E-TONGUE: A SMART TOOL TO PREDICT THE SAFE CONSUMPTION OF GROUNDWATER

Project Id: 2020-164

Final Report Draft (Individual)

A.M.P.B. Alahakoon

IT16175280

B.Sc. (Hons) Degree in Information Technology Specializing in Software Engineering

Department of Information Systems Engineering

Sri Lanka Institute of Information Technology

Sri Lanka

September 2020

E-TONGUE: A SMART TOOL TO PREDICT THE SAFE CONSUMPTION OF GROUNDWATER

Project Id: 2020-164

Final Report Draft (Individual)

B.Sc. (Hons) Degree in Information Technology Specializing in Software Engineering

Department of Information Systems Engineering

Sri Lanka Institute of Information Technology

Sri Lanka

February 2020

DECLARATION

We declare that this is our own work and this report does not incorporate without acknowledgment any material previously submitted for a degree or diploma in any other university or institute of higher learning and to the best of our knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgment is made in the text.

Name	Student ID	Signature
A.M.P.B. Alahakoon	IT16175280	

The supervisor/s should certify the final draft report with the following declaration.

The above candidates are carrying out research for the undergraduate Dissertation under my supervision.

Signature of the supervisor: Date

....

ABSTRACT

Resourceful explanations for the water aspect are picking up significance with headway in information technology. Because of the quick improvement and urbanization, the nature of water is getting debase overstep by step, and it prompts water-borne illnesses, and it makes a terrible effect on living beings. National Water Supply and Drainage Board (NWSDB) is the national foundation for the arrangement of consumable water and the NWSDB has so far had the option to give 43.4% [1] of the populace with pipe-borne water. There are different types of water supply through country water plans run by local governments just as buyer social societies and hand siphons and so forth that represent an extra 25% of the populace. In that capacity, the absolute safe drinking water supply inclusion in the nation is nearer to 65% of the populace. The rest of the society consumes unsafe water which tends to vast outbreaks in the future. Out of safe water consumes most of them unaware of the safeness of the consumable water. There is no portable and accurate mechanism to check the safeness of water when and there except TDS meter which has a limited range of capability of measuring water quality.

This research explores about groundwater safety checking mechanism based on the Internet of Things technology using pH, temperature, conductivity, and turbidity sensors. The model created is utilized for testing water tests and the information transferred over the Internet to the cloud server and predicts the safeness of water samples for the end-users with higher accuracy. Also, an algorithm that can join these strategies to give an exact outcome alongside an alignment system that can lessen the lapse rate. The product has been planned with the goal that it tends to be utilized for applications and by a wide range of end-users.

Keywords - Total dissolved solids (TDS), Water quality, internet of things,

ACKNOWLEDGEMENT

I would want to accept this open door to stretch out my appreciation to the individuals who have helped me through this research. Right off the beginning, I would praise my supervisors, Ms. Chathurangika Kahandawaarachchi and Ms. Narmada Gamage for the patient direction, support, and guidance they gave me over the span of this exploration.

Also, my much gratitude goes to National Water Supply and Drainage Board for the direction, advices and providing the previous information of water consumptions and parameters. The immense gratitude extended to Mr. W.G.T. Indragith, the Manager of the groundwater- Studies, Senior Hydrologist of National Water Supply and Drainage Board, Ratmalana helped me throughout the project. Much obliged to you my companions, for your significant recommendations and direction in the research outcomes.

At last, pledge my gratitude to everybody in my family who supported me during this Academic period.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Abbreviation	**Description**
IEEE	Institute of Electrical and Electronic Engineers
IDE	Integrated Development Environment
IoT	Internet of Things
GPRS	General Packet Radio Service
GSM	Global System for Mobile Communications
LDR	Light Dependent Resistor
NWSDB	National Water Supply and Drainage Board
TDS	Total Dissolved Solids
WQI	Water Quality Index
WQP	Water Quality Parameters
WSN	Wireless Sensor Network
SMS	Short Message Service
CKDu	Chronic Kidney disease

1. INTRODUCTION

Water is one of the basic regular assets that have been gifted to humankind. Be that as it may, the fast advancement of the public and various human exercises speeded up the defilement and disintegrated the water assets. The water quality index (WQI) is constantly utilized to legitimize or order the degree of water quality. WQI goes about as a meaningful sign of water quality estimation through the assurance of parameters of groundwater [2]. It is imperative to gauge the degree of water quality before devouring or safe use for different purposes. For water quality characterization, a few physical, natural, and synthetic parameters that have a critical effect on the water quality must be recognized.

The conventional approach for observing the water quality is with the end goal that the water test is taken and sent to the lab to be tried physically by expository strategies. Even though by this strategy the substance, physical, and organic particles of the water can be dissected, it comes with disadvantages. Right off, it is tedious and works escalated. Besides, the expense of this strategy is high because of the activity cost, work cost, and gear cost, and it is hard to detect analytical choices in real-time [2].

To guarantee the safety of the nature of water, it ought to be checked continuously. To make the way toward testing the constant nature of water basic and simple for everybody, ease and convenient water quality observing the device is structured and created. This a portable device implanted with several sensors to capture diversions in water formation. Mainly pH sensor, conductivity, temperature, calcium, and turbidity sensors will be embedded in this proposed system. It is a user-friendly framework which much of the time can test the nature of water and sends an alert to the authorities if there should arise an occurrence of any variation in any parameter of the water which is not consumable by checking formation of water sample through the neural network after storing the initial data in the cloud.

Due to the absence of water safeness checking equipment in Sri Lanka which performs in real-time this stores the critical data in the cloud. Our research team aims to develop a piece of equipment to cater to this requirement which is beneficial mainly to the families in the north-central province.

1.1 Background & Literature survey

This is an enormous water amount and if 1% of water misfortune is spared it will be adequate to give water to 20,000 families [1]. The related cost that the nation could be spared will be roughly Rs. 3,000 million every year. This sparing is adequate to build 1 km of interstates for every year. Along these lines, if more levels of water misfortunes can be wiped out the advantages are so high. Supply of consumable water is a costly issue and accordingly, the capital speculation required for the arrangement of consumable water surpasses Rs. 175,000 to 300,000 for each family. Consequently, the measure of cash required for water supply is enormous and this has gained the ground of providing water slower than usual. Simultaneously if the water created is utilized for expected purposes, without wastage and ill-advised utilization, the inclusion of the population can be expanded without going for additional speculations [1].

Our proposed approach, a possess collected Arduino microcontroller is utilized as the central controller of the system. When the code is transferred to the microcontroller. Right now, sensors are utilized to gauge the fundamental water parameters. As it was considered from the past investigates, the most fundamental water parameters should have been checked by the normal clients are water pH level, water turbidity (darkness), water temperature, conductivity, and the calcium level. Along these lines, Sensors' circuits are associated with the microcontroller and the tests of the turbidity, pH, conductivity, and temperature sensors put inside the water. A waterproof temperature sensor is utilized to dodge any harm or electrical stun to the device and the end-user. All sensors read the water quality parameters and send the information to the microcontroller as electrical signs and it will be transmitted to the cloud server through the WI-FI module.

So, there are several kinds of research regarding water quality management are conducted. We will elaborate on some of them. Quoi Tie-Zahn, 2010 [4] assembled up an online water quality checking entity dependent on GPRS/GSM. The transmission of data packets done through a GPRS enabled network, which assisted with checking remotely the WQP.

Actualized the WSN platform in the water observing mechanism. The SquidBee [5] which is a commonplace IEEE 802.15.4 ZigBee based bits were utilized as the WSN hubs. All sensor hubs were drifted on the seashore. It observed the water pH level and water temperature of the contemplated seashores. The device comprises three key parts: the sensor hub, cluster head hub [5], and network portal. The algorithm was written in C programming language to get the information from sensor hubs for each 5s of time interim. The approved end-user can screen the constant information of pH level, and water temperature.

This developed was a WQM [6] system based on WSN. The remote sensor was based on the Sigsbee network. WSN tested WQP and sent data to the Internet using GPRS. With the help of the Web, information was gathered at a remote server.

M.K.Khurana and his colleagues proposed a water quality observing device that can examine the nature of water and impart a caution signal to the authorities through Wi-Fi. If the water parameter is certainly not an ideal worth [7]. This platform gives an exact estimation of the water parameters since right now pH sensor is twofold aligned. Be that as it may, this is only capable of showing to the pH level of the water and no other water quality parameters.

N. Vijayakumar and R.Ramya concocted a thought for the constant water quality observing in the IoT (Internet of Things) condition. Their framework comprises a few sensors which can quantify some basic parameters of the water, for example, temperature, pH level, conductivity, turbidity, and the information can be seen on the web utilizing distributed computing. Center Micro controller implement here is raspberry Pi, its detriment is that it is run on LINUX utilizing the console. It requires the clients to include a command each time when they need to realize the sensors perusing [3].

A.S. Rao, built up a water quality observing framework utilizing Arduino Mega 2560 microcontroller and separate sensors to screen the temperature, conductivity, pH, broke up oxygen, light, and oxidation decrease capability of the water. Even though it comprises of complex wiring and requires a PC or an additional Beagle board XM ARM processor for correspondence interface and activity [4].

M Deqing, Z. Ying and C.Shangsong, in [8], utilized the Global System for Mobile Communications to detect the nature of water remotely. In their proposed framework, the basic water quality parameters, in particular, pH level, conductivity, dissolved oxygen, and turbidity are perused from the water through the individual sensors and it is then dissected by the controller and in the event that it is past the standard range, it is sent to relevant parties through an SMS, simultaneously. The information is also put away in a database and it is plotted to a graph for additional analysis. Be that as it may, this product is moderate for large water provider organizations or ventures since it comprises of costly parts.

Fu Qi-Feng et al's. framework contains an alert component that informs the production line representatives when the deliberate boundaries for example ammonia nitrogen and several other parameters are out of range. if the boundaries are not in the ostensible territory, at that point it very well may be sent to a treatment tank which it tends to be dealt with once more. Baihaqi Siregar et al. utilized a straighter forward methodology utilizing Waspmote as a micro controller as a remote association with the sensors. 3G component was utilized for sending information to a cloud which is then shown in diagrams and if any of the qualities are far-off, a SMS warning is transmitted to alarm the client. Here used methodology is major pertinent contrasted and the past undertaking [20].

A portion of the activities, likewise, incorporate dissolved oxygen as a component of the boundaries being estimated like Yang Su et al (2017) [15]. Additionally, a significant pointer for water quality. however, DO sensor tests are very exorbitant essentially because of the probe development. The goal of this undertaking is to build a moderate water observing framework for different mechanical processing plants managing wastewater, thusly DO sensor is not suited for this venture.

Yang Su et al. marine-based water standard observing framework centers around the strategies to get precise water boundaries utilizing water profundity. The correspondence utilized between the sensors and the microcontroller is very intricate yet proficient contrasted with the recently referenced works: boundaries are sent utilizing multi hop routing through remote sensors. This specialized technique can enforce to this sector where the plant has various water therapy frameworks. A case of multi bounce directing is appeared in Figure 1.

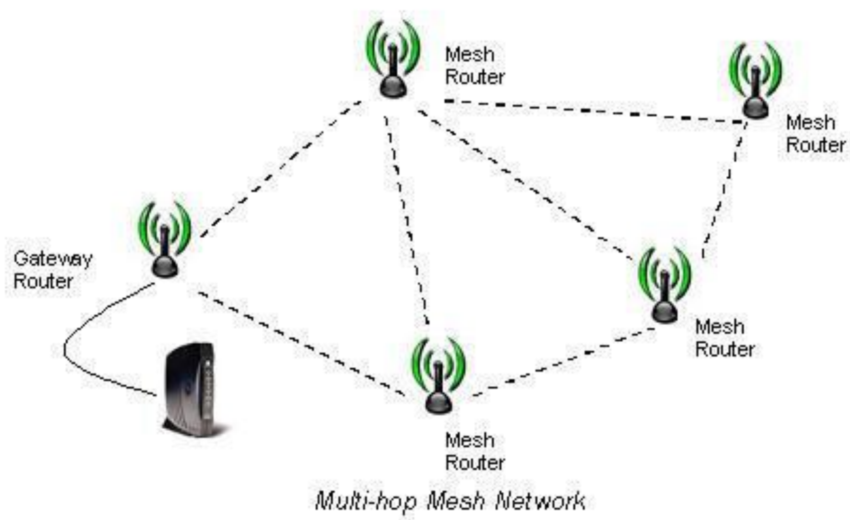

Figure 1: Mesh Network

1.2 Components used in the device

A. Temperature sensor

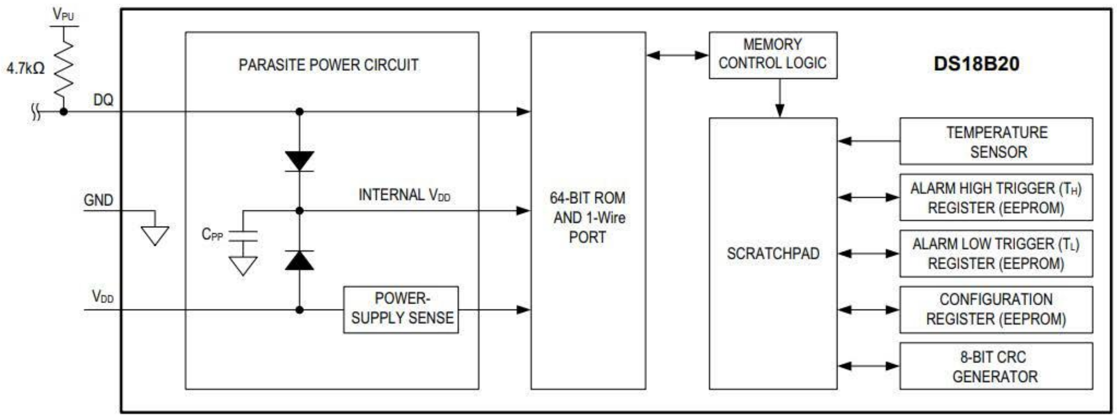

Figure 2: DS18B20 Sensor Circuit

DS18B20 is utilized in the project, which comprises of a temperature and an inbuilt IC containing an alert trigger that sends an admonition if the temperature is out of range. Through one data pin and ground pin it communicates with microcontroller. A draw up resistor is utilized which is associated from the data pin to the force flexibly all together for the microcontroller to pull it low while mentioning the information.

Temperature sensor can operate without an energy source even, as the inward capacitor stores vitality from the high signals, anyway as current is required current during transformation season of the temperature information to computerized bits [16]. The utilization of temperature sensor for this framework is used because turbidity and conductivity depend on temperature.

B. Turbidity Sensor

Measure of clearness and darkness of a fluid is known as turbidity, because of the existing immersed fragments. In figure 3 it shows water samples of different levels of turbidity.

Figure 3: Turbidity of different water levels

For groundwater turbidity is a significant factor. It is a pointer of the measure of immersed silt in water. The dangling dregs which because turbidity can convey microorganisms, for example, lead and microbes, these are toxic to mankind [9].

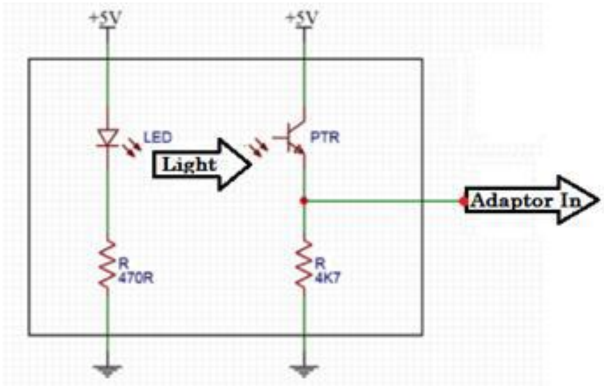

Figure 4: Turbidity sensor circuit

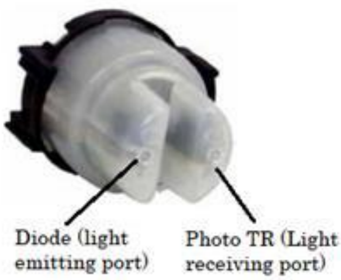

Figure 5: Turbidity Sensor

Turbidity sensor or turbidity tube used to capture turbidity in groundwater. This sensor work by estimating the measure of light conveyance and dissipating rate. The probe transmits a light pillar into the water, when light is obstructed and dissipated by particles which are in water before it is gotten by the photograph semiconductor (PTR). As indicated by Figure 4, the PTR is associated in corresponding with a resistor. For more clear water, the light is more grounded, thus, the yield voltage is higher. It can gauge turbidity from 0 to 2500 ntu. Yet, for our implementation, the turbidity is adjusted to be exact inside a scope of 0-290 ntu [17].

C. pH Sensor

This is characterized as portrayal of hydrogen particle movement in a fluid. As the majority of the handled groundwater is not sent to the water bodies for purification, it is significant that pH esteem is between 6 to 9, in such a case that the water is acidic, calcium creating life forms like mussels and shellfishes would experience issues in delivering calcium based shells.

In figure 6 it comprises of an external cylinder having a soaked KCl reference arrangement that allows regulate potential and the inward cylinder comprising of a pH 7 cradle mixture. The base of the internal cylinder is consisting of a delicate glass film which permits substitution of the hydrogen particles between the arrangement. Every one of cylinders has a silver wire covered that is then associated with the contribution of the sensor enhancer unit.

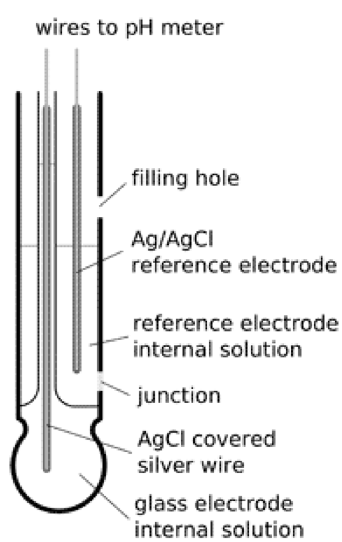

Figure 6: ph sensor probe

The hydrogen particles go within the glass film and restores the metal particles in the silver wire, along these lines a voltage is created. The voltage is transmitting as a contribution to the intensification course where it is handled by the microcontroller to get the pH esteem.

Because of essential arrangements, there can have denial voltage which the microcontroller will most likely be unable to measure, henceforth a reference is required. A straightforward intensifier would likewise be expected to enhance the little voltage from the pH test all together for the microcontroller to peruse the qualities precisely and to lessen blunders.

D. Electric Conductivity Sensor

Conductivity can be characterized as the capacity of fluid to direct electrical flow. In water and fluid structure materials. Electrical conductivity generally relies upon the accompanying boundaries:

- solution temperature
- versatility of particles
- valence of particles

The conductivity sensor can be utilized in different applications, for example, however not restricted to evaporator water treatment, and converse assimilation checking. The key factor to pick a legitimate test is to decide the necessary conductivity cell consistent to improve precision and guarantee life span of the hardware [18].

Conductivity of an answer is regularly estimated in milli Siemens or micro Siemens. So as to guarantee normalization of electrical conductivity estimations, units of explicit conductivity are utilized. Explicit conductivity is communicated as milli Siemens per centimeter (mS/cm) or micro Siemens per centimeter (μS/cm) [19].

The conductivity of arrangement is exceptionally subject to the grouping of particles and the temperature of the arrangement. The assurance of the electrical conductivity is a fast and helpful methods for assessing the centralization of particles in arrangement. Grouping of the particles basically speaks to the quantity of the particles. Conductivity has likewise a considerable reliance on temperature of the liquid too. The conductivity of normal electrolytes regularly increments with expanding temperature.

These is significant as it influences disintegrated oxygen dissolvability. Increasing conductivity bring down the broke up oxygen focus. The impact of conductivity on the solvency of broke up gases is because of Henry's Law; the steady utilized will changes dependent on salt particle fixations. Most water species can just endure explicit scope of conductivity.

By and large, conductivity offers a dependable technique for estimating the ionic substance of a solution. Unlike estimation with particle specific cathodes, for example, pH sensors, the reaction of a conductivity sensor will not change over time. There are drawbacks of conductivity estimations. They are not particle specific, estimating the boundary dependent on the consolidated impact of all the disintegrated particles, likewise for estimating various arrangements the conductivity test ought to cleaned with cleaning solutions appropriately before setting in another arrangement and if not focused the estimations may show off base qualities.

Conductivity of arrangement can be communicated as saltiness too. The test for saltiness and conductivity estimation is same, anyway for saltiness estimation a revision factor ought to be applied to conductivity perusing.

E. Arduino Nano Microcontroller

The research team decided to go with Arduino Nano microcontroller due its flexibility and breadboard friendliness. Microcontrollers has low force utilization. For this component, the microcontroller picked ought to have an adequately high check speed to decrease the operational time.

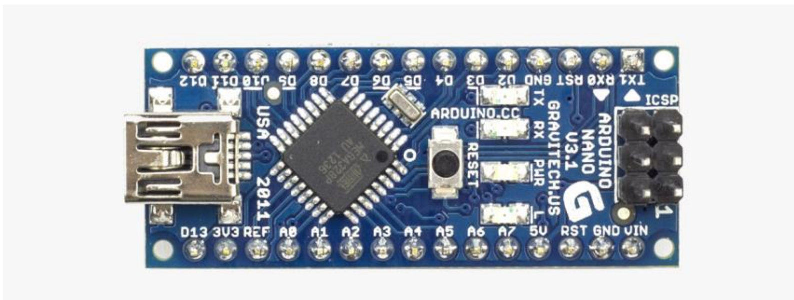

Figure 7: Arduino Nano Microcontroller broad

Arduino Nano is a prominently utilized microcontroller board which has 8-piece ATmega328P microcontroller. Arduino Nano inbuilt with 14 digital input pins, 8 analog pins, a mini USB association, and a reset button.

The USB association will be utilized for connecting purposes. The Arduino IDE is utilized which is written in Java, yet C language can be utilized to programming, making it simple to use.

Table 1: Arduino nano broad parameters

Paramater	Detail
Microcontroller	ATmega328P
Clock speed	16MHz
Flash memory	32KB *3
Static RAM (SRAM)	2KB
EEPROM	1KB
Supply voltage	5V
Recommended DC input voltage range	7V to 12V *2
Digital I/O pins	14
Digital I/O pins available for PWM output	6 *1
Analogue input pins	6
Max. DC current output per pin	20mA
Max. DC current available from 3.3V supply	50mA
Dimensions	68.6 × 53.4mm
Weight	25g

According to the details mention in above table, the most suited microcontroller for this research was Arduino Nano.

F. Nodemcu ESP8266

The Nodemcu board has a genuine Amica ESP8266 processor. Wireless transmission utilized for sending information from the microcontroller to the smart application. Key highlights of this broad are low direct power utilization, large sending range, high information rate, and most significantly similarity with the smart phone application. The following table gives us a clear understanding for the reasons behind choosing this type of transmission medium.

G. Communication Medium

Remote correspondence is utilized for communicating information from the microcontroller to the smart tool. The key factor for picking a decent communication mode are less power utilization, enormous communicating range, high information rate, and compatibility with the smart application. The following table 2 will display transmission mediums.

Table 2 : Comparison of transmission mediums

	Bluetooth (Eg: Bluetooth Serial Transceiver HC06)	Zigbee	ESP8266 Wi-Fi module	Radio transmission (Eg: nRF24L01)
Description	Wireless technology for exchanging data over short distances using short-wavelength radio waves in from 2.4 to 2.49 GHz	High-level communication protocols used to create WPANs built from low power digital radios	Provides a wireless connection to the internet, can be connected to the microcontroller used to receive and transmit data	A single chip radio transceiver for the worldwide 2.4 - 2.5 GHz ISM band.
Power consumption	Low power consumption (8mA while communicating at 3.3V)	Low power consumption (about 20mA while communicating)	High (has a rating of 3.3V, 0.5A)	Very low power consumption (rating of 3.6V, 9mA)
Transmission range	<10m	10-100m	20-100m	Around 250m
Data rate	9.6KB/s (default)	40-250 KB/s	1MB/s to 6.75GB/s	1MB/s to 2MB/s
extra features	-	Need to send through a mesh of connected Zigbee receivers to pass through, thus may be costly	Uses AT commands to control the WI-FI module which involves simple coding	Need some application to communicate with the smartphone app

1.3 Research Gap

In recent times safeness of water plays a vital role in Sri Lanka. The conventional strategy for water quality testing is not pertinent any longer. To handle the issue, a few electronic-based water qualities observing frameworks were created in the previous decade. However, as the greater parts of these frameworks were examined, other than their qualities, every one of them has its own drawbacks. Therefore, there is a clear gap remaining to be explored. So, we try to come up with an optimal solution to outfit this problem. Table 2.1 will give you and clear understanding of what to be considered to have a good quality product. From our research team's feasibility study, we discovered these areas to be carried forward for a better and accurate product.

Table 3: Comparisons of existing systems

Attributes	Currently Available Equipment's	Proposed Solution
Basic Sensors	Yes	Yes
Calcium Sensor	No	Yes
Portability	Moderate	High
Sensor Calibration	No	Yes
Predict WQI	No	Yes
Forecast water parameters	No	Yes
CKDu predication	No	Yes
Water precautions Visualization	No	Yes
Cost	Moderate	Low
Accuracy	Moderate	High
Real-Time	No	Yes

In this study, we present a low cost, IOT based prototype water quality monitoring and analysis-based prediction system which sends the sensor data to the server along with the water parameters values and location information of the device. Also, this system consists of real time data uploading capability, so that we can minimize the product usage cost too. Operations power up in the IOT device through 12V power adapter or 9V battery. The energy consumption is minimized through this device.

On the server or cloud side, the water quality parameter data collected from each location are processed and published through the mobile application ‘E-Tongue’. Mobile application users are able to figure out suitability or consumption of water, monitor the current and future water quality levels at any specific area. It also provides a water pollution predicting system that would be very important in identifying some natural water sources which are unsuitable for consumption purposes.

1.4 Research Problem

The assurance of water is an objection due to the top wellsprings of contaminations. The fundamental driver for water quality issues is the overexploitation of characteristic assets. When the safeness of water decreases numerous harmful water-related diseases may arise. In Sri Lanka, the main resources of water are surface water and groundwater. While groundwater assets are limited to wells, tube wells, and Argo wells. Around 80 percent of this groundwater is being utilized for high-esteem farming and staying 20% for residential use including flushing requests of toilets in urban zones. There are more than 300 urban and provincial water supply plots across 23 areas under the NWSDB, 33% of them depend on provisions from shallow and deep groundwater sources.

There is a direct impact of quality water for the well-being of people in Sri Lanka. Currently, there is no mechanism provided to society to check the safeness of groundwater before consumption by the authorities. Due to this large number of humankinds have water-borne diseases. So, our research team had a feasibility study on this matter and has a solution we proposed to develop a portable IoT device to check safeness in groundwater in real-time. This will cater to the problem mainly the people facing in rural cities. This will be more beneficial in time to come to the Sri Lankan society.

1.5 Objectives

1.5.1 Main Objectives

The elementary purpose of this component is to facilitate IoT enabled product to read fundamental water parameters through calibrated sensors and feeding collected data into the cloud server and visualize it in a smart phone application. Later, this will help to predict the safeness of water by going through several algorithms and convey end user’s consumption ability of tested sample.

1.5.2 Specific Objectives

- Implement a cost-effective water safeness measuring equipment.
- Integrated smartphone application for water quality measuring with high performance.
- Improving the accuracy of sensor readings
- Portable solution for the end-user.
- Comprehensive literature review on combination of sensors to be implemented.
- Minimize the error rate of each sensor.

2. METHODOLOGY

In this section we will elaborate more how we handle our objectives and specific objectives. This involves in step by step process to achieve the final product which helps to predict water quality index, seasonal parameter variations and forecasting chronic kidney disease outbreak.

2.1 Overview of the system

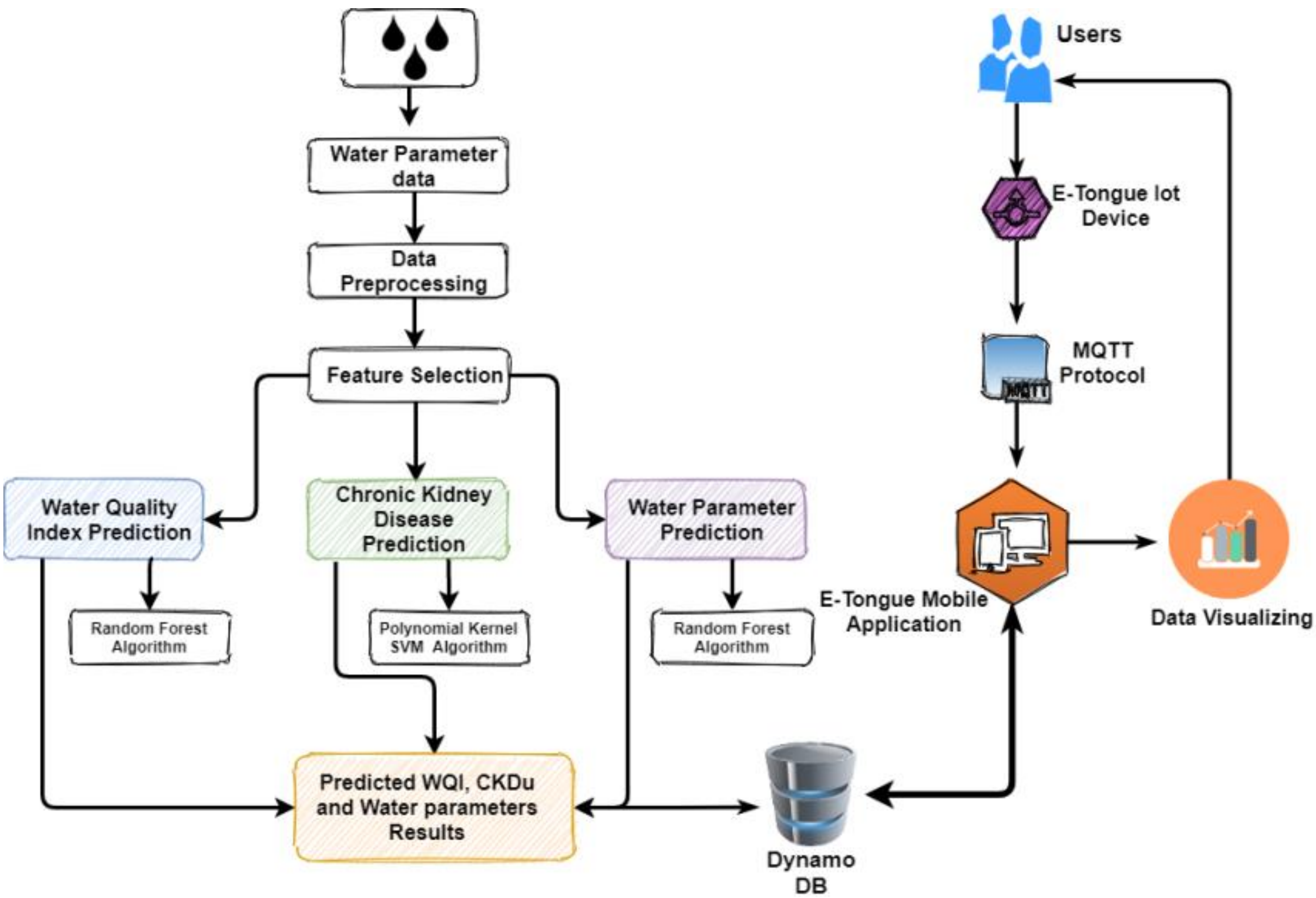

Figure 8: System Overview

E-Tongue is a smart app to predict the quality of ground water. Which beneficial the community to consume high quality water from natural groundwater resources. Above figure 8 is displays the system architecture diagram. This overall system has four main functionalities:

- Developing a smart gadget to get water samples for predictions purposes.

- Predicting the WQI value for ground water.
- Forecasting water quality parameters.
- Envision to CKDu outbreak and water borne diseases.

2.2 Overview of the component

Water safety estimating appliance was to implement as shown in figure 9, In this implemented methodology, a nodemcu microcontroller is utilized as the fundamental controller of the unit and Arduino Nano microcontroller for control purpose of sensors and for data storing dynamo database was utilized. As we are targeting this for normal end-users the device should be affordable. The framework capacities naturally and freely as per the code transferred to the microcontroller. Here, five sensors are utilized to measure the fundamental parameters. Water pH level, turbidity (darkness), temperature, turbidity, electric conductivity and total dissolved solids to measure groundwater.

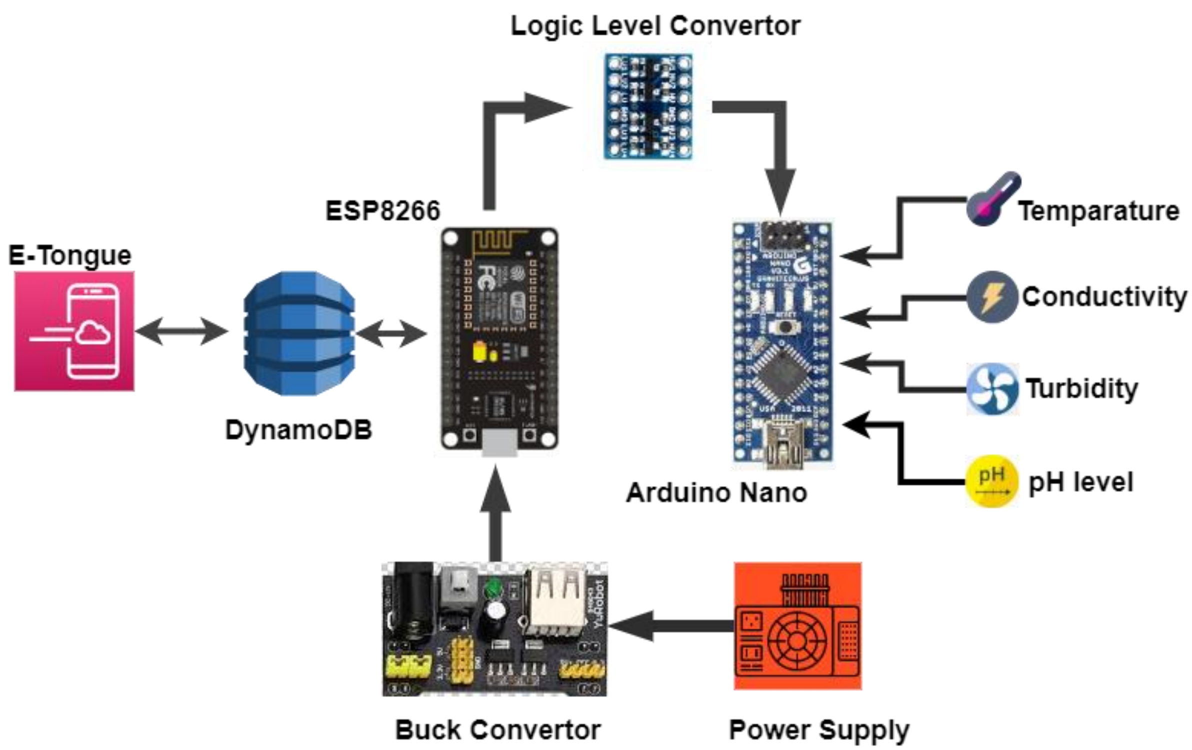

Figure 9: Component system overview

All the circuits and sensors read the parameters and transmit packets to the microcontroller as electrical signs. The Nodemcu ESP8266 was programmed to transmit the measured parameters into the cloud space and the smartphone application (E-Tongue) was developed to receive the parameters data from dynamo database for visualizing and analyzing .

The pin connections for the Arduino NANO shown in the below figure 10.

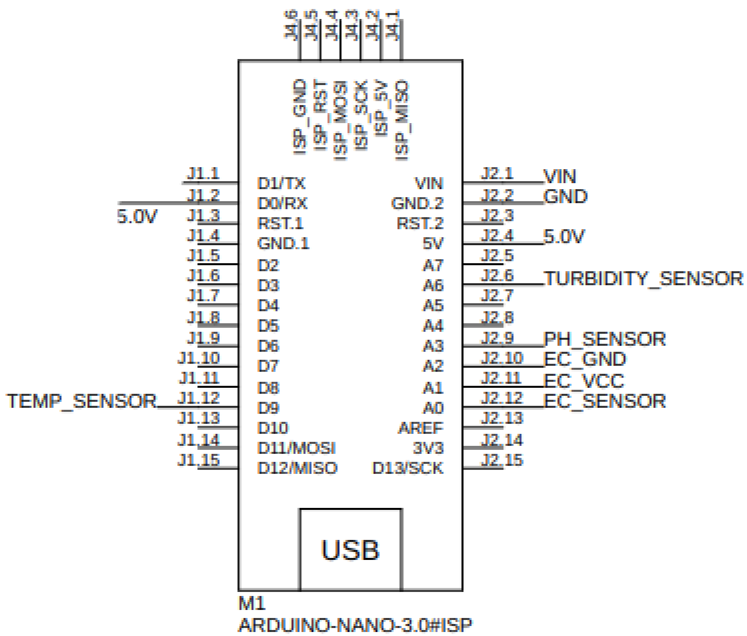

Figure 10: Arduino nano pin connection

2.2 Development of sensor circuities.

2.2.1 pH Sensor

A. pH sensor module diagram

After all this probe are not feasible to be straight forwardly associated with the microcontroller as a result of its low voltage, the pH module circuit utilized in this task. This module circuit is an advance variant of the earlier versions of pH module circuit where the operational components and the segments are changed concurring to the prerequisites [31].

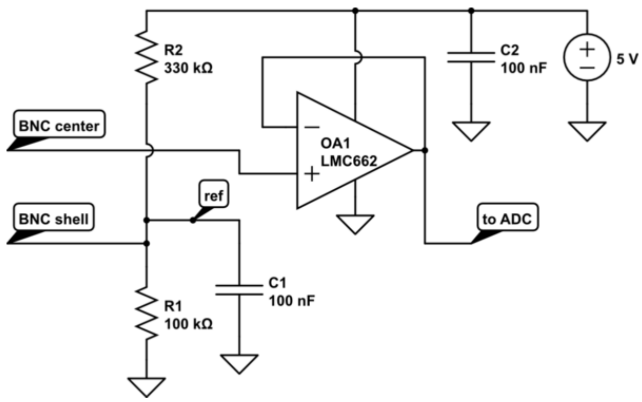

Figure 11: pH sensor module circuit

B. Calibration of pH sensor

For the calibration, voltage acquired is linear to the pH, pH 4 and pH 10 their separate voltages were gotten utilizing the known solutions. The table 4 and figure 12 shows the adjustments outputs.

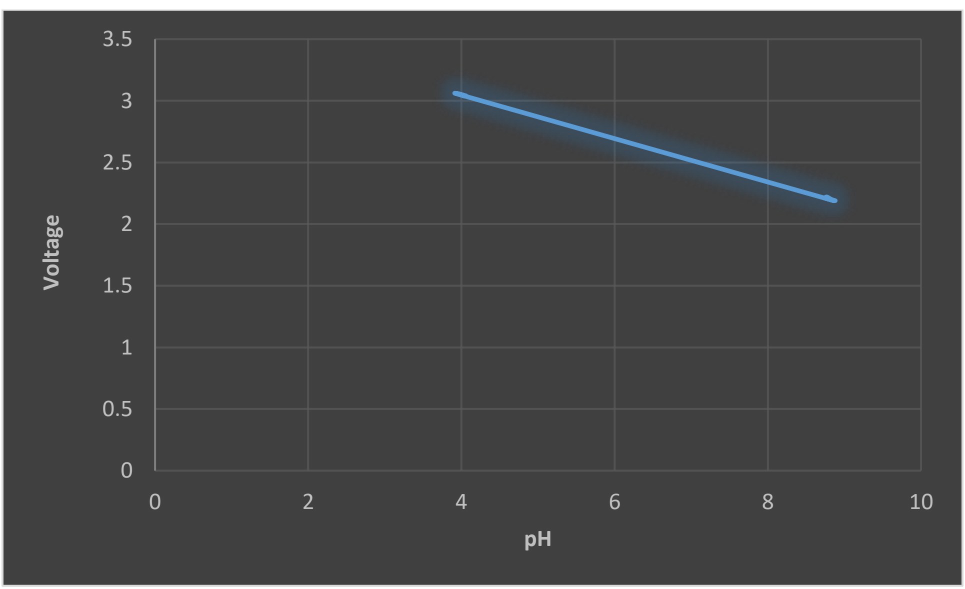

Figure 12: Graph of voltage vs ph value

Table 4: Values obtain for buffer solutions

ph	voltage
4.06	3.04
3.94	3.06
4.03	3.04
4	3.05
3.92	3.06
8.86	2.19
8.75	2.21
8.77	2.22
8.83	2.2
8.88	2.19

C. Coding part of pH sensor

```
void readPh() {
  for (int i = 0; i < 10; i++)
  {
    buf[i] = analogRead(PHPIN);
    delay(30);
  }
  for (int i = 0; i < 9; i++)
  {
    for (int j = i + 1; j < 10; j++)
    {
      if (buf[i] > buf[j])
      {
        pHtemp = buf[i];
        buf[i] = buf[j];
        buf[j] = pHtemp;
      }
    }
  }
  avgValue = 0;
  for (int i = 2; i < 8; i++)
    avgValue += buf[i];
  float pHVol = (float)avgValue * 5.0 / 1024 / 6;
  phValue = -5.70 * pHVol + calibration;
}
```

Figure 13: Coding of ph sensor

The pH variable is characterized as global parameter and so as to compute the pH value through sensor reading, 'readPh()' functionality is characterized which does the accompanying:

1. When 12v given to the unit pH circuit turns on.

2. It gather 10 consecutive values from analog pin A3 from nano microcontroller.

3. Next ordering values taken and discarding the highest and the lowest of the values and ***c***alculating the average with the six remaining samples.

4. The 'pHVol' is then calculated by converting the above value to voltage.

5. Using ph references ‘pHVol’ is converted to ‘phValue’.

2.2.2 Temperature Sensor

A. Temperature probe circuit

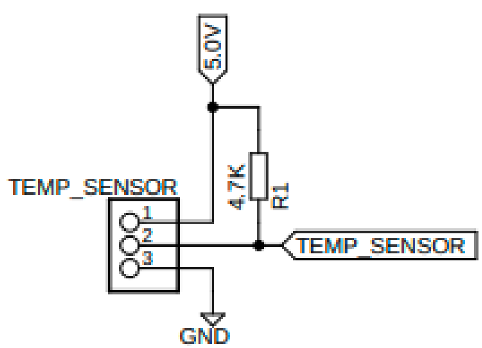

Figure 14: Circuit of temperature probe

The sensor functions with the method of 1-Wire communication. It need only the data pin connected to the microcontroller. It has a pull up resistor (4.7K) and the other two pins are used for power purposes ground and 5V as shown above. The 4.7K resistor is utilized to keep the line in high state when the controller bus is not being used. The temperature sensor reading estimated by the sensor will be put away in a 2-byte register inside the sensor. This information can be perused by the utilizing the 1-wire strategy by sending in an arrangement of information.

B. Calibration of temperature sensor.

For sensor calibration, you need to gauge something of which you know the temperature. The easiest method to do it at home is utilizing boiling water and a shower of ice, additionally called a "triple-point" shower. Remember that to make an exact estimation you should know atmospheric pressure and compute the best possible boiling temperature there.

Discerning the value that the sensor read and the value that ought to be, we can change the basic estimation of the DS18B20 into something more legit.

Fill up a container with water and heat up until it boils. Next, submerge the sensor probe and keep it for couple of seconds and take readings to the serial monitor. Mark this reading as ‘Basic_High’. Afterwards fill up another container with small ice cubes and ice water. Clean up the sensor probe and submerge it in ice container and set that reading as ‘Basic_Low’. References measured at this point. For boiling point, it was 99.9 Celsius and melting ice 0 Celsius. Set the basic range and reference range by taking the difference of the values. The following formula was built up with the calibration of the sensor.

$$\boldsymbol{Precise\,Value} = \left(\frac{\left(\left(\boldsymbol{Sensor_{Reading}} - \boldsymbol{Basic_{Low}}\right) * \boldsymbol{ReferenceRange}\right)}{} \boldsymbol{BasicRange}\right) \quad (1)$$

C. Coding of the sensor

```
#include <OneWire.h>
#include <DallasTemperature.h>
// Data wire is plugged into pin D9 on the Arduino
#define ONE_WIRE_BUS 9
// Setup a oneWire instance to communicate with any OneWire
devices
OneWire oneWire(ONE_WIRE_BUS);
// Pass our oneWire reference to Dallas Temperature.
DallasTemperature sensors(&oneWire);
/
*****************************************************************
****/
void setup(void)
{
 // start serial port
 Serial.begin(9600);
 Serial.println("Dallas Temperature IC Control Library Demo");
 // Start up the library
 sensors.begin();
}
void loop(void)
{
 Serial.print(" Requesting temperatures...");
 sensors.requestTemperatures(); // Send the command to get
temperature readings
 Serial.println("DONE");

 float precise_value = (((sensors.getTempCByIndex(0) -
Basic_low) *ReferenceRange) / BasicRange)
 Serial.print("Temperature is: ");
 Serial.print(sensors.getTempCByIndex(0));
   delay(1000);
}
```

Figure 15: Coding of temperature sensor

‘DallasTemperature.h’ and ‘OneWire.h’ libraries are included here. Through this sensor we can directly get the reading in digital form. The only data pin is linked to Arduino nano digital pin 9.

Set up the instances where it used to communicate with the sensor.

1. Firstly sensor is loaded using the sesnors.begin() .
2. It opens a function where it fetches for temperature using ‘sensors.requestTemperatures()’ . it works has a listener function.

3. The probe data read byte and byte and convert them to and analog data pattern and stored in sensors.getTempCByIndex(0).
4. At last precise value is taken by sending above value to the formula.

The value stored in 'temperature' is then sent to AWS DynamoDB to show the temperature of water in mobile interface.

2.2.3 Turbidity Sensor

A. Turbidity sensor module circuit diagram

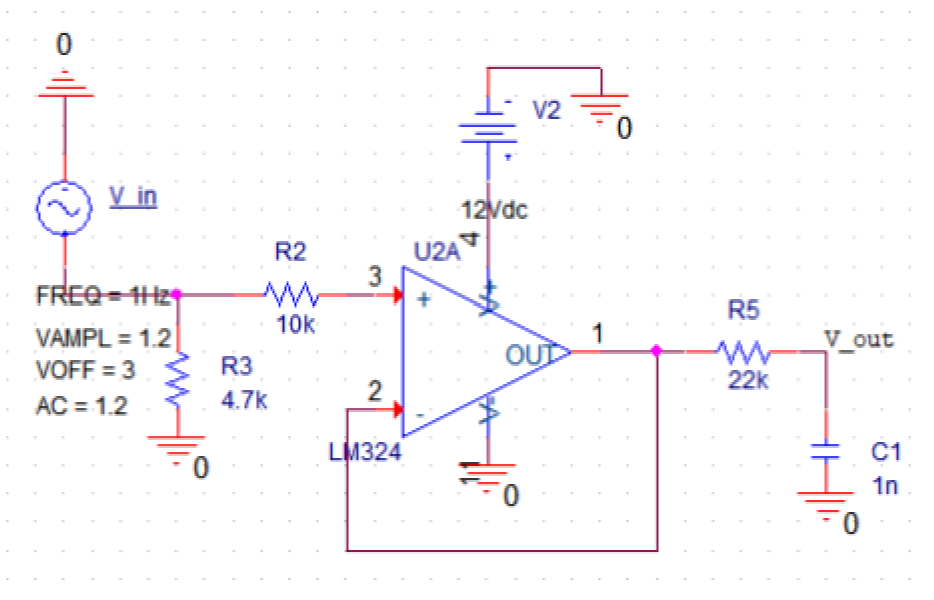

Figure 16: Circuit of turbidity module

The Arduino turbidity sensor recognizes water quality by estimating the degrees of turbidity, or the obscurity. It utilizes light to identify suspended particles in water by estimating the light conveyance and dispersing rate. A few alterations are done to improve execution. Likewise, operational amplifier LMV358 is utilized, furthermore this is simpler to be patched physically.

Turbidity module which fills in as a segregation between probe and Arduino broad by connecting analog wire A6 pin. With this connector circuit, turbidity value will be more accurate.   V_out in the circuit, will be associated with analog pin A6. Reading will be available in voltage. So, we have converted it from voltage to nephelometric turbidity units(ntu).

B. Calibration of turbidity sensor

The turbidity is adjusted with the assistance of tube. It is a proficient and minimal effort strategy to acquire the turbidity   in water. This built using a glass chamber with a distance across of 3cm, a print height unit and a review plate. It can gauge turbidity from 0 to 280 ntu. The following equation is used to test changes over the length in cm to turbidity in ntu.

$$Depth\ in\ CM\ =\ 244.14 * (Turbidity)^{-0.662} \qquad (2)$$

We prepared a solution mixing water with milk. Which will be utilized as adjustment answer for the turbidity sensor. At that point, the sensor's test was embedded in various examples, giving the yield voltage at various turbidity. To improve exactness, the first analog information from Arduino (0-1024) is utilized for calibration rather than voltage. A table of simple incentive at various turbidity was developed. The **figure 3** shows the procedure to follow to capture these readings.

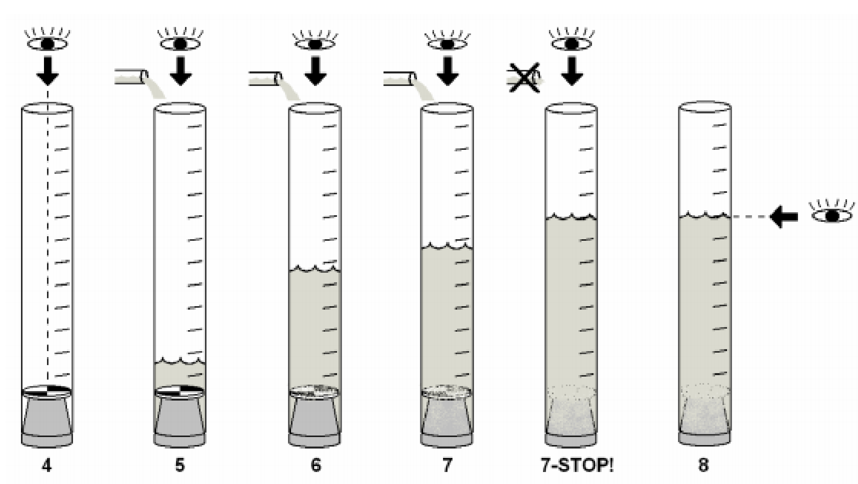

Figure 17: Turbidity tube

Table 5: Readings of the turbidity tube

Length/cm	Turbidity/NTU	Analog Value
6	280	495
8	175	565
11	108	637.5
13	84	663.5
15	68	689.5
17	56	706
18	51	714
19	47	721
20	44	730
21	41	735.5
23	35	748
24	33	750
25	32	755.5
26	31	762
27	30	765

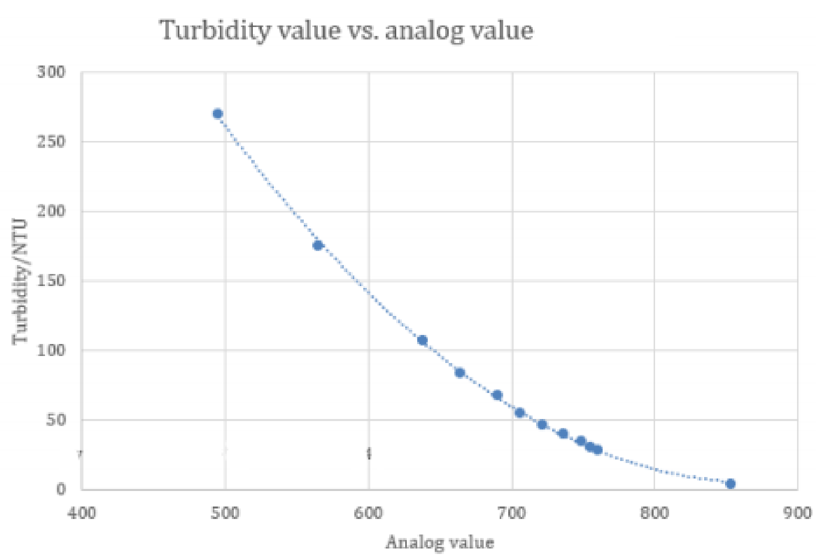

Figure 18: Graph of turbidity vs Analog value

From the figure 18, it very well may be seen that simple incentive from 766 to 495 relates to 30~280 Ntu. The fluctuation is excluded by taking the mean of six continuous measurements. The connection between turbidity and analog value is in equation 3.

$$y = -1120.4\,x^2 - 5742.3x - 4353.8 \quad (3)$$

The adjustment of turbidity sensor was done at 26.3ºC steady temperature, anyway at various temperatures, the analog output can shift generously. As per figure 19, condition temperature will change the yield voltage of the probe, the slant for analog output are - 2.05 ºC.

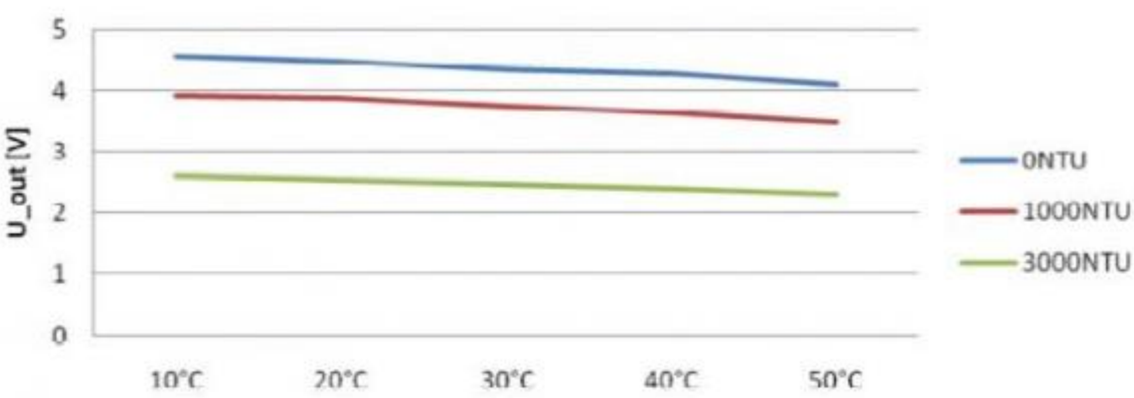

Figure 19: Graph of voltages

If the steady temperature is 26.3 then the equation 4 will build up in this manner.

$$Precise\ analog\ value\ \approx\ actual\ analog\ value\ +\ (T\ -\ 26.3)\ \times\ 2.05 \quad (4)$$

The turbidity probe is adjusted in a room at 26.3ºC and the water is 26.3ºC too, so the alignment condition is substantial since given that the water is at 26.3ºC, the temperature must be mulled over to ensure the test's precision is independent.

The yield from turbidity sensor in arrangement at fixed turbidity 100 NTU, when the temperature is between 26 ºC to 35 ºC is appeared in table underneath.

Table 6: Temperature readings

Temperature	Analog Value
26.6	631
28.5	627.3
29.4	624.1
30.4	621.5
32.3	617.2
33.4	614.2
34.1	613.3
34.7	612.8

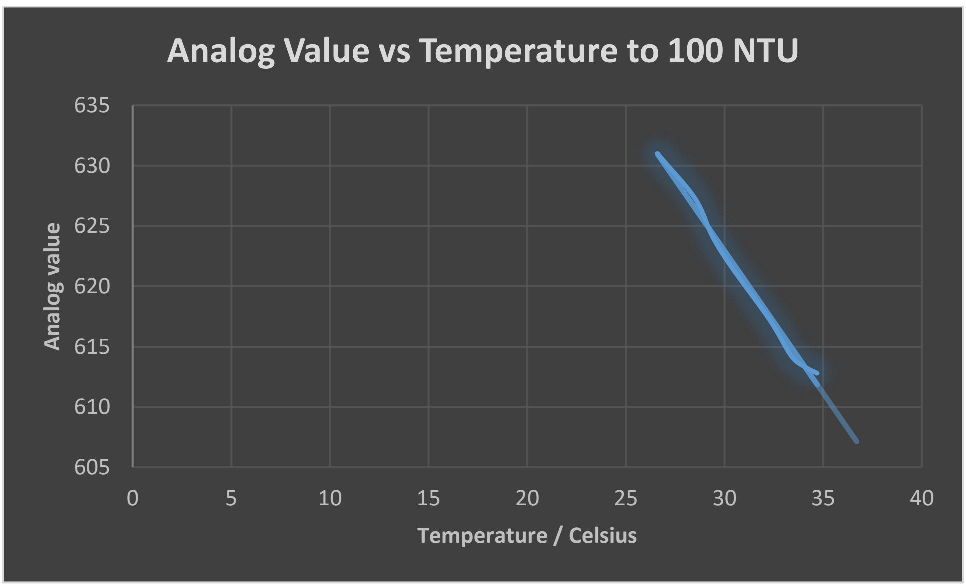

Figure 20: Graph of analog value vs temperature

As expressed in table 6, a chart is outlined to discover the connection among analog values and temperature. It is seen that the information focuses structure a straight line with a slant of -2.3562. It is close to the reference esteem (2.05). The underneath equation is utilized to change over the yield at any temperature.

$$Precise\ analog\ value\ \approx\ actual\ analog\ value + (T - 26.3) \times 2.3562 \qquad (5)$$

By taking temperature into thought, the polynomial condition equation 5 is combined with temperature adjustment condition equation 6, the resulting equation will be as follows.

$$Turbidity\ (NTU) = -1120.4 \times (x + (T - 26.3) \times 2.3116)\ 2 + 5742.3 \times (x + (T - 26.3) \times 2.3116) - 4353.8 \qquad (6)$$

C. Coding of the sensor

```
void readTurbidity() {
  turbSensorVal = analogRead(TURBIDPIN);
  turbSesnorVal = turbSensorVal + (temp - 26.3)* 2.3562
  turbVolt = turbSensorVal * (5.0 / 1024.0);
  if (turbVolt > 4.20024)
    turbVolt = 4.20024;
  turbidity = -1120.4 * sq(turbVolt);
  turbidity = turbidity + (5742.3 * turbVolt);
  turbidity = turbidity - 4352.9;
}
```

Figure 21: coding of temperature sensor

The following are the process of handling turbidity sensor.

1. power on the circuit to capture sensor readings.

2. Through the analog pin A6 analog value is taken to 'TurbSensorVal' variable.

3. To eliminate the impact of temperature its use the equation X.

4. Finally with calibration equation analog value is converted to turbidity value in the units of ntu.

2.2.4 Conductivity Sensor

A. Conductivity Circuit Diagram

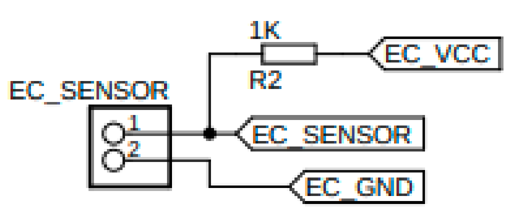

Figure 22: Conductivity circuit

Conductivity is the capacity of a mixture, a metal to pass an electric flow. Current is conveyed by cations and anions while in metals it is conveyed by electrons. In liquid plans the level of ionic quality contrasts.

Conductivity might be estimated by applying a rotating electrical flow (I) to two cathodes drenched in a solution and estimating the subsequent voltage (V). The cations move to the negative terminal, the anions to the positive cathode and the arrangement act as an electrical transmitter.

Hence the conductivity sensors in the market and much expensive and due to covid pandemic we were unable to purchase this conductivity sensor. We had the basic idea behind the sensor, at last we decide to design and build a conductivity sensor.

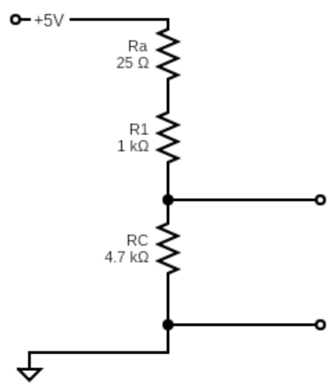

Figure 23: Flow of current with resistors

The functionality of the circuit diagram will be discussed here.

- Ra = 25Ω - Internal resistance in Arduino pin.
- R1 = 1000Ω - This resistance used for voltage divider circuit. The value of this resistance defines the resolution of the measuring range of the sensor.
- Rc – Calculated resistance between two terminals.

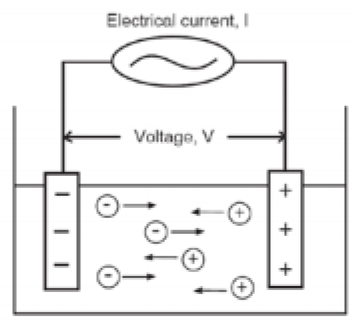

Figure 24: Voltage transfer pattern

Using Ohm’s law, we can determine the water resistance (R).

$$R = \frac{v}{I} \tag{7}$$

The reciprocal of the resistance (R) of a water between two electrodes can be used define conductance (G). In the above equation ‘V’ and ‘I’ are voltage and current, respectively. Conductance measured in Siemens (S).

$$G = \frac{1}{R} \tag{8}$$

The proportion to the displacement (D) between the electrodes to the area (A) of the electrodes known as cell constant (K).

$$K = \frac{D}{A} \tag{9}$$

This demonstrates particles in arrangement will direct electricity. The ability of conductivity is to pass electricity. The conductivity perusing of a testing samples will change with temperature. The conductivity (C) is characterized as:

$$C = K * G \tag{10}$$

B. Calibration of the sensor

The adjustment cycle of the sensor incorporates two arrangements with 1.515 (mS) and 10.86 (mS) conductivity. These arrangements are reasonable for estimating conductivity with K=1.76 test.

The alignment cycle follows the beneath steps:

1.Dip the temperature probe in the solution for at any rate 1-2 minutes all together for the arrangement to balance out its temperature.

2. Spot the conductivity sensor into 1.515 (mS/cm) arrangement and mix the solution.
3. Ensure the arrangement is settled at 26.3°C degree and watch the acquired conductivity voltage.
4. Subsequently the test the setup inside the 10.88 (mS/cm) arrangement and like the last advance the conductivity arrangement ought to be watched also, recorded.
5. The graph can be plotted using above values for the temperature of 26.3 ^{o}C.

While playing out the alignment cycle, a steady voltage was gotten from every arrangement table 7. The alignment chart was portrayed dependent on the acquired outcomes from the beneath table.

Table 7: Conductivity readings

Solution	Voltage
1.515	392.34
10.88	3350.3

C. Coding

```
void readEC() {
  pinMode(ECVCC, OUTPUT);
  pinMode(ECGND, OUTPUT);
  delay(30);

  digitalWrite(ECVCC, HIGH);
  digitalWrite(ECGND, LOW);
  ecSensorVal = analogRead(ECPROBE);
  digitalWrite(ECVCC, LOW);
  pinMode(ECVCC, INPUT);
  pinMode(ECGND, INPUT);

  ecVolt = ecSensorVal * (5.0 / 1024.0);
  Rc = (ecVolt * R1) / (5.0 - ecVolt);
  Rc = Rc - Ra;
  EC = 1000 / (Rc * K);
  EC25   =  EC / (1 + tempCoef * (temp - 25.0));
  ppm = (EC25) * (PPMconv * 1000);
}
```

Figure 25: Coding of conductivity sensor

2.2.5 Communication Structure

A. Overview

The communication medium to transmit sensors data to the user is achieved through WI-FI. In here, we have implemented two broads Nodemcu ESP8266 and Arduino Nano. For data communication within the broads done via serial communication using a logic level convertor. At last, transmission between the smart tool and the AWS DynamoDB built up using Message Queuing Telemetry Transport (MQTT) protocol.

B. Coding

```
#include <ESP8266WiFi.h>
#include <PubSubClient.h>
#include <ArduinoJson.h>
#include <NTPClient.h>
#include <WiFiUdp.h>
#include "FS.h"
#define WIFI_SSID "###"
#define WIFI_PASSWORD "#######"

const char* AWS_endpoint = "a2w43nj3we329s-ats.iot.us-
west-2.amazonaws.com";

char receivedData [256];
const size_t jsonCapacity = JSON_OBJECT_SIZE(10) + 30;
char msgOut[256];

WiFiUDP ntpUDP;
NTPClient timeClient(ntpUDP, "pool.ntp.org");

void callback(char* topic, byte* payload, unsigned int length) {
  Serial.print("Message arrived [");
  Serial.print(topic);
  Serial.print("] ");
  for (int i = 0; i < length; i++) {
    Serial.print((char)payload[i]);
  }
  Serial.println();
}

WiFiClientSecure espClient;
PubSubClient client(AWS_endpoint, 8883, callback, espClient);

void setup() {
  Serial.begin(9600);
  Serial.setDebugOutput(true);
  pinMode(LEDPIN, OUTPUT);
  digitalWrite(LEDPIN, LOW);
  setup_wifi();
  delay(1000);
}
```

Figure 26: Coding of Wi-Fi connection

2.3 Commercialization ascepts of the product

The convenient smartphone pursues water quality estimation solution to quantify water quality for various applications. The ‘E-tongue application interface with different progressed highlights, for example, area-based information collection, stockpiling on the cloud server, water borne disease prediction, precaution and healthcare, sensor alignment and a lot more has been presented. The solution has been tried for different water quality estimation applications. The polynomial Kernel svm and random forest algorithms have been implemented to chronic kidney disease and predict water quality index, respectively. Introduced work comprises of different advancements, for example, location water quality estimation, water quality parameters forecast, hassle-free activity, chronic kidney disease predictions and more. It has been discovered that research product could be an ease apparatus for water quality estimation at different country and metropolitan spots. The general framework is financially savvy, compact, and simple to use for any individual.

2.4 Testing and implementation

2.4.1 Implementation

E-tongue tool was implemented as the solution for identification of good quality water. Basically, the following items were used to build the structure.

- Three feet water pipe
- Sockets
- Stop value
- Wooden blocks

Figure 27: Prototype model

As in figure 27, we have made a portable device which can carried out at any place to capture water samples for real-time predictions. To capture sensor readings, we have built three cylinders. From that two of them are utilized to gather sensor data, and the remaining one as water inlet. All the sensor probes are placed on the lids of the cylinders.

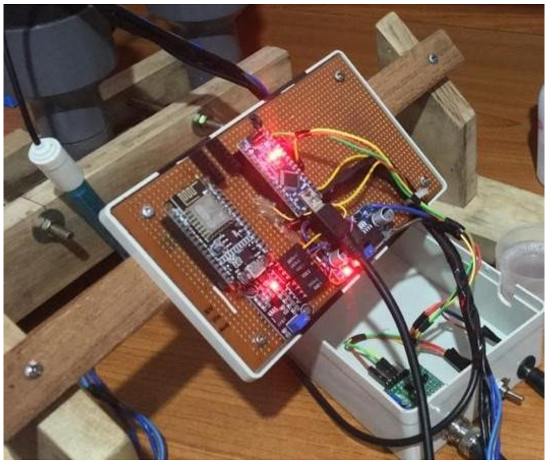

Figure 28: Circuits embedded in e-tongue

In Figure 28, it shows how the circuit was implement foe ‘E-tongue’ tool. The followings are the components used.

- AC Adapter
- Buck Convertor
- Nodemcu ESP8266
- Arduino Nano
- Logic level Convertor
- Ph Module
- Turbidity Module

With the use of above components, we were able to construct this intelligent tool. 12v current supply is given to operate this unit. Since 12v is high to operate these broad we have safely reduced the voltage up to 7v with the help of buck convertor. This is used to protect the broads. Nodemcu operates with 3.3v while nano works with 5v. For bridging esp8266 and Arduino nano we utilized logic level convertor to act as receiver and transmitter. In this manner we were able to accomplish hardware sector.

Backend created using node language to fetch real-time for predictions. It takes values from the sensors and send it to the database as well as for the device. We keep log of readings for seasonal forecasting.

2.4.2 Testing

In the testing phrase, we had several testing stages. Testing is the most prominent step in software life cycle which helps to give a standard product with error free at the end. In this application we started testing with unit testing. Where you are testing every component, functionality is working up to required level. It will be error free unit after you pass this testing and step of testing is integrated testing. There we combine all the components together and check for the main functionalities. Afterwards system testing is carried out where all system gets evaluated. Finally, user acceptance test carried with peer members. With their responses we came up the most suited solution after rectifying bugs and hardware issues. The following tables shows some test cases carried out during testing.

Test case ID	001
Test case description	Analyze pH probe readings
Test procedure	a. Remove the cleaning solution from the probe and wipe out with a paper. b. Pour water to the upper level of the cyclinders. c. Power on the circuit with 12v. d. Display the pH value in the serial monitor
Test data	Water sample
Expected result	pH reading within the range of 6-8. Because water is a cleaned sample from a ground water resource
Actual result	6.5
Final result	Pass

Test case ID	002
Test case description	Fetching readings from all the sensors.
Test procedure	a. Remove the cleaning solution from the probe and wipe out with a paper. b. Pour water to the upper level of the cyclinders. c. Immerse the sensors by closing the two lids of the cylinders. d. Power on the circuit with 12v. e. Display all the sensor values in the application monitor

Test data	Water sample
Expected result	All the readings should be within the reference ranges.
Actual result	Ph= 6.6 , EC= 0.13 , Temperature= 26.8 , Turbidity= 0.02 , TDS= 103
Final result	Pass

Test case ID	003
Test case description	check readings from all the sensors.
Test procedure	a. Pour water to the upper level of the cyclinders. b. Immerse the sensors by closing the two lids of the cylinders. c. Power on the circuit with 12v. d. Press display button.
Test data	Water sample
Expected result	All the readings should be within the reference ranges and it should fetch from the node api .
Actual result	As expected results were fetched through api
Final result	Pass

3. RESULTS AND DISCUSSION

3.2 Results

The research is focused on implementing a water quality measuring device with the smart application ensuring real time data. The goals were to predict water quality index, outbreak of water borne diseases and forecasting water quality parameters seasonal wise. In this section it provides the results of the experiments performed to solve the main research question.

E-Tongue device shows the ph level, temperature, turbidity, and total dissolved solids of the tested water samples from groundwater resources. The following table 8 shows some samples gathered from different water sources and figure 29 and 30 show user interfaces.

Table 8: Results of groundwater samples

Electric Conductivity (mS/cm	ph	TDS (ppm)	Temperature (C)	Turbidity (ntu)
0.23	7.16	161	25.44	0.02
0.14	5.07	97	25.56	197.51
0.23	6.95	159	25.25	0.02
7.98	5.58	5586	26.69	0.02
0.31	5.83	218	27.06	3.77
0.24	5.84	171	27.19	3.77
0.24	6.82	169	25.69	0.02
0.08	6.99	58	28.69	0.02
0.08	6.79	53	28.19	0.02

Figure 29: Home Screen

Figure 30 : WQI prediction screen

3.2 Research Findings

As stated in introduction, initial concept of this research is to implement a smart device to predict water quality parameters, water borne disease predictions and forecasting water parameters which helps end-users to identify the quality of daily consumption of their water sources and mean while this can be used in industrial level to test water samples in rea-time. Since the current procedure is time consuming.

A. Selecting the best combinations of sensors.

The key component of the device was the sensors. Analyzing literature surveys and earlier findings we were able to out the most suitable sensors which is need for predictions. They were ph probe, turbidity, temperature, and electric conductivity and by using conductivity there was a possibility of calculating total dissolved solids. Since conductivity sensors are much expensive. We had to build the conductivity sensor by utilizing basic concepts of science.

Among these parameters conductivity and turbidity depend on environment temperature. We took this also into consideration to give an accurate result by calibrating the sensors accordingly.

3.3 Discussion

E-tongue smart application and intelligent device would be beneficial in enhancing the prevailing procedure of checking groundwater quality immensely in next couple of years to come. consequently, this tool will empower in national drainage and water supply department plants which checks groundwater. Anyway, the accompanying would be thought of while operating this gadget. The smart app and smart device should be within the range of WI-FI. in any case different methods of innovation like Bluetooth would be a reasonable choice.

The sensors implement in this system are fragile, this should use with high care and ranges of this sensors are comparatively less when compared to industrial ones. long use of these sensors is not advised to follow. Thus, this device can give a better prediction than current devices in the market.

3.4 Summary of student contribution

Member	Component	Tasks
AMPB Alahakoon	Device implementation and Sensor calibrations	• Feasibility study for the requirement needed. • Identify the most suitable sensors which will be used to measure water quality by training models. • Developing a smart IoT embedded device for measure quality of water. • Constructing new sensor to measure conductivity levels in water. • Implementing data transmission media to transfer data between two broads and between database and smart application • Calibrating sensors to give away readings with minimum error fluctuation. • Improving accuracy of sensors by considering environment factors. • Implement an inception environment for the research team to test water samples.

4. CONCLUSION

A comprehensive approach is presented to detect the water quality and predict water quality index levels by location wise. This efficient and portable prototype design is implemented to run over from the limitation of prevailing water quality measuring solutions. The intelligent system subsists with five parameters, real-time database, integrated circuits, and a smart mobile application. A complete study was conducted in section 1 to figure out the best water parameters to use in this development.

In here four sensors are utilized to get five water parameters. These sensors are pH, turbidity, temperature, electric conductivity. Using conductivity sensor value, we would calculate total dissolved solids in water samples. After testing a sample these values will send to the cloud database through wireless transmission. From the cloud server, readings are fetched to the smart mobile application for analyzing and to do the water quality index prediction, Chronic kidney disease outbreak prediction from location wise at last water parameter forecasting.

Despite the fact that these sensors are not a grade standards, the E-tongue intelligent device prototype looks good to measure water parameters and it can even now be beneficial over traditional strategy for estimating water parameters use in the national drainage and water supply department which is tedious and requires labor power for operations. The accuracy of sensors are increased with proper calibration methodologies and it gives smaller error rate which is close to 0.4%.

As in the research question , we were able to fulfill the objectives and specific objectives of this research and through this smart device (E-tongue) it is possible to help the community in sri lanka in the betterment of consuming clean and quality water.

5. REFERENCES

[1]Hand Book for Water Consumers. (2014). 1st ed. Kandy: Greater Kandy Water Supply Project, Pahala Kondadeniya, Katugastota, Sri Lanka.

[2] R. Nithyanandam, T. W. Huan and N. H. T. Thy. (2015). Case Studies: Analysis of Water Quality in Sungai Batu Ferringhi . Journal of Engineering Science and Technology, EURECA 2014 Special Issue, pp. 15 – 25.

[3] N. Vijayakumar, R. Ramya. (2015). The Real Time Monitoring of Water Quality in IoT Environment. International Conference on Circuit, Power and Computing Technologies [ICCPCT].

[4] A.S. Rao, S. Marshall, J. Gubbi, M. Palaniswami, R. Sinnott, V. Pettigrove.(2013). Design of Low-cost Autonomous Water Quality Monitoring System. International Conference on Advances in Computing, Communications and Informatics (ICACCI).

[5] A. Alkandari, M. alnasheet, Y. Alabduljader and S. M. Moein. (2012). Water monitoring system using Wireless Sensor Network (WSN): Case study of Kuwait beaches. 2012 Second International Conference on Digital Information Processing and Communications (ICDIPC), Klaipeda City, pp. 10-15.

[6]Dong He, Li-Xin Zhang. (2012).The Water Quality Monitoring System based on Wireless Sensor Network. Report: Mechanical and Electronic Information Institute, China University of Geo Science, Wu Hen, China.

[7]M. K. Khurana, R. Singh, A. Prakash, R. Chhabra.(2016). An IoT Based Water Health Monitoring System. International Journal of Computer Technology and Applications (IJCTA). pp. 07-13.

[8] M. Deqing, Z. Ying, C. Shangsong. (2012). Automatic Measurement and Reporting System of Water Quality Based on GSM. International Conference on Intelligent System Design and Engineering.

[9]Geetha, S. and Gouthami, S. (2016). Internet of things enabled real time water quality monitoring system. Smart Water, 2(1).

[10]Gokulanathan, S., Manivasagam, P., Prabu, N. and Venkatesh, T. (2019). GSM Based Water Quality Monitoring System Using Arduino. Shanlax International Journal of Arts, Science and Humanities, 6(4), pp.22-26.

[11]Muhammad, R. and Aisha, H. (2017). Low-cost, Real-Time, Autonomous Water Quality Testing and Notification System. 17th ed. [ebook] Available at: http://paper.ijcsns.org/07_book/201705/20170537.pdf [Accessed 29 Feb. 2020].

[12]Samsudin, S., Salim, S., Osman, K., Sulaiman, S. and Sabri, M. (2018). A Smart Monitoring of a Water Quality Detector System. 10th ed. [ebook] Available at: https://pdfs.semanticscholar.org/8f4d/898cd93bb54be6b6237946dd8ba1e42f28f0.pdf [Accessed 29 Feb. 2020].

[13]Ieeexplore.ieee.org. (2020). Water quality monitoring with internet of things (IoT) - IEEE Conference Publication. [online] Available at: https://ieeexplore.ieee.org/abstract/document/8313015 [Accessed 29 Feb. 2020].

[14]K. H. Kamaludin and W. Ismail, "Water quality monitoring with internet of things (IoT)," 2017 IEEE Conference on Systems, Process and Control (ICSPC), Malacca, 2017, pp. 18-23.

[15] Y. Xu and F. Liu, "Application of Wireless Sensor Network in Water Quality Monitoring - IEEE Conference Publication", Ieeexplore.ieee.org, 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8006031. [Accessed: 26- Apr- 2019].

[16] “DS18B20-PAR 1-Wire Parasite-Power Digital Thermometer” ", maximintegrated.com, 2019. [Online]. Available: https://datasheets.maximintegrated.com/en/ds/DS18B20-PAR.pdf. [Accessed: 26- Apr- 2019].

[17] "Turbidity_sensor_SKU__SEN0189-DFRobot", Dfrobot.com, 2019. [Online]. Available: https://www.dfrobot.com/wiki/index.php/Turbidity_sensor_SKU:_SEN0189. [Accessed: 26- Apr- 2019]

[18] Pasco.com, 2019. [Online]. Available: https://www.pasco.com/support/. [Accessed: 27- Apr- 2019].

[19] "Analog EC sensor SKU SEN-DFRobot", Dfrobot.com, 2019. [Online]. Available: https://wiki.dfrobot.com/Analog_EC_Meter_SKU_DFR0300 .[Accessed: 26- Apr- 2019].

6. GLOSSARY

End User	People who interact with the smart mobile application
Sensor data	Data which have gathered from water quality measuring device from main system (e-tongue).
Arduino Circuits	circuit type that we use to implement this device
Machine learning	Computational learning mechanism

7. APPENDICES

7.1 Coding snippets

A. ESP8266 Module

```
#include <ESP8266WiFi.h>
#include <PubSubClient.h>
#include <ArduinoJson.h>
#include <NTPClient.h>
#include <WiFiUdp.h>
#include "FS.h"

#define WIFI_SSID "pandu"
#define WIFI_PASSWORD "a5470234Kb"

#define LEDPIN D0

const char* AWS_endpoint = "a2w43nj3we329s-ats.iot.us-
west-2.amazonaws.com";

char receivedData [256];
const size_t jsonCapacity = JSON_OBJECT_SIZE(10) + 30;
char msgOut[256];

double temp = 0.0;
double turbid = 0.0;
double ec = 0.0;
int ppm = 0;
double ph = 0.0;

unsigned long prevMillis = 0;
const long ECPeriod = 5500;

WiFiUDP ntpUDP;
NTPClient timeClient(ntpUDP, "pool.ntp.org");
```

```
#include "etongueESP.h"

void callback(char* topic, byte* payload, unsigned int length) {
  Serial.print("Message arrived [");
  Serial.print(topic);
  Serial.print("] ");
  for (int i = 0; i < length; i++) {
    Serial.print((char)payload[i]);
  }
  Serial.println();
}

WiFiClientSecure espClient;
PubSubClient client(AWS_endpoint, 8883, callback, espClient);

void setup() {

  Serial.begin(9600);
  Serial.setDebugOutput(true);
  pinMode(LEDPIN, OUTPUT);
  digitalWrite(LEDPIN, LOW);
  setup_wifi();
  delay(1000);
  if (!SPIFFS.begin()) {
    Serial.println("Failed to mount file system");
    return;
  }

  Serial.print("Heap: "); Serial.println(ESP.getFreeHeap());

  // Load certificate file
  File cert = SPIFFS.open("/cert.der", "r"); //replace cert.crt
eith your uploaded file name
  if (!cert) {
    Serial.println("Failed to open cert file");
  }
  else
    Serial.println("Success to open cert file");

  delay(1000);
```

```
  if (espClient.loadPrivateKey(private_key))
    Serial.println("private key loaded");
  else
    Serial.println("private key not loaded");

  // Load CA file
  File ca = SPIFFS.open("/ca.der", "r"); //replace ca eith your
uploaded file name
  if (!ca) {
    Serial.println("Failed to open ca ");
  }
  else
    Serial.println("Success to open ca");

  delay(1000);

  if (espClient.loadCACert(ca))
    Serial.println("ca loaded");
  else
    Serial.println("ca failed");

  Serial.print("Heap: "); Serial.println(ESP.getFreeHeap());

  digitalWrite(LEDPIN, HIGH);
}

void loop() {

  if (!client.connected()) {
    reconnect();
  }
  client.loop();

  if (millis() - prevMillis >= ECPeriod) {
    prevMillis = millis();
    serialLoop();
    delay(300);
    pushData();
  }
}
```

```
void serialLoop() {
  boolean newMsgReceived = false;
  int i = 0;
  while (Serial.available() > 0) {
    receivedData[i] = (int)Serial.read();
    i++;
    yield();
    newMsgReceived = true;
  }
  if (newMsgReceived) {
    Serial.print("Data received: ");
    Serial.print(receivedData);
    Serial.println();
    parseJsonData();
  }
}
void parseJsonData() {
  while (!Serial) continue;
  DynamicJsonDocument doc(jsonCapacity);
  DeserializationError error = deserializeJson(doc,
receivedData);
  if (error) {
    Serial.print(F("deserializeJson() failed: "));
    Serial.println(error.c_str());
    return;
  }
  temp = doc["temp"];
  turbid = doc["turbid"];
  ec = doc["ec"];
  ppm = doc["ppm"];
  ph = doc["ph"];
  String T = String(temp);
  String TU = String(turbid);
  String PHv = String(ph);
  String TDS = String(ppm);
  String ELEC = String(ec);
  snprintf (msgOut, 256, "{\"ec\" : \"%s\", \"temp\" : \" %s\",
\"ph\" : \"%s\" , \"turb\" : \"%s\", \"tds\" : \"%s\"}",
ELEC.c_str(), T.c_str(), PHv.c_str(), TU.c_str(), TDS.c_str());
}
```

B. Arduino Nano Microcontroller

```
#include <OneWire.h>
#include <DallasTemperature.h>
#include <ArduinoJson.h>
#define ECPROBE A0     // Analog read pin for EC probe
#define ECVCC 15       // Digital VCC pin for EC probe
#define ECGND 16       // Digital GND pin for EC probe
#define ONE_WIRE_BUS 9    // Temperature sensor on D9
#define TURBIDPIN A6   // Analog read pin for turbidity sensor
#define PHPIN A3       // Analog pin for pH sensor

int R1  = 1000;
int Ra = 25;
int ppm = 0;
float PPMconv = 0.7;
float tempCoef = 0.019;
float K = 1.76;    // EU plug K=1.76; US plug K=2.88
float ecSensorVal = 0;
float ecVolt = 0;
float Rc = 0;
float temp = 10;
float EC = 0;
float EC25 = 0;

unsigned long prevMillis = 0;
const long ECPeriod = 5500;

float turbSensorVal = 0;
float turbVolt = 0.0;
float turbidity = 0.0;
float phValue = 0.0;
float calibration = 21.40; //change this value to calibrate
int sensorValue = 0;
unsigned long int avgValue;
int buf[10], pHtemp;

const size_t jsonCapacity = JSON_OBJECT_SIZE(5);
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature tempSensor(&oneWire);
```

```
#include "etongueNANO.h"
void setup() {
  /*
     ECVCC and ECGND pins are set at high-impedance state
     when the EC probe is not used. This is required to
     avoid ground loop issues during simultaneous operation
     of sensors.
  */
  pinMode(ECVCC, INPUT);
  pinMode(ECGND, INPUT);
  Serial.begin(9600);
  delay(300);
  //  Serial.println("Initializing...");
  pinMode(LED_BUILTIN, OUTPUT);
  digitalWrite(LED_BUILTIN, HIGH);
  delay(900);
  tempSensor.begin();
  delay(900);

  R1 = (R1 + Ra);
  //  readEC();
  delay(6000);
  //  Serial.println("Done!");
  digitalWrite(LED_BUILTIN, LOW);
}

void loop() {
  if (millis() - prevMillis >= ECPeriod) {
    digitalWrite(LED_BUILTIN, HIGH);
    prevMillis = millis();
    readEC();
    readTurbidity();
    readPh();
    sendJSON();
    //    printData();  // Use only for debugging
    delay(100);
    digitalWrite(LED_BUILTIN, LOW);
  }
}
```