./

Report – Remote Water Quality Monitoring System

Course Code: <CODE>



Version Number:

Team Members :

Team No:

Module: Model Based System Engineering

OBJECTIVE

The objective of this project is to design and develop a system which can effectively monitor the quality of water with the help of several sensors and transmit the sensor data over Internet for remote monitoring and also ensure fail-safe operation of the system using RTOS.

**REQUIREMENTS:**

1. **HIGH LEVEL REQUIREMENTS:**

|  |  |  |
| --- | --- | --- |
| **ID** | **Description** | **Status** |
| HR\_01 | Collect sensor data | Implemented |
| HR\_02 | Remote Monitoring | Implemented |
| HR\_03 | Robust System | Implemented |
| HR\_04 | Modular design | Implemented |

1. **LOW LEVEL REQUIREMENTS:**

|  |  |  |
| --- | --- | --- |
| **ID** | **Description** | **Status** |
| LR\_01 | Control Action | Implemented |
| LR\_02 | Analysis of sensor data | Partially Implemented |
| LR\_03 | Clean interface for website | Implemented |
| LR\_04 | Email alert | Implemented |
| LR\_05 | Scalability | Implemented |
| LR\_06 | Prognostic function | Future |

**4W’s and 1H:**

**What:**

This system can effectively monitor the quality of water with the help of several sensors and transmit the sensor data over Internet for remote monitoring and also ensure fail-safe operation of the system using RTOS.

**Who:**

This system can be used by people who needs to monitor water quality to ensure the water is safe to drink or to utilize for other purposes.

**Where:**

This system can be used in places where we need to measure and monitor water quality like water reservoirs (lakes, ponds, rivers, aquifers etc.,) and water purification units.

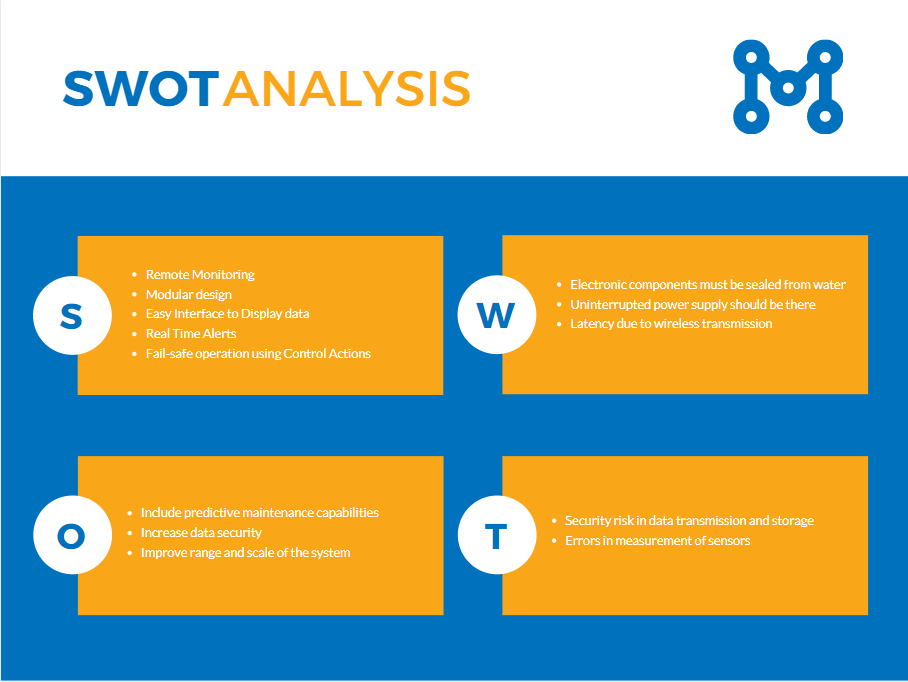
**When:**

This system is useful when we need to monitor water quality parameters of multiple water bodies/ reservoirs remotely which also shows trends of past data collected and shows visualizations of those data for easy monitoring.

**How:**

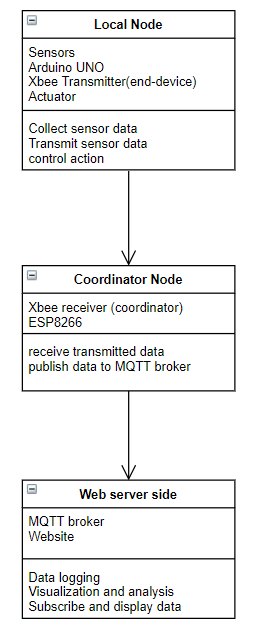
This system has two separate nodes, Local node (or End node) and Coordinator Node. Local node consists of the sensors, such as TDS sensor, pH sensor, temperature sensor and flow sensor, which measures the parameters and send the data to Arduino UNO. Arduino UNO is programmed with the help of FreeRTOS, a Real Time Operating System, which adds important multi-tasking and control functionality to Arduino. Arduino is interfaced with XBee module which acts as end node and transmits data wirelessly to XBee coordinator interfaced with ESP8266 microcontroller, which has built-in Wi-Fi to publish data securely to MQTT broker. Web application which we designed will then subscribe to the topic in MQTT broker and displays it in the website.

**SWOT ANALYSIS:**

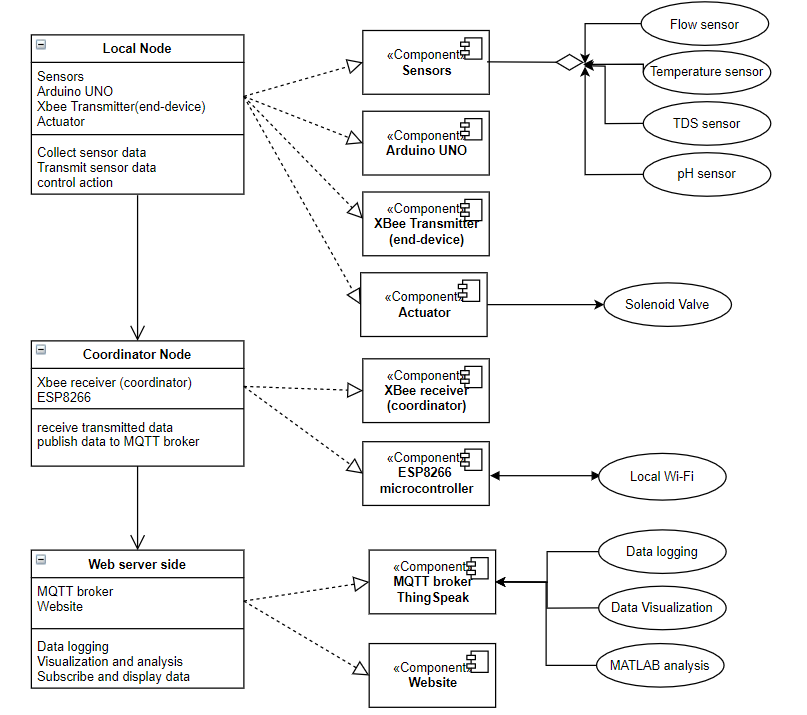


**ARCHITECTURE:**

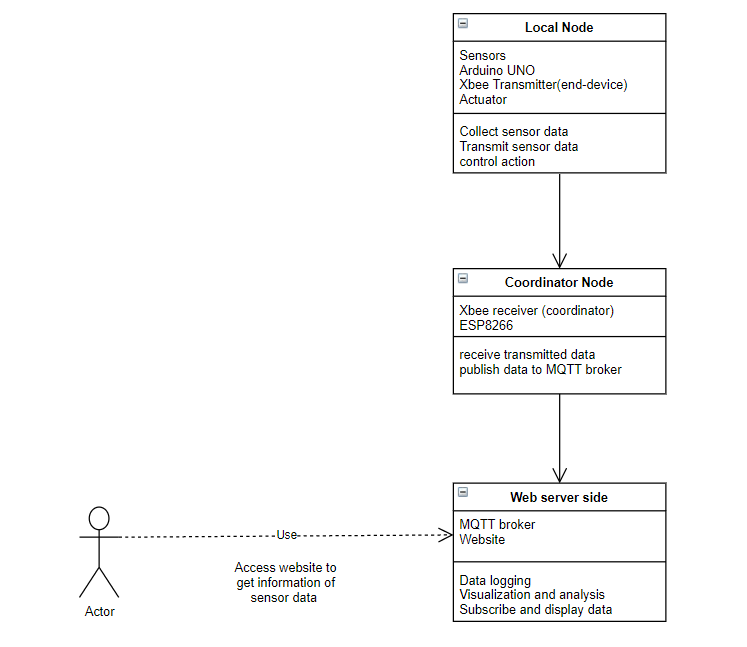
1. **STRUCTURE DIAGRAM:**
   1. **HIGH LEVEL**



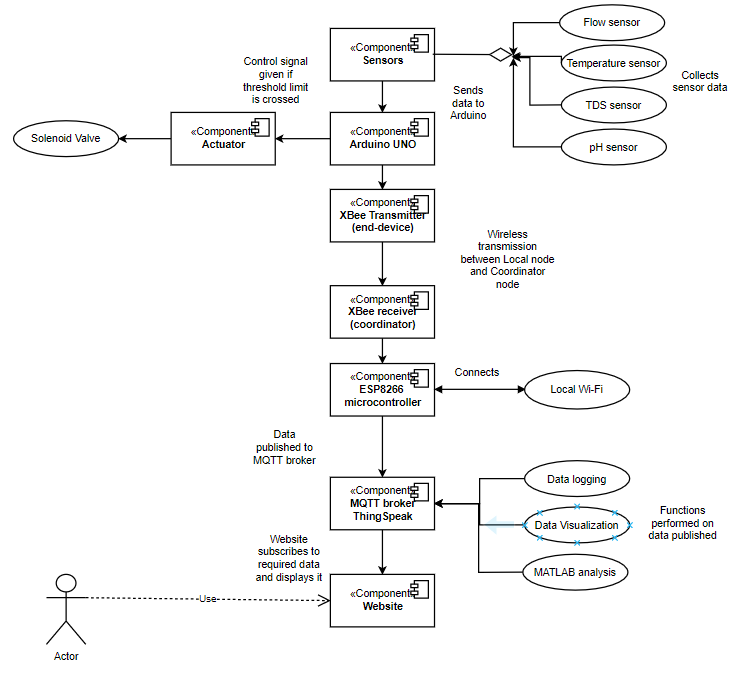
* 1. **LOW LEVEL**



1. **BEHAVIOUR DIAGRAM:**
   1. **HIGH LEVEL**



* 1. **LOW LEVEL**

**\**

**TEST PLAN AND TEST OUTPUT:**

1. **UNIT TEST – LOW LEVEL TESTING**

* Each individual hardware component to be tested and calibrated.
* Software side program codes are tested for functionality.

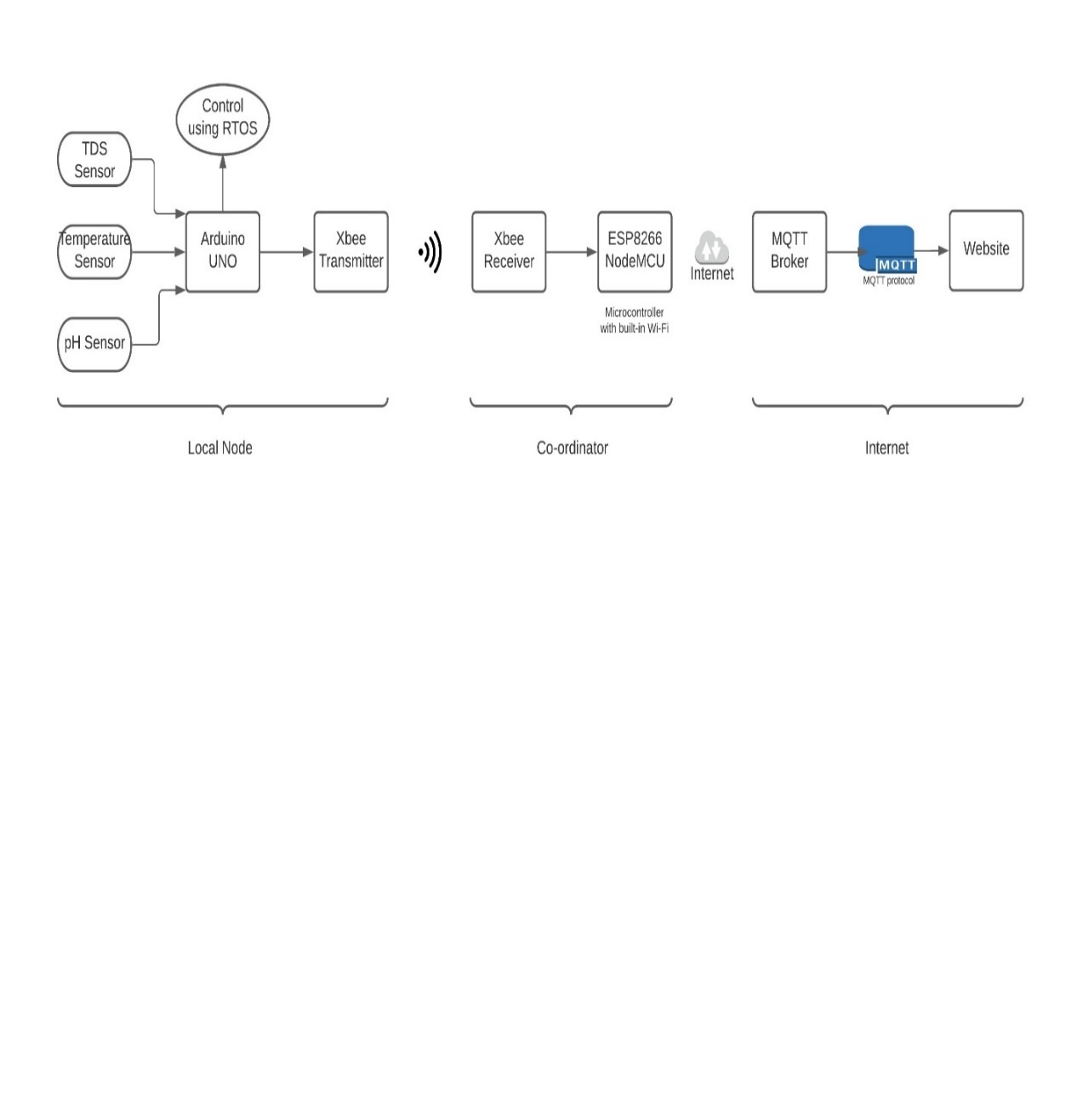
|  |  |  |  |
| --- | --- | --- | --- |
| Component | Test Type | Test conducted | Result |
| Arduino UNO | **Requirement based** | **sample coded uploaded** | **Passed** |
| ESP8266 | **Requirement based** | **sample code to connect to Wi-Fi** | **Passed** |
| XBee modules | **Requirement based** | **establish communication between XBee modules** | **Passed** |
| TDS sensor | **Scenario based** | **tested with different water samples with varying TDS** | **Passed** |
| TDS sensor | **Scenario based** | **added salt gradually to RO water** | **Passed** |
| pH sensor | **Scenario based** | **tested with solutions of known pH values** | **Passed** |
| Temperarture sensor | **Scenario based** | **tested for boundary conditions** | **Passed** |
| Flow sensor | **Scenario based** | **tested by varying flow rate of water** | **Passed** |
| Solenoid valve | **Requirement based** | **valve opening and closing when given control signal** | **Passed** |
| Data logging in MQTT | **Requirement based** | **data published to MQTT broker** | **Passed** |
| Data Analysis and Visualization | **Requirement based** | **custom functions implemented** | **Passed** |
| Website | **Requirement based** | **subscribe and display live data** | **Passed** |

1. **INTEGRATION TEST – HIGH LEVEL TESTING**

* The combined working of sensors, microcontrollers and web components are tested

|  |  |  |  |
| --- | --- | --- | --- |
| Component | Test Type | Test conducted | Result |
| Local Node | Requirement based | Combined working of components - Data collection and Transmission | Passed |
| Coordinator Node | Requirement based | Combined working of components - Data receiving and publishing | Passed |
| Web server side | Requirement based | Datalogging, visualization, analysis and display in website | Passed |

**IMPLEMENTATION:**



In this project, we have designed a system which have capable hardware and software components to achieve remote monitoring of water quality by measuring and analyzing drinking water quality parameters as stated by IS 10500:2012. The system has three overall sub-systems and two hardware sub-system or nodes, one being the local node, comprises of the sensors, Arduino UNO, XBee end device and actuator components, while the other being the coordinator, comprises of ESP8266 microcontroller and XBee coordinator. Each of the nodes perform their specific function with the help of embedded code flashed on to the microcontroller boards.

**Local Node**

The data collected by the sensors are transmitted with help of XBee modules using RF communication to coordinator node. Arduino UNO is programmed with the help of FreeRTOS, to help the Arduino to multi-task as we have created three tasks (or threads) TaskSensors - to collect and transmit sensor data, TaskpH – to check whether pH values are within permissible limit, TaskTds – to check whether TDS values are within permissible limits by. The last two tasks also ensure the fail-safe operation by closing the outlet in case if the permissible limits of sensor values are not met. This is done with the help of a solenoid valve, which is also interfaced with Arduino board. The Arduino and solenoid valve are powered with the help of step-down transformer and relay assembly. Thus, local node is capable of collecting sensor data, transmitting it and implement control action based on the collected data.

**Coordinator Node**

The data transmitted from the local node is received by XBee coordinator and sent to ESP8266 for publishing to the MQTT broker. ESP8266 has built-in Wi-Fi, so it can connect to the local available Wi-Fi and transmit data over Internet. XBee coordinator has to be powered ON always. It has the ability to communicate to multiple routers and end devices. It enables the system to be expanded in future. ESP8266 once connected to Wi-Fi published data to ThingSpeak using MQTT protocol.

**Web server side**

The data published by the coordinator is stored in a private channel in ThingSpeak (MQTT broker) containing four fields corresponding to temperature, TDS, pH, flow sensors values respectively. These sensor data are logged in the server and used for further analysis and visualizations using MATLAB. The channel is secured with the help of a unique channel ID and 16-bit authentication codes called Read API key and Write API key, without which data can neither be read from the channel nor written in to the channel. We have also designed a web application, which subscribes to the data in ThingSpeak using MQTT protocol and displays it in the website for remote monitoring purpose of clients and service providers. Only authorized personnel with username and password can access the website, which ensures the safety and reliability of the website. Each unit which is monitored will have data stored in different channels, so there won’t be any mismatch in the data shown in the website. Furthermore, MATLAB based analysis is done one the sensor data to remove anomalies or outliers in data to represent the trend of changes in data accurately.

FUTURE SCOPE

At present, the project only has one local node and one coordinator. The project is designed to be modular and scalable, meaning the components can be individually changed, if found damaged and new local nodes can be added to expand the system. The system can be expanded by installing local nodes in places where we need to measure water quality like water dispensers, water reservoirs, tanks etc., With just one coordinator and increasing the number of local nodes according to need and increasing number of routers to extend the range, we can expand the system significantly. Prognostic capabilities can be added to the system by implementing advanced machine learning algorithms to analyze the data collected. Therefore, predictive maintenance can be also be implemented which would enable us to prolong the life-span of the water purification units and also prevent any catastrophic malfunction of the units.

REFERENCES:

[1] IS 10500:2012 – Drinking water – specification

[2] Water Quality Monitoring and Notification System Using Arduino Based GSM System [Online], Available - http://www.instructables.com/id/Water-Quality-Monitoring-and-Notification-SystemU/

[3] NR Pavana and M. C. Padma, "Design of Low-Cost System for Real Time Monitoring of Water Quality Parameters in IOT Environment", International Journal of Advance Research incomputer Science and Management Studies, pp. 12-15, 2016

[4] CSIRO, “Wireless sensor network devices,” April 2013. [Online].

Available: http://www.ict.csiro.au/page.php?cid=87[5] S. Silva, N. Hoang Nghia, V. Tiporlini, and K. Alameh, “Web based water quality monitoring with sensor network: Employing zigbee and wimax technologies,” in High Capacity Optical Networks and Enabling Technologies (HONET). IEEE, 2011, pp. 138–142.

[6] M. A. Nasirudin, U. N. Za’bah, and O. Sidek, “Fresh water real-time monitoring system based on wireless sensor network and gsm,” in 2011 IEEE Conference on Open Systems (ICOS). IEEE, 2011, pp. 354–357.

[7] M. Zennaro, A. Floros, G. Dogan, S. Tao, C. Zhichao, H. Chen, M. Bahader, H. Ntareme, and A. Bagula, “On the design of a water quality wireless sensor network (wqwsn): An application to water quality monitoring in malawi,” in International Conference on Parallel Processing Workshops (ICPPW ’09). IEEE.

[8] T. L. Dinh, W. Hu, P. Sikka, P. Corke, L. Overs, and S. Brosnan, “Design and deployment of a remote robust sensor network: Experiences from an outdoor water quality monitoring network,” in 32nd IEEE Conference on Local Computer Networks (LCN 2007). IEEE, 2007, pp. 799–806.

[9] B. O’Flyrm, R. Martinez, J. Cleary, C. Slater, F. Regan, D. Diamond, and H. Murphy, “Smartcoast: A wireless sensor network for water quality monitoring,” in 32nd IEEE Conference on Local Computer Networks (LCN 2007). IEEE, 2007, pp. 815–816.

[10] Zulhani Rasin, “Water Quality Monitoring System Using Zigbee Based Wireless Sensor Network”, in International Journal of Engineering & Technology IJET Vol: 9 No: 10

[11] N. R. Moparthi, C. Mukesh and P. Vidya Sagar, "Water Quality Monitoring System Using IOT," 2018 Fourth International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB), 2018, pp. 1-5, doi: 10.1109/AEEICB.2018.8480963.

[12] A.N.Prasad, K. A. Mamun, F. R. Islam, H. Haqva, “Smart Water Quality Monitoring System” in conference paper DOI - 10.1109/APWCCSE.2015.7476234