# DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA ELECTRÓNICO PARA EL MONITOREO POSTURAL EN TIEMPO REAL DE LA CADERA Y LOS HOMBROS EN PERSONAS QUE PERMANECEN SENTADAS

IVÁN RICARDO GÓMEZ CALDERÓN
ERIKA ALEJANDRA VALCARCEL RINCÓN

REALIZADO CON LA ASESORÍA DE:

EMILIO ALEJANDRO SÁNCHEZ PINZÓN

UNIVERSIDAD EL BOSQUE
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA ELECTRÓNICA
OCTUBRE, 2024

UNIVERSIDAD EL BOSQUE
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA ELECTRÓNICA

ÁREA DE ÉNFASIS: EQUIPOS MÉDICOS

# DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA ELECTRÓNICO PARA EL MONITOREO POSTURAL EN TIEMPO REAL DE LA CADERA Y LOS HOMBROS EN PERSONAS QUE PERMANECEN SENTADAS

IVÁN RICARDO GÓMEZ CALDERÓN

ERIKA ALEJANDRA VALCARCEL RINCÓN

REALIZADO CON LA ASESORÍA DE:

EMILIO ALEJANDRO SÁNCHEZ PINZÓN

Página de Aprobación. Inclusión de Acta de grado.

## NOTA DE SALVEDAD

Según el artículo 37 del 14 de diciembre de 1989 del acuerdo 017, “La Universidad El Bosque, no se hace responsable de los conceptos emitidos por los investigadores en su trabajo, solo velará por el rigor científico, metodológico y ético del mismo en aras de la búsqueda de la verdad y la justicia”.

## AGRADECIMIENTOS

Agradecemos profundamente a nuestras familias, quienes han sido nuestro apoyo incondicional durante este proceso. Su comprensión y amor han sido nuestra motivación. Sin su respaldo y sacrificio este logro no habría sido posible.

A Emilio Alejandro Sánchez Pinzón, por su invaluable guía y apoyo en el desarrollo de este proyecto de grado. Su paciencia, dedicación y conocimientos han sido fundamentales para nuestra formación académica, profesional y humana.

Agradecemos a Ana Isabel Duran, por brindarnos su valiosa colaboración y asesoría en la revisión del documento.

Finalmente, agradecemos a cada uno de los docentes que han sido partícipes de nuestra formación académica.

**RESUMEN**

Actualmente, la mayoría de personas presentan problemas de mala postura debido al largo tiempo que permanecen sentados y aunque esto se resuelve con correctores posturales, no son lo suficientemente personalizados, ni tienen en cuenta estructuras fundamentales que intervienen en la posición de sedestación, como lo son los hombros y la cadera. En este contexto, el siguiente proyecto de grado presenta el desarrollo de un dispositivo de monitoreo postural que analiza la posición de la cadera y los hombros. Tiene como ventaja el desarrollo de una aplicación móvil y una base de datos que permite el almacenamiento en la nube. Este dispositivo permite validar y analizar en tiempo real los ángulos de inclinación en los ejes: x, y, z, de hombros y cadera. Luego, compara esta información con los datos ingresados y almacenados en la aplicación, e identifica si el usuario se encuentra en una correcta postura y notifica el resultado. En conclusión, este dispositivo cumple con la función de monitorear y de promover la conciencia de la correcta postura.

*Palabras Clave:* Postura, Sedestación, Monitoreo, Aplicación Móvil

**ABSTRACT**

Currently, most people have problems of poor posture due to the long time they remain seated and although this is solved with postural correctors, they are not sufficiently customized, nor do they take into account fundamental structures involved in the sitting position, such as the shoulders and hips. In this context, the following degree project presents the development of a postural monitoring device that analyzes the position of the hips and shoulders. It has as an advantage the development of a mobile application and a database that allows storage in the cloud. This device allows to validate and analyze in real time the x, y, z tilt angles of shoulders and hips. Then, it compares this information with the medical data entered and stored in the application, and identifies if the user is in a correct posture and notifies the result. In conclusion, this device fulfills the function of monitoring and promoting awareness of correct posture.

*Keywords*: Posture, Sitting, Monitoring, Mobile Application

## LISTA DE SIMBOLOS

**V** Voltaje

**C** Corriente

° Grados sexagesimales

**TABLA DE CONTENIDOS**

Pág.


1. INTRODUCCIÓN ..... 22

2. PLANTEAMIENTO DEL PROBLEMA ..... 23

2.1Antecedentes y Estado del Arte ..... 23

2.1.1 Nivel Internacional ..... 23

2.1.2 Nivel Nacional ..... 25

2.2Descripción y Formulación del Problema ..... 27

2.3Justificación ..... 28

2.4Objetivos ..... 29

2.4.1 Objetivo General ..... 29

2.4.2 Objetivos Específicos ..... 29

2.5Alcance y Limitaciones del Proyecto ..... 30

3. MARCO DE REFERENCIA ..... 31

3.1Marco Teórico o Conceptual ..... 31

3.1.1 Anatomía de la cadera ..... 31

3.1.2 Biomecánica de la cadera ..... 33

3.1.3 Anatomía de los hombros ..... 36

3.1.4 Biomecánica de los hombros ..... 38

3.1.5 Sedestación ..... 41

3.1.6 Movimientos de rotación Pitch, Yaw y Roll ..... 43

4. DESARROLLO DEL PROYECTO DE GRADO ..... 45

4.1Requerimientos ..... 45

4.1.1 Requerimientos funcionales ..... 45

4.1.2 Requerimientos de Calidad ..... 45

4.1.3 Requerimientos restrictivos ..... 46

4.2Metodología del Diseño ..... 46

4.2.1 Metodología desarrollada ..... 46

4.2.2 Planeación de actividades ..... 47
4.2.3 Recursos ..... 48
4.2.4 Plan de Pruebas ..... 49
4.2.5 Diseño conceptual ..... 49
4.3Descripción Técnica del Producto ..... 54
4.4Identificación de movimientos que se realizan y articulaciones que intervienen en la posición de sedestación ..... 55
4.4.1 Identificación teórica de movimientos ..... 55
4.4.2 Identificación práctica de movimientos y articulaciones con un fisioterapeuta experto en ergonomía ..... 58
4.4.3 Identificación práctica de articulaciones ..... 59
4.5Selección de componentes principales ..... 61
4.6Calibración sensor inercial ..... 71
4.7Comunicación I2C, sensores inerciales y tarjeta de programación ..... 75
4.8Integración batería en el dispositivo ..... 76
4.9Relación entre movimientos de los hombros y la cadera y los movimientos de Roll, Pitch y Roll del sensor inercial ..... 78
4.10Diseño prenda de vestir para ubicación de sensórica ..... 80
4.11Almacenamiento de datos ..... 82
4.12Diseño aplicación móvil ..... 86
4.12.1 Diseño preliminar ..... 86
4.12.2 Diseño Final ..... 88
5. RESULTADOS Y ANÁLISIS DE RESULTADOS ..... 94
5.1Desarrollo del sistema electrónico para el monitoreo postural de la cadera y los hombros en personas que permanecen sentadas ..... 94
5.2Diseño del dispositivo electrónico para detectar la posición de hombros y cadera cuando una persona se encuentra sentada ..... 94
5.2.1 Validación calibración sensores inerciales: ..... 94
5.2.2 Captura de información del sensor inercial en movimientos de hombros y cadera en posición de sedestación ..... 97
5.3Implementación del dispositivo electrónico ..... 99

5.4Desarrollo del software para dispositivos móviles, capaz de monitorear y almacenar los datos recolectados ..... 101
5.4.1 Desarrollo de la aplicación móvil y conexión Firebase ..... 101
5.4.2 Aplicación móvil y conexión bluetooth con el dispositivo. 107
5.4.3 Aplicación móvil y monitoreo ..... 110
5.5Integración y evaluación del dispositivo electrónico y el software ..... 114
5.6Cumplimiento de requerimientos: ..... 120
5.6.1 Requerimientos funcionales:..... 120
5.6.2 Requerimientos de calidad:..... 121
5.6.3 Requerimientos restrictivos ..... 121
6. CONCLUSIONES ..... 123
7. RECOMENDACIONES..... 125
8. REFERENCIAS BIBLIOGRÁFICAS ..... 126

## LISTA DE TABLAS

Tabla Pág.

Tabla 1. Recursos Fisicos .....49

Tabla 2. Prueba calcar estructura física en papel. .....60

Tabla 3. Comparación de las características de los sensores inerciales.....62

Tabla 4. Comparación de microcontroladores. .....65

Tabla 5. Comparación de multiplexores. .....68

Tabla 6. Comparación de las características de las baterías. .....70

Tabla 7. Validación calibración sensor inercial 1. .....94

Tabla 8. Validación calibración sensor inercial 2. .....95

Tabla 9. Validación calibración sensor inercial 3. .....95

Tabla 10. Validación calibración sensor inercial 4. .....95

Tabla 11. Integración de dispositivo electrónico y aplicación móvil. .....117

Tabla 12. Cumplimiento de requerimientos funcionales .....120

Tabla 13. Cumplimiento de Requerimientos de calidad.....121

Tabla 14. Cumplimiento de requerimientos restictivos .....121

## LISTA DE FIGURAS

Figura Pág.

Figura 1. Pelvis, vista anterior [25].....31

Figura 2. Movimiento de flexión y extensión, abducción y aducción y rotación de la cadera [ 23].....33

Figura 3. Anteversión y retroversión de la cadera [24]. .....35

Figura 4. Inclinación lateral externa e inclinación lateral interna [24]. .....35

Figura 5. Rotación interna y rotación externa del ilíaco [24]. .....36

Figura 6. Omóplato o escápula [25].....36

Figura 7. Elevación, descenso y abducción del hombro respecto al omóplato [24].....38

Figura 8. Aducción, campaneo interno y campaneo externo del hombro respecto al omóplato [24].....39

Figura 9. Antepulsión y retropulsión del brazo respecto al omóplato [24]. .....39

Figura 10. Abducción y aducción del brazo respecto al omóplato [24] .....39

Figura 11. Rotación interna y externa del brazo respecto al omóplato [24]. .....40

Figura 12. Antepulsión y retropulsión del hombro a mayor amplitud [24]. .....40

Figura 13. Abducción y aducción del hombro a mayor amplitud [24]. .....41

Figura 14. Rotación interna y externa del hombro a mayor amplitud [24]. .....41

Figura 15. Roll, Pitch y Yaw [30].....44

Figura 16. Diagrama de caja negra, entradas y salidas del sistema [Propiedad de los autores].....50

Figura 17. Diagrama detallado de entradas y salidas del sistema [Propiedad de los autores]. 53

Figura 18. Movimiento de hombros y análisis de omoplatos [Propiedad de los autores]. 58

Figura 19. Prueba 1 articulaciones que se involucran en la posición de sedestación [Propiedad de los autores].....60

Figura 20. Prueba 2 articulaciones que se involucran en la posición de sedestación [Propiedad de los autores].....60

Figura 21. Prueba 3 articulaciones que se involucran en la posición de sedestación [Propiedad de los autores].....60

Figura 22. Sensor inercial MPU6500 [33].....61

Figura 23. Sensor inercial CJMCU-103 [34].....62

Figura 24. Sensor inercial MPU6050 [35].....62

Figura 25. Arduino nano [36]. .....64

Figura 26. ESP-32 [37 - 38].....64

Figura 27. Raspberry Pi Zero 2W [39]. .....64

Figura 28. TSTM32F103C8T6 [40] .....65

Figura 29. Multiplexor 1 a 8 I2C [41] .....67

Figura 30. Multiplexor 1 a 8 I2C [42] .....67

Figura 31. Multiplexor 1 a 8 I2C [43] .....67

Figura 32. Batería Litio 18650 - 2500mah 3.7v [44].....69

Figura 33. Pilas AA alcalina 1.5v [45]. .....69

Figura 34. Power Bank 12v [46].....69

Figura 35. Medición física para eje Y [Propiedad de los autores].....72

Figura 36. Medición física para eje Z [Propiedad de los autores]. .....72

Figura 37. Curva de calibración sensor 1 [Propiedad de los autores].....73

Figura 38. Curva de calibración sensor 2 [Propiedad de los autores].....73

Figura 39. Curva de calibración sensor 3 [Propiedad de los autores].....73

Figura 40. Curva de calibración sensor 4 [Propiedad de los autores].....74

Figura 41. Conexión Multiplexor 1 a 8 I2C [Propiedad de los autores].....76

Figura 42. Integración Batería [Propiedad de los autores]. .....77

Figura 43. Diseño preliminar del arnés [Propiedad de los autores].....81

Figura 44. Diseño preliminar  camiseta [Propiedad de los autores]. .....81

Figura 45. Diseño Camiseta[Propiedad de los autores].....82

Figura 46. Diagrama Entidad relación [Propiedad de los autores]. .....83

Figura 47. Diseño servicio Firebase Authentication [Propiedad de los autores].....84

Figura 48.Diseño servicio Cloud Firestore [Propiedad de los autores]. .....84

Figura 49. SDK de  Firebase [Propiedad de los autores]….....85

Figura 50. Conexión Android Studio y  Firebase Authentication [Propiedad de los autores]…….85

Figura 51 .Conexión Android Studio y  Cloud Firestore [Propiedad de los autores].....86

Figura 52. Diseño preliminar aplicación móvil [Propiedad de los autores]. .....87

Figura 53. Frontend ventana de Login [Propiedad de los autores]. .....89

Figura 54. Frontend ventana de Manual de usuario [Propiedad de los autores]. .....89

Figura 55. Frontend ventana de Información personal [Propiedad de los autores]. .....90

Figura 56. Frontend ventana de Información médica [Propiedad de los autores]. .....92

Figura 57. Frontend ventana de Monitoreo [Propiedad de los autores].....93

Figura 58. Sensorica en postura correcta [Propiedad de los autores]. .....97

Figura 59. Sensorica en abducción de hombros [Propiedad de los autores].....98

Figura 60. Sensorica en inclinación lateral de cadera [Propiedad de los autores]. .....98

Figura 61. Sensorica en elevación y descenso de hombros [Propiedad de los autores]. ..99

Figura 62. Cable ribbon AWG 10 Hilos [55]. .....100

Figura 63. Posicionamiento de correas en prenda[Propiedad de los autores]. .....101

Figura 64. Notificación de error Ventana de Login [Propiedad de los autores]. .....103

Figura 65. Ventana de Login [Propiedad de los autores]. .....104

Figura 66. Funcionamiento conexión Android Studio y Firebase Authentication [Propiedad de los autores]. .....104

Figura 67 . Funcionamiento conexión Android Studio y Cloud Firestore, tabla de usuarios [Propiedad de los autores].....105

Figura 68. Ventana de Información personal [Propiedad de los autores].....105

Figura 69. Funcionamiento conexión Android Studio y Cloud Firestore, tabla de información médica [Propiedad de los autores]. .....106

Figura 70 .Ventana de Información médica [Propiedad de los autores].....106

Figura 71. Librerías bluetooth Arduino [Propiedad de los autores]. .....107

Figura 72. Lista de conexiones bluetooth Aplicación móvil[Propiedad de los autores]. 108

Figura 73. Notificación de conexión bluetooth [Propiedad de los autores].....109

Figura 74. Notificación de vinculación de dispositivos bluetooth rechazada [Propiedad de los autores]. .....110

Figura 75. Ventana de Monitoreo sin notificaciones [Propiedad de los autores]. .....111

Figura 76. Configuración bluetooth SerialPrint Arduino [Propiedad de los autores]. ....112

Figura 77. Ventana de Monitoreo con notificaciones [Propiedad de los autores]. .....113

Figura 78. Postura correcta [Propiedad de los autores]. .....114

Figura 79. Abducción de hombros[Propiedad de los autores]. .....115

Figura 80. Elevación y descenso de hombros [Propiedad de los autores]. .....115

Figura 81. Ventana de Monitoreo integrada con dispositivo electrónico [Propiedad de los autores].....116

# 1. INTRODUCCIÓN

En la actualidad, la mayoría de personas enfrentan problemas de postura, debido a los extensos periodos de tiempo que permanecen sentados, ya sea por motivos laborales, académicos o por el uso de vehículos. Para resolver esta problemática se suelen utilizar correctores posturales textiles o inteligentes que pueden llegar a ser invasivos, poco cómodos, genéricos y no fomentan una conciencia corporal adecuada en quienes los usan.

En este contexto, el siguiente proyecto tiene como objetivo desarrollar un sistema de monitoreo postural personalizable. Este sistema analiza estructuras principales que intervienen en la posición de sedestación y que en los correctores de postura no son tenidos en cuenta, como lo son: hombros y cadera. Además, el dispositivo incluye una aplicación móvil y una base de datos para que tanto el usuario pueda monitorear su postura.

Es importante señalar que este proyecto no está destinado a la venta al público, ya que, requiere la supervisión de un profesional médico para su uso. Además, el dispositivo se enfoca en el monitoreo de la postura, más no en la corrección, promoviendo así que el usuario tome conciencia de su postura.

Este proyecto se enmarca en la línea de dispositivos médicos y busca que la intersección entre la Medicina y la Ingeniería Electrónica propenda por una mejor calidad de vida para las personas. La exploración de esta área del conocimiento demuestra cómo la tecnología puede ser una herramienta valiosa tanto para los fisioterapuetas como para los pacientes, favoreciendo una relación más consciente con su propio cuerpo.

En este orden de ideas, el texto cuenta con los siguientes capítulos: 1) contextualización del problema 2) marco teórico 3) requerimientos 4) diseño del dispositivo 5) pruebas 6) análisis de resultados y 7) conclusiones.

# 2. PLANTEAMIENTO DEL PROBLEMA

## 2.1 Antecedentes y Estado del Arte

Se consideran diversos proyectos de grado, artículos y dispositivos que se encuentran en el mercado nacional e internacional, lo que proporciona una visión más amplia sobre los distintos sistemas destinados corregir, monitorear o entrenar la postura en movimientos como bipedestación, sedestación y marcha. Además, la investigación permite identificar los sensores más efectivos y comunes para el análisis postural, así como los métodos de visualización y almacenamiento de datos.

### 2.1.1 Nivel Internacional

**Estudios y prototipos en el mercado internacional para corregir, monitorear o entrenar la postura**

La tesis "Activity and posture recognition for the elderly based on a wearable device" [1] plantea el diseño de StraightenUp, un dispositivo capaz de monitorear las actividades de personas mayores, específicamente medir estáticamente la inclinación de la columna vertebral mediante un sistema de tres sensores inerciales (acelerómetro y giroscopio LilyPad ADXL335) distribuidos en un arnés a lo largo de la columna vertebral.

Dichos sensores son capaces de clasificar 6 posturas humanas en tiempo real, con la ayuda de un multiplexor (TCA9548A) que permite que los sensores cuenten con la misma dirección para minimizar errores de orientación y una batería de litio de 3.7V para la alimentación de la tarjeta. Los datos de inclinación son captados por los sensores y enviados a través de Bluetooth a un computador portátil para su respectivo análisis [1].

Esta sistema no solamente busca monitorear con precisión la postura, también pretende que el usuario se sienta cómodo y tranquilo, buscando que prime la experiencia del usuario al utilizar el sistema de monitoreo, para esto su diseño consta

de tres bandas (superior, central, inferior) que son ajustadas en la parte frontal (hombros, pecho, cintura) generando así un diseño de fácil ajuste y siendo adaptable para los adultos mayores [1].

En el artículo "Chaleco Inteligente para Corrección de Postura con Retroalimentación" [2] se observa el diseño de un chaleco con dos sensores acelerómetros y giroscopios (MPU6050) ubicados en los hombros para detectar la mala postura, adicional cuenta con un mecanismo de estimulación, específicamente impulsos eléctricos que informan al usuario de la mala postura y fomentan la corrección física a través de un zumbador.

Adicionalmente, el dispositivo permite la visualización en tiempo real del ángulo de inclinación de los hombros del usuario, mediante una aplicación móvil diseñada en AppInventor, permitiendo graficar los datos recolectados y configurar las funciones del chaleco, además de realizar un inicio de sesión que permite guardar los datos de corrección de cada usuario incluyendo ejercicios de estiramiento para mejorar la estabilidad del cuerpo y aliviar el dolor causado por una mala postura [2].

El diseño del chaleco busca la ergonomía, comodidad y facilidad de uso utilizando correas y un cinturón que permiten el ajuste para diferentes tipos de cuerpo, para que el usuario se adapte fácilmente u convierta en un hábito la utilización del mismo [4].

Para corroborar la correcta funcionalidad del chaleco se utiliza un goniómetro que permite medir los ángulos de inclinación de los hombros y compararlos con los valores del sensor y así validar que los datos recolectados por el dispositivo son verdaderos, para esto se utiliza la prueba t pareada, la cual facilita analizar la diferencia promedio entre dos conjuntos de datos relacionados, lo que facilita la determinación de si esa diferencia es estadísticamente significativa y si son lo suficientemente proporcionales para considerarse similares [2].

**Dispositivos disponibles en el mercado internacional para corregir, monitorear o entrenar la postura**

Upright Go2 es uno de los correctores de postura más avanzados, con dos sensores giroscopios integrados, capaces de brindar la información necesaria para entrenar la

postura, ya sea en posición de sedestación, bipedestación o marcha. Dicho dispositivo se debe ubicar en la parte superior de la espalda mediante un adhesivo, el cual avisa con una suave vibración cuando el usuario se encorva. Adicionalmente cuenta con una aplicación capaz de almacenar y sincronizar automáticamente la actividad postural detallada durante varios días [3].

Wearlumb es un maillot inteligente con diferentes sensores localizados en los brazos, la zona lumbar, la columna y las cervicales, permitiendo medir los ángulos de flexión lumbar, desviación lateral y rotación de la columna, corrigiendo e indicando cuando se tiene una mala postura. Adicionalmente, cuenta con una aplicación móvil capaz de recopilar información de postura en tiempo real, la cual debe ser interpretada por un especialista en ergonomía, siendo así, un dispositivo de carácter de estudio y análisis de prevención de riesgos laborales en compañías específicas que detectan problemas de postura en sus trabajadores y no de carácter comercial [4].

### 2.1.2 Nivel Nacional

**Estudios y prototipos en el mercado nacional para corregir, monitorear o entrenar la postura**

En el trabajo de grado “ Diseño e implementación de un prototipo de dispositivo biomédico no invasivo para la prevención de la postura incorrecta a nivel dorsal ” [5] se observa la implementación de un sensor MPU6050, un motor vibrador redondo y un microcontrolador NodeMCU como componentes principales para el diseño del instrumento de medición de posición corporal.

El sensor detecta el ángulo de inclinación en el eje Y implementando la función de acelerómetro del dispositivo y basándose en el Teorema de Pitágoras. Adicional, se ubica en la región dorsal de la espalda, específicamente sobre la séptima vértebra de la parte cervical, ya que, es donde se presenta la mayor curvatura cifótica y es el lugar donde el sensor puede leer fácilmente el cambio de ángulo en la posición de la espalda del usuario. Por otra parte, el motor funciona como alarma vibratoria para informar al usuario cuando se adopta una postura incorrecta, activándose cuando la posición supera los 45° [5].

Adicionalmente, el dispositivo permite la visualización en tiempo real del ángulo de inclinación de la espalda del usuario, mediante una aplicación móvil. Además esta permite almacenar la información recolectada en una base de datos ThingSpeak realizando un monitoreo diario y constante de la postura del usuario [5].

Por otra parte, el trabajo de grado " Diseño de un sistema de monitoreo en tiempo real por medio de iot para la medición y análisis de datos sobre el nivel de riesgo ergonómico de trabajadores del sector metalmecánico" [6] presenta el diseño de un dispositivo de monitoreo postural para la medición del riesgo ergonómico que pueden enfrentar las extremidades superiores de los trabajadores en el sector metalmecánica, enfocándose específicamente en la articulación del brazo, la muñeca y la mano, utilizando como elemento principal el sensor MPU6050 y su función de giroscopio para detectar el ángulo de inclinación en los ejes x,y y z basándose en los movimientos que realiza un avión; Yaw, Pitch y Roll respectivamente [6].

Para efectuar la conexión de los sensores ubicados en las tres articulaciones mencionadas anteriormente se implementa el protocolo I2C, donde se conecta el pin de cambio de memoria a los 3 sensores desde el AD0 de cada uno a un pin diferente de la tarjeta para poder controlar el cambio de dirección de memoria [6].

Los datos recolectados por el dispositivo se visualizan en una aplicación móvil, tanto numérica como gráficamente, presentando el ángulo de inclinación en el eje x, y y z, , además de almacenar dichas variables e información de usuario en un base de datos de MySQL.

**Dispositivos disponibles en el mercado nacional para corregir, monitorear o entrenar la postura**

El corrector de postura de espalda universal es el más vendido a nivel nacional, teniendo diferentes variables: Algunos con correas de velcro elásticas que solo abarcan la parte superior del cuerpo, específicamente la clavícula, el pecho y la espalda [7] . Los corset ajustables de espalda superior [8] los cuales incluyen varillas metálicas ubicadas entre los omoplatos, hasta la espalda media sin abarcar la zona lumbar.

Otros más completos como lo son los correctores con soporte de espalda largo, los cuales cuentan con varillas metálicas flexibles y correas que se envuelven sobre los

hombros y alrededor de la espalda para proporcionar un mayor apoyo. También existen los chalecos ortopédicos para corregir postura [10], los cuales integran la parte superior e inferior de la espalda, hombros, zona lumbar, cintura y abdomen, siendo dispositivos ergonómicamente diseñados con correas ajustables y unisex, para corregir la postura de manera genérica [9].

Los correctores de postura inteligentes con soporte de espalda monitorean la posición de esta mediante un sensor, el cual detecta en tiempo real la mala postura y activa una alarma de vibración cuando la espalda se dobla más de 25°, avisando inmediatamente al usuario que debe corregir su postura, para que su cintura, espalda y columna vertebral queden alineadas y desactivando la alarma hasta que la corrección se realice. Dicho dispositivo es unisex, cuenta con correas ajustables en cinturón de nylon elástico alrededor de los hombros y un soporte rígido en la parte superior de la espalda de aproximadamente 7cm donde se encuentra la sección electrónica [11].

## 2.2 Descripción y Formulación del Problema

El dolor de espalda es una de las patologías más comunes en la población, está relacionada con el teletrabajo, trabajos sedentarios o con actividades que exigen permanecer sentado [12]. Estas dolencias incrementan cuando no se mantiene una postura correcta, debido a que influye directamente en la actividad muscular del tronco [13], aumento de la presión intradiscal y en las fuerzas de compresión en las vértebras, [14] afectando la productividad y calidad de vida del individuo[15].

Según la Asociación Colombiana contra el Dolor en su octavo estudio nacional, se encontró que el 64.5% de las personas encuestadas han sufrido de dolores osteomusculares y de allí el 23.6% son dolores de espalda [16]. La solución que se le suele dar a las personas es un corrector de postura, sin embargo, este método no tiene en cuenta la complejidad de variables y articulaciones que intervienen en la correcta posición de la espalda, como lo son los hombros y la cadera [15].

Actualmente en el mercado existen dispositivos electrónicos y guías de cuidado de postura, que buscan la corrección postural y tratar dolencias de espalda. Pero estos dispositivos tienen algunos problemas, realizan una corrección, pero no se monitorea el proceso, solo consideran la columna vertebral e ignoran las demás articulaciones que intervienen en la posición de

sedestación, además, no tienen en cuenta prescripciones médicas, como problemas osteomusculares, inclinación de hombros, hombros adelantados, entre otras. Estos factores hacen que la corrección postural sea generalizada, y excluye situaciones específicas donde el ajuste de postura debe ser personalizable, ya que, es posible que mal usado puede generar atrofia en el músculo.

### 2.3 Justificación

Actualmente, la solución que se propone para evitar posturas incorrectas son los correctores posturales, donde, el modelo más común es el textil [17], el cual no permite generar conciencia cognitiva debido a la falta de recordatorios y visualizadores, además de llegar a ser incómodos y genéricos. Asimismo, existen correctores de postura inteligentes [18]–[19]–[20], los cuales cuentan con indicadores lumínicos y vibraciones que alertan al usuario de su mala postura, pero no tienen en cuenta articulaciones diferentes a la columna vertebral que también intervienen mientras la persona se encuentra sentada.

En la posición de sedestación el cuerpo realiza cambios posturales inconscientemente para encontrar un equilibrio de alivio muscular, donde las curvas fisiológicas normales de la columna se ven afectadas por la posición de la cadera y los hombros, generando sobrecargas en las estructuras osteomusculares intervenidas en dicha posición [21]. De allí nace la importancia de realizar un análisis del comportamiento de la cadera y hombros cuando una persona se encuentra sentada.

El desarrollo de este proyecto de grado busca el diseño de un sistema de monitoreo postural analizando hombros y cadera en personas que permanecen sentadas. Este se justifica, en primer lugar, porque en la detección de los movimientos de los hombros y cadera en posición de sedestación relacionándolos con los ángulos de inclinación en el eje x, y y z que capta el sensor inercial y los movimientos de rotación (yaw, roll y pitch) de una avión, realizando al mismo tiempo un estudio de las articulaciones que intervienen durante estos movimientos, para así conocer la ubicación exacta donde deben ir los sensores.

En segundo lugar, se justifica por la calibración de los sensores, utilizando un transportador como punto de referencia. Donde, se relacionan los datos medidos por los sensores con los datos reales del transportador mediante una línea de tendencia y su correspondiente ecuación lineal. En tercer lugar, por la aplicación móvil que facilita la interacción entre el dispositivo electrónico, el fisioterapeuta y el usuario, permitiendo que la información se lea y visualice de manera sencilla.

Finalmente, se justifica con el diseño de una base de datos en la nube, la cual, guarda la información personal y médica del usuario, para que el fisioterapeuta tenga acceso a lo que requiera y el usuario pueda validar como se encuentra su postura.

## 2.4 Objetivos

### 2.4.1 Objetivo General

Desarrollar un sistema electrónico para el monitoreo postural en tiempo real de la cadera y los hombros en personas que permanecen sentadas.

### 2.4.2 Objetivos Específicos

1. Diseñar un dispositivo electrónico para detectar la posición de hombros y cadera cuando una persona se encuentra sentada.
2. Implementar el dispositivo electrónico.
3. Desarrollar un software para dispositivos móviles, capaz de monitorear y almacenar los datos recolectados por la sensórica implementada.
4. Integrar el dispositivo electrónico y el software para tener un adecuado funcionamiento.
5. Evaluar el sistema postural desarrollado.

### 2.5 Alcance y Limitaciones del Proyecto

El producto final constará de un sistema electrónico para el monitoreo postural de hombros y cadera, incluyendo un dispositivo electrónico vestible realizador de la toma de mediciones y un software para dispositivos móviles diseñado para la visualización, almacenamiento y notificación de los datos recibidos.

El trabajo de grado propuesto cuenta con las siguientes limitaciones:

1. El diseño incluirá una estructura ósea grande.
2. El diseño no tendrá en cuenta una población menor de edad.
3. El diseño será adaptable para estructura ósea mediana.
4. La aplicación móvil solo será para sistemas operativos Android.
5. El diseño solo funcionará cuando la persona esté sentada.
6. El diseño se puede sobredimensionar en caso de no conseguir los componentes con las características necesarias al momento de la implementación.
7. El diseño no realiza la corrección postural, solo el monitoreo.

# 3. MARCO DE REFERENCIA

## 3.1 Marco Teórico o Conceptual

### 3.1.1 Anatomía de la cadera

La cadera es una de las articulaciones más estables y grandes del cuerpo, además de tener un amplio rango de movimiento que permite la locomoción normal en la ejecución de actividades diarias, como lo son: la bipedestación, sedestación y la marcha [22 -23].

La articulación de la cadera está compuesta por la cabeza del fémur y el acetábulo de la pelvis, los cuales son simétricos y cuentan con un espacio articular igual en todos los puntos, permitiendo la acción muscular y la rotación alrededor de un eje fijo [22 -23].

La pelvis está compuesta por tres elementos fundamentales: el sacro y dos iliacos o huesos coxales, como se muestra en la siguiente figura:

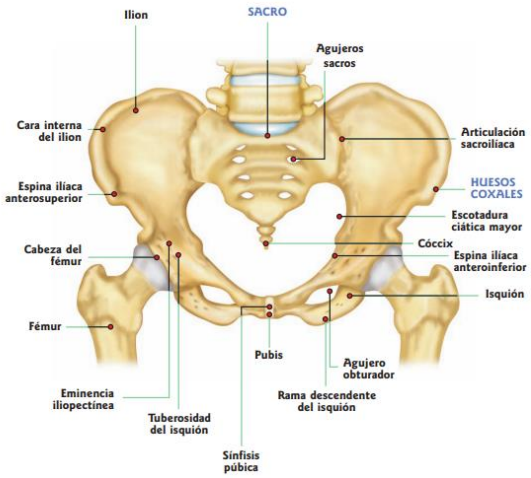



*Figura 1- Pelvis, vista anterior [25].*

Los iliacos se encuentran ubicados simétricamente respecto a la columna vertebral, cada uno está formado por tres huesos planos: ilion, isquion y pubis, donde la unión de estos huesos ocurre en el acetábulo.

Los dos huesos coxales o ilíacos se articulan en posición superior y posterior al sacro formando la articulación sacroilíaca, la cual tiene como función principal soportar el peso de la parte superior del cuerpo o del esqueleto axial y transferirlo a los huesos coxales. Por otra parte, los dos huesos coxales se articulan en posición inferior y anterior a la sínfisis púbica [24 -25 - 26].

El ilion está compuesto por dos caras (interna y externa) y cuatro bordes( superior, inferior, anterior y posterior), donde en el extremo superior se encuentra la fosa iliaca, cuya superficie auricular se articula con el sacro, además de estar rodeada por el borde superior donde se forma la cresta ilíaca. El isquion es la parte posterior e inferior del hueso ilíaco el cual se fusiona con el pubis formando así el agujero obturadorador y permitiendo cierto grado de movimiento gracias a la unión del pubis de la parte anterior del hueso coxal con el pubis del lado opuesto [24 -26].

El sacro es un hueso triangular rodeado de los músculos piramidales, ubicado entre los dos iliacos en la base de la columna vertebral. Está formado por la fusión de 5 vértebras sacras, además de estar ubicado en la parte posterior de la pelvis [24 - 26].

La estructura ósea de la pelvis forma un anillo pélvico compuesto por tres articulaciones; dos sacroiliacas y la sínfisis del pubis que trabajan sincrónicamente junto a las articulaciones coxofemorales y las de la columna lumbar, donde en conjunto tienen la forma de una cuenca que soporta el peso de la parte superior del cuerpo, trabajando como un elemento de transmisión de presiones y contrapresiones, donde el peso corporal recae sobre la quinta vértebra lumbar, transmitiendolo al sacro, posteriormente el peso pasa a través de las articulaciones sacroilíacas a los iliones en los cuales se neutraliza la fuerza ejercida sobre los fémures, produciendo una compresión en las tuberosidades isquiáticas en la posición de sedestación y en la bipedestación produce fuerzas de compresión en los acetábulos [22 - 24 -26].

La forma y las proporciones de la pelvis varían de una persona a otra, donde; el estrecho superior puede tener una forma: redondeada, sagital o alargada, a la

concavidad del sacro puede estar más o menos acentuada, la separación entre dos tuberosidades isquiáticas puede ser mayor o menor y donde el género influye; la pelvis del hombre es más estrecha a comparación de la pelvis de la mujer, lo cual está ampliamente relacionado con el papel que desempeña esta región en el cuerpo del hombre y la mujer [24].

### 3.1.2 Biomecánica de la cadera

La amplitud de los movimientos de la cadera tiene lugar en los tres planos: sagital, frontal y transverso; los movimientos de flexion y extension, abduccion y aduccion y rotacion pertenecen respectivamente a cada uno de los planos, los cuales se generan cuando el ilíaco se encuentra fijo y el fémur se desplaza en relación a él [24]. La siguiente figura permite identificar cada uno de los movimientos mencionados anteriormente:

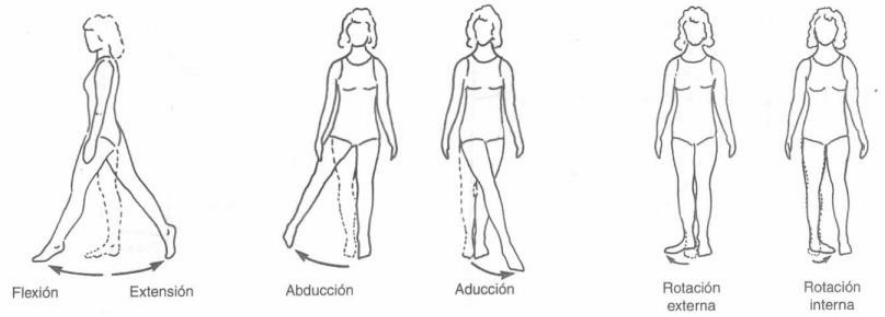



*Figura 2. Movimiento de flexión y extensión, abducción y aducción y rotación de la cadera [ 23]*

En el plano sagital se presenta el movimiento de flexión, el cual consiste en acercar las caras anteriores del muslo y del tronco, buscando reducir el ángulo entre los huesos de la articulación. La amplitud de la flexión es mayor entre más doblada se encuentre la rodilla y menor en cuanto más extendida se encuentre, esto, debido a la tensión producida en los músculos isquio- tibilaes. El rango de flexión oscila dentro de los 0° a 140° y generalmente conlleva una retroversión de la pelvis [23 - 24 -27].

Por otra parte, se puede observar el movimiento de extensión, en el cual, se acercan las caras posteriores del muslo y del tronco. La amplitud de la extensión es mayor entre más extendida se encuentre la rodilla y menor en cuanto más flexionada se encuentre, esto, debido a la tensión producida en el músculo recto anterior. El rango

de extensión oscila dentro de los 0° a 15° y generalmente conlleva una anteversión de la pelvis [23 - 24 -27].

En el plano frontal se presentan los movimiento de abducción y aducción de la cadera, donde, como se observa en la figura (x), el movimiento de abducción consiste en acercar las caras externas del muslo y del tronco, generando que el hueso se aleje de la línea media del cuerpo. El rango de abducción oscila dentro de los 0° a 30°, ya que, la parte superior del cuello topa con el techo del cotilo [23 - 24 - 27].

En la figura 2 se observa el movimiento de aducción, el cual se genera cuando el muslo se desplaza hacia adentro, generando que el hueso se acerque a la línea media del cuerpo. El rango de aducción va de 0° a 25° [23 - 24 - 27].

En el plano transverso se presenta la rotación externa y la rotación interna, donde lo que se produce es el movimiento del fémur sobre su eje longitudinal. En la rotación interna el pie se dirige hacia dentro en un rango de 0° a 70°, mientras que en la rotación externa el pie se dirige hacia afuera en un rango de 0° a 90°. La amplitud de la rotación es mayor entre más flexionada se encuentre la cadera y menor en cuanto más extendida se encuentre, esto, debido a la limitación en la función de los tejidos blandos [23 - 24 - 27].

Por otra parte, se presentan movimientos como lo son la anteversión, retroversión, inclinación lateral externa e inclinación lateral interna, rotación interna y externa, los cuales, se generan cuando el fémur se encuentra fijo y el ilíaco se desplaza sobre él [24].

El movimiento de anteversión de la cadera consiste en desplazar hacia delante la espina iliaca antero- superior generando que la curvatura de la región lumbar se acentúe produciendo una lordosis, como se observa en la siguiente figura [24].

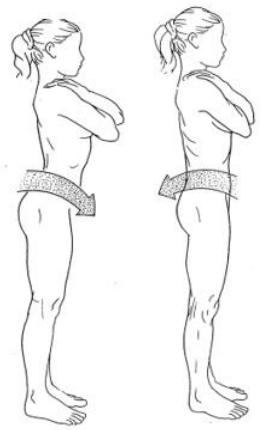


*Figura 3 Anteversión y retroversión de la cadera [24].*

Por otra parte, el movimiento de anteversión de la cadera consiste en desplazar hacia atrás la espina iliaca antero- superior generando la disminución en la curvatura de la región lumbar, como se observa en la figura 3 [24].

La exageración de la anteversión o de la retroversión generan inconvenientes en la columna lumbar, debido a que no se genera una repartición de cargas, lo que produce compresión en la región lumbar.

Los movimientos de Inclinación lateral externa e inclinación lateral interna consisten en desplazar hacia fuera la espina iliaca antero- superior y desplazar hacia dentro la espina iliaca antero- superior, como se observa en la figura 4. Cuando estos movimientos son efectuados de pie, suelen provocar una inclinación lateral en sentido inverso de la zona lumbar[24].

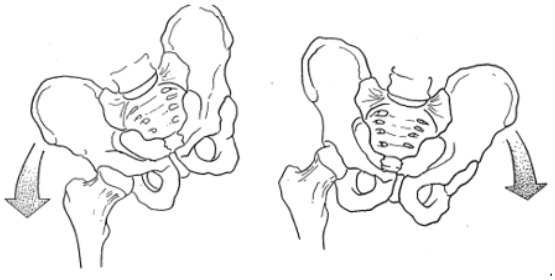


*Figura 4 .Inclinación lateral externa e inclinación lateral interna [24].*

Los movimientos de rotación interna y rotación externa consisten en rotar internamente el hueso ilíaco y externamente respectivamente, como se observa en la figura 5:

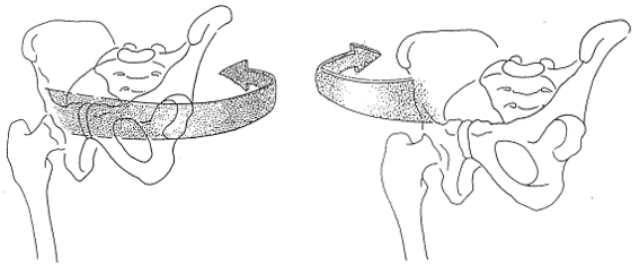


*Figura 5. Rotación interna y rotación externa del ilíaco [24].*

### 3.1.3 Anatomía de los hombros

El hombro es un conjunto anatómico y funcional que permite unir el miembro superior con el tórax. Está formado por las clavículas por delante, el omoplato o escápula por detrás y el esternón delante y en medio [25 - 26].

La clavícula es un hueso largo ubicado entre el omóplato y el esternón, además de articularse con ellos, se encuentra ubicada a ambos lados de la columna vertebral formando la parte superior de los hombros [25].

El omóplato es un hueso plano y cóncavo con forma triangular ubicado en la parte superior y posterior del tórax, con: dos caras (anterior y posterior), tres bordes y tres ángulos. Se articula con la caja torácica convexa y con el húmero por medio de una superficie cóncava: la articulación glenohumeral, capaz de soportar la extremidad superior, como se observa en la siguiente imagen [22- 24 - 25 ].

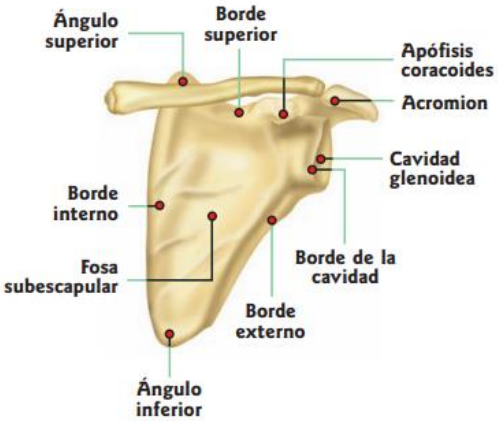



*Figura 6. Omóplato o escápula [25].*

La articulación glenohumeral está compuesta por una superficie redondeada en la parte proximal del húmero formando un arco aproximado de 120°, denominada bola o cabeza humeral y una superficie cóncava, ubicada en la zona cuenca de la escápula, denominada glenoide o cavidad glenoidea, la cual se encuentra rodeada de cartílago, incrementado así la profundidad de la cavidad y permitiendo que el brazo realice movimientos en su mayoría rotacionales debido a la escasez ósea, ya que, la estabilidad que se le brinda a la articulación es gracias a los ligamentos, músculos y estructuras capsulares la rodean [23].

Adicional a la articulación glenohumeral, el hombro cuenta con tres articulaciones: acromioclavicular, escapulotorácica y esternoclavicular, además de estructuras musculares que actúan sobre ellas para generar los respectivos movimientos del hombro [23].

La articulación acromioclavicular está formada por la clavícula y la escápula, permitiendo la unión ósea entre la escápula y el esqueleto apendicular y axial, teniendo como función principal limitar la rotación axial y la traslación posterior de la clavícula [23].

La articulación escapulotorácica es la región del cuerpo donde se encuentra la escápula y la pared torácica, permitiendo los movimientos de abducción, aducción, elevación, depresión, rotación interna y externa [24].

La articulación esternoclavicular se encarga de unir la extremidad superior con el tórax, específicamente el esternón con la clavícula, permitiendo que se puedan deslizar uno en relación al otro y coordinando los movimientos de la extremidad superior respecto al tronco. Dicha articulación contiene un disco artificial fibrocartilaginoso que limita el desplazamiento de la clavícula, ligamentos posterior y anterior, los cuales generan movimientos de traslación además del desplazamiento superior, un ligamento interclavicular el cual limita la articulación superiormente y

un ligamento costoclavicular limitador de movimiento de la articulación esternoclavicular [23].

### 3.1.4 Biomecánica de los hombros

Los movimientos del hombro pueden ser de dos tipos, ya que, se emplean regiones funcionales diferentes:

La primera región hace referencia a los movimientos del hombro en la región escapular, específicamente en la articulación escapulotorácica . Donde se presenta la elevación, que consiste en subir el hombro o una porción de la parte superior del cuerpo hacia arriba en el plano frontal, el descenso, que consiste en bajar el hombro o una porción de la parte superior del cuerpo hacia abajo respecto a su posición inicial, la abducción en la cual se separa el hombro de la columna vertebral produciendo a la vez un pequeño movimiento hacia adelante como se observa en la siguiente figura [24 - 27].

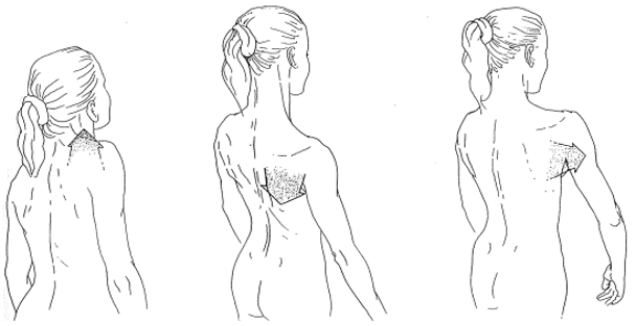


*Figura 7 Elevación, descenso y abducción del hombro respecto al omóplato [24].*

También se presente el movimiento de aducción que consiste en acercar el hombro hacia la columna vertebral, generando un movimiento hacia atrás en el plano transverso, el campaneo interno que consiste en generar un movimiento de péndulo en el hombro con la punta de la escápula hacia adentro y finalmente el campaneo externo que consiste en generar un movimiento de péndulo en el hombro con la punta de la escápula hacia afuera, como se observa en la siguiente figura [24 - 27].

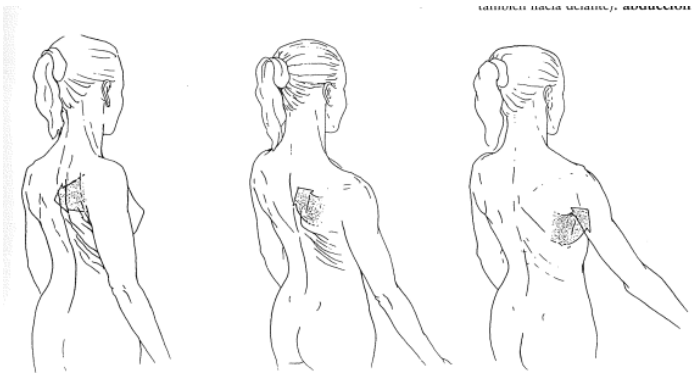


*Figura 8. Aducción, campaneo interno y campaneo externo del hombro respecto al omóplato [24].*

La segunda región hace referencia a los movimientos del brazo con respecto al omoplato. En el cual se presenta la antepulsión que consiste en mover el brazo hacia delante y la retropulsión que consiste en mover el brazo hacia atrás, como se observa en la siguiente figura [24].

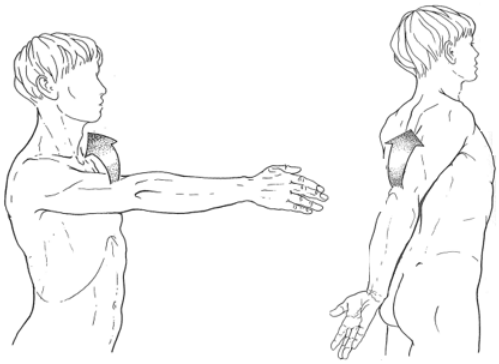


*Figura 9. Antepulsión y retropulsión del brazo respecto al omóplato [24].*

También se presenta la abducción que consiste en mover el brazo hacia afuera y la aducción que consiste en mover el brazo hacia adentro, como se observa en la siguiente figura [24].

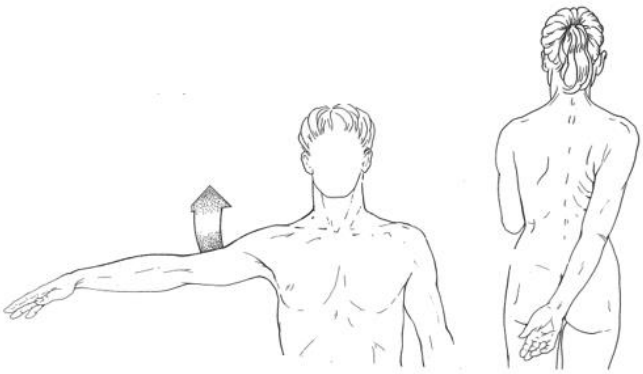


*Figura 10. Abducción y aducción del brazo respecto al omóplato [24]*

La rotación interna y la rotación externa consisten en rotar hacia adentro o hacia fuera el brazo con el codo flexionado, como se observa en la siguiente figura [24].

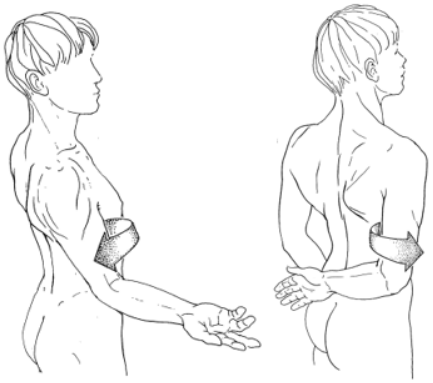


*Figura 11Rotación interna y externa del brazo respecto al omóplato [24].*

Cuando los movimientos del hombro alcanzan amplitudes mayores intervienen articulaciones como lo son el tórax y la columna vertebral, donde, en el movimiento de antepulsión se puede alcanzar una amplitud mayor al producir una extensión vertebral y una abertura torácica, mientras que en el movimiento de retropulsión se puede alcanzar una amplitud mayor al producir una flexión dorsal y un cierre del tórax, como se observa en la siguiente figura [24].

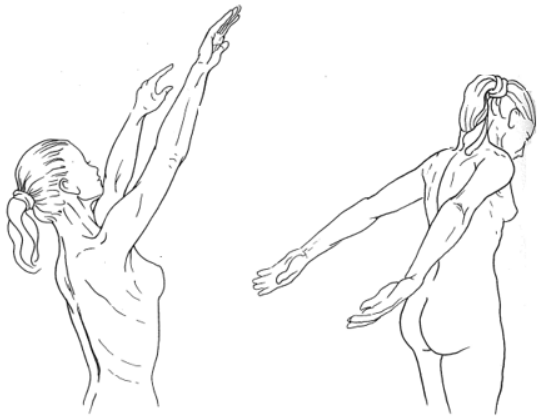


*Figura 12Antepulsión y retropulsión del hombro a mayor amplitud [24].*

En la abducción se puede alcanzar una amplitud mayor al producir una inclinación lateral dorsal hacia el lado opuesto y una abertura del hemitorax del lado donde se efectúa el movimiento, mientras que en la aducción se puede alcanzar una amplitud mayor al producir una inclinación lateral dorsal hacia el lado en que se realiza el

movimiento y un cierre de hemitorax del mismo lado, como se observa en la siguiente figura [24].

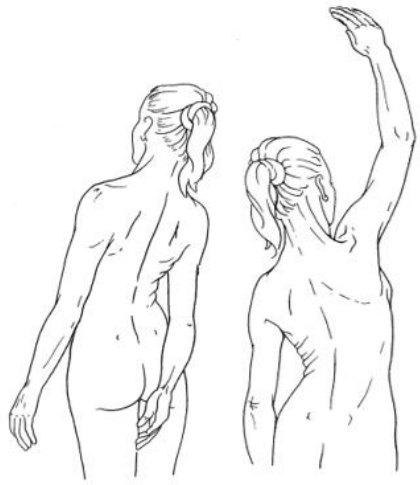


*Figura 13Abducción y aducción del hombro a mayor amplitud [24].*

En la rotación interna y la rotación externa se alcanza una mayor amplitud cuando rota la columna dorsal hacia adentro o hacia fuera en relación al omóplato, como se observa en la siguiente figura [24].

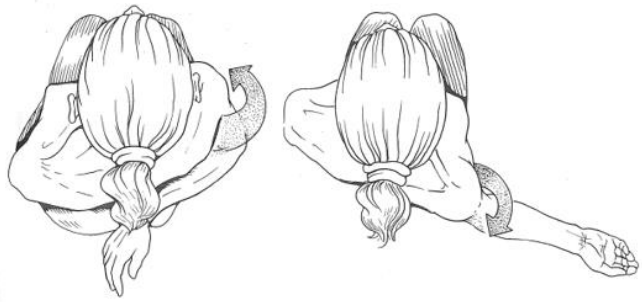


*Figura 14Rotación interna y externa del hombro a mayor amplitud [24].*

### 3.1.5 Sedestación

La sedestación es la posición básica en la que el ser humano permanece sentado, manteniendo una postura ergonómica enfocada en la verticalidad del tronco y la alineación de la columna vertebral . Dicha posición es más estable que la bipedestación e incluso la marcha, por ende, implica menor fatiga y gasto energético, aún así, esta posición es antinatural y produce una basculación y modificación de la cadera y lla columna vertebral, generando que la pelvis adquiera una postura desviada

de su plano y a su vez alteraciones en la columna vertebral debido a la compensación muscular [31].

La primera norma para una correcta sedestación es la capacidad de cambiar la postura regularmente, ya que, permanecer estáticos con la misma postura puede producir malestar articular y muscular en diferentes extremidades del cuerpo. De igual manera, mantener una buena postura ergonómica durante el tiempo que se permanezca sentado genera un esfuerzo muscular mínimo y un adecuado soporte [23- 31-32].

La zona más afectada es la lumbar, donde, diferentes estudios de laboratorio [32] han encontrado un aumento de la incomodidad de esta área al pasar mucho tiempo en posición de sedestación, esta incomodidad puede surgir por la activación y la carga sostenida de bajo nivel de los tejidos pasivos o por el aplanamiento de la curva lordótica lumbar y el desacondicionamiento muscular, donde, la traslación del tronco hacia delante incrementa la lordosis lumbar, mientras que una transición hacia atrás mueve el centro de gravedad por encima de las tuberosidades isquiáticas hacia una posición inestable [23 - 32].

Las extremidades inferiores corresponden a la segunda zona con más afectación. Según estudios [32], la cadera, el muslo y las nalgas son las áreas con mayor malestar y está ampliamente relacionado con la presión que se ejerce sobre éstas, específicamente sobre la región isquiática, esto, debido a que el cuerpo busca una articulación de apoyo para la estabilidad del mismo, donde la pelvis cumple esta función y genera activación muscular, concentrando las presiones de apoyo en zonas como el fémur, los isquiones y el sacro, provocando una retroversión de la pelvis y una cifosis lumbar [23 -31 -32].

Los estudios de laboratorio también han encontrado un aumento en el malestar de cuello y hombros, asociado con estar sentado durante períodos prolongados, encontrado, que la incomodidad es mayor al sentarse que realizando otros

movimientos como marcha o bipedestación, esto, debido a que existe una mayor carga en los músculos del cuello y los hombros al sentarse [32].

### 3.1.6 Movimientos de rotación Pitch, Yaw y Roll

Los ángulos de Yaw, Roll y Pitch se utilizan para describir el movimiento de rotación de un barco o aeronave, pero también permiten identificar la orientación de un dispositivo robótico, siguiendo una secuencia de rotaciones relativas a lo largo de los tres ejes espaciales (x,y y z) [28].

Para entender las rotaciones que se realizan sobre los ejes espaciales se imaginan tres líneas atravesando un avión y cruzándose en ángulo recto por el centro de gravedad del mismo, representado así los ejes x,y y z .

El ángulo de Roll o “balanceo” se entiende como la rotación que se realiza alrededor del eje longitudinal. Para comprender esta rotación se genera una línea imaginaria desde la nariz hasta la cola del avión, donde, al girar el avión sobre este eje, produce que mientras un ala ascienda la otra descienda, representando así el movimientos de “rollo” de un avión, como se observa en la figura x [29].

El ángulo de Pitch o “ cabeceo” se entiende como la rotación que se realiza alrededor del eje lateral.. Para comprender esta rotación se genera una línea imaginaria que va de punta a punta de ala, donde, al girar el avión sobre este eje, produce que la nariz del avión se incline hacia arriba o hacia abajo, representando así el ascenso o descenso de un avión, como se observa en la figura x [29].

El ángulo Yaw o “giñada” se entiende como la rotación que se realiza alrededor del eje vertical. Para comprender esta rotación se genera una línea imaginaria que atraviesa el centro del avión desde la parte superior hasta la inferior, donde, al girar el avión sobre este eje, produce que el avión se desvíe a lo largo del eje vertical y la nariz del avión realice un movimiento de derecha a izquierda, así que, mientras la

nariz apunta a la izquierda, la cola del avión se desviará hacia la derecha y viceversa, como se observa en la figura x [29].

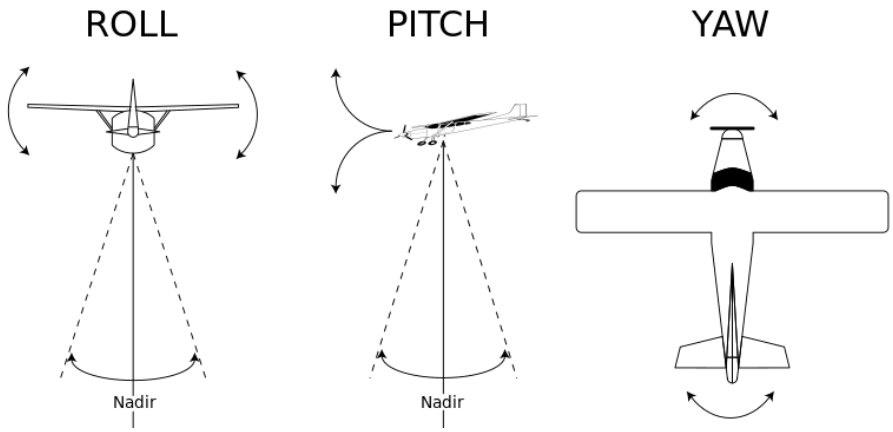



*Figura 15Roll, Pitch y Yaw [30].*

# 4. DESARROLLO DEL PROYECTO DE GRADO

## 4.1 Requerimientos

### 4.1.1 Requerimientos funcionales

1. El dispositivo electrónico medirá el ángulo de inclinación de cada hombro de manera independiente en los ejes x,y y z, teniendo en cuenta los movimientos de: elevación, descenso, abducción, aducción, rotación externa e interna del hombro a mayor amplitud, en un rango de: 0° a 180°.
2. El dispositivo electrónico medirá el ángulo de inclinación de la cadera en los ejes x,y y z teniendo en cuenta los movimientos de anteversión, retroversión, inclinación lateral externa, inclinación lateral interna, rotación interna y rotación externa, en un rango respectivamente de 0° a 90°.
3. El dispositivo electrónico medirá la simetría entre los omoplatos.
4. El dispositivo electrónico medirá la simetría entre los iliacos.
5. En la aplicación móvil se visualizará la postura del usuario.
6. La aplicación móvil procesará y analizará los valores de ángulo y simetría recolectados por el dispositivo electrónico y por el médico especialista, para identificar la correcta postura del usuario.
7. La aplicación móvil almacenará la información recolectada por medio de la implementación de una base de datos.
8. La aplicación móvil enviará notificaciones de alerta de mala postura (los valores de mala postura varían según el usuario).
9. El dispositivo tendrá un sistema de detección de fallas, el cual alertará al usuario de posibles desconexiones en el circuito, batería baja y mala posición de sensores.

### 4.1.2 Requerimientos de Calidad

1. Los valores de ángulo recolectados por el dispositivo electrónico tendrán un error máximo del ±5°.

2. El arnés será adaptable para una estructura ósea mediana masculina y femenina con las siguientes dimensiones: 93cm a 103cm pecho, 100cm a 110 cadera y 56 cm a 66 cm altura de hombros a cadera.
3. El sistema electrónico deberá incluir manual de usuario, donde se explique la manera correcta de usar el dispositivo y la aplicación, este se obtendrá a partir de la generación de protocolos de uso, conexión e indicadores de fallas, a partir de pruebas del prototipo en voluntarios reales; Estos voluntarios deben tener edades entre los 18 y 50 años con estructuras osteomusculares medianas.
4. Los sensores de distancia tendrán una dimensión máxima de 2cm x 8cm.
5. Los sensores de ángulo tendrán una dimensión máxima de 3 cm x 3 cm.

### 4.1.3 Requerimientos restrictivos

1. El dispositivo electrónico deberá tener un soporte de batería.
2. El dispositivo deberá ser alimentado por una batería recargable cuya capacidad sea suficiente para mantener un tiempo de trabajo de mínimo 5 horas.
3. El dispositivo deberá tener un microcontrolador con capacidad mínima para procesar simultáneamente información de cuatro sensores y contar con tecnología FPC.
4. El protocolo de comunicación del dispositivo deberá ser compatible con Android.
5. El dispositivo electrónico será diseñado para una estructura ósea mediana.
6. El dispositivo electrónico estará diseñado con tecnología FPC (flexible Printed Circuit).
7. El sistema electrónico diseñado será un prototipo.

## 4.2 Metodología del Diseño

### 4.2.1 Metodología desarrollada

La metodología implementada para este proyecto corresponde a una metodología CDIO (Conceptualización, Diseño, Implementación y Operación). Donde,

inicialmente se realiza toda la investigación relacionada a la postura durante la posición de sedestación, desde proyectos realizados, dispositivos en el mercado, hasta investigaciones específicas de las articulaciones y movimientos que realizan los hombros y cadera y que articulaciones se involucran en cada movimiento.

La siguiente etapa corresponde a la de diseño, donde se seleccionan los componentes para la creación del dispositivo, al igual que el software y la base de datos para el desarrollo de la aplicación móvil. En esta etapa se identifican los movimientos y las articulaciones que se deben analizar, ya que, el monitoreo postural que se realiza es específicamente en la posición de sedestación. Se realiza la calibración de los sensores a implementar, el diseño del circuito electrónico, el diseño de la prenda de vestir (camiseta) donde se ubica la sensórica para la toma de ángulos de inclinación de los hombros y cadera. Y finalmente el diseño de la aplicación móvil y la base de datos.

En la etapa de implementación,se realiza el análisis y los resultados obtenidos en el desarrollo del proyecto, implementando tanto el diseño del dispositivo, como el de la aplicación móvil y la base de datos, para probar individualmente el funcionamiento.

La última etapa corresponde a la operación donde se realiza la unión entre el dispositivo, la base de datos y la aplicación móvil, para validar en conjunto el funcionamiento del sistema de monitoreo por medio de pruebas y validaciones. Adicionalmente se realiza el manual de usuario para un correcto uso del sistema.

### 4.2.2 Planeación de actividades

Se plantean las siguientes actividades para el desarrollo del proyecto:

1. Revisión literaria de tecnologías y dispositivos para la corrección postural.
2. Revisión literaria de la anatomía y biomecánica de los hombros y la cadera.
3. Revisión literaria de la sedestación y los movimientos de: roll,yaw y pitch.
4. Identificación de los movimientos que realizan los hombros y cadera en la posición de sedestación.

5. Selección de componentes y softwares para el diseño del dispositivo.
6. Calibración de sensores inerciales.
7. Diseño dispositivo electrónico.
8. Diseño de la prenda donde irá ubicada la sensorica.
9. Diseño aplicación móvil y base de datos.
10. Implementación de dispositivo electrónico.
11. Pruebas funcionales del dispositivo electrónico.
12. Implementación de aplicación móvil y base de datos.
13. Integración aplicación móvil y base de datos.
14. Pruebas funcionales entre aplicación móvil y base de datos.
15. Integración de software y dispositivo electrónico.
16. Pruebas funcionales del sistema de monitoreo.
17. Modificaciones y ajustes.

### 4.2.3 Recursos

#### 4.2.3.1 Recursos Software

1. Arduino: Este recurso se implementa para el desarrollo del código de programación del dispositivo electrónico, permitiendo programar los sensores inerciales por medio de la tarjeta ESP32 devkit v1.
2. Android Studio: Este recurso se implementa para el desarrollo del código de programación de la aplicación móvil. Donde, para el frontend se utiliza HTML y para el backend Kotlin.
3. Firebase: Este recursos se implementa para el desarrollo de la base de datos y la autenticación del usuario a la aplicación móvil.

#### 4.2.3.2 Recursos fisicos

A partir de la selección de componentes, los recursos adquiridos para la implementación física del proyecto son los siguientes:

*Tabla 1Recursos Fisicos*

| **Elemento** | **Cantidad** | **Costo Unidad** | **Costo Total** |
|---|---|---|---|
| Computador | 1 | X | X |
| Batería 3,7V - 2000mAh | 1 | $12.000 | $12.000 |
| Sensor Inercial MPU6050 | 4 | $14.500 | $58.000 |
| Multiplexor 1 a 8 | 1 | $41.000 | $41.000 |
| TP4056 (Modulo cargador de baterías) | 1 | $5.000 | $5.000 |
| XL6009 (elevador de tensión) | 1 | $9.000 | $9.000 |
| ESP32 (microcontrolador) | 1 | $72.000 | $72.000 |
| Carcasa (impresión 3D) | 1 | $100.000 | $100.000 |
| Cable bus de datos (m) | 2 | $2.500 | $5.000 |
| Jumpers Hembra-Hembra | 1 | $8.000 | $8.000 |
| Correa textil (m) | 4 | $2.500 | $10.000 |
| Camiseta | 1 | $70.000 | $70.000 |
| **Total** | | | **$ 390.000** |

### 4.2.4 Plan de Pruebas

A partir de la planeación de las actividades y las diferentes etapas que se desarrollan en el proyecto se plantea el siguiente plan de pruebas:

1. Calibración y caracterización del sensor inercial.
2. Conexión I2C con cuatro sensores inerciales y tarjeta de programación ESP 32.
3. Lectura de sensores inerciales ubicados en las 4 articulaciones: omoplato derecho e izquierdo, cresta ilíaca derecha e izquierda.
4. Conexión aplicación móvil y base de datos.
5. Conexión aplicación móvil y dispositivo electrónico vía bluetooth.
6. Integración de dispositivo electrónico, aplicación móvil y base de datos.

### 4.2.5 Diseño conceptual

A partir de la investigación realizada en los antecedentes e historia del arte, se establece que los parámetros principales para el desarrollo del proyecto corresponden

a: ángulos de inclinación en los ejes x,y, y z de los hombros y la cadera. A continuación se presenta el diseño conceptual del dispositivo:

#### 4.2.5.1 SuperSistema

*Figura 16. Diagrama de caja negra, entradas y salidas del sistema [Propiedad de los autores].*

**Entradas:**

El dispositivo está compuesto por un total de 24 entradas para su correcto funcionamiento, estas son adquiridas de dos diferentes maneras; Las primeras 12 por medio de la sensórica a implementar y las restantes a través de mediciones manuales tomadas por un fisioterapeuta especializado. Esto con el fin de tener un valor de referencia acorde a la morfología del usuario.

Las entradas que se obtienen por medio de la sensórica y el especialista son las mismas, pero estas son procesadas y almacenadas por independiente, a continuación, se expresan cada una de ellas:

- **Sensórica:**

  **ax:** Ángulo de inclinación del omóplato derecho en el eje longitudinal; 0° a 90°; ±5°

  **ay:** Ángulo de inclinación del omóplato derecho en el eje lateral; 0° a 180°; ±5°

**az:** Ángulo de inclinación del omóplato derecho en el eje vertical; 0° a 45°; ±5°

**bx:** Ángulo de inclinación del omóplato izquierdo en el eje longitudinal; 0° a 90°; ±5°

**by:** Ángulo de inclinación del omóplato izquierdo en el eje lateral; 0° a 180°; ±5°

**bz:** Ángulo de inclinación del omóplato izquierdo en el eje vertical; 0° a 45°; ±5°

**cx:** Ángulo de inclinación de la cresta iliaca derecha en el eje longitudinal; 0° a 45°; ±5°

**cy:** Ángulo de inclinación de la cresta iliaca derecha en el eje lateral; 0° a 90°; ±5°

**cz:** Ángulo de inclinación de la cresta iliaca derecha en el eje vertical; 0° a 45°; ±5°

**dx:** Ángulo de inclinación de la cresta iliaca izquierda en el eje longitudinal; 0° a 45°; ±5°

**dy:** Ángulo de inclinación de la cresta iliaca izquierda en el eje lateral; 0° a 90°; ±5°

**dz:** Ángulo de inclinación de la cresta iliaca izquierda en el eje vertical; 0° a 45°; ±5°

- **Especialista:**

**Ax:** Ángulo de inclinación del omóplato derecho en el eje longitudinal; 0° a 90°; ±5°. Valor dado por especialista del área de la salud

**Ay:** Ángulo de inclinación del omóplato derecho en el eje lateral; 0° a 180°; ±5°. Valor dado por especialista del área de la salud

**Az:** Ángulo de inclinación del omóplato derecho en el eje vertical; 0° a 45°; ±5°. Valor dado por especialista del área de la salud

**Bx:** Ángulo de inclinación del omóplato izquierdo en el eje longitudinal; 0° a 90°; ±5°. Valor dado por especialista del área de la salud

**By:** Ángulo de inclinación del omóplato izquierdo en el eje lateral; 0° a 180°; ±5°. Valor dado por especialista del área de la salud

**Bz:** Ángulo de inclinación del omóplato izquierdo en el eje vertical; 0° a 45°; ±5°. Valor dado por especialista del área de la salud

**Cx:** Ángulo de inclinación de la cresta ilíaca derecha en el eje longitudinal; 0° a 45°; ±5°. Valor dado por especialista del área de la salud

**Cy:** Ángulo de inclinación de la cresta ilíaca derecha en el eje lateral; 0° a 90°; ±5°. Valor dado por especialista del área de la salud

**Cz:** Ángulo de inclinación de la cresta ilíaca derecha en el eje vertical; 0° a 45°; ±5°. Valor dado por especialista del área de la salud

**Dx:** Ángulo de inclinación de la cresta ilíaca izquierda en el eje longitudinal; 0° a 45°; ±5°. Valor dado por especialista del área de la salud

**Dy:** Ángulo de inclinación de la cresta ilíaca izquierda en el eje lateral; 0° a 90°; ±5°. Valor dado por especialista del área de la salud

**Dz:** Ángulo de inclinación de la cresta ilíaca izquierda en el eje vertical; 0° a 45°; ±5°. Valor dado por especialista del área de la salud

Los ángulos de inclinación en los ejes x, y y z se miden con el fin de identificar el movimiento que realiza la articulación y de esta manera la posición en la que se encuentra, permitiendo reconocer una correcta o incorrecta postura de acuerdo a los valores medidos por el especialista.

**Bloque central:**

El bloque central se encarga de analizar y procesar la información obtenida en cada una de las entradas a través de diferentes métodos y elementos electrónicos que intervienen en el proceso para realizar un correcto monitoreo y visualización de la postura del usuario.

Por una parte el dispositivo electrónico capta y procesa los datos de inclinación de las articulaciones cadera y hombros, donde posteriormente se almacenan y analizan en una base de datos junto a los valores de inclinación remitidos por un médico especialista (datos que deben ser ingresados manualmente).

**Salida:**

Visualización gráfica en aplicación móvil, utilizando un diagrama de la parte posterior del torso donde se evidencian los hombros y cadera para poder observar la posición real del usuario y la posición de referencia dada por el especialista. Si los valores de inclinación captados por el dispositivo electrónico difieren con los valores de inclinación remitidos por el especialista se emitirá un mensaje informando que la articulación se encuentra en una posición incorrecta.

**Excepciones del Super Sistema:**

El sistema no es responsable de un mal monitoreo de postura corporal si se presenta:

- Sudor corporal.
- Accesorios o prendas que afecten la ubicación de los sensores.
- Mala postura del arnés (prenda donde irá el dispositivo electrónico).
- Mal uso (que el usuario utilice el dispositivo electrónico en posiciones diferentes a la sedestación).
- Mala posición de las piernas en la posición de sedastación (piernas cruzadas)

4.2.5.2 Diagrama detallado

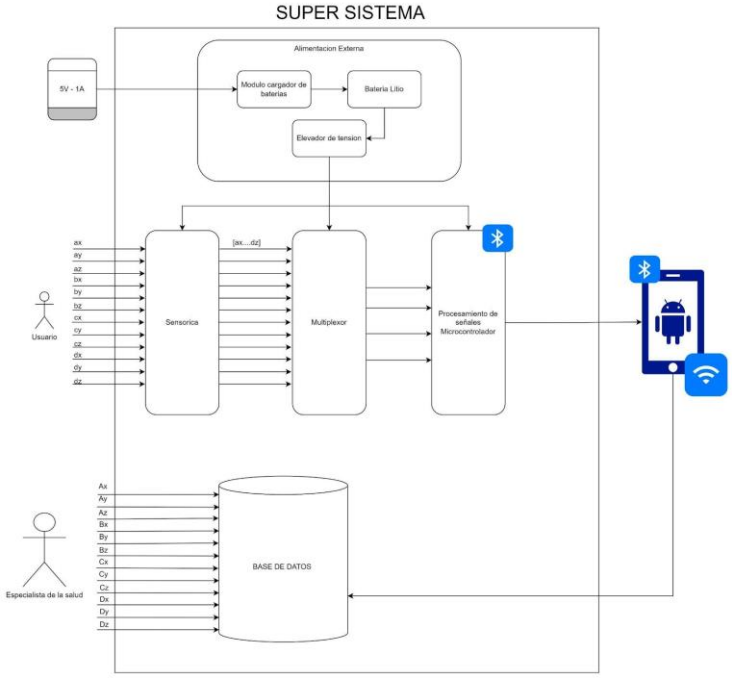



*Figura 17. Diagrama detallado de entradas y salidas del sistema [Propiedad de los autores].*

En este diagrama se visualiza detalladamente las entradas y salidas del sistema, teniendo en cuenta las diferentes etapas que se necesitaron para desarrollar el proyecto.

Se observa que para el super sistema diseñado es necesario 25 entradas, 12 de ellas son las posiciones angulares de las articulaciones (hombros y cadera) del usuario en los ejes x, y ,z; otras 12 son los valores físicos medidos por un especialista de la salud y el último es la alimentación hacia el dispositivo.

Las entradas (ax,ay,az … , dz) del usuario son captadas por la sensórica requerida, para conectar varios sensores a un solo controlador, se debe usar un multiplexor que envíe los datos hacia el microcontrolador sin tener pérdidas de información, ni lentitud en el envio; alli desde el microcontrolador se hace el envío de toda la información recopilada hacia un dispositivo móvil por medio de bluetooth, el cual contiene una aplicación móvil, donde se visualizan todos los datos obtenidos por la sensorica, adicional a ello, la aplicación móvil está conectada a una base de datos que alberga las mediciones físicas que un especialista tomó directamente al usuario.

Para la alimentación del dispositivo se requiere un batería de litio, la cual permitirá al dispositivo ser portátil y fácil de usar; para el uso de esta batería se debe implementar un circuito de carga de baterías de litio y un elevador a 5 voltios para mantener una tensión constante en todo el dispositivo.

Toda la información recopilada por la sensórica y el especialista de la salud, es mostrada en una aplicación móvil intuitiva y amigable con el usuario.

### 4.3 Descripción Técnica del Producto

El dispositivo consta de un sistema electrónico para el monitoreo postural de personas que permanecen mucho tiempo sentadas, analizando dos estructuras principales del cuerpo humano: hombros y cadera.

Donde, por medio de la implementación física del sistema, el usuario puede realizar la corrección de la postura cuando se encuentra sentado. De tal manera, que el sistema brinda información en tiempo real de los hombros y cadera para que con ayuda de esta el usuario monitoree su postura.

Para el monitoreo se desarrolla un software interactivo y de almacenamiento de datos, el cual permite configurar datos personales y médicos (ángulos máximos de inclinación de las articulaciones hombros y cadera) teniendo en cuenta patologías diagnosticadas y prescripciones médicas, para de esta manera realizar un monitoreo postural personalizado. La aplicación móvil permite la visualización, registro y modificación de información, además de notificaciones indicando la postura del usuario.

El dispositivo cuenta con tecnología vestible, específicamente una camiseta, la cual permite ajustar la posición de la sensórica, en relación a la posición y ubicación de las articulaciones hombros y cadera, generando que sea adaptable a la estructura ósea del usuario. Este dispositivo busca ser mínimamente invasivo e imperceptible por parte del usuario para no generar estrés, estado de alerta ni dependencia al usarlo.

## 4.4 Identificación de movimientos que se realizan y articulaciones que intervienen en la posición de sedestación

La identificación de los movimientos que se realizan y las articulaciones que intervienen cuando una persona se sienta es la base principal para el desarrollo del proyecto, ya que, esto permite identificar la ubicación de los sensores y los movimientos que se deben analizar para validar la postura del usuario.

### 4.4.1 Identificación teórica de movimientos

Los libros, artículos y estudios sobre la posición de sedestación no brindan información específica para analizar y entender todas las articulaciones y movimientos que se involucran cuando una persona se sienta. La mayoría de la literatura identifica a la columna vertebral como parte fundamental de dicha posición, pero ignora articulaciones muy importantes como lo son los hombros, la cadera e incluso el cuello y las piernas (partes del cuerpo que no se analizan en este proyecto).

Los hombros y la cadera son partes fundamentales de un análisis postural para la posición de sedestación Un ejemplo sencillo de la importancia de estas articulaciones se observa cuando una persona se sienta con la billetera en el bolsillo trasero del pantalón; a simple vista, puede parecer que no es algo relevante, pero la billetera

genera que la articulación de la cadera se encuentre ligeramente elevada de uno de sus lados, afectando el punto de gravedad del cuerpo y generando una descompensación de cargas, donde, los músculos del cor buscan compensar ese desequilibrio inclinando la articulación más proximal a la cadera, que, en este caso corresponde a los hombros [17].

Después de entender por qué estas dos articulaciones son esenciales para el análisis postural en posición de sedestación, es necesario identificar los movimientos específicos que se ejercen en dicha posición. Para esto se realiza un estudio previo de todos los movimientos que realizan las articulaciones de los hombros y la cadera (revisar el marco teórico) identificando empíricamente los movimientos que se asume se ejercen en la posición de sedestación, esto, ya que, ningún libro, artículo o estudio brinda información suficiente para concluir que estos movimientos efectivamente corresponden a los que se realizan mientras una persona se encuentra sentada.

En relación a la cadera se identifican los movimientos de anteversión, retroversión, inclinación lateral externa, inclinación lateral interna, rotación interna y rotación externa [23-24-27]. Para reconocer cómo estos movimientos se involucran en la posición de sedestación se relaciona cada uno con situaciones específicas en las que se realizan dichos movimientos y las posibles afectaciones de postura que generan.

En este contexto, la anteversión se vincula al momento en el cual la persona toma una postura recta muy pronunciada, la cual termina convirtiéndose en una hiperlordosis, donde, la columna genera una curvatura interna exagerada y las caderas y los glúteos sobresalen, produciendo afectaciones en el soporte del tronco. La retroversión se vincula al momento en el cual una persona se encuentra prácticamente acostada o encorvada sobre la silla, donde, se produce una lordosis lumbar disminuida y una reducción de movimiento en la pelvis [23-24-27].

La inclinación lateral interna y externa se vinculan al momento en el cual se almacenan objetos en la parte trasera del pantalón, los cuales generan la elevación de uno de los lados de la cadera, produciendo mayor carga sobre uno de los isquiones y llegando a producir una posible escoliosis. La rotación interna y externa se vinculan al momento en el cual una persona mueve uno de los lados de la cadera hacia el borde de la silla, generando una asimetría entre los extremos de la articulación [23-24-27].

En relación a los hombros se identifican los movimientos de elevación, descenso, abducción, aducción, rotación externa e interna del hombro a mayor amplitud [24-27]. Para reconocer cómo estos movimientos se involucran en la posición de sedestacion se relaciona cada uno con situaciones específicas en las que se realizan dichos movimientos y las posibles afectaciones de postura que generan.

En este contexto los movimientos de elevación y descenso se vinculan al momento en el cual una persona utiliza el mouse o el path mouse del computador, ya que, se realiza la elevación o el descenso de uno de los hombros, generando una descompensación de cargas y una futura posición de agotamiento de los brazos [24-27].

La aducción se vincula al encorvamiento de la columna, ya que, no solo se realiza un desplazamiento de la cadera, sino de los hombros, produciendo afectaciones desde la región dorsal hasta la lumbar. La aducción se genera por acostarse en la silla o al encorvarse. La abducción se vincula a la hiperlordosis del tronco, que sucede cuando los hombros se llevan hacia atrás, al igual que la cadera, generando una tensión entre la columna y las articulaciones mencionadas [24-27].

Por otra parte, la rotación interna y externa de mayor amplitud se vinculan al momento en el cual la persona gira sus hombros y región dorsal, ya sea a la derecha o izquierda, produciendo que la cadera y la zona lumbar quede estática y generando asimetría entre la región dorsal y lumbar [24-27].

### 4.4.2 Identificación práctica de movimientos y articulaciones con un fisioterapeuta experto en ergonomía

Para confirmar que los movimientos detectados empíricamente se efectúan en la posición de sedestación, se consulta a un fisioterapeuta experto en ergonomía. Él cual nos ayuda a realizar los movimientos presentados anteriormente y a observar qué articulaciones específicas de los hombros y la cadera son las que realizan dichos movimientos, como se observa en la siguiente figura:

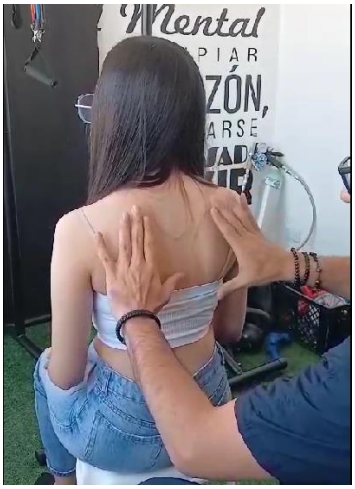



*Figura 18Movimiento de hombros y análisis de omoplatos [Propiedad de los autores].*

Esto nos permite identificar que estos movimientos si se desarrollan durante la posición de sedestación, corroborando así que el análisis teórico realizado anteriormente es correcto. Tras validar los movimientos, el siguiente paso es identificar las articulaciones específicas que los ejecutan, con el fin de conocer la ubicación exacta donde deben ir los sensores. Esto es importante, ya que al referirnos a hombros y cadera se abarcan zonas extensas del cuerpo, incluyendo articulaciones que pueden  no estar presentes o involucradas durante esos movimientos.

Al realizar movimientos de anteversión, retroversión, inclinación lateral externa, inclinación lateral interna, rotación interna y rotación externa de la cadera, se observa que la articulación más prominente y fácil de identificar corresponde a la articulación sacroilíaca. Para confirmar que esa articulación es la responsable de los movimientos mencionados anteriormente, se consulta la teoría [24], donde se observa que los

movimientos se generan cuando el fémur se encuentra fijo y el ilíaco se desplaza sobre él, reconociendo, que los ilíacos son los encargados de realizar los movimientos mencionados.

Al realizar movimientos de elevación, descenso, abducción, aducción, rotación externa e interna de los hombros, se observa que la articulación más prominente y fácil de identificar corresponde a los omoplatos o escápula. Para confirmar que esa articulación es la responsable de los movimientos se consulta la teoría [24-27], donde se observa que los movimientos mencionados anteriormente corresponden a movimientos de la región escapular, los cuales se efectúan en la articulación escapulotorácica, la cual está compuesta por la escápula en la parte posterior del tronco y el tórax en la parte frontal.

### 4.4.3 Identificación práctica de articulaciones

Con el fin de validar si los omoplatos e ilíacos son las articulaciones que intervienen en el gesto de la sedestación y son buenos candidatos para la ubicación de los sensores, se realiza una prueba con ayuda de 3 voluntarios, donde se calculan las estructuras del cuerpo que intervienen a la hora de sentarse, de la siguiente manera:

1. En una silla recta se ubica un papel cubriéndola en su totalidad.
2. Se pinta la parte posterior del cuerpo del voluntario, con el fin de calcar su estructura en el papel.
3. Se le indica al voluntario sentarse con una postura adecuada sobre el papel y hacer presión sobre la silla para marcar las estructuras que intervienen.
4. Sobre la figura obtenida se midien tres segmentos de la siguiente manera:
    a. Separación de las escápulas.
    b. Desde el borde medial de las escápulas hasta la zona lumbar-sacro.
    c. Desde la zona lumbar-sacro hasta el apoyo de los huesos isquiones.

Obteniendo los siguientes resultados

*Tabla 2Prueba calcar estructura física en papel.*

| **Voluntario** | **Características** | **Mediciones** | **Evidencia** |
|---|---|---|---|
| Voluntario 1 | Sexo: Femenino<br>Edad: 20 años<br>Estatura: 1,53 m | A: 13.5 cm<br>B: 25 cm<br>C: 13 cm | <br>*Figura 19Prueba 1 articulaciones que se involucran en la posición de sedestación [Propiedad de los autores].* |
| Voluntario 2 | Sexo: Masculino<br>Edad: 23 años<br>Estatura: 1,65 cm | A: 13 cm<br>B: 25,5 cm<br>C: 13,5 cm | <br>*Figura 20. Prueba 2 articulaciones que se involucran en la posición de sedestación [Propiedad de los autores].* |
| Voluntario 3 | Sexo: Femenino<br>Edad: 37 años<br>Estatura: 1,63 cm | A: 12 cm<br>B: 20,5 cm<br>C: 12,5 cm | <br>*Figura 21. Prueba 3 articulaciones que se involucran en la posición de sedestación [Propiedad de los autores].* |

A partir de las observaciones, se encuentra que las estructuras que se apoyan en la posición de sedestación corresponden a: las escápulas, la zona lumbar y sacroilíaca, y los huesos isquiones de la parte de la pelvis. Identificando así a los omóplatos o escápulas como la articulación principal de los hombros en la posición de sedestación

y asimismo la ubicación exacta de los sensores, ya que, aparte de ser la región específica donde se efectúa el movimiento, es una zona fácilmente detectable por el sensor para captar los cambios en la posición de la articulación del hombro.

También se identifica la articulación sacroilíaca como la articulación principal de la cadera en la posición de sedestación, pero, en este caso los ilíacos son los responsables específicamente de los movimientos que se realizan en la posición mencionada, por lo cual, se observa que la región más prominente y detectable para el sensor corresponde a las crestas ilíacas, identificando así la ubicación exacta donde deben ir los sensores.

## 4.5 Selección de componentes principales

En esta sección se presenta la selección de los componentes electrónicos principales para el diseño del dispositivo:

**Sensor inercial:**

Para cumplir el requerimiento de medir el ángulo de inclinación de los hombros y la cadera en los ejes x,y y z, se requieren 4 sensores inerciales capaces de medir la variación angular de las articulaciones, además de ser fáciles de programar, resistentes y precisos. A continuación, se presentan y comparan 3 sensores inerciales:

1. MPU6500

*Figura 22. Sensor inercial MPU6500 [33].*

## 2. CJMCU-103

*Figura 23. Sensor inercial CJMCU-103 [34].*

## 3. MPU6050

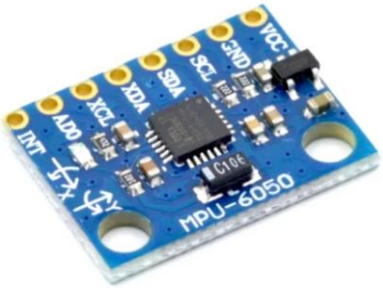



*Figura 24. Sensor inercial MPU6050 [35].*

*Tabla 3Comparación de las características de los sensores inerciales.*

| Sensor Inercial | Características | Ventajas | Desventajas | Precio | Proveedor |
|---|---|---|---|---|---|
| MPU6500 | -Acelerómetro de tres ejes.<br>-Giroscopio de tres ejes.<br>-Rango de aceleración: ± 2, 4, 8, 16G.<br>-Rango del Giroscopio: 250,500, 1000,2000°/s.<br>-Dimensiones: 15x25mm.<br>-Conexión I2C, SPI.<br>-Tensión de alimentación: 1.71 - 3.5v. | -Filtro digital que ayuda a reducir los ruidos.<br>Comunicación I2C y SPI.<br>- Configuración de baja potencia, ideal para sistemas que requieren bajo consumo energético. | -Complejidad en la programación debido a las funcionalidades adicionales con las que cuenta. -<br>Requiere un micro controlador con mayor procesamiento y memoria.<br>-Costo relativamente alto. | 14,000 COP | Mactronica |

| | | | | | |
|---|---|---|---|---|---|
| CJMCU-103 | -Ángulo de rotación: eléctrico, mecánico: 0 ° ~ 333,3°.<br>-Salida de voltaje analógica.<br>Linealidad: ± 2%.<br>-Resistencia (ohmios): 10k.<br>-Tolerancia de resistencia: ± 30%.<br>- Temperatura de funcionamiento: -40 ° C + 85 ° C. | -Bajo consumo de energía.<br>-Menor precio.<br>-Fácil de usar y programar. | -No cuenta con documentación suficiente.<br>-Baja precisión.<br>- Limitación en sus funcionalidades.<br>-Sensibilidad al ruido.<br>-Retardos en el procesamiento de señales. | 20,000 COP | Mercado Libre |
| MPU6050 | -Acelerómetro de tres ejes.<br>-Giroscopio de tres ejes.<br>-Rango de aceleración: ± 2, 4, 8, 16G.<br>-Rango del Giroscopio: 250,500, 1000,2000°/s.<br>-Tamaño: 14x21mm.<br>-Conexión I2C, SPI.<br>-Tensión de alimentación: 3.3 - 5v. | -Amplia capacidad de procesamiento de señales digitales. -Cuenta con documentación y estudios previos.<br>-Alto rendimiento.<br>Comunicación I2C. | -Baja precisión.<br>- Sensibilidad al ruido. | 15,000 COP | Dynamo Electronics |

A partir de las referencias comerciales encontradas, se identifica que la referencia del sensor inercial MPU6050, es el más acorde a la necesidad de este proyecto, ya que, contiene un giroscopio de tres ejes, además de permitir una comunicación I2C de alto rendimiento y de su velocidad para procesar señales digitales, sus rangos de medicion angular son acordes a lo requerido (0° a 180° para articulaciones de los hombros y 0° a 90° para articulación de cadera). Adicionalmente cuenta con bastante documentación para validar su funcionamiento y diversos usos.

**Microcontrolador:**

Para realizar la programación de los sensores, la comunicación entre el dispositivo, la aplicación móvil y la base de datos se requiere un microcontrolador fácil de implementar y programar, además de contar con comunicación WIFI y bluethoot, que sea de tamaño

reducido y poco consumo de corriente, además contar con el protocolo I2C que permita la conexión de los sensores seleccionados. A continuación, se presentan y comparan 4 microcontroladores:

1. Arduino NANO:

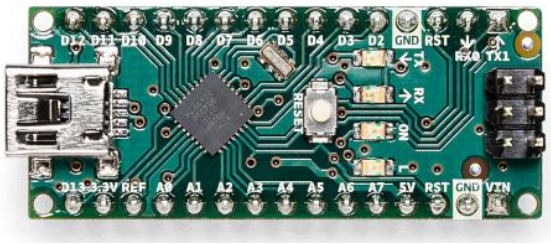



*Figura 25. Arduino nano [36].*

2. ESP32:

*Figura 26. ESP-32 [37 - 38].*

3. Raspberry Pi Zero 2W:

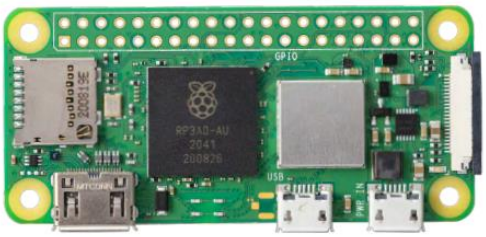



*Figura 27. Raspberry Pi Zero 2W [39].*

## 4. STM32

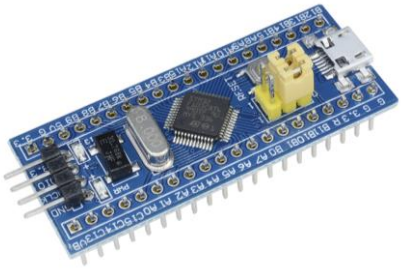


*Figura 28. TSTM32F103C8T6 [40]*

*Tabla 4. Comparación de microcontroladores.*

| **Sensor Inercial** | **Características** | **Ventajas** | **Desventajas** | **Precio** | **Proveedor** |
|---|---|---|---|---|---|
| Arduino Nano | - Alimentación de 5V y consumo de 800mA.<br>- Interfaces de comunicación UART, I2C y SPI.<br>- 20 pines digitales, 8 análogos y 8 PWM.<br>-Procesador Atmega 8bits - 16MHz<br>-Memoria Flash,SRAM, EEPROM | - Amplia variedad de librerías.<br>- Tamaño compacto.<br>- Compatibilidad con diversos módulos.<br>- Accesibilidad en el mercado. | - Rendimiento limitado en comparación a otros microcontroladores.<br>-Sin conectividad nativa a WiFi o Bluetooth.<br>- | $39,000 a $142,800 COP | Mercado Libre, Bigtronica, Sigma Electronica |
| ESP32 | -Wi-Fi 2.4GHz y Bluetooth<br>-Alimentación 3.6V y consumo 500mA<br>- Interfaz de comunicación GPIO, SPI, I2C, UART y SDIO.<br>- Memoria ROM y SRAM.<br>- 34 pines GPIO<br>- Procesador Xtensa LX6 - 32 Bits 240MHz<br>- | -Amplia compatibilidad con entornos de programación.<br>-Bajo Consumo.<br>-Wifi y Bluetooth Nativos.<br>- Alta velocidad de procesamiento.<br>- Gran variedad de protocolos de comunicación | -Programación más compleja.<br>- | $29,750 a $35.000 COP | Mercado Libre, Sigma Electronica, TNC Electronica |

| | | | | | |
|---|---|---|---|---|---|
| Raspberry Pi Zero 2W | – Wi-Fi 2.4GHz, Bluetooth y BLE.<br>- Alimentación 5V y consumo de 2.5 A.<br>- Interfaces de comunicación GPIO, HDMI, USB.<br>- Memoria RAM y Micro.<br>- 36 Pines GPIO.<br>- Broadcom Arm Cortex-A53 - 64 bits 1GHz | -Alta velocidad de procesamiento.<br>- Wi-Fi y Bluetooth integrados.<br>- Versatilidad de usos. | -Alto Consumo energetico.<br>-Alto precio.<br>-Configuración compleja | 95,200 COP | Sigma Electronica |
| STM32F103 | - Alimentación 3.6V y consumo 150mA.<br>-Protocolos de comunicación ISP, I2C, USART, CAN y USB.<br>-Memoria SRAM y Flash.<br>-39 pines GPIO y 10 Analogos.<br>-ARM Cortex - 32 Bits 72MHz | - Bajo consumo.<br>-Alto Rendimiento.<br>-Bajo costo | - Programación Compleja.<br>- Poca documentación de desarrollo.<br>- | $23.800 - $35.000 COP | Mercado Libre |

A partir de las referencias comerciales encontradas, se identifica que la tarjeta de programación ESP 32, es la más acorde a la necesidad de este proyecto, ya que, cuenta con un módulo WiFi y Bluetooth integrados que facilita la conexión entre el dispositivo y la aplicación sin necesidad de módulos externos, el tamaño es conveniente con la escala del prototipo, es de bajo consumo energético y un buen costo teniendo en cuenta sus capacidades de memoria, procesamiento de datos y gran alcance de implementación.

Es posible usar los otros microcontroladores, pero es necesario adicionar módulos externos que permitan una conexión inalámbrica, además el consumo de corriente aumenta y así mismo las características energéticas de la batería deben ser robustas.

**Multiplexor:**

Debido a la capacidad del protocolo I2C en los microcontroladores del mercado, se requiere implementar un multiplexor I2C que nos permita la conexión simultánea de los 4 sensores por los mismos pines del protocolo I2C, se realizó el análisis de diferentes Multiplexores I2C que ofrecen en el mercado:

1. Multiplexor 1 a 8 I2C

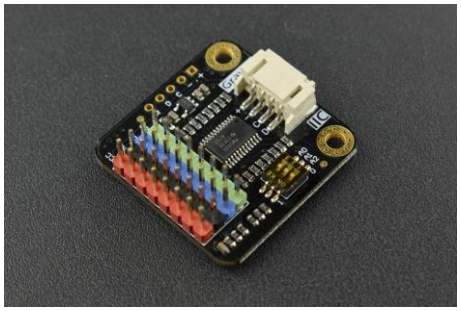


*Figura 29. Multiplexor 1 a 8 I2C [41]*

2. Módulo Multiplexor I2C TCA9548A

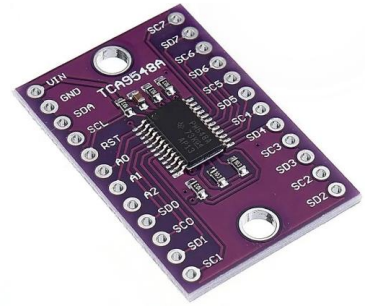



*Figura 30. Multiplexor 1 a 8 I2C [42]*

3. Multiplexor I2C TCA9548A

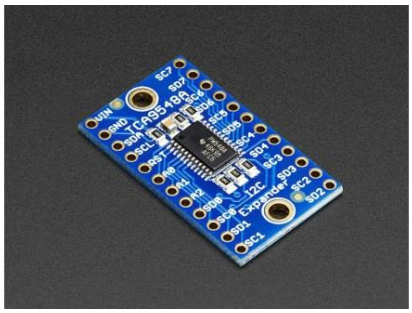


*Figura 31. Multiplexor 1 a 8 I2C [43]*

*Tabla 5. Comparación de multiplexores.*

| **Multiplexor** | **Características** | **Precio** |
|---|---|---|
| Multiplexor 1 a 8 I2C DFRobot | -Voltaje de operación de 3.3 - 5V<br>- Conector Pitch 4 pines<br>-Velocidad del reloj de 100k - 400k<br>- 32mm X 32mm | 40,100 COP |
| Módulo Multiplexor I2C TCA9548A | -Voltaje de operación de 3.3 - 5V<br>-Velocidad del reloj de 100k - 400k<br>- 30,6mm X 17,6mm | 8,000 COP |
| Multiplexor I2C TCA9548A | -Voltaje de operación de 3.3 - 5V<br>- 30,6mm X 17,6mm<br>- Incluye tira de pines para soldar | 36,500 COP |

Se encuentra que técnicamente tienen las mismas características, se diferencian en sus dimensiones, composición física y precio. Se elige el multiplexor DFRobot debido a su composición física y su fácil adquisición, además cumple con las características técnicas para la conexión I2C, al tener un Conector Pitch y pines de conexión sin necesidad de soldar regleta genera robustez y seguridad de implementación.

**Batería:**

Para alimentar el dispositivo se requiere una batería, portable y con suficiente amperaje para mantener el dispositivo funcionando por un mínimo de 5 horas.

Teniendo en cuenta que el consumo total de corriente del microcontrolador es de 500mA, con la conexión bluetooth activa y los sensores conectados directamente a la alimentación de la ESP32, la capacidad mínima de la batería debe ser de  como se muestra en el siguiente modelo matemático:

$$Capacidad\ de\ la\ bateria = Consumo\ de\ corriente \times Tiempo\ de\ funcionamiento \quad (1)$$

$$Capacidad\ de\ la\ bateria\ (mAh) = 500 \times 10^{-3}(5) = 2500mAh \qquad (2)$$

A continuación, se presentan y comparan 3 baterias:

1. Batería 3.7v 2000mah Litio

*Figura 32. Batería Litio 18650 - 2500mah 3.7v [44].*

2. Pila alcalina AA

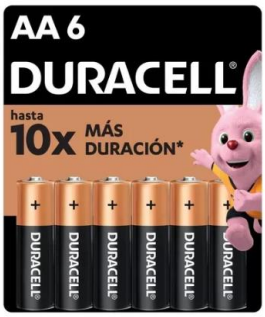



*Figura 33Pilas AA alcalina 1.5v [45].*

3. Power Bank

*Figura 34. Power Bank 12v [46].*

*Tabla 6Comparación de las características de las baterías.*

| **Batería** | **Características** | **Ventajas** | **Desventajas** | **Precio** | **Proveedor** |
|---|---|---|---|---|---|
| Batería 3.7v 2500 mah Litio | Batería de Polímero Ion-litio de 3.7v 2500mah, recargable, dimensiones: Alto 65mm x ancho 18mm | Se puede cargar la batería mientras el circuito está funcionando. Mantiene una carga estable durante largos períodos, aún si no se encuentra en uso. Permite una carga rápida Larga vida útil. Tiene un voltaje estable durante la mayor parte de su descarga | La tensión de la batería es demasiado alta para alimentar la ESP32 por el pin de 3.3v, por ende, se requiere un regulador de voltaje LDO que reduzca el voltaje de la batería a 3.3V. Requiere circuito de protección para evitar sobrecargas. | 32,000 COP | Ferretronica |
| Pila alcalina AA | Pila Alcalina AA de 1.5v 700mah, dimensiones: Alto 65mm x ancho 18mm. | Las pilas alcalinas AA tienen un voltaje nominal de 1.5V , generando que al conectar dos baterías en serie se obtenga un voltaje de 3V, permitiendo alimentar directamente al pin de 3.3V del microcontrolador | La corriente que suministra la pila alcalina AA es solo de 700 mA, por ende, el microcontrolador consumiría toda la corriente, generando una caída de voltaje y produciendo el reinicio del mismo. No es recargable. Tiene efecto memoria. Pérdida de voltaje a medida que se descargan | 3,586 COP | Mercado Libre |
| Power Bank | Power Bank de 12v 20000mah, cuenta con un puerto de salida de hasta 5v 2100mah dimensiones: Alto 28mm x ancho 69mm x largo 140mm. | Recargable Permite recargar dispositivos en corto tiempo debido a su carga rápida Cuenta con circuitos de protección contra sobrecargas y cortocircuitos Están diseñadas para utilizarse diariamente | No es eficiente energéticamente, ya que, realiza múltiples conversiones de voltaje, consumiendo cantidades considerables de energía permanentemente. La batería se puede deteriorar si no se utiliza regularmente. Se puede desperdiciar energía durante la carga. La tensión de la batería es demasiado alta para alimentar la ESP32 por el pin de 3.3v, por ende, se requiere un regulador de voltaje LDO que reduzca el voltaje de la batería a 3.3V. | 94,900 COP | Mercado Libre |

A partir de las referencias comerciales encontradas, se identifica que la batería recargable de 3.7v 2500mah Litio es la más acorde a la necesidad de este proyecto, ya que, cuenta con el amperaje suficiente para que el sistema funcione correctamente.

Teniendo en cuenta que la tarjeta de programación ESP 32 DevKit V1, se observa que la la corriente de operación es de 80mA hasta 500mA y el voltaje de operación de la tarjeta es de 3.3v con un rango de 2.7v a 3.6v , para el cual se puede realizar la alimentación por dos pines principales: VIN el cual soporta 5v y el pin de 3.3v el cual soporta 3.3v, permitiendo así alimentar la tarjeta por el pin 3.3v con la ayuda de un regulador de voltaje, para ofrecer la cantidad de voltaje adecuada y la suficiente corriente para que el sistema no presente inconvenientes.

Adicional la batería se puede cargar mientras el circuito está funcionando, manteniendo una carga estable durante largos períodos, aún si no se encuentra en uso.

### 4.6 Calibración sensor inercial

Para iniciar con el diseño del dispositivo electrónico, es fundamental calibrar cada uno de los sensores inerciales. Esto implica realizar todas las conexiones físicas necesarias entre los pines de comunicación y alimentación de los sensores, dirigiéndolos hacia el multiplexor 1 a 8 y de allí hacia la ESP32.

La calibración se realiza mediante una curva de calibración, donde se relaciona cada una de las mediciones físicas de ángulos de inclinación con los valores leídos por la ESP32. Todos estos datos son tabulados y graficados con el fin de obtener una ecuación que linealice el comportamiento de cada uno de los sensores.

Los valores relacionados con las mediciones físicas, son los de aceleración respecto a la gravedad, ya que, gracias al principio de funcionamiento del sistema microelectromecanico que contiene el módulo MPU6050, dependiendo de su posición física el conjunto de masas ejerce fuerzas en los muelles haciendo que la elongación de los mismos cambie y así mismo se procesen diferentes datos para cada posición del muelle y obteniendo una lectura para cada una de las posiciones angulares físicas.

Las medidas físicas tomadas, se realizan con ayuda de un transportador, ubicando el sensor en cada una de las posiciones angulares, de la siguiente manera:

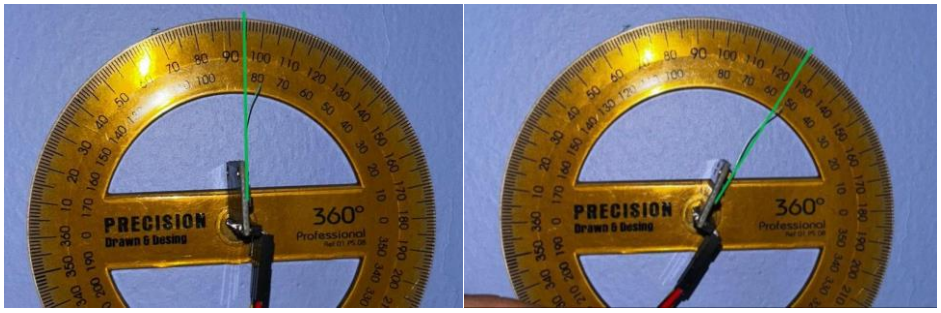



*Figura 35. Medición física para eje Y [Propiedad de los autores].*

*Figura 36Medición física para eje Z [Propiedad de los autores].*

Al ubicar el sensor inercial en cada una de las posiciones angulares, se debe mantener en esta posición por mínimo 2 segundos para que la masa y los muelles se estabilicen correctamente y tener una lectura más precisa.

Una vez recopilados todos los datos de las lecturas realizadas por la ESP32 de los sensores inerciales en cada una de las posiciones angulares, se realiza la tabulación de estos y posterior se grafican teniendo en cuenta que en el eje de las abscisas se ubican las lecturas obtenidas por la ESP32 y en el eje de las ordenadas cada una de las posiciones angulares, obteniendo las siguientes curvas de calibración:

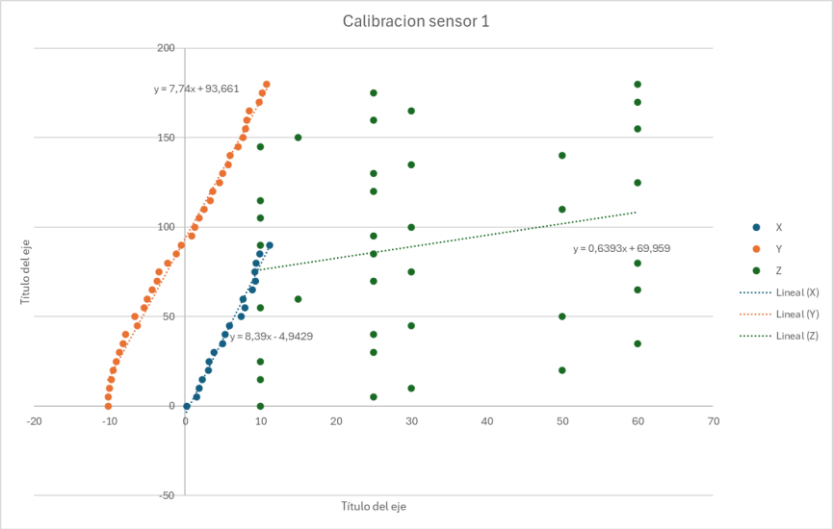



*Figura 37Curva de calibración sensor 1 [Propiedad de los autores].*

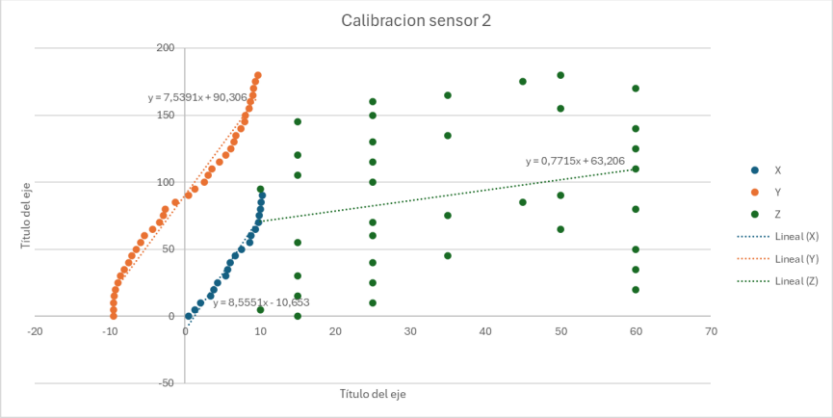



*Figura 38. Curva de calibración sensor 2 [Propiedad de los autores].*

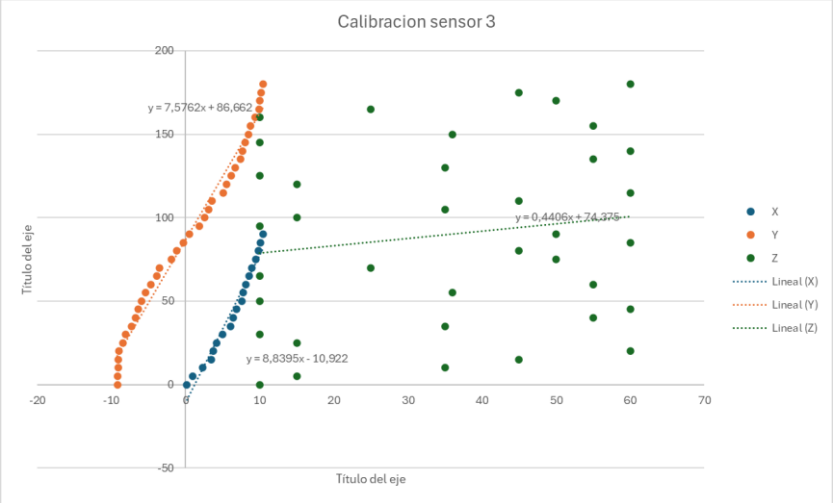



*Figura 39. Curva de calibración sensor 3 [Propiedad de los autores]..*

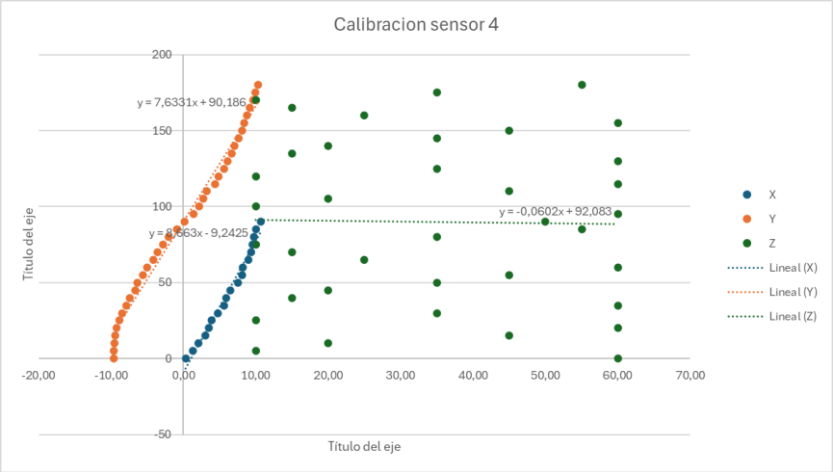



*Figura 40. Curva de calibración sensor 4 [Propiedad de los autores].*

Con ayuda de excel se agregan las líneas de tendencia y las ecuaciones que describen el comportamiento lineal del sensor en cada uno de los ejes y de esta manera se obtienen datos más precisos sobre el sensor que se está trabajando:

Las ecuaciones encontradas son las siguientes:

**Sensor 1:**

**Eje Y:**

$$y = 7.8785x + 94.621 \quad (3)$$

**Eje X:**

$$y = 8.4919x - 5.414 \quad (4)$$

**Eje Z:**

$$y = 0.6393x + 69.959 \quad (5)$$

**Sensor 2:**

**Eje Y:**

$$y = 7.5391x + 90.306 \quad (6)$$

**Eje X:**

$$y = 8.5551 - 10.653 \quad (7)$$

**Eje Z:**

$$y = 0.7715x + 63.206 \quad (8)$$

**Sensor 3:**

**Eje Y:**

$$y = 7.5762x + 86.662 \quad (9)$$

**Eje X:**

$$y = 8.8395x - 10.922 \quad (10)$$

**Eje Z:**

$$y = 0.4406x + 74.374 \quad (11)$$

**Sensor 4:**

**Eje Y:**

$$y = 7.6704x + 90.477 \quad (12)$$

**Eje X:**

$$y = 8.6845x - 9.3315 \quad (13)$$

**Eje Z:**

$$y = 0.0602x + 92.083 \quad (14)$$

### 4.7 Comunicación I2C, sensores inerciales y tarjeta de programación

Para comunicar los 4 sensores inerciales requeridos en el diseño, se utiliza el protocolo I2C. Este protocolo es comúnmente usado para la comunicación de dispositivos digitales, este se encarga de enviar uno o varios bytes de información dependiendo del conjunto de bits en cada trama y sus conexiones físicas, permitiendo interconectar hasta 127 dispositivos [47].

Se caracteriza por tener dos cables de comunicación: SDA (Serial Data) que se encarga de suministrar la información entre el dispositivo maestro y los esclavos, y el SCL (Serial Clock) que controla el reloj permitiendo iniciar o detener la comunicación. [47].

Es necesario utilizar el protocolo I2C en el dispositivo debido a la cantidad de sensores a implementar y la información que se requiere recibir y transmitir, para ello se selecciona el multiplexor 1 a 8 DFRobot, el cual permite recibir información por sus 8 canales. Además, este se conecta solo a dos pines (SCL y SDA) de la ESP32 para una comunicación correcta. [48].

Para la lectura y configuración de los puertos I2C del multiplexor se utiliza el entorno de Arduino IDE, en donde se incluye la librería (DFRobot_I2C_Multiplexer.h) requerida para lectura de datos en cada uno de los canales del multiplexor [48].

La conexión física de los componentes se realiza conectando cada uno de los pines (SDA,SCL,VCC, GND) del conector pitch del multiplexor a la ESP32 y los sensores a los canales 1 al 4 del multiplexor de la siguiente manera:

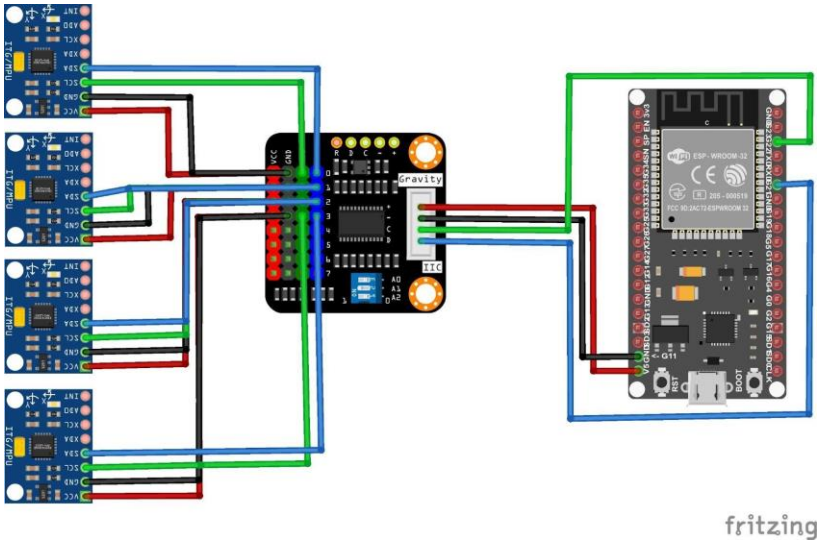



*Figura 41. Conexión Multiplexor 1 a 8 I2C [Propiedad de los autores].*

Además, del conector pitch, el multiplexor también ofrece una conexión por pines, en caso de que se requieran usar en una protoboard o simplemente adaptarlo a un circuito impreso. Cabe resaltar que gracias a las estructuras físicas de cada uno de los componentes usados, el circuito es modular y fácil de implementar, permitiendo cambiar las conexiones físicas entre ellos en cualquier instante, disminuyendo los riesgos a errores por mala implementación o fallas en circuitos impresos.

### 4.8 Integración batería en el dispositivo

Para que el dispositivo sea portátil e inalámbrico, se integra una batería en el diseño; con el fin de proteger la vida útil de la batería y del dispositivo en general, se adiciona un circuito

basado en módulos, para que este se pueda cargar sin necesidad de retirar la batería del dispositivo y se puedan generar errores por mala manipulación.

Se integra un módulo cargador de baterías de litio TP4056, que contiene un circuito de protección de corriente, tensión inversa y sobre tensiones, protegiendo la batería y el circuito[49], este permite una tensión de entrada alrededor de 4.5 a 5 voltios, comúnmente usada en múltiples dispositivos con baterías de litio.

Debido a que la descarga de la batería disminuye la eficiencia y el correcto funcionamiento del dispositivo, se integra un elevador a 5V XL6009 el cual mantiene una tensión constante hacia el microcontrolador hasta que la batería contenga menos del 50% de su almacenamiento.

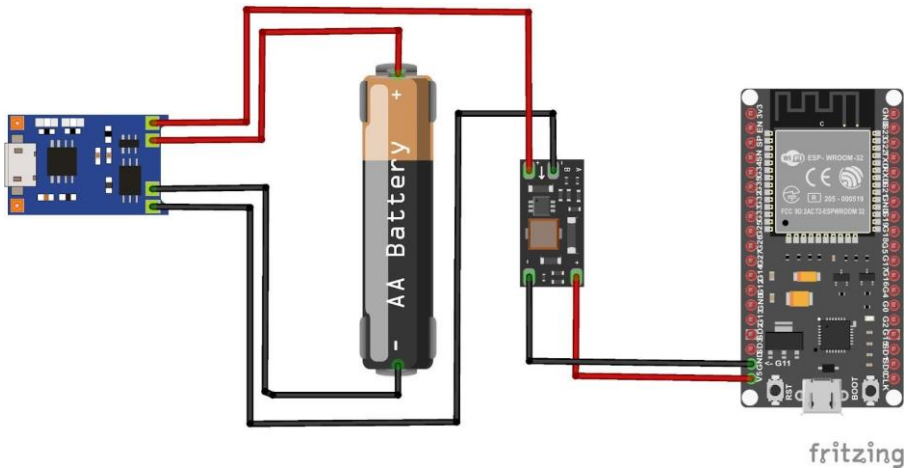



*Figura 42. Integración Batería [Propiedad de los autores].*

Este circuito tiene dos funciones principales, la primera se encarga de cargar la batería y proveer energía al dispositivo, y la segunda se basa en la protección de la batería, el microcontrolador y el usuario, ya que el módulo TP4056 contiene un comparador interno hecho con un transistor que apaga y prende el circuito, además de un conjunto de diodos que limita el flujo de corriente.

### 4.9 Relación entre movimientos de los hombros y la cadera y los movimientos de Roll, Pitch y Roll del sensor inercial

Para poder entender cómo los sensores captan los movimientos de rotación que se realizan sobre los ejes x, y, z, es importante relacionar los movimientos de los hombros y cadera en posición de sedestación con los movimientos de roll, pitch y roll del sensor inercial.

El sensor inercial gracias a su funcionamiento de giroscopio permite detectar los ángulos de inclinación en el eje x, y, z. Al relacionarlo con los movimientos de yaw, roll y pitch de un avión, se asume que el sensor se comporta como uno. De modo que, se observa que el movimiento de yaw el cual se realiza sobre un eje vertical se relaciona a un movimiento sobre el eje z, que visto desde el cuerpo humano sería un eje que atraviesa desde el centro de la cabeza hasta el centro de los pies. El movimiento de pitch el cual se realiza sobre un eje lateral se relaciona a un movimiento sobre el eje y, que visto desde el cuerpo humano sería un eje que atraviesa de hombro a hombro o cadera a cadera. El movimiento de roll el cual se realiza sobre un eje longitudinal se relaciona a un movimiento sobre el eje x, que visto desde el cuerpo humano sería un eje que atraviesa la cadera desde la parte frontal hasta la lateral.

Después de haber relacionado los movimientos de yaw, roll y pitch con los ejes de inclinación x, y, z, se pasa a entender la similitud entres los movimientos de rotación de yaw roll y pitch con los movimientos de las articulaciones hombros y cadera en la posición de sedestación de la siguiente manera:

**Hombros:**

1. El movimiento de Roll de un avión es cuando una de las alas del avión asciende y la otra desciende [29], dicho movimiento se relaciona con los movimientos de elevación y descenso de los hombros, ya que, los hombros pueden ascender y descender tal como si cada uno fuera un ala de un avión. Ejerciendo esos movimientos sobre el eje x..

2. El movimiento de Pitch de un avión es cuando la nariz del avión asciende y la cola desciende o viceversa [29], dicho movimiento se relaciona con los movimientos de abducción y aducción del hombro respecto al omoplato. Ejerciendo esos movimientos sobre el eje y.
   Donde, la abducción equivale a inclinar los hombros hacia al frente o siendo más específicos a encorvarse, haciendo referencia al momento donde la cola del avión se eleva y la nariz del avión se inclina, generando un movimiento de inclinación hacia el frente.
   La aducción que equivale a mover los hombros hacia atrás, siendo más específicos a tomar una postura recta exagerada, haciendo referencia al momento donde la nariz del avión se eleva y la cola del avión se inclina, generando un movimiento de inclinación hacia atrás.
3. El movimiento de Yaw de un avión es cuando de la nariz al centro del avión gira a la derecha y del centro del avión a la cola gira a la izquierda sobre su propio eje, como si hubiera una línea imaginaria atravesando el avión desde la parte superior hasta la inferior en su centro [29]. Dicho movimiento se relaciona con los movimientos de rotación externa y rotación interna del hombro a mayor amplitud, ya que, equivale a que la línea imaginaria pase desde el centro de la cabeza hasta el centro de los pies, generando que los hombros roten sobre su mismo eje. Ejerciendo esos movimientos sobre el eje z.

**Cadera:**

1. El movimiento de Roll de un avión es cuando una de las alas del avión asciende y la otra desciende [29]. Dicho movimiento se relaciona con los movimientos de inclinación lateral interna e inclinación lateral externa. Siendo cada cadera un ala del avión, de modo que se puede ascender o descender.
2. El movimiento de Pitch de un avión es cuando la nariz del avión asciende y la cola desciende o viceversa [29]. Dicho movimiento se relaciona con los movimientos de anteversión y retroversión. Ejerciendo esos movimientos sobre el eje y.
   Donde, la abducción equivale a inclinar la cadera o la pelvis específicamente hacia al frente, produciendo un movimiento casi acostado sobre la silla, haciendo referencia

al momento donde la cola del avión se eleva y la nariz del avión se inclina, generando un movimiento de inclinación hacia el frente. La aducción que equivale a mover la cadera o la pelvis hacia atrás, siendo más específicos a tomar una postura recta exagerada, haciendo referencia al momento donde la nariz del avión se eleva y la cola del avión se inclina, generando un movimiento de inclinación hacia atrás.

3. El movimiento de Yaw de un avión es cuando de la nariz al centro del avión gira a la derecha y del centro del avión a la cola gira a la izquierda sobre su propio eje, como si hubiera una línea imaginaria atravesando el avión desde la parte superior hasta la inferior en su centro [29]. Dicho movimiento se relaciona con los movimientos de rotación externa y rotación interna del ilíaco, ya que, equivale a que la línea imaginaria pase desde el centro de la cabeza hasta el centro de los pies, generando que la cadera rote sobre su mismo eje. Ejerciendo esos movimientos sobre el eje z.

### 4.10 Diseño prenda de vestir para ubicación de sensórica

Para posicionar los sensores inerciales sobre cada una de las articulaciones se diseña una prenda que permite ajustar cada uno de los sensores dependiendo de la morfología del usuario, además de contar con un compartimiento o bolsillo para el almacenamiento de la batería y carcasa del dispositivo. Para esto, se selecciona una camiseta como diseño base, donde, se consulta con una experta en diseño y confección de prendas, quien es informada sobre el objetivo del proyecto y la importancia de una adecuada ubicación de los sensores.

El primer boceto se centra en la estructura de un arnés de seguridad para trabajos en alturas. Al desarrollar este diseño, se observa que esta estructura textil puede integrar los sensores en los puntos identificados teniendo una correcta lectura, además de ser fácil de usar y llevar a cualquier parte.

*Figura 43. Diseño preliminar del arnés [Propiedad de los autores].*

Posterior a ello, se analiza cómo integrar todo el dispositivo a una sola prenda, teniendo en cuenta que esta debe ser lo menos invasiva posible, además de ser fácil de portar y transportar. Esto lleva al diseño de una camiseta con tela deportiva, que incorpora las guías necesarias para replicar la estructura del arnés, con el fin de poder ubicar los sensores de manera correcta. Además, el diseño cuenta con un compartimiento para guardar la circuitería asociada al dispositivo.

*Figura 44. Diseño preliminar camiseta [Propiedad de los autores].*

A partir de los diseños mencionados anteriormente , se elabora la camiseta siguiendo las características mínimas requeridas, obteniendo la siguiente prenda:

*Figura 45. Diseño Camiseta[Propiedad de los autores].*

La camiseta contiene 5 líneas guía donde en 4 de ellas se ubican los sensores y la quinta dirige todos los cables hacia el multiplexor y microcontrolador. Cada guía contiene unos orificios en sus extremos con el fin de poder manipular el sensor y ubicarlo en la posición correcta, además el bolsillo de la parte delantera permite ubicar el microcontrolador, multiplexor, batería y demás componentes asociados al dispositivo.

### 4.11 Almacenamiento de datos

Para identificar las variables a almacenar en la base de datos se realiza un diagrama entidad relación, en el cual se plantean 3 entidades: Autorización usuario, Información personal e Información médica, donde existe una relación 1:1, debido a que el usuario que ingresa solo se puede autenticar una vez, tener una sola información personal y una sola información médica, adicional la llave entre las entidades corresponde al “usuario” el cual hace referencia al correo electrónico con el cual se realiza el logueo a la aplicación móvil, como se observa en la siguiente figura:

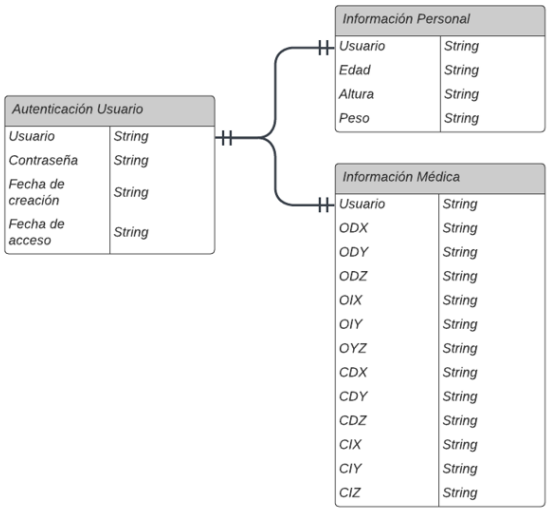



*Figura 46. Diagrama Entidad relación [Propiedad de los autores].*

El diseño de la base de datos se realiza utilizando FireBase, una plataforma en la nube, ideal para el desarrollo de aplicaciones. Firebase simplifica la creación del backend al ofrecer una variedad de servicios que automatizan tareas comunes necesarias para el funcionamiento de las aplicaciones móviles. Además cuenta con el respaldo de google y permite la integración de otras aplicaciones como lo es Android Studio [50].

Para acceder a la plataforma es necesario ingresar con la cuenta aliada de google y seleccionar el motivo por el cual se va a implementar, en donde se informa que es 100% de carácter académico.

Al ingresar a la plataforma se seleccionan dos servicios, los cuales corresponden al almacenamiento principal de datos que se va a realizar en la aplicación, el primero es Firebase Authentication [51], el cual identifica a los usuarios, permitiendo que la aplicación guarde los datos de usuario y contraseña en la nube de forma segura. Además, de conceder desde Firebase la validación del usuario, la fecha de creación, el proveedor con el que fue creado, la última fecha de acceso y realizar operaciones como: restablecer contraseña, inhabilitar cuenta, agregar usuarios y borrar cuenta, tal como se observa en la siguiente figura:

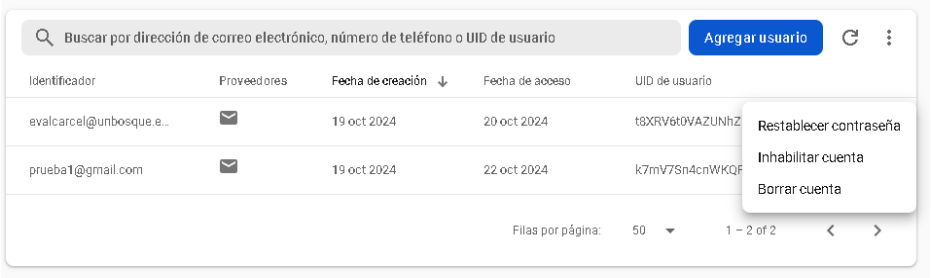



*Figura 47 .Diseño servicio Firebase Authentication [Propiedad de los autores].*

Por otra parte, el segundo servicio es Cloud Firestore, el cual corresponde a una " base de datos NoSQL flexible, escalable y en la nube, creada en la infraestructura de Google Cloud, a fin de almacenar y sincronizar datos para el desarrollo tanto del lado del cliente como del servidor" [52].

La cual nos permite la creación de dos tablas principales: usuarios e información médica, donde se almacena información personal del usuario, como lo es: edad, peso y altura y la información médica, como lo es: ángulos de inclinación en el eje x, y, z del omóplato derecho, ángulos de inclinación en el eje x, y, z del omóplato izquierdo, ángulos de inclinación en el eje x, y, z de la cresta iliaca derecha y ángulos de inclinación en el eje x, y, z de la cresta iliaca izquierda.

Adicional, desde Android Studio se realiza la creación de las tablas y subcategorías, utilizando Kotlin como lenguaje de programación, permitiendo realizar consultas a la base de datos, cómo: recuperar, modificar o eliminar información y brindando la opción de tener variables de tipo string, int, float, entre otras, como se observa en la siguiente figura:

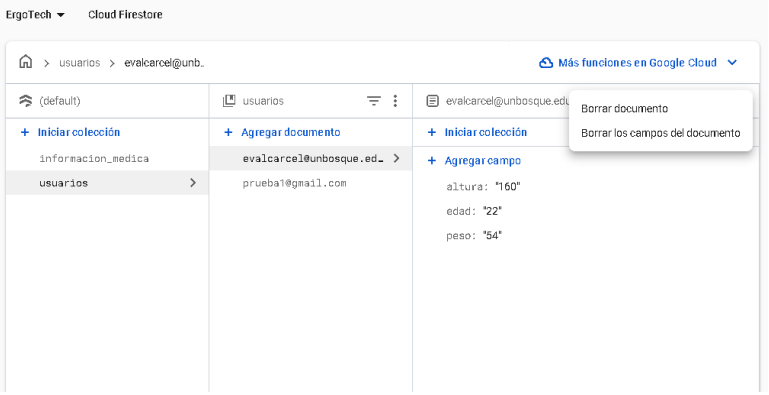



*Figura 48 .Diseño servicio Cloud Firestore [Propiedad de los autores].*

Para realizar la conexión entre Firebase Authentication, Cloud Firestore y la aplicación desarrollada en Android Studio se realizan los siguientes pasos:

1. Crear un proyecto en Firebase, el cual se denomina ErgoTech, ya que, corresponde al nombre de la aplicación móvil.
2. Descargar el archivo google-services.json directamente desde Firebase y pegarlo en el directorio raíz del módulo de la aplicación.
3. Agregar el complemento de Google Services como dependencia en el archivo de Gradle a nivel de proyecto, adicionando la línea id("com.google.gms.google-services") version "4.3.15", en la sección de plugins
4. Agregar el complemento de Google Services como dependencia en el archivo de Gradle a nivel de módulo, adicionando la línea id("com.google.gms.google-services"), en la sección de plugins
5. Agregar los SDK de firebase a la aplicación, en el archivo Gradle a nivel de módulo, en la sección de dependencias, como se observa en la figura x (en esta sección se agregan todas las librerías y los servicios de la cual depende la aplicación).

```
implementation(platform("com.google.firebase:firebase-bom:33.4.0"))
implementation("com.google.firebase:firebase-analytics")
implementation("com.google.firebase:firebase-auth:23.0.0")
implementation("com.google.firebase:firebase-firestore:25.1.0")
```

*Figura 49.SDK de Firebase [Propiedad de los autores].*

6. Modificar la versión SDK de la aplicación de 21 a 23.
7. Sincronizar el proyecto de Android con los archivos de Gradle.
8. Verificar la conexión entre la aplicación y el servicio de Authentication desde Android Studio, en la sección de herramientas y la opción de Firebase, como se observa en la siguiente figura:

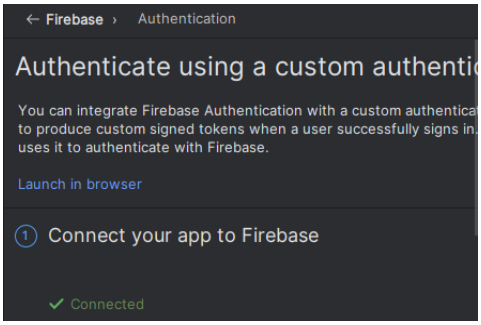



*Figura 50.Conexión Android Studio y Firebase Authentication [Propiedad de los autores].*

9. Verificar la conexión entre la aplicación y el servicio Cloud Firestore desde Android Studio, en la sección de herramientas y la opción de Firebase, como se observa en la siguiente figura:

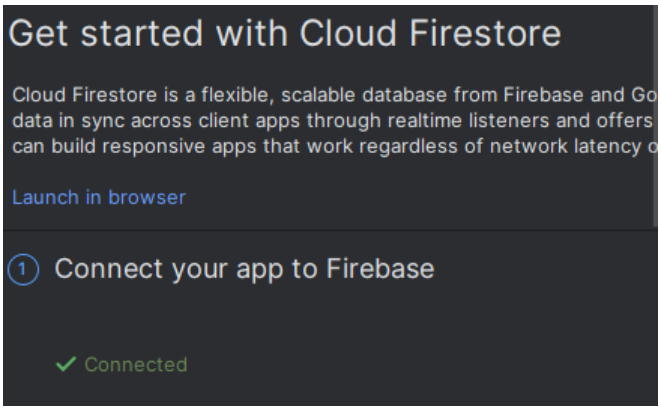



*Figura 51 .Conexión Android Studio y Cloud Firestore [Propiedad de los autores].*

10. Declarar los servicios de Firebase como variables globales en cada una de las ventanas que realizan conexión a Firebase.

## 4.12 Diseño aplicación móvil

### 4.12.1 Diseño preliminar

Como parte del desarrollo del proyecto, se realiza el diseño preliminar de aplicación móvil, para identificar las ventanas que se requieren para la creación del sistema de monitoreo postural.

El diseño preliminar se realiza en la plataforma web Figma, ya que, es una herramienta de diseño UI y UX, que permite crear maquetas y prototipos interactivos de páginas web, aplicaciones móviles, entre otros, en un entorno colaborativo, donde varios miembros del equipo pueden comentar y editar sobre un mismo proyecto. [54]

*Figura 52 .Diseño preliminar aplicación móvil [Propiedad de los autores].*

El diseño de la aplicación consta de 6 pantallas, donde al iniciar la aplicación el usuario se encuentra con una pantalla de inicio de sesión, la cual permite su identificación, actuando como llave de acceso a la aplicación y a la información almacenada en la base de datos.

Una vez el usuario ingresa, puede navegar por medio de un menú interactivo hacia las demás pantallas donde puede: ingresar información personal, ingresar información médica, conocer el manual de usuario y monitorear su postura.

Este diseño establece una base sólida para el desarrollo de la aplicación móvil, ya que, incluye las pantallas esenciales, botones, campos de texto e información general. Además, sirve como

herramienta para evaluar la experiencia del usuario, permitiendo realizar una simulación real del funcionamiento de la aplicación.

### 4.12.2 Diseño Final

Para el desarrollo de la aplicación móvil se selecciona Android Studio como entorno de programación, ya que, permite la creación de aplicaciones móviles compatibles con Android, además de implementar varios lenguajes de programación como: Java, Kotlin, Java Script, C++ entre otros.

Se inicia con el desarrollo de diferentes activitys o ventanas, creando una para cada servicio que va a estar disponible dentro del software. Donde el frontend de cada vista se desarrolla en HTML, ya que, brinda facilidad en la organización de botones, recuadros de texto, textos editables, fondos, entre otros, permitiendo hacer el diseño mucho más amigable con el usuario y de fácil entendimiento. Kotlin es el lenguaje seleccionado para el desarrollo del backend, ya que, es un lenguaje multiplataforma con bases de Java, funciones más desarrolladas, menos limitaciones y restricciones.

La primera ventana corresponde al login del usuario. Donde el frontend de la aplicación cuenta con 2 textos editables y 2 botones, como se observa en la figura 52. Los textos editables permiten el ingreso del correo y la contraseña, mientras que los botones brindan el acceso o el registro a la aplicación. Internamente el botón de registro toma los campos correo y contraseña digitados y los almacena en el servicio Authentication de Firebase.

Mientras que el botón de acceso compara los datos ingresado en los campos de usuario y contraseña con los datos almacenados en Firebase de usuario y contraseña, generando que si los datos son correctos y efectivamente se encuentran almacenados el usuario pueda ingresar, pero si los datos no existen o son incorrectos genere una notificación informando que no es posible el acceso.

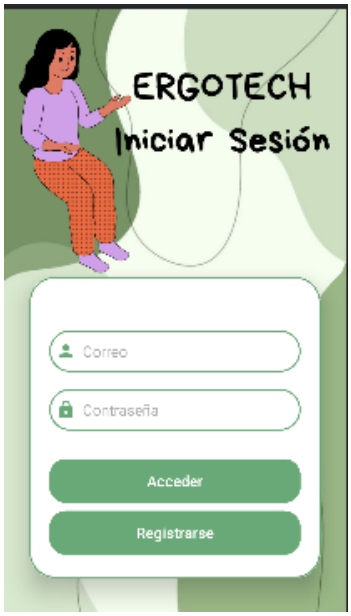



*Figura 53.Frontend ventana de Login [Propiedad de los autores].*

La segunda ventana corresponde al manual de usuario. Donde el frontend de la aplicación cuenta con una imagen que trae el manual y un botón de regreso, como se observa en la figura 53, el cual permite al usuario regresar al menú para seleccionar la opción o la venta a la que desee dirigirse.

*Figura 54 .Frontend ventana de Manual de usuario [Propiedad de los autores].*

La tercera ventana corresponde a la información personal. Donde el frontend de la aplicación cuenta con 4 textos editables y tres botones, como se observa en la figura 54. El texto editable de correo permite traer información de la base de datos automáticamente se abre la ventana, relacionando así el usuario que se loguea en la aplicación con el usuario que va a almacenar la información, por ende, dicho campo no permite digitar si no solo visualizar el correo con el cual se realizó el ingreso.

Los textos editables de peso, altura y edad permiten tanto el ingreso del correo y la contraseña, mientras que los botones brindan el acceso o el registro a la aplicación. Internamente el botón de registro toma los campos correo y contraseña digitados y los almacena en el servicio Authentication de Firebase.

Mientras que el botón de acceso compara los datos ingresado en los campos de usuario y contraseña con los datos almacenados en Firebase de usuario y contraseña

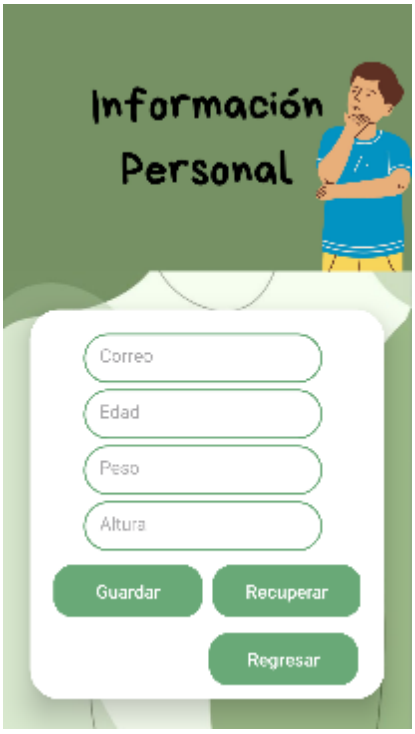



*Figura 55 .Frontend ventana de Información personal [Propiedad de los autores].*

La tercera ventana corresponde a la información personal. Donde el frontend de la aplicación cuenta con 4 textos editables y tres botones, como se observa en la figura 55. El texto

editable de correo permite traer información de la base de datos automáticamente se abre la ventana, relacionando así el usuario que se logueo en la aplicación con el usuario que va a almacenar lainformación, por ende, dicho campo no permite digitar si no solo visualizar el correo con el cual se realizó el ingreso.

Los textos editables de peso, altura y edad permiten tanto el ingreso de información como la visualización, de modo que el presionar el botón de guardar todo lo que haya escrito en esos campos se va a almacenar en la base de datos, mientras que al presionar el botón de recuperar, toda la información almacenada en la base de datos relacionada al usuario que se encuentra en el campo correo sera traida y buscada en cada uno de los textos editables según sea el caso.

Por otra parte, el botón de regreso permite al usuario regresar al menú para seleccionar la opción o la ventana a la que desee dirigirse.
La cuarta ventana corresponde a la información médica. Donde el frontend de la aplicación cuenta con 13 textos editables y tres botones, como se observa en la figura x. El texto editable de correo permite traer información de la base de datos automáticamente se abre la ventana, relacionando así el usuario que se loguea en la aplicación con el usuario que va a almacenar la información, igual que en la venta expuesta anteriormente.

Los textos editables se dividen en 4 subcategorías, relacionando 3 editables con cada una de las cuatro articulaciones a tener en cuenta, de modo que: los primeros 3 textos editables aceptan el ingreso del máximo valor del ángulo de inclinación que se permite para el omóplato derecho, otros 3 editables para el ingreso del máximo valor del ángulo de inclinación que se permite para el omóplato izquierdo, otros 3 editables para el ingreso del máximo valor del ángulo de inclinación que se permite para la cresta iliaca derecha y los últimos 3 editables para el ingreso del máximo valor del ángulo de inclinación que se permite para la cresta iliaca izquierda.

Los 12 textos editables mencionados anteriormente permiten tanto el ingreso de información como la visualización, de modo que el presionar el botón de guardar todo lo que haya escrito

en esos campos se va a almacenar en la base de datos, mientras que al presionar el botón de recuperar, toda la información almacenada en la base de datos relacionada al usuario que se encuentra en el campo correo sera traida y buscada en cada uno de los textos editables según sea el caso.

Por otra parte, el botón de regreso permite al usuario regresar al menú para seleccionar la opción o la ventana a la que desee dirigirse.

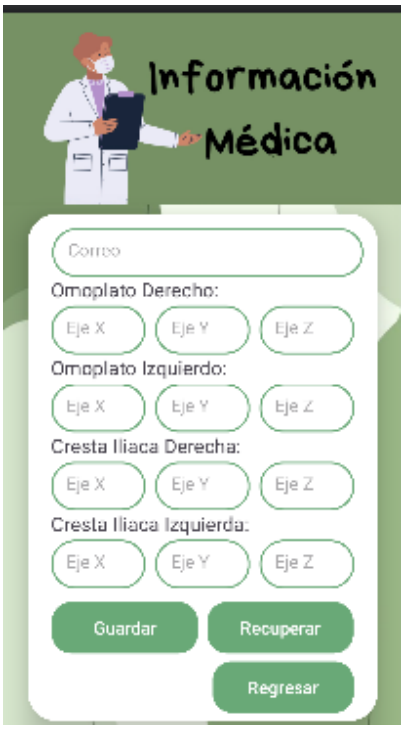



*Figura 56 .Frontend ventana de Información médica [Propiedad de los autores].*

La quinta ventana corresponde al monitoreo de postura. Donde el frontend de la aplicación cuenta con 10 textview  y un botón, como se observa en la figura 56. Los textview o textos de vista se dividen en 4 categorías, la primera correspondiente a datos captados por el dispositivo médico, donde 4 de los textview se destinan a la visualización de los ejes x, y, z del omóplato izquierdo, omoplato derecho, cresta iliaca izquierda y cresta iliaca derecha.

La segunda categoría responde a datos médicos, donde 4 de los textview se destinan a la visualización de los ejes x, y, z del omóplato izquierdo, omoplato derecho, cresta iliaca izquierda y cresta iliaca derecha almacenados en la base de datos.

La tercera categoría responde al textview que le indica al usuario si se genera una posición incorrecta y en qué articulación y eje se presenta. Para este visualizador de texto se implementan diferentes operaciones y variables internas, donde, se relaciona tanto la información médica de los ejes x, y, z de las 4 articulaciones mencionadas anteriormente como la información de los ejes x, y, z captada por el dispositivo. Donde, si alguno de los valores captados por el sensor supera el valor brindado por el médico especialista se genera un mensaje de alerta informando que se encuentra en una postura incorrecta y la articulación y el eje que está superando el valor estipulado por el especialista.

La cuarta categoría corresponde a un textview que permite visualizar si los 2 omoplatos y las 2 crestas iliacas se encuentran en simetría, de modo que, desde Arduino se comparan los ejes z de la parte derecha e izquierda de las articulaciones y se envía a la aplicación un mensaje informando si son o no simétricos.

Por otra parte, el botón de regreso permite al usuario regresar al menú para seleccionar la opción o la ventana a la que desee dirigirse.

*Figura 57 .Frontend ventana de Monitoreo [Propiedad de los autores].*

# 5. RESULTADOS Y ANÁLISIS DE RESULTADOS

## 5.1 Desarrollo del sistema electrónico para el monitoreo postural de la cadera y los hombros en personas que permanecen sentadas

Para desarrollar el sistema, se llevaron a cabo cuatro fases que están relacionados a los objetivos específicos:

1. Diseño del dispositivo electrónico para detectar la posición de hombros y cadera cuando una persona se encuentra sentada
2. Implementación del dispositivo electrónico.
3. Desarrollo del software para dispositivos móviles, capaz de monitorear y almacenar los datos recolectados por la sensórica implementada.
4. Integración y evaluación del dispositivo electrónico y el software

## 5.2 Diseño del dispositivo electrónico para detectar la posición de hombros y cadera cuando una persona se encuentra sentada

### 5.2.1 Validación calibración sensores inerciales:

Para verificar la calibración adecuada de los sensores, se ubica cada uno en cuatro ángulos aleatorios diferentes, donde, con ayuda de un transportador se miden los angulos de inclinacion en los ejes x y y de cada sensor , obteniendo los siguiente resultados:

*Tabla 7Validación calibración sensor inercial 1.*

| **Ángulos de inclinación eje Y Sensor 1** | | **Ángulos de inclinación eje x Sensor 1** | |
|---|---|---|---|
| **Medido** | **Leído** | **Medido** | **Leido** |
| 180 | 179.7 | 90 | 89.6 |
| 170 | 171.8 | 60 | 59.9 |
| 90 | 90.6 | 45 | 44.6 |
| 45 | 44.98 | 10 | 10.7 |

*Tabla 8. Validación calibración sensor inercial 2.*

| **Ángulos de inclinación eje Y Sensor 2** | | **Ángulos de inclinación eje x Sensor 2** | |
|---|---|---|---|
| **Medido** | **Leído** | **Medido** | **Leido** |
| 180 | 179.9 | 90 | 90.3 |
| 170 | 169.5 | 60 | 59.7 |
| 90 | 89.8 | 45 | 44.6 |
| 45 | 45.6 | 10 | 10.9 |

*Tabla 9. Validación calibración sensor inercial 3.*

| **Ángulos de inclinación eje Y Sensor 3** | | **Ángulos de inclinación eje x Sensor 3** | |
|---|---|---|---|
| **Medido** | **Leído** | **Medido** | **Leido** |
| 180 | 180.6 | 90 | 89.5 |
| 170 | 171.2 | 60 | 60.2 |
| 90 | 89.9 | 45 | 44.9 |
| 45 | 45.1 | 10 | 9.8 |

*Tabla 10. Validación calibración sensor inercial 4.*

| **Ángulos de inclinación eje Y Sensor 4** | | **Ángulos de inclinación eje x Sensor 4** | |
|---|---|---|---|
| **Medido** | **Leído** | **Medido** | **Leido** |
| 180 | 180.5 | 90 | 89.3 |
| 170 | 169.1 | 60 | 60.6 |
| 90 | 90.3 | 45 | 45.2 |
| 45 | 45.7 | 10 | 10.7 |

A partir de estos resultados se observa que existe un error de calibración promedio es del 0.5%, calculado de la siguiente manera:

- Para cada uno de los valores obtenidos se calcula su error porcentual.
- Posterior a ello, se promedia cada uno de los errores porcentuales calculados, se promedian y se obtiene un error porcentual medio.

**Sensor - Y**

$$\%Error = \left|\frac{179.9 - 180}{180}\right| \times 100\% = 0.05\% \quad (15)$$

$$\%Error = \left|\frac{171.8 - 170}{170}\right| \times 100\% = 1.05\% \quad (16)$$

$$\%Error = \left|\frac{90.6 - 90}{90}\right| \times 100\% = 0.67\% \quad (17)$$

$$\%Error = \left|\frac{44.98 - 45}{45}\right| \times 100\% = 0.04\% \quad (18)$$

$$Promedio\ Error\ Porcentual = 0.45\%$$

**Sensor - X**

$$\%Error = \left|\frac{89.6 - 90}{90}\right| \times 100\% = 0.44\% \quad (19)$$

$$\%Error = \left|\frac{59.9 - 60}{60}\right| \times 100\% = 0.17\% \quad (20)$$

$$\%Error = \left|\frac{44.6 - 45}{45}\right| \times 100\% = 0.89\% \quad (21)$$

$$\%Error = \left|\frac{10.7 - 10}{10}\right| \times 100\% = 0.7\% \quad (22)$$

$$Promedio\ Error\ Porcentual = 0.55\%$$

A partir de los resultados obtenidos se puede concluir que la calibración de los sensores respecto a los ejes de inclinación en “X” y “Y”, tienen un alta precisión con una tolerancia por medición de ±0,5°, según los obtenido por medio del promedio de Error porcentual y el análisis de las tablas de calibración obtenidas.

Por otra parte, durante la calibración de los sensores, se encuentra que no es posible calibrar de forma correcta el eje z, debido a que los muelles del sistema electromecánico del sensor no se elongan ni reciben ninguna excitación por la posición física en la que se encuentra, es decir, que al estar a 90° en el eje y, se debe girar el sensor sobre ese mismo eje para encontrar algún valor en z, pero durante este movimiento la masa del sistema electromecánico se mantiene en la misma posición, y de esta manera los muelles se mantienen igual o varían abruptamente,

generando que el sensor no perciba ningún cambio en el movimiento o genere saltos muy grandes entre cada uno.

### 5.2.2 Captura de información del sensor inercial en movimientos de hombros y cadera en posición de sedestación

Después de validar la calibración de los sensores e identificar cómo se relacionan los movimientos de yaw, roll y pitch del sensor inercial con los movimientos de los hombros y cadera en posición de sedestación, se pasa a ubicar los sensores con cinta en la posición deseada para observar el comportamiento de los ángulos de inclinación en cada una de las posturas, de la siguiente manera:

*Figura 58,Sensorica en postura correcta [Propiedad de los autores].*

En la imagen se observa una postura correcta, donde se identifica que en el eje x, y de los sensores el ángulo oscila entre los 90°. Al ubicar los hombros en posición de abducción, se encuentra que el eje y de los sensores de los omoplatos aumenta su posición angular dependiendo de qué tan inclinados se encuentren los hombros, mientras que en el eje x los ángulos de inclinación siguen oscilando entre los 90°.

*Figura 59.Sensorica en abducción de hombros [Propiedad de los autores].*

Cuando se eleva un hombro o una cadera, el eje x de los sensores aumenta o disminuye dependiendo del hombro o cadera que se incline. En la imagen 59 se observa una inclinación lateral de cadera, esto se traduce a que el ángulo en el eje x del sensor de la cadera inclinada disminuye, mientras que el sensor de la cadera que se encuentra en posición correcta se mantiene alrededor de los 90°.

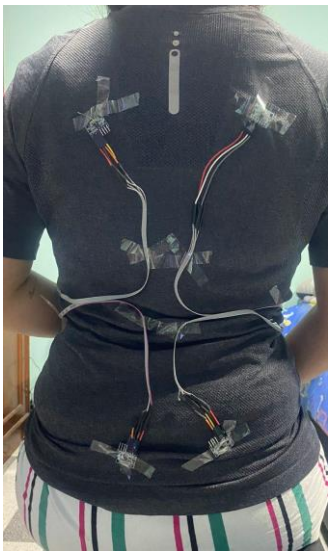


*Figura 60 .Sensorica en inclinación lateral de cadera [Propiedad de los autores].*

En el caso de los hombros, el dispositivo funciona de la misma manera, al tener el hombro izquierdo elevado y el derecho en descenso, el sensor posicionado en el hombro izquierdo

aumenta significativamente, mientras que el sensor del hombro derecho oscila alrededor de los disminuye.

*Figura 61.Sensorica en elevación y descenso de hombros [Propiedad de los autores].*

### 5.3 Implementación del dispositivo electrónico

Después de calibrar los sensores e identificar la correcta captura de los ángulos de inclinación de los omoplatos y las crestas iliacas se ubican los sensores y la circuitería en la prenda. En la parte posterior se ubican los sensores en cada carril diseñado para cada una de las articulaciones, y los demás componentes son ubicados en la parte delantera de la prenda en el compartimiento diseñado. Los sensores están conectados al multiplexor por medio de cable tipo ribbon AWG 28 de 4 hilos, este cable es ideal para el dispositivo debido a que es plano y a la hora de usar la prenda no es perceptible por el usuario, además se manejan distancias aproximadamente de 80 cm entre el sensor y el multiplexor, sin generar capacitancias debido a la longitud corta y el diámetro reducido del cable.

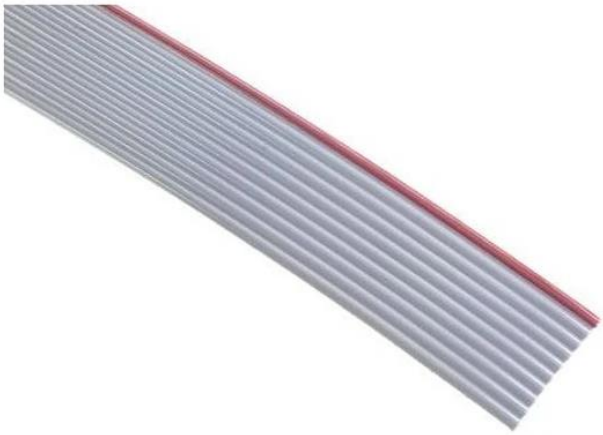


*Figura 62. Cable ribbon AWG 10 Hilos [55].*

Al validar el funcionamiento del dispositivo se observa que las mediciones no se ven alteradas por la integración de la prenda, y el funcionamiento eléctrico no se ve alterado en ningún momento, debido al aislamiento eléctrico que posee la tela, protegiendo al usuario y al dispositivo,  además el cable no genera ningún tipo de ruido, ni capacitancia,

Para realizar el desplazamiento de la sensórica según la estructura ósea del usuario se integran dos correas a cada uno de los sensores, las cuales tienen el objetivo de reubicar el sensor dependiendo del sentido en el que el usuario hala la correa, también los carriles de la prenda, limitan el espacio donde el sensor se debe mover, aumentando las probabilidades de una correcta ubicación del sensor, cabe resaltar que la ubicación del sensor depende del usuario, debido a que no se mide la ubicación lineal de cada sensor en la prenda, solo se cuenta con el sistema de desplazamiento.Es importante tener presente que la ubicación de los sensores como se ha mencionado anteriormente se ha identificado gracias a la literatura y a la ayuda del fisioterapeuta especializado en ergonomía.

*Figura 63. Posicionamiento de correas en prenda[Propiedad de los autores].*

## 5.4 Desarrollo del software para dispositivos móviles, capaz de monitorear y almacenar los datos recolectados

### 5.4.1 Desarrollo de la aplicación móvil y conexión Firebase

Para validar la conexión entre la aplicación móvil y los servicios de Firebase Authentication y Cloud Firestore, se comienza revisando solo el enlace entre el servicio Authentication y Android Studio, pero, al ejecutar la aplicación por primera vez, se observa que no permite realizar la sincronización, ya que arroja un error con la versión del servicio SDK(servicio que permite el ingreso de librerías a Android Studio). Por tal motivo, la primera opción es modificar la librería principal de Firebase y utilizar la última desarrollada, en la cual no es necesario especificar las versiones de las librerías, ya que la librería principal de Firebase abarca las versiones pasadas y actuales.

A pesar de que se logra resolver la primera dificultad, al momento de sincronizar el proyecto nuevamente, se presenta el mismo error, por lo cual en la aplicación de Android Studio se valida en: herramientas y Firebase la conexión con el servicio, observando que si se encuentra y además es correcta.

Como el error se sigue presentado, se valida en la documentación publicada por el proveedor de Android Studio, observando que la versión del servicio SDK de Android se configura

automáticamente en 21, debido a que es la version mas comun para implementar proyectos en Android, por tal motivo, dicho servicio no acepta librerías superiores a la 21 y menos de un servicio externo a los determinados por Android Studio. Por tal motivo, es necesario modificar la versión del servicio SDK de Android Studio para que sea compatible con las librerías de Firebase, cambiando la versión SDK de 21 a 23 dentro del archivo Gradle a nivel de módulo dentro de Android Studio.

Adicionalmente, es importante validar las versiones individuales de cada una de las librerías a importar, ya que, aún teniendo el SDK de Android Studio en una versión compatible con Firebase, se sigue presentando el mismo error de SDK al sincronizar el proyecto. Por tal motivo, es necesario remitirse a la documentación publicada por Firebase, repositorios y canal de preguntas de la plataforma, encontrando así que, además de modificar la versión del servicio SDK de Android Studio, es necesario modificar la versión de cada una de las librerías, aún sí se implementa una librería que en teoría abarca cualquiera de las versiones actuales y antiguas. Por tal motivo, se modifican las versiones de las librerías de acuerdo a la información publicada por el servicio de Firebase Authentication.

Al momento de sincronizar el proyecto se observa que la conexión finalmente es exitosa y permite validar en la ventana de Firebase Authentication el enlace entre la aplicación y Firebase. No obstante, al momento de ingresar un usuario y presionar el botón de acceder o registrar trae siempre la ventana de error informando que algo ha salido mal, como se observa en la siguiente figura:

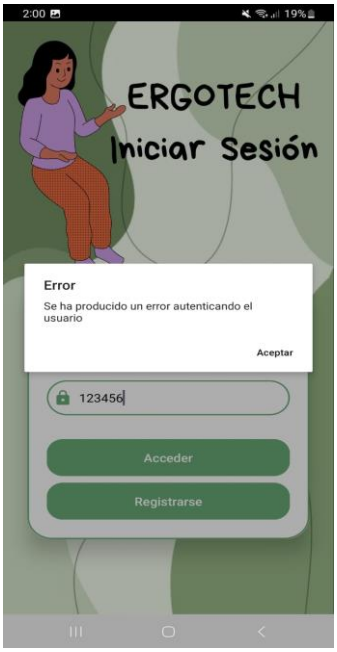



*Figura 64.Notificación de error Ventana de Login [Propiedad de los autores].*

Para solucionar el inconveniente, es necesario crear una variable global e inicializarla, la cual permite traer la librería para así poderla implementar en la vista correspondiente al login, ya que, es donde se utiliza el servicio de Authentication. Se observa, que este error se presenta en la última versión de Android y en las aplicaciones desarrolladas en el lenguaje de programación Kotlin, ya que, las actualizaciones recientes requieren declarar las variables de los servicios externos a Android Studio.

Finalmente, la aplicación permite el logueo del usuario y la conexión entre Android Studio y Firebase Authentication, como se observa en la figura 62, de modo que al ingresar los datos de correo y contraseña y presionar el botón de acceso si el usuario no se ha registrado anteriormente se notifica por medio de un banner que no se puede realizar dicho proceso. Si se presiona el botón de registrar y no se encuentra registrado automáticamente crea el usuario en Firebase, como se observa en la figura 63 y le permite el acceso a la aplicación. Si el usuario presiona el botón de acceder y ya se ha registrado anteriormente puede ingresar a la aplicación y por último si el usuario ya se encuentra registrado y digita mal la contraseña genera una notificación informando que se presenta un error en la autenticación.

*Figura 65.Ventana de Login [Propiedad de los autores].*

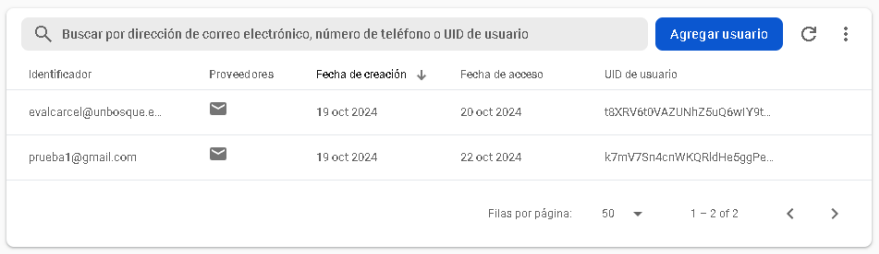



*Figura 66. Funcionamiento conexión Android Studio y Firebase Authentication [Propiedad de los autores].*

Después de validar la conexión y el funcionamiento entre Android Studio y el servicio Firebase Authentication se pasa a sincronizar el proyecto de Android Studio con el servicio Cloud Firestore, el cual logra realizar la conexión automáticamente, debido a que ya se ha solucionado cualquier posible inconveniente en la conexión hacia Firebase.

La aplicación permite guardar y recuperar tanto la información personal como la información médica del usuario de la base de datos. De modo que, las dos ventanas traen automáticamente el correo del usuario que se loguea, esto, ya que en la base de datos es necesario utilizar una llave primaria, la cual en este caso corresponde al correo, para así, poder realizar las consultas, ya sea para guardar, actualizar o recuperar información relacionadas al usuario.

En el caso de la ventana de información personal, los campos de: edad, peso y altura permiten el ingreso de datos, de modo que, al presionar el botón de guardar la información diligenciada se almacena en la base de datos en variables individuales y de tipo float, dicha información se relaciona en la tabla de "usuarios" y se almacena en el respectivo usuario que se loguea, como se observa en la siguiente figura:

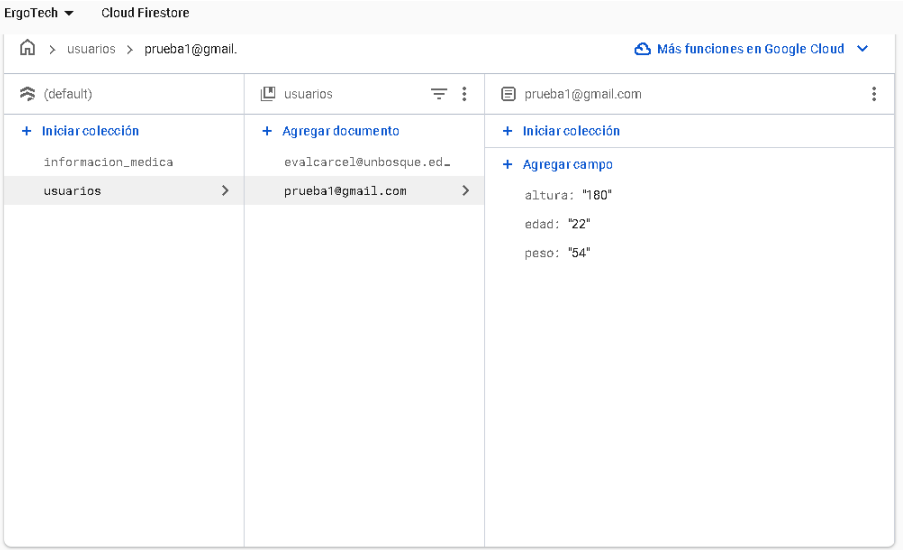



*Figura 67 . Funcionamiento conexión Android Studio y Cloud Firestore, tabla de usuarios [Propiedad de los autores].*

Por otra parte, el botón de recuperar trae la información de: edad, peso y altura relacionada al usuario que ingresa a la aplicación, como se observa en la siguiente figura:

*Figura 68.Ventana de Información personal [Propiedad de los autores].*

En el caso de la ventana de información médica, los campos de los ejes de inclinación en x, y y z para cada articulación permiten el ingreso de datos, de modo que, al presionar el botón de guardar, la información en la base de datos se almacena en variables individuales y de tipo float, donde se relaciona en la tabla de "información médica", como se observa en la siguiente figura:

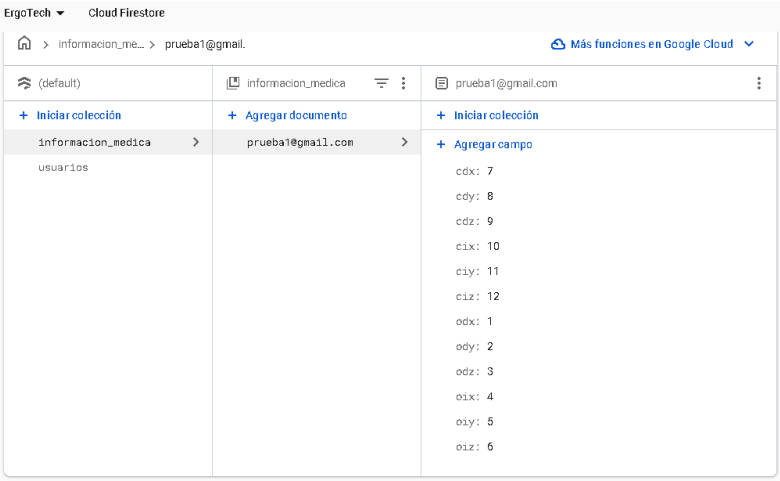



*Figura 69. Funcionamiento conexión Android Studio y Cloud Firestore, tabla de información médica [Propiedad de los autores].*

Por otra parte, el botón de recuperar trae la información de los ejes de inclinación en x, y, z para cada articulación relacionada al usuario que ingreso a la aplicación, como se observa en la siguiente figura:

*Figura 70 .Ventana de Información médica [Propiedad de los autores].*

En el caso de la ventana de información médica, es necesario realizar un modificación interna en el código, para evitar conflictos durante la recuperación de datos, ya que, en la base de datos, la información se almacena en variables de tipo float, mientras que para visualizar la información en los campos de texto (edittext) se requieren variables de tipo string. Por lo tanto, es necesario transformar las variables de tipo float a string, para así, poder mostrar correctamente la información almacenada.

Esta conversión no sólo permite la visualización de la información, sino que también simplifica el proceso de comparación entre los datos leídos por el sensor y los datos médicos almacenados en la base de datos, ya que, al tener los valores médicos en variables tipo float, se optimiza el desarrollo de las funciones de comparación, asegurando una integración eficiente de los datos.

### 5.4.2 Aplicación móvil y conexión bluetooth con el dispositivo

Para poder visualizar la información recolectada por los sensores inerciales es necesario conectar la aplicación móvil con el bluetooth de la tarjeta de programación ESP32. En primera instancia, se prueba solamente el funcionamiento del bluetooth de la tarjeta, para corroborar que no presente ningún inconveniente en la conexión, utilizando Arduino Bluetooth Controller (extension de Arduino) el cual permite tener desde el celular varias funcionalidades del software Arduino que se instala directamente en el PC, como lo es el Puerto Serial, donde se pueden observar los datos que está recibiendo la tarjeta.

Para que la tarjeta active la funcionalidad del Bluetooth es necesario adjuntar al código de Arduino las siguientes librerías:

```
#include "BluetoothSerial.h"
#if !defined(CONFIG_BT_ENABLED) || !defined(CONFIG_BLUEDROID_ENABLED)
#error Bluetooth is not enabled! Please run `make menuconfig` to and enable it
#endif
BluetoothSerial SerialBT;
```

*Figura 71Librerías bluetooth Arduino [Propiedad de los autores].*

Además, de modificar todos los comandos de impresión de información en el serial: Serial.print a SerialBT. print, para que así en el Puerto Serial remoto se imprima toda la información requerida.

Después de corroborar que el Bluetooth de la tarjeta de programación funciona correctamente, es necesario desarrollar el código en Android Studio para activar la funcionalidad dentro de la aplicación móvil. Por lo cual, se inicia con la modificación en el fronted de la ventana, implementando un listview para traer en forma de lista todas las conexiones bluetooth que se encuentren disponibles, como se observa en la figura 69, además de configurar el listview para que al vincularse con un dispositivo finalice la tarea y muestre inmediatamente la vista de la ventana de monitoreo.

*Figura 72. Lista de conexiones bluetooth Aplicación móvil[Propiedad de los autores].*

Para la conexión entre dispositivo y aplicación se implementa una librería bluetooth dentro de la programación backend de la ventana, donde, primero se notifica al usuario la activación bluetooth por medio de un banner interactivo, que al presionarlo permite activar el servicio de bluetooth del dispositivo móvil donde se encuentra descargada la aplicación, como se observa en la figura 70.

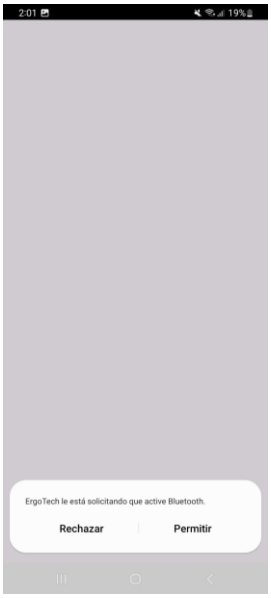



*Figura 73.Notificación de conexión bluetooth [Propiedad de los autores].*

Cuando se realiza la activación del bluetooth, automáticamente se despliega la lista de dispositivos conectados, la cual funciona como un listado de botones que al presionarlos realizan la vinculación con el dispositivo seleccionado, donde, en la parte inferior de la pantalla se puede observar una pequeña notificación de Android Studio informando si la conexión está en proceso, fue exitosa o fallo, como se observa en la figura 71. Si la conexión falla, automáticamente regresa a la ventana de conexión bluetooth, con el listado de dispositivos disponibles para la vinculación, pero si la conexión es exitosa el listview finaliza y desaparece, dando visibilidad del frontend de la pantalla de monitoreo.

*Figura 74 .Notificación de vinculación de dispositivos bluetooth rechazada [Propiedad de los autores].*

### 5.4.3 Aplicación móvil y monitoreo

El diseño de la ventana de monitoreo carga la información médica almacenada en la base de datos y la información detectada por la sensorica. Al iniciar el proceso se observan datos en la parte izquierda, derecha y central de la ventana. Donde, en la parte izquierda se presentan los siguientes datos en respectivo orden: valores en el eje x, y, z del omóplato izquierdo captados por los sensores, valores en el eje x, y, z del omóplato izquierdo almacenados en la base de datos, valores en el eje x, y, z de la cresta iliaca izquierda captados por los sensores, valores en el eje x, y, z de la cresta iliaca izquierda almacenados en la base de datos (los datos captados por los sensores se encuentran de color verde y los datos medicos almacenados en la base de datos se encuentran en color negro).

En la parte derecha se presentan los siguientes datos en respectivo orden: valores en el eje x, y, z del omóplato derecho captados por los sensores, valores en el eje x, y, z del omóplato derecho almacenados en la base de datos, valores en el eje x, y, z de la cresta iliaca derecha captados por los sensores, valores en el eje x, y, z de la cresta iliaca derecha almacenados en la base de datos (los datos captados por los sensores se encuentran de color verde y los datos medicos almacenados en la base de datos se encuentran en color negro).

En la parte central se observa primero la sección de notificación de postura, la cual se activa si el usuario se encuentra en una mala postura, trayendo información de la articulación y el eje que están en una mala posición, como se observa en la siguiente figura:

*Figura 75.Ventana de Monitoreo sin notificaciones [Propiedad de los autores].*

Para identificar si es una postura correcta o incorrecta, el código de programación backend trae los valores almacenados en la base de datos y los valores captados por la sensórica y las almacena en variables individuales, donde, se convierten los valores de tipo string a float para realizar las respectivas operaciones. Para esto es necesario tener en cuenta 3 etapas:

La primera, donde se realiza la conversión de las variables. Esto es necesario, ya que, al efectuar las comparaciones con los datos como strings y se obtiene un análisis incorrecto, debido a que, el sistema comparaba la extensión del string mas no el valor, por tal motivo, es necesario modificar los datos de tipo string a datos tipo float para así realizar comparaciones entre sus valores y no entre la extensión.

La segunda etapa, donde se realiza un almacenamiento individual de variables, esto, debido a que la lectura que realizan los sensores se visualiza en un solo texto editable para cada una de las articulaciones, trayendo la información de los ejes x, y, z de cada articulación de manera grupal y no individual como en la venta de información médica. En este caso es importante realizar la separación de datos, ya que, la comparación entre información médica e información captada por el dispositivo se debe realizar de manera individual entre cada eje de inclinación para identificar la correcta o incorrecta postura del usuario.

Para realizar la separación de la información de los ejes x, y, z de cada articulación, se modifica el código de Arduino agregando un signo para separar la información leída en cada ángulo de inclinación, como se observa en la figura 73, para, posteriormente en Android Studio por medio de un arreglo, separar los datos, identificando que después de cada signo se debe almacenar el valor leído en una variable individual.

```
// Ompoplato derecho
SerialBT.print(angulox);
SerialBT.print("|");
SerialBT.print(anguloy);
SerialBT.print("|");
SerialBT.println(anguloz);
```

*Figura 76 .Configuración bluetooth SerialPrint Arduino [Propiedad de los autores].*

En la tercera etapa se realizan las comparaciones entre los valores médicos que corresponden a los ángulos de inclinación máximos que puede tener una articulación, respecto a los valores captados por el sensor, que corresponden a los ángulos de inclinación en tiempo real realizados por el usuario. Es importante tener presente que los ángulos de inclinación máximos son diferentes entre los usuarios, ya que, estos, permiten identificar hasta donde se puede mover la articulación y aún así seguir teniendo una postura correcta.

Las comparaciones realizadas analizan que: si el valor captado por el sensor supera al valor determinado por el médico especialista en cada uno de los ejes de inclinación de las articulaciones se genera una alerta informando una mala postura, la articulación que se encuentra en un posición incorrecta y en qué eje de inclinación se presenta. Adicionalmente,

se implementa una herramienta visual, cambiando el color del texto a rojo en la articulación que presenta una mala postura, para que así sea más sencillo para el usuario observar cuál es la articulación afectada, como se observa en la siguiente figura:

*Figura 77.Ventana de Monitoreo con notificaciones [Propiedad de los autores].*

Otra de las comparaciones realizadas corresponde a la simetría, donde, en uno de los visualizadores de texto se puede observar si las articulaciones de los omoplatos y las crestas iliacas se encuentran simétricas. En este caso, las comparaciones se realizan dentro del código de Arduino enviando a la aplicación un texto informando si existe o no simetría, para esto, se comparan los ejes x, y del omóplato derecho e izquierdo y los ejes x, y de la cresta iliaca izquierda y derecha, notificando una simetría cuando los valores en el eje x o y de la articulación de los hombros o la cadera tiene el mismo valor y notificando una asimetría cuando los valores son diferentes.

### 5.5 Integración y evaluación del dispositivo electrónico y el software

Después de validar el funcionamiento del sistema electrónico, la aplicación móvil y la base de datos individualmente se procede a realizar pruebas con la integración completa del dispositivo. Donde, el sistema electrónico ya se encuentra ubicado sobre la prenda de vestir (camiseta) y es ajustable gracias a las correas frontales que permiten mover los sensores por los carriles diseñados y así ubicarlos en los omoplatos y las crestas iliacas según la morfología del usuario.

Se inicia una prueba completa, desde el logueo del usuario, el ingreso y almacenamiento de información personal e información médica y el respectivo monitoreo, para este último se presiona el botón de inicio del dispositivo electrónico, el cual se encuentra en la parte frontal de la camiseta para así encender el bluetooth del dispositivo y poderlo vincular a la aplicación móvil. Después de la vinculación se abre inmediatamente la ventana de monitoreo donde se observan los datos médicos almacenados en la base de datos y los datos captados por el dispositivo electrónico.

Para validar si la aplicación identifica una correcta o mala postura se efectúan varias pruebas donde el usuario realiza diferentes movimientos mientras se encuentra sentado, ubicando las articulaciones en las siguientes posiciones:

- Postura correcta:

*Figura 78.Postura correcta [Propiedad de los autores].*

- Abducción de hombros:

*Figura 79 .Abducción de hombros[Propiedad de los autores].*

- Elevación y descenso de hombros:

*Figura 80 .Elevación y descenso de hombros [Propiedad de los autores].*

Para realizar las respectivas comparaciones, se ingresa en la sección de información médica de la aplicación que: en cualquiera de los ejes x, y, z de las 4 articulaciones el ángulo de inclinación máximo debe ser de 45°. Esto significa, que si alguna articulación supera los 45° de inclinación, se considera que el usuario se encuentra en una posición incorrecta y por tal motivo debe ser notificado en la aplicación, como se observa en la siguiente figura:

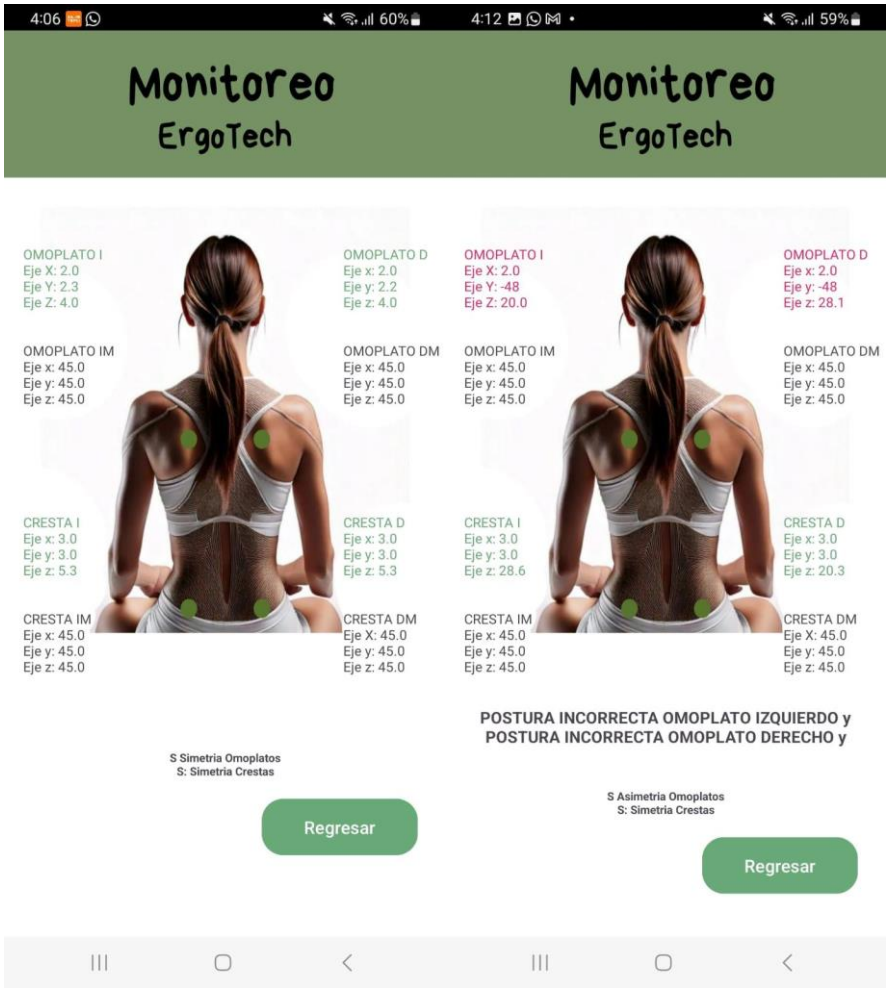



*Figura 81 .Ventana de Monitoreo integrada con dispositivo electrónico [Propiedad de los autores].*

Se realiza una tabla con la información recolectada por el dispositivo electrónico y la aplicación móvil, donde se observan los ángulos de inclinación máximos, la postura en la que se encuentra el usuario, los ángulos captados por el dispositivo electrónico y la notificación de la aplicación móvil.

Es importante tener presente que los ángulos máximos de inclinación que se ingresan en la sección información médica son ángulos que funcionan tanto para valores positivos como negativos, por tal motivo, si en la lectura de los sensores se observan valores de ángulos negativos hace referencia a que el movimiento se está realizando hacia atrás, izquierda, o abajo según sea el eje respecto al cual se está realizando la rotación.

*Tabla 11. Integración de dispositivo electrónico y aplicación móvil.*

| **Ángulos de inclinación máximos ingresados a la sección "Información médica"** | **Postura de usuario** | **Ángulos de inclinación captados por el dispositivo electrónico** | **Notificación aplicación móvil** |
|---|---|---|---|
| Para las 4 articulaciones<br>Eje x: 45<br>Eje y: 45<br>Eje z: 45 | Postura correcta | Omóplato derecho<br>Eje x: 2<br>Eje y: 2<br>Eje z: 4<br><br>Omóplato Izquierdo<br>Eje x: 2<br>Eje y: 2<br>Eje z: 4<br><br>Cresta derecha<br>Eje x: 3<br>Eje y: 3<br>Eje z: 5<br><br>Cresta izquierda<br>Eje x: 3<br>Eje y: 3<br>Eje z: 5 | Nada (significa que la postura es correcta) |
| Para las 4 articulaciones<br>Eje x: 45<br>Eje y: 45<br>Eje z: 45 | Aducción de hombros | Omoplato derecho<br>Eje x: 2<br>Eje y: -48<br>Eje z: 20<br><br>Omoplato izquierdo<br>Eje x: 2<br>Eje y: -48<br>Eje z: 28<br><br>Cresta derecha<br>Eje x: 3<br>Eje y: 3<br>Eje z: 28<br><br>Cresta izquierda<br>Eje x: 3<br>Eje y: 3<br>Eje z: 20 | Postura incorrecta omoplato derecho y omóplato izquierdo<br>y<br>Asimetría omoplatos |
| Para las 4 articulaciones<br><br>Eje x: 45<br>Eje y: 45<br>Eje z: 45 | Elevación y descenso de hombros | Omoplato derecho<br>Eje x: 49<br>Eje y: 2<br>Eje z: 30<br><br>Omoplato izquierdo<br>Eje x: -49<br>Eje y: 2<br>Eje z: 32<br><br>Cresta derecha<br>Eje x: 13<br>Eje y: 3<br>Eje z: 34<br><br>Cresta izquierda<br>Eje x: -7<br>Eje y: 3<br>Eje z: 32 | Postura incorrecta omoplato derecho x omoplato izquierdo<br>x<br>Asimetría omoplatos |

| | | | |
|---|---|---|---|
| Para las 4 articulaciones<br><br>Eje x: 45<br>Eje y: 45<br>Eje z: 45 | Rotación interna y externa de hombros | Omóplato derecho<br>Eje x: 2<br>Eje y: 2<br>Eje z: 46<br><br>Omóplato Izquierdo<br>Eje x: 2<br>Eje y: 2<br>Eje z: 52<br><br>Cresta derecha<br>Eje x: 3<br>Eje y: 3<br>Eje z: -46<br><br>Cresta izquierda<br>Eje x: 3<br>Eje y: 3<br>Eje z: 53 | Postura incorrecta omoplato derecho z omoplato izquierdo z<br><br>cresta derecha z<br>cresta izquierda z |
| Para las 4 articulaciones<br><br>Eje x: 45<br>Eje y: 45<br>Eje z: 45 | Inclinación lateral interna e inclinación lateral externa de cadera | Omóplato derecho<br>Eje x: 2<br>Eje y: 13<br>Eje z: 50<br><br>Omóplato Izquierdo<br>Eje x: 2<br>Eje y: -7<br>Eje z: 58<br><br>Cresta derecha<br>Eje x: 3<br>Eje y: 46<br>Eje z: 50<br><br>Cresta izquierda<br>Eje x: 3<br>Eje y: -46<br>Eje z: 58 | Postura incorrecta<br>cresta derecha y<br>cresta izquierda y Asimetría crestas<br><br>omoplato derecho z omoplato izquierdo z<br><br>cresta derecha z<br>cresta izquierda z |
| Para las 4 articulaciones<br><br>Eje x: 45<br>Eje y: 45<br>Eje z: 45 | Rotación externa y rotación interna de la cadera | Omóplato derecho<br>Eje x: 2<br>Eje y: 2<br>Eje z: 60<br><br>Omóplato Izquierdo<br>Eje x: 2<br>Eje y: 2<br>Eje z: 69<br><br>Cresta derecha<br>Eje x: 3<br>Eje y: 3<br>Eje z: -60<br><br>Cresta izquierda<br>Eje x: 3<br>Eje y: 3<br>Eje z: 72 | Postura incorrecta<br>cresta derecha z<br>cresta izquierda z<br><br>omoplato derecho z omoplato izquierdo z |

| | | | |
|---|---|---|---|
| Para las 4 articulaciones<br><br>Eje x: 45<br>Eje y: 45<br>Eje z: 45 | Abducción de cadera | Omóplato derecho<br>Eje x: 2<br>Eje y: 2<br>Eje z: 63<br><br>Omóplato Izquierdo<br>Eje x: 2<br>Eje y: 2<br>Eje z: 65<br><br>Cresta derecha<br>Eje x: 46<br>Eje y: 3<br>Eje z: 63<br><br>Cresta izquierda<br>Eje x: 46<br>Eje y: 3<br>Eje z: 66 | Postura incorrecta<br>cresta derecha x<br>cresta izquierda x Asimetría crestas<br><br>omoplato derecho z omoplato izquierdo<br><br>cresta derecha z<br>cresta izquierda z |

Se observa que la ventana de monitoreo de la aplicación móvil cuenta con un retardo de ±4 segundos en la lectura de la información enviada desde el dispositivo. Esta demora se debe al procesamiento de datos en Android Studio, el cual se ve afectado por la cantidad de información que debe analizarse, tanto de la base de datos como del dispositivo electrónico.

Aún así, la aplicación realiza correctamente la comparación entre los datos médicos ingresados a la aplicación móvil y los datos captados por el dispositivo electrónico en los ejes x y y, permitiendo identificar si el usuario se encuentra en una correcta o incorrecta postura e informando la articulación y el eje de inclinación en los cuales se está tomando una posición incorrecta. Adicional, notificando si hay simetría o no entre las articulaciones de los hombros y la cadera.

Por otra parte, el eje z presenta errores en los valores captados por el dispositivo electrónico, debido a varios inconvenientes durante la calibración. Estos incluyen ruidos y errores en las mediciones, así como problemas relacionados con la orientación y el marco de referencia de los sensores inerciales. Debido a que estos sensores miden la aceleración y la rotación en relación con un punto de referencia, produciendo que cualquier inestabilidad en el sensor dificulte la obtención correcta de medidas. Además, el eje z es uno de los más sensibles, ya que los movimientos en el eje x y y afectan directamente su funcionamiento.

A pesar de la lectura incorrecta de los ángulos de inclinación en el eje z, se observa que la aplicación realiza correctamente la comparación entre los ángulos de inclinación ingresados a la aplicación y los ángulos de inclinación medidos, ya que, si los valores ingresados superan a los medidos arroja inmediatamente la notificación informando que hay una mala postura y notificando la articulación y el eje correspondiente.

## 5.6 Cumplimiento de requerimientos

### 5.6.1 Requerimientos funcionales

*Tabla 12. Cumplimiento de requerimientos funcionales*

| Requerimiento | Cumple | Observaciones |
|---|---|---|
| El dispositivo electrónico medirá el ángulo de inclinación de cada hombro de manera independiente en los ejes x,y y z, teniendo en cuenta los movimientos de: elevación, descenso, abducción, aducción, rotación externa e interna del hombro a mayor amplitud, en un rango de: 0° a 180°. | SI | Se selecciono el sensor MPU6050 que tiene la capacidad de medir los tres ejes en el rango requerido. (ver sección 4.5. Selección de componentes principales) |
| El dispositivo electrónico medirá el ángulo de inclinación de la cadera en los ejes x,y y z teniendo en cuenta los movimientos de anteversión, retroversión, inclinación lateral externa, inclinación lateral interna, rotación interna y rotación externa, en un rango respectivamente de 0° a 90°. | SI | Se selecciono el sensor MPU6050 que tiene la capacidad de medir los tres ejes en el rango requerido. (ver sección 4.5. Selección de componentes principales) |
| El dispositivo electrónico medirá la simetría entre los omoplatos. | SI | Se comparan las posiciones angulares en el eje x de los sensores ubicados en los omoplatos, si este valor es igual o no difieren en más de 3° se deduce que hay simetría, en caso contrario, no hay simetría |
| El dispositivo electrónico medirá la simetría entre los iliacos. | SI | Se comparan las posiciones angulares en el eje x de los sensores ubicados en los iliacos, si este valor es igual o no difieren en más de 3° se deduce que hay simetría, en caso contrario, no hay simetría |
| En la aplicación móvil se visualizará la postura del usuario. | SI | Se desarrolla una aplicación que contiene login de usuario, visualización de postura, manual de usuario, información personal e información médica (Ver sección 5.4. Desarrollo del software para dispositivos móviles, capaz de monitorear y almacenar los datos recolectados) |
| La aplicación móvil procesará y analizará los valores de ángulo y simetría recolectados por el dispositivo electrónico y por el médico especialista, para identificar la correcta postura del usuario. | SI | Se desarrolla una aplicación que procesa y analiza información de los sensores (5.5. Integración y evaluación del dispositivo electrónico y el software) |

| | | |
|---|---|---|
| La aplicación móvil almacenará la información recolectada por medio de la implementación de una base de datos. | SI | Se desarrolla una aplicación que almacena información personal e información médica (Ver sección 5.4. Desarrollo del software para dispositivos móviles, capaz de monitorear y almacenar los datos recolectados) |
| La aplicación móvil enviará notificaciones de alerta de mala postura (los valores de mala postura varían según el usuario). | SI | Se desarrolla una aplicación que muestra alertas de mala postura (Ver sección 5.4. Desarrollo del software para dispositivos móviles, capaz de monitorear y almacenar los datos recolectados) |
| El dispositivo tendrá un sistema de detección de fallas, el cual alertará al usuario de posibles desconexiones en el circuito, batería baja y mala posición de sensores. | NO | Aunque cuenta con un sistema de detección de conexión de sensores por medio de la programación, no es posible alertar al usuario sobre desconexiones, ni mala posición de sensores, ya que esto es directamente controlado por el usuario dependiendo del correcto uso que se le de al dispositivo. |

### 5.6.2 Requerimientos de calidad

*Tabla 13. Cumplimiento de Requerimientos de calidad*

| Requerimiento | Cumple | Observaciones |
|---|---|---|
| Los valores de ángulo recolectados por el dispositivo electrónico tendrán un error máximo del ±5°. | SI | A partir del calculo del error se dedujo que el promedio de error es de ±0.5°. (Ver sección 5.2.1. Validación calibración sensores inerciales) |
| El arnés será adaptable para una estructura ósea mediana masculina y femenina con las siguientes dimensiones: 93cm a 103cm pecho, 100cm a 110 cadera y 56 cm a 66 cm altura de hombros a cadera. | SI | El arnés o prenda de vestir contiene una sistema de correas que permite adaptar el dispositivo a las estructuras Oseas (Ver sección 5.3. Implementación del dispositivo electrónico) |
| El sistema electrónico deberá incluir manual de usuario, donde se explique la manera correcta de usar el dispositivo y la aplicación, este se obtendrá a partir de la generación de protocolos de uso, conexión e indicadores de fallas, a partir de pruebas del prototipo en voluntarios reales; Estos voluntarios deben tener edades entre los 18 y 50 años con estructuras osteomusculares medianas. | SI | Se incluye manual en anexo de este proyecto, además se puede consultar en el aplicativo móvil en la opción de manual de usuario. |
| Los sensores de ángulo tendrán una dimensión máxima de 3 cm x 3 cm. | SI | Las dimensiones del sensor MPU6050 son de 1.5 cm x 2.5 cm |

### 5.6.3 Requerimientos restrictivos

*Tabla 14. Cumplimiento de requerimientos restictivos*

| Requerimiento | Cumple | Observaciones |
|---|---|---|
| El dispositivo electrónico deberá tener un soporte de batería. | SI | El dispositivo cuenta con una caja que contiene todo el circuito. |

| | | |
|---|---|---|
| El dispositivo deberá ser alimentado por una batería recargable cuya capacidad sea suficiente para mantener un tiempo de trabajo de mínimo 5 horas. | SI | El dispositivo cuenta con una batería recargable de 3.7V 2500mAh suficiente para el tiempo de trabajo de 5 horas (Ver sección 4.5. Selección de componentes principales – Batería) |
| El dispositivo deberá tener un microcontrolador con capacidad mínima para procesar simultáneamente información de cuatro sensores y contar con tecnología FPC. | SI | El dispositivo cuenta con una ESP32 capaz de procesar simultáneamente información de los cuatro sensores por medio de un multiplexor y el protocolo I2C. (Ver sección 4.7. Comunicación I2C, sensores inerciales y tarjeta de programación) |
| El protocolo de comunicación del dispositivo deberá ser compatible con Android. | SI | El dispositivo y aplicativo móvil se comunican por medio de bluetooth. |
| El dispositivo electrónico será diseñado para una estructura ósea mediana. | SI | La prenda de vestir es de tallaje mediano (M) y cuenta con correas que adaptan los sensores a estructuras oseas medianas. |
| El dispositivo electrónico estará diseñado con tecnología FPC (flexible Printed Circuit). | NO | Los componentes principales usados no cuentan con tecnología de circuito impreso flexible, se tendrían que diseñar e imprimir completamente partiendo de un modelo básico para cada uno de los componentes. |
| El sistema electrónico diseñado será un prototipo. | SI | |

# 6. CONCLUSIONES

- Dado que el cuerpo humano realiza cambios de postura inconscientes tras periodos prolongados de estar sentado, un sistema de monitoreo postural puede convertirse en un aliado para la concientización del ser humano y su postura, siendo así fundamental que la Medicina y la Ingeniería propendan por el desarrollo de proyectos para promover el bienestar y la mejora de la calidad de vida como lo busca el modelo biopsicosocial de la universidad.
- Además, del gran valor que tiene este trabajo a nivel humano, cabe destacar que la investigación detallada de los términos médicos relacionados con las articulaciones hombros y cadera, así como los movimientos del sensor inercial, y las rotaciones de yaw, roll y Pitch fueron fundamentales para el desarrollo del proyecto. Esta comprensión permitio identificar los movimientos que ocurren durante la sedestación y las articulaciones involucradas en esta posición, lo que facilitó la ubicación de los sensores y la identificación de la postura del usuario. Permitiendo así, poder discutir con otras disciplinas, enriqueciendo no solo el entendimiento de los términos técnicos de la Ingeniería, sino también los conceptos médicos pertinentes al proyecto.
- Los sensores inerciales presentan un margen de error de ±0.5° en las mediciones de los ángulos de inclinación en los ejes x y y, mientras que, en el eje z el error super los ±5°, superando el porcentaje de error esperado de acuerdo a los requerimientos y esto se refleja de forma aleatoria. Esto se debe a que el eje z de los sensores inerciales es el más sensible, lo que provoca que pequeños movimientos en los otros dos ejes afecten sus lecturas, igualmente, es el eje más complejo de analizar.
- Si bien se supera el margen de error los movimientos que se realizan en el eje z no son significativos, ya que involucra la rotación de estructuras óseas más robustas. Así, desde la parte biomecánica los movimientos se realizarían de forma consciente de modo que el usuario podría reconocer más fácil la postura de las articulaciones. Frente a esto, el dispositivo diseñado busca detectar los movimientos sutiles e inconscientes que muchas veces son difíciles de detectar por el usuario y generan una mala postura.
- Dado que no todos los ejes tienen el mismo nivel de importancia, se asigna un mayor peso a los ejes x, y (más del 50%). Esto se debe a que, para el diseño, es preferible

comprometer el eje z antes que los ejes x, y ya que no es posible realizar una calibración precisa de los ejes x, y sin afectar el z, y viceversa.

- El dispositivo tiene la capacidad de sensar movimientos en la articulación de la cadera en un rango de 0° a 90° y en la articulación de los hombros de 0° a 180°. Sin embargo, la biomecánica natural de la cadera humana permite un rango de movimiento de hasta 45°, mientras que en los hombros puede alcanzar los 180°. Esto permite que el dispositivo tenga un rango de movimiento superior, lo que facilita la medición y el análisis de factores que pueden influir en el rango de movimiento del usuario.
- El desarrollo de este proyecto de grado muestra la importancia de realizar un diseño amigable con el usuario y pensado para su bienestar, buscando la concientización de su postura y su propio cuerpo sin generar molestias o estrés. Esto se debe a que el sistema de notificación de mala postura no tiene alarmas ni vibraciones, pues presenta de manera gráfica la articulación que el usuario debe corregir. Además, el diseño promueve una fácil interacción con el dispositivo por medio de un frontend intuitivo.
- En conclusión este proyecto de grado nos ha permitido comprender por qué las articulaciones hombros y cadera son muy importantes en la posición de sedestación, aunque no sean las únicas presentes durante este proceso. Esto hace que el desarrollo de la tecnología presentada sea la base para futuros desarrollos de sistemas de monitoreo o corrección postural más complejos, donde se analice la estructura ósea completa que interviene cuando una persona se encuentra sentada. A continuación se presentan algunas recomendaciones.

# 7. RECOMENDACIONES

- Se recomienda mejorar el dispositivo mediante el análisis de articulaciones adicionales que intervienen en la posición de sedestación, como el cuello y las piernas. Esto permitirá desarrollar un dispositivo que abarque de manera integral todas las articulaciones que se involucran cuando una persona permanece sentada.

- Además, se recomienda implementar sensores para medir la simetría entre las articulaciones. En este caso, la simetría se evaluó de forma comparativa a través del código de programación. Sin embargo, para obtener mediciones más precisas, sería beneficioso incorporar sensores de distancia o fibras ópticas y emisores de luz, lo que generaría datos más reales y completos sobre la simetría del dispositivo.

- Se recomienda realizar la investigación de la tecnología FPC (Flexible Printed Circuit) y microelectrónica con el fin de implementar el dispositivo a escala reducida, y así disminuir el consumo de energía y materiales, ahorrar espacio y tener mayor portabilidad del dispositivo.

- Se recomienda implementar la autenticación a través de cuentas de Google, Facebook o Instagram, ya que, esto permite a los usuarios iniciar sesión de manera más rápida y recibir notificaciones en dichas aplicaciones. Adicionalmente, se sugiere integrar otros servicios de Firebase, como Google Analytics, el cual proporciona un análisis en tiempo real de los distintos servicios utilizados por los usuarios, incluyendo el tiempo que pasan en la aplicación, el tiempo de ingreso que realizan, cuántos usuarios ingresan, la cantidad de información que se almacena, entre otros.

## 8. REFERENCIAS BIBLIOGRÁFICAS

[1] M. G. Cajamarca, “Activity and posture recognition for the elderly based on a wearable device”, tesis de maestría, Ingeniería, Pontifica Universidad Católica de Chile, Santiago de Chile, 2018.

[2] A. P. Alfaro Camacho, E. A. Castañeda Villatoro y C. R. Ibañez Nangüelú, “Chaleco Inteligente para Corrección de Postura con Retroalimentación”, Ciencia Latina, vol. 7, nº 6, 2024.

[3] Desconocido, “Upright go 2”, Upright, [En línea]. Disponible: https://store.uprightpose.com/products/upright-go2.

[4] SGS, “Wearlumb: La "camiseta inteligente" que corrige la postura”, SGS, 3 Octubre 2019. [En línea]. Disponible: https://www.sgs.com/es-es/noticias/2019/10/wearlumb.

[5] M. H. Chapeton, “Diseño e implementación de un prototipo de dispositivo biomédico no invasivo para la prevención de la postura incorrecta a nivel dorsal”, Fusagasuga, Colombia: Universidad de Cundinamarca, 2020.

[6] A. E. Araujo Julio, “Diseño de un sistema de monitoreo en tiempo real por medio de IOT para la medicion y analisis de datos sobre el nivel de riesgo ergonomico de trabajadores del sector metalmecánico”, Bogotá, Universidad Santo Tomas, 2020.

[7] Desconocido, “Corrector De Postura Recovery Medium”, Futura - ortopedicos y droguerias, [En línea]. Disponible: https://www.ortopedicosfuturo.com/corrector-de-postura-recovery-medium-17004603/p?idsku=1642&srsltid=AfmBOoqnYVdLgnrGgqOcRnhFDl_LZk7Ewivtd5h454ig667GOzHWS7xCz38.

[8] Falabella, “Corrector Postura Espalda Superior Ergonómico Flechas Negro Unisex”, Falabella, [En línea]. Disponible: https://www.falabella.com.co/falabella-co/product/125233714/Corrector-Postura-Espalda-Superior-Ergonomico-Flechas-Negro-Unisex/125233715.

[9] Y. Store, “Sicheer – Corrector de postura para mujeres y hombres, correctores de espalda, entrenador de soporte vertical de hombros para corregir la postura”, Yaxa Store, [En línea]. Disponible: https://store.yaxa.co/products/sicheer-corrector-de-postura-para-

mujeres-y-hombres-correctores-de-espalda-entrenador-de-soporte-vertical-de-hombros-para-corregir-la-postura/.

[10] M. Libre, “Chaleco Corrector De Postura Ortopédico Unisex Para Columna”, Mercado Libre, [En línea]. Disponible: https://articulo.mercadolibre.com.co/MCO-2630129094-chaleco-corrector-de-postura-ortopedico-unisex-para-columna-_JM?matt_tool=19390127&utm_source=google_shopping&utm_medium=organic.

[11] M. Libre, “Corrector Postura Inteligente Soporte Espalda Zx808”, Compraloencasa, [En línea]. Disponible: https://articulo.mercadolibre.com.co/MCO-807803141-corrector-postura-inteligente-soporte-espalda-zx808-_JM#polycard_client=recommendations_vip-pads-up&reco_backend=pads-retrieval-model-odin_marketplace&reco_model=pads_retrieval_model&reco_client=vip-pads-.

[12] N. A. Elorza, M. Bedoya Ortiz, J. E. Diaz Viloria, M. A. Gonzalez Rios, E. Rendon Martinez y M. Rodriguez Echeverri, “Resumen de Sedestación ó permanecer sentado mucho tiempo: riesgo ergonómico para los trabajadores expuestos”, Fundacion Dialnet, vol. 8, nº 1, pp. 134-147, 2017.

[13] A. Wong, T. Chan, A. Chau, H. T. Cheung, K. Kwan, A. Lam, P. Wong y D. De Carvalho, “Do different sitting postures affect spinal biomechanics of asymptomatic individuals”, Sciencedirect, vol. 67, pp. 230-235, 2019.

[14] F. Arippa, A. Nguyen, M. Pau y C. Harris-Adamson, “Postural strategies among office workers during a prolonged sitting bout”, ScienceDirect, vol. 102, pp. 103-723, 2022.

[15] S. Arce Eslava, E. Garcia Lozano, E. Parra Gonzalez y A. Cruz Libreros, “Costos por Dolor Lumbar en una EPS en Cali, Colombia”, Revista Colombiana de Salud Ocupacional, vol. 3, nº 2, pp. 22-25, 2013.

[16] C. SAS, “VIII Estudio Nacional de Dolor – Prevalencia del dolor crónico en Colombia”, ConsultorSalud SAS, 6 Noviembre 2014. [En línea]. Disponible: https://consultorsalud.com/viii-estudio-nacional-de-dolor-prevalencia-del-dolor-cronico-en-colombia/.

[17] N. A. J. Otero, “Diseño de correctores y relajadores de postura desde el diseño textil”, Universidad de los Andes, Facultad de Arquitectura, Diseño Textil, p. https://repositorio.uniandes.edu.co/bitstream/handle/1992/22063/u258834.pdf?sequence=1&isAllowed=y, 2004.

[18] Amazon, “ZZBB Dispositivo De Corrección De Postura Inteligente En Tiempo Real Científico Entrenamiento De La Postura De La Espalda Monitoreo De La Postura Corrector De Enderezadora”, Amazon, [En línea]. Disponible: https://www.amazon.es/ZZBB-Dispositivo-Inteligente-Entrenamiento-Enderezadora/dp/B08QZKCCT4?th=1.

[19] Exito, “Corrector De Postura Inteligente Vibracion Recargable Ajustable Comodo Gris”, Exito, [En línea]. Disponible: https://www.exito.com/corrector-de-postura-inteligente-vibracion-recargable-ajustable-comodo-gris-101520258-mp/p.

[20] Y. Store, “Bestand Corrector de postura inteligente y entrenador para espalda sin tirantes recordatorio de vibración”, Yaxa Store, [En línea]. Disponible: https://colombia.yaxa.co/products/bestand-corrector-de-postura-inteligente-profesional-y-entrenador-para-espalda-sin-tirantes-recordatorio-de-vibracion-uso-con-aplicacion-para-seguimiento-de-postura-y-entrenamiento-colombia/.

[21] J. Fraga Rodriguez y J. Gonzalez Diaz , “Estudio de la efectividad de un programa de estiramientos e higiene postural en la corrección de la postura en trabajadores administrativos: un proyecto de investigación”, Universidad de la Laguna, España, 2020.

[22] V. H. Frankel y M. Nordin, “Biomecanica basica del sistema musculoesqueletico 3 Ed”, Argentina, McGraw-Hill Interamericana, 2004.

[23] R. Cailliet, “Anatomía Funcional Biomecánica”, Marbán, 2004.

[24] L. L. d. Marzo, “Anatomie pour le mouvement”, Barcelona: La Liebre de Marzo, 2007.

[25] A. G. Cuenca, W. Garcia , P. Soveron, M. N. Stradella, J. M. Dos Santos y M. V. Garcia Arias, “Anatomia y Fisiologia del Cuerpo Humano”, Argentina, Cultural Librera Americana S.A.

[26] I. A. Guzman Tolosa y M. C. Trillos Chacon, “Articulacion Sacroiliaca una revision anatomofisiologica”, Revista de la Asociación Colombiana de Fisioterapia, 2004.

[27] C. Jarmey, Atlas conciso de los musculos, Barcelona, Editorial Paidotribo, 2008.

[28] J. Mathhew, “Tha basics of Aircraft axes: Pitch, Yaw and Roll”, California Aeronautical University, 3 Octubre 2023. [En línea]. Disponible: https://calaero.edu/aeronautics/aerodynamics/aircraft-axes-pitch-yaw-roll/.

[29] Desconocido, “Aerodinámica Básica”, Direccion Nacional de Seguridad Operacional, [En línea]. Disponible:

http://www.anac.gov.ar/anac/web/uploads/pers_aeron/examenes/ppa/teor-a-y-analisis-de-respuestas-ppa.pdf.

[30] "Pitch Roll and Yaw", WikiMedia Commons, 05 Diciembre 2012. [En línea]. Disponible: https://commons.wikimedia.org/wiki/File:Pitch_Roll_and_Yaw.svg.

[31] D. A. Alonso Becerra, "Biomecánica", Ingeniería Industrial, Instituto Superior Politécnico Jose A. Echeverria, Ciudad de la Habana, 2007.

[32] R. Baker, P. Coenen, E. Howie, A. William y L. Straker, "The Short Term Musculoskeletal and Cognitive Effects of Prolonged Sitting During Office Computer Work", National Library of Medicine, vol. 15, nº 8, p. 1678, 2018.

[33] Mactronica, "Sensor giroscopio acelerómetros GY-6500 MPU6500", Mactronica, [En línea]. Disponible: https://www.mactronica.com.co/sensor-giroscopio-acelerometro-gy-6500-mpu6500.

[34] Dielect, "CJMCU-103 Sensor de ángulo giro rotatorios encoder posición", [En línea]. Disponible: https://ssdielect.com/movimiento-y-velocidad-1/3861-cjmcu-103.html.

[35] D. Electronics, "MPU-6050 IMU 6 grados acelerómetro y giroscopio marca Genérica", [En línea]. Disponible: https://www.dynamoelectronics.com/tienda/mpu-6050-imu-6-grados-acelerometro-giroscopio/.

[36] Arduino, "Arduino Nano", 28 10 2024. [En línea]. Disponible: https://docs.arduino.cc/resources/datasheets/A000005-datasheet.pdf.

[37] E. S. (Shangai), "ESP32 Series", Septiembre 2024. [En línea]. Disponible: https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf.

[38] S. Electronica, "ESP-32", [En línea]. Disponible: https://www.sigmaelectronica.net/producto/esp-32/.

[39] R. Pi, "Rapberry Pi Zero 2 W", Abril 2024. [En línea]. Disponible: https://datasheets.raspberrypi.com/rpizero2/raspberry-pi-zero-2-w-product-brief.pdf.

[40] S. NV, "STM32F103x8", [En línea]. Disponible: https://www.sigmaelectronica.net/wp-content/uploads/2018/09/stm32f103c8.pdf.

[41] BricoGeek, "Multiplexor I2C TCA9548A", [En línea]. Disponible: https://tienda.bricogeek.com/circuitos-integrados/1060-multiplexor-i2c-tca9548a.html.

[42] S. Electronica, "DFR0576 - Multiplexor I2C", [En línea]. Disponible: https://www.sigmaelectronica.net/producto/dfr0576/.

[43] Mactronica, “Modulo multiplexor I2C TCA9548A”, [En línea]. Disponible: https://www.mactronica.com.co/modulo-multiplexor-i2c-tca9548a.

[44] Ferretrónica, “ Batería Recargable 18650 3.7V - 2500 mAh”, Ferretónica, [En línea]. Disponible: https://ferretronica.com/products/bateria-recargable-18650-3-7v-2600-mah?srsltid=AfmBOorAEYAu95dKTKkDs7P4SN4IsSGxBcdR6DrAAs-MS-I77IG3pKQR.

[45] M. Libre, “Pila Duracell MN1500B6 pilas AA alcalinas 1.5V 6 unidades”, Gen Salud, [En línea]. Disponible: https://www.mercadolibre.com.co/pila-duracell-mn1500b6-pilas-aa-alcalinas-15v-6-unidades/p/MCO21431772#polycard_client=search-nordic&searchVariation=MCO21431772&position=13&search_layout=stack&type=product&tracking_id=da3a5802-f378-4fb7-981c-0e5673106bb3&.

[46] M. Libre, “Adata C20, Cargador 20000mah Smart Power Bank Usb-c/a, Negro”, ADATA, [En línea]. Disponible: https://articulo.mercadolibre.com.co/MCO-2321624682-adata-c20-cargador-20000mah-smart-power-bank-usb-ca-negro-_JM#polycard_client=search-nordic&position=50&search_layout=stack&type=item&tracking_id=0fb8932c-3d07-497b-ad03-a67a32220c0d.

[47] R. Marmolejo Estrada, “Todo sobre el puerto I2C, su protocolo, tramas”, HETPRO, Guadalajara, MExico, 2017.

[48] DFRobot, “SKU:DFR0576”, [En línea]. Disponible: https://wiki.dfrobot.com/Gravity__Digital_1-to-8_I2C_Multiplexer_SKU_DFR0576.

[49] Ferretronica, “Modulo Cargador Baterías Litio TP4056 con Proteccion”, [En línea]. Disponible: https://ferretronica.com/products/modulo-cargador-baterias-litio-tp4056-con-proteccion?variant=37318758039713&currency=COP&utm_medium=product_sync&utm_source=google&utm_content=sag_organic&utm_campaign=sag_organic&utm_campaign=gs-2021-10-19&utm_source=goo.

[50] Firebase, “Crea tu app lo mejor posible con Firebase y la IA Generativa”, [En línea]. Disponible: https://firebase.google.com/?hl=es-419.

[51] Firebase, “Firebase Authentication”, [En línea]. Disponible: https://firebase.google.com/docs/auth?hl=es-419#:~:text=El%20SDK%20de%20Firebase%20Authentication%20proporciona%20m%C

3%A9todos%20a%20fin%20de,electr%C3%B3nicos%20para%20restablecer%20la%20contrase%C3%B1a.

[52] Firebase, “Cloud Firestore”, [En línea]. Disponible: https://firebase.google.com/docs/firestore?hl=es-419.

[53] AGElectronica, “MPU6050”, [En línea]. Disponible: https://agelectronica.lat/pdfs/textos/O/OKY3234.PDF.

[54] Figma, “Figma Design”, [En línea]. Disponible: https://figma.com/es-es/design/.

[55] Z. Electronica, “Cable cinta Ribbon 10 Hilos Gris X Metro”, Zamux Electronica, 2024. [En línea]. Disponible: https://www.zamux.co/cable-ribbon-10-hilos-gris-x-metro.

# ANEXOS

## ANEXO A: CÓDIGO FUENTE DE PROGRAMACIÓN APLICACION

- Código HTML para el desarrollo del frontend de la aplicación
  Ventana de Login

```
<?xml version="1.0" encoding="utf-8"?>

<LinearLayout
   xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   android:orientation="vertical"
   android:gravity="center"
   android:background="@drawable/inicio"
   tools:context=".VentanaInicial">

   <androidx.cardview.widget.CardView
       android:layout_width="350dp"
       android:layout_height="wrap_content"
       android:layout_marginTop="270dp"
       app:cardCornerRadius="30dp"
       app:cardElevation="20dp">

       <LinearLayout
           android:layout_width="match_parent"
           android:layout_height="wrap_content"
           android:layout_gravity="center_horizontal"
           android:background="@drawable/custom_edittext"
           android:orientation="vertical"
           android:padding="24dp">



           <EditText
               android:layout_width="match_parent"
               android:layout_height="50dp"
               android:id="@+id/inicio_correo"
               android:background="@drawable/custom_edittext"
               android:layout_marginTop="40dp"
               android:padding="8dp"
               android:hint="Correo"
               android:drawableLeft="@drawable/baseline_person_24"
               android:drawablePadding="8dp"
               android:textColor="@color/black"/>

           <EditText
               android:layout_width="match_parent"
               android:layout_height="50dp"
               android:id="@+id/inicio_contrasena"
               android:background="@drawable/custom_edittext"
               android:layout_marginTop="20dp"
               android:padding="8dp"
```

```
                android:hint="Contraseña"
                android:drawableLeft="@drawable/baseline_lock_24"
                android:drawablePadding="8dp"
                android:textColor="@color/black"/>

            <Button
                android:layout_width="match_parent"
                android:layout_height="60dp"
                android:text="Acceder"
                android:id="@+id/accederboton"
                android:textSize="18sp"
                android:layout_marginTop="30dp"
                android:backgroundTint="@color/green"
                app:cornerRadius = "20dp" />

            <Button
                android:id="@+id/registrarseboton"
                android:layout_width="match_parent"
                android:layout_height="60dp"
                android:layout_marginTop="2dp"
                android:textSize="18sp"
                android:backgroundTint="@color/green"
                android:text="Registrarse"
                app:cornerRadius="20dp" />


        </LinearLayout>

    </androidx.cardview.widget.CardView>
</LinearLayout>
```

Ventana de Manú

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:gravity="center"
    android:background="@drawable/fondo"
    tools:context=".MainActivity">

    <androidx.cardview.widget.CardView
        android:layout_width="357dp"
        android:layout_height="390dp"
        android:layout_marginTop="270dp"
        app:cardCornerRadius="30dp"
        app:cardElevation="20dp">


        <LinearLayout
            android:layout_width="match_parent"
```

```
            android:layout_height="wrap_content"
            android:layout_gravity="center_horizontal"
            android:orientation="vertical"
            android:padding="24dp">

            <EditText
                android:id="@+id/txt_correo"
                android:layout_width="250dp"
                android:layout_height="50dp"
                android:layout_marginTop="15dp"
                android:layout_marginStart="25dp"
                android:background="@drawable/custom_edittext"
                android:hint="Correo"
                android:padding="16dp"
                android:textColor="@color/black" />

            <Button
                android:id="@+id/infop"
                android:layout_width="150dp"
                android:layout_height="60dp"
                android:backgroundTint="@color/green"
                android:text="Información Personal"
                android:textSize="18sp"
                app:cornerRadius="20dp" />
            <Button
                android:id="@+id/informacionm"
                android:layout_width="150dp"
                android:layout_height="60dp"
                android:backgroundTint="@color/green"
                android:text="Informacion Medica"
                android:textSize="18sp"
                app:cornerRadius="20dp" />

            <Button
                android:id="@+id/monitoreo"
                android:layout_width="150dp"
                android:layout_height="60dp"
                android:backgroundTint="@color/green"
                android:text="Monitoreo"
                android:textSize="18sp"
                app:cornerRadius="20dp" />

            <Button
                android:id="@+id/cerrar_boton"
                android:layout_width="150dp"
                android:layout_height="60dp"
                android:backgroundTint="@color/green"
                android:text="Cerrar Sesión"
                android:textSize="18sp"
                app:cornerRadius="20dp" />


        </LinearLayout>

    </androidx.cardview.widget.CardView>
```

```
</LinearLayout>
```

Ventana de Manual de Usuario

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
   xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   android:orientation="vertical"
   android:gravity="center"
   android:background="@drawable/manual"
   tools:context=".manual">
   <Button
       android:id="@+id/login_regresar"
       android:layout_width="145dp"
       android:layout_height="60dp"
       android:layout_marginStart="120dp"
       android:layout_marginTop="600dp"
       android:backgroundTint="@color/green"
       android:text="Regresar"
       android:textSize="18sp"
       app:cornerRadius="20dp" />

</LinearLayout>
```

Ventana de Informacion Personal

```
<?xml version="1.0" encoding="utf-8"?>

       <LinearLayout
          xmlns:android="http://schemas.android.com/apk/res/android"
            xmlns:app="http://schemas.android.com/apk/res-auto"
            xmlns:tools="http://schemas.android.com/tools"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:orientation="vertical"
            android:gravity="center"
            android:background="@drawable/fondoinfop"
            tools:context=".Login">

   <androidx.cardview.widget.CardView
       android:layout_width="357dp"
       android:layout_height="390dp"
       android:layout_marginTop="270dp"
       app:cardCornerRadius="30dp"
       app:cardElevation="20dp">


       <LinearLayout
           android:layout_width="match_parent"
           android:layout_height="wrap_content"
```

```
        android:layout_gravity="center_horizontal"
        android:orientation="vertical"
        android:padding="24dp">

        <EditText
            android:id="@+id/txt_correo"
            android:layout_width="240dp"
            android:layout_height="50dp"
            android:layout_marginTop="0dp"
            android:layout_marginStart="30dp"
            android:background="@drawable/custom_edittext"
            android:hint="Correo"
            android:padding="16dp"
            android:textColor="@color/black" />




        <EditText
                android:id="@+id/txt_edad"
                android:layout_width="240dp"
                android:layout_height="50dp"
                android:layout_marginTop="6dp"
                android:layout_marginStart="30dp"
                android:background="@drawable/custom_edittext"
                android:hint="Edad"
                android:padding="16dp"
                android:textColor="@color/black" />

        <EditText
            android:id="@+id/txt_peso"
            android:layout_width="240dp"
            android:layout_height="50dp"
            android:layout_marginTop="6dp"
            android:layout_marginStart="30dp"
            android:background="@drawable/custom_edittext"
            android:hint="Peso"
            android:padding="16dp"
            android:textColor="@color/black" />

        <EditText
            android:id="@+id/txt_altura"
            android:layout_width="240dp"
            android:layout_height="50dp"
            android:layout_marginTop="6dp"
            android:layout_marginStart="30dp"
            android:background="@drawable/custom_edittext"
            android:hint="Altura"
            android:padding="16dp"
            android:textColor="@color/black" />


        <LinearLayout
            android:layout_width="match_parent"
```

```
                android:layout_height="wrap_content"
                android:orientation="horizontal"
                android:layout_marginTop="10dp"
                android:gravity="center">

                <Button
                    android:id="@+id/login_guardar"
                    android:layout_width="150dp"
                    android:layout_height="60dp"
                    android:backgroundTint="@color/green"
                    android:text="Guardar"
                    android:textSize="18sp"
                    app:cornerRadius="20dp" />

                <Button
                    android:id="@+id/login_recuperar"
                    android:layout_width="150dp"
                    android:layout_height="60dp"
                    android:layout_marginStart="10dp"
                    android:backgroundTint="@color/green"
                    android:text="Recuperar"
                    android:textSize="18sp"
                    app:cornerRadius="20dp" />
            </LinearLayout>

            </LinearLayout>

        <Button
            android:id="@+id/login_regresar"
            android:layout_width="150dp"
            android:layout_height="60dp"
            android:layout_marginStart="180dp"
            android:layout_marginTop="320dp"
            android:backgroundTint="@color/green"
            android:text="Regresar"
            android:textSize="18sp"
            app:cornerRadius="20dp" />

    </androidx.cardview.widget.CardView>


</LinearLayout>
```

Ventana de Información Médica

```
<?xml version="1.0" encoding="utf-8"?>

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:gravity="center"
```

```
    android:background="@drawable/fondomedico"
    tools:context=".Medico">

    <androidx.cardview.widget.CardView
        android:layout_width="360dp"
        android:layout_height="520dp"
        android:layout_marginTop="200dp"
        app:cardCornerRadius="30dp"
        app:cardElevation="20dp">

        <EditText
            android:id="@+id/txt_correo"
            android:layout_width="320dp"
            android:layout_height="50dp"
            android:layout_marginTop="15dp"
            android:layout_marginStart="25dp"
            android:background="@drawable/custom_edittext"
            android:hint="Correo"
            android:padding="16dp"
            android:textColor="@color/black" />


        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="40dp"
            android:layout_gravity="center_horizontal"
            android:orientation="vertical"
            android:padding="24dp">


            <TextView
                android:layout_width="280dp"
                android:layout_height="30dp"
                android:text="Omoplato Derecho:"
                android:layout_marginStart="0dp"
                android:textSize="20sp"
                tools:ignore="MissingConstraints" />

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:orientation="horizontal"
                android:layout_marginTop="0dp"
                android:gravity="center">

                <EditText
                    android:id="@+id/txt_odx"
                    android:layout_width="0dp"
                    android:layout_height="50dp"
                    android:layout_weight="1"
                    android:layout_marginStart="0dp"
                    android:background="@drawable/custom_edittext"
                    android:hint="Eje X"
                    android:padding="16dp"
```

```
                android:textColor="@color/black" />

            <EditText
                android:id="@+id/txt_ody"
                android:layout_width="0dp"
                android:layout_height="50dp"
                android:layout_weight="1"
                android:layout_marginStart="10dp"
                android:background="@drawable/custom_edittext"
                android:hint="Eje Y"
                android:padding="16dp"
                android:textColor="@color/black" />

            <EditText
                android:id="@+id/txt_odz"
                android:layout_width="0dp"
                android:layout_height="50dp"
                android:layout_weight="1"
                android:layout_marginStart="10dp"
                android:background="@drawable/custom_edittext"
                android:hint="Eje Z"
                android:padding="16dp"
                android:textColor="@color/black" />

        </LinearLayout>

        <TextView
            android:layout_width="280dp"
            android:layout_height="30dp"
            android:text="Omoplato Izquierdo:"
            android:layout_marginStart="0dp"
            android:textSize="20sp"
            tools:ignore="MissingConstraints" />

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:layout_marginTop="0dp"
            android:gravity="center">

            <EditText
                android:id="@+id/txt_oix"
                android:layout_width="0dp"
                android:layout_height="50dp"
                android:layout_weight="1"
                android:layout_marginStart="0dp"
                android:background="@drawable/custom_edittext"
                android:hint="Eje X"
                android:padding="16dp"
                android:textColor="@color/black" />

            <EditText
                android:id="@+id/txt_oiy"
                android:layout_width="0dp"
```

```
                android:layout_height="50dp"
                android:layout_weight="1"
                android:layout_marginStart="10dp"
                android:background="@drawable/custom_edittext"
                android:hint="Eje Y"
                android:padding="16dp"
                android:textColor="@color/black" />

            <EditText
                android:id="@+id/txt_oiz"
                android:layout_width="0dp"
                android:layout_height="50dp"
                android:layout_weight="1"
                android:layout_marginStart="10dp"
                android:background="@drawable/custom_edittext"
                android:hint="Eje Z"
                android:padding="16dp"
                android:textColor="@color/black" />

        </LinearLayout>

        <TextView
            android:layout_width="280dp"
            android:layout_height="30dp"
            android:text="Cresta Iliaca Derecha:"
            android:layout_marginStart="0dp"
            android:textSize="20sp"
            tools:ignore="MissingConstraints" />

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:layout_marginTop="0dp"
            android:gravity="center">

            <EditText
                android:id="@+id/txt_cdx"
                android:layout_width="0dp"
                android:layout_height="50dp"
                android:layout_weight="1"
                android:layout_marginStart="0dp"
                android:background="@drawable/custom_edittext"
                android:hint="Eje X"
                android:padding="16dp"
                android:textColor="@color/black" />

            <EditText
                android:id="@+id/txt_cdy"
                android:layout_width="0dp"
                android:layout_height="50dp"
                android:layout_weight="1"
                android:layout_marginStart="10dp"
                android:background="@drawable/custom_edittext"
                android:hint="Eje Y"
```

```
                android:padding="16dp"
                android:textColor="@color/black" />

            <EditText
                android:id="@+id/txt_cdz"
                android:layout_width="0dp"
                android:layout_height="50dp"
                android:layout_weight="1"
                android:layout_marginStart="10dp"
                android:background="@drawable/custom_edittext"
                android:hint="Eje Z"
                android:padding="16dp"
                android:textColor="@color/black" />

        </LinearLayout>

        <TextView
            android:layout_width="280dp"
            android:layout_height="30dp"
            android:text="Cresta Iliaca Izquierda:"
            android:layout_marginStart="0dp"
            android:textSize="20sp"
            tools:ignore="MissingConstraints" />

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:layout_marginTop="0dp"
            android:gravity="center">

            <EditText
                android:id="@+id/txt_cix"
                android:layout_width="0dp"
                android:layout_height="50dp"
                android:layout_weight="1"
                android:layout_marginStart="0dp"
                android:background="@drawable/custom_edittext"
                android:hint="Eje X"
                android:padding="16dp"
                android:textColor="@color/black" />

            <EditText
                android:id="@+id/txt_ciy"
                android:layout_width="0dp"
                android:layout_height="50dp"
                android:layout_weight="1"
                android:layout_marginStart="10dp"
                android:background="@drawable/custom_edittext"
                android:hint="Eje Y"
                android:padding="16dp"
                android:textColor="@color/black" />

            <EditText
                android:id="@+id/txt_ciz"
```

```
                    android:layout_width="0dp"
                    android:layout_height="50dp"
                    android:layout_weight="1"
                    android:layout_marginStart="10dp"
                    android:background="@drawable/custom_edittext"
                    android:hint="Eje Z"
                    android:padding="16dp"
                    android:textColor="@color/black" />

            </LinearLayout>

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:orientation="horizontal"
                android:layout_marginTop="10dp"
                android:gravity="center">

                <Button
                    android:id="@+id/login_guardar"
                    android:layout_width="150dp"
                    android:layout_height="60dp"
                    android:backgroundTint="@color/green"
                    android:text="Guardar"
                    android:textSize="18sp"
                    app:cornerRadius="20dp" />

                <Button
                    android:id="@+id/login_recuperar"
                    android:layout_width="150dp"
                    android:layout_height="60dp"
                    android:layout_marginStart="10dp"
                    android:backgroundTint="@color/green"
                    android:text="Recuperar"
                    android:textSize="18sp"
                    app:cornerRadius="20dp" />
            </LinearLayout>

        </LinearLayout>

        <Button
            android:id="@+id/login_regresar"
            android:layout_width="145dp"
            android:layout_height="60dp"
            android:layout_marginStart="190dp"
            android:layout_marginTop="455dp"
            android:backgroundTint="@color/green"
            android:text="Regresar"
            android:textSize="18sp"
            app:cornerRadius="20dp" />

    </androidx.cardview.widget.CardView>
```

```
</LinearLayout>
```

Ventana de Monitoreo

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".Monitoreo">


    <ListView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:visibility="gone"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        android:id="@+id/listDeviceBluetooth"
        />


    <LinearLayout
        android:id="@+id/viewConn"
        android:layout_width="match_parent"
        android:visibility="visible"
        android:layout_height="match_parent"
        android:orientation="vertical"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        android:background="@drawable/monitoreo"
        app:layout_constraintTop_toTopOf="parent">

        <LinearLayout
           xmlns:android="http://schemas.android.com/apk/res/android"
            android:layout_width="410dp"
            android:layout_height="80dp"
            android:layout_marginTop="200dp"
            android:orientation="horizontal">

            <TextView
                android:id="@+id/txtDisplay"
                android:layout_width="130dp"
                android:layout_height="80dp"
                android:layout_marginTop="0dp"
                android:layout_marginStart="18dp"
                android:text="OmoplatoD"
                android:layout_marginEnd="8dp" />

            <TextView
```

```
            android:id="@+id/txtDisplay2"
            android:layout_width="100dp"
            android:layout_height="80dp"
            android:layout_marginTop="0dp"
            android:layout_marginStart="160dp"
            android:text="OmoplatoI" />
    </LinearLayout>

    <LinearLayout
       xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="410dp"
        android:layout_height="80dp"
        android:layout_marginTop="10dp"
        android:orientation="horizontal">

        <TextView
            android:id="@+id/txtDisplayA"
            android:layout_width="130dp"
            android:layout_height="80dp"
            android:layout_marginTop="0dp"
            android:layout_marginStart="18dp"
            android:text="OmoplatoD"
            android:layout_marginEnd="8dp" />

        <TextView
            android:id="@+id/txtDisplayB"
            android:layout_width="100dp"
            android:layout_height="80dp"
            android:layout_marginTop="0dp"
            android:layout_marginStart="160dp"
            android:text="OmoplatoI" />
    </LinearLayout>


    <LinearLayout
       xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="410dp"
        android:layout_height="80dp"
        android:layout_marginTop="80dp"
        app:cardCornerRadius="30dp"
        android:orientation="horizontal">

        <TextView
            android:id="@+id/txtDisplay3"
            android:layout_width="130dp"
            android:layout_height="80dp"
            android:layout_marginTop="0dp"
            android:layout_marginStart="18dp"
            android:text="CrestaD"
            android:layout_marginEnd="8dp" />

        <TextView
            android:id="@+id/txtDisplay4"
            android:layout_width="130dp"
            android:layout_height="80dp"
```

```
            android:layout_marginTop="0dp"
            android:layout_marginStart="160dp"
            android:text="CrestaI" />
    </LinearLayout>

    <LinearLayout
       xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="410dp"
        android:layout_height="80dp"
        android:layout_marginTop="10dp"
        app:cardCornerRadius="30dp"
        android:orientation="horizontal">

        <TextView
            android:id="@+id/txtDisplayC"
            android:layout_width="130dp"
            android:layout_height="80dp"
            android:layout_marginTop="0dp"
            android:layout_marginStart="18dp"
            android:text="CrestaD"
            android:layout_marginEnd="8dp" />

        <TextView
            android:id="@+id/txtDisplayD"
            android:layout_width="130dp"
            android:layout_height="80dp"
            android:layout_marginTop="0dp"
            android:layout_marginStart="160dp"
            android:text="CrestaI" />
    </LinearLayout>


    <TextView
        android:id="@+id/txtComparacion"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:textSize="17sp"
        android:textStyle="bold"
        android:text="Postura"
        android:textAlignment="center"
        />

    <TextView
        android:id="@+id/txtDisplay5"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textSize="12sp"
        android:textStyle="bold"
        android:layout_marginTop="20dp"
        android:textAlignment="center"
        android:text="Simetrias"
        />

    <Button
```

```
            android:id="@+id/monitoreo_regresar"
            android:layout_width="145dp"
            android:layout_height="60dp"
            android:layout_marginStart="240dp"
            android:layout_marginTop="15dp"
            android:backgroundTint="@color/green"
            android:text="Regresar"
            android:textSize="18sp"
            app:cornerRadius="20dp" />

    </LinearLayout>




</androidx.constraintlayout.widget.ConstraintLayout>
```

- Código Kotlin para el desarrollo del backend de la aplicación
  Ventana de Login

```
package com.example.ergtech

//import android.content.ContentProvider
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.content.Intent
//import android.provider.ContactsContract.CommonDataKinds.Email
import android.widget.Button
import android.widget.EditText
import androidx.appcompat.app.AlertDialog
//import com.google.firebase.auth.auth
import com.google.firebase.auth.FirebaseAuth
//import com.google.firebase.auth.FirebaseUser
//import com.google.firebase.auth.auth
//import com.google.firebase.Firebase


class VentanaInicial : AppCompatActivity() {

    private lateinit var auth: FirebaseAuth
    private lateinit var iniciocorreo: EditText
    private lateinit var iniciocontrasena: EditText
    private lateinit var accederboton: Button
    private lateinit var registrarseboton: Button

    companion object{
        var emailPrincipal = ""
    }


    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_ventana_inicial)
```

```
        auth = FirebaseAuth.getInstance()
        iniciocorreo = findViewById(R.id.inicio_correo)
        iniciocontrasena = findViewById(R.id.inicio_contrasena)
        accederboton=findViewById(R.id.accederboton)
        registrarseboton=findViewById(R.id.registrarseboton)

        crear()
    }


    private fun crear(){

        registrarseboton.setOnClickListener {
                                if(iniciocorreo.text.   isNotEmpty()   &&
iniciocontrasena.text.isNotEmpty() ) {
                FirebaseAuth.getInstance().createUserWithEmailAndPas
sword(iniciocorreo.text.toString(),
                    iniciocontrasena.text.toString()).addOnCompleteL
istener {
                        if(it.isSuccessful){
                                                    emailPrincipal  =
it.result?.user?.email.toString()
                            irmenu(it.result?.user?.email?:"")
                        }
                        else{
                            alerta()
                        }
                }
            }
        }

        accederboton.setOnClickListener {
                                if(iniciocorreo.text.   isNotEmpty()   &&
iniciocontrasena.text.isNotEmpty() ) {
                FirebaseAuth.getInstance().signInWithEmailAndPasswor
d(iniciocorreo.text.toString(),
                    iniciocontrasena.text.toString()).addOnCompleteL
istener {
                    if(it.isSuccessful){
                   emailPrincipal = it.result?.user?.email.toString()
                        irmenu(it.result?.user?.email?:"")
                    }
                    else{
                        alerta()
                    }
                }
            }
        }

    }

    private fun alerta(){
        val builder=AlertDialog.Builder(this)
        builder.setTitle("Error")
```

```
        builder.setMessage("Se ha producido un error autenticando el
usuario")
        builder.setPositiveButton("Aceptar", null)
        val dialog: AlertDialog=builder.create()
        dialog.show()
    }

    private fun irmenu(email: String){

        val i = Intent(this, MainActivity::class.java)
        startActivity(i)
    }
}
```

Ventana de Manú

```
package com.example.ergtech

import android.content.Intent
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
//import android.provider.ContactsContract.CommonDataKinds.Email
import android.widget.Button
//import android.widget.EditText
import android.widget.TextView
import com.example.ergtech.VentanaInicial.Companion.emailPrincipal
import com.google.firebase.auth.FirebaseAuth
import com.google.firebase.firestore.FirebaseFirestore


class MainActivity : AppCompatActivity() {

    private lateinit var cerrarboton: Button
    private lateinit var monitorear: Button
    private lateinit var infp: Button
    private lateinit var infm: Button
    private lateinit var correo: TextView

    private var email = ""

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)


        cerrarboton = findViewById(R.id.cerrar_boton)
        monitorear=findViewById(R.id.monitoreo)
        infp=findViewById(R.id.infop)
        infm=findViewById(R.id.informacionm)
        correo=findViewById(R.id.txt_correo)

        email = emailPrincipal

        setup(email?:"")
```

```
        cerrar()
        infp()
        infm()
        monc()

    }

    private fun monc() {
        monitorear.setOnClickListener {
            startActivity(Intent(this,Monitoreo::class.java))
        }
    }


    private fun cerrar() {
        cerrarboton.setOnClickListener {
            val intent = Intent(this, VentanaInicial::class.java)
            startActivity(intent)
            intent.putExtra("email",email)
        }
    }

    private fun infp() {
        infp.setOnClickListener {
            val intent1 = Intent(this, Login::class.java)
            intent1.putExtra("email",email)
            startActivity(intent1)
        }


    }

    private fun infm() {
        infm.setOnClickListener {
            val intent2 = Intent(this, Medico::class.java)
            intent2.putExtra("email",email)
            startActivity(intent2)
        }
    }

    private fun setup(email: String) {
        correo.text= email
    }

    }
```

Ventana de Manual de Usuario

```
package com.example.ergtech

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
```

```
class manual : AppCompatActivity() {
   override fun onCreate(savedInstanceState: Bundle?) {
       super.onCreate(savedInstanceState)
       setContentView(R.layout.activity_manual)
   }
}
```

Ventana de Informacion Personal

```
package com.example.ergtech

import android.content.Intent
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Button
import android.widget.TextView
import com.google.firebase.auth.FirebaseAuth
import com.google.firebase.firestore.FirebaseFirestore

//import androidx.appcompat.widget.AppCompatButton

class Login : AppCompatActivity() {

   private lateinit var regresar: Button
   private lateinit var guardar: Button
   private lateinit var recuperar: Button
   private lateinit var correo: TextView
   private lateinit var edad: TextView
   private lateinit var peso: TextView
   private lateinit var altura: TextView
   private val db= FirebaseFirestore.getInstance()

   override fun onCreate(savedInstanceState: Bundle?) {
       super.onCreate(savedInstanceState)
       setContentView(R.layout.activity_login)

       regresar=findViewById(R.id.login_regresar)
       guardar=findViewById(R.id.login_guardar)
       recuperar=findViewById(R.id.login_recuperar)
       correo=findViewById(R.id.txt_correo)
       edad=findViewById(R.id.txt_edad)
       peso=findViewById(R.id.txt_peso)
       altura=findViewById(R.id.txt_altura)

       val email= VentanaInicial.emailPrincipal


       setup(email?:"")

       volver()

       guardar.setOnClickListener {
           val userEmail = email ?: return@setOnClickListener
           db.collection("usuarios").document(userEmail).set(
```

```
                hashMapOf("edad" to edad.text.toString(), "peso" to
peso.text.toString(), "altura" to altura.text.toString())
            )
        }

        recuperar.setOnClickListener {
            val userEmail = email ?: return@setOnClickListener
             db.collection("usuarios").document(userEmail).get().addO
nSuccessListener {
                edad.setText(it.get("edad")as String?)
                peso.setText(it.get("peso")as String?)
                altura.setText(it.get("altura")as String?)
            }
        }

    }

    private fun volver() {
        regresar.setOnClickListener {
            val intent = Intent(this, MainActivity::class.java)
            startActivity(intent)
        }
    }

    private fun setup(email: String) {
        correo.text= email
    }
```

Ventana de Información Médica

```
package com.example.ergtech

import android.content.Intent
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Button
import android.widget.TextView
import com.google.firebase.firestore.FirebaseFirestore

class Medico : AppCompatActivity() {

    private lateinit var regresar: Button
    private lateinit var guardar: Button
    private lateinit var recuperar: Button
    private lateinit var correo: TextView
    private lateinit var odx: TextView
    private lateinit var ody: TextView
    private lateinit var odz: TextView
    private lateinit var oix: TextView
    private lateinit var oiy: TextView
    private lateinit var oiz: TextView
    private lateinit var cdx: TextView
```

```
    private lateinit var cdy: TextView
    private lateinit var cdz: TextView
    private lateinit var cix: TextView
    private lateinit var ciy: TextView
    private lateinit var ciz: TextView
    private val db= FirebaseFirestore.getInstance()


    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_medico)

        regresar=findViewById(R.id.login_regresar)
        guardar=findViewById(R.id.login_guardar)
        recuperar=findViewById(R.id.login_recuperar)
        correo=findViewById(R.id.txt_correo)
        odx=findViewById(R.id.txt_odx)
        ody=findViewById(R.id.txt_ody)
        odz=findViewById(R.id.txt_odz)
        oix=findViewById(R.id.txt_oix)
        oiy=findViewById(R.id.txt_oiy)
        oiz=findViewById(R.id.txt_oiz)
        cdx=findViewById(R.id.txt_cdx)
        cdy=findViewById(R.id.txt_cdy)
        cdz=findViewById(R.id.txt_cdz)
        cix=findViewById(R.id.txt_cix)
        ciy=findViewById(R.id.txt_ciy)
        ciz=findViewById(R.id.txt_ciz)

        val email = VentanaInicial.emailPrincipal


        setup(email?:"")

        volver()

        guardar.setOnClickListener {
            val userEmail = email ?: return@setOnClickListener
              val odxValue = odx.text.toString().toFloatOrNull() ?:
return@setOnClickListener
              val odyValue = ody.text.toString().toFloatOrNull() ?:
return@setOnClickListener
              val odzValue = odz.text.toString().toFloatOrNull() ?:
return@setOnClickListener
              val oixValue = oix.text.toString().toFloatOrNull() ?:
return@setOnClickListener
              val oiyValue = oiy.text.toString().toFloatOrNull() ?:
return@setOnClickListener
              val oizValue = oiz.text.toString().toFloatOrNull() ?:
return@setOnClickListener
              val cdxValue = cdx.text.toString().toFloatOrNull() ?:
return@setOnClickListener
              val cdyValue = cdy.text.toString().toFloatOrNull() ?:
return@setOnClickListener
```

```
                val cdzValue = cdz.text.toString().toFloatOrNull() ?:
return@setOnClickListener
                val cixValue = cix.text.toString().toFloatOrNull() ?:
return@setOnClickListener
                val ciyValue = ciy.text.toString().toFloatOrNull() ?:
return@setOnClickListener
                val cizValue = ciz.text.toString().toFloatOrNull() ?:
return@setOnClickListener

            db.collection("informacion_medica").document(userEmail).
set(
                hashMapOf("odx" to odxValue, "ody" to odyValue,"odz"
to odzValue,"oix" to oixValue,"oiy" to oiyValue, "oiz" to oizValue,
                    "cdx" to cdxValue, "cdy" to cdyValue,"cdz" to
cdzValue,"cix" to cixValue,"ciy" to ciyValue, "ciz" to cizValue))
        }

        recuperar.setOnClickListener {
            val userEmail = email ?: return@setOnClickListener
            db.collection("informacion_medica").document(userEmail).
get().addOnSuccessListener {
                val odxValue = it.get("odx")
                val odyValue = it.get("ody")
                val odzValue = it.get("odz")
                val oixValue = it.get("oix")
                val oiyValue = it.get("oiy")
                val oizValue = it.get("oiz")
                val cdxValue = it.get("cdx")
                val cdyValue = it.get("cdy")
                val cdzValue = it.get("cdz")
                val cixValue = it.get("cix")
                val ciyValue = it.get("ciy")
                val cizValue = it.get("ciz")

                                    if (odxValue is Number) {
odx.setText(odxValue.toFloat().toString())
                } else { odx.setText("") }

                                    if (odyValue is Number) {
ody.setText(odyValue.toFloat().toString())
                } else { ody.setText("") }

                                    if (odzValue is Number) {
odz.setText(odzValue.toFloat().toString())
                } else { odz.setText("") }

                                    if (oixValue is Number) {
oix.setText(oixValue.toFloat().toString())
                } else { oix.setText("") }

                                    if (oiyValue is Number) {
oiy.setText(oiyValue.toFloat().toString())
                } else { oiy.setText("") }
```

```
                                    if (oizValue is Number) {
oiz.setText(oizValue.toFloat().toString())
                } else { oiz.setText("") }

                                    if (cdxValue is Number) {
cdx.setText(cdxValue.toFloat().toString())
                } else { cdx.setText("") }

                                    if (cdyValue is Number) {
cdy.setText(cdyValue.toFloat().toString())
                } else { cdy.setText("") }

                                    if (cdzValue is Number) {
cdz.setText(cdzValue.toFloat().toString())
                } else { cdz.setText("") }

                                    if (cixValue is Number) {
cix.setText(cixValue.toFloat().toString())
                } else { cix.setText("") }

                                    if (ciyValue is Number) {
ciy.setText(ciyValue.toFloat().toString())
                } else { ciy.setText("") }

                                    if (cizValue is Number) {
ciz.setText(cizValue.toFloat().toString())
                } else { ciz.setText("") }

            }

    }
}

    private fun volver() {
        regresar.setOnClickListener {
            val intent = Intent(this, MainActivity::class.java)
            startActivity(intent)
        }
    }

    private fun setup(email: String) {
        correo.text= email
    }
```

Ventana de Monitoreo

```
package com.example.ergtech

import android.content.Intent
import android.os.Build
import androidx.appcompat.app.AppCompatActivity
```

```
import android.os.Bundle
import android.util.Log
import android.view.View
import android.widget.ArrayAdapter
import android.widget.Toast
import androidx.core.content.ContextCompat
import com.example.ergtech.databinding.ActivityMonitoreoBinding
import com.google.firebase.firestore.FirebaseFirestore
import com.ingenieriajhr.blujhr.BluJhr
import ingenieria.jhr.bluetoothjhr.BluetoothJhr

lateinit var blue: BluJhr

var devicesBluetooth = ArrayList<String>()

private val db= FirebaseFirestore.getInstance()

//Omoplato derecho
private var omoplatoDerecho = ""
private var omoplatoIzquierdo = ""
private var crestaDerecha = ""
private var crestaIzquiera = ""


private var msgGeneral = ""

lateinit var vb:ActivityMonitoreoBinding
class Monitoreo : AppCompatActivity() {
   override fun onCreate(savedInstanceState: Bundle?) {
       super.onCreate(savedInstanceState)
       vb=ActivityMonitoreoBinding.inflate(layoutInflater)
       setContentView(vb.root)




       blue = BluJhr(this)
       blue.onBluetooth()
        vb.listDeviceBluetooth.setOnItemClickListener { adapterView,
view, i, l ->
           if (devicesBluetooth.isNotEmpty()){
               blue.connect(devicesBluetooth[i])
                blue.setDataLoadFinishedListener(object:BluJhr.Conne
ctedBluetooth{
                override fun onConnectState(state: BluJhr.Connected)
{
                        when(state){
                            BluJhr.Connected.True->{
                                 Toast.makeText(applicationContext,"T
rue",Toast.LENGTH_SHORT).show()
                                  vb.listDeviceBluetooth.visibility =
View.GONE
                                vb.viewConn.visibility = View.VISIBLE
                                rxReceived()
                            }
```

```
                            BluJhr.Connected.Pending->{
                                 Toast.makeText(applicationContext,"P
ending",Toast.LENGTH_SHORT).show()

                            }

                            BluJhr.Connected.False->{
                                 Toast.makeText(applicationContext,"F
alse",Toast.LENGTH_SHORT).show()
                            }

                            BluJhr.Connected.Disconnect->{
                                 Toast.makeText(applicationContext,"D
isconnect",Toast.LENGTH_SHORT).show()
                                   vb.listDeviceBluetooth.visibility =
View.VISIBLE
                                vb.viewConn.visibility = View.GONE
                            }

                        }
                    }
                })
            }
        }
        vb.monitoreoRegresar.setOnClickListener {
            finish()
        }
        rxDataFirestore()

    }

    private fun rxDataFirestore() {
         db.collection("informacion_medica").document(VentanaInicial.
emailPrincipal).get().addOnSuccessListener {
            val odxValue = it.get("odx")
            val odyValue = it.get("ody")
            val odzValue = it.get("odz")
            val oixValue = it.get("oix")
            val oiyValue = it.get("oiy")
            val oizValue = it.get("oiz")
            val cdxValue = it.get("cdx")
            val cdyValue = it.get("cdy")
            val cdzValue = it.get("cdz")
            val cixValue = it.get("cix")
            val ciyValue = it.get("ciy")
            val cizValue = it.get("ciz")

            vb.txtDisplayD.text = "CRESTA DM\n" +
                    "Eje X: $cdxValue\n" +
                    "Eje y: $cdyValue\n" +
                    "Eje z: $cdzValue\n"
```

```
            vb.txtDisplayC.text = "CRESTA IM\n" +
                    "Eje x: $cixValue\n" +
                    "Eje y: $ciyValue\n" +
                    "Eje z: $cizValue\n"

            vb.txtDisplayB.text = "OMOPLATO DM\n" +
                    "Eje x: $odxValue\n" +
                    "Eje y: $odyValue\n" +
                    "Eje z: $odzValue\n"

            vb.txtDisplayA.text = "OMOPLATO IM\n" +
                    "Eje x: $oixValue\n" +
                    "Eje y: $oiyValue\n" +
                    "Eje z: $oizValue\n"


            omoplatoIzquierdo = "$oixValue|$oiyValue|$oizValue"
            omoplatoDerecho = "$odxValue|$odyValue|$odxValue"
            crestaIzquiera = "$cixValue|$ciyValue|$cizValue"
            crestaDerecha = "$cdxValue|$cdyValue|$cdzValue"


        }
    }

    /**
     * Recibimos los datos desde arduino
     */
    private fun rxReceived() {
        blue.loadDateRx(object:BluJhr.ReceivedData{
            override fun rxDate(rx: String) {
               Log.d("DATOSRX",rx)
                runOnUiThread {

                    //sensor 1
                    if (rx.contains("|")){
                        msgGeneral = ""
                        //default color
                        defaultColor()
                        val dataRx = rx.split("|")
                        val sensX = dataRx[0]
                        val sensY = dataRx[1]
                        val sensZ = dataRx[2]
                        vb.txtDisplay.text = "OMOPLATO I\n" +
                                "Eje X: $sensX\n" +
                                "Eje Y: $sensY\n" +
                                "Eje Z: $sensZ"
                      comparacion(rx, omoplatoIzquierdo,"|","OMOPLATO
IZQUIERDO")
                    }
                    //sensor 2
                    if (rx.contains("+")){
                        val dataRx = rx.split("+")
                        val sensX = dataRx[0]
                        val sensY = dataRx[1]
```

```
                        val sensZ = dataRx[2]
                        vb.txtDisplay2.text = "OMOPLATO D\n" +
                                "Eje x: $sensX\n" +
                                "Eje y: $sensY\n" +
                                "Eje z: $sensZ"
                        comparacion(rx, omoplatoDerecho,"+","OMOPLATO
DERECHO")

                    }

                    if (rx.contains("!")){
                        val dataRx = rx.split("!")
                        val sensX = dataRx[0]
                        val sensY = dataRx[1]
                        val sensZ = dataRx[2]
                        vb.txtDisplay3.text = "CRESTA I\n" +
                                "Eje x: $sensX\n" +
                                "Eje y: $sensY\n" +
                                "Eje z: $sensZ"
                           comparacion(rx, crestaIzquiera,"!","CRESTA
IZQUIERDA")

                    }

                    if (rx.contains("&")){
                        val dataRx = rx.split("&")
                        val sensX = dataRx[0]
                        val sensY = dataRx[1]
                        val sensZ = dataRx[2]
                        vb.txtDisplay4.text = "CRESTA D\n" +
                                "Eje x: $sensX\n" +
                                "Eje y: $sensY\n" +
                                "Eje z: $sensZ"
                            comparacion(rx, crestaDerecha,"&","CRESTA
DERECHA")

                    }
                    //


                    if (rx.contains("%")){
                        val dataRx = rx.split("%")
                        val sensX = dataRx[0]
                        val sensY = dataRx[1]
                        vb.txtDisplay5.text = "" +
                                "S $sensX\n" +
                                "S: $sensY"
                    }

                }
            }
        })
    }
```

```
private fun defaultColor() {
    vb.txtDisplay.setTextColor(resources.getColor(R.color.green))
   vb.txtDisplay2.setTextColor(resources.getColor(R.color.green))
   vb.txtDisplay3.setTextColor(resources.getColor(R.color.green))
   vb.txtDisplay4.setTextColor(resources.getColor(R.color.green))
}

private fun comparacion(rx: String,
                        dataSever: String,
                        separator: String,
                        mensaje:String) {

    //mensaje de error


    //vb.txtComparacion
    if (rx.isEmpty()) return
    if (dataSever.isEmpty())return
    //| separa los datos del servidor
    // mala posicion en el omoplato izquierdo x

    //data from arduino
    var dataBody = rx.split("$separator")
    //data from server

    var dataServerBody = dataSever.split("|")

    if (dataBody[0].toFloat() > dataServerBody[0].toFloat() ||
        dataBody[0].toFloat() < dataServerBody[0].toFloat()*-1){
        msgGeneral += "POSTURA INCORRECTA $mensaje x\n"
        changeColorSensor(getPosChangecolor(separator))
    }
    if (dataBody[1].toFloat() > dataServerBody[1].toFloat()||
        dataBody[1].toFloat() < dataServerBody[1].toFloat()*-1){
        msgGeneral += "POSTURA INCORRECTA $mensaje y\n"
        changeColorSensor(getPosChangecolor(separator))

    }
    if (dataBody[2].toFloat() > dataServerBody[2].toFloat()||
        dataBody[2].toFloat() < dataServerBody[2].toFloat()*-1){
        msgGeneral += "POSTURA INCORRECTA $mensaje z\n"
        changeColorSensor(getPosChangecolor(separator))

    }

    if (separator=="&"){
        vb.txtComparacion.text = msgGeneral
    }
}


fun getPosChangecolor(separator: String):Int{
    return  when(separator){
        "|"->{
```

```
                0
            }
            "+"->{
                1
            }
            "!"->{
                2
            }
            "&"->{
                3
            }
            else -> {0}
        }
    }


    fun changeColorSensor(color:Int){
        when(color){
            0->{
                 vb.txtDisplay.setTextColor(resources.getColor(R.colo
r.red))
            }
            1->{
                 vb.txtDisplay2.setTextColor(resources.getColor(R.col
or.red))
            }
            2->{
                 vb.txtDisplay3.setTextColor(resources.getColor(R.col
or.red))
            }
            3->{
                 vb.txtDisplay4.setTextColor(resources.getColor(R.col
or.red))
            }
        }
    }


    override fun onActivityResult(requestCode: Int, resultCode: Int,
data: Intent?) {
        if (!blue.stateBluetoooth() && requestCode == 100){
            blue.initializeBluetooth()
        }else{
            if (requestCode == 100){
                devicesBluetooth = blue.deviceBluetooth()
                if (devicesBluetooth.isNotEmpty()){
                                                    val   adapter   =
ArrayAdapter(this,android.R.layout.simple_expandable_list_item_1,de
vicesBluetooth)
                    vb.listDeviceBluetooth.adapter = adapter
                }else{
                          Toast.makeText(this, "No tienes vinculados
dispositivos", Toast.LENGTH_SHORT).show()
                }
```

```
            }
        }
        super.onActivityResult(requestCode, resultCode, data)
    }

    /**
     * pedimos los permisos correspondientes, para android 12 hay que
pedir los siguientes BLUETOOTH_SCAN y BLUETOOTH_CONNECT
     * en android 12 o superior se requieren permisos adicionales
     */
        override  fun  onRequestPermissionsResult(requestCode:  Int,
permissions: Array<out String>, grantResults: IntArray) {
        if (blue.checkPermissions(requestCode,grantResults)){
            Toast.makeText(this, "Exit", Toast.LENGTH_SHORT).show()
            blue.initializeBluetooth()
        }else{
            if(Build.VERSION.SDK_INT < Build.VERSION_CODES.S){
                blue.initializeBluetooth()
            }else{
                                Toast.makeText(this,  "Algo  salio  mal",
Toast.LENGTH_SHORT).show()
            }
        }
          super.onRequestPermissionsResult(requestCode, permissions,
grantResults)
    }




}
```

ANEXO B: TABLAS CALIBRACIÓN DE SENSORES INERCIALES

| Sensor 1 | | | |
|---|---|---|---|
| **Angulo** | **Valor x** | **Valor Y** | **Valor en Z** |
| **180** | 0,55 | 10,8 | 60 |
| **175** | 1 | 10,2 | 25 |
| **170** | 1,9 | 9,8 | 60 |
| **165** | 2,5 | 8,5 | 30 |
| **160** | 3,3 | 8,2 | 25 |
| **155** | 4,4 | 8 | 60 |
| **150** | 5,2 | 7,7 | 15 |
| **145** | 5,4 | 7 | 10 |

| Sensor 2 | | | |
|---|---|---|---|
| **Angulo** | **Valor x** | **Valor Y** | **Valor en Z** |
| **180** | 0,9 | 9,7 | 50 |
| **175** | 1,5 | 9,4 | 45 |
| **170** | 2,1 | 9,1 | 60 |
| **165** | 2,5 | 9 | 35 |
| **160** | 3,8 | 8,7 | 25 |
| **155** | 4,4 | 8,5 | 50 |
| **150** | 5,3 | 8 | 25 |
| **145** | 6,4 | 7,9 | 15 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| **140** | 6,9 | 6 | 50 | **140** | 6,7 | 7,4 | 60 |
| **135** | 7,2 | 5,7 | 30 | **135** | 7,5 | 6,8 | 35 |
| **130** | 7,9 | 5 | 25 | **130** | 8 | 6,5 | 25 |
| **125** | 8,1 | 4,6 | 60 | **125** | 8,4 | 6,1 | 60 |
| **120** | 8,5 | 3,7 | 25 | **120** | 8,9 | 5,4 | 15 |
| **115** | 8,7 | 3,3 | 10 | **115** | 9,2 | 4,6 | 25 |
| **110** | 9,1 | 2,5 | 50 | **110** | 9,5 | 3,6 | 60 |
| **105** | 9,3 | 1,9 | 10 | **105** | 9,7 | 3,1 | 15 |
| **100** | 9,6 | 1,3 | 30 | **100** | 9,9 | 2,6 | 25 |
| **95** | 9,8 | 0,9 | 25 | **95** | 10,1 | 1,3 | 10 |
| **90** | 11,2 | -0,5 | 10 | **90** | 10,3 | 0,5 | 50 |
| **85** | 9,9 | -1,2 | 25 | **85** | 10,1 | -1,3 | 45 |
| **80** | 9,4 | -2,3 | 60 | **80** | 10 | -2,6 | 60 |
| **75** | 9,2 | -3,5 | 30 | **75** | 9,9 | -2,9 | 35 |
| **70** | 9,3 | -3,7 | 25 | **70** | 9,8 | -3,4 | 25 |
| **65** | 8,9 | -4,4 | 60 | **65** | 9,4 | -4,3 | 50 |
| **60** | 7,7 | -5 | 15 | **60** | 8,8 | -5,4 | 25 |
| **55** | 7,9 | -5,4 | 10 | **55** | 8,6 | -5,9 | 15 |
| **50** | 7,4 | -6,7 | 50 | **50** | 7,5 | -6,5 | 60 |
| **45** | 5,9 | -6,3 | 30 | **45** | 6,7 | -7,1 | 35 |
| **40** | 5,3 | -7,9 | 25 | **40** | 6 | -7,5 | 25 |
| **35** | 5 | -8,2 | 60 | **35** | 5,7 | -8,1 | 60 |
| **30** | 3,8 | -8,7 | 25 | **30** | 5,4 | -8,6 | 15 |
| **25** | 3,2 | -9,1 | 10 | **25** | 4,3 | -8,9 | 25 |
| **20** | 3,1 | -9,5 | 50 | **20** | 3,8 | -9,3 | 60 |
| **15** | 2,3 | -9,8 | 10 | **15** | 3,4 | -9,4 | 15 |
| **10** | 1,9 | -10 | 30 | **10** | 2,1 | -9,5 | 25 |
| **5** | 1,5 | -10,2 | 25 | **5** | 1,3 | -9,5 | 10 |
| **0** | 0,2 | -10,2 | 10 | **0** | 0,5 | -9,5 | 15 |

| Sensor 3 | | | | Sensor 4 | | | |
|---|---|---|---|---|---|---|---|
| **Angulo** | **Valor x** | **Valor Y** | **Valor Z** | **Angulo** | **Valor x** | **Valor Y** | **Valor Z** |
| **180** | 0,6 | 10,5 | 60 | **180** | 0,7 | 10,3 | 55 |
| **175** | 1,1 | 10,2 | 45 | **175** | 1,2 | 9,9 | 35 |
| **170** | 2,7 | 10 | 50 | **170** | 2,2 | 9,6 | 10 |
| **165** | 3 | 9,9 | 25 | **165** | 2,7 | 9,1 | 15 |
| **160** | 3,5 | 9,4 | 10 | **160** | 3,5 | 8,8 | 25 |
| **155** | 4,3 | 8,8 | 55 | **155** | 4,4 | 8,4 | 60 |
| **150** | 5,1 | 8,5 | 36 | **150** | 5,2 | 8,1 | 45 |
| **145** | 5,5 | 8 | 10 | **145** | 5,8 | 7,6 | 35 |
| **140** | 6,5 | 7,7 | 60 | **140** | 6,7 | 7,0 | 20 |
| **135** | 7,4 | 7,4 | 55 | **135** | 7,4 | 6,6 | 15 |
| **130** | 8,1 | 6,7 | 35 | **130** | 8,0 | 6,1 | 60 |
| **125** | 8,5 | 6,2 | 10 | **125** | 8,3 | 5,6 | 35 |
| **120** | 9,3 | 5,5 | 15 | **120** | 8,9 | 4,9 | 10 |
| **115** | 9,6 | 5,1 | 60 | **115** | 9,2 | 4,3 | 60 |
| **110** | 9,9 | 3,6 | 45 | **110** | 9,5 | 3,2 | 45 |
| **105** | 10,11 | 3,1 | 35 | **105** | 9,7 | 2,7 | 20 |
| **100** | 10,3 | 2,6 | 15 | **100** | 9,9 | 2,2 | 10 |
| **95** | 10,5 | 1,9 | 10 | **95** | 10,1 | 1,4 | 60 |
| **90** | 10,5 | 0,5 | 50 | **90** | 10,7 | 0,2 | 50 |
| **85** | 10,1 | -0,3 | 60 | **85** | 10,0 | -0,9 | 55 |
| **80** | 9,8 | -1,2 | 45 | **80** | 9,7 | -2,0 | 35 |
| **75** | 9,5 | -1,9 | 50 | **75** | 9,5 | -2,8 | 10 |
| **70** | 8,9 | -3,5 | 25 | **70** | 9,3 | -3,5 | 15 |
| **65** | 8,6 | -3,9 | 10 | **65** | 9,0 | -4,2 | 25 |
| **60** | 8,1 | -4,7 | 55 | **60** | 8,2 | -5,0 | 60 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| **55** | 7,8 | -5,4 | 36 | **55** | 8,1 | -5,6 | 45 |
| **50** | 7,6 | -5,9 | 10 | **50** | 7,5 | -6,4 | 35 |
| **45** | 6,9 | -6,4 | 60 | **45** | 6,5 | -6,6 | 20 |
| **40** | 6,4 | -6,7 | 55 | **40** | 5,9 | -7,4 | 15 |
| **35** | 6,1 | -7,3 | 35 | **35** | 5,6 | -7,9 | 60 |
| **30** | 5 | -8,1 | 10 | **30** | 4,7 | -8,5 | 35 |
| **25** | 4,2 | -8,4 | 15 | **25** | 3,9 | -8,8 | 10 |
| **20** | 3,7 | -9 | 60 | **20** | 3,5 | -9,3 | 60 |
| **15** | 3,5 | -9,1 | 45 | **15** | 3,1 | -9,4 | 45 |
| **10** | 2,3 | -9,1 | 35 | **10** | 2,1 | -9,5 | 20 |
| **5** | 1 | -9,2 | 15 | **5** | 1,3 | -9,6 | 10 |
| **0** | 0,2 | -9,2 | 10 | **0** | 0,3 | -9,6 | 60 |

## ANEXO C: CÓDIGO PROGRAMACIÓN HARDWARE

Librerias, creación de objetos y bluetooth

```
#include <DFRobot_I2C_Multiplexer.h>
#include <Adafruit_MPU6050.h>
#include <Adafruit_Sensor.h>
#include <Wire.h>
#include <SPI.h>
#include "BluetoothSerial.h"
#if !defined(CONFIG_BT_ENABLED) || !defined(CONFIG_BLUEDROID_ENABLED)
#error Bluetooth is not enabled! Please run make menuconfig to and enable it
#endif


BluetoothSerial SerialBT;
Adafruit_MPU6050 mpu;
DFRobot_I2C_Multiplexer multiplexer(&Wire, 0x70);
```

Inicializacion de cada uno de los sensores en cada puerto

```
void setup() {
  Serial.begin(115200);
  Wire.begin();
  multiplexer.begin();

  multiplexer.selectPort(0);
  if (!mpu.begin(0x68)) {
    SerialBT.println("Failed to find MPU6050 chip");
    while (1)
      ;
  }
  SerialBT.println("MPU6050 Found!");

  multiplexer.selectPort(1);
  if (!mpu.begin(0x68)) {
    SerialBT.println("Failed to find MPU6050 chip");
    while (1)
      ;
  }
  SerialBT.println("MPU6050 Found!");

  multiplexer.selectPort(2);
  if (!mpu.begin(0x68)) {
    SerialBT.println("Failed to find MPU6050 chip");
    while (1)
      ;
  }
  SerialBT.println("MPU6050 Found!");

  multiplexer.selectPort(3);
  if (!mpu.begin(0x68)) {
    SerialBT.println("Failed to find MPU6050 chip");
    while (1)
      ;
  }
  SerialBT.println("MPU6050 Found!");
}
```

Lectura de datos en cada canal

```
void loop() {
  sensors_event_t a, g, temp;
  multiplexer.selectPort(0);
  mpu.getEvent(&a, &g, &temp);
  SerialBT.print("Hombro Izquierdo");
  SerialBT.print("Angulo Y:");
  float gradesX1 = (((8.4919) * (a.acceleration.x)) - 5.4148);
  SerialBT.print(gradesX1);
  SerialBT.print("º, Angulo Y: ");
  float gradesY1 = ((7.8785 * (a.acceleration.z)) + 94.621);
  SerialBT.print(gradesY1);
  SerialBT.print("º, Angulo Z: ");
  float gradesZ1 = ((0.6393 * (a.acceleration.y)) + 69.959);
  SerialBT.print(gradesZ1);
  SerialBT.print(" º");
  SerialBT.print("-");
  delay(500);

  multiplexer.selectPort(1);
  mpu.getEvent(&a, &g, &temp);
  SerialBT.print("Hombro Derecho");
  SerialBT.print("Angulo Y:");
  float gradesX2 = (((8.5551) * (a.acceleration.x)) - 10.653);
  SerialBT.print(gradesX2);
  SerialBT.print("º, Angulo Y:");
  float gradesY2 = ((7.5391 * (a.acceleration.z)) + 90.306);
  SerialBT.print(gradesY2);
  SerialBT.print("º, Angulo Z: ");
  float gradesZ2 = ((0.7715 * (a.acceleration.y)) + 63.206);
  SerialBT.print(gradesZ2);
  SerialBT.print(" º");
  SerialBT.print("-");
  delay(500);
```

```
  multiplexer.selectPort(2);
  mpu.getEvent(&a, &g, &temp);
  SerialBT.print("Cadera Izquierda");
  SerialBT.print("Angulo Y:");
  float gradesX3 = (((8.8359) * (a.acceleration.x)) - 10.922);
  SerialBT.print(gradesX3);
  SerialBT.print("°, Angulo Y:");
  float gradesY3 = ((7.5762 * (a.acceleration.z)) + 86.662);
  SerialBT.print(gradesY3);
  SerialBT.print("°, Angulo Z: ");
  float gradesZ3 = ((0.4406 * (a.acceleration.y)) + 74.374);
  SerialBT.print(gradesZ3);
  SerialBT.print(" °");
  SerialBT.print("-");
  delay(500);

  multiplexer.selectPort(3);
  mpu.getEvent(&a, &g, &temp);
  SerialBT.print("Cadera derecha ");
  float gradesX4 = (((8.6845) * (a.acceleration.x)) - 9.3319);
  SerialBT.print(gradesX4);
  SerialBT.print("°, Angulo Y:");
  float gradesY4 = ((7.6704 * (a.acceleration.z)) + 90.477);
  SerialBT.print(gradesY4);
  SerialBT.print("°, Angulo Z: ");
  float gradesZ4 = ((0.0602 * (a.acceleration.y)) + 92.083);
  SerialBT.print(gradesZ4);
  SerialBT.print(" °");
  SerialBT.println("");
  delay(500);
}
```

# ANEXO D: MANUAL DE USUARIO

## MANUAL DE USUARIO ERGOTECH

1. **Descripción**
   ErgoTech es un sistema de monitoreo postural en tiempo real de la cadera y los hombros en personas que permanecen sentadas. Este sistema cuenta con un dispositivo electrónico, una aplicación móvil y una base de datos. El prototipo analiza articulaciones específicas como lo son: omoplatos y crestas iliacas.

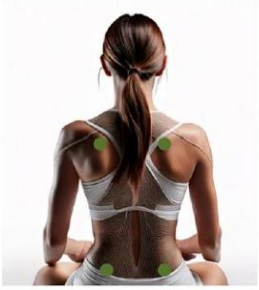
2. **Especificaciones**

   1. Si va a utilizar baterías externas asegúrese que estas sean de litio o recargables, que no superen los 3.6 V y que brinde un mínimo de 1000 mA.
   2. Las medidas recolectadas por el dispositivo corresponden a ángulos de inclinación en el eje x, y, z de las articulaciones hombros y cadera.

3. **Guía rápida de cómo utilizar las funciones principales del sistema**

   1. Póngase la camisa con el bolsillo central en la parte frontal y los carriles de los sensores en la parte posterior del torso. Como se observa en la siguiente figura:

   2. Siéntese en una silla o en algún lugar donde pueda tener la espalda completamente recta y los hombros alineados con la cadera.
   3. Ajuste las correas verticales y horizontales para que los sensores queden ubicados en los omoplatos y en las crestas iliacas, como se observa en la siguiente figura:



   4. Ingrese a la aplicación móvil Ergotech y regístrese con su correo y contraseña.

5. Seleccione en el menú interactivo la pantalla de "Información personal" e ingrese sus datos personales.
6. Seleccione en el menú interactivo la pantalla de "Información médica" e ingrese los ángulos máximos de inclinación de las articulaciones de los omoplatos y las crestas iliacas, recuerde que estos valores deben ser brindados por un fisioterapeuta especializado en ergonomía.
7. Diríjase a las herramientas de su celular y active el bluetooth.
8. Presione el botón de encendido que se encuentra en la parte delantera de la camiseta.
9. Espere 30 segundos mientras realiza la calibración el sistema.
10. Seleccione en el menú interactivo la pantalla de "Monitoreo", si realizo los pasos anteriormente mencionados se desplegara una pantalla con los dispositivos disponibles para vincularse por conexión bluetooth.

11. Seleccione la opción "ESP32test" y espere que la aplicación se vincule con el dispositivo.
12. Si la vinculación es correcta se abrirá la ventana de monitoreo.

13. Realice pequeños movimientos inclinando los hombros, subiendo y bajando la cadera, rotando los hombros y cadera, entre otros.

4. **Visualización de datos**

1. La pantalla de monitoreo le permitirá observar los ángulos de inclinación captados por el dispositivo electrónico en los ejes x, y, z de cada una de las articulaciones en color verde y los ángulos de inclinación ingresados a la aplicación en color negro, para así identificar cuál es su postura actual.

   Si se encuentra en una postura incorrecta la aplicación le notificara informando la articulación y el eje de inclinación el cual se encuentra mal posicionado, igualmente cambiara de color verde a rojo la articulación que encuentra en una mala posición.
3. Si desea puede consultar el manual de usuario de manera digital en la opción "Manual de Usuario.
4. En la ventana de "Información personal" podrá ingresar información, actualizarla y recuperarla. De modo que si requiere conocer que ingreso solo debe presionar el botón de recuperar y si desea modificarla solo debe editar los campos y presionar el botón de guardar.
5. En la ventana de "Información médica" podrá ingresar información, actualizarla y recuperarla. De modo que si requiere conocer que ingreso solo debe presionar el botón de recuperar y si desea modificarla solo debe editar los campos y presionar el botón de guardar.

5. **Mantenimientos y cuidados**

1. No alimentar la tarjeta de programación con baterías que superen los 3.6, ya que, podría causar daño en el módulo.
2. No alimentar los sensores MPU6050 con un pin diferente a los 3.3V, ya que, podría causar daños en los componentes.
2. Utilizar el dispositivo sobre ropa delgada y ajustada al cuerpo.
3. No permitir el contacto directo del dispositivo con la piel.
4. No sumergir el dispositivo en líquidos ni permitir el ingreso de este a los sensores o a la palca de programación.
5. No utilice el dispositivo en si no se encuentra sentado.
6. Apague el dispositivo cuando no se encuentre en uso y enciéndalo cuando ya se encuentre en posición de sed estación, ya que, si o enciende en otra posición puede generar una afectación en la calibración.

**5. Solución de problemas**

1. **Problema:** Error de autenticación.

   **Solución:** Valide que la contraseña y el usuario con el que se registró sean correctos, recuerde que no puede acceder sin antes haberse registrado y si ya se registró anteriormente no lo podrá hacer de nuevo.

2. **Problema:** La alimentación a través del puerto USB falla.

1. **Solución:** Seleccione como fuente de alimentación una batería externa, la cual puede ser de litio o recargable, que no superen los 3.6 V y que brinde un mínimo de 1000 mA.

3. **Problema:** No se puede vincular el dispositivo electrónico y la aplicación móvil.

   **Solución:** Cierre la aplicación, desactive el bluetooth del celular y apague el dispositivo. Espere 30 segundos, encienda nuevamente el dispositivo y realice de nuevo el proceso de conexión.

4. **Problema:** Falla en la lectura de datos, se mezcla la información de los diferentes sensores.
   **Solución:** Apague el dispositivo electrónico, espere un minuto, enciéndalo e ingrese de nuevo a la aplicación móvil.