**CHAPTER 1**

**INTRODUCTION**

Water is one of the most valuable resources on earth, as it is essential for the survival of all living organisms, including humans. However, the quality of water is being threatened by the increasing levels of pollution caused by various human activities. According to the United Nations world water development report 2017, around 80% of the world's wastewater is discharged back into the environment, mainly untreated, causing severe damage to rivers, lakes, and seas. Water pollution is a significant environmental and public health concern, and if not addressed, the problem will only worsen, especially with the projected increase in global freshwater demand by 2050.

To ensure the safety of water resources and human health, water quality monitoring and prediction have become essential. Traditional methods of water quality monitoring, such as manual analysis, are inefficient and time-consuming, requiring the collection of water samples and subsequent laboratory analysis. As a result, there is a growing need for an improved system to monitor and predict water quality in real-time.

With the advent of the Internet of Things (IoT) and machine learning (ML) technologies, it is now possible to develop an accurate and efficient water quality prediction system. The proposed system integrates IoT sensors to collect real-time data from water. The data is then processed using ML algorithms, including support vector machine (SVM), random forest (RF), support vector classifier (SVC), and extreme gradient boosting (XG Boost) to predict water quality using the parameters pH, solids, hardness, chloramines, trihalomethanes, organic carbon, sulfates, conductivity and turbidity.

The proposed water quality prediction system offers an accurate, efficient, and real-time monitoring and prediction platform that can be used by environmentalists, water resource managers, and policymakers to make informed decisions about water resource management. The system's potential impact is immense, as it has the potential to revolutionize water quality monitoring and prediction, leading to better water resource management, environmental sustainability, and improved public health.

**1.1 INTERNET OF THINGS**

The Internet of Things (IoT) refers to the interconnectivity of physical devices and objects through the internet, allowing them to collect and exchange data without human intervention. This network of connected devices includes everything from smartphones and smart homes to industrial machinery and city infrastructure. The IoT has the potential to transform numerous industries, including healthcare, transportation, agriculture, and manufacturing, by increasing efficiency, reducing costs, and enabling new services and products. However, the proliferation of IoT devices also raises concerns around privacy, security, and data management.

**1.1.1 Purpose of Internet of Things**

The purpose of using the Internet of Things (IoT) in water quality prediction is to enable real-time monitoring, analysis, and management of water resources. IoT devices, such as sensors and smart meters, can be deployed in water bodies, treatment plants, and distribution networks to collect and transmit data on water quality parameters, such as pH, temperature, and turbidity.

In addition to water quality prediction, IoT can also contribute to more efficient water management practices by enabling better resource allocation, reducing water waste, and improving the accuracy of billing and metering. Overall, the purpose of IoT in water quality prediction is to promote sustainable and effective water management practices that ensure the safety and reliability of our water resources.

**1.2 MACHINE LEARNING**

Machine learning is a subset of artificial intelligence that involves the use of algorithms and statistical models to enable computers to learn from data, without being explicitly programmed. Machine learning systems can identify patterns, make predictions, and take actions based on the data they receive.

There are three main types of machine learning: supervised learning, unsupervised learning, and reinforcement learning. Supervised learning involves training a model on a labelled dataset, where the output is known, to predict the output for new data. Unsupervised learning involves discovering patterns and relationships in unlabelled data, without specific target variables. Reinforcement learning involves training a model to take actions in an environment to maximize a reward.

Machine learning has numerous applications in various industries, including healthcare, finance, transportation, and marketing. It is used to detect fraud, diagnose diseases, and personalize user experiences.

However, the effectiveness of machine learning models depends on the quality and quantity of data used to train them, as well as the algorithms and techniques used. Additionally, ethical considerations such as bias and privacy must be taken into account when implementing machine learning systems.

**1.2.1 Purpose of Machine Learning**

The purpose of using machine learning in water quality prediction is to improve the accuracy and efficiency of predicting and monitoring water quality parameters. Machine learning algorithms can analyze vast amounts of data from sensors and other sources to identify patterns and trends that may be difficult for humans to detect.

**1.3 DESCRIPTION OF THE PROJECT**

This project aims to develop a water quality prediction system by leveraging the power of machine learning and internet of things (IoT) technologies. The system will consist of multiple sensor nodes that monitor various water quality parameters such as temperature, pH, solids, hardness, chloramines, trihalomethanes, organic carbon, sulphates, conductivity and turbidity.

Machine learning algorithms are then applied to the collected data to analyze it and make predictions about the water quality. The system can be trained to detect abnormal changes in the water quality and alert the authorities in case of potential water pollution or contamination.

The system can also be used to provide insights into the water quality and help authorities make informed decisions on water management and treatment. For instance, the system can predict the amount of chlorine required to treat the water, which can help in reducing the costs of water treatment.

Overall, the Water quality prediction system using IoT and Machine learning project can help in the management and protection of water resources by providing real-time data on water quality and predicting potential pollution events.

**1.4 PROBLEM ANALYSIS**

One of the main challenges in water quality management is the timely and accurate detection of contaminants. Traditional methods of water quality monitoring involve manual sampling and laboratory analysis, which can be time-consuming, expensive and prone to errors.

The current water quality monitoring systems typically provide only periodic measurements at discrete locations, which can result in missed events or insufficient spatial and temporal coverage.

The IoT sensors and connectivity technologies for real-time monitoring of water quality need to be deployed and configured properly to capture accurate and reliable data. With the large amounts of data generated by IoT sensors, it can be challenging to process, store, and analyze the data in real-time.

Machine learning algorithms can be used to predict water quality levels based on historical data, but they require large amounts of reliable and diverse data for accurate predictions.

The IoT sensors and the machine learning algorithms need to be maintained and updated regularly to ensure their accuracy and reliability over time.

**CHAPTER 2**

**LITERATURE SURVEY**

**TITLE:** Water Quality Classification Using SVM And XGBoost Method (2022)

**AUTHORS:** Hasriq Izzuan Hasnol Yusri, A’zraa Afhzan Ab Rahim, Siti Lailatul Mohd Hassan, Ili Shairah Abdul Halim, Noor Ezan Abdullah

Various pollutants have been endangering water quality over the past decades. As a result, predicting and modeling water quality have become essential to minimizing water pollution. This research has developed a classification algorithm to predict the water quality classification (WQC). The WQC is classified based on the water quality index (WQI) from 7 parameters in a dataset using Support Vector Machine (SVM) and Extreme Gradient Boosting (XGBoost). The results from the proposed model can accurately classify the water quality based on their features. The research outcome demonstrated that the XGBoost model performed better, with an accuracy of 94%, compared to the SVM model, with only a 67% accuracy. Even better, the XGBoost resulted in only 6% misclassification error compared to SVM, which had 33%. On top of that, XGBoost also obtained consistent superior results from 5-fold validation with an average accuracy of 90%, while SVM with an average accuracy of 64%. Considering the enhanced performance, XGBoost is concluded to be better at water quality classification.

**TITLE:** Interpretable tree-based ensemble model for predicting beach

water quality (2022)

**AUTHORS:** L. Li et al

Tree-based machine learning models based on environmental features offer low-cost and timely solutions for predicting microbial fecal contamination in beach water to inform the public of the health risk. However, many of these models are black boxes that are difficult for humans to understand, which may cause severe consequences such as unexplained decisions and failure in accountability. To develop interpretable predictive models for beach water quality, we evaluate five tree-based models, namely classification tree, random forest, CatBoost, XGBoost, and LightGBM, and employ a state-of-the-art explanation method SHAP to explain the models. When tested on the Escherichia coli (E. coli) concentration data collected from three beach sites along Lake Erie shores, LightGBM, followed by XGBoost, achieves the highest averaged precision and recall scores. For all three sites, both models suggest lake turbidity as the most important predictor, and elucidate the crucial role of accurate local data of wave height and rainfall in the model development. Local SHAP values further reveal the robustness of the importance of lake turbidity as its SHAP value increases nearly monotonically with its value and is minimally affected by other environmental factors. Moreover, we found an intriguing interaction between lake turbidity and day-of-year. This work suggests that the combination of LightGBM and SHAP has a promising potential to develop interpretable models for predicting microbial water quality in freshwater lakes.

**TITLE:** Prediction of groundwater quality using efficient machine learning technique (2021)

**AUTHORS:** Lingbo Li, Jundong Qiao, Guan Yu, Leizhi Wang, Hong-Yi Li, Chen Liao, Zhenduo Zhu

To ensure safe drinking water sources in the future, it is imperative to understand the quality and pollution level of existing groundwater. The prediction of water quality with high accuracy is the key to control water pollution and the improvement of water management. In this study, a deep learning (DL) based model is proposed for predicting groundwater quality and compared with three other machine learning (ML) models, namely, random forest (RF), eXtreme gradient boosting (XGBoost), and artificial neural network (ANN). A total of 226 groundwater samples are collected from an agriculturally intensive area Arang of Raipur district, Chhattisgarh, India, and various physicochemical parameters are measured to compute entropy weight-based groundwater quality index (EWQI). Prediction performances of models are determined by introducing five error metrics. Results showed that DL model is the best prediction model with the highest accuracy in terms of R2, i.e., R2 = 0996 against the RF (R2 = 0.886), XGBoost (R2 = 0.0.927), and ANN (R2 = 0.917). The uncertainty of the DL model output is cross-verified by running the proposed algorithm with newly randomized dataset for ten times, where minor deviations in the mean value of performance metrics are observed. Moreover, input variable importance computed by prediction models highlights that DL model is the most realistic and accurate approach in the prediction of groundwater quality.

**TITLE:** Proposed formulation of surface water quality and modelling using gene expression, machine learning, and regression techniques (2021)

**AUTHORS:** M. I. Shah, M. F. Javed, and T. Abunama

Water pollution is an international concern that impedes human health, ecological sustainability, and agricultural output. This study focuses on the distinguishing characteristics of an evolutionary and ensemble machine learning (ML) based modelling to provide an in-depth insight of escalating water quality problems. The 360 temporal readings of electric conductivity (EC) and total dissolved solids (TDS) with several input variables are used to establish multi-expression programming (MEP) model and random forest (RF) regression model for the assessment of water quality at Indus River. New hydrological insight for the region: The developed models were evaluated using several statistical metrics. The findings reveal that the determination coefficient (R2) in the testing phase (subject to unseen data) for the all the developed models is more than 0.95, indicating the accurateness of the developed models. Furthermore, the error measurements are much lesser with root mean square logarithmic error (RMSLE) nearly equals to zero for each developed model. The mean absolute percent error (MAPE) of MEP models and RF models falls below 10% and 5%, respectively, in all three phases (training, validation and testing). According to the sensitivity study of generated MEP models about the relevance of inputs on the predicted EC and TDS, shows that bi-carbonates and chlorine content have significant influence with a sensitiveness score more than 0.90, whereas the impact of sodium content is less pronounced. All the models (RF and MEP) have lower uncertainty based on the prediction interval coverage probability (PICP) calculated using the quartile regression (QR) approach. The PICP% of each model is greater than 85% in all three stages. Thus, the findings of the study indicate that developing intelligent models for water quality parameter is cost effective and feasible for monitoring and analyzing the Indus River water quality.

**TITLE:** Prediction of groundwater quality using efficient machine learning technique (2021)

**AUTHORS:** Sudhakar Singha, Srinivas Pasupuleti, Soumya S Singha, Rambabu Singh, Suresh Kumar

To ensure safe drinking water sources in the future, it is imperative to understand the quality and pollution level of existing groundwater. The prediction of water quality with high accuracy is the key to control water pollution and the improvement of water management. In this study, a deep learning (DL) based model is proposed for predicting groundwater quality and compared with three other machine learning (ML) models, namely, random forest (RF), eXtreme gradient boosting (XGBoost), and artificial neural network (ANN). A total of 226 groundwater samples are collected from an agriculturally intensive area Arang of Raipur district, Chhattisgarh, India, and various physicochemical parameters are measured to compute entropy weight-based groundwater quality index (EWQI). Prediction performances of models are determined by introducing five error metrics. Results showed that DL model is the best prediction model with the highest accuracy in terms of R2, i.e., R2 = 0996 against the RF (R2 = 0.886), XGBoost (R2 = 0.0.927), and ANN (R2 = 0.917). The uncertainty of the DL model output is cross-verified by running the proposed algorithm with newly randomized dataset for ten times, where minor deviations in the mean value of performance metrics are observed. Moreover, input variable importance computed by prediction models highlights that DL model is the most realistic and accurate approach in the prediction of groundwater quality.

**TITLE:** Modelling and Prediction of Water Quality by Using Artificial Intelligence (2021)

**AUTHORS:** Mosleh Hmoud Al-Adhaileh,

With fast economic growth and increased urbanization, water pollution has become grimmer. Understanding the issues and patterns of water quality is also critical for water pollution reduction and regulation. Most countries around the world have started to develop environmental water management schemes to truly understand the quality of the marine ecosystem. Water is life’s most important substance. Although 71% of the Earth’s surface is covered with water, the vast majority of it (95%) is salt water. Thus, conserving the quality of fresh water is essential. Almost one billion people do not have access to adequate drinking water sources, and two million people die every year from contaminated water and poor sanitation and hygiene. With advanced computing using artificial intelligence (AI) techniques, the modelling of water quality has been developed to resolve water quality issues. Artificial neural networks (ANNs) have aided in the monitoring of water quality systems by predicting changes in water quality. They can immensely improve the efficiency of aquaculture. The simulation of water quality conditions has difficulties and challenges regarding the use of the hydrodynamic and water quality model, a relatively novel computational approach. ANNs have been widely established in many disciplines and provide an alternative technique for understanding and monitoring water quality in reservoirs. ANNs have been successfully applied to simulate and forecast water quality in water bodies. Numerous ANN methods, such as feed-forward neural networks, have been used in various applications. The fuzzy logic system has been developed to solve complex nonlinear systems. ANN applications have been successfully used as tools to compute and predict the quality of water bodies.

**CHAPTER 3**

**SYSTEM ANALYSIS**

**3.1 EXISTING SYSTEM (ML)**

There are several existing systems of water quality management that utilize machine learning. One such example is the use of artificial neural networks (ANNs) to predict water quality parameters based on historical data.

ANNs are a type of machine learning algorithm that can learn complex patterns in data and make predictions based on that learning. In water quality management, ANNs can be trained on historical water quality data to predict future water quality parameters such as dissolved oxygen levels, pH, and turbidity.

Another example of a machine learning-based system for water quality management is the use of sensor networks. These sensor networks can collect real-time data on water quality parameters such as temperature, pH, and dissolved oxygen levels. Machine learning algorithms can be applied to this data to identify patterns and anomalies that could indicate a potential water quality problem.

Additionally, machine learning algorithms can be used for water quality monitoring and prediction. For example, predictive models can be developed using machine learning algorithms to forecast water quality parameters based on weather patterns, land use, and other environmental factors.

Overall, the use of machine learning in water quality management can improve the accuracy and efficiency of water quality monitoring, prediction, and management.

**EXISTING SYSTEM (IoT)**

The Internet of Things (IoT) is being increasingly used in water quality management systems. IoT refers to a network of physical devices that are connected to the internet and can collect and exchange data. Here are some examples of existing water quality management systems that use IoT:

Smart sensors: IoT sensors can be used to collect real-time data on water quality parameters such as temperature, pH, dissolved oxygen levels, and turbidity. This data can be transmitted to a central database or cloud-based system for analysis.

Automated water quality monitoring: IoT can be used to automate water quality monitoring by deploying sensors throughout a water system to collect data. This data can then be analyzed in real-time to detect any anomalies or potential issues.

Leak detection: IoT sensors can also be used to detect leaks in a water system. These sensors can be placed at various points in the system to detect changes in pressure or flow that could indicate a leak.

Remote management: IoT can be used to remotely manage water quality systems, such as water treatment plants or distribution systems. This can enable operators to monitor and control systems from a central location, reducing the need for on-site visits and improving efficiency.

Predictive maintenance: IoT sensors can be used to monitor the condition of equipment, such as pumps and valves, in a water system. This data can then be used to predict when maintenance is needed, reducing downtime and maintenance costs.

While IoT-based water quality management systems offer many potential benefits, there are also some challenges to consider. For example, IoT systems require reliable connectivity, and data security and privacy concerns need to be addressed. Additionally, the cost of implementing and maintaining IoT systems can be significant. However, with proper planning and management, the benefits of IoT-based water quality management systems can outweigh these challenges.

**3.1.1 Disadvantages of Existing System**

While there are many potential advantages to using the Internet of Things (IoT) in water quality management systems, there are also some potential disadvantages to consider:

1. Security risks: IoT devices are often connected to the internet, which can make them vulnerable to cyber-attacks. This is particularly concerning in water quality management systems, as a security breach could potentially impact public health and safety.

2. Reliability: IoT devices can be prone to malfunctions, such as connectivity issues or power outages. This can impact the reliability of water quality data and may require additional monitoring or backup systems.

3. Complexity: IoT-based water quality management systems can be complex and require specialized expertise to design, install, and maintain. This can increase costs and make it more challenging to implement these systems in smaller communities or organizations with limited resources.

4. Data overload: IoT devices can generate large amounts of data, which can be difficult to manage and analyze. Without proper data management and analysis tools, this data can be overwhelming and difficult to use effectively.

5. Cost: Implementing and maintaining an IoT-based water quality management system can be costly, particularly for small or rural communities with limited resources. This may make it challenging to justify the expense of these systems.

While there are many advantages to using machine learning in water quality management, there are also some potential disadvantages to consider. Here are a few:

1. Dependence on data: Machine learning algorithms rely heavily on the quality and quantity of the data they are trained on. If the data is incomplete, inaccurate, or biased, the resulting predictions may be unreliable. Additionally, if the data is not regularly updated, the model may become less effective over time.

2. Complexity: Machine learning algorithms can be complex and difficult to understand. This can make it challenging for non-experts to interpret the results and make informed decisions based on them.

3. Limited interpretability: Some machine learning algorithms, such as deep learning neural networks, can be difficult to interpret. This can make it challenging to understand how the model is making its predictions and can make it harder to identify potential problems or errors.

4. Resource-intensive: Training and running machine learning algorithms can be resource-intensive, requiring significant computational power and large amounts of data. This can make it more challenging and expensive to implement machine learning-based water quality management systems.

5. Need for ongoing maintenance: Like any technology, machine learning algorithms require ongoing maintenance and updates to remain effective. This can require significant resources and expertise, particularly if the system is large or complex.

Overall, while there are many potential benefits to using machine learning in water quality management, it's important to carefully consider these potential disadvantages and ensure that the benefits outweigh the costs and challenges.

**3.2 PROPOSED SYSTEM**

A proposed system of water quality management using both the Internet of Things (IoT) and machine learning could offer many potential benefits. Here's an overview of how such a system might work:

IoT sensors would be deployed throughout a water system to collect real-time data on water quality parameters such as temperature, pH, dissolved oxygen levels, and turbidity. This data would be transmitted to a central database or cloud-based system for analysis.

Machine learning algorithms would be used to analyze the data collected by the IoT sensors. These algorithms could detect patterns and anomalies in the data that might indicate water quality issues, such as contamination or changes in water flow or pressure.

The machine learning algorithms could also be used to predict future water quality issues based on historical data. For example, the algorithms could identify trends in water quality data that might suggest an upcoming issue, allowing operators to take preventative action.

The system could use automated alerts to notify operators of potential water quality issues in real-time. This could enable operators to respond quickly to address the issue and prevent any negative impacts on public health or safety.

The system could also incorporate predictive maintenance features, using data from IoT sensors to predict when equipment maintenance is needed. This could reduce downtime and maintenance costs.

While a system of water quality management using both IoT and machine learning offers many potential benefits, it's important to carefully consider the challenges and potential disadvantages as well. These might include the complexity and cost of implementing and maintaining such a system, as well as concerns around data security and privacy. However, with proper planning and management, a system of water quality management using IoT and machine learning could offer an effective and efficient way to monitor and maintain water quality, helping to ensure public health and safety.

**3.2.1 Advantages of Proposed System**

The proposed system of water quality management using internet of things and machine learning has several advantages:

1. Real-time monitoring: The system enables real-time monitoring of water quality parameters such as temperature, pH, and turbidity. This allows for immediate action to be taken in case of any abnormal values or fluctuations.

2. Cost-effective: The system is cost-effective compared to traditional methods of water quality management, which require manual sampling and laboratory analysis.

3. Improved accuracy: The use of sensors and machine learning algorithms in the system improves the accuracy and reliability of the water quality data, reducing the chances of errors and false readings.

4. Early detection of anomalies: The system can detect anomalies in the water quality parameters, which can indicate the presence of contaminants or other harmful substances. Early detection enables prompt action to be taken to prevent the spread of contaminants.

5. Remote monitoring: The system allows for remote monitoring of water quality parameters, which reduces the need for physical inspection and saves time and resources.

6. Predictive maintenance: The system can use machine learning algorithms to predict the need for maintenance of the sensors and other components, reducing downtime and improving efficiency.

Overall, the proposed system of water quality management using internet of things and machine learning has the potential to significantly improve the efficiency and accuracy of water quality management, while reducing costs and improving environmental sustainability.

**CHAPTER 4**

**SYSTEM SPECIFICATION**

**SOFTWARE REQUIREMENTS:**

Operating System : Windows OS

Programming Language : Python 3.11

Installation Software : Python 3.11

Platform : Visual Studio Code

**HARDWARE REQUIREMENTS:**

System Specification : Intel i5

Clock speed : 3.0 GHz

Ram Size : 512 M

Hard disk capacity : 40 GB

Sensors : DSB18B20 Temperature sensor, pH Sensor

Turbidity Sensor

Arduino UNO

ESP8266 Microcontroller

16\*2 LCD Display

**4.1 PYTHON**

**4.1.1 Introduction**

Python is a widely used general-purpose, high level programming language. It was created by Guido van Rossum in 1991 and further developed by the Python Software Foundation. It was designed with an emphasis on code readability, and its syntax allows programmers to express their concepts in fewer lines of code.

Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc). Python has a simple syntax similar to the English language. Python has syntax that allows developers to write programs with fewer lines than some other programming languages. Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick. Python can be treated in a procedural way, an object-oriented way or a functional way.

Python was designed for readability, and has some similarities to the English language with influence from mathematics. Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses. Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for this purpose.

**4.1.2 Python Library**

A Python library is a collection of related modules. It contains bundles of code that can be used repeatedly in different programs. It makes Python Programming simpler and convenient for the programmer. As we don't need to write the same code A Python library is a collection of related modules. It contains bundles of code that can be used repeatedly in different programs. It makes Python Programming simpler and convenient for the programmer. As we don't need to write the same code again and again for different programs. Python libraries play a very vital role in fields of Machine Learning, Data Science, Data Visualization, etc.

**4.1.3 Working of Python Library**

A Python library is simply a collection of codes or modules of codes that we can use in a program for specific operations. We use libraries so that we don’t need to write the code again in our program that is already available. But how it works. Actually, in the MS Windows environment, the library files have a DLL extension (Dynamic Load Libraries). When we link a library with our program and run that program, the linker automatically searches for that library. It extracts the functionalities of that library and interprets the program accordingly. That’s how we use the methods of a library in our program. We will see further, how we bring in the libraries in our python programs.

**4.1.4 Python Standard Library**

The Python Standard Library contains the exact syntax, semantics, and tokens of Python. It contains built-in modules that provide access to basic system functionality like I/O and some other core modules. Most of the Python Libraries are written in the C programming language. The Python standard library consists of more than 200 core modules. All these works together to make Python a high-level programming language. Python Standard Library plays a very important role. Without it, the programmers can’t have access to the functionalities of Python. But other than this, there are several other libraries in Python that make a programmer’s life easier. Let’s have a look at some of the commonly used libraries:

**Matplotlib:** This library is responsible for plotting numerical data. And that’s why it is used in data analysis. It is also an open-source library and plots high-defined figures like pie charts, histograms, scatterplots, graphs, etc.

**Pandas:** Pandas are an important library for data scientists. It is an open-source machine learning library that provides flexible high-level data structures and a variety of analysis tools. It eases data analysis, data manipulation, and cleaning of data. Pandas support operations like Sorting, Re-indexing, Iteration, Concatenation, Conversion of data, Visualizations, Aggregations, etc.

**Numpy:** The name “Numpy” stands for “Numerical Python”. It is the commonly used library. It is a popular machine learning library that supports large matrices and multi-dimensional data. It consists of in-built mathematical functions for easy computations. Even libraries like TensorFlow use Numpy internally to perform several operations on tensors. Array Interface is one of the key features of this library.

**SciPy:** The name “SciPy” stands for “Scientific Python”. It is an open-source library used for high-level scientific computations. This library is built over an extension of Numpy. It works with Numpy to handle complex computations. While Numpy allows sorting and indexing of array data, the numerical data code is stored in SciPy. It is also widely used by application developers and engineers.

**Scikit-learn:** Scikit-learn is a popular open-source Python library for machine learning, built on top of the NumPy and SciPy libraries. It provides a wide range of machine learning algorithms and tools for tasks such as classification, regression, clustering, and dimensionality reduction, as well as tools for model selection and evaluation. Scikit-learn is known for its ease of use, with a consistent API and comprehensive documentation, making it accessible for both beginners and experienced machine learning practitioners. It also offers a range of features for data pre-processing, feature engineering, and model tuning, making it a powerful tool for building end-to-end machine learning pipelines. Scikit-learn is widely used in industry and academia for a variety of applications, and its large community ensures that it is continually evolving and improving.

**XG Boost:** XGBoost is a popular open-source Python library for gradient boosting that has gained widespread popularity in recent years due to its high performance and scalability. Gradient boosting is a machine learning technique that combines the predictions of multiple weak models to improve accuracy, and XGBoost is a powerful implementation of this technique. It is designed to be fast and efficient, making it ideal for working with large datasets and high-dimensional feature spaces. XGBoost offers a range of features for model training, including early stopping, cross-validation, and regularization, and it can be used for a variety of machine learning tasks, including classification, regression, and ranking. Its popularity has led to its use in many industry applications, including search engines, recommendation systems, and fraud detection. XGBoost is a valuable tool for any data scientist or machine learning practitioner looking to improve their model performance and accuracy.

**Flask:** Flask is a popular open-source Python web framework that is used for developing web applications. It is known for its simplicity, flexibility, and lightweight design, making it an ideal choice for small to medium-sized applications. Flask is built on top of the WSGI toolkit and provides a range of features for building web applications, including URL routing, request handling, and template rendering. Flask is also highly customizable, allowing developers to choose their own libraries and tools for building their applications. It is widely used in industry and has a large and active community, with many third-party extensions and plugins available to extend its functionality. Flask is a valuable tool for anyone looking to build a Python-based web application, whether for personal or professional use.

**4.2 HARDWARE COMPONENTS**

**4.2.1 DS18B20 temperature sensor**

The DS18B20 is a digital temperature sensor that can measure temperatures ranging from -55°C to 125°C with an accuracy of ±0.5°C. It uses a one-wire interface, which means that multiple sensors can be connected to a single data line, making it easy to monitor temperature in different locations. Additionally, the DS18B20 has a unique 64-bit serial code that can be used to identify individual sensors in a network. This sensor is commonly used in a wide range of applications such as environmental monitoring, HVAC systems, and industrial automation, among others. It is a digital temperature sensor that uses a 1-Wire communication interface. This means that it can be connected to a microcontroller or computer using just one wire for data transfer and ground connection. The sensor has a resolution of 9 to 12 bits, which means that it can detect temperature changes as small as 0. 0625°C.It has a wide operating voltage range of 3.0V to 5.5V, making it suitable for use with a variety of microcontrollers and other electronics. This sensor has a variety of applications including industrial temperature monitoring, HVAC systems, weather stations, and home automation systems, among others. It has a waterproof variant which is ideal for use in outdoor applications where the sensor may be exposed to moisture. The sensor can be powered using parasitic power, which means that it can draw power from the data line, eliminating the need for an external power supply.



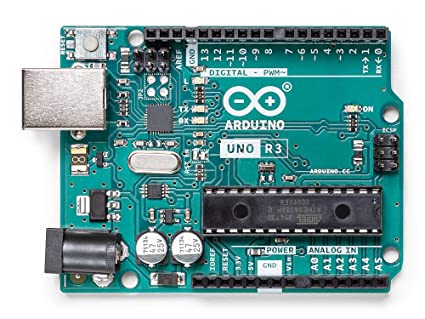
**Figure 4.2.1** DS18B20 Temperature Sensor

**4.2.2 Arduino UNO**

The Arduino UNO is a small and affordable microcontroller board that is commonly used for prototyping and DIY electronics projects. It has a variety of input and output pins that can be used to connect sensors, LEDs, motors, and other electronic components. The board is also compatible with a wide range of shields, which are add-on boards that can provide additional functionality such as wireless communication, motor control, or LCD displays.

The board is powered by either a USB cable or an external power supply and can communicate with a computer or other devices through its USB port. The board also includes a reset button that can be used to restart the program running on the microcontroller. The Arduino UNO is programmed using the Arduino IDE, which is a software development environment that is free and open-source. The IDE includes a code editor, compiler, and uploader, making it easy to write, test, and upload code to the board.

The Arduino UNO has a large and active community of users who share their projects and code online. This makes it easy for beginners to find examples and tutorials that can help them get started with the board. Additionally, the board is often used in educational settings to teach programming and electronics, and there are many resources available for teachers and students. Overall, the Arduino UNO is a powerful and versatile tool that can be used for a wide range of projects and applications.



**Figure 4.2.2** Arduino UNO

**4.2.3 pH Sensor:**

pH sensors are commonly used in water quality monitoring applications to measure the acidity or alkalinity of water. pH is a measure of the concentration of hydrogen ions in a solution, and it is expressed on a scale of 0 to 14, with 7 being neutral, values below 7 indicating acidity, and values above 7 indicating alkalinity.

Water with a pH level outside of the range of 6.5 to 8.5 can be harmful for human consumption and can affect the performance of water treatment systems. pH sensors for water quality monitoring are typically designed to be submerged in water and can measure pH levels with high accuracy and repeatability.

The sensors can be integrated into water quality monitoring systems, which can measure pH levels continuously and provide real-time data to operators. This data can be used to identify changes in water quality, detect potential problems in water treatment systems, and monitor the effectiveness of treatment processes.

Overall, pH sensors are an essential component of water quality monitoring systems and play a crucial role in ensuring the safety and sustainability of our water resources.

****

**Figure 4.2.3** pH Sensor

**4.2.4 Turbidity Sensor:**

Turbidity sensors are used to measure the number of suspended particles in water or other fluids. Turbidity is a measure of the cloudiness or haziness of a fluid, caused by the presence of particles such as sediment, algae, or organic matter. Turbidity can be an important indicator of water quality, as high levels of turbidity can indicate the presence of pollutants or other contaminants in the water. Turbidity sensors work by shining light through the fluid and measuring the amount of light that is scattered or absorbed by the particles. The sensors can be integrated into water quality monitoring systems and can provide continuous measurement of turbidity levels in real-time.

Turbidity sensors are commonly used in a variety of applications, including drinking water treatment, wastewater treatment, aquaculture, and environmental monitoring. They can also be used in research and development applications, such as in the study of sediment transport in rivers and estuaries.

Overall, turbidity sensors are an important tool for water quality monitoring and can help to ensure the safety and sustainability of our water resources.



**Figure 4.2.4** Turbidity Sensor

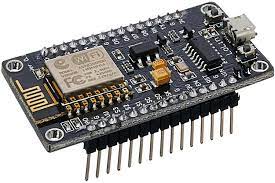
**4.2.5 ESP266 Micro Controller:**

The ESP8266 is a low-cost, highly-integrated Wi-Fi microcontroller designed for Internet of Things (IoT) applications. It is based on the Ten silica L106 32-bit microcontroller and features built-in Wi-Fi connectivity, making it ideal for applications that require wireless connectivity. The ESP8266 also includes a range of input/output (I/O) pins, allowing it to be used with a wide range of sensors, actuators, and other electronic components.

The ESP8266 is often used as a standalone microcontroller or as a Wi-Fi module for other microcontroller boards such as the Arduino. It can be programmed using a variety of programming languages and environments, including C++, Lua, and the Arduino IDE.

The ESP8266 is known for its low power consumption and small form factor, making it ideal for battery-powered or space-constrained applications. It also supports a range of communication protocols, including HTTP, MQTT, and TCP/IP, making it easy to integrate into existing IoT ecosystems.

Overall, the ESP8266 is a powerful and versatile microcontroller that has become a popular choice for IoT applications. Its low cost, built-in Wi-Fi connectivity, and range of I/O pins make it an attractive option for a wide range of projects and applications.



**Figure 4.2.5** ESP8266 Microcontroller

**4.2.6 16x2 LCD Display**

A 16x2 LCD display, also known as a 16-character by 2-line display, is a popular type of alphanumeric display commonly used in various electronic devices. It consists of 16 columns and 2 rows, allowing for the display of up to 32 characters at a time. These displays are characterized by their compact size and simplicity, making them ideal for applications where space is limited. The LCD technology used in these displays typically employs a backlight to enhance visibility in different lighting conditions. With their simple interface and wide compatibility with microcontrollers, 16x2 LCD displays are widely utilized in embedded systems, digital instruments, home appliances, and many other electronic devices, providing clear and concise visual information to users.



**Figure 4.2.6** 16\*2 LCD Display

**CHAPTER 5**

**FEASIBILITTY STUDY**

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential. Feasibility Analysis of a project is performed with the aim to determine that whether it would technically and economically feasible to undertake the project. In other words, is there sufficient resources, technical and economical, available to develop the product. It involves the analysis of the problem and collection of all the relevant information relating to the project such as inputs to the system, processing of the inputs, the output required to be produced by the system and various constraints on the behaviour of the system.

Three key considerations involved in the feasibility analysis are

* Economical Feasibility
* Technical Feasibility
* Operational Feasibility

**5.1 ECONOMICAL FEASIBILITY**

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus, the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased. Whether the project to be undertaken is economically feasible or not is determined by the Economic Feasibility Analysis. It involves the study that if there is sufficient finance available to complete the project. It is quite obvious that in order to develop a product, technical backup alone is not sufficient. Adequate capital is very much necessary for the successful completion of the project. The evaluation and analysis of the potential of a project to support the decision-making process, through the objective and rational identification of its strengths, weaknesses, opportunities and associated risks. In addition, the resources that will be needed to implement the project and an assessment of its chances of success. Moreover, it also has to be determined that whether the capital spent on developing the project would fetch handsome returns or not; otherwise, there is no point in developing the product, if it does not fetch any profit at all. The economic feasibility analysis is not necessarily difficult or expensive, but it must be comprehensive, taking into account all potential challenges and problems. Performing an economic feasibility analysis is an important step in evaluating the costs, benefits, risks, and benefits of a new business. In this case, the software is scheduled to develop with optimized cost in order to make the project economically feasible. The economic feasibility based on the terms of Time, Cost and Man power.

**5.2 TECHNICAL FEASIBILITY**

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system. Technical feasibility is a standard practice for companies to conduct feasibility studies before commencing work on a project. Businesses undertake a technical feasibility study to assess the practicality and viability of a product or service before launching it. Technical feasibility refers to the analysis that whether the technical support required to develop the product is available in sufficiency or not. Moreover, using HTML5, CSS to develop the front-end provide an excellent layout to the project. Thus, it is quite obvious that the project is technically feasible.

**5.3 OPERATIONAL FEASIBILITY**

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

**CHAPTER 6**

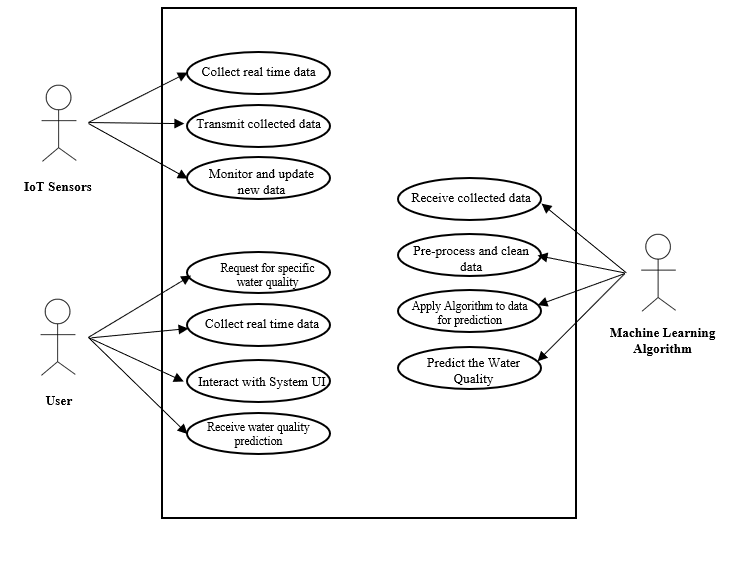
**SYSTEM DESIGN**

**6.1 DATA FLOW DIAGRAM**

**6.1.1 Use Case Diagram**

A use case diagram is a way to summarize details of a system and the users within that system. It is generally shown as a graphic depiction of interactions among different elements in a system. Use case diagrams will specify the events in a system and how those events flow, however, use case diagram does not describe how those events are implemented. A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. Use case diagrams have only 4 major elements: The actors that the system you are describing interacts with, the system itself, the use cases, or services, that the system knows how to perform, and the lines that represent relationships between these elements.

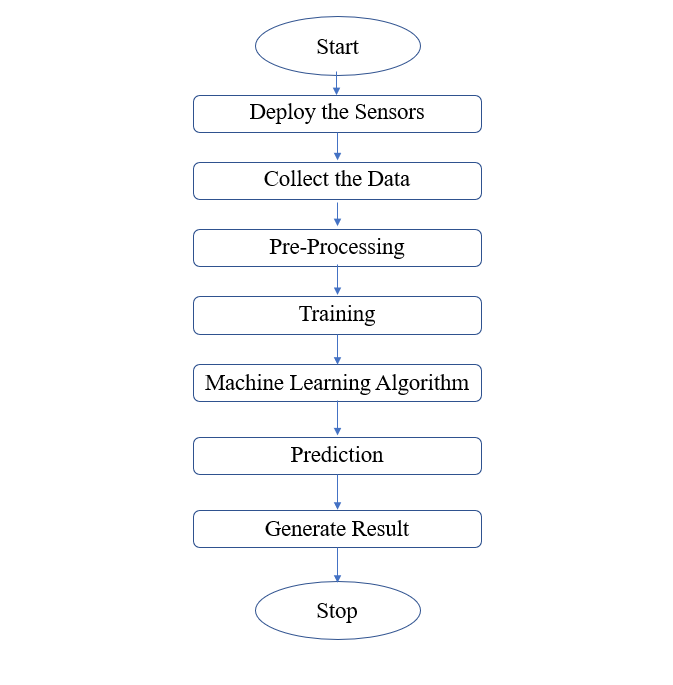
Use case diagram shows a set of use actor and relationship as shown in figure 6.1. A use case diagram contains actors, use cases and interactions or relationships. Use case diagram can be used during analysis to capture the system requirement and to understand how the system should work. A use case is a description of how a person who actually uses that process or system will accomplish a goal. It's typically associated with software systems, but can be used in reference to any process. A description of system behaviour in terms of sequence of action is called use case. A use case should yield an observable result of value to an actor. A use case contains all alternative flows of event related to producing the “observable of result value”.



**Figure 6.1** Use Case Diagram

**6.1.2 Flow Diagram**

It is a step-by-step action taking places process. It is an interaction diagram that emphasis time ordering of messages. It shows the object participating in the interaction by their “Life lines” and the message that they send to each other.



**Figure 6.2** Flow Diagram

**6.2 MODULE DESCRIPTION**

**6.2.1 IoT Sensor Module**

An IoT (Internet of Things) sensor module is a small electronic device that is designed to detect and transmit data about physical phenomena to a central system or network. These modules are typically equipped with sensors that can detect things like temperature, humidity, light, motion, and more, and they can be configured to communicate with a wide range of other devices and systems over a wireless network. IoT sensor modules play a key role in enabling the collection of real-time data from a variety of sources, which can be used to inform decision-making and improve efficiency in a wide range of industries, from manufacturing and logistics to healthcare and smart home automation. With the proliferation of IoT technology, sensor modules are becoming increasingly sophisticated and versatile, offering greater accuracy, longer battery life, and more advanced features like machine learning and edge computing capabilities. IoT sensor modules can be used for a variety of applications, such as environmental monitoring, asset tracking, predictive maintenance, and even predictive healthcare. They can also be integrated into larger IoT ecosystems, such as smart cities or industrial automation systems, to enable real-time monitoring and control of a wide range of devices and processes. As the demand for IoT technology continues to grow, IoT sensor modules are expected to become even more ubiquitous and essential to our daily lives.

**6.2.2 Data Acquisition and Transmission Module**

A data acquisition and transmission module are an essential component of many electronic systems, particularly those involving sensors or measurement devices. This module is responsible for collecting data from various sources, such as sensors, and transmitting it to a central processing unit for analysis. The data acquisition portion typically involves signal conditioning, which refers to the amplification, filtering, and conversion of the signal into a suitable format for processing. The transmission portion involves sending the data to a receiver, which may be located nearby or at a remote location. The choice of transmission medium depends on various factors, such as the distance between the transmitter and receiver, the required data rate, and the level of interference in the environment. Common transmission media include wired connections such as Ethernet and USB, and wireless connections such as Bluetooth, Wi-Fi, and cellular networks. The data acquisition and transmission module are a critical component of many systems, enabling real-time monitoring, control, and analysis of various physical processes.

**6.2.3 Data Storage and Management Module**

This module is responsible for storing the collected data in a database or cloud-based platform, which can be accessed by the machine learning module for analysis and prediction.

**6.2.4 Water Quality Prediction Module**

This module would use the data from the IoT sensors and the trained machine learning models to predict when equipment maintenance is needed. This would include developing appropriate algorithms to predict equipment failures based on historical data, and generating alerts or work orders when maintenance is predicted.

**6.2.5 User Interface Module**

A user interface module (UIM) is a software component that enables users to interact with a system through a graphical interface. The UIM acts as a bridge between the user and the system, providing an intuitive and easy-to-use interface that allows users to perform tasks and access information. UIMs are typically designed with a focus on usability and user experience, with features such as simple navigation, clear and concise labelling, and intuitive controls. A well-designed UIM can greatly enhance the usability of a system and improve user satisfaction. UIMs can be found in a wide range of applications, including desktop software, mobile apps, and web-based platforms, and are an essential component of modern software development.

**CHAPTER 7**

**SYSTEM TESTING**

The System testing is stage in which is the system tested to check whether the system works accurately and efficiently before it was implemented. Testing is vital to the success of the system. System testing makes a logical assumption that if all the parts of the system are correct, the goal will be successfully achieved.

**7.1 TYPES OF TESTING**

• Unit Testing

• Integration Testing

• Validation Testing

• Output Testing

• System Testing

• Performance Testing

• Procedure Testing

**7.1.1 Unit Testing**

Unit testing focuses verification efforts on the smallest unit of software design, the module. The objective in unit testing is to isolate a unit and validate its correctness. A manual approach to unit testing may employ a step-by-step instructional document. However, automation is efficient for achieving this, and enables the many benefits listed in this article. Conversely, if not planned carefully, a careless manual unit test case may execute as an integration test case that involves many software components.

Thus preclude the achievement of most if not all of the goals established for unit testing. This is also known as “Module Testing”. The modules are tested separately. This testing is carried out during programming stage itself.

During unit testing, modules are tested in isolation:

• If all modules were to be tested together it may not be easy to determine which module has the error.

• Unit testing reduces debugging effort several folds.

• Programmers carry out unit testing immediately after they complete the coding of a module.

**7.1.2 Integration Testing**

Data can be lost across the interface; one module can have an adverse effect on others. Integration testing is a systematic testing for constructing program structure. While at the same time conducting tests to uncover errors associated within the interface. Integration testing addresses the issues associated with the dual problems of verification and program construction. After the software has been integrated a set of high order sets are conducted. The objective is to take unit tested modules and combine them test it as a whole. Thus, in the integration-testing step, all the errors uncovered are corrected for the next testing steps.

After different modules of a system have been coded and unit tested

•Big Bang Approach

•Top-Down Approach

•Bottom Approach

•Mixed Approach

**7.1.3 System Testing**

System testing involves:

•Validating a fully developed system against its requirements.

•System testing is done against the requirements in the SRS

•This is the last phase of testing before the product is delivered

•System testing consists of three kinds

•Alpha testing

•Beta testing

•Acceptance testing

•Alpha - System testing is carried out by the test team within the developing

organization.

•Beta - System testing performed by a select group of friendly customers.

•Acceptance - System testing performed by the customer himself to determine whether the system should be accepted or rejected.

**7.1.4 Validation Testing**

The outputs that come out of the system are as a result of the inputs that go in to the system. So, for the correct and the expected outputs the inputs that go in to the system should be correct and proper.

So, this testing is done to check if the inputs are correct and they are validated before it goes in to the system for processing.

**7.1.5 Output Testing**

After performing the validation testing, the next step is output testing of the proposed system, since no system could be useful if it does not produce the required output in the specified format. Asking the users about the format required by them tests the outputs generated or displayed by the system under consideration. Hence the output format is considered in 2ways-one is on screen and another is printed format.

**7.1.6 Performance Testing**

Performance testing is designed to test the run-time performance of software within the context of an integrated system. It requires both hardware and software instrumentation. It is often necessary to measure resource utilization in an exacting fashion.

**7.1.7 Procedure Testing**

Determine the clarity of the documentation on operation and the user of the system by having users do exactly what manual request. In case of this project work system testing and unit testing are mainly used.

**CHAPTER 8**

**CONCLUSION AND FUTURE ENHANCEMENT**

**8.1 CONCLUSION**

A water quality prediction system that utilizes the Internet of Things (IoT) and machine learning can provide valuable insights into the safety and sustainability of our water resources. By collecting and analyzing data from various sensors and sources, such a system can predict changes in water quality and identify potential issues before they become major problems. The integration of IoT devices, such as pH sensors and turbidity sensors, can provide continuous monitoring of water quality in real-time. Machine learning algorithms can then be used to analyze this data and predict future trends or issues. This can help water treatment facilities and environmental agencies make informed decisions about water management and ensure the safety of our water resources. Overall, a water quality prediction system using IoT and machine learning has the potential to revolutionize the way we manage and protect our water resources. It can help to ensure the sustainability of our water supply, protect the health of aquatic life, and promote the overall well-being of our communities.

**8.2 FUTURE ENHANCEMENT**

There are several potential future enhancements for water quality management using internet of things and machine learning:

1. Integration with other systems: The system could be integrated with other water management systems, such as water treatment plants and distribution networks, to create a more comprehensive water management system.

2. Expansion of sensor capabilities: The system could be expanded to include additional sensors for detecting other water quality parameters such as dissolved oxygen, conductivity, and total dissolved solids.

3. Use of advanced machine learning techniques: The system could incorporate advanced machine learning techniques such as deep learning and neural networks to improve the accuracy and reliability of the data.

4. Cloud-based data storage and analysis: The data collected by the system could be stored in the cloud, allowing for easier access and analysis of the data by water management professionals and researchers.

5. Predictive modelling: The system could use machine learning algorithms to predict future water quality trends based on historical data, enabling proactive management and mitigation of potential issues.

6. Autonomous decision-making: The system could be designed to make autonomous decisions based on the water quality data collected, such as adjusting water treatment processes or alerting water management professionals in case of abnormal readings.

Overall, future enhancements of water quality management using internet of things and machine learning could lead to more efficient and effective water management systems, improving the safety and sustainability of our water resources.

**APPENDIX I**

**SOURCE CODE**

**App.py**

from flask import Flask, render\_template, request

import pandas as pd

import numpy as np

from sklearn.preprocessing import StandardScaler

from sklearn.ensemble import RandomForestClassifier

app = Flask(\_name\_)

# Load the dataset

df = pd.read\_csv('water\_potability.csv')

df.drop\_duplicates(inplace=True)

df.dropna(how='all', inplace=True)

idx1 = df.query('Potability == 1')['ph'][df.ph.isna()].index

df.loc[idx1, 'ph'] = df.query('Potability == 1')['ph'][df.ph.notna()].mean()

idx0 = df.query('Potability == 0')['ph'][df.ph.isna()].index

df.loc[idx0,'ph'] = df.query('Potability==0')['ph'][df.ph.notna()].mean()

idx1 = df.query('Potability == 1')['Sulfate'][df.Sulfate.isna()].index

df.loc[idx1, 'Sulfate'] = df.query('Potability == 1')['Sulfate'][df.Sulfate.notna()].mean()

idx0 = df.query('Potability == 0')['Sulfate'][df.Sulfate.isna()].index

df.loc[idx0,'Sulfate'] = df.query('Potability==0')['Sulfate'][df.Sulfate.notna()].mean()

idx1 = df.query('Potability == 1')['Trihalomethanes'][df.Trihalomethanes.isna()].index

df.loc[idx1, 'Trihalomethanes'] = df.query('Potability == 1')['Trihalomethanes'][df.Trihalomethanes.notna()].mean()

idx0=df.query('Potability== 0')['Trihalomethanes'][df.Trihalomethanes.isna()].index

df.loc[idx0,'Trihalomethanes']= df.query('Potability==0')['Trihalomethanes'][df.Trihalomethanes.notna()].mean()

df.loc[~df.ph.between(6.5, 8.5), 'Potability'] = 0

X = df.drop(['Potability'], axis = 1).values

y = df['Potability'].values

sc = StandardScaler()

X = sc.fit\_transform(X)

rf = RandomForestClassifier(n\_estimators=500, min\_samples\_leaf=10, random\_state=42)

rf.fit(X, y)

# Define the home page route

@app.route('/')

def home():

return render\_template('index.html')

# Define the prediction page route

@app.route('/predict', methods=[ 'POST'])

def predict():

# Check if the method used in the request is 'POST'

if request.method == 'POST':

# Get the form data

data = request.form

ph = float(data['ph'])

Hardness = float(data['Hardness'])

Solids = float(data['Solids'])

Chloramines = float(data['Chloramines'])

Sulfate = float(data['Sulfate'])

Conductivity = float(data['Conductivity'])

Organic\_carbon = float(data['Organic\_carbon'])

Trihalomethanes = float(data['Trihalomethanes'])

Turbidity = float(data['Turbidity'])

# Create a dataframe with the form data

input\_data = pd.DataFrame({'ph': [ph], 'Hardness': [Hardness], 'Solids': [Solids],

'Chloramines': [Chloramines], 'Sulfate': [Sulfate], 'Conductivity': [Conductivity],

'Organic\_carbon': [Organic\_carbon], 'Trihalomethanes': [Trihalomethanes],

'Turbidity': [Turbidity]})

# Scale the input data

input\_data = sc.transform(input\_data)

# Make a prediction

prediction = rf.predict(input\_data)

# Return the prediction result

if prediction == 1:

return render\_template('result.html', result=' is potable.')

    else:

return render\_template('result.html', result=' is not potable.')

if \_name\_ == '\_main\_':

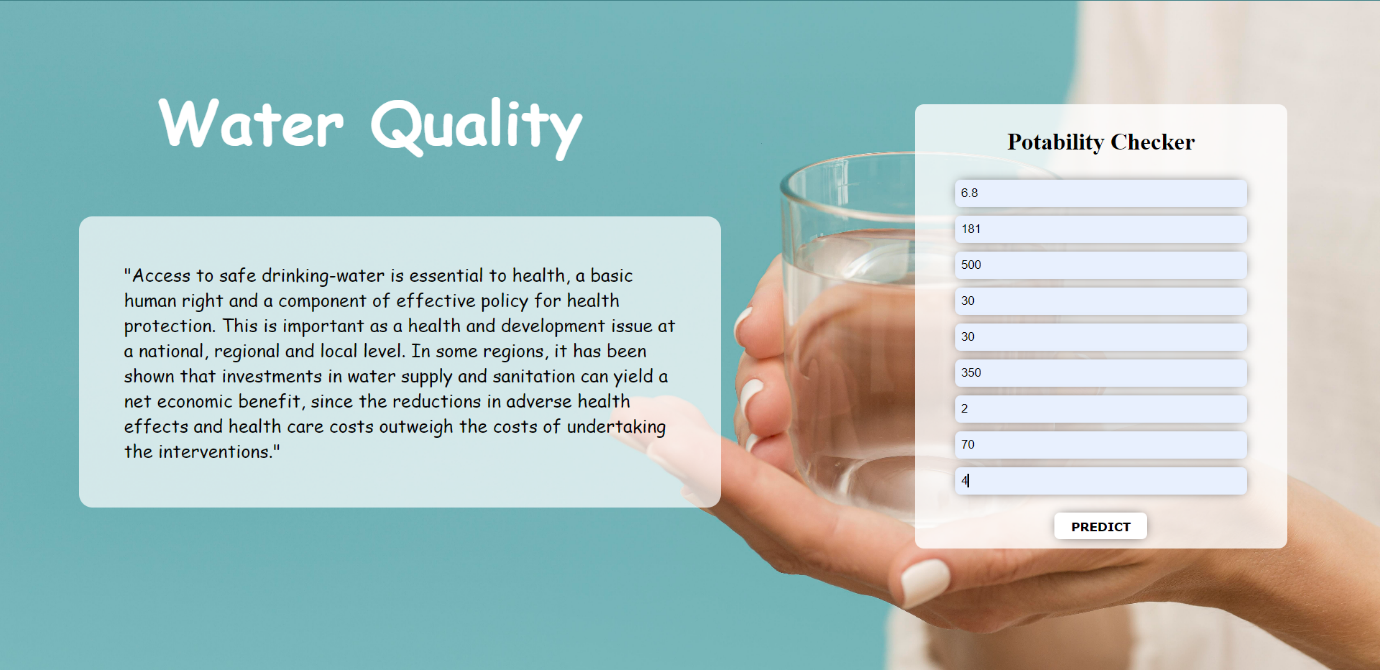
app.run(debug=True)

**APPENDIX II**

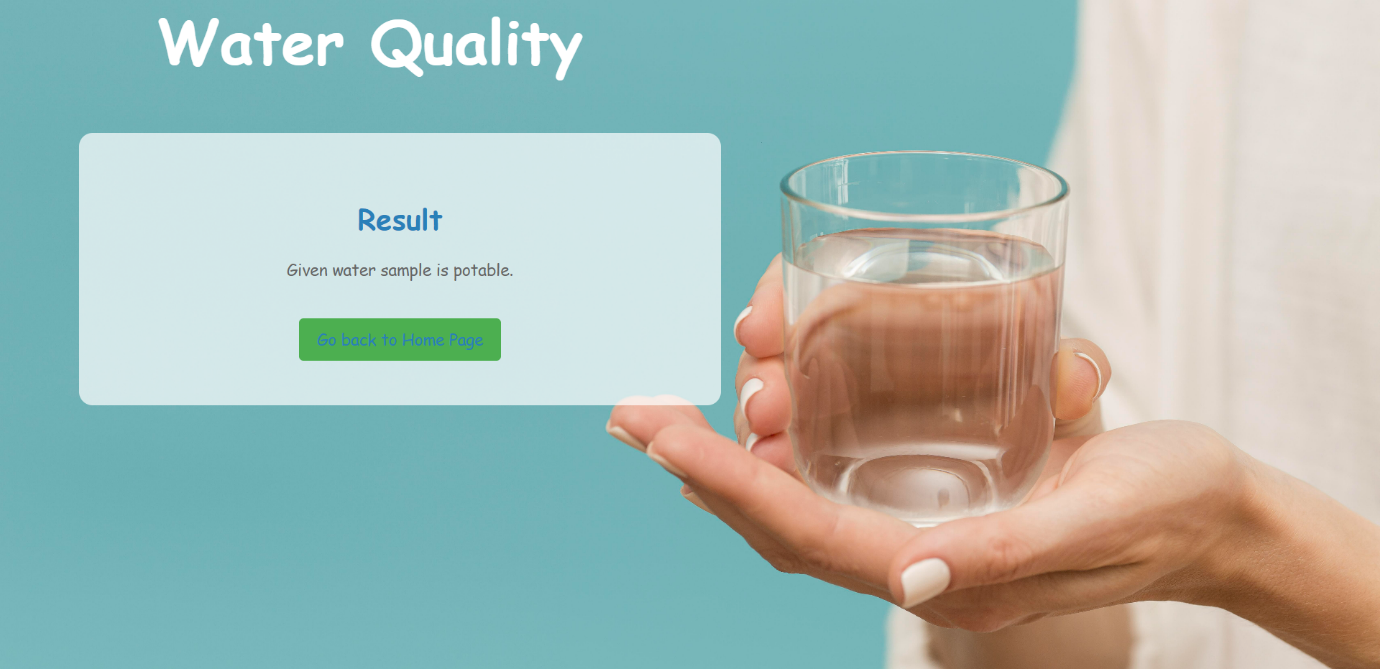
**SCREENSHOTS**

****

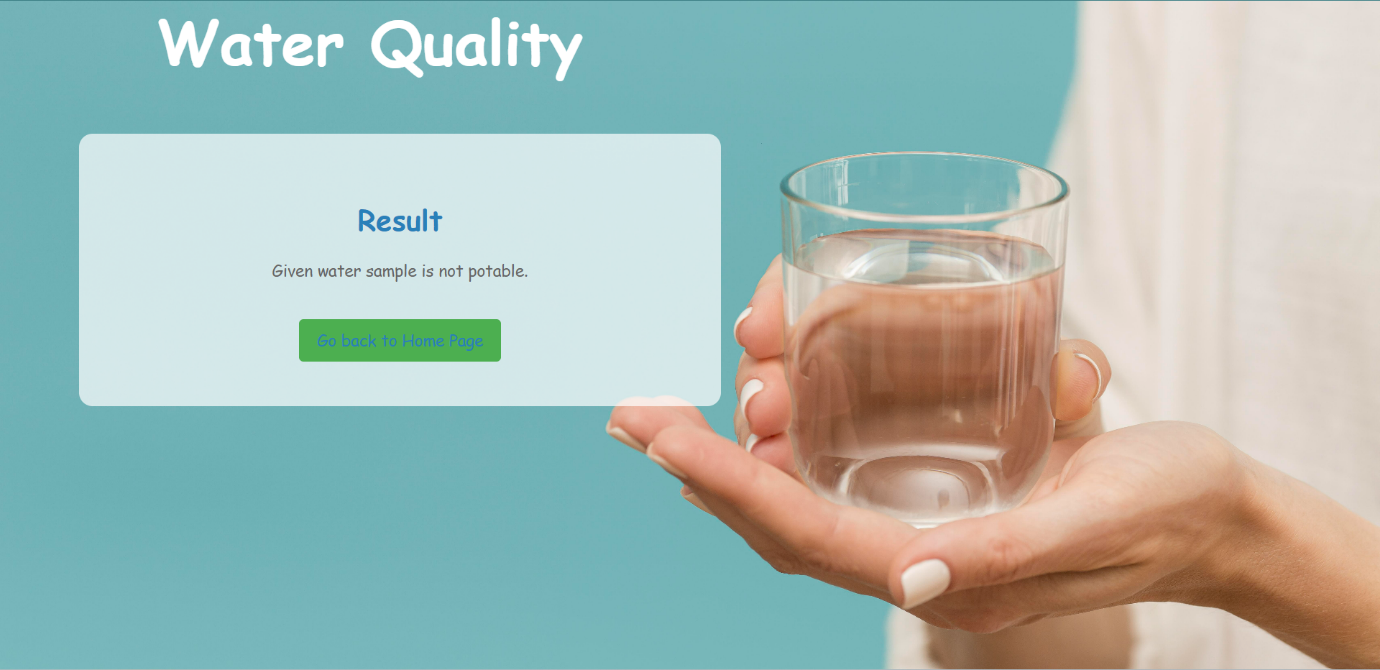
**Figure A.2.1** Index Page

****

**Figure A.2.2** Predict

****

**Figure A.2.3** Result

****

**Figure A.2.4** Result

**REFERENCES**

[1] World Water Assessment Programme (United Nations), *Wastewater : the untapped resource : the United Nations world water development report 2017*.

P. Burek *et al.*, "The Water Futures and Solutions Initiative of IIASA," 2016.

[2] A. Danades, D. Pratama, D. Anggraini, and D. Anggriani, "Comparison of accuracy level K-Nearest Neighbor algorithm and support vector machine algorithm in classification water quality status," in *Proceedings of the 2016 6th International Conference on System Engineering and Technology, ICSET 2016*, Feb. 2017, pp. 137–141. DOI: 10.1109/FIT.2016.7857553.

[3] K. P. Singh, N. Basant, and S. Gupta, "Support vector machines in water quality management," *Analytica Chimica Acta*, vol. 703, no. 2, pp. 152–162, Oct. 2011, DOI: 10.1016/j.aca.2011.07.027.

[4] T. Eitrich and B. Lang, "Efficient optimization of support vector machine learning parameters for unbalanced datasets," *Journal of Computational and Applied Mathematics*, vol. 196, no. 2, pp. 425–436, Nov. 2006, DOI: 10.1016/j.cam.2005.09.009.

[5] Z. Pang and K. Jia, "Designing and accomplishing a multiple water quality monitoring system based on SVM," in *Proceedings - 2013 9th International Conference on Intelligent Information Hiding and Multimedia Signal Processing, IIH-MSP 2013*, 2013, pp. 121–124. DOI: 10.1109/IIH- MSP.2013.39.

[6] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Aug. 2016, vol. 13-17-August-2016, pp. 785–794. DOI: 10.1145/2939672.2939785.

[7] D. N. Myers, "Why monitor water quality?" [Online]. Available: https://[www.epa.gov/assessing](http://www.epa.gov/assessing)

[8] "Artificial Neural Network Modeling of the Water Quality Index Using Land Use Areas as Predictors".

[9] M. Bouamar and M. Ladjal, "Evaluation of the performances of ANN and SVM techniques used in water quality classification."

[10] F. Hassanbaki Garabaghi, "Performance Evaluation of Machine Learning Models with Ensemble Learning approach in Classification of Water Quality Indices Based on Different Subset of Features," 2021, DOI: 10.21203/rs.3.rs- 876980/v1.

[11] L. Li *et al.*, "Interpretable tree-based ensemble model for predicting beach water quality," *Water Research*, vol. 211, Mar. 2022, DOI: 10.1016/j.watres.2022.118078.

[12] N. Nasir *et al.*, "Water quality classification using machine learning algorithms," *Journal of Water Process Engineering*, vol. 48, p. 102920, Aug. 2022, DOI: 10.1016/j.jwpe.2022.102920.

[13] D. Dezfooli, S. M. Hosseini-Moghari, K. Ebrahimi, and S. Araghinejad, "Classification of water quality status based on minimum quality parameters: application of machine learning techniques," *Modeling Earth Systems and Environment*, vol. 4, no. 1, pp. 311–324, Apr. 2018, DOI: 10.1007/s40808-017- 0406-9.

[14] T. H. H. Aldhyani, M. Al-Yaari, H. Alkahtani, and M. Maashi, "Water Quality Prediction Using Artificial Intelligence Algorithms," *Applied Bionics and Biomechanics*, vol. 2020, 2020, DOI: 10.1155/2020/6659314.

[15] "Indian water quality data | Kaggle." https://[www.kaggle.com/datasets/anbarivan/indian-water-quality-data](http://www.kaggle.com/datasets/anbarivan/indian-water-quality-data) (accessed Jul. 01, 2022).

[16] S. C. Dendukuri, D. S. Chandra, M. V. S. Raju, and S. S. Asadi, "Estimation of Water Quality Index By Weighted Arithmetic Water Quality Index Method: A Model Study," *International Journal of Civil Engineering and Technology*, vol. 8, no. 4, pp. 1215–1222, 2017, [Online]. Available: http://www.iaeme.com/IJCIET/index.asp1215<http://www.iaeme.com/IJCIET/i> ssues.asp?JType=IJCIET&VType=8&IType=4<http://www.iaeme.com/IJCIET>

/issues.asp?JType=IJCIET&VType=8&IType=4<http://www.iaeme.com/IJCIE> T/index.asp1216

[17] F. M. Kizar, "A comparison between weighted arithmetic and Canadian methods for a drinking water quality index at selected locations in shatt al- kufa," in *IOP Conference Series: Materials Science and Engineering*, Nov. 2018, vol. 433, no. 1. DOI: 10.1088/1757-899X/433/1/012026.

[18] M. D. Noori, "Comparative analysis of weighted arithmetic and CCME Water Quality Index estimation methods, accuracy and representation," in *IOP Conference Series: Materials Science and Engineering*, Mar. 2020, vol. 737, no. 1. DOI: 10.1088/1757-899X/737/1/012174.

[19] C. K. Ojukwu, G. O. C. Okeah, and P. C. Mmom, "A Comparative Analysis of the Weighted Arithmetic and Canadian Council of Ministers of the Environment Water Quality Indices for Water Sources in Ohaozara, Ebonyi State, Nigeria," *International Journal of Engineering Research & Technology (IJERT)* [*www.ijert.org*,](http://www.ijert.org/) vol. 10, 2021, [Online]. Available: [www.ijert.org](http://www.ijert.org/)

**PUBLICATIONS**

****

****

****

****

****