Carlos Antonio Buendia López   A01379471

Luis Ángel Terrazas García   A01377440

Creación del código en arduino para conseguir lecturas de los datos de los sensores de temperatura y oximetría.

```
Recolector_archivosNodeMCU
#include <ESP8266WiFi.h>
#include <ESP8266HTTPClient.h>
#include <Arduino.h>
#include <Wire.h>
#include <SPI.h>
#include "algorithm_by_RF.h"
#include "max30102.h"
#include <DHT.h>

DHT dht(D4, DHT11);

const char *red = "IZZI-4EB6";
const char *password = "8C61A3BB4EB6";

String urlPulso = "http://189.228.161.43/Grabar/InsertarPulso.php?idPaciente=1&idsensorpulso_oxi=1&valor=";
String urlOximetria = "http://189.228.161.43/Grabar/InsertarOximetria.php?idPaciente=1&idsensorpulso_oxi=1&valor=";
String urlTemperatura = "http://189.228.161.43/Grabar/InsertarTemperatura.php?idPaciente=1&idsensorTemperatura=1&valor=";
WiFiClient clienteWiFi;
HTTPClient http;

#define dht_dpin 2
//#define DEBUG // Uncomment for debug output to the Serial stream
//#define USE_ADALOGGER // Comment out if you don't have ADALOGGER itself but your MCU still can handle this code
#define TEST_MAXIM_ALGORITHM // Uncomment if you want to include results returned by the original MAXIM algorithm
//#define SAVE_RAW_DATA // Uncomment if you want raw data coming out of the sensor saved to SD card. Red signal first, IR second.


#ifdef TEST_MAXIM_ALGORITHM
  #include "algorithm.h"
#endif

// Interrupt pin
const byte oxiInt = 10; // pin connected to MAX30102 INT

uint32_t elapsedTime,timeStart;

uint32_t aun_ir_buffer[BUFFER_SIZE]; //infrared LED sensor data
uint32_t aun_red_buffer[BUFFER_SIZE];  //red LED sensor data
float old_n_spo2;  // Previous SPO2 value
uint8_t uch_dummy,k;

void setup() {
  dht.begin();
  Serial.begin(115200);
  delay(1000);
  Serial.println("\n");
  pinMode(oxiInt, INPUT);  //pin D10 connects to the interrupt output pin of the MAX30102
  Serial.println("\nConectando a la red");
  WiFi.begin(red, password);
```

Creación de script para insertar datos en mysql.

```php
<?php

    try {
        $pdo = new PDO("mysql:host=localhost;dbname=estacion_6",'root', '');
        echo "Conexion exitosa a la base de datos";
    } catch(PDOExceptions $e){
        echo "<h1>Error conectando a la base de datos</h1>";
        print 'Error, ' . $e->getMessage();
        die();
    }
```

```php
<?php

    include 'conexiones.php';
    if($_GET){
        $idPaciente = $_GET["idPaciente"];
        $idsensorpulso_oxi = $_GET["idsensorpulso_oxi"];
        $valor = $_GET["valor"];

        $sql_agregar = "INSERT INTO oximetria (idPaciente,idsensorpulso_oxi,valor)
                VALUES (?,?,?)";
        $sentencia_agregar = $pdo->prepare($sql_agregar);
        $resultado = $sentencia_agregar->execute(array($idPaciente,$idsensorpulso_oxi,$valor));

        if($resultado==true) {
            $sentencia_agregar = null;
            $pdo = null;
            echo "\nSe insertarion los datos de una manera correcta";
        } else{
            echo"\nError al insertar en la BD";
        }
    }
    else{
        echo"Faltan los datos";
    }

?>
```

Diagrama base de datos Modelo E-R
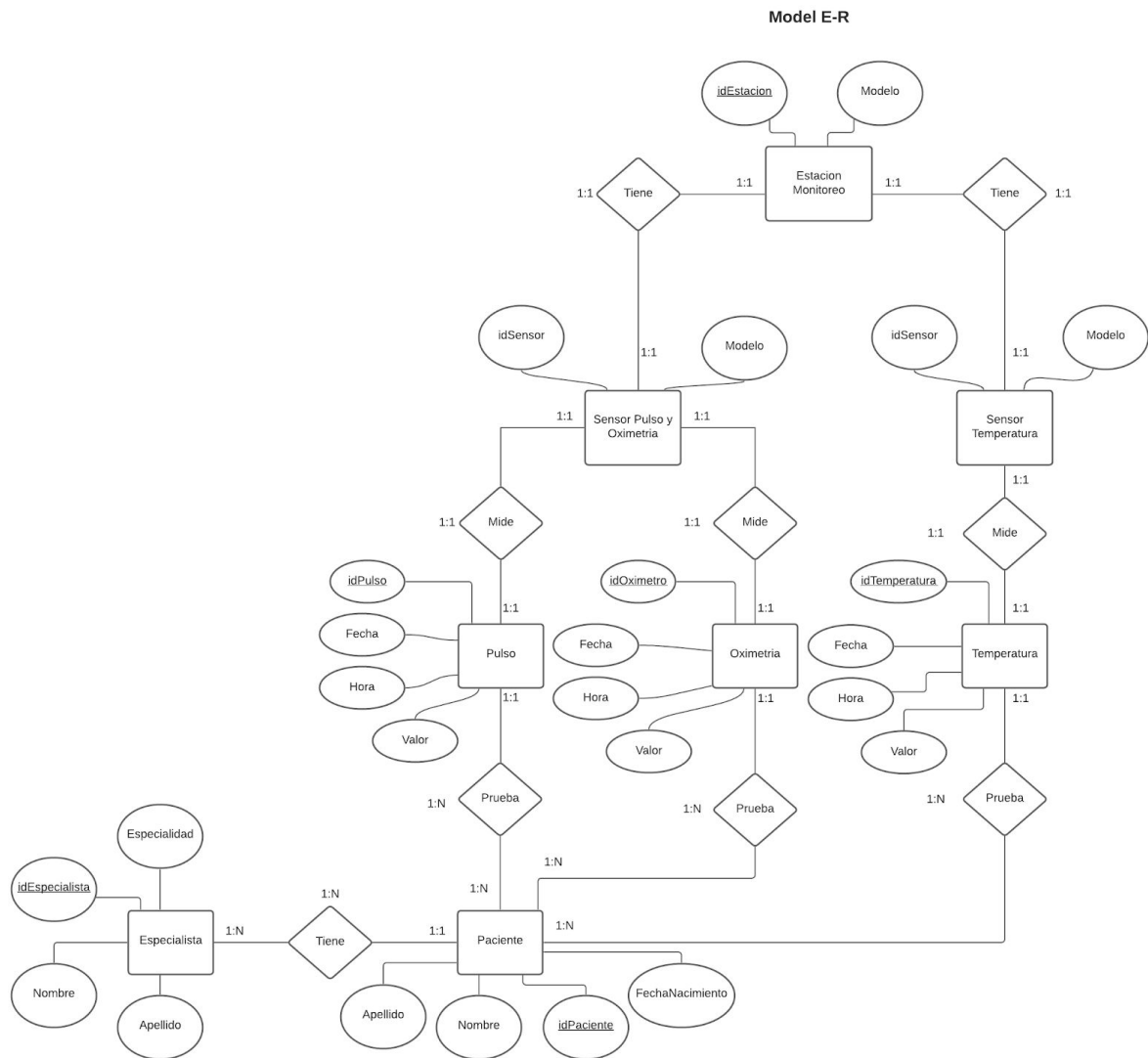
Script sql para creación de base de datos automática:
https://drive.google.com/file/d/1vbPgNNKdjBn4bUnM_D_78_k-yzASUxyt/view?usp=sharing

Revisión de datos por código en python que se conecta a la base de datos.

```python
import matplotlib.pyplot as plt

class Database:

    def __init__(self, base_datos, host):
        self.connection = pymysql.connect(
            host = _host,
            user = 'root',
            password = '',
            db = _base_datos
        )
        self.cursor = self.connection.cursor()
        print("Conexion establecida")
        self.matriz = []
        self.lista = []

    def conseguirDatos(self, id, idPaciente, tabla):
        try:
            sql = f'SELECT * FROM {tabla} WHERE id{tabla} = {id}'
            self.cursor.execute(sql)
            user = self.cursor.fetchone()
            for i in range(6):
                if idPaciente == user[1]:
                    self.lista.append(user[i])

                if idPaciente == user[1]:
                    self.matriz.append(self.lista)
        except Exception as e:
            raise
        self.lista = []

    def conseguir_tamano(self, tabla):
        self.cursor.execute(f"SELECT COUNT(*) FROM {tabla}")
        valor = self.cursor.fetchone()
        return valor[0]
```