

S O U P

SOILLESS CULTURE UPGRADE

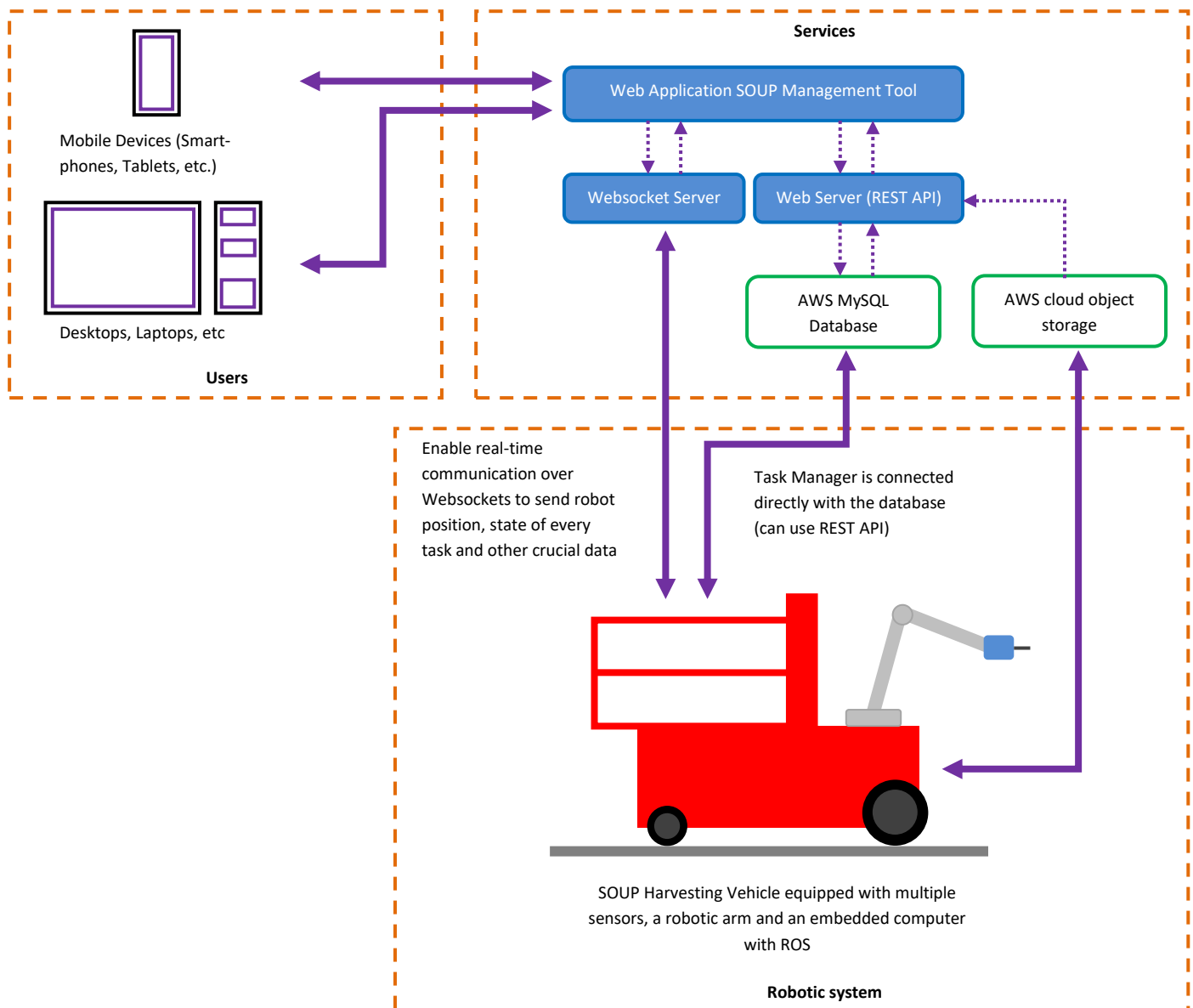
I

Ενιαίο λογισμικό ελέγχου και διεπαφής χρήστη, για την ρύθμιση όλων των παραμέτρων λειτουργίας του θερμοκηπίου και της υδροπονικής καλλιέργειας ντομάτας (SOUP)

SOUP Web Application Documentation

Το Soilless Culture Upgrade Management Tool είναι ένα πρωτότυπο σύστημα για τη διαχείριση, την παρακολούθηση και γενικότερα για τη σωστή διασύνδεση του αυτοματοποιημένου οχήματος διαχείρισης του θερμοκηπίου ντομάτας με τον χρήστη. Ο χρήστης μπορεί να διαχειριστεί, να επιβλέπει σε πραγματικό χρόνο και να εκτελεί τις διάφορες διαδικασίες που χρειάζεται ένα θερμοκήπιο με ντομάτες από οποιοδήποτε μέρος του κόσμου αρκεί να έχει πρόσβαση στο Internet. Όλο το σύστημα αποτελείται από την **εφαρμογή χρήστη** (Web Application), το web server, ο οποίος τρέχει το **REST API** για τη διαχείριση της βάσης δεδομένων και το web server για τη διασύνδεση του **Websocket Server**. Παρακάτω παρουσιάζεται η αρχιτεκτονική συστήματος με τη διασύνδεση με το υπόλοιπο ρομποτικό σύστημα. Τα κομμάτια που αποτελούν το σύστημα διαχείρισης, παρουσιάζονται με μπλε χρώμα.

Αρχιτεκτονική συστήματος



Τεχνολογίες Ανάπτυξης Συστήματος

Για την υλοποίηση του συστήματος χρησιμοποιήθηκαν αρκετές τεχνολογίες. Όλες οι τεχνολογίες χρησιμοποιήθηκαν και δουλεύουν ταυτόχρονα αλληλεπιδρώντας η μία με την άλλη για την εύκολη διαχείριση του συστήματος δίνοντας όσο το δυνατό περισσότερο έλεγχο στο χρήστη αυτοματοποιώντας ταυτόχρονα χρονοβόρες λειτουργίες. Οι τεχνολογίες που χρησιμοποιήθηκαν σε κάθε ένα από τα υποσυστήματα παρουσιάζονται παρακάτω.

- **Διαχειριστικό Περιβάλλον Χρήστη - Web Application**

1. Angular: Το framework της Angular χρησιμοποιήθηκε για τη δημιουργία της Web Εφαρμογής την οποία θα χρησιμοποιεί ο χρήστης και θα αλληλεπιδρά για τις εργασίες του θερμοκηπίου.
2. Typescript: Γλώσσα προγραμματισμού για την δημιουργία web εφαρμογών στο framework της Angular.
3. Bootstrap: Είναι η βιβλιοθήκη που προσφέρει για τη διαχείριση του DOM της εφαρμογής χρήστη.
4. jQuery: Βιβλιοθήκη που προσφέρει διάφορες λειτουργίες για τη σωστή λειτουργία της εφαρμογής χρήστη.

- **Web Server για το REST API (διασύνδεση βάσης δεδομένων με το Web App)**

1. Nodejs: Το περιβάλλον ανάπτυξης διαδικτυακών εφαρμογών για τις υπηρεσίες διαχείρισης της βάσης δεδομένων, συνδέσεων και άλλα.
2. Javascript: Γλώσσα προγραμματισμού για το περιβάλλον του Nodejs. Επίσης χρησιμοποιείται και δε κομμάτια της Web Εφαρμογής.
3. MySQL Database: Η βάση δεδομένων στην οποία αποθηκεύονται όλες οι σημαντικές πληροφορίες από τις διάφορες εργασίες που θα εκτελεί το όχημα. Ο όχημα και η εφαρμογή του χρήστη έχουν τη δυνατότητα να διαχειριστούν τη βάση δεδομένων.
4. JSON Web Tokens: Είναι το σύστημα το οποίο χρησιμοποιείται για την ασφαλή σύνδεση των χρηστών.
5. OpenCV.js: Επεξεργασία εικόνων για την παρουσίαση των αποτελεσμάτων για τον εντοπισμό καρπών.

- **Web Server για τη διασύνδεση μέσω Websockets (ενημερώσεις σε πραγματικό χρόνο)**

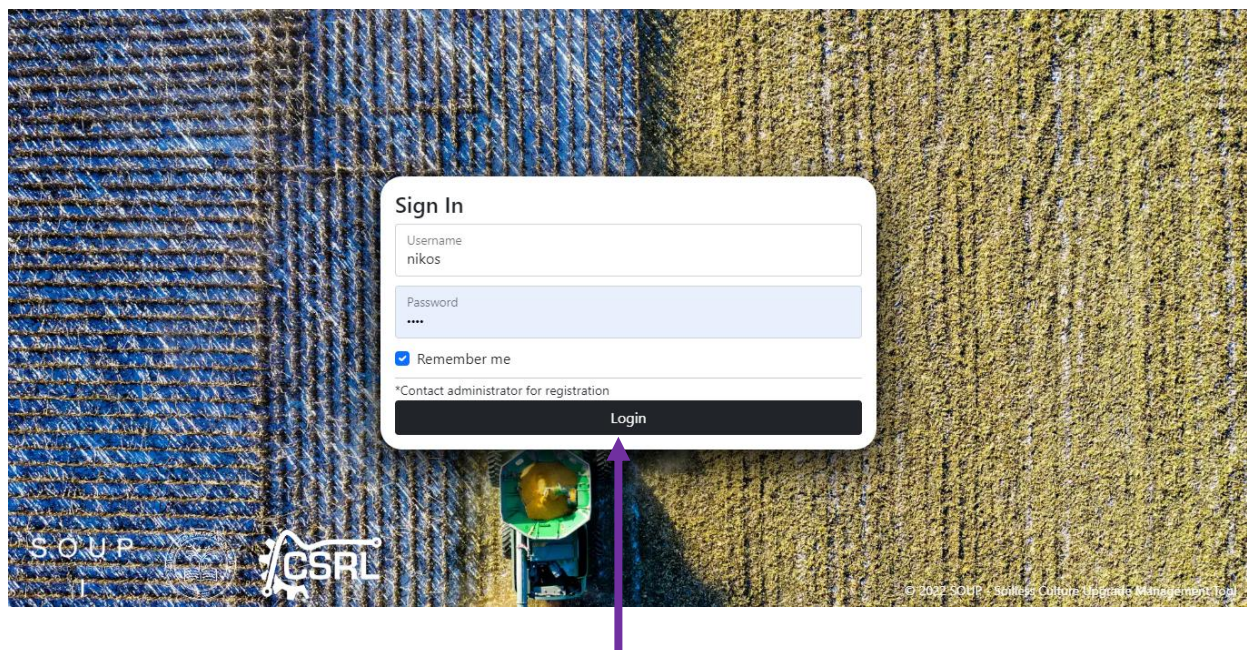
1. WebSocket Connection: Η σύνδεση με WebSocket χρησιμοποιείται για την άμεση ανταλλαγή μηνυμάτων του οχήματος προς την Web εφαρμογή του χρήστη σε πραγματικό χρόνο.

Ανάλυση Λειτουργίας Συστήματος

Εφαρμογή Χρήστη - Web Application

Αρχική Οθόνη - Είσοδος Χρήστη

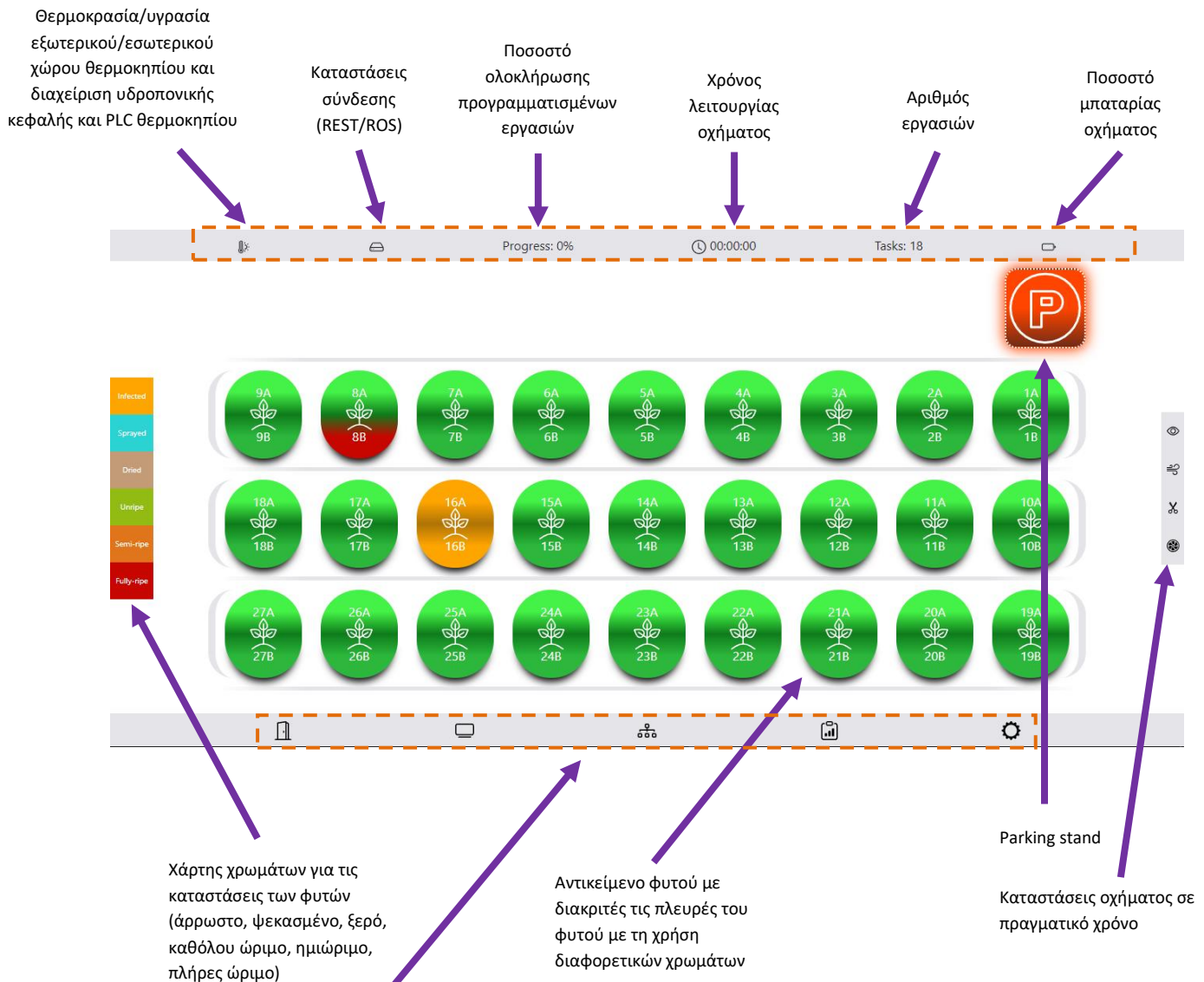
Παρακάτω παρουσιάζεται η οθόνη σύνδεσης του χρήστη. Ο χρήστης και διαχειριστής των εργασιών του αυτοματοποιημένου οχήματος θερμοκηπίου, πρέπει να έχει τα κλειδιά για να συνδεθεί στο σύστημα. Με αυτό τον τρόπο παρέχεται το επίπεδο ασφαλείας, έτσι ώστε να μην υπάρχει μη εξουσιοδοτημένη είσοδος και διαχείριση του οχήματος. Ο χρήστης πρέπει να ανοίξει το παρακάτω σύδενδρο στον περιηγητή του έτσι ώστε να εκτελεστεί η Web εφαρμογή: <https://georgealexakis.github.io/soup-app/>



Πλαίσια εισόδου για credentials για είσοδο
χρήστη στην εφαρμογή

Παρακολούθηση Θερμοκηπίου

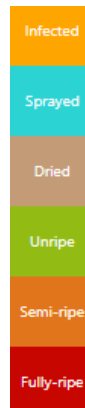
Παρακάτω παρουσιάζεται η οθόνη οπτικοποίησης των φυτών του θερμοκηπίου. Η παρακάτω διάταξη είναι η αναπαράσταση του πραγματικού θερμοκηπίου με διάφορα οπτικά αντικείμενα για την ευκολότερη και απλούστερη διαχείριση του θερμοκηπίου. Κάθε οβάλ αντικείμενο αντιπροσωπεύει ένα φυτό, στο οποίο έχουν διαχωριστεί οι πλευρές του με διαφορετικές αποχρώσεις. Ανάμεσα στα φυτά υπάρχουν οι διάδρομοι στους οποίους μπορεί να περάσει το όχημα και ενημερώνουν το χρήστη για την ακριβή τοποθεσία μέσα στο θερμοκήπιο. Η θέση στην οποία σταθμεύει το όχημα βρίσκεται πάνω δεξιά όπως και στο πραγματικό θερμοκήπιο. Για την κατάσταση των φυτών χρησιμοποιείται ένας χάρτης χρωμάτων όπως φαίνεται στην αριστερή πλευρά της παρακάτω εικόνας.



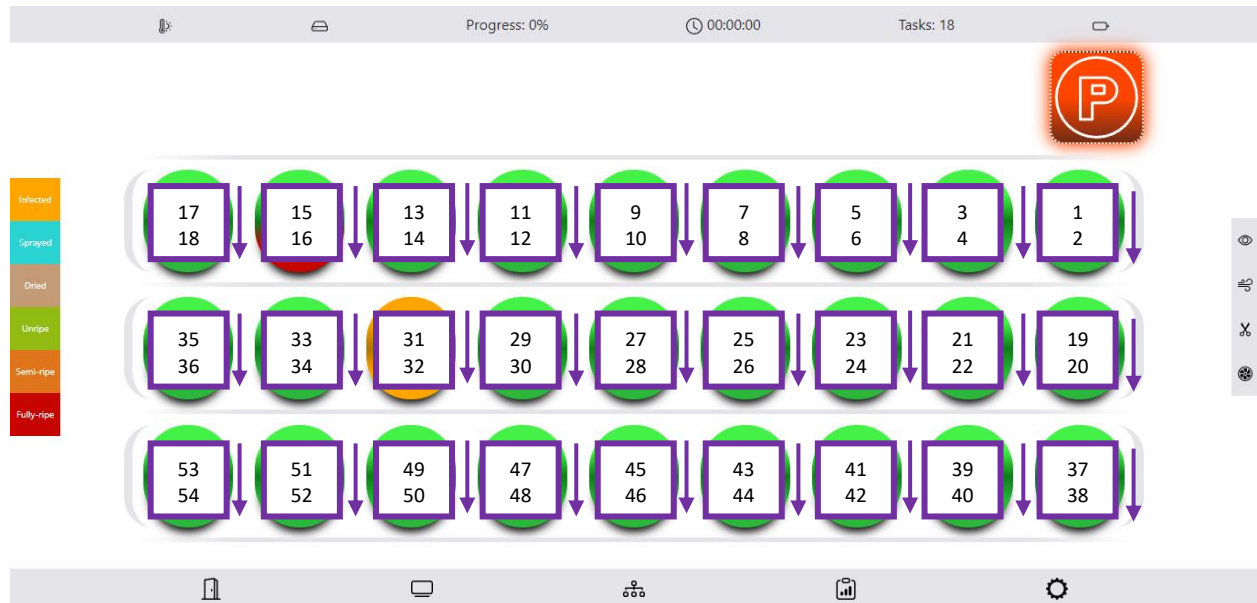
Navigation menu (logout, monitoring, task management, reports, settings)

Βασικό εργαλείο οπτικοποίησης των καταστάσεων των φυτών του θερμοκηπίου είναι μέσω της χρήσης των χρωμάτων. Στο κεντρικό πάνελ που παρουσιάζεται παραπάνω, φαίνεται αριστερά η χρωματική παλέτα, όπου παρουσιάζει τις καταστάσεις των φυτών. Κάθε φυτό μπορεί να έχει τις παρακάτω καταστάσεις:

- **Infected:** Το φυτό έχει κάποια ασθένεια, η οποία αναγνωρίστηκε κατά τη διάρκεια της εργασίας της επιθεώρησης. Όταν σε μια πλευρά του φυτού αναγνωριστεί μια ασθένεια τότε όλο το φυτό θεωρείται ότι έχει ασθένεια.
- **Sprayed:** Ένα φυτό θεωρείται ψεκασμένο όταν έχει ολοκληρωθεί η εργασία του ψεκασμού με κάποιο σκεύασμα. Ένα ψεκασμένο φυτό πρέπει να έχει ειδική μεταχείριση ως προς την συγκομιδή (μπορεί να είναι μόνο η μία πλευρά).
- **Dried:** Ξεραμένο φυτό όπου θέλει αντικατάσταση (μπορεί να είναι μόνο η μία πλευρά).
- **Unripe:** Φυτό με αγίνωτες ντομάτες (μπορεί να είναι μόνο η μία πλευρά).
- **Semi-ripe:** Φυτό με ντομάτες σε μέση κατάσταση ωρίμανσης (μπορεί να είναι μόνο η μία πλευρά).
- **Fully-ripe:** Φυτό με ώριμες ντομάτες οι οποίες είναι έτοιμες για συγκομιδή (μπορεί να είναι μόνο η μία πλευρά).



Κάθε φυτό αναπαρίσταται στη βάση ως δύο διαφορετικές εγγραφές, έτσι ώστε να υπάρχει ακριβέστερη καταγραφή των αποτελεσμάτων. Με αυτό τον τρόπο γίνεται δυνατή η διαχείριση των διαφορετικών πλευρών του κάθε φυτού. Παρακάτω παρουσιάζεται η αρίθμηση των φυτών όπως περιγράφεται στη βάση δεδομένων (με το **plantid** κάθε πλευράς).



Κάθε φυτό παρέχει διάφορες πληροφορίες στο χρήστη για τον αριθμό του καρπών, την ωριμότητα, τις ασθένειες και άλλες χρήσιμες πληροφορίες για τη γενικότερη κατάσταση των φυτών. Αυτές οι πληροφορίες εμφανίζονται πατώντας την κάθε πλευρά του κάθε φυτού. Παρακάτω παρουσιάζεται το πλαίσιο πληροφοριών ενός φυτού. Ο χρήστης έχει τη δυνατότητα να αλλάξει κάποιες πληροφορίες του κάθε φυτού.

Information Maturity Diseases

Plant ID: 8

Name: 88 (16)

Position (x, y): 0, 0

Last Gathering: January 1st 1999

Planting Date: January 1st 1999

Gathered Fruits: 0

Plant Type:

Plant Height (meters): 1

Update Close

Στοιχεία του φυτού που μπορούν να μεταβληθούν

Για την ωριμότητα των φυτών, συλλέγονται τα δεδομένα από τα predictions, όταν έχει πραγματοποιηθεί επιθεώρηση του κάθε φυτού και στο τέλος υπολογίζονται τα ποσοστά ωριμότητας επί του συνόλου των predictions. Παρακάτω παρουσιάζεται ο τύπος για τον υπολογισμό των ποσοστών:
$$ripeness (\%) = \frac{ripe_predictions}{total_predictions} * 100.$$

Information Maturity Diseases

Total Predictions:	61
Unripe Tomatoes:	22 (27.16%)
Semi-ripe Tomatoes:	11 (13.58%)
Fully-ripe Tomatoes:	48 (59.26%)
Unknown Predictions:	0 (0.00%)

*Unknown Predictions indicator consists of no predictions or unknown state of the predictions

Update Close

Οι ασθένειες των φυτών εντοπίζονται κατά τη όταν έχει ολοκληρωθεί η επιθεώρηση των φυτών. Το ανεπτυγμένο σύστημα μπορεί να αναλύσει τις εικόνες κατά την επιθεώρηση και να παρέχει πληροφορίες ασθενειών για κάθε φυτό.

#	Task	Plant	Growth Level	Disease	Disease Level	Pest	Pest Level	Deficiency Level	Confidence	Image	Mask URL
1058	1093	1		TUTA		2			NULL	1987	s3://souponserver-u2m/soup_data/upatras
1059	1094	1		TUTA		2			NULL	1988	s3://souponserver-u2m/soup_data/upatras

Αποτελέσματα
ανάλυσης
φυτών για
ασθένειες

Από το πλακάκι του parking stand μπορεί ο χρήστης να δει τις πληροφορίες του οχήματος, όπως επίσης να κάνει ανανέωση των πληροφοριών του πάνελ.

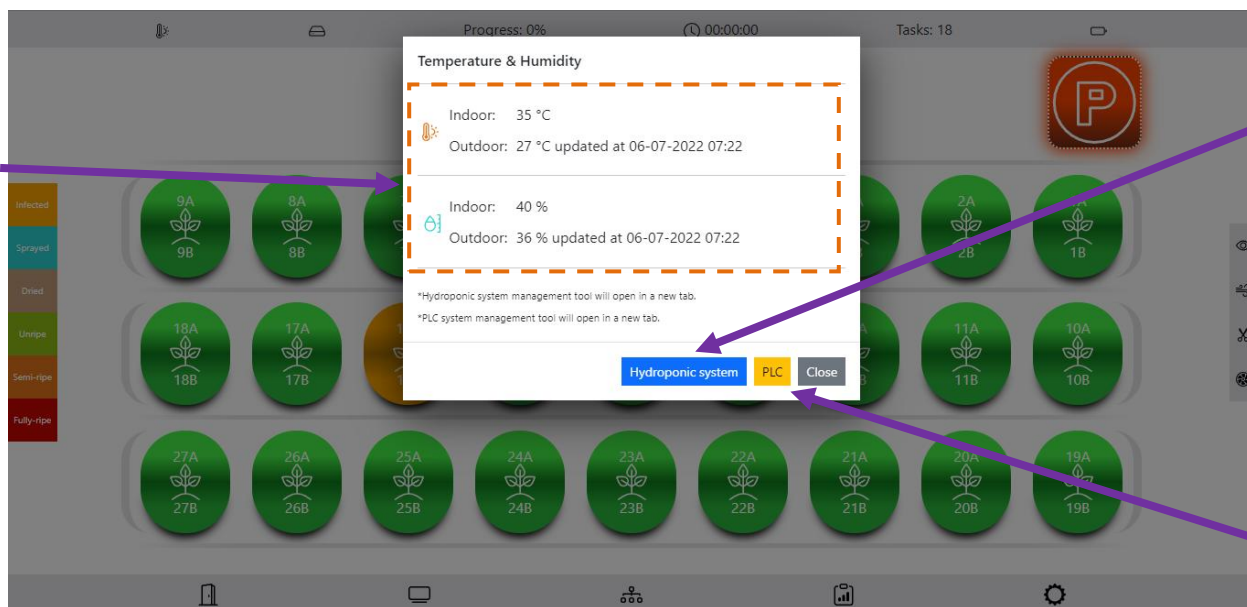
Robot Information

- Temperature: 40°C
- Status: Parked

Robot Equipment

- Bravi Leonardo HD Vehicle
- Embedded Box PC 5000
- Phoenix Inverter 12/3000 Smart
- RoboteQ MGSW1600
- RoboteQ FLW100
- Universal Robots UR10e Robotic Arm
- NVIDIA Jetson Nano Developer Kit
- Realsense D435 Depth Camera
- Multispectral Camera Muses9MSPL

Θερμοκρασία και υγρασία εξωτερικού και εσωτερικού χώρου



Πλοήγηση στο διαχειριστικό της υδροπονικής κεφαλής του θερμοκηπίου

Πλοήγηση στο διαχειριστικό του PLC του θερμοκηπίου

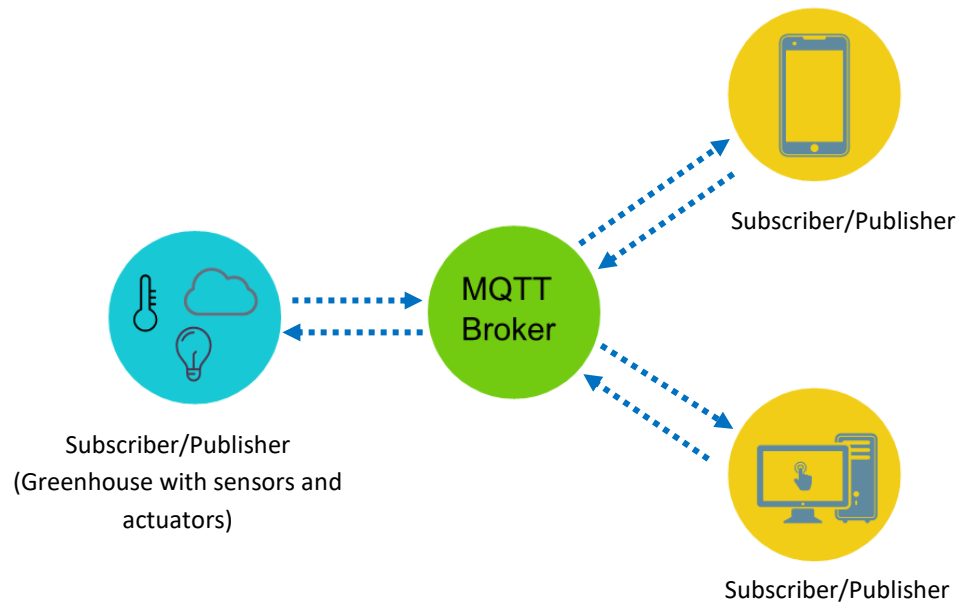
Παρακάτω παρουσιάζεται το διαχειριστικό περιβάλλον της υδροπονικής κεφαλής. Για τη λειτουργία του ο χρήστης πρέπει να βρίσκεται εντός του δικτύου του θερμοκηπίου για λόγους ασφαλείας. Με το παρακάτω σύστημα μπορούν να ελεγχτούν τα διαλύματα τα οποία θα χρησιμοποιηθούν στα φυτά, τη λίπανση, την άρδευση και την προβολή διαφόρων αναφορών του συστήματος.



Παρακάτω παρουσιάζεται το διαχειριστικό περιβάλλον των αυτοματισμών του θερμοκηπίου με το οποίο ο χρήστης μπορεί να παρακολουθήσει την κατάσταση των αισθητήρων του θερμοκηπίου και να ελέγξει πλήρως τα συστήματα εξαερισμού και φωτισμού του θερμοκηπίου. Η διασύνδεση πραγματοποιείται με το πρωτόκολλο MQTT το οποίο παρέχει μια σχεδόν άμεση αποστολή επικοινωνία με μικρό όγκο πακέτων τα οποία αποστέλλονται. Οι καταστάσεις του θερμοκηπίου που μπορεί να παρακολουθήσει ο χρήστης είναι η θερμοκρασία, η υγρασία, ο δείκτης Photosynthetically active radiation (PAR) και το ποσοστό του διοξειδίου του άνθρακα. Τέλος, ο χρήστης μπορεί να ελέγξει το φωτισμό, τον ανεμιστήρα, το rad ψύξης και τους φεγγίτες για τον εξαερισμό του θερμοκηπίου.



Η αρχιτεκτονική της σύνδεσης του συστήματος διαχείρισης των αυτοματισμών του θερμοκηπίου μέσω του PLC παρουσιάζεται παρακάτω. Στην περίπτωση του θερμοκηπίου ο publisher είναι το θερμοκήπιο και ο subscriber είναι ο κάθε χρήστης που θέλει να εποπτεύσει το θερμοκήπιο. Επίσης για την αλλαγή της κατάστασης κάποιων συστημάτων του θερμοκηπίου ο χρήστης είναι και publisher και το θερμοκήπιο subscriber.



Ο ενδιάμεσος κόμβος ανάμεσα στο θερμοκήπιο και του χρήστες είναι ο MQTT Broker. Η επιλογή του πρωτοκόλλου MQTT έγινε για την αποστολή μηνυμάτων μικρού μεγέθους χωρίς καθυστερήσεις, έτσι ώστε να υπάρχει όσο το δυνατόν πιο άμεση ενημέρωση των καταστάσεων του θερμοκηπίου. Είναι ένα πρωτόκολλο το οποίο χρησιμοποιείται για εφαρμογές IoT. Ο MQTT Broker λειτουργεί ως αναμεταδότης μηνυμάτων και έχει υλοποιηθεί στο Web Server που τρέχουν όλες οι υπηρεσίες του συστήματος ελέγχου του θερμοκηπίου. Για την υλοποίηση έχει χρησιμοποιηθεί το Eclipse Mosquitto, το οποίο είναι ένα open source λογισμικό για τη λειτουργία ενός MQTT Broker.

Διαχείριση Εργασιών

Στο μενού διαχείρισης των εργασιών ο χρήστης μπορεί να εισάγει, να τροποποιήσει και να διαγράψει εργασίες. Στην παρακάτω εικόνα παρουσιάζεται η οθόνη με τη λίστα εργασιών. Οι δυνατές εργασίες που μπορεί να εκτελέσει το σύστημα είναι η επιθεώρηση (inspection), συγκομιδή (harvesting), ψεκασμός (spraying) και η γονιμοποίηση (pollination).

Progress: 0%			Tasks: 18	
Task 1057 - No name	Plants 13A, 13B, planned to execute at 9/10/21, 16:30:16 ⌵ Inspection	Completed		
Task 1058 - inspect	Plants 8A, planned to execute at 9/2/20, 14:00:00 ⌵ Inspection	Completed		
Task 1065 - inspect	Plants 3A, planned to execute at 9/2/20, 14:00:00 ⌵ Inspection	Completed		
Task 1076 - inspect	Plants 1A, planned to execute at 9/2/20, 14:00:00 ⌵ Inspection	Assigned		
Task 1077 - inspect	Plants 1A,	Assigned		

Εργασία επιθεώρησης με εμφάνιση βασικών πληροφοριών στη λίστα εργασιών

Προσθήκη εργασίας, ανανέωση εργασιών, αυτόματη λειτουργία, διαγραφή όλων των εργασιών

Το σύστημα υποστηρίζει 4 διαφορετικούς τύπους εργασιών οι οποίες είναι τα Inspection, Harvesting, Spraying και Pollination. Ο χρήστης μπορεί να επιλέξει τον τύπο, το όνομα, τα φυτά, την ημερομηνία και κάποιες άλλες πληροφορίες για κάθε εργασία.

Τύπος εργασίας

Type: ☒ Inspection ☐ Harvesting ☐ Spraying ☐ Pollination

General:

Task name

Description

Plants:

1 2 3 4 5 6 7 8 9

10 11 12 13 14 15 16 17 18

19 20 21 22 23 24 25 26 27

Select multiple plants:

Row 1 Row 2 Row 3 All Plants

Planned date:

June 21st 2022 01:30:37 PM

Status:

Pending

Comments

Close Add Task

Type: ☐ Inspection ☐ Harvesting ☒ Spraying ☐ Pollination

General:

Task name

Description

Plants:

1 2 3 4 5 6 7 8 9

10 11 12 13 14 15 16 17 18

19 20 21 22 23 24 25 26 27

Select multiple plants:

Row 1 Row 2 Row 3 All Plants

Planned date:

June 21st 2022 01:30:37 PM

Status:

Pending

Spray type:

Pesticide

Comments

Φυτά για την επιλεγμένη εργασία

Ημερομηνία εκτέλεσης εργασίας

Αναφορές Συστήματος

Βασικό στοιχείο του συστήματος είναι η εξαγωγή των αναφορών. Ο χρήστης έχει τη δυνατότητα να προβάλει και να εξαγάγει σε μορφή PDF, τις αναφορές του συστήματος που περιλαμβάνουν τις αναφορές εικόνων, αποτελέσματα για ασθένειες, αποτελέσματα των εργασιών, τις προβλέψεις για την αναγνώριση καρπών και άλλα. Παρακάτω παρουσιάζονται οι αναφορές ασθενειών. Ο χρήστης μπορεί επίσης να προβάλει τα αποτελέσματα των εικόνων των φυτών με τις προσβεβλημένες περιοχές του φυτού επιλέγοντας το Mask URL κάθε αποτελέσματος.

Reports												
Results										Download		
											1	
#	Task	Plant	Growth Level	Disease	Disease Level	Pest	Pest Level	Deficiency	Deficiency Level	Confidence	Image	Mask URL
1042	1057	31			HIGH							
1043	1057	31										
1048	1076	1				TUTA	2			NULL	1975	s3://soupserver-u2m/soup_data/upatras_test_cube/Mask_TUTA.png
1049	1077	1				TUTA	2			NULL	1976	s3://soupserver-u2m/soup_data/upatras_test_cube/Mask_TUTA.png
1050	1083	1				TUTA	2			NULL	1977	s3://soupserver-u2m/soup_data/upatras_test_cube/Mask_TUTA.png
1051	1084	1				TUTA	2			NULL	1978	s3://soupserver-u2m/soup_data/upatras_test_cube/Mask_TUTA.png
1052	1087	1				TUTA	2			NULL	1981	s3://soupserver-u2m/soup_data/upatras_test_cube/Mask_TUTA.png
1053	1088	1				TUTA	2			NULL	1982	s3://soupserver-u2m/soup_data/upatras_test_cube/Mask_TUTA.png
1054	1089	1				TUTA	2			NULL	1983	s3://soupserver-u2m/soup_data/upatras_test_cube/Mask_TUTA.png
1055	1090	1				TUTA	2			NULL	1984	s3://soupserver-u2m/soup_data/upatras_test_cube/Mask_TUTA.png
1056	1091	1				TUTA	2			NULL	1985	s3://soupserver-u2m/soup_data/upatras_test_cube/Mask_TUTA.png
1057	1092	1				TUTA	2			NULL	1986	s3://soupserver-u2m/soup_data/upatras_test_cube/Mask_TUTA.png
1058	1093	1				TUTA	2			NULL	1987	s3://soupserver-u2m/soup_data/upatras_test_cube/Mask_TUTA.png
1059	1094	1				TUTA	2			NULL	1988	s3://soupserver-u2m/soup_data/upatras_test_cube/Mask_TUTA.png

Εικόνα με τονισμένα τα σημεία όπου υπάρχει ασθένεια με το ποσοστό επιτυχούς πρόβλεψης.



Στις αναφορές εικόνων ο χρήστης μπορεί να προβάλει και να κατεβάσει όλα τα δεδομένα που έχει αποθηκεύσει το σύστημα στη βάση δεδομένων κατά τη διάρκεια της εργασίας της επιθεώρησης επιλέγοντας το URL τις κάθε εγγραφής.

Progress: 0%

00:00:00

Tasks: 18

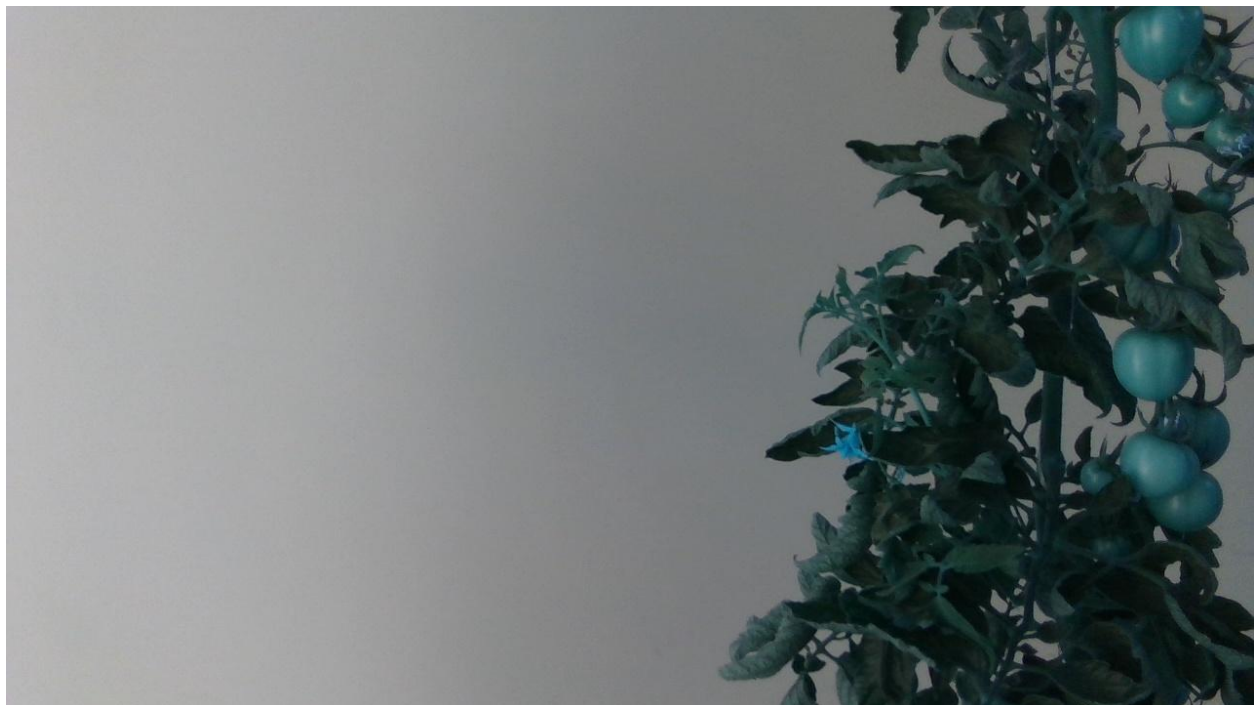
Reports

Images

Download

1

#	Task	Plant	Camera	URL	Status	Position X	Position Y	Position Z	Orientation X	Orientation Y	Orientation Z	Orientation W	Date	Time	Comments
57	33	34	6	soup_data/20210311_145452.jpg	completed	3.14	3.15	3.16					2021-03-11	14:54:52	
58	33	34	6	soup_data/20210311_145442.jpg	completed	3.14	3.15	3.16					2021-03-11	14:54:42	
59	33	34	6	soup_data/20210311_145446.jpg	completed	3.14	3.15	3.16					2021-03-11	14:54:46	
60	33	34	6	soup_data/20210311_145444.jpg	completed	3.14	3.15	3.16					2021-03-11	14:54:44	
61	33	34	6	soup_data/20210311_145450.jpg	completed	3.14	3.15	3.16					2021-03-11	14:54:50	
62	33	34	6	soup_data/20210311_145448.jpg	completed	3.14	3.15	3.16					2021-03-11	14:54:48	
67	33	34	6	soup_data/20210312_103843.jpg	completed	3.14	3.15	3.16					2021-03-12	10:38:43	



Στις αναφορές των prediction ο χρήστης έχει τη δυνατότητα να προβάλει την εικόνα του φυτού με τα predictions για καρπούς επιλέγοντας το εικονίδιο του μεγεθυντικού φακού κάθε εγγραφής. Όλη η διαδικασία γίνεται δυναμικά στο web server που έχει υλοποιηθεί. Παρακάτω παρουσιάζονται οι οθόνες των αναφορών για τα predictions και μια εικόνα να τις προβλέψεις καρπών.

Progress: 0%

00:00:00

Tasks: 18

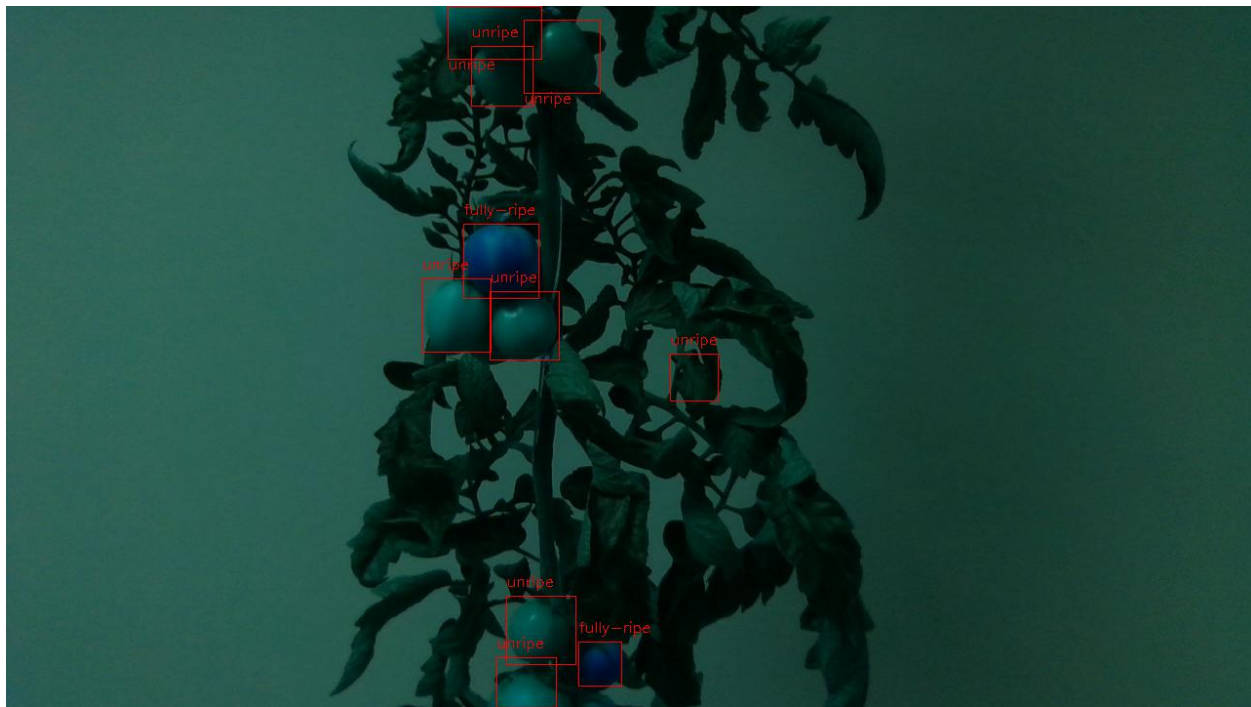
Reports

Predictions

Download

47

#	Image ID	Bbox X	Bbox Y	Bbox W	Bbox H	Space X	Space Y	Space Z	Category	Confidence	Inspection
100	713	477	41	63	61	-0.087063357234001	-0.18586209416389	0.57599008083344	unripe	0.99988758563995	
101	713	587	651	44	45	-0.024199785664678	0.20095671713352	0.59566009044647	fully-ripe	0.99986207485199	
102	713	531	14	78	75	-0.045809786766768	-0.18952958285809	0.5497100353241	unripe	0.99984979629517	
103	713	513	604	71	70	-0.061093688011169	0.17064779996872	0.56989008188248	unripe	0.99918383359909	
104	713	496	292	71	70	-0.069791465997696	-0.023309733718634	0.55508005619049	unripe	0.99914772789	
105	713	453	0	96	54	-0.085600428283215	-0.19991818070412	0.53904002904892	unripe	0.99894493818283	
106	713	426	279	71	75	-0.11183232814074	-0.029881108552217	0.55150008201599	unripe	0.99770146608353	
107	713	503	667	61	52	-0.070064179599285	0.20324932038784	0.49250999093056	unripe	0.99286663532257	





Ρυθμίσεις Συστήματος

Στην οθόνη των ρυθμίσεων, ο χρήστης έχει τη δυνατότητα να αλλάξει κάποιες ρυθμίσεις που αφορούν την αστικοποίηση των αποτελεσμάτων, όπως επίσης και να συνδεθεί τοπικά με το ρομποτικό όχημα σε περίπτωση συντήρησης.

The screenshot displays the 'General Settings' and 'Connection Settings' panels. The 'General Settings' panel includes a 'Confidence from 0 to 1' slider set to 1, a 'Confidence Threshold' checkbox (unchecked), a 'Maturity threshold 0.50 to 1.0' slider set to 1, a 'Tomatoes maturity' checkbox (unchecked), and a 'Refresh rate of the app' dropdown menu set to 'None'. The 'Connection Settings' panel includes a 'REST URL' field with 'https://soup-rest-api.herokuapp.com/', a 'Websocket URL' field with 'wss://soup-websocket.herokuapp.com/', a 'ROS IP' field with '192.168.1.65', a 'ROS Port' field with '9090', and an 'Auto reconnect' checkbox (unchecked). At the bottom of the 'General Settings' panel are 'Save' and 'Restore defaults' buttons. The top status bar shows 'Progress: 0%', '00:00:00', and 'Tasks: 18'.

Στις γενικές ρυθμίσεις ο χρήστης μπορεί να αλλάξει τη μεταβλητή του confidence όπου θα θεωρείται ένα prediction ότι το κάθε φυτό έχει κάποια ασθένεια. Επίσης ο χρήστης μπορεί να αλλάξει τον τρόπο οπτικοποίησης της ωριμότητας των καρπών. Οι καρποί ενός φυτού θεωρούνται ώριμοι να το ποσοστό των ώριμων καρπών είναι μεγαλύτερο από το ποσοστό των άλλων δύο καταστάσεων μαζί και αντίστοιχα για τις άλλες καταστάσεις. Ο χρήστης μπορεί να θεωρεί ότι οι καρποί είναι ώριμοι μόνο από το ποσοστό τις κάθε καταστάσεις (μη ώριμο, ημι-ώριμο, πλήρες ώριμο).

This detailed view of the 'General Settings' panel highlights two specific settings with dashed orange boxes and purple arrows. The first box, labeled 'Ρύθμιση προβολής ασθενειών' (Disease display setting), encloses the 'Confidence from 0 to 1' slider and the 'Confidence Threshold' checkbox. The second box, labeled 'Ρύθμιση προβολής ωριμότητας' (Maturity display setting), encloses the 'Maturity threshold 0.50 to 1.0' slider and the 'Tomatoes maturity' checkbox. Both sliders are currently set to 1. The 'Refresh rate of the app' dropdown is set to 'None'. At the bottom are 'Save' and 'Restore defaults' buttons.

Η εφαρμογή υποστηρίζει και την τοπική διασύνδεση με το ρομποτικό όχημα μέσω του ROS. Αυτή η λειτουργία είναι διαθέσιμη μόνο όταν ο διαχειριστής βρίσκεται στο τοπικό δίκτυο του ρομποτικού οχήματος. Παρακάτω παρουσιάζονται κάποιες ρυθμίσεις για την κίνηση του οχήματος.

Connection Settings

REST URL
https://soup-rest-api.herokuapp.com/

Websocket URL
wss://soup-websocket.herokuapp.com/

ROS IP
192.168.1.65

ROS Port
9090

☐ Auto reconnect

*ROS connection can be established in the same network

Connect Disconnect Settings

← Ρυθμίσεις ROS

Αφού συνδεθεί ο χρήστης στο ρομποτικό όχημα, ο χρήστης έχει τη δυνατότητα να ενεργοποιήσει/απενεργοποιήσει τους τροχούς και να δώσει εντολή να μετακινηθεί σε συγκεκριμένες θέσεις.

Progress: 0% 00:00:00 Tasks: 18

General Settings

Confidence from 0 to 1
1

☐ Confidence Threshold
*Confidence will be ignored for the monitor

Maturity threshold 0.50 to 1.0
1

☐ Tomatoes maturity
*Tomatoes maturity will be ignored for the monitoring panel if not set

Refresh rate of the app
None

*User settings will be saved locally

Save Restore defaults

Position x 0

Position y 0

Status: false

Close Arm Wheels Disarm Wheels Set

ROS IP
192.168.1.65

ROS Port
9090

☐ Auto reconnect

*ROS connection can be established in the same network

Connect Disconnect Settings

Οι δύο τελευταίες κάρτες παρουσιάζουν τους χρήστες που μπορούν αν συνδεθούν στο σύστημα και κάποιες πληροφορίες του συστήματός όπως επίσης και την πλήρη τεκμηρίωση του συστήματος.

Users

1. George Alexakis - geosalexs@gmail.com
2. Nikos Kounalakis - nikkoun96@gmail.com
3. upatras spclab - pgeorgant@ceid.upatras.gr
4. panos upatras -
5. kkonstantino upatras -

*All users have the same privileges

Information

SOUP Harvesting Tool is a Web App that is connected through a REST API, directly with the central Database of the robot. Moreover a WebSocket connection provides a real time data transmission through Internet.

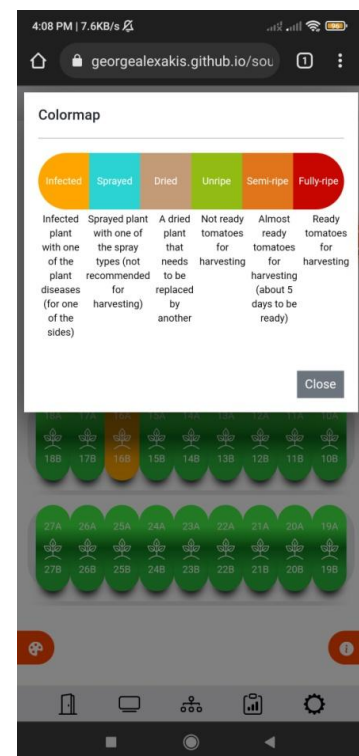
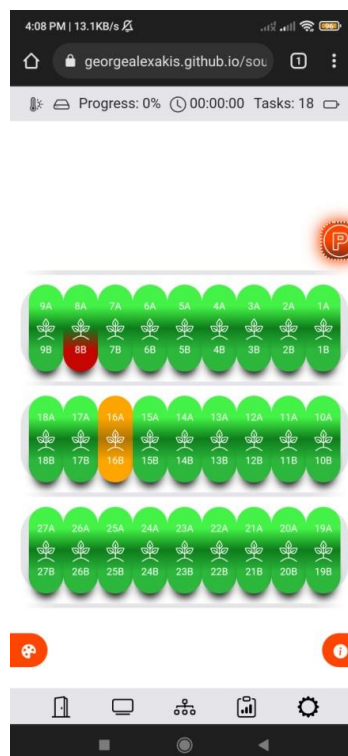
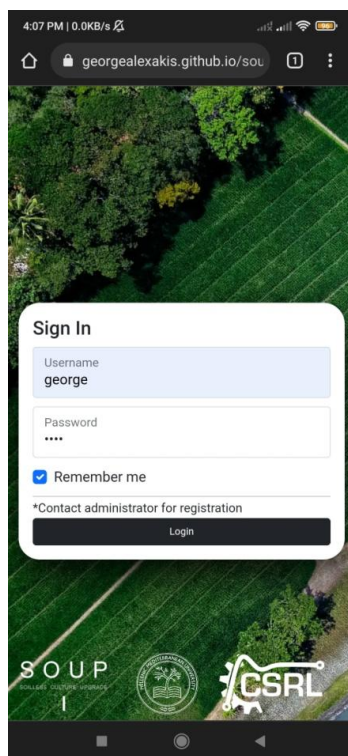
ROS local websocket connection is also available to the user for maintenance purposes of the robotic platform.

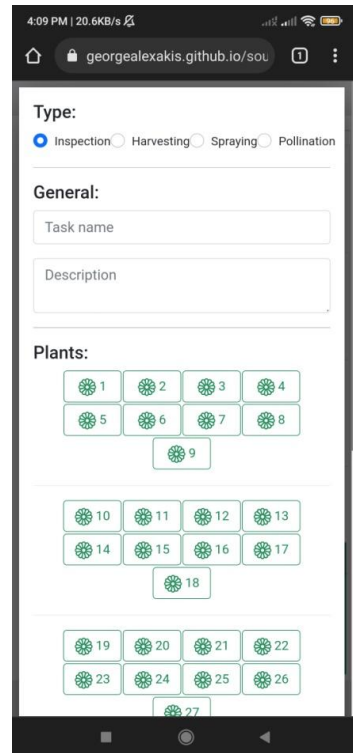
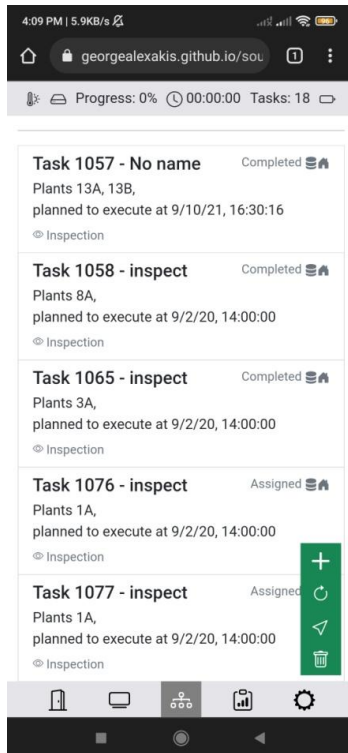
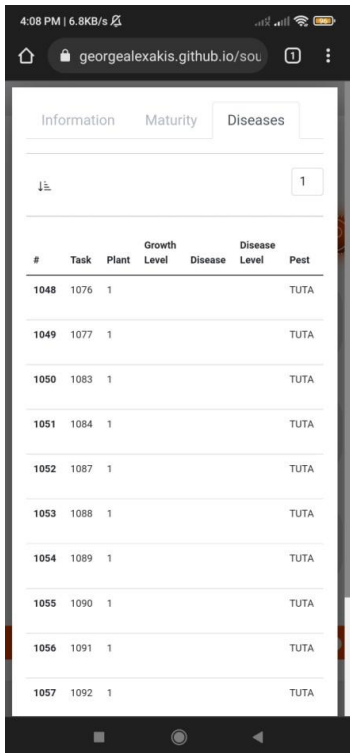
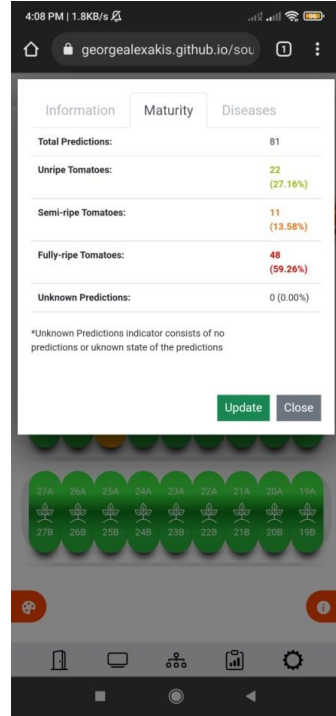
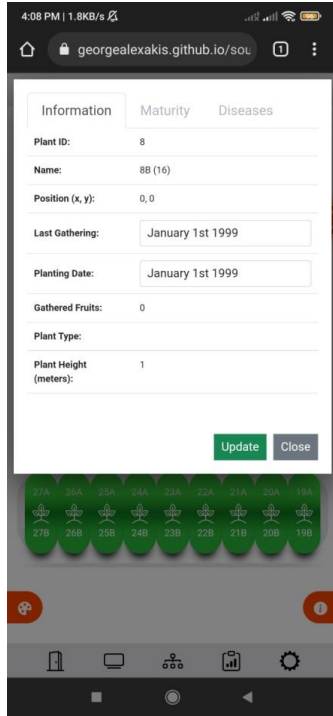
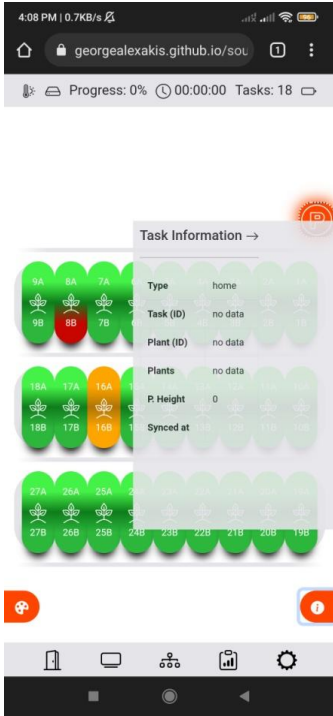
Tasks are saved on the server and locally on the client device. The basic functions consist of the task management, robot autonomous mode, movement of the robot on localy network, etc.

For more information, download the full documentation: [here](#)

Mobile Version

Όλο το σύστημα είναι χτισμένο, έτσι ώστε να το χειρίζεται ο χρήστης από ένα έξυπνο κινητό τηλέφωνο ή από οποιαδήποτε mobile συσκευή. Όλες οι λειτουργίες που είναι διαθέσιμες σε ένα υπολογιστή, είναι διαθέσιμες και στη mobile version της εφαρμογής. Παρακάτω παρουσιάζονται κάποιες οθόνες από τη λειτουργία της εφαρμογής σε ένα κινητό τηλέφωνο.





4:09 PM | 3.6KB/s

georgealexakis.github.io/sou

14

15

16

17

18

19

20

21

22

23

24

25

26

27

Select multiple plants:

Row 1

Row 2

Row 3

All Plants

Planned date:

June 21st 2022

4:08:57 PM

Status:

Pending

Comments

Close

Add Task

4:09 PM | 10.5KB/s

georgealexakis.github.io/sou

Warning!

Are you sure that you want to delete all tasks?

Close

Delete all

Task 1058 - inspect

Completed

Plants 8A,

planned to execute at 9/2/20, 14:00:00

Inspection

Task 1065 - inspect

Completed

Plants 3A,

planned to execute at 9/2/20, 14:00:00

Inspection

Task 1076 - inspect

Assigned

Plants 1A,

planned to execute at 9/2/20, 14:00:00

Inspection

Task 1077 - inspect

Assigned

Plants 1A,

planned to execute at 9/2/20, 14:00:00

Inspection

+

↻

🗑

4:09 PM | 2.2KB/s

georgealexakis.github.io/sou

Progress: 0% 00:00:00 Tasks: 18

Reports

Predictions

Download

1

#	Image ID	Bbox X	Bbox Y	Bbox W	Bbox H	Space X
100	713	477	41	63	61	-0.08706335723400
101	713	587	651	44	45	-0.02419978566467
102	713	531	14	78	75	-0.04580978676676
103	713	513	604	71	70	-0.06109368801116
104	713	496	292	71	70	-0.06979146599769
105	713	453	0	96	54	-0.08560042828321
106	713	426	279	71	75	-0.11183232814074
107	713	503	667	61	52	-0.07006417959928
108	713	681	356	49	48	0.034041032195091

4:09 PM | 1.7KB/s

105	713	453	0	96	54	-0.08560042828321
106	713	426	279	71	75	-0.11183232814074
107	713	503	667	61	52	-0.07006417959928
108	713	681	356	49	48	0.034041032195091
109	713	469	223	77	76	-0.08175620436668
110	714	409	579	27	27	-0.10448998212814
111	714	280	492	32	35	0
112	714	307	211	59	61	0
113	715	839	150	43	43	0
114	715	962	114	59	59	0.19894701242447
115	715	876	360	63	64	0.13821475207806
116	715	812	361	47	48	0
117	715	837	453	41	40	0.11865223199129
118	715	998	665	55	52	0
119	715	882	104	41	41	0.15629300475121

Previous 1 / 196 Next

4:09 PM | 1.4KB/s

georgealexakis.github.io/sou

Progress: 0% 00:00:00 Tasks: 18

General Settings

Confidence from 0 to 1

1

☐ Confidence Threshold

*Confidence will be ignored for the monitoring panel if not set

Maturity threshold 0.50 to 1.0

1

☐ Tomatoes maturity

*Tomatoes maturity will be ignored for the monitoring panel if not set

Refresh rate of the app

None

*User settings will be saved locally

Save

Restore defaults

4:09 PM | 0.0KB/s

*User settings will be saved locally

Save

Restore defaults

Connection Settings

REST URL

https://soup-rest-api.herokuapp.com/

Websocket URL

wss://soup-websocket.herokuapp.co

ROS IP

192.168.1.65

ROS Port

9090

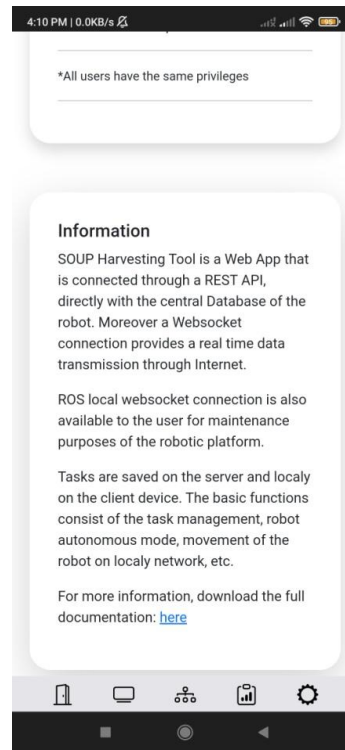
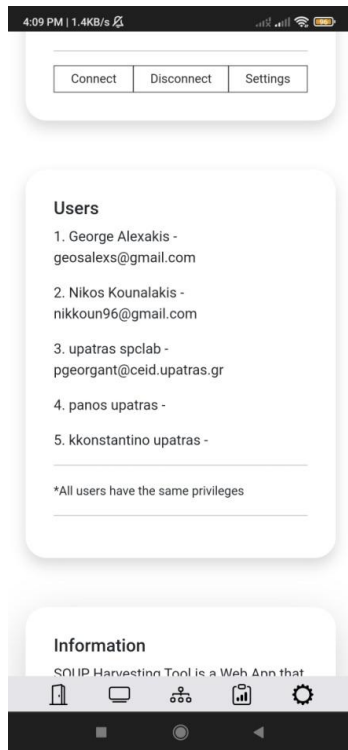
☐ Auto reconnect

*ROS connection can be established in the same network

Connect

Disconnect

Settings

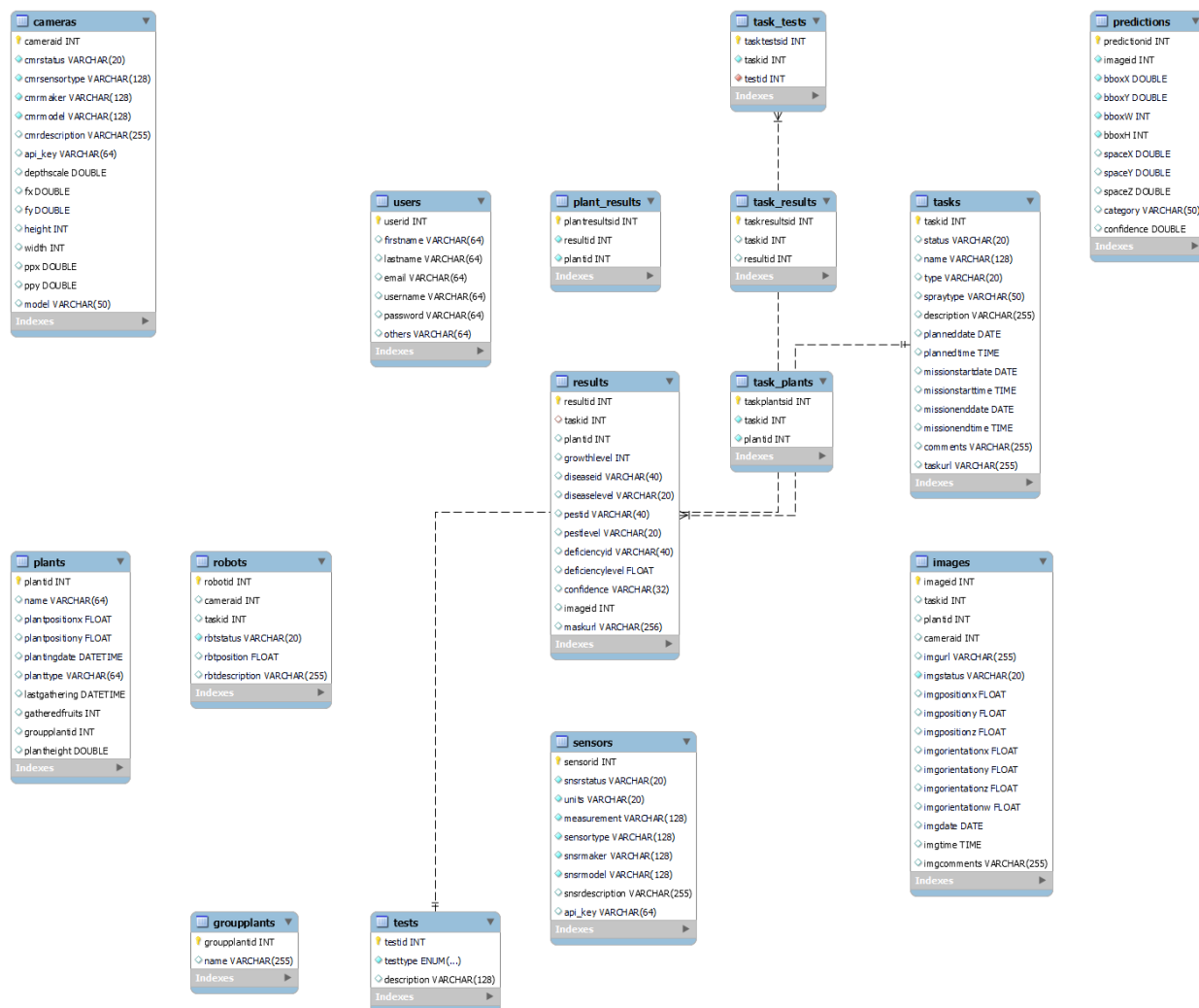


Web Server

Η διασύνδεση της εφαρμογής του χρήστη με τη βάση δεδομένων και η εκτέλεση κρίσιμων λειτουργιών του συστήματος πραγματοποιείται με τη χρήση του Web Server. Ειδικότερα, ο Web Server είναι υπεύθυνος για τη διασύνδεση της βάσης δεδομένων με της εφαρμογή του χρήστη μέσω ενός REST API όπου η εφαρμογή χρήστη εκτελεί GET, POST, DELETE requests και λαμβάνει τα απαραίτητα για τη λειτουργία δεδομένα. Γενικότερα ο Web Server είναι ο ενδιάμεσος κόμβος για τη λειτουργία του συστήματος. Παρακάτω παρουσιάζεται η δομή της βάσης δεδομένων και ο τρόπος που αποθηκεύονται τα δεδομένα.

Δομή Βάσης Δεδομένων Συστήματος Ελέγχου

Η βάση δεδομένων που αναφέρθηκε παραπάνω και αποτελεί βασικό δομικό στοιχείο του ενιαίου συστήματος ελέγχου και διεπαφής με το χρήστη είναι τύπου MySQL και η λειτουργία της βασίζεται στις Amazon Relational Database Services. Παρακάτω παρουσιάζονται οι πίνακες της βάσης δεδομένων που διαχειρίζεται ο Web Server.



REST API Requests

/auth: POST (Authentication)

/plants: GET (Get all plants)

/plants/length: GET (Get the number of plants)

/plant/:id: GET (Get single plant)

/plant/:id: PUT (Update plant)

/plant: GET (Get plant by argument)

/tasks: GET (Get all tasks)

/tasks/length: GET (Get the number of tasks)

/tasks: DELETE (Delete all tasks)

/task: POST (Create new task)

/task/:id: GET (Get single task)

/task/:id: PUT (Update task)

/task/:id: DELETE (Delete task)

/task: GET (Get task by argument)

/users: GET (Get all users)

/users/length: GET (Get the number of users)

/user/:id: GET (Get single user)

/user/:id: PUT (Update user)

/user/:id: DELETE (Delete user)

/images: GET (Get all images)

/images/length: GET (Get the number of images)

/image: POST (Create new image)

/image/:id: GET (Get single image)

/results: GET (Get all results)

/results/length: GET (Get the number of results)

/results/:id: GET (Get result)

/results/task/:id: GET (Get result by task)

/results/length/task/:id: GET (Get the number of results by task)

/results/plant/:id: GET (Get results by plant)

/results/length/plant/:id: GET (Get the number of results by plant)

/cameras: GET (Get all cameras)

/cameras/length: GET: (Get the number of cameras)

/robots: GET (Get all robots)

/robots/length: GET (Get the number of sensors)

/sensors: GET (Get all sensors)

/sensors/length: GET (Get the number of sensors)

/tests: GET (Get all tests)

/tests/length: GET (Get the number of tests)

/predictions: GET (Get all predictions);

/predictions/length: GET (Get the number of predictions)

/prediction/:id: GET (Get predictions by plant)

/predictions/image/:id: GET (Get predictions by image)

/prediction/length/:id: GET (Get the number of predictions);

/reports: GET (Get all reports);

Ο πίνακας **Tasks** περιλαμβάνει πληροφορία που αφορά τον προγραμματισμό και την εκτέλεση μιας εργασίας. Μπορεί να περιγράφει κάθε είδους εργασία: εξέταση φυτών (**inspection**), ψεκασμό (**spraying**) ή συγκομιδή (**harvesting**). Η εφαρμογή χρήστη είναι υπεύθυνη για τη δημιουργία νέων εργασιών και το ρομποτικό σύστημα ανατρέχει σε αυτόν προκειμένου να εκτελέσει κάποια νέα εργασία.

α/α	Όνομα πεδίου	Περιγραφή
1	taskid	Χαρακτηριστικός αύξων αριθμός έργου (πρωτεύον κλειδί πίνακα)
2	status	Δείκτης κατάστασης εργασίας (pending, processing, completed, canceled)
3	name	Όνομα εργασίας (καθορίζεται από το χρήστη)
4	type	Τύπος εργασίας (επιθεώρηση φυτών, συγκομιδή καρπών, ψεκασμός)
5	spraytype	Είδος ψεκαστικού σκευάσματος (σε περίπτωση ψεκασμού)
6	description	Περιγραφή εργασίας (καθορίζεται από τον χρήστη)
7	planneddate	Επιθυμητή ημερομηνία εκτέλεσης εργασίας
8	plannedtime	Επιθυμητή ώρα εκτέλεσης εργασίας
9	missionstartdate	Πραγματική ημερομηνία έναρξης εργασίας
10	missionstarttime	Πραγματική ώρα έναρξης εργασίας
11	missionenddate	Πραγματική ημερομηνία ολοκλήρωσης εργασίας
12	missionendtime	Πραγματική ώρα ολοκλήρωσης εργασίας
13	comments	Σχόλια (καθορίζονται από το χρήστη ή το σύστημα)
14	pointcloud	Αρχείο pointcloud της αναπαράστασης των φυτών της εργασίας
15	pointcloudview	Εικόνα της 3D αναπαράστασης των φυτών της εργασίας

Ο πίνακας **Images** συγκεντρώνει τα δεδομένα που συνοδεύουν μια φωτογραφία και περιγράφουν τις συνθήκες λήψης. Το ρομποτικό σύστημα είναι υπεύθυνο για την εισαγωγή ενός νέου πεδίου, κάθε φορά που γίνεται λήψη φωτογραφίας ανάλογα με την κάμερα λήψης που χρησιμοποιείται.

α/α	Όνομα πεδίου	Περιγραφή
1	imageid	Χαρακτηριστικός αύξων αριθμός φωτογραφίας (πρωτεύον κλειδί πίνακα)
2	taskid	Χαρακτηριστικός αριθμός έργου, στα πλαίσια του οποίου έγινε η λήψη (ξένο κλειδί)
3	plantid	Χαρακτηριστικός αριθμός κάμερας που χρησιμοποιήθηκε για την λήψη φωτογραφίας (ξένο κλειδί)
4	cameraid	Χαρακτηριστικός αριθμός φυτού φωτογραφίας (ξένο κλειδί)
5	imgurl	Διεύθυνση αποθήκευσης φωτογραφίας στο AWS Bucket Storage
6	imgstatus	Δείκτης κατάστασης της ανάλυσης (pending, processing, completed, canceled)
7	imgpositionx	Θέση κάμερας, συνταγμένες X
8	imgpositiony	Θέση κάμερας, συνταγμένες Y
9	imgpositionz	Θέση κάμερας, συνταγμένες Z
10	imgorientationx	Προσανατολισμός κάμερας X
11	imgorientationy	Προσανατολισμός κάμερας Y
12	imgorientationz	Προσανατολισμός κάμερας Z
13	imgorientationw	Προσανατολισμός κάμερας W
14	imgdate	Ημερομηνία λήψης φωτογραφίας
15	imgtime	Ώρα λήψης φωτογραφίας
16	imgcomments	Σχόλια (καθορίζονται από το σύστημα)

Ο πίνακας **Plants** συγκεντρώνει πληροφορίες σχετικά με την πλευρά του κάθε φυτού και πληροφορίες θέσης, ύψους και συλληφθέντων καρπών. Κάθε φυτό αποτελείται από δυο εγγραφές στον πίνακα plants.

α/α	Όνομα πεδίου	Περιγραφή
1	plantid	Χαρακτηριστικός αύξων αριθμός φυτού (πρωτεύον κλειδί πίνακα)
2	name	Όνομα φυτού (χρησιμοποιείται για την αστικοποίηση στο Web Application)
3	plantpositionx	Συντεταγμένες Χ θέσης φυτού στο θερμοκήπιο
4	plantpositiony	Συντεταγμένες Υ θέσης φυτού στο θερμοκήπιο
5	plantingdate	Ημερομηνία φύτευσης φυτού
6	planttype	Τύπος φυτού
7	lastgathering	Ημερομηνία τελευταίας συγκομιδής καρπών
8	gatheredfruits	Αριθμός καρπών που συλλέχθηκαν
9	groupplantid	Χαρακτηριστικός αριθμός πλευράς φυτού ως ενοποίηση σε ένα ενιαίο φυτό (ξένο κλειδί)
10	plantheight	Υψος φυτού

Ο πίνακας **Results** συγκεντρώνει τα αποτελέσματα από τον έλεγχο μιας φωτογραφίας για πιθανές ασθένειες και τροφοπενίες. Ο διακομιστής επεξεργασίας είναι υπεύθυνος για την εισαγωγή ενός νέου πεδίου, κάθε φορά που ολοκληρώνεται ένα τεστ.

α/α	Όνομα πεδίου	Περιγραφή
1	resultid	Χαρακτηριστικός αύξων αριθμός αποτελεσμάτων ανάλυσης (πρωτεύον κλειδί πίνακα)
2	taskid	Χαρακτηριστικός αριθμός εργασίας, στα πλαίσια του οποίου έγινε η λήψη (ξένο κλειδί)
3	plantid	Χαρακτηριστικός αριθμός φυτού (ξένο κλειδί)
4	growthlevel	Δείκτης ανάπτυξης φυτού
5	diseaseid	Φυτοπαθολογική ασθένεια (none, phytophthora infestans, alternaria solani, leveillula Taurica)
6	diseaselevel	Έκταση φυτοπαθολογικής ασθένειας (low, medium, high, total)
7	pestid	Εντομολογική προσβολή (none, tuta absoluta, myzus persicae, tetranychus uticae)
8	pestlevel	Έκταση εντομολογικής προσβολής (low, medium, high, total)
9	deficiencyid	Ελάττωμα φυτού (none, nitrogen deficiency, calcium deficiency, potassium deficiency)
10	deficiencylevel	Έκταση ελαττώματος φυτού (low, medium, high, total)
11	confidence	Αυτοπεποίθηση συστήματος ως προς το αποτέλεσμα
12	imageid	Χαρακτηριστικός αριθμός φωτογραφίας (ξένο κλειδί)
13	maskurl	Διεύθυνση αποθήκευσης αποτελεσμάτων επεξεργασία στο AWS Bucket Storage

Ο πίνακας **Predictions** συγκεντρώνει τα αποτελέσματα από τις προβλέψεις ωριμότητας των καρπών των φυτών. Σε κάθε φωτογραφία μπορεί να υπάρχουν παραπάνω από μια προβλε΄ψεις.

α/α	Όνομα πεδίου	Περιγραφή
1	predictionid	Χαρακτηριστικός αύξων αριθμός πρόβλεψης ωριμότητας (πρωτεύον κλειδί πίνακα)
2	imageid	Χαρακτηριστικός αριθμός εικόνας (ξένο κλειδί)
3	bboxX	Θέση κουτιού αποτελέσματος Χ
4	bboxY	Θέση κουτιού αποτελέσματος Υ
5	bboxW	Πλάτος κουτιού αποτελέσματος
6	bboxH	Υψος κουτιού αποτελέσματος
7	spaceX	Θέση στο 3D περιβάλλον Χ
8	spaceY	Θέση στο 3D περιβάλλον Υ
9	spaceZ	Θέση στο 3D περιβάλλον Ζ
10	category	Κατηγορία
11	confidence	Αυτοπεποίθηση συστήματος της πρόβλεψης αποτελέσματος

Ο πίνακας **Cameras** περιλαμβάνει πληροφορίες που αφορούν τις κάμερες και την κατάστασή τους.

α/α	Όνομα πεδίου	Περιγραφή
1	cameraid	Χαρακτηριστικός αύξων αριθμός κάμερας (πρωτεύον κλειδί πίνακα)
2	cmrstatus	Κατάσταση (εν λειτουργία ή αναμονή ή βλάβη)
3	cmrsensortype	Τύπος κάμερας
4	cmrmaker	Κατασκευαστής
5	cmrmodel	Μοντέλο
6	cmrdescription	Περιγραφή
7	api_key	API KEY κάμερας
8	depthscale	Κλίμακα βάθους
9	fx	Παράμετρος fx κάμερας
10	fy	Παράμετρος fy κάμερας
11	height	Ύψος κάμερας
12	width	Πλάτος κάμερας
13	ppx	Παράμετρος rpx κάμερας
14	ppy	Παράμετρος rpy κάμερας
15	model	Μοντέλο distortion

Στον πίνακα **Sensors** δίνονται πληροφορίες που αφορούν τους αισθητήρες του θερμοκηπίου και τις μετρήσεις τους.

α/α	Όνομα πεδίου	Περιγραφή
1	sensorid	Χαρακτηριστικός αύξων αριθμός αισθητήρα (πρωτεύον κλειδί πίνακα)
2	snsrstatus	Κατάσταση (εν λειτουργία ή βλάβη)
3	units	Μονάδες μέτρησης
4	measurement	Τρέχουσα μέτρηση του αισθητήρα
5	sensortype	Τύπος αισθητήρα
6	snsrmaker	Κατασκευαστής
7	snsrmodel	Μοντέλο
8	snsrdescription	Περιγραφή
9	api_key	Χαρακτηριστικός αύξων αριθμός αισθητήρα (πρωτεύον κλειδί πίνακα)

Ο πίνακας **Robots** δίνει πληροφορίες που αφορούν το ρομπότ του θερμοκηπίου και την κατάστασή του.

α/α	Όνομα πεδίου	Περιγραφή
1	robotid	Χαρακτηριστικός αύξων αριθμός ρομποτικού συστήματος (πρωτεύον κλειδί πίνακα)
2	cameraid	Χαρακτηριστικός αριθμός κάμερας που έχει συνδεθεί στον ρομποτικό βραχίονα (ξένο κλειδί)
3	taskid	Χαρακτηριστικός αριθμός έργου που έχει αναλάβει το ρομπότ (ξένο κλειδί)
4	rbtstatus	Κατάσταση (εν λειτουργία ή αναμονή ή βλάβη)
5	rbtposition	Τρέχουσα θέση
6	rbtdescription	Περιγραφή

Ο πίνακας **Tests** περιγράφει όλους τους ελέγχων που έγιναν κατά την διάρκεια μιας εργασίας.

α/α	Όνομα πεδίου	Περιγραφή
1	testid	Χαρακτηριστικός αύξων αριθμός (πρωτεύον κλειδί πίνακα)
2	testtype	Χαρακτηριστικός αριθμός τεστ (DISEASE, PEST, DEFICIENCY)
3	description	Περιγραφή τεστ

Ο πίνακας **Users** περιέχει όλες τις απαραίτητες πληροφορίες για τους χρήστες που μπορούν να διαχειριστούν το σύστημα.

α/α	Όνομα πεδίου	Περιγραφή
1	userid	Χαρακτηριστικός αύξων αριθμός (πρωτεύον κλειδί πίνακα)
2	firstname	Όνομα χρήστη
3	lastname	Επώνυμο χρήστη
4	email	Email χρήστη
5	username	Κλειδί σύνδεσης χρήστη
6	password	Κωδικός σύνδεσης χρήστη
7	others	Πεδίο για γενικές πληροφορίες

Οι πίνακες **task_plants**, **task_results**, **task_tests**, **plant_results**, **groupplants** είναι πίνακες που χρησιμοποιούνται για τα JOIN σε δύο πίνακες. Εκεί αποθηκεύονται τα ids των πινάκων που γίνεται ο συσχετισμός με τις εγγραφές του ενός πίνακα με τον άλλο πίνακα.

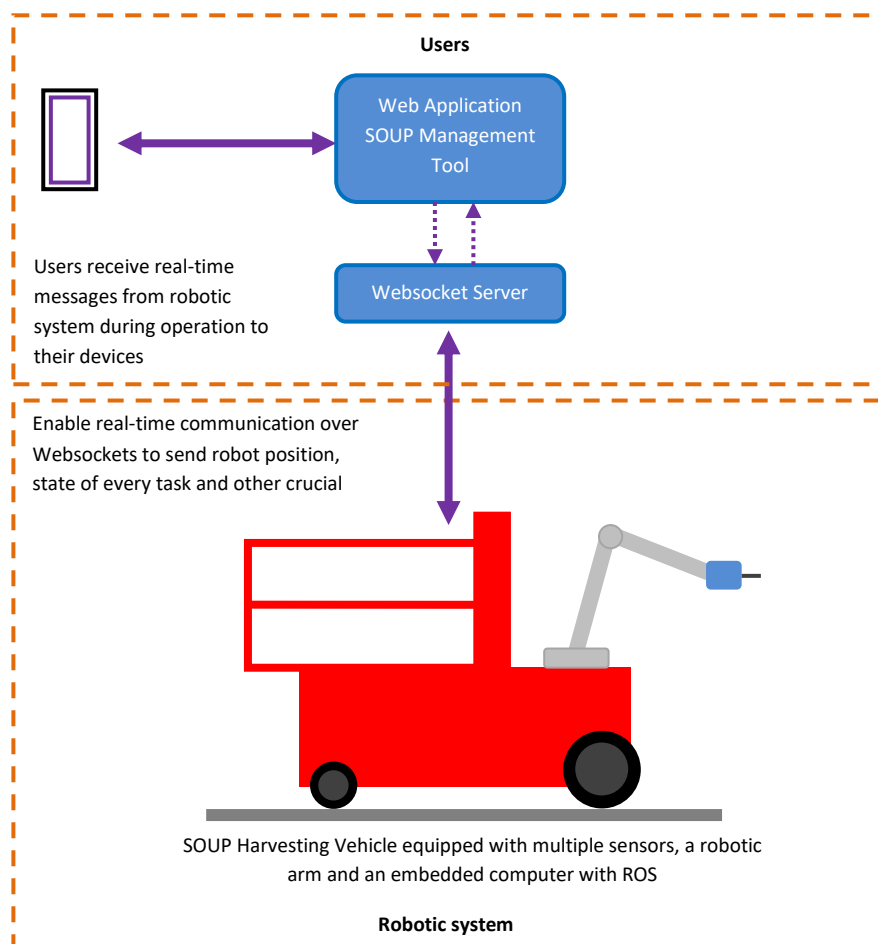
Για την εξαγωγή αρκετών αποτελεσμάτων γίνεται η χρήση σύνθετων ερωτημάτων από τη βάση δεδομένων. Παρακάτω παρουσιάζονται κάποια σύνθετα ερωτήματα που χρησιμοποιούνται.

- **Εξαγωγή εργασιών σε αντιστοιχία με τα φυτά:** `SELECT * FROM soupdb.tasks as tasks INNER JOIN soupdb.task_plants as tplants ON tasks.taskid = tplants.taskid INNER JOIN soupdb.plants as plants ON plants.plantid = tplants.plantid WHERE tasks.taskid = 'id';`
- **Εξαγωγή αποτελεσμάτων ωριμότητας:** `SELECT * FROM soupdb.predictions INNER JOIN soupdb.images ON soupdb.predictions.imageid=soupdb.images.imageid WHERE soupdb.images.plantid='id';`

Βασικό στοιχείο του συστήματος είναι η ασφάλεια των δεδομένων. Για τη διαχείριση δεδομένων όπως επίσης ακόμα και για την προβολή τους, πρέπει να γίνει επαλήθευση με τα κλειδιά χρήστη. Για τη λειτουργία σύνδεσης κάθε χρήστη χρησιμοποιείται η τεχνολογία JSON Web Token (JWT). Το JSON Web Token (JWT) είναι ένα ανοιχτό πρότυπο που χρησιμοποιείται για την κοινή χρήση πληροφοριών ασφαλείας μεταξύ client και server. Κάθε JWT περιέχει κωδικοποιημένα αντικείμενα JSON, όπου υπογράφονται χρησιμοποιώντας έναν κρυπτογραφικό αλγόριθμο για να διασφαλιστεί η επικοινωνία χωρίς να μπορούν να τροποποιηθούν μετά την έκδοση του κάθε κλειδιού. Με αυτόν τον τρόπο διασφαλίζεται μια ασφαλή επικοινωνία μεταξύ χρήστη και του web server.

Websocket Server

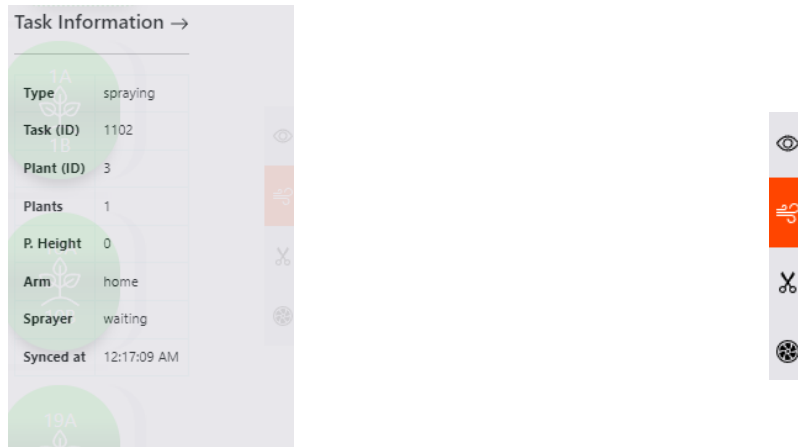
Κατά τη λειτουργία του συστήματος και ειδικότερα κατά την κίνηση του ρομποτικού οχήματος για την εκτέλεση εργασιών συγκομιδής, επιθεώρησης και άλλων, κρίθηκε αναγκαίο να υπάρχει δυνατότητα παρακολούθησης του οχήματος σε πραγματικό χρόνο. Δίνοντας αυτή τη δυνατότητα στο χρήστη, παρέχεται η αίσθηση ότι βρίσκεται στο χώρο του θερμοκηπίου παρέχοντας ταυτόχρονα πλήρη έλεγχο στο σύστημα. Ειδικότερα για την υλοποίηση του υποσυστήματος ανταλλαγής δεδομένων σε πραγματικό χρόνο (R.T. data exchange) χρησιμοποιήθηκε το πρωτόκολλο Websocket το οποίο σε αντίθεση με το πρωτόκολλο HTTP, το Websocket παρέχει πλήρης αμφίδρομη επικοινωνία μεταξύ client και server. Με αυτό τον τρόπο μπορεί να επιτευχθεί η επικοινωνία σε πραγματικό χρόνο όπου το όχημα στέλνει μηνύματα τις κατάστασης του καθώς εκτελεί εργασίες. Ο Websocket Server χρησιμοποιείται ως αναμεταδότης μηνυμάτων μεταξύ του οχήματος και της εφαρμογής του χρήστη. Τα μηνύματα που λαμβάνει η εφαρμογή είναι της μορφής JSON και πρέπει να έχουν συγκεκριμένη δομή.



Παρακάτω παρουσιάζεται το πλαϊνό πάνελ, το οποίο εμφανίζει τις καταστάσεις του οχήματος σε πραγματικό χρόνο. Ο χρήστης έχει τη δυνατότητα να δει σε πραγματικό χρόνο τα παρακάτω:

- Τον τύπο της εργασίας που εκτελείται.
- Την εργασία που εκτελείται.
- Το φυτό που εκτελείται η εργασία.
- Το σύνολο των φυτών που περιέχει η εργασία.
- Την κατάσταση της πολυφασματικής κάμερας.
- Την κατάσταση της κάμερας βάθους.
- Την κατάσταση του βραχίονα.
- Την κατάσταση του ψεκαστικού συστήματος.
- Το ύψος της πλατφόρμας.
- Την κατάσταση των μπαταριών του οχήματος.
- Την τρέχουσα θέση του οχήματος μέσα στο θερμοκήπιο.
- Την κατάσταση της αρπάγης του βραχίονα.





Παρακάτω παρουσιάζεται η δομή ενός μηνύματος JSON που χρησιμοποιείται για τη λειτουργία του συστήματος με τις περιγραφές του κάθε κλειδιού.

```
{
  task: string,    -->  values accepted values are (inspection, harvesting, spraying, pruning, home, moving, stopped, unknown)

  status: string   -->  values accepted values are (pending, completed, processing, canceled, none)

  meta: int[],     -->  values TASKID|PLANTID|PLANTS_TOTAL default is -1|-1|-1 no meta is [-1, -1, -1]

  scissors: int,   -->  values ON/OFF for 1/0

  gripper: int,    -->  values ON/OFF for 1/0

  spray: int,      -->  values ON/OFF for 1/0

  multispectral: int, -->  values ON/OFF for 1/0

  realsense: int,  -->  values ON/OFF for 1/0

  arm: int,        -->  values ON/OFF for 1/0

  platform: float, -->  values float value of height

  odometry: string, -->  values choose the format

  batteries: int[], -->  values array with system/platform batteries example [90, 89]

  timers: int[]    -->  values infinite array with timers []
}
```