Certification Page

This page must be the first page of your uploaded document.
Your assignment will not be graded without this page (completed with your full name in the area provided) as the first page of your uploaded document.

I, ___***Ulvi Bajarani***_____, certify that the work I am uploading represents efforts of my team member and mine per the following task-contribution table, and is not copied from anyone else or any other resource (such as Internet).

	Tasks on the schedule for this quiz are 10, 11. You may break these tasks further down into subtasks as needed.	Percent contributed	Percent contributed	Total percent
#	*Subtask* Description	Ulvi Bajarani	Gwendolyn Poulos	
10.1	Design Login for Website	50%	50%	100%
10.2	Design Webpage	50%	50%	100%
11.1	Interface RPi with DB	50%	50%	100%
11.2	Interface App with DB	50%	50%	100%
11.3	Interface Website with DB	50%	50%	100%

The upgraded code of sync_rpi_data.php

```
<?php
require_once __DIR__ . '/../../required/db_connect.php';
$input = file_get_contents("php://input");
$error = 0;
$out_json = array();
$out_json['success'] = 1; //assume success
$SW1_status = 0;
$SW2_status = 0;
$LED1_status = 0;
$LED2_status = 0;
if ($input)
{
   $json = json_decode($input, true); //check if it json input
   if (json_last_error() == JSON_ERROR_NONE)
   {
      if (isset($json["username"]) && isset($json["password"]) && isset($json["SW1"]) &&
isset($json["SW2"]) && isset($json["LED1"]) && isset($json["LED2"]))
      {
         $in_username = $json["username"];
         $in_password = $json["password"]; //if the expected fields are not null, get them
         $in_SW1 = $json["SW1"];
                        $in_SW2 = $json["SW2"];
         $in_LED1 = $json["LED1"];
         $in_LED2 = $json["LED2"];
         if ($stmt = $mysqli->prepare("SELECT password FROM Users WHERE pname = ? LIMIT
1"))
         {
            $stmt->bind_param('s', $in_username);
            $stmt->execute();
            $stmt->store_result(); //store_result to get num_rows etc.
            $stmt->bind_result($db_password); //get the hashed password
            $stmt->fetch();
            if ($stmt->num_rows == 1)
            { //if user exists, verify the password
               if (password_verify($in_password, $db_password))
               {
                                                $stmt->close();
                  if ($stmt = $mysqli->prepare("UPDATE SensorsAndActuators SET DeviceStatus=?
WHERE DevID = 'SW1'"))
                  { //update SW1
                     $stmt->bind_param('i', $in_SW1);
                     $stmt->execute();
                  }
                  else
                  {
                     $error = 1;
                  }
```

```
                $stmt->close();
                if (!$error && ($stmt = $mysqli->prepare("SELECT DeviceStatus FROM
SensorsAndActuators WHERE DevID = 'SW1'")))
                { //read SW1
                   $stmt->execute();
                   $stmt->bind_result($SW1_status);
                   $stmt->fetch();
                }
                else
                {
                   $error = 2;
                }
                $stmt->close();
                if ($stmt = $mysqli->prepare("UPDATE SensorsAndActuators SET DeviceStatus=?
WHERE DevID = 'SW2'"))
                { //update SW1
                   $stmt->bind_param('i', $in_SW2);
                   $stmt->execute();
                }
                else
                {
                   $error = 1;
                }
                $stmt->close();
                if (!$error && ($stmt = $mysqli->prepare("SELECT DeviceStatus FROM
SensorsAndActuators WHERE DevID = 'SW2'")))
                { //read SW1
                   $stmt->execute();
                   $stmt->bind_result($SW2_status);
                   $stmt->fetch();
                }
                else
                {
                   $error = 2;
                }
                $stmt->close();
                if (!$error && ($stmt = $mysqli->prepare("SELECT DeviceStatus FROM
SensorsAndActuators WHERE DevID = 'LED1'")))
                { //read LED1
                   $stmt->execute();
                   $stmt->bind_result($LED1_status);
                   $stmt->fetch();
                }
                else
                {
                   $error = 3;
                }
                $stmt->close();
```

```
                    if (!$error && ($stmt = $mysqli->prepare("SELECT DeviceStatus FROM
SensorsAndActuators WHERE DevID = 'LED2'")))
                    { //read LED1
                       $stmt->execute();
                       $stmt->bind_result($LED2_status);
                       $stmt->fetch();
                    }
                    else
                    {
                       $error = 3;
                    }
                    $stmt->close();
                 }
                 else
                 {
                    $error = 4;
                 }
              }
              else
              {
                 $error = 5;
              }
           }
           else
           {
              $error = 6;
           }
        }
        else
        {
           $error = 7;
        }
     }
     else
     {
        $error = 8;
     }
}
else
{
   $error = 9;
}
if ($error)
{
   $out_json['success'] = 0; //flag failure

}
$out_json['SW1'] = $SW1_status;
$out_json['SW2'] = $SW2_status;
```

```
$out_json['LED1'] = $LED1_status;
$out_json['LED2'] = $LED2_status;
$out_json['error'] = $error; //provide error (if any) number for debugging
echo json_encode($out_json); //encode the data in json format

?>
```

The upgraded code of sync_android_data.php

```
<?php
require_once __DIR__ . '/../../required/db_connect.php';
$input = file_get_contents("php://input");
$error = 0;
$out_json = array();
$out_json['success'] = 1; //assume success
$SW1_status = 0;
$SW2_status = 0;
$LED1_status = 0;
$LED2_status = 0;
if ($input)
{
   $json = json_decode($input, true); //check if it json input
   if (json_last_error() == JSON_ERROR_NONE)
   {
      if (isset($json["username"]) && isset($json["password"]) && isset($json["SW1"]) &&
isset($json["LED1"]) && isset($json["SW2"]) && isset($json["LED2"]))
      {
         $in_username = $json["username"];
         $in_password = $json["password"]; //if the expected fields are not null, get them
         $in_SW1 = $json["SW1"];
         $in_SW2 = $json["SW2"];
         $in_LED1 = $json["LED1"];
         $in_LED2 = $json["LED2"];
         if ($stmt = $mysqli->prepare("SELECT password FROM Users WHERE pname = ? LIMIT
1"))
         {
            $stmt->bind_param('s', $in_username);
            $stmt->execute();
            $stmt->store_result(); //store_result to get num_rows etc.
            $stmt->bind_result($db_password); //get the hashed password
            $stmt->fetch();
            if ($stmt->num_rows == 1)
            { //if user exists, verify the password
               if (password_verify($in_password, $db_password))
               {
                  $stmt->close();
                  if ($stmt = $mysqli->prepare("UPDATE SensorsAndActuators SET DeviceStatus=?
WHERE DevID = 'SW1'"))
                  { //update SW1
```

```
                $stmt->bind_param('i', $in_SW1);
                $stmt->execute();
            }
            else
            {
                $error = 1;
            }
            $stmt->close();
            if (!$error && ($stmt = $mysqli->prepare("SELECT DeviceStatus FROM
SensorsAndActuators WHERE DevID = 'SW1'")))
            { //read SW1
                $stmt->execute();
                $stmt->bind_result($SW1_status);
                $stmt->fetch();
            }
            else
            {
                $error = 2;
            }
            $stmt->close();
            if ($stmt = $mysqli->prepare("UPDATE SensorsAndActuators SET DeviceStatus=?
WHERE DevID = 'SW2'"))
            { //update SW1
                $stmt->bind_param('i', $in_SW2);
                $stmt->execute();
            }
            else
            {
                $error = 1;
            }
            $stmt->close();
            if (!$error && ($stmt = $mysqli->prepare("SELECT DeviceStatus FROM
SensorsAndActuators WHERE DevID = 'SW2'")))
            { //read SW1
                $stmt->execute();
                $stmt->bind_result($SW2_status);
                $stmt->fetch();
            }
            else
            {
                $error = 2;
            }
            $stmt->close();
            if (!$error && ($stmt = $mysqli->prepare("SELECT DeviceStatus FROM
SensorsAndActuators WHERE DevID = 'LED1'")))
            { //read LED1
                $stmt->execute();
                $stmt->bind_result($LED1_status);
                $stmt->fetch();
```

```
                    }
                    else
                    {
                        $error = 3;
                    }
                    $stmt->close();
                    if (!$error && ($stmt = $mysqli->prepare("SELECT DeviceStatus FROM
SensorsAndActuators WHERE DevID = 'LED2'")))
                    { //read LED1
                        $stmt->execute();
                        $stmt->bind_result($LED2_status);
                        $stmt->fetch();
                    }
                    else
                    {
                        $error = 3;
                    }
                    $stmt->close();
                }
                else
                {
                    $error = 4;
                }
            }
            else
            {
                $error = 5;
            }
        }
        else
        {
            $error = 6;
        }
    }
    else
    {
        $error = 7;
    }
  }
  else
  {
    $error = 8;
  }
}
else
{
  $error = 9;
}
if ($error)
```

```
{
   $out_json['success'] = 0; //flag failure

}
$out_json['SW1'] = $SW1_status;
$out_json['SW2'] = $SW1_status;
$out_json['LED1'] = $LED1_status;
$out_json['LED2'] = $LED1_status;
$out_json['error'] = $error; //provide error (if any) number for debugging
echo json_encode($out_json); //encode the data in json format

?>
```

The upgraded code of p2_t8.py

```
#!/usr/bin/python
import requests #import JSONRequests library
import time #import time library for sleep function
import datetime #import datetime library for timestamp
import RPi.GPIO as GPIO #import GPIO library
GPIO.setmode(GPIO.BCM) #set the pins according to BCM scheme
GPIO.setup(4,GPIO.OUT) #configure BCM Pin #4 as OUTPUT
GPIO.setup(6,GPIO.OUT) #configure BCM Pin #6 as OUTPUT
GPIO.setup(17,GPIO.IN) #configure BCM Pin #17 as INPUT
GPIO.setup(22,GPIO.IN) #configure BCM Pin #17 as INPUT
i=0; n=3; delay=5  #limit number of tries to 5 (initially set it to 1 for debugging)
while i<n:
        LED1=GPIO.input(4) #read what BCM Pin #4 is set to (LED1)
        LED2=GPIO.input(6) #read what BCM Pin #6 is set to (LED2)
        SW1=GPIO.input(17) #read the status of BCM Pin #17 (SW1)
        SW2=GPIO.input(22) #read the status of BCM Pin #22 (SW2)
        data = {'username': 'Test', 'password': 'Tester3', 'SW1': SW1, 'SW2': SW2, 'LED1': LED1,
'LED2': LED2}
        res = requests.post("https://team2project3342.online/scripts/sync_rpi_data.php", json=data)
        #in case of errors (especially, syntax) , you may want to print res.text and comment out the
statements below
        r = res.json()
        ts = datetime.datetime.now() #get the time stamp
        print "==============Server Response at " + str(ts) + "=============="
        if r['success']==1:
                print "+++++Server request successful: "
                if LED1!=r['LED1']:
                        print "Changing LED status as requested by the server"
                        if r['LED1']==1:
                                GPIO.output(4,GPIO.HIGH)
                        else: GPIO.output(4,GPIO.LOW)
                if LED2!=r['LED2']:
                        print "Changing LED status as requested by the server"
                        if r['LED2']==1:
```

```
                                GPIO.output(6,GPIO.HIGH)
                            else: GPIO.output(6,GPIO.LOW)
                print "The status of LED1 is " + str(r['LED1'])
                print "The status of LED2 is " + str(r['LED2'])
                print "The status of SW1 is " + str(r['SW1'])
                print "The status of SW2 is " + str(r['SW2'])
        else:
                print ">>>>> Server request failed - Error #" + str(r['error'])
        time.sleep(delay) #wait for delay seconds before sending another request
        i+=1
GPIO.cleanup()
```

The code of Projectv3.py, which is the android app on kivy.

```
import time
import kivy
from kivy.app import App
from kivy.uix.button import Button
from kivy.uix.boxlayout import BoxLayout
from kivy.uix.label import Label
from kivy.uix.switch import Switch
from kivy.clock import Clock
from functools import partial
import requests

class BoxLayoutApp(App):

    #def __init__(self, **kwargs):
        #super(BoxLayoutApp,self).__init__(**kwargs)
    def build(self):
        self.superBox = BoxLayout(orientation ='vertical')

        self.VB1 = BoxLayout(orientation ='vertical')

        self.lbl1 = Label(text="Monitoring System")
        self.VB1.add_widget(self.lbl1)

        localtime = time.asctime(time.localtime(time.time()))
        self.lbl2 = Label(text=localtime)
        self.VB1.add_widget(self.lbl2)

        self.HB1 = BoxLayout(orientation ='horizontal')

        self.lbl3 = Label(text="Status SW1:")
        self.HB1.add_widget(self.lbl3)

        self.sw1=Switch(active=False,
                disabled=True)
```

```
self.HB1.add_widget(self.sw1)
#self.sw1.bind(active=switch_callback1)

self.HB2 = BoxLayout(orientation ='horizontal')
self.lbl4 = Label(text="Status SW2:")
self.HB2.add_widget(self.lbl4)

self.sw2=Switch(active=False,
            disabled=True)
self.HB2.add_widget(self.sw2)
#self.sw2.bind(active=switch_callback2)

self.HB3 = BoxLayout(orientation ='horizontal')
self.lbl5 = Label(text="Status LED1:")
self.HB3.add_widget(self.lbl5)

self.led1=Switch(active=False,
             disabled=False)
self.HB3.add_widget(self.led1)
self.led1.bind(active=switch_callback3)

self.HB4 = BoxLayout(orientation ='horizontal')
self.lbl6 = Label(text="Status LED2:")
self.HB4.add_widget(self.lbl6)

self.led2=Switch(active=False,
             disabled=False)
self.HB4.add_widget(self.led2)
self.led2.bind(active=switch_callback4)

self.HB5 = BoxLayout(orientation ='horizontal')
self.lbl7 = Label(text="Alarm1")
self.HB5.add_widget(self.lbl7)

self.settings5=Switch(active=False)
self.HB5.add_widget(self.settings5)
self.settings5.bind(active=self.switch_callback5)

self.lbl71=Label(text="On since: ")
self.HB5.add_widget(self.lbl71)
self.btn5 = Button(text="acknowledge")
self.btn5.bind(on_press=self.pressed2)
self.HB5.add_widget(self.btn5)

self.HB6 = BoxLayout(orientation ='horizontal')
self.lbl8 = Label(text="Alarm2")
self.HB6.add_widget(self.lbl8)

self.settings6=Switch(active=False)
```

```
        self.HB6.add_widget(self.settings6)
        self.settings6.bind(active=self.switch_callback6)

        self.lbl81=Label(text="On since: ")
        self.HB6.add_widget(self.lbl81)

        self.btn6 = Button(text="acknowledge")
        self.btn6.bind(on_press=self.pressed3)
        self.HB6.add_widget(self.btn6)

        self.HB7 = BoxLayout(orientation ='horizontal')
        self.btn7 = Button(text="Dispense Norepinephrine")
        self.btn7.bind(on_press=self.pressed1)
        self.HB7.add_widget(self.btn7)
        self.btn8 = Button(text="Dispense Nitroglycerin")
        self.btn8.bind(on_press=self.pressed)
        self.HB7.add_widget(self.btn8)
        self.btn9 = Button(text="Dispense Auto")
        self.btn9.bind(on_press=self.presseda)
        self.HB7.add_widget(self.btn9)

        self.superBox.add_widget(self.VB1)
        self.superBox.add_widget(self.HB1)
        self.superBox.add_widget(self.HB2)
        self.superBox.add_widget(self.HB3)
        self.superBox.add_widget(self.HB4)
        self.superBox.add_widget(self.HB5)
        self.superBox.add_widget(self.HB6)
        self.superBox.add_widget(self.HB7)

        #schedule the JSONrequest function to trigger every 5 seconds to read/write database
        event = Clock.schedule_interval(partial(self.JSONrequest),5)

        return self.superBox

    def JSONrequest(self, *largs):
        if(self.sw1.active==True):
            SW1=1
        else:
            SW1=0
        if(self.sw2.active==True):
            SW2=1
        else:
            SW2=0
        if(self.led1.active==True):
            LED1=1
        else:
            LED1=0
        if(self.led2.active==True):
```

```
        LED2=1
    else:
        LED2=0
    data={'username':'Test', 'password':'Tester3','SW1':SW1,'LED1':LED1,'SW2':SW2,'LED2':LED2}
    res=requests.post("http://team2project3342.online/scripts/sync_android_data.php",json=data)
    print(res)
    r=res.json()
    if(SW1!=r['SW1']):
      print("Changing SW1 status to the value in the database.")
      if self.sw1.active==True:
          self.sw1.active=False
          self.led1.active=False
      else:
          self.sw1.active=True
    if(SW2!=r['SW2']):
      print("Changing SW2 status to the value in the database.")
      if self.sw2.active==True:
          self.sw2.active=False
          self.led2.active=False
      else:
          self.sw2.active=True
    return

def pressed(self,instance):
    print ("you picked " + instance.text)
    self.led2.active=True

def pressed1(self,instance):
    print ("you picked " + instance.text)
    self.led1.active=True

def presseda(self,instance):
    print ("you picked " + instance.text)
    if self.sw1.active==True:
        self.led1.active=True
    else:
        self.led1.active=False
    if self.sw2.active==True:
        self.led2.active=True
    else:
        self.led2.active=False

def pressed2(self,instance):
    print ("you acknowleged Alarm1")
    self.settings5.active=False
    print("turning alarm1 off")
def pressed3(self,instance):
    print ("you acknowleged Alarm2")
    self.settings6.active=False
```

```
        print("turning alarm1 off")
    def switch_callback5(self, switchObject, switchValue):
        print('Value of Alarm1: ', switchValue)
        if self.settings5.active==True:
            ltm = time.asctime(time.localtime(time.time()))
            self.lbl71.text+=ltm
        else:
            self.lbl71.text = "On since: "
    def switch_callback6(self, switchObject, switchValue):
        print('Value of Alarm2: ', switchValue)
        if self.settings6.active==True:
            ltm = time.asctime(time.localtime(time.time()))
            self.lbl81.text+=ltm
        else:
            self.lbl81.text = "On since: "

def switch_callback1(switchObject, switchValue):
        print('Value of SW1: ', switchValue)
def switch_callback2(switchObject, switchValue):
        print('Value of SW2: ', switchValue)
def switch_callback3(switchObject, switchValue):
        print('Value of LED1: ', switchValue)
def switch_callback4(switchObject, switchValue):
        print('Value of LED2: ', switchValue)

#class myApp(App):
#    def build(self):
 #       return BoxLayout()

if __name__=="__main__":
    myApp=BoxLayoutApp()
    myApp.run()
   # myApp().run()
```

The code of index.php

```
<?php
require_once __DIR__ . '/../required/db_connect.php';
?>
<html>
    <head>
        <title>Team Project</title>
        <link rel="stylesheet" type="text/css" href="prjct.css"/>
    </head>
    <body>
        <div id="Header">
            <h1>Monitoring System</h1>
        </div>
```

```
<div id="body">
  <h3>Login</h3>
  <form id="form1" method="post" action="">
    <div>
      <label for="usernm">Username: </label>
      <input type="text" name="usernm" id="usernm" required><br><br>
    </div>
    <div>
      <label for="psswrd">Password: </label>
      <input type="password" name="psswrd" id="psswrd" required><br><br>
    </div>
    <button id="sub" type="Submit">Submit</button>
  </form>
</div>
<div id="check">
  <?php
  if(isset($_POST['usernm']) and isset($_POST['psswrd'])){
    $uname=$_POST['usernm'];
    $passwrd=$_POST['psswrd'];

    if($query = $mysqli->prepare("SELECT * FROM Users WHERE pname='$uname'")){
      $query->execute();
      $query->bind_result($username,$password);
      $count=0;
      while($query->fetch()){
        $count=$count+1;
      }
      if ($count == 1){
        if(password_verify($passwrd,$password)){
          $query->close();
          $id='9';
          date_default_timezone_set("America/Chicago");
          $date=date("Y-m-d H:i:s");
          $stmt=$mysqli->prepare("INSERT INTO
TransactionalLogs(TimestampInfo,MessageID,DataInformation) VALUES(?,?,?)");
          $stmt->bind_param('sss',$date,$id,$uname);
          $stmt->execute();
          $stmt->close();
          echo "<script type='text/javascript'>alert('Login Credentials verified')</script>";
          echo "<script> window.location.assign('welcome.php');</script>";
        }
        else{
        //echo "Login Credentials verified";
        echo "<script type='text/javascript'>alert('Login Credentials do not match')</script>";

        }
      }
      else{
      echo "<script type='text/javascript'>alert('Invalid Login Credentials')</script>";
```

```
                //echo "Invalid Login Credentials";
                }
            }
            else{
                echo "Error in if 2";
            }
        }
        ?>
    </div>
  </body>
</html>
```

The code of DBDevice.php

```
<?php
require_once __DIR__ . '/../required/db_connect.php';
?>
<?php
date_default_timezone_set("America/Chicago");
echo "Last update: " . date('m/d/y') . " " . date('h:i:sa') . "<br>";
echo "<h3>Status of Sensors and Acutators</h3>";
if ($stmt=$mysqli->prepare("SELECT * FROM SensorsAndActuators LIMIT 100")) {
   $stmt->execute();
   $stmt->bind_result($devid,$devtype,$devfun,$ctrl,$status);
   echo "<table><tr><td>Device Name</td><td>Status</td></tr>";
   while ($stmt->fetch()) {
      echo "<tr><td>$devid</td><td>$status</td></tr>";
   }
   echo "</table>";
   $stmt->close();
}
else {
   echo "error";
   $mysqli->close();
}
echo "====================================" . "<br>";
echo "<h3>Alarm Status</h3>";
if ($stmt=$mysqli->prepare("SELECT a1.AlarmID, a1.SinceTS,
a1.AcknowledgementStatus,c1.MessageDescription FROM ActiveAlarms a1 LEFT JOIN Alarms a2
ON a1.AlarmID=a2.AlarmID LEFT JOIN CannedMessages c1 ON a2.MessageID=c1.MessageID")) {
   $stmt->execute();
   $stmt->bind_result($alarmid,$since,$acknowl,$description);
   echo "<table><tr><td>Alarm</td><td>Description</td><td>Since</td><td>Acknowledged</td></
tr>";
   while ($stmt->fetch()) {
      if ($acknowl!=1){
         echo "<tr><td>$alarmid</td><td>$description</td><td>$since</td><td>$acknowl</td></tr>";
      }
```

```
    }
    echo "</table>";
    $stmt->close();
}
else {
    echo "error";
    $mysqli->close();
}
echo "====================================" . "<br>";
echo "<h3>Charts</h3>";
if($stmt = $mysqli->prepare("Select MessageID, TimestampInfo FROM TransactionalLogs WHERE
MessageID=13 OR MessageID=14 ORDER BY TimestampInfo")){
    $stmt->execute();
    $stmt->bind_result($idNum,$time);
}
$datat[]=array();
$datan[]=array();
while($stmt->fetch()){
    $datat[]=$time;
    $datan[]=$idNum;
}
?>
<script src='https://cdn.jsdelivr.net/npm/chart.js@2.8.0'></script>
<script>var ctx = document.getElementById('chart1').getContext('2d');
    var chart=newChart(ctx,{
        type:'line',
        data:{
            labels:$datat,
            datasets:[{label:'LED Tracking',
                data:$datan
            }]
        },
        options:{}
    });
</script>";
```

Monitoring System

Login

Username:

Password:

Submit

File Edit View History Bookmarks Tools Help

RHYMEBACK | Tournam | Недавняя вики-деятель | Транслит по-русски | MySQL Databases | hPar | auth-db136.hostinger.co | Armando Herrera / Infol | • Team Project

team2project3342.online

Search

Monitoring System

Login

Username:

Password:

Submit

Invalid Login Credentials

OK

Transferring data from team2project3342.online...

ENG 11:48 26/06/2020

File Edit View History Bookmarks Tools Help

RHYMEBACK | Tournam | Недавняя вики-деятель | Транслит по-русски | MySQL Databases | hPar | auth-db136.hostinger.co | Armando Herrera / Infol | • Team Project

team2project3342.online

Search

Monitoring System

Login

Username:

Password:

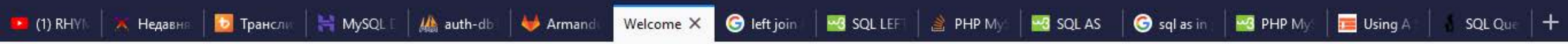
Submit

Login Credentials verified

OK

Transferring data from team2project3342.online...

ENG 11:50 26/06/2020



Welcome

Logout

File Edit View History Bookmarks Tools Help

(1) RHYM | Недавни | Трансли | MySQL E | auth-db | Armand | Welcome | left join | SQL LEFT | PHP My | SQL AS | sql as in | PHP My | Using A | SQL Que

team2project3342.online/welcome.php

Search

Welcome

Last update: 06/26/20 12:58:40pm

Status of Sensors and Acutators

Device Name	Status
LED1	0
LED2	0
LED3	0
LED4	0
SW1	0
SW2	0
SW3	0
SW4	0

==========

Alarm Status

Alarm	Description	Since	Acknowledged
3	BLOOD SUGAR HIGH	2020-03-11 14:25:21	0

==========

Charts

";

ENG 12:58 26/06/2020

File Edit View History Bookmarks Tools Help

Device Name	Status
LED1	0
LED2	0
LED3	0
LED4	0
SW1	0
SW2	0
SW3	0
SW4	0

==========

Alarm Status

Alarm	Description	Since	Acknowledged
3	BLOOD SUGAR HIGH	2020-03-11 14:25:21	0

==========

Charts

";

Logout

ENG 12:58 26/06/2020

File Edit View History Bookmarks Tools Help

RHYMEBACK | Tournam | Недавняя вики-деятель | Транслит по-русски | MySQL Databases | hPar | auth-db136.hostinger.co | Armando Herrera / Infol | Welcome

https://auth-db136.hostinger.com/sql.php?server=1&db=u5

Search

phpMyAdmin

Recent Favorites

- information_schema
- u551001383_team
  - New
  - ActiveAlarms
  - Alarms
  - CannedMessages
  - device
  - SensorsAndActuators
  - TransactionalLogs
  - Users
  - webuser

Server: 127.0.0.1:3306 » Database: u551001383_team » Table: TransactionalLogs

Browse | Structure | SQL | Search | Insert | Export | Import | Operations | Triggers

Showing rows 0 - 3 (4 total, Query took 0.0005 seconds.)

```
SELECT * FROM `TransactionalLogs`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

+ Options

	logID	TimestampInfo	MessageID	DataInformation
Edit Copy Delete		2020-06-26 11:50:11	9	Test
Edit Copy Delete	1	2020-03-11 12:23:16	7	
Edit Copy Delete	2	2020-03-12 10:8:37	9	ben
Edit Copy Delete	3	2020-03-12 10:8:37	11	ben

Check all *With selected:* Edit Copy Delete Export

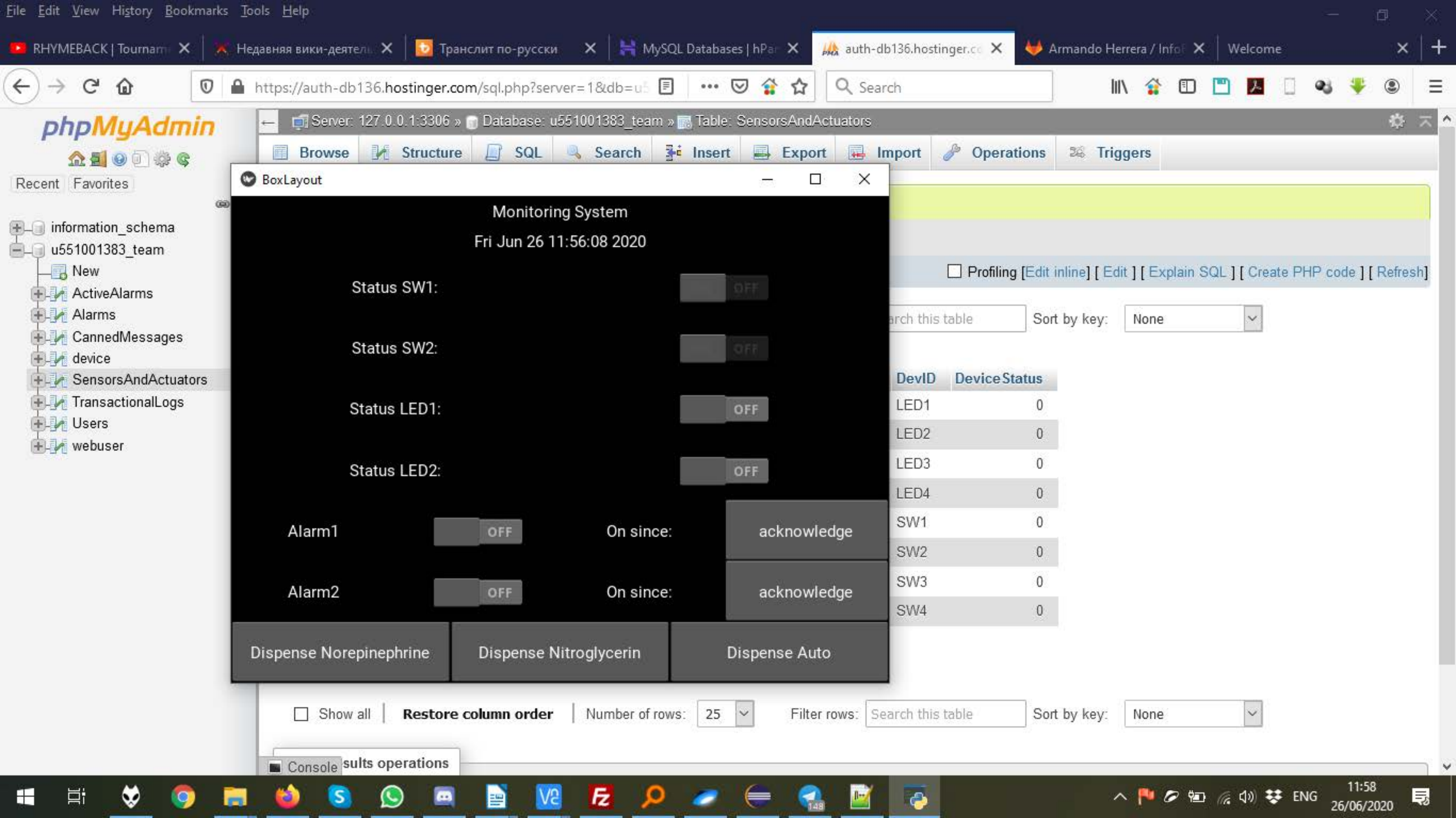
Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

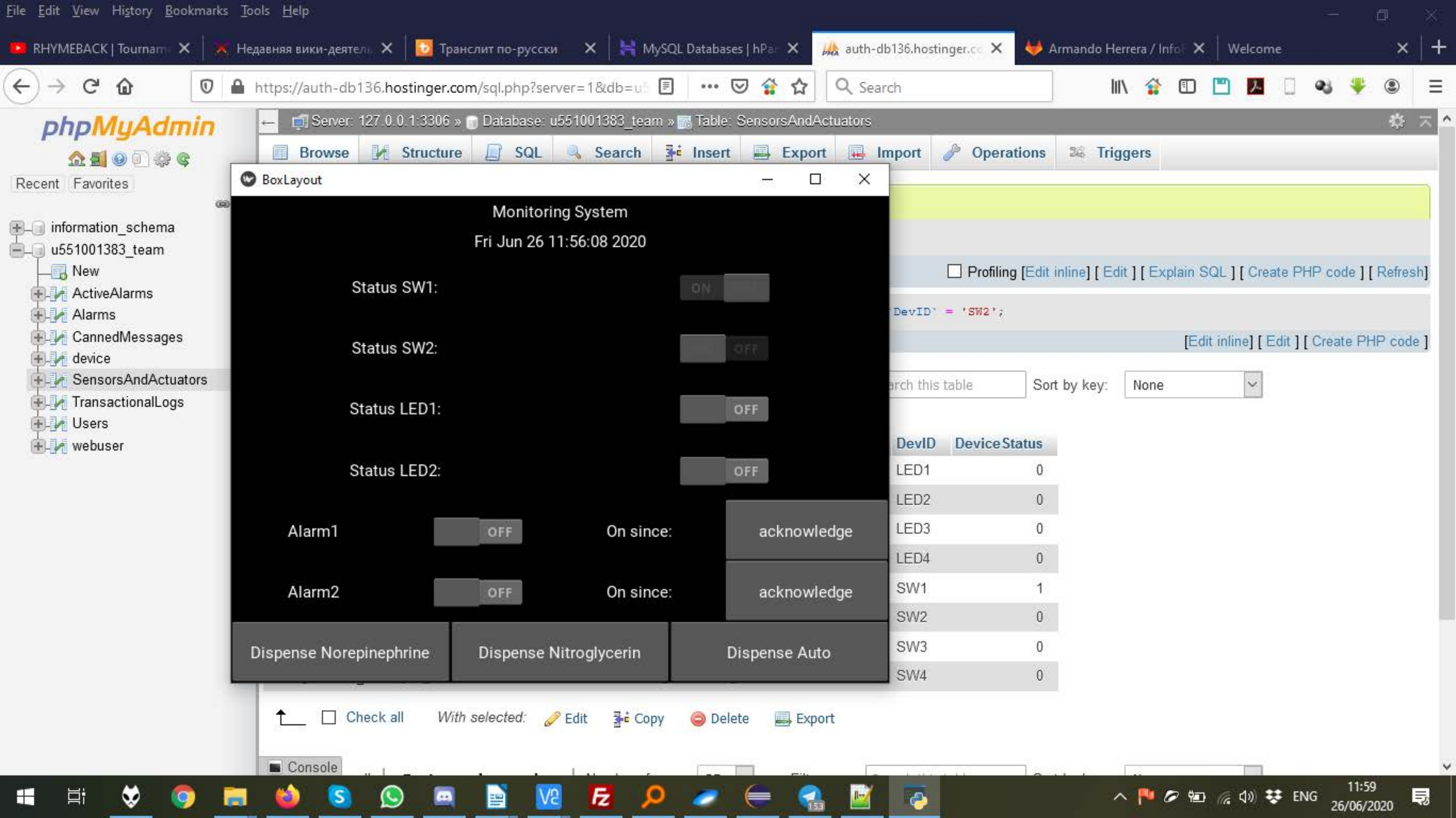
Query results operations

Print Copy to clipboard Export Display chart Create view

Console

ENG 12:07 26/06/2020


File Edit View History Bookmarks Tools Help
RHYMEBACK | Tournam
Недавняя вики-деятель
Транслит по-русски
MySQL Databases | hPar
auth-db136.hostinger.co
Armando Herrera / Infol
Welcome
https://auth-db136.hostinger.com/sql.php?server=1&db=u
Search
phpMyAdmin
Recent Favorites
information_schema
u551001383_team
New
ActiveAlarms
Alarms
CannedMessages
device
SensorsAndActuators
TransactionalLogs
Users
webuser
Server: 127.0.0.1:3306 » Database: u551001383_team » Table: SensorsAndActuators
Browse Structure SQL Search Insert Export Import Operations Triggers
Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
Sort by key: None
DevID Device Status
LED1 0
LED2 0
LED3 0
LED4 0
SW1 0
SW2 0
SW3 0
SW4 0
Show all | Restore column order | Number of rows: 25 Filter rows: Search this table Sort by key: None
Console
sults operations
BoxLayout
Monitoring System
Fri Jun 26 11:56:08 2020
Status SW1: OFF
Status SW2: OFF
Status LED1: OFF
Status LED2: OFF
Alarm1 OFF On since: acknowledge
Alarm2 OFF On since: acknowledge
Dispense Norepinephrine
Dispense Nitroglycerin
Dispense Auto
ENG
11:58
26/06/2020


BoxLayout
Monitoring System
Fri Jun 26 11:56:08 2020
Status SW1:
ON
Status SW2:
OFF
Status LED1:
OFF
Status LED2:
OFF
Alarm1
OFF
On since:
acknowledge
Alarm2
OFF
On since:
acknowledge
Dispense Norepinephrine
Dispense Nitroglycerin
Dispense Auto
phpMyAdmin
Server: 127.0.0.1:3306 » Database: u551001383_team » Table: SensorsAndActuators
Browse
Structure
SQL
Search
Insert
Export
Import
Operations
Triggers
information_schema
u551001383_team
New
ActiveAlarms
Alarms
CannedMessages
device
SensorsAndActuators
TransactionalLogs
Users
webuser
DevID
DeviceStatus
LED1 0
LED2 0
LED3 0
LED4 0
SW1 1
SW2 0
SW3 0
SW4 0
Check all
With selected:
Edit
Copy
Delete
Export
Console

File Edit View History Bookmarks Tools Help

RHYMEBACK | Tournam | Недавняя вики-деятель | Транслит по-русски | MySQL Databases | hPar | auth-db136.hostinger.co | Armando Herrera / InfoF | Welcome

https://auth-db136.hostinger.com/sql.php?server=1&db=u5

Search

phpMyAdmin

Recent Favorites

- information_schema
- u551001383_team
  - New
  - ActiveAlarms
  - Alarms
  - CannedMessages
  - device
  - SensorsAndActuators
  - TransactionalLogs
  - Users
  - webuser

Server: 127.0.0.1:3306 » Database: u551001383_team » Table: SensorsAndActuators

Browse | Structure | SQL | Search | Insert | Export | Import | Operations | Triggers

Showing rows 0 - 7 (8 total, Query took 0.0003 seconds.)

```
SELECT * FROM `SensorsAndActuators`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | **Restore column order** | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

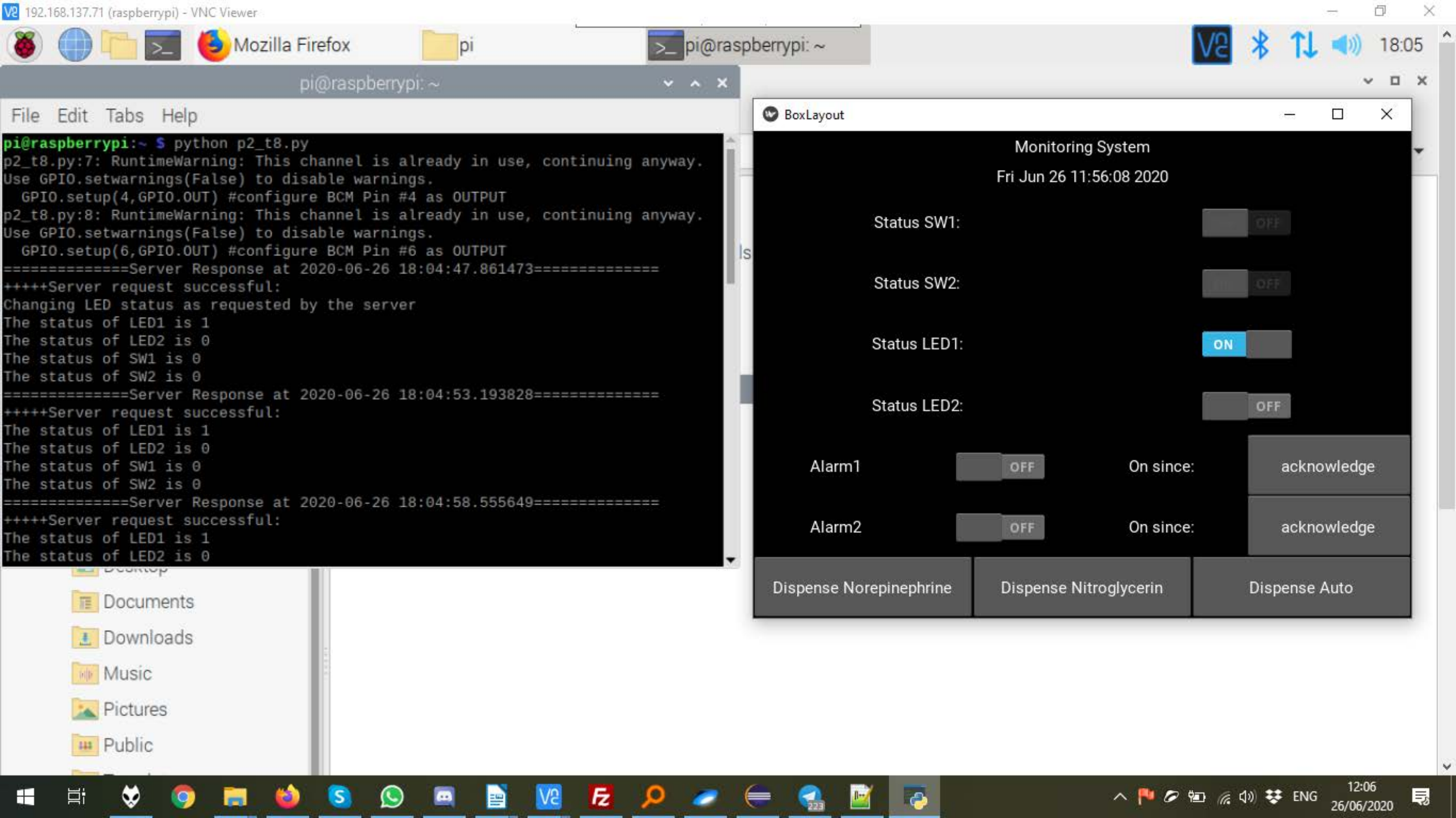
	DevType	FunctionDescription	Control	DevID	DeviceStatus
Edit Copy Delete	ACTUATOR	IV infusion of pressor to increase BP	ANDROID	LED1	1
Edit Copy Delete	ACTUATOR	IV infusion of Y to decrease BP	ANDROID	LED2	0
Edit Copy Delete	ACTUATOR	IV infusion of W to decrease blood sugar	ANDROID	LED3	0
Edit Copy Delete	ACTUATOR	IV infusion of Z to increase blood sugar	ANDROID	LED4	0
Edit Copy Delete	SENSOR	sense systolic BP > 140 mm Hg	RPI	SW1	0
Edit Copy Delete	SENSOR	sense diastolic BP < 30 mm Hg	RPI	SW2	0
Edit Copy Delete	SENSOR	sense blood sugar > 120 mg/dL	RPI	SW3	0
Edit Copy Delete	SENSOR	sense blood sugar < 50 mg/dL	RPI	SW4	0

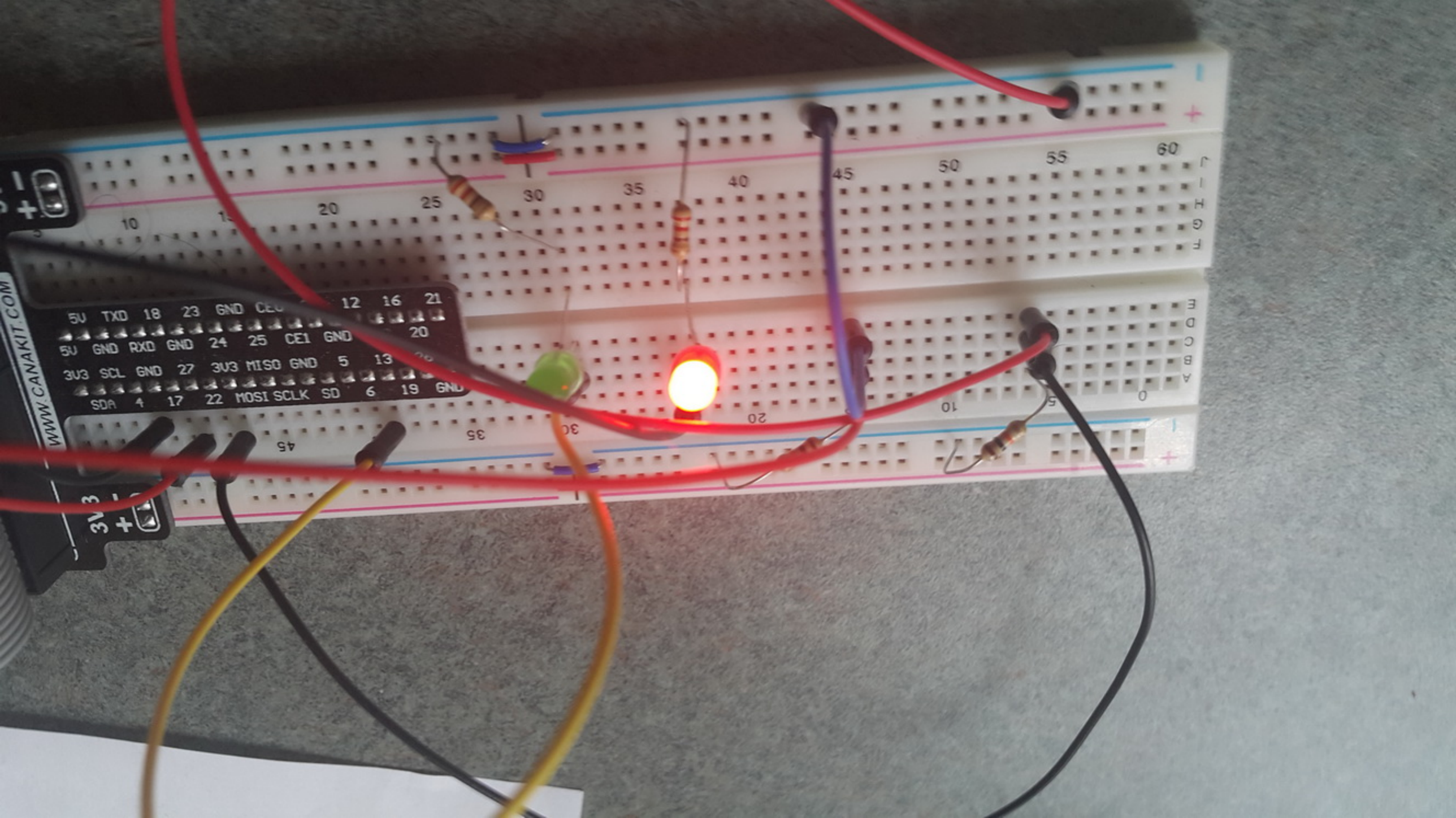
Check all *With selected:* Edit Copy Delete Export

Show all | **Restore column order** | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Console sults operations

ENG 12:05 26/06/2020


192.168.137.71 (raspberrypi) - VNC Viewer
Mozilla Firefox
pi
pi@raspberrypi: ~
18:05
pi@raspberrypi: ~
File Edit Tabs Help
pi@raspberrypi:~ $ python p2_t8.py
p2_t8.py:7: RuntimeWarning: This channel is already in use, continuing anyway.
Use GPIO.setwarnings(False) to disable warnings.
GPIO.setup(4,GPIO.OUT) #configure BCM Pin #4 as OUTPUT
p2_t8.py:8: RuntimeWarning: This channel is already in use, continuing anyway.
Use GPIO.setwarnings(False) to disable warnings.
GPIO.setup(6,GPIO.OUT) #configure BCM Pin #6 as OUTPUT
==========Server Response at 2020-06-26 18:04:47.861473==========
+++++Server request successful:
Changing LED status as requested by the server
The status of LED1 is 1
The status of LED2 is 0
The status of SW1 is 0
The status of SW2 is 0
==========Server Response at 2020-06-26 18:04:53.193828==========
+++++Server request successful:
The status of LED1 is 1
The status of LED2 is 0
The status of SW1 is 0
The status of SW2 is 0
==========Server Response at 2020-06-26 18:04:58.555649==========
+++++Server request successful:
The status of LED1 is 1
The status of LED2 is 0
Documents
Downloads
Music
Pictures
Public
BoxLayout
Monitoring System
Fri Jun 26 11:56:08 2020
Status SW1:
OFF
Status SW2:
OFF
Status LED1:
ON
Status LED2:
OFF
Alarm1
OFF
On since:
acknowledge
Alarm2
OFF
On since:
acknowledge
Dispense Norepinephrine
Dispense Nitroglycerin
Dispense Auto
ENG
12:06
26/06/2020


WWW.CANAKIT.COM
5V TXD 18 23 GND
5V GND RXD GND 24 25 CE1 GND
3V3 SCL GND 27 3V3 MISO GND 5 13
SDA 4 17 22 MOSI SCLK SD 6 19 GND
12 16 21
20
10 15 20 25 30 35 40 45 50 55 60
A B C D E F G H I J
0 5 10 20 30 35 45