



UNIVERSIDAD DE VALLADOLID

ESCUELA TÉCNICA SUPERIOR DE INGENIEROS DE TELECOMUNICACIÓN

TRABAJO FIN DE MÁSTER

MÁSTER UNIVERSITARIO DE INVESTIGACIÓN EN TIC
INGENIERÍA DE SISTEMAS TELEMÁTICOS

Diseño y evaluación de un sistema vestible para captura de movimientos orientado a aplicaciones de evaluación ergonómica y monitorización de terapias de rehabilitación física

Autor:

D. Javier González Alonso

Tutor:

Dr. D. Mario Martínez Zarzuela

Valladolid, septiembre de 2018

TÍTULO: **Diseño y evaluación de un sistema vestible para captura de movimientos orientado a aplicaciones de evaluación ergonómica y monitorización de terapias de rehabilitación física**

AUTOR: **D. Javier González Alonso**

TUTOR: **Dr. D. Mario Martínez Zarzuela**

DEPARTAMENTO: **Teoría de la Señal y Comunicaciones e Ingeniería Telemática**

TRIBUNAL

PRESIDENTE: **Dr. D. Fco. Javier Díaz Pernas-IT**

SECRETARIO: **Dra. D. Míriam Antón Rodríguez-IT**

VOCAL: **Dr. D. Carlos Gómez Peña-TSC**

SUPLENTE 1: **Dr. D. José Fernando Díez Higuera-IT**

SUPLENTE 2: **Dr. D. Javier Aguiar Pérez-TSC**

SUPLENTE 2: **Dra. D. María García Gadañón-TSC**

FECHA:

CALIFICACIÓN:

Deseo manifestar mi más sincero agradecimiento a todo nuestro equipo del laboratorio 12 del Grupo de Telemática e Imagen y en especial al Dr. D. Mario Martínez Zarzuela, por su interés, sus consejos y su inestimable ayuda durante el desarrollo de éste y otros proyectos.

También al resto de integrantes del Grupo de Telemática e Imagen de la Universidad de Valladolid, en especial a David, por dar esos pasos que convierten un proyecto pequeño en uno grande; a Héctor por prestarse a ayudar con las pruebas de goniometría y a Raúl por ayudarnos a avanzar todos juntos, y en general a todos aquellos que han convertido el laboratorio en mi casa durante los últimos meses.

Por último, a mi familia y a todos los que han estado apoyando desde la distancia, pero con mucha fuerza.

Per aspera ad astra

Resumen

El objetivo de este Trabajo Fin de Máster consiste en el desarrollo y posterior evaluación de varios sistemas integrados formados por dispositivos hardware vestibles y programas software a ejecutar en PC, cuya aplicación se centra en la valoración ergonómica y la rehabilitación. El trabajo constará de la realización de tres prototipos, con un mismo objetivo común: conformar la parte hardware del sistema, común a ambas aplicaciones.

Primero se estudiará la necesidad de sistemas vestibles como alternativa a los sistemas tradicionales utilizados para estas aplicaciones, comúnmente basados en análisis por imagen. Se constatará la creciente aparición de estos sistemas que buscan ofrecer soluciones más baratas y precisas a los dispositivos de captura de movimientos para aplicaciones de evaluación ergonómica y rehabilitación, tanto tradicional como mediante juegos serios.

Expondremos a continuación los requisitos de cada aplicación, el desarrollo de cada sistema electrónico y su implementación, procediendo a la descripción pormenorizada de los prototipos y ofreciendo los manuales de uso. Comprobaremos además la precisión y fiabilidad del sistema mediante una batería de pruebas tanto en entorno industrial (evaluación de puestos de trabajo) como médico (rehabilitación).

Finalmente, abordaremos las conclusiones alcanzadas tras todo el proceso y plantearemos las posibles líneas futuras a seguir.

Palabras clave

Co-diseño HW-SW, Wi-Fi, Bluetooth, Ergonomía, Rehabilitación, Cinemática.

Abstract

The objective of this Master's Thesis is the development and subsequent evaluation of various integrated systems formed by wearable hardware devices and their program interface. These devices will be used in PC applications focused on ergonomic assessment and rehabilitation. The work will consist of the production of three prototypes, with the same common objective: to conform the hardware part of the system, common to both applications.

First, we will study the needs of the emergence of these wearable systems, mainly as an alternative to the traditional systems used for these applications, commonly based on image analysis. We will confirm the increasing development of these systems as an alternative that offer cheaper and more accurate solutions to motion capture devices for ergonomic evaluation and rehabilitation applications (both traditional and serious games approach).

We will explain the requirements of each application, the development of each electronic system and its implementation, and a detailed description of the prototypes and manuals of use. We will also check the accuracy and reliability of the system through a battery of tests in both industrial (job evaluation) and medical (rehabilitation) environments.

Finally, we will address the conclusions reached after the entire process and we will outline the possible future guidelines.

Keywords

HW-SW Codesign, Wi-Fi, Bluetooth, Ergonomics, Rehabilitation, Kinematics.

ÍNDICE GENERAL

CAPÍTULO 1	INTRODUCCIÓN	10
1.1	MOTIVACIÓN	11
1.2	OBJETIVOS	12
1.2.1	Objetivo global	12
1.2.2	Objetivos parciales	13
1.3	ESTADO DE LA TÉCNICA	14
1.3.1	Estudios previos	14
1.3.2	Sector análisis médico y rehabilitación	15
1.3.3	Sector valoración ergonómica industrial	16
1.3.4	Necesidad del sistema	16
1.4	ESTRUCTURA DE LA MEMORIA	16
CAPÍTULO 2	ENTORNO MÉDICO Y REHABILITACIÓN	18
2.1	ESTADO ACTUAL DE LA REHABILITACIÓN	18
2.2	REVISIÓN SISTEMÁTICA	19
2.2.1	Necesidad	19
2.2.2	Estrategia de búsqueda de literatura	20
2.2.3	Resultados de la revisión	21
2.3	PROPUESTA DE SISTEMA DE REHABILITACIÓN	22
2.3.1	TELEKIN	23
2.3.2	Sistema integrado	23
2.3.3	Suite de rehabilitación	24
2.3.4	Estado actual	24
CAPÍTULO 3	EVALUACIÓN ERGONÓMICA	26
3.1	RIESGOS POR EL DISEÑO DEL PUESTO DE TRABAJO	26
3.1.1	Atura, Profundidad y/o alcances	26
3.1.2	Espacio de trabajo	27
3.1.3	Movimiento de materiales	28
3.1.4	Riesgo por la postura o repetitividad	29
3.2	INTEGRACIÓN DEL SISTEMA EN LA VALORACIÓN ERGONÓMICA	31
3.3	APLICACIÓN DEL SISTEMA DE VALORACIÓN ERGONÓMICA	32
CAPÍTULO 4	SISTEMA HARDWARE	34
4.1	DISPOSITIVO HARDWARE V1: VERSIÓN CABLEADA	34
4.1.1	Placa con la Unidad de Procesamiento	34

4.1.2	Sensores	36
4.1.3	Multiplexación cableada de sensores	37
4.1.4	Bluetooth	39
4.1.5	Alimentación	41
4.1.6	Encapsulado final	43
4.2	DISPOSITIVO HARDWARE V2: VERSIÓN BT AUTÓNOMA	44
4.2.1	Placa con la Unidad de Procesamiento	45
4.2.2	Encapsulado final	46
4.3	DISPOSITIVO HARDWARE DEFINITIVO (ESP32)	46
4.3.1	Placa con la Unidad de Procesamiento	47
4.3.2	Sensores	48
4.3.3	Alimentación	49
4.3.4	Pruebas protoboard	51
4.3.5	Conexiones	52
4.3.6	Encapsulado	55
CAPÍTULO 5 ESTUDIO DE LA COMUNICACIÓN		56
5.1	TRANSMISIÓN DE LOS DATOS	56
5.1.1	Conexión USB - Serie	56
5.1.2	Convertidor de niveles	57
5.1.3	Funcionamiento sobre Wi-Fi	58
5.1.4	Protocolos de comunicación internos Sensores-Microprocesador	66
5.2	INTERFAZ DEL SISTEMA	69
CAPÍTULO 6 DESARROLLO Y PLANIFICACIÓN		72
6.1	CONCEPTOS TEÓRICOS	72
6.1.1	Goniometría	72
6.1.2	Cuaterniones	78
6.1.3	Dinámica del hombro	80
6.2	ESTUDIO PRELIMINAR DE LOS REQUISITOS	81
6.2.1	Requisitos generales del proyecto	81
6.2.2	Requisitos particulares del proyecto	82
6.3	PLANIFICACIÓN DEL PROYECTO	83
6.3.1	Metodología de planificación y desarrollo de proyectos	83
6.3.2	Calendario temporal	84
6.4	ESTABLECIMIENTO DE LA BATERÍA DE PRUEBAS	85
6.4.1	Limitaciones de sistemas basados en cámaras	86
6.4.2	Pruebas evaluación ROM en tiempo real	86

6.4.3	Pruebas herramienta de análisis ergonómico (a posteriori)	87
CAPÍTULO 7	RESULTADOS	89
7.1	RESULTADOS	89
7.1.1	Resultados Dispositivo V1: versión cableada	89
7.1.2	Resultados Dispositivo Hardware Definitivo (ESP32)	90
	Limitaciones de sistemas basados en cámaras	91
	Pruebas evaluación ROM en tiempo real	99
	Pruebas herramienta de análisis ergonómico	105
7.2	DISCUSIÓN DE LOS RESULTADOS	113
7.3	PRESUPUESTO	115
7.3.1	Dispositivo Hardware V1: versión cableada	116
7.3.2	Dispositivo Hardware V2: versión BT autónoma	117
7.3.3	Dispositivo Hardware Definitivo (ESP32)	118
CAPÍTULO 8	CONCLUSIONES Y LÍNEAS FUTURAS	120
8.1	CONCLUSIONES	120
8.2	OBJETIVOS ALCANZADOS	122
8.3	LÍNEAS FUTURAS	123
REFERENCIAS		125

ÍNDICE DE FIGURAS

<i>Figura 1: Reproductor sesión con datos procesados/video grabado.</i>	25
<i>Figura 2: FT232RL FTDI USB to TTL Serial Adapter Module, modelo utilizado.....</i>	35
<i>Figura 3: Arduino Pro Mini pinout.....</i>	35
<i>Figura 4: Chip MPU6050 de Invensense (izq.) vs placa completa MPU6050.....</i>	36
<i>Figura 5: Placa TCA9548A 1-to-8 mux de Adafruit.....</i>	37
<i>Figura 6: Cable ethernet plano con corte.....</i>	38
<i>Figura 7: Conexión entre los dos MPU6050 mediante ethernet plano.....</i>	38
<i>Figura 8: Conexión del último sensor (antebrazo).</i>	39
<i>Figura 9: Módulo Bluetooth HC-06 (izq.) vs HC-05 (der.).....</i>	39
<i>Figura 10: Conexión del módulo Bluetooth HC-05 en el Arduino Uno.....</i>	41
<i>Figura 11: batería de litio utilizada con conector de 2 pines JST-PH.....</i>	42
<i>Figura 12: Adafruit Lilon/LiPoly Backpack Add-On con interruptor añadido.</i>	42
<i>Figura 13: Núcleo del Sistema (Arduino + BNO055 + HC-05 + I2C MUX + Batería)43</i>	
<i>Figura 14: Núcleo del Sistema (Arduino + BNO055 +HC-05 + I2C MUX + Batería) 43</i>	
<i>Figura 15: Introducción del núcleo del sistema en el encapsulado.....</i>	44
<i>Figura 16: Sistema completo listo para su colocación.....</i>	44
<i>Figura 17: Arduino Pro Trinket 3V.....</i>	45
<i>Figura 18: Arduino Pro Trinket 5V pinout (equivalente al del modelo de 3V).....</i>	45
<i>Figura 19: Dispositivo bluetooth autónomo, encapsulado final.....</i>	46
<i>Figura 20: Dispositivo bluetooth autónomo colocado para medidas de muñeca. ..</i>	46
<i>Figura 21: BNO055: Placa de Adafruit Abs. Orientation (izq.) frente a placa de proveedor chino CJMCU-055. Mayor tamaño en el caso de la placa de Adafruit.</i>	49
<i>Figura 22: Batería lipo utilizada en el dispositivo.....</i>	50
<i>Figura 23: Sistema de carga. Cara superior (izq.) e inferior (der.).....</i>	51
<i>Figura 24: Primera prueba en protoboard, movimiento 1.....</i>	51
<i>Figura 25: Primera prueba en protoboard, movimiento 2.....</i>	52
<i>Figura 26: Wemos ESP32 con placa de carga.</i>	52
<i>Figura 27: Inductor 4R7 de 4.7 μH (izq.) frente a su posición en la placa (der.).....</i>	53
<i>Figura 28: Conexión placa de carga a placa sensor CMJU 055 (D1, D2, GND, 3V3).....</i>	53
<i>Figura 29: Conexión del pulsador a la placa (Pines D7 y 3V3).....</i>	54

<i>Figura 30: Conexionado final de dos de los prototipos.</i>	54
<i>Figura 31: Encapsulado vista lateral izq. (izq.) vs Encapsulado vista lateral der. (der.)</i>	55
<i>Figura 32: Encapsulado final tras lijado y pintado.</i>	55
<i>Figura 33: convertidor de niveles FT232RL de FTDI</i>	57
<i>Figura 34: Pines de la placa Wemos MINI D1 ESP32</i>	59
<i>Figura 35: Imagen real de la placa Wemos MINI D1 ESP32</i>	59
<i>Figura 36: posibles arquitecturas del controlador Bluetooth y el host.</i>	64
<i>Figura 37: modelo de protocolo, dependencias de los perfiles</i>	66
<i>Figura 38: Conexionado de varios dispositivos al bus I2C.</i>	67
<i>Figura 39: Condición de inicio de I2C.</i>	67
<i>Figura 40: Comienzo de trama en I2C</i>	68
<i>Figura 41: Condición de parada de I2C</i>	68
<i>Figura 42: Esquema comunicación genérica I2C – Wemos ESP32 D1mini.</i>	68
<i>Figura 43: Interfaz TELEKIN con sensores inalámbricos.</i>	69
<i>Figura 44: Interfaz programa de captura en Unity3D</i>	70
<i>Figura 45: Interfaz de programación Python utilizando Jupyter notebooks</i>	71
<i>Figura 46: intersección de la prolongación de los ejes longitudinales del primer metacarpiano (a) y de la primera falange (b). Goniometría.</i>	72
<i>Figura 47: Posición neutra o posición cero.</i>	74
<i>Figura 48: Planos y ejes del cuerpo.</i>	75
<i>Figura 49: Flexión-extensión del hombro derecho: a) posición neutra de hombro; b) flexión de hombro y c) extensión del hombro.</i>	76
<i>Figura 50: Movimientos de abducción-aducción sobre el eje anteroposterior del hombro derecho en el plano frontal</i>	76
<i>Figura 51: Movimientos de rotación del hombro derecho sobre el eje vertical en el plano transversal</i>	77
<i>Figura 52: Movimientos del hombro en los distintos planos.</i>	80
<i>Figura 53: Diagrama de Gantt del proyecto.</i>	85
<i>Figura 54: plantilla goniométrica a utilizar en las pruebas.</i>	88
<i>Figura 55: Dispositivo cableado pruebas en línea de montaje.</i>	89
<i>Figura 56: Gráfica Flexo-extensión dispositivo cableado</i>	90
<i>Figura 57: Detección incorrecta de la mano por oclusión</i>	91

<i>Figura 58: Detección incorrecta de las piernas por oclusión.</i>	92
<i>Figura 59: Comportamiento normal detección antebrazo sin oclusión.</i>	92
<i>Figura 60: Detección incorrecta del antebrazo por auto-oclusión.</i>	93
<i>Figura 61: Detección incorrecta general por posición tumbada del usuario.</i>	93
<i>Figura 62: Detección incorrecta general por posición tumbada del usuario.</i>	94
<i>Figura 63: Detección incorrecta general por posición de espaldas a la cámara.</i>	94
<i>Figura 64: Detección incorrecta por auto-oclusión y posición de espaldas a la cámara.</i>	95
<i>Figura 65: Detección correcta de la mano posición 1 (caso oclusión cámara).</i>	96
<i>Figura 66: Detección correcta de la mano posición 2 (caso oclusión cámara).</i>	96
<i>Figura 67: Comportamiento normal detección antebrazo sin oclusión.</i>	97
<i>Figura 68: Detección correcta del antebrazo con auto-oclusión para cámara.</i>	97
<i>Figura 69: Posición tumbada del usuario. Detección correcta del sistema.</i>	98
<i>Figura 70: Detección correcta general en posición de frente a la cámara.</i>	98
<i>Figura 71: Detección correcta general en posición de espaldas a la cámara.</i>	99
<i>Figura 72: Abducción: modelo (izq.) vs valor ROM obtenido (135.4°).</i>	100
<i>Figura 73: Abducción programa Unity3D en tiempo real: 132.56° obtenidos.</i>	100
<i>Figura 74: Flexión: modelo (izq.) vs valor ROM obtenido (125.7°).</i>	101
<i>Figura 75: Flexión programa Unity3D en tiempo real: 123.92° obtenidos.</i>	101
<i>Figura 76: Extensión: modelo (izq.) vs valor ROM obtenido (24.4°).</i>	102
<i>Figura 77: Extensión programa Unity3D en tiempo real: 22.65° obtenidos.</i>	102
<i>Figura 78: Rotación externa: modelo (izq.) vs valor ROM obtenido (89.1°).</i>	103
<i>Figura 79: Rotación externa programa Unity3D en tiempo real: 88.35° obtenidos.</i>	103
<i>Figura 80: Rotación interna valor ROM obtenido (35.9°).</i>	104
<i>Figura 81: Rotación interna programa Unity3D en tiempo real: 34.53° obtenidos.</i>	104
<i>Figura 82: Abducción 0° (izq.) vs 45° (der.)</i>	106
<i>Figura 83: Abducción 90°(izq.) vs 110° (der.)</i>	106
<i>Figura 84: Flexión 0°(izq.) vs 20° (der.)</i>	107
<i>Figura 85: Flexión 45°(izq.) vs 90° (der.)</i>	107
<i>Figura 86: Flexión 110°(izq.)</i>	108
<i>Figura 87: Rotación 0° (izq.) vs rotación interna 45° (der.)</i>	108

<i>Figura 88: Rotación interna 45° (izq.) vs rotación externa 90° (der.)</i>	109
<i>Figura 89: Rotación interna 70° (no confundir el ángulo con el marcado por la mano, estamos midiendo la articulación del hombro).</i>	109
<i>Figura 90: Gráfica Abducción. Puntos señalados de izq. a der. 0°, 45° y 90°</i>	110
<i>Figura 91: Zoom Gráfica Abducción. Punto señalado del valor ya estabilizado</i>	110
<i>Figura 92: Gráfica Flexo-extensión. (Los valores iniciales “-10°” corresponden a extensión de hombro, el resto a flexión desde los “0°” en adelante)</i>	111
<i>Figura 93: Gráfica Flexo-extensión con principales valores anotados.</i>	111
<i>Figura 94: Gráfica Rotación. Puntos señalados de izq. a der. rotación interna 45°, externa 45°, externa 90° e interna 70°</i>	112
<i>Figura 95: Zoom Gráfica Rotación. Punto señalado del valor ya estabilizado</i>	112
<i>Figura 96: ejemplos tableros Kanban. Kanban vs. Scrum, de “Henrik Kniberg” et al.</i>	133
<i>Figura 97: Proceso de proyecto: Kanban vs. Scrum, de Henrik Kniberg & Mattias Skarin.</i>	135
<i>Figura 98: pequeña comparativa entre SVN y Git</i>	136
<i>Figura 99: Orientación proporcionada por una IMU, ejemplo aviación.</i>	139
<i>Figura 100: prototipo final colocado en espalda, brazo y antebrazo junto con el</i>	143
<i>Figura 101: software desarrollado en Unity3D de captura de movimientos.</i>	144
<i>Figura 102: gráficas resultantes de software de evaluación ergonómica, caso de análisis del movimiento del hombro: Abducción/Adducción, flexión/extensión y rotación interna/externa.</i>	145
<i>Figura 103: gráficas en forma de “tarta” o “donut centradas en el análisis temporal de una serie de rangos de movimiento.</i>	145
<i>Figura 104: herramienta de reproducción en vídeo simultánea. El archivo .txt se corresponde con la información de la sesión capturada.</i>	146
<i>Figura 105: Abducción 0° (izq.) vs 45° (der.)</i>	147
<i>Figura 106: Abducción 90° (izq.) vs 110° (der.)</i>	148
<i>Figura 107: Flexión 0° (izq.) vs 20° (der.)</i>	148
<i>Figura 108: Flexión 45°(izq.) vs 90° (der.)</i>	149

ÍNDICE DE TABLAS

<i>Tabla 1: Palabras clave de la búsqueda.....</i>	<i>20</i>
<i>Tabla 2: Principales comandos AT del módulo HC-05</i>	<i>40</i>
<i>Tabla 3: Consumo de potencia para distintos modos de trabajo.</i>	<i>50</i>
<i>Tabla 4: Plantilla ángulos de grabación para la valoración articular.</i>	<i>87</i>
<i>Tabla 5: Comparación cálculo ROM goniómetro digital vs Sensores</i>	<i>105</i>
<i>Tabla 6: Parte de la tabla de artículos separados por partes del cuerpo y articulación.....</i>	<i>113</i>
<i>Tabla 7: Presupuesto ingeniero.</i>	<i>115</i>
<i>Tabla 8: Presupuesto Prototipo V1.</i>	<i>116</i>
<i>Tabla 9: Presupuesto Prototipo V2.</i>	<i>117</i>
<i>Tabla 10: Presupuesto Prototipo final.</i>	<i>118</i>

Capítulo 1 Introducción

En la actualidad, un **27%** de la población sufre de dolencias óseas o musculares, incluyendo el dolor de espalda crónico lumbar, que afecta a más del **17%** de la población y el cervical, afectando a cerca del **15%**. Además, el 42% de las mujeres y el 31% de los hombres tampoco realizan actividades físicas fuera de su horario de trabajo (INE, 2016).

Solamente las enfermedades provocadas por posturas forzadas y movimientos repetitivos en el trabajo, constituyeron más del 70% de partes de enfermedad profesionales comunicados. Así pues, los trastornos musculoesqueléticos por exposición a riesgos ergonómicos en el trabajo son los problemas de salud de origen laboral más frecuentes en términos de incidencia y prevalencia. Comparando con Europa, Francia y España son los países con mayor número de enfermedades causadas por el trabajo declaradas, con un total de bajas que asciende a más del 55% en trabajos cualificados de la industria manufacturera, y operadores y montadores de instalaciones y maquinaria (CEPROSS, PANOTRATSS, 2012). Siendo la mayor causa de partes de enfermedad comunicados en España las enfermedades profesionales causadas por agentes físicos, tanto con baja (casi un 40%) como sin baja (más del 40%) según el Instituto Nacional de Seguridad, Salud y Bienestar en el Trabajo (2017).

Teniendo en cuenta esto, cabe pensar que las lesiones y dolencias relacionadas con la ergonomía de los puestos de trabajo, sobre todo en las grandes empresas, toma un papel muy relevante dentro del sistema de salud actual y se traduce en un sistema sanitario centrado en la rehabilitación de trastornos musculoesqueléticos con un alto número de pacientes y un gran desembolso económico por parte de la sanidad pública.

Una de las posibles soluciones para estos problemas sería centrar esfuerzos en la prevención de dichos trastornos y facilitar a su vez la rehabilitación en las primeras semanas de convalecencia, lo que reduce la incidencia de estas enfermedades.

En este trabajo de Fin de Máster, se propone la primera versión de un sistema capaz de incidir en ambos aspectos, primero mediante la valoración ergonómica de puestos de trabajo y después con la incorporación al sistema de rehabilitación de forma más dinámica y efectiva, todo con dispositivos vestibles encargados de sensorizar a los pacientes, permitiendo la rehabilitación temprana durante los periodos iniciales, actualmente caracterizados por largas listas de espera e inactividad de los pacientes.

Este Trabajo de Fin de Master se enmarca dentro de un proyecto coordinado que busca el desarrollo de un entorno integrado de evaluación ergonómica, así como de rehabilitación gracias a dispositivos sensores. Todo ello por medio del desarrollo de un sistema vestible que integra varios dispositivos, que serán utilizados tanto en

aplicaciones de valoración ergonómica en el trabajo de puestos de la industria manufacturera de automoción, además de en aplicaciones del sistema de rehabilitación del laboratorio del Grupo de Telemática e Imagen (TELEKIN) para suplir carencias del sistema Kinect® (Microsoft, 2013).

Este sistema estará diferenciado en dos aplicaciones, que permiten su uso en distintas plataformas y presentan distintas características:

- **Valoración ergonómica:** implementación del sistema hardware sobre la web, en un primer momento basado en Python, con un algoritmo de análisis de movimientos registrados que nos permite obtener estadísticas valiosas para la futura simulación de puestos de trabajo. Dichos datos de movimiento serán almacenados mediante servidores y pretenden constituir la base para un futuro sistema de aprendizaje automático mediante Deep Learning.
- **Sistema de rehabilitación:** implementación que continúa la línea de investigación que desarrolla el Grupo de Telemática e Imagen de la Universidad de Valladolid, para la rehabilitación de discapacidades motoras y cognitivas. Este sistema, conocido como sistema TELEKIN o *Tele-Rehabilitation System for musculoskeletal and cognitive disorders* (Caminero et al., 2017), tiene por objetivo fomentar la terapia entre los pacientes mejorando su grado de adecuación. Permitiendo, a su vez, el seguimiento del progreso de los pacientes por parte de los especialistas desde cualquier lugar. Se pretende la incorporación total del sistema Hardware desarrollado en este TFM al sistema TELEKIN, constituyendo un sustituto de la cámara Kinect® de Microsoft y proporcionando solución a sus principales debilidades.

1.1 Motivación

Durante los últimos años, se ha producido un escalado exponencial del interés en sistemas vestibles para aplicaciones médicas o deportivas, todo ello además fomentado por el creciente uso de dispositivos de esta índole, como los monitores de actividad y ritmo cardíaco vestibles como Fitbit o Garmin, los smartwatches, o diversos sistemas de control médico.

Su uso se ha extendido en diversos sectores, tanto robótica, ocio o deporte, como en medicina. Y están basados en el uso de diferentes elementos, como cámaras, sensores ópticos, acelerómetros, giroscopios, magnetómetros, GPS, etc.

En el caso que nos ocupa, se ha estudiado la realización mediciones fiables y precisas gracias a sistemas basados en IMUs (unidad de medida inercial). Para ello se ha conducido una revisión sistemática de artículos científicos, que buscaban comprobar la fiabilidad de los sistemas IMU frente a los utilizados como “golden standard” en cada campo de aplicación.

Gracias a nuestro estudio previo, llevado a cabo mediante esta revisión sistemática, se ha podido concluir la madurez de la tecnología, suficiente como para desarrollar un sistema que haga frente a las principales soluciones actuales para valoración ergonómica en tiempo real y asistencia de terapias rehabilitadoras físicas, basadas en su mayoría en dispositivos hardware de análisis de imagen (sistemas Mocap o de

captura de movimiento tanto con visual tracking como non-visual tracking: VICON, Kinect®, etc.)

El presente trabajo busca presentar una solución con características de fiabilidad suficiente como para utilizar en entornos reales de la industria automovilística para la valoración ergonómica, y una precisión tal que pueda utilizarse en entornos médicos como una herramienta válida de goniometría. Se presentarán para ello una batería de pruebas en el apartado **6. Desarrollo y batería de pruebas**, para determinar en parte el grado de adecuación del sistema para posibles futuras aplicaciones de medición clínica y ergonómica.

La solución se desarrollará en forma de tres dispositivos Hardware, implementando su comunicación y las distintas aplicaciones software que permitan su uso y evaluación. Estos dispositivos serán totalmente funcionales y pretenden ser un complemento a los sistemas médicos tradicionales, y un posible sustituto de las metodologías utilizadas actualmente para valoración ergonómica, basadas en su mayoría en análisis de imagen, permitiéndonos agregar funciones como análisis automático de movimientos y una mayor protección de los datos personales.

1.2 Objetivos

1.2.1 Objetivo global

El propósito general de este Trabajo Fin de Máster es diseñar y desarrollar un sistema formado por varios dispositivos hardware y su comunicación con un software también a desarrollar, ya sea para valoración ergonómica de puestos de trabajo, como para una suite de rehabilitación basada en el sistema TELEKIN.

Debido a la complejidad del proyecto, y de las múltiples versiones del hardware que se ha ido desarrollando, se ha optado por dividirlo en tres prototipos para una mejor observación y análisis de características, así como la mejor comprensión del trabajo realizado y la presentación de distintas opciones de implementación.

A continuación, expondremos los principales **objetivos globales** del proyecto:

- Conocer las características más relevantes de los potenciales usuarios, en este caso, tanto trabajadores de cadena de producción, como profesionales médicos y adultos a rehabilitar; para que la implementación de los dispositivos sea intuitiva, ergonómica y comprensible.
- Realizar una revisión del conocimiento sobre los ámbitos a simular por ambos sistemas, tanto el de valoración ergonómica como el de rehabilitación. Así estableceremos las características necesarias de los sensores para un desempeño lo más preciso y fiable posible.
- Estudiar sistemas previos de rehabilitación utilizados en otros proyectos anteriores, obteniendo así, los conocimientos adecuados para la elección de mecanismos de percepción a implementar.

- Determinar las funcionalidades disponibles en el hardware para conseguir cubrir las necesidades del sistema de valoración de puestos y tomar en consideración los posibles problemas a considerar en el transcurso del desarrollo. Todo ello en base a la información analizada tras la adquisición de conocimientos previos.
- Obtener un sistema cuya interacción con el usuario final sea sencilla y permita el adecuado progreso de los distintos casos de uso.
- Conseguir que la utilización de la aplicación completa HW+SW pueda estar al alcance de todo tipo de usuarios dentro del objetivo de cada sistema, ya sea por su accesibilidad, facilidad de uso e incluso respecto al ámbito económico. Además, cada sistema Hardware debe estar adaptado para cumplir con sus requerimientos en el contexto propio de uso ya sea un entorno industrial con sus consecuentes campos electromagnéticos espurios o en el entorno clínico para rehabilitación.
- Finalmente establecer conclusiones, plantear futuras líneas de mejora y propuestas de implementación de los dispositivos Hardware para otros ámbitos, incluso incluyendo futuras variaciones en el propio diseño.

1.2.2 Objetivos parciales

Revisión sistemática de sistemas vestibles para rehabilitación

En primer lugar, se realizará una revisión sistemática de artículos científicos referentes a la aplicación de sistemas vestibles basados en IMUs en el ámbito de la rehabilitación física. Esto nos permitirá tener una visión clara del estado de la técnica y poder asegurar la fiabilidad y precisión de este tipo de sistemas para aplicaciones médicas o ergonómicas.

Recolección y documentación de la aplicación real

Proceder a la delimitación del proyecto en base a los datos aportados por los distintos actores presentes en cada ámbito. Quienes, en nuestro caso, nos advirtieron de distintas necesidades del hardware antes de proceder con el desarrollo del prototipo final.

Búsqueda de dispositivos que cumplan con las funciones deseadas

En el ámbito de la electrónica y más concretamente de los dispositivos sensores, existe una gran variedad de dispositivos concretos que se pueden utilizar para llegar al mismo fin. Por ello se realizó una labor de búsqueda y concreción para determinar los elementos específicos que mejor respondían a las necesidades del proyecto.

Desarrollo de los prototipos

El tercer hito es el desarrollo de los prototipos una vez determinados los elementos a utilizar en cada aplicación concreta. Varios prototipos están basados en configuraciones Arduino comerciales, salvo el prototipo final, desarrollado buscando la mayor integración, potencial de comunicación (Bluetooth y WiFi) y potencia de cálculo, que por tanto será realizado gracias a un SoC (system on a chip) basado en procesadores Xtensa de *espressif systems*. Dichos prototipos se expondrán de forma más precisa en el apartado correspondiente.

Desarrollo de varias interfaces de usuario para la integración del proyecto

Tanto un programa Python de análisis de datos de movimiento, integrado en una solución basada en servidores de almacenamiento de dichos datos, como un programa para PC basado en Unity3D que permitirá la observación de los movimientos en tiempo real mediante un avatar 3D y la grabación de los mismos, además de distintas aplicaciones de juegos similares a los presentes en el sistema TELEKIN.

Integración final del proyecto

Finalmente se integrarán ambos proyectos y se comprobará el cumplimiento de los requisitos y la usabilidad del producto.

1.3 Estado de la técnica

Se ha llevado a cabo una revisión sistemática en el campo de los sensores vestibles para aplicaciones relacionadas con la rehabilitación: *Wearable IMUbased systems for rehabilitation a systematic review on reliability (pendiente de publicación)*. De ésta se han sacado las conclusiones pertinentes que apoyan el desarrollo acometido en este proyecto, así como se han obtenido resultados a cerca de la viabilidad de los sistemas del mismo modo que se pretenden alcanzar en el capítulo **7 Resultados**.

1.3.1 Estudios previos

En dicha revisión, se han analizado 20 artículos científicos de entre enero de 2012 y abril de 2018, que giraban alrededor de la aplicación de sensores IMU en dispositivos vestibles para su aplicación en rehabilitación (resultados extrapolables a la valoración ergonómica). Dichos estudios se han dividido en cuatro categorías según la calidad de su evaluación, obteniendo un total de 3 estudios preliminares, 13 evaluaciones con resultados cuantitativos llevadas a cabo en laboratorios, 2 estudios orientados a la aplicación de estos dispositivos fuera del laboratorio, y 5 estudios clínicos certificados.

Aun dentro de esta primera clasificación, cabe destacar la variedad de *outputs* vertidos por cada estudio, centrándose algunos en la comparación de resultados de los sistemas basados en IMUs con los considerados como “golden standard” (como

goniómetros o sistemas basados en análisis de imagen) o los que evalúan los resultados absolutos obtenidos directamente.

Por tanto, podemos considerar que existe una buena cantidad de estudios en dicho tema, y que resulta muy adecuada nuestra aproximación de desarrollar un sistema basado en sensores IMUs conectados inalámbricamente para obtener datos válidos para rehabilitación y ergonomía.

1.3.2 Sector análisis médico y rehabilitación

La inclusión de mediciones precisas de la actividad física es un paso crucial hoy en día para los programas de rehabilitación. Los métodos convencionales generalmente incorporan instrumentos de medición tradicionales que pueden ser reemplazados por tecnologías actuales y precisas, como sensores de movimiento de pequeño tamaño incorporables en vestibles.

Estos instrumentos convencionales, como el goniómetro, el ortorreceptor o el dinamómetro isocinético, son bastante costosos, difíciles de utilizar y obtener datos de ellos, y en la mayoría de los casos, son menos precisos que las necesidades de los programas de rehabilitación actuales. Algunas aproximaciones para estos programas se basaban en incorporar sistemas de captura de movimiento óptico (MAC) basados en cámara, pero no conseguían los resultados deseados, ya que necesitan de hardware costoso como estaciones o receptores y condiciones exigentes de espacio, lo que hace inviable su implementación en sistemas de aplicación real.

El seguimiento de movimiento basado en unidades de medición inerciales comerciales (IMU) se ha estudiado ampliamente en los últimos años como una tecnología rentable para aquellas aplicaciones en las que el seguimiento de movimiento basado en tecnologías ópticas no es adecuado.

Aunque hay estudios sobre este tema, no se han evaluado convenientemente los sistemas en todos ellos, sin demostrar en algunos el grado de fiabilidad de éstos para el análisis de rehabilitación. Por ello, hemos conducido una revisión sistemática sobre este ámbito, previa a la realización de este proyecto, con el objetivo de clarificar qué parámetros de rehabilitación podemos medir con estos dispositivos, cuáles son sus aplicaciones en el campo médico, y si finalmente constituyen un método válido y fiable de medición de precisión.

Revisando los diferentes enfoques de rehabilitación, relevancia clínica, métodos de recolección de datos, factibilidad y confiabilidad, obtuvimos unos resultados que mostraban un número creciente de estudios con evidencias constatables sobre la efectividad y viabilidad de la implementación de sistemas IMU. Convirtiéndolos así en el enfoque principal, desde nuestro punto de vista, para la mejora de los instrumentos de rehabilitación clínica necesarios en los tiempos modernos.

1.3.3 Sector valoración ergonómica industrial

El sistema actual de valoración ergonómica industrial, está basada en análisis por visión. Ya sea en casos en los que se utilizan sistemas Mocap o de captura de movimiento por imagen para la predicción de puestos (requiriendo grandes espacios y presentando grandes dificultades), como en el caso más común, en que se evalúan los puestos pregrabados desde distintos ángulos de cámara. Aquí, los movimientos son evaluados directamente por expertos forenses o profesionales médicos equivalentes en base a dichas grabaciones.

Aun en el caso de los sistemas Mocap por imagen, en que los datos son recogidos directamente y son fácilmente interpretables por los expertos a posteriori; no se consigue una solución a la altura de la precisión y fiabilidad esperable con la tecnología más avanzada, o en otros casos, las necesidades de espacio o la complejidad del vestible impiden su incorporación directa en cadena.

Por ello, frente a las dos soluciones más comunes presentes en el mercado, pero claramente insuficientes tecnológicamente, pretendemos proveer una posible tercera solución que con tecnología basada en IMUs pueda mejorar los métodos de evaluación actuales, recogiendo los datos automáticamente y proporcionando directamente su análisis a los especialistas médicos encargados de la evaluación.

1.3.4 Necesidad del sistema

La necesidad de la propuesta, además de por los puntos anteriores, está apoyada a su vez por las tendencias globales marcadas por el WEF (foro económico mundial). En primer lugar, las megatendencias asumidas como retos globales por la WEF, incluyen en su análisis para 2018 la necesidad de evolución en el ámbito del empleo, salud pública, habilidades y capital humano («Global megatrends assessment, 2015»). Nuestro proyecto de evaluación ergonómica de puestos de trabajo, responde a dicha necesidad, permitiendo la adaptación del puesto según capacidades y la mejoría de las condiciones de salud de los trabajadores.

Como proyección de futuro, también respondiendo a los retos definidos por la WEF, en esta ocasión para 2030 («4 mega-trends that could change the world by 2030», WEF), consideramos la propuesta de sistema de rehabilitación, prevención y entrenamiento que permitirán responder a los patrones demográficos (el ya conocido *active aging*) y las tendencias globales hacia el empoderamiento personal, con futuros sistemas de rehabilitación personalizada, al modo de los sistemas de entrenamiento personal, cada vez más presentes en nuestra vida diaria.

1.4 Estructura de la memoria

Estructuraremos la memoria en capítulos que abordarán los diferentes pasos realizados para alcanzar nuestro objetivo principal:

- En el primer capítulo hemos definido las motivaciones que impulsaron este trabajo, a la vez que hemos expuesto las metas de desarrollo planteadas.
- En el segundo capítulo estudiaremos el estado actual de los sistemas de rehabilitación, comentaremos la revisión sistemática llevada a cabo para motivar nuestra investigación, así como presentaremos la solución planteada como continuación del sistema TELEKIN controlado mediante el nuevo Hardware desarrollado.
- En el tercer capítulo abordaremos la valoración ergonómica desde distintos puntos: los distintos riesgos debidos al diseño del puesto de trabajo, la integración del sistema Hardware desarrollado en los métodos de análisis de la valoración ergonómica de puestos de trabajo, y la aplicación final de este nuevo método de análisis en las cadenas de producción de la industria automovilística.
- El cuarto capítulo abarca el estudio de los distintos requisitos y posibilidades de cada prototipo. En primer lugar, estudiaremos los componentes principales del Hardware y luego detallaremos los distintos prototipos desarrollados a lo largo del proyecto, sus características y el proceso de construcción de los mismos. y después comentaremos los requisitos de cada modelo, y las posibilidades para cumplirlos con elementos como procesadores, sensores o sistemas disponibles en el mercado.
- En el quinto capítulo, entraremos en detalle en cuanto a los requisitos y protocolos de comunicación a utilizar en los distintos prototipos, incluyendo un estudio detallado de los procesos de comunicación intra- e inter- dispositivo.
- En el sexto capítulo abordaremos el método de planificación de proyectos utilizado, los conceptos teóricos base de los mecanismos de evaluación, el proceso de evaluación pormenorizado, y los resultados obtenidos.
- En el séptimo capítulo, detallaremos los resultados de la batería de pruebas definida en el capítulo anterior, lo que pretende venir a justificar la fiabilidad y precisión del sistema. Comentaremos unas pequeñas conclusiones respecto a dichas pruebas y sus resultados. Finalmente calcularemos un presupuesto estimado del desarrollo.
- Y, por último, en el octavo capítulo, recogeremos tanto las conclusiones que hemos alcanzado tras todo el proceso, como las posibles líneas futuras derivadas del presente Trabajo Fin de Máster.
- A mayores, incluiremos una breve explicación de los principales sistemas sensores incluidos en los prototipos; y por último un manual de usuario, para la aplicación desarrollada, para ayudar en la comprensión de su funcionamiento y posibilidades.

Capítulo 2 Entorno médico y rehabilitación

Nuestro desarrollo, en el ámbito de aplicación médica y más concretamente en el entorno de la rehabilitación en España, pretende emplear las últimas tecnologías de sensorización y telemática para complementar las terapias tradicionales. Este sistema, busca permitir a los terapeutas ofrecer a los pacientes sesiones virtuales de actividades para terapias de rehabilitación y entrenamiento de habilidades físicas con su correspondiente supervisión a distancia.

Responde a la evolución natural del sistema TELEKIN (Caminero et al., 2017) desarrollado por el laboratorio del Grupo de Telemática e Imagen de la Universidad de Valladolid. En este caso, se ha sustituido la cámara Kinect® utilizada hasta la fecha en el sistema TELEKIN por unos sensores de desarrollo propio basados en el Trabajo de Fin de Grado anterior al presente Trabajo de Fin de Máster, titulado: *“Periféricos basados en Arduino para interacción con sistemas médicos de simulación y rehabilitación”*.

2.1 Estado actual de la rehabilitación

El sistema nacional de salud de España está centrado principalmente en los procesos agudos de la enfermedad. En las fases de convalecencia o rehabilitación la atención se diluye conllevando a grandes periodos de espera y saturación de clínicas. Es recomendado aumentar los recursos para el tratamiento de la fase subaguda o encontrar formas de mantener el nivel de atención ofrecido para la fase aguda sin disminuir la calidad de la atención.

El caso del Ictus, por ejemplo, es la primera causa de discapacidad en España. Ciento veinte mil personas son dadas de alta todos los años en los hospitales de agudos, y un 70 % de ellos necesita rehabilitación. Esta rehabilitación debe ser intensiva y temprana si queremos lograr buenos resultados. Más de un 50 % de esas personas no recibe tratamiento rehabilitador adecuado condenándoles de por vida a una dependencia grave o severa en el 60% de los casos. Esto da lugar a que cada año treinta mil personas sufrirán una dependencia grave a lo largo de su vida que podría haberse evitado. Todo ello conllevará consecuencias, no solo para la persona afectada, sino para el sistema de salud en general que tendrá que proporcionar recursos para ayudarle en su dependencia o en su defecto pensiones de invalidez.

La aplicación de los dispositivos desarrollados busca poder permitir al personal médico rehabilitador supervisar la correcta realización de los ejercicios por parte de los pacientes de una forma precisa y a distancia, posibilitando evitar la saturación actual del proceso de rehabilitación presente en nuestras entidades públicas y privadas. Nuestros sensores facilitan que los pacientes puedan tener un control

sobre la correcta realización de los ejercicios sin la presencia directa de los profesionales supervisores. Pueden aportar un gran apoyo en los procesos de post-rehabilitación sustituyendo la rehabilitación tradicional en casa, además de incorporarse en cualquier momento del proceso de rehabilitación para poder acortar las esperas de los pacientes que aún no han recibido ninguna sesión de rehabilitación.

2.2 Revisión sistemática

“Una revisión sistemática de la literatura es un medio para identificar, evaluar e interpretar toda la investigación disponible relevante para una pregunta de investigación particular, o área temática, o fenómeno de interés”. (Kitchenham, 2004).

Una revisión sistemática puede llevarse a cabo por varios motivos: resumir la evidencia existente sobre un tratamiento o tecnología, identificar cualquier vacío en la investigación actual para sugerir nuevas áreas para investigación, o proporcionar un marco para posicionar apropiadamente las nuevas actividades de investigación, entre otros. Además, la mayoría de las investigaciones comienzan con una revisión de literatura, sin embargo, a menos que una revisión de la literatura sea minuciosa y justa, tendrá poco valor científico. Esta es la principal razón para elegir realizar una revisión sistemática frente a las revisiones tradicionales. Aunque las revisiones sistemáticas requieren un esfuerzo considerablemente mayor que las revisiones tradicionales, poseen varias ventajas frente a éstas. Si los estudios dan resultados consistentes, las revisiones sistemáticas proporcionan evidencia de que el fenómeno es robusto y transferible. Si los estudios arrojan resultados inconsistentes, se pueden estudiar las fuentes de variación.

Para el desarrollo de esta revisión, se han seguido las pautas establecidas por Mulrow (1994) y Kitchenham (2004).

2.2.1 Necesidad

La inclusión de mediciones precisas de la actividad física es un paso crucial hoy en día para los programas de rehabilitación. Éstas generalmente se obtienen mediante los instrumentos de medición tradicionales como el goniómetro, el ortorreceptor o el dinamómetro isocinético, que son bastante costosos, difíciles de usar y no permiten la automatización del proceso de obtención de datos. Además, en la mayoría de los casos su precisión depende directamente de la habilidad del profesional médico.

El seguimiento de movimiento basado en unidades de medición inerciales comerciales (IMU) se ha estudiado ampliamente en los últimos años como una tecnología alternativa para aquellas aplicaciones en las que el seguimiento de movimiento basado en tecnologías ópticas no es adecuado. Sin embargo, no se han evaluado convenientemente y, aunque hay estudios sobre este tema, no se ha demostrado el grado de fiabilidad de estos sistemas para el análisis de rehabilitación. En el artículo desarrollado en el transcurso de este TFM: *“Wearable*

IMU-based systems for rehabilitation: a systematic review on reliability” nuestro objetivo es aclarar qué parámetros de rehabilitación podemos medir con estos dispositivos, cuáles son sus aplicaciones en el campo médico, y si finalmente constituyen un método válido y confiable de medida de precisión.

Para ello se revisaron los diferentes enfoques de rehabilitación, relevancia clínica, métodos de recolección de datos, factibilidad y confiabilidad.

2.2.2 Estrategia de búsqueda de literatura

Primero definimos las palabras clave que conformarán nuestra búsqueda en las distintas fuentes. Ésta selección se puede observar en la tabla a continuación:

Table 1 Literature search strategy	
Rehabilitation measures	“rehabilitation” OR “postural sway” OR “injury prevention” OR “therapeutic exercises” OR “fitness” AND
Body measurement	“kinematics” OR “range of movement” OR “body tracking” AND
Wearable devices	“wearable” OR “body area networks” OR “smart textiles” OR “mobile” AND
IMUs	“inertial sensor” OR “gyroscope” OR “accelerometer” OR “magnetometer” AND
*Type of study	“clinical study” OR “feasibility study”
(*) In the ACM database, the search was specified with the inclusion of other keywords to reduce the amount of primary results.	

Tabla 1: Palabras clave de la búsqueda.

Sobre la elección de dichos términos, las “medidas de rehabilitación” es una parte nuclear de nuestra investigación, y los sensores deben proporcionar suficiente información precisa sobre la “cinemática” humana para este fin. La sección de “medición corporal” limita los trabajos que incluyen algunos análisis cinemáticos o de movimiento relacionados con las articulaciones corporales válidas para nuestras medidas de rehabilitación. “Dispositivos portátiles” y la sección de “IMUs”, se centran en evitar dispositivos no portátiles y documentos seleccionados sobre otras tecnologías. Como estos sistemas deben proporcionar una cinemática 3D confiable y un rango de alta precisión, vamos a excluir la tecnología no IMU y algunos estudios que solo tienen en cuenta uno de los parámetros para medir como la aceleración o el giroscopio exclusivamente. La última sección, “tipo de estudio”, fue una precisión tomada en caso de que la cantidad de trabajos resultante fuera demasiado alta.

Con la selección de palabras clave ya definida, deberemos proceder con la búsqueda de artículos científicos. Para ello se ha introducido la cadena de búsqueda, en diversas bases de datos, y se han ido eliminando los artículos que no encajen con

nuestro criterio de inclusión/exclusión (definido con anterioridad) hasta llegar a los 22 artículos seleccionados para el análisis final. Podemos ver de forma más detallada en el siguiente gráfico el proceso detallado de búsqueda:

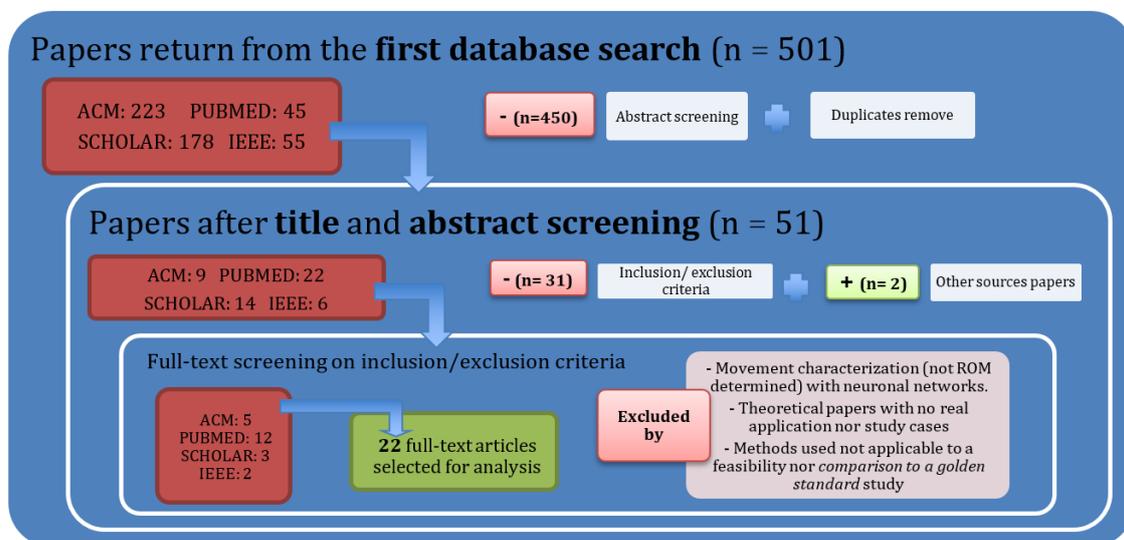


Gráfico 1: Proceso de búsqueda.

2.2.3 Resultados de la revisión

Aunque comentaremos detalladamente los resultados que más nos afectan cuando comparemos los resultados obtenidos por nuestra batería de pruebas con nuestros dispositivos propios, con los resultados de estos artículos en el apartado **7.2 Discusión de los resultados**, comentaremos brevemente los distintos aspectos analizados y las conclusiones alcanzadas.

En primer lugar, nos centramos en el lugar de colocación de los dispositivos, o lo que es lo mismo, la articulación a medir por el sistema. Primero dividimos el estudio por partes del cuerpo: *tronco*, *extremidad superior*, *extremidad inferior*, para después centrarnos en cada articulación en concreto, analizando el número de artículos centrados en cada una y la necesidad de medir una articulación sobre otra.

La extremidad superior y más concretamente el análisis del hombro alcanzaba el mayor número de artículos, lo que corroboraba nuestra idea inicial de centrar el análisis en esta zona debido al alto número de lesiones registradas en esa zona frente a las otras. También le seguía muy de cerca el análisis del tronco, más concretamente la zona lumbar y de cadera, conocida también por la gran cantidad de lesiones producidas en dichas articulaciones.

Nuestro siguiente análisis buscó agrupar distintos patrones de análisis en los artículos basándonos en los tipos de medidas que se recogían en cada uno. Creímos conveniente dividirlos en varios grupos:

- *localización espacial 3D*: de elementos como un hueso en específico o articulación,
- *identificación de patrones*: normalmente ligado al análisis con técnicas de Machine Learning,

- análisis “*in vivo*”: cuántos de los análisis proporcionaban información en tiempo real,
- *repetición motora*: cuando se analizaba la capacidad de realizar un movimiento repetitivo correctamente,
- *análisis del movimiento*: de movimientos complejos con más de una articulación con algún algoritmo del rendimiento del paciente,
- *ROM*: cuando se mida el rango de cada movimiento en grados,
- *Otros*: referido a aquellos que no se incluyeran en las categorías anteriores.

Además, los estudios incluidos también tenían diferentes objetivos como es determinar la fiabilidad de la obtención de datos precisos con la tecnología IMU en comparación con otras tecnologías clínicas; analizar la capacidad de estos sensores para discriminar los movimientos del paciente según diferentes patrones; evaluar el progreso de alguna enfermedad; proporcionar varios sistemas de ayudas válidas para terapeutas, etc.

También centramos el análisis en la evaluación de la viabilidad y limitaciones de cada sistema. Primero dividiendo los artículos por tipo de estudio según la fiabilidad y el valor del mismo: se incluían tanto *estudios preliminares*, con reducido número de pacientes o condiciones limitantes; *evaluaciones cuantitativas en laboratorio*, sin grupo de control y por tanto con menor validez estadística; *estudios fuera del laboratorio*, similar al anterior pero sin limitaciones por las condiciones de laboratorio; y *ensayos clínicos*, el tipo de estudio más fiable y preciso que cuenta con varios grupos de participantes incluyendo un grupo de control, y que desarrolla las pruebas siguiendo un estricto protocolo.

A continuación, se analizó la calidad del método de establecimiento de la fiabilidad del sistema en cada artículo, también aquí intentando englobarlos en varias categorías. Éstas incluían la *identificación correcta de movimientos*, que podía realizarse con la simple observación o mediante vídeos, en los artículos centrados en el análisis de movimientos; la comparación con *servo-motores* o *brazos robot*, lo que permitía comparar con precisión las medidas introducidas y obtenidas; la validación de los experimentos por parte de un *profesional médico o fisioterapeuta*; la obtención de *gráficas estándar* ampliamente aceptadas con valores analizables; o la comparación con cualquier instrumento considerado como “*golden standard*” como los anteriormente comentados: sistemas Mocap, goniómetros, etc.

Finalmente, como conclusión de esta revisión sistemática, podemos decir que los resultados mostraron un número creciente de estudios que demuestran evidencias sobre la efectividad y la viabilidad de la implementación de sistemas IMU para medidas y evaluación del movimiento corporal, convirtiéndolos en un gran enfoque para la mejora de los instrumentos de rehabilitación clínica necesarios en la actualidad.

2.3 Propuesta de sistema de Rehabilitación

El sistema en que se basa este Trabajo de Fin de Máster, es un proyecto ambicioso que integra diversos desarrollos (como el sistema TELEKIN, paso previo al sistema

actual), buscando transformar el paradigma actual del ámbito de la rehabilitación física en España.

Para ello se han incorporado dispositivos sensores de desarrollo propio primero a TELEKIN y posteriormente a un sistema propio que pretende adherir distintas posibilidades y necesidades de las aplicaciones de telerehabilitación y juegos serios para rehabilitación disponibles actualmente.

2.3.1 TELEKIN

Continuando la línea de investigación que desarrolla el Grupo de Telemática e Imagen de la Universidad de Valladolid, para la rehabilitación de discapacidades motoras y cognitivas, se ha decidido evolucionar el sistema TELEKIN o Tele-Rehabilitation System for musculoskeletal and cognitive disorders (Caminero et al., 2017).

Este sistema se compone de módulos diferenciados, para poder integrar nuevas partes de modo sencillo sin afectar al resto del trabajo. Presenta tanto un *Módulo Web* encargado de gestionar las sesiones y el acceso a los resultados de los pacientes, como un *Módulo de Juego* que contiene las propias actividades de rehabilitación.

Aquí es donde se han integrado las modificaciones realizadas, desarrollando un script de comunicación con las actividades de la plataforma, que permitieron además añadir funcionalidades y juegos extra que el dispositivo hardware nos permitió incorporar.

Se ha incorporado la compatibilidad con nuestros dispositivos sensores para sustituir a la cámara Kinect® de Microsoft. Permitiendo por tanto su uso con los mismos módulos de Juego ya desarrollados para la cámara, y favoreciendo la utilización de las nuevas características de un Hardware más preciso como es el desarrollado para este proyecto.

Además, se ha incorporado algún módulo de juego nuevo más orientado a la gamificación de ejercicios de rehabilitación como los de la escuela de espalda. Para ello se recogerán los datos de los sensores en tiempo real y la aplicación en Unity3D los recibirá para controlar el juego seleccionado con el que pretendemos ejercitar la espalda a la vez que guardamos la información relevante como rango articular para que la gestione nuestro Módulo Web (Through the Interface, 2017), encargado de gestionar las sesiones y el acceso a los resultados de los pacientes. El módulo incorporado no es objeto de este TFM, pero sí se ha utilizado para comprobar el funcionamiento de los sensores.

2.3.2 Sistema integrado

Se ha planteado, como parte de otro proyecto futuro, aunque no se ha completado su desarrollo, la integración en un sistema independiente de Unity3D y de TELEKIN para la utilización de los sensores en juegos serios fuera de dichos entornos. Esta migración ha sido fomentada debido a las grandes dificultades que presenta el porteo del sistema TELEKIN a dispositivos móviles, uno de los principales targets

objetivos del sistema, para facilitar el uso de la rehabilitación a distancia sin supervisión en vivo.

El desarrollo pretende utilizar el entorno de desarrollo Flutter SDK, para combinar la potencia del desarrollo nativo en dispositivos móviles con la versatilidad de las aplicaciones web o del lenguaje de programación Dart de Google. Dicho desarrollo se finalizará en un futuro proyecto del Grupo de Telemática e Imagen.

2.3.3 Suite de rehabilitación

Finalmente se plantea desarrollar en otro proyecto una especie de “suite” virtual que oferte las opciones del sistema TELEKIN como juegos serios y almacenamiento de sesiones en la nube, pero sin contar con las limitaciones del desarrollo original ni del hardware Kinect®.

El sistema final contará con múltiples módulos de juego configurables para cualquier articulación a rehabilitar, además de un sistema fiable de ayudas auditivas para la correcta ejecución de los ejercicios. En futuras implementaciones se buscará incluir otras ayudas sensoriales como con la inclusión de dispositivos hápticos o vibradores en los sensores.

La interfaz será sencilla y clara para el usuario final (paciente) y guiará mediante sencillas instrucciones a la autocolocación de los sensores y la realización de los ejercicios, además de contar con un sistema de calibrado automático para asegurar la fiabilidad de los datos recibidos. Para el profesional rehabilitador, el módulo web se encargará de presentar los informes, tablas o gráficas de resultados de cada paciente, siendo éstos personales e intransferibles entre clínicas o médicos y respetando la privacidad y protección de datos de cada usuario.

Finalmente, el sistema presentará un sistema de almacenamiento en la nube que aportará interoperabilidad a los distintos niveles de usuario, posibilitando el acceso al sistema desde cualquier tipo de dispositivo, tanto móvil, Tablet, PC, Mac, etc., en cualquier momento o lugar simplemente con acceso a internet.

2.3.4 Estado actual

Para este Trabajo de Fin de Máster se ha conseguido implementar nuestro sistema vestible de sensores en la plataforma TELEKIN además de desarrollar un programa en Unity3D capaz de monitorizar en tiempo real la actividad del usuario y mostrar el rango articular instantáneo por pantalla. Este programa también permite la grabación de los datos capturados en forma de datos en crudo de los sensores para ser posteriormente analizados por nuestra herramienta de valoración ergonómica; o los datos modificados por el programa en Unity3D, que nos permiten, mediante un reproductor (también desarrollado en Unity3D) reproducir las sesiones completas por medio de la lectura de archivos de texto “.txt” de poco peso, pudiendo comparar al mismo tiempo con videos grabados.



Figura 1: Reproductor sesión con datos procesados/video grabado.

Capítulo 3 Evaluación Ergonómica

La ergonomía representa la aplicación científica concerniente al diseño de instalaciones, herramientas y métodos de trabajo para el correcto desempeño del ser humano. El principal parámetro que analiza esta rama es el sistema hombre-objeto-entorno. La evaluación ergonómica busca poder diseñar el entorno y los objetos que el ser humano emplea.

Se realiza comúnmente para conocer los parámetros los cuales afectan al trabajador en su área de trabajo, los conocidos como factores de riesgo. Estos podrían provocar alguna enfermedad, como es el caso de las posturas inadecuadas que se realizan cuando el puesto de trabajo no está diseñado correctamente. Por tanto, se busca minimizar al máximo posible enfermedades como problemas cardiovasculares o trastornos musculo-esqueléticos, causados entre otros por malas posturas.

En este apartado se describirán algunos métodos utilizados en esta evaluación, incluyendo parámetros de postura, repetición, duración, carga estática, aplicación de la fuerza, uso de las herramientas y el medio ambiente. En este caso verificando las posturas estresantes, esfuerzos excesivos, sostenidos, vibración, estrés por contacto, etc.

3.1 Riesgos por el diseño del puesto de trabajo

Las siguientes directrices están guiadas de acorde con las guías redactadas por el Instituto Nacional de Seguridad e Higiene en el Trabajo del Ministerio de Trabajo y Asuntos Sociales (International Labour Office & International Ergonomics Association, 2000; INSHT, 2003; INSHT, 2004; *INSHT, "Factores de riesgo del trabajo repetitivo"*) Según éstas, para evaluar correctamente los riesgos remitidos por el diseño de los puestos de trabajo, se deberá comprobar si se dan alguna de las siguientes situaciones, enmarcadas dentro de las situaciones de riesgo no tolerables, siendo mayor el riesgo cuanto mayor sea el número de éstas.

3.1.1 Altura, Profundidad y/o alcances

Como hemos comentado, la inclusión de alguna de las siguientes características, harán al puesto de trabajo más propenso a permitir situaciones de riesgo para los trabajadores.

En primer lugar, cuando la altura de trabajo no se adapte al tipo de tarea y a las dimensiones de cada trabajador. En concreto, se presentarán alguna de las siguientes situaciones estando el trabajador de pie:

- En las tareas de precisión, la altura de trabajo no estará 5-10 cm por encima de la altura de los codos del trabajador.

- En las tareas ligeras, la altura de trabajo no estará 10-15 cm por debajo de la altura de los codos del trabajador.
- En las tareas pesadas, la altura de trabajo no estará 15-30 cm por debajo de la altura de codos del trabajador.

En el caso de que el trabajador se encuentre sentado, la altura de la superficie de trabajo no estará aproximadamente al nivel de los codos.

La zona de trabajo estará alejada del trabajador en alguna de las siguientes situaciones: los elementos de uso muy frecuente están a más de 25 cm del borde de la mesa de trabajo, los de uso medio a más de 50 cm del borde y los de uso ocasional a más de 70 cm del borde.

Se dan alcances por encima del nivel del hombro (brazos elevados y sin apoyo de manera frecuente o prolongada), alcances laterales o por detrás del cuerpo y hay apoyo de los antebrazos en bordes no redondeados o cantos agudos de mesas u otras superficies de trabajo.

Nuestro sistema

Podemos evitar muchos de los problemas de ergonomía debidos a la altura, profundidad y alcances ya que con el estudio del rango articular y movimientos (ROM, estudio cinemático, etc.) observaremos las consecuencias de un mal diseño de puestos, no permitiendo la sucesión de las características a evitar comentadas en este apartado.

Por ejemplo, nuestro sistema proporcionará los ángulos de los movimientos del trabajador, y una altura de trabajo inadecuada se observará mediante unos ángulos de movimiento muy elevados en la flexo-extensión y abducción del hombro, para alcanzar el objeto situado a una altura inadecuada según la altura del trabajador elegido para dicho puesto. Además, podremos detectarlo a simple vista si observamos la representación visual de los movimientos por medio del sistema de visionado en 3D con avatar que incluye nuestra implementación.

3.1.2 Espacio de trabajo

En este punto se trata de definir el correcto espacio de trabajo, tanto como por los elementos utilizados por los trabajadores como mesas o sillas como por elementos auxiliares de trabajo.

Se considera de riesgo cuando no hay espacio suficiente encima de la superficie de trabajo (mesa, banco de trabajo, etc) para distribuir adecuadamente los diferentes elementos u objetos utilizados, cuando no hay espacio suficiente para mover cómodamente las piernas (por ejemplo, por debajo de la mesa o banco de trabajo) o el cuerpo y cuando la superficie libre en el entorno del puesto de trabajo es menor de 2 m².

Así pues, se entiende que los trabajadores que de forma habitual trabajan de pie deben disponer de banquetas o sillas (por ejemplo, sillas de tipo semi-sentado) para sentarse ocasionalmente y que en trabajos que requieren desplazamientos o ejercer fuerzas el trabajador no estará sentado. Ha de prestarse atención a los casos en que la silla de trabajo no sea adecuada; por ejemplo, cuando los pies cuelguen del asiento sin poderse apoyar en el suelo, o el respaldo no permita un apoyo adecuado del tronco, y nunca trabajando de pie sobre superficies inestables o irregulares.

En este caso, nuestro sistema se propondrá cuando las limitaciones por espacio de trabajo ya hayan sido planteadas y corregidas por el personal cualificado en tareas de prevención ya que obedecen a planteamientos de planificación espacial y no son posibles de evitar. Nuestro sistema se limitará a encontrar la consecuencia de una mala planificación, en forma de movimientos más limitados o extraños, que se vean forzados por una inadecuada planificación del espacio de trabajo.

3.1.3 Movimiento de materiales

El punto relacionado con el transporte de materiales por pasillos y área circundante, ayudas mecánicas para el movimiento de materiales y correcto almacenaje de los mismos, no corresponde directamente con la evaluación del puesto si no con la planificación general del funcionamiento de la cadena de montaje.

Con respecto a la manipulación de materiales producida en el puesto, se evaluará sobre todo la correcta utilización de herramientas para cada tarea, las posibles posturas forzadas de muñeca en su uso, y sobre todo la utilización de herramientas de peso inadecuado para la tarea desempeñada.

En este caso se considerará una planificación inadecuada cuando:

- Se empleen herramientas que no son las específicas para la tarea que se realiza.
- Cuando obliga a una postura forzada de la muñeca (flexión, extensión, giro o inclinación hacia un lado elevados).
- El mango de las herramientas no es cómodo (grosor, longitud, forma o material inadecuados).
- El trabajador utiliza herramientas de peso > 2.3 kg que no están suspendidas.
- El trabajador utiliza herramientas de peso > 0.4 kg en tareas de precisión.
- Las herramientas más pesadas y de uso frecuente en una misma zona de trabajo no están suspendidas.
- Hay herramientas en mal estado.
- Se transmiten vibraciones molestas de las herramientas.

Nuestro sistema

Nuestros dispositivos pueden ser colocados en la muñeca para poder observar posturas inadecuadas, torsiones y vibraciones excesivas. En próximas

implementaciones se planea incorporar un sistema de detección de peso o fuerza ejercida, para posibilitar la evaluación de dichos puntos.

Además, nuestro sistema permitirá seleccionar entre si la postura se ha adquirido libre de peso, o en cambio, con levantamiento de cargas (tanto si es con postura de pie, sentada o transporte de cargas). Se habrá estipulado por tanto el cálculo de un peso aceptable de levantamiento, que se comparará con el peso real de la carga manipulada. Si el peso real manipulado es mayor que el Peso aceptable del levantamiento, el riesgo no es tolerable por lo que deben rediseñarse las condiciones de trabajo. Este estudio es todavía más complejo, y deberá completarse con el resto de recomendaciones establecidas en el documento del Instituto Nacional de Seguridad e Higiene en el Trabajo, incluyendo la aplicación del método de la Universidad de Lovaina , conocido como método LUBA (INSHT, 2004).

3.1.4 Riesgo por la postura o repetitividad

En el documento oficial del Ministerio de Trabajo y Asuntos Sociales, el análisis del riesgo postural se considera a realizar mediante la observación directa o de video. Por tanto, se estipula en las directrices que los expertos deberán evaluar de forma visual la realización de posturas incorrectas y decidir si se encuentran o no dentro de los rangos articulares estipulados. Éste análisis es uno de los principales motivos para la incorporación de nuestro sistema a los procesos de valoración ergonómica de puestos de trabajo, ya que pretendemos pasar de una validación meramente visual a la obtención de datos de medida precisos y automáticos que puedan aportar fiabilidad y factibilidad a la forma de obtener dicha evaluación actualmente. Como hemos constatado en el apartado **1.3.3 Sector valoración ergonómica industrial**, los métodos previstos de evaluación no son los más adecuados para las capacidades tecnológicas actuales y deberían ser actualizados por sistemas de adquisición de datos por sensores para evitar evaluaciones erróneas y derivas en las medidas debidas a diferencias en la percepción de cada experto.

En este apartado se pretende evaluar el riesgo asociado a la carga física de las posturas de trabajo mediante la denominada como Ficha de Evaluación. En esta Ficha se analiza de manera independiente el tronco, los brazos, las muñecas, el cuello y las piernas, considerando varios aspectos como la posición del segmento corporal (elevación, inclinación hacia un lado, giro, etc.), la frecuencia de movimientos (repeticiones/minuto), si la postura se mantiene de manera sostenida (estática o más de un minuto) o si la postura se adopta mucho tiempo (no necesariamente seguido) durante los distintos puestos.

TRONCO: Flexión/ extensión

- Tronco flexionado $>20^\circ$ de manera sostenida (estática) y sin apoyar.
- Tronco flexionado $>20^\circ$ de manera repetida (>2 veces/minuto).
- Tronco flexionado $>60^\circ$ mucho tiempo (no necesariamente seguido) durante la jornada.

- Tronco en extensión (inclinado hacia detrás) sin apoyo, de manera sostenida (estática).
- Tronco en extensión (inclinado hacia detrás) sin apoyo, de manera repetida (>2 veces/minuto).
- Tronco en extensión (inclinado hacia detrás) sin apoyo, mucho tiempo (no necesariamente seguido) durante la jornada.

TRONCO: Inclinación lateral o giro

- Tronco inclinado hacia un lado o girado (claramente visible) de manera sostenida (estática).
- Tronco inclinado hacia un lado o girado (claramente visible) mucho tiempo (no necesariamente seguido) durante la jornada.
- Tronco inclinado hacia un lado o girado (claramente visible) de manera repetida (>2 veces/minuto).

BRAZO

- Brazo elevado >20° de manera sostenida (estática), y sin apoyar.
- Brazo elevado entre 20 y 60°, de manera repetida (>10 veces/minuto).
- Brazo elevado >60°, de manera repetida (>2 veces/minuto).
- Brazo elevado >60° mucho tiempo (no necesariamente seguido) durante la jornada.
- Brazo en extensión (hacia detrás), de manera sostenida (estática).
- Brazo en extensión (hacia detrás) mucho tiempo (no necesariamente seguido) durante la jornada.
- Brazo en extensión (hacia detrás), de manera repetida (>2 veces/minuto).

MUÑECA / CODO

- Muñeca muy flexionada/extendida, muy inclinada lateralmente o muy girada, de manera sostenida (estática).
- Muñeca muy flexionada/extendida, muy inclinada lateralmente o muy girada, de manera repetida (>2 veces/minuto).
- Codo muy flexionado o muy extendido de manera sostenida (estática) o repetida (>2 veces/minuto).

CABEZA Y CUELLO: Línea de visión

- El trabajador tiene su línea de visión por debajo de 40° respecto de la horizontal, de manera sostenida (estática).
- El trabajador tiene su línea de visión por debajo de 40° respecto de la horizontal mucho tiempo (no necesariamente seguido) durante la jornada.
- El trabajador tiene su línea de visión por debajo de 40° respecto de la horizontal, de manera repetida (>2 veces/ minuto).
- El trabajador tiene su línea de visión por encima de la horizontal, de manera sostenida (estática).

- El trabajador tiene su línea de visión por encima de la horizontal mucho tiempo (no necesariamente seguido) durante la jornada.
- El trabajador tiene su línea de visión por encima de la horizontal, de manera repetida (>2 veces/minuto).

CABEZA Y CUELLO: Inclinación lateral o giro

- Cabeza inclinada hacia un lado o girada (claramente visible) de manera sostenida (estática).
- Cabeza inclinada hacia un lado o girada (claramente visible) mucho tiempo (no necesariamente seguido) durante la jornada.
- Cabeza inclinada hacia un lado o girada (claramente visible) de manera repetida (>2 veces/minuto).

PIERNAS

- El trabajador está de pie, sin desplazarse (por ejemplo, frente a un banco de trabajo o una cinta transportadora), >2 horas seguidas.
- El trabajador mantiene una postura de pie (estática) con las rodillas flexionadas.
- El trabajador está de pie y flexiona las rodillas de manera repetida (>2 veces/minuto).
- El trabajador mantiene una postura de pie (estática) en la que carga casi todo el peso del cuerpo sobre una de las dos piernas.
- El trabajador está de rodillas o en cuclillas de manera sostenida (estática) o mucho tiempo (no necesariamente seguido) durante la jornada.

Finalmente, para una evaluación más precisa en la aplicación final de evaluación ergonómica (aplicación Python) se han tomado algunas de las recomendaciones establecidas por los siguientes métodos para complementar al anteriormente mencionado método LUBA:

- Método Ergo/IBV (IBV, 2000a)
- OCRA (Occhipinti, 1998)
- REBA (Hignett y McAtamney, 2000)
- RULA (McAtamney y Corlett, 1993)
- Strain Index (Moore y Garg, 1995)

3.2 Integración del sistema en la valoración ergonómica

Comentados los principales puntos de riesgo del diseño de puestos de trabajo, pasaremos a definir cómo nuestro sistema puede ofrecer una solución automatizada de la valoración ergonómica de puestos de trabajo en cadenas de producción, principalmente de automoción.

Nuestro sistema recibe los datos obtenidos por los sensores colocados en el trabajador en las distintas articulaciones como el tronco, los brazos, las muñecas, el cuello y las piernas entre otros. Cada articulación requiere un mínimo de dos sensores colocados, uno para servir como base comparativa o eje inicial del movimiento y otro como eje final del mismo. Así pues, los sensores determinarán el ángulo entre los dos, obteniendo el rango articular del movimiento en grados (otros factores como aceleración, velocidad o fuerza magnética/gravedad también serán obtenidos) y replicándose el movimiento completo en un avatar en la aplicación de visionado en tiempo real. Finalmente, todos los valores instantáneos de los ángulos serán almacenados en un archivo codificado que será almacenado a su vez en la nube para posteriores análisis. Los resultados podrán ser presentados de distinta forma según las necesidades, pues el sistema recuperará estos datos de la nube y trabajará con ellos para obtener distintos parámetros evaluadores sobre el correcto desempeño del puesto. Comentaremos más al respecto en el apartado de las pruebas realizadas.

Un ejemplo de estos parámetros serán los vertidos por nuestro software de análisis estadístico, incluyendo porcentajes de ocupación respecto a determinado grado de movimiento máximo establecido para analizar la adecuación a los criterios de planificación. Estos resultados podrán ser reflejados en tablas, o gráficas de fácil observación y recopilados en informes técnicos de generación semi-automática, sólo necesitando la elección de parámetros a incorporar al informe y los niveles de movimiento deseados.

Cabe aclarar que el sistema que controla los sensores es independiente del software que analiza los datos obtenidos y proporciona tablas, gráficas e informes. Por la disposición en la nube de los registros guardados, se hace posible la independencia de ambos sistemas, contando el sistema de análisis con un desarrollo basado en Python, y el sistema de visionado 3D en tiempo real y recogida de datos en Unity3D.

Todo ello posibilita que dispongamos de un amplio número de posibilidades de evolución del sistema, ya sea desde el punto de vista de Unity3D, optimizando el sistema de visualización en tiempo real, con más opciones como el análisis in vivo del movimiento o incorporando un entorno más realista de acorde con el puesto en concreto; o desde el punto de vista del software de análisis estadístico, con nuevas opciones analíticas o de visualización para facilitar el estudio de resultados finales.

3.3 Aplicación del sistema de valoración ergonómica

Nuestro sistema pretende incorporar a la suite definida en el punto **2.3.3 Suite de rehabilitación**, un sub-módulo encargado del proceso de valoración ergonómica. Compartirá la estructura del almacenamiento en la nube, pero la aplicación final del usuario será diferente, aunque en una primera versión heredará el sistema Unity3D basado en TELEKIN.

Permitirá a los usuarios visualizar en tiempo real la captura de movimientos del puesto y poder realizar un pilotaje para simulación de puestos, anterior a la puesta en práctica de una determinada modificación a un puesto de trabajo. Esto facilitará a los diseñadores de los puestos, poder evaluar los riesgos y futuras lesiones en que incurrirán los trabajadores ante un determinado plan de modificación de un puesto. Además, las estadísticas finales de evaluación estarán disponibles en cualquier dispositivo para su consulta y dispondrán de distintas opciones de acuerdo a las necesidades de valoración como son: peso cargado, repetición de posturas de riesgo, superación de rangos articulares recomendados, etc.

La visualización final de los datos capturados podrá ser tanto en tiempo real por medio de un avatar y movimientos 3D, como reproducir los datos guardados de un puesto a posteriori (en cualquier momento tras la captura) también en un avatar 3D, como en forma de informe resumido con gráficas y tablas de la evaluación (todo ello accediendo a los datos de los puestos almacenados en la nube).

Capítulo 4 Sistema Hardware

En este trabajo fin de máster se abarcará la realización de los prototipos para distintas aplicaciones exigidas en distintos puntos del proyecto.

En este capítulo entraremos a detallar las tres versiones Hardware desarrolladas para abordar la misma solución, con distintas características implementadas en cada una según surgían las necesidades del proyecto. La primera versión busca ofrecer una solución cableada al problema completo: un dispositivo de detección de movimientos que supliera al sistema Kinect®. La necesidad de la misma venía anticipada por las distintas limitaciones que presenta la cámara Kinect®, así como los distintos sistemas de captura de movimiento basados en imagen, como veremos en el apartado **6.4.1 Limitaciones de sistemas basados en cámaras**. La segunda versión busca ofrecer un sistema de detección de movimientos totalmente inalámbrico para responder a detección de movimientos simples con una única articulación a detectar, compatible con el sistema de juegos serios TELEKIN. Y la última versión consigue condensar las capacidades de los dos primeros prototipos en una versión totalmente inalámbrica, pero permitiendo además la interconexión inalámbrica entre dispositivos para conseguir monitorizar más de una articulación simultáneamente y así como tener compatibilidad con TELEKIN, consiguiendo una alternativa viable a sistemas de detección cinemática basados en cámara como Kinect®.

4.1 Dispositivo Hardware V1: versión cableada

La primera versión del dispositivo de monitorización de movimientos está constituida por un Arduino Pro mini, un Módulo Bluetooth HC-05 para la comunicación, el Adafruit LiIon/LiPoly Backpack Add-On para la carga de la batería de 500 mAh incorporada y un multiplexor I2C Adafruit TCA9548A 1-to-8 mux para conectar al procesador los distintos sensores MPU 6050 6DoF unidos por cable ethernet plano.

4.1.1 Placa con la Unidad de Procesamiento

Se escogió en primer lugar, para la versión cableada del dispositivo, el Arduino Pro mini por ser una placa de microcontrolador basada en ATmega328, con el que ya habíamos trabajado y que nos permitía reaprovechar el trabajo realizado en proyectos anteriores. Además, permitía la programación mediante el Arduino IDE, solamente incorporando las librerías modificadas de los sensores utilizados (MPU6050). Pasaremos a comentar sus principales características:

Posee 14 pines digitales de entrada/salida (de los cuales 6 se pueden usar como salidas PWM), 6 entradas analógicas, un resonador integrado, un botón de reinicio y orificios para el montaje de los pines. El resto de los pines serán los de la UART SPI e I2C (permite la conexión de 1 de cada tipo). La corriente por pin de I/O será de 40mA, y contará con una memoria Flash de 32KB, de los cuales 2KB estarán reservados al bootloader. Incorpora también 2KB de SRAM y 1KB de EEPROM, además de un procesador con velocidad de reloj de 8Hz en la versión de 3.3V.

Se puede conectar un conector de seis pines a un cable FTDI o placa de arranque para proporcionar alimentación y comunicación USB a la placa. No cuenta por tanto de chip conversor UART a USB y deberá ser conectado por una placa intermedia como la utilizada, basada en el chip FT232RL.

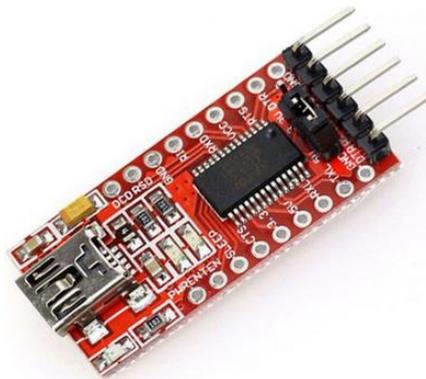


Figura 2: FT232RL FTDI USB to TTL Serial Adapter Module, modelo utilizado.

La placa del Arduino Pro Mini, por su parte, viene sin encabezados preinstalados, lo que permite el uso de varios tipos de conectores o la soldadura directa de cables. El diseño del pinout es compatible con el Arduino Mini.

Hay dos versiones del Pro Mini. Uno funciona a 3.3V y 8MHz, el otro a 5V y 16MHz. Para nuestro prototipo, escogimos la versión de 3.3V, ideal para alimentación mediante batería. Podemos observar la función de cada uno de sus pines:

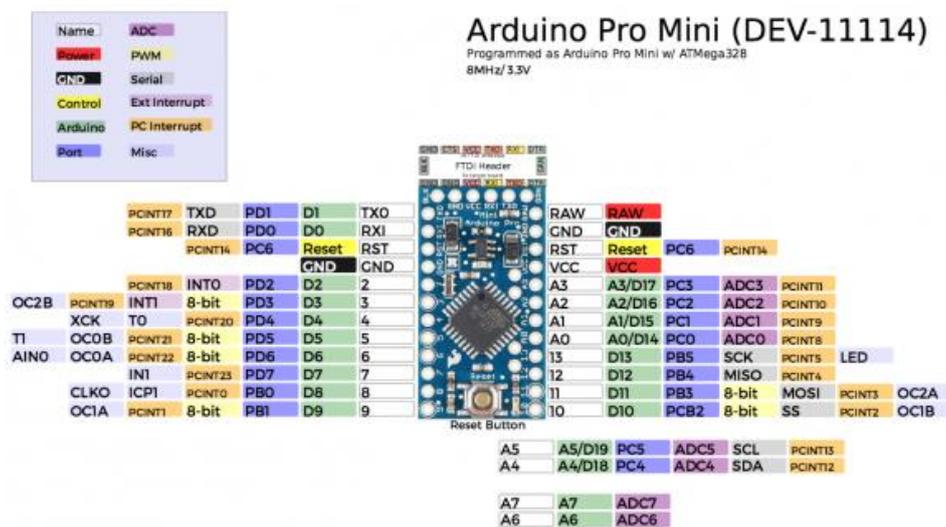


Figura 3: Arduino Pro Mini pinout.

4.1.2 Sensores

Una Unidad de Medida Inercial, o IMU, es un dispositivo capaz de medir la fuerza (aceleración) y la velocidad. Genéricamente consta de un Acelerómetro y un Giroscopio como mínimo, y en este caso (MPU-6050) no presenta ningún sensor extra. El MPU-6050 es un sensor inercial de 6DOF (6 grados de libertad), lo que significa 3 ejes por acelerómetro y 3 por giroscopio, combinados junto con un procesador a bordo de Movimiento Digital (DMP) que nos permite obtener 3 ejes más en función de estos 6 primeros (emulando los 9DOF).

El giroscopio cuenta con un rango de escala de ± 250 , ± 500 , ± 1.000 y ± 2.000 $^{\circ}/s$ (dps) y el acelerómetro un rango de ± 2 g, ± 4 g, ± 8 g, y ± 16 g. Opera con 3.3 voltios, aunque algunas versiones incluyen un regulador que permite conectarlo a 5V.

Resumen de las características principales:

- Salida I2C digital de 6 o 9 ejes. Incorpora MotionFusion para el cálculo de las matrices de rotación, cuaterniones, ángulos de Euler, etc.
- Giroscopio de 3 ejes con una sensibilidad de hasta 131 LSBs/dps and rango de escala de ± 250 , ± 500 , ± 1000 , and ± 2000 dps
- Acelerómetro de 3 ejes programable of ± 2 g, ± 4 g, ± 8 g and ± 16 g
- Procesador a bordo Digital Motion Processing™ (DMP™)
- Algoritmos internos para la calibración del dispositivo
- Incluye un sensor de temperatura digital con una oscilación de +1%.
- Es el sensor de menor coste de los estudiados, unos 6/7€ en modelos de fabricación china.

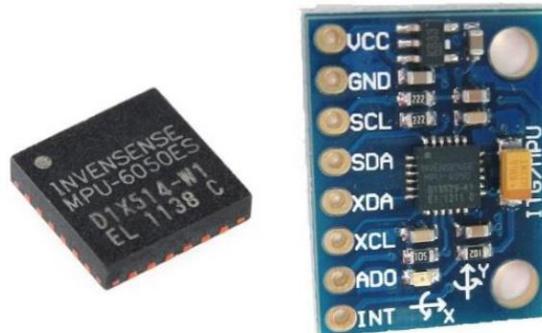


Figura 4: Chip MPU6050 de Invensense (izq.) vs placa completa MPU6050.

Para corregir los errores y ruido, que serán grandes por basar el cálculo del ángulo en solamente 2 valores medidos por eje (uno de aceleración y otro de giroscopio), se utilizan distintos métodos; nosotros utilizaremos el procesador digital de movimiento (DMP) incorporado dentro de la MPU. Éste descargará el cálculo de los algoritmos de procesamiento de movimiento desde el procesador host del MPU 6050 hasta el procesador digital de movimiento, los datos resultantes pueden ser leídos de los registros de la DMP, o pueden almacenarse en una FIFO.

Su propósito será descargar los requisitos de temporización y la potencia de procesamiento del procesador anfitrión. Además, el DMP tiene acceso a uno de los pines externos de la MPU, que pueden ser utilizados para la generación de interrupciones.

Este sensor es bastante preciso, ya que contiene una conversión hardware de 16 bits de A/D por cada canal, cuyo objetivo será digitalizar las salidas del acelerómetro. Para ello captará los canales x, y, z al mismo tiempo. Como se ha comentado, el sensor utiliza el bus I2C para conectarse con el Arduino.

Finalmente, cabe aclarar, que nuestro sistema cableado utilizará 2 MPU 6050 para brazo y antebrazo respectivamente, y el tercer sensor, colocado en el mismo encapsulado que el Arduino, el bluetooth y la batería, será un BNO055 para mayor estabilidad en una zona que comúnmente se encuentra en posición vertical como es la espalda. Comentaremos dicho sensor en detalle próximos apartados.

4.1.3 Multiplexación cableada de sensores

En este caso, los sensores estarán interconectados con el arduino, varios simultáneamente, gracias a una placa con un multiplexor I2C: la TCA9548A 1-to-8 mux de Adafruit.

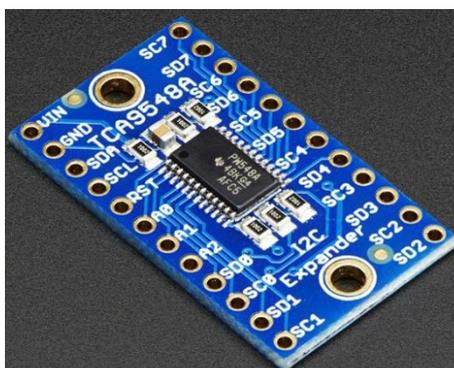


Figura 5: Placa TCA9548A 1-to-8 mux de Adafruit.

Es una forma de obtener hasta 8 dispositivos de la misma dirección I2C conectados a un microcontrolador; actuando como un controlador de acceso, enviando los comandos al conjunto seleccionado de pines I2C. Permite que, ocupando únicamente 1 par de pines I2C del Arduino, obtengamos hasta 8 posibles dispositivos conectados a él por I2C, controlando la dirección de cada dispositivo por Software.

Así conseguiremos recibir la información de todos los MPU6050 simultáneamente. Y será el procesador el encargado de convertir dichos datos en información útil sobre el posicionamiento de cada sensor, equivalente a una parte del cuerpo concreta.

Por último, aclararemos que los sensores se unieron físicamente mediante un cable ethernet plano, pues resulta la elección más adecuada ya que cuenta dentro del cable de protección con los 8 pines necesarios para nuestra conexión.

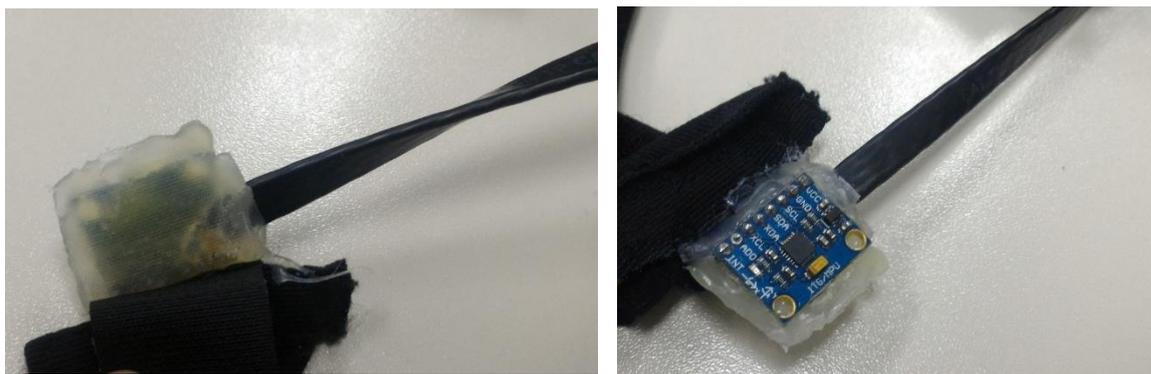


Figura 6: Cable ethernet plano con corte.

Esto es así, ya que necesitamos de 4 pines por sensor: VCC, GND, SCL y SDA; dos de alimentación y dos de I2C, aunque los dos de alimentación son comunes y solamente necesitaríamos 6 cables. Como veremos en las siguientes imágenes, se realizó un corte a longitud intermedia del cable ethernet, de longitud la distancia entre la espalda donde irá el encapsulado principal y el brazo donde irá el primer sensor. Este corte deberá ser calculado para sólo dejar a la vista 4 de los 8 cables del ethernet, y se procederá a la soldadura a los 4 pines del primer sensor. Así, quedarán ocultos y protegidos los 4 pines del segundo sensor que se soldarán a la distancia entre el brazo y el antebrazo.



Figura 7: Conexión entre los dos MPU6050 mediante ethernet plano.



*Figura 8: Conexión del último sensor (antebrazo).
Vista inferior (izq.) y posterior (der.)*

4.1.4 Bluetooth

La placa encargada de facilitarnos la comunicación del sistema con un dispositivo móvil o PC mediante Bluetooth, será el módulo HC-05 que puede configurarse tanto como Maestro como Esclavo, y que además dispone de bastante más parámetros de configuración que su hermano pequeño, el módulo HC-06 (que además sólo podía ser esclavo).

El aspecto externo es bastante similar entre ellos, pero su principal diferencia es tanto el firmware flasheado en la placa, como por los pines que cuentan, 4 para el HC-06 y 6 para el HC-05.

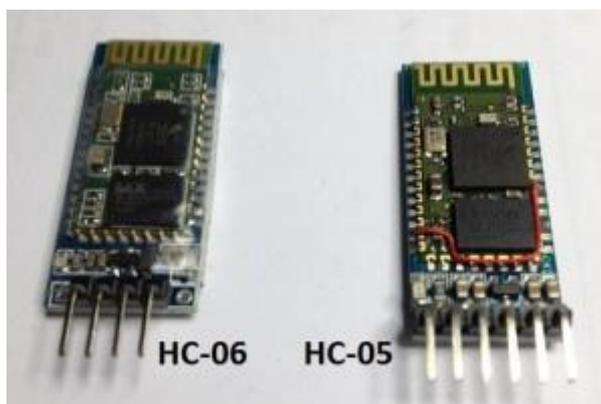


Figura 9: Módulo Bluetooth HC-06 (izq.) vs HC-05 (der.)

El HC-05 es ligeramente más complicado de colocar en modo comandos que el HC-06, que entra en modo de programación automáticamente en su encendido, si no hay nadie conectado a él. El HC-05, por su parte, requiere una cierta manera de arrancado para el modo de recepción de comandos, concretamente que el pin KEY, este en HIGH en cuanto encendamos el modulo. Este modo se denomina modo de comandos AT y permite reprogramarlos con nuestras preferencias, las que se

mantendrán hasta que decidamos cambiarlas mediante estos comandos. Así pues, adjuntaremos la tabla de comandos principales del módulo, de la que cabe destacar que algunos comandos varían según el modelo del chip de la placa (como la elección del BaudRate o el nombre):

AT COMMAND LISTING

COMMAND	FUNCTION
AT	Test UART Connection
AT+RESET	Reset Device
AT+VERSION	Query firmware version
AT+ORGL	Restore settings to Factory Defaults
AT+ADDR	Query Device Bluetooth Address
AT+NAME	Query/Set Device Name
AT+RNAME	Query Remote Bluetooth Device's
AT+ROLE	Query/Set Device Role
AT+CLASS	Query/Set Class of Device CoD
AT+IAC	Query/Set Inquire Access Code
AT+INQM	Query/Set Inquire Access Mode
AT+PSWDAT+PIN	Query/Set Pairing Passkey
AT+UART	Query/Set UART parameter
AT+CMODE	Query/Set Connection Mode
AT+BIND	Query/Set Binding Bluetooth Address
AT+POLAR	Query/Set LED Output Polarity
AT+PIO	Set/Reset a User I/O pin

Tabla 2: Principales comandos AT del módulo HC-05

Para su conexionado, simplemente deberemos guiarnos por el nombre de las patillas que se encuentra serigrafiado en la cara inferior, como vemos en la imagen para el Arduino Uno, pues será equivalente en el Arduino Pro mini:

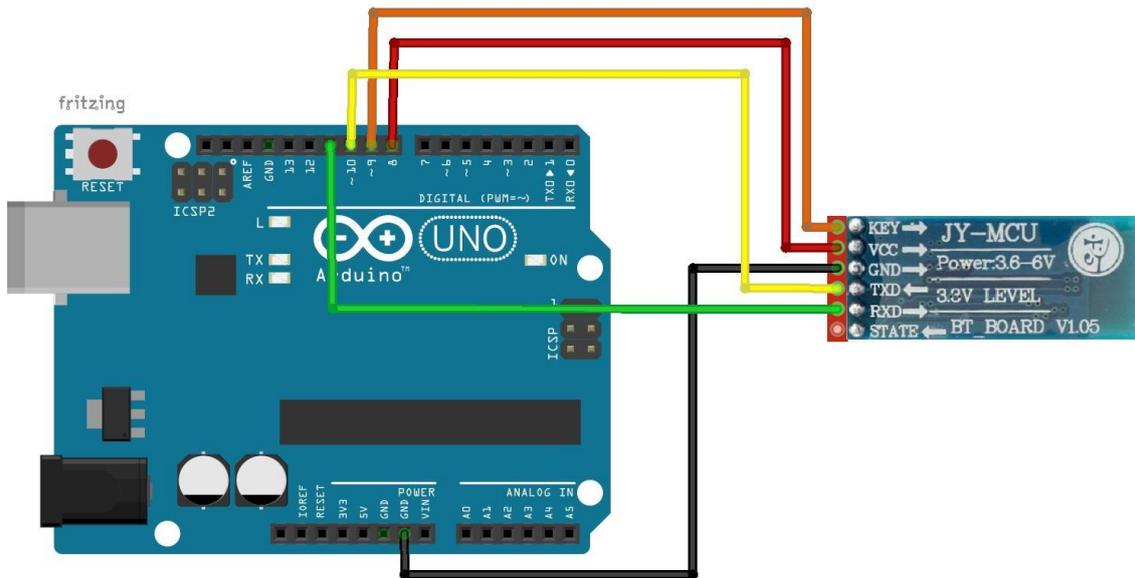


Figura 10: Conexión del módulo Bluetooth HC-05 en el Arduino Uno.

Una vez configurado el módulo, la información será transmitida cuando se establezca comunicación bluetooth con nuestro dispositivo móvil o PC.

4.1.5 Alimentación

Las baterías de Polímero de iones de litio (también conocido como “lipo” o “LiPoly”) son delgadas, ligeras y de gran capacidad. En nuestro caso la salida deberá ser de por lo menos 3.3V o cercana a 5V si lo conectamos a una entrada “5V” que cuente con un divisor de tensión. La batería a utilizar en nuestro caso tendrá una capacidad de 500mAh.

La batería dispondrá de un conector de 2 pines JST-PH y circuito de protección. El circuito de protección incluido evita que la tensión de la batería suba demasiado (exceso de carga) o baje (descarga excesiva), lo cual, conllevaría posiblemente la muerte súbita de la batería por una descarga que llegue a los 3.0V. También estará protegida contra cortocircuitos de salida y contra subidas excesivas de temperatura gracias al termistor interno del chip de carga de la batería.

Sin embargo, incluso con esta protección, es muy importante que únicamente se utilice un cargador de baterías Li-Ion / LiPoly de tensión constante a una velocidad entre 150 mA y 500mA.

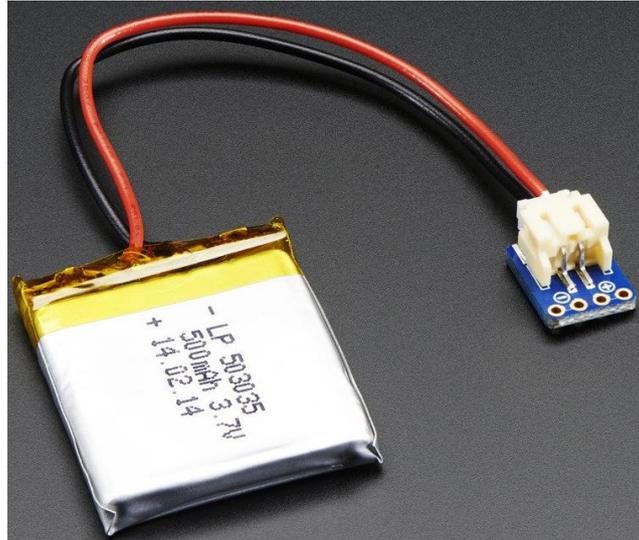


Figura 11: batería de litio utilizada con conector de 2 pines JST-PH.

Dicha batería se conectará al Arduino gracias a la placa Adafruit Lilon/LiPoly Backpack Add-On, que permite al soldarse encima de la placa del Arduino Pro Mini, que la batería se recargue automáticamente al enchufar el cable USB al Arduino, cambiando a la batería recargable al desenchufarlo.

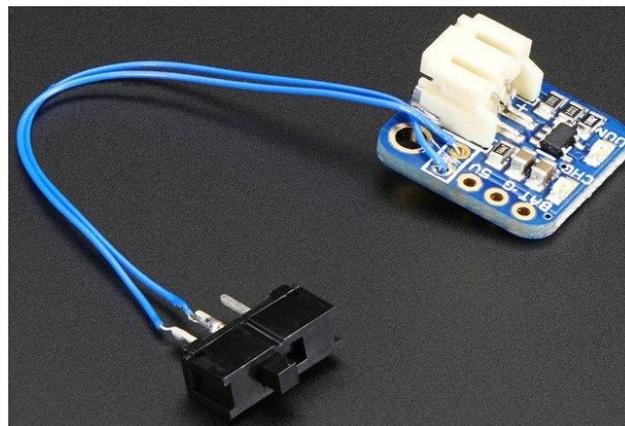
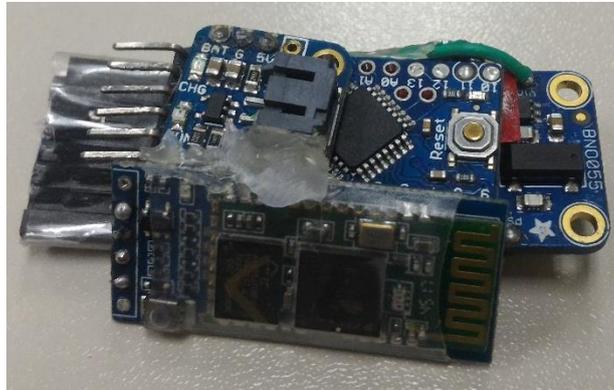


Figura 12: Adafruit Lilon/LiPoly Backpack Add-On con interruptor añadido.

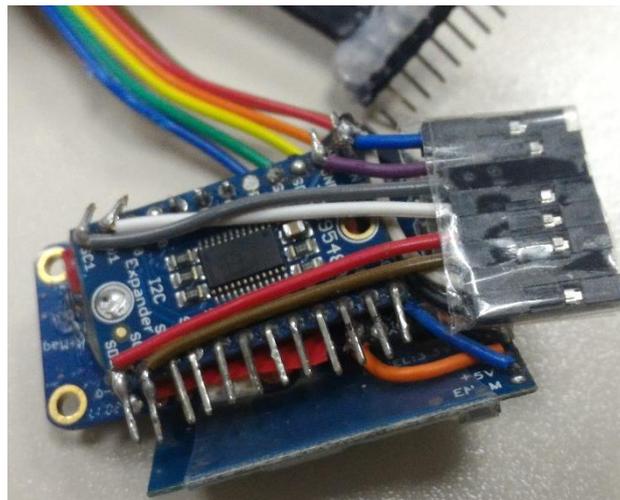
La carga se realiza en tres etapas: primero una carga de preconditionamiento, luego una carga rápida de corriente constante y finalmente una carga lenta de voltaje constante para mantener la batería recargada. La corriente de carga es de 100 mA por defecto, por lo que funcionará con cualquier tamaño de batería y puerto USB. Se puede cambiar al modo 500mA simplemente soldando el jumper disponible en la parte inferior, para cuando solo se estén cargando baterías con un tamaño de 500mAh o mayores. Además, permite de forma muy sencilla introducir un interruptor de encendido al sistema.

4.1.6 Encapsulado final

Por último, veremos el resultado del encapsulado final del sistema, tanto en el núcleo de procesamiento (Arduino + HC-05 + I2C MUX + Batería) como en el resto del sistema cableado:



*Figura 13: Núcleo del Sistema (Arduino + BNO055 + HC-05 + I2C MUX + Batería)
Cara superior.*



*Figura 14: Núcleo del Sistema (Arduino + BNO055 + HC-05 + I2C MUX + Batería)
Cara inferior.*

Como podemos apreciar en la cara superior, el sistema cuenta con las patillas “al aire” de la conexión al convertidor UART a USB, pues se pretende permitir su programación, pero sin la necesidad de introducir el convertidor dentro del encapsulado, esto es, conectándolo únicamente en su programación (sin soldarlo).

Además, podemos observar en la cara inferior, cómo el cable ethernet plano se conectará a una cabecera de pines sin estar soldado, permitiendo la desconexión entre el núcleo y los dos MPU6050, en caso de ser necesaria.

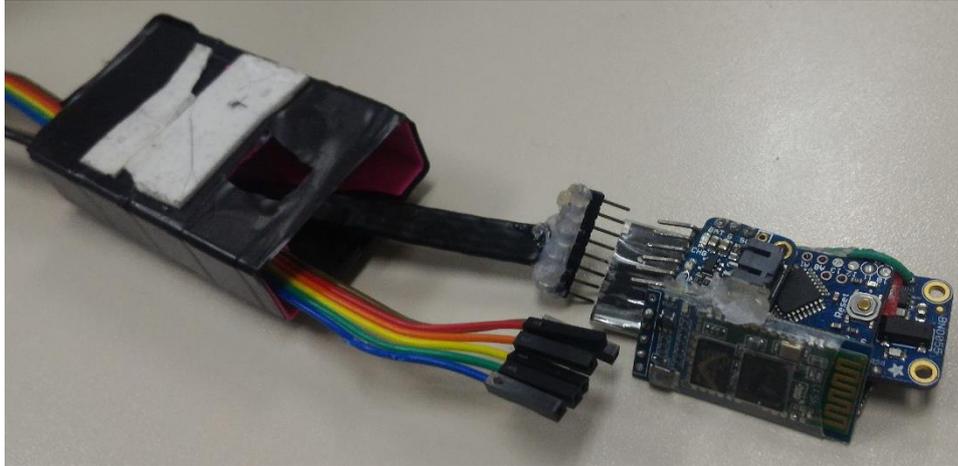


Figura 15: Introducción del núcleo del sistema en el encapsulado con colocación en la espalda.



Figura 16: Sistema completo listo para su colocación.

4.2 Dispositivo Hardware V2: versión BT autónoma

Basada en la versión anterior, se ideó un dispositivo que pudiera monitorizar únicamente una articulación, y enviar los datos por bluetooth en tiempo real.

Pretendía funcionar de forma similar al núcleo central del dispositivo hardware V1, pero con un único sensor, reduciendo de forma drástica el tamaño, para que pudiera colocarse hasta en los lugares más complicados como la muñeca.

Para este modelo, se utilizó como procesador el Arduino Pro Trinket, con un MPU6050 como sensor, una batería de 240 o 500 mAh según el caso, y el Adafruit LiIon/LiPoly Backpack Add-On.

4.2.1 Placa con la Unidad de Procesamiento

Como placa de procesamiento se escogió el Arduino Pro Trinket, al observar un mejor rendimiento que con el Arduino Pro mini, sin aumentar prácticamente el tamaño.

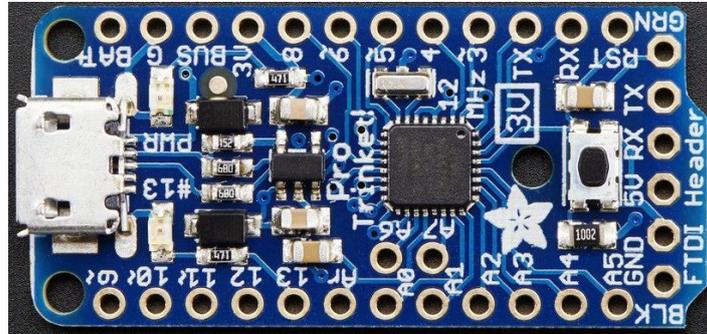


Figura 17: Arduino Pro Trinket 3V

El Arduino Pro Trinket combina el núcleo del chip Arduino ATmega328, como en el caso anterior. Es como un Arduino Pro Mini con más pines y con USB incluido en la placa (con convertor UART-USB incluido). Frente a éste, cuenta con 18 GPIO, 2 entradas analógicas adicionales, 28K de flash y 2K de RAM.

La PCB del Pro Trinket, aun con su reducido tamaño, tiene la misma capacidad que un Arduino UNO. En este caso concreto, El Pro Trinket 3V utiliza el chip Atmega328P, que es el mismo chip central en el Arduino UNO/Mini etc. Sin embargo, utiliza una lógica de 3V y un reloj de 12MHz, lo que significa que nos permite una configuración de baja potencia y baja tensión.

Incluye un regulador de potencia de 3,3 V con capacidad de salida de 150 mA y caída ultra-baja. Permite hasta 16 V de entrada, con protección de polaridad inversa, protección térmica y de límite de corriente. La carga, además, estará indicada inequívocamente, al cambiar automáticamente el color del LED de alimentación incluido en la placa, de verde a rojo al comenzar la carga. Por último, incluye un botón de reset para ingresar al gestor de arranque o reiniciar el programa.

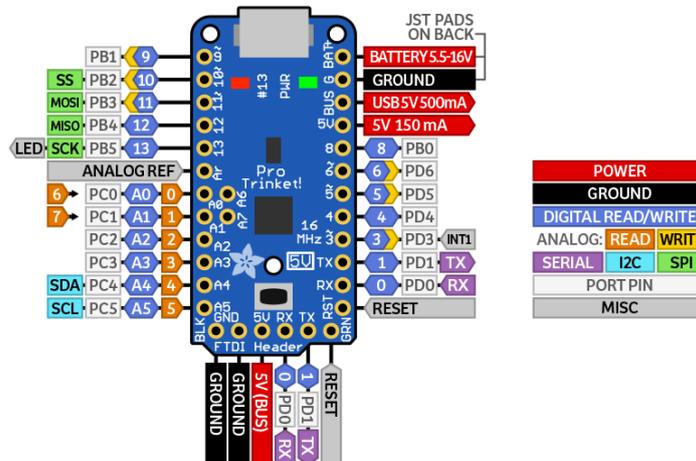


Figura 18: Arduino Pro Trinket 5V pinout (equivalente al del modelo de 3V)

4.2.2 Encapsulado final

En este prototipo, no hemos comentado ninguno de los demás apartados por ser un análisis análogo al realizado en el modelo anterior, al compartir la mayoría de los componentes utilizados. Nos centraremos ahora, por tanto, en el encapsulado final del prototipo:



*Figura 19: Dispositivo bluetooth autónomo, encapsulado final.
Vista lateral izq. (izq.), trasera (centro) y lateral der. (der.)*



Figura 20: Dispositivo bluetooth autónomo colocado para medidas de muñeca.

4.3 Dispositivo Hardware Definitivo (ESP32)

En este apartado se comentará la última versión del dispositivo, totalmente inalámbrica y basada en el procesador ESP32 de *espressif*. Dicha implementación, constituye la evolución lógica de los planteamientos anteriores, eliminando la conexión cableada de sensores, y permitiendo el escalado del sistema llegando a permitir la incorporación de 8 puntos corporales distintos.

Esta versión, contará de batería, procesador, sensor y antena todo en el mismo encapsulado; y definirá el sistema como conjunto de uno o más de dichos dispositivos. Como veremos, será necesario que uno de ellos se establezca como

Maestro o “Master”, para recibir los datos del resto de unidades y transmitirlos de forma conjunta al dispositivo externo encargado de visualizar los datos en la aplicación (de forma gráfica normalmente). Cabe aclarar que tanto el dispositivo Maestro como sus Esclavos, serán intercambiables pues poseerán la misma base Hardware, aunque estarán flasheados con una versión distinta del Firmware.

4.3.1 Placa con la Unidad de Procesamiento

Se ha elegido la placa de desarrollo Wemos ESP32 D1 mini, que lleva incrustado el módulo WROOM ESP32 de *espressif*. Es una placa de desarrollo al modo de las Arduino, y presenta una alternativa más barata y de menor tamaño a las distintas placas de desarrollo oficial como la ESP32-DevKitC o la ESP32-PICO-KIT que por su forma alargada no se ajustaban a nuestros requisitos de dimensiones de placa.

El WROOM ESP32 utilizado incluye como procesador al ESP32-D0WDQ6 (D0WD en otras versiones) que contiene dos microprocesadores LX6 Xtensa® de 32 bits de baja potencia.

La **memoria interna**, a su vez, incluye:

- 448 kB de ROM para el arranque y las funciones principales.
- 520 kB de SRAM en el mismo chip para datos e instrucciones.
- 8 kB de SRAM en el RTC, la denominada RTC FAST Memory, que se puede utilizar para el almacenamiento de datos; es accedido por la CPU principal durante el arranque del RTC desde el modo de Deep-sleep.
- 8 kB de SRAM en RTC, la denominada RTC SLOW Memory, que es accedido por el coprocesador durante el modo de sueño profundo.
- 1 kbit de eFuse: se utilizan 256 bits para el sistema (dirección MAC y configuración de chip) y los 768 bits restantes estarán reservados para las aplicaciones de los clientes, incluidos el cifrado flash y el ID de chip.

Respecto a la utilización de Flash y SRAM **externos**, el ESP32 admite tanto múltiples memorias flash QSPI externas como chips SRAM conectados por SPI. Permitiendo distintos modos de funcionamiento como el SPI estándar, el SPI dual, o el Quad SPI (ESP32 Technical Reference Manual V3.6, 2018). Por otro lado, también soporta la encriptación/descriptación hardware basada en AES para proteger los programas y datos de los desarrolladores introducidos en la flash.

El ESP32 puede acceder a la flash QSPI externa y a la SRAM a través de cachés de alta velocidad. Permite el mapeo simultáneo de hasta 16 MB de flash externa en el espacio de la memoria de la instrucción de la CPU y en el espacio de la memoria de solo lectura:

- Cuando el flash externo se asigna al espacio de memoria de la CPU, se pueden mapear hasta 11 MB + 248 KB a la vez.

- Cuando el flash externo se asigna a un espacio de memoria de datos de solo lectura, se pueden mapear hasta 4 MB a la vez. Se admiten lecturas de 8 bits, 16 bits y 32 bits.

La SRAM externa se puede mapear en el espacio de la memoria de datos de la CPU. Se admite SRAM de hasta 8 MB y se pueden mapear hasta 4 MB a la vez. Se admiten lecturas y escrituras de 8 bits, 16 bits y 32 bits.

El ESP32-WROOM-32 integra 4 MB de flash SPI externo. El flash SPI de 4 MB se puede mapear en la memoria en el fragmento de código de la CPU, y admite el acceso a 8, 16 y 32 bits. El flash SPI integrado está conectado a GPIO6, GPIO7, GPIO8, GPIO9, GPIO10 y GPIO11. Estos seis pines no se podrán usar como GPIO regulares.

4.3.2 Sensores

En el Trabajo Fin de Grado previo a este Trabajo Fin de Máster, se realizó una explicación de los distintos dispositivos sensores comerciales disponibles para la medida de movimiento inercial. No se ha percibido ninguna evolución clara en este aspecto, por lo que seguiremos considerando el BNO055 como la opción más adecuada para nuestro proyecto.

No obstante, resumiremos las principales características del sensor elegido y aclararemos su propósito:

BNO055 9 DOF

Basada en componentes específicos de BOSCH, el BNO055 cuenta con los medios necesarios para obtener una salida de forma absoluta de los ángulos en todos los ejes.

Incluye compensación de temperatura, así como el potente microcontrolador ARM Cortex-M0 que procesará y realizará los cálculos matemáticos necesarios para llegar a obtener estos ángulos, en el mismo chip. El ratio de actualización llegará hasta los 100 Hz.

Las salidas posibles sin necesidad de ningún cálculo por parte del procesador del ESP32 serán:

- Orientación Absoluta (Vector Euler, 100Hz),
- Orientación Absoluta (cuaternión, 100Hz),
- Vector Velocidad Angular en rad/s (100Hz),
- Vector de Aceleración en m/s^2 (100Hz), (gravedad + movimiento lineal),
- Vector de Fuerza de Campo Magnético en micro Tesla (μT) (20Hz),
- Vector de Aceleración Lineal en m/s^2 (100Hz) (aceleración menos gravedad),
- Vector de Gravedad en m/s^2 (menos movimiento), Temperatura Ambiente en grados celsius (1Hz).

En algunos modelos de placa, posee un regulador de tensión de 3.3V, un convertor de niveles, pines I2C para su control, así como un cristal de cuarzo (en este caso de 32768 KHz) para un funcionamiento mucho más estable y preciso. Posee las siguientes dimensiones: 20mm x 27mm x 4mm.

Para éste proyecto, se ha utilizado un modelo equivalente comercializado por distribuidores provenientes de China, el CJMCU 055, disminuyendo su elevado precio (en torno a los 50€ en Europa) a en torno a 10€. No podemos aclarar el modelo del chip del sensor utilizado por falta de información del distribuidor, pero es totalmente equivalente al oficial de BOSH.

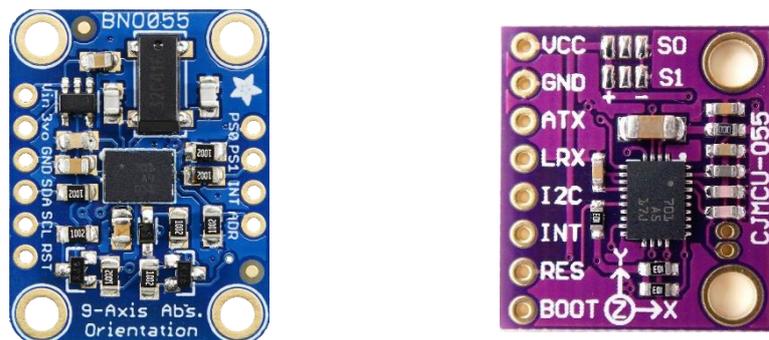


Figura 21: BNO055: Placa de Adafruit Abs. Orientation (izq.) frente a placa de proveedor chino CJMCU-055. Mayor tamaño en el caso de la placa de Adafruit.

4.3.3 Alimentación

El procesador ESP32 cuenta con tecnologías avanzadas de **administración de energía**, pudiendo cambiar entre diferentes modos de funcionamiento según el consumo de energía deseado:

- Modo activo: el chip de radio está encendido. Puede recibir, transmitir o escuchar.
- Modem-sleep mode: la CPU está operativa y el reloj es configurable. La banda base Wi-Fi/Bluetooth y radio están desactivadas.
- Light-sleep mode: la CPU está en pausa. La memoria y los periféricos RTC, así como el coprocesador ULP están en funcionamiento. Cualquier evento de activación o wake-up (MAC, host, temporizador RTC o interrupciones externas) activará el chip.
- Deep-sleep mode: solo la memoria RTC y los periféricos RTC están encendidos. Los datos de conexión de Wi-Fi y Bluetooth se almacenan en la memoria RTC. El coprocesador ULP está en funcionamiento.
- Modo de Hibernación: el oscilador interno de 8 MHz y el coprocesador ULP están deshabilitados. La memoria de recuperación del RTC está apagada. Solo están activos un temporizador RTC en el slow clock y ciertos GPIO RTC. El temporizador RTC o los GPIO RTC pueden activar el chip en este modo.

El consumo de energía varía con los diferentes modos de potencia y estados de trabajo de los módulos funcionales como vemos en la siguiente tabla:

Power mode	Description	Power consumption
Active (RF working)	Wi-Fi Tx packet	Please refer to ESP32 Datasheet .
	Wi-Fi / BT Tx packet	
	Wi-Fi / BT Rx and listening	
Modem-sleep	The CPU is powered on.	Max speed 240 MHz: 30 mA ~ 50 mA
		Normal speed 80 MHz: 20 mA ~ 25 mA
		Slow speed 2 MHz: 2 mA ~ 4 mA
Light-sleep	-	0.8 mA
Deep-sleep	The ULP co-processor is powered on.	150 μ A
	ULP sensor-monitored pattern	100 μ A @1% duty
	RTC timer + RTC memory	10 μ A
Hibernation	RTC timer only	5 μ A
Power off	CHIP_PU is set to low level, the chip is powered off	0.1 μ A

Tabla 3: Consumo de potencia para distintos modos de trabajo.

Con el uso de las anteriormente mencionadas tecnologías de control de consumo, hemos elegido alimentar todo el sistema (cada dispositivo vestible) con una batería lipo de 350mAh. Con dicha alimentación el sistema tiene una autonomía de aproximadamente 8h para los esclavos y 6h para el maestro. Esto es debido a que el dispositivo maestro utiliza Bluetooth a la vez que Wi-Fi, lo que aumenta el consumo. Aunque en este caso no existe tanta diferencia debido al elevado consumo de la tecnología Wi-Fi, frente al bajo consumo de Bluetooth.

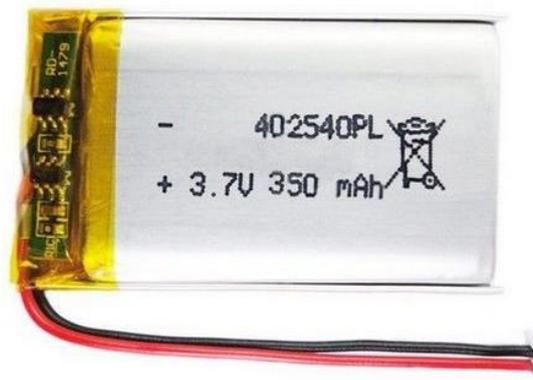


Figura 22: Batería lipo utilizada en el dispositivo.

El dispositivo a su vez, contará con un sistema de carga en el propio vestible, que permitirá conectarse a una fuente de alimentación por micro-USB, siendo válido cualquier cargador de dispositivos móviles o puerto USB. La batería estará conectada al sistema de carga por medio de un conector JST XH-2.54mm.

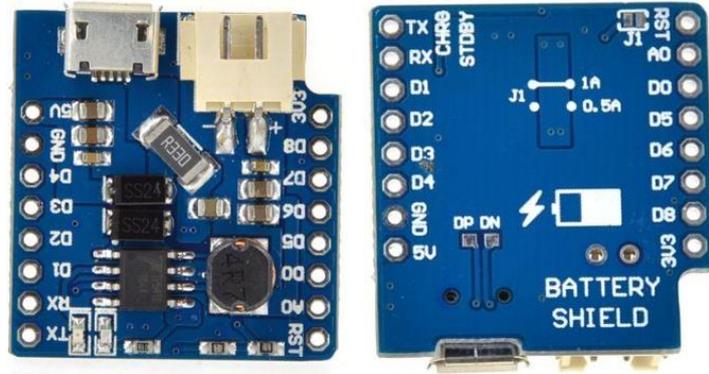


Figura 23: Sistema de carga. Cara superior (izq.) e inferior (der.).

4.3.4 Pruebas protoboard

La primera toma de contacto con todo el sistema completo se realizó en una placa de desarrollo protoboard, conectando el Wemos ESP32 D1 mini por usb directamente al PC. No se utilizó conexión Bluetooth para intercambiar la información de los sensores, si no que se transmitió la información con los cuaterniones por puerto serie COM. Como vemos en las siguientes imágenes se consiguió animar las extremidades de un avatar.

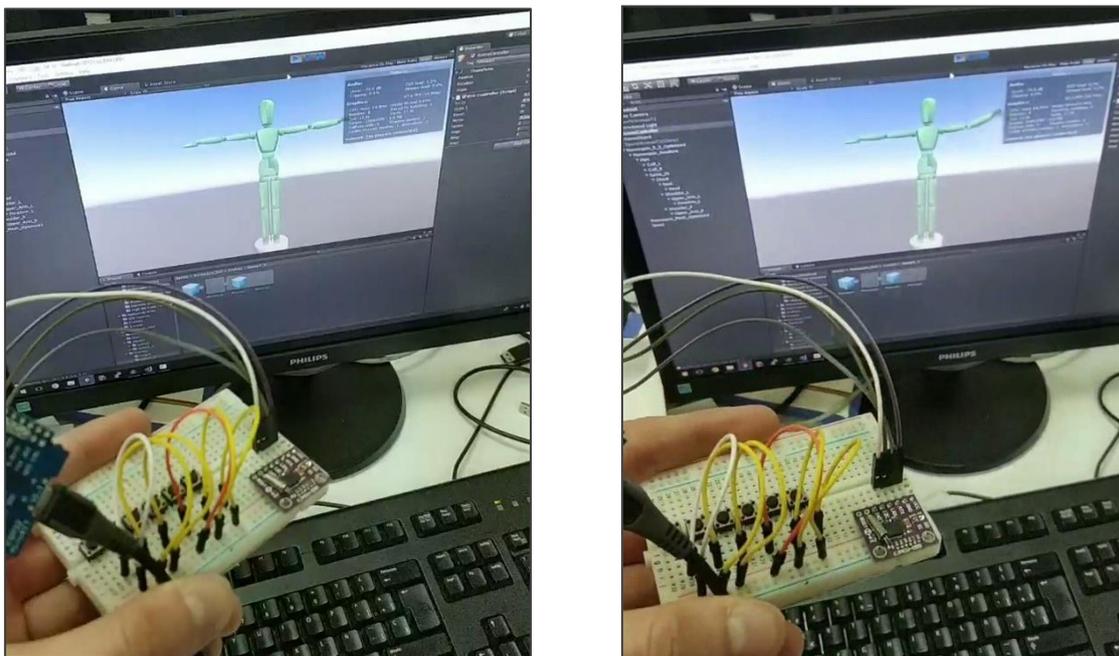


Figura 24: Primera prueba en protoboard, movimiento 1.

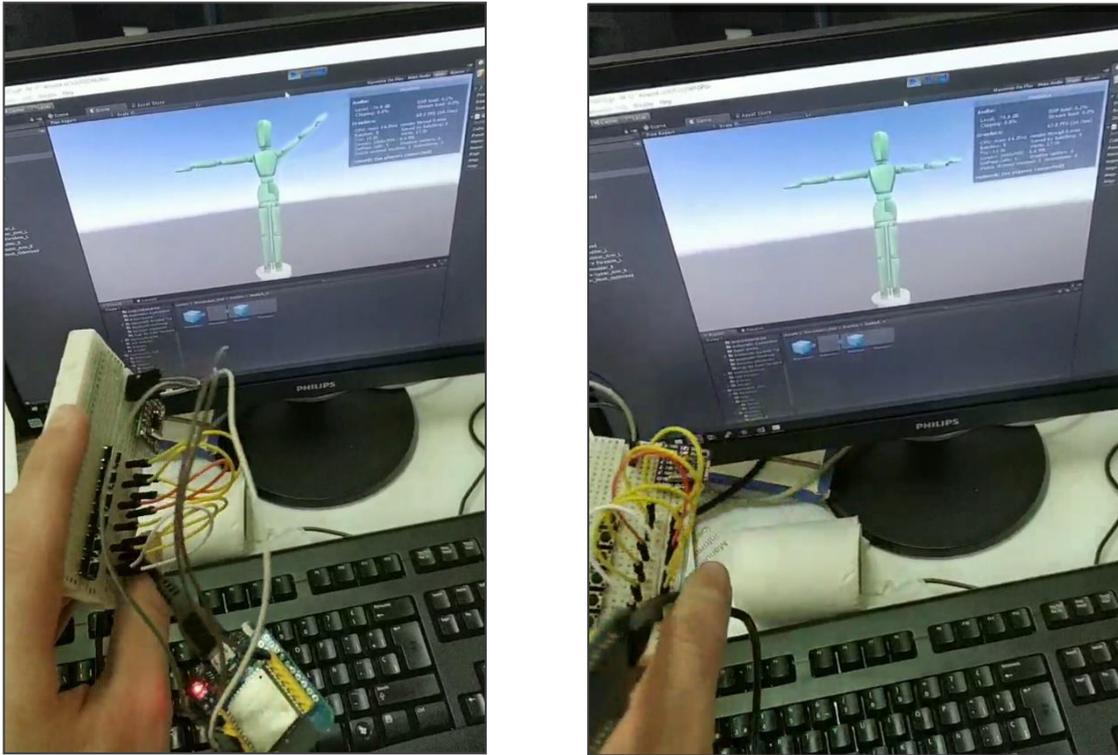


Figura 25: Primera prueba en protoboard, movimiento 2

4.3.5 Conexiones

En primer lugar, soldaremos el sistema de carga o “battery shield” a los pines correspondientes del Wemos ESP32 D1 mini. Esto nos permitirá cargar la batería lipo que conectemos al conector JST – XH que incorpora como hemos comentado anteriormente. Aunque no hemos utilizado los conectores pines hembra que nos proporcionaba el fabricante para otorgar separación entre el WROOM-32 y la placa de carga, hemos dejado un espacio entre el encapsulado metálico del WROOM-32 y ésta, para evitar contactos accidentales, pero reduciendo el tamaño total (con los pines hembra el tamaño sería mucho mayor).



Figura 26: Wemos ESP32 con placa de carga.

Ahora, conectaremos el sensor (placa basada en Adafruit BNO055 Absolute Orientation Sensor) encima de las patillas de la placa de carga, para lo cual deberemos recortar los bordes de la placa del sensor mediante lijado, para que no se toque con el bobinado del inductor 4R7.

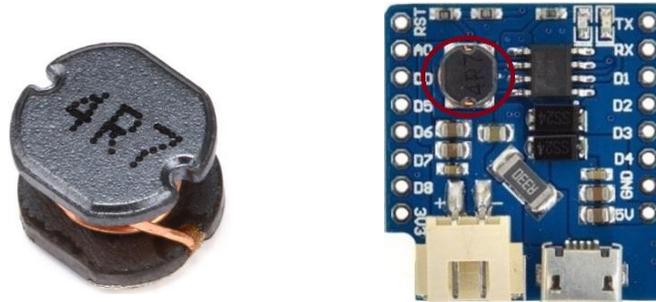


Figura 27: Inductor 4R7 de $4.7 \mu\text{H}$ (izq.) frente a su posición en la placa (der.)

La conexión de las patillas se realizará de la siguiente forma: la patilla SDA o ATX (según modelo de placa del BNO055) de la placa del sensor se conectará a la patilla D2 de la placa de carga, que se corresponde con el GPIO21; la patilla SCL o LRX de la placa del sensor se conectará a la patilla D1 de la placa de carga, que se corresponde con el GPIO22; y por último conectaremos el pin de GND y de 3V3 del sensor a los de la placa de carga correspondientes. Cabe aclarar que en algunos otros modelos de fabricantes chinos de la placa del sensor, deberemos conectar también o el pin de interrupciones "int" o el de selección de I2C "I2C_SEL" a tierra.

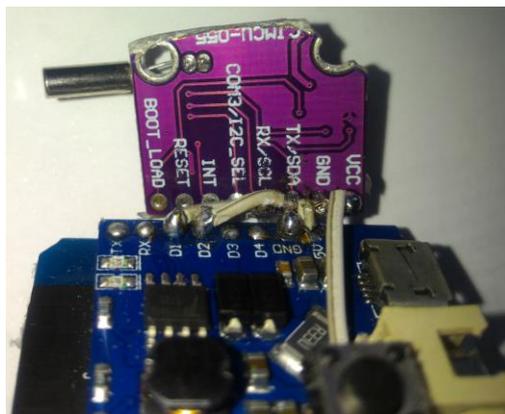


Figura 28: Conexión de la placa de carga a la placa sensor CMJU 055 (D1, D2, GND, 3V3)

Conectaremos el resto de elementos a la placa, el pulsador y el led RGB. Para el pulsador hemos elegido conectar sus patillas entre el pin D7 correspondiente a GPIO23 y 3V3 simplemente por la posición del botón según el encapsulado. El LED RGB lo podremos soldar a cualquiera de las patillas que admitan PWM. En ambos casos, pulsador y LED, podremos conectarlos a prácticamente cualquier patilla de

nuestra placa, simplemente teniendo en cuenta si el LED es de ánodo o cátodo común, y respecto al pulsador si está conectando la patilla elegida a GND o a 3V3 (y lo detectaremos como alta o baja en el pin elegido).

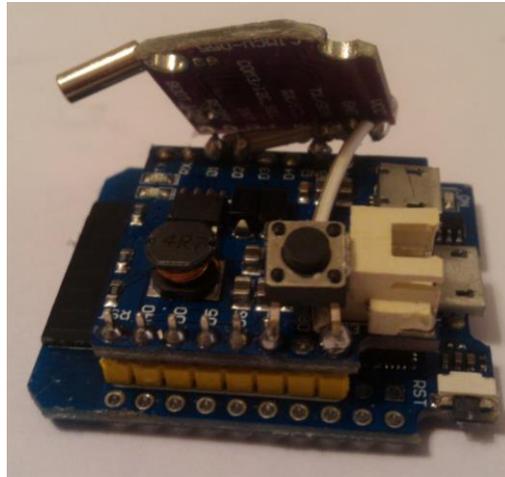


Figura 29: Conexión del pulsador a la placa (Pines D7 y 3V3).

Por último, conectaremos un interruptor entre la batería lipo y el conector JST- XH para poder elegir cuándo el sistema estará alimentado por batería o cuándo funcionará solamente conectado por micro-USB. En la versión final del producto, habrá una única placa que junte la parte del sistema de carga, con el ESP32, con el sensor, y podremos ajustar mejor el funcionamiento por baterías tanto por la reducción del consumo como por un conexionado de la batería al circuito mejorado. Colocaremos una fina capa de material aislante por la cara inferior de la placa del Wemos D1 mini ya que la batería se encontrará colocada debajo de la placa en el encapsulado final.

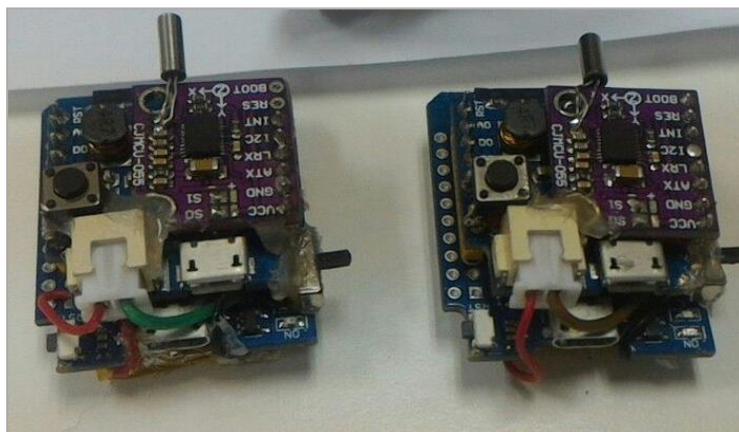


Figura 30: Conexión final de dos de los prototipos.

4.3.6 Encapsulado

En el primer prototipo, se construirá mediante varias capas de plástico rígido pegadas entre sí dando lugar a una forma de cubo. Se realizarán pequeñas perforaciones en el plástico para permitir el acceso al pulsador, interruptor de encendido y micro-USB de carga. El resultado final será pintado para disimular imperfecciones y se pegará una capa de velcro por la cara inferior del producto, que permita su fácil colocación en distintas partes del cuerpo mediante cintas.

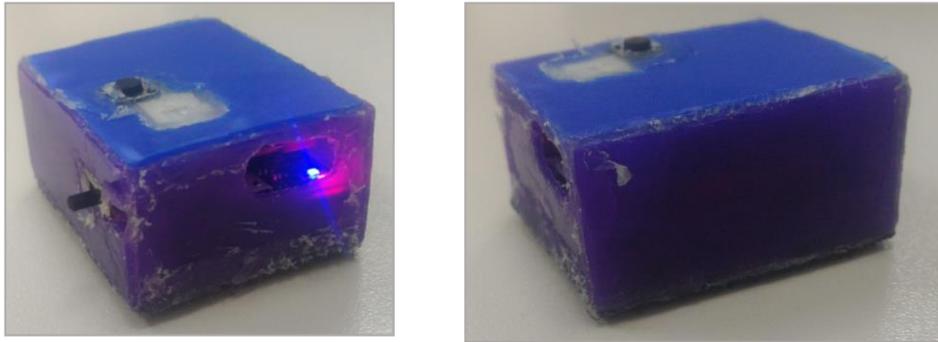


Figura 31: Encapsulado vista lateral izq. (izq.) vs Encapsulado vista lateral der. (der.)



Figura 32: Encapsulado final tras lijado y pintado.

Capítulo 5 Estudio de la comunicación

En todo proyecto que involucre el **co-diseño Hardware-Software**, es muy importante estudiar la comunicación tanto intrasistema como entre el dispositivo y el sistema al que se conectará (en nuestro caso un ordenador). Todo ello debe ser minuciosamente calculado, para permitir el perfecto funcionamiento conjunto HW-SW, antes de determinar los elementos finales a incorporar al dispositivo.

A continuación, desglosaremos las distintas opciones de diseño, analizadas previamente al comienzo de la toma de decisiones y el inicio del desarrollo.

5.1 Transmisión de los datos

5.1.1 Conexión USB - Serie

El *Universal Serial Bus* o **USB**, es un bus estándar industrial que define los cables, conectores y protocolos usados para conectar, comunicar y proveer de alimentación eléctrica a dispositivos electrónicos. Fue desarrollado para tratar de unificar la forma de conectar periféricos entre los dispositivos de las empresas del sector que hasta entonces eran poco compatibles entre sí.

La serialización es el proceso de transformar estructuras de datos y objetos en un formato que pueda ser almacenado y transmitido correctamente, de manera que pueda ser recuperado íntegramente en su forma original. Esto se consigue eliminando las relaciones internas entre los bytes, reduciéndolo a un simple *stream* o flujo de bytes que permita su tratamiento directo.

Por su parte la **UART** o *USART Universal Asynchronous Receiver-Transmitter* es el dispositivo que controla los puertos y dispositivos serie, base de casi cualquiera de las comunicaciones de los microcontroladores.

La comunicación del SoC con el ordenador y su programación se realizará mediante USB. Al conectar un Arduino al ordenador mediante un cable USB, el ordenador instala un puerto serie virtual (COM) para acceder al puerto serie de Arduino. Casi todos los modelos de Arduino y compatibles utilizan una interfaz USB pues integran algún chip que hará la función de convertir el interfaz serie de la UART de Arduino a un interfaz USB.

Muchos Arduino disponen de microcontroladores que no tienen un interfaz USB incorporado, impidiendo la conexión directa a un ordenador. Para solucionarlo utilizan un segundo microcontrolador intermediario, generalmente un ATmega16u2 que cuenta con un interfaz USB que mediante un firmware especial facilita la carga del programa en la memoria flash del Arduino para la comunicación posterior por USB.

En la práctica, esto nos permite tener un puerto serie virtual que utilizaremos para conectar con Arduino y para ello necesitaremos un driver o software que implemente la comunicación con el USB y monte este puerto serie virtual. Los drivers deberán poseer certificación de “usb.org” pagando las licencias de uso correspondientes.

5.1.2 Convertidor de niveles

Su función será convertir la señal USB a **TTL** para poder conectar el resto de la placa del Arduino, que sólo admite señales TTL, y se conectará con el ordenador a través de las líneas de datos D+ y D- del conector mini USB. Su comunicación con el módulo Arduino (el resto del dispositivo) será efectiva a través de la UART de éste (TXD y RXD). Los pines *_DTR* (*Data Terminal Ready Control Output*, Terminal de Salida de Datos para Control) y *_RTS* (*Request to Send Control Output*, Petición para enviar) estarán conectados al *bootloader* y *reset* del Arduino.

El conversor de transmisiones serie a USB **FT232R** está alimentado a 5 voltios y las señales a su salida son de 5 voltios en alto, dado que el puerto USB del ordenador proporciona 5 voltios. Por ello, no se podría conectar directamente al Arduino, ya que sólo admite 3,3 voltios en sus patillas. Así pues, en el circuito del Arduino existirá un divisor de tensión que reduzca los 5 voltios a 3,3 voltios, en las líneas de transmisión de datos, *reset* y *bootloader*.

En ciertos Arduino que utilizan tecnologías USB-Serie como FTDI, los drivers suelen estar incluidos en el sistema operativo y tiene su correspondiente licencia de uso, permitiéndonos el reconocimiento automático del dispositivo al conectarlo.

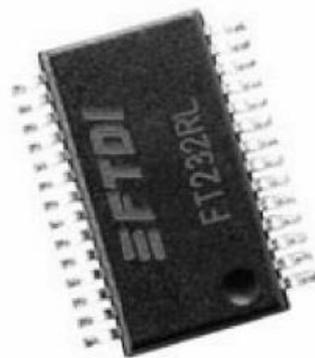


Figura 33: convertidor de niveles FT232RL de FTDI

Pero no todos los chips USB-Serie que presentan los diferentes Arduino utilizan FTDI como el *FT232RL*, y algunas de las otras tecnologías utilizadas presentan problemas de firma de los drivers sobre todo los de procedencia china (pues suelen evitar pagar licencias).

Listado de otras tecnologías sustitutas de FT232RL de FTDI:

- *Atmel ATMEGA8U2 16U2 con Firmware Arduino USBSerial (como en Arduino UNO R3)*
- *ASIX MCS7810*
- *Cypress CY7C65211 & Cypress CY7C65213*
- *Prolific PL2303HX (Pin compatible con FT232R)*
- *Microchip MCP2200*
- *Silicon Labs CP210x*
- *Texas Instruments TUSB3410*

Y el utilizado por nosotros en una de las versiones:

- *WCH-IC Chinese CH340*

La identificación de los dispositivos USB, se verá determinada por dos identificadores: el USB Vendor ID (VID) que responde al fabricante y el Product ID (PID) que identifica el producto de ese fabricante. Gracias a esto, es posible identificar inequívocamente un dispositivo USB.

Problemas derivados de la utilización de tecnologías sin firmar

En nuestro prototipo, utilizamos la tecnología WCH-IC Chinese **CH340**, que nos proporciona interfaz USB-Serie, pero no tiene controladores firmados oficiales. Esto nos planteó ciertos problemas en sistemas operativos OSX, pues al contrario que en Windows que permite de forma sencilla instalar los drivers, necesitamos realizar ciertos ajustes para poder conectar nuestro dispositivo y utilizar la comunicación serie. Lo primero de todo es habilitar las *kexts (kernel extensions)* no firmadas o en su defecto desactivar el SIP (*System Integrity Protection*). Una *kext* es un conjunto de código cargado dinámicamente que corre en el espacio del *kernel*, y que permite normalmente realizar tareas a bajo nivel no permitidas en el espacio usuario, en nuestro caso estaríamos hablando de cargar drivers de dispositivo a bajo nivel. Después ya podríamos instalar los drivers del CH340 (la última versión, teniendo cuidado de versiones que pueden provocar un *crash* del sistema) y reestablecer el SIP o deshabilitar las *kext* no firmadas de nuevo.

Si necesitáramos que nuestro prototipo funcionara en OSX, se buscará utilizar el modelo Arduino Nano con chip FTDI pues éste no necesita de cambios en la protección del sistema para instalar sus drivers.

5.1.3 Funcionamiento sobre Wi-Fi

Hardware del módulo Wi-Fi / BT

Como ya se ha comentado anteriormente, el módulo que permita la comunicación Wi-Fi y Bluetooth debe tener un microcontrolador interno que se pueda programar. Debido a esto se escogerá una placa de desarrollo de la familia **ESP32**, como es el

modelo Wemos D1 mini con el ESP32-S (figuras 34 y 35). Se ha elegido este modelo por la familiaridad con la familia de controladores de *espressif*, por la continuación del trabajo realizado en el TFG previo con las placas ESP8266 (sólo Wi-Fi).

Además, se presenta en una amplia gama de módulos de Wi-Fi y BT totalmente certificados que funcionan con los SoC propios de *espressif*. En nuestro caso el Wemos D1 mini cuenta con un módulo WROOM-32, que contiene el chip ESP32 junto con algunos componentes clave que incluyen un oscilador de cristal y un circuito de adaptación de antena todo en el mismo módulo. Esto hace que sea más fácil proporcionar una solución basada en ESP32 lista para integrarse en el producto final. A éste módulo, la placa Wemos D1 mini, añade algunos componentes adicionales, como una interfaz de programación USB, resistencias de arranque y encabezados de salida. Los pines clave de esta placa se resumen en la siguiente imagen:

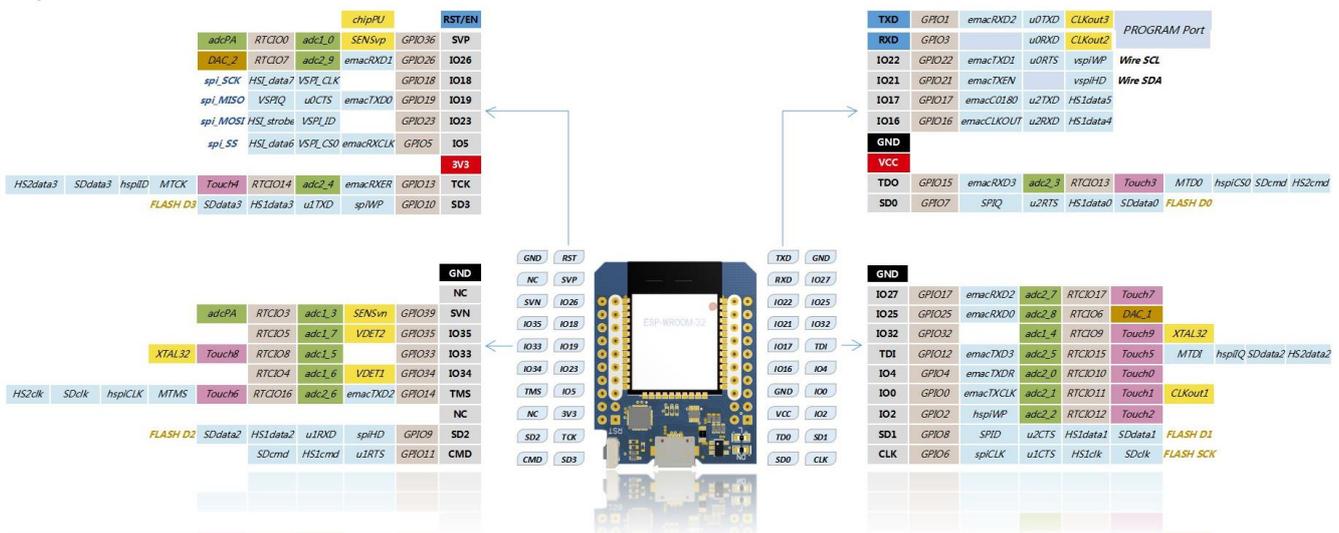


Figura 34: Pines de la placa Wemos MINI D1 ESP32



Figura 35: Imagen real de la placa Wemos MINI D1 ESP32

A continuación, vamos a comentar las características principales del módulo WROOM ESP-32.

Características Hardware:

- Microprocesador de bajo consumo ESP32-D0WD (o D0WD Q6), de doble núcleo corriendo a velocidades de hasta 160MHz
- 4MB de FLASH
- Antena integrada en pista MIFA (Meandered Inverted-F Antenna)
- Tensión de alimentación/operación 2.7 ~ 3.6V
- Reloj incorporado: oscilador/cristal de 40Mhz (permitiendo funcionamiento a 40 MHz, 80 MHz y 160 MHz)
- Sensor incorporado de efecto Hall
- Corriente de operación media de 80mA
- Rango de temperatura de operación recomendada: -40°C a 85°C
- Interfaces del módulo disponibles: tarjeta SD, UART, SPI, SDIO, I2C, LED PWM, Motor PWM, I2S, IR, GPIO, sensor táctil capacitivo, ADC, DAC.
- Tamaño (módulo WROOM): (18±0.2) mm x (25.5±0.2) mm x (3.1±0.15) mm
- *Chip puente UART-USB incorporable (incluido en la placa Wemos D1 mini)*

Características Software:

- Modo Wi-Fi: Station/SoftAP/SoftAP+Station/P2P
- Seguridad Wi-Fi: WPA/WPA2/WPA2-Enterprise/WPS
- Encriptación: AES/RSA/ECC/SHA
- Actualización de firmware: UART Download / OTA (descarga y escribe el firmware a través de la red o el host)
- Desarrollo de software: Admite Cloud Server Development / SDK para desarrollo de firmware personalizado.
- Protocolos de Red: IPv4, IPv6, SSL, TCP/UDP/HTTP/FTP/MQTT.
- Configuración de Usuario: set de instrucciones AT, servidor en la nube, Android/iOS app

Certificación y tecnologías soportadas:

- Certificación RF: FCC / CE / IC / TELEC / KCC / SRRC / NCC
- Certificación Wi-Fi: Wi-Fi Alliance
- Certificación Bluetooth: BQB
- Certificación verde: RoHS / REACH
- Protocolos Wi-Fi: 802.11 b / g / n (802.11n hasta 150 Mbps) con agregación A-MPDU y A-MSDU y soporte de *guard interval* de 0.4 μ s
- Rango de frecuencia Wi-Fi: 2.4 GHz ~ 2.5 GHz
- Protocolos Bluetooth: v4.2, BR / EDR y especificación BLE
- Radio Bluetooth: receptor NZIF con sensibilidad de -97 dBm, transmisor de clase 1, clase 2 y clase 3 y AFH
- Audio Bluetooth: CVSD y SBC

Comunicación Wi-Fi

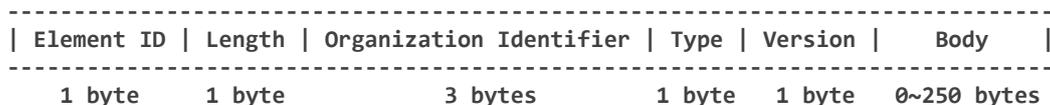
La comunicación Wi-Fi se producirá entre los distintos dispositivos. Los esclavos se conectarán al maestro mediante dicha tecnología, aunque los datos serán transmitidos al dispositivo final (PC, dispositivo móvil, etc.) mediante Bluetooth.

Concretamente utilizaremos la tecnología **ESP-NOW**, un tipo de protocolo de comunicación **WiFi sin conexión** definido por *espressif*. En ESP-NOW, los datos de la aplicación se encapsulan en una trama activa específica del proveedor y luego se transmiten de un dispositivo WiFi a otro sin conexión. El CTR con protocolo CBC-MAC (CCMP) se utiliza para proteger la trama activa aportando la capa de seguridad. ESP-NOW es ampliamente utilizado en sistemas de luz inteligente, control remoto, sensórica, etc.

Respecto al formato de la trama, ESP-NOW utiliza una trama activa específica del proveedor para transmitir datos con el siguiente formato:



- **Category Code:** el campo Categoría se establece con un valor de (127) que indica la categoría específica del proveedor.
- **Organization Identifier:** el identificador de organización contiene un identificador único (0x18fe34) que son los primeros tres bytes de la dirección MAC especificada por *espressif*.
- **Vendor Specific Content:** el contenido específico del proveedor contiene el campo específico del proveedor de la siguiente manera:



- **Element ID:** el campo Element ID se establece con un valor de (221) que indica el elemento específico del proveedor.
- **Length:** la longitud es la longitud total del Organization Identifier, junto con el campo tipo, versión y cuerpo.
- **Organization Identifier:** el identificador de organización contiene un identificador único (0x18fe34) que son los primeros tres bytes de la dirección MAC especificada por *espressif*. (Como en el caso anterior)
- **Type:** El campo Tipo se establece en el valor (4) que indica ESP-NOW.

- **Versión:** el campo Versión está configurado a la versión de ESP-NOW.
- **Body:** El cuerpo contiene los datos ESP-NOW.

Como ESP-NOW es no orientado a conexión, el encabezado MAC es un poco diferente al de las tramas estándar. Los bits FromDS y ToDS del campo FrameControl son ambos 0. El primer campo de dirección se asigna a la dirección de destino. El segundo campo de dirección se asigna a la dirección de origen. El tercer campo de dirección está configurado para transmitir a la dirección de broadcast (0xff:0xff:0xff:0xff:0xff:0xff).

Para la **seguridad**, ESP-NOW usa el método CCMP, que se puede consultar en IEEE Std. 802.11-2012, para proteger la trama activa. El dispositivo WiFi mantiene una clave maestra primaria (PMK) y varias claves maestras locales (LMK). La longitud de ellas es de 16 bytes. PMK se usa para cifrar LMK con el algoritmo AES-128 configurable con la llamada a `esp_now_set_pmk()`. Si la PMK no está configurada, se usará una PMK predeterminada. Si se ha establecido una LMK para el dispositivo emparejado, se usará para encriptar la trama activa específica del proveedor con el método CCMP. La cantidad máxima de diferentes LMK es seis. No está soportado la encriptación de una trama multicast (o de multidifusión).

Los datos con ESP-NOW deben transmitirse después de haber iniciado el WiFi, por lo que se recomienda iniciar WiFi antes de la necesaria inicialización de ESP-NOW y detener WiFi después de la **inicialización** de ESP-NOW. Estas se realizarán llamando a `esp_now_init()` para inicializar ESP-NOW y `esp_now_deinit()` para desinicializarlo. Cuando se llama a `esp_now_deinit()`, se eliminará toda la información de los dispositivos emparejados.

Antes de enviar datos a otro dispositivo, deberemos llamar a `esp_now_add_peer()` para agregarlo primero a la lista de **dispositivos vinculados**. La cantidad máxima de dispositivos emparejados es de veinte. Si la seguridad está habilitada, se debe configurar el LMK. Los datos con ESP-NOW pueden enviarse desde la interfaz de estación o softap, la cual deberemos asegurarnos antes de enviar datos de que esté activada. Si realizamos una difusión, deberemos agregar previamente un dispositivo con una dirección MAC de difusión. El rango del canal del dispositivo emparejado va de 0 a 14. Si el canal está configurado en 0, los datos se enviarán en el canal actual. De lo contrario, el canal debe configurarse con el número en el que se encuentra el dispositivo local.

Los datos se enviarán llamando a `esp_now_send()` y `esp_now_register_send_cb` devolverá `ESP_NOW_SEND_SUCCESS` si los datos se reciben con éxito en la capa MAC. De lo contrario, devolverá `ESP_NOW_SEND_FAIL`. Hay varios motivos por los que el envío podría fallar, por ejemplo, si el dispositivo de destino no existe, si los canales de los dispositivos no son iguales, o si la trama activa se pierde al transmitir por el aire, etc. No está garantizado que la capa de aplicación pueda recibir los datos.

Si es necesario, se puede exigir confirmación (ack) cuando se reciban los datos. Si se supera el tiempo de espera de recepción de datos, se pueden retransmitir los datos. Además, se puede asignar un número de secuencia a los datos transmitidos con ESP-NOW para eliminar los datos duplicados.

Si hay muchos datos para enviar con ESP-NOW, se llamará a `esp_now_send ()` para enviar hasta 250 bytes o menos de datos cada vez. Un intervalo demasiado corto entre el envío de dos datos ESP-NOW puede provocar un desorden en el **envío** de la función de *callback* (por ser sin conexión). Por lo tanto, se recomienda enviar los siguientes datos de ESP-NOW después de que haya regresado la función *callback* del envío anterior. Con `esp_now_register_rcv_cb` registraremos la función *callback* recibida.

La función *callback* se ejecuta desde una tarea WiFi de alta prioridad. Por lo tanto, no realizaremos operaciones largas en la función *callback*, lo sustituiremos por una cola de datos desde una tarea de menor prioridad.

Por último, aclarar que ESP-NOW, el protocolo creado por *espressif*, no necesita una red WiFi, siendo similar a otros protocolos de bajo consumo utilizados en la banda de los 2.4Ghz (como algunos ratones Wireless), pero sí necesita el emparejado inicial anteriormente comentado, y después de éste, la conexión es persistente de igual a igual (P2P).

Comunicación Bluetooth

El maestro será el único de los dispositivos que aprovechará el **Bluetooth** que incorpora la placa. Enviará los datos obtenidos en el resto de dispositivos incluyendo los suyos propios al dispositivo final para su visualización en tiempo real, análisis o representación del movimiento 3D en vivo mediante la aplicación destinada a tal fin.

Bluetooth, es un estándar de tecnología inalámbrica para el intercambio de datos en distancias cortas, con ventajas que incluyen la robustez, un bajo consumo de energía y bajo coste.

Para el diseño, decidimos optar por el **Bluetooth tradicional**, frente al **Bluetooth Low Energy** utilizado por muchas aplicaciones actuales, aunque no descartamos que en futuras versiones pasemos a utilizar BLE. El principal problema de BLE es que no permite enviar paquetes de datos de gran tamaño, un inconveniente importante cuando hablamos de datos de precisión, con respuesta en tiempo real. El tamaño máximo de paquete para transferir desde la capa de aplicación a las capas inferiores es de 20 Bytes. Para nuestra primera versión solventaremos este problema utilizando el Bluetooth clásico, aunque no descartamos trocear los paquetes de datos en unidades más pequeñas y a una mayor tasa para equilibrar el envío y hacerlo equivalente al utilizado en nuestro prototipo con **Bluetooth clásico**.

En el ESP32, básicamente, la pila del protocolo Bluetooth se divide en dos partes: una "Pila del controlador" y una "Pila de host". La **pila del controlador** contiene el PHY, Baseband, Link Controller, Link Manager, Device Manager, HCI y otros módulos, y se utiliza para la administración de la interfaz de hardware y la administración de enlaces. La **pila de host** contiene L2CAP, SMP, SDP, ATT, GATT, GAP y varios perfiles, y funciona como una interfaz para la capa de aplicación, lo que facilita a la capa de aplicación el acceso al sistema Bluetooth. El Bluetooth Host puede implementarse en el mismo dispositivo que el Controlador o en diferentes dispositivos. Ambos enfoques son compatibles en el ESP32.

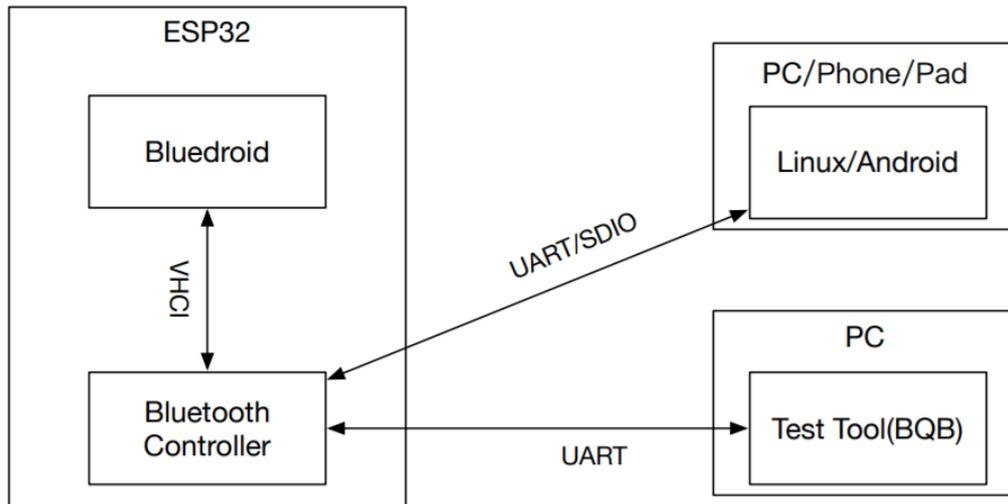


Figura 36: posibles arquitecturas del controlador Bluetooth y el host.

Hay, por tanto, distintas implementaciones posibles:

- **Escenario 1 (configuración ESP32 predeterminada):** BLUEDROID se selecciona como el host Bluetooth, y el VHCI (interfaz HCI virtual implementada por software) se utiliza para la comunicación entre el host Bluetooth y el controlador. En este escenario, tanto el BLUEDROID como el controlador se implementan en el mismo dispositivo (es decir, el chip ESP32), lo que elimina la necesidad de una PC extra u otros dispositivos host que ejecuten el host Bluetooth.
- **Escenario 2:** el sistema ESP32 se usa solo como un Controlador Bluetooth, y se requiere un dispositivo adicional que ejecute el Host Bluetooth (como un PC Linux que ejecute BlueZ o un dispositivo Android que ejecute BLUEDROID, etc.). En este escenario, controlador y host se implementan en diferentes dispositivos, lo cual es bastante similar al caso de los teléfonos móviles, tabletas o PCs.
- **Escenario 3:** este escenario es similar al Escenario 2. La diferencia radica en que, en la prueba de controlador BQB (u otras certificaciones), el ESP32 puede realizar tests conectándose a las herramientas de prueba, con el UART habilitado como la interfaz de entrada/salida.

Por otra parte, el **entorno operativo** predeterminado de ESP32 Bluetooth es un FreeRTOS (un kernel de sistema operativo en tiempo real para dispositivos integrados) de doble núcleo. El Bluetooth puede asignar tareas basadas en funciones con diferentes prioridades. Las tareas con mayor prioridad son las que ejecuta el controlador. Las tareas del controlador, que tienen mayores requisitos de tiempo real, tienen la mayor prioridad en el sistema FreeRTOS, excepto en las tareas de IPC, que son principalmente para la comunicación entre procesos de las distintas CPUs del doble núcleo. BLUEDROID (el host del sistema Bluetooth del ESP32 predeterminado) contiene cuatro tareas en total, que ejecutan el BTC, BTU, HCI UPWARD y HCI DOWNWARD.

El **controlador Bluetooth** de ESP32 admite tanto el BT clásico como el BLE (V4.2). El controlador integra una variedad de funciones, incluyendo el protocolo H4, HCI, Link Manager, Link Controller, Device Manager y HW Interface. Estas funciones se proporcionan a los desarrolladores en forma de bibliotecas, mientras que también se proporcionan algunas API que pueden acceder al controlador.

La **pila de host Bluetooth** está basada en **BLUEDROID** y se ha adaptado a aplicaciones integradas. En la capa inferior, la pila de host Bluetooth se comunica con el controlador Bluetooth a través de la interfaz HCI virtual. En la capa superior, la pila del Host Bluetooth proporciona los perfiles y las API para la administración de la pila a las aplicaciones del usuario.

Los distintos protocolos definen los formatos de mensaje y los procedimientos destinados a cumplir funciones específicas, como transporte de datos, control de enlace, servicio de seguridad o intercambio de información de servicio. Los perfiles de Bluetooth, por otro lado, definen las funciones y características requeridas de cada capa en el sistema Bluetooth, desde PHY hasta L2CAP, y cualquier otro protocolo fuera de la especificación del núcleo.

A continuación, se muestran los perfiles y protocolos clásicos de BT actualmente admitidos en la pila de host.

- Perfiles: GAP, A2DP (SNK), AVRCP (CT)
- Protocolos: L2CAP, SDP, AVDTP, AVCTP

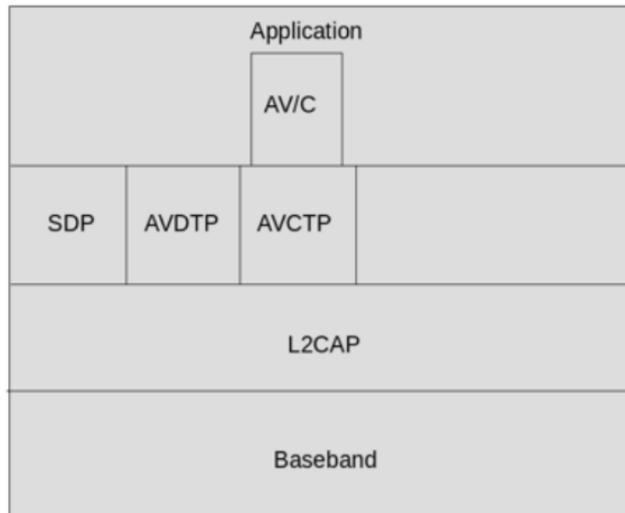


Figura 37: modelo de protocolo, dependencias de los perfiles

5.1.4 Protocolos de comunicación internos Sensores-Microprocesador

Protocolo I2C

El principal sensor utilizado en los dispositivos hardware, el BNO055, utiliza el protocolo *Inter-Integrated Circuit* o **I2C**. Este protocolo pertenece a la familia de protocolos serie síncrona, como *SPI* o *MicroWire*. Aunque también disponga de la posibilidad de utilizar otros protocolos como el anteriormente citado *SPI*, nos decantaremos por *I2C* al ser el recomendado para utilizar intra-placa. Además, preferiremos *I2C* porque utiliza menos líneas, no necesita una línea específica de control, utiliza bus de colector abierto otorgando mayor flexibilidad de voltaje y es multimaestro (H. Mangas, 2014). La velocidad de los sensores por su parte, no presenta cuellos de botella con la tecnología *I2C*, aunque sea de menor velocidad.

Respecto al origen de *I2C*, fue desarrollado por Philips en los años 80. Es un protocolo de bus multipunto bidireccional que emplea dos líneas referidas a masa: *SDA*, Señal de datos y *SCL*, Señal de reloj. Además de síncrono, es *half-duplex*, no pudiendo transmitir y recibir a la vez, además, establece una comunicación maestro-esclavo en la que pueden existir varios maestros. Los dispositivos se conectan al bus con driver de tipo colector/drenador abierto (conexión tipo AND: el 0 vence al 1). En este, para que aparezca un 1 lógico se necesitan resistencias de *pull-up* externas (*R1* y *R2* en la imagen).

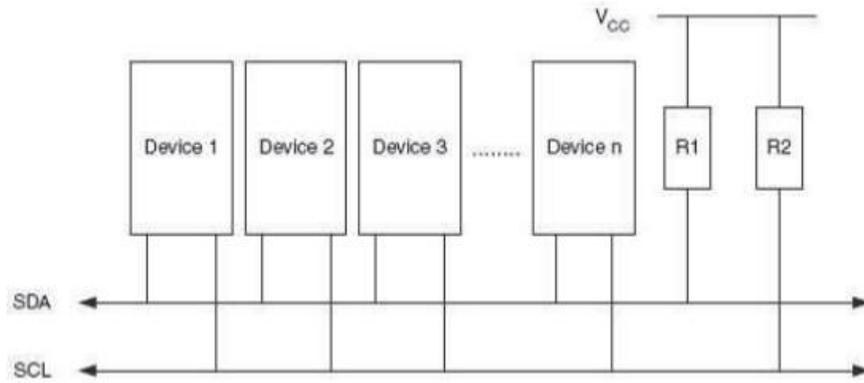


Figura 38: Conexión de varios dispositivos al bus I2C.

El protocolo cuenta con tres modos de funcionamiento:

- *Modo estándar*: alcanza velocidades de 100 kbps y la máxima capacidad del bus es de 400 pF.
- *Modo rápido*: alcanza velocidades de 400 kbps y la máxima capacidad del bus es de 400 pF.
- *Modo alta velocidad*: en este modo se pueden distinguir dos casos, en uno alcanza velocidades de 1700 kbps y la capacidad del bus es de 400 pF, mientras que en el otro alcanza velocidades de 3400 kbps y la máxima capacidad del bus es 100 pF.

En este protocolo, únicamente los maestros pueden iniciar la comunicación. La condición inicial, es cuando las señales SDA y SCL están en alto. En este estado cualquier maestro puede establecer la condición de *start*.

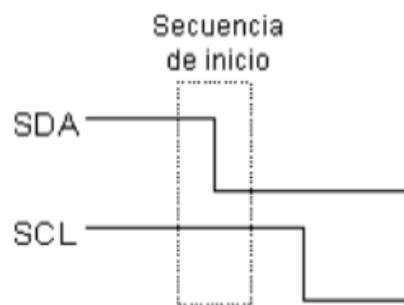


Figura 39: Condición de inicio de I2C.

Una vez cumplida la condición de *start*, se envía un byte con la dirección de 7 bits del dispositivo con el que se desea comunicar y un bit que indica escritura o lectura. Si el esclavo está disponible contestará con un bit de ACK (0 lógico). Si el bit de lectura/escritura fue puesto a 0 (escritura), el dispositivo maestro envía datos al esclavo. Por cada byte que envía recibe un ACK del esclavo.

Si el bit de lectura/escritura fue puesto a 1 (lectura), el esclavo envía los datos y el maestro envía un ACK por cada byte que recibe.



Figura 40: Comienzo de trama en I2C.

Una vez enviados los datos tanto en un sentido como en el otro, el maestro genera una señal de stop, para ello, cambia la línea de datos de bajo a alto mientras el reloj está en alto.



Figura 41: Condición de parada de I2C.

Indicaremos a continuación el conexionado de los distintos pines que conforman la comunicación I2C del Wemos D1 mini con una placa sensora genérica:

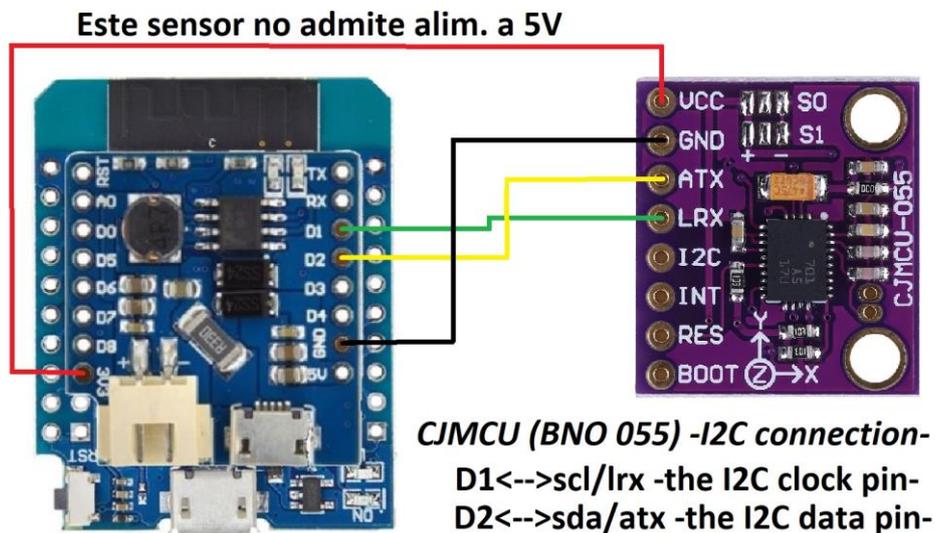


Figura 42: Esquema comunicación genérica I2C – Wemos ESP32 D1mini.

Pines de entrada/salida Digital

Utilizaremos los pines de propósito general GPIO como pines de entrada/salida digital para distintas aplicaciones como un pulsador y un Led RGB. El pulsador estará conectado entre un pin digital y tierra/3V3 según la aplicación. Así, después de eliminar rebotes, obtendremos un 1 o 0 lógico dependiendo del conexionado, cuando activemos el pulsador. En el caso del Led, los pines utilizados deberán contar con modulación PWM y darán información del color al Led RGB.

Pines I2C

En nuestra aplicación sólo necesitaremos un par de pines I2C (SDA, SCL) para conectar nuestro sensor a la placa del ESP32, y por tanto utilizaremos los pines GPIO 21 y 22 que están establecidos para realizar dicha función. Se conectará SDA al pin 21 y SCL al 22, aunque existen librerías que permiten la conexión de cualquier pin GPIO a las líneas de I2C.

5.2 Interfaz del sistema

El sistema conjunto abarca más del contenido de este TFM, pues incluye el software TELEKIN desarrollado en el Grupo de Telemática e Imagen. Para la integración con la suit TELEKIN, se eligió comunicar el dispositivo maestro por Bluetooth a un PC, para sustituir en un futuro la conexión de una cámara Kinect®. Una vez emparejado, el sistema reconocerá el dispositivo e indicará su estado (conectado/no conectado) permitiendo incorporar los sensores inalámbricos para la ejecución de los ejercicios del programa.

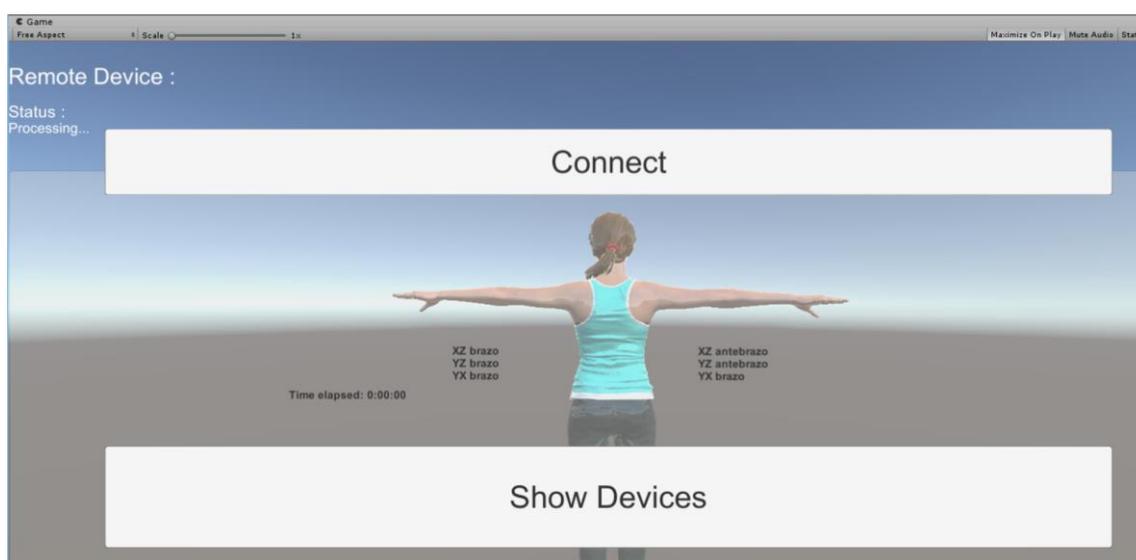


Figura 43: Interfaz TELEKIN con sensores inalámbricos.

Además de la integración con TELEKIN, se ha realizado un programa software en Unity3D para la visualización en tiempo real de los movimientos, y la captura de los mismos para su posterior estudio. Los datos se pueden capturar tanto en crudo, como de forma visual obteniendo un vídeo de los movimientos del avatar al final de la ejecución del software de captura.



Figura 44: Interfaz programa de captura en Unity3D.

Por último, los datos en crudo podrán ser introducidos en el software de análisis, desarrollado en Python mediante notebooks de Jupyter, que cuenta con diversas librerías de representación gráfica y análisis matemático, incluyendo una propia consistente en una modificación de la librería Cuaternión.js de Noble (código abierto).

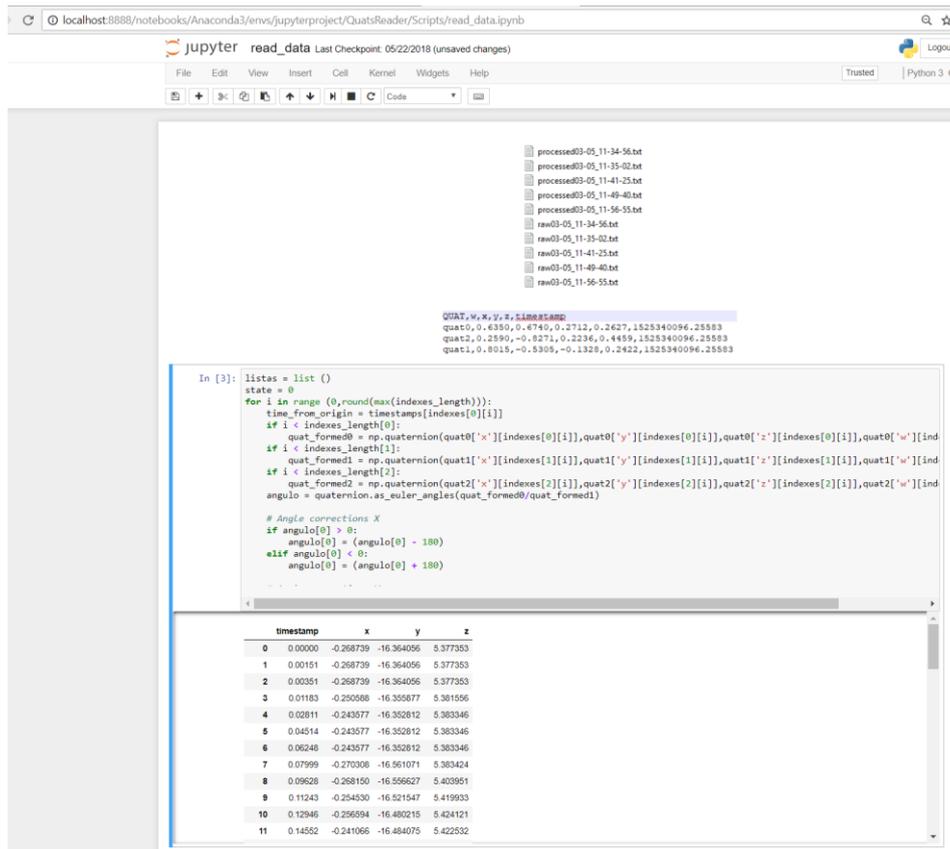


Figura 45: Interfaz de programación Python utilizando Jupyter notebooks

Aclarar que las gráficas que veremos en próximos apartados, resultado de este programa Python, utilizan para la representación de datos la librería Cufflinks de Plotly.

Capítulo 6 Desarrollo y planificación

En este capítulo, primero describiremos unos conceptos teóricos básicos para el entendimiento de los distintos procedimientos y elementos de evaluación seleccionados. Estos serán nociones básicas de *goniometría*, *cuaterniones*, y *dinámica del hombro*, ya que en nuestro trabajo nos centraremos en el estudio de dicha articulación para comprobar el sistema.

Después se describe la batería de pruebas **desarrolladas** para la integración en los sistemas TELEKIN y captura de movimientos para valoración ergonómica.

6.1 Conceptos teóricos

En este apartado, introduciremos distintos conceptos teóricos que vamos a utilizar en nuestro análisis posterior, sobre todo en el capítulo **7 Resultados**, a la hora de evaluar las capacidades del sistema y su precisión y fiabilidad.

6.1.1 Goniometría

Goniometría es la técnica de medición de los ángulos creados por la intersección de los ejes longitudinales de los huesos a nivel de las articulaciones. Las siguientes explicaciones están basadas en el trabajo de Taboadela (2007).

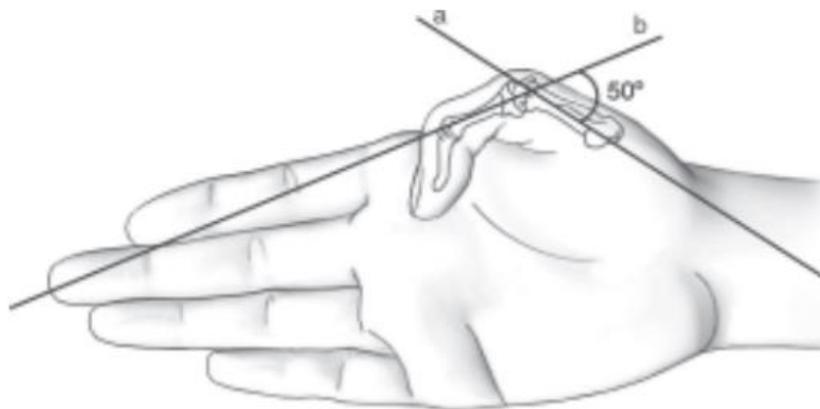


Figura 46: intersección de la prolongación de los ejes longitudinales del primer metacarpiano (a) y de la primera falange (b). Goniometría.

La goniometría en Medicina tiene dos objetivos principales: evaluar la posición de una articulación en el espacio y evaluar el arco de movimiento de una articulación

en cada uno de los tres planos del espacio (para objetivar y cuantificar la movilidad de una articulación).

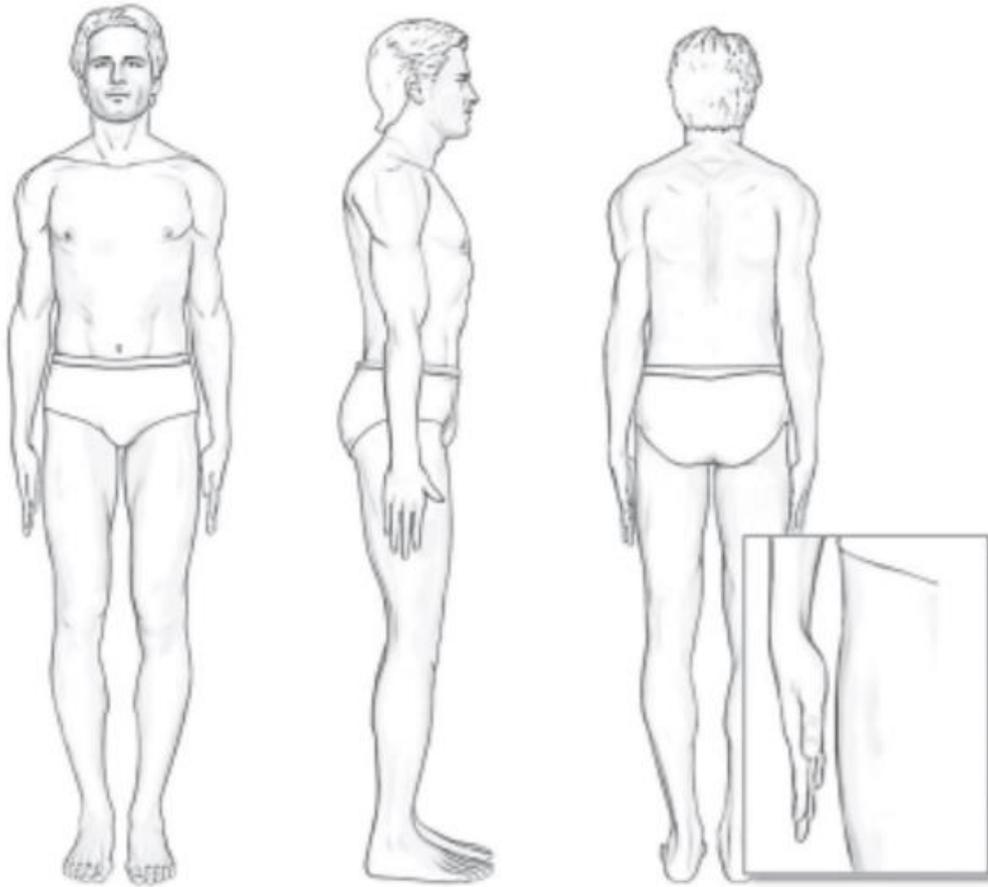
Cuenta con muchos ámbitos de aplicación, tanto ortopedia, traumatología, biomedicina o rehabilitación:

- En Ortopedia y Traumatología y en Reumatología, la goniometría se aplica para describir la presencia de deseos a nivel del sistema osteoarticular con fines diagnósticos, pronósticos, terapéuticos y de investigación.
- En la industria biomédica, la goniometría se aplica en la fabricación y el diseño de aparatos de medición, de instrumental quirúrgico, de prótesis y de ortesis.
- **En Rehabilitación, se utiliza para determinar el punto de inicio de un tratamiento, evaluar su progresión en el tiempo, motivar al paciente, establecer un pronóstico, modificar el tratamiento o darle un punto final, y, finalmente, evaluar la secuela.**
- En Medicina del Deporte, se utiliza para cuantificar la evolución del entrenamiento de los deportistas.
- En Administración de Salud y en Epidemiología, el registro goniométrico estandarizado facilita la revisión e interpretación de datos en las historias clínicas.
- **Por último, en Medicina Legal, en Medicina Previsional y en Medicina del Trabajo, la goniometría se utiliza para la evaluación de incapacidades, producto de secuelas de accidentes o enfermedades que afectan al sistema osteoarticular.**

En nuestro caso el sistema estará pensado para aplicación directa en Medicina del Trabajo y Rehabilitación, aunque el sistema permite su utilización en distintos ámbitos.

En primer lugar, deberemos escoger la posición inicial que nos servirá como posición humana de referencia para valer nuestras medidas, a escoger entre la **posición neutra** o posición cero, la **posición anatómica** o la **posición funcional** (fisiológica).

Elegiremos por tanto la posición neutra, pues es una posición humana de referencia que se adopta como punto de inicio para realizar las mediciones goniométricas comúnmente.



*Figura 47: Posición neutra o posición cero.
En dicha posición el pulgar apunta hacia delante.*

Después de comentar las posiciones de partida, comentaremos los distintos planos en que nos fijaremos para evaluar las medidas.

La **Planimetría** es la descripción de los planos imaginarios que sirven de referencia para dividir el cuerpo humano en diferentes zonas y facilitar su estudio. En el cuerpo humano, se reconocen tres planos perpendiculares entre sí: plano **sagital**, plano **frontal** o coronal y plano **transversal** o axial.

Cada uno de estos planos son cruzados perpendicularmente por un eje: eje **mediolateral**, eje **anteroposterior** y eje **vertical** respectivamente. Sobre estos tres ejes, se producen los movimientos articulares:

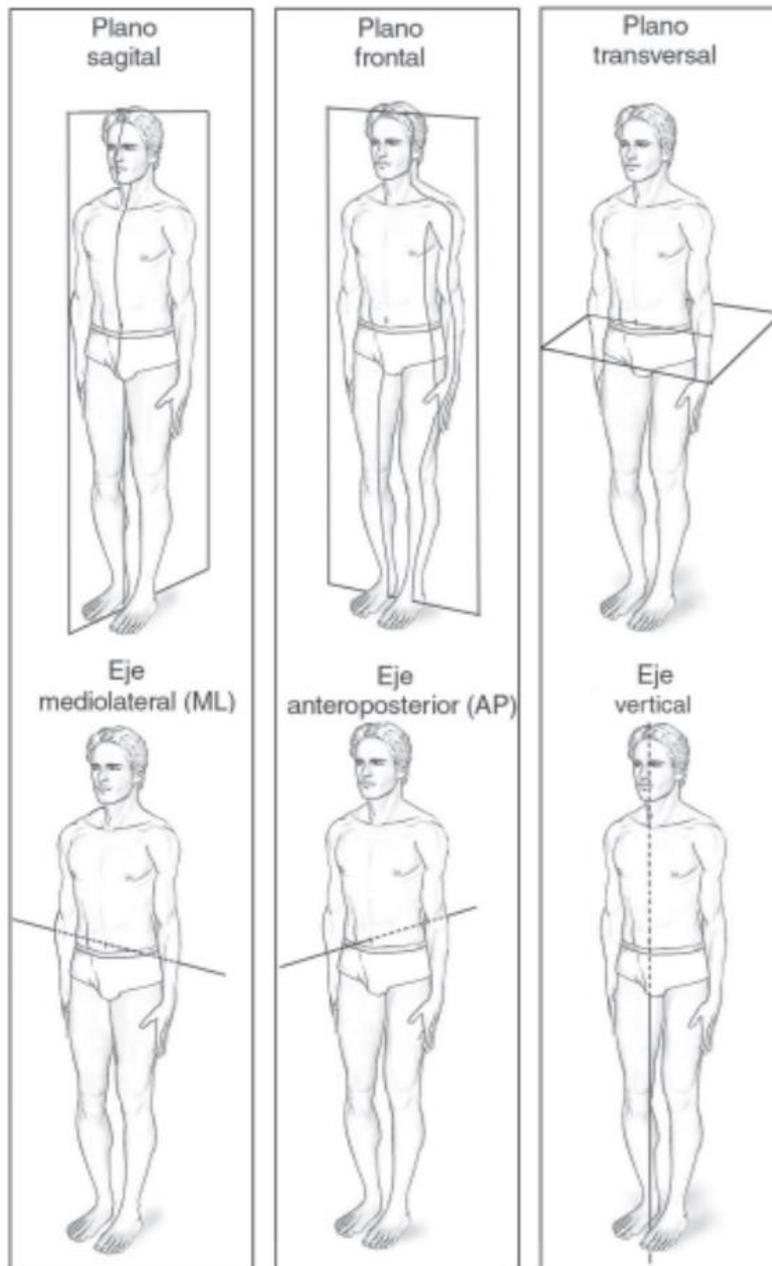


Figura 48: Planos y ejes del cuerpo.

Los distintos movimientos que una articulación puede realizar con respecto al **plano sagital**, están definidos como:

- **Flexión:** Es todo movimiento en el plano sagital que desplaza una parte del cuerpo hacia delante de la posición anatómica.
- **Extensión:** Es todo movimiento en el plano sagital que desplaza una parte del cuerpo hacia atrás de la posición anatómica.

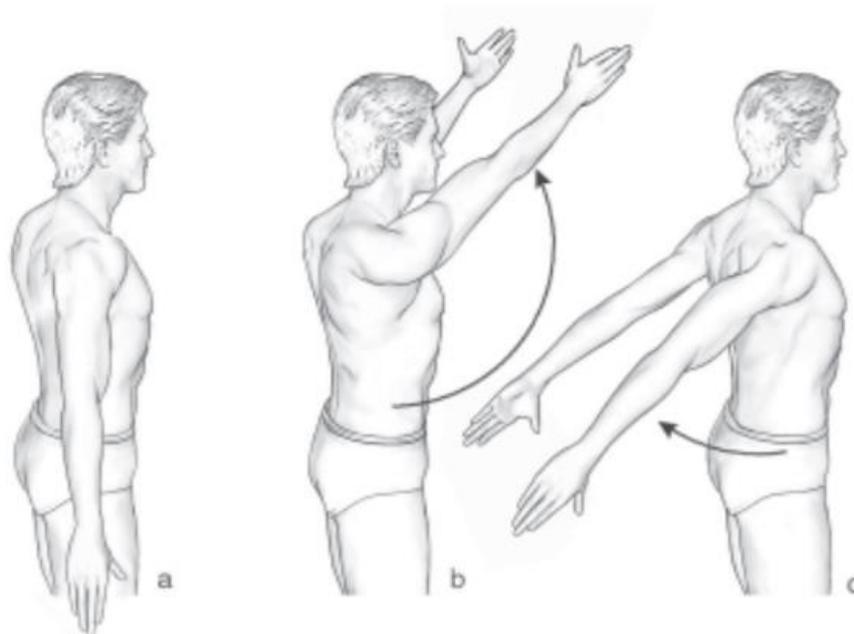


Figura 49: Flexión-extensión del hombro derecho: a) posición neutra de hombro; b) flexión de hombro y c) extensión del hombro.

Los distintos movimientos que una articulación puede realizar con respecto al **plano frontal**, están definidos como:

- **Abducción:** Es todo movimiento en el plano frontal que aleja una parte del cuerpo de la línea media.
- **Aducción:** Es todo movimiento que en el plano frontal acerca una parte del cuerpo a la línea media.

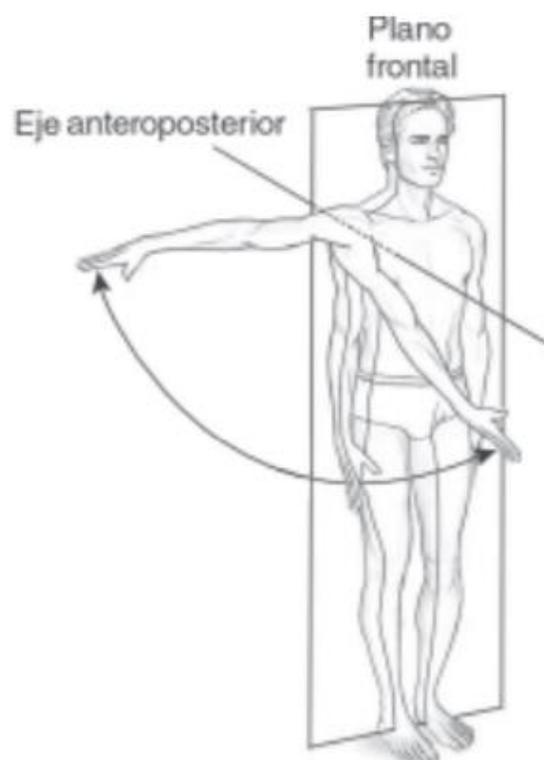


Figura 50: Movimientos de abducción-aducción sobre el eje anteroposterior del hombro derecho en el plano frontal

Los distintos movimientos que una articulación puede realizar con respecto al **plano transversal**, están definidos como:

- **Rotación externa:** Es todo movimiento en el plano transversal que desplaza una parte del cuerpo hacia fuera.
- **Extensión:** Es todo movimiento en el plano transversal que desplaza una parte del cuerpo hacia dentro.

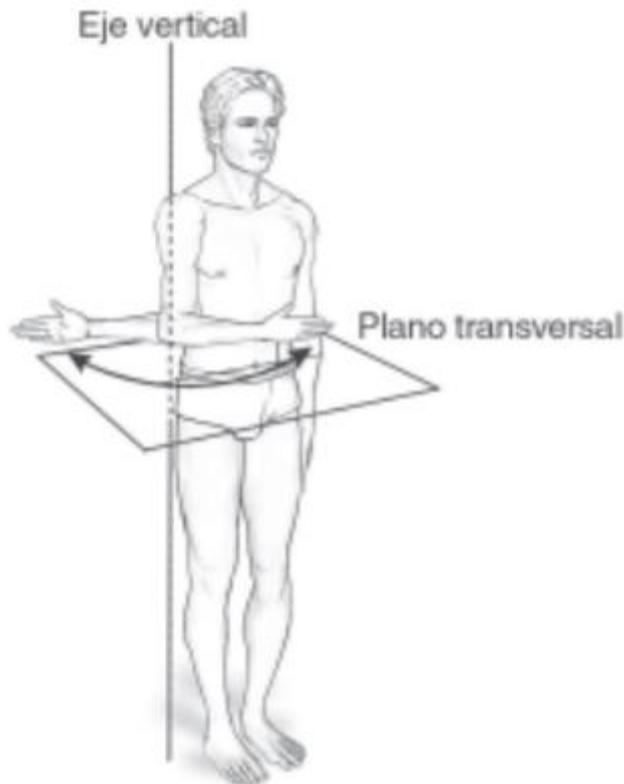


Figura 51: Movimientos de rotación del hombro derecho sobre el eje vertical en el plano transversal

Por último, cabe aclarar que sobre el nombre de los movimientos articulares hay algunas **excepciones**. Es el caso de la rodilla, por ejemplo, denominándose paradójicamente extensión al movimiento que lleva la pierna hacia delante de la posición neutra, y flexión, al movimiento que la lleva hacia atrás. También sucede en otras articulaciones como el tobillo, en la que se denomina extensión al movimiento que lleva el segmento distal (pie) hacia delante, y flexión, cuando lo lleva hacia atrás

6.1.2 Cuaterniones

Los **cuaterniones** o números cuaternios, descubiertos por parte del matemático irlandés William Rowan Hamilton en el siglo XIX, llevaron a la extensión del plano complejo a las tres dimensiones abriendo el paso al estudio y el desarrollo de las nuevas álgebras no conmutativas y a una nueva interpretación tridimensional de la realidad física (Sánchez Muñoz, 2011)

Mientras que los números complejos son una extensión de los reales por la adición de la unidad imaginaria i , tal que $i^2 = -1$, los cuaterniones son una extensión generada de manera análoga añadiendo las unidades imaginarias: i, j y k a los números reales y tal que $i^2 = j^2 = k^2 = -1$.

Los cuaterniones, en general, satisfacen casi todas las propiedades usuales del álgebra común, a excepción de una (bastante importante): la conmutativa del producto de cuaterniones (" $q1 \cdot q2$ " no es lo mismo que " $q2 \cdot q1$ "). La prueba más básica de que el producto de cuaterniones no es conmutativo viene directamente de la identidad de Hamilton:

$$ij = -ji = k ; jk = -kj = i ; ki = -ik = j \quad (1)$$

Estos números son actualmente de amplia utilización en procedimientos de programación sobre todo utilizados como operadores matemáticos para rotar y estirar vectores (Van Verth, 2013).

Comentaremos algunas propiedades importantes para entender el desarrollo posterior:

- Se puede verificar fácilmente que, siendo $q1$ y $q2$ cuaterniones de la forma:

$$q1 \cdot q2 = (s1, v1) \cdot (s2, v2) = (s1 \cdot s2 - v1 \cdot v2, s1 \cdot v2 + s2 \cdot v1 + v1 \wedge v2) \quad (2)$$

donde la operación " \cdot " para vectores representa el producto escalar, y " \wedge " representa el producto vectorial.

- Además, si:

$$q = w + x \cdot i + y \cdot j + z \cdot k \quad (3)$$

entonces un Cuaternión conjugado con q es:

$$q' = w - x \cdot i - y \cdot j - z \cdot k ; \text{ es decir: } q' = (s, -v) \quad (4)$$

- La norma (longitud) de un Cuaternión q se define como:

$$|q| = \text{sqrt}(w \cdot w + x \cdot x + y \cdot y + z \cdot z) \quad (5)$$

donde sqrt es la operación de extracción de raíz cuadrada.

- Un cuaternión con la norma igual a la unidad se llama unitario.

- Un cuaternión inverso con respecto al cuaternión \mathbf{q} , se define por la expresión:

$$\text{inv}(q) = \frac{q'}{(q \cdot q')} \quad (6)$$

- Para un cuaternión unitario su inverso coincide con su conjugado:

$$\text{inv}(q) = q' \text{ si } |q| = 1 \quad (7)$$

Sea un punto \mathbf{p} en el espacio de 3 dimensiones: " $\mathbf{p} = \mathbf{p}(x, y, z)$ ". Para poder rotarlo en el espacio alrededor de un eje pasando por el origen del sistema de coordenadas el punto se representa en forma de un **cuaternión** " $\mathbf{P} = (0, \mathbf{p})$ ". El eje de rotación se determina por un **cuaternión unitario** " $c = (0, \mathbf{u})$ ", donde el vector \mathbf{u} coincide con la dirección del eje de rotación. (Markelov, 2002).

La operación de **rotación** en un **ángulo** ϕ alrededor del eje definido por el cuaternión unitario " $c = (0, \mathbf{u})$ " se representa por un cuaternión unitario:

$$\mathbf{q} = (s, \mathbf{v}); \text{ donde } s = \cos \frac{\phi}{2} \text{ y } \mathbf{v} = \mathbf{u} \cdot \sin \frac{\phi}{2}; \quad (8)$$

$$\text{es decir, } \mathbf{q} = \left(\cos \frac{\phi}{2}, \mathbf{u} \cdot \sin \frac{\phi}{2} \right)$$

El resultado de la rotación, un punto \mathbf{r} , se obtiene en forma de un cuaternión " $\mathbf{R} = (0, \mathbf{r})$ " por medio de la siguiente operación de multiplicación de tres cuaterniones: " $\mathbf{R} = \mathbf{q} \cdot \mathbf{P} \cdot \mathbf{q}'$ ".

Sean dos cuaterniones unitarios $\mathbf{q1}$ y $\mathbf{q2}$ que representan dos rotaciones, si primero se realiza la rotación $\mathbf{q1}$ y después la rotación $\mathbf{q2}$, entonces la posición final de un punto determinado por un cuaternión \mathbf{P} se obtiene por medio de las siguientes operaciones:

$$\begin{aligned} \mathbf{q2} \cdot (\mathbf{q1} \cdot \mathbf{P} \cdot \mathbf{q1}') \cdot \mathbf{q2}' &= (\mathbf{q2} \cdot \mathbf{q1}) \cdot \mathbf{P} \cdot (\mathbf{q1}' \cdot \mathbf{q2}') = \\ &= (\mathbf{q2} \cdot \mathbf{q1}) \cdot \mathbf{P} \cdot (\mathbf{q2} \cdot \mathbf{q1})' = \mathbf{q} \cdot \mathbf{P} \cdot \mathbf{q}' \end{aligned} \quad (9)$$

Es decir, la realización de dos rotaciones en serie primero $\mathbf{q1}$ y después $\mathbf{q2}$ es equivalente a una sola rotación \mathbf{q} igual al producto de dos rotaciones $\mathbf{q2} \cdot \mathbf{q1}$. Este resultado es muy práctico pues se puede siempre sustituir una serie de rotaciones determinadas por cuaterniones por una rotación resultante determinada por un solo cuaternión. De hecho, esto permite trabajar con cuaterniones de forma más sencilla que con matrices de rotación o procedimientos de rotación basados en el uso de ángulos de Euler (los que además pueden inducir a error debido al Gimball Lock – ver **Anexo II: IMU Funcionamiento** –)

Esta característica distintiva de los cuaterniones, nos permitirá analizar las rotaciones de una forma matemáticamente más sencilla, aunque de mayor complejidad para su interpretación directa.

6.1.3 Dinámica del hombro

Para la evaluación de nuestro prototipo, hemos considerado el análisis de la dinámica del hombro, crucial para muchas tareas dentro de un puesto de trabajo de cadena de producción. Es por tanto una articulación clave en nuestra labor de valoración ergonómica.

Desde la posición anatómica decúbito supino, buscaremos medir con precisión los distintos movimientos de Abducción/Aducción, Flexión/Extensión y Rotación medial/lateral que corresponden con los tres ejes de movimiento. Nuestra labor será determinar con los datos recibidos por parte de los sensores, la diferencia angular en grados entre diferentes partes del cuerpo, para así determinar la variación angular de cada movimiento por cada articulación.

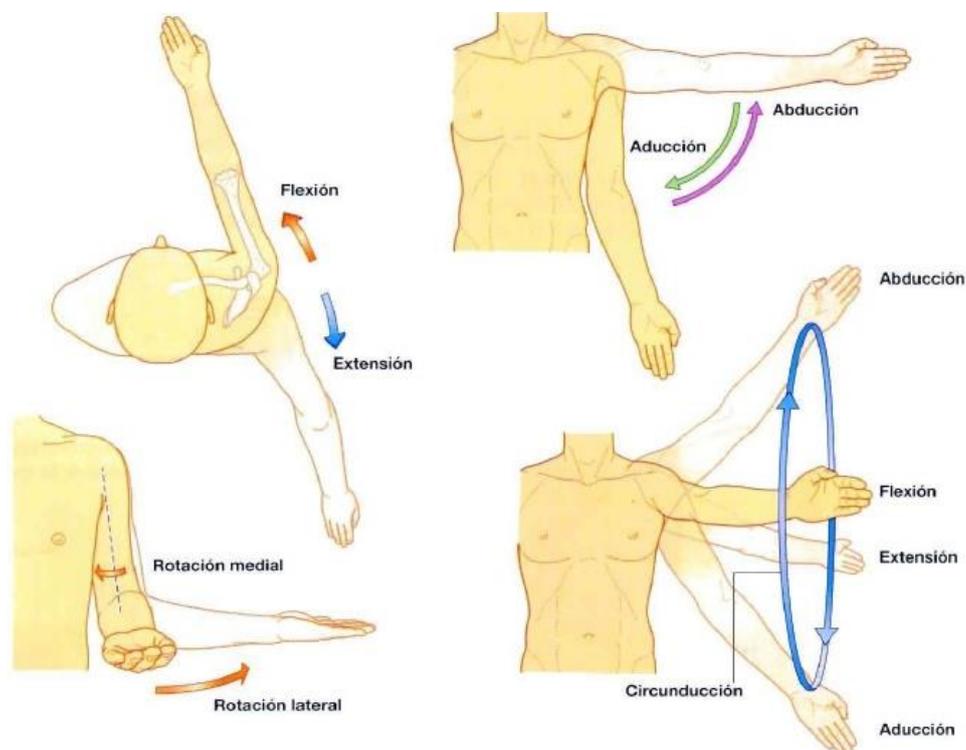


Figura 52: Movimientos del hombro en los distintos planos.

Para ello nos basaremos en el análisis con cuaterniones desarrollado en el apartado anterior, que mediante la división de dos cuaterniones definidos por cada una de las partes del cuerpo que conforman la articulación nos determinará un cuaternión resultante, que definirá el “desplazamiento angular” de la articulación.

Mediante las fórmulas matemáticas de conversión de cuaterniones a ángulos en los distintos ejes, obtendremos los grados de movimiento de la articulación:

En dinámica de vuelo sobre todo (para posicionamiento de drones, aviones o vehículos), de forma similar a los ángulos de Euler, usamos los ángulos de Tait Bryan:

- Bank – ϕ : rotación sobre el nuevo eje X
- Attitude – θ : rotación sobre el nuevo eje Y
- Heading – ψ : rotación sobre el eje Z

Con el eje X apuntando hacia delante, el eje Y hacia la derecha y el eje Z hacia abajo, con ángulos definidos para la rotación con sentido derecha a izquierda.

$$\begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} = \begin{bmatrix} \text{atan2}(2(q_0 q_1 + q_2 q_3), 1 - 2(q_1^2 + q_2^2)) \\ \text{asin}(2(q_0 q_2 - q_3 q_1)) \\ \text{atan2}(2(q_0 q_3 + q_1 q_2), 1 - 2(q_2^2 + q_3^2)) \end{bmatrix} \quad (10)$$

***Atan2 o Arctan2:**

Su principal utilidad, la expresión $\text{Atan2}(y, x) - \arctan(y/x)$ no evalúa a 0 para $x < 0$, sino a $+\pi$ (180 °, para $y > 0$), y $-\pi$ (-180 °, para $y < 0$). Así resuelve la inconveniencia de los valores principales de \arctan que cubren solo dos cuadrantes del plano euclidiano, además de evitar el problema de la división por cero.

6.2 Estudio preliminar de los requisitos

6.2.1 Requisitos generales del proyecto

Comentaremos a continuación los requisitos generales del proyecto desarrollado en este Trabajo Fin de Máster, teniendo en cuenta que, debido a los posibles propósitos comerciales próximos, no se permite la utilización de materiales o propiedad intelectual que hayan sido restringidos a otros para cualquier uso que esté contemplado en este documento.

1. Responderá a una **perspectiva general**, en que aparezca la descripción de características principales, aplicabilidad, etc. Estos requerimientos serán principalmente funcionales, mecánicos y de hardware; apareciendo requerimientos de conformidad como regulaciones en el futuro proceso de comercialización.
2. El proyecto, por estar relacionado con software ya desarrollado, deberá utilizar unos **estándares** concretos para su integración en el sistema. Esos estándares determinarán el desarrollo de cada prototipo hardware.

3. Los sistemas desarrollados deben responder a unas características de **usuario** bien conocidas y sobre las cuales se deberá orientar todo el desarrollo para facilitar las tareas de usabilidad y control del producto.
4. El producto deberá tener en cuenta la descripción del **ambiente** operacional en el cual funcionará en su uso ordinario.
5. Deberá responder a unas **restricciones** generales, ya sean políticas de regulación, costo, limitaciones de hardware, interfaz a otras aplicaciones, operación paralela, funciones de control, protocolos, u otras consideraciones.
6. Se deberán tener en cuenta las ocasiones en que el cliente será partícipe del proceso de desarrollo, así como la **comunicación** con el otro equipo (encargado del software) de desarrollo y los requerimientos de infraestructura.

6.2.2 Requisitos particulares del proyecto

Analizaremos los requisitos para cada caso particular de estudio:

Software de valoración ergonómica

1. El sistema deberá responder en tiempo real a las **interacciones**, enviando los datos de los sensores a una velocidad suficiente para una reacción fluida de la simulación y para evitar la pérdida de datos.
2. La aplicación se destinará a profesionales de la evaluación ergonómica, y por tanto debe resultar **intuitiva** y **realista** respecto a las metodologías reales de evaluación ergonómica. En nuestro caso concreto, nos encargaremos de la ergonomía del hardware y de su facilidad de uso, además de la claridad de las gráficas resultantes del análisis.
3. El dispositivo se utilizará comúnmente en puestos reales de trabajo de empresas de cadena de producción. Por tanto, se deberá tener en cuenta para un correcto encapsulado que elimine emisiones indeseadas y espurios que puedan afectar a la comunicación Bluetooth, WiFi y al magnetómetro del sensor.
4. Las **limitaciones de hardware** vendrán dadas principalmente por el tamaño reducido necesario y el bajo coste. El coste deberá ser ajustado para presentar una verdadera alternativa ante los dispositivos de simulación comerciales, ya que el principal inconveniente de estos, es su elevado precio además de ser la mayoría soluciones cableadas. Los protocolos e interfaz utilizados se describirán en posteriores apartados.
5. El **cliente**, en este caso profesionales de la valoración ergonómica en trabajos de cadena, formará parte del proceso de desarrollo definiendo los requisitos y proporcionando **feedback** periódicamente a lo largo del proceso de desarrollo.

Sistema de Rehabilitación

1. El primer punto es común para ambos apartados, añadiendo en este caso una correcta **sincronización** del sistema Wi-Fi intra-dispositivos y Bluetooth con el PC que ejecute la aplicación.
2. El prototipo final deberá poderse incorporar correctamente al sistema TELEKIN, para ello deberá ser capaz comunicarse con **Unity3D** a través del Bluetooth incorporado. Además, su funcionamiento deberá poder ser controlado mediante un script en C# que detalle su actividad.
3. La aplicación debe ser fácilmente incorporable a cualquier módulo de trabajo del sistema **TELEKIN**. Y lo suficientemente ergonómica como para que su colocación en las extremidades de nuestro cuerpo por medio de velcro no resulte incómoda.
4. El dispositivo se utilizará dentro de un **entorno** de realidad virtual **doméstico o médico**, pues estará destinado a la rehabilitación de personas adultas en su propio domicilio o en centros médicos. Por tanto, no presentará más restricciones que su correcta sujeción en las extremidades, espalda, cadera etc. del usuario.
5. El **coste** deberá también ser **reducido**, así como su **tamaño**, ya que se pretende la utilización de más de un dispositivo colocado en distintas articulaciones del paciente. Además, la unidad principal se colocará comúnmente en la espalda por medio de cintas o un cinturón corrector de postura, ya que las lesiones de espalda concentran el mayor porcentaje de lesiones musculoesqueléticas. El protocolo e interfaz utilizados también se describirá posteriormente en detalle.
6. En este caso, al ser muy amplio el target de población al que pertenece, se deberá realizar un proceso largo de **testeo** antes de su comercialización, ya que, además uno de los rangos de clientes potenciales pertenece a la **tercera edad**, la cual es propensa a presentar dificultades de uso de estos dispositivos electrónicos. Por tanto, la interconexión de sistemas deberá ser ajena al usuario, automática e intuitiva.

6.3 Planificación del proyecto

6.3.1 Metodología de planificación y desarrollo de proyectos

En un proyecto de dimensiones considerables es necesario incorporar herramientas de **organización y control de versiones** eficaces. Para ello utilizaremos diversos mecanismos que se complementarán para facilitar la tarea de control y certeza obteniendo un código limpio y totalmente funcional.

A lo largo del proceso de desarrollo del proyecto, se establecerá en primer lugar la comunicación tanto con médicos rehabilitadores como con profesionales de la

ergonomía en el trabajo, para establecer los requisitos de los productos, y en segundo lugar con el resto del equipo de desarrollo. En nuestro caso, las propuestas se realizaron a nuestro laboratorio del Grupo de Telemática e Imagen que ya tenía en marcha varios proyectos de rehabilitación con dispositivos de captura de imagen proponiéndose una migración a dispositivos inalámbricos. Dichos dispositivos serían desarrollados exprofeso para estas propuestas no sin considerar el amplio abanico de posibilidades que brindaría un desarrollo de dichas características obteniendo un prototipo único y fiable.

Al ser un proyecto de futuro, la comunicación entre el equipo encargado del desarrollo del Hardware y el encargado del Software debe ser constante. En este caso, aunque sea un proyecto desarrollado íntegramente en el laboratorio del GTI, entre distintos miembros del equipo.

En este caso concreto, utilizaremos en el equipo interno del laboratorio, la herramienta GitLAB para control de proyectos, seguridad y limpieza de código. Estos mecanismos de desarrollo y control están expuestos de forma extendida en el ***Anexo I: Metodología de planificación y desarrollo de proyectos.***

6.3.2 Calendario temporal

Al comienzo del Trabajo de Fin de Máster, se planificó un **plan de trabajo** para llevar a cabo el proyecto antes de la fecha de presentación. Según la normativa debería conllevar alrededor de 450 horas, aunque debido al número de iteraciones del proyecto y la continuidad con el Trabajo Fin de Grado que lo precedía, la cantidad fue en este caso superior.

Para ello, se dividió el proyecto en varias fases y se elaboró un diagrama de Gantt que estimara el tiempo de cada fase del desarrollo. En la siguiente figura se muestra el calendario temporal con las horas llevadas a cabo durante el proyecto:

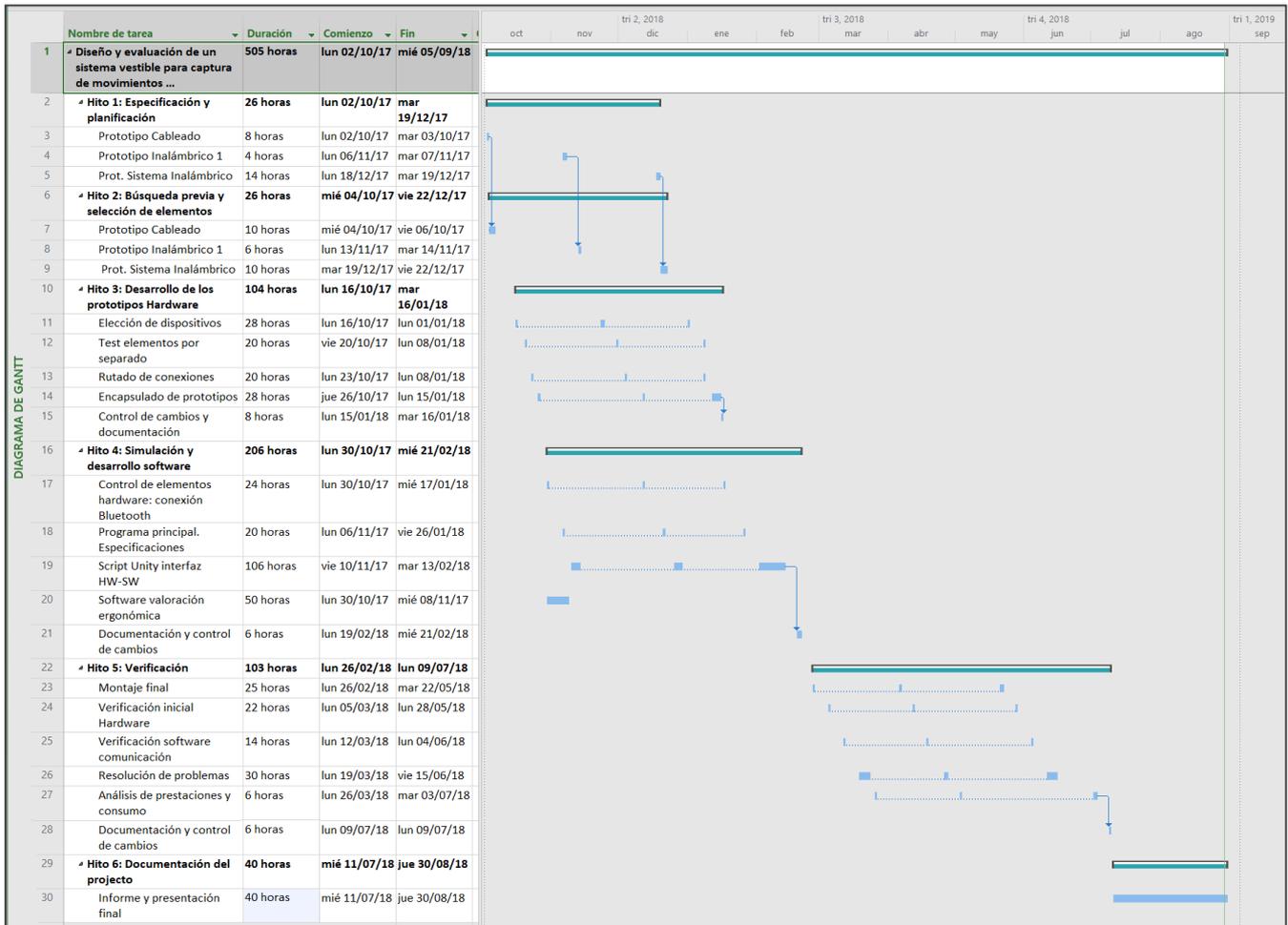


Figura 53: Diagrama de Gantt del proyecto.

6.4 Establecimiento de la batería de pruebas

Estas pruebas constituyeron una forma fiable de determinar la validez del sistema final (modelo ESP32). Para ello, se comparó el desempeño de la cámara Kinect® dentro o fuera de la plataforma TELEKIN con los sensores inalámbricos.

Además, se estudiaron las capacidades de nuestro sistema vestible tanto para medición de rango articular en tiempo real, estudio del ROM y análisis ergonómico.

Se comprobó, por tanto, la fiabilidad de los prototipos inalámbricos desarrollados según los criterios aplicados a Kinect® y a otros dispositivos de medición considerados como “golden standard”. Además, se constató la facilidad del nuevo sistema para solucionar algunas limitaciones tecnológicas de los sistemas por cámaras como Kinect®.

6.4.1 Limitaciones de sistemas basados en cámaras

Respecto a las limitaciones propias del Hardware, los sistemas de monitorización basados en cámaras no están exentos de ellas. Son conocidos los problemas asociados a este Hardware, y para conocer cuántos son solucionados por la nueva implementación se establecieron una serie de parámetros a evaluar:

- Oclusiones por materiales
- Auto-occlusiones (por el propio cuerpo)
- Posición tumbada
- Posiciones indetectables (como a través de superficies)
- Posición “de espaldas a la cámara”

Como en el caso de las encuestas, fueron puntuados del 1 al 5 para representar la menor o mayor capacidad del sistema inalámbrico de resolver las limitaciones observadas con Kinect® u otros sistemas de reconocimiento basados en cámaras.

6.4.2 Pruebas evaluación ROM en tiempo real

Para evaluar la capacidad del sistema software desarrollado en Unity3D para la visualización en tiempo real de movimientos con fin de incorporarse al sistema TELEKIN, se realizaron mediciones del rango de movimiento mediante un goniómetro digital para comparar posteriormente con los datos recogidos en tiempo real por el sistema software.

Para comparar los resultados obtenidos gracias a nuestros dispositivos sin cables, primero vestimos a un usuario con los sensores necesarios para la captura de las distintas articulaciones del miembro superior, y al mismo tiempo, como medida de comprobación, grabaríamos los movimientos llevados a cabo.

El software, nos permitiría conocer de forma exacta el movimiento registrado, ya que dispone de opciones para medir los grados de movimiento entre articulaciones en tiempo real.

Para poder realizar una correcta comparación, elegiríamos movimientos que involucraran solamente al miembro superior, divididos según articulación involucrada. Podemos observar los distintos ejercicios de test realizados en la siguiente tabla a rellenar una vez por cada ejecución de la batería de pruebas:

La comparación se realizaría mediante una tabla en la que se recogerán los movimientos realizados, comparándolos con los observados en el vídeo centrándonos sobre todo en el ángulo máximo. Se trata así de suplir la no implementación de la cámara Kinect® en el sistema de valoración ergonómica, ya que éste ha sido desarrollado explícitamente para el sistema inalámbrico.

Los ángulos por tanto se verán en el vídeo en la pantalla del goniómetro digital, lo que podrá ser complementado con el análisis de forma gráfica, contando para ello con varios ángulos de visionado para la correcta valoración del rango articular.

		<i>Espalda</i>	<i>Hombro</i>	<i>Codo</i>	
<i>Abducción</i>	Vídeo	... °	... °	... °	<i>Rango articular máximo</i>
	Sensores	... °	... °	... °	
<i>Aducción</i>	Vídeo	... °	... °	... °	<i>Rango articular máximo</i>
	Sensores	... °	... °	... °	
<i>Rotación interna</i>	Vídeo	... °	... °	... °	<i>Rango articular máximo</i>
	Sensores	... °	... °	... °	
<i>Rotación externa</i>	Vídeo	... °	... °	... °	<i>Rango articular máximo</i>
	Sensores	... °	... °	... °	
<i>Flexión</i>	Vídeo	... °	... °	... °	<i>Rango articular máximo</i>
	Sensores	... °	... °	... °	
<i>Extensión</i>	Vídeo	... °	... °	... °	<i>Rango articular máximo</i>
	Sensores	... °	... °	... °	

Tabla 4: Plantilla ángulos de grabación para la valoración articular. (para nuestras pruebas nos centraremos únicamente en el hombro)

6.4.3 Pruebas herramienta de análisis ergonómico (a posteriori)

Se ha realizado una evaluación angular a posteriori de una serie de movimientos estándar que implicaran la articulación del hombro. Para ello, se ha utilizado una plantilla goniométrica y con los sensores correctamente colocados se han llevado a cabo distintos movimientos articulares siguiendo un patrón fácilmente comparable con los resultados vertidos por el software de análisis ergonómico. Así podremos comparar los distintos ángulos que realicemos en cada uno de los planos según lo explicado en la sección 4.1.3, con los ángulos alcanzados en la gráfica resultado del análisis, para determinar la fiabilidad del sistema.

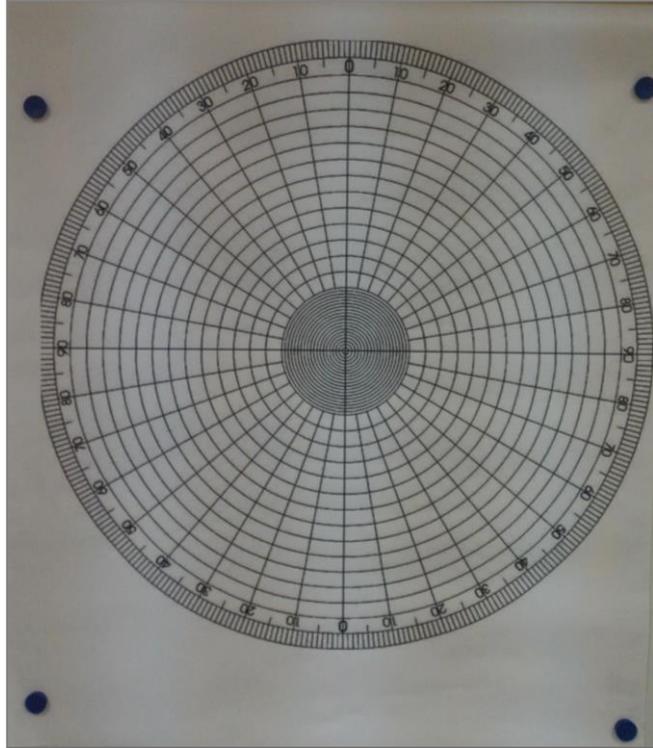


Figura 54: plantilla goniométrica a utilizar en las pruebas.

Aunque debido a la complejidad de los movimientos posibles, se ha centrado el estudio en realizar un barrido angular, con parada cada 10/15° en cada uno de los distintos ejes de movimiento, ya sea flexoextensión, abducción/adducción o rotación interna o externa del hombro.

Capítulo 7 Resultados

En este capítulo se muestran los resultados recogidos por la batería de pruebas llevadas a cabo con el prototipo final, así como se discuten los principales resultados obtenidos tras el pilotaje.

7.1 Resultados

En este apartado se van a exponer los resultados tras las pruebas con los dispositivos ya montados y completamente funcionales, según la planificación detallada en el punto anterior.

Cabe destacar que no se realizaron pruebas extensas con los prototipos anteriores al prototipo final: la versión *V2: versión BT autónoma* se utilizó únicamente para el desarrollo de juegos serios, los cuales no entran dentro del contenido de este TFM; y con la *V1: versión cableada* sólo se realizaron pruebas de ergonomía que se comentarán con brevedad por la limitación observada en los rangos articulares máximos.

7.1.1 Resultados Dispositivo V1: versión cableada

En este caso, sólo realizamos las pruebas del punto **6.4.3 Pruebas herramienta de análisis ergonómico (a posteriori)**, simplemente para validar la fiabilidad del sistema.



Figura 55: Dispositivo cableado pruebas en línea de montaje.

Antes de proceder a comentar los resultados, es destacable comentar la aparatosidad de estas soluciones cableadas, que serán más complejas cuantas más articulaciones permitan medir.

Es por esa misma aparatosidad, concretamente debido a la utilización de cables de varias líneas (concretamente Vcc, GND, SDA y SCL por sensor), que las dimensiones de estas impidieron la observación correcta de movimientos de amplitud superior a los 80° modificando además las medidas posteriores del sistema. Todo ello se producía por la tensión (física) que sufría el cable en dichos movimientos amplios, trasladándose a la modificación de la posición de los sensores en el brazo para los movimientos posteriores.



Figura 56: Gráfica Flexo-extensión dispositivo cableado.

Como podemos apreciar en la gráfica, en el sistema cableado no se incluía un mecanismo de detección de posición inicial (como si ocurre en la versión inalámbrica), por lo que a los grados obtenidos deberemos sumarle "+10°". Esto nos deja con valores que nunca superan los 80-85° de extensión, aunque si se hayan superado en la realidad.

La aparición de estos problemas en repetidas tomas, fue uno de los motivos principales que fomentó el desarrollo de las versiones inalámbricas.

7.1.2 Resultados Dispositivo Hardware Definitivo (ESP32)

Con el dispositivo final, se pudo registrar toda la batería de pruebas pensadas para este proyecto. Pasaremos a continuación a comentar los resultados de cada una de ellas.

Limitaciones de sistemas basados en cámaras

Pasaremos a comentar las distintas limitaciones de sistemas basados en cámaras como Kinect® de Microsoft, comparando los distintos casos con el comportamiento de nuestro sistema. Para ello, primero observaremos los fallos de detección que se producen utilizando los sistemas basados en cámaras y a continuación los buenos resultados obtenidos por nuestro sistema vestible ante la misma situación:

KINECT®:

- **Oclusiones por materiales:**

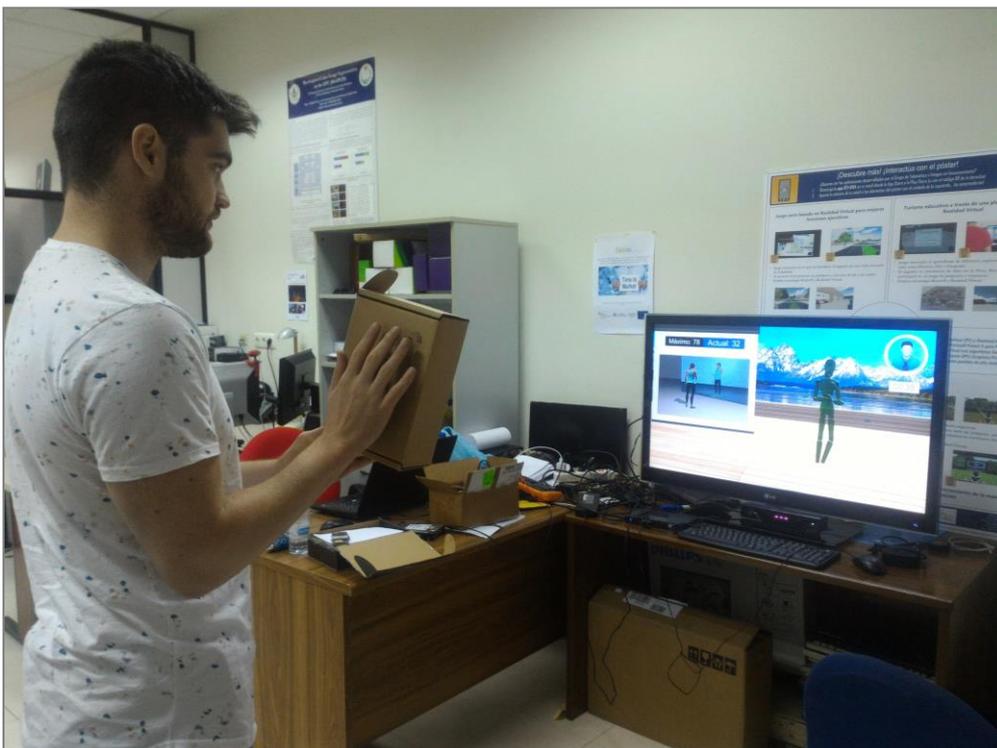


Figura 57: Detección incorrecta de la mano por oclusión.



Figura 58: Detección incorrecta de las piernas por oclusión.

- **Auto-oclusiones (por el propio cuerpo):**

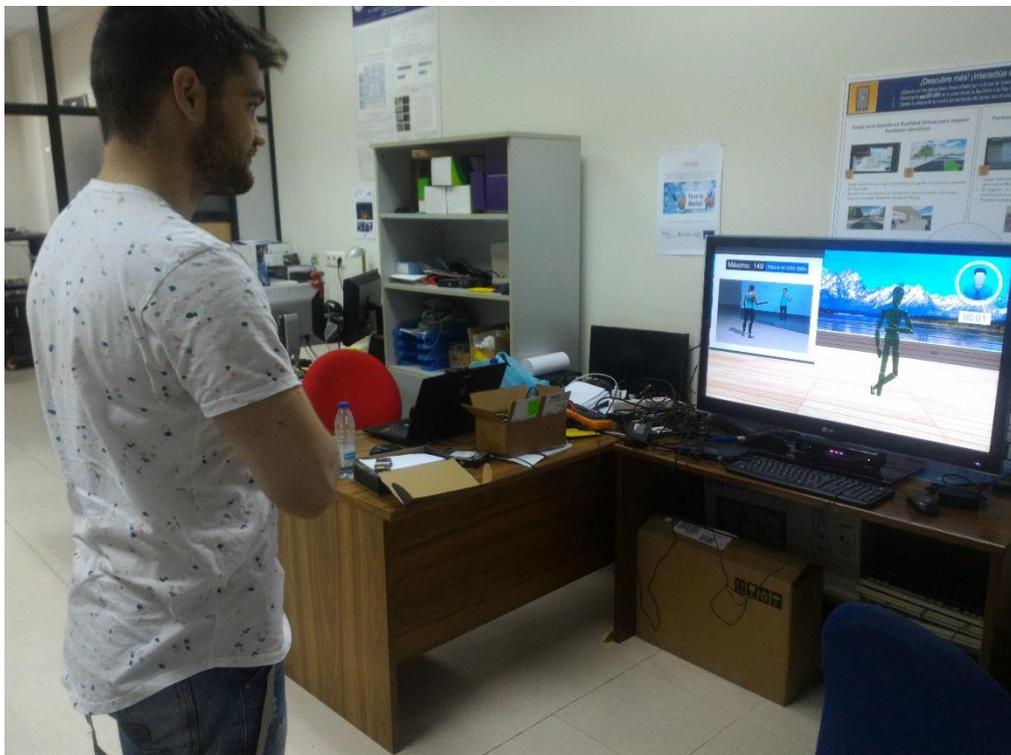


Figura 59: Comportamiento normal detección antebrazo sin oclusión.



Figura 60: Detección incorrecta del antebrazo por auto-oclusión.

- **Posición tumbada:**

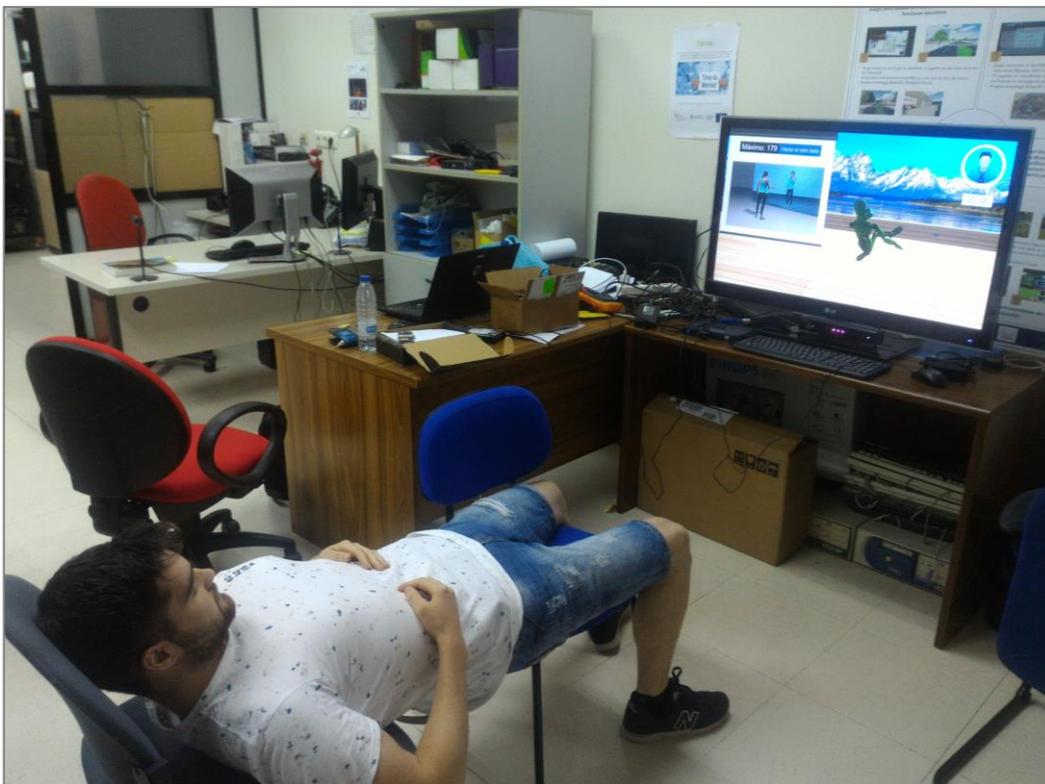


Figura 61: Detección incorrecta general por posición tumbada del usuario.



Figura 62: Detección incorrecta general por posición tumbada del usuario.

- **Posición “de espaldas a la cámara”:**

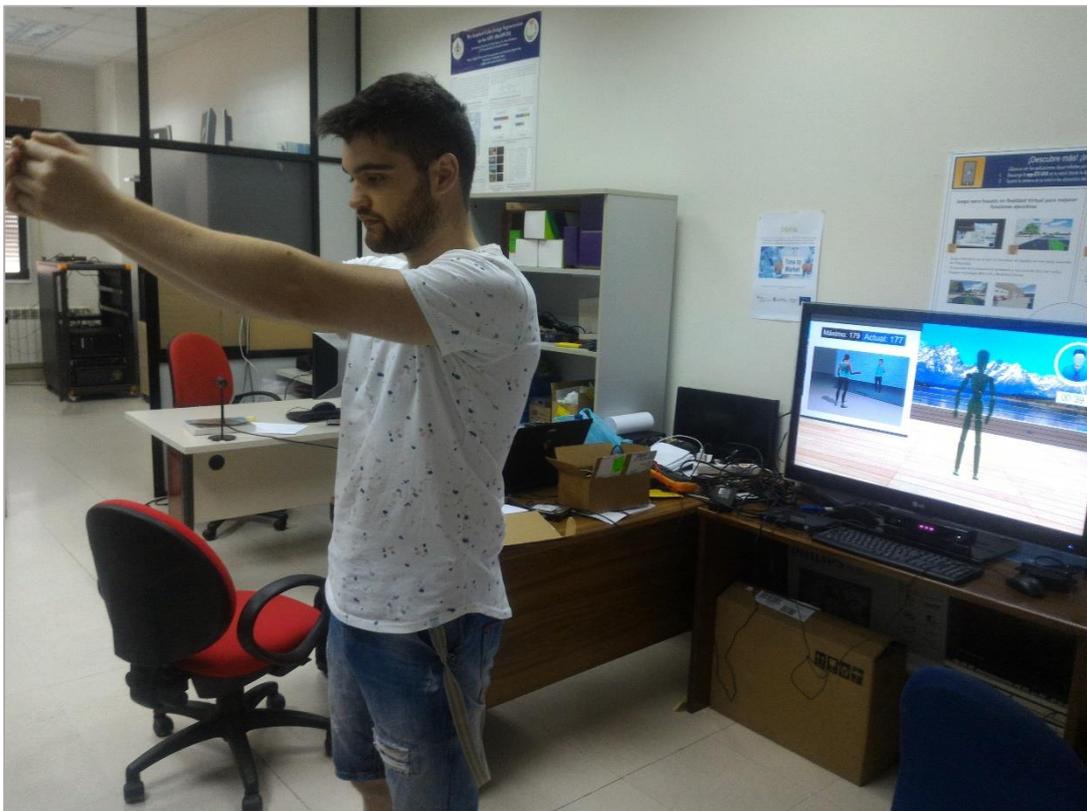


Figura 63: Detección incorrecta general por posición de espaldas a la cámara.

- **Posiciones que incluyan varios de los puntos anteriores:**



Figura 64: Detección incorrecta por auto-oclusión y posición de espaldas a la cámara.

SISTEMA VESTIBLE:

A continuación, expondremos los mismos casos que produjeron problemas de detección con sistemas basados en cámaras como Kinect®, utilizando nuestro sistema vestible. Como podemos observar, no se produce ninguna detección incorrecta ante los mismos supuestos.

- **Oclusiones por materiales:**



Figura 65: Detección correcta de la mano posición 1 (caso oclusión cámara).

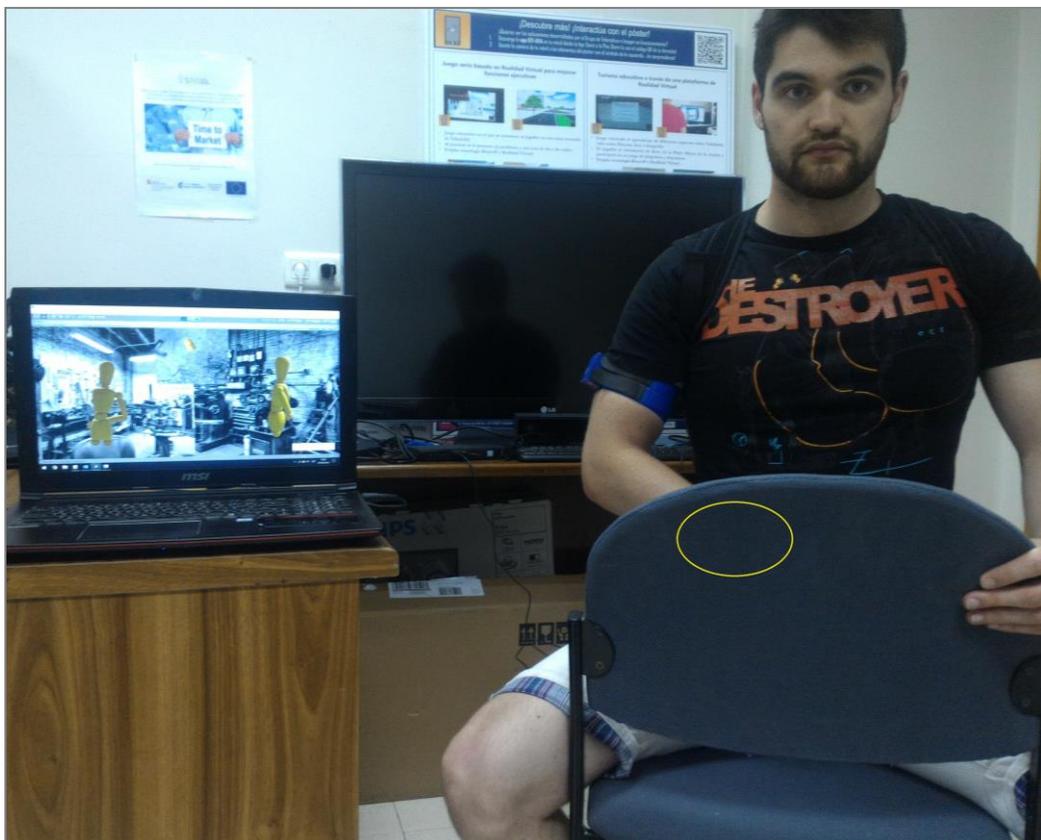


Figura 66: Detección correcta de la mano posición 2 (caso oclusión cámara).

- **Auto-oclusiones (por el propio cuerpo):**



Figura 67: Comportamiento normal detección antebrazo sin oclusión.

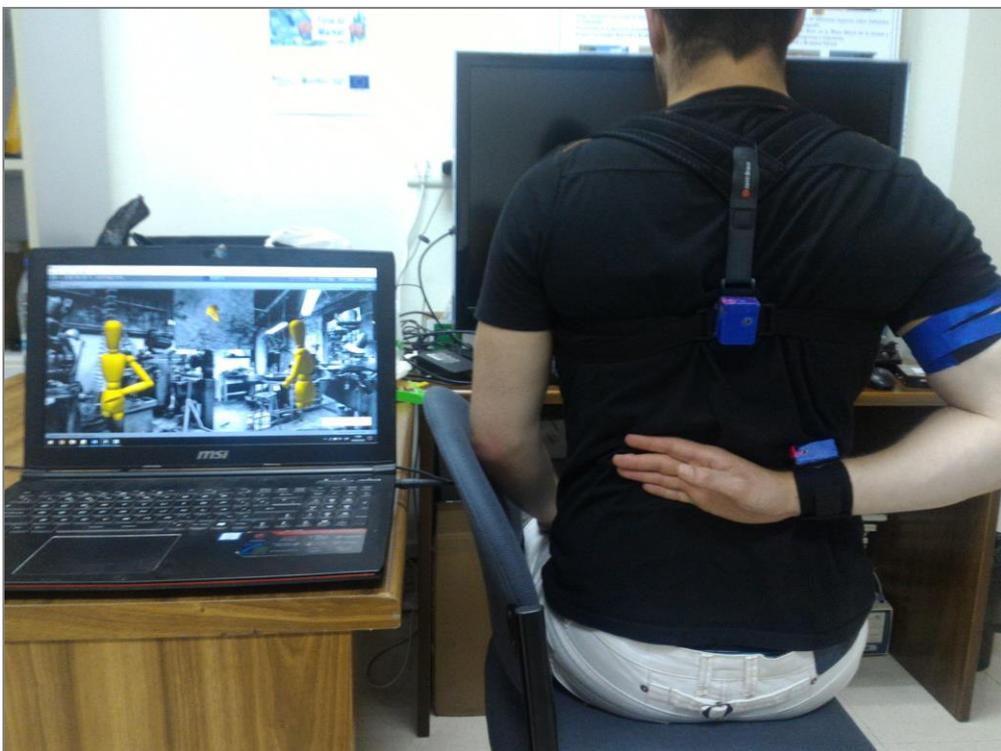


Figura 68: Detección correcta del antebrazo con auto-oclusión para cámara.

- **Posición tumbada:**

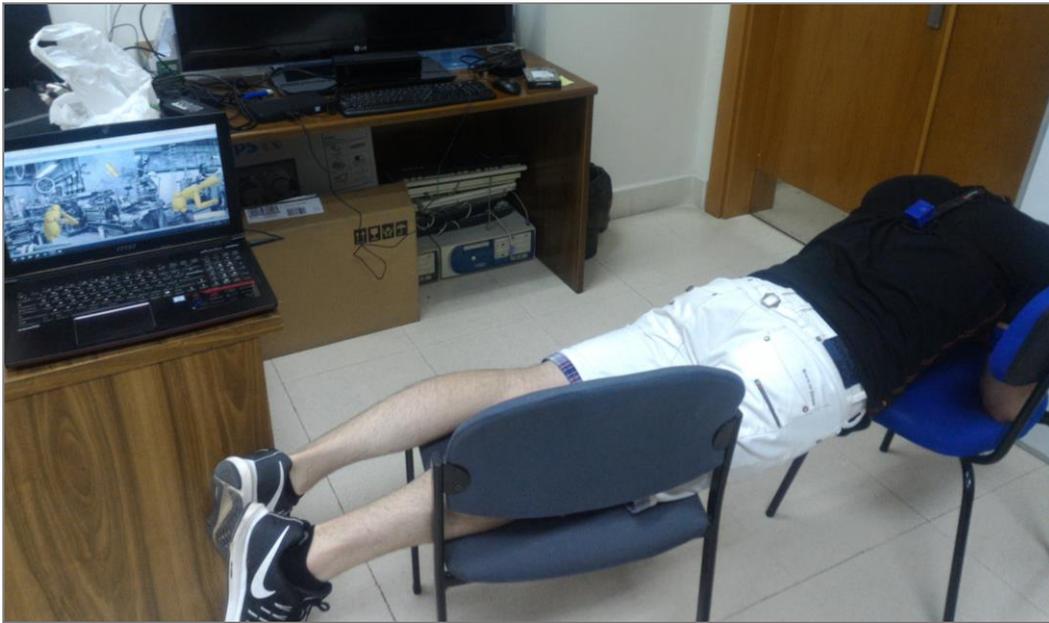


Figura 69: Posición tumbada del usuario. Detección correcta del sistema.

Podemos ver cómo la cadera en el programa no está en la posición propia para un cuerpo tumbado, pero esto es debido a que no hemos colocado sensores en la cadera del usuario; las zonas con sensores se recogen correctamente.

- **Posición “de espaldas a la cámara”:**



Figura 70: Detección correcta general en posición de frente a la cámara.



Figura 71: Detección correcta general en posición de espaldas a la cámara.

El general el sistema en cuanto a posiciones no tolerables para una correcta detección es mucho más robusto que los sistemas basados en cámaras. Podemos capturar incluso los movimientos que se producen en habitaciones contiguas sin visión directa del usuario o a varios metros de distancia (dependiendo directamente de la calidad de la conexión Bluetooth).

Pruebas evaluación ROM en tiempo real

Para el análisis en este caso comparativo entre vídeo y datos de rango articular máximo, hemos realizado las medidas del ROM típicas, colocándonos para ello en posiciones estándar y grabando en video el movimiento hasta alcanzar el rango máximo, ayudándonos de un goniómetro digital para cerciorarnos de la medida. Después hemos comparado los valores con los ángulos máximos obtenidos con los sensores. La captura del programa Unity3D 3D puede resultar confusa respecto a la articulación de la cadera, pero únicamente representa que no se han colocado sensores para capturar la posición de la cadera, observándose vertical.

Los principales movimientos para medida del rango articular máximo del hombro que hemos capturado son:

- **Abducción/Aducción**

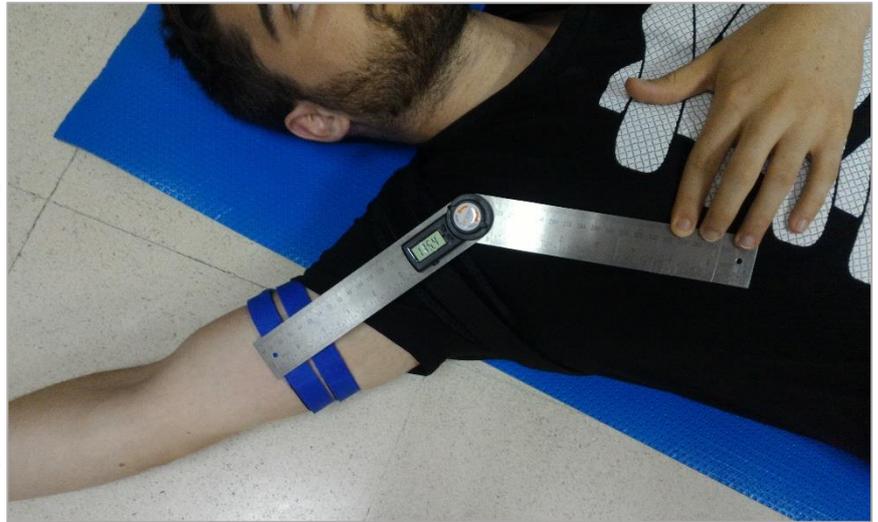
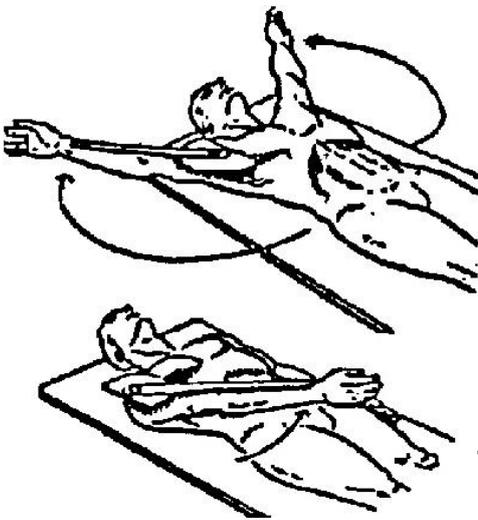


Figura 72: Abducción: modelo (izq.) vs valor ROM obtenido (135.4°).

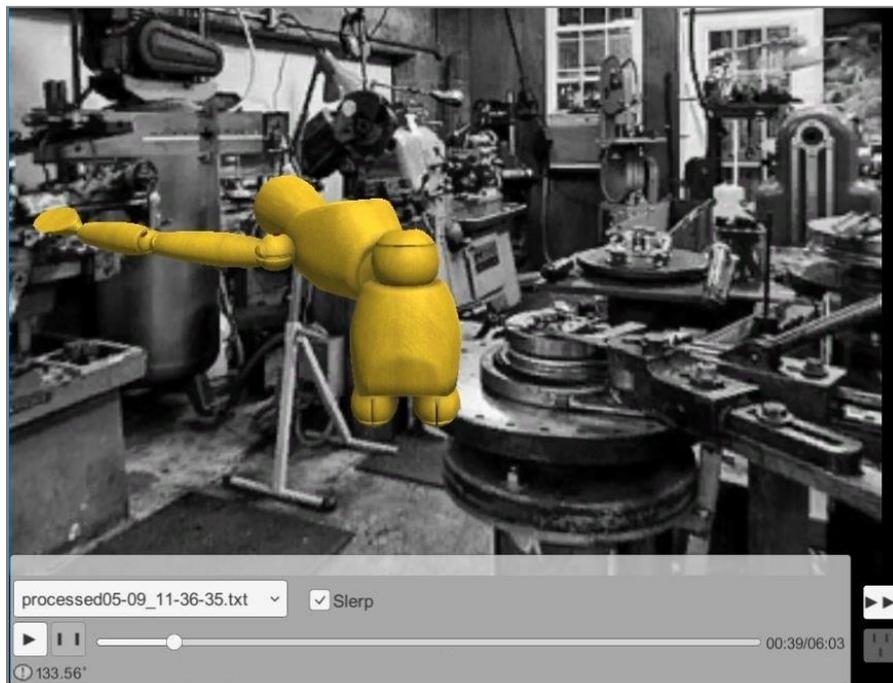


Figura 73: Abducción programa Unity3D en tiempo real: 132.56° obtenidos.

- No hemos realizado prueba de Aducción por ser un movimiento muy limitado en cuanto a ROM y difícil de evaluar con estas mismas herramientas, por el valor de ROM tan pequeño, lo que puede inducir a mayor error.

- Flexión

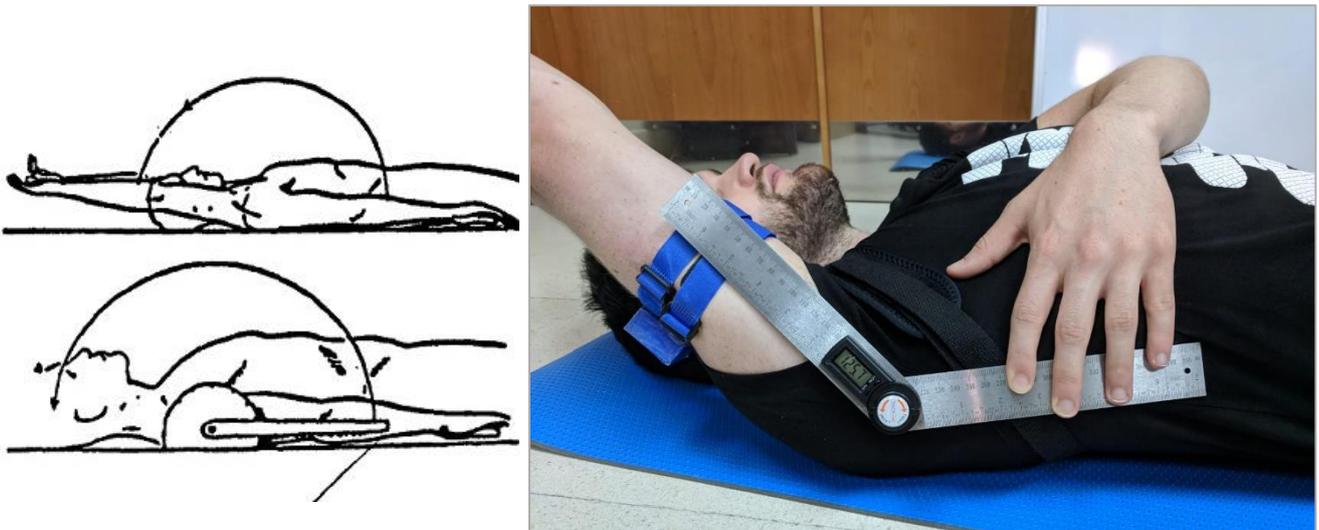


Figura 74: Flexión: modelo (izq.) vs valor ROM obtenido (125.7°).

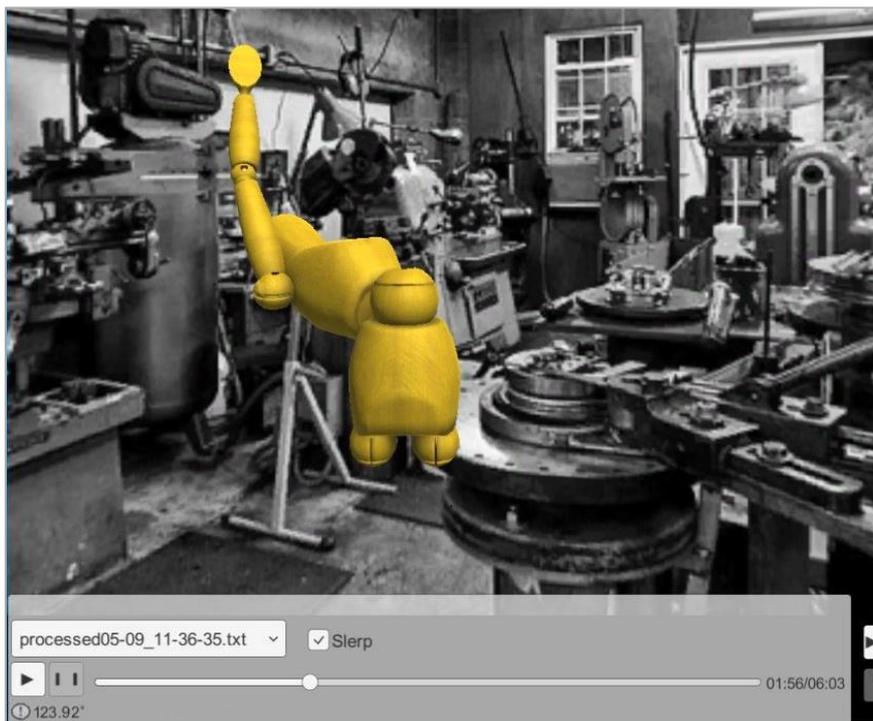


Figura 75: Flexión programa Unity3D en tiempo real: 123.92° obtenidos.

- Extensión

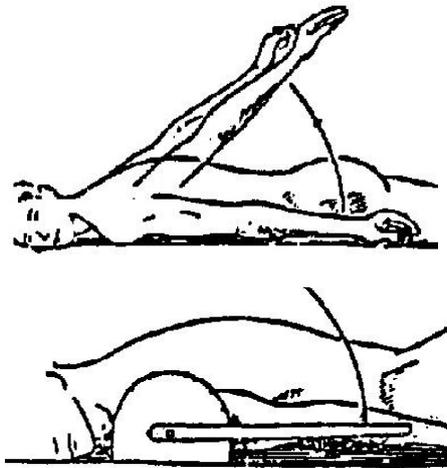


Figura 76: Extensión: modelo (izq.) vs valor ROM obtenido (24.4°).

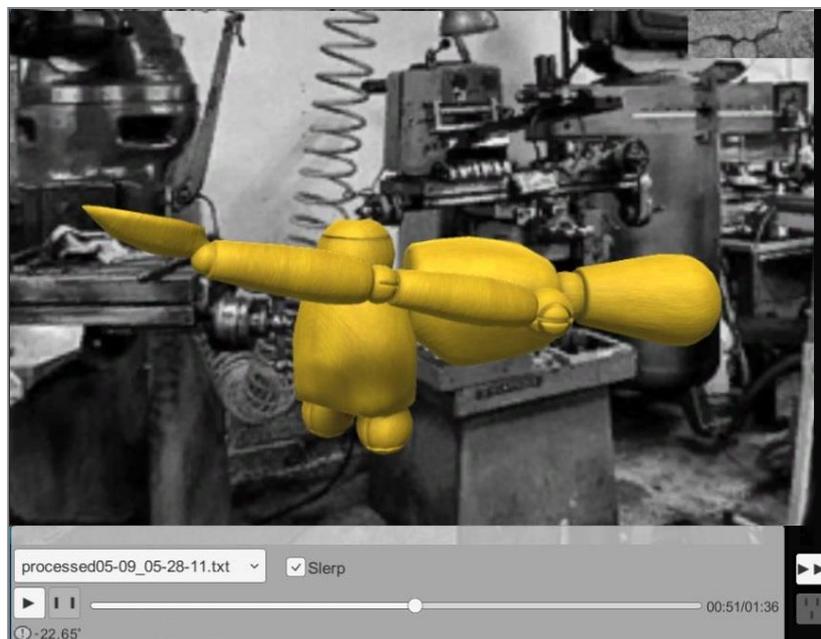


Figura 77: Extensión programa Unity3D en tiempo real: 22.65° obtenidos.
(El programa indica con el signo menos que estamos ante una extensión y no flexión)

- Rotación externa



Figura 78: Rotación externa: modelo (izq.) vs valor ROM obtenido (89.1°).

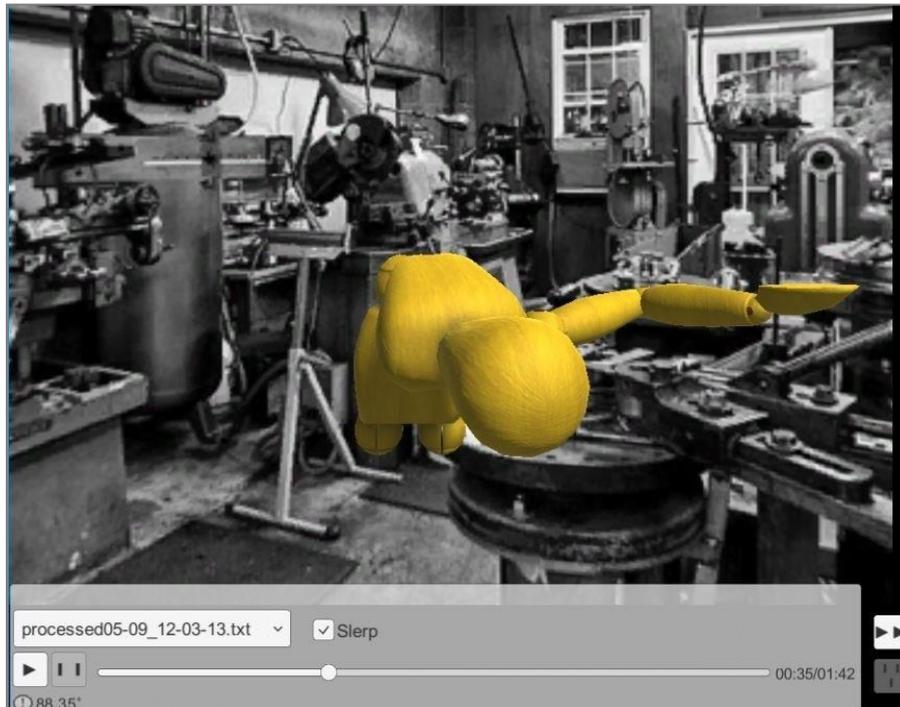


Figura 79: Rotación externa programa Unity3D en tiempo real: 88.35° obtenidos.

- **Rotación interna**

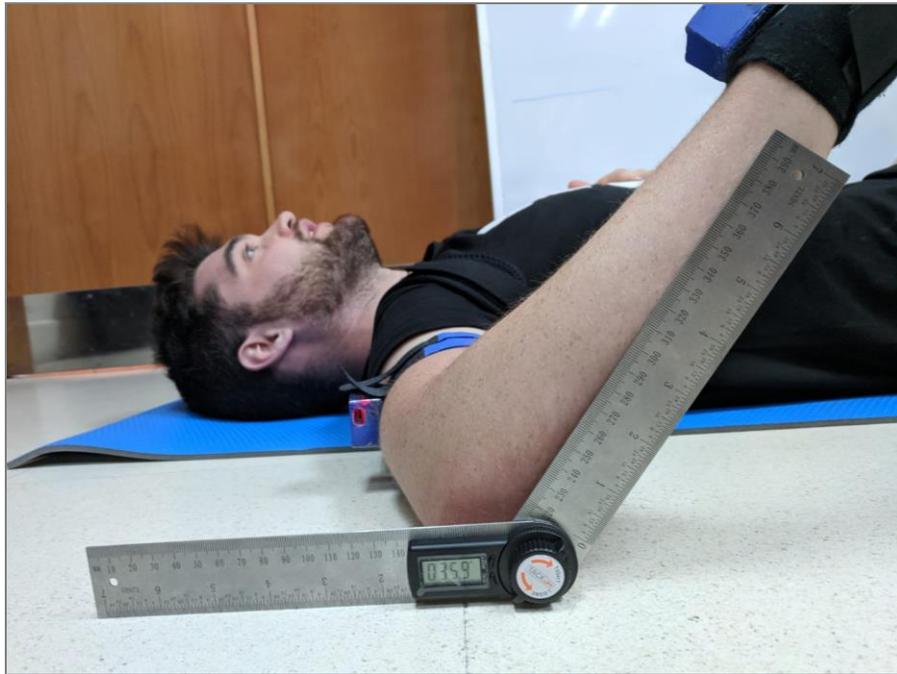


Figura 80: Rotación interna valor ROM obtenido (35.9°).

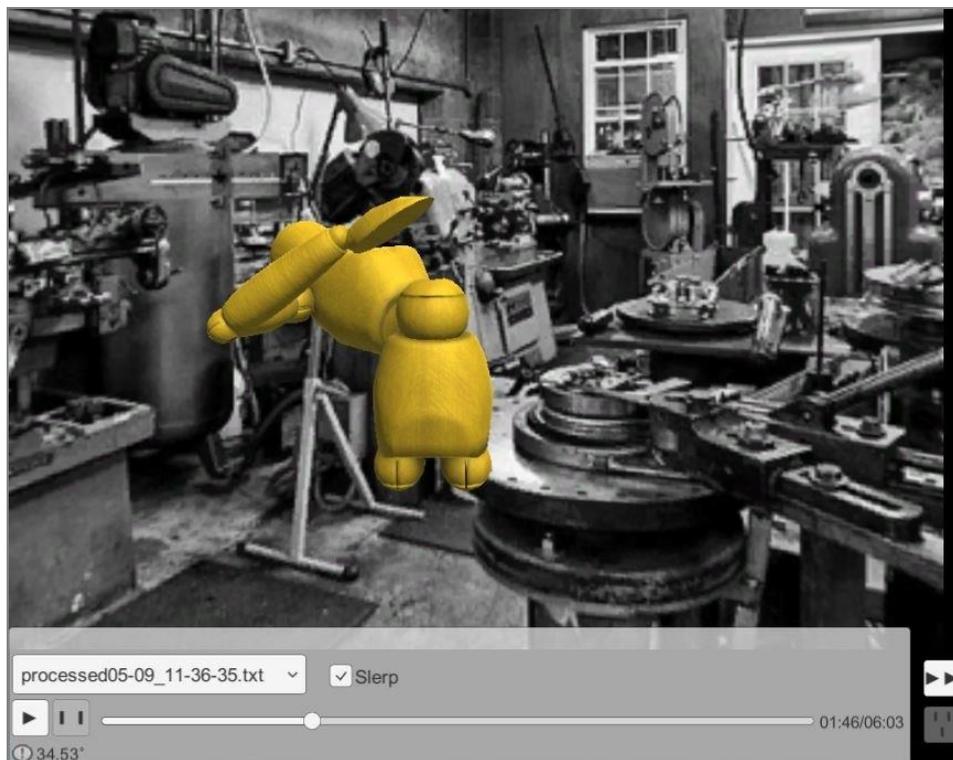


Figura 81: Rotación interna programa Unity3D en tiempo real: 34.53° obtenidos.

Veamos a continuación una **tabla resumen**:

		<i>Articulación Hombro</i>	
<i>Abducción</i>	Vídeo	135.46 °	<i>Rango articular máximo</i>
	Sensores	133.56°	
<i>Flexión</i>	Vídeo	125.7 °	<i>Rango articular máximo</i>
	Sensores	123.92 °	
<i>Extensión</i>	Vídeo	24.4 °	<i>Rango articular máximo</i>
	Sensores	22.65 °	
<i>Rotación externa</i>	Vídeo	89.1 °	<i>Rango articular máximo</i>
	Sensores	88.35 °	
<i>Rotación interna</i>	Vídeo	35.9 °	<i>Rango articular máximo</i>
	Sensores	34.53 °	

Tabla 5: Comparación cálculo ROM goniómetro digital vs Sensores

Pruebas herramienta de análisis ergonómico

Por último, para comparar nuestro sistema de evaluación a posteriori con las medidas reales, realizaremos movimientos en cada uno de los ejes que definen los ejercicios anteriores de flexo-extensión, abducción/aducción y rotación del hombro. Estos movimientos pretenderán, guiados por una plantilla goniométrica, acercarnos a determinar la fiabilidad del sistema y precisión de medidas ya que pararemos el movimiento en tramos de 10 o 15° (según el goniómetro) para corroborar que observamos saltos de 10/15° en la gráfica obtenida por el software de valoración ergonómico. Además, las variaciones de movimiento dentro de cada tramo (sin contar el salto) deben ser mínimas, por la alta precisión de los sensores reduciendo las alteraciones a la estabilidad de la posición del usuario.

Veamos primero la posición del usuario con el goniómetro a su espalda, para **abducción/aducción**. El resto de valores que no aparecen en las fotos, se analizarán directamente en la gráfica del software:



Figura 82: Abducción 0° (izq.) vs 45° (der.)



Figura 83: Abducción 90°(izq.) vs 110° (der.)

Veamos ahora la posición del usuario con el goniómetro a su espalda, para **flexión**:

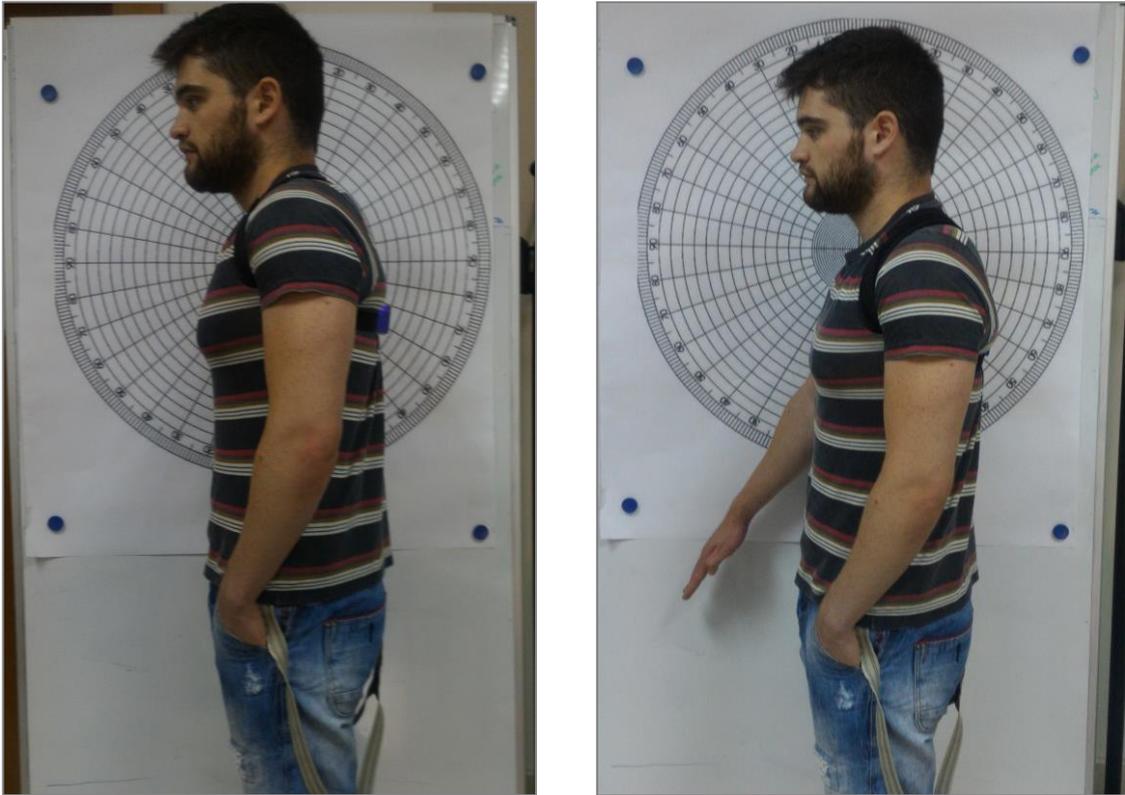


Figura 84: Flexión 0°(izq.) vs 20° (der.)

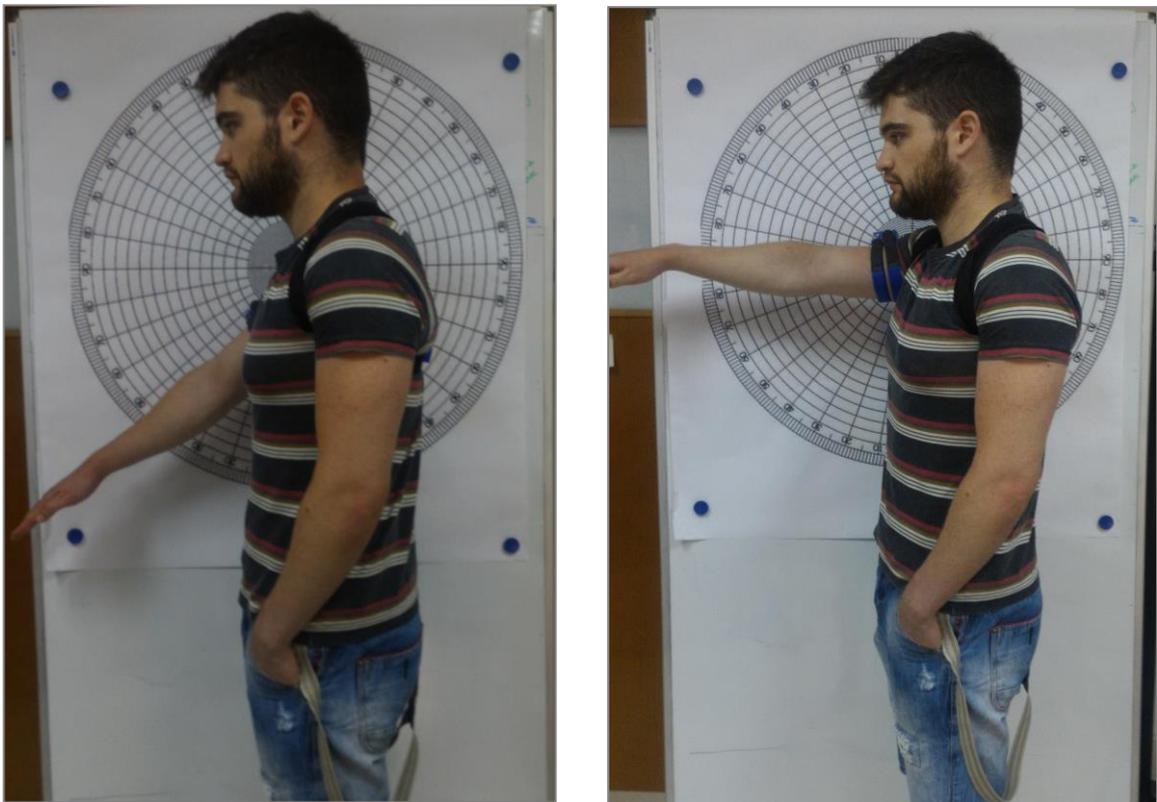


Figura 85: Flexión 45°(izq.) vs 90° (der.)



Figura 86: Flexión 110°(izq.)

Por último, veamos la posición del usuario con el goniómetro a su espalda, para la **rotación interior/exterior del hombro**:

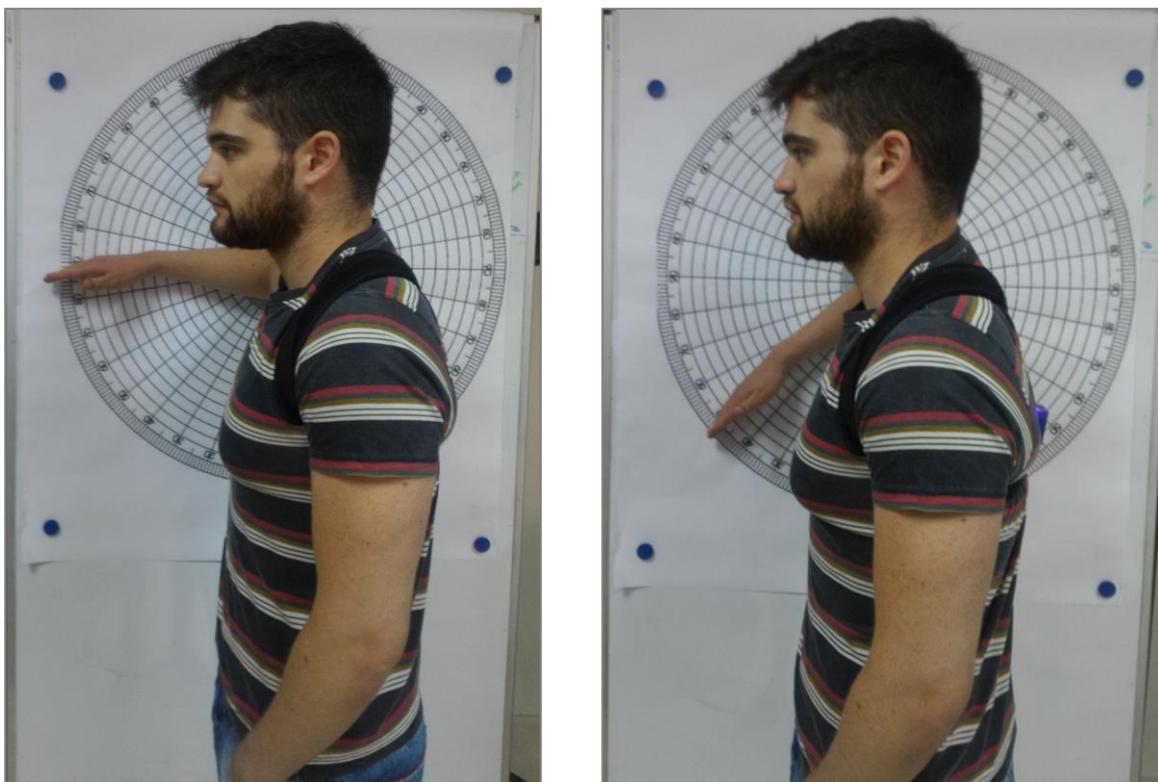


Figura 87: Rotación 0° (izq.) vs rotación interna 45° (der.)

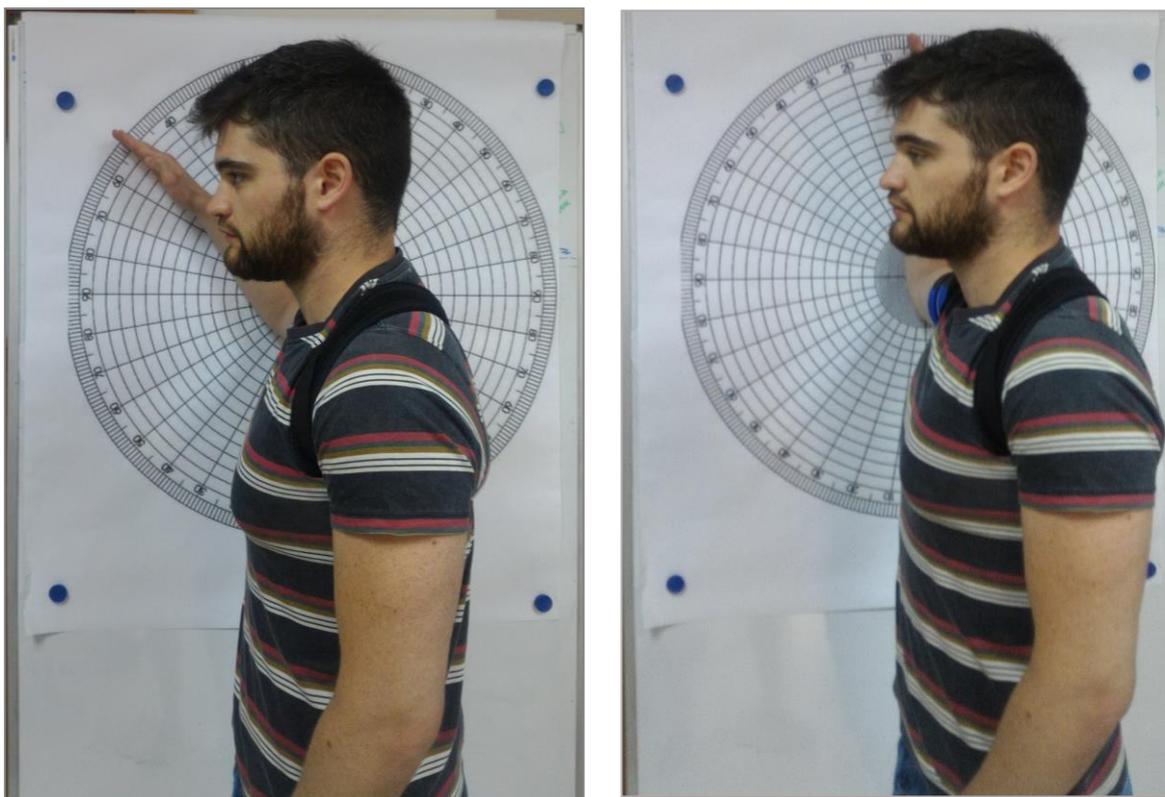


Figura 88: Rotación interna 45° (izq.) vs rotación externa 90° (der.)



Figura 89: Rotación interna 70° (no confundir el ángulo con el marcado por la mano, estamos midiendo la articulación del hombro).

A continuación, observemos la **gráfica** obtenida con la herramienta software de evaluación ergonómica, en la que podemos apreciar los distintos valores estables que hemos capturado para **abducción/aducción**:



Figura 90: Gráfica Abducción. Puntos señalados de izq. a der. 0°, 45° y 90°.

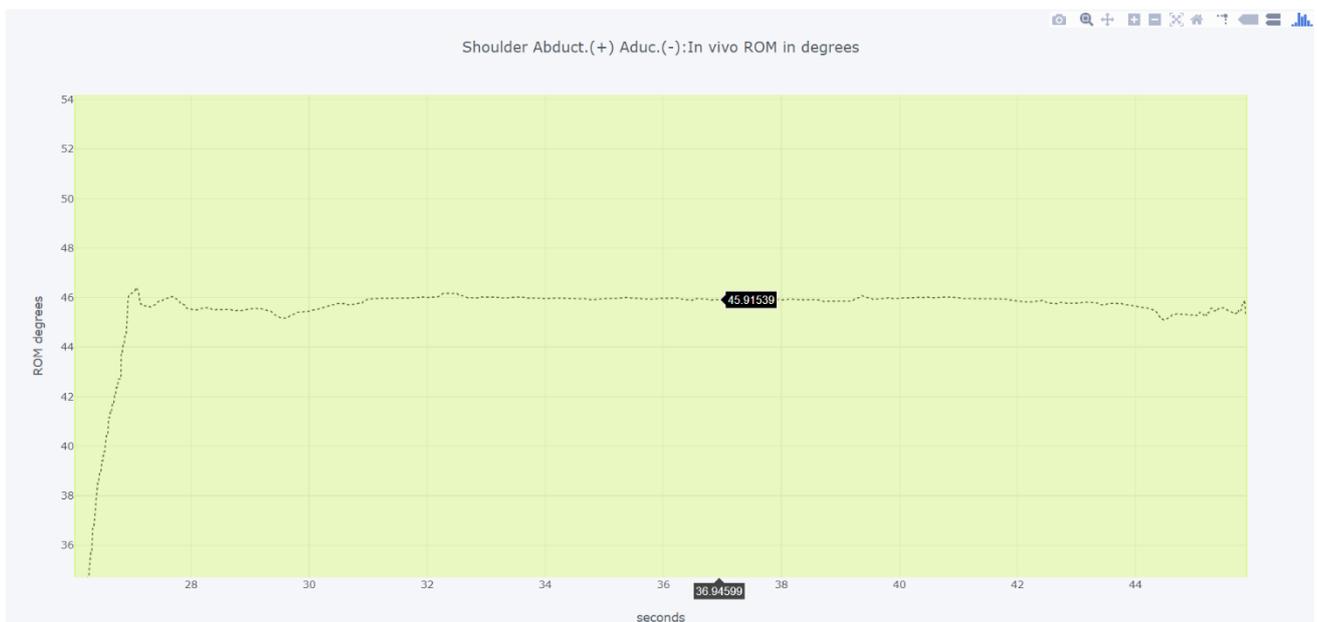


Figura 91: Zoom Gráfica Abducción. Punto señalado del valor ya estabilizado

Veamos ahora los resultados para **flexo-extensión**:



Figura 92: Gráfica Flexo-extensión. (Los valores iniciales “-10°” corresponden a extensión de hombro, el resto a flexión desde los “0°” en adelante)

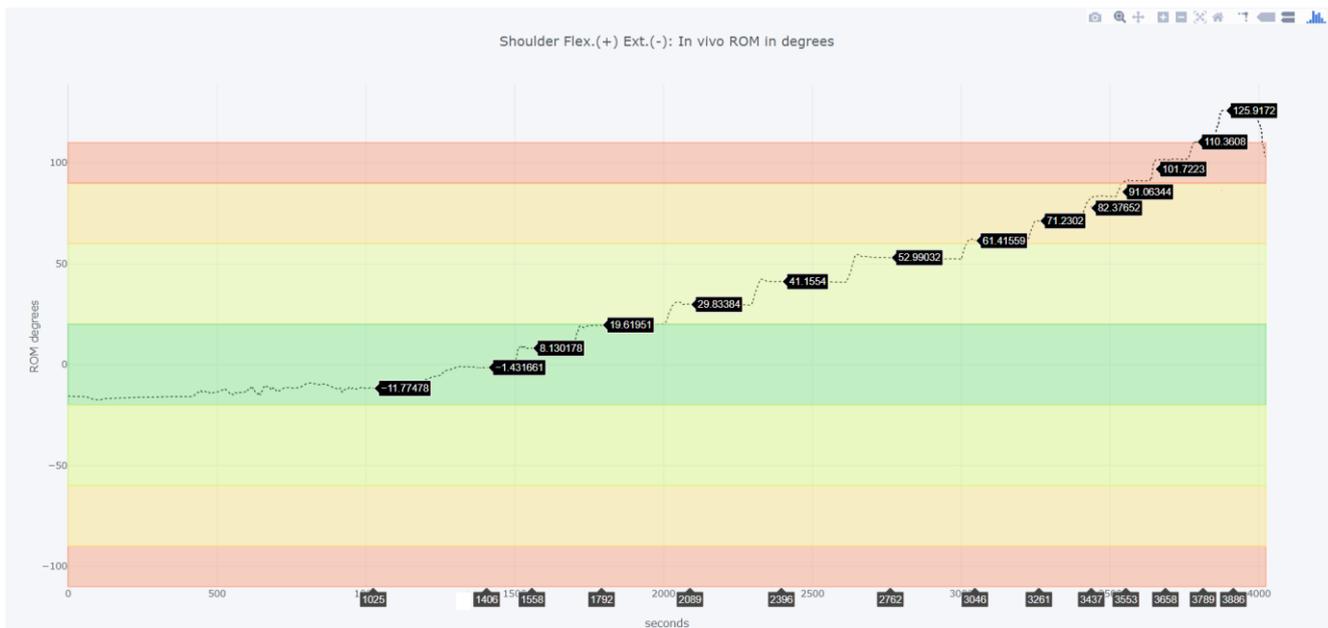


Figura 93: Gráfica Flexo-extensión con principales valores anotados.

Y, por último, los de **rotación externa e interna** del hombro:

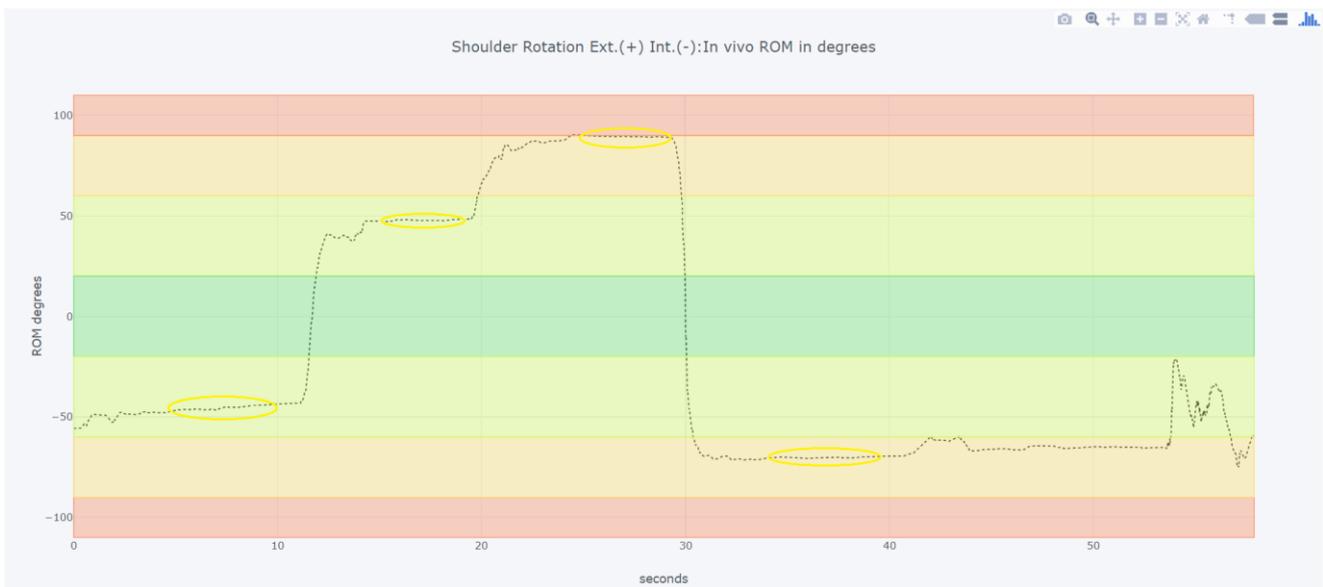


Figura 94: Gráfica Rotación. Puntos señalados de izq. a der. rotación interna 45°, externa 45°, externa 90° e interna 70°

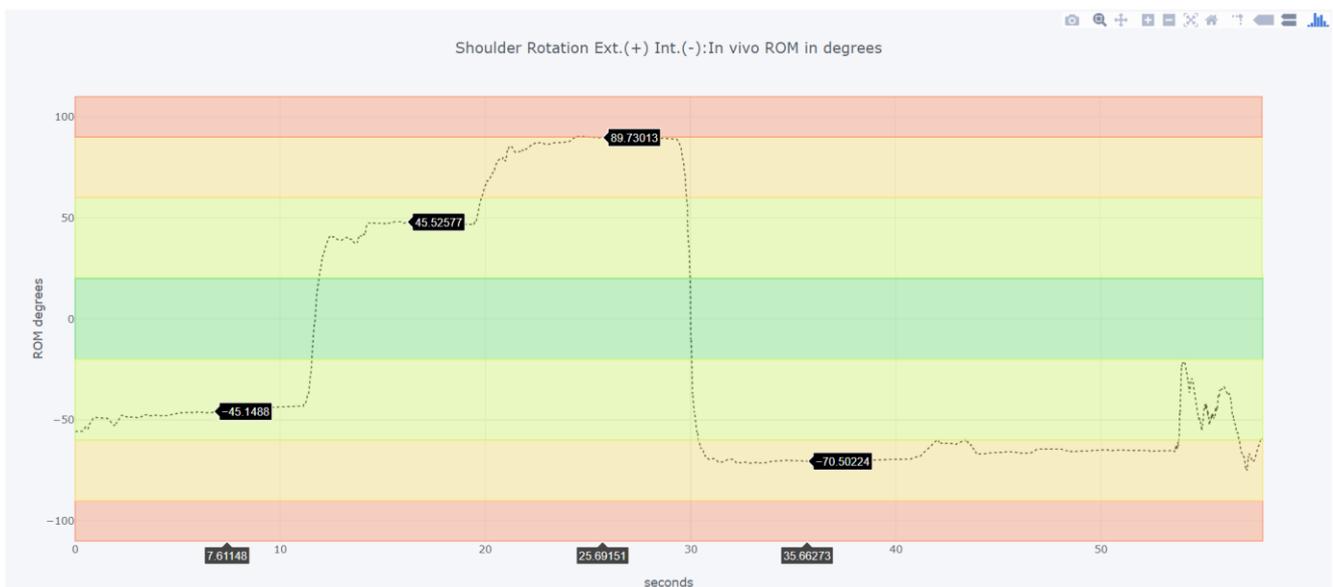


Figura 95: Zoom Gráfica Rotación. Punto señalado del valor ya estabilizado

Como podemos observar en ambos casos, los valores son bastante estables unos “ ± 0.2 grados” del valor medio únicamente sobrepasados por el movimiento del usuario. Además, hemos procurado parar el movimiento cada 10/15° para obtener un resultado claramente apreciable en las gráficas.

7.2 Discusión de los resultados

La batería de pruebas llevada a cabo ha servido para establecer si el sistema desarrollado cumple con los requisitos establecidos en la fase de arranque. Principalmente esta evaluación se corresponde con el análisis de la precisión del sistema a la hora de determinar con exactitud el rango de movimiento de una determinada articulación, en este caso, el hombro.

Del trabajo llevado a cabo previamente al desarrollo de las pruebas, de nuestra revisión sistemática, se ha obtenido un listado de artículos científicos que analizan la validez de sistemas similares a nuestro desarrollo. En una de las tablas de dicho artículo, se separaban los resultados por articulación, por lo que nos basaremos en ella para determinar y comparar los resultados previsibles de nuestro ensayo.

Body part	Joint	# of papers*	References
Upper limb (13)	Arm	5	(Cancela et al., 2014), (Crabolu et al., 2017), (El-Gohary & McNames, 2015), (Kirking et al., 2016), (Ploderer et al., 2016)
	Forearm	1	(El-Gohary & McNames, 2015)
	Shoulder	11	(Aslani et al., 2017), (El-Gohary & McNames, 2015), (Hellmers et al. 2017), (Khurelbaatar et al. 2015), (Kirking et al., 2016), (Lin et al., 2017), (Oarde et al. 2014), (Pereira et al. 2017), (Picerno et al., 2015), (Ploderer et al., 2016), (Robert-Lachaine et al. 2017)
	Elbow	9	(Crabolu et al., 2017), (El-Gohary & McNames, 2015), (Hellmers et al. 2017), (Khurelbaatar et al. 2015), (Lin et al., 2017), Oarde et al. (2014), (Oarde et al. 2014), (Pereira et al. 2017), (Ploderer et al., 2016), (Robert-Lachaine et al. 2017)
	Wrist	6	(El-Gohary & McNames, 2015), (Hellmers et al. 2017), (Khurelbaatar et al. 2015), (Lin et al., 2017), (Ploderer et al., 2016), (Robert-Lachaine et al. 2017)

**Note: Full-body systems like Xsens are considered as "+1" in # of papers category for each joint.*

*Tabla 6: Parte de la tabla de artículos separados por partes del cuerpo y articulación.
En: Wearable IMU-based systems for rehabilitation: a systematic review on reliability*

Podemos entonces basarnos en los trabajos de al menos 11 artículos científicos para determinar si nuestros resultados se encuentran dentro de lo esperable, y para apoyarnos a la hora de determinar la usabilidad del sistema para poder sustituir en un futuro a las soluciones basadas en cámaras como Kinect® (tanto en TELEKIN como en sistemas de valoración ergonómica actuales).

En relación con la validez de la tecnología adoptada, se debe tener en cuenta que todos los sistemas comerciales utilizados en los estudios anteriores tienen elementos de hardware que permiten determinar los parámetros de medición con gran precisión. Además, de los artículos escogidos, aquellos que utilizan sistemas no comerciales o ad hoc, deben escoger algoritmos de fusión para los datos de giro,

aceleración y magnetismo lo suficientemente buenos como para no acumular errores y obtener mediciones tan precisas como los sistemas comerciales.

En nuestro caso, cabe aclarar que nuestros sensores de movimiento BNO055 superan en gran parte de los artículos a los sensores inerciales escogidos para aquellos sistemas, contando con mayor precisión de medidas y estabilidad con el tiempo. Así pues, en los artículos anteriores, el chip de medición inercial más común es el MPU-9250 de invensense, con características muy similares al nuestro.

Aun así, hemos pasado a comparar los resultados obtenidos por nuestro pequeño estudio con los de alguno de los artículos, y puesto que no todos obtienen resultados de fiabilidad por medio de los mismos procedimientos, nos centraremos en aquellos fácilmente comparables a los nuestros. Estos son, los que comparan los resultados obtenidos con los golden-standard, es decir, sistemas de medición que se han considerado de validez en el campo de aplicación.

Uno de los casos es el de El-Gohary & McNames (2015), que comparaban sus sensores con los movimientos controlados de un brazo robot, obteniendo un error $\leq 3^\circ$ durante un movimiento lento de menos de $180^\circ/\text{sec}$ y un error máximo de 7.8° para el eje vertical. Kirking et al. (2016) también realiza una comparación con un brazo robot, y en este caso obtiene un RMSE de 2° para las actividades de la vida cotidiana o ADLs evaluadas.

Otro caso es el de Hellmers et al. (2017), que compara el traje de captura de movimientos de todo el cuerpo de Xsens con los golden-standard basados en cámaras. Con él consiguieron lograr un alto grado de precisión con una media de error en los ángulos de las articulaciones de $1.35^\circ \pm 2^\circ$. También se utiliza el sistema Xsens en Khurelbaatar et al. (2015), comparándolo con el Hawk system, un sistema de captura basado en cámaras que es considerado como golden-standard. El error cuadrático medio (RMSE) para todos los ejes fue inferior a $1,1^\circ$ (un único eje con rotación libre de 3 ejes).

En Robert-Lachaine et al. (2017) se evaluó la precisión de sistemas de sensores comerciales frente a un sistema optoelectrónico (el sistema Optotrak de 8 cámaras en este caso) como referencia en todas las articulaciones durante tareas manuales de manipulación de materiales de 32 min. Se obtuvieron para tareas largas y complejas una media en todas las articulaciones de $\leq 2.8^\circ$ de error en comparación con los $\leq 1.2^\circ$ registrados durante movimientos funcionales cortos.

Por último, comentaremos el caso de Picerno et al. (2015) que utiliza un dinamómetro isocinético comparado con un sistema de medida inercial comercial. En este caso, el error resultó menor en la estimación del ROM, siempre $< 1^\circ$ sin verse afectado por el número de repeticiones ni las velocidades angulares.

En nuestro caso, tanto la evaluación con sistemas basados en cámaras como por medio de goniometría, obtienen unos muy buenos resultados comparativos. Los casos en que hemos cuantificado este error, apartados ***Pruebas evaluación ROM en tiempo real*** y ***Pruebas herramienta de análisis ergonómico***, incluyen tanto el análisis in vivo: comparando con un goniómetro digital y obteniendo una media de

$\leq 1.51^\circ$ de error; como el análisis a posteriori mediante las gráficas del estudio ergonómico, con $\pm 0.2^\circ$ de estabilidad en las medidas. Este resultado es equiparable a los resultados de los artículos referenciados.

Por otra parte, frente a los sistemas basados en cámara, en este caso comparando con el sistema Kinect® que en otros artículos ha sido utilizado como golden-standard como en Hossein Mousavi Hondori & Maryam Khademi (2014); un sistema como el nuestro presenta ventajas de usabilidad tan grandes que permiten la ampliación de las tecnologías de captura de movimientos inalámbricas al mundo de la valoración ergonómica industrial. Esto es así, porque eliminamos la problemática del espacio, el campo de visión, las oclusiones o auto-occlusiones y las posiciones indetectables. Aun así, hay que incluir como efectos colaterales de utilizar estas tecnologías, un rango de alcance limitado y la necesidad de evitar grandes campos magnéticos para el correcto funcionamiento de los sensores.

En conclusión, podemos decir que nuestro sistema responde satisfactoriamente a todos los requisitos planteados al inicio de este Trabajo Fin de Máster, y que se han evaluado y comparado con publicaciones científicas nuestros resultados, obteniendo unos resultados muy similares a los ya conocidos por otros sistemas que han comenzado su implantación como sistemas comerciales.

7.3 Presupuesto

La inversión necesaria para desarrollar un hardware y aplicación de estas características se puede dividir en: gastos de equipamiento, gastos en personal, y gastos de fabricación del dispositivo en sí. En este capítulo, se pretende determinar el presupuesto de lo que cuesta diseñar nuestro dispositivo. Como se busca que el producto sea comercializado en posteriores versiones, se buscará conseguir un precio razonable para la venta al cliente. Vendiendo la mayor cantidad posible, el precio de desarrollo del ingeniero se repartirá entre todos los dispositivos.

En la tabla siguiente desglosamos y detallamos esta inversión. El gasto en personal lo calcularemos a partir del tiempo empleado en desarrollar la aplicación y construir los prototipos, obviando la etapa de aprendizaje del software de desarrollo.

Horas	Precio por hora (\€)	Total (€)	Total por Prototipo (€)
505	15	7575	2525

Tabla 7: Presupuesto ingeniero.

El presupuesto en personal se dividirá entre 3, siendo el coste de un investigador en formación durante 4 meses, ya que se repartirá entre los 3 prototipos.

7.3.1 Dispositivo Hardware V1: versión cableada

GASTOS Prototipo V1	€/Unidad	Unidades	Subtotal (€)
GASTOS DE EQUIPAMIENTO			400,00€
Portatil de uso amortizado	400,00€	1	400,00€
GASTOS Prototipo 1			106,19€
FT232RL TTL to USB Serial Converter	7,00€	1	7,00€
Arduino Pro Mini 3V	9,90€	1	9,90€
BNO 055 (CJMCU)	10,00€	1	10,00€
MPU 6050	6,99€	2	6,99€
TCA9548A I2C mux	10,87€	1	10,87€
Cable ethernet plano	8,99€	1	8,99€
Interruptor	1€	1	1€
Batería de litio 3.7V – 500mAh (Adafruit)	15,00€	1	15,00€
Adafruit LiPoly Backpack Add-On	9,45€	1	9,45€
Encapsulado (+ cableado y soldadura)	20,00€	1	20,00€
PERSONAL			2525,00€
Investigador con Seguridad Social	2525,00€	1	2525,00€
SUBTOTAL			3031,19€
GASTOS INDIRECTOS (15%)			454,68€
IMPORTE TOTAL			3485,87€

Tabla 8: Presupuesto Prototipo V1.

7.3.2 Dispositivo Hardware V2: versión BT autónoma

GASTOS Prototipo V2	€/Unidad	Unidades	Subtotal (€)
GASTOS DE EQUIPAMIENTO			400,00€
Portatil de uso amortizado	400,00€	1	400,00€
GASTOS Prototipo 2			72,68€
Arduino Pro Trinket 3V	15,23€	1	15,23€
Módulo Bluetooth HC-05	7,00€	1	7,00€
BNO 055 (CJMCU)	10,00€	1	10,00€
Interruptor	1€	1	1€
Batería de litio 3.7V – 500mAh (Adafruit)	15,00€	1	15,00€
Adafruit LiPoly Backpack Add-On	9,45€	1	9,45€
Encapsulado (+ cableado y soldadura)	15,00€	1	15,00€
PERSONAL			2525,00€
Investigador con Seguridad Social	2525,00€	1	2525,00€
SUBTOTAL			2997,68€
GASTOS INDIRECTOS (15%)			449,65€
IMPORTE TOTAL			3447,33€

Tabla 9: Presupuesto Prototipo V2.

En este caso, el prototipo deberá contar con una batería de 500mAh para que tenga una vida útil mínima. Esto es debido a que los sistemas Arduino son muy pobres en optimización del consumo, y a diferencia de lo que veremos en el prototipo final, el consumo es tan elevado que 500mAh resulta el mínimo necesario para obtener una cierta autonomía.

7.3.3 Dispositivo Hardware Definitivo (ESP32)

GASTOS Prototipo Final	€/Unidad	Unidades	Subtotal (€)
GASTOS DE EQUIPAMIENTO			400,00€
Portatil de uso amortizado	400,00€	1	400,00€
GASTOS Prototipo Final		1	58,30€
Placa ESP32	16,00€	1	16,00€
BNO 055 (CJMCU)	10,00€	1	10,00€
Wemos Battery shield	3€	1	3€
Batería de litio 3.7V - 350mA	8,00€	1	8,00€
Pulsador de membrana	0,30€	1	0,30€
Interruptor	1€	1	1€
Encapsulado (+ cableado y soldadura)	20,00€	1	20,00€
PERSONAL			2525,00€
Investigador con Seguridad Social	2525,00€	1	2525,00€
SUBTOTAL			2983,3€
GASTOS INDIRECTOS (15%)			447,50€
IMPORTE TOTAL			3430, 80€

Tabla 10: Presupuesto Prototipo final.

El presupuesto indicado en la tabla de arriba, se refiere a una única unidad sensora, capaz de colocarse en una única articulación cada vez. Para obtener el presupuesto total del sistema, se deberá multiplicar los **GASTOS Prototipo Final** por el número de articulaciones a medir. El número mínimo de sensores será de 2 unidades, ya que normalmente para obtener una medida precisa del movimiento de una articulación se necesitarán de dos puntos de referencia, salvo por ejemplo en el caso de la espalda, que se podrá presuponer recta en un primer momento de calibración del sensor, aunque se recomienda colocar otra unidad por ejemplo en la cadera, para obtener una medida realista y precisa del movimiento.

Como un ejemplo del número de sensores a utilizar por aplicación, comentaremos el caso del miembro superior, utilizado en los test de evaluación, que consta de hombro, codo y muñeca, por ejemplo. Así pues, colocaremos una unidad en la espalda (como referencia), otra en la parte superior del brazo, otra en el antebrazo y una última en la mano, para obtener estas articulaciones al comparar los datos de los sensores 2 a 2.

Capítulo 8 Conclusiones y líneas futuras

Trabajar en líneas de investigación relacionadas con la tecnología de más reciente aparición puede ser uno de los acercamientos a la investigación más aventurada y arriesgada, pero sin duda, puede ser de gran interés tanto para el investigador, como para los futuros receptores de los frutos de dicho trabajo.

Los nuevos avances tecnológicos tienden a incorporarse paulatinamente, ya sea la mejora de infraestructuras para la industria de manufactura (industria 4.0) como la incorporación de nuevo instrumental, o cambio en los métodos actuales de recopilación e interpretación de datos biomédicos en el ámbito de la rehabilitación médica.

Tener la posibilidad de desarrollar varios dispositivos para aplicación en campos tan relevantes como la rehabilitación médica y la valoración ergonómica industrial, constituye un trabajo de gran interés y de gran valor investigador.

Este Trabajo Fin de Master propone el uso de las nuevas tecnologías de sensorización y comunicación inalámbrica en chip, como posible solución a los sistemas actuales de captura de movimiento basados en cámaras. Esto constituye un aporte más al creciente mundo de la sensorización inalámbrica y los datos biomédicos comúnmente recogidos por cualquier dispositivo vestible presente en nuestra vida diaria (como los smartwatches, dispositivos Garmin, etc.),

8.1 Conclusiones

En este trabajo nos hemos centrado en el desarrollo tanto de un sistema Hardware, como Software, a partir de sensores de medición inercial y sistemas en un chip comerciales, fabricando nuestra propia solución para incorporar a un sistema de rehabilitación y también de evaluación ergonómica.

En primer lugar, se analizó el estado de la técnica, mediante estudios previos en ambos sectores, y se justificó finalmente la necesidad del sistema. Con el fin de apoyar esta justificación se llevó a cabo también una revisión sistemática sobre el tema: *“Wearable IMU based systems for rehabilitation: a systematic review on reliability”*.

A continuación, se detalló la propuesta de sistema de rehabilitación, que pretende complementar al sistema actual del Grupo de Telemática e Imagen de la Universidad de Valladolid para rehabilitación TELEKIN. Para ello se plantearon tanto las metas a alcanzar con este trabajo como los próximos hitos a desarrollar en el futuro.

Respecto a la evaluación ergonómica, se expusieron los principales riesgos causados por el incorrecto diseño de los puestos de trabajo, según algunas de las principales guías utilizadas por el *Instituto Nacional de Seguridad e Higiene en el Trabajo*, así como el modelo de integración de nuestro sistema en los métodos de evaluación

ergonómica de puestos de trabajo y la forma de la aplicación encargada de dicha tarea. También se estudió el actual paradigma de valoración ergonómica en la industria de manufactura, concluyendo que el método utilizado en general se ha observado pobre y desactualizado, siendo muy necesaria la introducción de las nuevas tecnologías en dichos procedimientos.

Una vez se presentó el estado de ambos ámbitos de aplicación de nuestro proyecto, y la forma de incorporar nuestro sistema a ellos, se comenzó a describir la parte Hardware del sistema vestible desarrollado. Este desarrollo se dividió en distintos prototipos que contaban con características distintas, la V1: versión cableada, la V2: versión BT autónoma y la definitiva basada en el ESP32.

La primera versión (V1), está basada en arduino y cuenta con un módulo Bluetooth para comunicación con el PC o dispositivo móvil, aunque los distintos sensores se comunican entre sí unidos por cable mediante el protocolo I2C.

La segunda aproximación (V2), está también basada en arduino pero pretende monitorizar únicamente una articulación, contando para ello con una pequeña batería y centrándose en reducir drásticamente su tamaño frente al prototipo V1.

Por último, el dispositivo final, combina las ideas del tamaño reducido y la multiplicidad de sensores, para conseguir una solución totalmente inalámbrica, basada en el procesador ESP32 de *espressif*, que permite la incorporación de hasta 8 puntos corporales distintos.

Tras estudiar las distintas tecnologías que podremos introducir en nuestros dispositivos Hardware, observamos que se trata de un mundo en continuo crecimiento y expansión, y que la facilidad de obtención de los elementos Hardware, así como su reducido precio, presentan una línea de futuro a tener en cuenta en los proyectos "*Do It Yourself*" frente a las soluciones comerciales ya existentes. De hecho, el SoC elegido en este caso (ESP32) acaba de comenzar su andadura en el mundo de los sistemas todo en un chip pensados para el IoT o internet de las cosas.

Aunque todos estos dispositivos necesitarán, además, la definición de unos protocolos de comunicación tanto inter- como intra- dispositivo, que se detallaron a continuación de la elección del Hardware, junto con la explicación de distintas interfaces de los sistemas de rehabilitación y valoración ergonómica. A este respecto, hemos constatado que las distintas tecnologías incorporadas son cada vez más apreciadas en cualquier campo de la electrónica y la ingeniería de telecomunicaciones, tomando un papel principal la sensorización y comunicación Bluetooth o WiFi en cualquier dispositivo electrónico presente en todos los aspectos de nuestra vida cotidiana.

Una vez que se definió completamente el Hardware y el Software, se prosiguió con la planificación de una serie de pruebas que permitieran demostrar que el sistema desarrollado constituía un sistema vestible funcional, ampliable en número de articulaciones y con un software que lo apoyara tanto en la tarea de rehabilitación sensorizada como de valoración ergonómica de puestos de trabajo. No sin antes introducir los conceptos teóricos que apoyaron el establecimiento de la batería de pruebas y la metodología de las mismas.

A continuación, se procedió con las pruebas, en las que se analizaron los resultados tanto para el dispositivo *V1: versión cableada* como para el *dispositivo Hardware definitivo basado en el ESP32*. No fue así con la versión *V2: BT autónoma*, de la que simplemente se comprobó su compatibilidad con el sistema TELEKIN.

Centrándonos en el sistema final, obtuvimos buenos resultados, sobre todo comparándolo con los sistemas actualmente utilizados para rehabilitación sensorizada (en este caso basados en cámaras) y valoración ergonómica (grabaciones de video revisadas por expertos). Estos resultados, junto con los apuntados en la revisión sistemática realizada en el transcurso de este Trabajo Fin de Máster, fueron buenos indicadores de la posible futura incorporación de sistemas basados en unidades de medida inercial para rehabilitación y medicina.

Además, con el desglose de los distintos presupuestos por versión, se ha permitido apreciar que la inversión necesaria para proyectos de este tipo es principalmente de investigación y tiempo invertido, y no tanto de gastos de equipamiento o fabricación.

Con todo esto, podemos decir que este proyecto ha permitido afianzar y desarrollar tanto competencias genéricas de ingeniería, llevando a cabo desde el diseño del producto hasta su construcción final y desarrollo de sus aplicaciones; como específicas del Máster de Investigación, realizando una tarea de documentación y revisión de artículos científicos como base para la justificación del proyecto.

Finalmente, recordar que se ha buscado utilizar diversas tecnologías incluyendo dispositivos de reciente aparición en el mercado como es el SoC (sistema en un chip) ESP32. Esta novedad también implicó basar el desarrollo en un sistema todavía sin explotar, lo que conllevó un amplio trabajo de documentación previa y comprensión del sistema.

8.2 Objetivos alcanzados

En el presente Trabajo Fin de Máster, se han conseguido alcanzar los objetivos tanto globales como parciales establecidos en el punto **1.2 Objetivos**. Ya que hemos comprobado en las pruebas que, según los resultados obtenidos y los datos comparativos con nuestros sistemas de medida, el prototipo final responde correctamente a sus requisitos iniciales.

En primer lugar, se ha conseguido la redacción de un artículo científico (pendiente de publicación), en el cual se realiza una revisión sistemática de las aplicaciones de sistemas vestibles de sensores en el ámbito de la rehabilitación médica.

Además, se adquirió la información y documentación que fue necesaria previamente al proceso del desarrollo de los prototipos, en base a la información aportada por los distintos actores participantes en cada ámbito como son profesionales médicos o trabajadores de la industria de manufactura.

También se desarrolló una labor de búsqueda y concreción para escoger entre distintos dispositivos que cumplieran con las funciones deseadas, adaptándonos a las necesidades definidas en el proyecto.

En cuanto al propio desarrollo, una vez determinados los elementos base, se determinaron las características concretas de cada prototipo a fabricar, dividiéndolo en tres prototipos Hardware distintos con distintas configuraciones, incluso utilizando un microprocesador distinto en el prototipo final respecto a las versiones V1 y V2. El desarrollo de los prototipos en sí mismo fue exitoso, sobre todo en cuanto al prototipo final, que agrupó todas las características deseables en un producto de sensorización vestibular totalmente funcional.

Por último, se consiguió la integración del proyecto, consiguiendo que los dispositivos ayudasen tanto en la rehabilitación sensorizada de pacientes como en la valoración ergonómica eficaz de puestos de trabajo de nuestra industria. Para ello hemos abarcado desde la programación de microprocesadores (tanto de un sistema basado en Arduino como en un SoC de *espressif*), el desarrollo de una aplicación software de análisis en Python con Jupyter Notebook, o incluso de scripts para Unity3D en C# incorporables al sistema TELEKIN.

Con ello, hemos conseguido resolver todas las necesidades desde las inicialmente planteadas, hasta las que han ido surgiendo en la etapa de desarrollo. Este tipo de proyectos, constituyen a su vez una aplicación directa tanto de los conocimientos adquiridos durante el grado en Telecomunicaciones y mención de “Sistemas Electrónicos” como de las competencias del Master de Investigación en Sistemas de la Información y Comunicación, sobre todo en el proceso de revisión sistemática previa como en la planificación y desempeño de la batería de pruebas.

8.3 Líneas futuras

Tras el cumplimiento de los requerimientos principales del Trabajo de Fin de Máster se abre una nueva línea de trabajo con respecto a la mejora de los prototipos.

Una de las **posibilidades** es la incorporación de nuevos dispositivos sensores al prototipo final basado en el ESP32. Se podrán añadir sensores mioeléctricos a algunas extremidades, como por ejemplo brazos o piernas. Este tipo de sensores miden, filtran, rectifican y amplifican la actividad eléctrica de un músculo y producen una señal análoga que puede ser fácilmente leída por un microcontrolador haciendo que podamos desarrollar un dispositivo sensible a gestos (siendo una de las líneas principales de investigación en este ámbito hoy en día, el *Gesture Control* o control gestual de aplicaciones).

También podríamos sustituir la comunicación dispositivo-PC por una comunicación *bluetooth LE* en el caso de que las pruebas resulten satisfactorias y el dispositivo se preste a ello. Teniendo en cuenta que para ello deberemos enviar la información a un menor bitrate, seguramente condensando la información de las tramas.

El resto de modificaciones serán orientadas al software, incorporando las nuevas funcionalidades al simulador o al sistema TELEKIN y buscando nuevas formas de aprovechar el Hardware ya presente en el dispositivo para nuevas aplicaciones.

Además, aprovechando que los sistemas de captura de movimientos basados en cámaras obtienen generalmente peores resultados que los que utilizan sensorización mediante vestibles, como en nuestro caso, se podrían intentar sustituir algunos sistemas comerciales pensados para otras aplicaciones. Además, se conseguían solucionar una serie de problemas ligados a la tecnología utilizada para éstas.

Respecto al ámbito de la valoración ergonómica en la industria, tras nuestro paso por diversos entornos industriales de manufactura, hemos constatado cómo los métodos actualmente utilizados para esta valoración son cuanto menos limitados, lo cual podría justificar el aportar nuestra solución a la aplicación de las nuevas tecnologías (industria 4.0) al ámbito más olvidado de estas industrias.

REFERENCIAS

- Aslani, N., Noroozi, S., Davenport, P., Hartley, R., Dupac, M., & Sewell, P. (2017). Development of a 3D workspace shoulder assessment tool incorporating electromyography and an inertial measurement unit—a preliminary study. *Medical & biological engineering & computing*, 1–9.
- Asociación de Especialistas en Prevención y Salud Laboral, (2017) Estudio epidemiológico de EEPP reconocidas en España de 1990 a 2014.
- Borenstein G., (2012) “Making Things See: 3D vision with Kinect®, Processing, Arduino, and MakerBot”. MakeMedia. Make: Books.
- Caminero, R. V., Méndez-Herrero, L. A., Díaz-Pernas, F. J., Calabia del Campo, J., Rodríguez, M. A., & Martínez-Zarzuela, M. (2017). Telekin - Tele-rehabilitation System for Musculoskeletal and Cognitive Disorders using Natural Movement Interface Devices: En *Proceedings of the 3rd International Conference on Information and Communication Technologies for Ageing Well and e-Health* (pp. 198-203). Porto, Portugal: SCITEPRESS - Science and Technology Publications.
- Cancela, J., Pastorino, M., Arredondo, M. T., Nikita, K. S., Villagra, F., & Pastor, M. A. (2014). Feasibility study of a wearable system based on a wireless body area network for gait assessment in Parkinson’s disease patients. *Sensors*, 14(3), 4618–4633.
- CEPROSS, PANOTRATSS, (2012). Observatorio de enfermedades profesionales (CEPROSS) y de enfermedades causadas o agravadas por el trabajo (PANOTRATSS): “Informe anual de 2012”.
- Clara Díaz Aramburu Departamento de Investigación e Información (INSSBT), (2017). Accidentes de trabajo por sobreesfuerzos. 2016. Instituto Nacional de Seguridad, Salud y Bienestar en el Trabajo (INSSBT), O.A., M.P.
- Crabolu, M., Pani, D., Raffo, L., Conti, M., Crivelli, P., & Cereatti, A. (2017). In vivo estimation of the shoulder joint center of rotation using magneto-inertial sensors: MRI-based accuracy and repeatability assessment. *Biomedical engineering online*, 16(1), 34.
- El-Gohary, M., & McNames, J. (2015). Human joint angle estimation with inertial sensors and validation with a robot arm. *IEEE Transactions on Biomedical Engineering*, 62(7), 1759–1767.
- Espressif Systems. (2018, V3). ESP32 Technical Reference Manual.
- European Environment Agency, (2015) "Global megatrends assessment".
- Hellmers, S., Fudickar, S., Lange, E., Lins, C., & Hein, A. (2017). Validation of a motion capture suit for clinical gait analysis. En *Proceedings of the 11th EAI International Conference on Pervasive Computing Technologies for Healthcare* (pp. 120–126). ACM.

- Hossein Mousavi Hondori, & Maryam Khademi. (2014). A Review on Technical and Clinical Impact of Microsoft Kinect® on Physical Therapy and Rehabilitation. *Hindawi Journal of Medical Engineering*.
- Instituto Nacional de Estadística (2016). Estado de salud (estado de salud percibido, enfermedades crónicas, calidad de vida, dependencia funcional).
- Instituto Nacional de Estadística (2016). Tablas Estadística de Accidentes de Trabajo.
- Instituto Nacional de Seguridad e Higiene en el Trabajo, (2003). Manual para la evaluación y prevención de riesgos ergonómicos y psicosociales en la PYME.
- Instituto Nacional de Seguridad e Higiene en el Trabajo. (2004). Evaluación de la carga postural: método de la Universidad de Lovaina; método LUBA. Instituto Nacional de Seguridad e Higiene en el Trabajo.
- Instituto Nacional de Seguridad e Higiene en el Trabajo [online]. Factores de riesgo del trabajo repetitivo. Instituto Nacional de Seguridad e Higiene en el Trabajo. Fecha de última consulta: Junio 2018.
- International Labour Office, & International Ergonomics Association (2000). Lista de comprobación ergonómica: ergonomic checkpoints: soluciones prácticas y de sencilla aplicación para mejorar la seguridad, la salud y las condiciones de trabajo. Ginebra: OIT.
- Hernández Mangas, J., (2014). “Microcontroladores y DSPs”. Universidad de Valladolid.
- Jain, P., “Position sensor: Types of position sensor” [online]. Disponible en: <https://www.engineersgarage.com/> Fecha de última consulta: Diciembre 2017.
- Khurelbaatar, T., Kim, K., Lee, S., & Kim, Y. H. (2015). Consistent accuracy in whole-body joint kinetics during gait using wearable inertial motion sensors and in-shoe pressure sensors. *Gait & posture*, 42(1), 65–69.
- Kirking, B., El-Gohary, M., & Kwon, Y. (2016). The feasibility of shoulder motion tracking during activities of daily living using inertial measurement units. *Gait & posture*, 49, 47–53.
- Kitchenham, B. (2004). Procedures for performing systematic reviews. *Keele, UK, Keele University*, 33(2004), 1–26.
- Kniberg H. & Skarin M., (2010) “Kanban vs. Scrum”. C4Media.
- Lin, L. F., Lin, Y. J., Lin, Z. H., Chuang, L. Y., Hsu, W. C., & Lin, Y. H. (2017). Feasibility and efficacy of wearable devices for upper limb rehabilitation in patients with chronic stroke: a randomized controlled pilot study. *European journal of physical and rehabilitation medicine*.
- Markelov, A. (2002). Uso de Cuaterniones para Representar Rotaciones. *Tecnología Hoy*, 6(1), 21-22.
- Microsoft, [online]. Kinect® For Developers, “Características Kinect® 2.0 (2013)”. www.Kinectfordevelopers.com. Fecha de última consulta: Junio 2018.

- Mousavi Hondori H., & Khademi M., (2014). A Review on Technical and Clinical Impact of Microsoft Kinect® on Physical Therapy and Rehabilitation. *Hindawi Journal of Medical Engineering*, 2014.
- Mulrow, C. D. (1994). Systematic reviews: rationale for systematic reviews. *Bmj*, 309(6954), 597–599.
- Oarde, D. E., Libatique, N. C., Tangonan, G. L., Sotto, D. M., & Pacaldo, A. T. (2014). Digital motion analysis system for rehabilitation using wearable sensors. En *Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM), 2014 International Conference on* (pp. 1–7). IEEE.
- Oxer J., Blemings H., “Practical Arduino: Cool Projects for Open Source Hardware”. Apress, 26 ene. 2011.
- Pereira, A., Guimarães, V., & Sousa, I. (2017). Joint angles tracking for rehabilitation at home using inertial sensors: a feasibility study. En *Proceedings of the 11th EAI International Conference on Pervasive Computing Technologies for Healthcare* (pp. 146–154). ACM.
- Picerno, P., Viero, V., Donati, M., Triossi, T., Tancredi, V., & Melchiorri, G. (2015). Ambulatory assessment of shoulder abduction strength curve using a single wearable inertial sensor. *Journal of Rehabilitation Research & Development*, 52(2).
- Playground.Arduino.cc, “Comunicando Arduino con otros sistemas” [online]. Disponible en: <http://playground.Arduino.cc/ArduinoNotebookTraduccion/Appendix4>
Fecha de última consulta: Junio 2017.
- Ploderer, B., Fong, J., Withana, A., Klavic, M., Nair, S., Crocher, V., Nanayakkara, S. (2016). ArmSleeve: a patient monitoring system to support occupational therapists in stroke rehabilitation. En *Proceedings of the 2016 ACM Conference on Designing Interactive Systems* (pp. 700–711). ACM.
- Ramos Melgar E., Castro Diez C., (2012) “Arduino and Kinect® Projects: Design, Build, Blow Their Minds”. Apress.
- Robert-Lachaine, X., Mecheri, H., Larue, C., & Plamondon, A. (2017). Validation of inertial measurement units with an optoelectronic system for whole-body motion analysis. *Medical & biological engineering & computing*, 55(4), 609–619.
- Sánchez Muñoz J. M., (2011). Hamilton y el Descubrimiento de los Cuaterniones. *Revista de Investigación Pensamiento Matemático*.
- Taboadela, C. H. (2007). "Goniometría: Una herramienta para la evaluación de las incapacidades laborales". ASOCIART SA ART.
- Through the Interface, [online] “Calling a web-service from a Unity3D scene”. Fecha de última consulta: Diciembre 2017.
- Udemy courses: “Intro to C# Programming and Scripting for Games in Unity” [online]. Disponible en: <https://www.udemy.com/> Fecha de última consulta: Diciembre 2017.
- Udemy courses: “Learn Advanced C# Scripting in Unity 5 for Games” [online].

Disponible en: <https://www.udemy.com/> Fecha de última consulta: Diciembre 2017.

Udemy curses: "Learn Advanced Game Mechanics in Unity for Game Developers" [online].
Disponible en: <https://www.udemy.com/> Fecha de última consulta: Diciembre 2017.

Unity3D. ¿Qué es Unity? [online]. Recuperado 5 de marzo de 2018, de <https://unity3d.com> Fecha de última consulta: Diciembre 2017.

Unity documentation. "Scripting API: Vector3" [online]. Disponible en: <https://docs.unity3d.com/> Fecha de última consulta: Diciembre 2017.

Van Verth, J. (2013). Understanding quaternions. Game developers conference.

Wheat D., (2012) "Arduino Internals". Apress.

World Economic Forum, [online] "4 mega-trends that could change the world by 2030".
Recuperado 22 de agosto de 2018, de <https://www.weforum.org/agenda/2017/08/4-mega-trends-that-could-change-the-world-by-2030/>. Fecha de última consulta: agosto 2018.

Anexos

Anexo I – Metodología de planificación y desarrollo de proyectos

Esta metodología está particularmente aplicada al desarrollo Software Hardware de un producto, pero puede ser aplicable a cualquier desarrollo de un proyecto.

1 Roles del desarrollo:

Se definen los 4 distintos perfiles de desarrollo de un proyecto:

Product Owner (P.O.): es el encargado de dar la lista de especificaciones. No tiene obligatoriedad de pertenecer a la organización. Será el cliente o el que da la voz al cliente.

Scrum Master (S.M.): interpreta al P.O. traduciendo la lista de especificaciones a la realidad alcanzable. Añade además metainformación.

Quality Assurance (Q.A.): describe cómo tienen que ser los test unitarios (o *unit-test*), aunque puede no tener que escribir los test directamente, sólo definirlos.

Developer (Dev.): es necesario en todos los procesos de desarrollo, para realizar el proyecto de forma adecuada, al menos la presencia de 2 Dev. Un desarrollador escribe los test unitarios y el otro el código en sí de la aplicación y a continuación cada uno comprobará sin sesgo la parte del otro. Por ello la retroalimentación se producirá solamente al existir más de dos desarrolladores en el proyecto.

2 Fases del desarrollo:

Preplanificación e integración: Si ponemos el desarrollo de producto en una línea temporal comenzaremos por este procedimiento. Éste implicará la definición total del producto y sus requerimientos detallados. Es la fase más importante y de mayor duración, aunque en proyectos cortos se suele desestimar (incorrectamente).

Escritura del código/desarrollo del producto: En esta fase el comenzaremos el desarrollo del producto o la escritura del código, siendo esta fase, al final de la cual podemos obtener un resultado palpable del trabajo. Cada un determinado periodo

de tiempo (2 semanas, por ejemplo), se lanzará una *release* (desde el master habiendo realizado las *Pull Request* correspondientes anteriormente).

Code Freeze: Se denomina así al momento en que se paraliza el desarrollo del producto o escritura de código, no permitiendo el añadido a partir de este momento. Aparece por tanto un Buffer de tiempo destinado únicamente a la resolución de errores, sin añadir ninguna *feature*.

Release Date/Entrega: Es la fecha final de entrega de producto, como otra *release* desde el master, y que asegura el correcto funcionamiento y el cumplimiento de todos los requerimientos.

3 Modelos de desarrollo:

Principalmente en los proyectos se utiliza la organización en rama como modelo de desarrollo del proyecto.

Organización en Rama: Consiste en dividir los distintos procesos de trabajo del proyecto. Debería aparecer una rama por cada unidad atómica de trabajo del proyecto idealmente.

La línea principal será la denominada línea master, siendo la línea entregada en cada *release* en producción. Ésta línea nos asegura fiabilidad y cumplimiento de los mínimos para ser una versión finalmente lanzada.

A partir de ella se ramifica la línea de *develop*. o rama de desarrollo, que se dividirá a su vez en distintas subramas por cada unidad de trabajo del desarrollo. Estas subramas serán subidas a la línea inmediatamente superior con cada petición o *Pull Request* que el desarrollador mande (lo que se denominará un *commit*). Esta petición, si es aprobada, sustituirá la parte del código correspondiente actualizándolo y asegurando la fiabilidad y limpieza de código.

Organización en Cajas: Normalmente se utilizará para una aplicación exportable fácilmente a distinto Hardware. Utiliza normalmente para obtener su máximo partido la programación funcional, esto es, el dispositivo final sólo realizará llamadas a la parte del servidor que devolverá únicamente valores puros y sólo desarrollaremos la interfaz en el entorno de desarrollo del dispositivo objetivo. Así no será necesario exportar todo el código y podremos elegir el idioma de desarrollo que prefiramos para la parte del Framework (que se localizará en un servidor como una caja negra).

La propia organización del proyecto se denomina así por la distribución del trabajo en cajas negras, cada una pudiéndose asignar a un *developer* y así dividir el WIP (*work in progress*) en porciones accesibles. Para su *release*, dependiendo del sistema objetivo, se tomará un conjunto de cajas concéntricas que contendrán toda la parte del proyecto que necesitemos. Agruparemos entonces diferentes espacios de trabajo

para conformar nuestro producto sin utilizar *frameworks* distintos para distintos entornos.

Integración continua: La integración continua si está disponible en el sistema de control de versiones utilizado, permitirá enviar un paquete de los *commits* al CEO automáticamente. Aprovecha la organización en conjuntos como ramas o cajas para integrar automáticamente los sistemas de notificación.

Master: el master debe ser 100% funcional, 100% libre de errores (lo que nos permitirá *unit-test*), y 100% verificado por *Quality Assurance*.

Develop: simplemente debe asegurar al 100% la compilación y el funcionamiento

Subramas: no deben necesariamente cumplir ninguna de las 3 condiciones.

Para subir una subrama a Develop, el Quality Assurance, o encargado de verificar el código, mandará una Pull Request o petición de integración y el sistema determinará si es aceptada trabajando siempre desde la premisa de que el código sea incorrecto hasta corroborarlo.

En orden, primero se avisa al QA, después hasta que el QA no lo compruebe no continuará el commit, y por último el commit pasa por el servidor de integración (y los distintos test que tenga incorporados para determinar si el código está listo para incorporarse a la rama Develop.)

4 Metodologías de asignación de trabajo:

Las principales metodologías entre las que se debate cualquier equipo son **Scrum** y **Kanban**:

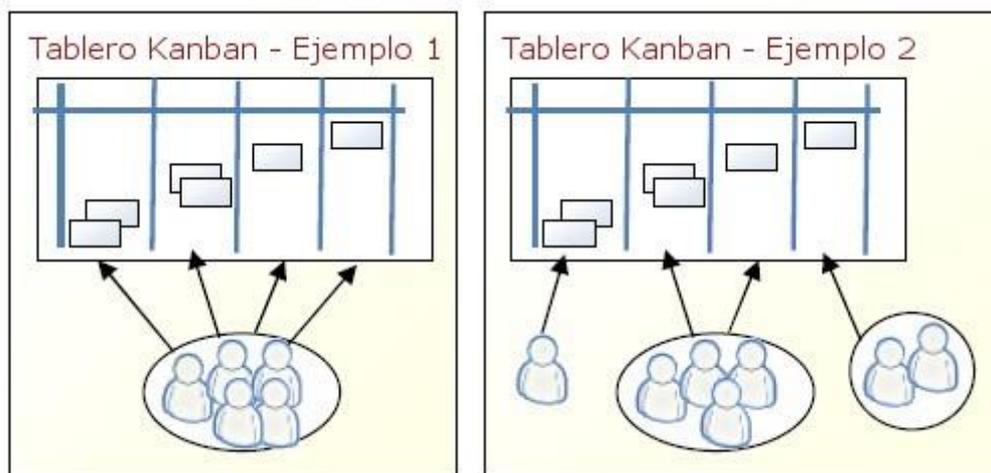


Figura 96: ejemplos tableros Kanban. Kanban vs. Scrum, de "Henrik Kniberg" et al.

Scrum organiza el equipo en pequeños equipos autogestionados y multidisciplinarios, el trabajo en una lista de entregables pequeños y concretos, el tiempo en iteraciones cortas, con código demostrable, al terminar cada iteración y optimiza la planificación de versiones, el proceso en cada iteración y actualiza las prioridades en colaboración con el cliente. Scrum prescribe roles y no así Kanban. Scrum está basado en iteraciones, en Kanban, elegimos iterar si sirve para cerrar una versión.

En Scrum, el trabajo en progreso (WIP - work in progress) es limitado por la unidad de tiempo o iteración. Este bloque temporal o iteración se denomina *sprint*. En Scrum, cuando termina el *sprint*, los ítems se sacan de la pizarra y comienza un nuevo *sprint*.

Kanban visualiza el flujo de trabajo al separar el trabajo en ítems, escribir cada ítem en una tarjeta de la pizarra Kanban y utilizar columnas que ilustran el estado de la tarjeta. Limita el trabajo en progreso: es recomendable asignar explícitamente cuantos ítems podemos tener en cada estado, un estado es una columna de la pizarra de Kanban. Mide el tiempo de iniciación de un ítem: con el objetivo de optimizar ese tiempo. En Kanban, los ítems persisten hasta que estén terminados, sin importar cuándo termine el *sprint*.

Ambos procesos están basados en el **desarrollo incremental**:

En Scrum, los equipos estiman. En Kanban, lo decide el equipo. Ambos procesos trabajan simultáneamente sobre múltiples productos. En Scrum, con diferentes pizarras. En Kanban, se diferencian los productos en la pizarra.

Scrum prescribe reuniones diarias de no más de 15 minutos. En Kanban, los equipos las hacen de igual manera. Scrum prescribe el gráfico de trabajo restante, por cada iteración. En Kanban no es un gráfico que tenga que utilizarse. En Kanban, el WIP es limitado por estados del flujo de trabajo.

En Scrum, se tiene un equipo con todos los conocimientos necesarios para completar los ítems de la iteración. En Kanban, no necesariamente hay un único equipo relacionado a un ítem.

Scrum, en general, no hace cambios en medio de una iteración. En el caso de Kanban, podemos agregar un nuevo ítem en alguna de las columnas, pero si estamos en el límite, tenemos que remover algún ítem de esa columna.

Ambos son **empíricos**, es decir, que necesitamos experimentar con el proceso y adaptarlo a nuestro contexto. Uno de los puntos a tratar, en Kanban, es el límite de trabajo en progreso (WIP), entonces, podemos experimentar y adaptarnos a nuestro proyecto.

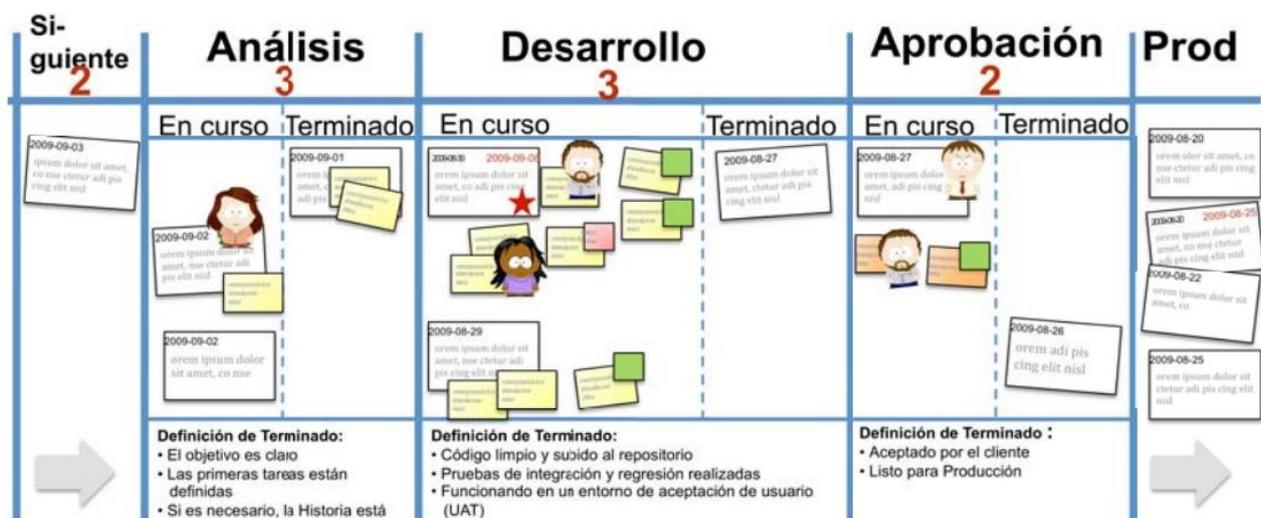


Figura 97: Proceso de proyecto: Kanban vs. Scrum, de Henrik Kniberg & Mattias Skarin.

4 Sistemas de control de versiones:

Las principales herramientas son Git y Subversion, siendo ambos sistemas que nos permitirán enviar peticiones o *commits*, controlar diferentes versiones de código y testear el contenido de los mismos, aunque de formas diferentes.

Git utiliza deltas temporales, almacenando las diferencias entre una versión del código u otra obteniendo ventajas en tiempo y coste de procesamiento. Subversion por su parte guarda el grosor del código sin dejar de almacenar el contenido ya presente en el código, sustituyéndolo.

Git trabaja en distintas ramas, frente a Subversion que trabaja en una sola línea. El sistema de control de ciclos de versión, controlará en el caso de Git si hay líneas de código incompatibles de las que no se puede hacer *merge*, posibilidad que aparece al estar Git basado en tomar deltas/diferencias de código.

Git cuenta con distintos interfaces gráficos, siendo un programa inicialmente en línea de comandos. Estos son Sourcetree y Tower entre otros, que, aunque son utilizados continuamente por desarrolladores, conviene complementarlos con conocimientos de los comandos principales de línea de códigos de Git, ya que al ser un proceso que maneja información muy importante, muchas veces ante un problema de pérdida de código serán necesarios estos comandos para revertir el proceso. Git viene instalado de serie en la mayoría de los sistemas operativos, salvo Windows.

Git cuenta con las siguientes ventajas:

- No depende de una conexión de red permanente, pudiendo trabajar en el proyecto desde cualquier lugar.
- Aporta seguridad en caso de fallo o pérdida de los repositorios principales.
- No necesita de permisos especiales de lectura y escritura para los diferentes directorios (aunque, de ser así, será posible y complejo implementarlo).
- La transmisión de los cambios es bastante más rápida.

Subversion por su parte:

- Permite permisos de acceso basados en rutas de acceso para las diferentes áreas del proyecto.
- Aporta ventajas al trabajar con numerosos archivos binarios de gran tamaño.
- También permite guardar la estructura de los directorios vacíos (estos son rechazados por Git, debido a que no contienen ningún tipo de contenido).

	SVN	Git
Control de versiones	Centralizada	Distribuida
Repositorio	Un repositorio central donde se generan copias de trabajo	Copias locales del repositorio en las que se trabaja directamente
Autorización de acceso	Dependiendo de la ruta de acceso	Para la totalidad del directorio
Seguimiento de cambios	Basado en archivos	Basado en contenido
Historial de cambios	Solo en el repositorio completo, las copias de trabajo incluyen únicamente la versión más reciente	Tanto el repositorio como las copias de trabajo individuales incluyen el historial completo
Conectividad de red	Con cada acceso	Solo necesario para la sincronización

Figura 98: pequeña comparativa entre SVN y Git

5 Integración:

El servidor de integración se encarga de comunicar la rama local o HEAD con el sistema ORIGIN, recibiendo la solicitud, *commit* o *git push* (en el caso de Git) y comprobando el cumplimiento del código corriendo los *tests* que posean.

Hay dos tipos de servidor de integración principalmente: Jenkins, gratuito, con muchos planes y servicios para aplicaciones concretas y BMBOO de pago, pero con distintos planes y un gran soporte. Además, BMBOO cuenta con un gran servicio de empaquetamiento que permite publicar en App Stores de forma automática (el servidor de integración se ocupa de todo el proceso).

Con la mayoría de los sistemas, los servidores tienen constancia cada vez que cogemos una tarea (mediante disparos). Queda asignada la tarea, creando automáticamente ramas y no permitiendo su asignación a otro *developer*.

6 Testeo de código:

Hay diferentes *tests* que se deberán correr y anteriormente programar para cada proyecto concreto, antes de comenzar a programar el prototipo en sí.

Un *unit-test* o test unitario es un método que prueba una unidad de código. Al hablar de una unidad de código nos referimos a un requerimiento. Por ello es recomendable utilizar hasta un 50% del tiempo de desarrollo del código en escribir los test y no desarrollar el propio código. Este test devolverá un booleano con el resultado tras pasar el código por él.

El modo de encontrar el Test al código exacto donde se encuentra la función que debe comprobar puede ser automático, con extensiones asociadas a Unit Test en algunas herramientas de desarrollo como Swift o no, siendo indicado expresamente en el archivo que contenga el código desarrollado de la aplicación.

Podemos definir los **distintos tipos de test** según sus objetivos de comprobación:

Test de aceptación: este test incluye imaginar todas las limitaciones que deba tener el programa y comprueba que las respuestas a las entradas del programa sean las adecuadas según lo recibido. Mucho más necesario en lenguajes de tipado débil, pues en los de tipado fuerte muchos casos ya están necesariamente requeridos por las exigencias de formato del lenguaje.

Test de regresión: busca ser capaz de regresar a los valores de la función original. En algunos casos es un test imperativo, cuando no sólo se puede llegar a desperdiciar memoria por un diseño de código poco exigente, sino que también se pueden perder Watts de potencia. En el caso del desarrollo de tecnología presente en vehículos, por ejemplo, la potencia estará ajustada al detalle y los valores si se pasan por referencia a memoria, estos no serán irrecuperables.

Test de integración: deben ser capaz de decidir si se integran bien dos funciones, además de que las variables sean correctas individualmente y en conjunto. Suelen estar presentes por colas, por ejemplo, de operaciones.

Los demás tipos de *tests* son básicamente combinación de estos tres.

Construir una batería de test, se podrá localizar tanto en el servidor de integración como de forma local. El servidor de integración lanzará para cada subrama sólo la batería de test específica necesaria para esa rama, igual que en master sólo correrá los test de producción.

6 Limpieza de código:

Se denomina "*Linting*" al proceso por el cual pasamos nuestro código a un análisis de errores potenciales, y se desarrollará antes de ser aceptado. La utilización de *lint* obliga a los desarrolladores a escribir bien el código, habiendo un *lint* específico para cada lenguaje de programación. Los motores de integración comentados anteriormente como BMBOO exigen en ocasiones el *linting*.

La compilación pasa por *lint*, por ejemplo, el máximo de caracteres, espacios, limpieza, tabulación, y se puede modificar el *linting* específico en cada lenguaje adaptándolo a las necesidades del programador, pero teniendo en cuenta mantener el *lint* local respecto al *lint* en el servidor de integración.

Todos estos métodos de trabajo son utilizados de forma metódica por las principales y grandes empresas de la industria. Podemos encontrar más información en el trabajo *Kanban vs. Scrum*, de Henrik Kniberg & Mattias Skarin.

A estos métodos podemos añadir la utilización de nuevos enfoques de programación como la programación orientada a protocolos, que permite la optimización del tiempo y del código.

Anexo II – IMU

Funcionamiento

Denominamos **Unidad de Medida Inercial** o **IMU** a un componente electrónico basado en sensores de aceleración y velocidad angular (acelerómetros y giróscopos respectivamente) la cual nos reporta el movimiento y orientación que sufre dicho dispositivo. Es el componente principal de sistemas de guía inercial utilizados en vehículos aéreos, espaciales, marinos y aplicaciones robóticas

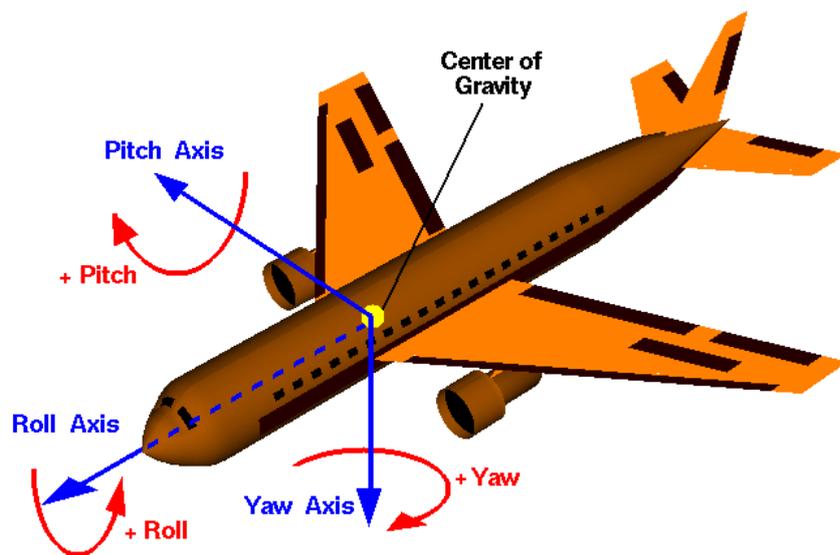


Figura 99: Orientación proporcionada por una IMU, ejemplo aviación.

1. Componentes de una IMU

Toda IMU está compuesta como mínimo por un acelerómetro y un giróscopo para captar una aceleración y una velocidad angular en concreto. Generalmente, es interesante que las IMUs capturen la aceleración y la velocidad angular en los tres ejes de coordenadas (*yaw*, *roll*, *pitch*) para conocer el movimiento exacto del componente.

También podemos encontrar unidades que incorporan un microprocesador que se encarga de recoger los datos de dichos sensores y enviarlos de forma ordenada al usuario mediante un protocolo de comunicación incorporado en la IMU.

A continuación, se detallarán los componentes nombrados:

Acelerómetro: instrumento capaz de medir aceleración en uno, dos o tres ejes. Existen varios tipos de acelerómetros, dependiendo de su fabricación y funcionamiento. Las IMUs incorporan acelerómetros integrados en silicio, utilizando la tecnología llamada MEMS, debido a la necesidad de reducir el tamaño total de la unidad. La mayoría de éstos son capacitivos, y calculan la aceleración mediante el voltaje obtenido entre dos placas una de las cuales varía su posición dependiendo del movimiento del acelerómetro. Se caracterizan por ser muy precisos en situaciones estables y tener un gran error en situaciones vibratorias o movimientos muy inestables.

Giróscopo o giroscopio: dispositivo que mide la orientación, basándose en los principios de la conservación del momento angular. Las unidades de medida inercial utilizan giróscopos MEMS, es decir, integrados y de tamaño reducido. La salida de dicho sensor es un voltaje, la variación del cual nos indica en grados por segundo ($V/^\circ/s$) la velocidad angular sufrida por el sensor. Se caracterizan por tener un error constante y lineal llamado *bias* el cual debemos tener en cuenta.

Microprocesador: algunas unidades de medida inercial, como ya hemos comentado, incorporan un microprocesador. Éste es programable, pero su principal función es recoger los datos entregados por los sensores, procesarlos según desee el usuario y enviarlos. En el microprocesador se define la frecuencia de trabajo de la unidad, que será el tiempo comprendido desde que recoge el dato del primer sensor hasta que envía al usuario el dato procesado del último sensor.

La mayoría de los microprocesadores incorporan un convertor analógico-digital, para así convertir el voltaje dado por el sensor en una muestra.

Protocolo de comunicación: los protocolos de comunicación alámbricos típicos en las IMUs son el UART, el RS-232 o el USB. Algunas unidades incluyen protocolos inalámbricos, siendo los más utilizados *ZigBee* y *bluetooth*.

Magnetómetro: algunas unidades de medida inercial incluyen también sensores magnetómetros. Estos dispositivos miden la fuerza y/o dirección de los campos magnéticos que los afectan respecto el campo magnético terrestre. Aunque cabe la posibilidad de que se vean afectados por variación de otros campos magnéticos en algunas zonas.

2. Caracterización de una IMU

Los diferentes tipos de unidades de medida inercial que podemos encontrar en el mercado se caracterizan generalmente por el tipo de sensores de que están compuestas.

Estos sensores, la frecuencia de trabajo, que puede interesar mayor o menor dependiendo de la aplicación para la que esté destinada la unidad, y el protocolo de comunicación son los que definen una IMU mejor que otra.

Los sensores de que se componen las unidades de medida inercial se definen principalmente por su **rango de trabajo** (máxima medida que soporta el sensor), su sensibilidad (relación entre la variación de la magnitud de salida y la de entrada) y su ancho de banda de respuesta (frecuencia de funcionamiento del sensor). Para el caso de los acelerómetros, el rango de trabajo se mide con la gravedad estándar, aceleración de $g = 9.8 \text{ m/s}^2$. Podemos encontrar IMUs con un rango de 1.5g, 2g, 4g, etc. Dependiendo de este rango, obtenemos una sensibilidad u otra, ya que el nivel de voltaje máximo que podemos obtener viene definido por la alimentación del sensor.

Entonces, para mayor rango, menor sensibilidad. En los giróscopos, el rango de medida se mide en $^\circ/\text{s}$, ya que obtenemos una velocidad angular, y la sensibilidad en $\text{mV}/^\circ/\text{s}$. Valores típicos de rango que podemos encontrar son $\pm 200 \text{ }^\circ/\text{s}$, $\pm 300 \text{ }^\circ/\text{s}$, $\pm 500 \text{ }^\circ/\text{s}$, etc.

Dado que el ángulo se calcula a partir de la gravedad, no es posible calcular el ángulo Z con esta fórmula ni con ninguna otra. Para hacerlo se necesita otro componente: el magnetómetro. El MPU-6050 no lleva, y por tanto nunca podrá calcular con precisión el ángulo Z. Sin embargo, en este caso, es posible calcular el eje Z mediante fórmulas matemáticas que nos resulten en el ángulo Z de forma aproximada.

En el caso de que la IMU tenga magnetómetros, el rango se mide en gauss = 1 maxwell / cm^2 .

La frecuencia de trabajo de la unidad de medida inercial viene definida por el reloj que use el microprocesador para enviar los datos al usuario y por la frecuencia de conversión del conversor analógico-digital. También influye la velocidad de transmisión, que depende del protocolo que utilicemos y de cómo lo definamos. Ésta velocidad se mide en baudios o bits por segundo y son valores típicos 38400 bps, 57600 bps, 115200 bps, etc.

3. Error en las medidas

Hay dos problemas muy importantes cuando hablamos del funcionamiento de un IMU: el ruido y los errores.

El **ruido** son todas aquellas interferencias que afectan a los dispositivos electrónicos. El acelerómetro es capaz de medir cualquier ángulo, sin embargo, sus lecturas son ruidosas y tienen un cierto margen de error.

Además, el acelerómetro también detecta cualquier aceleración que no sea la de la gravedad. Por tanto, si mueves la IMU sin girarla, al aplicar una aceleración en otro eje, la IMU lo detectará como un cambio de rotación.

Por otra parte, tenemos el giroscopio, que, a diferencia del acelerómetro, da las medidas con mucha precisión. Pero al realizar los cálculos del ángulo es inevitable que se produzca un pequeño error, que con el tiempo va acumulándose hasta que cualquier similitud con la realidad es pura coincidencia. Esto es el denominado *drift*.

Ante esta problemática, hay varias formas de combinar los datos del acelerómetro y el giroscopio para así obtener medidas precisas. Hay distintos **algoritmos**, llamados **filtros**, que hacen esta tarea. Uno de los mejores es el famoso Filtro de Kalman, utilizado tanto en los aviones, como en cohetes y en satélites geostacionarios. Es capaz de calcular el error de cada medida a partir de las medidas anteriores, eliminarlo y dar el valor real del ángulo. En cierto modo es un algoritmo que aprende en cada iteración. Sus principales problemas son que posee un coste de procesamiento algo elevado y que es, además, muy complejo.

Además de éste, tenemos la posibilidad de utilizar otros filtros, como el Filtro Complementario o *Complementary Filter*. Es ideal para implementar con un microprocesador: fácil de utilizar, bajo coste de procesamiento y con una precisión muy buena. Es en realidad una unión de dos filtros diferentes: un *High-pass Filter* para el giroscopio y un *Low-pass Filter* para el Acelerómetro. El primero deja pasar únicamente los valores por encima de un cierto límite, al contrario que el *Low-pass filter*, que sólo permite a los que están por debajo. En la fórmula resultante tendremos el AnguloGyro que será el ángulo del Giroscopio que hemos calculado previamente, y el AnguloAccel con el ángulo del Acelerómetro calculado con la fórmula de la tangente. También se tomará Δt como el tiempo que ha pasado (en segundos) desde la última vez que se ha calculado el filtro para tener en cuenta el error anterior.

$$Angulo = 0.98 * (Angulo + AnguloGyro * \Delta t) + 0.02 * AnguloAccel \quad (111)$$

Ec. 1: fórmula filtro complementario.

Por último, debemos hablar del **Gimbal Lock**, un efecto bastante molesto que se produce cuando se produce la rotación de cualquier entidad en el eje Y de 90/-90 grados. La consecuencia es que el eje X y el Z acaban apuntando en la misma dirección, perdiéndose el control sobre ellos. Ocurrirá en cualquier paquete de animación, programa de diseño 3D, o cualquier motor 3D que utilice para las rotaciones los ángulos de Euler.

Una de las formas que se proponen para acabar con el problema del Gimbal Lock es la utilización de cuaterniones en lugar de ángulos de Euler.

Tras todo esto conseguiremos un resultado muy fiable de la posición del dispositivo, que podrá ser utilizado para nuestros prototipos.

Anexo IV - Manual de uso

Software valoración ergonómica

1. Puesta en marcha

A continuación, se enumeran y explican brevemente los pasos para la puesta en marcha de la aplicación, tomando como punto de partida un sistema (PC) adecuadamente configurado para correrla. Previo a la ejecución del programa se deberá emparejar el dispositivo maestro (uno sólo de las unidades vestibles) mediante bluetooth al PC en el que ejecutemos la aplicación. Tras el primer emparejado no es necesario ningún otro ajuste para que el Hardware sea reconocido por la aplicación en posteriores ejecuciones.

Los dispositivos se colocan en la zona a valorar, en este caso, para el miembro superior derecho se debe ubicar un sensor en la espalda, uno en el brazo y otro en el antebrazo. De esta forma podremos conocer información acerca de al menos dos articulaciones: hombro y codo.

Los sensores se colocan de forma sencilla mediante velcro, como podemos apreciar en la figura:

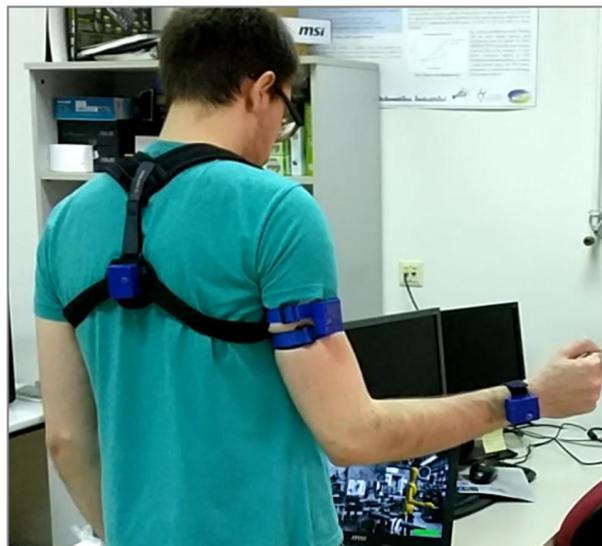


Figura 100: prototipo final colocado en espalda, brazo y antebrazo junto con el software de valoración ergonómica en ejecución.

Para inicial el funcionamiento del dispositivo, deberemos encender mediante sus interruptores las distintas unidades que utilizemos, las que después de su calibración inicial automática, pasaremos a colocar en su posición. Para esta primera versión del software de valoración ergonómica, deberemos grabar los datos en primer lugar utilizando en primer lugar el programa de visualización en tiempo real realizado en Unity3D.



Figura 101: software desarrollado en Unity3D de captura de movimientos.

Este programa generará un fichero de texto con información del movimiento registrado, el cual, será leído a continuación por el programa de valoración ergonómica basado en python. Tras la ejecución del algoritmo de evaluación cinemática que analiza los movimientos registrados en forma de cuaterniones y los transforma en ángulos absolutos respecto a una posición de cúbito supino, obtendremos diversas gráficas que nos mostrarán los distintos movimientos realizados en cada uno de los tres ejes de movimiento posibles. Estos movimientos se definen de acuerdo a lo visto en el punto **6.1.3 Dinámica del hombro**. Podemos observar a continuación, las gráficas obtenidas para la ejecución de un análisis ergonómico real:



Figura 102: gráficas resultantes de software de evaluación ergonómica, caso de análisis del movimiento del hombro: Abducción/Adducción, flexión/extensión y rotación interna/externa.

La herramienta software permitía presentar los resultados de forma interactiva y fácilmente interpretable para los expertos evaluadores, mediante archivos html. Además añadía información adicional gracias a análisis gráficos de tipo “tarta” en los que se detalla el porcentaje temporal respecto al total del puesto de trabajo, en que se superan ciertos márgenes de movimiento, permitiendo así la corrección o planificación de puestos según movimientos clave que sobrepasan cierto nivel angular.

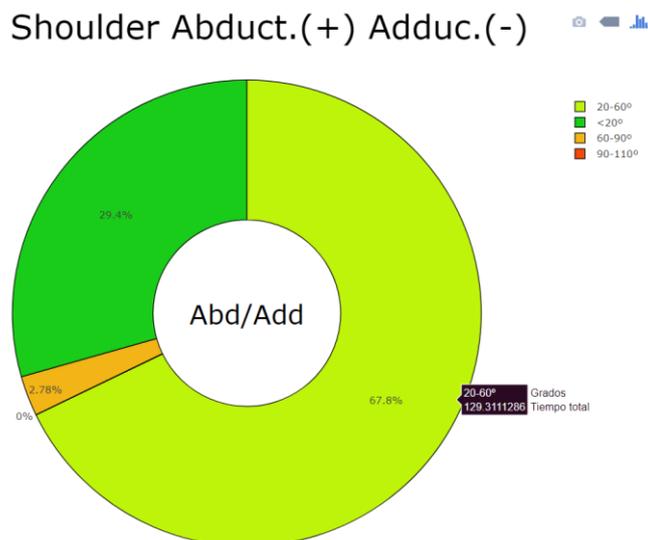


Figura 103: gráficas en forma de “tarta” o “donut centradas en el análisis temporal de una serie de rangos de movimiento.

Para su mejor observación se realizó un reproductor en Unity3D que permitía abrir los archivos de sesiones de evaluación ergonómica grabadas y superponer el video

grabado mediante cámaras externas en su puesto con el resultado de la captura de movimiento en tiempo real. Pudiendo así además analizar momentos temporales concretos que puedan ser de interés:



Figura 104: herramienta de reproducción en vídeo simultánea. El archivo .txt se corresponde con la información de la sesión capturada.

2. Manual de uso de la aplicación

La aplicación posee ciertas características, que pasaremos a describir:

- Consta de dos partes claramente diferenciadas, la aplicación desarrollada en Unity3D que permite el visionado del movimiento en tiempo real y la aplicación en Python que permite el análisis a posteriori de los puestos capturados.
- Permite la detección en tiempo real de nuestros movimientos y la presentación en un avatar de los mismos.
- Además, con un simple botón de la aplicación permite la grabación (captura de datos) de los movimientos tanto en video como en un archivo .txt con únicamente los datos de movimiento capturados.
- El programa en Python permite un análisis sencillo mediante gráficas de los datos previamente capturados que, además, son editables mediante el software de edición que incorpora Plot.ly.

- Los sensores son fáciles de colocar en distintas partes de nuestro cuerpo y sus baterías son recargables mediante micro-usb.

3. Ejemplos de funcionamiento

En este apartado, incluiremos las capturas realizadas durante la etapa de pruebas para la evaluación del sistema sensores/software ergonómico. Estas capturas pertenecen a los movimientos mostrados en el apartado **7.1.2.4** que recogimos para el análisis de movimiento posterior, pero muestran parte del sistema en tiempo real de recogida de datos, con el movimiento del avatar a la vez que del usuario. Podemos observar como la postura del usuario y del avatar coinciden en las capturas:

Abducción/aducción:

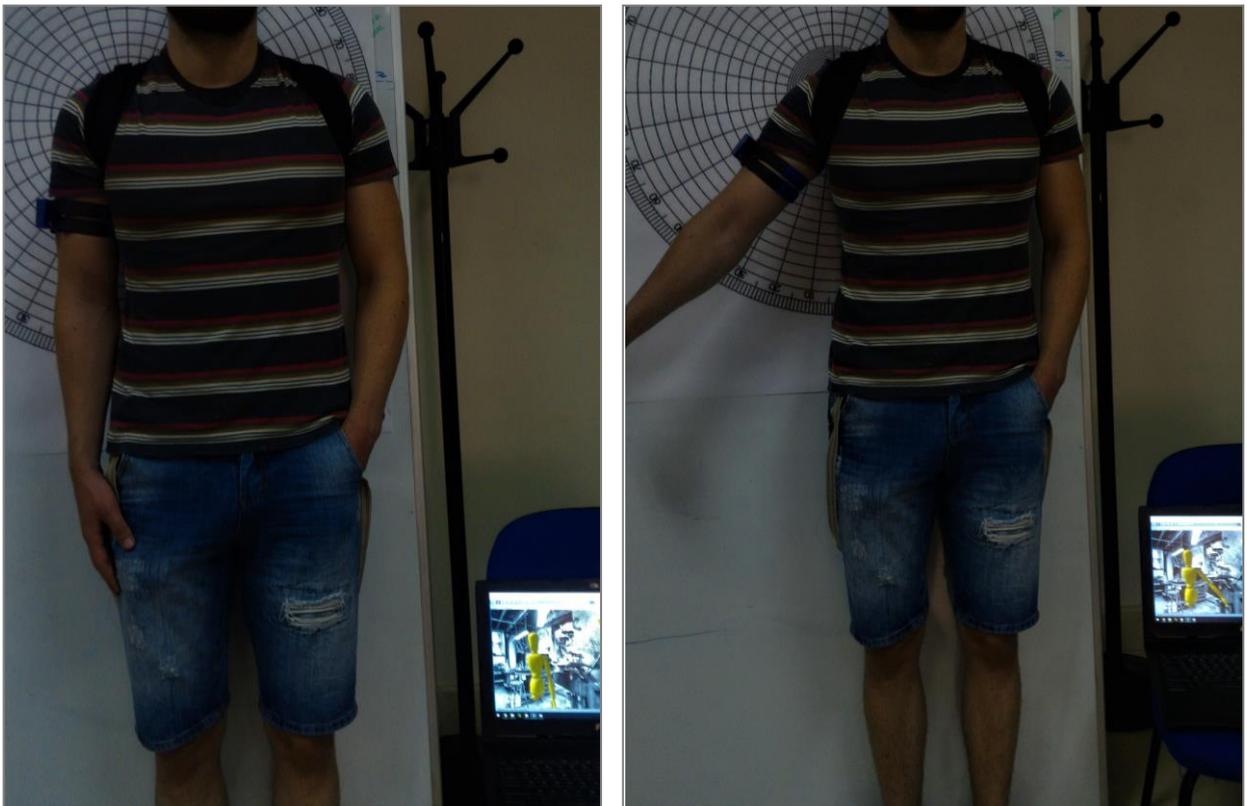


Figura 105: Abducción 0° (izq.) vs 45° (der.)



Figura 106: Abducción 90° (izq.) vs 110° (der.)

Flexo-extensión:



Figura 107: Flexión 0° (izq.) vs 20° (der.)

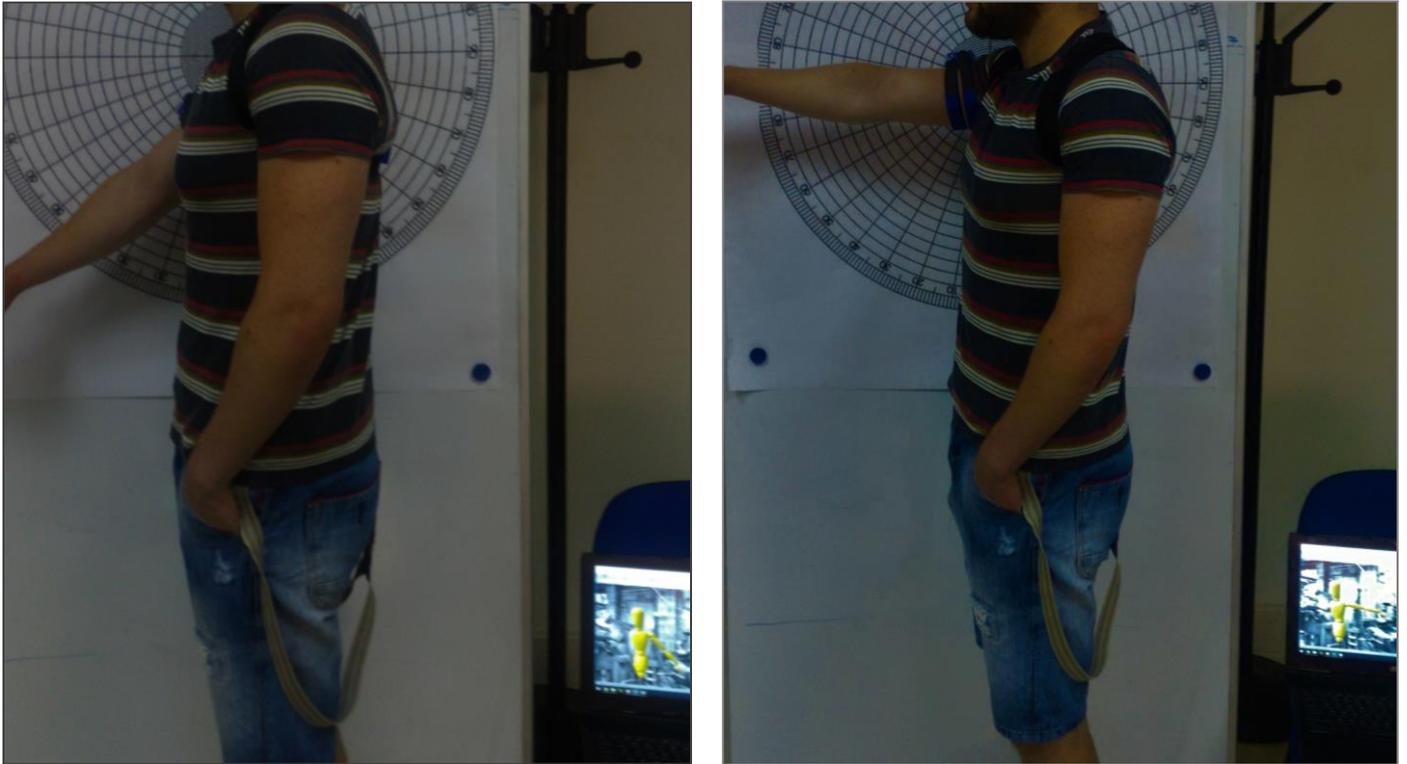


Figura 108: Flexión 45°(izq.) vs 90° (der.)

Las capturas de **rotación interna/externa** se han descartado debido a que eran movimientos poco claros de observar en el avatar, debido a que si se conectaban sólo dos sensores (tronco y brazo) se veía el movimiento en la posición de la palma de la mano del avatar, lo cual es difícilmente observable en las fotos. Por otra parte, si utilizáramos tres sensores (tronco, brazo y antebrazo) solamente nos fijaríamos en los movimientos de la articulación del codo y realmente podíamos caer en engaño visual al pretender observar la articulación del hombro. Además de que debemos ser especialmente cuidadosos al visualizar el avatar del programa de captura, pues el ángulo de cámara puede hacernos creer que no se esté visualizando el movimiento correctamente.