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 my own efforts, and is not copied from anyone else or any other resource (such as Internet). *Furthermore, I certify that I have not let anyone copy from my work.*

The code of python file for Android app (Unlike the tutorial, the Hostinger domain was used, because 000webhost had troubles with the requests, closing the application in the next request):

```
import kivy
from kivy.app import App
from kivy.uix.switch import Switch
from kivy.uix.gridlayout import GridLayout
from kivy.uix.label import Label
from kivy.clock import Clock
from functools import partial
import time
import requests

class SwitchContainer(GridLayout): #create a class that uses the GridLayout module

    def __init__(self, **kwargs):
        super(SwitchContainer, self).__init__(**kwargs)
        self.cols = 2

        self.add_widget(Label(text="SW 1: ")) #create a label for SW1
        self.sw1 = Switch(active=False) #create a SwitchCompat for SW1 (default to OFF)
        self.add_widget(self.sw1) #add the created SwitchCompat to the screen
        self.sw1.disabled = True #make SW1 unclickable on the app


        self.add_widget(Label(text="LED 1: ")) #create a label for LED1
        self.led1 = Switch(active=False) #create a SwitchCompat for LED1 (default to OFF)
        self.add_widget(self.led1) #add the created SwitchCompat to the screen
        self.led1.disabled = False #by default a created SwitchCompat is clickable; so, there is no need
        #for this statement

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


    def JSONrequest(self, *largs):

        if (self.sw1.active == True): #Get the sw1 active status and convert it to an integer
            SW1 = 1
        else:
            SW1 = 0
        if (self.led1.active == True): #Get the led1 active status and convert it to an integer
            LED1 = 1
        else:
            LED1 = 0

        #below are json request payload, the request itself, and the response
        data = {'username': 'Test','password':'Tester3', 'SW1': SW1, 'LED1': LED1} #json request payload
        res = requests.post("https://team2project3342.online/scripts/sync_app_data.php", json=data)
```

```
        r = res.json() #json response
        if SW1 != r['SW1']: #check the received value of SW1 & change it on the App if there is a
mismatch
            if self.sw1.active == True:
                self.sw1.active = False
            else:
                self.sw1.active = True
        else:
            return


class SwitchExample(App):
    def build(self): #build
        return SwitchContainer()

if __name__ == '__main__':
    SwitchExample().run() #run
```

The code of p2_t8.py file (Unlike the tutorial, the Hostinger domain was used, because 000webhost had troubles with the requests, closing the application in the next request. As a result, the username and password are different.):

```
#!/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(17,GPIO.IN) #configure BCM Pin #17 as INPUT
i=0; n=15; 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)
        SW1=GPIO.input(17) #read the status of BCM Pin #17 (SW1)
        data = {'username': 'Test', 'password': 'Tester3', 'SW1': SW1, 'LED1': LED1}
        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)
                print "The status of LED1 is " + str(r['LED1'])
```

```
                print "The status of SW1 is " + str(r['SW1'])
        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 sync_app_data.php (Unlike the tutorial, instead of webuser, Users database was used):

```
<?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;
$LED1_status = 0;
if ($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"]))
      {
         $in_username = $json["username"];
         $in_password = $json["password"]; //if the expected fields are not null, get them
         $in_SW1 = $json["SW1"];
         $in_LED1 = $json["LED1"];
         if ($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 device set status=? where devname = 'LED1'"))
                  { //update LED1
                     $stmt->bind_param('i', $in_LED1);
                     $stmt->execute();
                  }
                  else
                  {
```

```
                    $error = 1;
                }
                $stmt->close();
                if (!$error && ($stmt = $mysqli->prepare("SELECT status FROM device where
devname = 'SW1'")))
                { //read SW1
                    $stmt->execute();
                    $stmt->bind_result($SW1_status);
                    $stmt->fetch();
                }
                else
                {
                    $error = 2;
                }
                $stmt->close();
                if (!$error && ($stmt = $mysqli->prepare("SELECT status FROM device where
devname = 'LED1'")))
                { //read LED1
                    $stmt->execute();
                    $stmt->bind_result($LED1_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['LED1'] = $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 code of sync_rpi_data.php (Unlike the tutorial, instead of webuser, Users database was used):

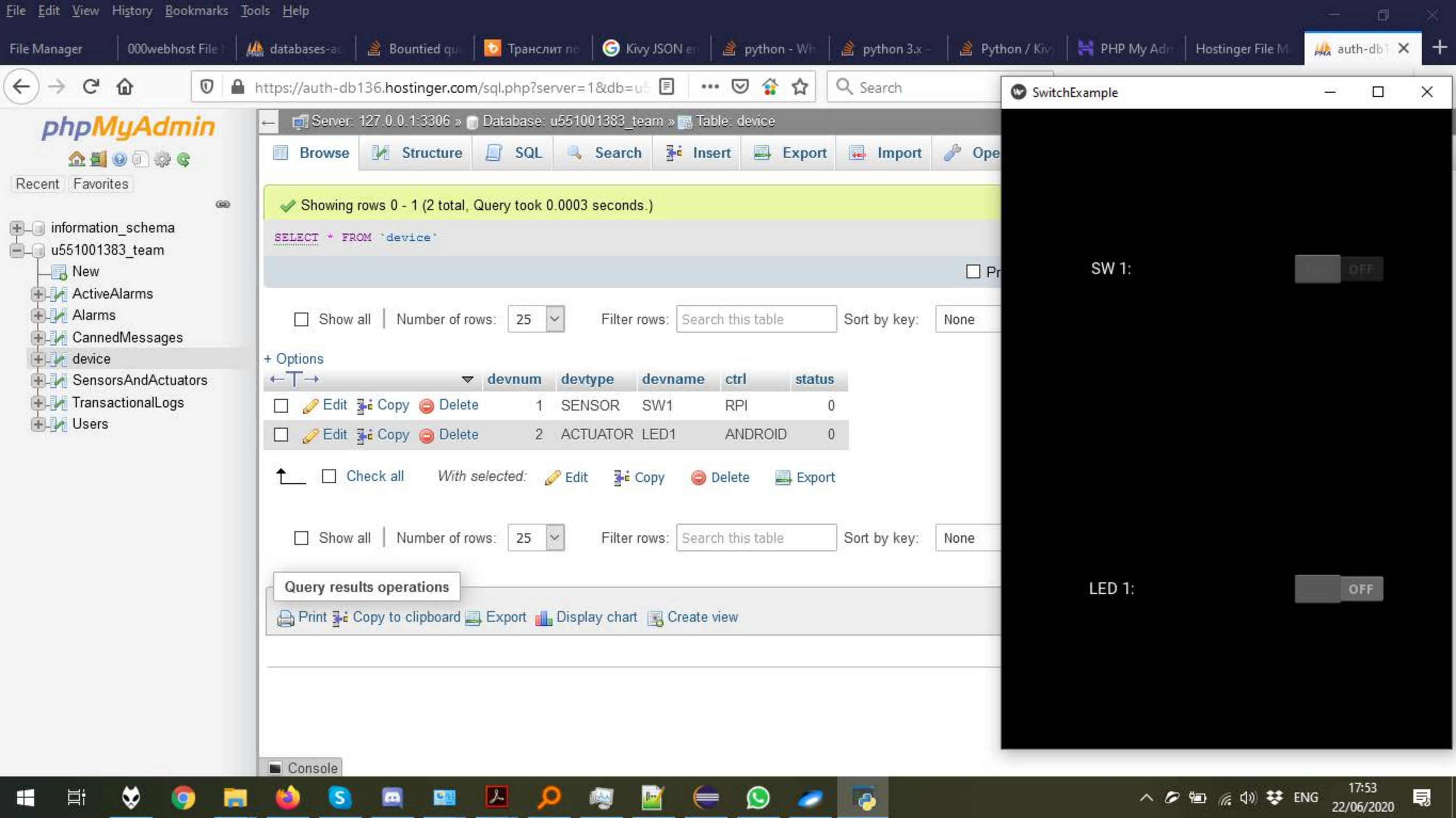
```
<?php
require_once __DIR__ . '/../../required/db_connect.php';
$input = file_get_contents("php://input");
$error = 0;
$out_json = array();
$out_json['success'] = 1; //assume success
$SW1_status = 0;
$LED1_status = 0;
if ($input)
{
    $json = json_decode($input, true); //check if it json input
    if (json_last_error() == JSON_ERROR_NONE)
    {
        if (isset($json["username"]) && isset($json["password"]) && isset($json["SW1"]) &&
isset($json["LED1"]))
        {
            $in_username = $json["username"];
            $in_password = $json["password"]; //if the expected fields are not null, get them
            $in_SW1 = $json["SW1"];
            $in_LED1 = $json["LED1"];
            if ($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.
```

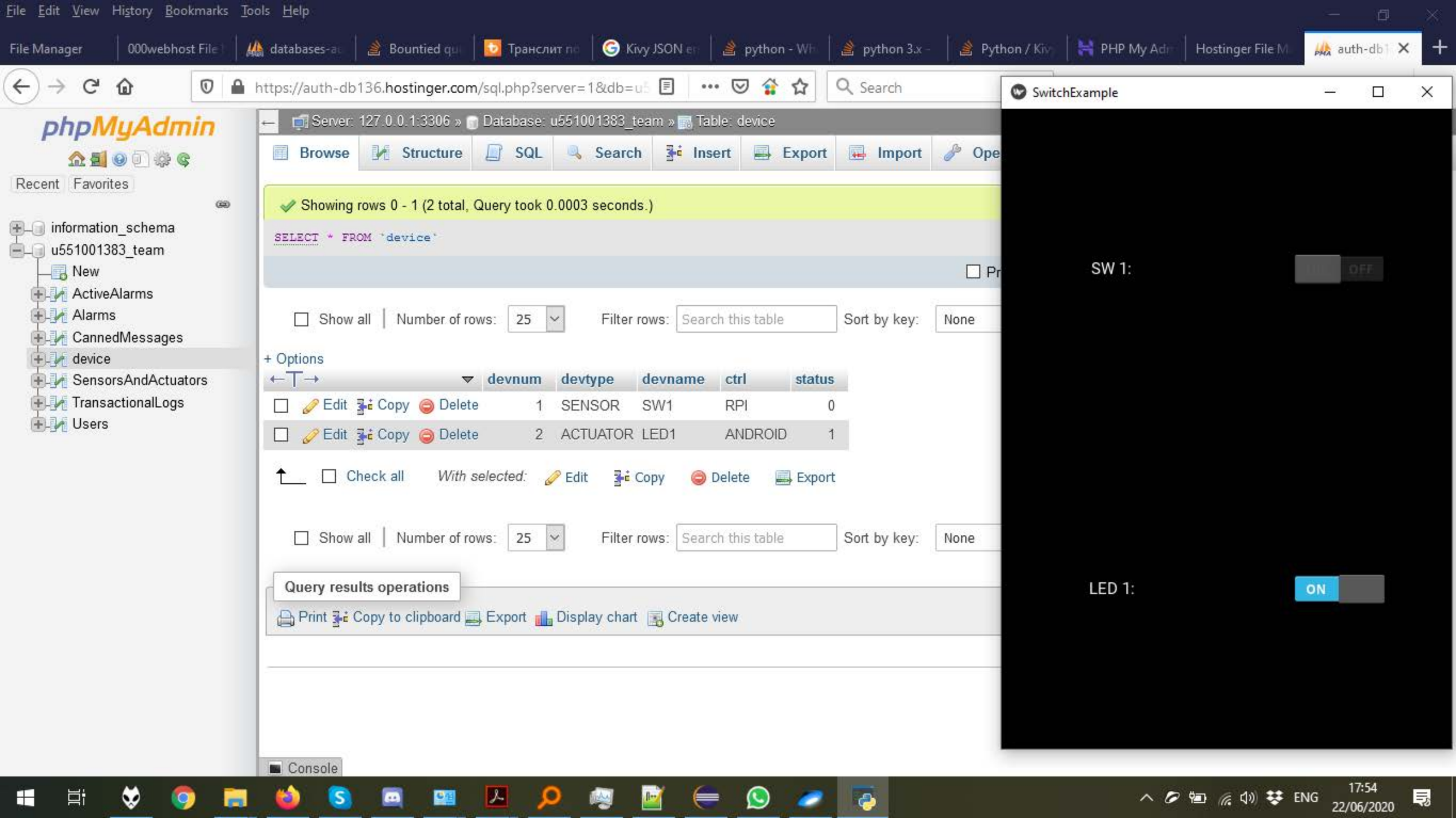
```
            $stmt->bind_result($db_password); //get the hashed password
            $stmt->fetch();
            if ($stmt->num_rows == 1)
            { //if user exists, verify the password
               if (password_verify($in_password, $db_password))
               {
                  $stmt->close();
                  if ($stmt = $mysqli->prepare("UPDATE device set status=?
where devname = 'SW1'"))
                  { //update SW1
                     $stmt->bind_param('i', $in_SW1);
                     $stmt->execute();
                  }
                  else
                  {
                     $error = 1;
                  }
                  $stmt->close();
                  if (!$error && ($stmt = $mysqli->prepare("SELECT status FROM device
where devname = 'SW1'")))
                  { //read SW1
                     $stmt->execute();
                     $stmt->bind_result($SW1_status);
                     $stmt->fetch();
                  }
                  else
                  {
                     $error = 2;
                  }
                  $stmt->close();
                  if (!$error && ($stmt = $mysqli->prepare("SELECT status FROM device
where devname = 'LED1'")))
                  { //read LED1
                     $stmt->execute();
                     $stmt->bind_result($LED1_status);
                     $stmt->fetch();
                  }
                  else
                  {
                     $error = 3;
                  }
                  $stmt->close();
               }
               else
               {
                  $error = 4;
               }
            }
            else
```

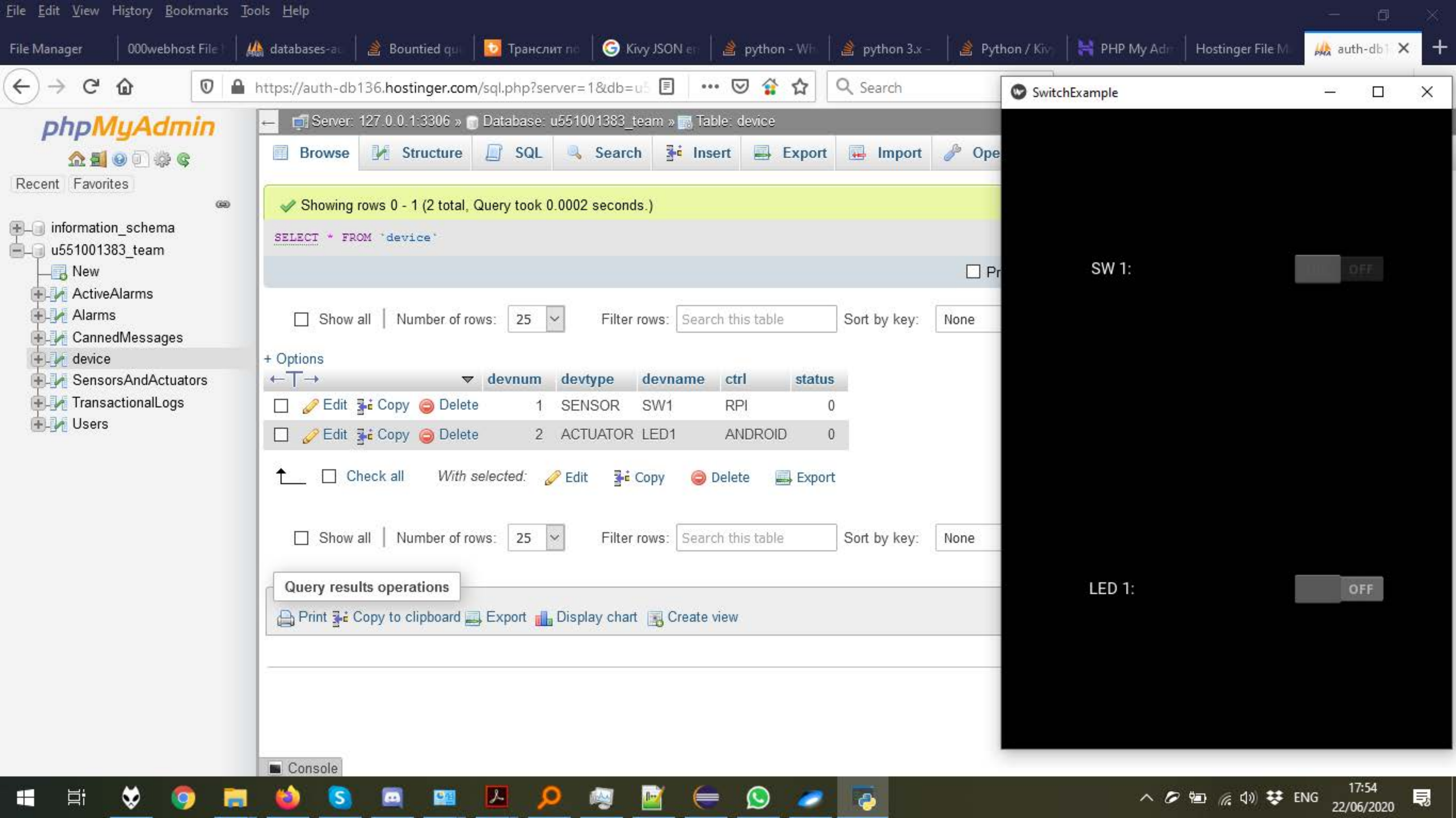
```
            {
               $error = 5;
            }
         }
         else
         {
            $error = 6;
         }
      }
      else
      {
         $error = 7;
      }
   }
   else
   {
      $error = 8;
   }
}
else
{
   $error = 9;
}
if ($error)
{
   $out_json['success'] = 0; //flag failure

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

?>
```


File Edit View History Bookmarks Tools Help
auth-db1
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
Server: 127.0.0.1:3306 » Database: u551001383_team » Table: device
Browse Structure SQL Search Insert Export Import
Showing rows 0 - 1 (2 total, Query took 0.0003 seconds.)
SELECT * FROM `device`
Show all Number of rows: 25 Filter rows: Search this table Sort by key: None
+ Options
devnum devtype devname ctrl status
Edit Copy Delete 1 SENSOR SW1 RPI 0
Edit Copy Delete 2 ACTUATOR LED1 ANDROID 0
Check all With selected: Edit Copy Delete Export
Query results operations
Print Copy to clipboard Export Display chart Create view
Console
SwitchExample
SW 1:
OFF
LED 1:
OFF
ENG
17:53
22/06/2020


File Edit View History Bookmarks Tools Help
auth-db1
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
Server: 127.0.0.1:3306 » Database: u551001383_team » Table: device
Browse Structure SQL Search Insert Export Import
Showing rows 0 - 1 (2 total, Query took 0.0003 seconds.)
SELECT * FROM `device`
Show all Number of rows: 25 Filter rows: Search this table Sort by key: None
+ Options
devnum devtype devname ctrl status
Edit Copy Delete 1 SENSOR SW1 RPI 0
Edit Copy Delete 2 ACTUATOR LED1 ANDROID 1
Check all With selected: Edit Copy Delete Export
Query results operations
Print Copy to clipboard Export Display chart Create view
Console
SwitchExample
SW 1:
OFF
LED 1:
ON
ENG
17:54
22/06/2020


File Edit View History Bookmarks Tools Help
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
Server: 127.0.0.1:3306 » Database: u551001383_team » Table: device
Browse Structure SQL Search Insert Export Import
Showing rows 0 - 1 (2 total, Query took 0.0002 seconds.)
SELECT * FROM `device`
Show all Number of rows: 25 Filter rows: Search this table Sort by key: None
+ Options
devnum devtype devname ctrl status
Edit Copy Delete 1 SENSOR SW1 RPI 0
Edit Copy Delete 2 ACTUATOR LED1 ANDROID 0
Check all With selected: Edit Copy Delete Export
Query results operations
Print Copy to clipboard Export Display chart Create view
Console
SwitchExample
SW 1:
OFF
LED 1:
OFF
ENG
17:54
22/06/2020

192.168.137.84 (raspberrypi) - VNC Viewer

pi@raspberrypi: ~

00:18

File Edit Tabs Help

```
pi@raspberrypi:~ $ python p2_t8.py
p2_t8.py:7: RuntimeWarning: This channel is already in use, cont
  GPIO.setup(4,GPIO.OUT) #configure BCM Pin #4 as OUTPUT
==============Server Response at 2020-06-23 00:18:03.462985=====
+++++Server request successful:
The status of LED1 is 0
The status of SW1 is 0
==============Server Response at 2020-06-23 00:18:08.814422=====
+++++Server request successful:
Changing LED status as requested by the server
The status of LED1 is 1
The status of SW1 is 0
==============Server Response at 2020-06-23 00:18:14.173016=====
+++++Server request successful:
The status of LED1 is 1
The status of SW1 is 0
==============Server Response at 2020-06-23 00:18:19.508408=====
+++++Server request successful:
The status of LED1 is 1
The status of SW1 is 0
==============Server Response at 2020-06-23 00:18:24.866639=====
+++++Server request successful:
The status of LED1 is 1
The status of SW1 is 0
==============Server Response at 2020-06-23 00:18:30.217454=====
+++++Server request successful:
The status of LED1 is 1
The status of SW1 is 0
```

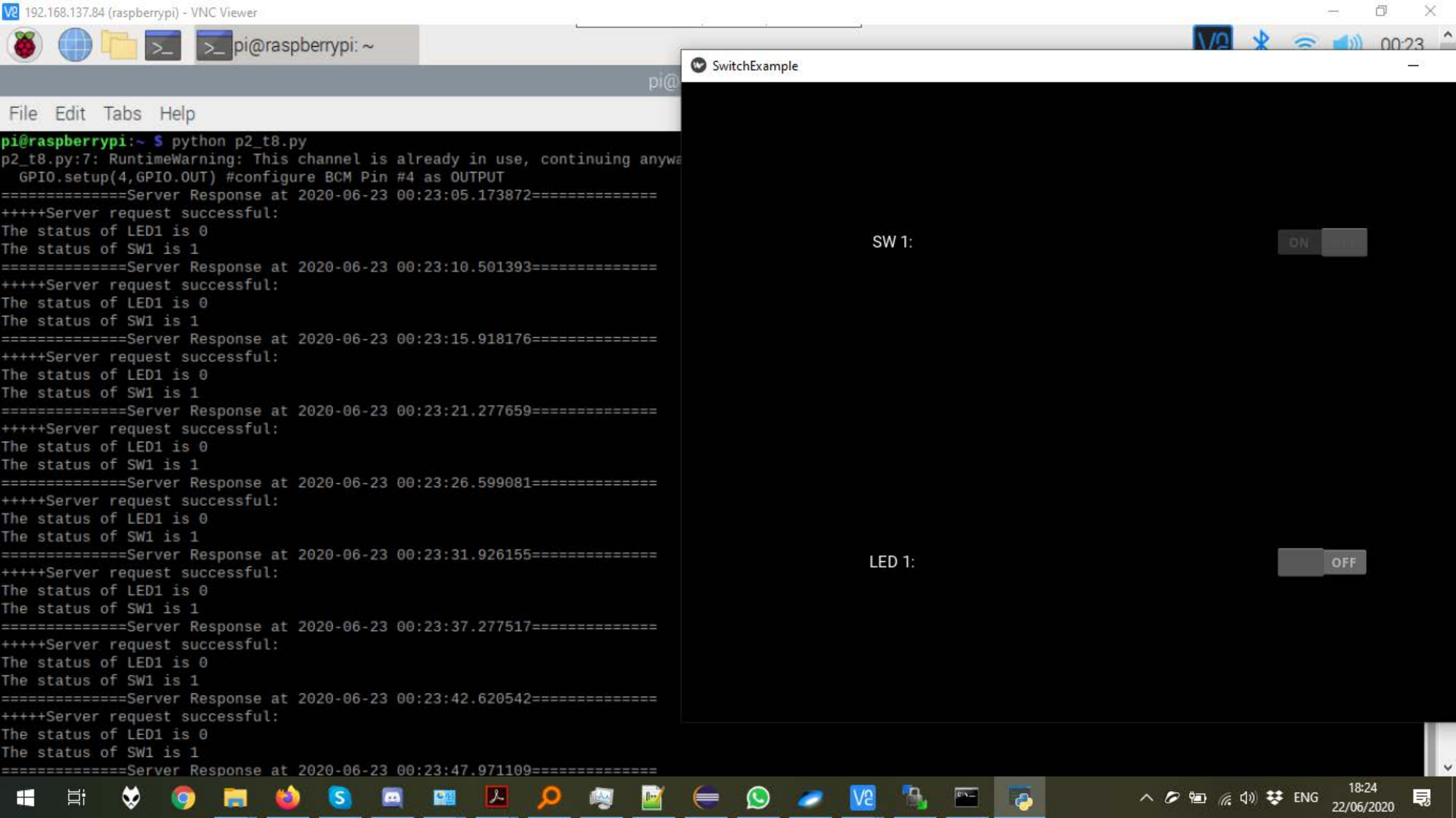
SwitchExample

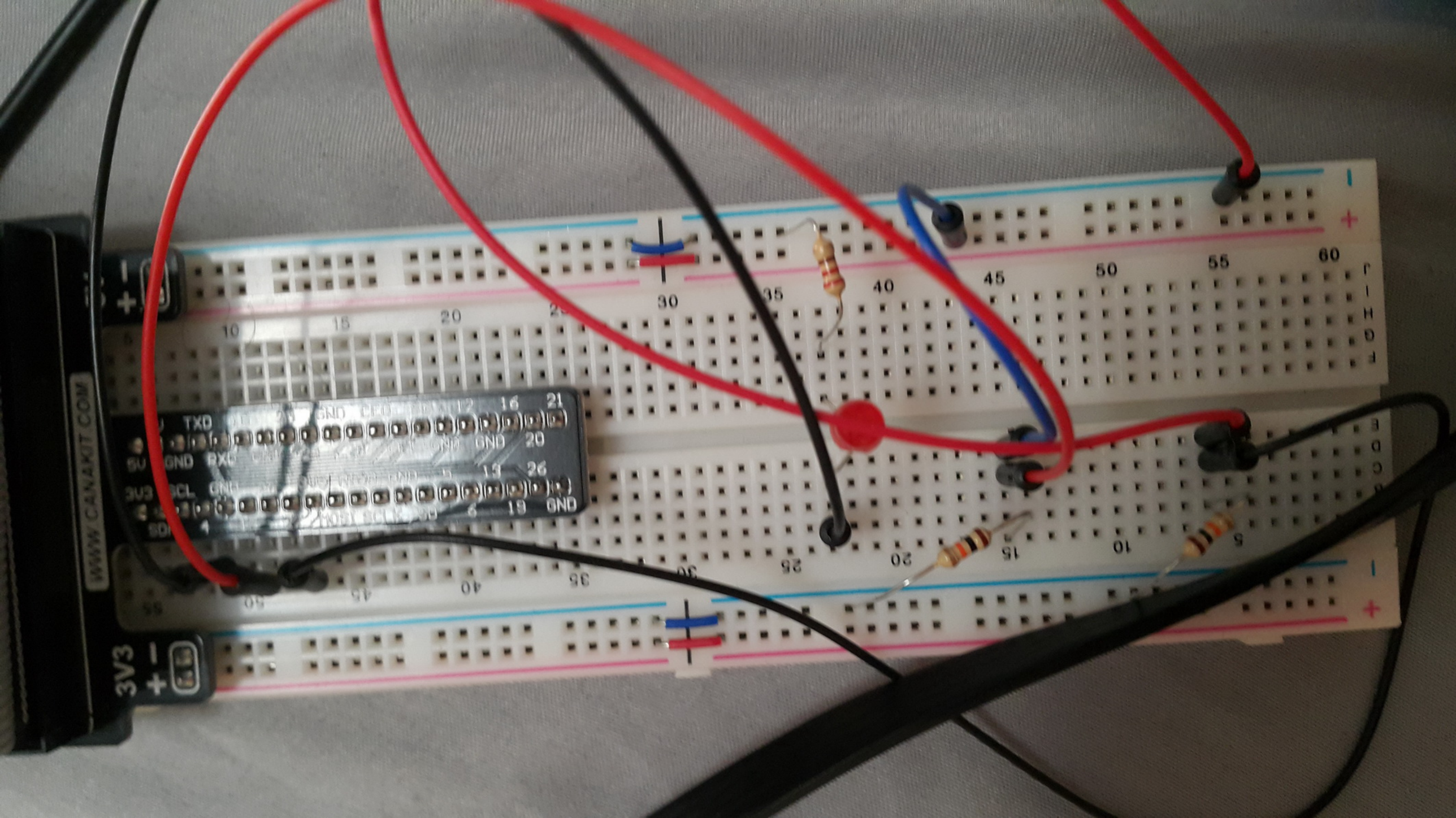
SW 1: OFF

LED 1: ON

ENG 18:18 22/06/2020

5V
3V3
WWW.CANAKIT.COM
GND CE0 SC 12 16 21
TXD GND 24 25 CE1 GND GND 20
3V3 MISO GND 5 13 26
MOSI SCLK SD 6 19 GND


192.168.137.84 (raspberrypi) - VNC Viewer
pi@raspberrypi: ~
00:23
File Edit Tabs Help
pi@raspberrypi:~ $ python p2_t8.py
p2_t8.py:7: RuntimeWarning: This channel is already in use, continuing anywa
GPIO.setup(4,GPIO.OUT) #configure BCM Pin #4 as OUTPUT
==========Server Response at 2020-06-23 00:23:05.173872==========
+++++Server request successful:
The status of LED1 is 0
The status of SW1 is 1
==========Server Response at 2020-06-23 00:23:10.501393==========
+++++Server request successful:
The status of LED1 is 0
The status of SW1 is 1
==========Server Response at 2020-06-23 00:23:15.918176==========
+++++Server request successful:
The status of LED1 is 0
The status of SW1 is 1
==========Server Response at 2020-06-23 00:23:21.277659==========
+++++Server request successful:
The status of LED1 is 0
The status of SW1 is 1
==========Server Response at 2020-06-23 00:23:26.599081==========
+++++Server request successful:
The status of LED1 is 0
The status of SW1 is 1
==========Server Response at 2020-06-23 00:23:31.926155==========
+++++Server request successful:
The status of LED1 is 0
The status of SW1 is 1
==========Server Response at 2020-06-23 00:23:37.277517==========
+++++Server request successful:
The status of LED1 is 0
The status of SW1 is 1
==========Server Response at 2020-06-23 00:23:42.620542==========
+++++Server request successful:
The status of LED1 is 0
The status of SW1 is 1
==========Server Response at 2020-06-23 00:23:47.971109==========
SwitchExample
SW 1:
ON
LED 1:
OFF
ENG
18:24
22/06/2020


WWW.CANAKIT.COM
3V3
5V
TXD
GND
RXD
3V3
SCL
GND
SDA
16
21
GND
20
13
26
19
GND
10
15
20
25
30
35
40
45
50
55
60