EFFICIENT WATER QUALITY ANALYSIS AND PREDICTION A PROJECT REPORT

Submitted by

|  |  |
| --- | --- |
| **S.PRIYADARSHINI** | **-(310119104061)** |
| **S.PRIYADHARSHINI** | **-(310119104062)** |
| **S.PUSHPAROJA** | **-(310119104063)** |
| **S.SOWMYA** | **-(310119104076)** |
| **D.PRIYADARSHINI** | **-(310119104060)** |

*in partial fulfillment for the award of the degree of*

# BACHELOR OF ENGINEERING

***in***

**COMPUTER SCIENCE AND ENGINEERING**

# ANAND INSTITUTE OF HIGHER TECHNOLOGY



**ANNA UNIVERSITY: CHENNAI 600 025**

**APRIL 2022**

|  |  |  |
| --- | --- | --- |
| **TABLE OF CONTENT** | | |
| **CHAPTER NO** | **TITLE** | **PAGE NO** |
| **1** | **INTRODUCTION** |  |
|  | 1.1 PROJECT OVERVIEW | 4 |
|  | 1.2 PURPOSE | 4 |
| **2** | **LITERATURE SURVEY** | 5 |
|  | 2.1 EXISTING PROBLEM | 7 |
|  | 2.2 REFERENCES | 8 |
|  | 2.3 PROBLEM SATEMENT DEFINITION | 9 |
| **3** | **IDEATION & PROPOSED SOLUTION** |  |
|  | 3.1 EMPATHY MAP CANVAS | 9 |
|  | 3.2 IDEATION & BRAINSTORMING | 10 |
|  | 3.3 PROPOSED SOLUTION | 13 |
|  | 3.4 PROBLEM SOLUTION FIT | 15 |
| **4** | **REQUIREMENT ANALYSIS** |  |
|  | 4.1 FUNCTIONAL REQUIREMENT | 16 |
|  | 4.2 NON-FUNCTIONAL REQUIREMENTS | 17 |
| **5** | **PROJECT DESIGN** |  |
|  | 5.1 DATA FLOW DIAGRAMS | 19 |
|  | 5.2 SOLUTION & TECHNICAL ARCHITECTURE | 20 |
|  | 5.3 USER STORIES | 23 |
| **6** | **PROJECT PLANNING & SCHEDULING** |  |
|  | 6.1 SPRINT PLANNING & ESTIMATION | 24 |
|  | 6.2 SPRINT DELLIVERY SCHEDULE | 25 |
|  | 6.3 REPORTS FROM **JIRA** | 26 |
| **7** | **CODING & SOLUTIONING (Explain the features added in the project along with code)** |  |

|  |  |  |
| --- | --- | --- |
|  | 7.1 FEATURE 1 | 27 |
|  | 7.2 FEATURE 2 | 29 |
|  | 7.3 DATABASE SCHEMA (if Applicable) |  |
| **8** | **TESTING** |  |
|  | 8.1 TEST CASES | 32 |
|  | 8.2 USER ACCEPTANCE TESTING | 34 |
| **9** | **RESULTS** |  |
|  | 9.1 PERFORMANCE METRICS | 34 |
| **10** | **ADVANTAGES & DISADVANTAGES** | 37 |
| **11** | **CONCLUSION** | 38 |
| **12** | **FUTURE SCOPE** | 38 |
| **13** | **APPENDIX** |  |
|  | SOURCE CODE | 39 |
|  | GITHUB & PROJECT DEMO LINK | 53 |

# INTRODUCTION

Water quality has a direct impact on public health and the environment. Water is used for various practices, such as drinking, agriculture, and industry. Recently, development of water sports and entertainment has greatly helped to attract tourists (Jennings 2007). Among various sources of water supply, due to easy access, rivers have been used more frequently for the development of human societies. Using other water resources such as groundwater and seawater sometimes assisted with problems. For example, using groundwater without suitable recharge will lead to land subsidence

# PROJECT OVERVIEW

With the rapid increase in the volume of data on the [aquatic environment](https://www.sciencedirect.com/topics/earth-and-planetary-sciences/aquatic-environment), machine learning has become an important tool for data analysis, classification, and prediction. Unlike traditional models used in water-related research, data-driven models based on machine learning can efficiently solve more complex nonlinear problems. In water environment research, models and conclusions derived from machine learning have been applied to the construction, monitoring, simulation, evaluation, and optimization of various [water treatment](https://www.sciencedirect.com/topics/earth-and-planetary-sciences/water-purification) and management systems.

Additionally, machine learning can provide solutions for water [pollution](https://www.sciencedirect.com/topics/earth-and-planetary-sciences/pollution-control) [control](https://www.sciencedirect.com/topics/earth-and-planetary-sciences/pollution-control), [water quality improvement](https://www.sciencedirect.com/topics/earth-and-planetary-sciences/water-quality-improvement), and [watershed](https://www.sciencedirect.com/topics/earth-and-planetary-sciences/watershed) ecosystem security management. In this review, we describe the cases in which machine learning algorithms have been applied to evaluate the water quality in different water environments, such as surface water, groundwater, drinking water, sewage, and seawater. Furthermore, we propose possible future applications of machine learning approaches to water environments.

**1.2.PURPOSE**

Hence, rapid industrial development has prompted the decay of water quality at a disturbing rate. Furthermore, infrastructures, with the absence of public awareness, and less hygienic qualities, significantly affect the quality of drinking water . In fact, the consequences of polluted drinking water are so dangerous and can badly affect health, the environment, and infrastructures. As per the United Nations (UN) report, about 1.5 million people die each year because of contaminated water-driven diseases. In developing countries, it is announced that 80% of health problems are caused by contaminated water. Five million deaths and 2.5 billion illnesses are reported annually .Such a mortality rate is higher than deaths resulting from accidents, crimes, and terrorist attacks .

Therefore, it is very important to suggest new approaches to analyze and, if possible, to predict the water quality (WQ). It is recommended to consider the temporal dimension for forecasting the WQ patterns to ensure the monitoring of the seasonal

change of the WQ . However, using a special variation of models together to predict the WQ grants better results than using a single model . There are several methodologies proposed for the prediction and modeling of the WQ. These methodologies include statistical approaches, visual modeling, analyzing algorithms, and predictive algorithms. For the sake of the determination of the correlation and relationship among different water quality parameters, multivariate statistical techniques have been employed . The geostatistical approaches were used for transitional probability, multivariate interpolation, and regression analysis .

Massive increases in population, the industrial revolution, and the use of fertilizers and pesticides have led to serious effects on the WQ environments . Thus, having models for the prediction of the WQ is of great help for monitoring water contamination.

# LITERATURE SURVERY

Many works had been conducted to predict water quality using Machine Learning (ML) approaches. Some researchers used the traditional Machine Learning models, such as Decision Tree [[13](https://encyclopedia.pub/entry/24076#ref_13)][[14](https://encyclopedia.pub/entry/24076#ref_14)], [Artificial Neural Network](https://encyclopedia.pub/entry/27684) [[2](https://encyclopedia.pub/entry/24076#ref_12)][[5](https://encyclopedia.pub/entry/24076#ref_15)][[6](https://encyclopedia.pub/entry/24076#ref_16)][[7](https://encyclopedia.pub/entry/24076#ref_17)], [Support Vector](https://encyclopedia.pub/entry/29353) [Machine](https://encyclopedia.pub/entry/29353) [[8](https://encyclopedia.pub/entry/24076#ref_18)][[9](https://encyclopedia.pub/entry/24076#ref_19)][[0](https://encyclopedia.pub/entry/24076#ref_20)], K-Nearest Neighbors [[21](https://encyclopedia.pub/entry/24076#ref_21)] and Naïve Bayes [[18](https://encyclopedia.pub/entry/24076#ref_18)][[22](https://encyclopedia.pub/entry/24076#ref_22)][[23](https://encyclopedia.pub/entry/24076#ref_23)]. However, in recent years, some researchers are moving towards more advanced ML ensemble models, such as Gradient Boosting and Random Forest [1]

Traditional Machine Learning models, such as the Decision Tree model, are frequently found in the literature and performed well on water quality data. However, decision-tree-based ensemble models, including Random Forest (RF) and Gradient Boosting (GB), always outperform the single decision tree [[4](https://encyclopedia.pub/entry/24076#ref_24)]. Among the reasons for this are its ability to manage both regular attributes and data, not being sensitive to missing values and being highly efficient. Compared to other ML models, decision-tree-based models are more favorable to short-term prediction and may have a quicker calculation speed [[6](https://encyclopedia.pub/entry/24076#ref_26)]. Gakii and Jepkoech [[3](https://encyclopedia.pub/entry/24076#ref_13)] compared five different decision tree classifiers, which are Logistic Model Tree (LMT), J48, Hoeffding tree, Random Forest and Decision Stump. They found that J48 showed the highest accuracy of 94%, while Decision Stump showed the lowest accuracy. Another study by Jeihouni et al. [[4](https://encyclopedia.pub/entry/24076#ref_14)] also compared five decision-tree-based models, which are Random Tree, Random Forest, Ordinary Decision Tree (ODT), Chi- square Automatic [Interaction](https://encyclopedia.pub/entry/6198) Detector and Iterative Dichotomiser 3 (ID3), to determine high water quality zones. They found that ODT and Random Forest produce higher accuracy compared to the other algorithms and the methods are more suitable for continuous datasets.

Another popular Machine Learning model to predict water quality is Artificial Neural Network (ANN). ANN is a remarkable data-driven model that can cater both linear and non-linear associations among output and input data. It is used to treat the non-linearity of water quality data and the uncertainty of contaminant source. However, the performance of ANN can be obstructed if the training data are imbalanced and when all initial weights of the parameter have the same value. In

India, Aradhana and Singh [[8](https://encyclopedia.pub/entry/24076#ref_8)] used ANN algorithms to predict water quality. They found that Lavenberg Marquardt (LM) algorithm has a better performance than the Gradient Descent Adaptive (GDA) algorithm. Abyaneh [[5](https://encyclopedia.pub/entry/24076#ref_5)] used ANN and multivariate linear regression models in his research and found that the ANN model outperforms the MLR model. However, the research only assessed the performance of the ANN model using root-mean-square error (RMSE), coefficient of correlation

(r) and bias values. Although ANN models are the most broadly used, they have a drawback as the prediction power becomes weak if they are used with a small dataset and the testing data are outside the range of the training data [[8](https://encyclopedia.pub/entry/24076#ref_28)].

Support Vector Machine has also been extensively used in water quality studies. Some studies proved that SVM is the best model in predicting water quality compared to other models. A study by Babbar and Babbar [[11](https://encyclopedia.pub/entry/24076#ref_11)] found that Support Vector Machine and Decision Tree are the best classifiers because they have the lowest error rate, which is 0%, in classifying water quality class compared to ANN, Naive Bayes and K-NN classifiers. It also revealed that ML models can quickly determine the water quality class if the data provided represent an accurate representation of domain knowledge. In China, Liu and Lu [[12](https://encyclopedia.pub/entry/24076#ref_12)] developed the SVM and ANN model to predict phosphorus and nitrogen. They found that SVM model achieves a better forecasting accuracy compared to the ANN model. This is because the SVM model optimizes a smaller number of parameters acquired from the principle of structural risk minimization, hence avoiding the occurrence of overtraining data to have a better generalization ability [[12](https://encyclopedia.pub/entry/24076#ref_12)]. This is supported by another study in Eastern Azerbaijan, Iran [[6](https://encyclopedia.pub/entry/24076#ref_16)]. They found that SVM has a better performance compared to the K-Nearest Neighbor algorithm in estimating two water quality parameters, which are total dissolved solid and conductivity. The results showed smaller error and higher R2 than the results attained in Abbasi et al.’s report [[4](https://encyclopedia.pub/entry/24076#ref_4)]. Naïve Bayes has also been widely used for predicting water quality. A study by Vijay and Kamaraj [[2](https://encyclopedia.pub/entry/24076#ref_22)] found that Random Forest and Naïve Bayes produce better accuracy and low classification error compared to the C5.0 classifier. However, traditional ML models, for example, Decision Tree, ANN, Naïve Bayes and SVM, do not perform well. They have some weaknesses, such as a high tendency to be biased and a high variance [[22](https://encyclopedia.pub/entry/24076#ref_22)]. For example, SVM uses the structural risk minimization principle to address overfitting problem in Machine Learning by reducing the model’s complexity and fitting the training data successfully [[9](https://encyclopedia.pub/entry/24076#ref_29)]. Meanwhile, the Bayes model uses prior and posterior probabilities in order to prevent overfitting problems and bias from using only sample information. In ANN, the training process takes a longer time and overfitting problems may occur if there are too many layers, while the prediction error may be affected if there are not enough layers [[30](https://encyclopedia.pub/entry/24076#ref_30)]. Overfitting is a fundamental issue in supervised Machine Learning that prevents the perfect generalization of the model to fit the data observed on the training data, as well as unseen data on the testing set. Hence, overfitting occurs due to the presence of noise, a limited training set size, and classifier complexity [[30](https://encyclopedia.pub/entry/24076#ref_30)]. One of the strategies considered by many previous works to reduce the effects of overfitting is to adopt more advanced methods, such as the

ensemble method.

The ensemble method is a Machine Learning technique that combines several base learners’ decisions to produce a more precise prediction than what can be achieved with having each base learner’s decision [[6](https://encyclopedia.pub/entry/24076#ref_16)]. This method has also gained wide attention among researchers recently. The diversity and accuracy of each base learner are two important features to make the ensemble learners work properly [[7](https://encyclopedia.pub/entry/24076#ref_17)]. The ensemble method ensures the two features in several ways based on its working principle. There are two commonly used ensemble families in Machine Learning, which are bagging and boosting. Both the bagging and boosting

methods provide a higher stability to the classifiers and are good in reducing variance. Boosting can reduce the bias, while bagging can solve the overfitting problem [[1](https://encyclopedia.pub/entry/24076#ref_31)]. A famous ensemble model that uses the bagging algorithm is Random Forest. It is a classification model that uses multiple base models, typically decision trees, on a given subset of data independently and makes decisions based on all models [[5](https://encyclopedia.pub/entry/24076#ref_25)]. It uses feature randomness and bagging when building each individual decision tree to produce an independent forest of trees. Random Forest carries all the advantages of a decision tree with the added effectiveness of using several models [[2](https://encyclopedia.pub/entry/24076#ref_32)]. Another popular ensemble model is Gradient Boosting. Gradient Boosting is a Machine Learning technique that trains multiple weak classifiers, typically decision trees, to create a robust classifier for regression and classification problems. It assembles the model in a stage-wise way similar to other boosting techniques and it generalizes them by optimizing a suitable cost function. In the GB algorithm, incorrectly classified cases for a step are given increased weight during the next step. The advantages of GB are that it has exceptional accuracy in predicting and fast process [[3](https://encyclopedia.pub/entry/24076#ref_33)]. Therefore, advanced models, such as Random Forest and Gradient Boosting, should be employed to cater for the lack of basic ML models.

# EXISITNG PROBLEM

the main problem lies here. For testing the water quality we have to conduct lab tests on the water which is costly and time-consuming as well. So, in this paper, we propose an alternative approach using artificial intelligence to predict water quality. This method uses a significant and easily available water quality index which is set by the WHO(World Health Organisation). The data taken in this paper is taken from the PCPB India which includes 3277 examples of the distinct wellspring. In this paper, WQI(Water Quality Index) is calculated using AI techniques. So in future work, we can integrate this with IoT based framework to study large datasets and to expand our study to a larger scale. By using that it can predict the water quality fast and more accurately than any other IoT framework. That IoT framework system uses some limits for the sensor to check the parameters like ph, Temperature, Turbidity, and so on. And further after reading this parameter pass these readings to the Arduino microcontroller and ZigBee handset for further prediction

# REFERENCES

* + 1. Ling, J.K.B. Water Quality Study and Its Relationship with High Tide and L ow Tide at Kuantan River. Bachelor’s Thesis, Universiti Malaysia Pahang, Gambang, Malaysia, 2010. Available online: [http://umpir.ump.edu.my/id/ep](http://umpir.ump.edu.my/id/eprint/2449/1/JACKY_LING_KUO_BAO.PDF) [rint/2449/1/JACKY\_LING\_KUO\_BAO.PDF](http://umpir.ump.edu.my/id/eprint/2449/1/JACKY_LING_KUO_BAO.PDF) (accessed on 22 February 202 2).
    2. Xu, J.; Gao, X.; Yang, Z.; Xu, T. Trend and Attribution Analysis of Runoff Changes in the Weihe River Basin in the Last 50 Years. Water 2022, 14, 47.
    3. Wahab, M.A.A.; Jamadon, N.K.; Mohmood, A.; Syahir, A. River Pollution Relationship to the National Health Indicated by Under-Five Child Mortalit y Rate: A Case Study in Malaysia. Bioremediat. Sci. Technol. Res. 2015, 3, 20–25.
    4. Abbasi, T.; Abbasi, S.A. Water Quality Indices; Elsevier: Amsterdam, The Netherlands, 2012.
    5. Abyaneh, H.Z. Evaluation of multivariate linear regression and artificial neu ral networks in prediction of water quality parameters. J. Environ. Health Sc i. Eng. 2014, 12, 40.
    6. Alias, S.W.A.N. Ecosystem Health Assessment of Sungai Pengkalan Chepa Basin: Water Quality and Heavy Metal Analysis. Sains Malays. 2020, 49, 1 787–1798.
    7. Al-Badaii, F.; Shuhaimi-Othman, M.; Gasim, M.B. Water quality assessmen t of the Semenyih river, Selangor, Malaysia. J. Chem. 2013, 2013, 871056.
    8. Asadollah, S.B.H.S.; Sharafati, A.; Motta, D.; Yaseen, Z.M. River water qu ality index prediction and uncertainty analysis: A comparative study of mac hine learning models. J. Environ. Chem. Eng. 2021, 9, 104599.
    9. Chen, K.; Chen, H.; Zhou, C.; Huang, Y.; Qi, X.; Shen, R.; Liu, F.; Zuo, M.; Zou, X.; Wang, J. Comparative analysis of surface water quality prediction performance and identification of key water parameters using different mac hine learning models based on big data. Water Res. 2020, 171, 115454.
    10. Lerios, J.L.; Villarica, M.V. Pattern Extraction of Water Quality Prediction Using Machine Learning Algorithms of Water Reservoir. Int. J. Mech. Eng. Robot. Res. 2019, 8, 992–997.
    11. Sengorur, B.; Koklu, R.; Ates, A. Water quality assessment using artificial i ntelligence techniques: SOM and ANN—A case study of Melen River Turk ey. Water Qual. Expo. Health 2015, 7, 469–490.
    12. Aradhana, G.; Singh, N.B. Comparison of Artificial Neural Network algorit hm for water quality prediction of River Ganga. Environ. Res. J. 2014, 8, 55

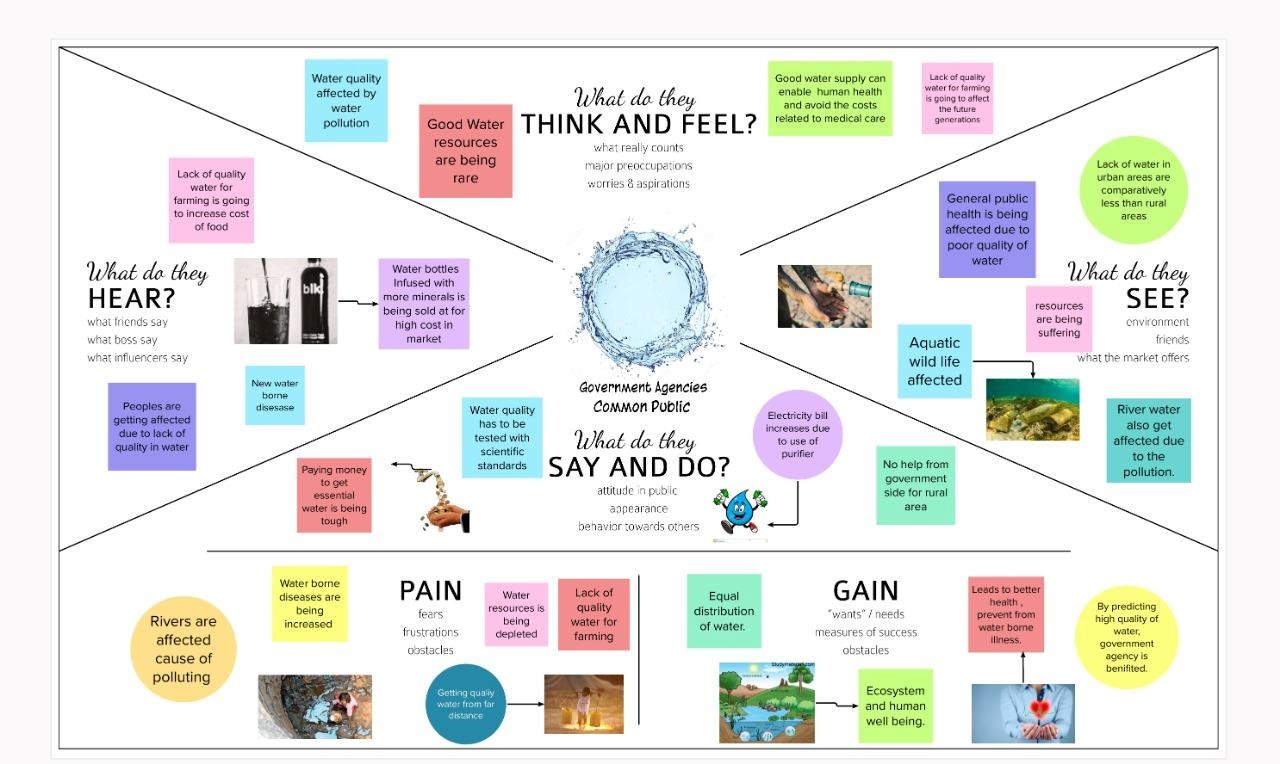
–63.

# PROBLEM STATEMENT DEFINITION

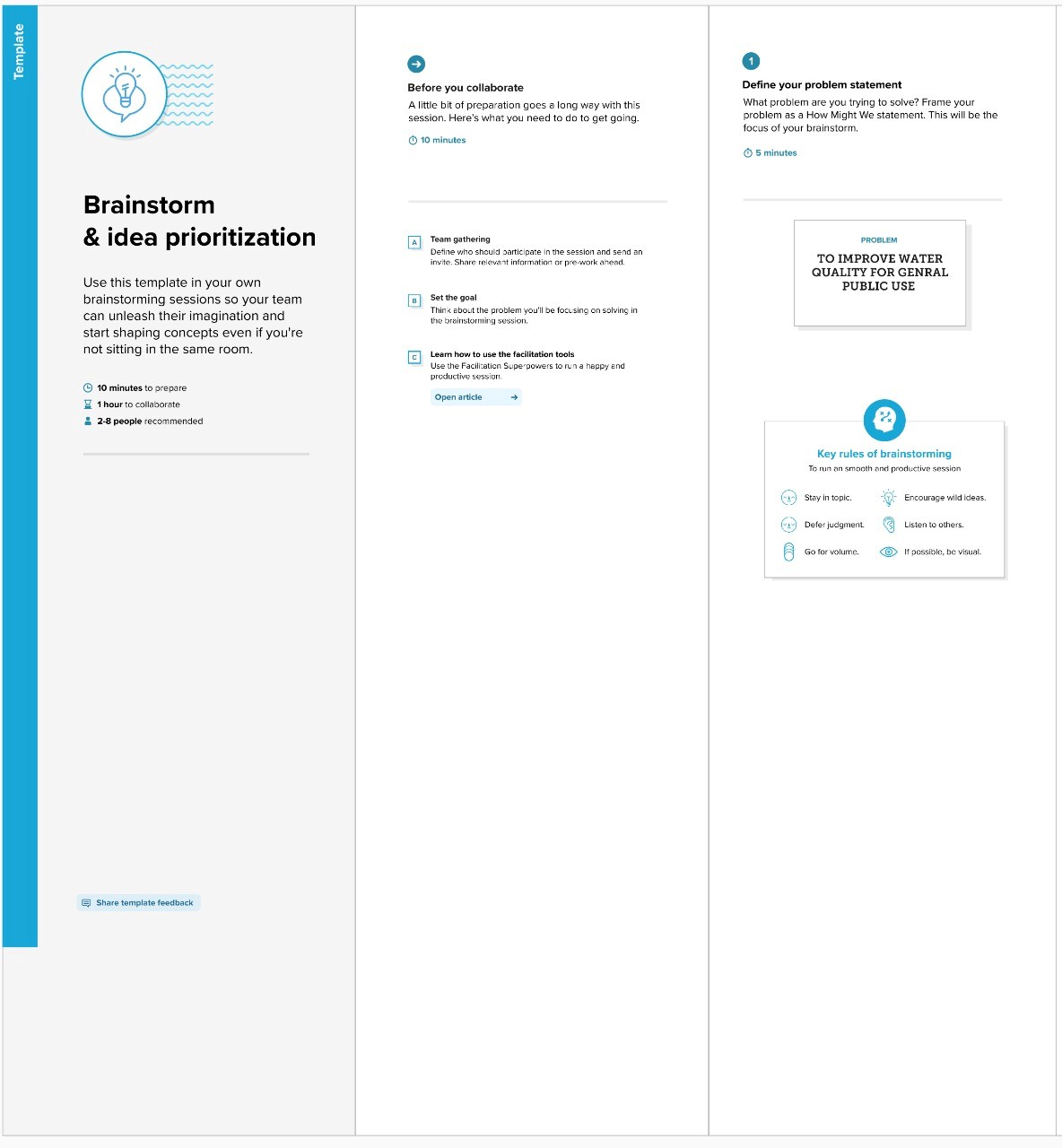
To predict the water safe or not for Access to safe drinking-water is essential to he alth, a basic human right and a component of effective policy for health protection. This is important as a health and development issue at a national, regional and loca l level. In some regions, it has been shown that investments in water supply and sa nitation can yield a net economic benefit, since the reductions in adverse health eff ects and health care costs outweigh the costs of undertaking the interventions.

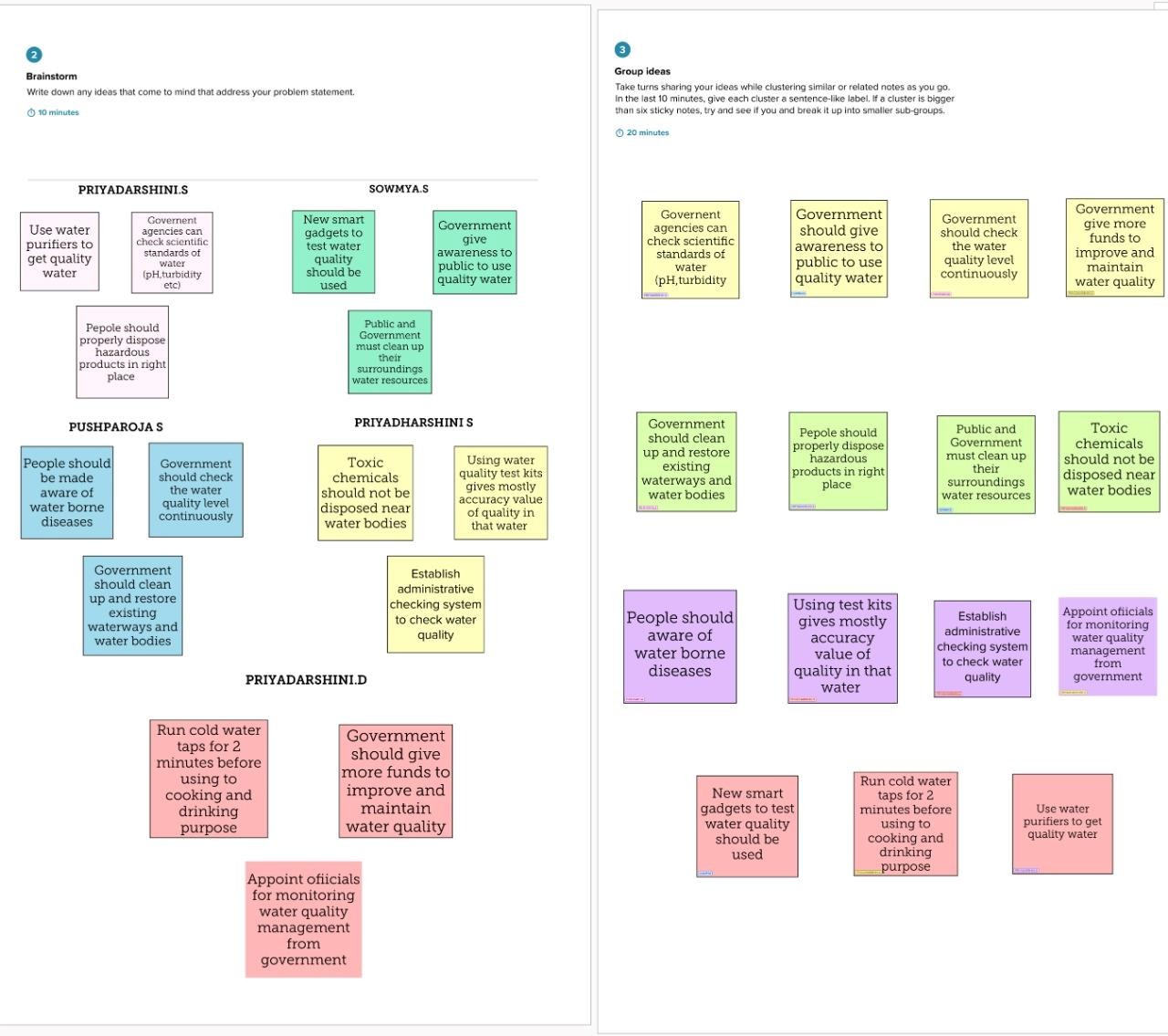
# IDEATION AND PROPOSED SOLUTION

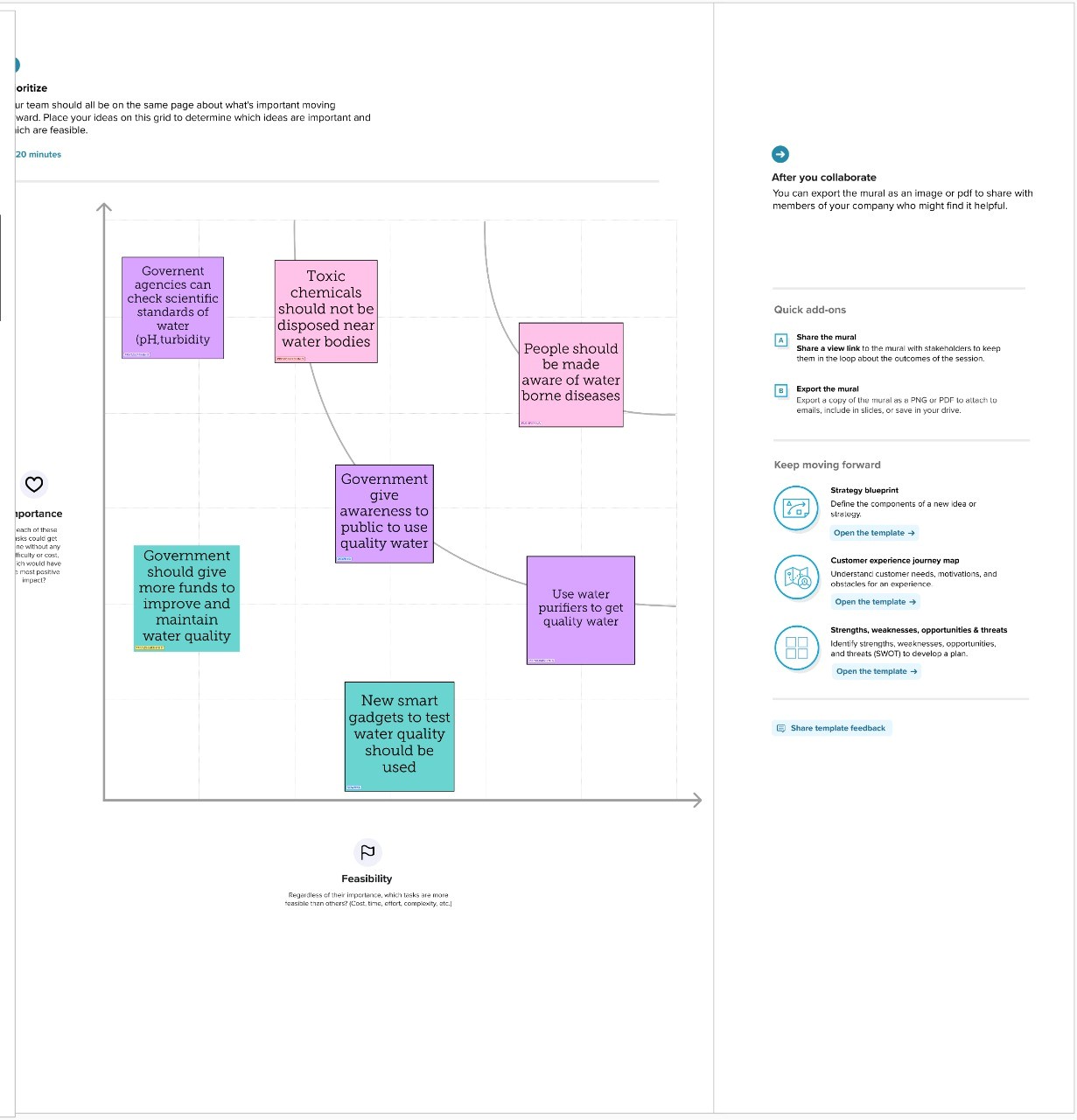
* 1. **EMPATHY MAP CANVAS**



# IDEATION AND BRAINSTORMING





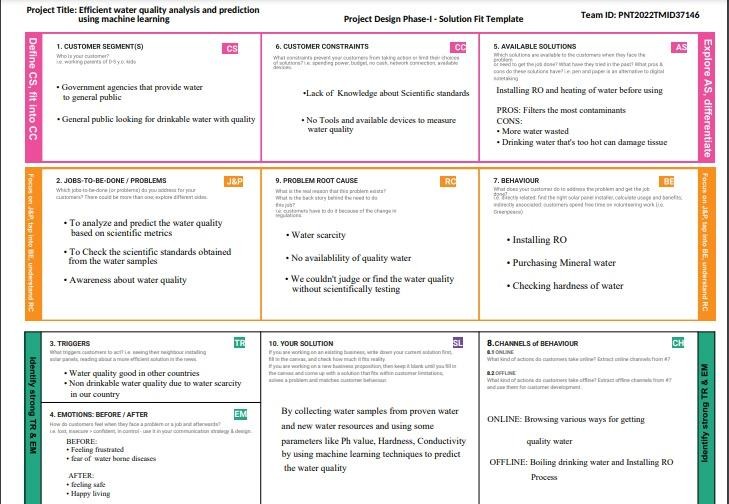


* 1. **PROPOSED SOLUTION**

|  |  |  |
| --- | --- | --- |
| **S NO** | **PARAMETER** | **DESCRIPTION** |
| 1. | Problem Statement (Problem to be solved) | * Water quality prediction using machine learning techniques. Our model predicts the drinkability of the water based parameters such as Ph value, conductivity, and hardness of the water,. |
| 2. | Idea / Solution description | * Water quality prediction model using the principal component analysis followed by decision tree classification. * Firstly, the water quality index (WQI) is calculated using the weighted arithmeticindex method. * Secondly, the principal component analysis (PCA) is applied to the dataset, and the most dominant WQI parametershave been extracted. * Thirdly, to predict the WQI, different regression algorithms are used to the PCAoutput. * Finally, the decision tree classifier model is utilized to classify the   waterquality status. |
| 3. | Novelty / Uniqueness | * In this prediction, the main uniqueness isutilization of PCA and decision tree   classifier model. |

|  |  |  |
| --- | --- | --- |
| 4. | Social Impact / Customer Satisfaction | * This work can demonstrate how setting ofmore stringent water quality objectives can enhance and protect environmental assets of water resources. * This work can aid in justifying the range of water quality metrics set by government initiatives and to minimize further damages in water resources. * This work can help to quickly identify drinkability of water from new   sources. |
| 5. | Business Model (Revenue Model) | * For Analyzing the metrics of each water resource a charge of Rs 100 will be collected. |
| 6. | Scalability of the Solution | * The solution is highly scalable as we use Machine learning technique . * A Automated system can be build to aidthe government, to collect the water metrics and quickly analyze and   predictthe water quality. |

# PROBLEM SOLUTION FIT



1. **REQUIREMENT ANALYSIS**

# FUNCTIONAL REQUIREMENT

|  |  |  |
| --- | --- | --- |
| **FR No.** | **Functional Requirement (Epic)** | **Sub Requirement (Story / Sub-Task)** |
| FR-1 | User Registration | * Registration through Form * Registration through Gmail * Registration through LinkedIN |
| FR-2 | User Confirmation | * Confirmation via Email * Confirmation via OTP |
| FR-3 | Select the water quality testingparameters | * Chemical contamination * Microbial contamination * Physical contamination |
| FR-4 | Physical contamination | * **PH**-is important when disinfecting water with chloride * **EC**-unusually high level may suggest chemical contamination. * **Turbidity**-High turbidity decreases water acceptability. |

|  |  |  |
| --- | --- | --- |
| FR-5 | Chemical contamination | * **Fluoride(1.5 mg/l)**-Fluoride is a naturally- occurring form of the element fluorine, which is sometimes found in groundwater at levels that exceed safe levels. * **Nitrate and Nitrite(50 mg/l)**-In most cases, these compounds aren’t a serious health risk. * **Arsenic(10μg/l)**-The EPA says studies link long- term exposure of arsenic to certain cancers as well as cardiovascular, neurological, and other   conditions.   * **Chlorine(5 mg/L**)-This value is the health-based guideline. Chlorine is often used for water treatment. |
| FR-6 | Microbial contamination | * **E. coli(0 MPN/100 ml**)-Provided indication of contamination by fecal coliforms or other harmful bacteria. This is important because   fecal pollution is the major cause of water- borne diseases in humans. |

* 1. **NON-FUNCTIONAL REQUIREMENT**

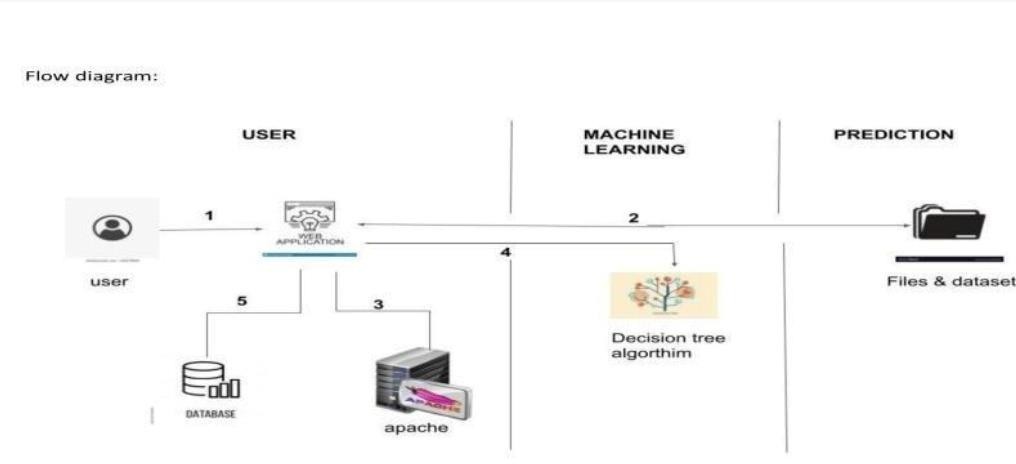
|  |  |  |
| --- | --- | --- |
| **FR No.** | **Non-Functional Requirement** | **Description** |
| NFR-1 | **Usability** | * To improve usability of data provided to water quality exchange,to monitor nutrient record in water. |

|  |  |  |
| --- | --- | --- |
| NFR-2 | **Security** | * The capacity of a population to safeguard sustainable access to adequate quantities of acceptable quality for sustaining livelihoods,human well-being,socio-   economic development,for ensuring  protection against water borne population and water related diseases. |
| NFR-3 | **Reliability** | * **System adequacy and system security**- A hierarchical framework approach to system adequacy evaluation is presented. Adequacy evaluation techniques for each hierarchical   level associated with basic probabilistic indices. |
| NFR-4 | **Performance** | * The presence of certain contaminants in our water can lead to health issues, including gastrointestinal illness, reproductive problems, and neurological disorders.   Infants, young children, pregnant women, the elderly, and people with weakened  immune systems may be especially at risk for illness. |
| NFR-5 | **Availability** | * Low levels of rainfall and high temperatures lead to water deficits . When rainfall is low, there is less water available. When temperatures are high, water   evaporates and so there is less available to  use. Water surpluses are common where rainfall is high and temperatures are lower. |
| NFR-6 | **Scalability** | * Scaling occurs when water has high levels of minerals like calcium carbonate, which can build-up on surfaces**.** Slight scaling can be considered beneficial in that the inside   surfaces of metal pipes become coated with harmless minerals that act as a barrier to  corrosion. |

# PROJECT DESIGN

* 1. **DATA FLOW DIAGRAM**

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



# SOLUTION AND TECHNINCAL ARCHITECTURE

There are basically 10 steps for making our model predict the water quality of the water samples. Those steps are:-

1. *Problem Identification*

In this step, we identify the problem which is solved by our model. So the problem to be solved by our model is water quality prediction using a dataset.

1. *Data Extraction:-*

In this, we extract the data from the internet to train our data and predict the water quality. So for that, we take the CPCB(Central Pollution Control Board India) dataset which contains 3277 instances of 13 different wellsprings which are collected between 2014 to 2020.

1. *Data Exploration:-*

In this step, we analyze the data visually by comparing some parameters of water with the WHO standards of water. It gives a slight overview of the data.

1. *Data Cleaning*

In this step, we clean that data like if there are some missing values in it so we replace them with mean and remove noise from the data..

1. *Data Selection*

In this step, we select the data types and source of the data. The essential goal of data selection is deciding fitting data type, source, and instrument that permit agents to respond to explore questions sufficiently

1. *Data Splitting*

In this step, we divide the dataset into smaller subsets for easing the complexity. Normally, with a two-section split, one section is utilized to assess or test the information and the other to prepare the model.

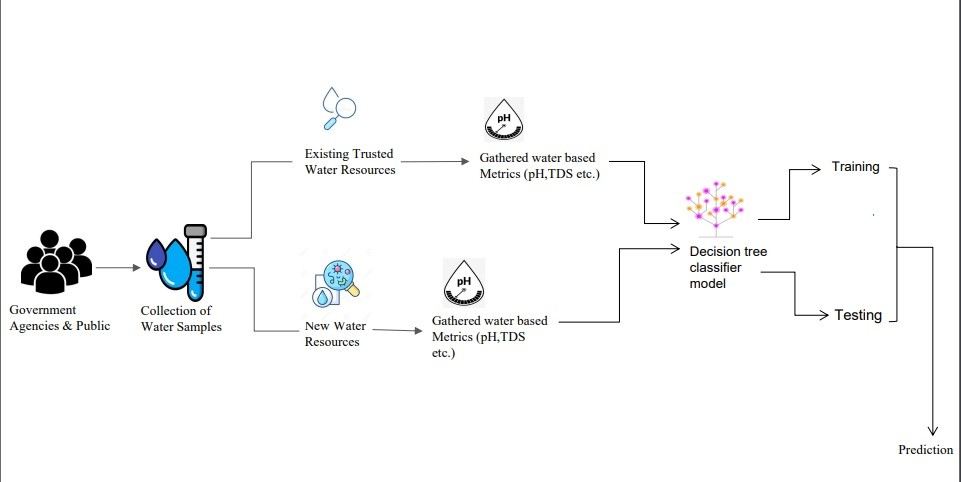
1. *Data Modeling*

In this step, we create a graph of the dataset for visual representation of data for better understanding. A Data Model is this theoretical model that permits the further structure of conceptual models and to set connections between data.

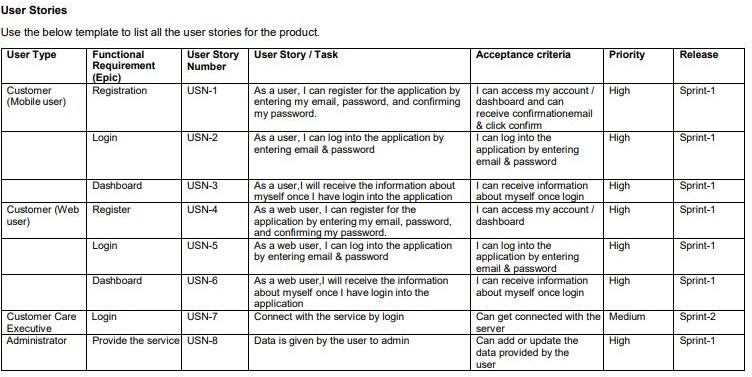
1. *Model Evaluation*

Model Evaluation is a fundamental piece of the model improvement process. In this step, we evaluate our model and check how well our model do in the future.

# SOLUTION ARCHITECTURE

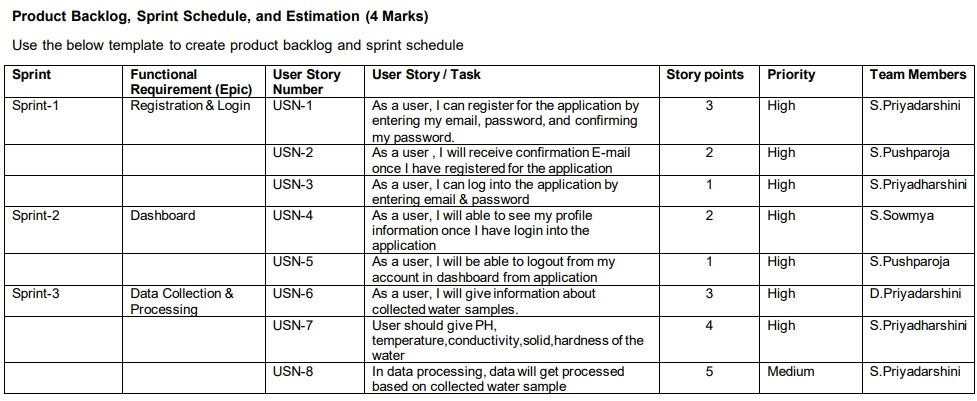


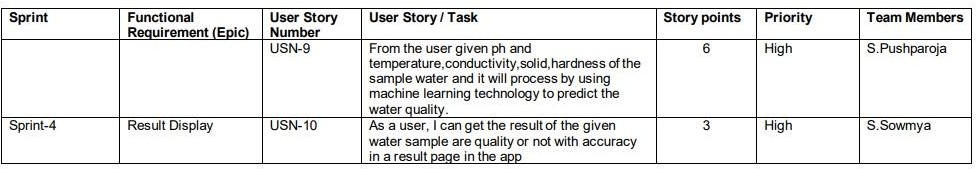
* 1. **USER STORIES**



# PROJECT PLANNING AND SCDULING

* 1. **SPRINT PLANNING AND ESTIMATION**

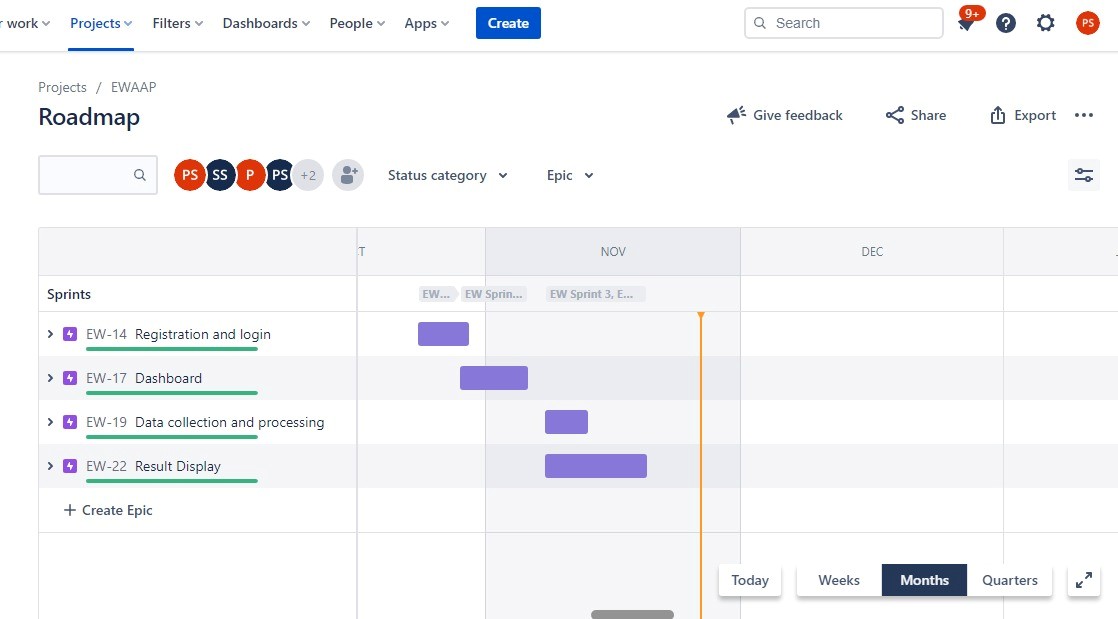




# SPRINT SCHEDULE



* 1. **REPORTS FROM JIRA**

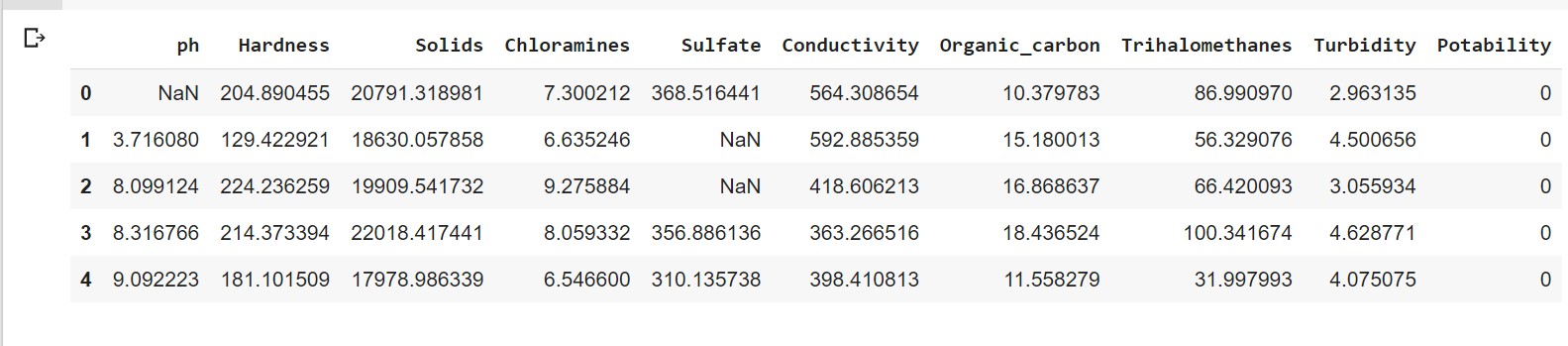


# CODING AND SOLUTIONS

* 1. **FEATURE 1**

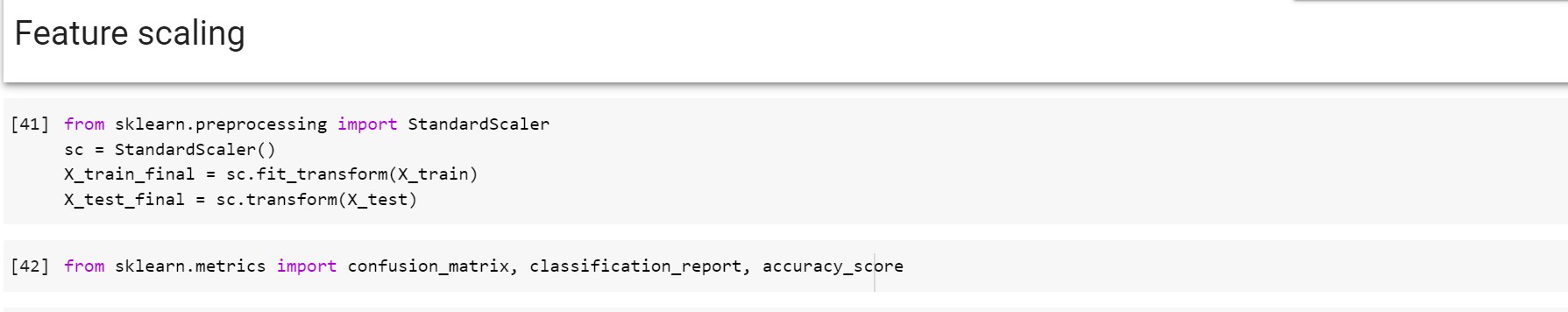
# Data collection and creation

Data mining techniques require domain knowledge in order to generate predictions. For water quality applications, it is vital to understand how various water quality parameters influence water quality. This information can come from a domain expert or historical data collections. For the forecasting task, two types of data sets were used: a carefully created huge synthetic data set and an available real data set



# Data Preprocessing

The processing phase is very important in data analysis to improve the data quality. In this phase, the WQI has been calculated from the most significant parameters of the dataset. Then, water samples have been classified on the basis of the WQI values. For obtaining superior accuracy, the -score method has been used as a data normalization technique.



# Water Quality Index Calculation

To measure water quality, WQI is used to be calculated using various parameters that significantly affect WQ [40–42]. In this study, a published dataset is considered to test the proposed model, and seven significant water quality parameters are included. The WQI has been calculated using the following formula:

where: is the total number of parameters included in the WQI calculations is the quality rating scale for each parameter calculated by equation ([2](https://www.hindawi.com/journals/abb/2020/6659314/#EEq1)) below, and is the unit weight for each parameter calculated by equation ([3](https://www.hindawi.com/journals/abb/2020/6659314/#EEq2)).



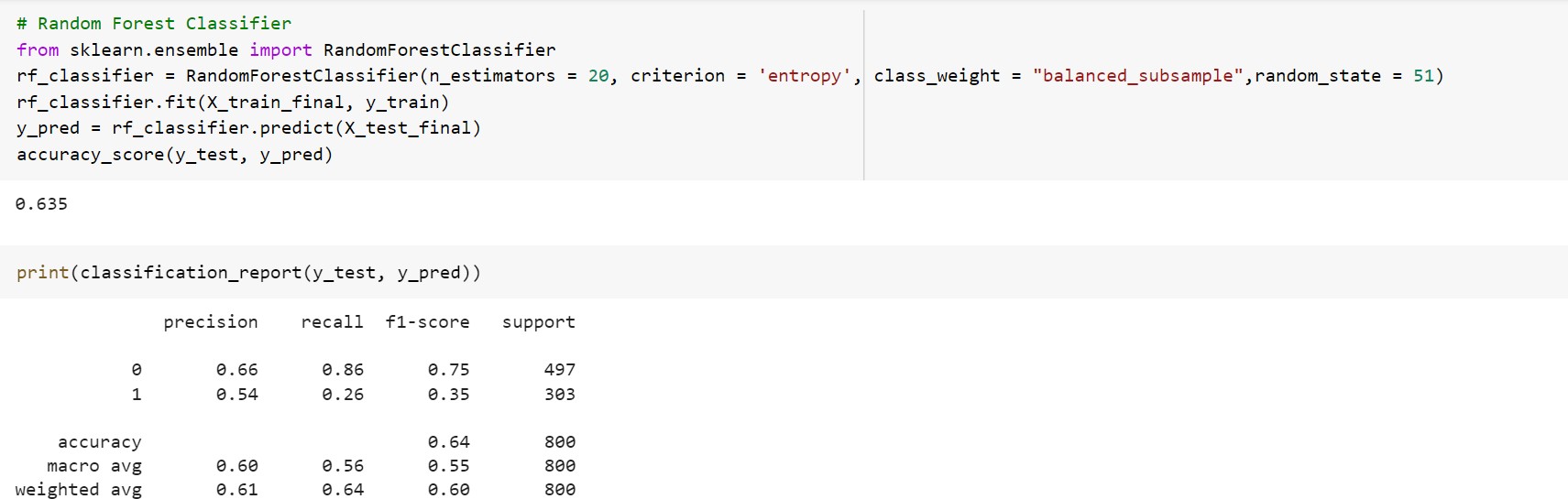
where: is the measured value of parameter in the tested water samples is the ideal value of parameter in pure water (0 for all parameters except and ), and is the recommended standard value of parameter (as shown in Table [1](https://www.hindawi.com/journals/abb/2020/6659314/tab1/))

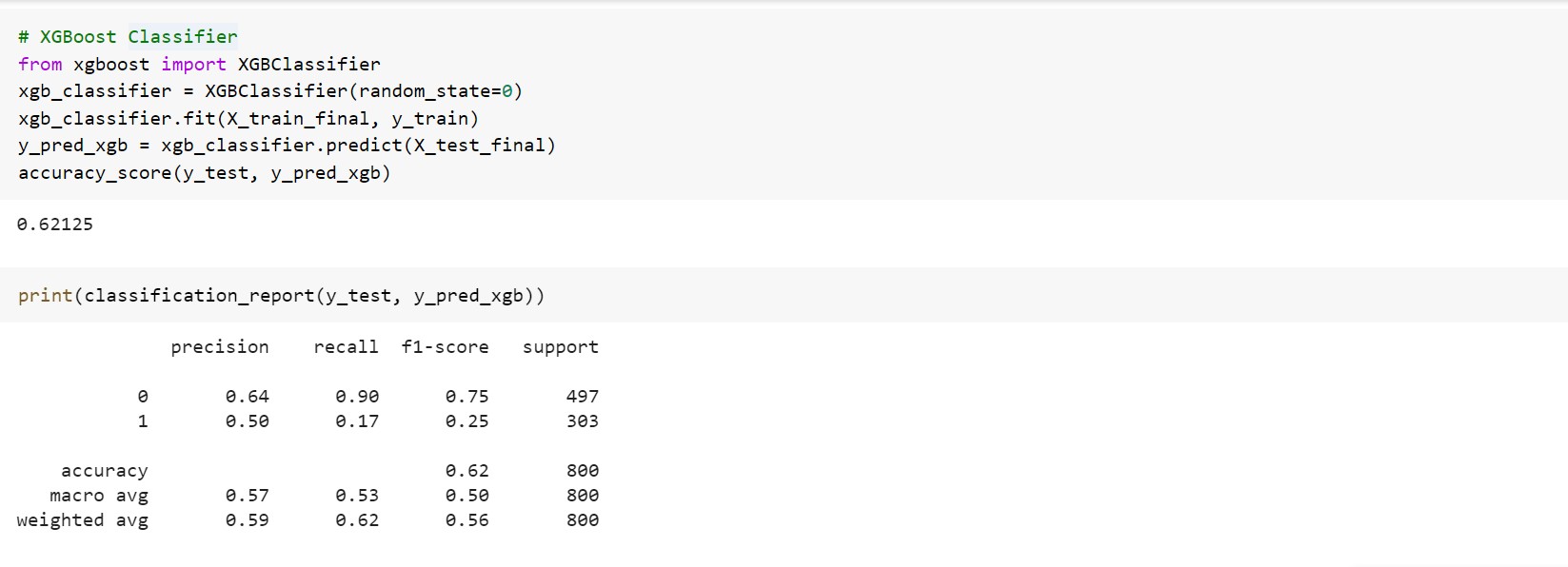


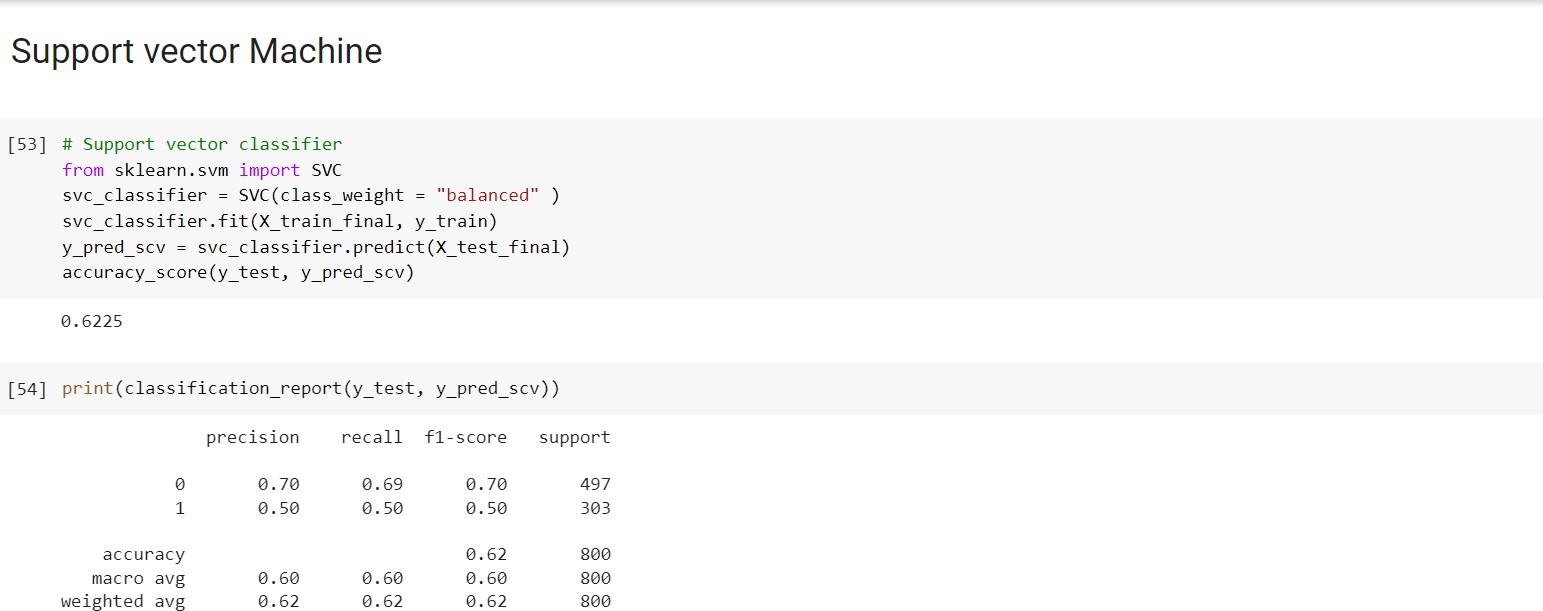
# FEATURE 2

Performance Measures Results True Positives (TP) are when the model predicts the positive class properly. True Negatives (TN) is one of the components of a confusion matrix designed to demonstrate how classification algorithms work. Positive outcomes that the model predicted incorrectly are known as False Positives (FP). False Negatives (FN) are negative outcomes that the model predicts negative class. Accuracy is the most basic and intuitive performance metric, consisting of the ratio of successfully predicted observations to total observations.

Accuracy = TP+TN/(TP+FP+FN+TN)

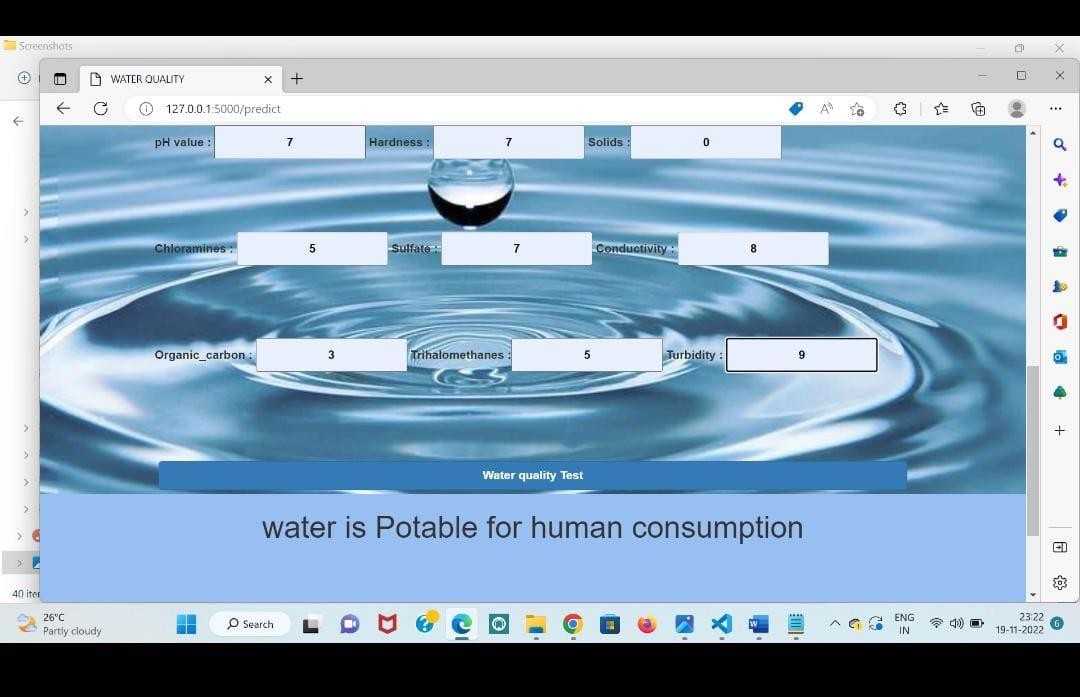




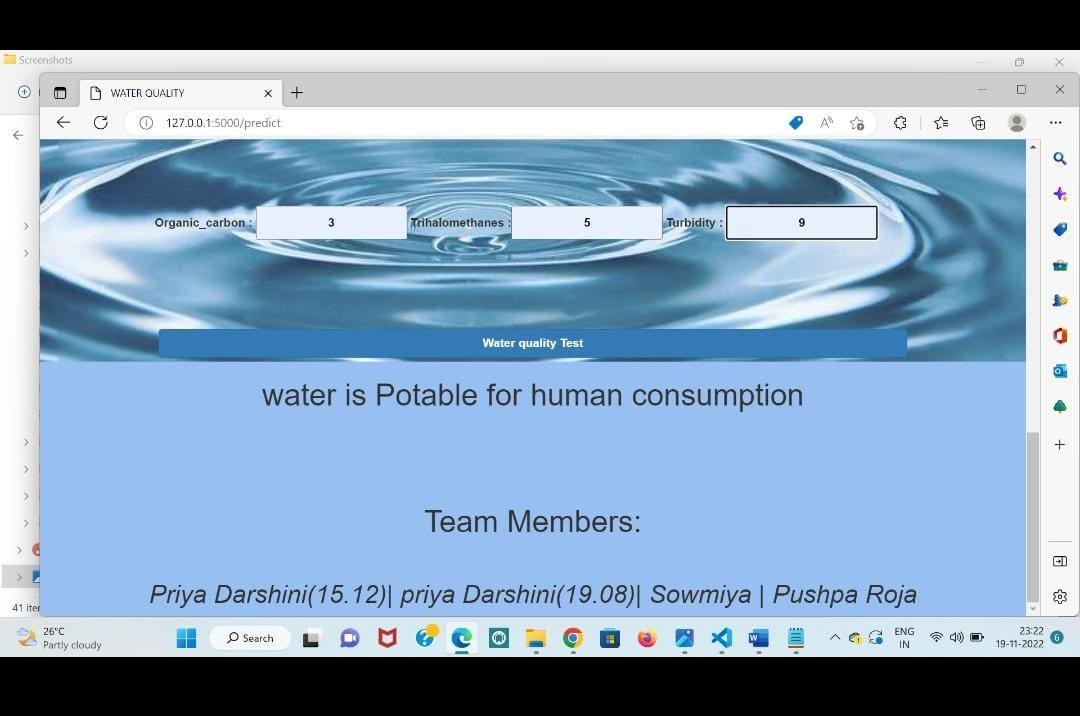
The SVM model was developed in 1995 by Corinna Cortes and Vapnik. It has several unique benefits in solving small samples, and nonlinear and high-dimensional pattern recognition. It can be extended to function in the simulation of other machine learning problems. It uses the hyperplane to separate the points of the input vectors and finds the needed coefficients. The best hyperplane is the line with the largest margin, which is meant the distance between the hyperplane and the nearest input objects. The input points defined in the hyperplane are called *support vectors*. In this work, the linear SVM model along with the Gaussian radial basis function (equation ([17](https://www.hindawi.com/journals/abb/2020/6659314/#EEq4))) is used to classify the tested water samples based on their quality.

# TESTING

* 1. **TEST CASES 1**



# TEST CASE 2



* 1. **USER ACCEPTANCE TESTING**

# Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the [ProductName] project at the time of the release to User Acceptance Testing (UAT).

# Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Resolution** | **Severity 1** | **Severity 2** | **Severity 3** | **Severity 4** | **Subtotal** |
| By Design | 10 | 4 | 2 | 3 | 20 |
| Duplicate | 1 | 0 | 3 | 0 | 4 |
| External | 2 | 3 | 0 | 1 | 6 |
| Fixed | 11 | 2 | 4 | 20 | 37 |
| Not Reproduced | 0 | 0 | 1 | 0 | 1 |
| Skipped | 0 | 0 | 1 | 1 | 2 |
| Won't Fix | 0 | 5 | 2 | 1 | 8 |
| Totals | 24 | 14 | 13 | 26 | 77 |

# Test Case Analysis

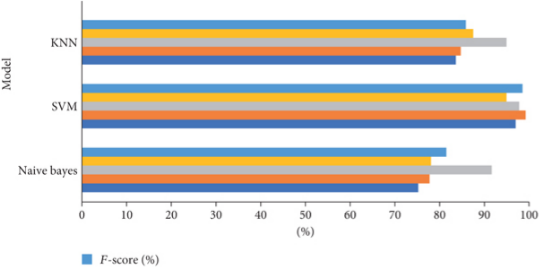
This report shows the number of test cases that have passed, failed, and untested

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Section** | **Total Cases** | **Not Tested** | **Fail** | **Pass** |
| Print Engine | 7 | 0 | 0 | 7 |
| Client Application | 51 | 0 | 0 | 51 |
| Security | 2 | 0 | 0 | 2 |
| Outsource Shipping | 3 | 0 | 0 | 3 |
| Exception Reporting | 9 | 0 | 0 | 9 |
| Final Report Output | 4 | 0 | 0 | 4 |
| Version Control | 2 | 0 | 0 | 2 |

# RESULT

* 1. **PERFORMANCE METRICS**

For validating the developed model, the dataset has been divided into 70% training and 30% testing subsets. While the ANN and LSTM models were used to predict the WQI, the SVM, KNN, and Naive Bayes were utilized for the water quality classification prediction



## SO ,WE ARE GOING TO USE SVC

Performance Measures Results True Positives (TP) are when the model predicts the positive class properly. True Negatives (TN) is one of the components of a confusion matrix designed to demonstrate how classification algorithms work. Positive outcomes that the model predicted incorrectly are known as False Positives (FP). False Negatives (FN) are negative outcomes that the model predicts negative class. Accuracy is the most basic and intuitive performance metric, consisting of the ratio of successfully predicted observations to total observations.

## Accuracy = TP+TN/(TP+FP+FN+TN)

**Table 1. Comparison of algorithms SN.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| SN. | Algorithm | Type | ACCURACY | Precision | Recall f1-Score |
| 1 | RANDOM FOREST | 58.5 | 0.42 | 0.38 | 0.40 |
| 2 | XGBOOST | 61.7 | 0.43 | 0.12 | 0.18 |

# ADVANTAGES

Whether it be for groundwater, surface water or open water, there are a number of reasons why it is important for you to undertake regular water quality testing. If you’re wanting to create a solid foundation on which to build a broader water management plan, then investing in water quality testing should be your first point of action. This testing will also allow you to adhere to strict permit regulations and be in compliance with Australian laws.

Identifying the health of your water will help you to discover where it may need some help. Ultimately, finding a source of pollution, or remaining proactive with your monitoring will enable you to save money in the long term. The more information that you can obtain will assist you with your decision on what product you may need to improve the condition of your water. Simply guessing and buying products based on a hunch or a general trend is ill-advised, as each body of water has unique properties that can only be discovered through testing.

Measuring the amount of dissolved oxygen in your water is another important advantage of water quality testing, as typically the less oxygen, the higher the water temperature, resulting in a more harmful environment for aquatic life. These levels do fluctuate slightly across the seasons, but regular monitoring of your water quality will allow you to discover trends over time, and whether there are other factors that may be contributing to the results you discover.

# DISADVANTAGES

Training necessary Somewhat difficult to manage over time and with large data sets Requires manual operation to submit data, some configuration required

Costly, usually only feasible under Exchange Network grants Technical expertise and network server required

Requires manual operation to submit data Cannot respond to data queries from other nodes, and therefore cannot interact with the Exchange Network Technical expertise and network server required

# CONCLUSION

Potability determines the quality of water, which is one of the most important resources for existence. Traditionally, testing water quality required an expensive and time-consuming lab analysis. This study looked into an alternative machine learning method for predicting water quality using only a few simple water quality criteria. To estimate, a set of representative supervised machine learning algorithms was used. It would detect water of bad quality before it was released for consumption and notify the appropriate authorities It will hopefully reduce the number of individuals who drink low-quality water, lowering the risk of diseases like typhoid and diarrhea. In this case, using a prescriptive analysis based on projected values would result in future capabilities to assist decision and policy makers.

# 12 . FUTURE SCOPE

Machine learning has been widely used as a powerful tool to solve problems in the water environment because it can be applied to predict water quality, optimize water resource allocation, manage water resource shortages, etc. Despite this, several challenges remain in fully applying machine learning approaches in this field to evaluate water quality: (1) Machine learning is usually dependent on large amounts of high-quality data. Obtaining sufficient data with high accuracy in [water treatment](https://www.sciencedirect.com/topics/earth-and-planetary-sciences/water-purification) and management systems is often difficult owing to the cost or technology limitations. (2) As the conditions in real water treatment and management systems can be extremely complex, the current algorithms may only be applied to specific systems, which hinders the wide application of machine learning approaches. (3) The implementation of machine learning algorithms in practical applications requires researchers to have certain professional background knowledge.

To overcome the above-mentioned challenges, the following aspects should be considered in future research and engineering practices: (1) More advanced sensors, including soft sensors, should be developed and applied in water quality monitoring to collect sufficiently accurate data to facilitate the application of machine learning approaches. (2) The [feasibility](https://www.sciencedirect.com/topics/earth-and-planetary-sciences/feasibility) and reliability of the algorithms should be improved, and more universal algorithms and models should be developed according to the water treatment and management requirements. (3) Interdisciplinary talent with knowledge in different fields should be trained to develop more advanced machine learning techniques and apply them in engineering practices.

**13. APPENDIX**

## SOURCE CODE

import pandas as pd import numpy as np

import matplotlib.pyplot as plt import seaborn as sns

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

df.shape df.info() df.describe()

sns.countplot(x='Potability',data=df ) df["Potability"].value\_counts()

print(f"0 : {round(1998 /3276 \* 100 , 2)}") print(f"1 : {round(1278 /3276 \* 100 , 2)}")

#EDA

df.isnull().sum()

for feature in df.columns:

if df[feature].isnull().sum()>0:

print(f"{feature} : {round(df[feature].isnull().mean(),4)\*100}%") ## Fill missing values with median

for feature in df.columns: df[feature].fillna(df[feature].median() , inplace = True)

## find dublicate rows in dataset duplicate = df[df.duplicated()] duplicate

for i in df.columns:

print(f" {i} : {len(df[i].unique())}")

for feature in df.columns:

if feature == "Potability": pass

else:

bar = sns.histplot(df[feature] , kde\_kws = {'bw' : 1} , ) bar.legend(["Skewness: {:0.2f}".format(df[feature].skew())]) plt.xlabel(feature)

plt.ylabel("Probability density") plt.title(feature)

plt.show()

# we don't have missing values in our dataset so we can skip if condition for feature in df.columns:

if 0 in df[feature].unique():# because log 0 is not defined thats why we are using this condition or we can also use log1p pass

else:

df.boxplot(column=feature) plt.ylabel(feature) plt.title(feature)

plt.show()

# removing outliers Q1 = df.quantile(0.25) Q3 = df.quantile(0.75) IQR = Q3 - Q1

print(IQR)

df = df[~((df < (Q1 - 1.5 \* IQR)) |(df > (Q3 + 1.5 \* IQR))).any(axis=1)] df.shape

df["Potability"].value\_counts()

## Correlation plt.figure(figsize=(25,25))

ax = sns.heatmap(df.corr(), cmap = "coolwarm", annot=True, linewidth=2)

Multivariate analysis

sns.pairplot(df , height=10 , size = 5 , hue = "Potability" ) ### splitting data into x and y

X = df.iloc[: , : -1]

y = df.iloc[ : , -1]

# split dataset into train and test

from sklearn.model\_selection import train\_test\_split

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size = 0.3, random\_state= 5)

#from collections import Counter

#from imblearn.over\_sampling import SMOTE

#counter = Counter(y\_train) #print(f"before oversampling: {counter}") #smt = SMOTE()

#X\_train , y\_train = smt.fit\_resample(X\_train , y\_train) #counter = Counter(y\_train)

#print(f"after oversampling : {counter}")

Feature scaling

from sklearn.preprocessing import StandardScaler sc = StandardScaler()

X\_train\_final = sc.fit\_transform(X\_train) X\_test\_final = sc.transform(X\_test)

from sklearn.metrics import confusion\_matrix, classification\_report, accuracy\_score

# Random Forest Classifier

from sklearn.ensemble import RandomForestClassifier

rf\_classifier = RandomForestClassifier(n\_estimators = 20, criterion = 'entropy', class\_weight = "balanced\_subsample",random\_stat e = 51)

rf\_classifier.fit(X\_train\_final, y\_train)

y\_pred = rf\_classifier.predict(X\_test\_final) accuracy\_score(y\_test, y\_pred)

print(classification\_report(y\_test, y\_pred))

# XGBoost Classifier

from xgboost import XGBClassifier xgb\_classifier = XGBClassifier(random\_state=0) xgb\_classifier.fit(X\_train\_final, y\_train) y\_pred\_xgb = xgb\_classifier.predict(X\_test\_final) accuracy\_score(y\_test, y\_pred\_xgb)

print(classification\_report(y\_test, y\_pred\_xgb))

Support vector Machine # Support vector classifier

from sklearn.svm import SVC

svc\_classifier = SVC(class\_weight = "balanced" ) svc\_classifier.fit(X\_train\_final, y\_train) y\_pred\_scv = svc\_classifier.predict(X\_test\_final) accuracy\_score(y\_test, y\_pred\_scv)

print(classification\_report(y\_test, y\_pred\_scv)) cm = confusion\_matrix(y\_test, y\_pred\_scv)

plt.title('Heatmap of Confusion Matrix', fontsize = 12) sns.heatmap(cm, annot = True, fmt = "d")

plt.show()

Hyperparameter Tuning with Support vector Machine # defining parameter range

param\_grid = {'C': [0.1, 1, 10, 100, 200 , 400 , 600 , 800],

'gamma': [1, 0.1, 0.01, 0.001, 0.0001],

'kernel': ['rbf']}

from sklearn.model\_selection import GridSearchCV

grid = GridSearchCV(SVC(), param\_grid, refit = True, verbose = 3) # fitting the model for grid search

grid.fit(X\_train\_final, y\_train)

# print best parameter after tuning print(grid.best\_params\_)

# print how our model looks after hyper-parameter tuning print(grid.best\_estimator\_)

# Support vector classifier from sklearn.svm import SVC

svc\_classifier = SVC(class\_weight = "balanced" , C=100, gamma=0.01) svc\_classifier.fit(X\_train\_final, y\_train)

y\_pred\_scv = svc\_classifier.predict(X\_test\_final) accuracy\_score(y\_test, y\_pred\_scv)

print(classification\_report(y\_test, y\_pred\_xgb))

cm = confusion\_matrix(y\_test, y\_pred\_scv) plt.title('Heatmap of Confusion Matrix', fontsize = 12) sns.heatmap(cm, annot = True , fmt = "d")

plt.show()

## Pickle

from sklearn.svm import SVC import pickle

# save model

pickle.dump(svc\_classifier, open('model.pkl', 'wb'))

# load model

water\_quality\_model = pickle.load(open('model.pkl', 'rb'))

# predict the output

y\_pred =water\_quality\_model.predict(X\_test\_final)

# confusion matrix

print('Confusion matrix of Support vector Machine : \n',confusion\_matrix(y\_test, y\_pred),'\n')

## APP.PY

from flask import Flask, request, render\_template import pickle

import pandas as pd import numpy as np import joblib

scaler = joblib.load("my\_scaler.save")

app = Flask( name )

model = pickle.load(open('model.pkl', 'rb'))

@app.route("/home") @app.route("/")

def hello():

return render\_template("home.html")

@app.route("/predict", methods = ["GET", "POST"]) def predict():

if request.method == "POST":

input\_features = [float(x) for x in request.form.values()] features\_value = [np.array(input\_features)]

feature\_names = ["ph", "Hardness" , "Solids", "Chloramines", "Sulfate", "Conductivity", "Organic\_carbon","Trihalomethanes", "Turbidity"]

df = pd.DataFrame(features\_value, columns = feature\_names) df = scaler.transform(df)

output = model.predict(df)

if output[0] == 1: prediction = "safe"

else:

prediction = "not safe"

return render\_template('home.html', prediction\_text= "water is {} for human consumption ".format(prediction)) if name == " main ":

app.run(debug=True)

## REQUIREMENT.TXT

Flask == 2.2.2

joblib == 1.2.0

numpy == 1.23.4

pandas == 1.5.1

scikit-learn == 1.1.3

xgboost == 1.7.1

gunicorn == 20.1.0

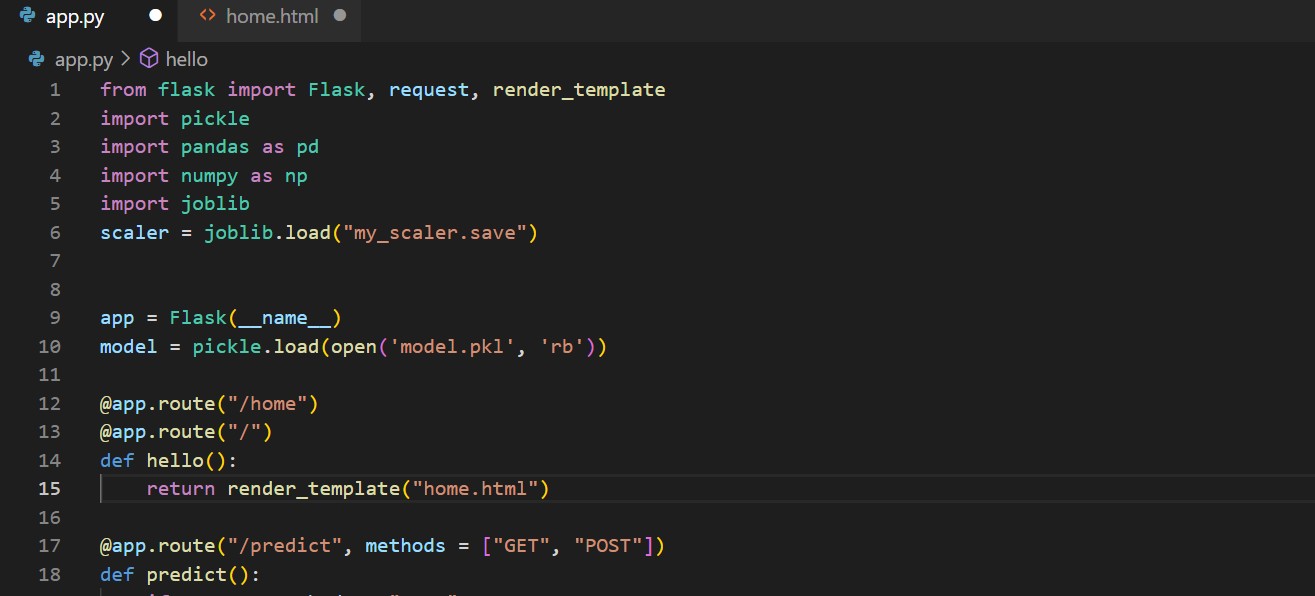
matplotlib == 3.6.2

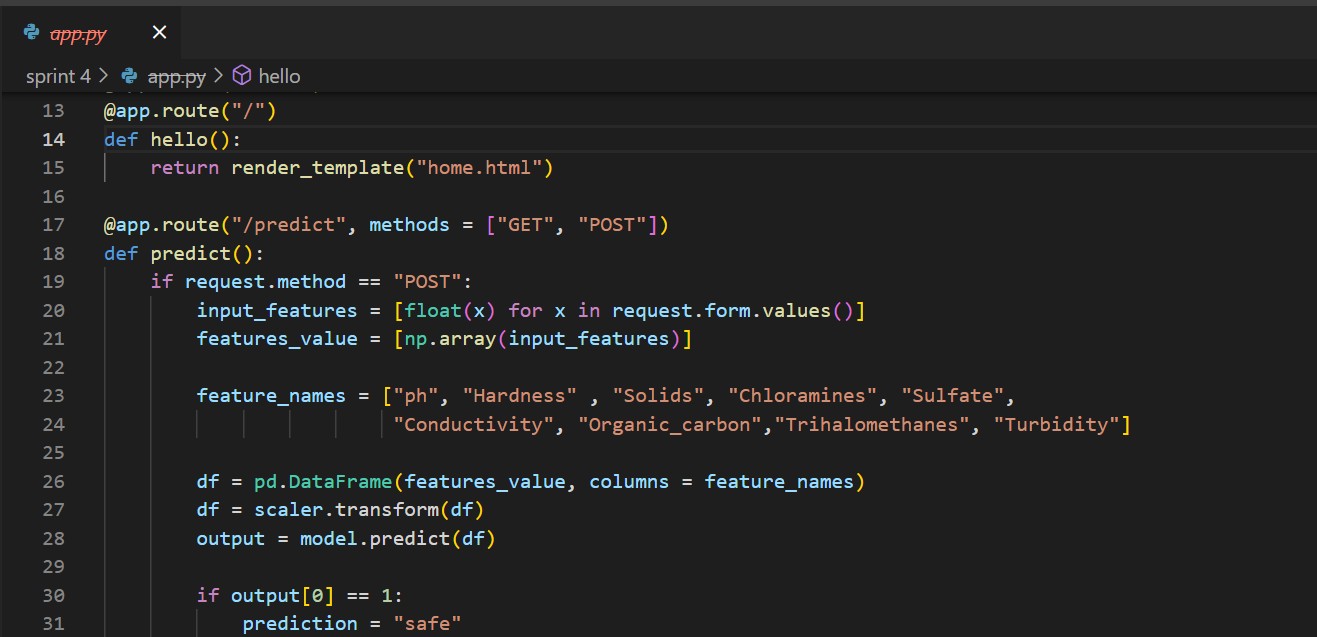
seaborn == 0.12.1 gevent

requests

flask-cors==3.0.10

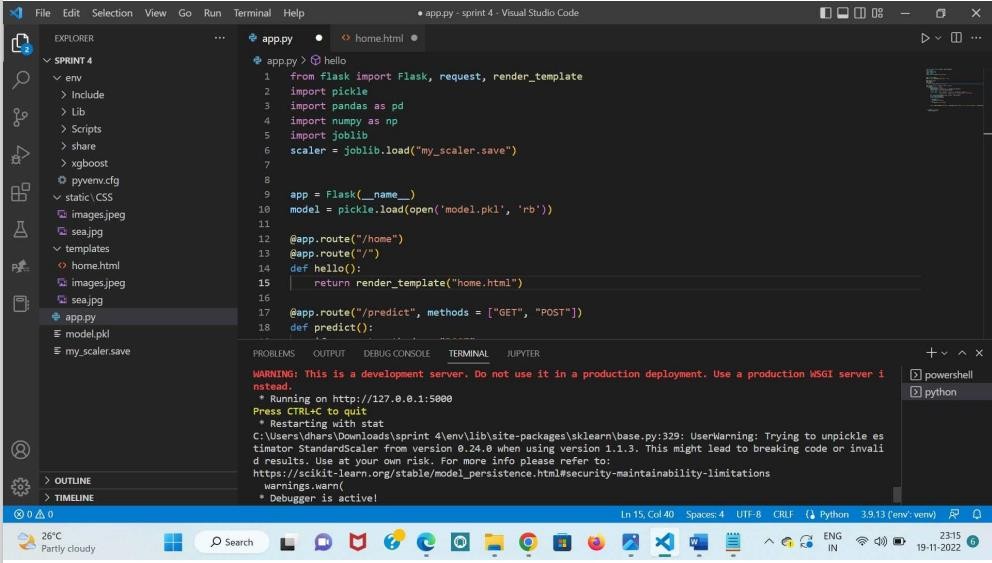
**APP.PY**





## WATER QUALITY.IPYNB

**HOME.HTML**



**LINKS:**

GITHUB :

https://github.com/IBM-EPBL/IBM-Project-15900-1659606030