CAREER EPISODE 02

A REAL-TIME WATER QUALITY MONITORING SYSTEM CAPABLE OF IOT FOR EARLY CONTAMINATION DETECTION

INTRODUCTION

CE 2.1

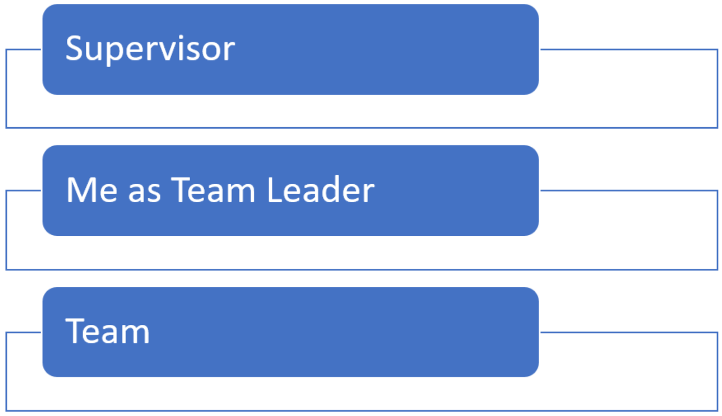
For the project explained in career episode 2, I have worked on a real time water quality monitoring system. The project was associated with my 6th semester course, "Fundamental of IOT". This project's primary goal was to provide a water monitoring system using IoT. I used a controller that processed incoming data from sensors to a local server or cloud. After processing, the sensor data is shown on a website in real-time charts and via a QR code that can be scanned on a mobile device to view the water quality condition in real time. My expertise in Computer Sciences and Engineering courses is fully showcased in this project. It spanned weeks with the help of my course instructor. I started working on it in [redacted] and completed it by the end of [redacted] It helped polishing my skills, technical knowledge and management strategies. I produced a project overview and guided my team in accordance with it as the team leader.

BACKGROUND

CE 2.2.1

To guarantee that people have a safe supply of drinking water for their consumption, an Internet of Things (IoT) based monitoring system has been designed to check water quality. It has a few water quality sensors that measure the turbidity, turbidity, temperature, and total dissolved solids (TDS) of the water. The sensor data was transmitted over Wi-Fi to a local server or cloud for real-time data display on a website after being processed by a microcontroller like an Arduino or ESP32. If the water quality changes, officials will be notified by email or SMS to prevent major health problems. A QR code has also been introduced to allow people to instantly check the quality of their water.

PROJECT REPORTING HIERARCHY



PERSONAL ENGINEERING ACTIVITY

CE 2.3.1

For this project, my idea was to place several water quality sensors in water resources and connect to a microcontroller. This data was subsequently sent to a local server or the cloud by the Wi-Fi-capable microcontroller. Following processing, the sensor data is shown in real-time charts on a website and as a QR code that can be scanned on a mobile device to view the current state of the water quality. In the event of a major change in the water quality such as pH, turbidity, or TDS, exceed the threshold level, an email or SMS alert is sent to the appropriate department for necessary action.

CE 2.3.2

From my previous knowledge, I already knew that water is affected by different types of pollutants. For example, chemical fertilizers are utilized to promote agricultural growth, they have flowed into lakes and streams due to heavy rainfall, causing hypertrophication. This has resulted in an excessive number of algae that pollutes the water. Traditional water quality monitoring involves collecting water samples from multiple water resource locations, analyzing them using a variety of analytical techniques, and diagnosing them in a laboratory. However, because these methods are slow to produce results and are not considered to be effective, therefore an efficient method of monitoring water quality was needed to be developed.

CE 2.3.3

I outlined the scope of this project so that users can examine the water quality in real time by scanning a QR code; they don't need to log in to use QR Scan. I designed it in a way that the user and microcontroller must be linked to the same Wi-Fi network in order to view the water quality status via a QR code scan. The administrator has the access to create a new login so that users can access the website. The administrator could download the sensor data and charts. The TDS sensor can operate between -10°C and 55°C. -55°C to 125°C is the temperature sensor's operational range. The turbidity sensor operates between 5 and 90°C. A Ph sensor has to know the water's temperature in order to measure the PH of the water precisely.

CE 2.3.4

To achieve the goals of system administrator, user, and admin, I made a list of use cases i.e. register for a new account, log out, see and download charts, download sensor data, scan a QR code, and adjust microcontrollers and sensors. I made a use case diagram that graphically depicted how a user interacts with the system. A case diagram shows the characters, use cases, and their connections.

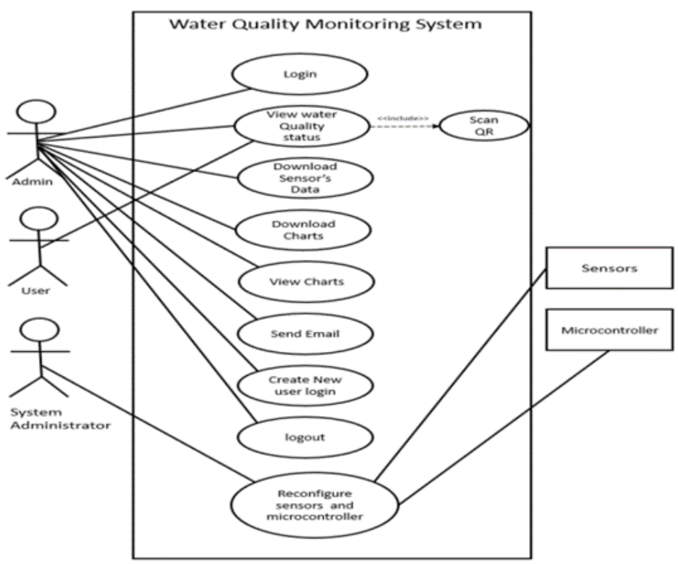


Figure 1 Use Case Diagram

Flow chart is shown below:

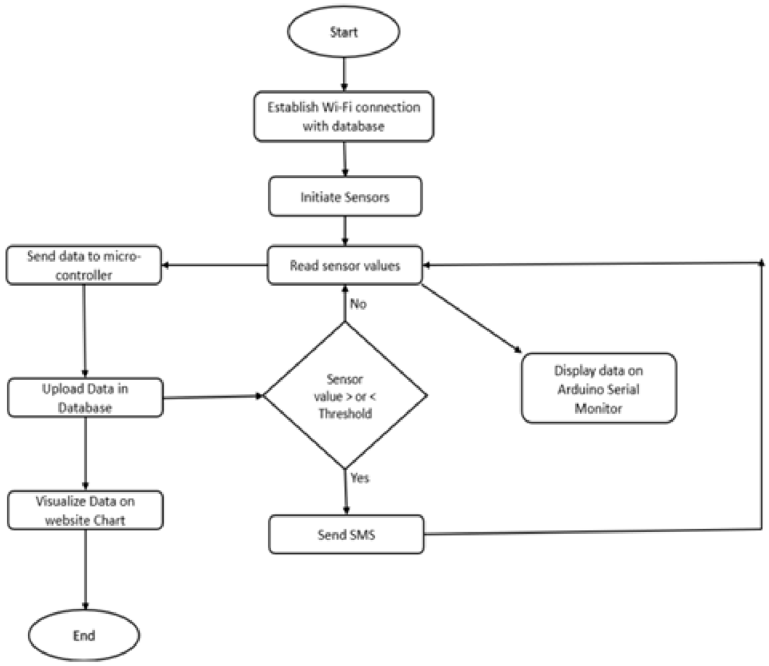


Figure 2 Flow Chart

CE 2.3.5

I created the system architecture because I knew from my earlier model that I had to use it to illustrate the components of the system and how they interacted. The application's "Three Tier Architecture" serves as its base. There were three layers i.e. presentation layer(browser), logic tier(PHP script and web server) and Data tier(database and database server). The ability to modify or redesign each of the three tiers independently without impacting the other two layers is the primary argument in favor of a three-tier architecture.

CE 2.3.6

I created a user interface that the user may interact with, and this layer interfaces with another tier to complete the user's request. Different user interfaces need information about user requirements in order to make sure that all users can use it easily. These interfaces will communicate with the application tier since it houses all of the insertion-related business logic. I then developed business logic. The functionality, including insertion and updating, will be controlled by this layer. To enter and retrieve data, this layer will speak with the presentation tier. Following data processing, I designed the application tier that exchanged information with the data tier in order to store the outcomes. After all processing was finished in the application tier, I used the third tier, the database, to store data that was used later. To retrieve data, this layer only interacted with the presentation tier.

CE 2.3.7

After this, I designed an activity diagram:

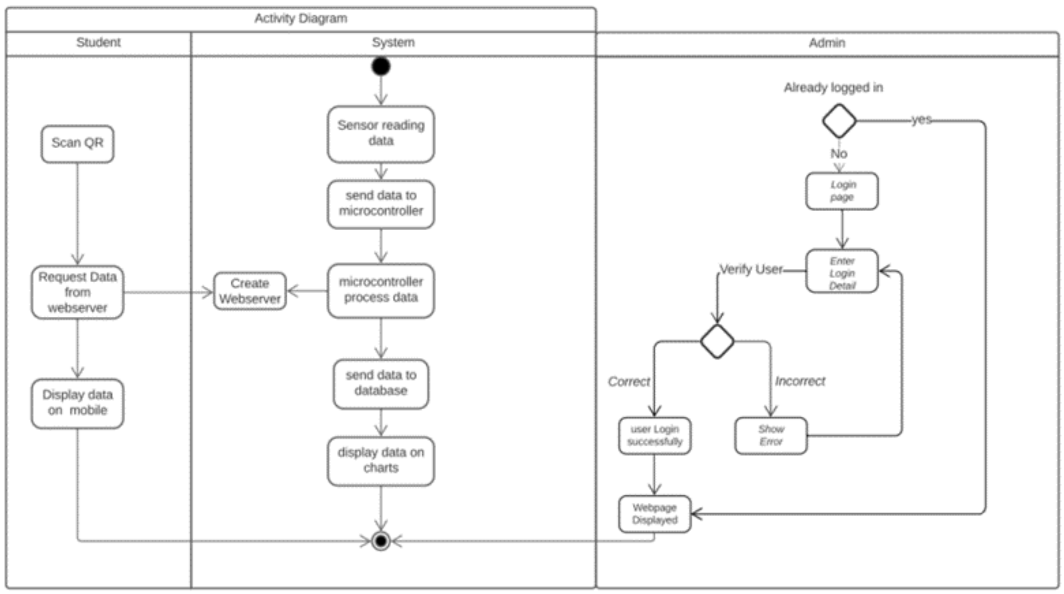


Figure 3 Activity Diagram

After visiting the market, I decided to use ESP32 microcontroller. The platform for the documentation has been selected to be Microsoft Word 2016. As design diagrams, Lucid Chart and Microsoft PowerPoint have been selected. I used Sublime Text 3 tool to code data visualization websites. I used a cross-platform program called Arduino IDE. I wrote and uploaded an Arduino application using this. I defined the initial state at boot and run using setup(). Additionally, I sketched using loop() and ran it after setup was finished.

CE 2.3.8

Based on my prior knowledge, I realized that in order to code and design web-based applications, I needed Sublime Text 3. I worked on the admin login action for the user interface, as seen below:

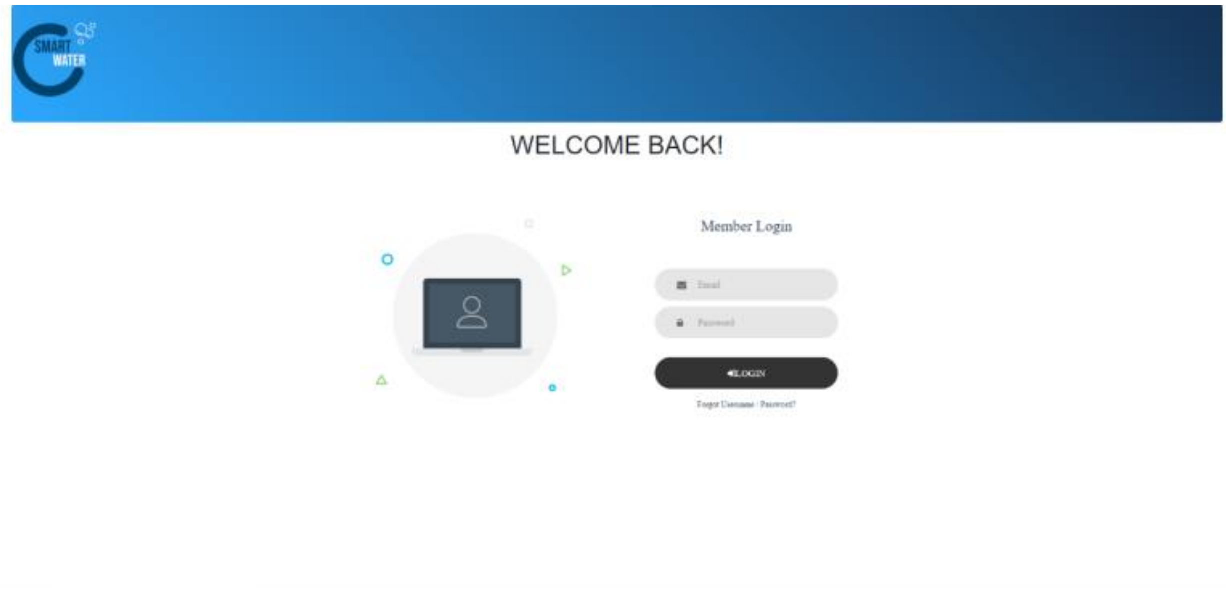


Figure 4 Admin Login Activity

Figure 5 Main Dashboard 1

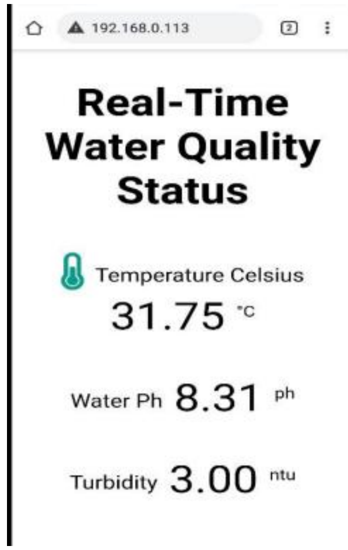


Figure 6 Water quality status on Mobile

CE 2.3.9

For this, I used local servers. Designing the hardware was my next task, and I decided to utilize an ESP32 microcontroller for this. I was aware that it was an inexpensive, that could interface with a variety of different devices, including boards, relays, and sensors. Additionally, because an analog pH sensor had a handy gravity hookup, I used it. For a pH measurement at 0.1 pH, an instant connection to the probe and controller was required. It has a BNC connector, an LED that indicated power, and a PH2.0 sensor interface. I connected the pH sensor to a connection and inserted the PH2.0 interface into the analog port of the Arduino controller.

CE 2.3.10

I made use of a turbidity sensor, which measured the amount of turbidity in the water to determine its quality. For identifying suspended particles in water, I monitored light transmittance and scattering rate. The level of liquid turbidity rises as TSS does. Additionally, DS18B20 temperature sensors were 1-wire accurate and had a precision of up to 12 bits, I chose to use them. I measured the amount of TDS in one liter of water using a total dissolved solid sensor. More soluble materials dissolve in water and the water is often less clean when the TDS is higher. Before soldering the sensors, I connected and tested them using a breadboard. I connected the pins at each end using jumper wires.

CE 2.3.11

This document's goal was to offer a methodical approach to system testing, which aids in determining the necessary testing effort. To determine whether the requirements of a specification are fulfilled, I employed an acceptance test approach and conducted tests. This test's primary goal is to assess how well the system complies with business requirements and confirm that it satisfies the standards needed to be delivered to end users. It also specifies the amount of time that will be spent on testing. Login, check water quality status, download charts, download sensor data, send email, test the Ph sensor using 4.0 and 10.0 PH buffer solution, test the turbidity sensor using clear and muddy water samples, and test the temperature sensor using cold normal hot water samples are among the features that were evaluated.

CE 2.3.12

As part of the software testing process, I decided to conduct user acceptance testing. One of the last and most important software project processes that needs to be completed before newly created software is released onto the market is user acceptance testing. I made the test cases to see if a new user could register and log in as the administrator. I tested the water's quality using a QR sensor, and the results were shown on the screen. I downloaded real-time charts of the water quality metrics turbidity, total dissolved solids, and pH while working on another test case. After choosing the chart's image format and clicking "download images," the test was successful. I chose to use CSV and XLS files to get sensor data from the website, and the test was successful. In the end, I verified that the data was transmitted by email, and this test was also successful. I investigated the project's cost, as indicated below:

Sr. No.	Items	Amount (Rs.)
1	ESP Microcontroller	950
2	Turbidity Sensor	2250
3	pH Sensor	6200
4	Temperature Sensor	220
5	Breadboard	130
6	Jumper Wires	350
7	Buffer Solution	600
8	Shipping Charges	350

	Total Cost	**11050**

SUMMARY

CE 2.4

I developed a system that monitors water quality in real-time and allowed users to rapidly check its status by scanning a QR code. The project's primary goal was to ensure that students, staff, and other [redacted] personnel may consume water safely. The administrator received SMS notifications in the event that there was a major divergence in the water quality parameters, and they can examine and download water quality data generated by the system on charts.