universität innsbruck

Plant Health

Take care of your beloved plants!

Giuliano Giambertone, Stefan Huter, Deniz Özkaya, Marco Ponti, Tom van Loon

Overview

- **Live Demo**
- **Software Architecture**
- **IoT and Reliability**
- **Software Quality**
- **User Interface**
- **Project Management**
- **Lessons Learned**

Motivation

Plants are fundamentally sovereign life forms. However, three challenges prevent most people from growing their own vegetables, herbs and more.

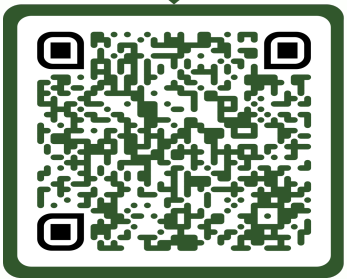
- Lack of time
- Knowledge
- Environment

Plant Health tries to offer a solution for that in form of a web app and sensor stations that are used to track the environment of the plant.

Live Demo

Feel free to try it out!

Plant Health



Username: FPAdmin

Password: passwd

[0] www.srh-softwaresolutions.com

Strengths of the project

- **Running server version**
- **Dark Mode**
- **Global Search Bar for Plants**
- **Users can change their user information incl. password**
- **Notifications for admins/gardeners**
- **Info buttons**
- **Ability to delete measurements**

Software Architecture

Objective of the project was to build a web app that solves the tasks mentioned in the Motivation slide.

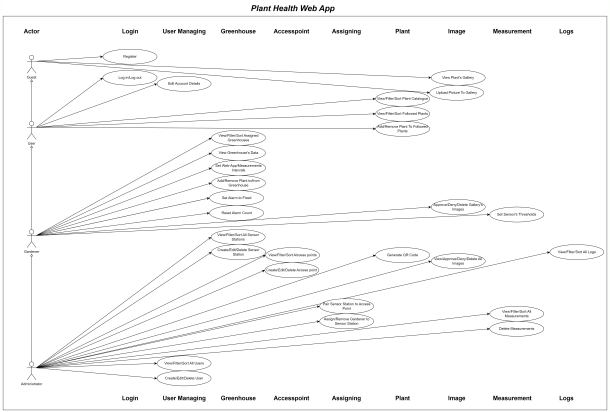
Roles:

- **User:** Track their followed plants.
- **Gardener:** Take care of assigned plants.
- **Admin:** Manage the users, plants, images, sensor stations and access points.

Software Architecture cont.

- **Sensor station:** Measure the environmental data at the plant and send it to the paired access point.
- **Access point:** Collect the data of the paired sensor stations and send it to the web app.
- **Web app:** Give the user a user-friendly User Interface to keep track of their plants.

Software Architecture: Use Case Diagram



Software Architecture: Pairing Sequence Diagram

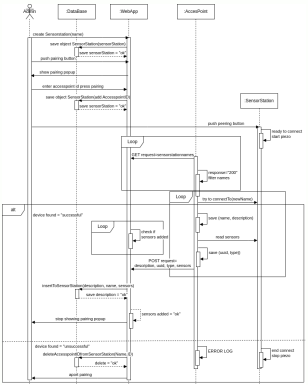


Image: Sequencediagram connection to sensor station

Internet of Things

- **Sensor station:** For the sensor station an Arduino NanoLE is used which sends the measurements via Bluetooth Low Energy to its connected access point.
- **Access point:** The access point is realized with a Raspberry Pi. It stores the data locally and transmits it to the web app in an adjustable time frame. After successfully transmitting the data to the web app, the access point deletes the local data.
- **Web app:** The web app is implemented using Spring Boot and JSF as a frontend framework. The web app stores the received measurements in a database and displays everything in a user-friendly way.

Reliability

- **T1 Restricted communication between sensor station and access point:** The data will be transmitted whenever there is a stable connection again. Until then, no measurements can be sent.
- **T2 Unexpected restart of an access point:** The access point will be restarted immediately using cronjob.
- **T3 Temporary failure of communication between access point and web server:** Data on the access point will only be deleted if a response of successful transmission is received from the web app. Otherwise it will still be stored at the access point and will be retransmitted in the next transmission frame.

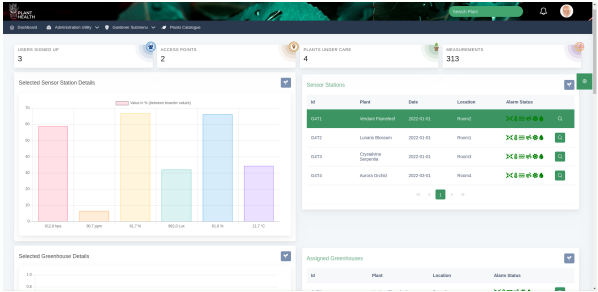
Reliability cont.

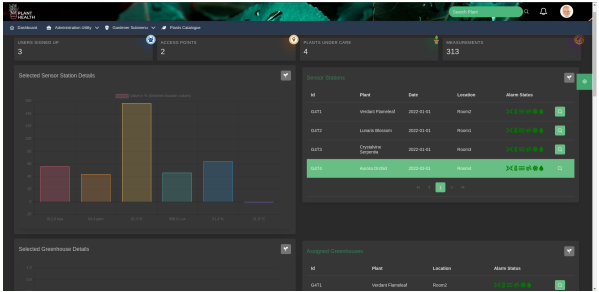
- **T4 Short term failure of the backend:** As explained in T3 this will result in no data loss from the access point since the access point will not delete any data before receiving an acknowledgment of successful transmission from the web app. Also data stored in the database of the web app is persistent and will therefore not be lost.
- **T5 Outage of the sensor station:** When pairing successfully, we use a switch on the sensor station, so that the sensor stations knows that it was already successfully paired. If this switch is set to on, after restart the sensor station enters advertising immediately.

Software Quality

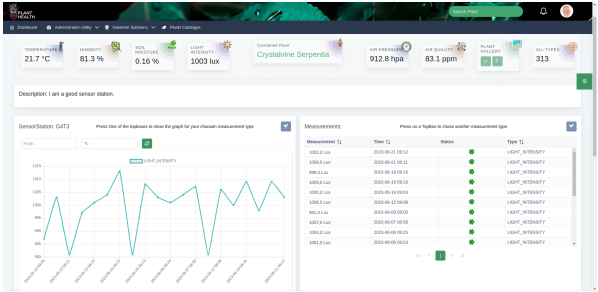
- System was tested with all use-cases covered.
- Test coverage of around 70% in SonarQube.
- No security issues were discovered in the process of testing.
- The provided system is organized well and extendable.
- Layer architecture is followed.

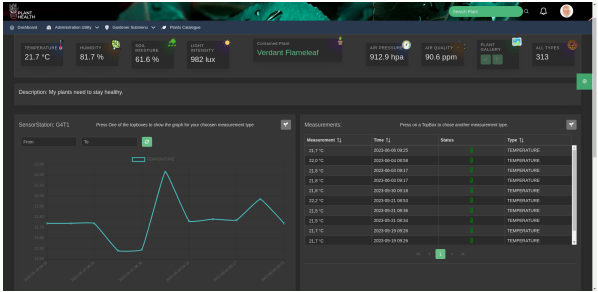
User Interface: Dashboard



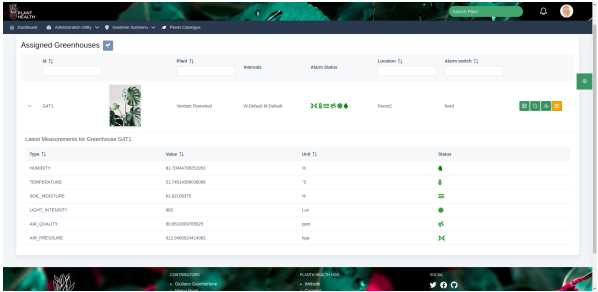


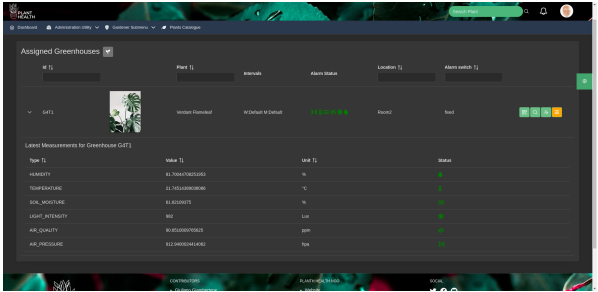
User Interface: Sensor Station Details





User Interface: Assigned Sensor Stations





Project Management

We split the team into 2 groups, one for the web app and the second one for the hardware part of the project.

Webapp:

- Giuliano Giambertone
- Deniz Özkaya
- Marco Ponti

Hardware:

- Stefan Huter
- Tom van Loon

Lessons Learned

- **A good initial concept can take you far!**
- **No matter how much time you expect a feature to take, it will take more!**
- **Good communication within the team is key!**
- **Working in a motivated team is fun!**

Thank you for your attention!

Giuliano Giambertone, Stefan Huter, Deniz Özkaya, Marco Ponti, Tom van Loon