**VISVESVARAYA TECHNOLOGICAL UNIVERSITY BELAGAVI-590014, KARNATAKA**

**

**A PROJECT REPORT ON**

**“IoT BASED AQUAPONICS MONITORING SYSTEM”**

**Submitted in partial fulfilment of requirement of 8th semester of B.E course during the year 2022-2023.**

**SUBMITTED BY**

**PADMAVATI (3GN19EC034)**

**SUKESHINI (3GN19EC052)**

**SUMAVATI (3GN19EC053)**

UNDER THE GUIDANCE OF

**Dr. SAVITA SOMA**



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING**

**GURU NANAK DEV ENGINEERING COLLEGE BIDAR, KARNATAKA**

**ACADEMIC YEAR 2022-2023**

**GURU NANAK DEV ENGINEERING COLLEGE BIDAR, KARNATAKA**



**DEPARTMENT OF**

**ELECTRONICS AND COMMUNICATION ENGINEERING**

**CERTIFICATE**

This is to certify that the project work entitled **“IoT BASED AQUAPONICS MONITORING SYSTEM”** is a Bonafede work carried out by PADMAVATI (3GN19EC034), SUKESHINI (3GN19EC052), SUMAVATI (3GN19EC053), in partial fulfilment of the requirements for the Bachelor’s degree in ELECTRONICS AND COMMUNICATION ENGINEERING of VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELGAUM during the year 2022-2023. It is certified that this Project Report has been approved as it satisfies the academic requirements.

……………… ……………… ………………

**Dr. Savita Soma Dr. Kishan Singh Dr. Dhananjay Maktedar**

(Project Guide) (HOD, ECE) (Principal, GNDECB)

**EXTERNAL VIVA**

**Name of the examiner Signature with Date 1)……………………. ………………………**

**2)……………………. ………………………**

**ACKNOWLEDGEMENT**

We would like to express our deep sense of gratitude to our principal **Dr. Dhananjay Maktedar**, Guru Nanak Dev Engineering College, Bidar for his inspiration and academic support by providing good facilities to complete the Project.

Our sincere thanks to **Dr. Kishan Singh**, Head of the Department, Electronics and Communication Engineering for his whole hearted support in completion of this Project.

We are highly indebted to our Project coordinators **Dr. Savita Soma / Prof. Namratha** for guiding and giving us timely advice and suggestions in the successful completion of the project.

We are highly indebted to our project guide **Dr. Savita Soma** for guiding and giving us timely advice and suggestions in the successful completion of the project.

We thank all the staff members for supporting the completion of our project.

Also, we are thankful to my parents and all my dear friends who have directly or indirectly helped me. Finally, we express gratefulness to all those who knowingly or unknowingly helped us in successful completion of this project.

**PADMAVATI (3GN19EC034)**

**SUKESHINI (3GN19EC052)**

**SUMAVATI (3GN19EC053)**

**ABSTRACT**

Agriculture is the art of nurturing the plants and cultivation of soil for the production of food. In India the major amount of food production is depended on the agriculture. But now days use of chemicals like Sulphur dioxide and fluorine etc. for protecting the plants against the pesticides and also the use of other chemicals for the fast growth of the plants have caused damage to human health as well as soil pollution. The soil pollution results in the decreased fertility of soil which affects the growth of the plants. So, the solution for this is the growing the plants without using soil and with the help of only water. The development and growth of the plants with the help of only water is called as Hydroponic system. As the plants require nutrition's for their growth, instead of using other sources we are using the natural fish extract by fish farming which is called as Aquaculture. As the plants are growing with the help of water and fish extract it is called as Aquaponic system. When the pH level of water exceeds more than its desired level it becomes harmful for the aquaculture and also for the plants as well the variation in the temperature, humidity, TDS causes harm to the aquaponic system so by using pH sensor, TDS sensor, DHT11 sensor all the required parameters are managed and monitored with the help of the IOT. All the parameters are displayed on the Blynk app.

**CONTENTS**

|  |  |  |
| --- | --- | --- |
| **SL NO.** | **TITLE** | **PAGE NO.** |
| **CHAPTER 1** | **INTRODUCTION** | **1** |
| **CHAPTER 2** | **LITERATURE SURVEY** | **2 - 11** |
| **CHAPTER 3** | **PROBLEM STATEMENT** | **12** |
| **CHAPTER 4** | **METHODOLOGY**  4.1 Block diagram  4.2 Data flow diagram  4.3 Implementation | **13 - 19** |
| **CHAPTER 5** | **HARDWARE REQUIREMENTS**  5.1 Hardware components | **20 - 23** |
| **CHAPTER 6** | **SOFTWARE REQUIREMENTS**  6.1 Arduino Ide Appearance  6.2 Toolbar Button  6.3 Menu Bar  6.4 Edit  6.5 Sketch  6.6 Tools  6.7 Help | **24 – 35** |
| **CHAPTER 7** | **RESULT AND DISCUSSION**  7.1 Rain Sensor  7.2 pH Sensor Output  7.2.1 Serial Monitor Output  7.3 Moisture Sensor Output  7.3.1 Serial Monitor Output of Moisture  7.4 TDS Sensor Output  7.4.1 Serial Monitor Output of TDS  7.5 Dht11 Sensor Output  7.5.1 Serial Monitor Output of DHT11  7.6 Output Results  7.6.1 Rain Sensor  7.6.2 TDS Sensor  7.6.3 pH Sensor  7.6.4 Dht11 Sensor  7.6.5 Moisture Sensor  7.7 Snapshots | **36 - 46** |
| **CHAPTER 8** | **ADVANTAGES AND DISADVANTAGES**  8.1 Advantages  8.2 Disadvantages | **47** |
| **CHAPTER 9** | **FUTURE SCOPE** | **48** |
| **CHAPTER 10** | **CONCLUSION** | **49** |
| **SOURCE CODE** | **Source Code** | **50 - 51** |
| **REFERENCES** | **References** | **52** |

**TABLE OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **FIGURES** | **TITLE** | **PAGE NO.** |
| **4.1.1** | **Block diagram of aquaponic system** | **13** |
| **4.2.1** | **Level 0 Data flow diagram** | **15** |
| **4.2.2** | **Level 1 Data flow diagram** | **15** |
| **4.2.3** | **Level 2 Data flow diagram** | **16** |
| **4.2.4** | **Sequence diagram for aquaponic system** | **17** |
| **4.3.1** | **Implementation diagram for aquaponic system** | **17** |
| **5.1.1** | **DHT11 Sensor** | **20** |
| **5.1.2** | **DS18B20 Sensor** | **20** |
| **5.1.3** | **pH Sensor** | **21** |
| **5.1.4** | **TDS Sensor** | **21** |
| **5.1.5** | **Moisture Sensor** | **22** |
| **5.1.6** | **Rain Sensor** | **22** |
| **5.1.7** | **ESP8266 nodeMCU** | **23** |
| **6.1.1** | **Arduino IDE** | **24** |
| **6.2.1** | **Arduino IDE Tool Bar** | **25** |
| **6.3.1** | **Arduino IDE file** | **26** |
| **6.4.1** | **Arduino IDE edit** | **28** |
| **6.4.2** | **Arduino IDE line bar** | **29** |
| **6.4.3** | **Arduino IDE find bar** | **30** |
| **6.5.1** | **Arduino IDE sketch** | **31** |
| **6.5.2** | **Arduino IDE Add file** | **32** |
| **6.6.1** | **Arduino IDE tools** | **33** |
| **6.7.1** | **Arduino IDE help** | **35** |
| **7.1.1** | **Serial monitor output of rain sensor** | **36** |
| **7.1.2** | **Blynk app output of rain sensor** | **36** |
| **7.2.1** | **Serial monitor output of pH sensor** | **37** |
| **7.2.2** | **Blynk app output of pH sensor** | **37** |
| **7.3.1** | **Serial monitor output of moisture sensor** | **38** |
| **7.3.2** | **Blynk app output of moisture sensor** | **38** |
| **7.4.1** | **Blynk app output of TDS sensor** | **38** |
| **7.5.1** | **Serial monitor output of DHT11 sensor** | **39** |
| **7.5.2** | **Blynk app output of DHT11 sensor** | **39** |
| **7.6.1** | **Rain Sensor Results** | **40** |
| **7.6.2** | **Rain Sensor and TDS Sensor Results** | **40** |
| **7.6.3** | **Rain Sensor and TDS Sensor, pH Results** | **41** |
| **7.6.4** | **Rain Sensor and TDS Sensor, pH, DHT11 Results** | **41** |
| **7.6.5** | **Rain Sensor and TDS Sensor, pH, DHT11, Moisture Sensor Results** | **42** |
| **7.7.1** | **Project Demonstration Snapshot 1** | **42** |
| **7.7.2** | **Project Demonstration Snapshot 2** | **43** |
| **7.7.3** | **Project Demonstration Snapshot 3** | **43** |
| **7.7.4** | **Project Demonstration Snapshot 4** | **44** |
| **7.7.5** | **Project Demonstration Snapshot 5** | **44** |
| **7.7.6** | **Project Demonstration Snapshot 6** | **45** |
| **7.7.7** | **Project Demonstration Snapshot 7** | **45** |

**INTRODUCTION**

Agriculture has provided the most of the population with natural food but nowadays the use of pesticides has been increased for fast growth of the crops because of which it has caused damage to human health and the problems related to soil also increased in the past few years. Because of which we have introduced aquaponics system for the growth of the plants naturally with the help of fish extract. Hydroponics is used to supply nutrient to plants without soil. And aquaculture is just growing of fish naturally like fish farming. This system is automatically getting monitor by using IOT domain. Water temperature, surrounding temperature, humidity. pH factor, total amount of dissolvent and water temperature can be monitored using sensors. Grow light and Water pump get automatically turn on/off using relays. User gets notification of all data about via message using BLYNK/TUYA application.

Using combination of Aquaponics and IOT user will be benefited by following:

• Less water is required as water get reused. So, farmers don’t have to depend on rainwater.

• Less space is required as layer wise farming is done.

• Farmers don’t have to worry about soil related problem as it is soilless planting. Nutrients for plant are provided using fish extract.

• Profit percentage is increased.

This project focuses on making the agriculture soilless as well as making it feasible to use anywhere. For which we will be using all the knowledge of hydroponics and aquaponics. The aim is to make an aquaponics system that will help grow fishes and plants together indoor. The sensors are used to take certain readings which will let us know whether the system is normal or not. That will go to Blynk/TUYA cloud, which will keep the readings on the cloud, then send the readings to the Blynk/TUYA application on the smart phone. Using the knowledge of Arduino and the cloud, we will send the data to the cloud that will display the results. Using the readings, we will be able to know the system is working properly or not. Sensors, Arduino and Cloud is used to take readings and display the readings too.

**CHAPTER 2**

**LITERATURE SURVEY**

**[1] Title:** **An autonomous aquaponics system using 6LoWPAN based WSN** || **Authors:** N Hari Kumar, Sandhya Baskaran, Sanjana Hariraj, Vaishali Krishnan||2016.

The authors N Hari Kumar et.al stated that Aquaponics is a groundbreaking food production technique that combines both traditional methods of aquaculture and hydroponics to grow both fish and crops in a single integrated system. Aquaponics system uses fish wastes to provide essential nutrients to the plants. In return plants serve as a bio-filter for the fish in a symbiotic relationship. The purpose of this paper is to showcase how to build an efficient Internet of Things (IoT) application for aquaponics in order to create an autonomous, self-regulating system with the help of Wireless Sensor Network (WSN). An open standard of WSN called 6LoWPAN is being used in this system which helps us to construct a global infrastructure. The designed aquaponic system is composed of sensor devices that can sense and collect information of the various dimensions of the water quality involved and store it in a cloud database. This means that the human intervention would be considerably less when compared to the traditional methods of aquaponics. In addition to the traditional technique, with the help of Next-Gen Telco technologies, i.e., using their high bandwidth and low latency, infected fishes are detected automatically on time and treat them to ensure the balance in our aquaponics ecosystem. In this paper we propose an end-to-end system to enable "Connected Aquaponics" which includes WSN and Next-Gen Telco to increase the crop yield and provide organic sustainable food to the world community.

**[2] Title: IoT fuzzy logic aquaponics monitoring and control hardware real-time system ||** **Authors:** Adnan Shaout, Spencer G Scot || 2017.

The author **Adnan Shaout, Spencer G Scot** have made research onthe Aquaponics is a growing field in which fish and plants are grown together and mutual benefit each other. Fuzzy logic is used to evaluate the inputs and automatically provide the proper output. The system will monitor water temperature, pH, air temperature, and luminance. The system will control a light, heater, and alarm. The Arduino Uno R3 board was selected to be the hardware interface for inputs/outputs. Selecting the Arduino was based on MATLAB having a support toolbox to interface with the Arduino (ATMEGA8U2-MU) microcontroller. Updating of the input value and triggering twitter alerts was all done through using the free Thingspeak server tool that connects nicely with MATLAB via a toolbox.

**[3]** **Title:** **Enhancing aquaponics management with IoT-based Predictive Analytics for efficient information utilization ||** **Authors:** Divas Karimanzira, Thomas Rauschenbach || 2019.

The author Divas Karimanzira et.al stated that the modern aquaponic systems can be highly successful, but they require intensive monitoring, control and management. Consequently, the Automation Pyramid (AP) with its layers of Supervisory Control and Data Acquisition (SCADA), Enterprise Resource Planning (ERP) and Manufacturing Execution System (MES) is applied for process control. With cloud-based IoT-based Predictive Analytics at the fore marsh, it is worth finding out if IoT will make these technologies obsolete, or they can work together to gain more beneficial results. In this paper, they discussed that the enhancement of SCADA, ERP and MES with IoT in aquaponics and likewise how IoT-based Predictive Analytics can help to get more out of it. An example use case of an aquaponics project with five demonstration sites in different geographical locations will be presented to show the benefits of IoT on example Predictive Analytics services. Innovative is the collection of data from the five demonstration sites over IoT to make the models of fish, tomatoes, technical components such as filters used for remote monitoring, predictive remote maintenance and economical optimization of the individual plants robust. Robustness of the various models, fish and crop growth models, models for econometric optimization were evaluated using Monte Carlo Simulations revealing as expected the superiority of the IoT-based models. Our analysis suggest that the models are generally tolerant to the temperature coefficient variations of up to 15% and the econometric models tolerated a variation of for example feed ration size for fish of up to 4% and by the energy optimization models a tolerance of up to 14% by variations of solar radiation could be noticed. Furthermore, from the analysis made, it can be concluded that MES has several capabilities which cannot be replaced by IoT such as responsiveness to trigger changes on anomalies. It act as proxy when there is no case for sensors and reliably ensure correct execution in the aquaponics plants. IoT systems can produce unprecedented improvements in many areas but need MES to leverage their true potential and benefits.

**[4] Title: Urban aquaponics farming and cities-a systematic literature review Authors**: Rahmita Wirza, Shah Nazir || 2021.

The author Rahmita Wirza et.al discussed that an aquaponic system is considered to be a sustainable food production solution that follows circular economy principles and the biomimetic natural system to reduce input and waste. It is the combination of two mainly productive systems, a recirculating aquaculture system consists of fish and crustaceans farmed in a tank and hydroponic cultivation consists of vegetable cultured in medium other than soil. Both these systems are well-known around the globe by their performance of production, quality, and verified food safety. An aquaponic system is an industrious mechanism which incorporates impeccably with sustainable growth of intensive agriculture. The existing literature regarding the aquaponic production covers different species of vegetables and fish, a variety of layouts of system, and climate conditions. However, there is a lack of knowledge that can systematically present the existing state-of-the-artwork in a systematic manner. To overcome this limitation, the proposed research presents a systematic literature review in the field of urban aquaponics. This systematic literature review will help practitioners to take help from the existing literature and propose new solutions based on the available evidence in urban aquaponics.

**[5] Title: IoT enabled Aquaponics with wireless sensor smart monitoring || Authors :** Praveen C Menon || 2020.

Here author Praveen C Menon mentioned that the Aquaponics is an advanced and emerging farming style in which fish farming and vegetable farming turned out to be more professional and precise. This approach uses the latest technology for hydration and nutrition supply to plants and fish. This paper addresses some technical issues faced by the farmers using aqua-ponics and proposes the technical solution of an identified problem in traditional aquaponics using wireless sensors and a communication network with a GUI application. AWSM is a GUI solution for the addressed issues. AWSM based end-user mobile application alerts the farmer online with the water quality notification and suggestions even when the farmer is away. Results showed considerable improvements when AWSM based IoT application was implemented for the Aquaponics system when compared with the conventional method.

**[6]** **Title: Survey on IoT based automated aquaponics gardening approaches Authors:** KS Aishwarya, M Harish, S Prathibhashree, K Panimozhi || 2018.

Here author KS Aishwarya et.al have mentioned that the Aquaponics is a system which combines the aquaculture & hydroponics that grows fish and plants together in one system. It uses fish wastes to provide essential nutrients to the plants and in return the plants will purify the water and gives it back to the fishes. The purpose of this paper is to build an efficient system by implementing aquaponics system by using the technology of IoT (Internet of Things). By creating an automated System with the help of sensors interfaced with the Arduino board, it possible to automate fish feeding and water supply to the plants at the regular interval of time. Existing system that combines these technologies must overcome the fundamental issues like cost, food quality control and limited grow. In this paper they intend to propose a kit which contains all these features mentioned above, and that is helpful to provide the basic organic vegetation for the home.

**[7] Title: Development of an IoT-based aquaponics monitoring and correction system with temperature-controlled greenhouse || Authors:** Lean Karlo S Tolentino, Edmon O Fernandez, Romeo L Jorda, Shayne Nathalie D Amora, Daniel Kristopher T Bartolata, Joshua Ricart V Sarucam, June Carlo L Sobrepeña, Kristine Yvonne P Sombol || 2019.

The authors Lean Karlo S Tolentino et.al discussed that monitoring and automatic correction system for an aquaponics set-up in a temperature-controlled greenhouse using an Android device through Internet of Things (IoT). The system involves the acquiring of real time data detected by the light intensity sensor, and air temperature and humidity sensor. It also includes the monitoring of the pH level and temperature of the recirculating water of the system and the canopy area of the plant. If the acquired data is not within the threshold range, the correcting devices, namely grow lights, exhaust and inlet fans, evaporative cooler, aerator, and peristaltic buffer device were automatically triggered by the system to correct and achieve its normal status. The Internet remote access includes the effective wireless transmission and reception of data report between the system and an Android unit with the Android application in real-time.

**[8]** **Title: Architecture design of monitoring and controlling of IoT-based aquaponics system powered by solar energy || Authors:** Taji Khaoula, Rachida Ait Abdelouahid, Ibtissame Ezzahoui, Abdelaziz Marzak || 2021.

The authors Taji Khaoula et.al and mentioned that the demand for food is persistently increasing, the world is facing cannot be addressed with the land’s additional natural resources and exploitation. Currently, the agricultural industry is the world’s largest water user and consumes about 70% of the water in different processes. In this food and environmental crisis, the new methods for growing plants in the agricultural domain are various, one of these methods is aquaponics which is a sustainable agriculture system that combines two names: Aquaculture, which is the farming of fish and Hydroponic which is the cultivation of plants (Off-soil) using water enriched with mineral materials. Aquaponics comes as a solution to enhance farming and agriculture productivity. The attention towards the aquaponics system is increasing due to its high efficiency, availability, and fewer resources consumption. However, there is intensive automation, monitoring, and control requirement for a smart aquaponics system. To enhance the productivity of aquaponics, it is required to utilize IoT-based technologies and artificial intelligence (AI) algorithms. The main aim of this paper at hand consists of proposing a new contribution based on the Internet of Things (IoT) solution that could control and monitor water quality and environmental parameters by using the maximum of sensors (water level, water temperature, electrical conductivity, CO2, Ammonia Nitrogen (TAN), etc.), and actuators (water pump, etc.,) in aquaponics farming. This paper provides an innovative solution based on an interoperable, secure, scalable, low-cost, fully self-powered, flexible, reliable and generic IoT architecture that meets the requirements of aquaponics.

**[9]** **Title: Smart aquaponic system-based Internet of Things (IoT) ||**

**Authors:** Haryanto, M Ulum, A F Ibadillah, R Alfita1, K Aji and R Rizkyandi || 2018.

Authors Haryanto et.al mentioned that getting appropriate water source for fish and plant cultivation seems difficult. Moreover, the agricultural production is decreasing due to narrower lands so that land and water-saving technology combined with a variety of vegetable is important to produce maximum yield. Aquaponics is a sustainable agriculture system in a symbiotic environment by combining aquaculture and hydroponics. This water system should flow on the planting medium periodically to ensure the plants get the nutrients, while the water can be filtered properly by the medium. This research designed a smart aquaponics system that could control and monitor the degree of acidity, water level, water temperature, and fish feed that were integrated with internet-based mobile application. In this system, there was a sensor installed to retrieve data, which was then transmitted to Ubuntu IoT Cloud server that could be accessed in real time through the internet network. Thus, the quality and water circulation were well-preserved. Results showed that the success rate of measurement for ultrasonic sensor was 99.94%, pH sensor of 92.35%, and temperature sensor of 97.91%. The temperature and pH water pool that were suitable for aquaponics ranged between 20-30°C and 7-75 and the monitoring system proceeded as expected.

**[10] Title: Aquaponics for Agriculture using IOT || Authors:** Maryam Jawadwala Information Technology VCET Mumbai, India Yogesh Pingle Information Tehnology VCET Mumbai, India **||** 2020.

The author **Yogesh Pingle** had made research on that this project focuses on making the agriculture soilless as well as making it feasible to use anywhere. For which they will be using all the knowledge of hydroponics and aquaponics. The aim is to make an aquaponics system that will help grow fishes and plants together indoor. The sensors are used to take certain readings which will let them know whether the system is normal or not. That will go to Blynk cloud, which will keep the readings on the cloud, then send the readings to the Blynk application on the smart phone. Using the knowledge of Arduino and the cloud, they will send the data to the cloud that will display the results. Using the readings, we will be able to know the system is working properly or not. Sensors, Arduino and Cloud is used to take readings and display the readings too. This is the main paper of this research, this paper totally focuses on recent development in soilless farming, hydroponics and aquaponics making it feasible for farming at individual level or even can be used by farmers. Also, the system once automated then it will reduce most of the manual work of farming and will help in having the natural growth of plants. The system was developed taking the idea of hydroponics and aquaculture. The fish extract contains ammonia and the nutrients needed by the plants too that’s why it became helpful in developing the system combining this both. The system is built keeping in mind that the fish should be grown naturally and the plants too simultaneously. The plants extract nitrogen which is needed from the fish and then it will grow naturally without any use of artificial chemicals and fertilizers. In fact, the system eliminates the use of artificial use of chemicals and other fertilizers. Information obtained from cloud is used to give accurate readings to determine the readings that affects the system.

**CHAPTER 3**

**PROBLEM STATEMENT**

The traditional method of growing plants and raising fish in separate systems has limitations, including the need for high levels of inputs, water, energy, and maintenance. Moreover, Due to artificial use of fertilizers and chemicals the soil has lost its originality. The farmers have no idea or either forced to use this for growing crops. It has become difficult for farmers too to waive off loans with the help of crops as they are dependent on nature of soil and climatic conditions and also the current pandemic crisis has led to significant supply chain disruptions, which have made food insecurity an even more pressing issue. In this context, an IoT-based aquaponic system can be a viable solution to address these problems by combining aquaculture and hydroponics to create a sustainable and efficient food production system. However, designing and implementing such a system requires overcoming several challenges such as optimizing water quality parameters, balancing fish and plant growth, monitoring and controlling the system remotely, and ensuring data security. Therefore, the problem statement for an IoT-based aquaponic system is to develop a cost-effective, energy-efficient, and user-friendly monitoring and control system that can optimize the growth of fish and plants, reduce water and energy consumption, and ensure food safety and security.

**CHAPTER 4**

**METHODOLOGY**

**4.1 BLOCK DIAGRAM**

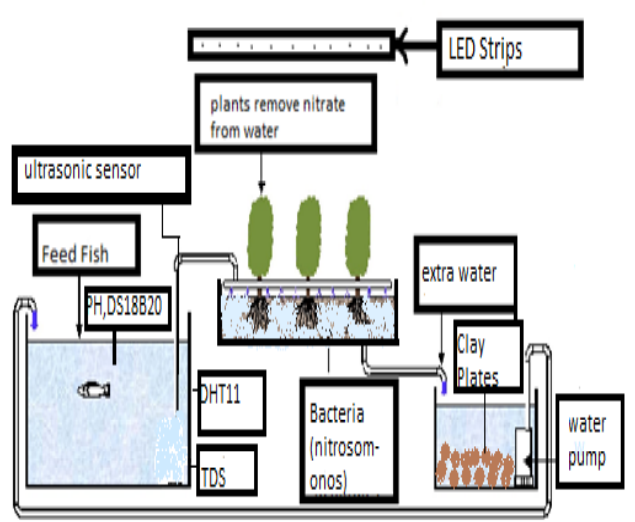
****

Figure 4.1.1: Block diagram of Aquaponic System

***Figure Labels****: Ultrasonic Sensor, Feed fish, pH, DS18b20, DHT11, TDS, plants remove nitrate from water, Bacteria (Nitrosomonas), LED Strips, Extra water, Clay Plates, Water Pump, Arduino UNO, Node MCU, BLYNK Application.*

Component used are mostly the sensors for the proper functioning of the system as if there are any inappropriate conditions arising then the sensors will always give the readings for awareness.

Aquaponics system consists of hardware components like DHT11, DS18b20, pH sensor, TDS Sensor, Ultrasonic Sensor, ESP8266 NodeMCU, 5V Relay, Water Pump, Arduino UNO, Wires, Breadboard, Resistors(4.7k), 3 Carters (Grow bed, Fish Tank, Sump Tank), Air Bubble Stone, Bacteria(Nitrosomonas), Clay Plates, LED Strips(Red, White as grow light), Grow box, Sponge, Fish (Koi, Tilapia), Fish food, Plant Seeds (Cucumber, Spinach, Tomato), Pipes, Sun mica Sheet and Software Requirements are Arduino Sketch, Blynk Application. Firstly, all 3 carters are placed one above the other in the order Sump Tank, Fish tank and Grow bed. All three carters are

connected with pipes in vertical manner. The sun mica sheet is placed above the last grow bed with holes for grow box fitting. The plant seeds are initially kept in a sponge with some water for growing into sapling. After one week of growth of sapling, it can be kept into the grow box which is placed in last grow bed. The sump tank below will have clay plates for filtering of water to some extent. The fishes in the fish tank, that is, the second tank will be fed with fish food for fish growth. Fishes will leave some extract which will settle down the tank and with help of pipes this wastewater will come into the sump tank which is having motor to circulate the water to the plants in the grow bed. The sump tank will have clay plates which will help storing some amount of water. The plants will get nutrients for its growth from fish extract water and with LED strips placed above the plants will help the plants to do photosynthesis process as well. More the water is recycled more efficient the system will be for future. The sensors are used for testing whether the system is working properly or not. The pH of the water in which fish exist can change which may be harmful for fish life. Also, the temperature of water shouldn’t exceed a certain limit which is necessary to be maintained. The ultrasonic Sensor is used to check the height of water in fish tank. Blynk app is freely available in Google Play store which is used to show sensor readings in graphical format which makes it easy to read it. Also, the TDS sensor is used to check the dissolved solvents in the fish tank and its readings is taken by the Arduino which is interfaced with Blynk App to get readings. Also, the relay is used to interface the Arduino for switching on and off automatically so that motor supplies water efficiently to the plants. DS18b20 is used to check the temperature inside the water and DHT11 is used to check the temperature outside the carter. All these sensors will be interfaced with NodeMCU to Blynk app. Also, the aquaponics system requires pH of 6.8. The air bubble stone will be used to supply oxygen for the fishes.

**4.2 DATA FLOW DIAGRAM**

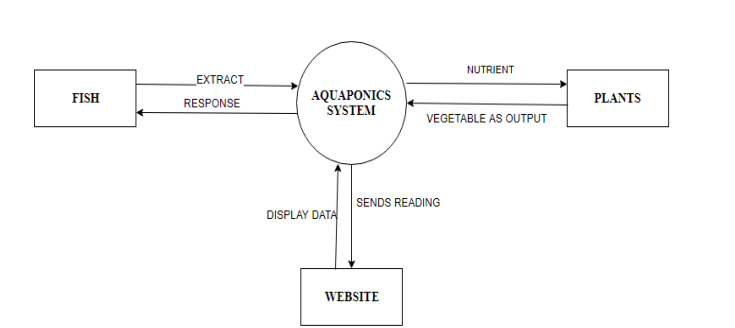
****

Figure 4.2.1: Level 0 Data flow Diagram for aquaponics system

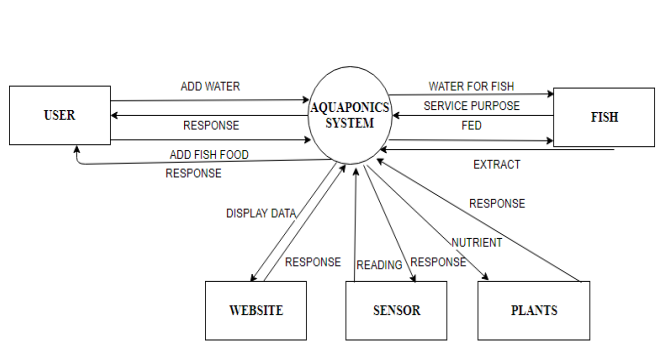


Figure 4.2.2: Level 1 Data flow diagram for aquaponics system

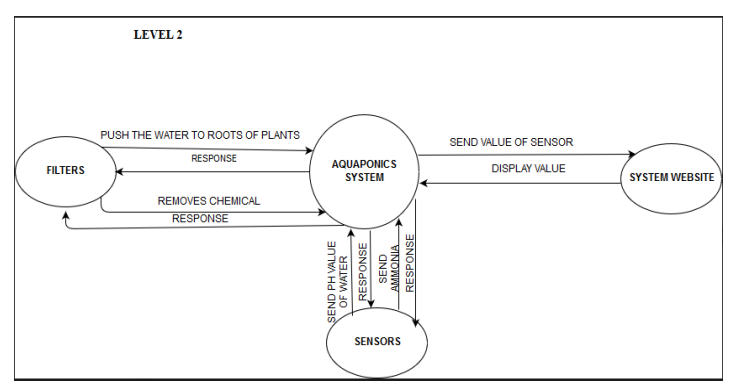


Figure 4.2.3: Level 2 Data flow Diagram for aquaponics system

A data-flow diagram (DFD) is a way of representing a flow of a data of a process or a system. The DFD also provides information about the outputs and inputs of each entity and the process itself. Level 0, level 1 and Level 2 DFD for aquaponics system is shown in figure 4.2.1, figure 4.2.2 and figure 4.2.3.

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the logical view of the system under development. Sequence diagrams are sometimes called event diagrams or event scenarios. Figure 4.2.4 shows sequence diagram for aquaponics system.

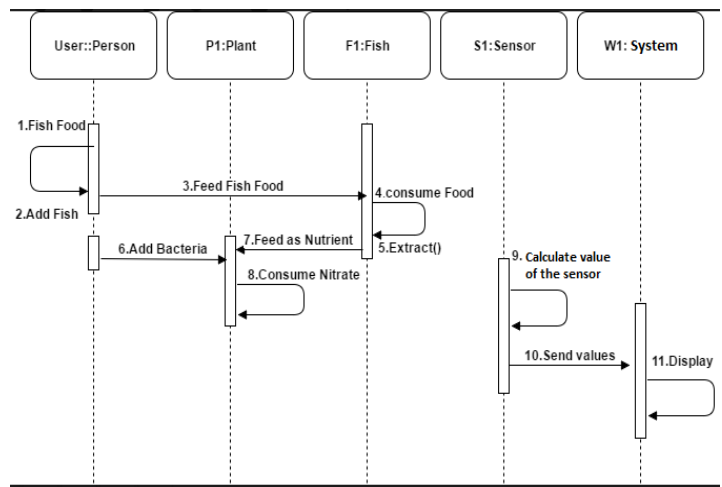


Figure 4.2.4: Sequence diagram for aquaponics system

**4.3 IMPLEMENTATION**

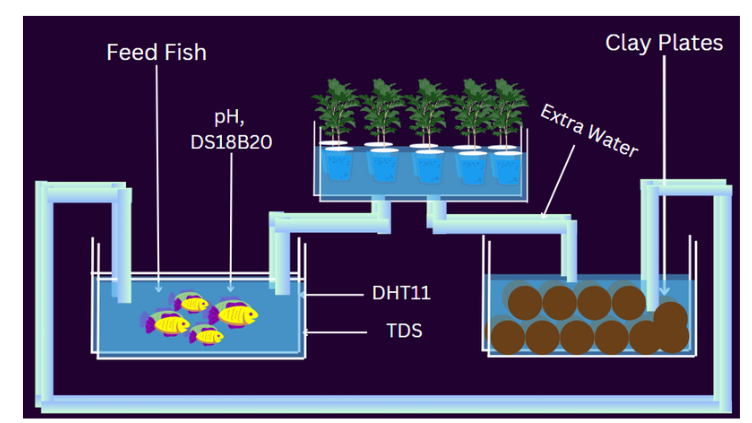
****

Figure 4.3.1: Implementation diagram for aquaponics system

The fish is fed with fish food in second carter (fish tank) to leave extract. The fish tank has sensor like TDS to check the solvents(extract) in water. Also, the carter has pH sensor to check the pH of the water. The fishes need pH between 6 to 7. Also, the fish extract will go into the sump tank with help of pipes in the fish tank. The sump tank consists of clay plates which will keep the water cool and will also act as filter. The fish extract in the sump tank will be sent to the first carter (Grow bed) with help of motor in the sump tank. The ultrasonic sensor is kept in sump tank to determine the water level and DS18b20 sensor is placed inside the tank to determine the water temperature. Also, the plants in the first carter will absorb the fish water obtained from the sump tank and will start growing. The LED light is placed above the plants as the plants need to do photosynthesis process. The DHT11 Sensor is placed outside the carter to determine surrounding temperature and humidity. This tank consists of clay plates which keeps the water cool and acts as filter. The tank consists of ultrasonic sensor to determine the water level and consists of DS18b20 sensor to determine the temperature of water. The tank consists of motor to draw water from sump tank to the grow bed. The fish tank consists of fishes which is fed with fish food to leave extract for plants. The fish tank consists of pH sensor to determine the pH level of water and consists of TDS Sensor to determine the dissolved solvent(extract). The grow bed consists of sun mica sheet with holes in which grow box is kept with clay plates. The plants are placed in the grow box and the LED light is placed above third carter which will keep in photosynthesis process. Koi fish is easily available and is cheaper than other fishes. Also, this fish is less prone to diseases compared to other fishes. The fish extract is rich in Nitrates and nitrites which is required by the plants. And the extract contains magnesium, phosphates, carbohydrates, sulphates in a very small amount which is also needed for plants. A koi’s diet should consist of six main building blocks: protein, fats, carbs, fiber, vitamins, and minerals. A young koi needs more protein than an older one. Koi will get their fats from wheat germ, fish meal, and corn oils in prepared foods. Avoid koi food made from rice, wheat, and corn they contain too many carbs. Instead, look for ones deriving carbohydrates from vegetables and fruit. Spinach plant can grow fast easily. It requires nutrients which is already available in fish extract. Spinach requires soil in the pH range of 6.2 to 6.9. Spinach plants require a fertilizer that is rich in nitrogen when they are about halfway through the growing cycle. Hence instead of using artificial fertilizer we can use fish extract which is natural fertilizer. The carters are placed one above the other and are having holes in first two carters which is fitted with pipes to height of carter. The last tank is not having any hole and is filled with water till half. It contains clay plates to keep water cool and to also filter water. And the motor is also placed in the tank to draw the water from sump tank to the first tank which is the grow bed. The sump tank consists of the ultrasonic sensor to determine the height of water level. The DS18b20 sensor is also placed in sump tank to determine the water temperature. The second tank is filled with water for fishes and the fish is given fish food to leave extract. The TDS sensor is placed in second tank to determine the dissolved soluble like extract of fish and contains the pH sensor to determine the pH of the fish tank. Also, DHT11 Sensor is placed outside the carters to determine surrounding temperature and humidity. The sensors are controlled with the help of microcontroller. The sensors like DHT11, DS18b20 and Ultrasonic are controlled with help of NodeMCU and sensors like TDS and pH is controlled with help of Arduino UNO microcontroller. The first carter contain pipe from sump tank for the supply of water. The carter is placed with sun mica sheet which is used to keep grow box by making holes. The grow box contains some clay plates and plants placed between them. The bacteria named Nitrosomonas is being sprayed for two to three times in a week for its proper growth. The fish water contains ammonia which needs to be broken down into smaller and simpler molecules by plants and Nitrosomonas help in that process. Figure 4.3.1 shows implementation of aquaponics system.

**CHAPTER 5**

**HARDWARE REQUIREMENTS**

**5.1 HARDWARE COMPONENTS**

1. **DHT11**

It's a digital sensor which is used for measuring humidity and temperature. It uses combination of capacitive humidity sensor and thermistor. For this sensor no analog pin is needed.

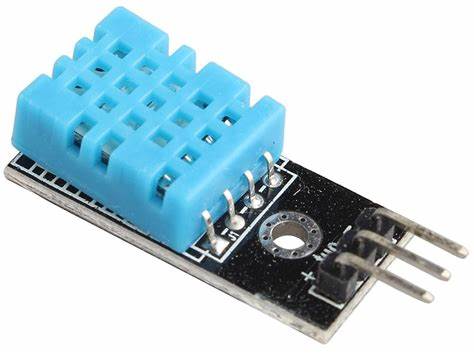


Figure 5.1.1: DHT11 Sensor

1. **DS18b20**

This is the temperature sensor which can measure the temperature between -50 to 125C. The main advantage of this sensor is it can measure minimum amount of temperature to maximum.



Figure 5.1.2: DS18B20 Sensor

1. **pH Sensor**

pH stands for potential of hydrogen it refers to the concentration of hydrogen ion in any solution. Which measure the acidity and alkalinity of a solution in the pH value ranges from scale 0 to 14.



Figure 5.1.3: pH Sensor

1. **TDS Sensor**

TDS refers to total dissolved solvent. This sensor is used to indicate the total dissolved solids in solution.



Figure 5.1.4: TDS Sensor

1. **Moisture Sensor**

Measures relative humidity from 0% humidity to 100% humidity.

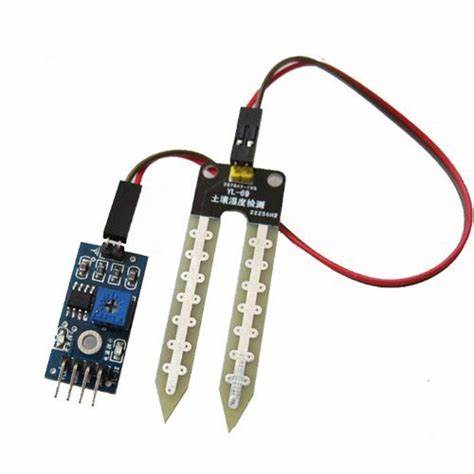


Figure 5.1.5: Moisture Sensor

1. **Rain Sensor**

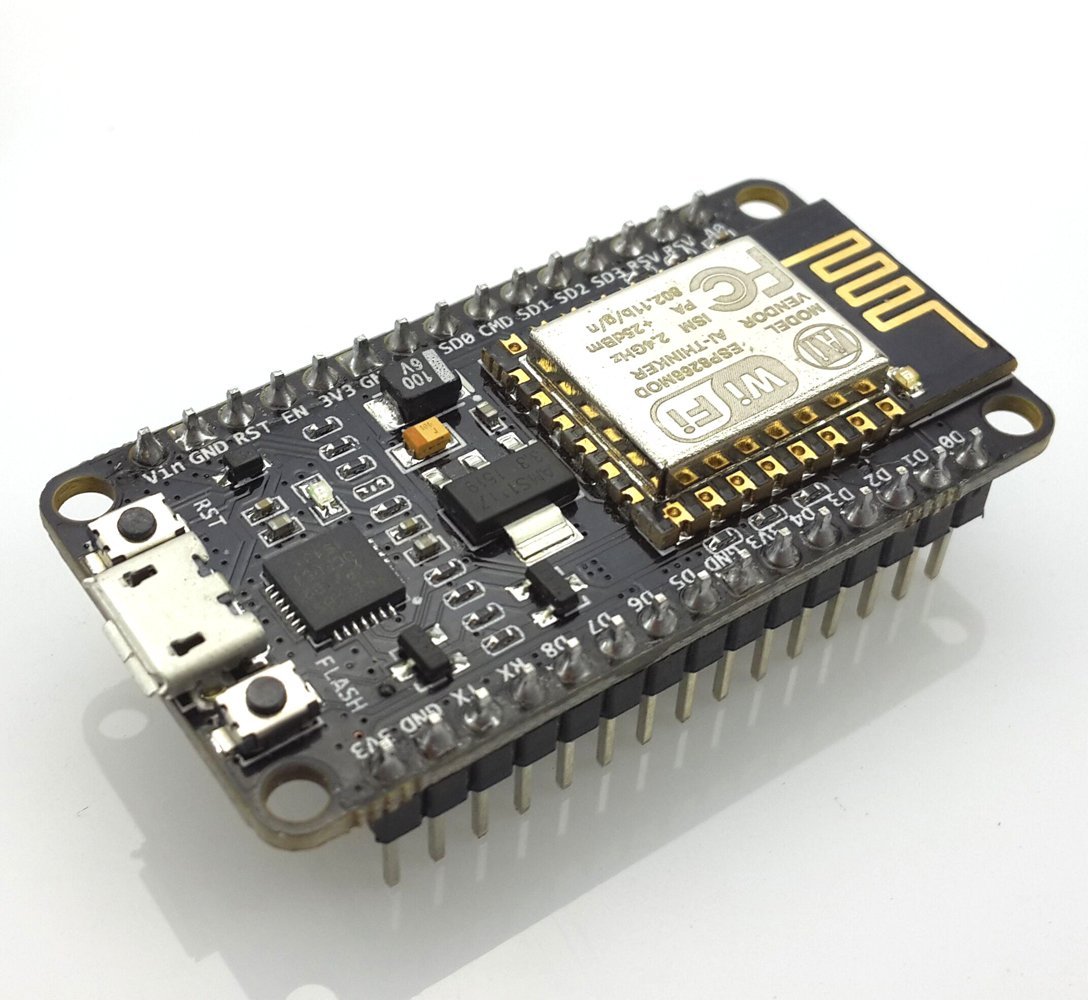
A rain sensor is activated by rainfall.



Figure 5.1.6: Rain Sensor

1. **ESP8266 NodeMCU**

Open-source LUA based firmware developed for the ESP8266 Wi-Fi chip



5.1.7 ESP8266 NODE MCU

**CHAPTER 6**

**SOFTWARE REQUIREMENTS**

**ARDUINO SOFTWARE IDE**

**6.1 Arduino IDE Appearance**

The Arduino IDE is an open-source software, which is used to write and upload code to the Arduino boards. The IDE application is suitable for different operating systems such as **Windows, Mac OS X, and Linux**. It supports the programming languages C and C++. Here, IDE stands for **Integrated Development Environment**.

The program or code written in the Arduino IDE is often called as sketching. We need to connect the Genuine and Arduino board with the IDE to upload the sketch written in the Arduino IDE software. The sketch is saved with the extension '.ino.'

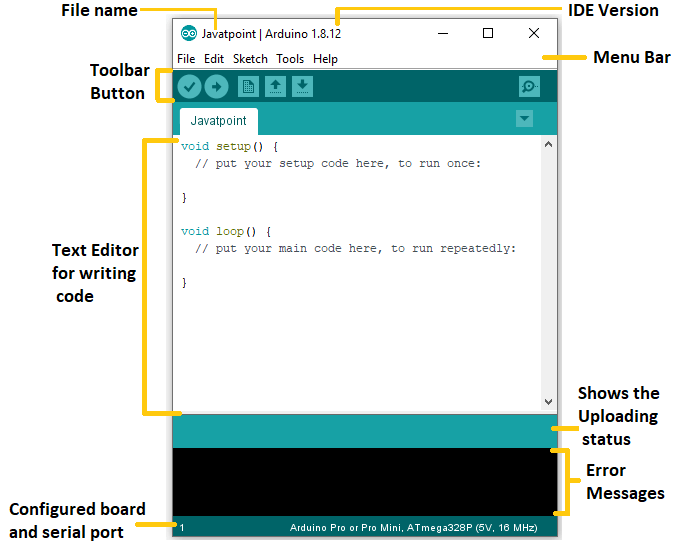


Figure 6.1.1: Arduino IDE

**6.2 Toolbar Button**

The icons displayed on the toolbar are New, Open, Save, Upload, and Verify.

It is shown below:

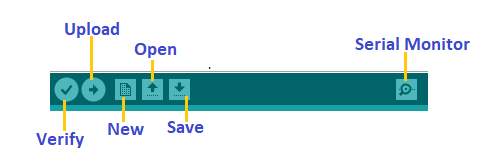


Figure 6.2.1: Arduino IDE Toolbar

### **Upload**

The Upload button compiles and runs our code written on the screen. It further uploads the code to the connected board. Before uploading the sketch, we need to make sure that the correct board and ports are selected.

We also need a USB connection to connect the board and the computer. Once all the above measures are done, click on the Upload button present on the toolbar.

### **Open**

The Open button is used to open the already created file. The selected file will be opened in the current window.

### **Save**

The save button is used to save the current sketch or code.

### **New**

It is used to create a new sketch or opens a new window.

### **Verify**

The Verify button is used to check the compilation error of the sketch or the written code.

### **Serial Monitor**

The serial monitor button is present on the right corner of the toolbar. It opens the serial monitor.

## 6.3 Menu Bar

**File**

When we click on the File button on the Menu bar, a drop-down list will appear. It is shown below:

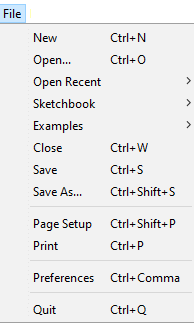


Figure 6.3.1: Arduino IDE File

**New**

The New button opens the new window. It does not remove the sketch which is already present.

**Open**

It allows opening the sketch, which can be browsed from the folders and computer drivers.

**Open Recent**

The Open Recent button contains the list of the recent sketches.

**Sketchbook**

It stores the current sketches created in the Arduino IDE software. It opens the selected sketch or code in a new editor at an instance.

**Close**

The Close button closes the window from which the button is clicked.

**Save**

The save button is used to save the current sketch. It also saves the changes made to the current sketch. If we have not specified the name of the file, it will open the '**Save As...'** window.

**6.4 Edit**

When we click on the Edit button on the Menu bar, a drop-down list appears. It is shown below:

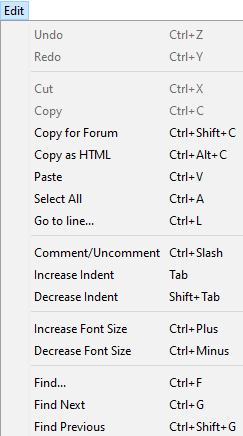


Figure 6.4.1: Arduino IDE Edit

**Undo**

The Undo button is used to reverse the last modification done to the sketch while editing.

**Redo**

The Redo button is used to repeat the last modification done to the sketch while editing.

**Cut**

It allows us to remove the selected text from the written code. The text is further placed to the clipboard. We can also paste that text anywhere in our sketch.

**Copy**

It creates a duplicate copy of the selected text. The text is further placed on the clipboard.

**Copy for Forum**

The 'Copy for Forum' button is used to copy the selected text to the clipboard, which is also suitable for posting to the forum.

**Copy as HTML**

The 'Copy for Forum' button is used to copy the selected text as HTML to the clipboard. It is desirable for embedding in web pages.

**Paste**

The Paste button is used to paste the selected text of the clipboard to the specified position of the cursor.

**Select All**

It selects all the text of the sketch.

**Go to line...**

It moves the cursor to the specified line number.

The window will appear as:

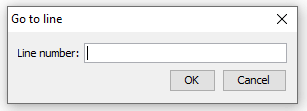


Figure 6.4.2: Arduino IDE Linebar

**Comment/De-comment**

The Comment/ De-comment button is used to put or remove the comment mark (**//**) at the beginning of the specified line.

**Increase Indent**

It is used to add the space at the starting of the specified line. The spacing moves the text towards the right.

**Decrease Indent**

It is used to subtract or remove the space at the starting of the specified line. The spacing moves the text towards the left.

**Increase Font Size**

It increases the font size of the written text.

**Decrease Font Size**

It decreases the font size of the written text.

**Find...**

It is used to find the specified text. We can also replace the text. It highlights the text in the sketch.

The window will appear as:

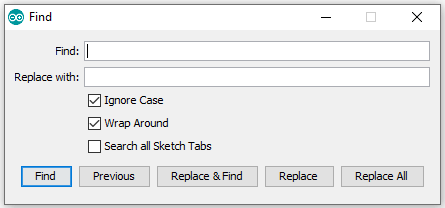


Figure 6.4.3: Arduino IDE Findbar

**Find Next**

It highlights the next word, which has specified in the '**Find...'** window. If there is no such word, it will not show any highlighted text.

**Find Previous**

It highlights the previous word, which has specified in the '**Find...'** window. If there is no such word, it will not show any highlighted text.

**6.5 Sketch**

When we click on the Sketch button on the Menu bar, a drop-down list appears. It is shown below:

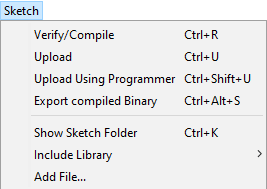


Figure 6.5.1: Arduino IDE Sketch

**Verify/Compile**

It will check for the errors in the code while compiling. The memory in the console area is also reported by the IDE.

**Upload**

The Upload button is used to configure the code to the specified board through the port.

**Upload Using Programmer**

It is used to override the Bootloader that is present on the board. We can utilize the full capacity of the Flash memory using the '**Upload Using Programmer**' option. To implement this, we need to restore the Bootloader using the **Tools**-> **Burn Bootloader** option to upload it to the USB serial port.

**Export compiled Binary**

It allows saving a .**hex** file and can be kept archived. Using other tools, .hex file can also be sent to the board.

**Show Sketch Folder**

It opens the folder of the current code written or sketch.

**Include Library**

Include Library includes various Arduino libraries. The libraries are inserted into our code at the beginning of the code starting with the #. We can also import the libraries from .zip file.

**Add File...**

The Add File... button is used to add the created file in a new tab on the existing file.

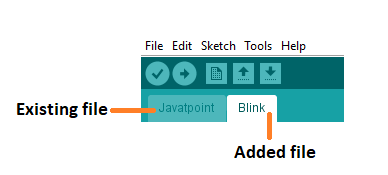


Figure 6.5.2: Arduino IDE Add file

We can also delete the corresponding file from the tab by clicking on the **small triangle** -> **Delete** option.

### **6.6 Tools**

When we click on the Tools button on the Menu bar, a drop-down list appears. It is shown below:

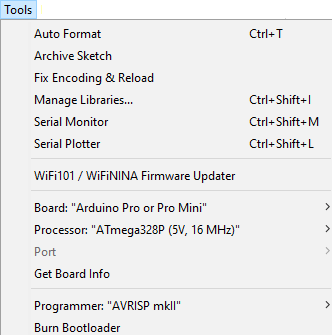


Figure 6.6.1: Arduino IDE Tools

**Auto Format**

The Auto Format button is used to format the written code. For example, lining the open and closed curly brackets in the code.

**Archive Sketch**

The copy of the current sketch or code is archived in the .zip format. The directory of the archived is same as the sketch.

**Fix Encoding and Reload**

This button is used to fix the inconsistency between the operating system char maps and editor char map encoding.

**Manage Libraries...**

It shows the updated list of all the installed libraries. We can also use this option to install a new library into the Arduino IDE.

**Serial Monitor**

It allows the exchange of data with the connected board on the port.

**Serial Plotter**

The Serial Plotter button is used to display the serial data in a plot. It comes preinstalled in the Arduino IDE.

**WiFi101/WiFiNINA Firmware Updater**

It is used to check and update the Wi-Fi Firmware of the connected board.

**Board**

We are required to select the board from the list of boards. The selected board must be similar to the board connected to the computer.

**Processor**

It displays the processor according to the selected board. It refreshes every time during the selection of the board.

**Port**

It consists of the virtual and real serial devices present on our machine.

**Get Board Info**

It gives the information about the selected board. We need to select the appropriate port before getting information about the board.

**Programmer**

We need to select the hardware programmer while programming the board. It is required when we are not using the onboard USB serial connection. It is also required during the burning of the Bootloader.

**Burn Bootloader**

The Bootloader is present on the board onto the microcontroller. The option is useful when we have purchased the microcontroller without the bootloader. Before burning the bootloader, we need to make sure about the correct selected board and port.

### **6.7 Help**

When we click on the Help button on the Menu bar, a drop-down list will appear. It is shown below:

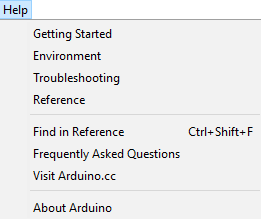


Figure 6.7.1: Arduino IDE Help

The Help section includes several documents that are easy to access, which comes along with the Arduino IDE. It consists of the number of options such as Getting Started, Environment, Troubleshooting, Reference, etc. We can also consider the image shown above, which includes all the options under the Help section.

**CHAPTER 7**

**RESULTS AND DISCUSSION**

**7.1 RAIN SENSOR**

int rainPin = D1; // Pin for rain sensor

int rainValue = 0; // Variable to store rain value

if (rainValue < 1) {

Blynk.virtualWrite(V5, "No Rain");

} else {

Blynk.virtualWrite(V5, "It’s Raining");

}

**Explanation**

* Rain Sensor pin is connected to D1 Digital pin of NodeMCU ESP8266.
* The Blynk virtual pin value V5 is assigned to display the rain value on Blynk App.
* If it is raining then it displays value as “It’s Raining”
* If it is not raining then it displays value as “No rain”

**7.1.1 Serial Monitor Output**

**It’s Raining**

**No Rain**

Figure 7.1.1: Serial Monitor output of Rain Sensor

**7.1.2 Blynk App Output**



Figure 7.1.2: Blynk App output of Rain Sensor

* 1. **pH sensor Output**

float pHvalue = analogRead(A0);

pHvalue = 14.0 - (pHvalue / 1024.0 \* 14.0);

Serial.print("pH value: ");

Serial.println(pHvalue);

Blynk.virtualWrite(V3, pHvalue);

**Explanation**

* pH Sensor pin is connected to A0 Digital pin of NodeMCU ESP8266.
* The Blynk virtual pin value V3 is assigned to display the pH value on Blynk App.
* pH Value Ranging from 0 to 14 is Displayed on Blynk App

**7.2.1 Serial Monitor Output**

**pH value is 4.895**

Figure 7.2.1: Serial Monitor output of pH sensor

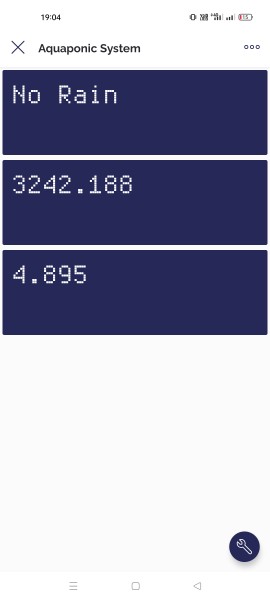


Figure 7.2.2: Blynk App Output of pH Sensor

**7.3 Moisture Sensor Output**

int moisturePin = A0; // Pin for moisture sensor

int moistureValue = 0; // Variable to store moisture value

Blynk.virtualWrite(V15, moistureValue);

Serial.println("Moisture Value is");

 Serial.println(moistureValue);

**Explanation**

* Moisture Sensor pin is connected to A0 Digital pin of NodeMCU ESP8266.
* The Blynk virtual pin value V15 is assigned to display the Moisture value on Blynk App.
* Moisture Value Ranging from 0 to 1023 is Displayed on Blynk App

**7.3.1 Serial Monitor Output of Moisture Sensor**

**Moisture value is 651**

Figure 7.3.1: Serial Monitor output of Moisture sensor

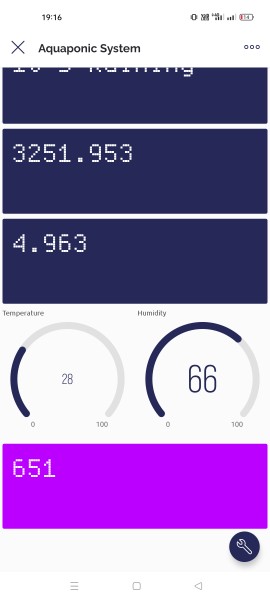


Figure 7.3.2: Blynk App output of Moisture sensor

**7.4 TDS Sensor Output**

int tdsPin = A0;

tdsValue = analogRead(tdsPin); // Read TDS sensor data

float tdsPPM = (tdsValue / 1024.0) \* 5000.0; // Convert TDS sensor data to PPM

Blynk.virtualWrite(V4,  tdsPPM);

Serial.println("TDS Value is");

Serial.println(tdsPPM);

**Explanation**

* TDS Sensor pin is connected to A0 Digital pin of NodeMCU ESP8266.
* The Blynk virtual pin value V4 is assigned to display the TDS value on Blynk App.
* TDS 0 to 1000ppm Value Ranging from 0 to 1023 is Displayed on Blynk App

**7.4.1 Serial Monitor Output of TDS Sensor**

**TDS value is 321**

Figure 7.4.1: Serial Monitor output of TDS sensor

**7.5 DHT11 Sensor Output**

#define DHTPIN D6

Blynk.virtualWrite(V1, temperature);

Serial.print("Temperature: ");

Serial.println(temperature); // Send temperature data to Blynk virtual pin V1

Blynk.virtualWrite(V2, humidity);

Serial.print("Humidity ");

Serial.println(humidity);

**Explanation**

* DHT11 Sensor pin is connected to D6 Digital pin of NodeMCU ESP8266.
* The Blynk virtual pin value V1 and V2 are assigned to display the Temperature and Humidity value on Blynk App.

**7.5.1 Serial Monitor Output of DHT11 Sensor**

**Temperature: 28**

**Humidity: 68**

Figure 7.5.1: Serial Monitor output of Moisture sensor



Figure 7.5.2: Blynk App output of DHT11 sensor

**7.6 Output Results**

**7.6.1 Rain Sensor**

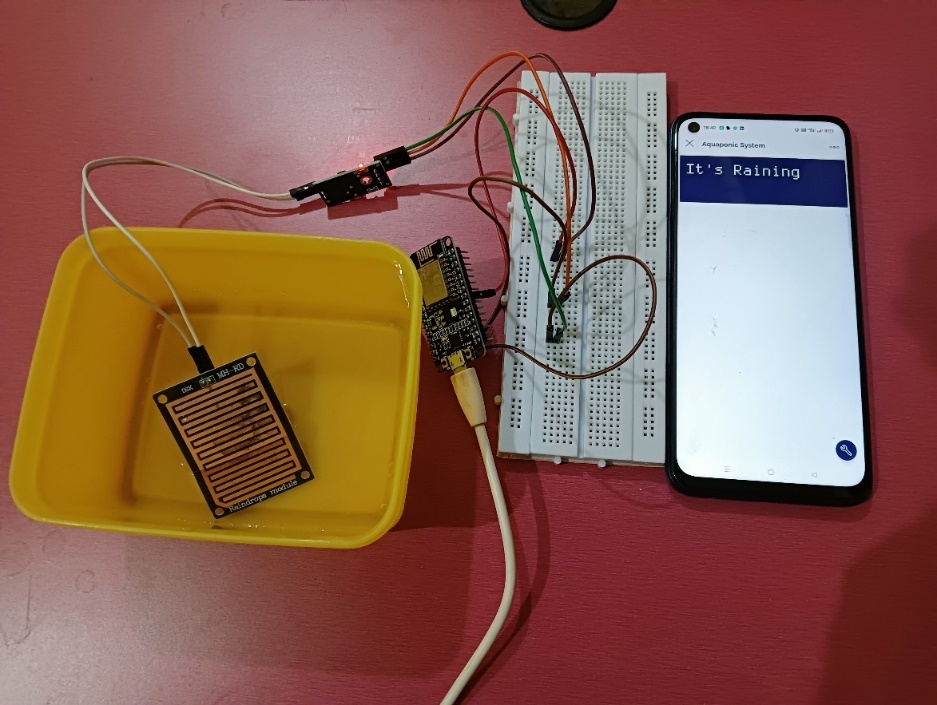


Figure 7.6.1: Rain Sensor Results

**7.6.2 TDS Sensor**

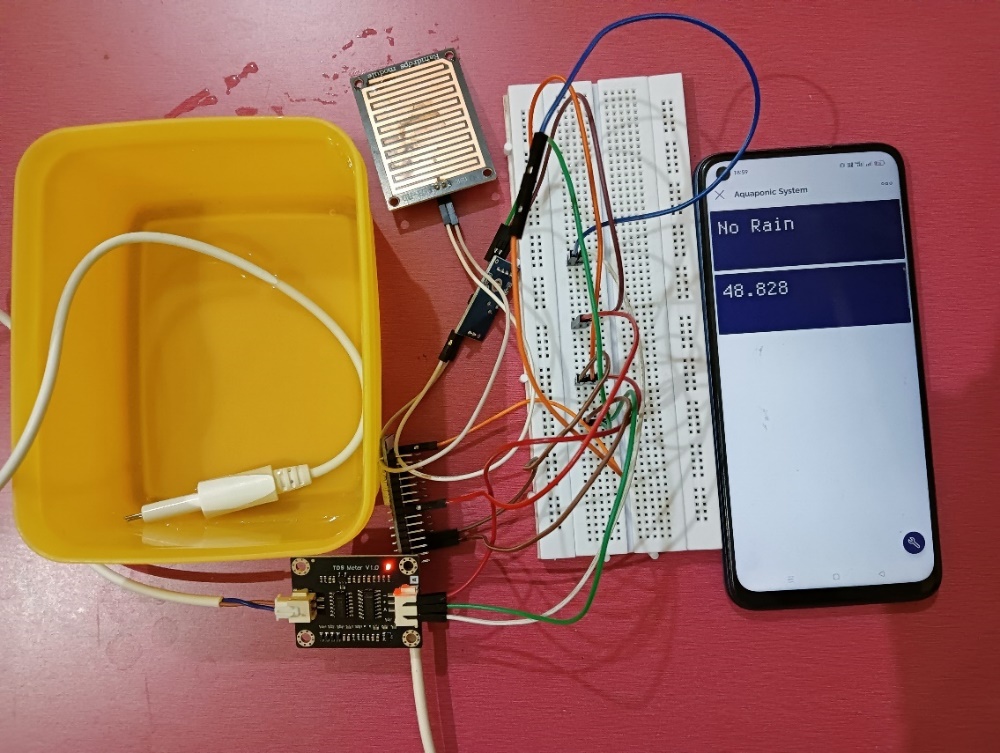


Figure 7.6.2: Rain Sensor and TDS Sensor Results

**7.6.3 pH Sensor**

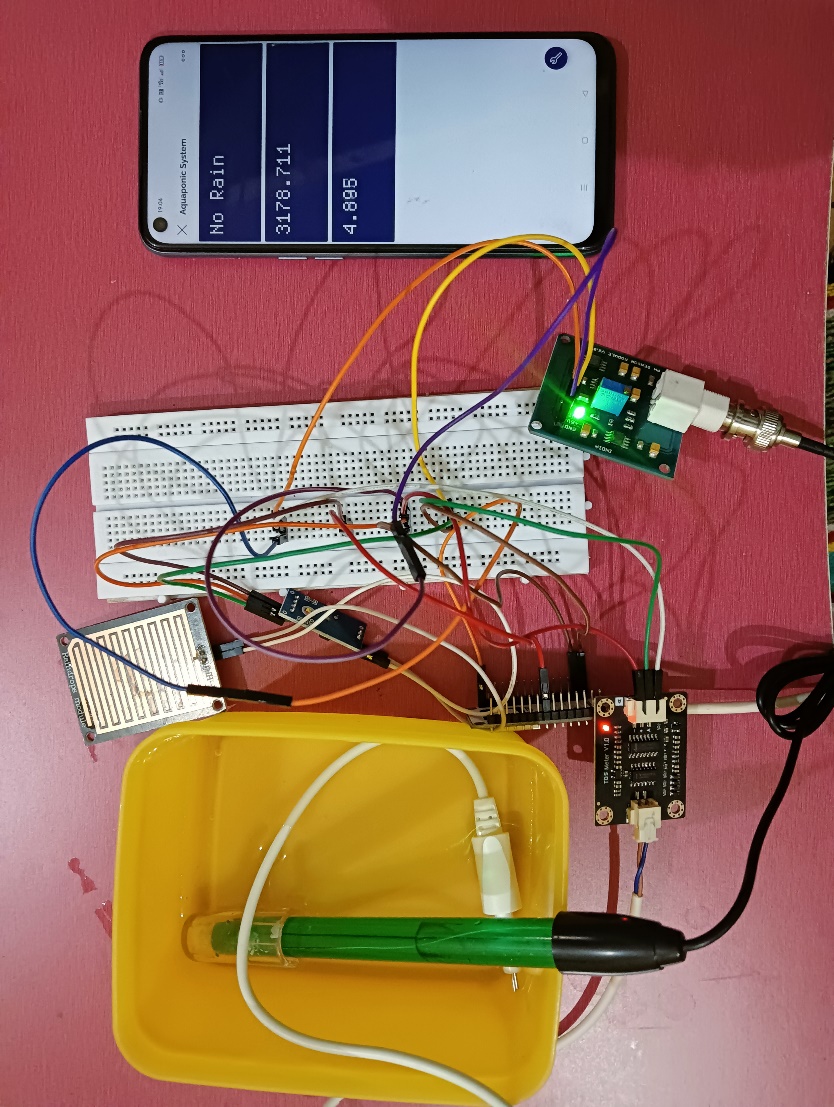


Figure 7.6.3: Rain Sensor, TDS Sensor, pH sensor Results

**7.6.4 DHT11 Sensor**

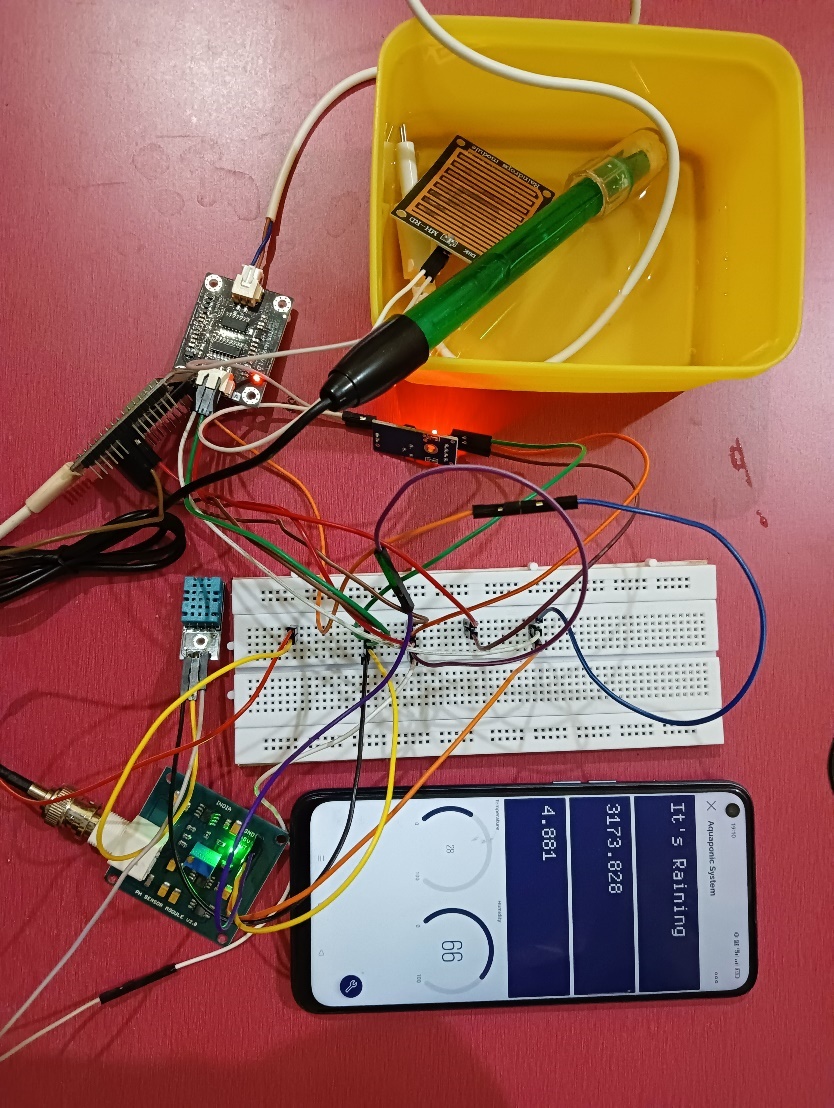


Figure 7.6.4: Rain Sensor, TDS Sensor, pH sensor, DHT11 Results

**7.6.5 MOISTURE Sensor**

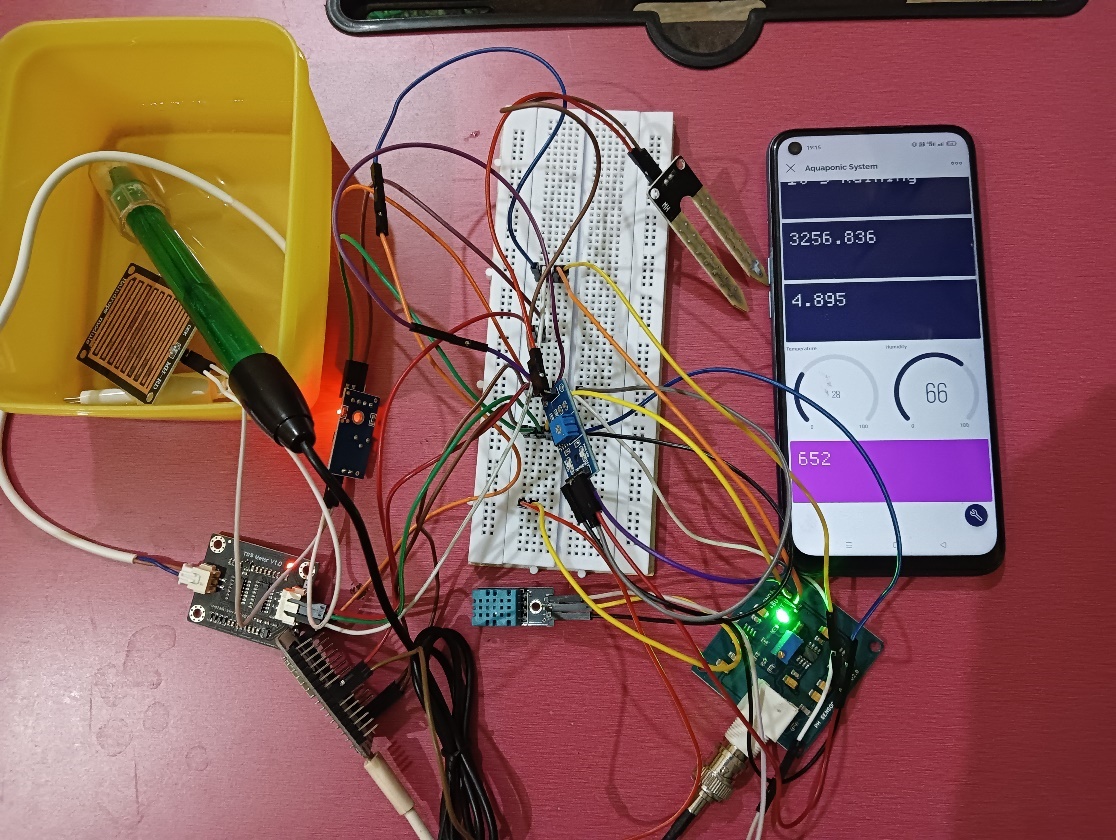


Figure 7.6.5: Rain Sensor, TDS Sensor, pH sensor, DHT1, Moisture Sensor Results

**7.7 Snapshots**

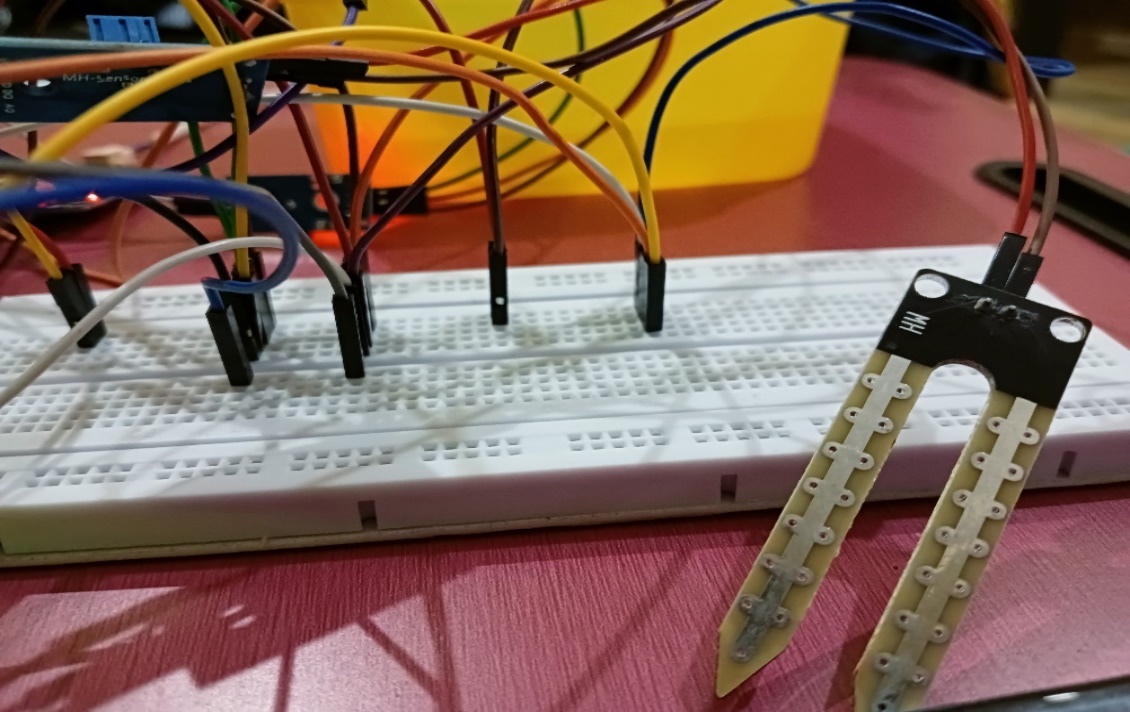


Figure 7.7.1: Project Demonstration Snapshot 1

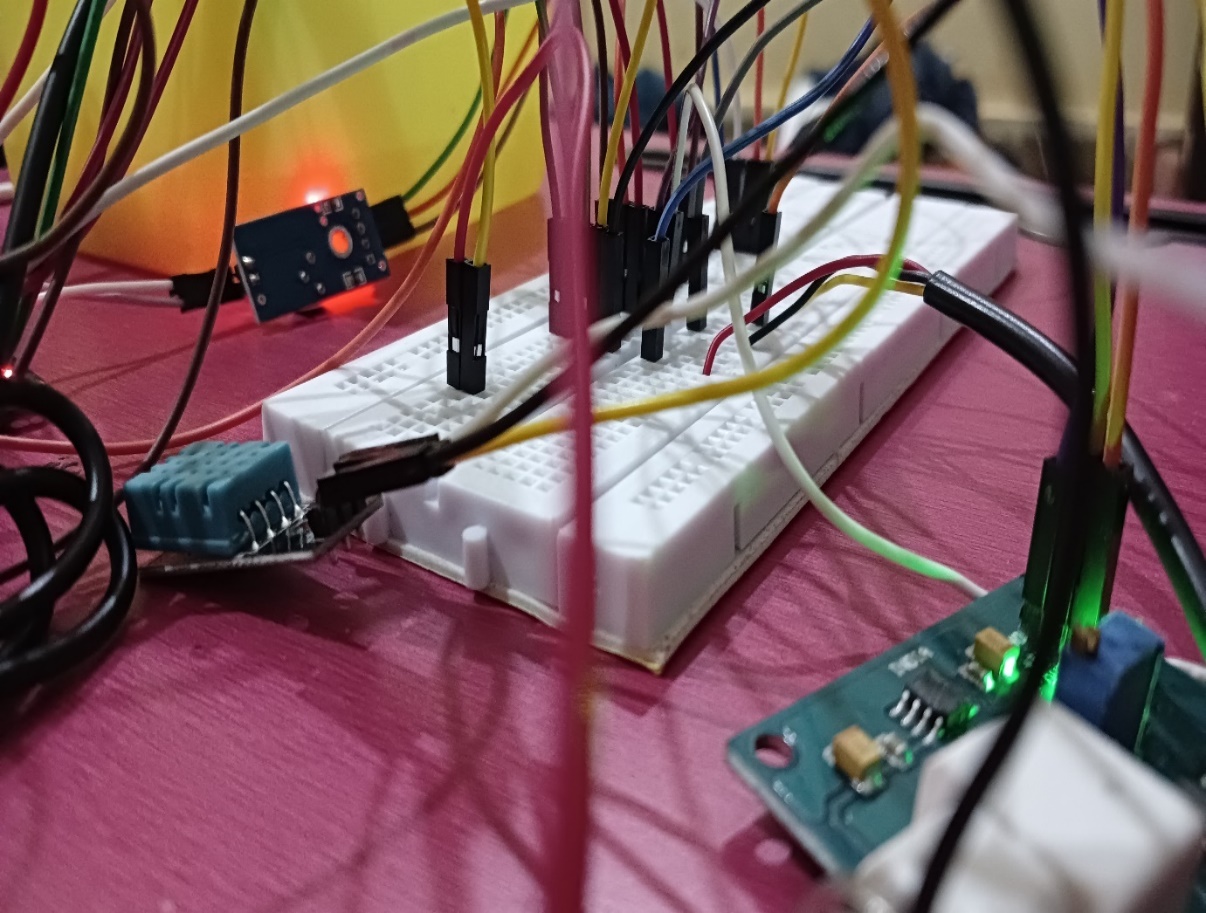


Figure 7.7.2: Project Demonstration Snapshot 2

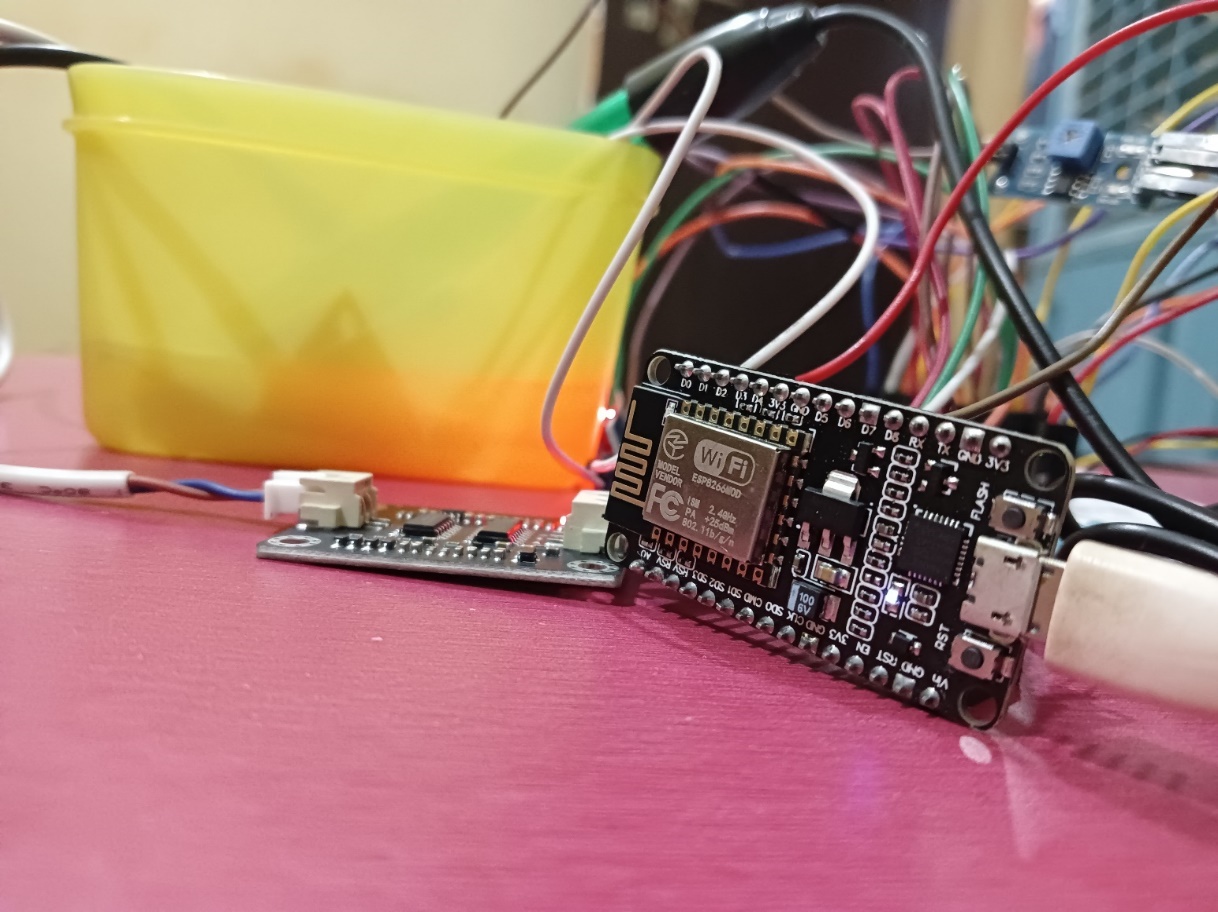


Figure 7.7.3: Project Demonstration Snapshot 3



Figure 7.7.4: Project Demonstration Snapshot 4

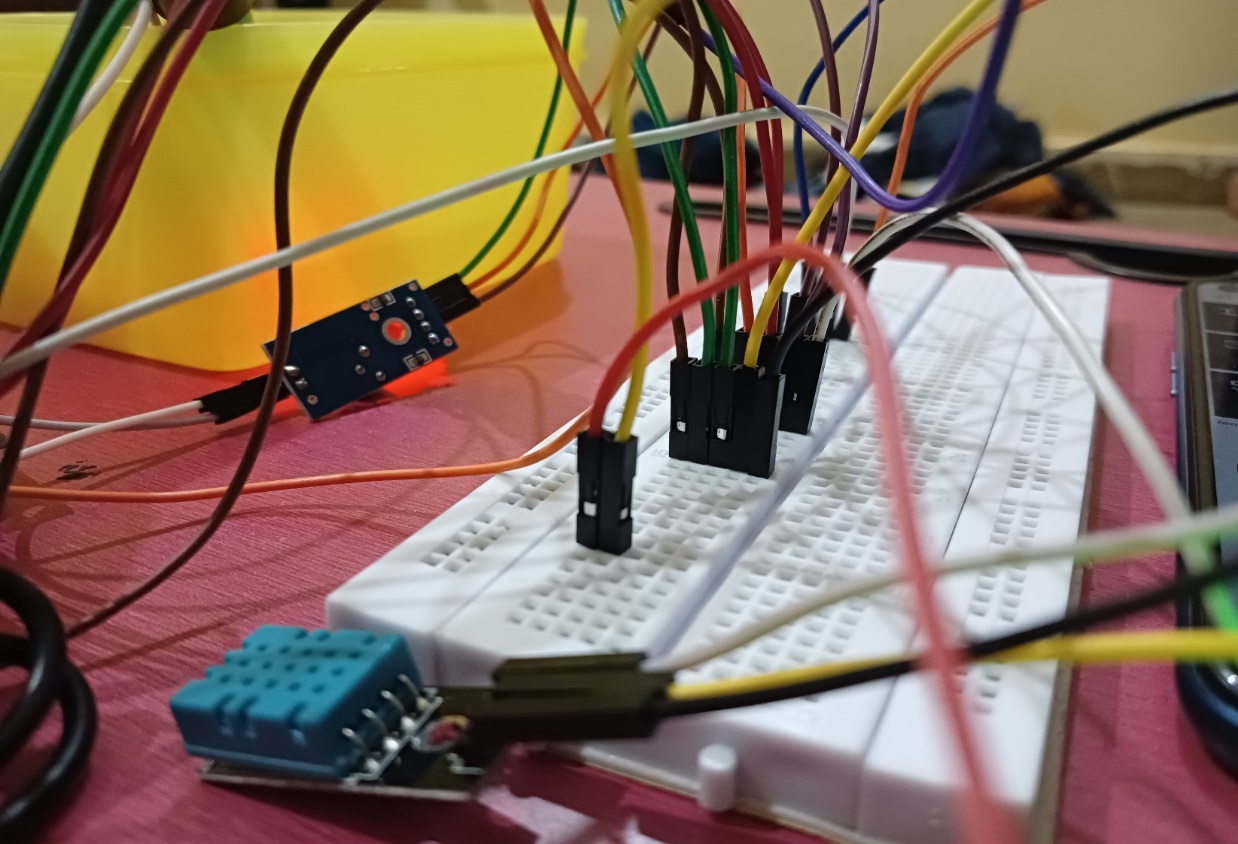


Figure 7.7.5: Project Demonstration Snapshot 5



Figure 7.7.6: Aquaponic Plants Snapshot 6



Figure 7.7.7: Aquaponic Plants Snapshot 7

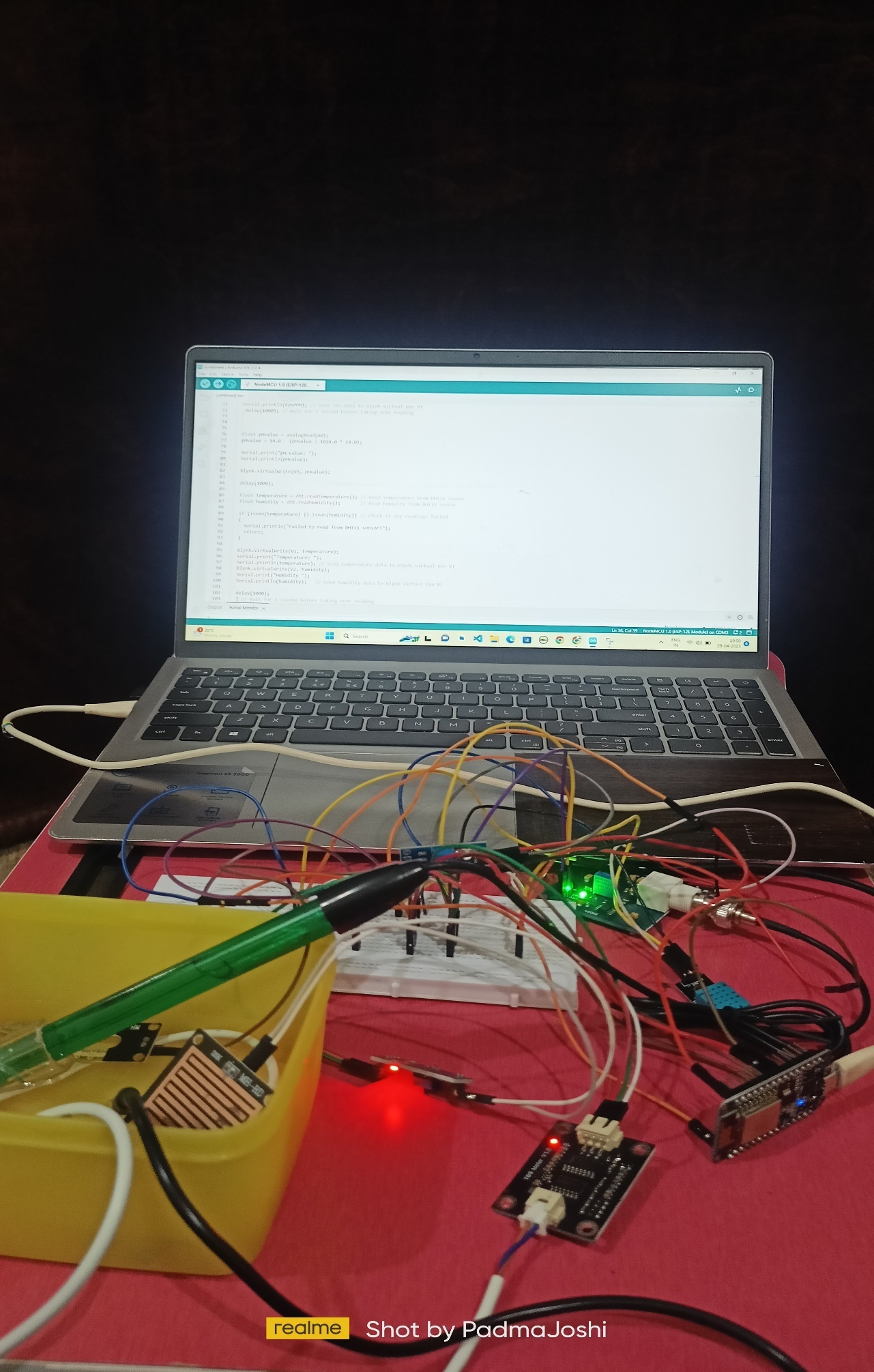


Figure 7.7.8: Aquaponic Plants Snapshot 8

**CHAPTER 8**

**ADVANTAGES AND DISADVANTAGES**

**8.1 ADVANTAGES**

* Produces fresh and organic fish and vegetables with aquaponics
* Plants grow faster and taste better with aquaponics
* Aquaponic systems are easy to build and cheap to run
* It requires 6x less space than traditional farming
* It require 90% less water than classical farming
* Aquaponic systems are easily expandable for commercial purposes
* It’s sustainable and eco-friendly way of food production
* Aquaponics is an efficient way to produce out of season products
* It’s employing the whole family in sustainable farming

**8.2 DISADVANTAGES**

* Aquaponics is Not Suitable for All Crops.
* Cost of Electricity.
* Set Up Costs.
* Expensive to Upscale.
* Needs Technical Knowledge.
* Not A Lot of Fish.
* One Part Fails, It All Fails

**CHAPTER 9**

**FUTURE SCOPE**

The future scope of IoT-based aquaponic systems is vast and holds great potential in various areas, including:

**1. Integration with AI and machine learning:** IoT-based aquaponic systems can be integrated with AI and machine learning algorithms to automate the system's operation, optimize yield, and reduce waste. For example, AI algorithms can analyze the system's data and adjust the feeding, lighting, and water parameters to optimize plant and fish growth.

**2. Scalability:** IoT-based aquaponic systems can be scaled up or down to suit different production needs. The integration of IoT sensors and control systems can help to increase the efficiency of larger-scale systems, making them more profitable and sustainable.

**3. Optimization of resource usage**: IoT sensors can help to optimize resource usage in the aquaponic system, including water, energy, and nutrients. This can help to reduce costs and improve the sustainability of the system.

**4. Integration with blockchain technology:** IoT-based aquaponic systems can be integrated with blockchain technology to create a transparent and traceable supply chain. This can help to improve the accountability of food producers and increase consumer trust in the food system.

**5. Remote access and monitoring:** IoT-based aquaponic systems can be accessed and monitored remotely, allowing farmers to manage the system from anywhere in the world. This can help to reduce labor costs and improve the efficiency of the system.

**6. Integration with urban farming:** IoT-based aquaponic systems can be integrated with urban farming to create a sustainable food production system in cities. This can help to reduce the carbon footprint of food production and improve access to fresh, locally-grown produce.

Overall, the future scope of IoT-based aquaponic systems is promising, and advancements in technology will continue to improve the efficiency and sustainability of aquaponics.

**CHAPTER 10**

**CONCLUSION**

In conclusion, an IoT-based aquaponic system is a promising technology that can help to increase the efficiency, sustainability, and scalability of aquaponics. The integration of IoT sensors and control systems allows for real-time monitoring and adjustment of water parameters, feeding, lighting, and energy usage. This can help to optimize plant and fish growth, reduce waste, and improve the overall sustainability of food production.

In the future, advancements in AI, machine learning, blockchain technology, and urban farming are expected to further improve the efficiency and sustainability of IoT-based aquaponic systems. As the world population continues to grow, the demand for sustainable food production will increase, and aquaponics has the potential to play an important role in meeting this demand.

Overall, an IoT-based aquaponic system is a promising technology that can help to create a more sustainable and efficient food production system for the future.

**SOURCE CODE**

#include <dummy.h>

#define BLYNK\_TEMPLATE\_ID "TMPL3TantPQZ\_"

#define BLYNK\_TEMPLATE\_NAME "Project"

#define BLYNK\_AUTH\_TOKEN "wM4CNG7M0kpGkBeTdeK9uGXbZ-Y3AMMV"

#include <ESP8266WiFi.h>

#include <BlynkSimpleEsp8266.h>

#include <DHT.h>

int rainPin = D5; // Pin for rain sensor

int rainValue = 0; // Variable to store rain value

int moisturePin = A0; // Pin for moisture sensor

int moistureValue = 0; // Variable to store moisture value

int tdsPin = A0; // Analog pin to read TDS sensor data

float tdsValue = 0; // Variable to store TDS sensor data

#define DHTPIN D6          // DHT11 sensor pin

#define DHTTYPE DHT11      // DHT11 sensor type

DHT dht(DHTPIN, DHTTYPE);

char auth[] = "wM4CNG7M0kpGkBeTdeK9uGXbZ-Y3AMMV"; // Put your Blynk auth token here

char ssid[] = "realme 7"; // Put your WiFi SSID here

char pass[] = "93805450"; // Put your WiFi password here

BlynkTimer timer;

void sendSensorData()

{

  moistureValue = analogRead(moisturePin); // Read moisture value from sensor

  Blynk.virtualWrite(V15, moistureValue);

  Serial.println("Moisture Value is");

  Serial.println(moistureValue);

rainValue = analogRead(rainPin); // Read rain value from sensor

  if (rainValue > 1) {

    Blynk.virtualWrite(V6, "No Rain");

  }else {

    Blynk.virtualWrite(V6, "It's Raining");

  }

void setup()

{ Serial.begin(9600);

  pinMode(moisturePin, INPUT); // Set moisture pin as input

  pinMode(rainPin, INPUT); // Set rain pin as input

  Blynk.begin(auth, ssid, pass);

   timer.setInterval(5000L, sendSensorData); // Send sensor data every 5 seconds

  dht.begin();

  timer.setInterval(5000L, sendSensorData); // Send sensor data every 5 seconds

}

void loop()

{

  Blynk.run();

  timer.run();

tdsValue = analogRead(tdsPin); // Read TDS sensor data

  float tdsPPM = (tdsValue / 1024.0) \* 5000.0; // Convert TDS sensor data to PPM

  Blynk.virtualWrite(V4,  tdsPPM);

  Serial.println("TDS Value is");

  Serial.println(tdsPPM); // Send TDS data to Blynk virtual pin V1

   delay(1000); // Wait for 1 second before taking next reading

  float pHvalue = analogRead(A0);

  pHvalue = 14.0 - (pHvalue / 1024.0 \* 14.0);

  Serial.print("pH value: ");

  Serial.println(pHvalue);

  Blynk.virtualWrite(V3, pHvalue);

  delay(1000);

  float temperature = dht.readTemperature(); // Read temperature from DHT11 sensor

  float humidity = dht.readHumidity();       // Read humidity from DHT11 sensor

  if (isnan(temperature) || isnan(humidity)) // Check if any readings failed

  {

    Serial.println("Failed to read from DHT11 sensor!");

    return;

  }

  Blynk.virtualWrite(V1, temperature)

Serial.print("Temperature: ");

  Serial.println(temperature); // Send temperature data to Blynk virtual pin V1

  Blynk.virtualWrite(V2, humidity);

  Serial.print("Humidity ");

  Serial.println(humidity);   // Send humidity data to Blynk virtual pin V2

  delay(1000);

  }

**REFERENCES**

1] Wanda Vernandhes, N.S Salahuddin, **Smart Aquaponic with Monitoring and Control System Based On IOT**, In IEEE, 2017.

[2] Analene Montesines Nagayo, Caesar Mendoza, Rodrigo S. Jamisola, **An Automated Solar-Powered Aquaponics System towards Agricultural Sustainability in the Sultanate Oman**, IEEE, pp.42-49, 2017.

[3] N Hari Kumar, Sandhya Baskaran, Sanjana Hariraj, Vaishali Krishnan, **An Autonomous Aquaponics System using 6LoWPAN based WSN IEEE**, pp. 125-132, 2016.

[4] Aquaponics:retrieved from **http://en.wikipedia.org/wikiiAquaponics**

[5] Megumi U. Leatherbury **Vegilab and Aquaponics Indoor Growing System** In IEEE ,2014.

[6] M.F. Saaid, N. S. M. Fadhil, M.S.A. Megat Ali, M.Z.H. Noor **Automated Indoor Aquaponic Cultivation Technique**, IEEE, pp 285- 289,2013.