

# **EFFICIENT WATER QUALITY ANALYSIS AND PREDICTION A PROJECT REPORT**

Submitted by

**V KRISHNAKUMAR** -(310119106015)  
**A DANIEL** - (310119106010)  
**Y GNANA PRAKASH** -(310119106013)  
**A MOHAMED AFSAL** -(310119106018)

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

**BACHELOR OF ENGINEERING**

*in*

**ELECTRONICS AND COMMUNICATION ENGINEERING**

**ANAND INSTITUTE OF HIGHER TECHNOLOGY**



**ANNA UNIVERSITY: CHENNAI 600 025**

**APRIL 2022**

# TABLE OF CONTENT

CHAPTER NO	TITLE	PAGE NO
<b>1</b>	<b>INTRODUCTION</b>	
	1.1 PROJECT OVERVIEW	4
	1.2 PURPOSE	4
<b>2</b>	<b>LITERATURE SURVEY</b>	5
	2.1 EXISTING PROBLEM	7
	2.2 REFERENCES	8
	2.3 PROBLEM STATEMENT DEFINITION	9
<b>3</b>	<b>IDEATION &amp; PROPOSED SOLUTION</b>	
	3.1 EMPATHY MAP CANVAS	9
	3.2 IDEATION & BRAINSTORMING	10
	3.3 PROPOSED SOLUTION	11
	3.4 PROBLEM SOLUTION FIT	12
<b>4</b>	<b>REQUIREMENT ANALYSIS</b>	
	4.1 FUNCTIONAL REQUIREMENT	13
	4.2 NON-FUNCTIONAL REQUIREMENTS	14
<b>5</b>	<b>PROJECT DESIGN</b>	
	5.1 DATA FLOW DIAGRAMS	15
	5.2 SOLUTION & TECHNICAL ARCHITECTURE	16
	5.3 USER STORIES	19
<b>6</b>	<b>PROJECT PLANNING &amp; SCHEDULING</b>	
	6.1 SPRINT PLANNING & ESTIMATION	21
	6.2 SPRINT DELIVERY SCHEDULE	22
	6.3 REPORTS FROM JIRA	23
<b>7</b>	<b>CODING &amp; SOLUTIONING (Explain the features added in the project along with code)</b>	

	7.1 FEATURE 1	24
	7.2 FEATURE 2	26
	7.3 DATABASE SCHEMA (if Applicable)	
<b>8</b>	<b>TESTING</b>	
	8.1 TEST CASES	29
	8.2 USER ACCEPTANCE TESTING	30
<b>9</b>	<b>RESULTS</b>	
	9.1 PERFORMANCE METRICS	31
<b>10</b>	<b>ADVANTAGES &amp; DISADVANTAGES</b>	33
<b>11</b>	<b>CONCLUSION</b>	34
<b>12</b>	<b>FUTURE SCOPE</b>	34
<b>13</b>	<b>APPENDIX</b>	
	SOURCE CODE	35
	GITHUB & PROJECT DEMO LINK	49

# 1.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

## 1.1 PROJECT OVERVIEW

With the rapid increase in the volume of data on the 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 and management systems. Additionally, machine learning can provide solutions for water pollution control, water quality improvement, and 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.

## 2. LITERATURE SURVEY

Many works have been conducted to predict water quality using Machine Learning (ML) approaches. Some researchers used the traditional Machine Learning models, such as Decision Tree <sup>[13][14]</sup>, Artificial Neural Network <sup>[2][5][6][7]</sup>, Support Vector Machine <sup>[8][9][10]</sup>, K-Nearest Neighbors <sup>[21]</sup> and Naïve Bayes <sup>[18][22][23]</sup>. However, in recent years, some researchers are moving towards more advanced ML ensemble models, such as Gradient Boosting and Random Forest <sup>[1]</sup>

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 <sup>[4]</sup>. 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 <sup>[6]</sup>. Gakii and Jepkoech <sup>[3]</sup> 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 Jelihouni et al. <sup>[4]</sup> also compared five decision-tree-based models, which are Random Tree, Random Forest, Ordinary Decision Tree (ODT), Chi-square Automatic Interaction 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 <sup>[8]</sup> 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 <sup>[5]</sup> 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 <sup>[8]</sup>.

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 <sup>[11]</sup> 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 <sup>[12]</sup> 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 <sup>[12]</sup>. This is supported by another study in Eastern Azerbaijan, Iran <sup>[6]</sup>. 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  $R^2$  than the results attained in Abbasi et al.'s report <sup>[4]</sup>. Naïve Bayes has also been widely used for predicting water quality. A study by Vijay and Kamaraj <sup>[2]</sup> 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 <sup>[22]</sup>. 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 <sup>[9]</sup>. 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 <sup>[30]</sup>. 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 <sup>[30]</sup>. 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 <sup>[6]</sup>. 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 <sup>[7]</sup>.

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 <sup>[1]</sup>. 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 <sup>[5]</sup>. 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 <sup>[2]</sup>. 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 <sup>[3]</sup>. Therefore, advanced models, such as Random Forest and Gradient Boosting, should be employed to cater for the lack of basic ML models.

## 2.1 EXISTING 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

## 2.2 REFERENCES

1. Ling, J.K.B. Water Quality Study and Its Relationship with High Tide and Low Tide at Kuantan River. Bachelor's Thesis, Universiti Malaysia Pahang, Gambang, Malaysia, 2010. Available online: [http://umpir.ump.edu.my/id/eprint/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 2022).

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 Mortality 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 neural networks in prediction of water quality parameters. *J. Environ. Health Sci. 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, 1787–1798.
7. Al-Badaii, F.; Shuhaimi-Othman, M.; Gasim, M.B. Water quality assessment 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 quality index prediction and uncertainty analysis: A comparative study of machine 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 machine learning models based on big data. *Water Res.* 2020, 171, 115454.
10. Leros, 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 intelligence techniques: SOM and ANN—A case study of Melen River Turkey. *Water Qual. Expo. Health* 2015, 7, 469–490.
12. Aradhana, G.; Singh, N.B. Comparison of Artificial Neural Network algorithm for water quality prediction of River Ganga. *Environ. Res. J.* 2014, 8, 55–63.

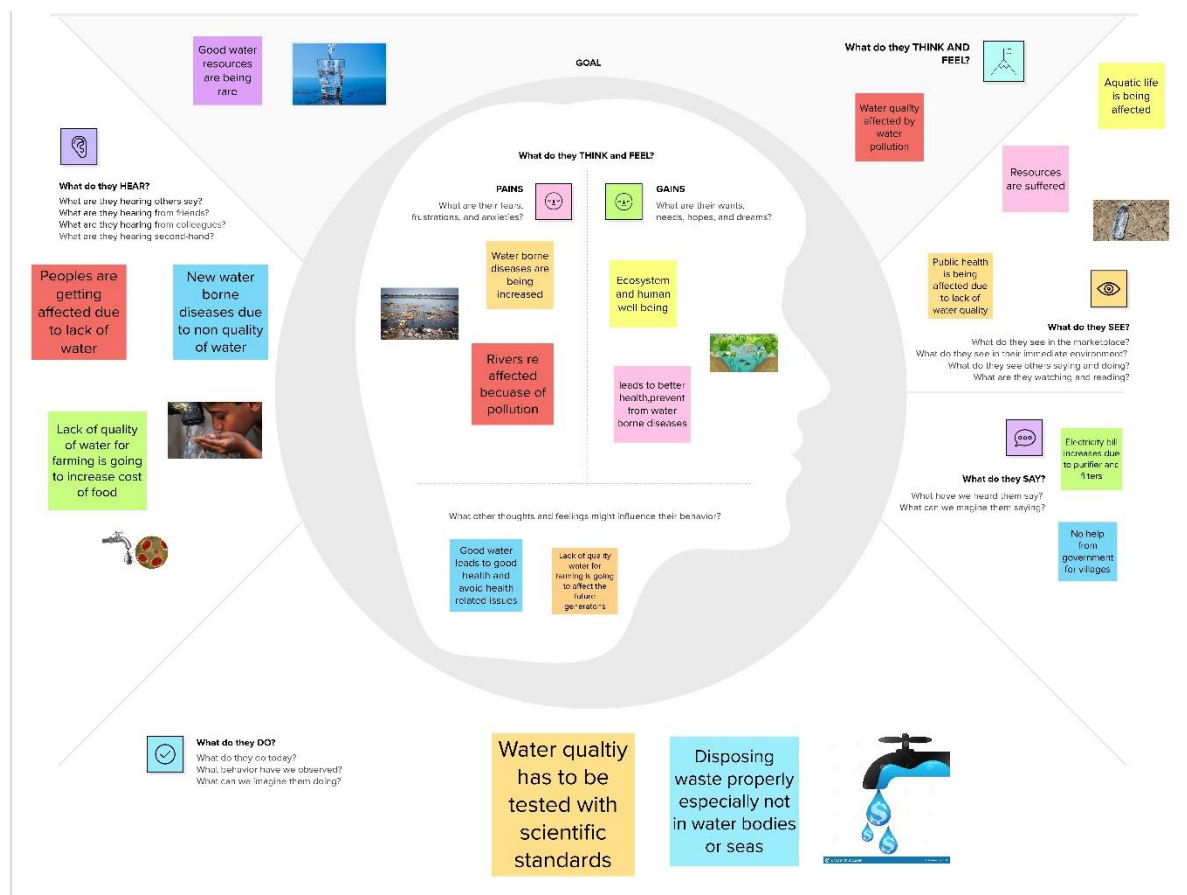


## 2.3 PROBLEM STATEMENT DEFINITION

To predict the water safe or not for Access to safe drinking-water is essential to health, 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 local level. In some regions, it has been shown that investments in water supply and sanitation can yield a net economic benefit, since the reductions in adverse health effects and health care costs outweigh the costs of undertaking the interventions.

## 3.IDEATION AND PROPOSED SOLUTION

### 3.1 EMPATHY MAP CANVAS



## 3.2 IDEATION AND BRAINSTORMING

Template

## Brainstorm & idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start sharing concepts even if you're not sitting in the same room.

- 1 10 minutes to prepare
- 2 1 hour to facilitate
- 3 4-6 people to brainstorm

4+

### Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get started.

- 1. [Get started](#)

1

#### Know your goal

Before you start brainstorming, it's important to clarify the goal of the session. What do you want to achieve?

2

#### Set the goal

Brainstorming is a creative process. It's important to set a clear goal for the session. What do you want to achieve?

3

#### Learn how to use the facilitation tool

Before you start brainstorming, it's important to learn how to use the facilitation tool. What do you want to achieve?

[Click to edit](#)

1

### Define your problem statement

What problem are you trying to solve? Define your problem in a clear, concise statement. This will be the focus of your brainstorming.

- 1. [Get started](#)

PROBLEM

### TO ANALYSE AND IMPROVE WATER QUALITY

#### Key rules of brainstorming

The main principle of brainstorming is:

- 1. Quantity over quality
- 2. No criticism
- 3. Encourage wild ideas
- 4. Stay focused
- 5. One idea per person
- 6. Go for quantity
- 7. Build on others' ideas

2

### Brainstorm

Write down any ideas that come to mind. It's important to have a problem statement to focus your brainstorming.

- 1. [Get started](#)

KIRAN/ANAN

DANIEL

Government agencies should check the standards of water

People should be aware of water borne diseases

Government should clean up water bodies now and then

Government should check quality level constantly

Water purifiers can be used to get quality water

People should be aware of water borne diseases

BETHESI

SHRADDHA/AN

Government should check the quality of water in all areas

Toxic chemicals should not be disposed near water bodies

Government should give more funds to improve water quality

Toxic chemicals should not be disposed in water bodies

Establish a wastewater treatment system to check water quality

People should properly dispose of waste in the trash can

SHRADDHA/AN

Boil water before using to cook and for drinking purposes

Government give awareness to public to use quality of water

New smart gadgets to test water quality

[Click to edit this feedback](#)

### 3 Group ideas

"Take 10 minutes to generate some ideas. Remember, it's a brainstorm so you go for as many ideas as you can. There are no bad ideas. The ideas that are the most creative and useful will stand out. If you have a lot of ideas, you can choose the best ones to develop further."

© 2019 Pearson Education Ltd. All rights reserved. No part of this publication may be reproduced without the prior written permission of Pearson Education Ltd. This material is protected by the copyright laws of your country.

### 4 Prioritize

"Now we need to decide which ideas are the most important. We will do this by ranking the ideas. The ideas that are the most important will be ranked 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195, 196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222, 223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 263, 264, 265, 266, 267, 268, 269, 270, 271, 272, 273, 274, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 290, 291, 292, 293, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 311, 312, 313, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323, 324, 325, 326, 327, 328, 329, 330, 331, 332, 333, 334, 335, 336, 337, 338, 339, 340, 341, 342, 343, 344, 345, 346, 347, 348, 349, 350, 351, 352, 353, 354, 355, 356, 357, 358, 359, 360, 361, 362, 363, 364, 365, 366, 367, 368, 369, 370, 371, 372, 373, 374, 375, 376, 377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388, 389, 390, 391, 392, 393, 394, 395, 396, 397, 398, 399, 400, 401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 411, 412, 413, 414, 415, 416, 417, 418, 419, 420, 421, 422, 423, 424, 425, 426, 427, 428, 429, 430, 431, 432, 433, 434, 435, 436, 437, 438, 439, 440, 441, 442, 443, 444, 445, 446, 447, 448, 449, 450, 451, 452, 453, 454, 455, 456, 457, 458, 459, 460, 461, 462, 463, 464, 465, 466, 467, 468, 469, 470, 471, 472, 473, 474, 475, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485, 486, 487, 488, 489, 490, 491, 492, 493, 494, 495, 496, 497, 498, 499, 500, 501, 502, 503, 504, 505, 506, 507, 508, 509, 510, 511, 512, 513, 514, 515, 516, 517, 518, 519, 520, 521, 522, 523, 524, 525, 526, 527, 528, 529, 530, 531, 532, 533, 534, 535, 536, 537, 538, 539, 540, 541, 542, 543, 544, 545, 546, 547, 548, 549, 550, 551, 552, 553, 554, 555, 556, 557, 558, 559, 560, 561, 562, 563, 564, 565, 566, 567, 568, 569, 570, 571, 572, 573, 574, 575, 576, 577, 578, 579, 580, 581, 582, 583, 584, 585, 586, 587, 588, 589, 590, 591, 592, 593, 594, 595, 596, 597, 598, 599, 600, 601, 602, 603, 604, 605, 606, 607, 608, 609, 610, 611, 612, 613, 614, 615, 616, 617, 618, 619, 620, 621, 622, 623, 624, 625, 626, 627, 628, 629, 630, 631, 632, 633, 634, 635, 636, 637, 638, 639, 640, 641, 642, 643, 644, 645, 646, 647, 648, 649, 650, 651, 652, 653, 654, 655, 656, 657, 658, 659, 660, 661, 662, 663, 664, 665, 666, 667, 668, 669, 670, 671, 672, 673, 674, 675, 676, 677, 678, 679, 680, 681, 682, 683, 684, 685, 686, 687, 688, 689, 690, 691, 692, 693, 694, 695, 696, 697, 698, 699, 700, 701, 702, 703, 704, 705, 706, 707, 708, 709, 710, 711, 712, 713, 714, 715, 716, 717, 718, 719, 720, 721, 722, 723, 724, 725, 726, 727, 728, 729, 730, 731, 732, 733, 734, 735, 736, 737, 738, 739, 740, 741, 742, 743, 744, 745, 746, 747, 748, 749, 750, 751, 752, 753, 754, 755, 756, 757, 758, 759, 760, 761, 762, 763, 764, 765, 766, 767, 768, 769, 770, 771, 772, 773, 774, 775, 776, 777, 778, 779, 780, 781, 782, 783, 784, 785, 786, 787, 788, 789, 790, 791, 792, 793

### 3.3 PROPOSED SOLUTION

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	<ul style="list-style-type: none"> <li>Water quality prediction using linear regression in machine learning techniques. In this technique, our model predicts that the water contaminants or using some parameters like Ph value, conductivity, hardness, etc.,</li> </ul>
2.	Idea / Solution description	<ul style="list-style-type: none"> <li>Water Quality Index (WQI) assessment is required for the purpose of public health.</li> <li>An effective linear regression model is developed to predict the WQI based on the parameters like temperature, dissolved oxygen, pH value, Turbidity, Nitrates and faecal coliform.</li> </ul>
3.	Novelty / Uniqueness	<ul style="list-style-type: none"> <li>In this prediction, the main uniqueness is that using linear regression</li> </ul>
4.	Social Impact / Customer Satisfaction	<ul style="list-style-type: none"> <li>One of the important factors leading to poverty is shortage of water resource.</li> <li>The shortage of water resource can lead to poor land productivity, low production level and low income level, and can also lead to drinking water shortage and poor life quality.</li> </ul>
5.	Business Model (Revenue Model)	<ul style="list-style-type: none"> <li>For Analysing the metrics of each water resource a charge will be collected.</li> </ul>
6.	Scalability of the Solution	<ul style="list-style-type: none"> <li>The result gives more accuracy of the water quality analysis by using linear regression.</li> </ul>

## 3.4 PROBLEM SOLUTION FIT

Define CS, fit into CC	<b>1. CUSTOMER SEGMENT(S)</b> <span>CS</span> Who is your customer? i.e. working parents of 0-5 y.o. kids <ul style="list-style-type: none"> <li>• Government agencies</li> <li>• General public</li> </ul>	<b>6. CUSTOMER CONSTRAINTS</b> <span>CC</span> What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available devices. <ul style="list-style-type: none"> <li>• Knowledge about Scientific standards</li> <li>• No Tools and available devices</li> <li>• Inference of Water quality</li> </ul>	<b>5. AVAILABLE SOLUTIONS</b> <span>AS</span> Which solutions are available to the customers when they face the problem or need to get the job done? What have they tried in the past? What pros & cons do these solutions have? i.e. pen and paper is an alternative to digital notetaking RO and heating of water before using PROS: Filters the most contaminants CONS: <ul style="list-style-type: none"> <li>• More water wasted</li> <li>• Drinking water that's too hot can damage tissue</li> </ul>	Explore AS, differentiate
	<b>2. JOBS-TO-BE-DONE / PROBLEMS</b> <span>J&amp;P</span> Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides. To Check the scientific standards obtained from the water samples	<b>9. PROBLEM ROOT CAUSE</b> <span>RC</span> What is the real reason that this problem exists? What is the back story behind the need to do this job? i.e. customers have to do it because of the change in regulations. <ul style="list-style-type: none"> <li>• Water scarcity</li> <li>• No available of quality water</li> <li>• We couldn't judge or find the water is quality or not</li> </ul>	<b>7. BEHAVIOUR</b> <span>BE</span> What does your customer do to address the problem and get the job done? i.e. directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace) <ul style="list-style-type: none"> <li>• RO</li> <li>• Mineral water</li> <li>• Checking hardness of water</li> </ul>	
Identify strong TR & EM	<b>3. TRIGGERS</b> <span>TR</span> What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news. <ul style="list-style-type: none"> <li>• Non drinkable water quality due to water scarcity in our country</li> </ul>	<b>10. YOUR SOLUTION</b> <span>SL</span> If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality. If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour. <p>By collecting water samples and using some parameters like Ph value, Hardness, Conductivity by using machine learning techniques to predict the water quality</p>	<b>8. CHANNELS of BEHAVIOUR</b> <span>CH</span> <b>8.1 ONLINE</b> What kind of actions do customers take online? Extract online channels from #7 <b>8.2 OFFLINE</b> What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development. <p>ONLINE: Browsing various ways for getting quality water</p> <p>OFFLINE: Boiling drinking water and RO Process</p>	Identify strong TR & EM
	<b>4. EMOTIONS: BEFORE / AFTER</b> <span>EM</span> How do customers feel when they face a problem or a job and afterwards? i.e. lost, insecure > confident, in control - use it in your communication strategy & design. <p>BEFORE:</p> <ul style="list-style-type: none"> <li>• Feeling Irritated</li> <li>• fear of water borne diseases</li> </ul> <p>AFTER:</p> <ul style="list-style-type: none"> <li>• feeling safe</li> <li>• Happy living</li> </ul>			

## 4. REQUIREMENT ANALYSIS

### 4.1 FUNCTIONAL REQUIREMENT

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	<ul style="list-style-type: none"> <li>☐ Registration through Form</li> <li>☐ Registration through Gmail</li> <li>☐ Registration through LinkedIn</li> </ul>
FR-2	User Confirmation	<ul style="list-style-type: none"> <li>☐ Confirmation via Email</li> <li>☐ Confirmation via OTP</li> </ul>
FR-3	Select the water quality testing parameters	<ul style="list-style-type: none"> <li>☐ Chemical contamination</li> <li>☐ Microbial contamination</li> <li>☐ Physical contamination</li> </ul>
FR-4	Physical contamination	<ul style="list-style-type: none"> <li>☐ <b>PH</b>-is important when disinfecting water with chloride</li> <li>☐ <b>EC</b>-unusually high level may suggest chemical contamination.</li> <li>☐ <b>Turbidity</b>-High turbidity decreases water acceptability.</li> </ul>
FR-5	Chemical contamination	<ul style="list-style-type: none"> <li>☐ <b>Fluoride(1.5 mg/l)</b>-Fluoride is a naturally-occurring form of the element fluorine, which is sometimes found in groundwater at levels that exceed safe levels.</li> <li>☐ <b>Nitrate and Nitrite(50 mg/l)</b>-In most cases, these compounds aren't a serious health risk.</li> <li>☐ <b>Arsenic(10µg/l)</b>-The EPA says studies link long-term exposure of arsenic to certain cancers as well as cardiovascular, neurological, and other conditions.</li> <li>☐ <b>Chlorine(5 mg/L)</b>-This value is the health-based guideline. Chlorine is often used for water treatment.</li> </ul>
FR-6	Microbial contamination	<ul style="list-style-type: none"> <li>☐ <b>E. coli(0 MPN/100 ml)</b>-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.</li> </ul>

## 4.2 NON-FUNCTIONAL REQUIREMENT

FR No.	Non-Functional Requirement	Description
NFR-1	<b>Usability</b>	☐ To improve usability of data provided to water quality exchange, to monitor nutrient record in water.
NFR-2	<b>Security</b>	☐ The capacity of a population to safeguard sustainable access to adequate quantities of acceptable quality for sustaining livelihoods.
NFR-3	<b>Reliability</b>	☐ <b>System adequacy and system security-</b> A hierarchical framework approach to system adequacy evaluation is presented. Adequacy evaluation techniques for each hierarchical level associated with basic probabilistic indices.
NFR-4	<b>Performance</b>	☐ 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	<b>Availability</b>	☐ 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	<b>Scalability</b>	☐ 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.

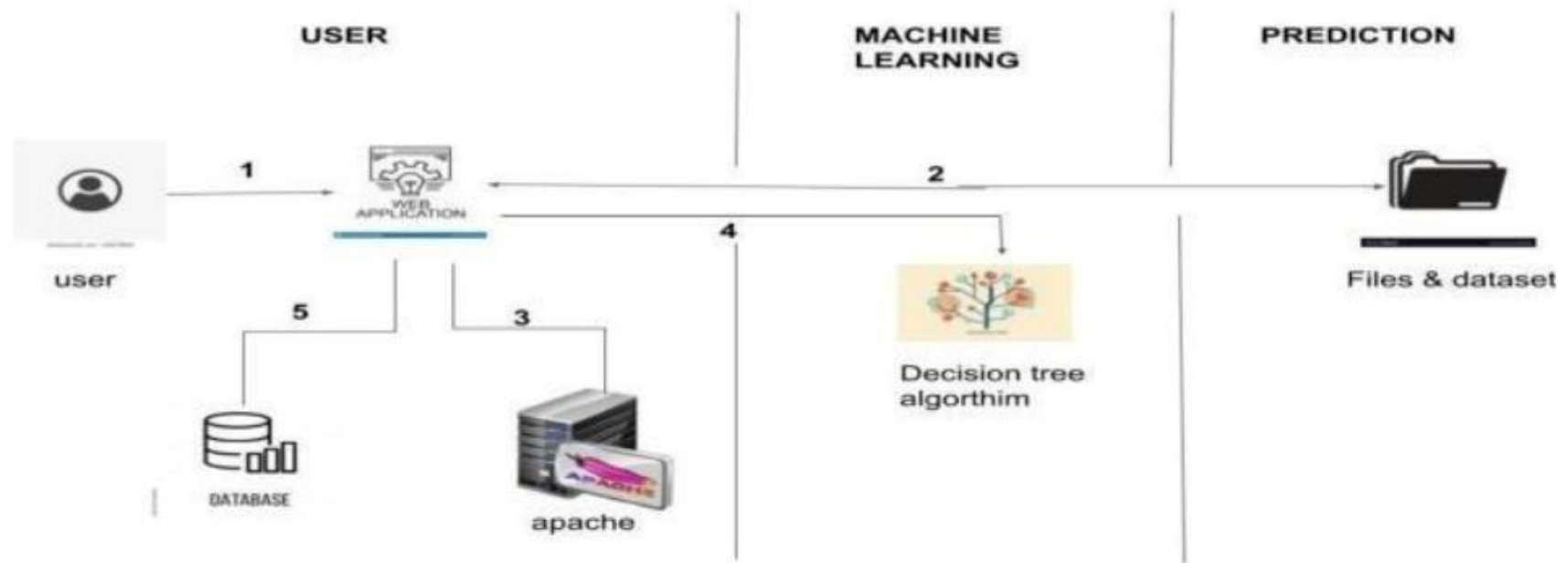


## 5.PROJECT DESIGN

### 5.1 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.

Flow diagram:



## 5.2 SOLUTION AND TECHNICAL ARCHITECTURE

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

### *A. 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.

### *B. 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.

### *C. 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.

### *D. 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..



### *F. 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

### *G. 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.

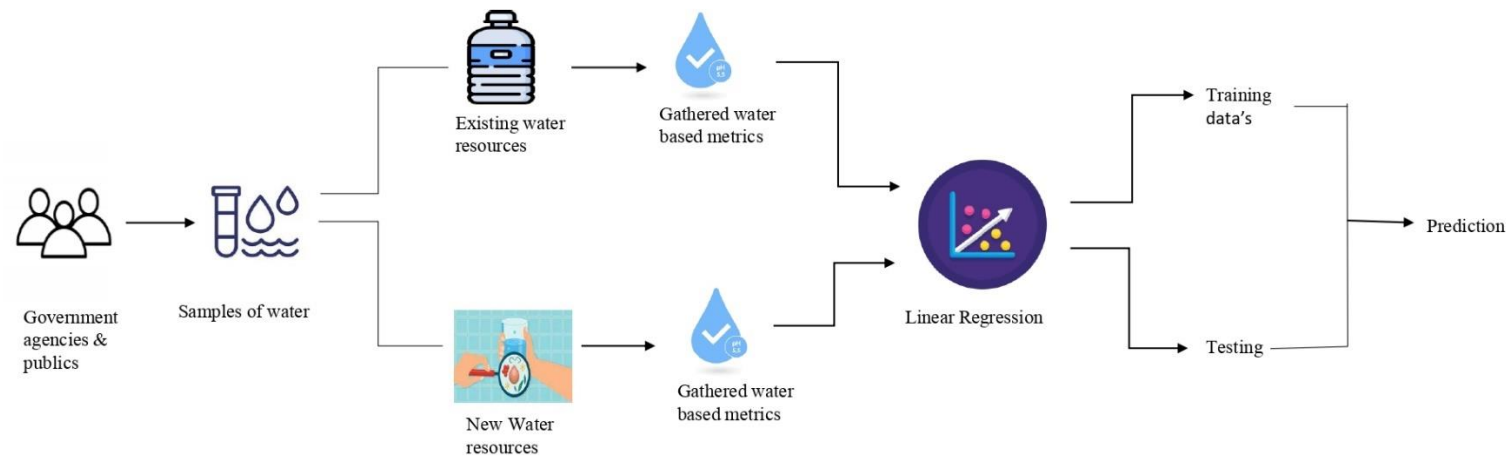
### *H. Data Modeling*

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.

### *I. 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



## 5.3 USER STORIES

### User Stories

Use the below template to list all the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail	I can register & access the dashboard with Gmail Login	Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password	I can log into the application by entering email & password	High	Sprint-1

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
	Dashboard	USN-6	As a user,I will receive the information about myself once I have login into the application	I can receive information about myself once login	High	Sprint-1
Customer (Web user)	Register	USN-7	As a web user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
	Login	USN-8	As a web user, I can log into the application by entering email & password	I can log into the application by entering email & password	High	Sprint-1
	Dashboard	USN-9	As a web user,I will receive the information about myself once I have login into the application	I can receive information about myself once login	High	Sprint-1
Customer Care Executive	Register	USN-10	As a customer care executive,I can register using G-mail and password	I can access my account	High	Sprint-1
	Login	USN-11	As a customer care executive,I can login by entering G-mail and password	I can log into the application by entering email & password	High	Sprint-1
	Dashboard	USN-12	As a customer care executive,I will receive the information about	I can receive issue from customer	High	Sprint-1

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
			issues of the customer once I can login into the application			
Administrator	Login	USN-13	As a administrator,I can login using Admin username and password	I can login to the application	High	Sprint-1
	Dashboard	USN-14	As a administrator, I can access the dashboard of the customer	I can access the dashboard of customer	High	Sprint-1

## 6. PROJECT PLANNING AND SCHEDULING

### 6.1 SPRINT PLANNING AND ESTIMATION

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story points	Priority	Team Members
Sprint-1	Registration & Login	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	3	High	V Krishnakumar
		USN-2	As a user,OTP is generated and the OTP is verified	1	High	Y Gnana Prakash
		USN-3	As a user , I will receive confirmation E-mail once I have registered for the application	1	High	A Daniel
		USN-4	As a user, I can log into the application by entering email & password	1	High	Y Gnana Prakash
Sprint-2	Dashboard	USN-4	As a user, I will able to see my profile information once I have login into the application	2	High	A Mohamed Afsal
		USN-5	As a user, I will be able to logout from my account in dashboard from application	1	High	V Krishnakumar
Sprint-3	Data Collection & Processing	USN-6	As a user, I will give information about collected water samples.	3	High	A Daniel
		USN-7	User should give PH, temperature, conductivity, solid, hardness of the water	4	High	Y Gnana Prakash

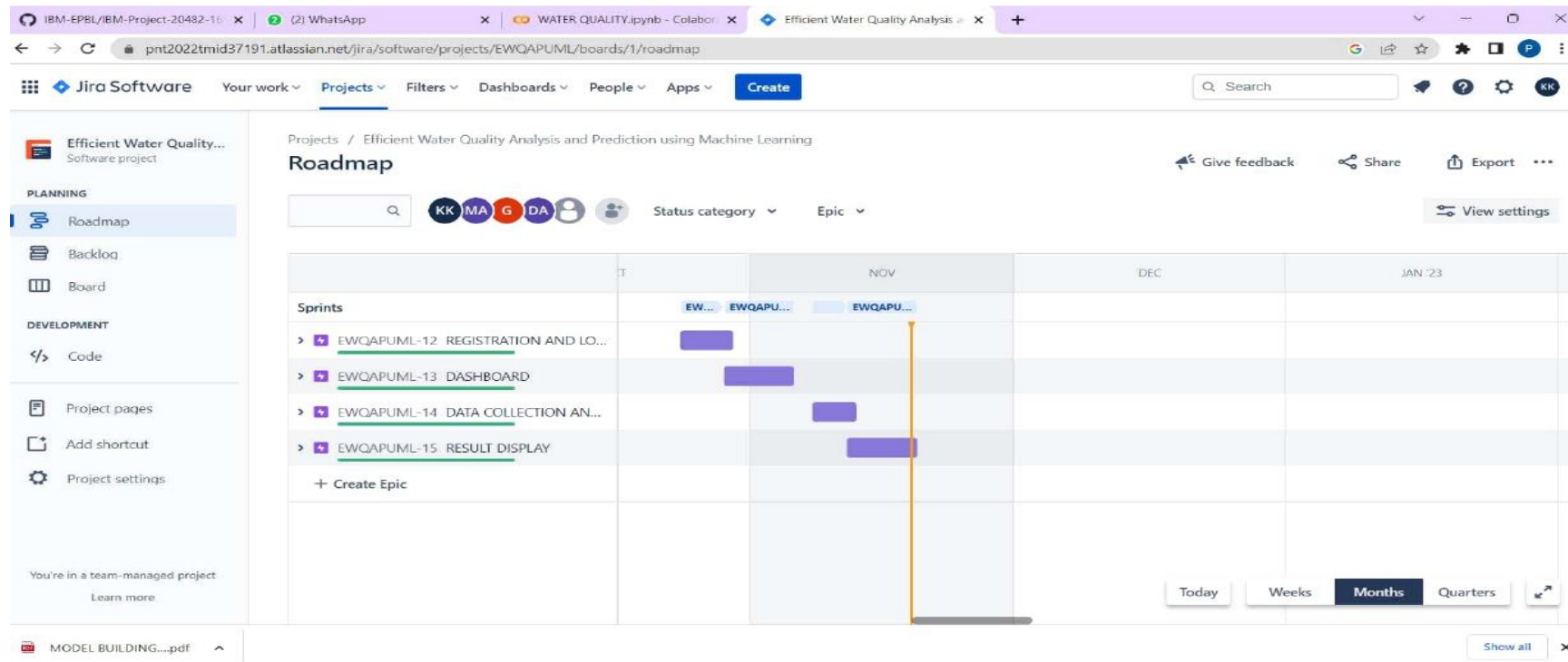
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story points	Priority	Team Members
		USN-8	In data processing, data will get processed based on collected water sample	5	Medium	V Krishnakumar
		USN-9	From the user given pH and temperature ,conductivity ,solid, hardness of the sample water and it will process by using machine learning technology to predict the water quality.	6	High	A Mohamed Afsal
Sprint-4	Result Display	USN-10	As a user, I can get the result of the given water sample are quality or not with accuracy in a result page in the app	3	High	A Daniel

## 6.2 SPRINT SCHEDULE

### Project Tracker, Velocity & Burndown Chart: (4 Marks)

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	6	6 Days	24 Oct 2022	29 Oct 2022	6	19 Nov 2022
Sprint-2	3	6 Days	31 Oct 2022	05 Nov 2022	3	19 Nov 2022
Sprint-3	18	6 Days	07 Nov 2022	12 Nov 2022	18	19 Nov 2022
Sprint-4	3	6 Days	14 Nov 2022	19 Nov 2022	3	19 Nov 2022

## 6.3 REPORTS FROM JIRA



## 7. CODING AND SOLUTIONS

### 7.1 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

	ph	Hardness	Solids	Chloramines	Sulfate	Conductivity	Organic_carbon	Trihalomethanes	Turbidity	Potability
0	NaN	204.890455	20791.318981	7.300212	368.516441	564.308654	10.379783	86.990970	2.963135	0
1	3.716080	129.422921	18630.057858	6.635246	NaN	592.885359	15.180013	56.329076	4.500656	0
2	8.099124	224.236259	19909.541732	9.275884	NaN	418.606213	16.868637	66.420093	3.055934	0
3	8.316766	214.373394	22018.417441	8.059332	356.886136	363.266516	18.436524	100.341674	4.628771	0
4	9.092223	181.101509	17978.986339	6.546600	310.135738	398.410813	11.558279	31.997993	4.075075	0

#### 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.



## Feature scaling

```
[41] from sklearn.preprocessing import StandardScaler
      sc = StandardScaler()
      X_train_final = sc.fit_transform(X_train)
      X_test_final = sc.transform(X_test)
```

```
[42] from sklearn.metrics import confusion_matrix, classification_report, accuracy_score
```

## 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:

$$WQI = \frac{\sum_{i=1}^N q_i \times w_i}{\sum_{i=1}^N w_i},$$

where:  $N$  is the total number of parameters included in the WQI calculations,  $q_i$  is the quality rating scale for each parameter calculated by equation (2) below, and  $w_i$  is the unit weight for each parameter calculated by equation (3).

$$q_i = 100 \times \left( \frac{V_i - V_{Ideal}}{S_i - V_{Ideal}} \right),$$

where:  $V_i$  is the measured value of parameter in the tested water samples,  $V_{Ideal}$  is the ideal value of parameter in pure water (0 for all parameters except  $pH$  and  $DO$ ), and  $S_i$  is the recommended standard value of parameter (as shown in Table 1)

$$w_i = \frac{K}{S_i},$$

## 7.2 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.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}}$$

```
# Random Forest Classifier
from sklearn.ensemble import RandomForestClassifier
rf_classifier = RandomForestClassifier(n_estimators = 20, criterion = 'entropy', class_weight = "balanced_subsample", random_state = 51)
rf_classifier.fit(X_train_final, y_train)
y_pred = rf_classifier.predict(X_test_final)
accuracy_score(y_test, y_pred)
```

0.635

```
print(classification_report(y_test, y_pred))
```

	precision	recall	f1-score	support
0	0.66	0.86	0.75	497
1	0.54	0.26	0.35	303
accuracy			0.64	800
macro avg	0.60	0.56	0.55	800
weighted avg	0.61	0.64	0.60	800

```
# 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)
```

0.62125

```
print(classification_report(y_test, y_pred_xgb))
```

	precision	recall	f1-score	support
0	0.64	0.90	0.75	497
1	0.50	0.17	0.25	303
accuracy			0.62	800
macro avg	0.57	0.53	0.50	800
weighted avg	0.59	0.62	0.56	800

## Support vector Machine

```
[53] # 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)
```

0.6225

```
[54] print(classification_report(y_test, y_pred_scv))
```

	precision	recall	f1-score	support
0	0.70	0.69	0.70	497
1	0.50	0.50	0.50	303
accuracy			0.62	800
macro avg	0.60	0.60	0.60	800
weighted avg	0.62	0.62	0.62	800

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)) is used to classify the tested water samples based on their quality.

## 8.TESTING

### 8.1 TEST CASES 1

By PNT2022TMID37191

Enter values

pH value :	<input type="text" value="pH value"/>	Hardness :	<input type="text" value="Hardness"/>	Solids :	<input type="text" value="Solids"/>
Chloramines :	<input type="text" value="Chloramines"/>	Sulfate :	<input type="text" value="Sulfate"/>	Conductivity :	<input type="text" value="Conductivity"/>
Organic_carbon :	<input type="text" value="Organic_carbon"/>	Trihalomethanes :	<input type="text" value="Trihalomethanes"/>	Turbidity :	<input type="text" value="Turbidity"/>

Water quality Test

water is safe for human consumption

Team Members:  
krishnakumar  
daniel  
gnanaprakash  
mohamed afsal

## 8.2 USER ACCEPTANCE TESTING

### 1. 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).

### 2. 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

### 3. Test Case Analysis

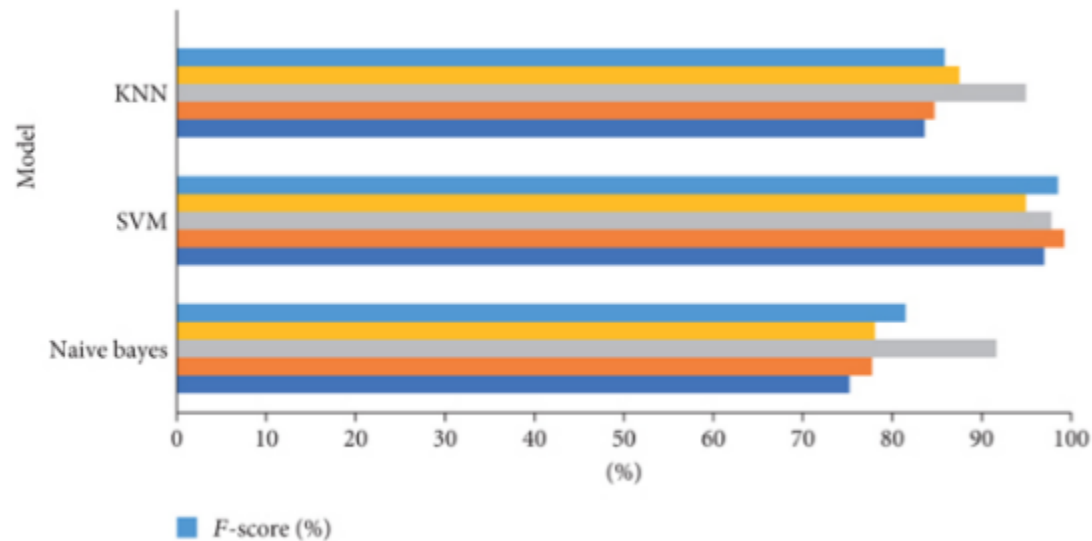
This report shows the number of test cases that have passed, failed, and untested

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

## 9.RESULT

### 9.1 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 =  $\frac{TP+TN}{TP+FP+FN+TN}$**

**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



## 10. 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

## 11. 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 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 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 duplicate 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: {:.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_state = 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

```
app.py  home.html
app.py > hello
1  from flask import Flask, request, render_template
2  import pickle
3  import pandas as pd
4  import numpy as np
5  import joblib
6  scaler = joblib.load("my_scaler.save")
7
8
9  app = Flask(__name__)
10 model = pickle.load(open('model.pkl', 'rb'))
11
12 @app.route("/home")
13 @app.route("/")
14 def hello():
15     return render_template("home.html")
16
17 @app.route("/predict", methods = ["GET", "POST"])
18 def predict():
```

app.py X

sprint 4 > app.py > hello

```
13 @app.route("/")
14 def hello():
15     return render_template("home.html")
16
17 @app.route("/predict", methods = ["GET", "POST"])
18 def predict():
19     if request.method == "POST":
20         input_features = [float(x) for x in request.form.values()]
21         features_value = [np.array(input_features)]
22
23         feature_names = ["ph", "Hardness" , "Solids", "Chloramines", "Sulfate",
24                           "Conductivity", "Organic_carbon","Trihalomethanes", "Turbidity"]
25
26         df = pd.DataFrame(features_value, columns = feature_names)
27         df = scaler.transform(df)
28         output = model.predict(df)
29
30         if output[0] == 1:
31             prediction = "safe"
```

## WATER QUALITY.IPYNB

### Support vector Machine

```
[53] # 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)
```

0.6225

```
[54] print(classification_report(y_test, y_pred_scv))
```

	precision	recall	f1-score	support
0	0.70	0.69	0.70	497
1	0.50	0.50	0.50	303
accuracy			0.62	800
macro avg	0.60	0.60	0.60	800
weighted avg	0.62	0.62	0.62	800

# HOME.HTML

The screenshot shows the Visual Studio Code interface with the following components:

- Explorer:** Displays the project structure. The 'templates' folder is expanded, showing 'home.html' as the active file. Other files include 'app.py', 'model.pkl', 'my\_scaler.save', and 'water\_quality.png'.
- Editor:** Shows the code for 'home.html'. The code includes a title 'Water Quality', a Bootstrap link, and a styled input field with the following CSS rules:

```
<style>
input {
  text-align: center;
  width: 20%;
  height: 70px;
  font-size: 14px;
  padding-top: 0px;
}
.thick {
  text-decoration-line: underline;
  text-decoration-style: solid;
  text-decoration-color: blue;
}
</style>
</head>
<body style="background-color: powderblue;">
```
- Terminal:** Shows the output of pip commands for installing dependencies:

```
Collecting pillow>=6.2.0
  Downloading Pillow-9.3.0-cp311-cp311-win_amd64.whl (2.5 MB)
    2.5/2.5 MB 4.8 MB/s eta 0:00:00
Collecting pyparsing>=2.2.1
  Downloading pyparsing-3.0.9-py3-none-any.whl (98 kB)
    98.3/98.3 kB 352.4 kB/s eta 0:00:00
Requirement already satisfied: python-dateutil>=2.7 in c:\users\priyadarshini\desktop\ibm kicha\sprint 4\env\lib\site-packages (from matplotlib (2.8.2))
Requirement already satisfied: six>=1.5 in c:\users\priyadarshini\desktop\ibm kicha\sprint 4\env\lib\site-packages (from python-dateutil>=2.7->matplotlib (1.16.0))
Installing collected packages: pyparsing, pillow, kiwisolver, fonttools, cycycler, contourpy, packaging, matplotlib
Successfully installed contourpy-1.0.6 cycycler-0.11.0 fonttools-4.38.0 kiwisolver-1.4.4 matplotlib-3.6.2 packaging-21.3 pillow-9.3.0 pyparsing-3.0.9
(env) PS C:\Users\Priyadarshini\Desktop\ibm kicha\sprint 4>
```



**LINKS:**

GITHUB :

<https://github.com/IBM-EPBL/IBM-Project-20482-1659720433>