

APPENDIX

Arduino Uno code

```
1. #include <DHT.h>
2. #include <HCSR04.h>
3. #define DHT_PIN A0
4. #define DHTTYPE DHT11
5. DHT dht(DHT_PIN, DHTTYPE);
6. //HCSR04 HC(2,new int [2]{5,6},2);
7. int Temp;
8. int y;
9. int x;
10. int vib = A5;
11. String life;
12. const int TriggerPin = 2;
13. const int TriggerPinx = 4;
14. const int EchoPin = 6;
15. const int EchoPinx = 5;
16. long duration, distance;
17. long durationx, distancex;
18.
19. void setup() {
20.   Serial.begin(9600);
21.   dht.begin();
22.   pinMode(TriggerPin,OUTPUT);
23.   pinMode(EchoPin,INPUT);
24.   pinMode(TriggerPinx,OUTPUT);
25.   pinMode(EchoPinx,INPUT);
26. }
27.
28. void loop() {
29.   String req = "";
30.   while(Serial.available()>0){
31.     req += char(Serial.read());
32.   }
33.   req.trim();
34.   if(req=="Yes"){
35.     sendata();
36.   }
37.   req = "";
38.   delay(1000);
39. }
40.
41. void sendata(){
42.   Temp = dht.readTemperature();
43.   int VibFix = analogRead(vib);
44.   digitalWrite(TriggerPin, LOW);
45.   delayMicroseconds(2);
46.   digitalWrite(TriggerPin, HIGH);
47.   delayMicroseconds(10);
48.   digitalWrite(TriggerPin, LOW);
49.   duration = pulseIn(EchoPin, HIGH);
50.   distance = (duration/2) /29.1;
```

```

51.     duration = 550-duration;
52.     y = abs(duration);
53.     delay(1000);
54.
55.     digitalWrite(TriggerPinx, LOW);
56.     delayMicroseconds(2);
57.     digitalWrite(TriggerPinx, HIGH);
58.     delayMicroseconds(10);
59.     digitalWrite(TriggerPinx, LOW);
60.     durationx = pulseIn(EchoPinx, HIGH);
61.     distancex = (durationx/2) /29.1;
62.     durationx = 1060-durationx;
63.     x = abs(durationx);
64.     delay(1000);
65.     if ((Temp <= 30) && (VibFix <= 50) && (x <= 250) && (y <= 150))
66.     {
67.         life=">5Tahun";
68.     }
69.     else if ((Temp <= 30) && (VibFix <= 50) && (x <= 250) && (y >
150))
70.     {
71.         life=">5Tahun";
72.     }
73.     else if ((Temp <= 30) && (VibFix <= 50) && (x > 250) && (y <=
150))
74.     {
75.         life=">5Tahun";
76.     }
77.     else if ((Temp <= 30) && (VibFix > 50) && (x <= 250) && (y <=
150))
78.     {
79.         life=">5Tahun";
80.     }
81.     else if ((Temp <= 30) && (VibFix <= 50) && (x > 250) && (y > 150))
82.     {
83.         life="<5Tahun";
84.     }
85.     else if ((Temp <= 30) && (VibFix > 50) && (x <= 250) && (y > 150))
86.     {
87.         life="<5Tahun";
88.     }
89.     else if ((Temp <= 30) && (VibFix > 50) && (x > 250) && (y <= 150))
90.     {
91.         life="<5Tahun";
92.     }
93.     else if ((Temp <= 30) && (VibFix > 50) && (x > 250) && (y > 150))
94.     {
95.         life="<5Tahun";
96.     }
97.     else if ((Temp > 30) && (VibFix <= 50) && (x <= 250) && (y <=
150))
98.     {
99.         life=">5Tahun";
100.    }
101.    else if ((Temp > 30) && (VibFix <= 50) && (x <= 250) && (y > 150))
102.    {

```

```

103.     life="<5Tahun";
104.     }
105.     else if ((Temp > 30) && (VibFix <= 50) && (x > 250) && (y <= 150))
106.     {
107.         life="<5Tahun";
108.     }
109.     else if ((Temp > 30) && (VibFix > 50) && (x <= 250) && (y <= 150))
110.     {
111.         life="<5Tahun";
112.     }
113.     else if ((Temp > 30) && (VibFix <= 50) && (x > 250) && (y > 150))
114.     {
115.         life="<5Tahun";
116.     }
117.     else if ((Temp > 30) && (VibFix > 50) && (x <= 250) && (y > 150))
118.     {
119.         life="<5Tahun";
120.     }
121.     else if ((Temp > 30) && (VibFix > 50) && (x > 250) && (y <= 150))
122.     {
123.         life="<5Tahun";
124.     }
125.     else
126.     {
127.         life="<5Tahun";
128.     }
129.     String data = String(Temp)+"#" +String(VibFix)+"#" +String(x)
+String(y)+"#" +String(life);
130.     Serial.println(data); //Send to NodeMCU
131. }

```

NodeMCU Code

```

1. #include <SoftwareSerial.h>
2. #include <ESP8266WiFi.h>
3. #include <ESP8266HTTPClient.h>
4.
5. SoftwareSerial DataSerial(12,13);
6. unsigned long previousMillis = 0;
7. const long interval = 3000;
8. String arrData[5];
9.
10. int Temperature;
11. int Vibration;
12. int X;
13. int Y;
14. String Life;
15.
16.
17. const char *ssid = "Bhaskara";
18. const char *password = "cobatanyampos";
19. const char *host = "192.168.1.12";
20.
21. void setup() {
22.     Serial.begin(9600);

```

```

23.   DataSerial.begin(9600);
24.   WiFi.mode(WIFI_STA);
25.   WiFi.begin(ssid, password);
26.   while (WiFi.status() != WL_CONNECTED){
27.       delay(500);
28.   }
29.   Serial.println("");
30. }
31.
32. void loop() {
33.     unsigned long currentMillis = millis();
34.     if(currentMillis - previousMillis >= interval){
35.         previousMillis = currentMillis;
36.         String data = "";
37.         while(DataSerial.available()>0){
38.             data += char(DataSerial.read());
39.         }
40.         data.trim();
41.         if(data!=""){
42.             int index = 0;
43.             for(int i=0; i<data.length(); i++){
44.                 char delimiter = '#';
45.                 if(data[i] != delimiter)
46.                     arrData[index] += data[i];
47.                 else
48.                     index++;
49.             }
50.             if(index == 4){
51.                 Serial.println("Temperature : " + arrData[0]);
52.                 Serial.println("Vibration : " + arrData[1]);
53.                 Serial.println("X : " + arrData[2]);
54.                 Serial.println("Y : " + arrData[3]);
55.                 Serial.println("Life : " + arrData[4]);
56.                 Serial.println();
57.             }
58.             Temperature = arrData[0].toInt();
59.             Vibration = arrData[1].toInt();
60.             X = arrData[2].toInt();
61.             Y = arrData[3].toInt();
62.             Life = arrData[4];
63.             arrData[0]="";
64.             arrData[1]="";
65.             arrData[2]="";
66.             arrData[3]="";
67.             arrData[4]="";
68.         }
69.         DataSerial.println("Yes");
70.     }
71.     WiFiClient client;
72.     const int httpPort = 80;
73.     if (!client.connect(host, httpPort)){
74.         Serial.println("Failed");
75.         return;
76.     }
77.     String url = "/Project/write-data.php?Temperature=";
78.     url += Temperature;

```

```

79. url += "&Vibration=";
80. url += Vibration;
81. url += "&X=";
82. url += X;
83. url += "&Y=";
84. url += Y;
85. url += "&Life=";
86. url += Life;
87. client.print(String("GET ") + url + " HTTP/1.1\r\n" +
88.             "Host: " + host + "\r\n" +
89.             "Connection: close\r\n\r\n");
90. unsigned long timeout = millis();
91. while (client.available() == 0) {
92.     if (millis() - timeout > 1000) {
93.         Serial.println(">>> Client Timeout !");
94.         client.stop();
95.         return;
96.     }
97. }
98. while (client.available()) {
99.     String line = client.readStringUntil('\r');
100.    Serial.print(line);
101. }
102. }

```

HTML Code

```

1. <?php
2.   require("koneksi.php");
3. ?>
4.
5. <!DOCTYPE html>
6. <html>
7.   <head>
8.     <meta http-equiv="refresh" content="5">
9.   </head>
10.  <body>
11.    <style>
12.      #wntable {
13.        border-collapse: collapse;
14.        width: 50%;
15.      }
16.
17.      #wntable td, #wntable th {
18.        border: 1px solid #ddd;
19.        padding: 8px;
20.      }
21.
22.      #wntable tr:nth-child(even){background-color: #f2f2f2;}
23.
24.      #wntable tr:hover {background-color: #ddd;}
25.
26.      #wntable th {
27.        padding-top: 12px;
28.        padding-bottom: 12px;

```

```

29.         text-align: left;
30.         background-color: #00A8A9;
31.         color: white;
32.     }
33. </style>
34.
35. <div id="cards" class="cards" align="center">
36.     <h1> Age Machine Sensor </h1>
37.     <table id="wntable">
38.         <tr>
39.             <th>No</th>
40.             <th>Temperature</th>
41.             <th>Vibration</th>
42.             <th>X</th>
43.             <th>Y</th>
44.             <th>Life</th>
45.         </tr>
46.         <?php
47.
48.             $sql = mysqli_query($koneksi, "SELECT * FROM tblProject
ORDER BY id DESC");
49.
50.             if(mysqli_num_rows($sql) == 0){
51.                 echo '<tr><td colspan="14">Data Tidak Ada.</td></tr>';
52.             }else{
53.                 $no = 1;
54.                 while($row = mysqli_fetch_assoc($sql)){
55.                     echo '
56.                     <tr>
57.                         <td>'. $no. '</td>
58.                         <td>'. $row['Temperature']. '</td>
59.                         <td>'. $row['Vibration']. '</td>
60.                         <td>'. $row['X']. '</td>
61.                         <td>'. $row['Y']. '</td>
62.                         <td>'. $row['Life']. '</td>
63.                     </tr>
64.                     ';
65.                     $no++;
66.                 }
67.             }
68.             ?>
69.         </table>
70.     </div>
71. </body>
72. </html>

```

C45 Algorithm Code

```
1. DROP DATABASE IF EXISTS final;
2. CREATE DATABASE final;
3. USE final;
4.
5. CREATE TABLE tblData(
6. id INT,
7. Temperature INT,
8. Vibration INT,
9. X INT,
10. Y INT,
11. Life varchar(20)
12. );
13.
14. LOAD DATA LOCAL INFILE 'myFile0.csv'
15. INTO TABLE tblData
16. FIELDS TERMINATED BY ','
17. ENCLOSED BY '"'
18. LINES TERMINATED BY '\n'
19. IGNORE 1 ROWS;
20.
21. create table tblCount(
22. atribut varchar(20),
23. info varchar(50),
24. amountdata int,
25. kurang5tahun int,
26. lebih5tahun int,
27. entropy double,
28. gain double
29. );
30.
31. create table tblResult(
32. nomer int primary key AUTO_INCREMENT,
33. atribut varchar(20)
34. );
35.
36. create table tblAtribut
37. (
38. atribut varchar(50),
39. info varchar(50),
40. amountdata int,
41. entropy double,
42. gain double
43. );
44.
45. create table printf
46. (
47. thetext TEXT NOT NULL
48. );
49.
50. delimiter ??
51. create procedure c45al()
52. BEGIN
53.
```

```

54.
55. declare a int default 0;
56.
57.
58. looping: WHILE (a <> 1) do
59.
60.
61. delete from tblCount;
62. delete from tblAtribut;
63.
64. select @amountdata:=count(*)
65. from tblData;
66.
67. select @kurang5tahun:=count(*)
68. from tblData
69. where Life LIKE ('%<5Tahun%');
70.
71. select @lebih5tahun:=count(*)
72. from tblData
73. where Life LIKE ('%>5Tahun%');
74.
75. select
    @entropy:=(-(@kurang5tahun/@amountdata)*log2(@kurang5tahun/@amountdata))
76. +
77. (-(@lebih5tahun/@amountdata)*log2(@lebih5tahun/@amountdata));
78.
79. select @amountdata as TOTALDATA,
80. @kurang5tahun as kurangdari5tahun,
81. @lebih5tahun as lebihdari5tahun,
82. @entropy as ENTROPY;
83.
84. insert into tblCount (atribut, amountdata, kurang5tahun, lebih5tahun,
    entropy)
85. values
86. ('TOTAL DATA', @amountdata, @kurang5tahun, @lebih5tahun, @entropy);
87. select *from tblCount;
88.
89. insert into tblCount
90. (info, amountdata, kurang5tahun, lebih5tahun)
91. select distinct(A.Temperature) as TEMPERATURE, count(A.Temperature)
    as JUMLAHDATA,
92. (
93.     select COUNT(*)
94.     from tblData as B
95.     where B.Life LIKE ('%<5Tahun%') and
96.     B.Temperature = A.Temperature
97. )AS RATINGLOW,
98. (
99.     select COUNT(*)
100.    from tblData as C
101.    where C.Life LIKE ('%>5Tahun%') and
102.    C.Temperature = A.Temperature
103. )as RATINGHIGH
104. from tblData as A
105. group by A.Temperature;
106.

```



```

107.
108.
109. update tblCount set atribut = 'Temperature'
110.     where atribut is NULL;
111.
112. insert into tblCount
113.     (info, amountdata, kurang5tahun, lebih5tahun)
114.     select distinct(A.Vibration) as VIBRATION, count(A.Vibration) as
        JUMLAHDATA,
115.     (
116.         select COUNT(*)
117.         from tblData as B
118.         where B.Life LIKE ('%<5Tahun%') and
119.         B.Vibration = A.Vibration
120.     )AS RATINGLOW,
121.     (
122.         select COUNT(*)
123.         from tblData as C
124.         where C.Life LIKE ('%>5Tahun%') and
125.         C.Vibration = A.Vibration
126.     )as RATINGHIGH
127. from tblData as A
128. group by A.Vibration;
129.
130.
131.
132. update tblCount set atribut = 'Vibration'
133.     where atribut is NULL;
134.
135.
136. insert into tblCount
137.     (info, amountdata, kurang5tahun, lebih5tahun)
138.     select distinct(A.X) as X, count(A.X) as JUMLAHDATA,
139.     (
140.         select COUNT(*)
141.         from tblData as B
142.         where B.Life LIKE ('%<5Tahun%') and
143.         B.X = A.X
144.     )AS RATINGLOW,
145.     (
146.         select COUNT(*)
147.         from tblData as C
148.         where C.Life LIKE ('%>5Tahun%') and
149.         C.X = A.X
150.     )as RATINGHIGH
151. from tblData as A
152. group by A.X;
153.
154.
155.     update tblCount set atribut = 'X'
156.         where atribut is NULL;
157.
158.
159. insert into tblCount
160.     (info, amountdata, kurang5tahun, lebih5tahun)
161.     select distinct(A.Y) as Y, count(A.Y) as JUMLAHDATA,

```

```

162.  (
163.      select COUNT(*)
164.      from tblData as B
165.      where B.Life LIKE ('%<5Tahun%') and
166.      B.Y = A.Y
167.  )AS RATINGLOW,
168.  (
169.      select COUNT(*)
170.      from tblData as C
171.      where C.Life LIKE ('%>5Tahun%') and
172.      C.Y = A.Y
173.  )as RATINGHIGH
174.  from tblData as A
175.  group by A.Y;
176.
177.
178.      update tblCount set atribut = 'Y'
179.      where atribut is NULL;
180.
181.
182.
183.
184.      update tblCount set entropy =
185.      (-(kurang5tahun/amountdata) *log2(kurang5tahun/amountdata))
186.      +
187.      (-(lebih5tahun/amountdata) *log2(lebih5tahun/amountdata));
188.      update tblCount set entropy = 0
189.      where entropy is NULL;
190.      update tblCount set entropy = ROUND(entropy,4);
191.      select *from tblCount;
192.
193.
194.      drop table if exists tblTampung;
195.      create temporary table tblTampung
196.      (
197.          atribut varchar(20),
198.          gain double
199.      );
200.
201.      insert into tblTampung(atribut, gain)
202.      select atribut, @entropy -
SUM((amountdata/@amountdata)*entropy) as COUNTGAIN
203.      from tblCount
204.      group by atribut;
205.
206.      select *from tblTampung;
207.
208.      update tblCount set gain =
209.      ROUND(
210.      (
211.          select tblTampung.gain
212.          from tblTampung
213.          where tblTampung.atribut = tblCount.atribut
214.      ),4);
215.      select *from tblCount;
216.      select * from tblCount order by gain;

```

```


217.      -- select * from tblResult;
218.
219.
220.
221.      insert into tblAtribut(atribut, info, amountdata, entropy,
gain)
222.      select atribut, info, amountdata, entropy, gain
223.      from tblCount
224.      where gain = (select max(gain)from tblCount);
225.
226.      select * from tblAtribut;
227.
228.      -- select info from tblAtribut;
229.
230.      -- menentukan entropy tertinggi
231.      select distinct @collectAtribut:= atribut
232.      from tblAtribut
233.      where gain = (select max(gain) from tblAtribut);
234.
235.      select @collectInformasi:= info
236.      from tblAtribut
237.      where entropy= (select max(entropy) from tblAtribut);
238.
239.      select @collectTotalData:= amountdata
240.      from tblCount
241.      where atribut = 'TOTAL DATA';
242.
243.      insert into tblResult (atribut)values(@collectAtribut);
244.      select * from tblResult;
245.
246.
247.
248.      -- iterasi 2
249.
250.      -- if untuk pilihan menentukan hasil dari
@collectAtribut
251.      if(@collectAtribut = 'Temperature')then
252.
253.
254.          create table tblData2 as
255.              select id from tblData where Temperature =
@collectInformasi;
256.
257.          delete from tblData
258.              where id not in (select id from tblData2);
259.
260.          drop table tblData2;
261.
262.      elseif(@collectAtribut = 'Vibration')then
263.
264.
265.          create table tblData2 as
266.              select id from tblData where Vibration =
@collectInformasi;
267.
268.          delete from tblData

```

```
269.         where id not in (select id from tblData2);
270.
271.     drop table tblData2;
272.
273.     elseif(@collectAtribut = 'X') then
274.
275.         create table tblData2 as
276.             select id from tblData where X = @collectInformasi;
277.
278.         delete from tblData
279.             where id not in (select id from tblData2);
280.
281.         drop table tblData2;
282.
283.
284.     elseif(@collectAtribut = 'Y') then
285.
286.         create table tblData2 as
287.             select id from tblData where Y = @collectInformasi;
288.
289.         delete from tblData
290.             where id not in (select id from tblData2);
291.
292.         drop table tblData2;
293.     end if;
294. -- loop iterasi
295. if (@collectTotalData <= 4) then
296. LEAVE looping;
297. end if;
298. end while looping;
299.
300. END ??
301. delimiter ;
```



5.62% PLAGIARISM
APPROXIMATELY

0.24% IN QUOTES 

Report #14324327

INTRODUCTION Background Machine life prediction plays an important role in the process various industries in the world. The condition of machine components is a special concern for the company, because if there is damage it can result in a decrease in productivity. Therefore, it is very important to diagnose and diagnose damage to the rotating machine before the damage occurs, so that a more appropriate method and time of maintenance can be planned. There are various methods that can be used to measure machine condition, such as vibration analysis, oil analysis, temperature measurement, pressure measurement, flow rate measurement, and others. Measurement of vibration, temperature, and machine displacement are several methods that are quite good for determining machine condition because vibration and temperature are indicators of mechanical conditions and early indicators of defects in a machine as a whole. The higher the vibration and the resulting temperature, this indicates a decrease in the condition of the machine. Productivity will