SAVONIA

THIS IS THE WAY TO CHANGE THE WORLD.

RAINMAS

Development and deployment of IoT devices to optimize the use of rainwater for garden irrigation

AUTHOR/S   Margarita Filenko

SAVONIA UNIVERSITY OF APPLIED SCIENCES

Field of Study		
Choose Field of Study		
Degree Programme		
Select Programme		
Author		
Margarita Filenko		
Title of Thesis		
Date 10.08.2025	Pages/Appendices 23	
Client Organisation /Partners		
CAPELENA, EU4DUAL, ESTIA, SAVONIA		

Abstract:
Nowadays, one of the main world problems is water scarcity. A lot of water is consumed senselessly, especially in agriculture. Water management has become an essential responsibility for people. Advanced IoT technologies could be very efficient in water optimization problem. RAINMAS project proposes effective IoT technology for optimizing the use of rainwater for garden irrigation. We developed and deployed demonstration model which gathers rainwater, checks its quality and provides it for the garden irrigation automatically. We showed great results and proved the efficiency of our device for the environment. This technology allows to decrease water usage in agriculture and helps to save more water.

Keywords
Water optimization, IoT, Irrigation, LoRa, Water quality, Automation

CONTENTS

TABLE OF FIGURES

1   INTRODUCTION

According to the latest Water Bank Water Resources Management Overview, water resources are experiencing enormous pressure by most of the countries at present. Estimates show that in the current world situation, humanity will face a 40% scarcity of water by 2030, with demand outstripping supply. It will require a 50% increase in agricultural production and a 15% increase in water withdrawals for feeding 10 billion people by 2050[2]. This indicates that development of water management has become crucial part of the world's global responsibilities. Managing water utilization in the agricultural sector is a particularly essential segment of optimizing water usage.

The significance of securing water quantity shows the increasing necessity of advanced technologies for monitoring the water and managing the water usage. The earlier methods required an individual to take samples of water and testing is performed done manually consuming a lot of time and labor[3]. Nowadays, the enhancement in IoT (Internet of Things) engineering has led to the development of automatic systems with LoRa technologies and sensors implementation.

The problem mentioned earlier correlates with one of the EU4DUAL Grand Challenge – Green Economy. As part of EU4DUAL projects we will focus on this significant problem, more precisely eco-friendly agriculture and food systems, and will provide a method of managing water in agriculture by utilizing IoT technologies and rainwater.

The structure of the paper is organized as follows: Chapter 2 covers the literature review. Chapter 3 covers the hardware description including schematic design. Chapter 4 covers the software description and Chapter 5 covers the results and future work. Chapter 7 covers the conclusions.

2   STATE OF ART

In this chapter, we present the previous studies that include water optimizing methods with usage of IoT technologies and eco-efficient irrigation systems. This study presents an IoT-based architecture for monitoring water level and quality in domestic tanks using customized hardware with 2.4 GHz RF communication and an ESP8266 Wi-Fi module for real-time data transmission. The system was designed and evaluated in the Proteus simulation environment, with calibrated pH and ultrasonic sensors to ensure accuracy. Real-time visualization and monitoring were achieved through the cloud-enabled Virtuino app[3]. This paper proposes an IoT-based Water Quality Monitoring (WQM) system that employs multiple sensors to measure pH, turbidity, water level, temperature, and humidity. The sensor data is processed via a microcontroller and transmitted to the cloud using the ThinkSpeak application for real-time monitoring. The system offers a cost-effective and efficient solution to ensure safe drinking water supply[4]. This study introduces a smart IoT-based fish monitoring and control system to address labor shortages in aquaculture by enabling real-time monitoring and adjustment of water quality and pond conditions. A deep learning model, optimized using Bayesian hyperparameter tuning, was developed to predict California Bass growth with high accuracy (R^2 = 0.94, MSE = 0.0015). The integration of this model into an autonomous feeding system demonstrates the potential of AIoT to enhance aquaculture efficiency, reduce waste, and support sustainable fish farming[5]. This paper reviews the role of IoT-based smart irrigation systems in advancing sustainable agriculture, particularly in support of UN SDG 6 and Target 6.4 on water efficiency. Using a qualitative, secondary-data approach, it highlights how sensor-driven automation improves soil and weather monitoring, optimizes water use, and reduces environmental impact. The study also discusses the benefits and challenges of implementing such systems, offering insights to guide farmers and researchers toward more sustainable and cost-effective irrigation practices[6]. This paper presents the design and development of a real-time water quality monitoring system capable of measuring physiochemical parameters such as flow, temperature, pH, conductivity, and oxidation-reduction potential. The system uses custom-designed sensors with signal conditioning circuits connected to a microcontroller and transmits data via ZigBee modules to a notification node, which provides visual readings and audio alerts when unsafe levels are detected. Validation tests confirmed the system's accuracy and reliability in processing, transmitting, and displaying water quality data[7]. This study proposes a cloud-based smart irrigation system that integrates IoT, renewable energy, and big data to optimize water usage in agriculture, particularly in arid regions. The system connects multiple small-scale smart farms, enabling real-time monitoring, weather-based irrigation control, and informed decision-making through data analysis. Implemented in a solar-powered testbed, the prototype demonstrated significant improvements in water conservation and offers a scalable deployment roadmap for sustainable smart farming[8].

3 DEMONSTRATION MODEL/HARDWARE DESCRIPTION

3.1 Overview

The main objective of the RAINMAS project is to build a demonstration model which consists of two parts: "Tank" part and "Garden" part and implement this model in a real-world environment afterwards. Both parts are connected together by garden hose. "Tank" part is responsible for gathering rainwater and checking its quality. "Garden" part is responsible for checking the necessity of watering the soil in the garden.

The whole architecture is shown in Figure 1. Both parts have the microcontroller as the main component in the structure. We decided on building the whole model around Arduino MKR WAN 1310 with the MKR Connector Carrier which allows us to connect a lot of sensors and add new ones whenever we want. This technology simplifies the process of using sensors and managing the whole system. In addition, our microcontrollers are equipped with LoRa chip and antenna which is a crucial part of data transmission. We send data from devices in the common server and use the connected database for storing it.
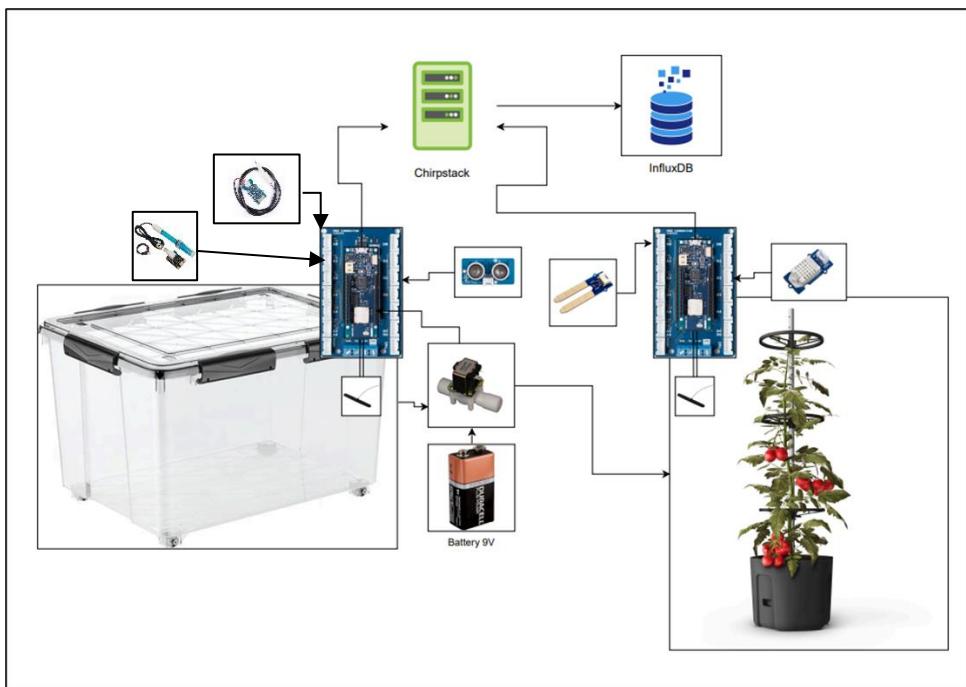
Figure 1. Architecture of the demonstration model

3.2 "Tank" architecture

In this section we will describe the components of the "Tank" part of the demonstration model. The whole architecture of the "Tank" is presented in the Figure 2 and Figure 3. "Tank" part of the model consists of microcontroller, 3 sensors (PH sensor, TDS sensor, Ultrasonic sensor), water tank, solenoid valve and electronic card for the valve.

All the sensors are connected to the microcontroller via MKR Connector Carrier, hence all the sensors have groove connection and supply voltage of +5V, as our Arduino is working with 5V logic.
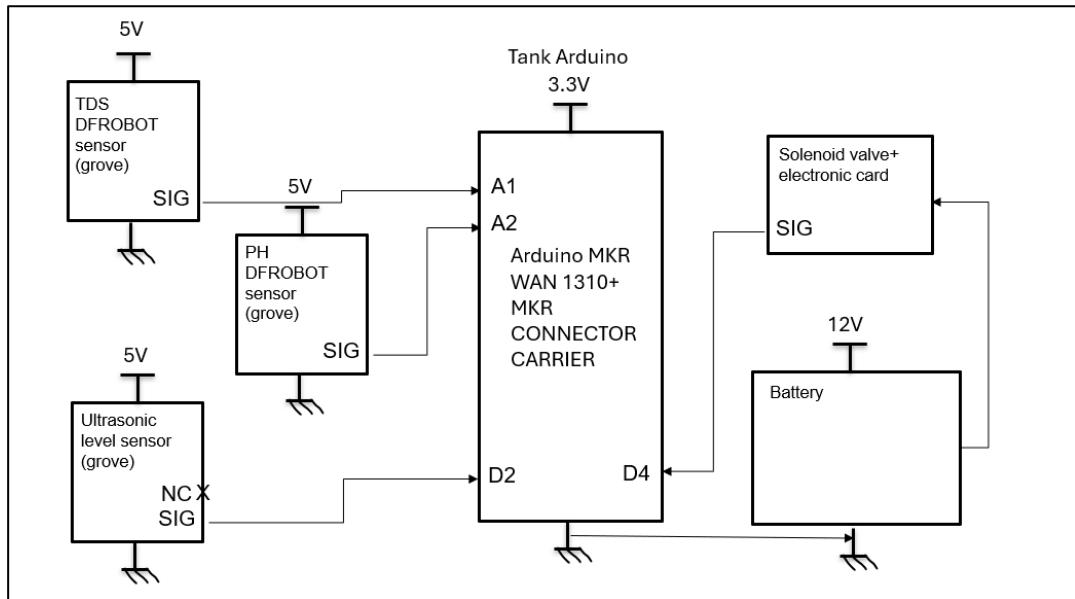
Figure 2. "Tank" architecture a
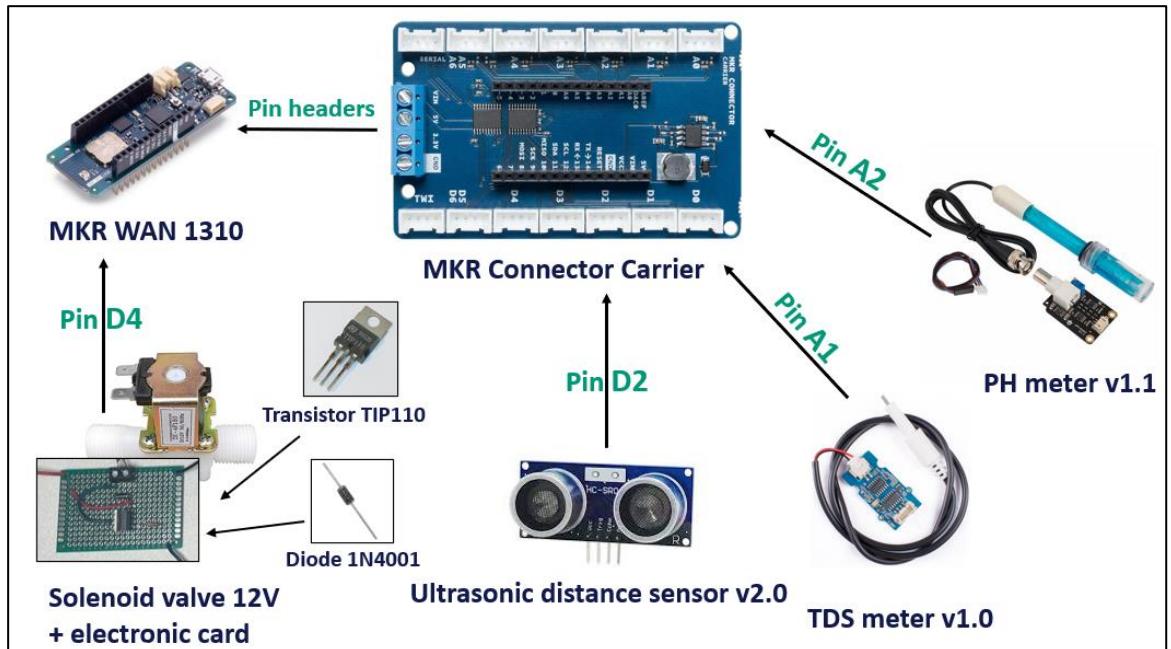
Figure 3. "Tank" Architecture b

3.2.1 Ultrasonic level sensor

We use ultrasonic sensor for measuring the water level in the tank. Model of the sensor is Ultrasonic Ranger v1.0 (grove). Sensor is placed on the top of the tank. The ultrasonic sensor provides a 2cm - 4cm measurement range. The sensor includes an ultrasonic transmitter (trigger), receiver (echo) and a control circuit. It generates a high-frequency sound wave of frequency 40 kHz, and it will be the valuation of the echo received by the sensor measures the interval between signal transmission from the pin trigger and receiving it back to the echo which further determines the distance to an object[4]. In our case, sound wave reflects from the water surface and shows the remaining water level in the tank. Ultrasonic sensor has digital signal output. Data format is floating numbers; measurements are done in centimeters.

3.2.2 TDS sensor

TDS sensor or Turbidity sensor calculates the quantity of total dissolved solids in water, more precisely quantity of their ions in water. The main principle of work is electrical conductivity. The more ions in water, the more electrical conductivity value will it have. TDS sensor passes a small current in the water and then calculates the voltage in it. Consequently, the voltage value is directly proportional to the quantity of ions in water.

The voltage value is converted into the ppm (parts per million) value for storing this data. Data format is integers. Range 300ppm-500ppm is considered to be ideal for irrigation water. Everything that below these numbers has lack of minerals and everything that over these numbers is unsafe for plants. TDS sensor has analog signal output.

tds=(133.42*compensationVolatge*compensationVolatge*compensationVolatge - 255.86*compensationVolatge*compensationVolatge + 857.39*compensationVolatge)*0.5;

3.2.3 PH sensor

PH (potential of hydrogen) sensor measures alkalinity of solution, specifically hydrogen ion concentration. Logic of its working is very similar to TDS sensor. Basic operation principle is related to electrical conductivity as well. PH sensor interacts with hydrogen ions in the water and creates a tiny voltage around the glass probe (part of the ph sensor). There is a reference electrode inside this probe, and the sensor measures the voltage difference between the electrode and the glass probe itself. Subsequently, this number converts to the PH value.

The ideal PH value for irrigation is 6.0-7.5, 5.5-8.5 is still acceptable but depends on the sort of plants. PH sensor has an analog signal output and data format is float numbers.

pHValue = 8.33*voltage-8.485;

3.2.4 Solenoid valve

Solenoid valve is the most significant part of the whole system because it is the key element in the automation process. This device is used to control the water flow by means of electric current. It consists of solenoid and movable plunger. When electrical energy is applied to the coil, magnetic field is generated, which pulls the plunger and lets water go through the valve.

Power supply required for solenoid valve is +12V. We use an external voltage consisting of 8 AA batteries, 1.5V each. In addition, solenoid valve should share the same ground with Arduino because it is necessary to have the same electrical reference for both devices to procure sufficient work. *For the same reason, it works strictly with electronic card which balance the circuit*.

3.2.5 Electronic card

Electronic card is additional part related to solenoid valve. It saves the whole circuit from overvoltage and prevents the influence of huge voltage difference between two devices (valve and microcontroller). The card circuit is shown in Figure 4 and in Figure 5.
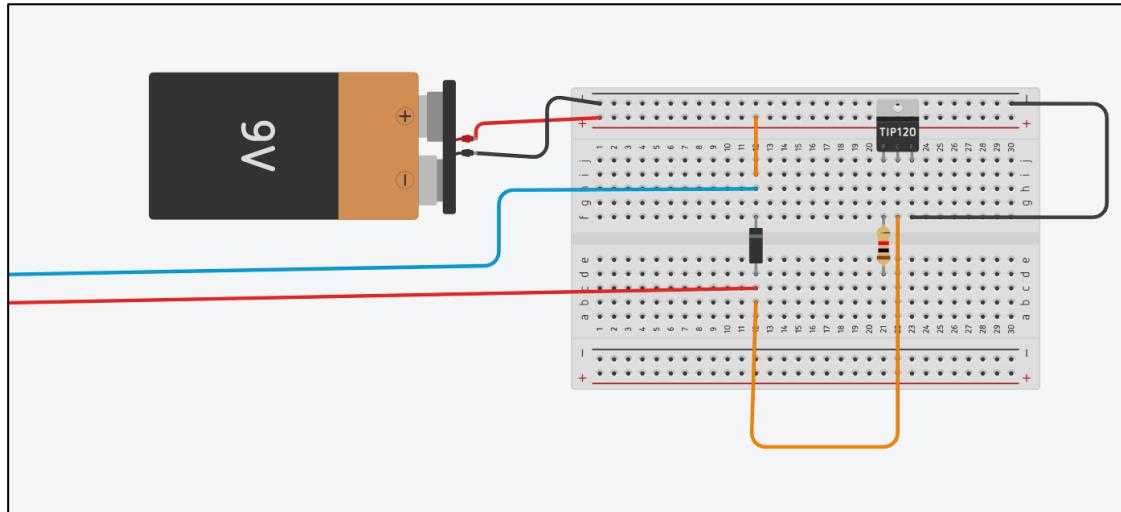
Figure 4. Circuit of electronic card
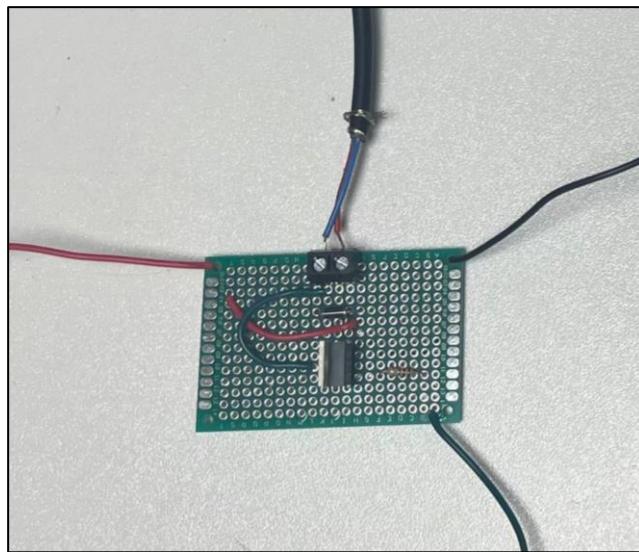
Figure 5. Real electronic card

Electronic card is connected to solenoid valve and microcontroller. Card serves as an intermediary link between these 2 devices. It connects with the Arduino with signal wire which is shown as green wire.

Card consists of 4 components: resistor, transistor, diode and solenoid valve's connector. Resistor value is 1kΩ, transistor is MOSFET type, exact model is TIP110, and diode is 1N4001 model.

3.3 "Garden" architecture

In this section we will describe the components of the "Garden" part of the demonstration model. The whole architecture of the "Garden" is presented in the Figure 6 and Figure 7. "Garden" part of the model consists of microcontroller, 2 sensors (Temperature sensor and Moisture sensor) and certainly any place which is representing garden. In our case it was a flower pot.

Same way as "Tank" part, all the sensors are connected to the microcontroller via MKR Connector Carrier, hence all the sensors have groove connection and supply voltage of +5V, as our Arduino is working with 5V logic.
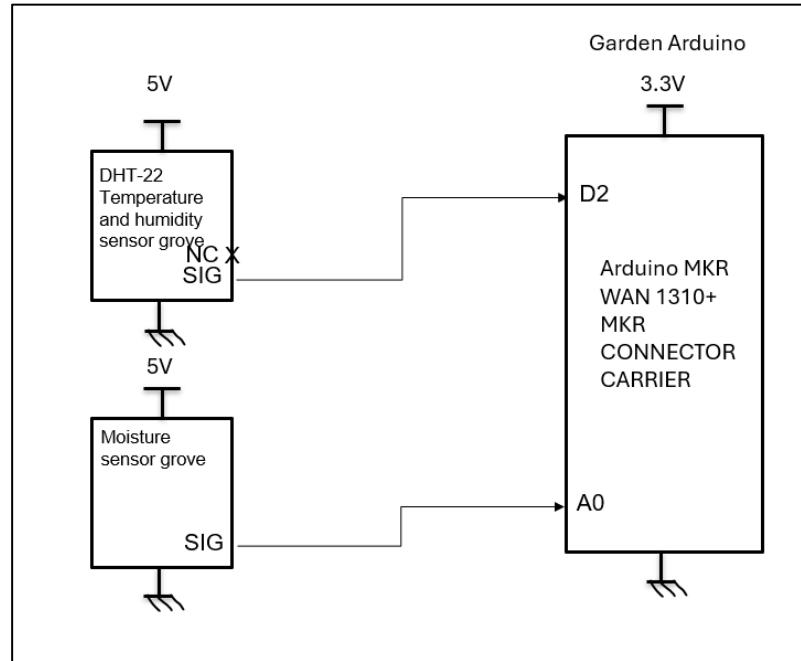
Figure 6. "Garden" Architecture a
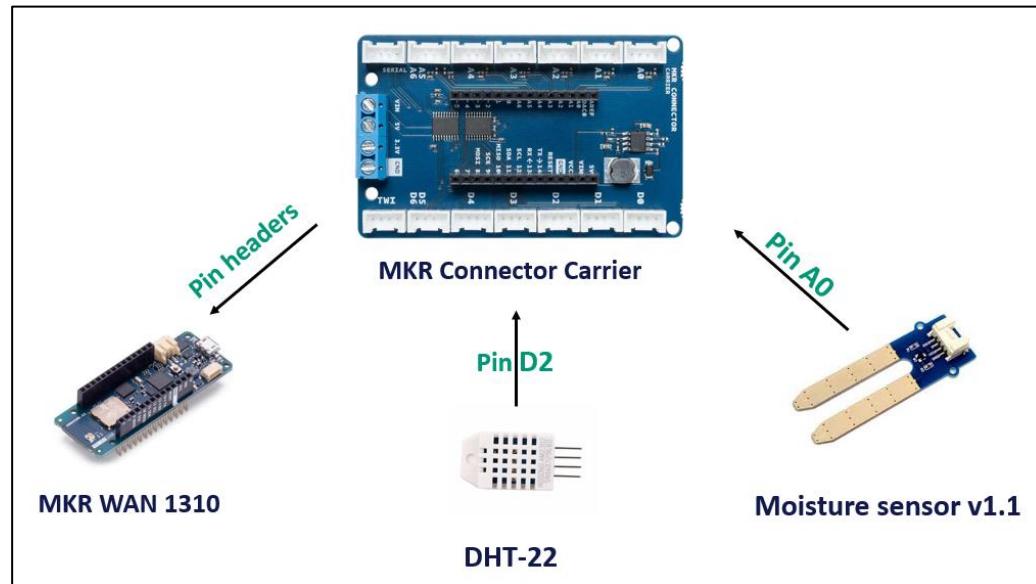
Figure 7. "Garden" Architecture b

3.3.1 DHT-22 sensor

DHT-22 is a simple temperature and humidity sensor. In RAINMAS project it is used only for temperature measurements. It is placed in a pre-root plants zone and collects temperature data of the surrounding atmosphere there. The sensor comes with a Negative Temperature Coefficient (NTC) for temperature measurement.

DHT-22 is sensor with digital signal output. Data format is integer numbers and measurements show in Celsius degrees.

3.3.2 Moisture sensor

Grove moisture sensor v1.1 is used for collecting soil moisture data. It is placed directly into the soil with its pads which work as probes. These probes work as a variable resistor. Water in the soil conducts electricity between the probes. Resistance changes depending on the humidity of soil. Consequently, the dryer the soil, the higher the resistance, because wet soil is more conductive than dry soil.

Sensor is measuring electrical resistance of the soil. This data can be decoded as so: 0-300 -> dry soil; 300-700 -> humid soil; 700-950 -> water. Consequently, in instances where the number is less than 300, the irrigation of the soil is necessary. Moisture sensor has an analog signal output and data format is integer numbers.

4   SOFTWARE DESCRIPTION

4.1   Algorithm

The system has its own algorithm and sequence of certain events which happen constantly. Every sensor is obligated to work simultaneously with others, and each event happens in specific order. Schematic of this algorithm is shown in Figure 8.

Even though everything always works at the same time, there are some nuances which change the normal sequence sometimes. For instance, PH and TDS sensors differ from the other sensors and their codes work differently. These codes are very similar to each other, and they work on the basis of arrays. Due to high sensitivity of these sensors and submerged placement, it is easy to get wrong results because of the noise. Therefore, instead of one measurement these sensors gather 10 samples, then calculate the average value and send this value to the server. It takes more time compared to other sensors calculations. Consequently, if we just have started the program, we will not be able to get data from all the sensors in the first place because arrays of TDS and PH sensor's codes will just be empty. Nevertheless, after 2 minutes of work, program starts sending data from all the sensors normally.

Another important clarification is that one part of the code works infinitely, so it makes decision based on the latest data which can be detected in the server.

The whole algorithm works once per certain period of time which can be changed by user depends on his/her purposes.
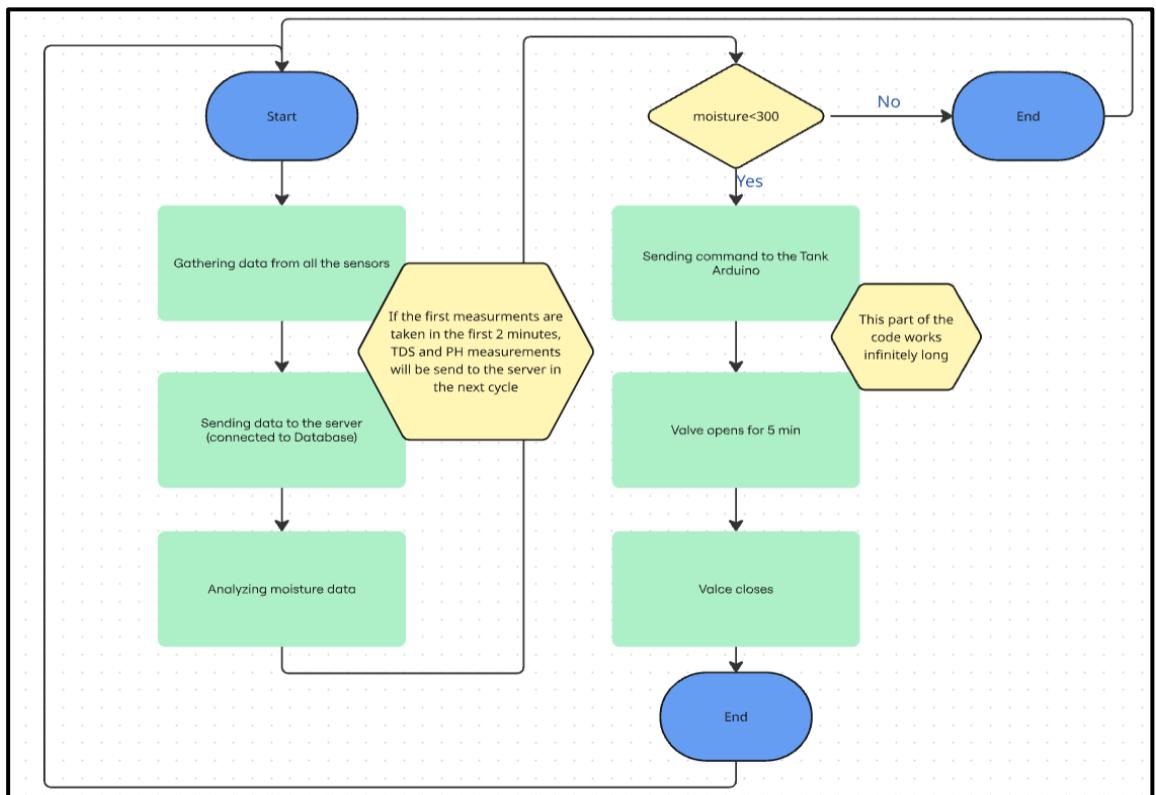

Figure 8. Algorithm

4.2 Server

Just after any sensor receives its measurements results, data from it is transmitted to the global ESTIA server. The name for this server is "ESTIA IoT Platform" and it works on the base of ChirpStack. "ESTIA IoT Platform" operates as a server for all devices that have access to it through the common gateway.

Furthermore, server is connected with the common database where all the data is stored. Database works on the basis of InfluxDB. Data can be sorted by time, device, location and separate sensor measurements. Example of the database is shown in Figure 9.

Server procures communication between node and server and between two nodes through the server.
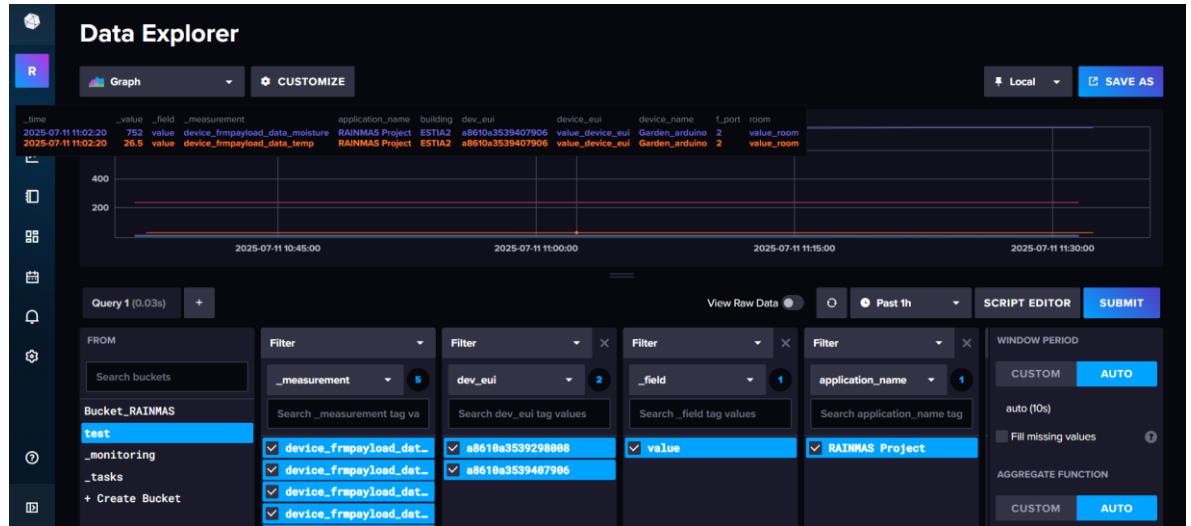
Figure 9. Database

4.3 Code

With regard to the programming element, we used 2 programming languages during work on the RAIMAS project: C++ language was utilized for Arduino's coding and data transmission functions, and the Python language for the automation of the system. Most of the coding was done on C++. Structure of the code consists of 18 different files: 10 files with sensor's codes and libraries, 2 files for solenoid valve's code and its library, 1 file with sensor's configuration, 1 file with sensor's credentials, 1 file with sensor's ids, 2 files responsible for data encoding and transmission by using LoRa technology and 1 main file. All the "libraries" files are simplifying the connection between the main file and all the other significant files. Files with sensor's ids and credentials are needed to connect devices to the server. LoRa files utilized for encoding the data from all the sensors using a byte-oriented format, where each sensor ID is followed by its corresponding measurement value. Afterwards, data is transmitted to the server where another C++ code decodes this data and stores it in the database in normal format after. Main file launches all the other files and sets the delay for the whole program.

Another part of the program is one python file. This file launches separately from the main program and runs infinitely long until user breaks it. Program reads the latest moisture sensor data from the server and if value is less than 300, it sends command to the "Tank" node which invokes solenoid valve to open and starts the irrigation process. After 5 minutes valve goes back to its "closed" status.

5 RESULTS AND PROSPECTS

5.1 Sensor Calibration

Sensor calibration is one of the most essential components in every resembling research project. RAINMAS is not an exception, and we calibrated all the 5 sensors that were used during the research. Tests showed that every single sensor operates properly, and range of measurements is acceptable in terms of the project objectives. Process of calibration is shown in Figures 10-14 and table with results of calibration is present in Figures 15-18.

All the sensors were tested in diverse conditions. Moisture sensor was tested in dry soil, humid soil and water. Ultrasonic sensor was tested with the water level of 5 cm, 10 cm and 20 cm. An additional cm appears due to the transmitter and receiver parts of the sensor which create an extra centimeter during measurements. This defect was fixed by changing the code formula after calibration. TDS sensor was tested in tap water, in salty water (salt consists of a lot of minerals, which increase the quantity of ions in water) and in rainwater. PH sensor was tested in special solutions for calibration with PH of 7.01 and 4.01, and with regular tap water.

Figure 10. Moisture sensor calibration

Figure 11. Ultrasonic sensor calibration
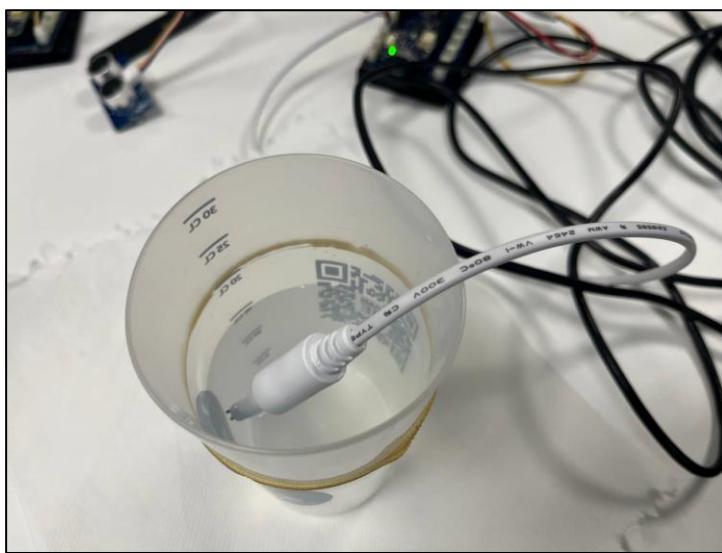
Figure 12. TDS sensor calibration
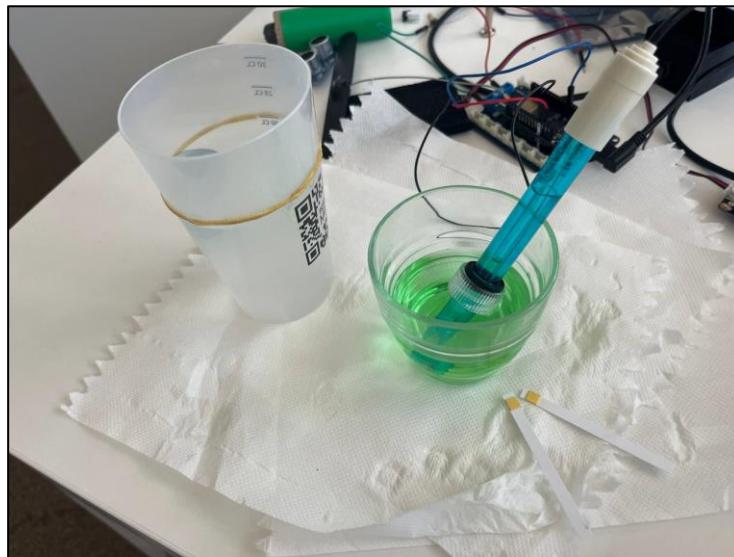
Figure 13. PH sensor calibration a) pH 7.01 solution

Figure 14. PH sensor calibration b) pH 4.01 solution

Soil moisture sensor	03.06.2025																												
	Dry									**Humid**										**Water**									
14:02	14:03	14:04	14:05	14:06	14:07	14:08	14:09	14:10	14:11	14:12	14:13	14:14	14:15	14:16	14:17	14:18	14:19	14:20	14:21	14:22	14:23	14:24	14:25	14:26	14:27	14:28	14:29	14:30	14:31
164	129	102	109	169	161	197	231	262	273	703	691	686	686	686	687	688	688	690	690	755	797	768	758	753	740	762	748	765	764

Figure 15. Soil moisture sensor calibration results

Water level sensor																													
	5 cm									**10 cm**										**20 cm**									
14:42	14:43	14:44	14:45	14:46	14:47	14:48	14:49	14:50	14:51	14:58	14:59	15:00	15:01	15:02	15:03	15:04	15:05	15:06	15:07	15:10	15:11	15:12	15:13	15:14	15:15	15:16	15:17	15:18	15:19
6.66	6.68	6.68	6.78	6.7	6.76	6.68	6.8	6.82	6.78	11.4	11.5	11.4	11.4	11.4	11.4	11.4	11.4	11.4	11.4	21.2	21.2	21.6	21.7	21.6	21.2	21.3	21.3	21.2	21.2

Figure 16. Water level sensor calibration results

TDS sensor	07.07.2025																													
		Tap water										**with salt**										**Rain water**								
	10:46	10:47	10:48	10:49	10:50	10:51	10:52	10:53	10:54	10:55	10:56	10:57	10:58	10:59	11:00	11:01	11:02	11:03	11:04	11:05	11:06	11:07	11:08	11:09	11:10	11:11	11:12	11:13	11:14	11:15
	221	221	221	221	221	221	221	221	221	221	1130	1130	1130	1130	1130	1130	1130	1130	1130	1130	506	506	506	506	506	506	506	506	506	506

Figure 17. TDS sensor calibration results

pH sensor	10.07.2025																													
		pH4										**pH7**										**Tap water**								
	10:36	10:37	10:38	10:39	10:40	10:41	10:42	10:43	10:44	10:45	10:46	10:47	10:48	10:49	10:50	10:51	10:52	10:53	10:54	10:55	10:56	10:57	10:58	10:59	11:00	11:01	11:02	11:03	11:04	11:05
	4.04	4.03	4.03	4.02	4.02	4.01	4.01	4.01	4	4.01	7	7	6.99	6.99	6.99	6.98	6.98	6.98	6.98	6.99	7.62	7.63	7.63	7.63	7.63	7.62	7.63	7.62	7.63	7.63

Figure 18. pH sensor calibration results

5.2 Results

We developed the demonstration model of IoT device for optimizing the rainwater for garden irrigation. Final results are shown in Figures 19-20.

Figure 19. Demonstration model a
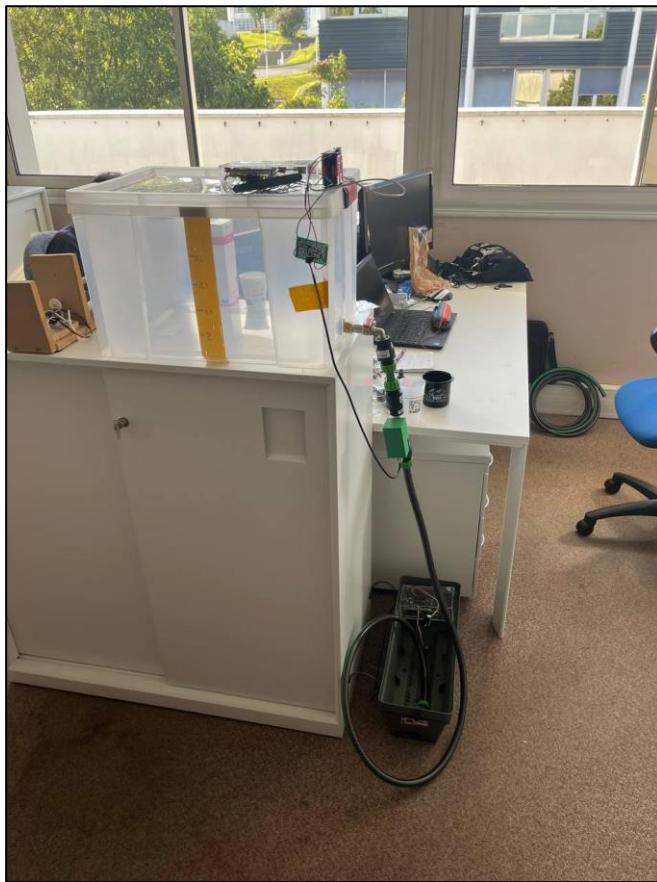
Figure 20. Demonstration model b

In addition, some isolation cases were created for the solenoid valve (Figure 21) and Arduino nodes (Figure 22). We used 3D-printing technology to create case for the solenoid valve, and we utilized plastic carving machine for isolation boxes for nodes.

Figure 21. Solenoid valve case
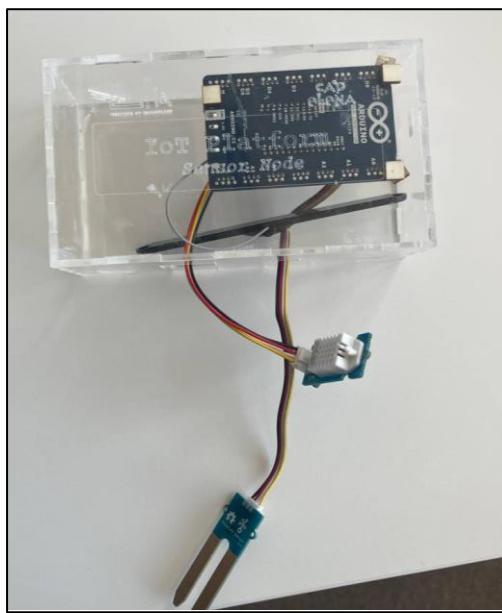
Figure 22. Arduino protection box

5.3   Future work

We have outlined several plans for the future work. First of all, we will test the demonstration model in the real-world conditions and analyze its work.

Secondly, we will solve the TDS sensor and PH sensor problem. The difficulty lies in inaccurate results during simultaneous work. TDS sensor and PH sensor have similar working principles, based on electrical conductivity. Both sensors are changing the voltage in the water a little bit. However, both sensors are very sensitive as well. Therefore, even small change in the voltage creates a noise which affects both sensor's measurements. We explored 2 possible solutions for this problem. First, use separate ground for these sensors. If we divide the ground for them, TDS current will not affect the PH measurements so much. Second, refuse from utilizing them simultaneously. It is possible to disconnect one sensor while another is gathering data – simply providing voltage to only one sensor at a time. Both sensors can work by timer and by rotation.

Thirdly, we will think about improving isolation for all the parts that could easily get damage from water.

Finally, we will implement more sensors for checking water quality by other parameters and we will add the automated water recycling system. For instance, if water has bad quality and cannot be used for irrigation, we will forward it for other utilization, e.g. toilets.

6   CONCLUSION

In the time of water scarcity all over the world, efficient water management is a primary task for everyone. With the developing IoT technologies we can optimize water usage successfully, especially in agriculture, and save huge amounts of water which is used senselessly for irrigation. In RAINMAS project we developed and deployed efficient IoT devices that helps in optimizing the use of rainwater for garden irrigation. The proposed system with Arduino MKR WAN 1310 and grove sensors works successfully and shows acceptable results. Automation process operates without any mistakes and plants are irrigated whenever they require. Algorithm has a strong structure and every part of the demonstration model works according to it. Code is structured well and has an easy-understanding logic. Data encodes and decodes correctly, information stays secure and all the measurements stored in the database.

REFRENCES

Artificial intelligence has been used in the work as follows:

ChatGPT 2024. OpenAI. GPT-3.5. Accessed for language check, May 2024.
https://chat.openai.com.

1. Water Resources Management Overview. Available online:
   https://www.worldbank.org/en/topic/waterresourcesmanagement (last updated: 5 October 2022).

2. EU4DUAL Three Grand Challenges. Available online: https://eu4dual.education/three-grand-challenges/ .

3. Rajesh Singh, Mohammed Baz, Anita Gehlot, Mamoon Rashid, Manpreet Khurana, Shaik Vaseem Akram, Sultan S. Alshamrani, Ahmed Saeed AlGhamdi. "Water Quality Monitoring and Management of Building Water Tank Using Industrial Internet of Things". 2021.

4. Sathish Pasika, Sai Teja Gandla. "Smart water quality monitoring system with cost-effective using IoT". 2020.

5. Min-Chie Chiu, Wei-Mon Yan, Showkat Ahmad Bhat, Nen-Fu Huang. "Development of smart aquaculture farm management system using IoT and AI-based surrogate models". 2022.

6. Khaled Obaideena, Bashria A.A. Yousef, Maryam Nooman AlMallahi, Yong Chai Tan, Montaser Mahmoud, Hadi Jaber, Mohamad Ramadan. "An overview of smart irrigation systems using IoT". 2022.

7. NIEL ANDRE CLOETE, REZA MALEKIAN, (Member, IEEE), AND LAKSHMI NAIR, (Member, IEEE). "Design of Smart Sensors for Real-Time Water Quality Monitoring". 2016.

8. Bouali Et-taibi, Mohamed Riduan Abid, El-Mahjoub Boufounas, Abdennabi Morchid, Safae Bourhnane, Tareq Abu Hamed, Driss Benhaddou. "Enhancing water management in smart agriculture: A cloud and IoT-Based smart irrigation system". 2024.

APPENDIX 1: CREATIVE COMMONS LICENCES – IMAGE COPYRIGHT

Image copyright can be ensured by setting search criteria in the image search that allow the image to be used and edited. An image taken from a source can be edited, but you need to check with the copyright whether this can be done. For example, cropping an image or annotating an image is editing. If a finished map is used to mark off an area, it is regarded as editing.

This is how to check image copyright: If you search for an image using Google's Image Search tool, then use the Searching Tools and select Access. Select one of the following options: A derivative work means that you have edited the image in question.

Attribution (BY) The work may be copied, distributed, shown and performed in public and derivative works may be created, provided that the name of the author or copyright holder is duly mentioned.

Non-Commercial (NC) The work may be copied, distributed, shown and performed in public and derivative works may be created only when they are not used for commercial purposes.

No Derivative Works (ND) The work may be copied, distributed, shown and performed in public, but no derivative works may be created from it.

Share Alike (SA) Derivative works may only be distributed under the same license as the original work.

Indicating CC licenses in combinations

The licenses are written with the letter combination CC first. This is followed by a space and a dashed list of abbreviations for license terms. The first of the conditions is always BY, followed by a possible NC and followed by a possible third condition.

The licenses obtained by combining the terms are:

- Attribution (CC BY)
- Attribution – Share-alike (CC BY-SA)
- Attribution – No Derivative Works (CC BY-ND)
- Attribution – Non-Commercial (CC BY-NC)
- Attribution – Non-Commercial – Share-alike (CC BY-NC-SA)
- Attribution –Non-commercial – No Derivative Works (CC BY-NC-ND)

In addition, there is a special CC0 license, which allows the author to waive all rights to the work to the extent permitted by law. In Finland, the author cannot waive his moral rights, thus the name of the author / photographer must always be mentioned.

Example: FIGURE 1. A detail of the Colosseum in Rome (Laamanen 2015, CC BY-SA)

The image may be edited and shared, but the original creator must be mentioned. A modified image may only be shared under the same license as the original image.