SOFIA UNIVERSITY “ST. KLIMENT OHRIDSKI”

FACULTY OF MATHEMATICS AND INFORMATICS

Course Practical Robotics And Smart Things

Instructor: Trayan Iliev

Project documentation for “Smart Plant”

Elizabet Svetlinova Mihaylova
FN: 45194
Program: Informatics

Date:

02.07.2019

Content:

Description of functional and non-functional project requirements

Monitoring plant health is very important for their fast growth. In this busy world, people usually forget to water their plants which leads to bad growth and health of their plants. For ensuring complete development of plants it is necessary to develop proper surrounding conditions in which plants grow.

The main purpose of automation is to provide comfort to the people by reducing the manual work and to improve the growth of plants without user interaction. The some of the important parameters for the quality and productivity of plant growth are soil, soil moisture and sunlight. Information to the user about the plant health and growth may be provided to the user by continuously monitoring and recording these garden parameters. It provides a better understanding of how each parameter affects the growth of plants. And this information about the garden can be directly monitored and controlled by the owner of the system through his/hers smartphone or laptop. Camera is provided in this system to monitor the garden through smart device. This smart gardening system will provide convenience and comfort to the user by sensing and controlling the parameters of the garden without their physical presence. All data are stored in the database and can be retrieved at any time. This will help the user to understand the relation between the plant growth and the mentioned garden parameters.

Description of used technologies, hardware component, and software libraries with references to sources, including PEAS

Technologies

The system program is deployed on Raspberry Pi 3 which is the server side and

1. Node-Red

It is a programming tool for wiring together hardware devices, APIs and online services in new and interesting ways. It provides a browser-based editor that makes it easy to wire together flows using the wide range of nodes in the palette that can be deployed to its runtime in a single-click.

Source: https://nodered.org/

2. Arduino IDE

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software.

Source: https://www.arduino.cc/en/Main/Software

Hardware components

1. Raspberry Pi 3
2. Arduino Duemilanove
3. Water pump
4. L293D Motor Driver
5. Soil moisture sensor
6. Led strip
7. MPD MINI LED Driver 12w
8. Relay 5v
9. USB Cannon camera
10. Cables and hose 0.24 mm

Software libraries with references to sources

A Node-RED node to talk to an Arduino running standard firmata 2.2
https://flows.nodered.org/node/node-red-node-arduino

Firmata sketch for Arduino
https://www.arduino.cc/en/Reference/Firmata

Node-RED Node to store/retrieve easily information stored into .json file.
https://flows.nodered.org/node/json-db-node-red

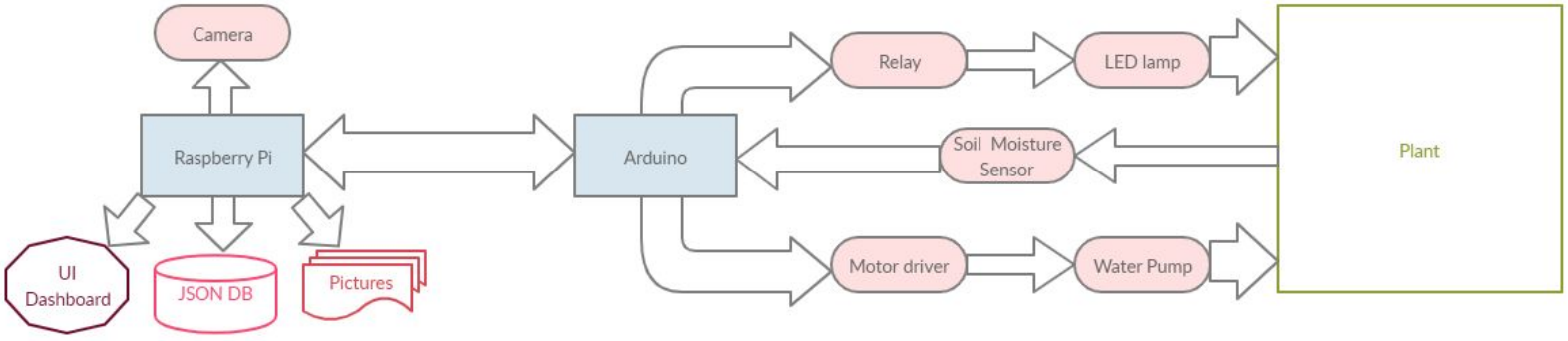
Scheduler for node-red which allows you to enter on/off times as 24hr clock (e.g. 01:10) or suncalc events (e.g. golden Hour). It also allows you to offset times and randomise the time within the offset.
https://flows.nodered.org/node/node-red-contrib-schedex

A Node-RED node to take photos on a Raspberry Pi with a USB camera.
https://flows.nodered.org/node/node-red-contrib-usbcamera

PEAS description

Agent Type	Performance Measure	Environment	Actuators	Sensors
Grow box	healthy and growing plants	plants, room	water pump, led strip	soil moisture sensor, camera

Architecture



The Raspberry is running Node-RED flow, which contains the program logic. The flow is controlling when to turn on and off the lamp, water pump and when to take a picture.
Node-RED also provides a user UI interface with Dashboard, where users can view and control remotely the plant health.
In the program there are plant configurations which describe the name of the plant, when to turn on and off the lamp and what is the minimum level of soil moisture. When the flow is started there is a current plant program which is used. It can be changed with other plant program even there is an option for adding new program, removing existing programs and editing the current plant configuration. The programs are saved into JSON file.
The Node-RED flow is reading the level of soil moisture from the Arduino and if the moisture is low then the water pump is started in intervals for 3 seconds (in order to prevent overflowing). The watering is automatically stopped when the moisture level is good.
Also the flow contains timer which controls the LED lamp and timer for taking picture every 24h.

Hardware architecture:

Raspberry

- connected to Arduino
- local network
- camera

Arduino

- motor driver which is connected to the water pump
- relay which is connected to LED lamp
- soil moisture sensor

Description of system installation and configuration.

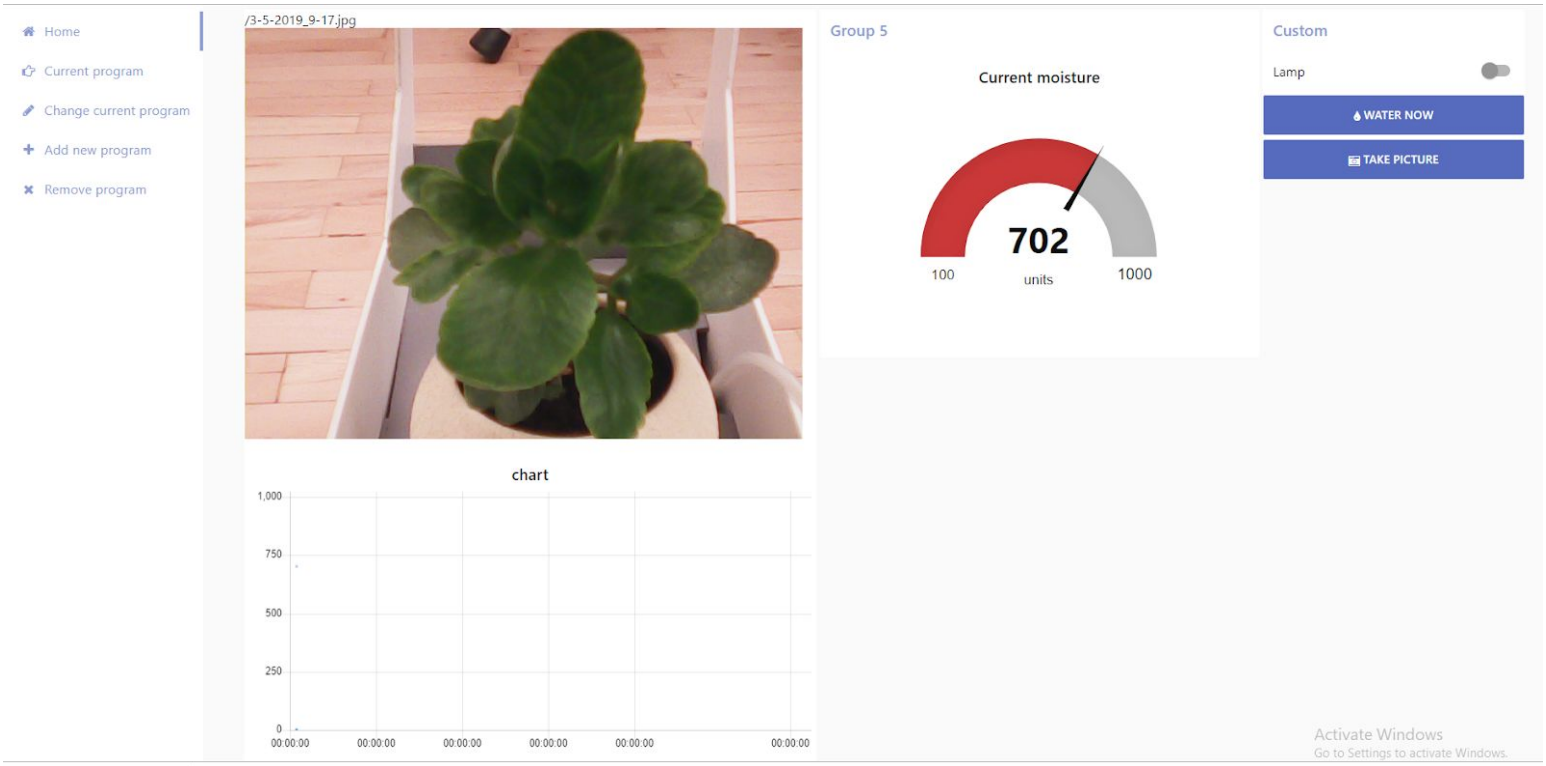
1. Set static ip address on the Raspberry 192.168.8.80
2. Install Node-RED on Raspberry Pi with command:

`bash <(curl -sL https://raw.githubusercontent.com/node-red/raspbian-deb-package/master/resources/update-nodejs-and-nodered)`

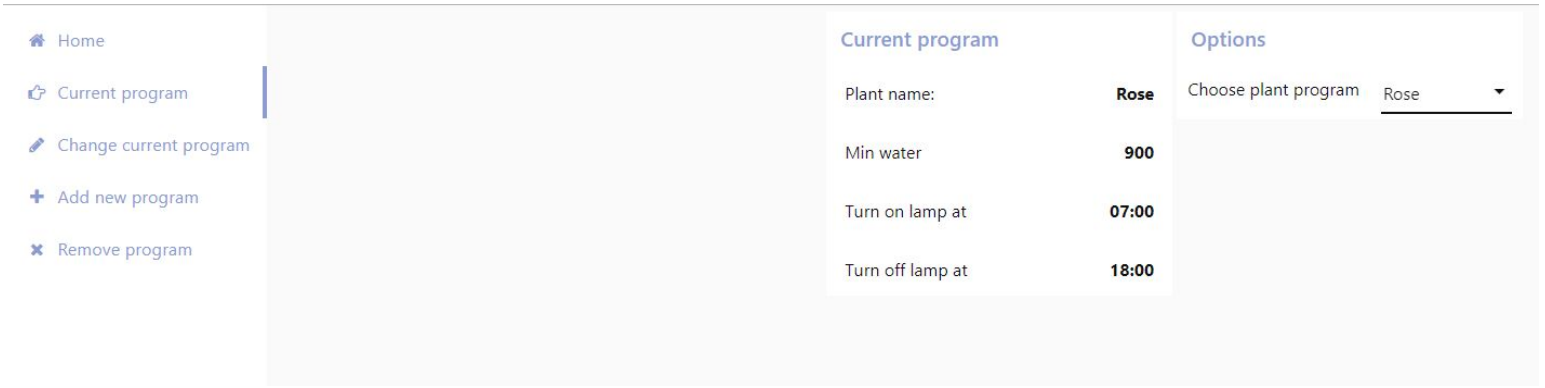
3. Install the libraries described above
4. Run Firmata sketch on Arduino
5. Import the flow into Node-Red
6. Deploy

User manual

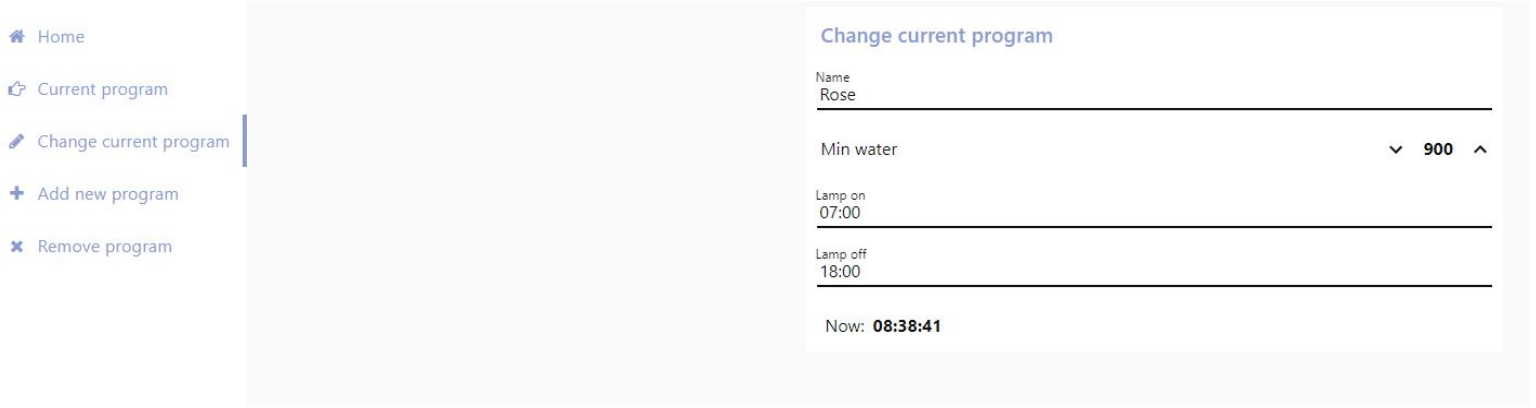
After everything is setup go to: http://192.168.8.80:1880/ui



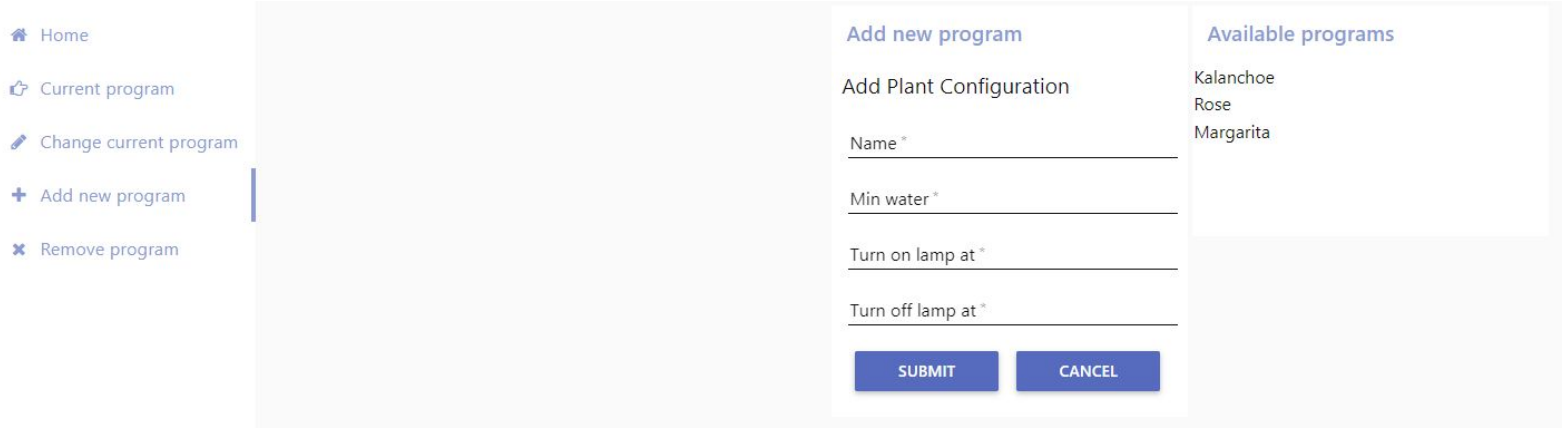
This is the main page, where the user can view the current moisture of the plant, graphic for moisture and when the pump is turned on. Also there are buttons for manual turning on/off the lamp, water for 3 seconds and take a picture.



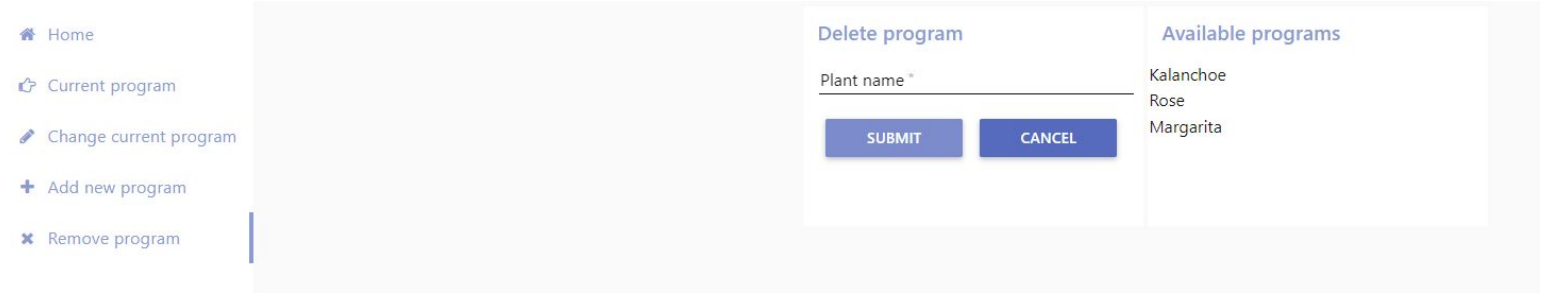
When choosing “Current program” from the navigation, the page is showing the current configuration and its properties. By choosing other plant program, the current configuration is changed.



When choosing “Change current program” the user can edit the properties of the configuration.



Adding a new plant program is by choosing “Add new program” from navigation, after filling the fields, click “SUBMIT” button and the new program name will appear into Available programs panel. Keep in mind that the names of the plants must be different, because the program uses them as identifiers.



For removing an existing plant program go to “Remove program” from the navigation and type the program name. After clicking “SUBMIT” button, the plant program will disappear from Available programs panel.

Conclusion

For future development the project can be expanded for more plants, by adding more soil moisture sensors and water pumps. By this way more plants can be monitored and taken care of.

References to used materials and technologies (including short description of each resource).

A Node-RED node to talk to an Arduino running standard firmata 2.2
https://flows.nodered.org/node/node-red-node-arduino

Firmata sketch for Arduino
https://www.arduino.cc/en/Reference/Firmata

Node-RED Node to store/retrieve easily information stored into .json file.
https://flows.nodered.org/node/json-db-node-red

Scheduler for node-red
https://flows.nodered.org/node/node-red-contrib-schedex

A Node-RED node to take photos on a Raspberry Pi with a USB camera.
https://flows.nodered.org/node/node-red-contrib-usbcamera

How to use 5v Relay
https://components101.com/5v-relay-pinout-working-datasheet