Water Control System for Al Nakheel Tree Using Fuzzy Logic with Trend Analysis

Conference Paper · October 2020

DOI: 10.1109/iCCECE49321.2020.9231256

3 authors:

Moayad Al-Falahi
Muscat College

Ferddie Quiroz Canlas
Muscat College

Sarachandran Nair
Muscat College

Water Control System for Al Nakheel Tree Using Fuzzy Logic with Trend Analysis

Moayad Salim Al Falahi
Department of Computing
Muscat College
Muscat, Oman
15128@email.muscatcollege.edu.om

Ferddie Quiroz Canlas
Department of Computing
Muscat College
Muscat, Oman
ferddie@muscatcollege.edu.om

Sarachandran Nair
Department of Computing
Muscat College
Muscat, Oman
saran@muscatcollege.edu.om

***Abstract*—The proposed water control system for Al Nakheel Tree is a multifaceted implementation model that combines the Fuzzy Logic Technique, Internet of Things (IoT), and Cloud Technologies. The NodeMCU microcontroller collects data from the temperature, and soil moisture sensors, and then performs Fuzzy Logic to determine the right amount of water the motor pump should supply to the palm trees. The ages of the palm trees are also used to determine the frequency of the watering process. These data, including the status of the system, are stored in a cloud database. Using the mobile application which was securely paired to the microcontroller helps the farmer to monitor the status and control the system. The system switches from Fuzzy Logic operation to trend analysis using the moving averaging method when one of the sensors fails.**

Keywords—palm trees, water control, internet of things, fuzzy logic, trend analysis

I. INTRODUCTION

Since the dawn of human history, farming is considered a vital endeavor not only to sustain human lives but also for other aesthetics purposes. It is known that the irrigation of plants is necessary to sustain the life of plants. The method of farming varies depending on the type of plantation and surrounding conditions (such as weather, temperature, and others). The process of cultivating plants has evolved considerably over the ages from manual to automated [1].

Palms trees (Al Nhakeel in Arabic) are considered as one of the most famous plantation in Gulf countries, such as Oman and Saudi Arabia. There is also prevalence of the tree on other Arabic speaking countries in Africa such as Egypt. Dates fruits are always part of every Omani families' daily meal. It has been observed that Omani women make sweet delicacies using the said fruit for specials occasions. Furthermore, both raw and ripe fruits are commonly seen in the market place alongside other important commodities [2].

Dates trees require meticulous attention when it comes to watering. They have varying needs of amount of water based on some parameters such as temperature, weather conditions, and palm age. They need more watering during the summertime than in the wintertime. According to Al Hyari [3], the following conditions must be observed in order to successfully cultivate palm trees: (1) If the temperature is high and the palm trees age is more than four years, then the palm tree needs one-time watering every week with more amount of water. (2) If the temperature is low and the palm tree's age is more than four years, then the palm tree needs one-time watering every week with less amount of water. (3) If the temperature is high and the palm trees age is less than or equal to four years, then the palm tree needs one-time watering every three days with more amount of water. (4) If the temperature is low and the palm trees age is less than or equal to four years, then the palm tree needs one-time watering every three days with less amount of water.

This paper attempts to develop an implementation model of a water control system for Al Nakheel trees using Fuzzy Logic to assist farmers. The said model involves an IoT based module that automates the settings of the water amount for the whole year. Also, using cloud databases, it records and retrieves the trends in temperature and soil moisture changes and uses the records as backup information in case the temperature sensor or soil moisture sensor fails.

II. REVIEW OF RELATED STUDIES

A. Internet of Things (IoT) Based Water Control System

A group of researchers from the Universitas Islam Sultan Agung in Indonesia [4] proposed an IoT based plant sprinkler system utilizing message queue telemetry transport (MSTT) protocol to enable users to access the system online. The prototype consists of Wemos D1 Uno microcontroller, solenoid sensors and controllers, temperature, humidity, and soil moisture sensors. The microcontroller (connected to a WIFI) processes the information fed by the sensors and uploads them to the cloud. The user then monitors the status of the system, specifically that of the sensors using a web client or their mobile phone. The study did not specify how the microcontroller manages the amount of water nor the time duration of the sprinkler. Also, it is not clear if the user can control the sprinkler using the interface provided.

Reghukumara and Vijayakumar [5] used the same MSTT protocol to enable the Arduino microcontroller to update the data in the cloud. In their model, the pH sensor was introduced to help monitor and predict the plants' health. Similarly, the prototype is connected to WIFI and utilizes the same cloud platform-Adafruit IO to store the data collected by the microcontroller. The microcontroller makes decisions based on a rule base and sensor threshold values giving emphasis when to pour water and the health of the plants (which is later sent as an email to farmers). Prediction of the plants' health was performed using a decision tree model which was encapsulated in the FindS algorithm. However, the study did not indicate where and how the algorithm was implemented. Also, the Boolean representation of the pouring of water did not indicate the amount of water and the duration of the pouring of water.

B. Water Control Systems Using Fuzzy Logic

Ying et al. [6] of the Universiti Tun Hussein Onn Malaysia proposed a water management system for oil palm nursery. Using Fuzzy Logic, the said system determined the amount of water-based on two inputs: (1) rainfall expressed in millimeter and (2) time to water expressed in a 24-hour unit. Furthermore, the system provides a web-based interface where a user can encode the parameters. In return, the system provides the calculated amount of water and visualization. Furthermore, it keeps records using its time data management tool. The study acts like a recommender system and no further automation was conducted both on the collection of information and the control of the amount of water to be supplied to palm trees.

Rahim et al. [7] proposed a simulated fuzzy logic-based irrigation system for tomato plants using soil moisture and plant root length as parameters. The fuzzy logic calculations were conducted using MATLAB Simulink. The test results have shown a significant difference in the efficiency of the simulated fuzzy logic controller system and the irrigation block diagram based on-off controller in terms of power and water consumption. The proposed model has never been tested with real hardware and software implementation.

C. Synthesis and Gap Analysis

None of the existing studies have made a dedicated model for Palm Trees and too general in their approach. Although there are existing similar projects that tackle the water control domain, it is evident in the literature surveys that some projects either used IoT but not fuzzy logic or fuzzy logic but not IoT. There is a project that attempted to incorporate both IoT and Fuzzy logic. However, the project was done in a simulation way and not implemented using real hardware and software. Also, existing systems did not apply any intervention in case one of the sensors fail.

III. METHODOLOGIES

A. Requirement Analysis

An expert from the Ministry of Agriculture and Fisheries of the Sultanate of Oman was consulted to provide the necessary information needed for the proposed model. The interview also confirmed that no similar existing system is in placed nor registered as currently in use within the sultanate. Also, testing was conducted by the said expert to give feedback on the performance of the system.

B. Proposed Architecture

The proposed model as depicted in Fig. 1 comprises seven (7) major components. Both the mobile application and the Arduino microcontroller require an internet connection to write and retrieve data from the Firebase and MySQL databases. The Apache webserver houses the PHP scripting language that enables the mobile application to read trends (parsed in a form of JSON array from MySQL. These trends consist of data from sensors (temperature and soil moisture), amount of water supplied to Palm Trees, and date of watering. Similarly, after processing sensor inputs and water amount, the Arduino microcontroller uses PHP in reading and writing the same data to MySQL.

Fig 1. Architecture of the Proposed Model

Both the mobile application and Arduino microcontroller use Software Development Kit (SDK) that has powerful libraries that allow both components to connect directly to the Firebase database. Firebase stores the user's settings for the Arduino and manipulates it through the mobile application. The same SDK enables the mobile application to receive push notifications about the sensor and microcontroller status recorded in Firebase in real-time.

For testing purpose, the architecture is implemented using free web and database services.

C. Fuzzy Logic Concept

The Fuzzy Logic technique is versatile in designing systems involving multi-parameter and non-linear control models. Normal computer technology uses either true or false (0 or 1) for reasoning. Therefore, many complex problems that need more precision cannot be solved by that approach [8].

The Fuzzy Logic-based systems involve not only the typical Boolean truth values of 0 and 1 but also any values in between them. This means variables can be categorized into true, almost true, or partially true. Therefore, Fuzzy Logic allows the system to take a varying degree of precision where the statement can be expressed as very low, low, medium, high, etc. [9].

D. Hardware and Software

a) The Microcontroller and Sensors: The layout of the prototype is shown in Fig. 2. Each component is labeled with numbers whose names are indicated in TABLE I.

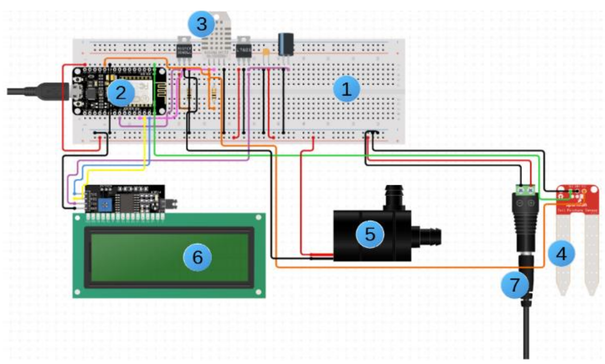

Fig 2. The Hardware Components

M. S. Al Falahi, F. Q. Canlas and S. Nair, "Water Control System for Al Nakheel Tree Using Fuzzy Logic with Trend Analysis," 2020 International Conference on Computing, Electronics & Communications Engineering (iCCECE), Southend, United Kingdom, 2020, pp. 143-149, doi: 10.1109/iCCECE49321.2020.9231256.

For the prototype, the NodeMCU is inserted into the breadboard and acts as the main component of the watering control system.

TABLE I. NAME OF COMPONENTS

Hardware Name	***Number***
Breadboard	1
NodeMCU	2
Temperature Sensor	3
Soil Moisture Sensor	4
Water Pump	5
LED Display (Serial Monitor)	6
Wall Adapter Power Supply	7

All the other components are connected to the NodeMCU through the same breadboard. For exceptional cases, jumper wires are used to connect other components. The system is powered using an adapter.

During the initial setup, the serial monitor is used to enter the WiFi SSID and WiFi key to connect the microcontroller to the internet. Similarly, it displays the verification key which must be entered into the mobile application to connect to the watering system and control it later.

The temperature sensor is placed in the open air. The soil moisture sensor will be placed nearby the plant. The temperature sensor and the soil moisture sensor provide inputs to NodeMCU as a basis for every watering operation.

Based on the results of the Fuzzy Logic Analysis, the NodeMCU turns on or off the water pump thus providing the appropriate amount of water.

b) *Firebase*: It is a backend as a service provided by Google for web and mobile application development. The developer can log in to the Firebase console website using Google account. Firebase is a server, database, and API at the same time. Also, Firebase can be called a Realtime Database which is a very fast service that can be used to sync the data automatically between the server and client, and gives a fast response to the client [10]. By using the Firebase service in app development, it will be easy to manage the cloud and the database and other services in one place. Fig. 1 shows the difference between the traditional app development and Firebase app development [11].

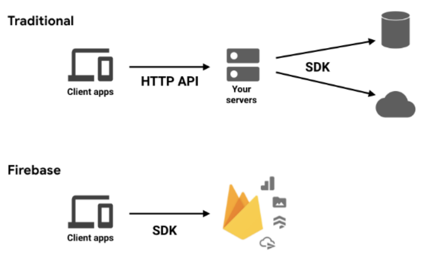

Fig. 2 Comparison Between Traditional and Firebase App Development

c) *MySQL, PHP and JSON:* MySQL is a database server, an open source software known for many features such as data security, on-demand scalability, high performance, round the clock uptime (using clusters) and comprehensive transactional support [12] best suited for storing multirow of data, i.e. trends. It works directly with PHP; a cross platform web scripting language has built-in database support and strong library [13] suitable for querying the database and performing computations. Database queries and computation results are transported over the web using JSON (JavaScript Object Notation) objects, a data format specified by RFC 4627. JSON provides a schema for organizing and sharing data for multiple platforms, and popularly used for parsing data from the database (through PHP) to mobile application [14].

d) *Android Studio:* The mobile application was developed using Android Studio, an integrated development environment IDE used to develop Android mobile applications. It uses Java and Kotlin as programming languages. It uses extensible markup language XML to design the application. One of the nice features in the Android studio support emulator which simulates the Android operating system. The application is uploaded in the mobile in APK format [15].

E. Context and Use Case Diagram

During the initial setup, WIFI login credentials and mobile application verification key are encoded in the microcontroller using the serial monitor connected to the computer. The mobile application supplies all the necessary information about the system. The microcontroller collects and processes the information fed by the temperature and soil moisture sensors. Using Fuzzy Logic, the microcontroller calculates the right amount of water which is then translated to appropriate duration of turning the water pump on and off. This information is then stored in the online database. Fig. 3 and Fig. 4 show the detailed interrelation and functions of each component in the system.

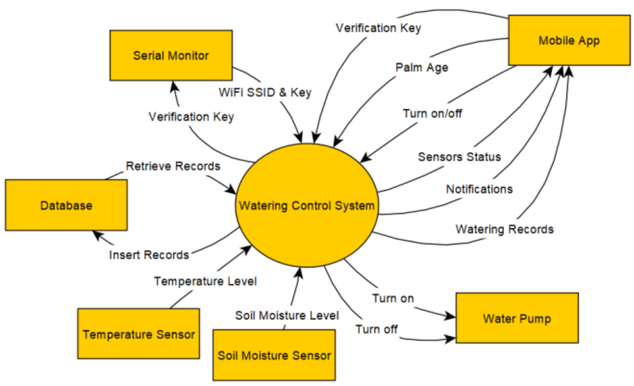

Fig. 3 Context Diagram

M. S. Al Falahi, F. Q. Canlas and S. Nair, "Water Control System for Al Nakheel Tree Using Fuzzy Logic with Trend Analysis," 2020 International Conference on Computing, Electronics & Communications Engineering (iCCECE), Southend, United Kingdom, 2020, pp. 143-149, doi: 10.1109/iCCECE49321.2020.9231256.

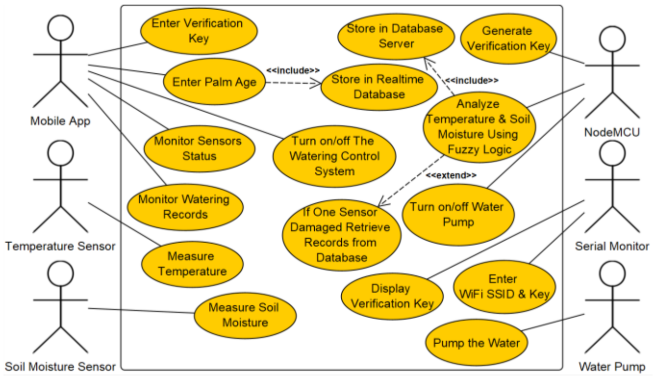

Fig. 4 Use Case Diagram

IV. Results and Discussion

A. Fuzzy Logic Membership

The Fuzzy Logic system takes two inputs: Temperature and Soil Moisture and one output: amount of water.

The temperature expressed in degree Celsius (Fig. 5) is set to 0 as minimum and 50 as maximum. These values are selected because the lowest temperature level recorded in Oman is 4.4 degrees [16], and the highest temperature level recorded is 49.8 degrees [17]. On the other hand, the soil moisture (Fig. 6) is set to 0-100 range since the moisture sensor returns values in percentages.

The for water (as per the Ministry of Agriculture and Fisheries) is set to 0 to 250 expressed in liters. Fig. 7 shows the range of water consumption.

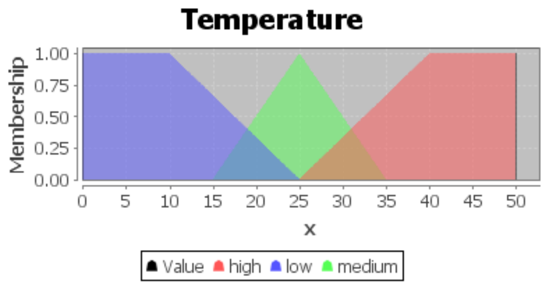

Fig. 5 Temperature Range

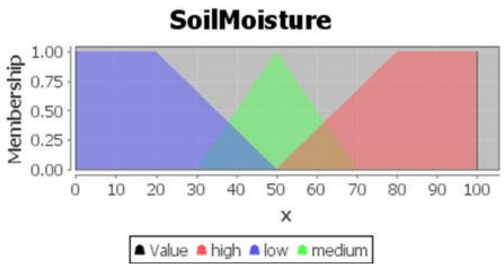

Fig. 6 Soil Moisture Range

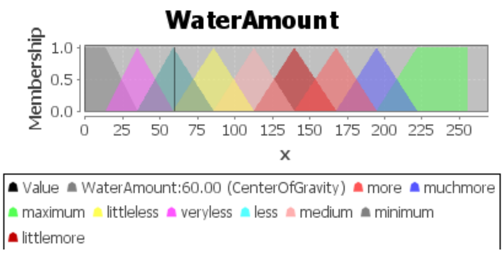

Fig. 7 Water Amount Range

B. Fuzzy Logic Rules

TABLE II shows the summary of the linguistic terms used in the Fuzzy Logic Rules. There are three parameters involved in the study: Temperature, Soil Moisture and Water Amount. Using the "AND" operator, the water amount is determined using the values supplied to temperature and soil moisture.

TABLE II. Fuzzy Logic Rules

Temperature	*Soil Moisture*	*Water Amount*
high	high	less
medium	high	very less
low	high	minimum
high	medium	little more
high	low	maximum
medium	low	much more
medium	medium	medium
low	medium	little less
low	low	more

C. Sample Source Codes

a) Fuzzification and Membership Functions: Fig. 8 shows the creation of fuzzy object and setting the membership value ranges for temperature, soil moisture, and water amount membership functions.

```
//Create Fuzzy Object
Fuzzy *fuzzy = new Fuzzy();

//Setting the membership value ranges for temperature
FuzzySet *lowTemperature = new FuzzySet(0, 0, 10, 25);
FuzzySet *mediumTemperature = new FuzzySet(15, 25, 25, 35);
FuzzySet *highTemperature = new FuzzySet(25, 40, 50, 50);

//Setting the membership value ranges for soil moisture
FuzzySet *lowSoilMoisture = new FuzzySet(0, 0, 20, 50);
FuzzySet *mediumSoilMoisture = new FuzzySet(30, 50, 50, 70);
FuzzySet *highSoilMoisture = new FuzzySet(50, 80, 100, 100);

//Setting the membership value ranges for water amount
FuzzySet *minimum = new FuzzySet(0, 0, 14, 35);
FuzzySet *veryless = new FuzzySet(14, 35, 35, 59);
FuzzySet *less = new FuzzySet(35, 59, 59, 86);
FuzzySet *littleless = new FuzzySet(59, 86, 86, 113);
FuzzySet *medium = new FuzzySet(86, 113, 113, 140);
FuzzySet *littlemore = new FuzzySet(113, 140, 140, 168);
FuzzySet *more = new FuzzySet(140, 168, 168, 195);
FuzzySet *muchmore = new FuzzySet(168, 195, 195, 222);
FuzzySet *maximum = new FuzzySet(195, 222, 250, 250);
```

Fig. 8 Source Code – Fuzzification and Membership Functions

b) Building the Fuzzy Rules: Fig. 9 and Fig. 10 show the BuildingFuzzyRule() and CreateFuzzyIO() functions to join the two conditions with AND operator and add the output to the fuzzy object based on the conditions.

```
//Building Fuzzy Rule function
void BuildingFuzzyRule(FuzzySet *condition1, FuzzySet *condition2, FuzzySet *result, int ruleNumber)
{
  FuzzyRuleAntecedent *fuzzyRuleAntecedent = new FuzzyRuleAntecedent();
  fuzzyRuleAntecedent->joinWithAND(condition1, condition2);

  FuzzyRuleConsequent *fuzzyRuleConsequent = new FuzzyRuleConsequent();
  fuzzyRuleConsequent->addOutput(result);

  FuzzyRule *fuzzyRule = new FuzzyRule(ruleNumber, fuzzyRuleAntecedent, fuzzyRuleConsequent);
  fuzzy->addFuzzyRule(fuzzyRule);
}
```

Fig. 9 Building Fuzzy Rules

M. S. Al Falahi, F. Q. Canlas and S. Nair, "Water Control System for Al Nakheel Tree Using Fuzzy Logic with Trend Analysis," 2020 International Conference on Computing, Electronics & Communications Engineering (iCCECE), Southend, United Kingdom, 2020, pp. 143-149, doi: 10.1109/iCCECE49321.2020.9231256.

```
//Create Fuzzy input and output
void CreateFuzzyIO()
{
  //Create temperature membership function as Input
  FuzzyInput *temperature = new FuzzyInput(1);
  temperature->addFuzzySet(lowTemperature);
  temperature->addFuzzySet(mediumTemperature);
  temperature->addFuzzySet(highTemperature);
  fuzzy->addFuzzyInput(temperature);

  //Create soil moisture membership function as Input
  FuzzyInput *soilMoisture = new FuzzyInput(2);
  soilMoisture->addFuzzySet(lowSoilMoisture);
  soilMoisture->addFuzzySet(mediumSoilMoisture);
  soilMoisture->addFuzzySet(highSoilMoisture);
  fuzzy->addFuzzyInput(soilMoisture);

  //Create water amount membership function as Output
  FuzzyOutput *waterAmount = new FuzzyOutput(1);
  waterAmount->addFuzzySet(minimum);
  waterAmount->addFuzzySet(veryless);
  waterAmount->addFuzzySet(less);
  waterAmount->addFuzzySet(littleless);
  waterAmount->addFuzzySet(medium);
  waterAmount->addFuzzySet(littlemore);
  waterAmount->addFuzzySet(more);
  waterAmount->addFuzzySet(muchmore);
  waterAmount->addFuzzySet(maximum);
  fuzzy->addFuzzyOutput(waterAmount);

  //Building Fuzzy Rules
  BuildingFuzzyRule(highTemperature, highSoilMoisture, less, 1);
  BuildingFuzzyRule(mediumTemperature, highSoilMoisture, veryless, 2);
  BuildingFuzzyRule(lowTemperature, highSoilMoisture, minimum, 3);
  BuildingFuzzyRule(highTemperature, mediumSoilMoisture, littlemore, 4);
  BuildingFuzzyRule(highTemperature, lowSoilMoisture, maximum, 5);
  BuildingFuzzyRule(mediumTemperature, lowSoilMoisture, muchmore, 6);
  BuildingFuzzyRule(mediumTemperature, mediumSoilMoisture, medium, 7);
  BuildingFuzzyRule(lowTemperature, mediumSoilMoisture, littleless, 8);
  BuildingFuzzyRule(lowTemperature, lowSoilMoisture, more, 9);
}
```

Fig. 10 Building Fuzzy Input and Output

c) Defuzzification: Fig. 11 shows the functions for defuzzification.

```
//Compute water amount "Fuzzyfication & Defuzzyfication"
float ComputeWaterAmount(float temp, float soiMoi)
{
  //Fuzzyfication
  fuzzy->setInput(1, temp);
  fuzzy->setInput(2, soiMoi);
  fuzzy->fuzzify();

  //Defuzzyfication
  return fuzzy->defuzzify(1);
}
```

Fig. 11 Defuzzification

d) Reading From Sensors: The succeeding figures show the source code for reading values from temperature (Fig. 12) and soil moisture (Fig. 13) sensors.

```
//Get temperature value
float getTemperature()
{
  Serial.println("\n\n*****************************************************************");
  Serial.print("Temperature: ");
  str = "0";
  startMillisTimer = millis();
  while(true)
  {
    Firebase.getString(firebaseData, "System Status", systemStatus);
    Firebase.getBool(firebaseData, "Connection Request", connectionRequest);
    Firebase.getBool(firebaseData, "Disconnection Request", disconnectionRequest);
    if(systemStatus=="Off" || connectionRequest || disconnectionRequest)
      break;
    if(Serial.available())
    {
      str = Serial.readString();
      Firebase.setString(firebaseData, "Temperature Sensor Status", "Running");
      tempSensorRunning = true;
      break;
    }
    if(millis() - startMillisTimer >= oneMinute)
    {
      Firebase.setString(firebaseData, "Temperature Sensor Status", "Not Running");
      sendMessage("Temperature sensor is not available/working");
      tempSensorRunning = false;
      break;
    }
  }
  value = str.toFloat();
  return value;
}
```

Fig. 12 Reading Data from Temperature Sensor

```
//Get soil moisture value
float getSoilMoisture()
{
  Serial.print("Soil Moisture: ");
  str = "0";
  startMillisTimer = millis();
  while(true)
  {
    Firebase.getString(firebaseData, "System Status", systemStatus);
    Firebase.getBool(firebaseData, "Connection Request", connectionRequest);
    Firebase.getBool(firebaseData, "Disconnection Request", disconnectionRequest);
    if(systemStatus=="Off" || connectionRequest || disconnectionRequest)
      break;
    if(Serial.available())
    {
      str = Serial.readString();
      Firebase.setString(firebaseData, "Soil Moisture Sensor Status", "Running");
      soilSensorRunning = true;
      break;
    }
    if(millis() - startMillisTimer >= oneMinute || str.toFloat() <= -1000)
    {
      Firebase.setString(firebaseData, "Soil Moisture Sensor Status", "Not Running");
      sendMessage("Soil Moisture sensor is not available/working");
      soilSensorRunning = false;
      break;
    }
  }
  value = str.toFloat();
  return value;
}
```

Fig. 13 Reading Data from Soil Moisture Sensor

D. User Interface

The succeeding discussions tackle the serial monitor and the mobile application.

a) Setting the WIFI connection and Verification Key for the mobile application: Fig. 14 and Fig. 15 show how to procedure on how to set the WIFI login credentials and verification key to enable mobile application to connect to the Arduino microcontroller.

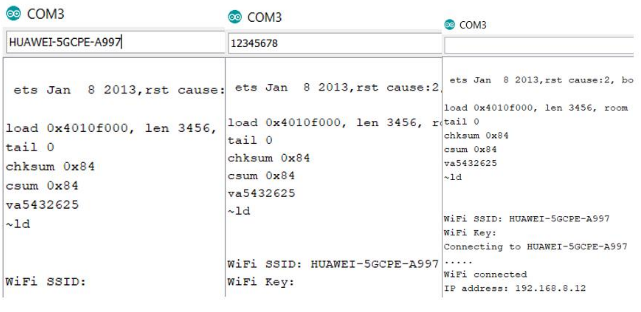

Fig. 14 Setting up the WIFI Connection

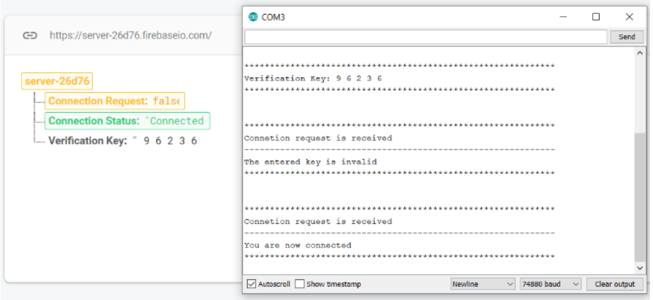

Fig. 15 Setting the Verification Key

b) Mobile Application: Users can monitor the status of the sensors, data read by Arduino and the trends in watering process. Fig. 16 and Fig. 17 show the mobile application interfaces and the serial monitor output.

M. S. Al Falahi, F. Q. Canlas and S. Nair, "Water Control System for Al Nakheel Tree Using Fuzzy Logic with Trend Analysis," 2020 International Conference on Computing, Electronics & Communications Engineering (iCCECE), Southend, United Kingdom, 2020, pp. 143-149, doi: 10.1109/iCCECE49321.2020.9231256.

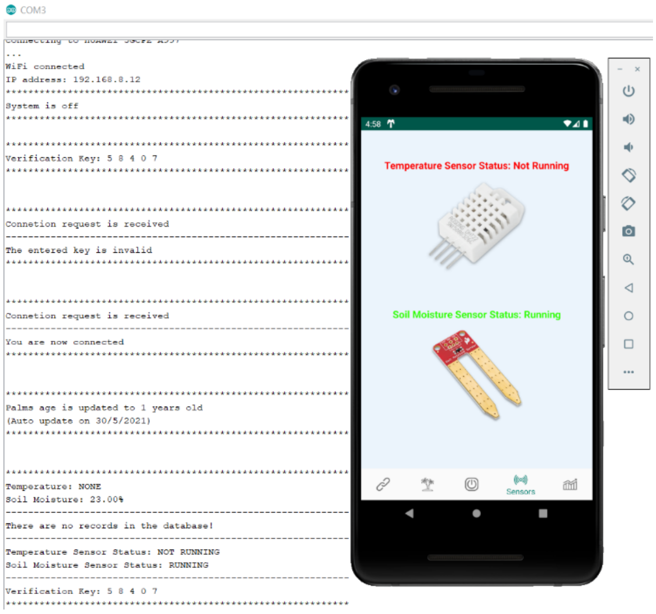

Fig. 16. Sensor Status

For unexpected situations wherein one of the sensors is not functioning, the system will switch from Fuzzy Logic operation to averaging operation. The average of the water amount for the past five days will be taken and fed to the Arduino that controls the water pump.

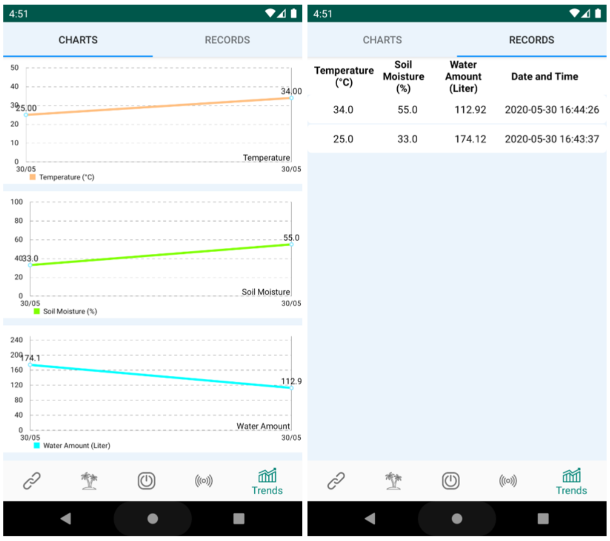

Fig. 17 Temperature, Soil Moisture and Water Amount Trends

E. User Acceptance Testing

The same expert from the Ministry of Agriculture and Fisheries conducted the testing vis-à-vis the requirements laid down during the initial interview. The test results showed favorable responses to all aspects of the proposed model.

V. Conclusion

Using Fuzzy Logic, the proposed model can irrigate the palm tree automatically with a significant amount of water for every watering operation without the need for any human intervention. Also, the system can record and notify users of the trends in the watering operations, and the status of the system. Furthermore, these records can be used to replace the fuzzy logic operation with the averaging method in the case that the temperature or soil moisture sensor is not working.

Cross-platform application for mobile app and web version of the user interface should be developed to provide access to all users. Although the discrepancy in the amount of water produced by fuzzy logic and the actual amount of water produced by the pump is less significant, the code for the motor pump should be refined and calibrated to achieve the precise amount.

References

[1] J. M. Reddy and N. D. Kumar, "Evolving Strategies for Crop Planning and Operation of Irrigation Reservoir System Using Multi-Objective Differential Evolution," Department of Civil Engineering-Indian Institute of Science, Bangalore, 2008.

[2] L. I. El-Juhany, "Degradation of Date Palm Trees and Date Production in Arab Countries: Causes and Potential Rehabilitation," *Australian Journal of Basic and Applied Sciences*, vol. 4, no. 8, pp. 3998-4010, 2010.

[3] I. Al Hyari, "Methods of Watering Palm Trees," Mawdoo, 14 May 2017. [Online]. Available: https://mawdoo3.com/%D8%B7%D8%B1%D9%82_%D8%B1%D9%8A_%D8%A7%D9%84%D9%86%D8%AE%D9%8A%D9%84 . [Accessed 10 January 2020].

[4] M. Qomaruddin, M. S. Mubarak and S. Mulyono, "Plant Watering System on The Basis of Internet of Things (IoT) With Protocol of Message Queue Telemetry Transport (MQTT)," in *The 18th International Conference on Positron Annihilation*, Orlando, 2019.

[5] A. Reghukumara and V. Vijayakumar, "Smart Plant Watering System with Cloud Analysis and Plant Health Prediction," in *INTERNATIONAL CONFERENCE ON RECENT TRENDS IN ADVANCED COMPUTING 2019, ICRTAC 2019*, Chennai, 2019.

[6] L. C. Ying, N. Arbaiy, M. Z. M. Salikon and H. A. Rahman, "Plant Watering Management System Using Fuzzy Logic Approach in Oil Palm Nursery," *Journal of Telecommunication, Electronic and Computer Engineering*, vol. 9, no. 3, pp. 129-134, 2017.

[7] S. Rahim, M. H. Hussain, S. R. Rahim, N. Hashim and H. A. Halim, "An automatic irrigation system for plants using fuzzy logic controller considering volumetric water content," in *First International Conference on Emerging Electrical Energy, Electronics and Computing Technologies 2019*, Melaka, 2020.

[8] A. N. Isizoh, S. O. Okide, A. E. Anazia and C. D. Ogu, "Temperature Control System Using Fuzzy Logic Technique," *International Journal of Advanced Research in Artificial Intelligence*, vol. 1, no. 3, pp. 27-31, 2012.

[9] P. Singhala, D. N. Shah and B. Patel, "Temperature Control using Fuzzy Logic," *International Journal of Instrumentation and Control Systems*, vol. 4, no. 1, pp. 1-10, 2014.

[10] C. Esplin, "What is Firebase?," HowtoFireBase.com, 24 October 2016. [Online]. Available: https://howtofirebase.com/what-is-firebase-fcb8614ba442. [Accessed 2 June 2020].

[11] D. Stevenson, "What is Firebase? The complete story, abridged.," Medium.com, 25 September 2018. [Online]. Available: https://medium.com/firebase-developers/what-is-firebase-the-complete-story-abridged-bcc730c5f2c0. [Accessed 4 June 2020].

[12] T. Branson, "8 Major Advantages of Using MySQL," Datamation, 16 November 2016. [Online]. Available: https://www.datamation.com/storage/8-major-advantages-of-using-mysql.html. [Accessed 1 June 2020].

[13] J. Hoo, "Advantages and Disadvantages of PHP," Big Cheap Hosting, 9 February 2018. [Online]. Available: https://bigcheaphosting.com/advantages-and-disadvantages-of-php/. [Accessed 1 June 2020].

[14] R. Sharma, "JSON – its advantages and disadvantages," ezeelive.com, 21 November 2020. [Online]. Available: https://ezeelive.com/json-advantages-disadvantages/. [Accessed 1 June 2020].

[15] M. Rouse, "Android Studio," TechTarget, October 2018. [Online]. Available: https://searchmobilecomputing.techtarget.com/definition/Android-Studio. [Accessed 4 June 2020].

[16] Times News Services, "The coldest place in Oman has a temperature of less than five degrees," Times of Oman, 3 December 2019. [Online]. Available: https://timesofoman.com/article/2326217/oman/environment/the-coldest-place-in-oman-has-a-temperature-of-less-than-five-degrees. [Accessed 28 May 2020].

[17] Times News Service, "Muscat records highest temperature in Oman," Times of Oman, 14 May 2018. [Online]. Available: https://timesofoman.com/article/134069/oman/environment/muscat-records-highest-temperature-in-oman. [Accessed 27 May 2020].

[18] C. Esplin, "What is Firebase?," 24 October 2016. [Online]. Available: https://howtofirebase.com/what-is-firebase-fcb8614ba442.

[19] D. Stevenson, "What is Firebase? The complete story, abridged.," 25 September 2018. [Online]. Available: https://medium.com/firebase-developers/what-is-firebase-the-complete-story-abridged-bcc730c5f2c0.

M. S. Al Falahi, F. Q. Canlas and S. Nair, "Water Control System for Al Nakheel Tree Using Fuzzy Logic with Trend Analysis," 2020 International Conference on Computing, Electronics & Communications Engineering (iCCECE), Southend, United Kingdom, 2020, pp. 143-149, doi: 10.1109/iCCECE49321.2020.9231256.