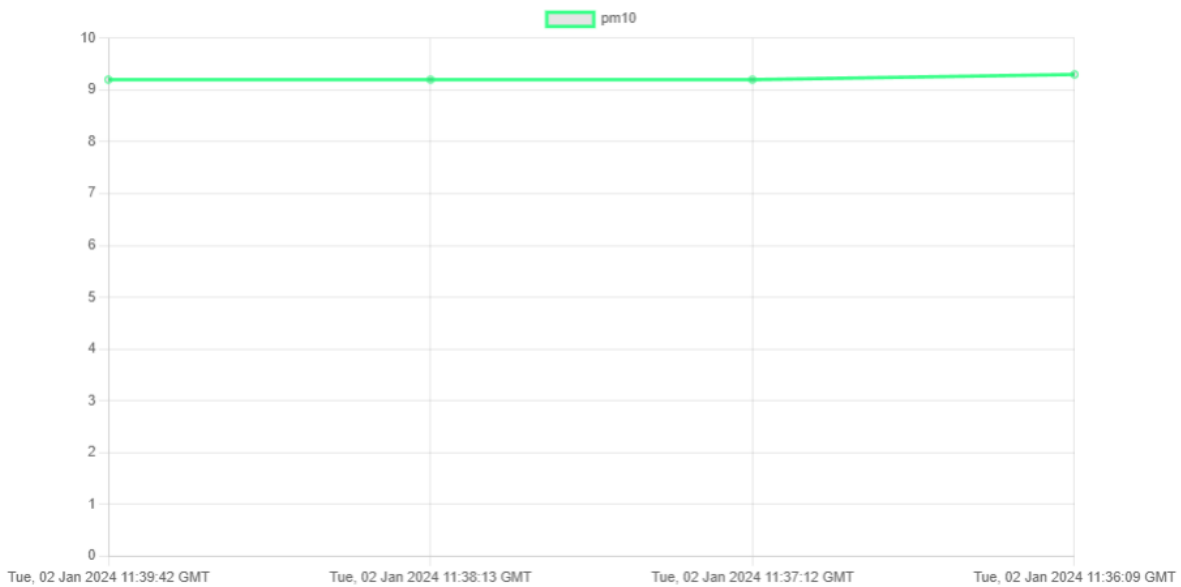
Εικόνα 4.9: Γράφημα με μετρήσεις σωματιδίων PM1.0

## PM2.5 Data



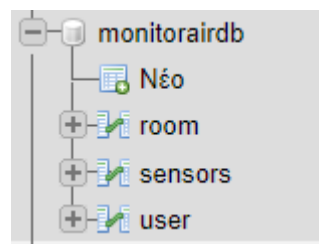
Εικόνα 4.10: Γράφημα με μετρήσεις σωματιδίων PM2.5

## PM10 Data



Εικόνα 4.11: Γράφημα με μετρήσεις σωματιδίων PM10

### 4.3 Η Βάση



Εικόνα 4.12: Η βάση monitorairdb με τους πίνακες

Η βάση δεδομένων κατασκευάστηκε χρησιμοποιώντας MySQL MariaDB και η δομή της, συμπεριλαμβανομένων των χρησιμοποιούμενων πινάκων, απεικονίζεται στην Εικόνα 4.24. Τα επόμενα σχήματα παρέχουν μια λεπτομερή προβολή των πινάκων, περιγράφοντας τα συγκεκριμένα πεδία που περιέχονται σε καθένα. Κάθε πίνακας είναι δομημένος γύρω από ένα πρωτεύον κλειδί, το πεδίο 'id', το οποίο χρησιμεύει ως μοναδικό αναγνωριστικό για εγγραφές και είναι ένας σειριακός αριθμός. Το πεδίο «ενεργό» είναι ενσωματωμένο για να υποδηλώνει την τρέχουσα κατάσταση μιας εγγραφής, υποδεικνύοντας εάν είναι ενεργή. Επιπλέον, διατηρούνται χρονικές σημάνσεις για κάθε εγγραφή μέσω του πεδίου "created", το οποίο καταγράφει τον ακριβή χρόνο της αρχικής δημιουργίας μιας εγγραφής.

#	Όνομα	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Σχόλια	Πρόσθετα
<input type="checkbox"/>	1	<b>id</b>	int(11)		Όχι	Καμία		AUTO_INCREMENT
<input type="checkbox"/>	2	<b>username</b>	varchar(30)	utf8mb4_general_ci	Όχι	Καμία		
<input type="checkbox"/>	3	<b>password</b>	varchar(30)	utf8mb4_general_ci	Όχι	Καμία		
<input type="checkbox"/>	4	<b>lastname</b>	varchar(30)	utf8mb4_general_ci	Όχι	Καμία		
<input type="checkbox"/>	5	<b>firstname</b>	varchar(30)	utf8mb4_general_ci	Όχι	Καμία		
<input type="checkbox"/>	6	<b>kind</b>	tinyint(4)		Όχι	Καμία		
<input type="checkbox"/>	7	<b>created</b>	datetime		Όχι	current_timestamp()		

Εικόνα 4.13: Η δομή του πίνακα User

				id	username	password	lastname	firstname	kind	created		
<input type="checkbox"/>		Επεξεργασία		Αντιγραφή		Διαγραφή	1	dimitris	123456	Καραγκούνης Δημήτρης	1	2024-02-02 11:29:30
<input type="checkbox"/>		Επεξεργασία		Αντιγραφή		Διαγραφή	2	kostas	123456	Παπάς Κώστας	1	2024-02-02 11:29:30

Εικόνα 4.14: Τα περιεχόμενα του πίνακα User

#	Όνομα	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Σχόλια	Πρόσθετα
<input type="checkbox"/>	1	<b>id</b>	int(11)		Όχι	Καμία		AUTO_INCREMENT
<input type="checkbox"/>	2	<b>enable</b>	tinyint(4)		Όχι	1		
<input type="checkbox"/>	3	<b>userid</b>	int(11)		Όχι	Καμία		
<input type="checkbox"/>	4	<b>roomid</b>	int(11)		Όχι	Καμία		
<input type="checkbox"/>	5	<b>temp</b>	double		Ναι	NULL		
<input type="checkbox"/>	6	<b>hum</b>	double		Ναι	NULL		
<input type="checkbox"/>	7	<b>pm1</b>	double		Ναι	NULL		
<input type="checkbox"/>	8	<b>pm2_5</b>	double		Όχι	Καμία		
<input type="checkbox"/>	9	<b>pm10</b>	double		Όχι	Καμία		
<input type="checkbox"/>	10	<b>created</b>	datetime		Όχι	current_timestamp()		

Εικόνα 4.15: Η δομή του πίνακα Sensor

				id	enable	userid	roomid	temp	hum	pm1	pm2_5	pm10	created		
<input type="checkbox"/>		Επεξεργασία		Αντιγραφή		Διαγραφή	1	1	1	19.8	44	8	7	9.3	2024-01-02 11:36:09
<input type="checkbox"/>		Επεξεργασία		Αντιγραφή		Διαγραφή	2	1	1	19.9	44.2	8.2	7.1	9.2	2024-01-02 11:37:12
<input type="checkbox"/>		Επεξεργασία		Αντιγραφή		Διαγραφή	3	1	1	19.7	44	8.2	7.1	9.2	2024-01-02 11:38:13
<input type="checkbox"/>		Επεξεργασία		Αντιγραφή		Διαγραφή	4	1	1	19.8	44.1	8.1	7.1	9.2	2024-01-02 11:39:42

Εικόνα 4.16: Τα περιεχόμενα του πίνακα Sensor

#	Όνομα	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Σχόλια	Πρόσθετα
<input type="checkbox"/>	1	id	int(11)		Όχι	Καμία		AUTO_INCREMENT
<input type="checkbox"/>	2	userid	int(11)		Όχι	Καμία		
<input type="checkbox"/>	3	roomid	int(11)		Όχι	Καμία		
<input type="checkbox"/>	4	roomname	varchar(30) utf8mb4_general_ci		Όχι	Καμία		

Εικόνα 4.17: Η δομή του πίνακα Room

				id	userid	roomid	roomname			
<input type="checkbox"/>		Επεξεργασία		Αντιγραφή		Διαγραφή	1	1	1	Εργαστήριο Δ1
<input type="checkbox"/>		Επεξεργασία		Αντιγραφή		Διαγραφή	2	2	2	Αίθουσα Β3

Εικόνα 4.18: Τα περιεχόμενα του πίνακα Room

#### 4.4 Ασφάλεια πρόσβασης στο σύστημα και στα δεδομένα

Η ασφάλεια των κωδικών πρόσβασης του ιστότοπου είναι ζωτικής σημασίας για τον έλεγχο ταυτότητας των χρηστών. Η πρόσβαση στο σύστημα παρέχεται μόνο όταν οι χρήστες εισάγουν με ακρίβεια τα διαπιστευτήριά τους, συγκεκριμένα το email και τον κωδικό πρόσβασής τους. Μετά την επιτυχή είσοδο, ξεκινά μια συνεδρία, διατηρώντας αποτελεσματικά τις πληροφορίες του χρήστη με ασφάλεια.

Η ασφάλεια που βασίζεται σε περιόδους λειτουργίας διαδραματίζει κεντρικό ρόλο στον έλεγχο ταυτότητας χρήστη και στην αλληλεπίδραση με το σύστημα. Κάθε δραστηριότητα χρήστη, είτε κάνει κλικ σε ένα κουμπί για μια συγκεκριμένη εργασία είτε ξεκινά οποιαδήποτε άλλη απαιτούμενη ενέργεια, επικυρώνεται σχολαστικά μέσω της συνεδρίας που καθιερώθηκε κατά τη σύνδεση. Αυτή η περίοδος λειτουργίας, που στεγάζεται με ασφάλεια στον διακομιστή, είναι χρονικά δεσμευμένη, διασφαλίζοντας ότι οι ενέργειες επικυρώνονται και επαληθεύονται εντός ενός καθορισμένου χρονικού πλαισίου.

Η διασφάλιση της επικοινωνίας δεδομένων είναι πρωταρχικής σημασίας, ειδικά μέσω του Διαδικτύου, και επιτυγχάνεται μέσω της κρυπτογράφησης SSL (Secure Sockets Layer) [17]. Αυτό το πρωτόκολλο διασφαλίζει ένα προστατευμένο κανάλι μεταξύ του χρήστη και του ιστότοπου του συστήματος, κρυπτογραφώντας όλα τα μεταδιδόμενα δεδομένα. Με την κρυπτογράφηση των πληροφοριών που ανταλλάσσονται μεταξύ των δύο συσκευών, το SSL δημιουργεί μια ενισχυμένη σύνδεση μέσω του Διαδικτύου, διασφαλίζοντας τη μετάδοση ευαίσθητων δεδομένων και ενισχύοντας τη συνολική ασφάλεια της επικοινωνίας.

## Κεφάλαιο 5ο: Συμπεράσματα και προτάσεις βελτίωσης

Το σύστημα μέτρησης ποιότητας αέρα σε πραγματικό χρόνο παρέχει μια πολύτιμη υπηρεσία στην παρακολούθηση και ανάλυση των περιβαλλοντικών συνθηκών σε κλειστούς χώρους. Αξιοποιώντας τη δύναμη του Raspberry Pi 3, μαζί με τους αισθητήρες PMS5003 και DHT11, αυτό το σύστημα παρέχει με συνέπεια κρίσιμα δεδομένα σχετικά με τις συγκεντρώσεις σωματιδίων, τη θερμοκρασία και την υγρασία, απαραίτητα για τη διατήρηση της υγείας και της άνεσης των ανθρώπων. Η ολοκληρωμένη προσέγγισή του, η οποία περιλαμβάνει τη μετάδοση δεδομένων σε διακομιστή, την ασφαλή αποθήκευση σε μια βάση δεδομένων και τη όμορφη παρουσίαση σε μια πλατφόρμα web, περιλαμβάνει μια ολιστική λύση για διαχείριση και ανάλυση δεδομένων. Αυτή η τεχνολογική ολοκλήρωση ανεβάζει σημαντικά το βιοτικό επίπεδο. Η καλά σχεδιασμένη αρχιτεκτονική του συστήματος διασφαλίζει την ασφαλή και αξιόπιστη απόκτηση δεδομένων, ενώ η διεπαφή οπτικοποίησης που βασίζεται στο web καθιστά σύνθετες πληροφορίες προσβάσιμες και εύπεπτες, εξυπηρετώντας τους χρήστες σε διάφορα επίπεδα. Να προστεθεί ότι η συμβολή του chatgpt βοήθησε στη βελτίωση του περιεχομένου της εργασίας.

Ωστόσο, υπάρχουν πάντα ευκαιρίες για βελτίωση. Η καινοτομία ενός εκλεπτυσμένου μηχανισμού για την επικύρωση και τον καθαρισμό δεδομένων θα ενισχύσει την ακρίβεια και την αξιοπιστία των μετρήσεων. Η αξιοποίηση των δυνατοτήτων των αλγορίθμων μηχανικής μάθησης ή των στατιστικών μοντέλων θα μπορούσε να αποκαλύψει πληροφορίες για πρόβλεψη τάσεων ή ανίχνευση ανωμαλιών, εμπλουτίζοντας έτσι την τιμή που προκύπτει από τα δεδομένα. Επιπλέον, η βελτίωση της διεπαφής χρήστη για τη βελτίωση της διαισθητικότητας και της προσαρμογής θα ενισχύσει την καλή πλοήγηση του χρήστη. Η εισαγωγή εξελιγμένων λειτουργιών για παρακολούθηση σε πραγματικό χρόνο, όπως εξατομικευμένες ειδοποιήσεις και ολοκληρωμένα αναλυτικά εργαλεία, θα μετατρέψει το σύστημα σε μια πιο προληπτική και ισχυρή πλατφόρμα. Επιπλέον, η ενίσχυση των μέτρων ασφαλείας των δεδομένων και η ενίσχυση των ελέγχων πρόσβασης θα διασφαλίσουν ότι η ακεραιότητα και η εμπιστευτικότητα των δεδομένων είναι αδιαπραγμάτευτη, προσβάσιμη αποκλειστικά σε πιστοποιημένους χρήστες.

Το σύστημα μέτρησης της ποιότητας του αέρα σε πραγματικό χρόνο είναι εφοδιασμένο με μια εκτεταμένη σειρά λειτουργιών, που αποδεικνύεται ως απαραίτητο στοιχείο για την περιβαλλοντική παρακολούθηση. Η ικανότητά του για εξέλιξη, μέσω συνεχών βελτιώσεων και εισαγωγής καινοτόμων χαρακτηριστικών, υπόσχεται όχι απλώς να διατηρήσει αλλά και να αυξήσει σημαντικά τη χρησιμότητα και την αποτελεσματικότητά του.

## ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] <https://www.techtarget.com/iotagenda/definition/Internet-of-Things-IoT>
- [2] <https://www.adafruit.com/product/3686>
- [3] [https://www.rigacci.org/wiki/doku.php/doc/appunti/hardware/raspberrypi\\_air](https://www.rigacci.org/wiki/doku.php/doc/appunti/hardware/raspberrypi_air)
- [4] <https://github.com/pimoroni/pms5003-python>
- [5] <https://www.circuitbasics.com/how-to-set-up-the-dht11-humidity-sensor-on-the-raspberry-pi/>
- [6] <https://pimylifeup.com/raspberry-pi-dht11-sensor/>
- [7] <https://www.raspberrypi.com/products/raspberry-pi-3-model-b-plus/>
- [8] <https://i.stack.imgur.com/VEBEs.png>
- [9] <https://www.theengineeringprojects.com/2019/03/introduction-to-dht11.html>
- [10] [https://media.distrelec.com/Web/WebShopImages/landscape\\_large/3-/01/Adafruit-3686-30139173-01.jpg](https://media.distrelec.com/Web/WebShopImages/landscape_large/3-/01/Adafruit-3686-30139173-01.jpg)
- [11] <https://aloha.co/blog/why-python>
- [12] <https://flask.palletsprojects.com/en/3.0.x/>
- [13] <http://www.w3schools.com/mysql/mysql-tutorial>
- [14] [https://www.w3schools.com/html/html\\_scripts.asp](https://www.w3schools.com/html/html_scripts.asp)
- [15] <https://www.chartjs.org/>
- [16] <https://www.chartjs.org/docs/latest/charts/line.html>
- [17] <https://www.cloudflare.com/learning/ssl/what-is-ssl/>
- [18] <https://chat.openai.com>

## ΠΑΡΑΡΤΗΜΑ Α

Στο παράρτημα αυτό αναφέρονται τα βασικά κομμάτια του κώδικα που χρησιμοποιήθηκε.

Python - server

```
from flask import Flask, render_template, request, redirect, url_for, session, jsonify
import mysql.connector
from mysql.connector import Error
import hashlib

app = Flask(__name__)
app.secret_key = '1234554321'

def get_db_connection():
    connection = None
    try:
        connection = mysql.connector.connect(
            host='localhost',
            user='root',
            password='',
            database='monitorairdb'
        )
    except Error as e:
        print("Error while connecting to MySQL", e)
    return connection

@app.route('/')
#def index():
    #return render_template('index.html')

@app.route('/main')
def main():
    # Check if user is loggedin
    if 'loggedin' in session:
        return render_template('index.html', username=session['username'], room=session['roomname'])
    # User is not loggedin, redirect to login page
    return redirect(url_for('login'))

@app.route('/public')
def public():
    return render_template('index.html', room=1)
```

```

@app.route('/')
def index():
    if 'loggedin' in session:
        # User is logged in already, redirect to main page
        return redirect(url_for('main'))
    return redirect(url_for('login'))

@app.route('/data')
def data():
    connection = get_db_connection()
    if connection is not None and connection.is_connected():
        cursor = connection.cursor(dictionary=True)
        #cursor.execute('SELECT temp,hum,pm1,pm2_5,pm10,created FROM sensors ORDER BY created DESC
LIMIT 10')
        #cursor.execute('SELECT temp,hum,pm1,pm2_5,pm10,created FROM sensors WHERE roomid = %s ORDER
BY created DESC LIMIT 10',(session['roomsid'], ))
        cursor.execute('SELECT temp,hum,pm1,pm2_5,pm10,created FROM sensors WHERE roomid = 1 ORDER BY
created DESC LIMIT 10')
        records = cursor.fetchall()
        print(records)
        cursor.close()
        connection.close()
        return jsonify(records)
    else:
        return jsonify([])

@app.route('/login', methods=['GET', 'POST'])
def login():
    msg = ''
    if request.method == 'POST':
        username = request.form['username']
        password = request.form['password']

        # Hash the password for security
        hashed_password = password ;#hashlib.sha256(password.encode()).hexdigest()

        # Connect to the database
        conn = mysql.connector.connect(
            host='localhost',
            user='root',
            password='',
            database='monitorairdb')
        cursor = conn.cursor(dictionary=True)

        # Check if the account exists
        cursor.execute('SELECT * FROM user WHERE username = %s AND password = %s', (username,
hashed_password,))
        account = cursor.fetchone()

```

```

if account:
    # Create session data
    session['loggedin'] = True
    session['id'] = account['id']
    session['username'] = account['username']
    print(account['id'])

    #get rooms
    cursor2 = conn.cursor(dictionary=True)
    cursor2.execute('SELECT * FROM room WHERE userid = %s', (account['id'],))
    rooms = cursor2.fetchone()
    session['roomsid'] = rooms['id']
    session['roomname'] = rooms['roomname']

    # Redirect to main page
    return redirect(url_for('main'))
else:
    msg = 'Incorrect username/password!'
    cursor.close()
    cursor2.close()
    conn.close()
return render_template('login.html', msg=msg)

@app.route('/logout')
def logout():
    # Remove session data
    session.pop('loggedin', None)
    session.pop('id', None)
    session.pop('username', None)
    session.pop('roomsid', None)
    session.pop('roomname', None)
    # Redirect to login page
    return redirect(url_for('login'))

if __name__ == '__main__':
    app.run(host='0.0.0.0', port=105)

```

login.html

```

<!DOCTYPE html>
<html>
<head>
    <title>Σύνδεση</title>
</head>
<body>
    <h2>Σύνδεση για Μέτρηση ποιότητας αέρα</h2>

```

```

<form action="/login" method="post">
  <div>
    <label for="username">Username:</label>
    <input type="text" name="username" id="username" required>
  </div>
  <br>
  <div>
    <label for="password">Password:</label>
    <input type="password" name="password" id="password" required>
  </div>
  <input type="submit" value="Σύμβαση">
</form>
{% if msg %}
  <p>{{msg}}</p>
{% endif %}
</body>
</html>

```

index.html

```

<!DOCTYPE html>
<html>
<head>
  <title>Charts</title>
  <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>

  <style>
    .chart-container {
width: 1000px;
height: 600px
margin-bottom: 20px;
    }
  </style>
</head>
<body>
  <h1>User:{{ username }}</h1>
  <h3>Room:{{ room }}</h3>
  <a href="/logout">Logout</a>

  <div class="chart-container">

```

```

<h1>Temperature Data</h1>
<canvas id="temperatureChart"></canvas>
  </div>
  <div class="chart-container">
<h1>Humidity Data</h1>
<canvas id="humidityChart"></canvas>
  </div>
  <div class="chart-container">
<h1>PM1.0 Data</h1>
<canvas id="pm1Chart"></canvas>
  </div>
  <div class="chart-container">
<h1>PM2.5 Data</h1>
<canvas id="pm2_5Chart"></canvas>
  </div>
  <div class="chart-container">
<h1>PM10 Data</h1>
<canvas id="pm10Chart"></canvas>
  </div>

<script>
  const tempCtx = document.getElementById('temperatureChart').getContext('2d');
  const humidityCtx = document.getElementById('humidityChart').getContext('2d');
    const pm1Ctx = document.getElementById('pm1Chart').getContext('2d');
    const pm2_5Ctx = document.getElementById('pm2_5Chart').getContext('2d');
    const pm10Ctx = document.getElementById('pm10Chart').getContext('2d');

  const temperatureChart = new Chart(tempCtx, {
    type: 'line',
    data: {
      labels: [], // Timestamps go here
      datasets: [{
        label: 'Temperature (°C)',
        data: [], // Temperature data goes here
        borderColor: 'rgb(255, 99, 132)',
      }]
    },
    options: {
      scales: {
        y: {

```

```

        beginAtZero: true
    }
}
});

const humidityChart = new Chart(humidityCtx, {
  type: 'line',
  data: {
    labels: [], // Timestamps go here
    datasets: [{
      label: 'Humidity (%)',
      data: [], // Humidity data goes here
      borderColor: 'rgb(54, 162, 235)',
    }]
  },
  options: {
    scales: {
      y: {
        beginAtZero: true
      }
    }
  }
});

const pm1Chart = new Chart(pm1Ctx, {
  type: 'line',
  data: {
    labels: [], // Timestamps go here
    datasets: [{
      label: 'pm1',
      data: [], //pm
      borderColor: 'rgb(54, 162, 135)',
    }]
  },
  options: {
    scales: {
      y: {
        beginAtZero: true
      }
    }
  }
});

```

```

    }
  }
});

    const pm2_5Chart = new Chart(pm2_5Ctx, {
type: 'line',
data: {
  labels: [], // Timestamps go here
  datasets: [{
    label: 'pm2.5',
    data: [], //pm
    borderColor: 'rgb(154, 162, 135)',
  }]
},
options: {
  scales: {
    y: {
      beginAtZero: true
    }
  }
}
});

    const pm10Chart = new Chart(pm10Ctx, {
type: 'line',
data: {
  labels: [], // Timestamps go here
  datasets: [{
    label: 'pm10',
    data: [], //pm
    borderColor: 'rgb(54, 262, 135)',
  }]
},
options: {
  scales: {
    y: {
      beginAtZero: true
    }
  }
}
});

```

```

async function fetchData() {
  const response = await fetch('/data');
  const data = await response.json();
  const timestamps = data.map(item => item.created);
  const temperatures = data.map(item => item.temp);
  const humidities = data.map(item => item.hum);
    const pm1s = data.map(item => item.pm1);
    const pm2_5s = data.map(item => item.pm2_5);
    const pm10s = data.map(item => item.pm10);

  temperatureChart.data.labels = timestamps;
  temperatureChart.data.datasets[0].data = temperatures;

  humidityChart.data.labels = timestamps;
  humidityChart.data.datasets[0].data = humidities;

    pm1Chart.data.labels = timestamps;
  pm1Chart.data.datasets[0].data = pm1s;

    pm2_5Chart.data.labels = timestamps;
  pm2_5Chart.data.datasets[0].data = pm2_5s;

    pm10Chart.data.labels = timestamps;
  pm10Chart.data.datasets[0].data = pm10s;

  temperatureChart.update();
  humidityChart.update();
    pm1Chart.update();
    pm2_5Chart.update();
    pm10Chart.update();
}

// Fetch data when the page loads
window.onload = fetchData;
</script>
</body>
</html>

```