CHAPTER V

IMPLEMENTATION AND TESTING

5.1 Implementation

5.1.1. NodeMCU & PHP Code

Programming languages on NodeMCU is using C++ language. This program is inputted so that the HC-SR04 ultrasonic sensor works properly and the data can be uploaded to the broker.

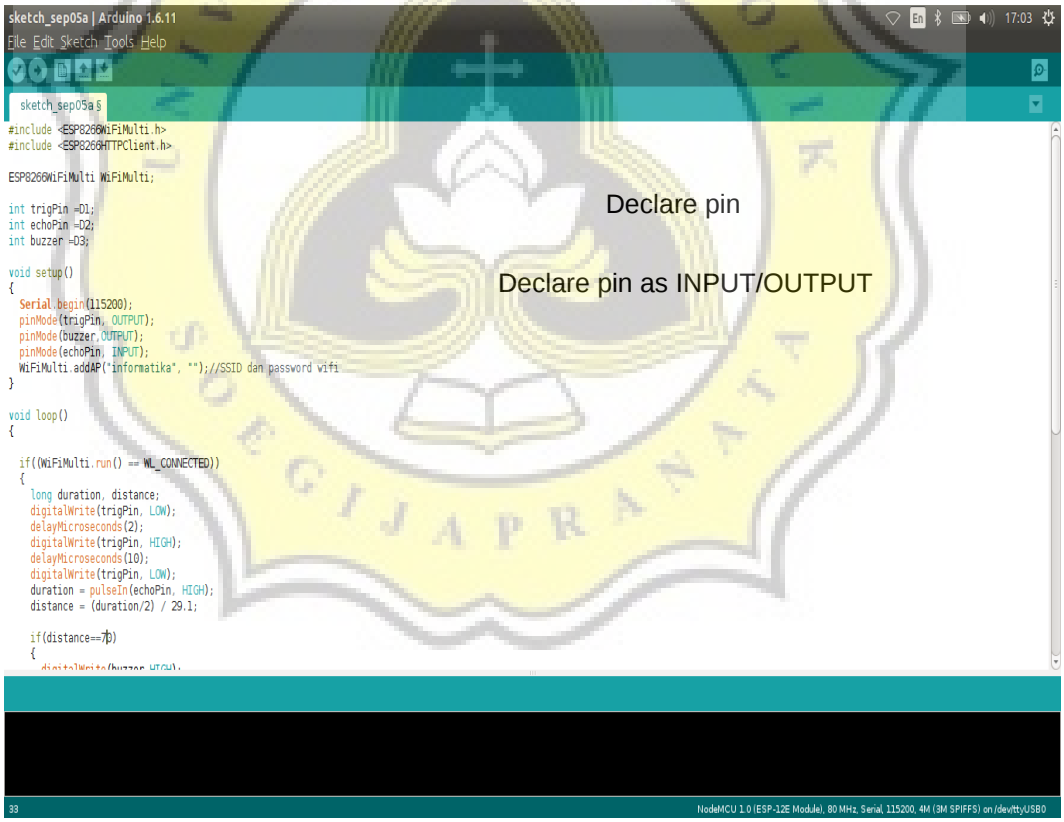


Figure5.8. Code NodeMCU

When the program starts, the first to do is read the pin that has been declared, then search the internet connection of the SSID and password have been set.

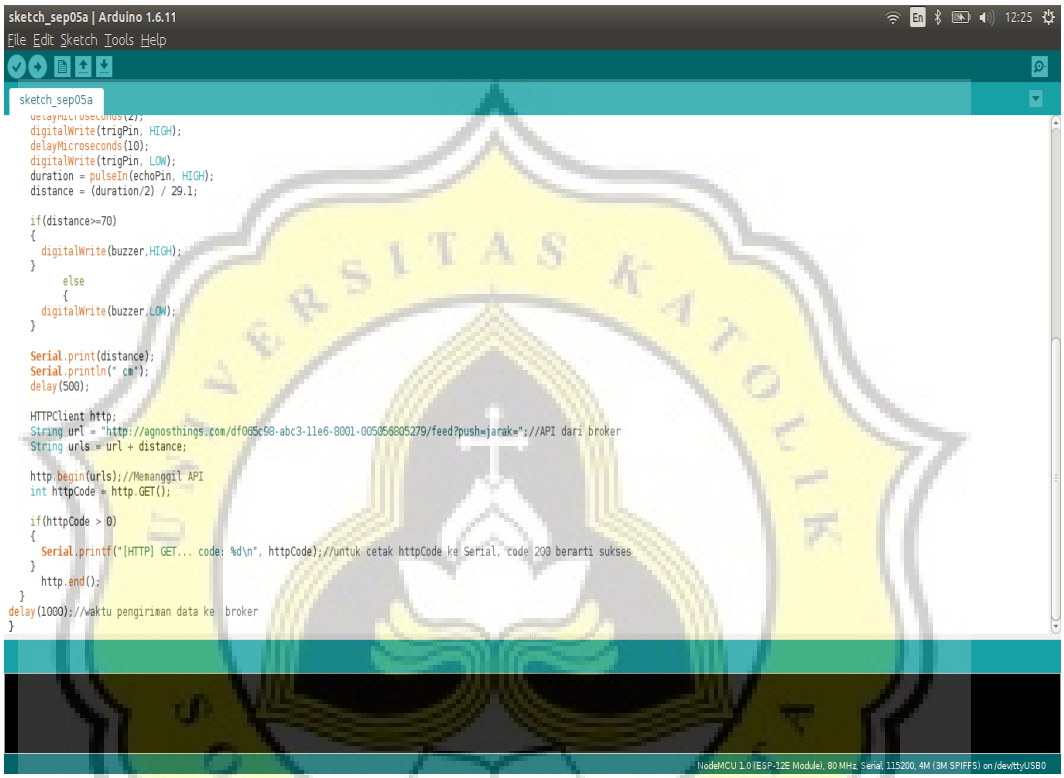


Figure5.9. Code NodeMCU

After that, the program will read the HC-SR04 ultrasonic sensor and obtains the data from sensor. When the distance is more than 70, the buzzer will sound to indicate a warning, then the data from the HC-SR04 ultrasonic sensor to be sent to the broker www.agnosthings.com via API and an internet connection.

node.php (ojo D /media/indra/ojo D/TA/TA) - gedit

```php
<?php
class node
{
        protected $no;
        protected $jarak;
        protected $tanggal;
        protected $next;

        public function setNo($par)
        {
                $this->no = $par;
        }
        public function getNo()
        {
                return $this->no;
        }
        public function setJarak($par)
        {
                $this->jarak = $par;
        }
        public function getJarak()
        {
                return $this->jarak;
        }
        public function setTanggal($par)
        {
                $this->tanggal = $par;
        }
        public function getTanggal()
        {
                return $this->tanggal;
        }
        public function setNext($par)
        {
                $this->next = $par;
        }
```

Figure5.10. Code PHP

Php program here is to create a table of ultrasonic sensor data using data structure arraylist, the data comes from a broker and then inserted into a txt file. First, create a class that contains set and get to put and call data.

ambilData.php (ojo D /media/indra/ojo D/TA/TA) - gedit

```php
<?php
$list = new nodeList();
$myfile = fopen("coba.txt", "r"); //open file txt dari baris pertama
while(!feof($myfile)) //mengecek kursor sampai akhir file
{
        $line = fgets($myfile); //baca 1 baris
        $col = explode('        ', $line); //memisahkan baris ke kolom dengan delimiter tab
        $hari = intval(substr(isset($col[2]) ? $col[2] : null,8,2)); //mencari hari(day of month)
        $hari=$hari-1; //convert ke index (0-30)
        if(isset($list->nodes[$hari]))
        {
                $temp = $list->nodes[$hari];
                while($temp->getNext() != NULL)
                {
                        $temp = $temp->getNext(); //cari node terakhir
                }
                $new = new node();
                $new->setNo(isset($col[0]) ? $col[0] : null); //kalo ga dikasih ini ttp jalan tapi ada error undefined offset
                $new->setJarak(isset($col[1]) ? $col[1] : null);
                $new->setTanggal(isset($col[2]) ? $col[2] : null);
                $temp->setNext($new);
        }
        else
        {
                $list->nodes[$hari] = new node();
                $list->nodes[$hari]->setNo(isset($col[0]) ? $col[0] : null);
                $list->nodes[$hari]->setJarak(isset($col[1]) ? $col[1] : null);
                $list->nodes[$hari]->setTanggal(isset($col[2]) ? $col[2] : null);
        }
}
        fclose($myfile);
?>
```

Figure5.11. Code PHP

After that, read txt file into the PHP program. The reading of files is done per row from the first row to the last row, then define the separator between the columns and make index. Index here using the date by a day, then data is entered into the node that has been made.

5.2 Testing

After ultrasonic sensors work properly and the data can be read using php the next step is the placement of the tool at location, the following is the placement and the program works.

Figure5.12. Placement at Location

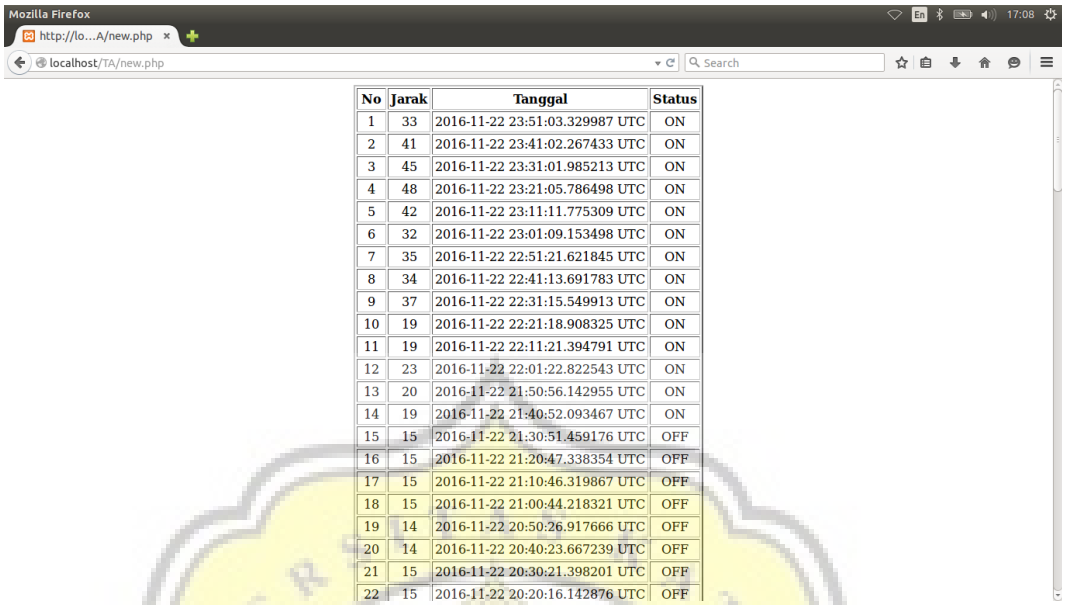
The combined NodeMCU and HC-SR04 ultrasonic sensor is placed on top of the water tank in order to be able to measure the water level.

First column is id, the second column is the distance, the third column is date and time of recording

1	33	2016-11-22 23:51:03.329987 UTC
2	41	2016-11-22 23:41:02.267433 UTC
3	45	2016-11-22 23:31:01.985213 UTC
4	48	2016-11-22 23:21:05.786498 UTC
5	42	2016-11-22 23:11:11.775309 UTC
6	32	2016-11-22 23:01:09.153498 UTC
7	35	2016-11-22 22:51:21.621845 UTC
8	34	2016-11-22 22:41:13.691783 UTC
9	37	2016-11-22 22:31:15.549913 UTC
10	19	2016-11-22 22:21:18.908325 UTC
11	19	2016-11-22 22:11:21.394791 UTC
12	23	2016-11-22 22:01:22.822543 UTC
13	20	2016-11-22 21:50:56.142955 UTC
14	19	2016-11-22 21:40:52.093467 UTC
15	15	2016-11-22 21:30:51.459176 UTC
16	15	2016-11-22 21:20:47.338354 UTC
17	15	2016-11-22 21:10:46.319867 UTC
18	15	2016-11-22 21:00:44.218321 UTC
19	14	2016-11-22 20:50:26.917666 UTC
20	14	2016-11-22 20:40:23.667239 UTC
21	15	2016-11-22 20:30:21.398201 UTC
22	15	2016-11-22 20:20:16.142876 UTC
23	16	2016-11-22 20:10:13.718966 UTC
24	15	2016-11-22 19:59:46.628805 UTC
25	15	2016-11-22 19:49:44.048629 UTC
26	15	2016-11-22 19:39:38.795920 UTC
27	15	2016-11-22 19:29:36.642309 UTC
28	16	2016-11-22 19:19:55.561503 UTC
29	16	2016-11-22 19:09:54.755243 UTC
30	14	2016-11-22 18:59:53.821571 UTC
31	15	2016-11-22 18:49:50.216205 UTC
32	15	2016-11-22 18:39:48.847232 UTC
33	15	2016-11-22 18:29:47.962165 UTC
34	16	2016-11-22 18:19:47.083885 UTC
35	16	2016-11-22 18:09:45.926468 UTC
36	16	2016-11-22 17:59:43.180443 UTC
37	16	2016-11-22 17:49:53.314897 UTC
38	14	2016-11-22 17:39:42.309367 UTC
39	14	2016-11-22 17:29:50.316729 UTC
40	14	2016-11-22 17:19:14.102003 UTC
41	15	2016-11-22 17:09:13.176534 UTC
42	17	2016-11-22 16:59:12.294894 UTC
43	16	2016-11-22 16:49:10.889878 UTC
44	15	2016-11-22 16:39:09.994563 UTC
45	15	2016-11-22 16:29:09.045309 UTC

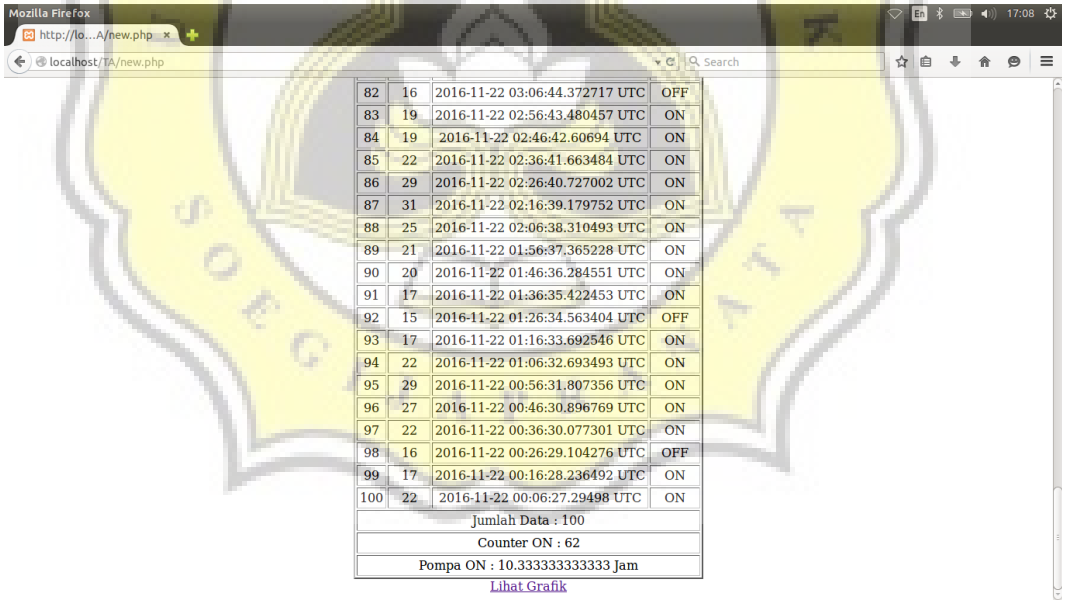
Figure5.13. File txt

Figure5.14. Insert Date to Search



No	Jarak	Tanggal	Status
1	33	2016-11-22 23:51:03.329987 UTC	ON
2	41	2016-11-22 23:41:02.267433 UTC	ON
3	45	2016-11-22 23:31:01.985213 UTC	ON
4	48	2016-11-22 23:21:05.786498 UTC	ON
5	42	2016-11-22 23:11:11.775309 UTC	ON
6	32	2016-11-22 23:01:09.153498 UTC	ON
7	35	2016-11-22 22:51:21.621845 UTC	ON
8	34	2016-11-22 22:41:13.691783 UTC	ON
9	37	2016-11-22 22:31:15.549913 UTC	ON
10	19	2016-11-22 22:21:18.908325 UTC	ON
11	19	2016-11-22 22:11:21.394791 UTC	ON
12	23	2016-11-22 22:01:22.822543 UTC	ON
13	20	2016-11-22 21:50:56.142955 UTC	ON
14	19	2016-11-22 21:40:52.093467 UTC	ON
15	15	2016-11-22 21:30:51.459176 UTC	OFF
16	15	2016-11-22 21:20:47.338354 UTC	OFF
17	15	2016-11-22 21:10:46.319867 UTC	OFF
18	15	2016-11-22 21:00:44.218321 UTC	OFF
19	14	2016-11-22 20:50:26.917666 UTC	OFF
20	14	2016-11-22 20:40:23.667239 UTC	OFF
21	15	2016-11-22 20:30:21.398201 UTC	OFF
22	15	2016-11-22 20:20:16.142876 UTC	OFF

Figure5.15. Table PHP From File txt



82	16	2016-11-22 03:06:44.372717 UTC	OFF
83	19	2016-11-22 02:56:43.480457 UTC	ON
84	19	2016-11-22 02:46:42.60694 UTC	ON
85	22	2016-11-22 02:36:41.663484 UTC	ON
86	29	2016-11-22 02:26:40.727002 UTC	ON
87	31	2016-11-22 02:16:39.179752 UTC	ON
88	25	2016-11-22 02:06:38.310493 UTC	ON
89	21	2016-11-22 01:56:37.365228 UTC	ON
90	20	2016-11-22 01:46:36.284551 UTC	ON
91	17	2016-11-22 01:36:35.422453 UTC	ON
92	15	2016-11-22 01:26:34.563404 UTC	OFF
93	17	2016-11-22 01:16:33.692546 UTC	ON
94	22	2016-11-22 01:06:32.693493 UTC	ON
95	29	2016-11-22 00:56:31.807356 UTC	ON
96	27	2016-11-22 00:46:30.896769 UTC	ON
97	22	2016-11-22 00:36:30.077301 UTC	ON
98	16	2016-11-22 00:26:29.104276 UTC	OFF
99	17	2016-11-22 00:16:28.236492 UTC	ON
100	22	2016-11-22 00:06:27.29498 UTC	ON
Jumlah Data : 100			
Counter ON : 62			
Pompa ON : 10.333333333333 Jam			

Lihat Grafik

Figure5.16. Table PHP From File txt

The first copy data from a broker to a txt file, then entered in the program. Insert date you want to search the data. After that the program will read input from the user and select it if it makes in the Type 0 (years), type 1 (month-year), or type 2 (date-month-year). Then the program will display the data corresponding to that sought by the user. The data is the id, distance, date and time of recording data, and status (obtained from the requirement if distance> 15). At the end of the table there is a summary of the data consists of the count of data, counter ON, pump ON, and graphs.

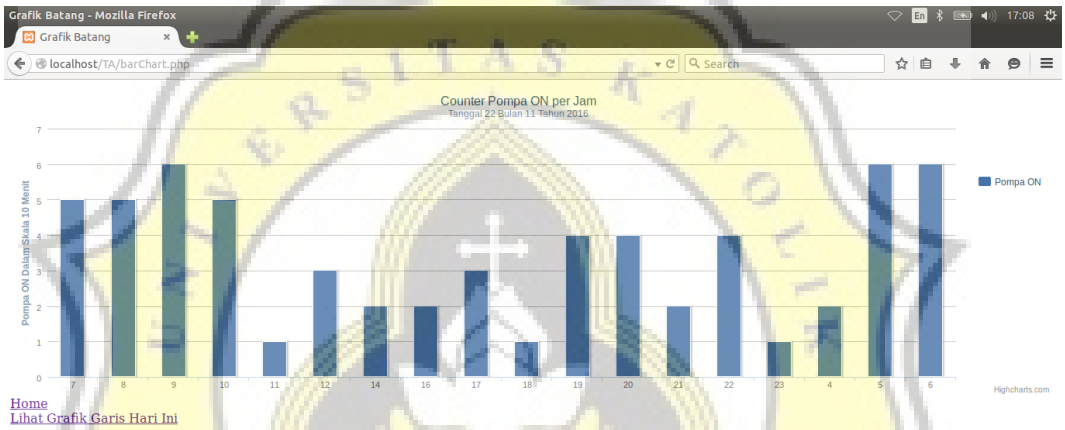


Figure5.17. Graph Hour

All the data in text file can be figured as the graph above. The graph represent data every 10 minutes in a day. It can be seen that at 4 am until 10 am the sensor show that water pump is continue working, while between 11 am until 11 pm water pump working only several minute/hour.

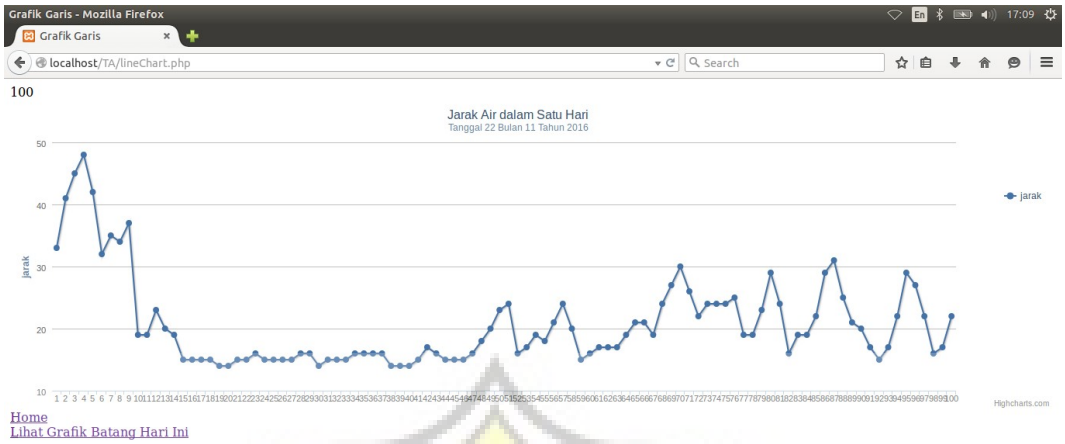


Figure5.18. Graph Data

The graph above shows the distance based on the amount of data recorded per day. From the graph it can be seen that data number 1 until 14 show that water pump continue working, while between data number 15 until 46 water pump are not too often work, then from data number 47 until 100 water pump continue work again.

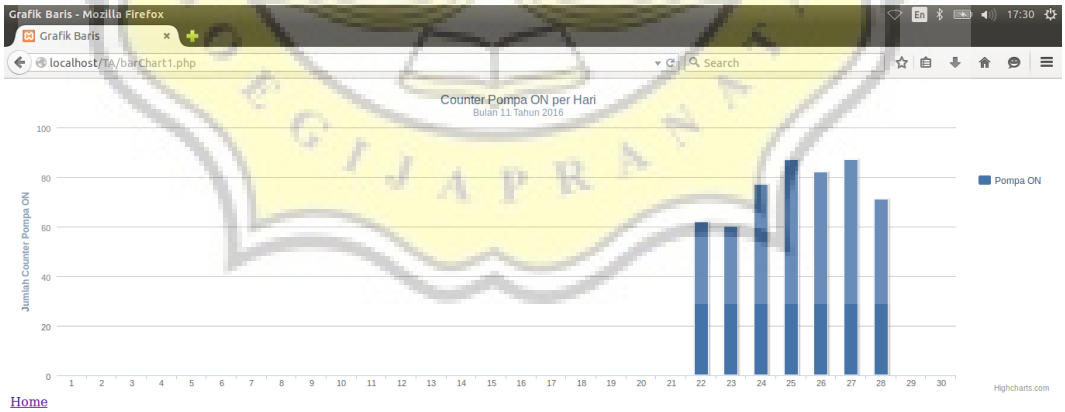


Figure5.19. Graph Day

The graph above shows the counter ON water pump per day in a month. It can seen every day the water pump works not always same, graph above show that on 22 until 23 percentage of working water pump lower, then on 24 until 28 looks that water pump more often works.

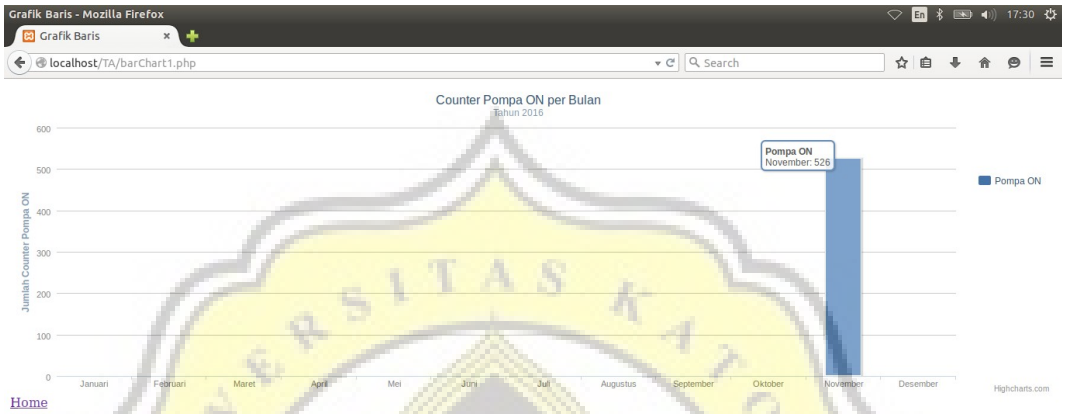


Figure5.20. Graph Month

This graph above shows the counter ON water pump per month in a year. It can seen water pump working in November