

# BARRAS PARA FLEXIONES

Alben Adrian Ramirez Gomez, Marlon Javier Cabrera Montenegro,

Henry David Bran Velasquez, Diego Fernando Cortez Lopez,

Katerine Adalinda Santos Ramirez, Julio José Orellana Ruíz

Universidad de San Carlos de Guatemala, Facultad de Ingeniería

## I. INTRODUCCIÓN

Las flexiones son uno de los ejercicios más conocidos por la mayoría de personas, siendo uno de los ejercicios más recomendados para ser realizado desde casa. Se debe desarrollar un objeto automatizado basado en técnicas de IoT capaz de recolectar datos sobre los ejercicios que se hayan realizado con los aparatos, considerando la construcción de las barras para flexiones, el posicionamiento de los sensores y el posicionamiento de un sensor a la altura del pecho para realizar mediciones correctas. Se debe desarrollar una web progressive app para poder monitorear los resultados en tiempo real y poder realizar un análisis del proceso de los entrenamientos.

## I. OBJETIVOS

### A. Generales

- Diseña un dispositivo que permita analizar los resultados de los entrenamientos de flexiones.

### B. Específicos

- Diseña una web progressive app que permite ver los resultados en tiempo real.
- Utilizar técnicas de IoT para recolectar datos sobre los ejercicios,
- Aprender a desarrollar una solución mediante la correcta implementación del framework de IoT.

## I. DESARROLLO DE LA PRÁCTICA

### A. Bill of Material

#### a. Listado de Materiales físicos:

- Arduino Mega 2560
- Sensor de ritmo cardiaco y oxígeno
- Sensor de ultrasonido
- Módulo bluetooth HC-05
- Placa perforada
- Cinta de aislar
- Madera
- Tornillos
- Goma
- Fomi
- Sellante adhesivo

#### b. Listado de Materiales digitales:

- API de ingreso de parámetros
- Componentes digitales de diseño

### B. Magnitudes físicas a medir

- Calorías quemadas

$$\text{Calorías} = 0.049 * \text{Peso} * 2.2 * \text{tiempo}$$

Ecuación 1: Calorías gastadas durante vrs Tiempo de la práctica

- Pulso cardiaco y oxígeno (bpm)
- Rango de movimiento (cm)

$$\text{rango} = \text{altura}_{\text{máxima}} - \text{altura}_{\text{mínima}}$$

Ecuación 2: Distancia recorrida por el movimiento en vertical del cuerpo

- Frecuencia de repeticiones

$$\text{repeticiones} = \text{No. de movimiento}$$

Ecuación 3: Mediante valores de activación de arriba y abajo del arduino

### C. Funciones Principales:

#### • Medición de la frecuencia cardiaca:

El dispositivo será capaz de medir e interpretar la frecuencia cardiaca del usuario a lo largo de su uso.

#### • Medición del rango de movimiento:

El dispositivo deberá ser capaz de medir el movimiento de arriba y abajo del usuario al realizar el desplazamiento en el ejercicio.

#### • Medición de la frecuencia de repeticiones:

La unidad debe poder medir el ritmo de trabajo del usuario, cuando inicia sus desplazamientos se calculan las repeticiones .

● **Medición de las calorías quemadas:**

El dispositivo deberá ser capaz de calcular la cantidad de calorías quemadas durante el entrenamiento..

## D. Procedimiento

### 1. Sensores

● Arduino Mega 2560

Tamaño	Lectura Sensor	Instalación	Microcontrolador	Medición voltaje
10.0 X 6.0 X 1.0 cm	Pines de E/S digitales: 54 (de los cuales 14 proporcionan salidas PWM) Pines de entrada analógica: 16	Objetivo: áreas exteriores, interiores.	ATMEGA2560	7 a 12V

● link datasheet:

[https://content.arduino.cc/assets/Pinout-Mega2560rev3\\_latest.pdf](https://content.arduino.cc/assets/Pinout-Mega2560rev3_latest.pdf)



Precio: Q 229.00

● Sensor de ultrasonido

Tamaño	Lectura Sensor	Instalación	Voltaje de Operación	Unidad de Medida
2.0 X 7.0 X 2.0 cm	Ultrasonido	Objetivo: Hacer un rango de movimiento y repeticiones..	3.3V a 5.0V	Digital, rango de medición de 2 a 450 cm

● link datasheet:

[http://www.energiazero.org/arduino\\_sensores/Arduino%20ultrasonic%20sensor%20\(HC-SR04%20or%20HY-SRF05\).pdf](http://www.energiazero.org/arduino_sensores/Arduino%20ultrasonic%20sensor%20(HC-SR04%20or%20HY-SRF05).pdf)



● Precio: Q25.00

● Sensor de ritmo cardiaco

Tamaño	Medición voltaje	Instalación	Rango de Medición	Unidad de Medida
1.4 X 1.7 X 0.3 cm	5.0 V	Objetivo: Percibir el ritmo cardiaco.	entre 50 y 60 Hz y temperatura de -40 a 85 Celsius.	Ritmo en BPM

● link datasheet:

<https://datasheets.maximintegrated.com/en/ds/MA-X30102.pdf>



● Precio: Q47.00

● Módulo bluetooth

Tamaño	Lectura Sensor	Instalación	Microcontrolador	Medición voltaje
2.4 X 1.4 X 0.8 cm	Pines de Gnd, RXD, TXD, Eneble, Vcc	Objetivo: Transportar la información mediante protocolos de bluetooth.	CSR BC417143	3.3 V

● link datasheet:

[https://components101.com/sites/default/files/component\\_datasheet/HC-05%20Datasheet.pdf](https://components101.com/sites/default/files/component_datasheet/HC-05%20Datasheet.pdf)



● Precio: Q60.00

## 2. Conectividad

● La transmisión de datos se realiza por medio del arduino que envía los datos a través del módulo bluetooth al pc, esto ocurre de forma inalámbrica.

● Los datos recibidos en la PC son almacenados en una base de datos persistente y este a su vez son procesados en la aplicación processing a través de una imagen.

- Datos que deben de enviar desde la unidad al PC mediante Bluetooth:
  - Medición Ritmo Cardiaco.
  - Rango de Movimiento.
  - Repeticiones del movimiento.

## ★ Arduino

El envío de datos al servidor se realizó de esta manera:

```
Loop (1 segundo) {
  Sensores Arduino (Memoria -> registros
  limitados)
  Módulo Ritmo Cardiaco
  Módulo de ultrasonido
  Módulo Bluetooth
}
```

Así mismo se envía los datos a la Base de Datos de la siguiente manera y mediante un switch de selección:

```
return "{" + RH+ "," + Constant_Ritmo_Cardiaco
"}";

return      "{"      +      Repeticiones"      +
Constante_Repetición "}";

return      "{"      +      Movimiento"      +
Constante_Movimiento + "};
```

Este a su vez es recibido y almacenado correctamente en la base de datos mediante el módulo bluetooth..

## ★ NodeJs

Se inicializa la configuración con la base de datos en MySQL, que está corriendo en un servidor de Google Cloud

```
var mysql = require('mysql')
var conexion = mysql.createConnection(
{
  host: '104.198.193.190',
  user: 'root',
  password: '1234',
  database: 'barras_paralelas',
  port: '3306'
})
```

```
conexion.connect(function(err){
  if(err) {
    throw err
  }else{
    console.log("Conection whit Mysql
ready")
  }
})
```

```
module.exports = conexion;
```

Se importan las librerías que se utilizarán para las peticiones en la base de datos.

```
var express = require('express')
var http = require('http');
var conexion = require('../DB/db')
```

```
var app = express()
app.use(express.json());
app.use(express.urlencoded({extended:
true}));
```

El servidor de Nodejs es utilizado para realizar peticiones a la base de datos de Google Cloud. Las peticiones utilizadas están descritas en el apartado 5.a

## ★ Dashboard de entrenamiento

El dashboard de entrenamiento se utiliza para seleccionar el tipo de entrenamiento a realizar y así de forma visual el usuario pueda obtener retroalimentación de su desempeño en tiempo real.



Figura 1: Inicio de sesión del usuario



Figura 2: Registro de usuario



Figura 3: Dashboard de entrenamiento para el usuario

## ★ Aplicación web

La aplicación web se utilizará para mostrar el rendimiento del atleta, los datos mostrados serán en un rango de fecha específico.

Los bocetos de la aplicación son los siguientes:



Figura 4: Dashboard propuesto para el login

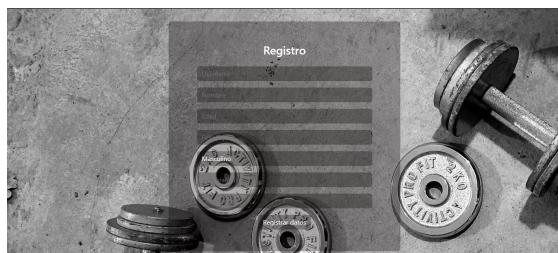


Figura 5: Dashboard propuesto para el registro



Figura 6: Dashboard propuesto para menu de entrenamiento



Figura 7: Dashboard propuesto para el reporte de entrenamiento

## ★ Entorno del Objeto



Este puede permanecer en un área exterior y seguir funcionando correctamente(suponiendo que esté sellado el prototipo). Por ejemplo:

- Patio
- Interiores de casa

El mejor caso para probar el prototipo es en un espacio semi cerrado donde no haya mucha interferencias del ambiente entre otros factores.

## ★ Tamaño del objeto

Cada barra paralela, posee una dimensión de 32 cm de largo, y en cada punta posee una base de 11 cm x 7 cm. además posee una pequeña base donde está el sensor de proximidad la cual tiene 11 cm de largo x 9 cm de ancho.

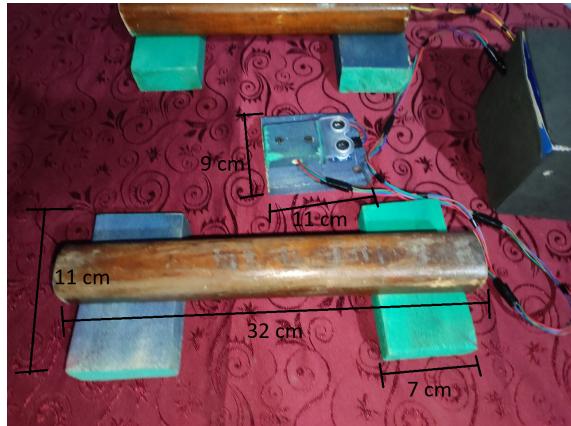


Figura 8: Dimensiones del prototipo

### 3. Bocetos del Prototipo

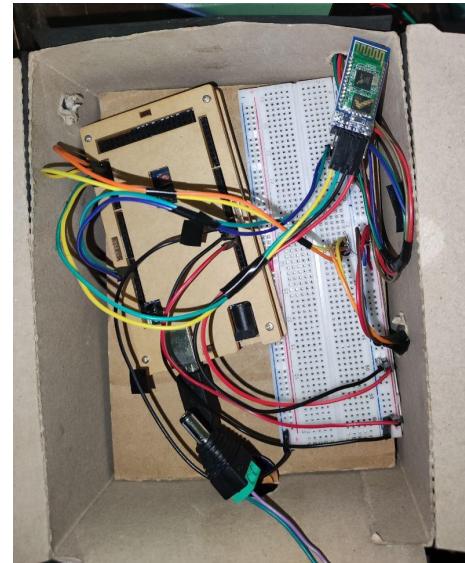


Figura 9: Prototipo visualizado de parte superior

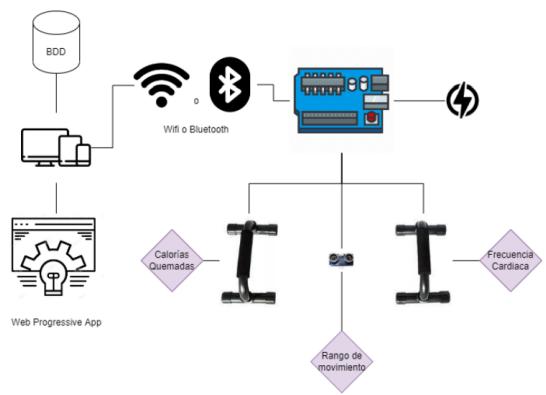


Figura 5: Comunicación entre los componentes al arduino y del arduino a el manejo de datos y software.

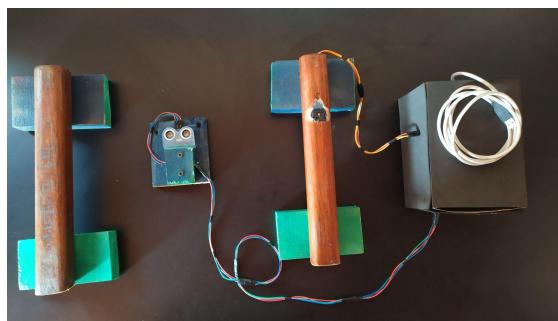


Figura 10: Prototipo

Figura 10: Dashboard visto desde un dispositivo móvil.

### 4. Base de Datos

En la base de datos se maneja Google Cloud mediante esta plataforma se aloja una instancia para la base de datos: la instancia de la base de datos es en MySQL. De la cual generamos una ER como almacenamiento de datos:

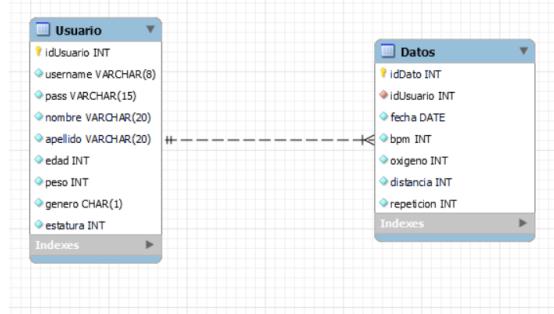


figura 11: ER de los datos

Los datos se representan de forma de tablas en donde se va estructurar los datos:

usuario -> datos

Los datos van a depender de un usuario, Cada usuario debe ser registrado e iniciado sesión para poder almacenar los datos, por el mismo motivo un entrenamiento dependerá de un usuario.

las tablas se muestran de la siguiente manera:

Column Name	Datatype
username	VARCHAR(8)
pass	VARCHAR(15)
nombre	VARCHAR(20)
apellido	VARCHAR(20)
edad	INT
peso	INT
genero	CHAR(1)
estatura	INT

Figura 12: Tabla Usuario

Los datos Usuario nos darán la clave foránea de la siguiente tabla, que es la que llevará los datos del entrenamiento, estos datos se verán representados en la aplicación web o móvil:

Column Name	Datatype
idData	INT
idUsuario	INT
fecha	DATE
bpm	INT
oxigeno	INT
distancia	INT
repeticion	INT

figura 13: Tabla Entrenamiento

### a. Análisis Descriptivo:

- Aplicar IOT para la resolución de problemas en la vida cotidiana a través del uso de sensores y mecanismos que permitan monitorear los datos obtenidos, en este caso el mecanismo se basa en un par de barras paralelas, en las cuales activaremos los músculos de los brazos, espalda y pecho, así mismo es recomendable mezclarlo con otro tipo de ejercicio para la parte inferior del cuerpo.

### b. Análisis de Diagnóstico:

- IOT, Aplicar el IOT para la resolución activación de la parte superior del cuerpo
- Conexión de sensores al arduino uno, para extraer los datos obtenidos que se ejecuten en el sensor.
- Conexión bluetooth para enviar los datos hacia una conexión cliente para recibir los datos que el arduino mandara.
- Por medio de programación entre software's, se puede obtener, relacionar y almacenar los datos que se obtengan.
- Los datos obtenidos se manejaron desde un servidor donde aloja los resultados claramente esclarecidos y según los resultados se ejecutará un servicio de front-end para los distintos dinamismos.

## 5. Conocimientos aplicados

### a. Recuperar datos de la base de datos

```
var query = coon.query(
  "SELECT * FROM Usuario;",
  function(err, result){
    if (err) throw err
    else res.send(result)
  }
)
```

### b. Ingresar datos a la base de datos. con la API

```
app.post("/register", function(req, res){
  console.log(req.body)
  var query = coon.query(
    `INSERT INTO Usuario (IdUser, Contra,
    Nombre, Edad, peso, Genero, Estatura) VALUES
    ("${req.body.IdUser}", "${req.body.Contra}", "${req.body.Nombre}", ${req.body.Edad},
    ${req.body.peso}, "${req.body.Genero}", ${req.body.Estatura});`,
    function(err, result){
      if (err) throw err
      else{
        console.log(result)
        res.send(result)
        console.log(result)
      }
    }
  )
});
```

### c. Puerto del servidor

```
app.listen(
  4001,
  ()=> console.log('Server Port: ', 4001)
)
```

### d. Conexión para la base de datos MySQL:

```
var mysql = require('mysql')
var coon = mysql.createConnection({
```

```

    {
      host: '104.198.193.190',
      user: 'root',
      password: '1234',
      database: 'barras_paralelas',
      port: '3306'
    }
  )

coon.connect(function(err){
  if(err){
    throw err
  }else{
    console.log("Conexion con la base de datos
lista!!!!!");
  }
})

```

module.exports = coon;

## **6. Link del repositorio de github**

- [https://github.com/adrianrg578/ACE2\\_2S22\\_G2.git](https://github.com/adrianrg578/ACE2_2S22_G2.git)

## **7. Link del video del prototipo**

- [https://youtube.com/playlist?list=PL6GFUI5F47n82oaGU8zTI\\_NskuJw2Fpl-](https://youtube.com/playlist?list=PL6GFUI5F47n82oaGU8zTI_NskuJw2Fpl-)

## **III. REFERENCIAS**

-  COMO HACER UNAS MINI PARALEL...