Manual Técnico

**NurseBot**

**INTEGRANTES**

SORIANO ALVARO

CEBALLOS EDUARDO

SALAMANCA RAUL

BAJAÑA MOISÉS



**Programación de Sistemas Telemáticos**

**1. Objetivos**

Dar a conocer una descripción precisa del funcioinamiento del sistema.

**2. Requerimientos Técnicos**

|  |  |
| --- | --- |
| Hardware  (Mínimo) | **Velocidad CPU**   * 1.2GHz   **RAM (GB)**   * 1GB   **Cámara principal – Resolución**  **No requerida**  **Interna (GB)**   * 2 GB   **Externa**   * No requerida   **Localización**   * No requerida   **Conexión**   * No requerida |
|  |  |
| Software | **Versión Android**  4.X, preferiblemente con Android 5  **Aplicaciones instaladas**   * GooglePlay |

Diagrama de red

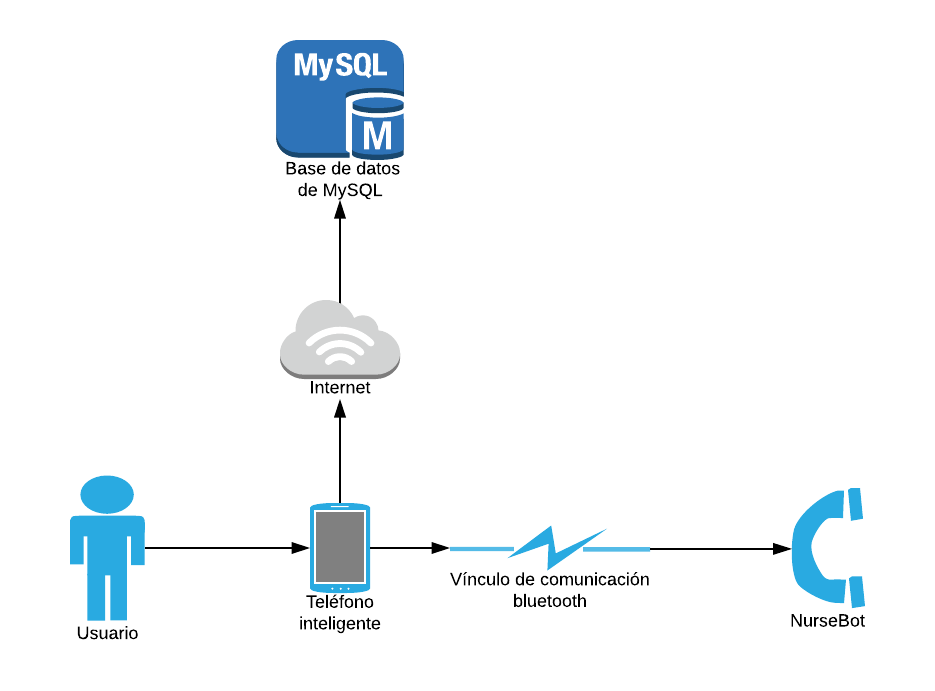
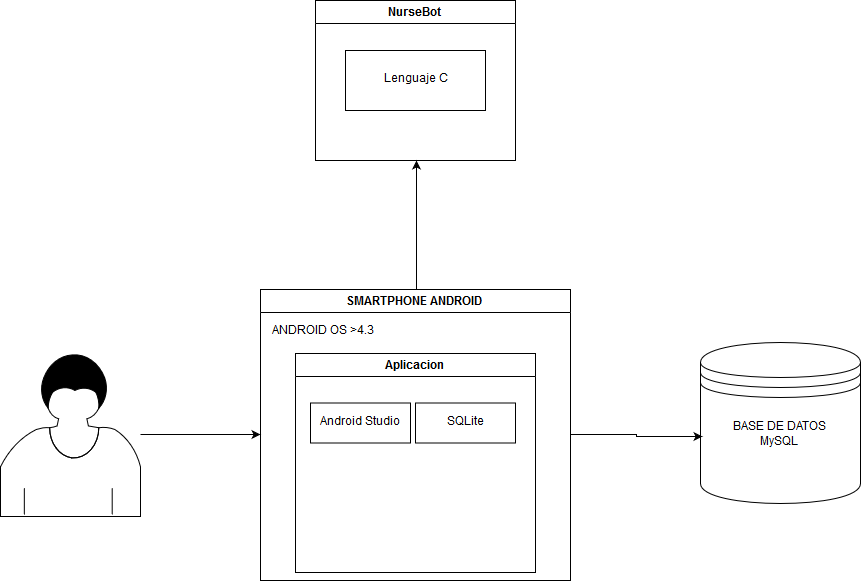
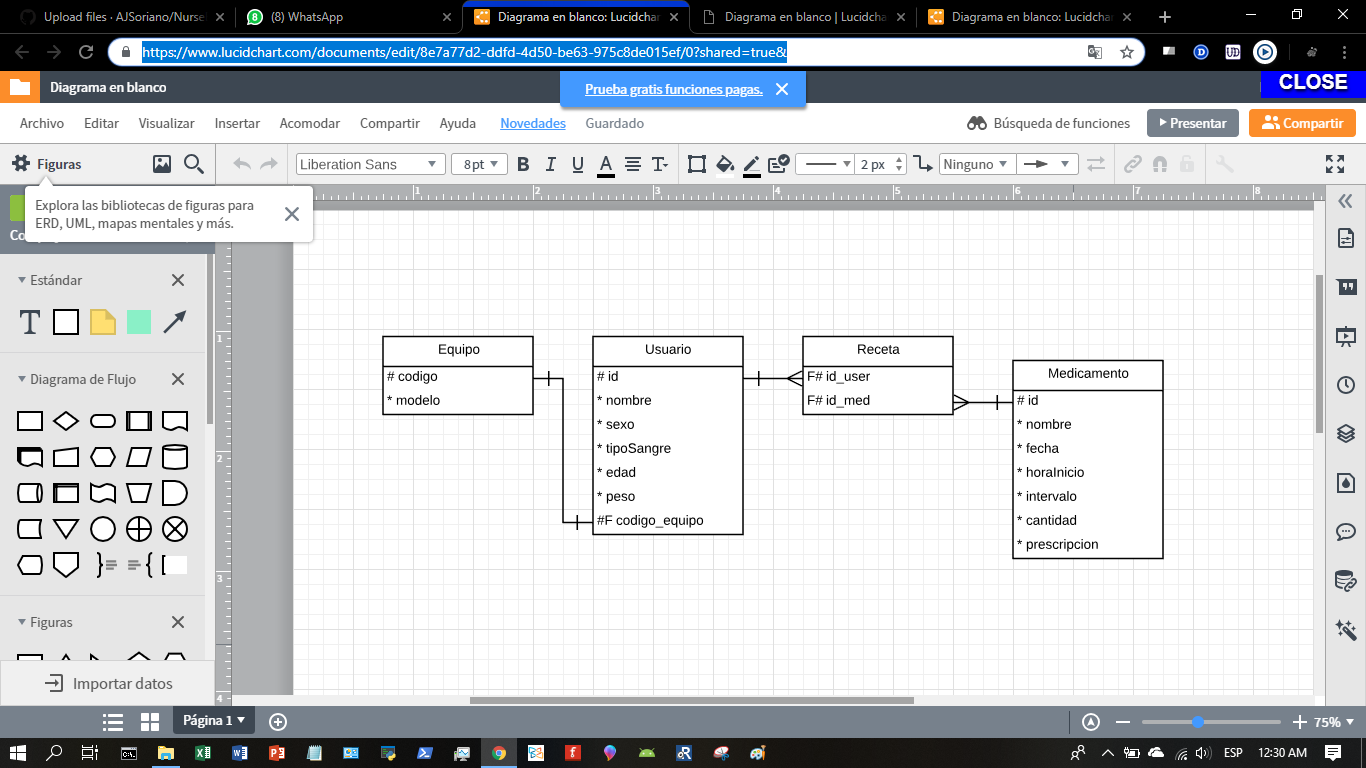


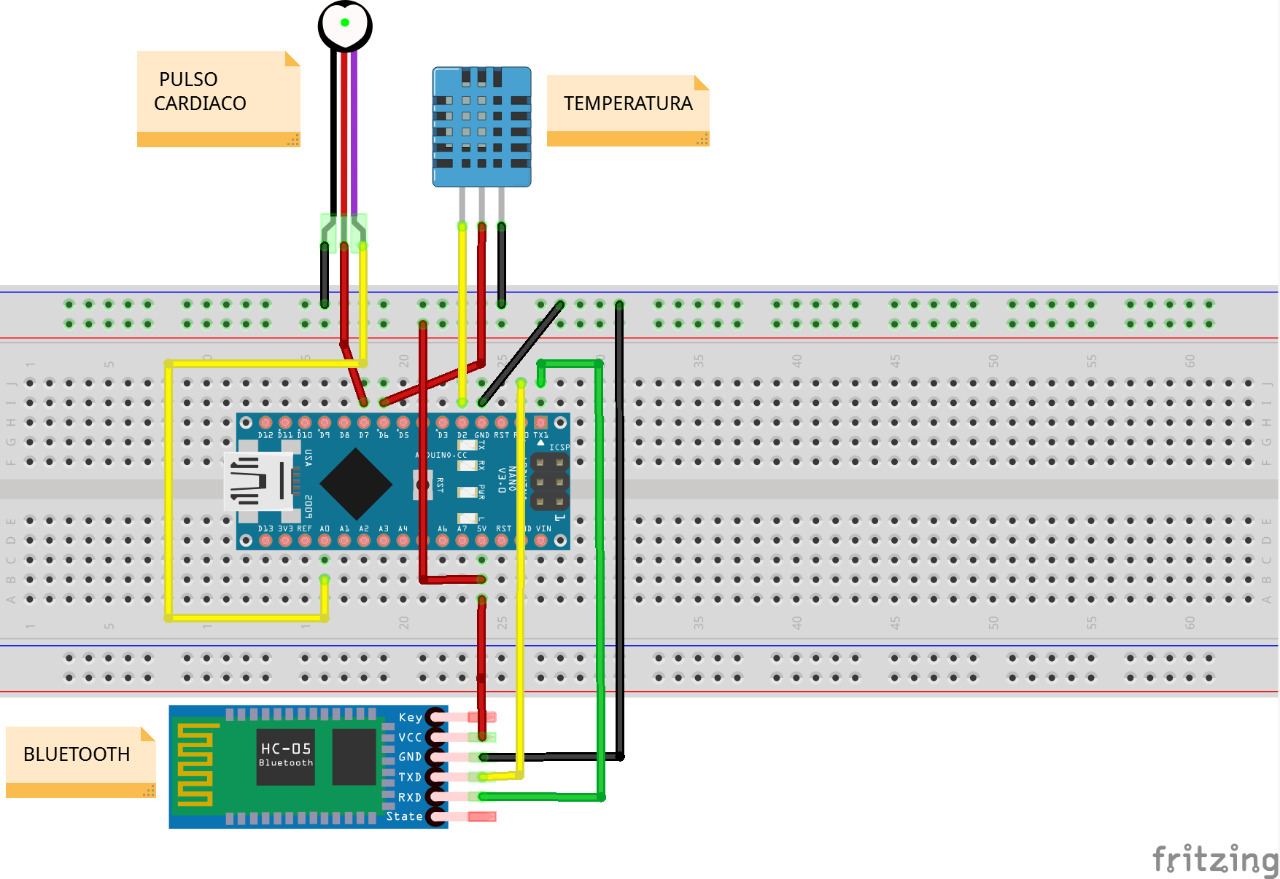
Diagrama de despliegue del sistema



Modelo entidad relación de la base de datos



Configuración de los dispositivos utilizados



Código fuente Arduino

#include <SoftwareSerial.h>

#define USE\_ARDUINO\_INTERRUPTS true

#include <PulseSensorPlayground.h>

#include "DHT.h"

#define DHTPIN 2

#define DHTTYPE DHT11

DHT dht(DHTPIN, DHTTYPE);

SoftwareSerial serial(0,1);

const int PulseWire = 0;

const int encenderPulso = 7;

const int encenderTemp = 6;

int Threshold = 550;

PulseSensorPlayground pulseSensor;

void setup() {

Serial.begin(9600);

pinMode(encenderPulso,OUTPUT);

pinMode(encenderTemp,OUTPUT);

dht.begin();

pulseSensor.analogInput(PulseWire);

pulseSensor.setThreshold(Threshold);

if (pulseSensor.begin()) {}

}

void loop() {

char c;

if(Serial.available()){

c = Serial.read();

if(c=='a'){

digitalWrite(encenderPulso,HIGH);

digitalWrite(encenderTemp,HIGH);

while(HIGH){

if(Serial.available()){

c = Serial.read();

}

if(c=='b'){

break;

}

float t = dht.readTemperature();

int myBPM = pulseSensor.getBeatsPerMinute();

pulseSensor.sawStartOfBeat();

Serial.print(myBPM);

Serial.println(" ♥");

if (isnan(t)){

Serial.print("0");

Serial.print(" C");

}

if (isnan(t)==LOW) {

Serial.print(t);

Serial.print(" C");

}

delay (1000);

}

}

digitalWrite(encenderPulso,LOW);

digitalWrite(encenderTemp,LOW);

}

Código fuente Android Studio

Activity LogIn

**package** com.example.nursebotpst;  
  
**import** android.content.Context;  
**import** android.database.sqlite.SQLiteDatabase;  
**import** android.database.sqlite.SQLiteOpenHelper;  
**import** android.view.View;  
  
**public class** AdminSQLiteOpenHelper **extends** SQLiteOpenHelper {  
 **public** AdminSQLiteOpenHelper(Context context, String name, SQLiteDatabase.CursorFactory factory, **int** version) {  
 **super**((Context) context, name, factory, version);  
 }  
  
  
 @Override  
 **public void** onCreate(SQLiteDatabase db) {  
 db.execSQL(**"create table medicamentos(codigo int primary key,medicamento text,cantidad int,dia int,mes int,ano int,hora int,intervalo int)"**);  
  
  
 }  
  
 @Override  
 **public void** onUpgrade(SQLiteDatabase db, **int** oldVersion, **int** newVersion) {  
  
 }  
}

Background Worker

**package** com.example.nursebotpst;  
  
**import** android.app.Activity;  
**import** android.app.AlertDialog;  
**import** android.content.Context;  
**import** android.content.Intent;  
**import** android.os.AsyncTask;  
**import** android.view.TextureView;  
**import** android.widget.TextView;  
**import** android.widget.Toast;  
  
**import** java.io.BufferedReader;  
**import** java.io.BufferedWriter;  
**import** java.io.IOException;  
**import** java.io.InputStream;  
**import** java.io.InputStreamReader;  
**import** java.io.OutputStream;  
**import** java.io.OutputStreamWriter;  
**import** java.net.HttpURLConnection;  
**import** java.net.MalformedURLException;  
**import** java.net.URL;  
**import** java.net.URLEncoder;  
**import** java.nio.Buffer;  
*/\*\*  
 \* Created by Estudiante on 24/01/2019.  
 \*/***public class** BackgroundWorker **extends** AsyncTask<String,Void,String>{  
 Context context;  
 AlertDialog alertDialog;  
 TextView textView;  
 BackgroundWorker (Context ctx){  
 context = ctx;  
 }  
 @Override  
 **protected** String doInBackground(String... voids) {  
 String type = voids[0];  
 String login\_url = **"http://192.168.1.17/NurseBot2.php"**;  
 **if**(type.equals(**"login"**)){  
 **try** {  
 String user = voids[1];  
 String pwd = voids[2];  
 URL url = **new** URL(login\_url);  
 HttpURLConnection httpURLConnection = (HttpURLConnection) url.openConnection();  
 httpURLConnection.setRequestMethod(**"POST"**);  
 httpURLConnection.setDoOutput(**true**);  
 httpURLConnection.setDoInput(**true**);  
 OutputStream outputStream = httpURLConnection.getOutputStream();  
 BufferedWriter bufferedWriter = **new** BufferedWriter(**new** OutputStreamWriter(outputStream,**"UTF-8"**));  
 String post\_data = URLEncoder.encode(**"codig"**,**"UTF-8"**)+**"="**+URLEncoder.encode(pwd,**"UTF-8"**)*/\*+"&" +  
 URLEncoder.encode("pwd","UTF-8")+"="+URLEncoder.encode(pwd,"UTF-8")\*/*;  
 bufferedWriter.write(post\_data);  
 bufferedWriter.flush();  
 bufferedWriter.close();  
 outputStream.close();  
  
 InputStream inputStream = httpURLConnection.getInputStream();  
 BufferedReader bufferedReader = **new** BufferedReader(**new** InputStreamReader(inputStream,**"iso-8859-1"**));  
  
 String result = **""**;  
 String line = **""**;  
  
 **while**((line = bufferedReader.readLine()) != **null**){  
 result += line;  
 }  
 bufferedReader.close();  
 inputStream.close();  
 httpURLConnection.disconnect();  
  
 **return** result;  
 } **catch** (MalformedURLException e) {  
 e.printStackTrace();  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
 }  
 **return null**;  
 }  
 @Override  
 **protected void** onPreExecute(){  
 *//textView = (TextView)((Activity)context).findViewById(R.id.lbl\_result);  
 //alertDialog = new AlertDialog.Builder(context).create();  
 //alertDialog.setTitle("Login Status");* }  
 @Override  
 **protected void** onPostExecute(String result){  
 *//alertDialog.setMessage("Usuario/Contraseña incorrecta");  
  
 //textView.setText(result);* **if**(result.contains(**"1"**)){  
 Toast.makeText((Activity)context, **"Ingreso exitoso"**, Toast.LENGTH\_SHORT).show();  
 Intent i = **new** Intent((Activity)context, MainActivity.**class**);  
 ((Activity)context).startActivity(i);  
 ((Activity)context).finish();  
 }  
 **else** Toast.makeText((Activity)context, **"Usuario/Contraseña incorrecta"**, Toast.LENGTH\_SHORT).show();  
 *//alertDialog.show();* }  
 @Override  
 **protected void** onProgressUpdate(Void... values){  
 **super**.onProgressUpdate(values);  
 }  
}

ACTIVITY CRONOGRAMA

**package** com.example.nursebotpst;  
  
**import** android.arch.lifecycle.CompositeGeneratedAdaptersObserver;  
**import** android.database.Cursor;  
**import** android.database.sqlite.SQLiteDatabase;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.os.Bundle;  
**import** android.view.View;  
**import** android.widget.EditText;  
**import** android.widget.TextView;  
**import** android.widget.Toast;  
  
**import** org.w3c.dom.Text;  
  
**public class** Cronograma **extends** AppCompatActivity {  
 **private** TextView medicamento, cantidad;  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_cronograma);  
 medicamento = (TextView) findViewById(R.id.IdMedCronograma);  
 cantidad = (TextView) findViewById(R.id.IdCantidadCronograma);  
  
 }  
  
 **public void** consultar(View v) {  
 AdminSQLiteOpenHelper admin = **new** AdminSQLiteOpenHelper(**this**,  
 **"administracion"**, **null**, 1);  
 SQLiteDatabase bd = admin.getReadableDatabase();  
  
 Cursor fila = bd.rawQuery(  
 **"select \* from medicamentos"**, **null**);  
 **if** (fila.moveToFirst()) {  
 medicamento.setText(fila.getString(1));  
 cantidad.setText(fila.getString(2));  
 }  
 Toast.makeText(**this**, **"Consulta realizada con exito"**, Toast.LENGTH\_SHORT).show();  
 bd.close();  
 }  
  
  
  
  
}

Main Activity **package** com.example.nursebotpst;  
  
**import** android.content.Intent;  
**import** android.database.Cursor;  
**import** android.database.sqlite.SQLiteDatabase;  
**import** android.os.PersistableBundle;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.os.Bundle;  
**import** android.view.View;  
**import** android.widget.Button;  
**import** android.widget.TextView;  
**import** android.widget.Toast;  
  
**import** org.w3c.dom.Text;  
  
  
**public class** MainActivity **extends** AppCompatActivity {  
 **private** TextView med, cant;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
 }

Activity Med **package** com.example.nursebotpst;  
  
**import** android.bluetooth.BluetoothAdapter;  
**import** android.bluetooth.BluetoothDevice;  
**import** android.bluetooth.BluetoothSocket;  
**import** android.content.Intent;  
**import** android.os.Handler;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.os.Bundle;  
**import** android.view.View;  
**import** android.widget.Button;  
**import** android.widget.EditText;  
**import** android.widget.TextView;  
**import** android.widget.Toast;  
  
**import** java.io.IOException;  
**import** java.io.InputStream;  
**import** java.io.OutputStream;  
**import** java.util.Set;  
**import** java.util.UUID;  
  
**public class** MedActivity **extends** AppCompatActivity {  
 **private final** String DEVICE\_ADDRESS=**"98:D3:37:91:0E:3E"**;  
 **private final** UUID PORT\_UUID = UUID.fromString(**"00001101-0000-1000-8000-00805f9b34fb"**);*//Serial Port Service ID* **private** BluetoothDevice device;  
 **private** BluetoothSocket socket;  
 **private** OutputStream outputStream;  
 **private** InputStream inputStream;  
 Button startButton, sendButton,clearButton,stopButton;  
 TextView textView;  
 TextView editText;  
 **boolean** deviceConnected=**false**;  
 Thread thread;  
 **byte** buffer[];  
 **int** bufferPosition;  
 **boolean** stopThread;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_med);  
 startButton = (Button) findViewById(R.id.start\_button);  
 stopButton = (Button) findViewById(R.id.stop\_button);  
 sendButton = (Button) findViewById(R.id.send\_button);  
 *//clearButton = (Button) findViewById(R.id.clear\_botton);  
 //editText = (EditText) findViewById(R.id.entrada);* textView = (TextView) findViewById(R.id.med\_pulso);  
 setUiEnabled(**false**);  
  
 }  
 **public void** setUiEnabled(**boolean** bool)  
 {  
 startButton.setEnabled(!bool);  
 stopButton.setEnabled(bool);  
 sendButton.setEnabled(bool);  
 textView.setEnabled(bool);  
  
 }  
 **public boolean** BTinit()  
 {  
 **boolean** found=**false**;  
 BluetoothAdapter bluetoothAdapter=BluetoothAdapter.getDefaultAdapter();  
 **if** (bluetoothAdapter == **null**) {  
 Toast.makeText(getApplicationContext(),**"Device doesnt Support Bluetooth"**,Toast.LENGTH\_SHORT).show();  
 }  
 **if**(!bluetoothAdapter.isEnabled())  
 {  
 Intent enableAdapter = **new** Intent(BluetoothAdapter.ACTION\_REQUEST\_ENABLE);  
 startActivityForResult(enableAdapter, 0);  
 **try** {  
 Thread.sleep(1000);  
 } **catch** (InterruptedException e) {  
 e.printStackTrace();  
 }  
 }  
 Set<BluetoothDevice> bondedDevices = bluetoothAdapter.getBondedDevices();  
 **if**(bondedDevices.isEmpty())  
 {  
 Toast.makeText(getApplicationContext(),**"Please Pair the Device first"**,Toast.LENGTH\_SHORT).show();  
 }  
 **else** {  
 **for** (BluetoothDevice iterator : bondedDevices)  
 {  
 **if**(iterator.getAddress().equals(DEVICE\_ADDRESS))  
 {  
 device=iterator;  
 found=**true**;  
 **break**;  
 }  
 }  
 }  
 **return** found;  
 }  
  
 **public boolean** BTconnect()  
 {  
 **boolean** connected=**true**;  
 **try** {  
 socket = device.createRfcommSocketToServiceRecord(PORT\_UUID);  
 socket.connect();  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 connected=**false**;  
 }  
 **if**(connected)  
 {  
 **try** {  
 outputStream=socket.getOutputStream();  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
 **try** {  
 inputStream=socket.getInputStream();  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
  
 }  
  
  
 **return** connected;  
 }  
  
  
  
  
 **public void** onClickStart(View view) {  
  
 **if**(BTinit()) {  
 setUiEnabled(**true**);  
 **if**(BTconnect())  
 {  
  
 deviceConnected=**true**;  
 beginListenForData();  
 textView.append(**"\nConnection Opened!\n"**);  
 }  
  
 }  
 }  
  
 **void** beginListenForData()  
 {  
 **final** Handler handler = **new** Handler();  
 stopThread = **false**;  
 buffer = **new byte**[1024];  
 Thread thread = **new** Thread(**new** Runnable()  
 {  
 **public void** run()  
 {  
 **while**(!Thread.currentThread().isInterrupted() && !stopThread)  
 {  
 **try** {  
 **int** byteCount = inputStream.available();  
 **if**(byteCount > 0)  
 {  
 **byte**[] rawBytes = **new byte**[byteCount];  
 inputStream.read(rawBytes);  
 **final** String string=**new** String(rawBytes,**"UTF-8"**);  
 handler.postDelayed(**new** Runnable() {  
 **public void** run()  
 {  
 textView.append(string);  
 }  
 },500);  
 handler.postDelayed(**new** Runnable() {  
 @Override  
 **public void** run() {  
 textView.setText(**""**);  
 }  
 },100);  
  
 }  
 }  
 **catch** (IOException ex)  
 {  
 stopThread = **true**;  
 }  
 }  
 }  
 });  
  
 thread.start();  
 }  
  
 **public void** onClickSend(View view) {  
 String string = **"a"**;  
 string.concat(**"\n"**);  
 **try** {  
 outputStream.write(string.getBytes());  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
 Toast.makeText(**this**, **"Medicion iniciada"**,  
 Toast.LENGTH\_LONG).show();  
  
  
  
 }  
  
 **public void** onClickStop(View view) **throws** IOException {  
 stopThread = **true**;  
 outputStream.close();  
 inputStream.close();  
 socket.close();  
 setUiEnabled(**false**);  
 textView.setText(**""**);  
 deviceConnected=**false**;  
 Toast.makeText(**this**, **"Conexion terminada"**,  
 Toast.LENGTH\_SHORT).show();  
  
 *//textView.append("\nConnection Closed!\n");* }  
  
 **public void** onClickOff(View view) {  
 String string1 = **"b"**;  
 string1.concat(**"\n"**);  
 **try** {  
 outputStream.write(string1.getBytes());  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
 textView.setText(**""**);  
 Toast.makeText(**this**, **"Medicion terminada"**,  
 Toast.LENGTH\_SHORT).show();  
  
  
  
  
 }  
  
  
  
}

Activity Medicamento

package com.example.nursebotpst; **package** com.example.nursebotpst;  
  
**import** android.bluetooth.BluetoothAdapter;  
**import** android.bluetooth.BluetoothDevice;  
**import** android.bluetooth.BluetoothSocket;  
**import** android.content.Intent;  
**import** android.os.Handler;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.os.Bundle;  
**import** android.view.View;  
**import** android.widget.Button;  
**import** android.widget.EditText;  
**import** android.widget.TextView;  
**import** android.widget.Toast;  
  
**import** java.io.IOException;  
**import** java.io.InputStream;  
**import** java.io.OutputStream;  
**import** java.util.Set;  
**import** java.util.UUID;  
  
**public class** MedActivity **extends** AppCompatActivity {  
 **private final** String DEVICE\_ADDRESS=**"98:D3:37:91:0E:3E"**;  
 **private final** UUID PORT\_UUID = UUID.fromString(**"00001101-0000-1000-8000-00805f9b34fb"**);*//Serial Port Service ID* **private** BluetoothDevice device;  
 **private** BluetoothSocket socket;  
 **private** OutputStream outputStream;  
 **private** InputStream inputStream;  
 Button startButton, sendButton,clearButton,stopButton;  
 TextView textView;  
 TextView editText;  
 **boolean** deviceConnected=**false**;  
 Thread thread;  
 **byte** buffer[];  
 **int** bufferPosition;  
 **boolean** stopThread;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_med);  
 startButton = (Button) findViewById(R.id.start\_button);  
 stopButton = (Button) findViewById(R.id.stop\_button);  
 sendButton = (Button) findViewById(R.id.send\_button);  
 *//clearButton = (Button) findViewById(R.id.clear\_botton);  
 //editText = (EditText) findViewById(R.id.entrada);* textView = (TextView) findViewById(R.id.med\_pulso);  
 setUiEnabled(**false**);  
  
 }  
 **public void** setUiEnabled(**boolean** bool)  
 {  
 startButton.setEnabled(!bool);  
 stopButton.setEnabled(bool);  
 sendButton.setEnabled(bool);  
 textView.setEnabled(bool);  
  
 }  
 **public boolean** BTinit()  
 {  
 **boolean** found=**false**;  
 BluetoothAdapter bluetoothAdapter=BluetoothAdapter.getDefaultAdapter();  
 **if** (bluetoothAdapter == **null**) {  
 Toast.makeText(getApplicationContext(),**"Device doesnt Support Bluetooth"**,Toast.LENGTH\_SHORT).show();  
 }  
 **if**(!bluetoothAdapter.isEnabled())  
 {  
 Intent enableAdapter = **new** Intent(BluetoothAdapter.ACTION\_REQUEST\_ENABLE);  
 startActivityForResult(enableAdapter, 0);  
 **try** {  
 Thread.sleep(1000);  
 } **catch** (InterruptedException e) {  
 e.printStackTrace();  
 }  
 }  
 Set<BluetoothDevice> bondedDevices = bluetoothAdapter.getBondedDevices();  
 **if**(bondedDevices.isEmpty())  
 {  
 Toast.makeText(getApplicationContext(),**"Please Pair the Device first"**,Toast.LENGTH\_SHORT).show();  
 }  
 **else** {  
 **for** (BluetoothDevice iterator : bondedDevices)  
 {  
 **if**(iterator.getAddress().equals(DEVICE\_ADDRESS))  
 {  
 device=iterator;  
 found=**true**;  
 **break**;  
 }  
 }  
 }  
 **return** found;  
 }  
  
 **public boolean** BTconnect()  
 {  
 **boolean** connected=**true**;  
 **try** {  
 socket = device.createRfcommSocketToServiceRecord(PORT\_UUID);  
 socket.connect();  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 connected=**false**;  
 }  
 **if**(connected)  
 {  
 **try** {  
 outputStream=socket.getOutputStream();  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
 **try** {  
 inputStream=socket.getInputStream();  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
  
 }  
  
  
 **return** connected;  
 }  
  
  
  
  
 **public void** onClickStart(View view) {  
  
 **if**(BTinit()) {  
 setUiEnabled(**true**);  
 **if**(BTconnect())  
 {  
  
 deviceConnected=**true**;  
 beginListenForData();  
 textView.append(**"\nConnection Opened!\n"**);  
 }  
  
 }  
 }  
  
 **void** beginListenForData()  
 {  
 **final** Handler handler = **new** Handler();  
 stopThread = **false**;  
 buffer = **new byte**[1024];  
 Thread thread = **new** Thread(**new** Runnable()  
 {  
 **public void** run()  
 {  
 **while**(!Thread.currentThread().isInterrupted() && !stopThread)  
 {  
 **try** {  
 **int** byteCount = inputStream.available();  
 **if**(byteCount > 0)  
 {  
 **byte**[] rawBytes = **new byte**[byteCount];  
 inputStream.read(rawBytes);  
 **final** String string=**new** String(rawBytes,**"UTF-8"**);  
 handler.postDelayed(**new** Runnable() {  
 **public void** run()  
 {  
 textView.append(string);  
 }  
 },500);  
 handler.postDelayed(**new** Runnable() {  
 @Override  
 **public void** run() {  
 textView.setText(**""**);  
 }  
 },100);  
  
 }  
 }  
 **catch** (IOException ex)  
 {  
 stopThread = **true**;  
 }  
 }  
 }  
 });  
  
 thread.start();  
 }  
  
 **public void** onClickSend(View view) {  
 String string = **"a"**;  
 string.concat(**"\n"**);  
 **try** {  
 outputStream.write(string.getBytes());  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
 Toast.makeText(**this**, **"Medicion iniciada"**,  
 Toast.LENGTH\_LONG).show();  
  
  
  
 }  
  
 **public void** onClickStop(View view) **throws** IOException {  
 stopThread = **true**;  
 outputStream.close();  
 inputStream.close();  
 socket.close();  
 setUiEnabled(**false**);  
 textView.setText(**""**);  
 deviceConnected=**false**;  
 Toast.makeText(**this**, **"Conexion terminada"**,  
 Toast.LENGTH\_SHORT).show();  
  
 *//textView.append("\nConnection Closed!\n");* }  
  
 **public void** onClickOff(View view) {  
 String string1 = **"b"**;  
 string1.concat(**"\n"**);  
 **try** {  
 outputStream.write(string1.getBytes());  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
 textView.setText(**""**);  
 Toast.makeText(**this**, **"Medicion terminada"**,  
 Toast.LENGTH\_SHORT).show();  
  
  
  
  
 }  
  
  
  
}

Activity Perfil

**package** com.example.nursebotpst;  
  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.os.Bundle;  
  
**public class** Perfil **extends** AppCompatActivity {  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_perfil);  
 }  
}

**Activity SIgnln**

**package** com.example.nursebotpst;  
  
**import** android.content.Intent;  
**import** android.database.sqlite.SQLiteDatabase;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.os.Bundle;  
**import** android.view.View;  
**import** android.widget.EditText;  
**import** android.widget.Toast;  
  
**public class** SignIn **extends** AppCompatActivity {  
 **private** EditText nombre,user,mail,pass,edad,sangre,peso,codigo;  
  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_sign\_in);  
 }  
  
 **public void** registrarUsuario(View v){  
  
  
  
  
  
  
  
 }  
}

**CÓDIGO PHP**

**Descripción Código PHP**

El siguiente código fue utilizado para conectar la base de datos externa en phpmyadmin con la aplicación en Android (NurseBotPST), la cual valida la contraseña ingresada por el usuario con el código del equipo existente en la base de datos.

<?php

// All database connection variables

$db\_user="root";

$db\_password="root1234";

$db\_name="NurseBot";

$db\_server="localhost";

$con = mysqli\_connect($db\_server,$db\_user,$db\_password,$db\_name);

$cod = $\_POST["codig"];

$result = mysqli\_query($con, "select \* from Equipo where codigo=".$cod);

while($row = mysqli\_fetch\_array($result)){

echo 1;//$row["codigo"];

//echo $row["modelo"];

}

//echo "Consulta realizada exitosamente...";

?>