**FORMAN CHRISTIAN COLLEGE**

**(A CHARTERED UNIVERSITY)**



**Embedded Systems (CSCS 306)**

**FALL-2019**

**LAB-Exam**

**Store Data on Ubuntu Server through Node MCU**

**Group Members:**

* Mohammad Usman 20-10558
* Abeer Butt 20-10596
* Haris Naseer 20-10612

**Introduction:**

**Tasks:**

* **LabTask1:** In this LabTask, we had to create a database in mysql server running on Ubuntu Server. Then, we had to create a table which should have the following fields (id, status, dt, tm).
* **LabTask2:** In this LabTask, we had to write a php file on the Server that should insert status, date and time in the table. We had to test the connection between the table and the php file by sending a request.
* **LabTask3:** In this LabTask,we just had to clear all the entries in the table from within the Server.
* **LabTask4:** In this LabTask, we had to write code on the Node MCU that should make an LED ON/OFF using a push button. Once the LED is ON, it will turn OFF only when the user presses the button again and vice versa.
* **LabTask5:** In this LabTask, we had to extend the code written in the above task sp that the Node MCU connects with the Server. Now, on each button press, we had to store the LED status along with date and time in the database on the Server through the Node MCU by making a POST request.

**LAMP Stack:**

* Install Apache 2.4 from the Ubuntu repository and start
  + sudo apt install apache2
  + /etc/init.d/apache2 start
* Install the mysql-server package
  + sudo apt install mysql-server
* Install PHP, the PHP Extension and Application Repository, Apache support, and MySQL support
  + sudo apt install php7.2 libapache2-mod-php7.2 php-mysql
* Install the MySQL client by using the Ubuntu package manager
  + sudo apt-get install mysql-client
  + During the installation process you will be prompted to enter a password for the MySQL root user. Once the installation is complete, the MySQL server should be started automatically. You can run the following command from a terminal prompt to check whether the MySQL server is running
  + $sudo netstat -tap | grep mysql
  + When you run this command, you should see the following line or something similar
  + tcp 0 0 localhost.localdomain:mysql \*:\* LISTEN -
  + If the server is not running correctly, you can type the following command to start it:
  + $ sudo /etc/init.d/mysql restart

**Connectivity Issues:**

* FCC’s Wifi was not assigning an IP to our Ubuntu Server VM even on Bridge Network settings
  + Solution: Used a hotspot from Mobile and connected our laptop to it instead of FCC’s Wifi

**Working Code:**

**LabExam:**

#include <ESP8266WiFi.h>

#include <ESP8266Ping.h>

#include <ESP8266WiFiMulti.h>

#include <WiFiClient.h>

#include <WiFiClientSecure.h>

#include <WiFiServer.h>

#include <WiFiServerSecure.h>

#include <WiFiUdp.h>

#include <ESP8266WebServer.h>

#include <ESP8266HTTPClient.h>

const int led = D3; // led pin number

const int btn = D2; // button pin number

bool stateLED = LOW;

int cState = 0; // current state of the inc. button

int pState = 0; // previous state of the inc. button

int i = 0;

int j = 0;

const char \*ssid = "mob"; // ENTER YOUR WIFI SETTINGS

const char \*password = "pakistan04";

String host = "192.168.43.111"; // server address to read/write from

void setup()

{

pinMode(btn, INPUT);

pinMode(led, OUTPUT); // configuring sensor pin as input

Serial.begin(9600);

delay(100);

WiFi.begin(ssid, password); // connect to your WiFi router

Serial.print("Connecting");

// wait for connection

while (WiFi.status() != WL\_CONNECTED)

{

delay(500);

Serial.print(".");

}

// if connection successful show IP address in serial monitor

Serial.println("");

Serial.print("Connected to ");

Serial.println(ssid);

Serial.print("IP address: ");

Serial.println(WiFi.localIP());

Serial.println(WiFi.gatewayIP());

// check if node can ping the server

IPAddress ip (192, 168, 43, 111); // the remote ip to ping

bool ret = Ping.ping(ip);

Serial.print("Pinging: ");

Serial.println(ret);

}

void loop()

{

cState = digitalRead(btn); // read button value

if (cState != pState)

{

if (cState == HIGH && i == 0)

{

j = 1;

digitalWrite(led, HIGH);

updateState(1);

}

else if (cState == LOW && j == 1)

{

i = 1;

}

else if (cState == HIGH && i == 1)

{

j = 0;

digitalWrite(led, LOW);

updateState(0);

}

else if (cState == LOW && j == 0)

{

i = 0;

}

}

pState = cState;

}

// function that sends the status of LED to the server

void updateState(int stat)

{

HTTPClient http; // HTTPClient object

String postData; // post Data

postData = "status=" + String(stat);

http.begin("http://" + host + "/labexam/insert.php"); // request destination

http.addHeader("Content-Type", "application/x-www-form-urlencoded"); // content-type header

int httpCode = http.POST(postData); // send the request

String payload = http.getString(); // get the response payload

Serial.println(httpCode); // print HTTP return code

Serial.println(payload); // print request response payload

http.end(); // close connection

}

**insert.php:**

<?php

$servername = "localhost”;

$username = 'root';

$password = '123456';

$dbname = 'labexamdb';

// Create connection

$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection

if ($conn->connect\_error) {

die("Connection failed: " . $conn->connect\_error);

}

$status = $\_POST["status"];

$date = date("Y-m-d");

$time = date("h:i:s");

$sql = "INSERT INTO tb (status, dt, tm)

VALUES ('$status', '$date', '$time')";

if ($conn->query($sql) === TRUE)

{

echo "OK";

}

else

{

echo "Error: " . $sql . "<br>" . $conn->error;

}

$conn->close();

?>