



**UNIVERSIDAD DE CASTILLA-LA MANCHA**  
**ESCUELA SUPERIOR DE INFORMÁTICA**

**GRADO EN INGENIERÍA INFORMÁTICA**

**Ingeniería de Computadores**

**TRABAJO FIN DE GRADO**

Sistema de ayuda a la rehabilitación a distancia para personas  
con problemas de visión

Mario Pérez Sánchez-Montañez

Septiembre, 2021





**UNIVERSIDAD DE CASTILLA-LA MANCHA  
ESCUELA SUPERIOR DE INFORMÁTICA**

**Tecnologías y Sistemas de Información**

**Ingeniería de Computadores**

**TRABAJO FIN DE GRADO**

**Sistema de ayuda a la rehabilitación a distancia para  
personas con problemas de visión**

Autor: Mario Pérez Sánchez-Montañez

Tutor: Dr. Javier Alonso Albusac Jiménez

Septiembre, 2021

Sistema de ayuda a la rehabilitación a distancia para personas con problemas de visión  
© Mario Pérez Sánchez-Montañez, 2021

Este documento se distribuye con licencia CC BY-NC-SA 4.0. El texto completo de la licencia puede obtenerse en <https://creativecommons.org/licenses/by-nc-sa/4.0/>.

La copia y distribución de esta obra está permitida en todo el mundo, sin regalías y por cualquier medio, siempre que esta nota sea preservada. Se concede permiso para copiar y distribuir traducciones de este libro desde el español original a otro idioma, siempre que la traducción sea aprobada por el autor del libro y tanto el aviso de copyright como esta nota de permiso, sean preservados en todas las copias.

Este texto ha sido preparado con la plantilla  $\LaTeX$  de TFG para la UCLM publicada por [Jesús Salido](#) en GitHub<sup>1</sup> o Overleaf<sup>2</sup> como parte del curso « *$\LaTeX$  esencial para preparación de TFG, Tesis y otros documentos académicos*» impartido en la Escuela Superior de Informática de la Universidad de Castilla-La Mancha.



---

<sup>1</sup>[https://github.com/JesusSalido/TFG\\_ESI\\_UCLM](https://github.com/JesusSalido/TFG_ESI_UCLM)

<sup>2</sup><https://www.overleaf.com/latex/templates/plantilla-de-tfg-escuela-superior-de-informatica-uclm/phjgscmfqtsw>

TRIBUNAL:

Presidente: \_\_\_\_\_

Vocal: \_\_\_\_\_

Secretario: \_\_\_\_\_

FECHA DE DEFENSA: \_\_\_\_\_

CALIFICACIÓN: \_\_\_\_\_

PRESIDENTE

VOCAL

SECRETARIO

Fdo.:

Fdo.:

Fdo.:



*A mi yayo por ser un ejemplo de superación ante la adversidad.*



## Resumen

La rehabilitación en pacientes que han sufrido alguna lesión severa implica la visita a un centro médico. Esto se convierte en un problema aún mayor para personas que viven en un radio de 100 km y deben desplazarse, teniendo que implicar a familiares o algún servicio de ambulancia. Además, esto supone un coste alto tanto para el paciente como para el sistema sanitario. La evolución tecnológica de los últimos años ha propiciado la aparición de sistemas informáticos que permiten que doctor y paciente no tengan que estar necesariamente en el mismo lugar. Sin embargo, gran parte de estos sistemas no están preparados para personas con problemas de visión.

En este proyecto se propone el diseño y desarrollo de un sistema que permita al personal sanitario definir el plan de ejercicios para cada paciente, y al paciente con problemas de visión realizar los ejercicios desde su hogar. El sistema constará de un microcontrolador con un conjunto de sensores y actuadores, principalmente elementos de vibración, que ubicados en diferentes partes del cuerpo podrán guiar al paciente para realizar los ejercicios correctamente. El sistema además permitirá hacer un seguimiento de como se han realizado estos ejercicios y tomar decisiones en consecuencia.



## **Abstract**

Rehabilitation in patients who have suffered a severe injury involves a visit to a medical center. This is an even greater problem for people who live within a radius of 100 km and must travel, having to involve family members or an ambulance service. In addition, this entails high costs for both the patient and the healthcare system. Technological developments in recent years have led to the appearance of computer systems that allow the doctor and patient to be in the same place. However, most of these systems are not prepared for visually impaired people.

This project proposes the design and development of a system that allows health personnel to define the exercise plan for each patient, and the visually impaired patient to perform the exercises from home. The system will consist of a microcontroller with a set of sensors and actuators, mainly vibration elements, which located in different parts of the body will be able to guide the patient to perform the exercises correctly. The system will also make it possible to monitor how these exercises have been performed and make decisions accordingly.



# AGRADECIMIENTOS

---

En primer lugar, agradecer a mis padres, Paco y Mayte, y a mi hermana, Celia. Por darme la oportunidad de estudiar lo que siempre había querido y por estar a mi lado día a día, sobre todo en los momentos difíciles.

Gracias a mi familia por acompañarme en esta etapa. En especial a mis tíos, Luis y Esther, y a mis yayos, Francis y Tere, por preocuparse siempre por mí y cuidarme como a un hijo.

Agradecer a mi compañero y amigo, David, las incontables horas de trabajo juntos durante esta etapa y todos los momentos vividos. Sin tu amistad no hubiera sido igual.

Por último, agradecer a mi tutor, Javier, por darme la oportunidad de realizar este proyecto y por guiarme durante su desarrollo.

*Mario Pérez Sánchez-Montañez*  
Ciudad Real, 2021



# ÍNDICE GENERAL

---

<b>Resumen</b>	<b>IX</b>
<b>Abstract</b>	<b>XI</b>
<b>Agradecimientos</b>	<b>XIII</b>
<b>Índice de figuras</b>	<b>XVII</b>
<b>Índice de tablas</b>	<b>XIX</b>
<b>Índice de listados</b>	<b>XXI</b>
<b>1. Introducción</b>	<b>1</b>
1.1. Estructura del documento . . . . .	5
<b>2. Objetivos</b>	<b>7</b>
2.1. Objetivo general . . . . .	7
2.2. Objetivos específicos . . . . .	7
<b>3. Estado del arte</b>	<b>9</b>
3.1. Juegos Serios y Rehabilitación . . . . .	9
3.1.1. Juegos serios y metodologías de diseño . . . . .	9
3.1.2. Taxonomía de juegos serios . . . . .	10
3.1.3. Gamificación . . . . .	12
3.2. Sistemas de ayuda para personas con discapacidad visual . . . . .	12
3.3. Sistemas de rehabilitación . . . . .	16
3.4. Estudio y elección del entorno hardware . . . . .	19
<b>4. Metodología</b>	<b>23</b>
4.1. Metodología elegida . . . . .	23
4.2. Distribución del trabajo . . . . .	24
4.2.1. Paquetes de trabajo . . . . .	24
4.2.2. Iteraciones . . . . .	28
4.3. Recursos . . . . .	30
4.3.1. Recursos Hardware . . . . .	30
4.3.2. Recursos <i>Software</i> . . . . .	31
<b>5. Arquitectura</b>	<b>33</b>
5.1. Visión general . . . . .	33
5.2. Arquitectura del sistema . . . . .	35
5.2.1. Aplicación del paciente . . . . .	35
5.2.2. Aplicación del terapeuta . . . . .	36

5.2.3.	Sistema de seguimiento de articulaciones . . . . .	37
5.2.4.	Sistema de interacción por audio . . . . .	37
5.2.5.	<i>Backend</i> . . . . .	38
5.3.	Sistema de guiado basado en vibraciones incorporadas a un guante . . . . .	39
5.3.1.	Proceso de guiado . . . . .	39
5.3.2.	Implementación . . . . .	41
5.3.3.	Comunicación . . . . .	43
5.4.	Seguimiento del movimiento de articulaciones . . . . .	44
5.4.1.	Visión general . . . . .	44
5.4.2.	Manejo del dispositivo . . . . .	45
5.4.3.	Obtención de datos corporales . . . . .	47
5.4.4.	<i>Tracking</i> de la mano . . . . .	48
5.4.5.	Procesamiento de datos tridimensionales . . . . .	49
5.4.6.	Comunicación con las aplicaciones de rehabilitación . . . . .	50
5.5.	Aplicación de escritorio con las funciones del terapeuta . . . . .	50
5.5.1.	Entradas y salidas . . . . .	51
5.5.2.	Definición de ejercicios . . . . .	51
5.5.3.	Seguimiento del paciente . . . . .	53
5.6.	Aplicación de escritorio para ejecución de ejercicios . . . . .	54
5.6.1.	Entradas y salidas . . . . .	55
5.6.2.	Representación del ejercicio . . . . .	56
5.6.3.	Realización de ejercicios . . . . .	57
5.7.	<i>Backend</i> . . . . .	59
5.7.1.	Base de datos . . . . .	60
5.7.2.	API Rest . . . . .	62
<b>6.</b>	<b>Resultados</b> . . . . .	<b>63</b>
6.1.	Sistema resultante . . . . .	63
6.1.1.	Aplicación del terapeuta . . . . .	63
6.1.2.	Aplicación del paciente . . . . .	65
6.2.	Experimentación . . . . .	67
6.2.1.	Ejercicio 1 . . . . .	68
6.2.2.	Ejercicio 2 . . . . .	72
6.2.3.	Valoración final . . . . .	75
<b>7.</b>	<b>Conclusiones</b> . . . . .	<b>77</b>
7.1.	Objetivos alcanzados . . . . .	78
7.2.	Justificación de competencias adquiridas . . . . .	79
7.3.	Trabajo futuro . . . . .	79
	<b>Bibliografía</b> . . . . .	<b>81</b>
	<b>A. Repositorios del sistema desarrollado</b> . . . . .	<b>89</b>

# ÍNDICE DE FIGURAS

---

3.1. Taxonomía de juegos serios [46] . . . . .	11
3.2. Primera versión comercial de <i>Tom Pouce</i> [37] . . . . .	13
3.3. Segunda versión comercial de <i>Teletact</i> [37] . . . . .	14
3.4. Primera versión de <i>Geotact</i> [37] . . . . .	14
3.5. Interfaz de usuario del sistema <i>BlindAid</i> [48] . . . . .	15
3.6. Entorno experimental (A) y la representación realizada por una persona invidente (B) [41] . . . . .	15
3.7. Ejemplo de detección de objetos [5] . . . . .	16
3.8. Imágenes de los ejercicios de rehabilitación [68] . . . . .	17
3.9. Ejecución de la flexión de rodilla [26] . . . . .	17
3.10. Escenario del videojuego [7] . . . . .	18
3.11. Ejemplo de ejecución [44] . . . . .	18
3.12. Imágenes del sistema <i>PhyReUP!</i> [71] . . . . .	19
3.13. Azure Kinect DK . . . . .	20
3.14. Ejemplos de guantes [30] . . . . .	21
4.1. Infografía desarrollo iterativo e incremental . . . . .	23
5.1. Esquema general del sistema . . . . .	34
5.2. Arquitectura del sistema propuesto . . . . .	35
5.3. Creación de un ejercicio . . . . .	37
5.4. Componentes <i>hardware</i> del dispositivo de guiado . . . . .	39
5.5. Diagrama de flujo del proceso de retroalimentación háptica . . . . .	40
5.6. Dispositivo de guiado . . . . .	41
5.7. Eje de coordenadas utilizado . . . . .	41
5.8. Diagrama de clases del sistema de guiado . . . . .	42
5.9. Comparación peticiones <i>API Rest</i> y <i>WebSocket</i> . . . . .	44
5.10. Esquema general del sistema de seguimiento de articulaciones . . . . .	45
5.11. Componentes principales del dispositivo <i>Azure Kinect DK</i> [57] . . . . .	46
5.12. Campo de visión de la cámara RGB y la profundidad [57] . . . . .	46
5.13. Articulaciones procesadas por <i>Azure Kinect DK</i> [56] . . . . .	48
5.14. Máquina de estados finitos para el procesamiento de los datos de la articulación . . . . .	49
5.15. Coordenadas del <i>Azure Kinect DK</i> [80] . . . . .	49
5.16. Esquema general de la aplicación del terapeuta . . . . .	51
5.17. Reproducción de los puntos de control . . . . .	54
5.18. Diagrama de casos de uso del paciente . . . . .	54
5.19. Esquema general de la aplicación del paciente . . . . .	55
5.20. Diagrama de secuencia para iniciar ejercicio . . . . .	56
5.21. Ejecuciones de un ejercicio en distintos espacios dimensionales . . . . .	57
5.22. Diagrama de flujo ejecución ejercicio . . . . .	58

---

5.23. Pacientes realizando un ejercicio . . . . .	59
5.24. Visión general del <i>backend</i> . . . . .	60
5.25. Esquema de la base de datos . . . . .	61
5.26. Estructura de las peticiones . . . . .	62
6.1. Proceso de creación del ejercicio por el terapeuta . . . . .	64
6.2. Ejecución ejercicio desde varias perspectivas . . . . .	65
6.3. Configuración de red del sistema de guiado . . . . .	65
6.4. Dispositivo de guiado conectado . . . . .	66
6.5. Interfaz del paciente . . . . .	67
6.6. Puntuación ejercicio 1 . . . . .	70
6.7. Variaciones ejercicio 1 . . . . .	70
6.8. Ejecuciones Ejercicio 1 . . . . .	71
6.9. Puntuación ejercicio 2 . . . . .	73
6.10. Variaciones ejercicio 2 . . . . .	73
6.11. Ejecuciones ejercicio 2 . . . . .	74
6.12. Puntuación . . . . .	75
6.13. Puntuación global . . . . .	76
6.14. Variación media total . . . . .	76
A.1. Código QR del repositorio de la aplicación rehabilitación . . . . .	89
A.2. Código QR del repositorio del sistema de guiado . . . . .	90
A.3. Código QR del repositorio del <i>backend</i> . . . . .	90

# ÍNDICE DE TABLAS

---

4.1. Resumen de la iteración 1 . . . . .	28
4.2. Resumen de la iteración 2 . . . . .	28
4.3. Resumen de la iteración 3 . . . . .	29
4.4. Resumen de la iteración 4 . . . . .	29
4.5. Resumen de la iteración 5 . . . . .	29
4.6. Resumen de la iteración 6 . . . . .	30
4.7. Resumen de la iteración 6 . . . . .	30
4.8. Resumen de la iteración 7 . . . . .	30
6.1. Resultados ejercicio 1 . . . . .	68
6.2. Resultados ejercicio 1 . . . . .	69
6.3. Comparación resultados ejercicio 1 sin guiado . . . . .	69
6.4. Resultados ejercicio 2 . . . . .	72
6.5. Resultados ejercicio 2 . . . . .	72
6.6. Comparación resultados ejercicio 2 sin guiado . . . . .	72
6.7. Resultados globales . . . . .	75



# ÍNDICE DE LISTADOS

---

5.1. Clase comunicador del sistema de guiado . . . . .	42
5.2. Algoritmo de definición de ejercicios . . . . .	52
5.3. Algoritmo para calcular las diferencias vectoriales entre puntos de control . . . . .	53
5.4. Algoritmo para recreación del ejercicio . . . . .	56
5.5. Ejemplo del modelo <i>Patient</i> . . . . .	61



# INTRODUCCIÓN

---

En la actualidad, la tecnología tiene un papel de gran importancia en la integración de personas en la sociedad con algún tipo de discapacidad, ayudando a que éstas puedan realizar de manera más autónoma multitud de actividades. Es por ello que se plantea la necesidad de que las personas con discapacidad puedan acceder a esta tecnología, lo cual no siempre es posible debido a la falta de recursos económicos de las personas con discapacidad y a la falta de información sobre estos recursos.

La medicina física y la rehabilitación como especialidad médica en los Estados Unidos surgió en la década de 1930. Posteriormente, se consolidó en la sociedad dando lugar a la especialidad que se conoce a día de hoy [51]. En ese momento, se trataba de una especialidad médica que se centraba en el tratamiento físico y motor de las personas con discapacidad, pero que no tenía en cuenta el aspecto psíquico y psicológico. La rehabilitación médica se ha convertido en una especialidad médica que abarca todos los aspectos del paciente y su entorno. El objetivo de ésta es el restablecimiento o mejora funcional de las personas con discapacidad o algún tipo de lesión que le incapacite, así como su reintegración social y laboral.

Las personas saludables a nivel articular gozan de la capacidad y libertad de movimientos durante su vida cotidiana, lo que les permite ejercer su actividad profesional o cualquier otra actividad sin problemas. Con el envejecimiento aparece un aumento de la densidad en el cartílago y en los tejidos asociados, sumado a la tendencia de pérdida de movilidad articular, desarrollo de artritis y de otras patologías, que intensifica la restricción de movimiento. Una flexibilidad adecuada posibilita al ser humano a realizar eficazmente distintos movimientos articulares. La ausencia de ésta conduce a un aumento de la probabilidad de lesión y a diversos problemas funcionales, sobretodo en personas sedentarias o de avanzada edad. La flexibilidad disminuye progresivamente con la edad, sufriendo una pérdida de 20 a 30 % entre los 30 y 70 años de edad [28].

Las lesiones del cartílago articular son una de las principales causas de atención en los sistemas de sanitarios, especialmente en países donde la esperanza de vida es alta. El cartílago articular es de vital importancia en las articulaciones con amplio rango de movimiento y sus funciones principales son transmitir las fuerzas sobre las superficies articulares, amortiguar las cargas y proveer una superficie de deslizamiento adecuada entre articulaciones. Las lesiones de origen degenerativo son las más frecuentes, convirtiéndose en un problema de salud pública por el alto impacto socio-económico que representan los gastos directos e indirectos relacionados con el tratamiento o ausencia laboral [29].

Además, de las lesiones puramente físicas o producidas por el paso de los años, existen multitud de lesiones de origen neurológico, las cuales necesitan de rehabilitación física. Una de las enfermedades neurológicas más comunes son los ictus, que producen secuelas discapacitantes o el fallecimiento. Entre las secuelas físicas más comunes que sufren las personas afectadas por enfermedades de este origen, se encuentra la pérdida de fuerza, movilidad o sensibilidad en distintas partes del sistema músculo-esquelético. Por ello, es muy común la aplicación de rehabilitación física para mejorar estas secuelas. Entre los tratamiento más comunes, se encuentran la realización de ejercicios terapéuticos

que impliquen realizar ciertos movimientos, para conseguir recuperar la movilidad articular y la fuerza del paciente en las zonas corporales afectadas [52].

Un ejercicio terapéutico consiste en ejecutar de forma sistemática y planificada movimientos corporales, posturas, actividades físicas, con el propósito de que el paciente disponga de medios para corregir o prevenir alteraciones corporales, mejorar o restablecer el funcionamiento físico o reducir factores de riesgo. Se considera una técnica fundamental dentro del ámbito fisioterapéutico, ya sea con fines terapéuticos o preventivos. Se han publicado numerosos estudios científicos [63] que han demostrado con la suficiente evidencia, que los programas de terapia activa en rehabilitación correctamente pautados son muy beneficiosos para el paciente. Los beneficios son muy diversos, a destacar que mejora el funcionamiento del sistema músculo-esquelético.

Para adaptar los ejercicios a las diferentes fases de recuperación, por las cuales pasará el paciente, se debe diseñar un plan de ejercicio terapéutico que periódicamente será revisado por el fisioterapeuta o especialista. Se realizarán las modificaciones que sean necesarias para conseguir una progresión que estimule la motivación del paciente, lo que favorecerá la constancia en la realización de los ejercicios [59]. Es aquí, donde la tecnología podría jugar un gran papel, ya que podría ayudar a motivar al paciente y a realizar los ejercicios de forma más precisa. Además, se podría evaluar el proceso forma mucho más exhaustiva que con métodos tradicionales, lo que ayudaría al terapeuta a tomar mejores decisiones durante el proceso de recuperación.

Los procesos de rehabilitación son muy comunes a lo largo de la vida de una persona, y estos se deben adaptar a la lesión en cuestión y a las circunstancias de la persona que se pretende rehabilitar. En el caso de las personas que sufren algún tipo de discapacidad, el proceso de rehabilitación tendría que ser adaptado de forma mucho más específica, dada la complejidad añadida. Este es el caso de las personas con discapacidad visual, que requieren de un programa de rehabilitación centrado en diversas disciplinas, cuyo objetivo será que el paciente logre desarrollar diferentes actividades lo más independiente posible, mejorando consigo su calidad de vida. Estos programas se enmarcan dentro de diferentes áreas como son: apoyo psicosocial, técnicas para la vida diaria, desarrollo de la habilidades cognitivas, elementos de la comunicación, orientación y movilidad, desarrollo de competencias para la lecto-escritura adaptada braille, desarrollo de habilidades para el cálculo en la vida diaria y acondicionamiento físico [33].

La ceguera es uno de los problemas de salud más incapacitantes en el ser humano. Se estima que 180 millones de personas en todo el mundo sufren una discapacidad visual, entre 40 y 45 millones son ciegos y 135 millones restante sufren algún tipo de deficiencia visual, según la Organización Mundial de la Salud. Más del 50 % de la discapacidad visual en el mundo es causada por la ceguera no tratable y no curable, y más del 75 % de la discapacidad visual en los países de ingresos bajos es causada por la ceguera no tratable. La mayoría de la población mundial no conoce las causas de la discapacidad visual, ni la manera de prevenirla y tratarla. El desafío de la discapacidad visual en el mundo es mayor en los países de ingresos bajos, donde el acceso a servicios de salud es limitado y la población no tiene acceso a servicios de salud oportunos para la discapacidad visual. Aunque el resto de países también tienen problemas para garantizar el acceso a servicios de salud y atención oportuna para la discapacidad visual [76].

La demanda social de atención a este tipo de personas, con necesidades especiales o cierto grado de dependencia, es cada vez mayor en nuestra sociedad constituyendo uno de los principales retos de la sociedad actual. Los sistemas sanitarios y de atención social, han habilitado una serie de recursos de hospitales, centros de rehabilitación y asistencia domiciliaria, entre otros, para hacer frente a estos problemas. El objetivo de estos es dotar a la persona del mayor bienestar y la mayor autonomía posible en el desempeño de sus tareas, desde las básicas de la vida diaria hasta las propias de la vida social, tales como la movilidad y la comunicación. La necesidad de hacer posible la interacción y comunicación de personas con discapacidad ha llevado a diferentes grupos de investigación a investigar y desarrollar interfaces persona computador basadas en nuevas formas de interacción [22].

Tal como se acaba de exponer, en el caso de personas con problemas de visión existe un amplio espectro de áreas de rehabilitación o mejora de la condición física, no sólo desde el punto de vista de la recuperación en lesiones articulares. Como el caso de la orientación y movilidad, que se puede definir como un proceso en el cuál con el uso de los sentidos restantes el paciente debe establecer su posición en relación a los objetos que le rodean realizando algún tipo de movimiento. Los movimientos están presentes en todas las actividades del día a día de cualquier persona, por ello, es necesario adaptarse y mejorar en este ámbito. Las personas con limitaciones visuales que desarrollan este tipo de destrezas pueden obtener un autodomínio e independencia que mejorarán su calidad de vida.

La rehabilitación convencional se realiza en clínicas especializadas con la supervisión presencial del especialista pertinente. Esta metodología conlleva una serie de problemas para personas que tienen dificultades para desplazarse a la clínica en cuestión, sobre todo si el paciente es una persona con discapacidad visual. La problemática expuesta tradicionalmente se ha solucionado desplazando al paciente por medio de un familiar o ambulancia al centro médico donde se deba realizar la actividad de rehabilitación. Esta solución conlleva un serie de problemas como son la falta de profesionales para atender presencialmente a todos los pacientes o los recursos limitados de movilidad del sector sanitario. Además, si nos referimos al contexto específico de este trabajo, las personas con problemas de visión tiene aún mayores dificultades para desplazarse y recibir rehabilitación. Dado este contexto se deberían pensar e implementar nuevas soluciones que permitan hacer un uso mas eficiente de los recursos sanitarios.

Además, la pandemia causada por el *COVID-19* ha obligado a modificar la forma de trabajar en la mayoría de ámbitos laborales, sobre todo el sanitario que se vio colapsado. A partir del sábado 14 de Marzo de 2020, el día que se publicó el Real Decreto de estado de alarma, hubo un punto de inflexión en el sistema sanitario español, obligando a cambiar muchos de los procesos convencionales seguidos hasta la fecha [6]. Durante el confinamiento vivido se han establecido nuevas prioridades de tratamiento. Además, de un cambio muy significativo en la forma de tratar las consultas o incidencias, estableciéndolas de forma no presencial como opción por defecto. Las intervenciones en rehabilitación se suelen realizar en diferentes áreas físicas, normalmente zonas comunes donde los pacientes ingresados en el hospital se desplazan para recibir su tratamiento rehabilitador. Este tipo de situaciones se tuvieron que ir cancelando o adaptando durante el estado de alarma. Si durante esta situación se hubiera tenido la posibilidad de realizar tratamientos de rehabilitación a distancia las lesiones articulares seguramente no habrían empeorado. Sin embargo, el hecho de no haber realizado rehabilitación en un momento determinado puede hacer que los pacientes tengan que enfrentarse a secuelas más graves o de más duración [50].

En el año 2005, la Organización Mundial de la Salud (OMS) en [65], advertía de que los modelos asistenciales de los sistemas sanitarios eran ineficientes en la asistencia de procesos crónicos. En el año 2014, la OMS publica el "Proyecto de acción mundial de la OMS sobre discapacidad 2014-2021". En cual plantea acciones concretas para mejorar la cobertura, la eficacia y la eficiencia de los servicios sanitarios [75]. Sin embargo, la pandemia por *COVID-19* está modificando el perfil de discapacidad descrito por la OMS por varias razones: el aumento de secuelas de la enfermedad por *COVID-19*, el incremento de déficit a nivel funcional en pacientes que no han recibido suficiente atención y el impacto del descenso en la actividad física por el confinamiento en personas de edad avanzada, o poco funcionales como los discapacitados. Este nuevo escenario supone para la disciplina de la medicina física y rehabilitación un nuevo reto y una oportunidad de mejora. Un nuevo reto para implementar soluciones organizativas innovadoras que den una respuesta efectiva y eficiente a la epidemia de discapacidad que ya padecíamos que ha sido agravada.

La implementación de servicios de rehabilitación a distancia mediante tecnologías de la información y comunicación (TIC), hasta la fecha, tiene una tasa de utilización baja en la práctica asistencial, a excepción del tratamiento de déficits cognitivos iniciales o en rehabilitación neurológica. En la situación actual, es esencial implementar consultas que no se requiera la presencialidad del paciente,

salvo que el proceso asistencial lo requiera. En el campo de la telemedicina existe multitud de aplicaciones prácticas que se podrían implementar, como realizar consultas telemáticas entre profesionales para ayudar en la toma de decisiones, monitorizar a distancia el progreso terapéutico del paciente o practicar ejercicios de rehabilitación supervisados o guiados por algún tipo de tecnología.

Disciplinas de la medicina como la fisioterapia, la terapia ocupacional o la logopedia también requieren transformaciones que reinventen la relación entre el terapeuta y el paciente, para poder atender al mayor número de pacientes posibles por terapeuta. Las soluciones tecnológicas que realizan una monitorización y evalúan los progresos a distancia en materia de rehabilitación son uno de los modelos de negocio en expansión en la industria biomédica. La rehabilitación de procesos músculo-esqueléticos, dada su elevada prevalencia, se puede beneficiar del empleo de estas herramientas. A día de hoy, con la movilidad restringida en todo el mundo, la telemedicina y telerrehabilitación pueden solventar algunas necesidades críticas en rehabilitación en el abordaje de procesos músculo-esqueléticos o cardiorrespiratorios, entre otros. La telemedicina y la telerrehabilitación son dos modelos de negocio que pueden ser utilizados en el abordaje de procesos músculo-esqueléticos, entre otros. La telemedicina es una disciplina que tiene como objetivo la realización de pruebas diagnósticas y terapéuticas a través de medios electrónicos y sistemas informáticos. La telerrehabilitación es una disciplina que tiene como objetivo el desarrollo de programas de rehabilitación a distancia, con el fin de mejorar la calidad de vida del paciente [60].

Desde el prisma económico, la pandemia ha dejado una situación bastante complicada, ya que todas las previsiones apuntan a que se va a producir un retroceso muy severo en el Producto Interior Bruto (PIB) del país. Además, se prevé un aumento sustancial de la deuda pública, pudiendo llegar a situarse en el 131,6 % del PIB [35]. Estos datos dejan entrever que se va a necesitar hacer un ajuste presupuestario para recuperar el nivel económico anterior a la pandemia.

Teniendo en cuenta las nuevas medidas de salud pública y las previsiones económicas, queda de manifiesto la necesidad de adaptarse a estas circunstancias. En la actualidad ya existen soluciones tecnológicas que ayudan en el proceso de rehabilitación, pero la mayoría de ellas se utilizan de forma presencial. En cambio, una solución que posibilite la rehabilitación telemática aportaría gran cantidad de beneficios al paciente y al terapeuta. Desde el punto de vista del paciente, se podrían destacar algunos beneficios como la libertad de realizar el proceso de rehabilitación cuando mejor convenga y la precisión a la hora de medir como se realizan los distintos movimientos. Para el terapeuta también supone varias ventajas, como la capacidad de atender a más pacientes o poder ver de forma precisa cómo se están realizando los ejercicios, dado que las distintas ejecuciones del paciente se almacenarán para estudiarlas posteriormente. Además de los potenciales beneficios descritos, también se podrían reducir costes económicos en el sistema sanitario.

El presente proyecto propone una solución tecnológica que podría ser de ayuda para realizar ciertos tratamientos de rehabilitación desde la comodidad del hogar. Este tipo de solución tecnológica es muy novedosa, dado que en la actualidad no existen demasiados sistemas que permitan realizar ejercicios de rehabilitación de forma telemática, sobre todo enfocados a personas invidentes. Con el sistema se podrán crear distintos tipos de ejercicios de rehabilitación, pautados y definidos por el terapeuta. Posteriormente el paciente realizaría estos ejercicios siguiendo las indicaciones que le de el sistema en tiempo real. El prototipo está ideado para personas con algún tipo de discapacidad visual, es decir, que no se requiera del uso de la vista durante su utilización. Por lo tanto, el paciente será guiado en la realización de los ejercicios mediante retroalimentación háptica, gracias a un dispositivo compuesto de sensores que guiará al usuario con vibraciones indicando la dirección que debe seguir la articulación para realizar correctamente el movimiento. Además, el sistema también utilizará retroalimentación acústica como apoyo en el guiado y para motivar al paciente durante la realización del ejercicio.

## 1.1. ESTRUCTURA DEL DOCUMENTO

El presente Trabajo de Fin de Grado está compuesto de los siguientes capítulos:

- **Capítulo 1: Introducción**

Se trata del capítulo actual del documento, en el cual se pretende presentar el contexto de este trabajo y la problemática existente que se pretende solucionar.

- **Capítulo 2: Objetivos**

El capítulo de los objetivos se centra en explicar detalladamente el objetivo principal del trabajo. El objetivo principal estará dividido en una serie de hitos u objetivos parciales que consiguiéndolos se logrará el objetivo principal.

- **Capítulo 3: Estado del arte**

El estado del arte del proyecto se trata de un estudio detallado de las soluciones existentes en el contexto actual del trabajo. Este estudio servirá de ayuda para conseguir los distintos objetivos definidos, creando una base teórica que ayudará a tomar mejores decisiones de diseño. En el caso del presente proyecto, se realizará un estudio detallado de distintos entornos de rehabilitación, así como de técnicas que serán de utilidad en el desarrollo del proyecto.

- **Capítulo 4: Metodología**

En este capítulo se presenta la metodología de desarrollo aplicada en el proceso de desarrollo del proyecto, además de las distintas herramientas y recursos que han sido de utilidad para llevarlo a cabo.

- **Capítulo 5: Arquitectura**

El capítulo de la arquitectura tiene como objetivo hacer una descripción técnica del sistema resultante del proyecto. Para ello, se explicarán los distintos problemas encontrados y qué soluciones se han elegido como las convenientes, explicando el por qué de estas decisiones.

- **Capítulo 6: Resultados**

En este capítulo se expondrá el sistema resultante y la experimentación llevada a cabo para comprobar la eficacia del dispositivo diseñado.

- **Capítulo 7: Conclusión**

Finalmente, en la conclusión se realizará una evaluación de los resultados obtenidos para comprobar si han cumplido los objetivos propuestos, justificando las competencias adquiridas y proponiendo potenciales líneas de trabajo futuro.



# OBJETIVOS

---

En este capítulo se va a mencionar el objetivo que pretende cumplir el desarrollo del proyecto. Además, de los diferentes hitos u objetivos parciales que se deben conseguir para lograr el objetivo principal en sí mismo.

## 2.1. OBJETIVO GENERAL

El objetivo general de este proyecto consiste en desarrollar un sistema para que personas con problemas de visión puedan realizar remotamente la rehabilitación de lesiones articulares.

Para llevar a cabo este objetivo, el sistema deberá permitir al paciente realizar distintos ejercicios de recuperación pautados por el especialista médico pertinente. Además, éste último deberá tener la capacidad de crear los diferentes ejercicios y poder realizar un seguimiento de la ejecución del paciente de manera adecuada.

## 2.2. OBJETIVOS ESPECÍFICOS

Una vez definido el objetivo general que se pretende cumplir, se van a describir una serie de objetivos específicos que cumpliéndolos se logrará el objetivo principal del proyecto descrito anteriormente. Los objetivos específicos son los siguientes:

### **Desarrollar un sistema de definición de ejercicios por parte del rehabilitador**

El sistema resultante de este trabajo debe ayudar a la rehabilitación a distancia mediante la realización de ejercicios terapéuticos. Para ello, se necesita que el terapeuta pueda definir estos ejercicios en el sistema, para que posteriormente el paciente los pueda realizar. Los ejercicios deben estar definidos de tal forma que la persona que posteriormente los va a realizar no tenga que estar en el mismo lugar que el terapeuta a la hora de crearlos. Por lo tanto, el sistema debe tener una independencia espacial para que el ejercicio creado pueda ser realizado en un lugar totalmente distinto. Para conseguir esto, será necesario elegir un dispositivo de tracking del cuerpo, que permita guardar información sobre el movimiento de las articulaciones involucradas en el ejercicio. Este dispositivo será utilizado tanto en la creación como en la ejecución de los ejercicios, ya que gracias a él se podrá lograr la independencia espacial comentada.

### **Realizar un análisis del movimiento en tiempo real**

Durante la realización de los distintos ejercicios definidos en el sistema se debe realizar un análisis y un procesamiento de los mismos. Dependiendo de como se esté realizando el movimiento el sistema enviará unas órdenes u otras con el objetivo de que el paciente realice el movimiento correctamente. Este procesamiento debe ser realizado por el computador que interactuará con el dispositivo de

tracking para captar la información del movimiento, y una vez obtenida dicha información la procesará y la analizará para enviar las instrucciones correspondientes al dispositivo de guiado que utilice el paciente.

### **Desarrollar un dispositivo el guiado que permita la correcta realización de ejercicios terapéuticos**

Los ejercicios definidos deben ser realizados por una persona con discapacidad visual, esto restringe mucho las posibilidades de guiar al paciente en la realización de los ejercicios. Por lo tanto, no se pueden usar dispositivos convencionales o de realidad virtual, ya que todos ellos dependen del sentido óptico para su uso. Para que la persona pueda realizar los ejercicios correctamente se necesita de la creación de un dispositivo que guíe al paciente durante su realización y que no depende del sentido de la vista. Por ello, este objetivo pretende diseñar un dispositivo que sea capaz de transmitir de manera háptica la información necesaria para que el paciente realice el ejercicio correctamente.

### **Aplicación de técnicas de gamificación para mejorar la motivación del paciente**

La motivación del paciente al realizar el tratamiento de rehabilitación es primordial para lograr una pronta recuperación. Para lograr esto se aplicarán técnicas de gamificación que persuadan al paciente a realizar los ejercicios pautados por el terapeuta con la recurrencia estipulada. En el caso del presente proyecto, esto cobra más importancia, porque la rehabilitación es telemática y el terapeuta no supervisará físicamente las ejecuciones de los ejercicios. Por lo tanto, se debe lograr que el paciente realice el tratamiento por iniciativa propia y evitar en la medida de lo posible el abandono del tratamiento. Las técnicas de gamificación a emplear deben estar enfocadas al tipo de paciente que se va tratar, es decir, sería incorrecto aplicar técnicas que sean visuales, ya que el paciente objetivo tiene discapacidad visual. En cambio, técnicas de modalidad acústica si podrían ser más convenientes.

### **Desarrollar un sistema de seguimiento para ver los resultados de la ejecución de los distintos ejercicios**

El terapeuta debe poder realizar un seguimiento del tratamiento del paciente, para que pueda ser adaptado durante el proceso de recuperación. Por ello, es de vital importancia que las distintas ejecuciones de los ejercicios sean persistentes en el sistema, para que se puedan analizar una vez se hayan realizados los ejercicios. De esta forma, se podrá ver la evolución que tenga el paciente durante la rehabilitación.

## ESTADO DEL ARTE

---

En este capítulo se van a presentar conceptos teóricos considerados relevantes para el desarrollo de este proyecto. Además, se van a estudiar trabajos de diversa índole, pero relacionados en mayor o menor medida con el proyecto y que servirán como guía para lograr los objetivos descritos. En primer lugar, se van a tratar conceptos relacionados con los juegos serios y la implicación de estos en el proceso de rehabilitación. En segundo lugar, se van a estudiar algunos trabajos considerados como relevantes en el ámbito de las tecnologías de rehabilitación. A continuación, se tratarán proyectos relacionados más específicamente con la discapacidad visual, para terminar presentando los dispositivos que se utilizarán en la solución propuesta.

### 3.1. JUEGOS SERIOS Y REHABILITACIÓN

Los procesos o técnicas de rehabilitación dependen mucho del paciente y la lesión en cuestión. En el caso de los pacientes con algún tipo de discapacidad, como en el caso de que algunos de los sentidos sufra algún tipo de trastorno, suele provocar que el proceso de desarrollo físico y cognitivo sea más lento. Por ello, las sesiones de rehabilitación suelen ser tediosas y muy poco satisfactorias para el paciente, debido a que la mayoría de estas sesiones son repetitivas, conllevando aburrimiento o cansancio temprano del paciente.

Los juegos serios se originan para dar respuesta a la necesidad de ofrecer herramientas con las cuales se puedan realizar procesos de rehabilitación a través de medios lúdicos tecnológicos, ya sea de forma digital o tangible. De esta forma, el paciente podrá hacer frente a los inconvenientes expuestos anteriormente [69]. Cabe destacar la distinción entre un juego serio y uno convencional, el primer tipo está enfocado en lograr un cambio en el jugador, ya sea aprender una habilidad o rehabilitar una zona corporal como es el caso que nos ocupa en este proyecto. El segundo tipo de juego, conocido tradicionalmente, tiene un objetivo meramente de entretenimiento.

#### 3.1.1. Juegos serios y metodologías de diseño

La primera aparición del término “juego serio” proviene de la época de los 70, donde no existían demasiadas formas de entretenimiento u ocio, siendo los juegos de mesa una de las formas más destacadas. Por esa razón, se impulsó la creación de juegos que involucrarán procesos más allá del mero entretenimiento, como la formación o el aprendizaje. Clark Abt, puntualizó el interés hacia los juegos serios en su libro llamado *Serious Game* [1], “*nos interesan los juegos serios porque tienen un propósito educativo explícito y cuidadosamente planeado, y porque no están pensados para ser jugados únicamente por diversión*”.

Gracias a la aparición de multitud de tipos de juegos serios, cada uno con unos objetivos específicos a lo largo de los años han aparecido distintas metodologías de diseño. En 2008 aparece una metodología apodada como *EMERGO*, que ofrece una guía de desarrollo de juegos basada en escenarios en los cuales se modelan situaciones del mundo real [61]. En 2010 se propone el desarrollo de herramientas

desde un punto de vista teórico y técnico enfocadas en enseñar competencias profesionales [53]. La investigadora Natalia Padilla, en el año 2011, propone una metodología enfocada en una arquitectura de análisis del aprendizaje colaborativo, que permite modelar los distintos elementos de un juego serio [67]. En 2015, se presenta una metodología para la implementación de un entorno lúdico desarrollado con el fin de tratar a pacientes con discapacidad mental [21]. En 2016 se presentó una metodología de diseño de juegos serios centrada en la rehabilitación del lenguaje y la comunicación de niños con discapacidad acústica [18]. Por último, destacar el artículo [74] que realiza un estudio en profundidad de diferentes juegos serios en el ámbito de la salud que destacan en la actualidad.

### 3.1.2. Taxonomía de juegos serios

Con el objetivo de clasificar las diferentes variedades de juegos serios se van a definir una serie de características consideradas como relevantes según [46].

#### Actividad

La primera característica a destacar sería el tipo de actividad que el jugador realiza según lo requiera el juego en cuestión. Se podría decir que esta es la función que realiza el jugador como respuesta al juego. Los tipos de actividad podrían ser el esfuerzo físico en juegos para el bienestar personal como en [15, 79], o juegos para mejorar una enfermedad como la obesidad en niños [77]. El tipo de actividad podría ser también el fisiológico, como en los juegos enfocados en la rehabilitación [17], o la detección del estado de la salud de las personas [55]. Otro tipo de actividad relevante podría ser mental, como en juegos con fines educativos [78] o de comunicación interpersonal [42].

#### Modalidad sensorial

La modalidad sensorial también se podría decir que es un criterio importante, ya que es aquí donde se define el canal por el cual la información se comunica desde el computador pertinente al jugador. Esto caracteriza las modalidades sensoriales que experimentarán los jugadores al interactuar con el juego. Las modalidades más utilizadas son la óptica, la acústica, la háptica y, en menor medida, la olfativa.

La modalidad debe ser elegida en beneficio del propósito del juego y la experiencia del usuario. En [25] se realizó un estudio en el cual se presentaba el mismo juego a dos grupos distintos de personas, en el primero se utilizaba una pantalla para que el jugador pudiese comprobar su desempeño y en el segundo grupo no se realizaba este tipo de retroalimentación. Los resultados del estudio en cuestión fueron que el uso de una pantalla que proporciona información al usuario fue un factor determinante en el éxito de este juego serio. En [86] se comprobó que la inclusión de música en el juego ayudaba a aumentar la motivación de los usuarios para practicar el juego. En el caso de la retroalimentación háptica en juegos serios, ésta tiene el potencial de mejorar el aprendizaje haciendo uso de la percepción táctil de los objetos [4] y de hacer más realista juegos de rehabilitación en personas que han padecido un ictus [66]. En el caso de la modalidad olfativa, existen algunas intentos como una aplicación para comprobar el uso del sentido del olfato en terapia [23], aunque esta modalidad no suele ser muy utilizada.

#### Estilo de interacción

El estilo de interacción define de qué forma o con qué elementos se va a realizar la interacción con el usuario dentro de un juego serio. En los juegos tradicionales es común el uso de periféricos como el teclado, el ratón o el *joystick*. Recientemente, también es común el uso de interfaces que permiten una interacción mediante el seguimiento de movimientos corporales. La elección del estilo de interacción resulta de vital importancia a la hora de que un juego serio cumpla su cometido.

En [85] se pudo comprobar como la elección de la interfaz, con la cual se interactúa con el jugador, es determinante, ya que mediante el uso de sensores se aseguraban que los jugadores movieran las articulaciones que el juego requería. De esta forma el jugador realizaba la actividad de la forma que se debía. En el ámbito concreto de la rehabilitación, se podría decir que es más primordial aún la forma de interactuar con el paciente, ya que el sistema debe asegurarse que la rehabilitación se realiza de la forma correcta, en caso contrario el paciente podría empeorar.

### Ambiente

El ambiente, en el contexto de los juegos serios, se podría definir como un conjunto de características que determinan ciertos aspectos relevantes del juego a desarrollar, como son el espacio, el entorno de realidad, la ubicación, la movilidad o la presencia social. Dependiendo del juego en cuestión unos aspectos serán más relevantes que otros, en este caso se ha realizado un estudio de los aspectos más relacionados con el presente trabajo.

El espacio se puede tratar desde un punto de vista bidimensional o tridimensional. La elección dependerá del tipo de actividad que debe realizar el jugador. Por ejemplo, si el jugador debe realizar movimientos en dos ejes dimensionales se elegiría un espacio bidimensional, en caso de que se requieran movimientos sobre los tres ejes del espacio se optaría por un espacio tridimensional. El entorno de realidad se podría definir como el conjunto de elementos con el cuál el jugador debe interactuar y si estos elementos son generados virtualmente o no, es decir, se refiere a conceptos como el de realidad virtual o realidad aumentada.

### Área de aplicación

El área de aplicación de juego serio indica el dominio de aplicación en el que se engloba. Este aspecto está estrechamente relacionado con el objetivo del juego. Las áreas de aplicación podrían ser muy diversas, incluso el mismo juego podría estar englobado en varias. Aunque, no debe perderse de vista el objetivo general de un juego serio, como es el de no ceñirse exclusivamente a fines de entretenimiento. Algunas áreas o dominios de aplicación relevantes podrían ser la educación, el cuidado de la salud, el bienestar o las relaciones interpersonales.

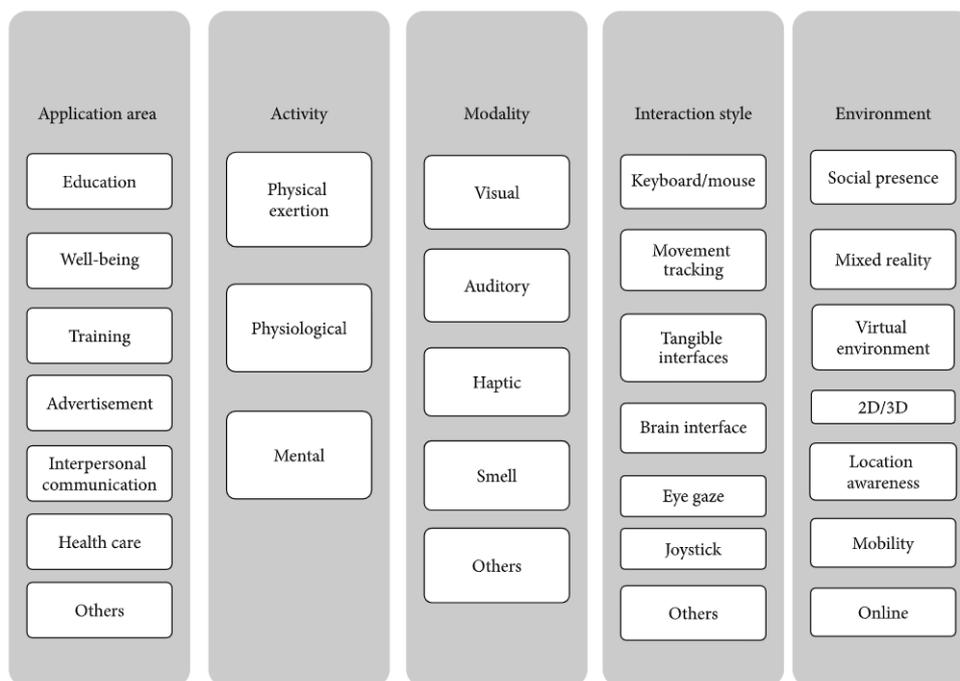


Figura 3.1: Taxonomía de juegos serios [46]

### 3.1.3. Gamificación

Uno de los principales objetivos de los juegos serios es que el usuario utilice el juego para su cometido, ya sea la rehabilitación de una lesión, el aprendizaje de una habilidad o cualquier otro. Esto se logra si el jugador tiene la suficiente motivación como para realizar la actividad de la manera correcta y el tiempo suficiente. Es aquí donde tiene cabida el concepto de gamificación, que consiste en el uso de principios de diseño, el uso de la mecánica y otros elementos propios de los videojuegos tradicionales. El objetivo es aprovechar la predisposición psicológica de las personas a participar en juegos y las técnicas propias del juego para motivar y mejorar el comportamiento de los jugadores [40].

Las principales técnicas de gamificación utilizan recompensas para los jugadores que logran las tareas o las actividades pertinentes. Otros jugadores o proporcionar tablas de clasificación son formas de alentar a los jugadores a competir. Hacer que las recompensas conseguidas sean visibles para, incrementando la práctica de las actividades objetivo. Algunas técnicas destacables podrían ser la utilización de algún tipo de puntuación para evaluar el rendimiento del usuario [87], insignias o representaciones visuales de logros [84], entre otras técnicas.

*NikeFuel* es un sistema de seguimiento de la actividad física desarrollada por la empresa *Nike, Inc.* Este sistema propició que en la comunidad de la compañía, *Nike+*, dos millones de usuarios realizaran ejercicio físico, gracias a que proporcionaba gran cantidad de estadísticas de la actividad física realizada por el usuario. Además, existía la posibilidad de visionar las estadísticas del resto de usuarios. Cuando completabas ciertos hitos particulares la plataforma otorgaba insignias que se podían exhibir ante la comunidad como si de un escaparate de trofeos se tratase. Con esta serie de técnicas, la empresa americana consiguió generar una especie de competencia entre usuarios, convirtiendo la práctica deportiva en un "juego". Es decir, se aplicaron técnicas de gamificación para influir positivamente en la motivación, la productividad y el comportamiento de los usuarios [12].

En [19] plantean la implementación de un juego serio para la rehabilitación del equilibrio en pacientes que han sufrido parálisis cerebral. En la implementación tiene mucha importancia la motivación del paciente, para evitar el abandono de la rehabilitación. Los usuarios del juego en cuestión eran pacientes que habían abandonado la terapia con anterioridad y los resultados obtenidos fueron una mejora en la motivación a la hora de realizar la rehabilitación, y en consecuencia, una mejora del equilibrio.

Otra contribución a destacar es el artículo [39], en el cual se trata a la gamificación como estrategia pedagógica, con el objetivo de mejorar el aprendizaje un segundo idioma. El objetivo era incrementar la participación utilizando técnicas como puntuación, tablas de clasificación o comentarios inmediatos que informaban del progreso. Es decir, se trataba de medir, evaluar e informar de la mejor forma posible el progreso, de esta forma se motivaba al usuario a seguir practicando para lograr el objetivo, en este caso mejorar su conocimiento sobre un segundo idioma. Los resultados fueron que los estudiantes incrementaron sus calificaciones en un 9% y el número de alumnos que suspendieron disminuyó un 16%.

## 3.2. SISTEMAS DE AYUDA PARA PERSONAS CON DISCAPACIDAD VISUAL

La ceguera impone numerosas restricciones a la capacidad de moverse y control sobre el cuerpo a la persona que la sufre, como resultado estas personas pueden llegar a ser muy dependientes e incapaces de moverse en entornos cotidianos [10].

La tecnología de asistencia para personas con discapacidad visual es vía de investigación que está adquiriendo gran importancia debido al relevante impacto social en la creciente población de personas invidentes y envejecidas que tenemos en la actualidad. En [11] se muestra el crecimiento que han tenido las publicaciones relacionadas con este ámbito, pasando de menos de 50 publicaciones

anuales en la década de 1990 a cerca de 400 publicaciones de carácter científico en el año 2014. Este estudio afirma que se espera que la tecnología de asistencia para personas con deficiencias visuales aumente rápidamente e impacte positivamente en las vidas de estas personas. En 2015 se publicó [27], en el cuál se evidenciaba un aumento en las aplicaciones para móviles para asistir o ayudar a personas con problemas de visión. El artículo [34], publicado en 2017, presenta una encuesta comparativa de los dispositivos de asistencia para personas con discapacidad visual, con el objetivo de comprobar el progreso de la tecnología de asistencia para este grupo de personas. En la actualidad, según [81] se podría observar un incremento de desarrollos tecnológicos para personas con esta discapacidad enfocados a captar más información del entorno de la persona, con el objetivo de aumentar o mejorar la movilidad de estas personas.

### Sistemas de ayuda a la orientación para personas ciegas

En [37] se hacen estudio de los sistemas desarrollados para ayudar a las personas con discapacidad visual a orientarse y evitar obstáculos, para que consigan moverse de forma segura en un medio físico.

Uno de los sistemas creados, apodado como *Tom Pouce*, es un sistema que analiza la proximidad mediante infrarrojos. El dispositivo creado tiene la intención de detectar obstáculos que se encuentre en frente del bastón que suele usar una persona invidente. Un ejemplo de detección sería un panel de gran tamaño con postes de soporte muy separados. El dispositivo emitiría varios haces de luz infrarrojos cercanos en diferentes direcciones y con disparidad de potencias de emisión, con el objetivo de cubrir de la mejor forma el espacio. Los detectores reciben la luz rebotada por los obstáculos, detectándolos y avisando a la persona que lleva el bastón, mediante un retroalimentación háptica en el dedo meñique.



Figura 3.2: Primera versión comercial de *Tom Pouce* [37]

*Teletact* es un telémetro láser de mano, es decir, un dispositivo capaz de medir distancias remotamente. La distancia hasta el primer obstáculo que se encuentre el rayo láser se mide con aproximadamente un 1 % de precisión en un rango de entre 10 cm a 10 m. El dispositivo requiere realizar un escaneo del entorno para detectar un obstáculo. Para comunicar la información al usuario se utiliza una retroalimentación háptica y acústica. En la interfaz háptica se utilizan dos dispositivos de vibración ubicados en dos dedos distintos, para el primero de ellos se codifican distancias de entre 3 y 6 metros; y para el segundo distancias entre 1,5 y 3 metros. En la interfaz acústica se codifican distancias de hasta 15 metros mediante 28 notas musicales de distinto tono.

*Geotact* es un sensor GPS que se diferencia con otros dispositivos convencionales en la forma en la cual la información es transmitida al usuario. No se utiliza el método tradicional de orientación, es decir, con comandos de voz como girar a la izquierda o a la derecha. En su lugar, se da la dirección imaginando un reloj tradicional, en el cual las distintas horas representan las direcciones a transmitir al usuario. La distancia se transmitiría en metros combinada con el mensaje de la dirección, por ejemplo dos en punto 160 metros.



Figura 3.3: Segunda versión comercial de *Teletact* [37]

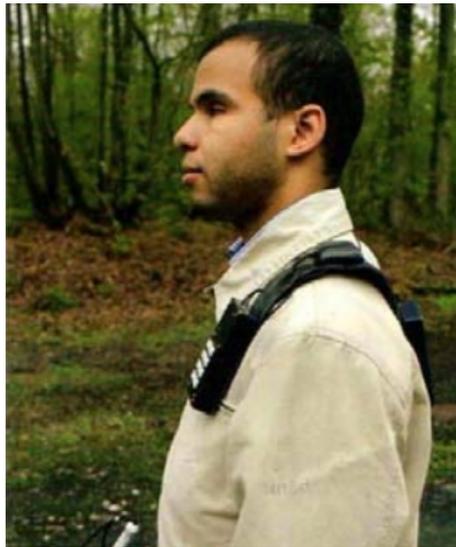


Figura 3.4: Primera versión de *Geotact* [37]

La perspectiva rehabilitadora de estos sistemas se centra en mejorar la movilidad teniendo en cuenta la fluidez, la anticipación y la seguridad, manteniendo el máximo potencial de orientación. Aunque se usen este tipo de dispositivos, hay que tener en cuenta que el usuario debe aprender las bases de movilidad con el bastón. Una vez aprendidas se pueden introducir los dispositivos a modo de entrenamiento de movilidad, consiguiendo mejores estrategias para que el usuario pueda moverse de la mejor forma posible dentro de un entorno, estimulando los sentidos no visuales de la persona. Los resultados obtenidos con los dispositivos han sido que los mejores usuarios alcanzan una movilidad muy similar a la de una persona vidente, cuando ésta pretende encontrar un camino en lugares como un centro comercial muy concurrido.

### **Sistema de rehabilitación que integra un entorno virtual para mejorar las habilidades de orientación y movilidad de las personas ciegas, *BlindAid***

En [48] se presenta la integración de un entorno virtual dentro de un programa de rehabilitación para mejorar la orientación y la movilidad de personas con algún tipo de discapacidad visual. *BlindAid* permite al usuario interactuar con diferentes estructuras y objetos virtuales a través de retroalimentación acústica y háptica. El objetivo de este artículo es comprobar si el uso del sistema resultante junto con un programa rehabilitación puede capacitar al paciente en actividades funcionales que requieran cierto grado de movilidad. Los resultados obtenidos demostraron que los pacientes sometidos al estudio muestran potenciales beneficios en movilidad y orientación al hacer uso de un entorno virtual. Se pudieron observar como las habilidades adquiridas durante el proceso de rehabilitación fueron extrapoladas a entornos reales de la vida cotidiana como la movilidad dentro de la residencia habitual

del paciente.



Figura 3.5: Interfaz de usuario del sistema *BlindAid* [48]

### Sistema de percepción espacial para personas ciegas

El objetivo del proyecto de investigación [41] es ofrecer a las personas invidentes una forma de obtener más información sobre su entorno inmediato de la que obtienen con los métodos tradicionales. Para ello, se ha desarrollado un dispositivo que captura la forma y el volumen del espacio situado en frente de la persona ciega en cuestión. Una vez recibida la información, esta se envía en tiempo real mediante un mapa de sonidos. El efecto producido por el mapa de sonidos en la persona sería como si los objetos que forman el entorno estuvieran cubiertos de fuentes de sonido que emiten señales continuamente. Esta nueva forma de recibir información del espacio tridimensional mediante retroalimentación acústica únicamente, mejora el conocimiento del usuario de su interacción con el entorno. De esta forma, se brinda a las personas invidentes de más independencia de orientación y movilidad.

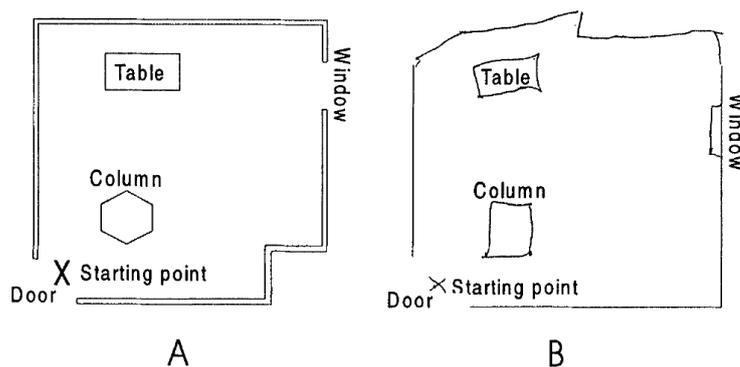


Figura 3.6: Entorno experimental (A) y la representación realizada por una persona invidente (B) [41]

En la Figura 3.6 se puede observar la representación que realiza una persona invidente de un entorno experimental haciendo uso del prototipo desarrollado, sin retroalimentación háptica del entorno. Aunque, la percepción sonora ha sido acompañada de una evocación visual simultánea utilizando puntos de luz puntiformes, ubicados en las fuentes sonoras virtuales provenientes de los objetos representados. Estas percepciones visuales son recibidas por la persona ciega como fosfenos<sup>1</sup>,

<sup>1</sup>Un fosfeno es un fenómeno descrito como la sensación de ver manchas luminosas. Esta sensación suele ser causa de la estimulación mecánica, eléctrica o magnética de la retina o corteza visual.

ayudando en gran medida a representar visualmente el entorno sin la capacidad de verlo.

Los resultados obtenidos por este proyecto abren una línea de investigación en el campo de la rehabilitación sensorial, ya que allana el camino para el desarrollo de aplicaciones que apliquen esta misma técnica y mejoren el desarrollo psicomotor de personas de edad temprana con ceguera congénita.

### Sistema de ayuda a invidentes basado en cámaras de profundidad

El proyecto [5] tiene como objetivo desarrollar un sistema de ayuda para personas invidentes aplicando técnicas de visión artificial. El algoritmo propuesto permite que el usuario obtenga información de los objetos que le rodean y pudiendo así detectarlos sin necesidad del sentido óptico. La información será transmitida al usuario mediante una técnica de localización sonora denominada como técnica binaural. El algoritmo utiliza el framework ROS, con el cual se puede obtener la información necesaria del dispositivo encargado de detectar tridimensionalmente los objetos del entorno. El dispositivo de tracking utilizando en el proyecto es la primera versión de *Kinect*.

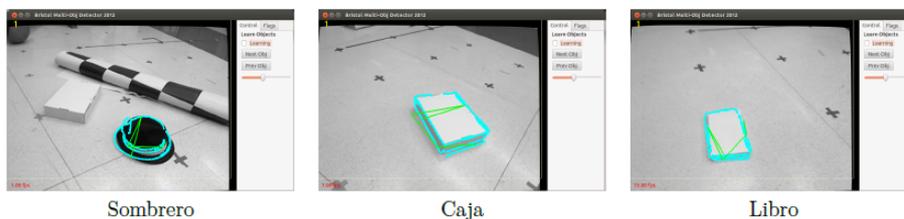


Figura 3.7: Ejemplo de detección de objetos [5]

Los trabajos mencionados se centran en rehabilitar la movilidad y la orientación de las personas con discapacidad visual. Se pueden extraer conclusiones que sirvan de utilidad para el presente trabajo, como es la manera de comunicarse con la persona invidente, ya que en la mayoría de casos se trata de explotar otras modalidades sensoriales de la persona, acústica y háptica principalmente.

## 3.3. SISTEMAS DE REHABILITACIÓN

Las tecnologías de rehabilitación se originaron justo después de la Segunda Guerra Mundial, porque se produjeron diferentes trastornos que afectaron al sistema musculoesquelético [14]. En la actualidad, como resultado del acelerado avance en tecnologías para rehabilitación contamos con un gran abanico de posibilidades para ofrecer soluciones tecnológicas en el tratamiento de lesiones articulares. En España, se están realizando contribuciones representativas en cuanto a tecnologías que sirven de complemento y soporte para la función rehabilitadora y asistencial [9]. La utilización de este tipo de tecnologías reportan beneficios sociales y económicos para los pacientes y el sistema sanitario.

### Rehabilitación mediante juegos basados en *Kinect* y realidad virtual, ReaKinG

Este artículo [68] tiene como objetivo el desarrollo de un juego serio utilizando la interfaz natural de usuario (NUI) de *Kinect*, para asistir la rehabilitación. La plataforma está compuesta por diferentes tipos de ejercicios para la rehabilitación de varios aspectos, como la fuerza o las capacidades aeróbicas y cognitivas. El sistema se desarrolló para que el paciente sea capaz de realizar el proceso de rehabilitación a distancia, sin la presencia del terapeuta.



Figura 3.8: Imágenes de los ejercicios de rehabilitación [68]

### Sistema de monitoreo para pacientes con tratamientos de tendinitis del tendón rotuliano utilizando *Kinect*

Este proyecto [26] tiene como objetivo la creación de un sistema de monitoreo a distancia para pacientes con tratamientos kinesiológicos de tendinitis del tendón rotuliano. El médico especialista define una colección de ejercicios que el paciente deberá realizar a distancia y asistido con el sistema. Entre sus funciones permite al paciente corregir los movimientos que realice y generar unos resultados, que determinan en qué grado está avanzando el paciente.

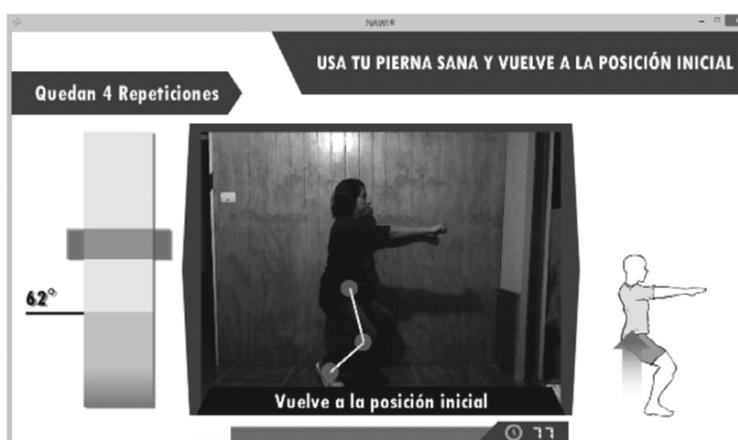


Figura 3.9: Ejecución de la flexión de rodilla [26]

### Sistema basado en visión por computador para la rehabilitación a distancia de la artritis reumatoide

En [31] se presenta un sistema de rehabilitación domiciliaria, haciendo uso del sensor *Kinect*. El sistema trata de supervisar las sesiones de rehabilitación. Además de supervisar la realización de los distintos ejercicios, también recopila y gestiona la información obtenida de las distintas sesiones de rehabilitación.

La artritis reumatoide es una enfermedad que afecta al sistema musco-esquelético y que normalmente afecta a personas de edad avanzada. Por esta razón, el sistema proporciona una interfaz natural que facilita la interacción.

### Sistema de rehabilitación basado en realidad virtual

Los objetivos principales de este artículo [7] son desarrollar una prueba de concepto de la integración de estas tecnologías en el ámbito de la salud y probar si un sistema basado en videojuegos podría acelerar la recuperación del paciente. Los dispositivos utilizados en este trabajo son las gafas de realidad virtual *Oculus Rift DK2* y el dispositivo de seguimiento de movimiento *Intel RealSense*.



Figura 3.10: Escenario del videojuego [7]

### Sistema de rehabilitación virtual basado en *Kinect* de bajo coste para la rehabilitación hospitalaria del miembro superior en pacientes con accidente cerebrovascular subagudo

En [44] se realiza un estudio en el cual un grupo de 23 personas con accidente cerebrovascular subagudo participan en una terapia ocupacional de 30 minutos diariamente para la recuperación de las extremidades superiores durante 10 días. El ensayo se realizó de forma aleatorizada, dividiendo el grupo en dos subgrupos, uno de ellos recibiría la rehabilitación virtual real y el otro grupo realizaría una simulación. Ni los pacientes ni los terapeutas sabía a que grupo se pertenecía. Para el ensayo se desarrollaron tres juegos serios denominados denominados *Push Museum*, *Apple Run* y *Fruit Market*, respectivamente. Los resultados obtenidos fueron que el sistema resultante no fue más eficaz en comparación rehabilitación virtual simulada. Aunque, el sistema indujo a un movimiento del brazo más controlado, lo que sugiere que el sistema podría ser de utilidad como terapia complementaria durante la rehabilitación hospitalaria.

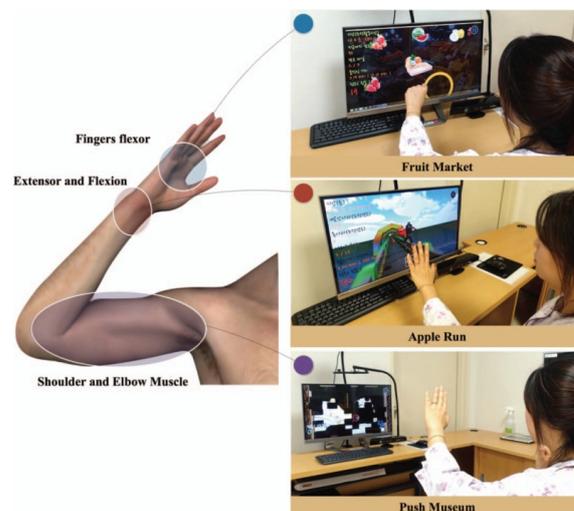


Figura 3.11: Ejemplo de ejecución [44]

### Sistema basado en realidad aumentada y gamificación para la rehabilitación física, *PhyReUP!*

En [71] se propone un sistema basado en realidad aumentada para la rehabilitación física a distancia haciendo uso del dispositivo *Azure Kinect DK* y de las *Hololens*. El sistema se divide en dos subsistemas principales, uno para que el terapeuta pueda definir los distintos ejercicios a realizar y el segundo de ellos es para la realización de los ejercicios creados. El primero de los dispositivos se utiliza para el *tracking* corporal de la persona, ya sea el terapeuta para definir los ejercicios o el paciente para realizarlos. El segundo otorgará al sistema la capacidad de representar hologramas tridimensionales en el espacio, pudiendo así representar los elementos necesarios para realizar el proceso de rehabilitación a distancia.

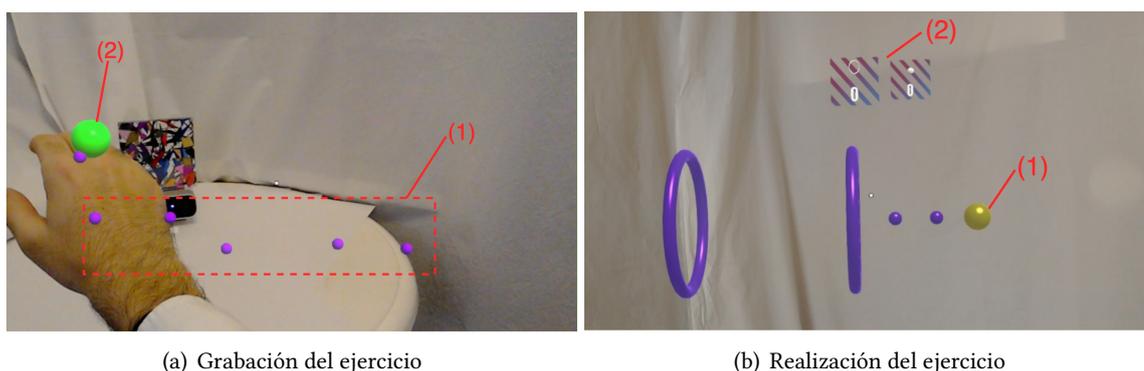


Figura 3.12: Imágenes del sistema *PhyReUP!* [71]

Los trabajos mencionados en esta sección tienen como objetivo la rehabilitación física de personas sin discapacidad visual. Las conclusiones de utilidad que se pueden extraer son que el uso de dispositivos de *tracking* es de vital importancia para hacer un correcto seguimiento de los movimientos del usuario. Además, destaca también la inclusión de diversas técnicas de gamificación para lograr incrementar la motivación del paciente como se ha explicado en la Sección 3.1.3. Para llevar a cabo trabajos de este tipo enfocados en personas invidentes, habría que adaptar la forma de comunicarse con el usuario, dado que la mayoría de trabajos hacen uso de la modalidad sensorial óptica. Se podría extraer como conclusión que para llevar a cabo un sistema de ayuda a la rehabilitación para personas con discapacidad visual habría que utilizar un dispositivo de seguimiento corporal combinado con un dispositivo que utilice las modalidades sensoriales no afectadas. Además, para incluir técnicas de gamificación se podría hacer transmitiendo de forma acústica la información pertinente.

## 3.4. ESTUDIO Y ELECCIÓN DEL ENTORNO HARDWARE

En esta sección se va a realizar un estudio para elegir el entorno hardware más apropiado para el proyecto teniendo en cuenta los proyectos mencionados. El principal inconveniente, hasta hace poco tiempo, al que había que hacer frente cuando se planteaba este tipo de prácticas de rehabilitación era el alto coste del *hardware* que se requería. Este problema ha quedado atrás desde que se han lanzado productos al mercado como el sensor *Kinect*, un dispositivo para practicar un gran abanico de juegos en la consola de *Microsoft, Xbox 360*. Posteriormente, la compañía ofreció nuevas versiones del dispositivo, *Kinect v2* y *Azure Kinect DK*. También se lanzaron al mercado otros dispositivos de *tracking* como *Leap Motion Controller* o *Intel RealSense*. Este tipo de dispositivos despertaron el interés investigador debido a su gran potencial en el ámbito del desarrollo de proyectos de seguimiento de movimientos.

En el proyecto a desarrollar se necesita el seguimiento de movimientos de personas de forma similar a como se realiza en los proyectos ya comentados [7, 26, 31, 68, 71]. El dispositivo de *tracking* resulta de vital importancia para la correcta consecución de los ejercicios a realizar en el sistema a

crear. También existen tecnologías *software* que no necesitan dispositivos específicos para realizar un seguimiento de movimientos, como el caso de *OpenCV*, una de las bibliotecas de *software* libre más relevantes en el ámbito de la visión por computador. El problema que podría existir es que las cámaras convencionales son monoculares, con lo cual existe una gran complejidad a la hora de realizar un seguimiento en tres dimensiones [8]. Por ello, en proyectos de este tipo se suele optar por dispositivos específicamente diseñados para este cometido, como los ya comentados.

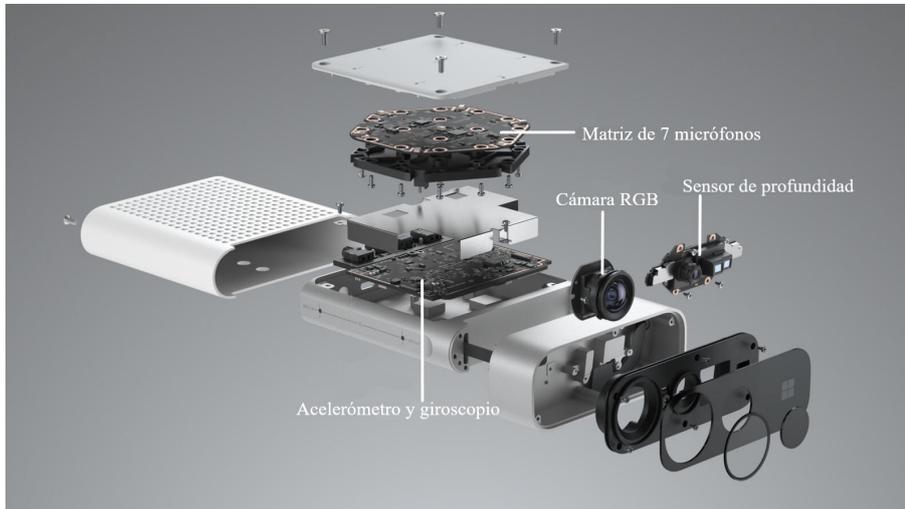


Figura 3.13: Azure Kinect DK

Centrándonos más específicamente en el dispositivo *Azure Kinect DK*, es uno de los más actuales en el ámbito del *tracking*, lanzado al mercado en Abril de 2020. Proporciona características muy útiles en el ámbito del seguimiento corporal como la segmentación de cuerpos o creación de un esqueleto anatómicamente correcto para cada cuerpo detectado, todo ello realizado tridimensionalmente [2]. El dispositivo está formado por una cámara RGB de 21 megapíxeles y una cámara de profundidad. La cámara de profundidad es la característica del dispositivo más relevante para este trabajo, ya que se utiliza para el seguimiento de movimientos tridimensionalmente. Esta cámara estima la profundidad haciendo uso del denominado principio del tiempo de vuelo (ToF), en el cual la distancia a un objeto se determina por el tiempo que tarda la luz emitida en llegar al objeto y volver al sensor [47], mismo método que utiliza la versión anterior del dispositivo. Este principio es muy común en dispositivos de este ámbito, pero lo que hace destacar al *Azure Kinect* es que tiene una precisión significativamente superior que el resto de dispositivos que se comercializan [3]. Además, la cámara *Kinect* tiene una unidad de medición inercial (IMU) integrada y una matriz de 7 micrófonos. Esto amplía el rango de posibles aplicaciones a otras áreas de investigación.

Una vez elegido el dispositivo de *tracking*, otro aspecto de gran importancia en sistemas de ayuda a la rehabilitación es la forma de realizar la retroalimentación al paciente, es decir, que el sistema se comunique con el usuario para que éste sepa en todo momento que acción debe realizar y cómo la está realizando. La primera opción que podríamos sopesar es que el usuario puede ver en una pantalla cómo esta realizando el ejercicio y así poder comprobar si lo está realizando correctamente o no, como en el caso de [26, 31]. Otra opción que se podría considerar es utilizar algún dispositivo de realidad aumentada, con el que el usuario podría ver en un entorno creado cómo está ejecutando las distintas acciones, como en [7, 71]. Estas opciones son totalmente válidas en sistemas enfocados en personas que no sufren discapacidad visual, pero el enfoque de este trabajo son las personas que sí sufren esta discapacidad. Con lo cual, se deben buscar alternativas de retroalimentación del usuario, en los proyectos [37, 41, 48] descritos anteriormente predomina la retroalimentación háptica y acústica como formas de informar al usuario sin necesidad del sentido óptico.

La comunicación háptica es muy común en sistemas diseñados para personas invidentes y existen multitud de aplicaciones existentes como: aplicaciones del sistema braille [13, 62, 64], el método

Tadoma [72], el alfabeto Malossi [20], sistemas táctiles [43, 73] entre otros. Para lograr realizar la retroalimentación háptica se pretende hacer uso de un dispositivo vibrotáctil, capaz de comunicar al paciente invidente mediante vibraciones las direcciones que deben seguir sus movimientos para lograr realizar correctamente los ejercicios terapéuticos definidos. Para el presente proyecto se ha decidido aplicar la retroalimentación en la mano y así poder guiar movimientos relacionados con las articulaciones pertenecientes al brazo.



Figura 3.14: Ejemplos de guantes [30]

En [30] se realiza un estudio de los sistemas basados en guantes y distintos tipos de aplicaciones de éstos. Se define un sistema basado en guantes como un sistema compuesto por una serie de sensores, electrónica para el procesamiento o adquisición de datos y suministro de energía. Los sensores elegidos dependerán del problema que se pretenda resolver. En la Figura 3.14 se pueden observar algunos ejemplos de guantes. Para el caso del presente proyecto, se deben elegir sensores que emitan vibraciones y que éstas puedan ser detectadas por el usuario. Además, se debe elegir un microcontrolador que se encargue de controlar correctamente estos sensores.

En [45] se patenta un sistema de comunicación para personas sordas, sordociegas o no vocales basado en un guante. El sistema está formado por un guante equipado con sensores flexibles diseñados exclusivamente para controlar los ángulos producidos por los dedos realizando distintos tipos de movimiento. Además del guante, el sistema está formado por un computador que recibe los datos de los sensores, y son transcritos a letras y señales de control. Todo ello gracias a un algoritmo de reconocimiento de letras. Cuando se reconoce una letra, ésta se comunica al usuario del guante

dependiendo si tiene discapacidad visual o no. En el caso de un usuario vidente se muestra en una pantalla y para un usuario invidente se muestra en una pantalla braille.

Cabe destacar que no se han encontrado contribuciones destacables para solucionar el problema que pretende resolver este trabajo. Aunque, después del estudio realizado se podría decir que para construir un sistema que resuelva el problema mencionado se necesitan tres dispositivos principales: i) un dispositivo que realiza un seguimiento de los movimientos corporales, ii) un dispositivo que ayude a guiar al paciente invidente y iii) un computador que procese los datos e interactúe con los dos dispositivos ya mencionados. El primer dispositivo debe ser capaz de detectar movimientos con profundidad, es decir, tridimensionalmente. Para el segundo tipo de dispositivo no existen en la actualidad dispositivos comerciales que puedan realizar las funciones requeridas para este trabajo. Por ello, se debe diseñar específicamente un prototipo que cumpla con los requisitos del proyecto. Este prototipo deberá incluir una serie de sensores que guíen al usuario y para controlar estos sensores se debe utilizar un microcontrolador que procese la información recibida por el computador principal.

# METODOLOGÍA

El uso de una metodología de trabajo es algo fundamental en el desarrollo de cualquier proyecto tecnológico, porque permite trabajar de forma eficiente y alcanzar los objetivos establecidos propuestos en la fase inicial del mismo. En este capítulo se va a explicar la metodología utilizada durante el desarrollo del proyecto, la distribución del trabajo y los recursos necesarios para llevarlo a cabo.

## 4.1. METODOLOGÍA ELEGIDA

La metodología elegida para la consecución del proyecto es el **desarrollo iterativo e incremental**. Basado en que el ciclo de vida del desarrollo del proyecto, en este caso de *software*, se divide en una secuencia de iteraciones, como se puede observar en la Figura 4.1. Cada una de ellas se podría considerar como un proyecto en sí, compuesto por varias fases: *análisis, diseño, implementación y pruebas* [49]. Al comienzo del proyecto, se deben acordar los requisitos del proyecto, tanto funcionales como no funcionales, así como el alcance del mismo. Una vez finalizadas todas las iteraciones, se realizará el despliegue en el entorno de ejecución correspondiente.

Cada iteración tiene como resultado una versión funcional del proyecto, implementado nuevas características, permitiendo que el proyecto avance de forma incremental hasta llegar a la versión final. La duración de la iteración será de entre dos y cuatro semanas. Una vez cumplido este tiempo se analizan los resultados obtenidos de la iteración y se contrastan con los objetivos pautados al inicio de la misma.

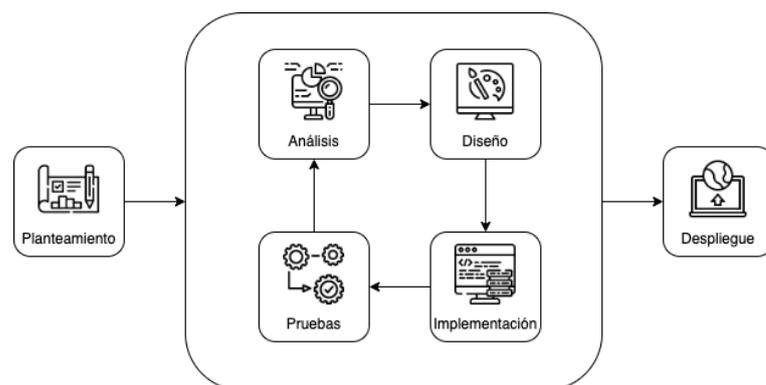


Figura 4.1: Infografía desarrollo iterativo e incremental

La decisión de elegir esta metodología se ha tomado en base a los objetivos y estructura del proyecto, que se podría dividir en dos partes fundamentales: la aplicación del terapeuta con la cual se crearán los ejercicios de rehabilitación y realizará un seguimiento del paciente; y la aplicación del paciente para la ejecución de sus respectivos ejercicios terapéuticos. Ambas partes dependen una

de la otra, por lo tanto, la inclusión de nuevas características en una de ellas influirá en la otra y viceversa. Por esta razón, un desarrollo de forma iterativa permitirá obtener una versión operativa y parcialmente funcional de cada una de las dos partes fundamentales del proyecto al finalizar cada iteración.

## 4.2. DISTRIBUCIÓN DEL TRABAJO

En esta sección se van a explicar las distintas iteraciones que han compuesto el desarrollo de este proyecto, siguiendo la metodología explicada anteriormente. La carga de trabajo se ha dividido en seis paquetes, y cada uno a su vez en distintas tareas. Los paquetes de trabajo son los siguientes:

1. Estudio inicial
2. Aplicación de *tracking*
3. Aplicaciones de rehabilitación terapeuta/paciente
4. Sistema de guiado
5. Sistema de interacción por audio
6. *Backend*

### 4.2.1. Paquetes de trabajo

#### Estudio inicial

El estudio inicial engloba todas las tareas de documentación y estudio para el correcto desarrollo del sistema resultado de este proyecto. Este paquete está compuesto por las siguientes tareas:

1. **Definición de requisitos del sistema.** En esta tarea se han definido los distintos requisitos no funcionales del sistema, acordados con el tutor del proyecto.
2. **Estudio de antecedentes de sistemas de rehabilitación.** La finalidad de esta tarea fue estudiar los distintos sistemas de rehabilitación desarrollados y su contexto, siempre enfocados a personas con problemas de visión.
3. **Estudios de modos de interacción.** Dado el contexto de rehabilitación de los pacientes que harán uso del sistema, el estudio y la elección de los correctos modos de interacción con el usuario fue una decisión de gran importancia para el desarrollo del sistema.
4. **Estudio de dispositivos hardware.** Una vez elegidos los distintos modelos de interacción que componen el sistema, otro aspecto de gran relevancia son los dispositivos hardware empleados para efectuar una correcta interacción con los usuarios del sistema.

#### Aplicación de *tracking*

El paquete de trabajo correspondiente al desarrollo del *tracking*, consiste en desarrollar la parte lógica que procese los datos recibidos del dispositivo de seguimiento de articulaciones elegido en una tarea anterior. Compuesto de tres tareas:

1. **Diseño e implementación del *tracking* tridimensional.** Con esta tarea se pretende desarrollar la lógica que procese la información de la posición de la articulación objetivo.
2. **Comunicación con las aplicaciones de rehabilitación.** Esta tarea tiene la finalidad de conseguir una comunicación eficiente entre la aplicación de *tracking* y las aplicaciones de rehabilitación.
3. **Validación y pruebas.** El objetivo de esta tarea es realizar las pruebas necesarias para validar que la aplicación funciona correctamente.

### Aplicaciones de rehabilitación terapeuta/paciente

Este paquete de trabajo está formado por las tareas de diseño y desarrollo de las aplicaciones de rehabilitación que usarán el terapeuta y el paciente. Han sido incluidas ambas aplicaciones en el mismo paquete, ya que comparten la gran mayoría de módulos, que serán abstraídos a un nivel superior para que se puedan usar en ambas aplicaciones. Las tareas que conforman este paquete son las siguientes:

1. **Obtención y representación de los datos del *tracking*.** En esta tarea se ha implementado la recepción y procesamiento de la información recibida de la aplicación de *tracking* y la representación de esta información según el rol del usuario.
2. **Desarrollo del módulo de definición de ejercicios.** A lo largo de esta tarea se ha desarrollado el módulo encargado de definir los ejercicios terapéuticos, que serán utilizados en la aplicación del paciente.
3. **Desarrollo del módulo de realización de ejercicios.** Durante esta tarea se ha desarrollado el módulo encargado de que el paciente pueda reproducir de la forma más correcta posible los ejercicios creados por el terapeuta.
4. **Comunicación con el sistema de guiado.** En esta tarea, se ha implementado la interacción con el sistema de guiado, encargado de guiar al paciente invidente durante la realización de ejercicios utilizando interacción háptica.
5. **Comunicación con el *Backend*.** Durante el desarrollo de esta tarea se ha implementado el envío y recepción de datos, tanto del terapeuta como del paciente, al subsistema encargado de la persistencia de datos.
6. **Comunicación con el Sistema de interacción por audio.** En esta tarea se ha desarrollado la interacción con el subsistema encargado de interactuar con el usuario mediante comandos de voz, para facilitar el uso del sistema tanto en la creación como en la ejecución de ejercicios.
7. **Validación y pruebas.** A lo largo de esta tarea se han realizado las pruebas pertinentes para corregir los problemas surgidos durante el desarrollo de ambas aplicaciones de rehabilitación y mejorar así su funcionamiento final.

### Sistema de guiado

El presente paquete de trabajo es el correspondiente al desarrollo tanto de la parte *hardware* como *software* del sistema de guiado, que ayuda al paciente invidente a utilizar la aplicación de rehabilitación sin hacer uso del sentido de la vista a la hora de realizar los ejercicios terapéuticos, ya que se emplea retroalimentación háptica.

1. **Diseño *hardware* del dispositivo.** Esta tarea consiste en realizar el diseño *hardware* del dispositivo de guiado en forma de guante, teniendo en cuenta el estudio realizado anteriormente.
2. **Diseño e implementación de la comunicación.** En esta tarea se realiza el desarrollo de la comunicación de este sistema con la aplicación del terapeuta, que permite que se puedan enviar señales de vibración.
3. **Desarrollo de la lógica interna con los actuadores.** Durante esta tarea, se desarrolla la lógica del dispositivo, para que una vez se reciban órdenes de vibración se realicen en el actuador correspondiente a la dirección que debe seguir el paciente.
4. **Validación y pruebas.** A lo largo de esta tarea, se realizaron las pruebas y las validaciones correspondientes para comprobar si el sistema de guiado se integraba correctamente con la aplicación del terapeuta a la hora de realizar los ejercicios.

### Sistema de interacción por audio

En este paquete de trabajo, se ha desarrollado un sistema de interacción por audio, con el cual en ambas aplicaciones, tanto la del terapeuta como la del paciente, se podrá interactuar mediante órdenes

de voz y audio. Esta interacción consiste en realizar indicaciones mediante audio e interpretando órdenes por voz a la hora de realizar ejercicios.

1. **Implementación del reconocimiento de voz.** Esta tarea consiste en implementar el reconocimiento de voz, que permite una conversión de voz a texto para que el sistema pueda tomar las decisiones pertinentes.
2. **Implementación de la transcripción de texto a audio.** Durante esta tarea, se ha realizado la conversión de texto a audio, para reproducir mensajes específicos al usuario.
3. **Implementación de la reproducción de sonidos.** Implementación de la lógica necesaria para la reproducción de sonidos específicos que mejoran la interacción del sistema con el usuario.
4. **Diseño e implementación de la API.** En esta tarea se ha implementado una *API Rest* que encapsula toda la lógica desarrollada en este paquete. Esta implementación permite a las aplicaciones del terapeuta y del paciente comunicarse de manera eficiente con este sistema.
5. **Validación y pruebas.** En la realización de esta tarea, se ha validado si las aplicaciones de rehabilitación interactúan de forma correcta con el sistema resultante de este paquete.

### **Backend**

Este paquete de trabajo tiene como objetivo el desarrollo de la persistencia de datos del sistema resultante siguiendo una estructura de creación, lectura, actualización y eliminación de datos, como se ha comentado anteriormente.

1. **Diseño e implementación de la base de datos.** En esta tarea se ha desarrollado una base de datos adecuada al sistema, que permite guardar la información necesaria para que el sistema cumpla su cometido.
2. **Diseño e implementación de la API.** Esta tarea tiene como objetivo encapsular la base de datos desarrollada en la tarea anterior en una **API Rest**.
3. **Validación y pruebas.** Durante esta tarea, se han realizado las pruebas necesarias para validar si el *Backend* desarrollado en este paquete se comporta correctamente dentro del sistema global.

	Sem 1	Sem 2	Sem 3	Sem 4	Sem 5	Sem 6	Sem 7	Sem 8	Sem 9	Sem 10	Sem 11	Sem 12	Sem 13	Sem 14	Sem 15	Sem 16	Sem 17	Sem 18	Sem 19	Sem 20	Sem 21
	Enero		Febrero				Marzo				Abril				Mayo				Junio		
<b>[1] Tarea [2]</b>																					
<b>1 Estudio Inicial</b>																					
1.1 Definición de requisitos del sistema																					
1.2 Estudio de antecedentes de sistemas de rehabilitación																					
1.3 Estudio de modos de interacción																					
1.4 Estudio de dispositivos hardware																					
<b>2 Sistema de seguimiento de articulaciones</b>																					
2.1 Implementación del tracking tridimensional																					
2.2 Comunicación con las aplicaciones de rehabilitación																					
2.3 Validación y pruebas																					
<b>3 Aplicaciones de rehabilitación terapeuta/paciente</b>																					
3.1 Obtención y representación de los datos del tracking																					
3.2 Desarrollo del módulo de definición de ejercicios																					
3.3 Desarrollo del módulo de realización de ejercicios																					
3.4 Comunicación con el Sistema de guiado																					
3.5 Comunicación con el Backend																					
3.6 Comunicación con el Sistema de interacción por audio																					
3.7 Validación y pruebas																					
<b>4 Sistema de guiado</b>																					
4.1 Diseño hardware del dispositivo																					
4.2 Diseño e implementación de la comunicación																					
4.3 Desarrollo de la lógica interna con los actuadores																					
4.4 Validación y pruebas																					
<b>5 Sistema de interacción por audio</b>																					
5.1 Implementación de la reproducción de sonidos																					
5.2 Implementación de la transcripción de texto a audio																					
5.3 Implementación del reconocimiento de voz																					
5.4 Diseño e implementación de la API																					
5.5 Validación y pruebas																					
<b>6 Backend</b>																					
6.1 Diseño e implementación de la base de datos																					
6.2 Diseño e implementación de la API																					
6.3 Validación y pruebas																					

### 4.2.2. Iteraciones

Este proyecto se ha desarrollado en siete iteraciones, siguiendo la metodología mencionada en 4.1. La primera iteración se compone de un estudio inicial de los antecedentes existentes para obtener los requisitos del proyecto. Las siguientes iteraciones se centran en el diseño, implementación y validación de los distintos subsistemas que componen el proyecto. Cada iteración se resume en una tabla en la que se indica el inicio y el fin de la misma, los paquetes de trabajo relacionados, las tareas realizadas y los objetivos alcanzados.

#### Iteración 1

La primera iteración se ha centrado en estudiar los antecedentes de sistemas de rehabilitación para personas invidentes, los modos de interacción con el usuario y los distintos dispositivos *hardware* existentes para el seguimiento corporal. De este estudio se han extraído los requisitos del proyecto, así como los dispositivos *hardware* a utilizar, uno de ellos para el *tracking* corporal y otro diseñado específicamente para guiar al paciente con problemas de visión mediante retroalimentación háptica.

Iteración 1		
27/01/2020 - 28/02/2020		Estimación: 100h
Paquete de trabajo	Tareas	Objetivos alcanzados
Pt.1	T1.1 T1.2 T1.3 T1.4	- Requisitos del sistema - Elección de hardware - Modos de interacción

**Tabla 4.1:** Resumen de la iteración 1

#### Iteración 2

Durante la segunda iteración se diseñó del *tracking* tridimensional con el dispositivo *Kinect Azure DK*. Puesto que los datos obtenidos del *tracking* deben ser servidos a las aplicaciones de rehabilitación se ha implementado la comunicación con estos subsistemas. La representación de los datos de *tracking* es un parte fundamental para el correcto funcionamiento del sistema global, ya que la eficiencia del sistema depende en gran parte de la representación y tratamiento de éstos.

Iteración 2		
01/03/2021 - 13/03/2021		Estimación: 40h
Paquete de trabajo	Tareas	Objetivos alcanzados
Pt.2 Pt.3	T2.1 T2.2 T3.1	- Tracking tridimensional - Comunicación con aplicaciones de rehabilitación - Representación de datos de tracking

**Tabla 4.2:** Resumen de la iteración 2

#### Iteración 3

La tercera de las iteraciones del desarrollo ha consistido en la optimización del seguimiento corporal para validar que el *tracking* se realiza correctamente. Posteriormente, se implementó una versión preliminar del algoritmo de creación de ejercicios, que ejecutará la aplicación del terapeuta.

Iteración 3		
15/03/2021 - 24/03/2021		Estimación: 40h
Paquete de trabajo	Tareas	Objetivos alcanzados
Pt.2 Pt.3	T2.3 T3.2	- Corrección de errores de tracking - Validación del sistema de tracking - Versión preliminar del algoritmo de creación de ejercicios

**Tabla 4.3:** Resumen de la iteración 3

#### Iteración 4

Durante esta iteración se consiguió una versión funcional del módulo de definición de ejercicios perteneciente a la aplicación del terapeuta. Al mismo tiempo, se ha desarrollado una versión preliminar del algoritmo de ejecución ejercicios, basado en los datos resultantes del módulo anterior. Además, se implementó la reproducción de sonidos y la transcripción de texto a audio, funcionalidades pertenecientes al sistema de interacción por audio.

Iteración 4		
25/03/2021 - 09/04/2021		Estimación: 60h
Paquete de trabajo	Tareas	Objetivos alcanzados
Pt.3 Pt.5 Pt.6	T3.2 T3.3 T5.1 T5.2 T6.1	- Versión funcional del módulo de definición de ejercicios - Versión preliminar del algoritmo de ejecución de ejercicios - Reproducción de sonidos - Transcripción de texto a audio - Persistencia de datos de los ejercicios definidos

**Tabla 4.4:** Resumen de la iteración 4

#### Iteración 5

En la iteración cinco se ha implementado una versión funcional del módulo de ejecución de ejercicios y con ello el prototipo del dispositivo de guiado. Además, se ha implementado la interacción por voz, por parte del paciente, funcionalidad incluida en el sistema de interacción por audio.

Iteración 5		
10/04/2021 - 25/04/2021		Estimación: 50h
Paquete de trabajo	Tareas	Objetivos alcanzados
Pt.3 Pt.4 Pt.5 Pt.6	T3.3 T4.1 T5.3 T5.4 T6.1	- Versión funcional del módulo de ejecución de ejercicios - Prototipo del dispositivo de guiado - Interacción por voz - Persistencia de datos de los resultados

**Tabla 4.5:** Resumen de la iteración 5

#### Iteración 6

La iteración seis ha consistido en desarrollar una versión funcional del sistema de guiado que será utilizado durante la ejecución de los ejercicios. Además, se ha implementado la comunicación con el resto de subsistemas por parte de las aplicaciones de rehabilitación.

Tabla 4.6: Resumen de la iteración 6

Iteración 6		
26/04/2021 - 16/05/2021		Estimación: 60h
Paquete de trabajo	Tareas	Objetivos alcanzados
Pt.3 Pt.4 Pt.5 Pt.6	T3.4	- Versión funcional del sistema de guiado - Comunicación con el sistema de guiado - Comunicación con el sistema de interacción por audio - Comunicación con el backend
	T3.5	
	T3.6	
	T4.2	
	T4.3	
	T5.4	
	T6.2	

Tabla 4.7: Resumen de la iteración 6

### Iteración 7

La última de las iteraciones ha consistido en corregir los errores encontrados durante la ejecución de las pruebas del sistema. Una vez corregidos los errores el sistema ha quedado validado y se ha procedido al despliegue. Además, durante esta iteración se terminó de documentar el presente documento que representa la memoria final del proyecto.

Iteración 7		
16/05/2021 - 25/05/2021		Estimación: 40h
Paquete de trabajo	Tareas	Objetivos alcanzados
Pt.3 Pt.4 Pt.5 Pt.6	T3.7	- Corrección de errores - Validación y pruebas - Despliegue
	T4.4	
	T5.5	
	T6.3	

Tabla 4.8: Resumen de la iteración 7

## 4.3. RECURSOS

Para el correcto desarrollo del proyecto será necesario hacer uso de recursos *hardware* y *software*. En esta sección, se van a mencionar los distintos recursos utilizados durante la realización del proyecto.

### 4.3.1. Recursos Hardware

Teniendo en cuenta el proyecto a desarrollar, los recursos de este tipo son un aspecto fundamental para el cumplimiento de los objetivos del mismo.

- **Computador de escritorio:** equipo utilizado durante el desarrollo del sistema, compuesto por el procesador *AMD FX(tm)-8350* con 8 núcleos y una frecuencia de 4 GHz.
- **Dispositivo de *tracking corporal*, *Azure Kinect DK*<sup>1</sup>:** Dispositivo desarrollado por *Microsoft* compuesto por una serie de sensores y cámaras que van a permitir obtener la información de las articulaciones del usuario en tres dimensiones.
- **Dispositivo de guiado (guante):** compuesto por una microcontrolador y seis actuadores.
  - **Microcontrolador *ESP8266* con pantalla *OLED* integrada de 0,91"**<sup>2</sup>: dispositivo diseñado por la empresa *AZ-Delivery* que destaca por tener una pantalla integrada, será utilizada para mostrar información relevante. Además, este microcontrolador posee un módulo *wifi* integrado, característica fundamental para la comunicación.

<sup>1</sup><https://azure.microsoft.com/es-es/services/kinect-dk/>

<sup>2</sup><https://www.az-delivery.de/es/products/esp8266-mikrocontroller-mit-integrierten-0-91-oled-display>

- **Motores de vibración:** actuadores conectados al microcontrolador que son capaces de convertir fácilmente señales eléctricas en vibraciones mecánicas.

### 4.3.2. Recursos *Software*

Los recursos *software* son también una pieza fundamental para el desarrollo del proyecto, ya que la correcta elección de los mismos fue primordial para sacar el máximo beneficio a los recursos *hardware* elegidos.

#### Sistemas operativos

- **Microsoft Windows 10 Pro:** es la versión profesional de *Windows 10*. La aplicación de escritorio ha sido desarrollada para este sistema operativo, por la gran compatibilidad con el dispositivo de *tracking* corporal.

#### Tecnologías de desarrollo

- **Node.js<sup>3</sup>:** es un entorno de desarrollo de aplicaciones web que permite programar en *JavaScript* para crear aplicaciones web y móviles. Utilizado en la aplicación de escritorio y en la parte *backend* del sistema
- **JavaScript:** lenguaje de programación interpretado, orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico. Empleado en la aplicación de escritorio y en la parte *backend* para desarrollar la lógica.
- **HyperText Markup Language (HTML):** lenguaje de marcado para la elaboración de páginas web. Utilizado en la aplicación principal para representar las distintas pantallas de la aplicación.
- **JSON:** formato de texto sencillo para el intercambio de datos basado en la notación literal de objetos *JavaScript*. Las respuestas del *backend* del sistema están representadas en este formato.
- **C++:** lenguaje de programación creado con el objetivo de extender al lenguaje de programación C, con mecanismos que permiten la manipulación de objetos. Para el desarrollo de la lógica del dispositivo de guiado se utilizó este lenguaje de programación.
- **MySQL<sup>4</sup>:** sistema de gestión de bases de datos relacional desarrollado bajo licencia pública general y licencia comercial por *Oracle Corporation*. La base de datos del sistema se ha desarrollado con *MySQL*.
- **LaTeX:** sistema de composición de textos, orientado a la creación de documentos de gran calidad tipográfica. Empleado para el desarrollo de la documentación de este proyecto.

#### Kits de desarrollo y bibliotecas *software*

- **Azure Kinect Sensor SDK<sup>5</sup>:** kit de desarrollo formado por todas las bibliotecas necesarias y *wrappers* para el manejo de los sensores y dispositivos del *Azure Kinect DK*.
- **Azure Kinect Body Tracking SDK<sup>6</sup>:** kit de desarrollo que permite construir aplicaciones de *tracking* corporal con el dispositivo *Azure Kinect DK*.
- **Electron<sup>7</sup>:** *framework open source* para *Node.js* que permite el desarrollo de aplicaciones gráficas de escritorio. Muchas aplicaciones comerciales han sido desarrolladas gracias a este *framework*, entre las que destacan *Visual Studio Code*, *WhatsApp* o *Twitch*. Ha sido elegido para el desarrollo de la aplicación de escritorio del sistema por la gran versatilidad que aporta.
- **Kinect Azure<sup>8</sup>:** librería de *Node.js* que permite el desarrollo de aplicaciones de escritorio haciendo uso de *Electron*. Se podría definir como un *wrapper* de los *SDK* oficiales comentados

<sup>3</sup><https://nodejs.org/>

<sup>4</sup><https://www.mysql.com/>

<sup>5</sup><https://docs.microsoft.com/en-us/azure/kinect-dk/sensor-sdk-download>

<sup>6</sup><https://docs.microsoft.com/en-us/azure/kinect-dk/body-sdk-download>

<sup>7</sup><https://www.electronjs.org/>

<sup>8</sup><https://github.com/wouterverweider/kinect-azure>

anteriormente, que permite programar aplicaciones más versátiles con el dispositivo *Azure Kinect* sin pérdida de rendimiento.

- **Express**<sup>9</sup>: *framework open source* para *Node.js* que permite el desarrollo *backend*. Utilizado para el desarrollo de la *API REST* del sistema.
- **Sequelize**<sup>10</sup>: *Object Relation Controller (ORM)* para *Node.js* muy destacado, con compatibilidad para con *MySQL*, *MariaDB*, *SQLite* y *Microsoft SQL Server*. El *backend* del sistema hace uso de este *ORM* para mapear los objetos las estructuras de la base de datos relacional construida con *MySQL*.

## Herramientas de desarrollo

- **Visual Studio Code**<sup>11</sup>: editor de código fuente desarrollado por *Microsoft*. Utilizando para el desarrollo de las distintas partes del sistema.
- **Platformio**<sup>12</sup>: entorno de desarrollo integrado en *Visual Studio Code*, que proporciona características modernas y poderosas para acelerar y simplificar el desarrollo de *software* en sistemas embebidos. Ha sido utilizado durante el desarrollo del dispositivo de guiado.
- **Git**: sistema de control de versiones empleado para el correcto desarrollo del *software* del proyecto.
- **Postman**<sup>13</sup>: herramienta de testeo de *APIs* utilizado durante el desarrollo de la *API REST* del sistema.
- **DBeaver**<sup>14</sup>: herramienta de administración de bases de datos que permite realizar todas las operaciones relacionadas con el desarrollo de bases de datos de forma gráfica. Durante la creación de la base de datos del sistema se ha utilizado esta herramienta.
- **Overleaf**<sup>15</sup>: editor colaborativo de *LaTeX*, utilizado en el desarrollo del presente documento.
- **Draw.io**<sup>16</sup>: herramienta de creación de distintos tipos de diagramas, como diagramas de flujo, diagramas de procesos, organigramas, UML, ER y diagramas de red. Los distintos esquemas o diagramas de este documento se han desarrollado con esta herramienta.

---

<sup>9</sup><https://expressjs.com/>

<sup>10</sup><https://sequelize.org/>

<sup>11</sup><https://code.visualstudio.com/>

<sup>12</sup><https://platformio.org/>

<sup>13</sup><https://www.postman.com/>

<sup>14</sup><https://dbeaver.io/>

<sup>15</sup><https://www.overleaf.com/>

<sup>16</sup><http://draw.io/>

# ARQUITECTURA

---

En el capítulo actual se va realizar una descripción técnica detallada del sistema diseñado y desarrollado. El contenido del capítulo incluye el modo en el que se han alcanzado los objetivos y resuelto los diferentes problemas que han surgido. La descripción parte de un enfoque más general para continuar hacia detalles más específicos, con el objetivo de aportar una visión global al lector y conocer los detalles de la propuesta gradualmente.

## 5.1. VISIÓN GENERAL

El sistema global está dividido a su vez en dos subsistemas con dos perfiles de usuario diferenciados, cada uno de ellos con un objetivo diferente. El primer perfil corresponde al *terapeuta*, el cual podrá crear ejercicios y realizar un seguimiento de la rehabilitación de los pacientes. Por otro lado, el segundo de ellos corresponde al *paciente*, cuyo propósito es el de realizar los diferentes ejercicios terapéuticos según las indicaciones del especialista médico.

Las diferentes partes del sistema mencionadas se pueden ver de forma resumida a continuación (ver Figura 5.1):

- **Aplicación del paciente:** encargada de guiar al paciente durante la realización de los distintos tipos de ejercicios de rehabilitación pautados por el terapeuta. Esta parte del sistema hará uso del dispositivo de seguimiento corporal de igual forma que la aplicación del terapeuta, interactuando con el sistema de seguimiento de articulaciones para obtener la información corporal y con el *Backend* para la persistencia de datos. El usuario de esta aplicación, al ser invidente, recibirá información auditiva y podrá hacer un control de la aplicación por medio de la voz. Ambas funciones realizadas por el sistema de interacción por audio.
  - **Sistema de guiado:** este sistema será utilizado únicamente por el paciente. Como su propio nombre indica, su función principal es guiar al paciente durante la realización de los ejercicios. El sistema guiará al usuario mediante vibraciones según la dirección del movimiento que deba seguir, teniendo en cuenta que el paciente no podrá hacer uso de una pantalla para guiarse, dada su discapacidad. Para ello, el sistema dispone de un guante compuesto por un microcontrolador y seis motores de vibración, situados sobre diferentes partes de la mano, y distribuidos de tal forma que se sitúan dos por cada uno de los ejes (X, Y y Z) en el espacio 3D. El computador principal se conectará de forma inalámbrica al dispositivo de guiado (guante) para enviar las instrucciones de guiado pertinentes (vibraciones) que, mediante estas vibraciones, guiará el movimiento de brazo y mano del paciente invidente.
- **Aplicación del terapeuta:** proporciona al terapeuta la herramienta que le permite crear ejercicios de rehabilitación y seguir la evolución del paciente. Dentro de esta aplicación se recibe la información relativa al *tracking* corporal gracias al sistema de seguimiento de articulaciones. Para realizar operaciones que requieran datos se conectará con el *Backend* del sistema. Durante

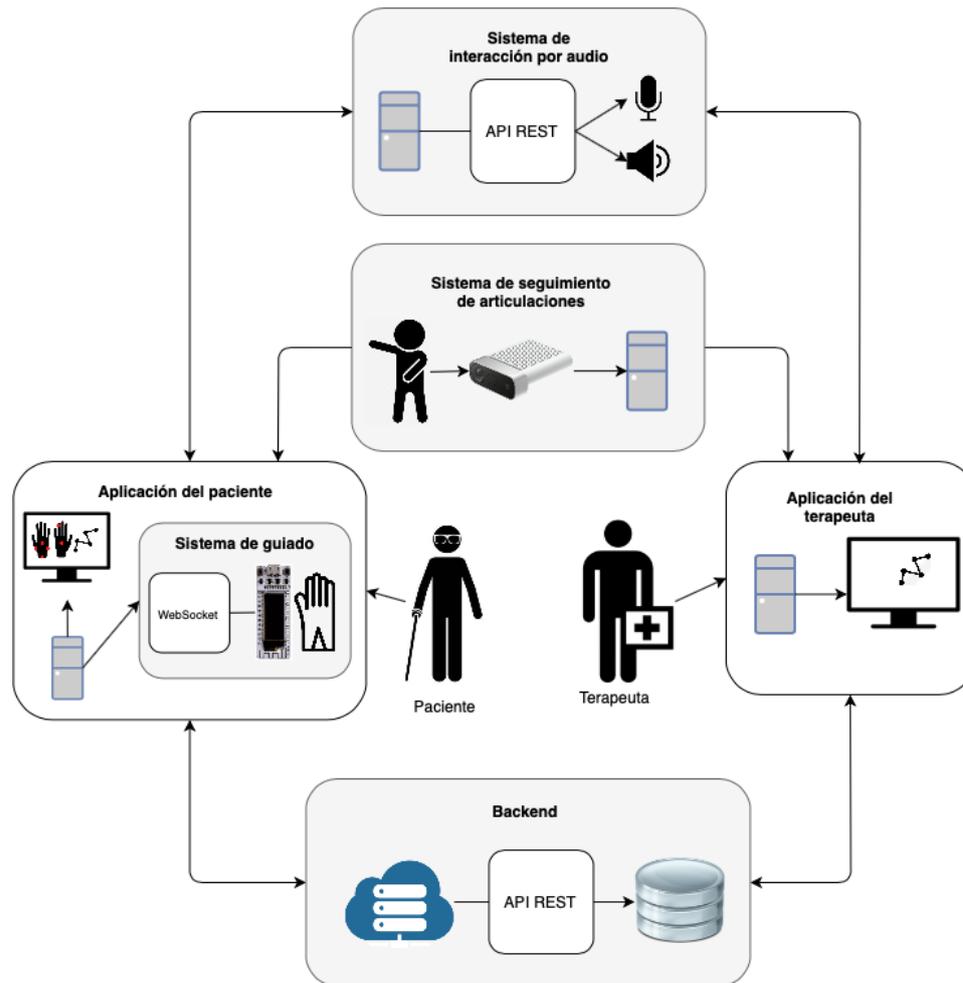


Figura 5.1: Esquema general del sistema

la ejecución de la aplicación el usuario recibirá información de forma sonora, que le permitirá tener una interacción con la aplicación más intuitiva, todo ello gracias al sistema de interacción por audio.

- **Sistema de seguimiento de articulaciones:** se encarga de obtener la información del movimiento de las articulaciones del usuario. El resto de aplicaciones dependen directamente de la información captada por esta aplicación, es decir, es de vital importancia tanto para la creación de los ejercicios terapéuticos como para su correcta ejecución por parte del paciente. El dispositivo utilizado en este sistema, es el *Azure Kinect DK*. En [82] se confirma que tanto la desviación estándar como el error sistemático del dispositivo están dentro de los valores especificados en la documentación oficial. La desviación estándar equivaldría a 17 mm, junto a un error sistemático de  $<11 \text{ mm} + 0,1\%$  de la distancia sin interferencia de trayectos múltiples.
- **Sistema de interacción por audio:** es el sistema encargado de interactuar con el usuario, ya sea paciente o terapeuta, mediante audio. Las funciones principales son la de reproducción tanto de sonidos como de mensajes de texto y el reconocimiento de voz. Esta última función tiene especial relevancia para la aplicación del paciente, ya que teniendo en cuenta su discapacidad tendrá la posibilidad de controlar aplicación de rehabilitación por medio de la voz. Toda la lógica de este sistema está encapsulada dentro de una *API Rest* que se ejecutará en el computador principal, de esta forma las aplicaciones de rehabilitación simplemente tendrán que realizar las peticiones pertinentes para su uso.
- **Backend de la aplicación:** se encargará de realizar operaciones sobre la base de datos y de interactuar con las aplicaciones anteriores cuando sea necesario realizar operaciones sobre los

datos persistentes del sistema. Para ello se ha desarrollado una API REST y una base de datos con las capacidades necesarias para mantener la persistencia de datos del sistema.

A continuación, se describirán con mayor grado de detalle cada una de las partes que componen el esquema general presentado en esta sección.

## 5.2. ARQUITECTURA DEL SISTEMA

En la sección anterior se ha descrito el sistema de forma general, en esta sección se procederá a hacer una descripción con más detalle, haciendo hincapié en detalles más técnicos del sistema. En la Figura 5.2 se pueden observar los componentes principales del sistema divididos en capas independientes, así como sus respectivos módulos, cada uno de ellos con una función específica dentro del sistema global.

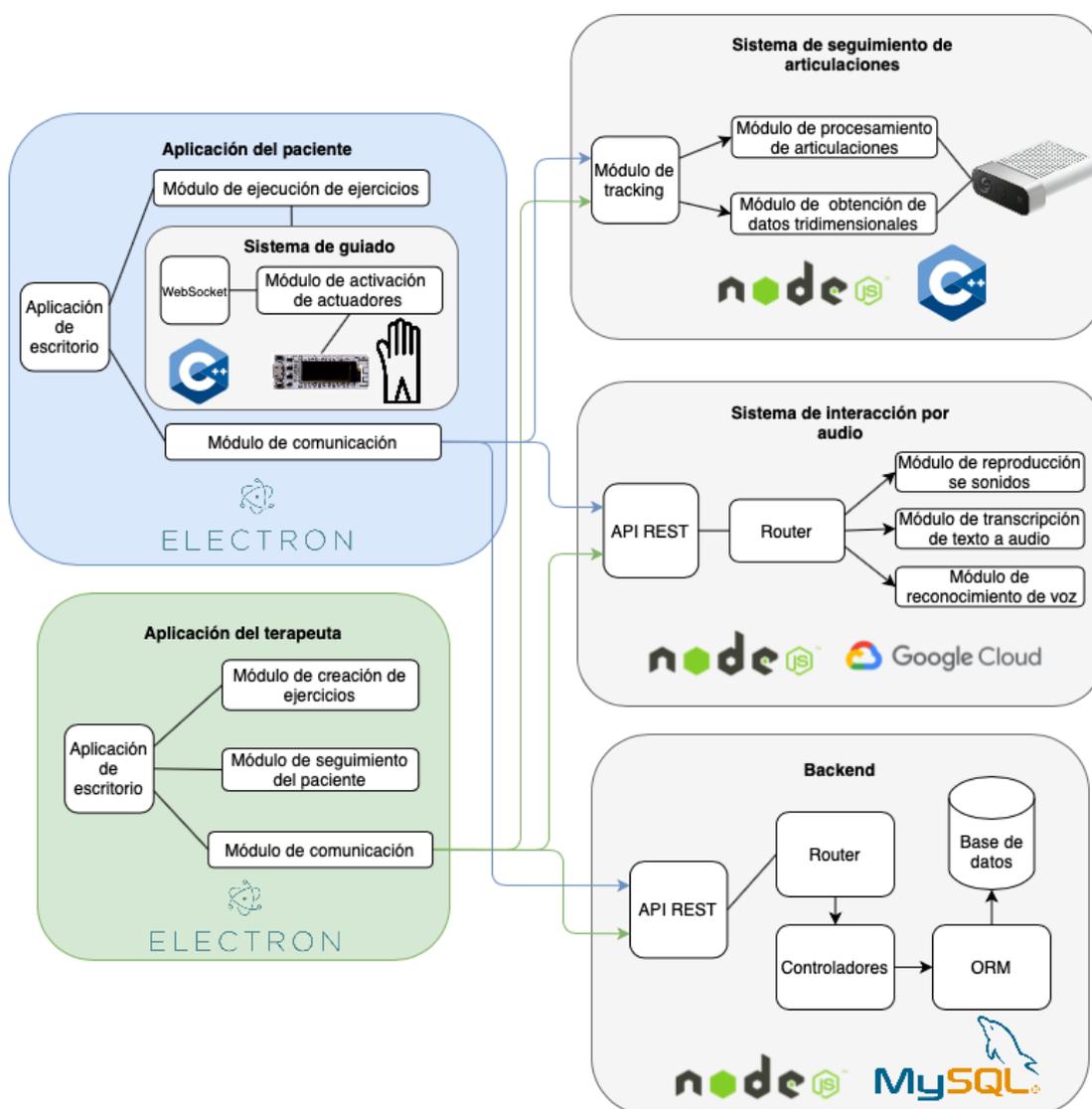


Figura 5.2: Arquitectura del sistema propuesto

### 5.2.1. Aplicación del paciente

La aplicación de rehabilitación para el paciente es una de las aplicaciones principales del sistema global, con el objetivo de capacitar al paciente para realizar los ejercicios terapéuticos para rehabilitarse. La aplicación está dividida en dos módulos principales, uno de ellos para ejecutar los ejercicios

terapéuticos creados por el terapeuta y el módulo de comunicación para comunicarse con el resto de subsistemas del sistema global.

El módulo de ejecución de ejercicios está enfocado en el procesamiento de los datos provenientes del sistema de seguimiento de articulaciones, según los cuales el paciente será guiado en la realización de un ejercicio. Este módulo se conecta con el sistema de guiado, formado por un microcontrolador y un conjunto de motores de vibración que representan las coordenadas espaciales, dando como resultado un guante que será usado por el paciente durante la realización del ejercicio. Cada coordenada está representada por dos actuadores, uno por cada sentido. El módulo de ejecución de ejercicios enviará órdenes al sistema de guiado para activar la vibración de los actuadores, según la coordenada más alejada y su sentido, guiando así de forma háptica al paciente invidente. El módulo de activación de actuadores no tiene lógica alguna, simplemente procesa órdenes activando o desactivando los actuadores. La conexión entre el módulo de ejercicios y el sistema de guiado es inalámbrica y continua, con la implementación de un *WebSocket* en el módulo de activación de actuadores. De esta forma, se consigue que las órdenes se procesen de manera eficiente y en tiempo real, ya que la conexión es síncrona, puesto que se presupone que la aplicación enviará gran cantidad de órdenes al sistema de guiado.

El módulo de comunicación es el encargado de comunicarse con los distintos subsistemas, dependiendo las acciones que se requieran. Para recibir los datos del *tracking* corporal se comunicará con el sistema de seguimiento de articulaciones. Cuando la aplicación requiera interactuar por voz o audio con el paciente realizará peticiones al sistema de interacción por audio. En cuanto a operaciones relacionadas con datos persistentes, la aplicación interactuará con el *backend* del sistema.

La aplicación de escritorio se ha desarrollado con un *framework JavaScript* enfocado en el desarrollo de *software* de escritorio, llamado *Electron.js*. Con esta elección, se consigue gran versatilidad en el desarrollo, puesto que se han podido utilizar librerías asociadas al desarrollo *web*. Para el sistema de guiado se ha utilizado *C++* como lenguaje de programación en el desarrollo del *software* que ejecuta el microcontrolador, debido a que ofrece un gran rendimiento en este tipo componentes físicos. El rendimiento es un fundamental, ya que es necesario que el sistema de guiado funcione en tiempo real.

### 5.2.2. Aplicación del terapeuta

La aplicación del terapeuta es la encargada de dotar al terapeuta con las herramientas necesarias para crear ejercicios de rehabilitación para pacientes invidentes y realizar el seguimiento de éstos. La aplicación esta formada por tres módulos, cada uno con funciones específicas. El módulo de ejecución de ejercicios es el utilizado durante la creación de los ejercicios. Cuando el terapeuta requiera realizar el seguimiento de un paciente, la aplicación hará uso del módulo de seguimiento del paciente. El módulo de comunicación será el encargado de dotar a la aplicación con la capacidad de comunicarse con otros subsistemas. Al igual que la aplicación del paciente, la aplicación de escritorio se ha desarrollado con *Electron.js*. De esta forma una parte del código desarrollado en ambas aplicaciones es compartido, consiguiendo que el proceso de desarrollo de *software* haya sido más eficiente.

Durante la creación de ejercicios el terapeuta dibujará en el espacio el movimiento deseado, como se puede ver en la Figura 5.3. Se puede apreciar como la persona con el rol de terapeuta realiza un movimiento frente al dispositivo de *tracking* y en la pantalla del computador principal se dibujan los puntos que representan la trayectoria. Una vez finalizada la creación el ejercicio será almacenado en el sistema, para su posterior reproducción en el entorno del paciente. Para llevar a cabo esta acción la aplicación requerirá datos de *tracking* corporal suministrados por el sistema de seguimiento de articulaciones. Durante la creación, la aplicación interactuará con el usuario mediante sonidos o mensajes dictados, gracias al sistema de interacción por audio. Cuando la creación finalice será necesario guardar el ejercicio para que sea persistente dentro del sistema global, para ello se realizarán peticiones al *backend* con este cometido. Para que el terapeuta pueda realizar un correcto seguimiento

del paciente es necesario que éste pueda comprobar cómo ha realizado el paciente los ejercicios pautados. Para ello es necesario recuperar los datos que representan la ejecución de los ejercicios. Esta información está almacenada en el *backend* del sistema, por lo tanto se realizarán las peticiones pertinentes para obtenerla. Todas estas acciones asociadas a subsistemas fuera de la aplicación del terapeuta son realizadas gracias al módulo de comunicación.

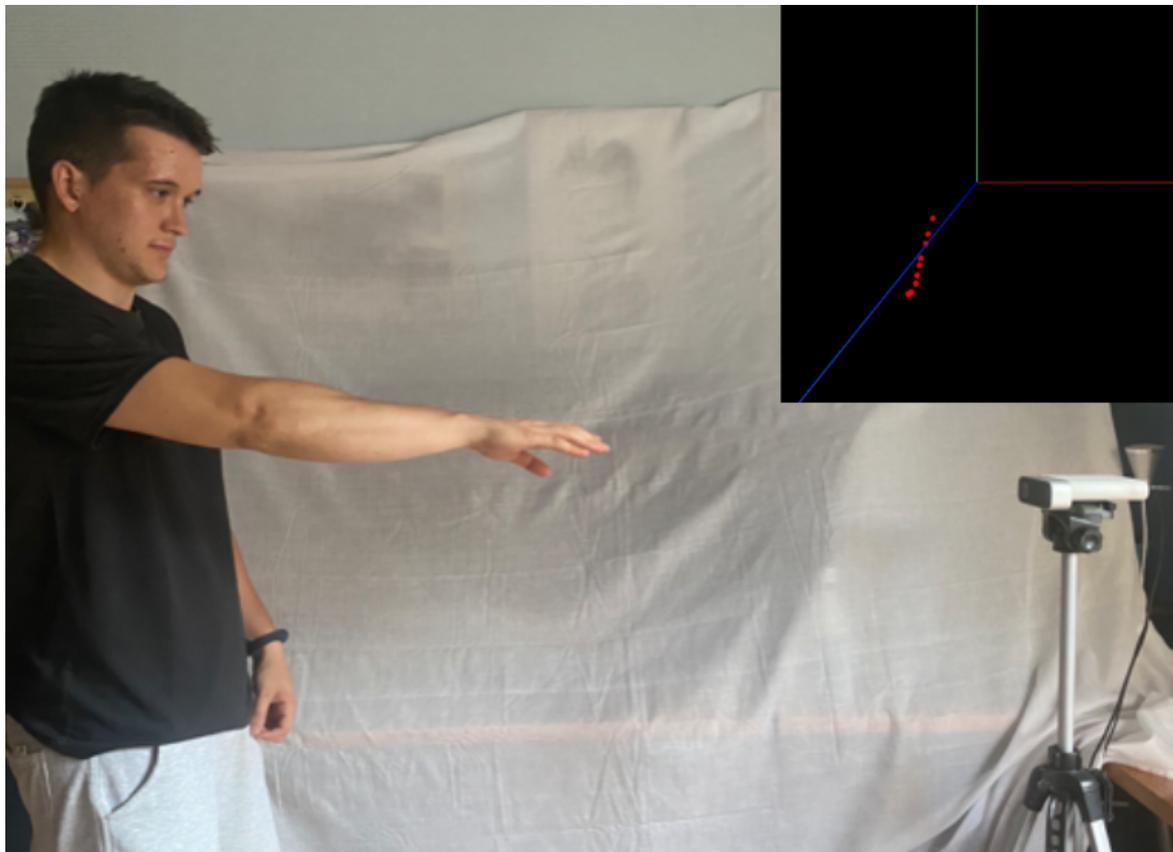


Figura 5.3: Creación de un ejercicio

### 5.2.3. Sistema de seguimiento de articulaciones

Tal y como se ha descrito anteriormente el sistema de seguimiento de articulaciones es el encargado de obtener los datos de *tracking* corporal. Este sistema está compuesto por un módulo de obtención de datos tridimensionales, el cual obtiene los datos corporales del usuario y de sus movimientos. El módulo recibe un flujo continuo de datos del dispositivo *Kinect Azure* que procesa y obtiene las coordenadas en tiempo real de las articulaciones detectadas. El segundo módulo de este sistema es el encargado del procesamiento de los datos de la articulación obtenidos en el módulo anterior, para que el módulo de *tracking* pueda servirlos a las aplicaciones de rehabilitación.

Los módulos que interactúan con el dispositivo de *tracking* están desarrollados con el lenguaje de programación *C++*, pero utilizando la API para integrar código de este lenguaje dentro del entorno de *Node.js*, para que el módulo de *tracking* pueda servir los datos en *JavaScript*, ya que este lenguaje es el elegido para el desarrollo de las aplicaciones de rehabilitación.

### 5.2.4. Sistema de interacción por audio

La interacción por audio es un aspecto de gran importancia en el proyecto, ya que el usuario objetivo al que va dirigido el sistema interactúa con el entorno sin el sentido de la vista. Esta interacción se va a realizar en dos modalidades: a) reproducción de sonidos y b) reconocimiento de voz. La primera modalidad será utilizada para reproducir mensajes específicos al usuario, así como sonidos.

Esta modalidad está presente tanto para el rol de paciente como el de terapeuta. La modalidad de reconocimiento de voz será utilizada para que el paciente pueda interactuar con el sistema mediante comandos de voz, de esta forma podrá elegir entre los ejercicios a realizar e iniciar la grabación, todo ello mediante la voz.

El sistema de interacción por audio esta compuesto de tres módulos bien diferenciados, cada uno con una función específica, y son los siguientes:

- **Módulo de reproducción de sonidos:** encargado de la reproducción de sonidos específicos, utilizados tanto en la creación de ejercicios como en la realización de los mismos.
- **Módulo de transcripción de texto a audio:** módulo encargado de transcribir cualquier texto a audio, para su reproducción. Este módulo se utiliza para reproducir mensajes específicos al paciente o al terapeuta durante la utilización del sistema.
- **Módulo de reconocimiento de voz:** este módulo es el encargado de transcribir la voz del usuario a texto, para que posteriormente este texto sea procesado por el sistema y se realicen las acciones pertinentes.

Los tres módulos se han desarrollado en *Node.js*, de tal forma que quedan encapsulados en una *API Rest*, para que las aplicaciones de rehabilitación puedan realizar peticiones específicas, según su requerimiento. Tanto en el caso del módulo de transcripción de texto a audio como el de reconocimiento de voz se ha hecho uso de una API de *Google Cloud* que permite realizar estas acciones con una alta precisión, ya que éstas conllevan un grado de complejidad que *Google* resuelve aplicando técnicas de inteligencia artificial. Se ha utilizado un servicio externo porque es gratuito y de gran calidad, esto ha permitido que el desarrollo se haya centrado directamente en los objetivos del proyecto.

### 5.2.5. Backend

El sistema desarrollado en este proyecto requiere persistencia de datos, ya que tanto la aplicación del paciente como la del terapeuta requieren guardar datos para posteriormente poder recuperarlos. Por lo tanto, garantizar el acceso de forma correcta a estos datos siguiendo una arquitectura acorde al proyecto es de vital importancia. La arquitectura elegida para satisfacer los requisitos de persistencia de datos es la de una *API Rest* formada por una estructura de rutas acorde a las necesidades de las aplicaciones. La base de datos será mapeada haciendo uso de un mapeo objeto-relacional. Por lo tanto, el *backend* quedaría dividido en tres capas bien diferenciadas que serían las siguientes:

- **Router:** englobará todas las rutas<sup>1</sup> necesarias y procesará las peticiones de las aplicaciones. Para satisfacer estas peticiones hará uso del controlador correspondiente según la ruta en cuestión.
- **Controladores:** los controladores son la capa intermedia entre el *router* y el *ORM*. El objetivo de esta capa es garantizar la seguridad del *backend*, procesando los datos provenientes del *router*, para posteriormente hacer uso del *ORM*.
- **Entidad objeto-relacional (ORM):** encargado de abstraer la base de datos del sistema creando una base de datos orientada a objetos virtual, sobre la base de datos relacional.

El *backend* del sistema se ha desarrollado en el entorno de *Node.js* utilizando un *framework*, llamado *Express.js*, específico para el desarrollo *backend* en este entorno. La implementación de la base de datos relacional elegida es *MySQL*.

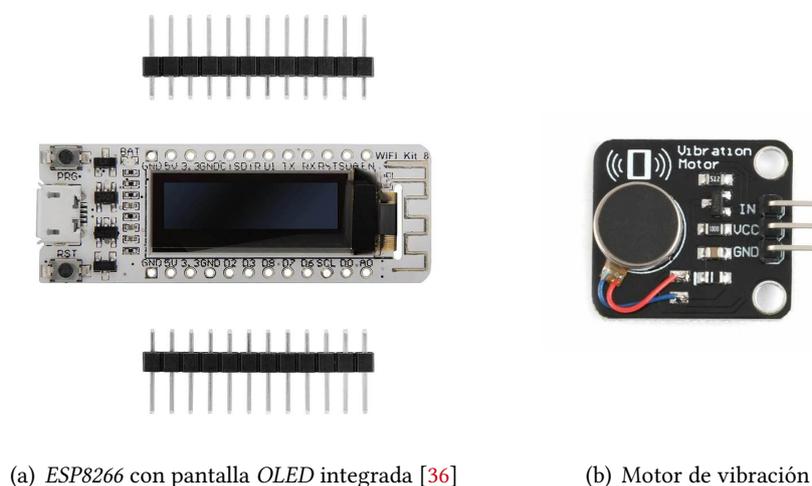
---

<sup>1</sup>Las rutas dirigen las solicitudes entrantes de la API a los recursos de *backend*

### 5.3. SISTEMA DE GUIADO BASADO EN VIBRACIONES INCORPORADAS A UN GUANTE

Los pacientes que utilizarán el sistema de guiado para rehabilitarse tendrán algún tipo de discapacidad visual. Este tipo de discapacidad les impedirá guiarse en los ejercicios terapéuticos con soluciones tecnológicas que requieran habilidades visuales, como se comentó en la Sección 3.4. Esta implicación ha supuesto estudiar nuevas formas de guiar al paciente en la ejecución de los ejercicios terapéuticos. Puesto que no existe un dispositivo que permita guiar al paciente sin utilizar el sentido de la vista para integrarlo en este proyecto, se ha optado por diseñar un dispositivo que lo resuelva.

El paciente no es consciente de cuál será la trayectoria del ejercicio terapéutico que debe realizar. Por lo tanto, debe ser guiado de principio a fin con el objetivo de alcanzar los distintos puntos de control que representan el movimiento. La retroalimentación para guiar al paciente será háptica, ya que únicamente con retroalimentación auditiva no se puede comprender con exactitud la trayectoria de un movimiento. Por ello, se emitirán vibraciones que indiquen la trayectoria objetivo, además de indicaciones en forma de audio que ayuden al paciente en la ejecución del ejercicio.



(a) ESP8266 con pantalla OLED integrada [36]

(b) Motor de vibración

**Figura 5.4:** Componentes *hardware* del dispositivo de guiado

El dispositivo de guiado conforma la parte *hardware* del sistema de guiado, formado por el microcontrolador ESP8266 con una pantalla OLED integrada y seis actuadores, resultando un prototipo en forma de guante que será utilizado por el paciente para la ejecución de los ejercicios terapéuticos. La elección de este microcontrolador está fundamentada en que gracias a su pequeño tamaño puede integrarse correctamente en el guante y poseer un módulo *wifi* integrado. Los actuadores son motores de vibración para que sean percibidos de forma háptica por el usuario.

#### 5.3.1. Proceso de guiado

El sistema de guiado se encarga de guiar al paciente para que complete los sucesivos puntos de control que forman el ejercicio. El sistema de guiado indica al paciente "de forma háptica como debe corregir la trayectoria para que el movimiento pautado por el terapeuta. La aplicación del paciente calculará la variación de cada coordenada respecto al punto de control actual. De la coordenada que más variación se calcule se obtendrá el sentido a corregir y será enviada al sistema de guiado para que active el actuador correspondiente que indique la dirección que debe seguir el paciente. En el caso de que la coordenada con más variación haya cambiado la aplicación de rehabilitación enviará esta nueva coordenada al sistema de guiado. En la Figura 5.5 se puede ver el proceso de guiado. Si la coordenada que más varía no cambia el actuador permanecerá encendido. De tal forma, que solo se

envía la coordenada más alejada una única vez, consiguiendo que el algoritmo sea eficiente y que el sistema de guiado pueda procesar los datos entrantes sin saturación.

El movimiento que forma un ejercicio terapéutico esta compuesto de un conjunto de puntos tridimensionales. Por tanto, el paciente debe ser guiado en las tres coordenadas que forman estos puntos. Cada una de estas coordenadas tiene un eje positivo y negativo, dependiendo cómo sea la trayectoria del movimiento los valores de estas coordenadas serán positivos o negativos. Por ello, se debe guiar al paciente en tres coordenadas distintas y el eje de cada una de ellas. Las coordenadas será indicadas al paciente de forma individual, es decir, no se corregirá la trayectoria del paciente en varios ejes de coordenadas al mismo tiempo, porque podría ser contra-intuitivo para el usuario, ya que se percibe vibraciones hápticas en dos direcciones en el mismo instante de tiempo podría resultarle difícil comprender hacia donde debe dirigirse. Por lo tanto, se indicará al paciente qué coordenada debe corregir y en qué sentido, dependiendo la lejanía de cada coordenada con el punto de control a conseguir. Es decir, se calculará la coordenada que más varíe con respecto al punto de control y se tratará de corregir, cuando se calcule una coordenada que varíe más que la anterior se realizará una nueva indicación. La indicación háptica consiste en que el actuador correspondiente vibre de forma continua para que el paciente perciba la dirección y el sentido que debe seguir en ese instante. Teniendo en cuenta que cada punto está formado por tres coordenadas y cada una de ellas tienen dos sentidos. Por cada coordenada se ha colocado dos actuadores en el dispositivo de guiado, de tal forma que se pueda guiar al paciente para cualquier dirección que se requiera en el ejercicio.

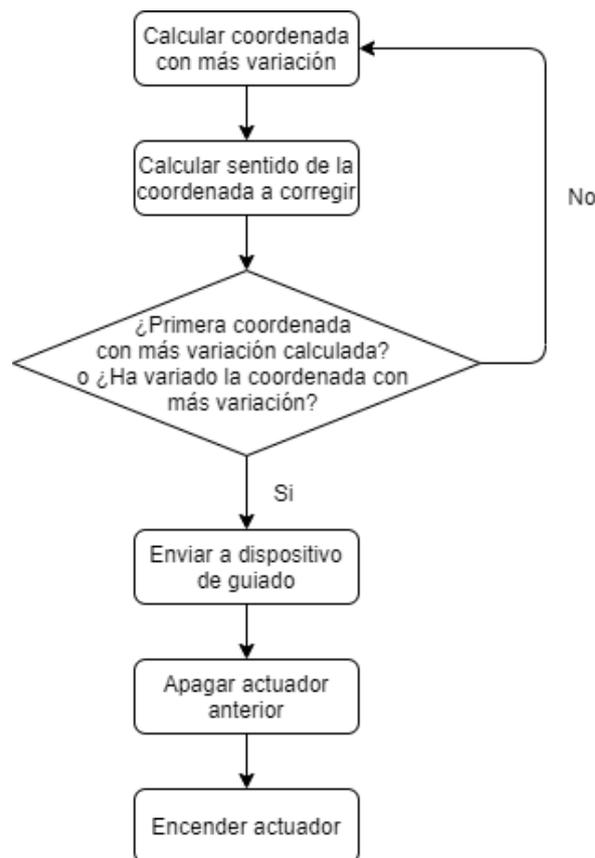


Figura 5.5: Diagrama de flujo del proceso de retroalimentación háptica

En la Figura 5.6 se puede ver el dispositivo de guiado desarrollado y su correspondiente representación virtual. También se puede apreciar la coordenada a corregir por cada actuador. Como ya se ha comentado anteriormente, cada actuador tendrá como objetivo corregir una coordenada y un sentido, la elección de éstos ha dependido del eje de coordenadas que utiliza el dispositivo de *tracking* que se puede ver en la Figura 5.7. Por lo tanto, con este eje de coordenadas y el dispositivo de guiado

orientado de tal forma que la palma de la mano queda colocada en el eje Y se pueden establecer las equivalencias entre los distintos actuadores. De tal forma que cada actuador tratará de impulsar al paciente al sentido contrario de la coordenada más alejada en cada instante. Por esta razón, se puede apreciar como los motores de vibración están situados en el sentido contrario al que pretenden generar movimiento.

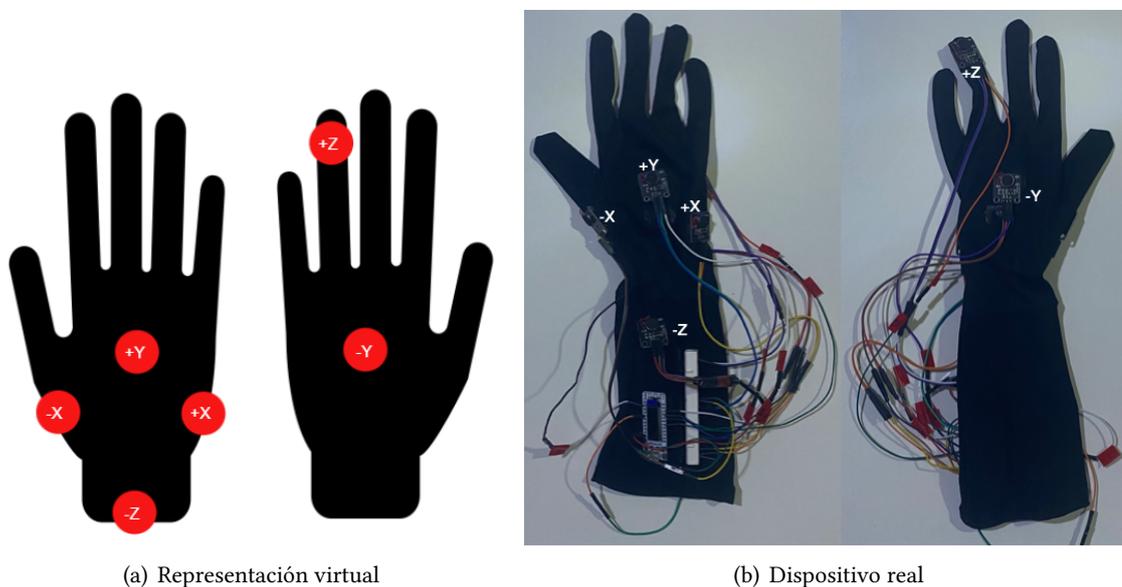


Figura 5.6: Dispositivo de guiado

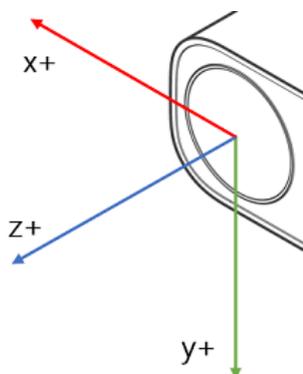


Figura 5.7: Eje de coordenadas utilizado

### 5.3.2. Implementación

La implementación del sistema de guiado se ha desarrollado utilizando el paradigma de programación de orientación a objetos en *C++*. De esta manera, se crean objetos que cumplen funciones específicas dentro del sistema haciendo que el código sea más escalable. En el diagrama de clases de la Figura 5.8 se pueden apreciar las distintas clases creadas para cumplir los requerimientos del sistema.

El objeto principal del sistema es el comunicador encargado de conectarse con el computador principal a la hora de realizar un ejercicio. Para implementar esta comunicación se ha desarrollado una clase, la cual encapsula toda la lógica necesaria (ver listado de código 5.1). El comunicador está formado por dos métodos principales:

- Método para iniciar la comunicación (*init*): en primer lugar realiza una conexión a la red mediante *WiFi* gracias a la librería *WifiManager*<sup>2</sup>. Una vez conectado el dispositivo se inicia

<sup>2</sup><https://github.com/tzapu/WiFiManager>

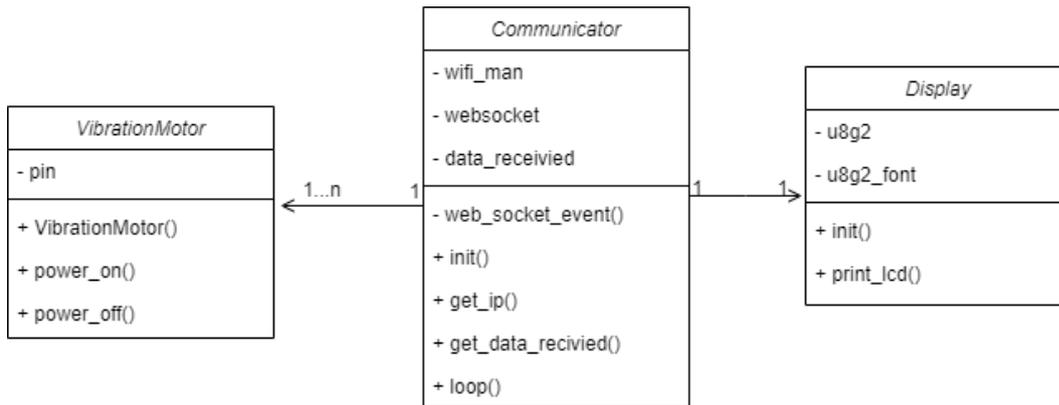


Figura 5.8: Diagrama de clases del sistema de guiado

el *WebSocket* y se establecen los eventos que se darán durante la comunicación. El evento principal para el caso de este sistema será el de recepción de datos en forma de texto, en el cual se recibe un identificador numérico en forma de carácter que será convertido a tipo entero y almacenado en un parámetro del objeto creado. Este parámetro será accedido para encender el actuador correspondiente.

- Método para ejecutar el *WebSocket* (*loop*): este método consiste en poner el *WebSocket* a la escucha para recibir los datos esperados. Dependiendo los datos que se reciban durante la ejecución, se lanzarán unos eventos u otros, definidos durante la creación del *WebSocket*.

Además del comunicador, el sistema está formado por otros dos objetos más. El primero de ellos será el que representa un motor de vibración; al iniciar el sistema se crea un vector de seis objetos de este tipo, uno por cada actuador *hardware* que tiene el guante. El segundo de los objetos representa la pantalla integrada del microcontrolador, con este objeto se muestra cierta información relevante como la dirección IP del dispositivo al iniciar o el actuador que está encendido en cada momento durante el proceso de rehabilitación. Esta información no es relevante para el paciente con problema de visión, ya que se presupone que no podrá apreciarla, aún así se ha utilizado porque no repercute significativamente en el rendimiento.

Listado 5.1: Clase comunicador del sistema de guiado

```

1 class Communicator
2 {
3 private:
4     WiFiManager wifi_man;
5     WebSocketsServer websocket{WEBSOCKET_PORT};
6
7     void web_socket_event(uint8_t num, WStype_t type, uint8_t ↵
8         ↵ *payload, size_t length)
9     {
10         switch (type)
11         {
12             case WStype_DISCONNECTED:
13                 break;
14             case WStype_CONNECTED:
15                 {
16                     websocket.sendTXT(num, CONNECT_MSG);
17                     break;
18                 }
19             case WStype_TEXT:
20                 {
21                     sscanf((char *)payload, "%d", &data_recivied);
22                 }
23                 break;
  
```

```

24     default:
25         break;
26     }
27 }
28
29 public:
30     int data_recivied = MY_NULL;
31
32     void init()
33     {
34         /* Init Wifi Connection */
35         wifi_man.autoConnect("ESP8266Temp");
36
37         /* Init WebSocket */
38         websocket.begin();
39         websocket.onEvent(std::bind(&Communicator::web_socket_event,
40         this, std::placeholders::_1, std::placeholders::_2,
41         std::placeholders::_3, std::placeholders::_4));
42     }
43
44     std::string get_ip()
45     {
46         return WiFi.localIP().toString().c_str();
47     }
48
49     void loop()
50     {
51         websocket.loop();
52     }
53 };

```

### 5.3.3. Comunicación

La conexión entre la aplicación de rehabilitación y el sistema de guiado debe ser eficiente garantizando que las órdenes lleguen en tiempo real a los actuadores, para que el paciente sea guiado correctamente. El sistema de guiado estará situado en la articulación del paciente, y es conveniente que la conexión se realice de forma inalámbrica. Esto es debido a que una conexión cableada entre el dispositivo de guiado y el computador principal limitaría los movimientos a realizar por el paciente. Por este motivo, se ha elegido un microcontrolador con un módulo *wifi* que implemente una conexión inalámbrica. El dispositivo de guiado será un dispositivo más en la red local del usuario y el computador principal será otro, por lo tanto, esta conexión se realiza entre dispositivos de la misma red, garantizando una conexión lo suficientemente estable. Durante el estudio inicial se eligieron dos posibilidades de comunicación entre el computador principal y el dispositivo de guiado: mediante una *API Rest*<sup>3</sup> o un *WebSocket*<sup>4</sup>.

La implementación de una *API Rest* en el sistema de guiado conllevaría que el computador principal realizara peticiones para enviar la dirección más alejada al sistema de guiado. Este tipo de conexión podría ser válida, en el caso de que no se requieran demasiadas peticiones, ya que si el computador necesita enviar un gran número de peticiones, el sistema de guiado podría saturarse. Esto es debido a que con este tipo de peticiones, se envía gran cantidad de datos, y se debe responder a cada una de ellas. Por lo tanto, en el supuesto de que se envíen un número muy alto de peticiones se tardaría en procesarlas demasiado tiempo, perdiendo la capacidad de guiar en tiempo real al paciente. Teniendo

<sup>3</sup>Una *API Rest* es una interfaz de programación de aplicaciones que se ajusta a la arquitectura cliente-servidor de transferencia de estado representacional (*REST*). De esta forma, es posible la creación de distintos tipos de servicios con peticiones *HTTP* [54]

<sup>4</sup>El protocolo *WebSocket* permite la comunicación bidireccional entre cliente y servidor sobre un *socket TCP* sin necesidad de abrir una conexión *HTTP* cada vez que necesiten enviar información [38]

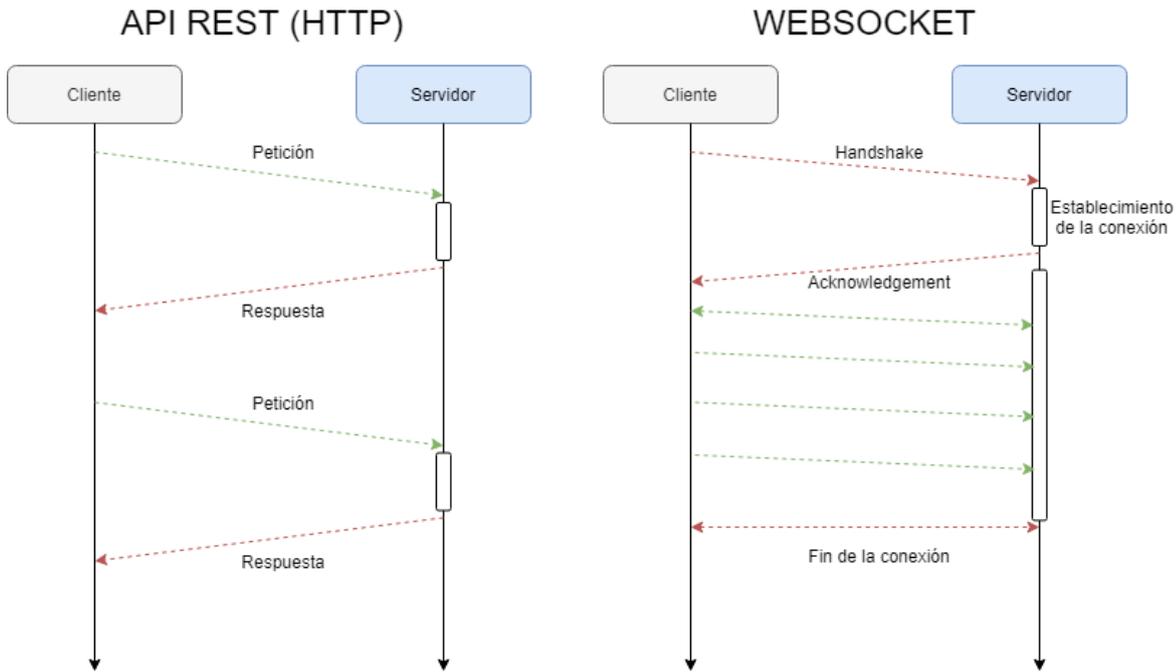


Figura 5.9: Comparación peticiones API Rest y WebSocket

en cuenta estos supuestos no se podría dar como válida esta solución, ya que no se puede asegurar que la cantidad de peticiones vaya a ser reducida.

La implementación mediante un *WebSocket* podría paliar los problemas comentados, ya que los datos se procesarían de forma mucho más eficiente. Como se puede observar en la Figura 5.9 con esta implementación se establecería una conexión bidireccional entre la aplicación de rehabilitación (cliente) y el sistema de guiado (servidor). Una vez establecida la conexión cliente y servidor podrán enviar datos en ambas direcciones por un canal abierto. Puesto que este canal de información no se cierra hasta finalizar el ejercicio se consigue que el envío y procesamiento de la información se realice de forma muy eficiente, garantizando que las órdenes al paciente sean captadas en tiempo real.

## 5.4. SEGUIMIENTO DEL MOVIMIENTO DE ARTICULACIONES

En el Capítulo 3 se justificó por qué es necesario utilizar un dispositivo de seguimiento corporal o *tracking* para el desarrollo de este proyecto. Este dispositivo deberá utilizar las tres dimensiones del espacio para representar los movimientos corporales correctamente. El dispositivo elegido ha sido el *Azure Kinect DK*, ya que cumple perfectamente las especificaciones del proyecto como se comenta en la Sección 3.4.

### 5.4.1. Visión general

La aplicación está construida en base a una librería *open source* de *Node.js* que procesa concurrentemente los datos que provienen del dispositivo de forma concurrente. Una vez que los datos han sido procesados se transforman para poder ser usados en *JavaScript* de forma nativa.

Este módulo se compone a su vez de otros tres módulos, como se puede observar en la Figura 5.10. El primero de ellos para obtener datos del dispositivo, que procesará el *SDK* en primer lugar. A continuación, los datos resultantes de este primer procesamiento serán accedidos de forma concurrente por el módulo de procesamiento, que obtendrá y procesará los datos de las distintas articulaciones

detectadas. Los datos resultantes de este segundo procesamiento serán tratados por el *wrapper*<sup>5</sup>, dando la posibilidad de acceder a ellos desde *JavaScript*. El módulo de comunicación será el encargado de comunicarse con el resto de aplicaciones para que accedan a la información procesada del *tracking corporal*.

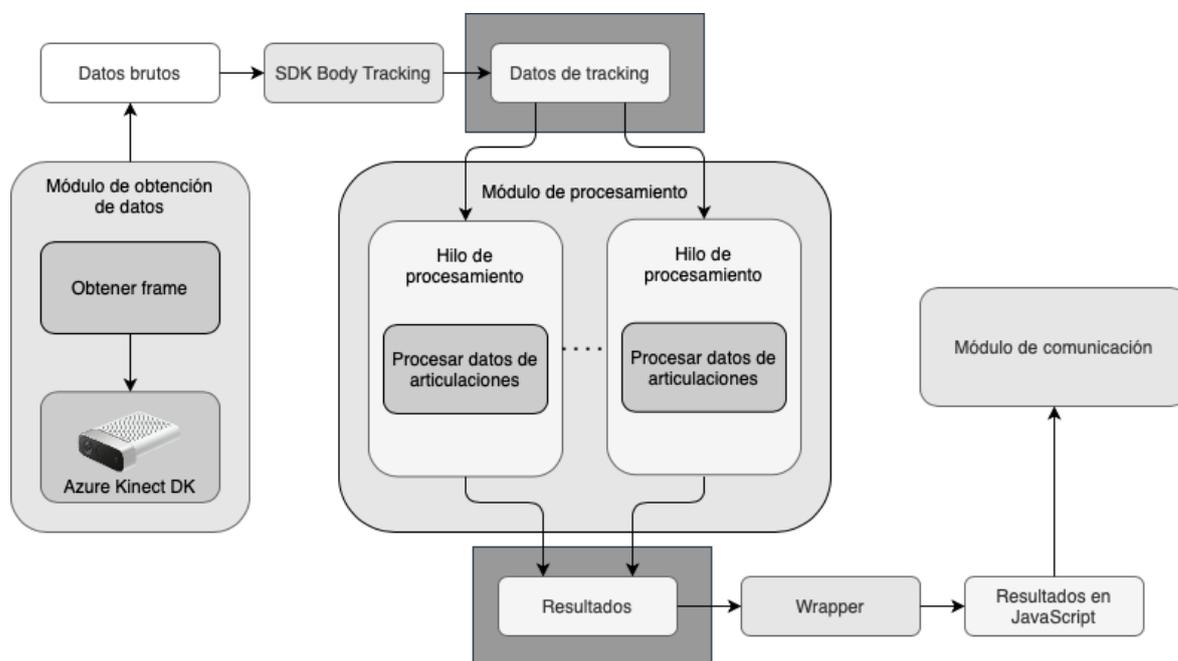


Figura 5.10: Esquema general del sistema de seguimiento de articulaciones

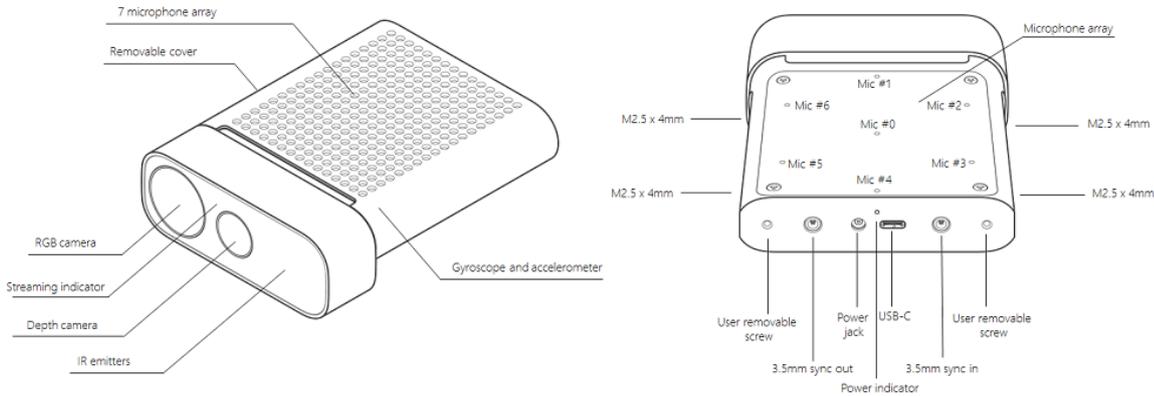
Una característica fundamental de esta aplicación es que debe procesar los datos del *tracking* en tiempo real, ya que el usuario debe recibir el *feedback* de la ejecución mientras realiza el movimiento pertinente. Por lo tanto, el procesamiento se realiza de forma concurrente para mejorar el rendimiento y así, tanto la aplicación del terapeuta como la del paciente recibirán la información del dispositivo adecuadamente, posibilitando a ambas interactuar con el usuario de la forma correcta.

#### 5.4.2. Manejo del dispositivo

El dispositivo *Azure Kinect* puede ser usado en diferentes modos de funcionamiento. En este proyecto se hace uso de la función de *tracking* corporal. Esta función permite obtener una imagen tridimensional del cuerpo humano, que posteriormente se puede usar para realizar diferentes tareas. Para poder hacer uso de la función de *tracking* del esqueleto será necesario usar la cámara de profundidad. El funcionamiento de la cámara de profundidad se basa en la medición del tiempo que tarda la luz en viajar desde el sensor hasta la escena y viceversa. Para poder obtener una imagen tridimensional del entorno se necesita medir la distancia a cada punto de la escena. Para ello, se produce una emisión de rayos de luz, que chocan con los objetos que hay en la escena y vuelven hacia la cámara. Cuanto más cerca esté el objeto con el que impacta el rayo, antes volverá y más cerca estará de la cámara. Esto permite determinar la configuración de la escena (distribución de objetos) y determinar los valores de los píxeles que forman la imagen que representan la escena. La fuente de luz que se usa en el dispositivo es una luz infrarroja modulada en el espectro NIR. Esta luz se proyecta sobre la escena y posteriormente se detecta cuando regresa al sensor. La medición del tiempo que tarda la luz en viajar desde el sensor hasta la escena y viceversa se realiza mediante el uso de un cristal de cuarzo. El

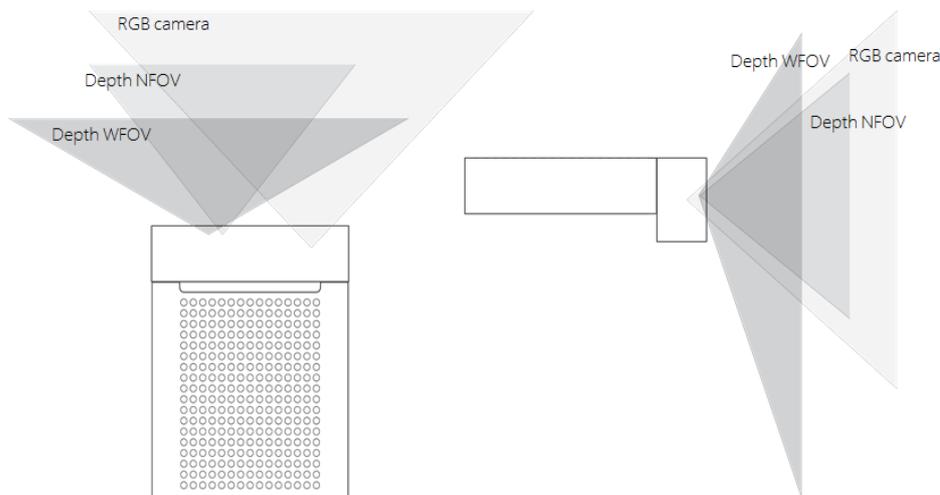
<sup>5</sup>Una función contenedora, *wrapper* en inglés, es una subrutina en una biblioteca de *software* cuyo propósito principal es llamar a una segunda subrutina con el mínimo cálculo adicional. *Node.js* está desarrollado en *C++*. Por lo tanto, no ejecuta código *JavaScript* directamente, sino que envuelve el código dentro de una función contenedora antes de la ejecución. El código desarrollado para utilizar el dispositivo de *tracking* está en *C++*. Por lo tanto, se llevará a cabo mediante un proceso similar al que realiza *Node.js* originalmente para acceder a ciertas partes del sistema operativo [16]

crystal de cuarzo es un material que tiene una propiedad única, que es la de tener una velocidad de propagación de la luz muy alta. Esto significa que la luz tarda muy poco tiempo en viajar desde el sensor hasta la escena y viceversa.



**Figura 5.11:** Componentes principales del dispositivo *Azure Kinect DK* [57]

La cámara toma las imágenes IR moduladas sin procesar y las envía al computador conectado al *Kinect*. Éste será el encargado de crear el mapa de profundidad de la imagen. Para ello, el kit de desarrollo de *Azure Body Tracking* utiliza el algoritmo de la técnica de extracción de contornos, que se basa en la extracción de una serie de líneas que forman los contornos de un objeto en una imagen. El algoritmo consiste en aplicar la función de diferencial sobre una imagen binaria, que permite identificar las diferencias entre dos imágenes consecutivas. La ejecución de este algoritmo para procesar las imágenes se puede realizar tanto en GPU como en CPU, aunque con el procesamiento con GPU normalmente se obtendrá un mejor rendimiento. El inconveniente es que solo son compatibles ciertas tarjetas gráficas de *Nvidia*, con lo cual el programa deberá estar capacitado para utilizar la CPU o GPU dependiendo el computador dónde se ejecute este procesamiento. Los modos de funcionamiento de la cámara de profundidad compatibles con el *tracking* corporal son *NFOV Binned* o *Unbinned* (modos de profundidad de campo de visión estrecho) y *WFOV Binned* o *Unbinned* (modo de profundidad de campo de visión ancho). Aunque, el *WFOV Unbinned* trabaja a un máximo de 15 fps, con lo cual su uso mermaría notablemente el rendimiento [57]. El tiempo de respuesta del dispositivo depende en gran medida de la elección entre estos modos de funcionamiento, por lo tanto, se debe elegir el que mejor rendimiento ofrezca.



**Figura 5.12:** Campo de visión de la cámara RGB y la profundidad [57]

En la aplicación de *tracking* se pueden configurar los siguientes aspectos relativos al dispositivo:

- **Tipo de procesamiento:** se puede optar por un procesamiento realizado por la CPU o por la

GPU, siendo este último el que normalmente proporcione mayor rendimiento. Aunque está limitado a cierto tipo de gráficas *NVIDIA*, por lo tanto, podría ser un inconveniente ya que limita el número de computadores compatibles con la aplicación. Por ello, se ha elegido el procesamiento por CPU porque el rendimiento es adecuado para el proyecto para el desarrollo y las pruebas del mismo.

- **Modo de funcionamiento de la cámara de profundidad:** se podrá elegir entre los modos *NFOV\_UNBINNED*, *NFOV\_BINNED* y *WFOV\_BINNED*, cada uno con sus características como se ha comentado anteriormente. Por defecto, se ha elegido el modo *NFOV\_UNBINNED*, modo de profundidad de campo de visión estrecho. Se ha elegido este modo, porque el *tracking* se hace solo de una persona simultáneamente, dado que en ambas aplicaciones de rehabilitación solo hay un usuario al mismo tiempo.

### 5.4.3. Obtención de datos corporales

Una vez iniciada la cámara se configura el objeto que representa el dispositivo en el computador, que utilizando funciones del SDK es capaz de transformar la imagen IR al mapa de profundidad y obtener la información captada del entorno. En este caso, se captará la información de las articulaciones del usuario de la aplicación.

El procesamiento de los datos provenientes del dispositivo se puede realizar de forma síncrona o asíncrona. De este tipo de procesamiento dependerá en gran medida el rendimiento de la aplicación, y con ello el *feedback* que recibirá el usuario durante su uso. Un procesamiento síncrono toma un *frame* para procesarlo, una vez procesado vuelve a tomar otro *frame* y así sucesivamente. Esto provocaría una gran pérdida de rendimiento al ser necesario esperar a que se complete el procesamiento de un *frame* para obtener uno nuevo. En cambio, el procesamiento de tipo asíncrono recibe un flujo continuo de *frames* que va almacenando en una cola. Los datos almacenados en esta cola los obtendrá la librería *Azure Kinect Body Tracking* que se encargará de procesarlos para obtener el mapa de profundidad de la imagen capturada, y posteriormente identificar las distintas extremidades del usuario. La cola de datos del *tracking* es accedida de forma concurrente para procesar los datos obtenidos y almacenarlos de forma correcta para su posterior tratamiento fuera de la aplicación de *tracking*. Al ser la cola una sección crítica se garantiza la consistencia de los datos mediante semáforos *mutex*, de esta forma solo un hilo de procesamiento podrá acceder a la estructura de datos al mismo tiempo.

El dispositivo es capaz de captar datos de las extremidades de varios cuerpos al mismo tiempo, dando como resultado un mapa de cuerpos, cada uno con sus correspondientes datos. Para esta aplicación solo se tendrá en cuenta la información de un cuerpo, ya que el sistema será usado por un único usuario al mismo tiempo. Cada una de las articulaciones (Figura 5.13) cuenta con las coordenadas en el espacio, un cuaternión<sup>6</sup> de las rotaciones de está y el nivel de confianza. Este último representado por cuatro niveles de confianza que van del menor nivel de confianza, que indica que la articulación se encuentra fuera del rango del dispositivo y por tanto no se pueda obtener información fiable, a el mayor nivel de confianza el cual indica que la información recibida de la articulación es fiable. Para el caso de este sistema se han utilizado las coordenadas espaciales, ya que con éstas se puede conseguir la independencia espacial entre usuarios sin excesivo coste computacional. El concepto de independencia espacial, se refiere a lograr que el usuario con rol de paciente pueda utilizar el dispositivo de *tracking* en un espacio diferente al usuario con el rol de terapeuta. De este modo, se consigue que el espacio del paciente y del terapeuta puedan ser distintos para realizar la rehabilitación de forma telemática.

---

<sup>6</sup>Los cuaterniones son una extensión de los números reales, proporciona una notación matemática para representar las orientaciones y las rotaciones de objetos en un espacio tridimensional.

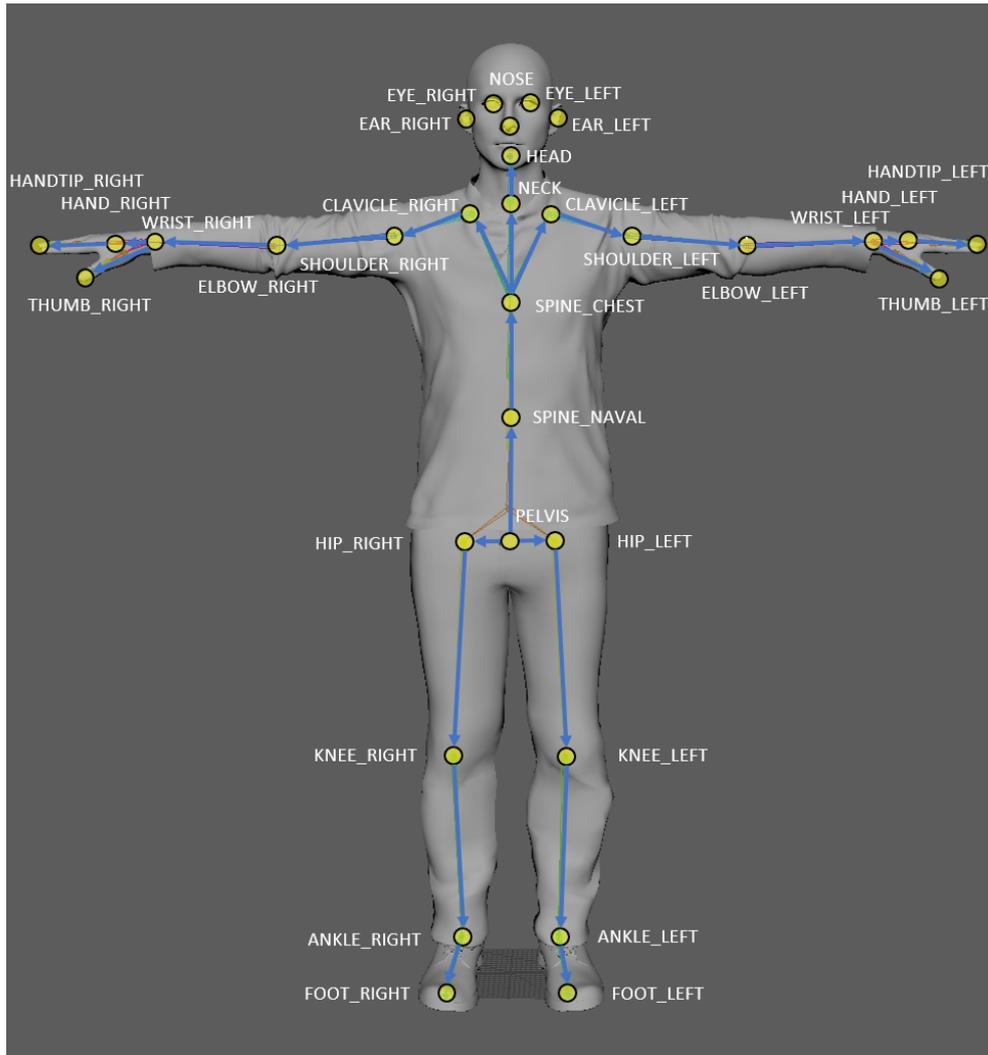


Figura 5.13: Articulaciones procesadas por *Azure Kinect DK* [56]

#### 5.4.4. *Tracking* de la mano

El sistema está diseñado en primera instancia para el seguimiento de movimientos de las manos, aunque el sistema es escalable, ya que se podrían realizar ejercicios terapéuticos para otras partes del cuerpo sin que el sistema sufra modificaciones de gran envergadura. El seguimiento de la articulación es igual tanto para la aplicación del terapeuta como para la del paciente, por esta razón se ha realizado una abstracción del *tracking* corporal en una aplicación que será utilizada de igual forma en ambos casos.

El *tracking* en el sistema se realiza con una máquina de estados finitos, formada por tres estados como se puede observar en la Figura 5.14. Tanto la aplicación del terapeuta como la del paciente se ejecutan con esta máquina de estados, aunque cada aplicación ejecuta su correspondiente lógica dentro de cada estado. El estado *STANDBY* se ejecuta al inicio, con el objetivo de captar la orden definida para iniciar el *tracking* corporal. El estado es igual en ambas aplicaciones, ya que se ha definido el mismo gesto para iniciar el *tracking*. En el estado *START* se procesarán los datos recibidos por el dispositivo con un objetivo específico, según la aplicación en cuestión. Se almacenarán datos en memoria principal para que posteriormente en el estado *END* sean procesados y se hagan persistentes en el sistema. En ambos estados la lógica de las aplicaciones difiere, porque el objetivo es distinto, como se verá posteriormente. El hecho de tener una máquina de estados otorga cierta escalabilidad, ya que se podría realizar una inclusión o modificación de estados en un futuro sin incurrir en un

desarrollo demasiado complejo.

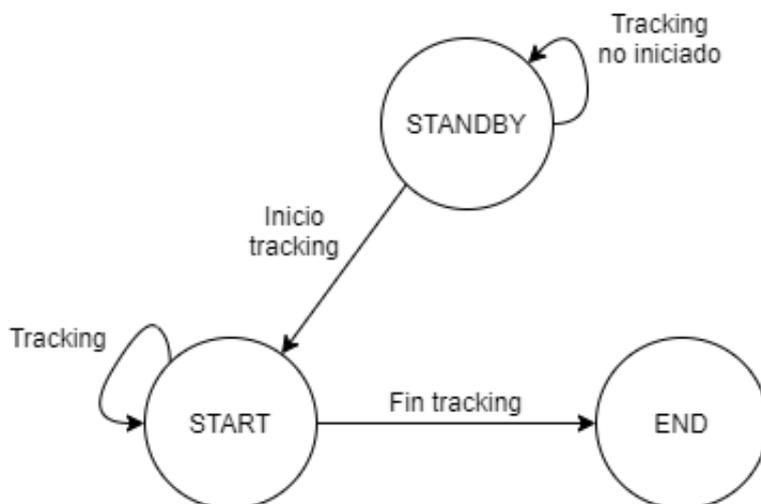


Figura 5.14: Máquina de estados finitos para el procesamiento de los datos de la articulación

#### 5.4.5. Procesamiento de datos tridimensionales

El origen de coordenadas del dispositivo se sitúa en el punto focal de la cámara. Como se puede apreciar en la Figura 5.15, el sistema de coordenadas se orienta de tal forma que el eje  $X$  positivo apunta a la derecha, el eje  $Y$  positivo hacia abajo y el  $Z$  positivo hacia delante, cada uno de ellos con su homólogo eje negativo en el sentido contrario. Teniendo en cuenta este sistema de coordenadas, para el correcto funcionamiento del sistema la mano del usuario estará orientada de tal forma que la palma de esta apuntará hacia el eje  $Y$  positivo.

La cámara de profundidad hace uso de dos iluminadores. Uno de ellos se utiliza en los modos de campo de visión de tipo estrecho (NFOV), que está alineado con la cámara de profundidad. El otro iluminador se utiliza en los modos de campo de visión amplio (WFOV), que está inclinado 1,3 grados más hacia abajo con respecto a la cámara de profundidad [80].

Los datos tridimensionales obtenidos del seguimiento de movimientos de las articulaciones son servidos a las aplicaciones de rehabilitación, de tal forma que cada una de ellas los utiliza para tomar las decisiones correspondientes según sus objetivos, como se explicará en las siguientes secciones del presente capítulo.

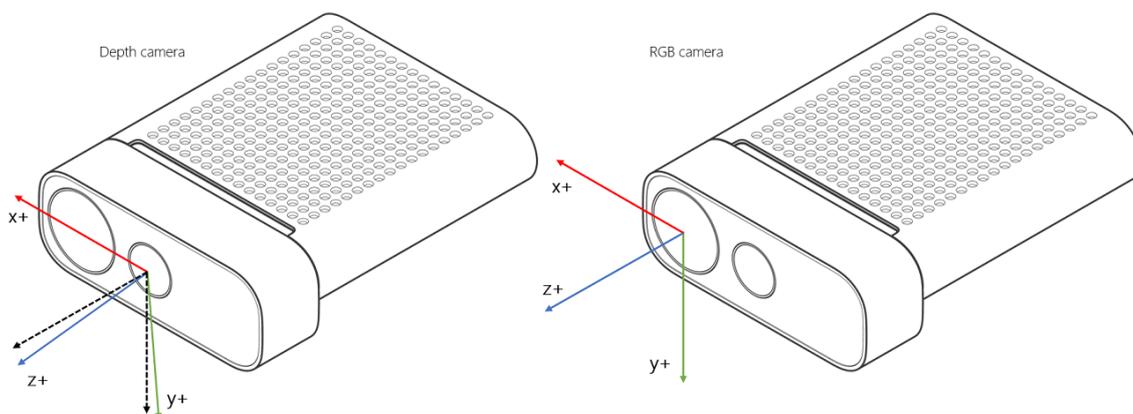


Figura 5.15: Coordenadas del Azure Kinect DK [80]

El dispositivo de *tracking* procesa los ejes de coordenadas de la forma que se muestra en la Figura 5.15. Teniendo en cuenta este procesamiento, para representar gráficamente las trayectorias de forma intuitiva la coordenada Y se procesará de forma antagonista a cómo lo hace el dispositivo. De esta manera, el eje positivo de la coordenada Y se situaría hacia arriba. Esto se realiza negando el valor de esta coordenada al representarlo gráficamente. Este procesamiento solo influye a la hora de representar gráficamente el ejercicio por pantalla.

#### 5.4.6. Comunicación con las aplicaciones de rehabilitación

Como se ha comentado anteriormente, la aplicación de *tracking* utiliza un *wrapper* para que se pueda utilizar el código C++ desde *JavaScript* con *Node.js*. *Node.js* está compuesto de una colección de muchas librerías de código abierto. Utiliza el motor V8<sup>7</sup> de código abierto de *Google*, escrito en C++, como motor *JavaScript* predeterminado que ejecuta el código. Cuando se instala *Node.js* en un equipo, generalmente instalamos una versión compilada de todo el código fuente junto con sus dependencias. De esta forma, no es necesario instalar estas bibliotecas manualmente. Sin embargo, en *Node.js* también se puede compilar a partir del código fuente de estas bibliotecas. Dado que *Node.js* está escrito en lenguajes de bajo nivel como *textitC* y C++, puede comunicarse con otros programas que están escritos en C y C++, y viceversa. Es posible cargar dinámicamente un archivo *DLL C* o C++ externo en tiempo de ejecución y utilizar su *API* para realizar algunas operaciones escritas en su interior desde un programa *JavaScript*. Esto es lo que se utiliza para cargar los ficheros necesarios del SDK de seguimiento de personas del dispositivo. De esta forma, se consigue hacer uso del dispositivo en aplicaciones desarrolladas en *JavaScript*, con todas las ventajas que esto implica, sin pérdida de rendimiento relevante que afecte negativamente al sistema. Las aplicaciones de rehabilitación importarán la librería directamente desde *JavaScript*, pudiendo hacer uso de todas las funciones del dispositivo.

### 5.5. APLICACIÓN DE ESCRITORIO CON LAS FUNCIONES DEL TERAPEUTA

El terapeuta es uno de los actores principales del sistema y, como tal, podrá realizar ciertas acciones dentro de él. La acción principal del terapeuta es crear los ejercicios terapéuticos para que el paciente puede rehabilitarse. En segundo lugar, el terapeuta podrá seguir las sesiones finalizadas por parte de los pacientes. Estas dos formas de interactuar con el sistema se realizarán gracias a la aplicación específicamente diseñada para el terapeuta, aunque, como ya se ha comentado, ciertas partes de esta aplicación están en un nivel de abstracción superior, ya que también serán utilizadas por la aplicación del paciente como se explicará posteriormente.

En la Figura 5.16 se muestra como la aplicación se ha dividido en varios módulos, cada uno de ellos con su respectiva funcionalidad y objetivo. El propósito de esta modularización es reutilizar estos módulos en la aplicación del paciente, ya que varias funcionalidades son similares en ambos perfiles de usuario. También se observa como los módulos interactúan con los otros dos subsistemas de la aplicación, como son la aplicación de *tracking*, explicada anteriormente, y el *backend* del sistema, que se explicará en la siguiente sección.

---

<sup>7</sup>V8 es un motor de código abierto para *JavaScript*, que puede ser usado en el lado del cliente y del servidor. [58].

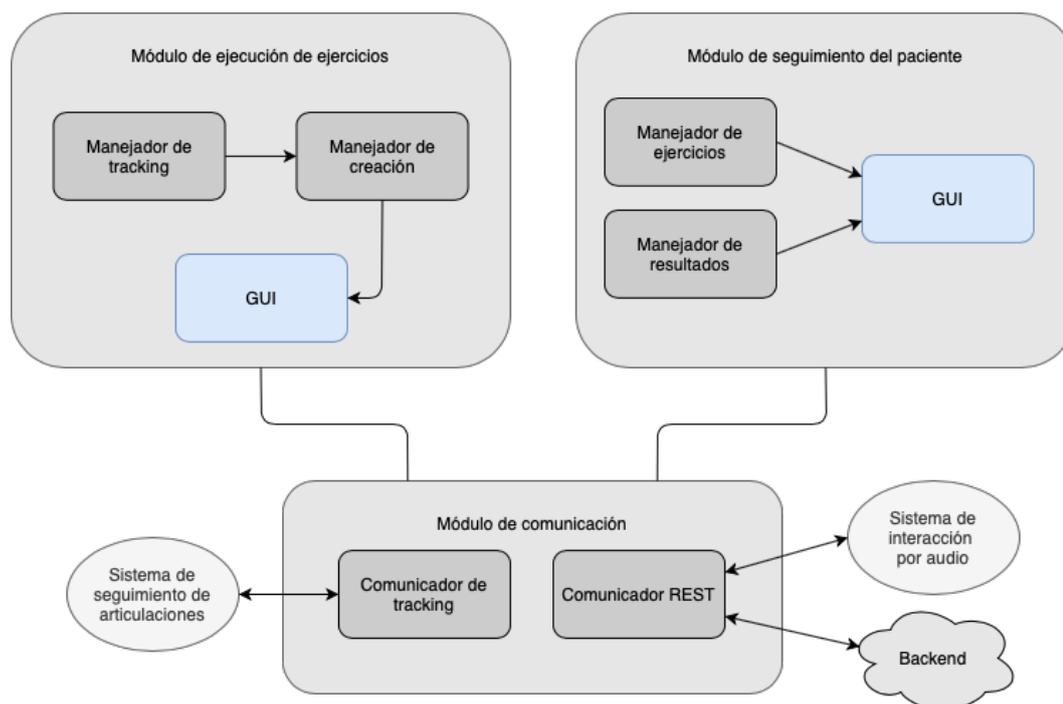


Figura 5.16: Esquema general de la aplicación del terapeuta

### 5.5.1. Entradas y salidas

La aplicación del terapeuta recibe diferentes datos de entrada al interactuar con el entorno. A través de estas entradas se recibe una serie de datos y con ellos el sistema produce diferentes salidas correspondientes.

Las entradas se podrían categorizar en dos tipos, que son las siguientes:

- **Computador:** los datos recibidos del computador principal serían las acciones relacionadas con los distintos periféricos, como son el teclado y el ratón en la aplicación de escritorio del sistema. Con estas entradas, el sistema guarda información e inicia las acciones correspondientes según el tipo de usuario, en caso del terapeuta todo lo relativo a la creación de ejercicios o el seguimiento del paciente.
- **Movimientos corporales:** los movimientos corporales se podrían considerar como el flujo de información más importante que recibe el sistema, gracias al dispositivo de *tracking* corporal. Gracias al dispositivo de *tracking*, se reciben gestos para iniciar la grabación, además de la representación del movimiento. Con esta última información, en el caso del terapeuta, se guarda una representación del movimiento para su posterior reproducción por parte del paciente. Esta representación consiste en un vector de diferencias entre coordenadas.

Este subsistema, una vez procesados los datos de las diferentes entradas, produce una serie de salidas o resultados. La principal salida del subsistema son los ejercicios terapéuticos destinados a los pacientes.

### 5.5.2. Definición de ejercicios

Uno de los objetivos del presente proyecto es la definición de ejercicios por parte del terapeuta, de tal forma que se pueda crear un plan de rehabilitación para que, posteriormente, el paciente lo reproduzca. La creación o definición de un ejercicio consiste en capturar un movimiento y almacenarlo de forma efectiva en el sistema. Se considera que el ejercicio ha sido almacenado de forma efectiva cuando éste se puede reproducir con independencia espacial, es decir, que la ejecución del ejercicio no dependa de

una posición específica del usuario al realizarlo, sino que pueda realizarse en otro lugar o situándose más lejos o más cerca del dispositivo de *tracking*. Esto es un punto de vital importancia en el proyecto, porque el objetivo principal es que el paciente pueda rehabilitarse de forma remota. Por tanto, se presupone que el terapeuta definirá los ejercicios en lugar y el paciente los llevará a cabo en otro lugar distinto. Por ello, el sistema debe poder adaptarse y tener la capacidad de reproducir el movimiento original del ejercicio en un ambiente distinto. La forma de conseguir esta independencia espacial al ejecutar un ejercicio en un lugar distinto al de creación, se explica de manera detallada en la Sección 5.6.2.

**Listado 5.2:** Algoritmo de definición de ejercicios

```

1 Algoritmo definicion_ejercicios
2   ejercicio<-[]
3   puntoActual<-[]
4   puntoAnterior<-[]
5   datosEsqueleto<-[]
6   estado<-STANDBY
7
8   Mientras grabando Hacer
9     Segun estado Hacer
10    STANDBY:
11      Si gestoInicio Entonces
12        estado = START
13      Fin Si
14    START:
15      puntoActual = datosEsqueleto[articulacionId]
16      Si !coordenadasAnteriores |
17        variacionMedia(puntoAnterior, puntoActual) > ←
18          ↪ UMBRAL_MAXIMO Entonces
19            puntoDeControl = puntoActual
20            ejercicio[indice]<-puntoDeControl
21          Fin Si
22
23        puntoAnterior=puntoActual
24
25      Si variacionMedia(ejercicio[0], puntoActual) < ←
26        ↪ UMBRAL_MAXIMO Entonces
27          estado=END
28        Fin Si
29      END:
30        finEjercicio()
31    Fin Segun
32  Fin Mientras
33 FinAlgoritmo

```

En el listado 5.2 se puede observar la máquina de estados con la implementación del terapeuta. Para el estado inicial *START*, el terapeuta se situaría delante del dispositivo de seguimiento corporal y con un gesto definido en el sistema se iniciará el *seguimiento* de la articulación objetivo. En el estado *START*, el terapeuta irá realizando el movimiento y el sistema guardará en memoria un punto de control cuando considera que se ha realizado el suficiente cambio en la trayectoria. De esta forma, se irá creando una estructura de datos con los puntos de control creados. La estructura de datos consiste en un *array* formado por un número variable de puntos, y cada uno de estos puntos representa la posición de la articulación con las coordenadas X, Y y Z. El ejercicio se considera finalizado cuando la articulación vuelve a la posición inicial con cierto umbral, una vez alcanzado este punto el terapeuta podrá elegir si la ejercicio se guarda en el sistema o no. En el caso de que el terapeuta decida guardarlo el sistema procesará los puntos de control creados, para calcular las diferencias entre ellos siguiendo la Función 5.1, como se puede ver en el listado de código 5.3. Este cálculo es primordial en el algoritmo, ya que este vector de diferencias vectoriales permitirá reproducir los puntos de control en el espacio vectorial del paciente, consiguiendo la independencia espacial comentada anteriormente.

$$\sum_{i=0}^{i=n} (x_{i+1} - x_i, y_{i+1} - y_i, z_{i+1} - z_i) \quad (5.1)$$

**Listado 5.3:** Algoritmo para calcular las diferencias vectoriales entre puntos de control

```

1 Algoritmo calculoDiferencias
2   ejercicio<-[]
3   diferencias<-[]
4
5   Para indice<-0 Hasta indice < longitud(ejercicio) && indice+1 ←
6     ← < longitud(ejercicio) Con Paso indice=indice+1 Hacer
7     diferencias[indice]=diferencia(ejercicio[indice], ←
8     ← ejercicio[indice+1])
9   Fin Para
10 FinAlgoritmo

```

### 5.5.3. Seguimiento del paciente

Una vez creados los ejercicios por parte del terapeuta, ya están a disposición de los pacientes. La correcta ejecución de éstos es primordial para que el paciente se recupere de su lesión, por ello se requiere llevar a cabo un seguimiento. En este seguimiento del paciente, el terapeuta debe ser capaz de discernir cuál es la calidad de ejecución que está llevando a cabo el paciente, por lo tanto es de vital importancia reflejar el movimiento del paciente en el espacio junto con una comparación con el ejercicio originalmente creado. De esta forma, se podrá observar cuánto difieren ambos ejercicios y tomar decisiones en base a ello, como podría ser cambiar de ejercicio.

Como ya se ha comentado, el ejercicio original es guardado en el sistema como un vector de diferencias entre los puntos de control creados durante la creación del ejercicio por parte del terapeuta. Esta decisión de diseño, cobra también gran relevancia en este punto porque para supervisar se necesita representación gráfica en el espacio vectorial del paciente junto con el ejercicio original. Ambos movimientos deben comenzar en el mismo origen de coordenadas, para que se puedan comparar correctamente. Conseguir esto es sencillo ya que al punto inicial del espacio vectorial del paciente se le sumará el primer elemento del vector de diferencias, consiguiendo el siguiente punto de la trayectoria de la representación del ejercicio original, y a este punto se le sumará el siguiente elemento del vector y así sucesivamente hasta conseguir todos los puntos. Uniendo estos puntos se conseguirá representar el movimiento original en el mismo espacio vectorial y con el mismo origen de coordenadas que la ejecución del paciente.

En la Figura 5.17 se puede apreciar gráficamente como se reproducen los distintos puntos de control que representan el ejercicio a realizar. Por un lado están las coordenadas del punto inicial y por otro un vector con todas las diferencias calculadas durante la creación del ejercicio. Este vector está conformado por una serie de tuplas de tres posiciones, cada una de ellas representa la diferencia en su correspondiente coordenada ( $difX$ ,  $difY$ ,  $difZ$ ). Con el primer punto y la primera tupla del vector de diferencias se obtendrá el primer punto de control. En base a este punto y a la siguiente tupla de diferencias se creará un nuevo punto de control, y así sucesivamente hasta representar todos los puntos de control en el espacio dimensional del paciente.

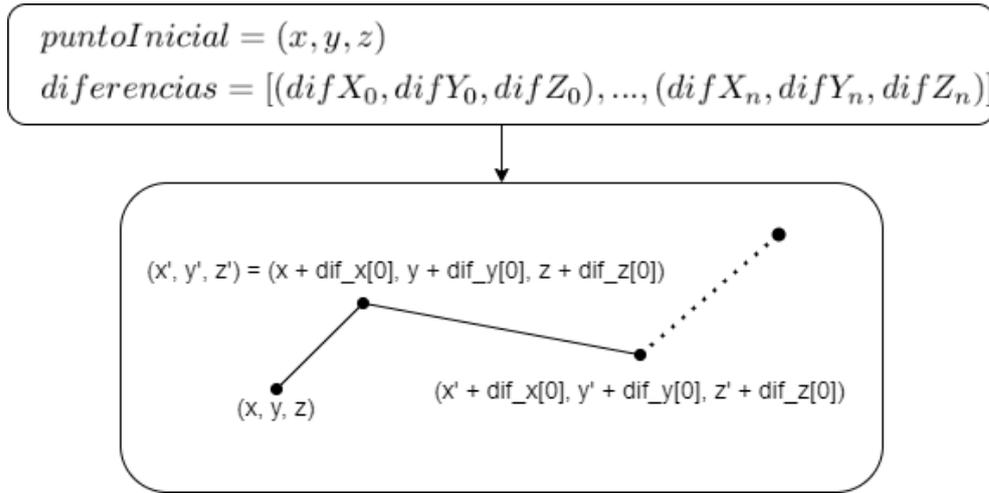


Figura 5.17: Reproducción de los puntos de control

## 5.6. APLICACIÓN DE ESCRITORIO PARA EJECUCIÓN DE EJERCICIOS

El actor principal del sistema es el paciente, porque el sistema desarrollado se enfoca en rehabilitar de lesiones físicas a personas invidentes con una serie de ejercicios terapéuticos, como se puede observar en la Figura 5.18.

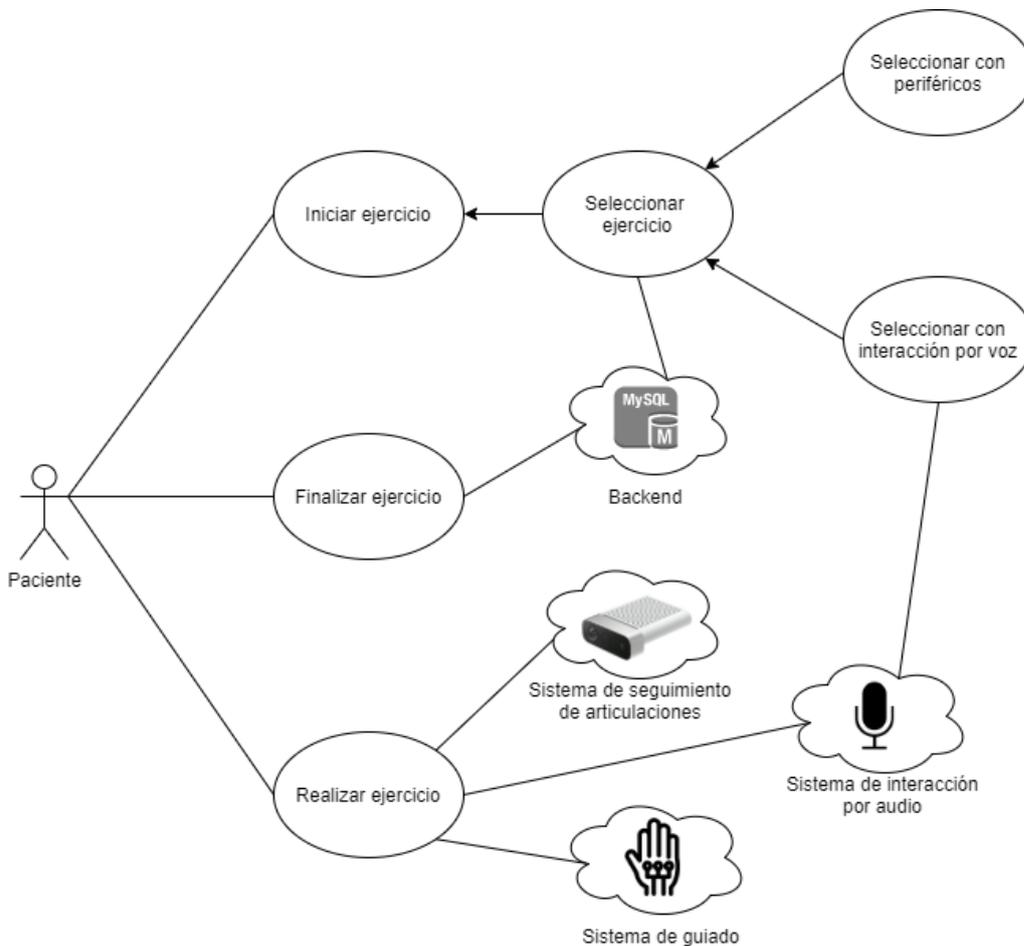


Figura 5.18: Diagrama de casos de uso del paciente

El usuario con rol de paciente utilizará este subsistema para rehabilitarse de forma efectiva, teniendo en cuenta su discapacidad. El paciente interactuará con el sistema mediante esta aplicación, diseñada específicamente para que los usuarios puedan utilizarla sin necesidad de visionar la pantalla. De esta forma, se garantiza que el usuario invidente sea prácticamente autónomo durante el proceso de rehabilitación. Esta aplicación comparte ciertas partes con la aplicación del terapeuta, haciendo que el sistema sea escalable y evitando duplicidades. Además, se comunicará con otros subsistemas según lo requiera. En la Figura 5.19 se puede observar la división de la aplicación en módulos, cada uno de ellos con su respectiva funcionalidad según su objetivo.

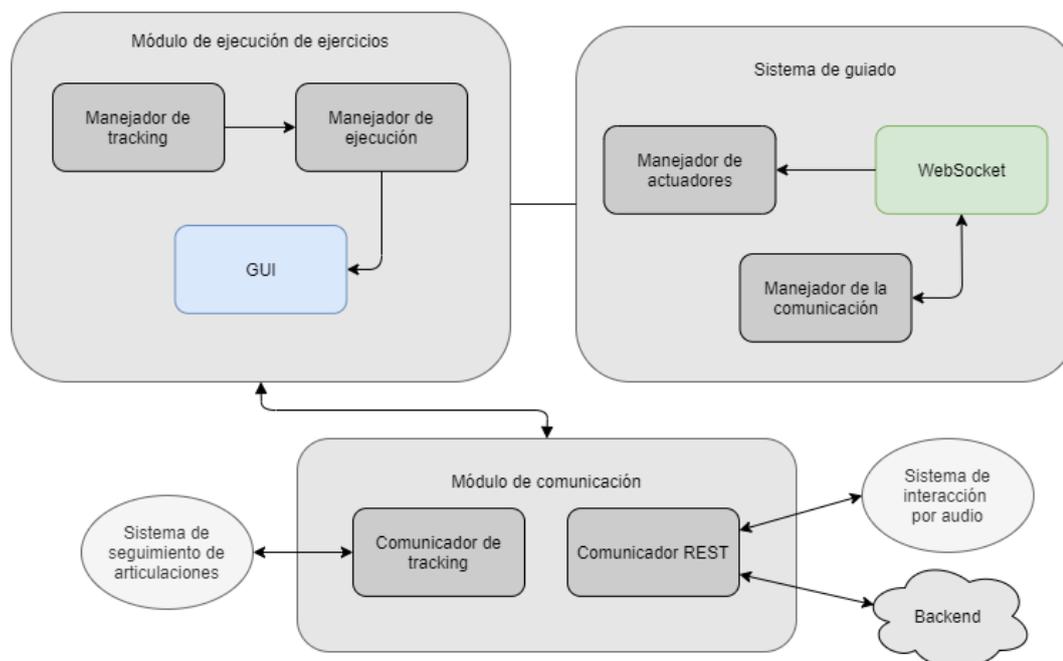


Figura 5.19: Esquema general de la aplicación del paciente

### 5.6.1. Entradas y salidas

La aplicación del paciente recibe datos de diferentes entradas cuando interactúa con el entorno y según los datos recibidos produce una serie de datos de salida.

Las entradas se podrían categorizar en dos tipos, que son los siguientes:

- **Computador:** los datos recibidos del computador principal serían las acciones relacionadas con el mismo, al igual que con el terapeuta.
- **Movimientos corporales:** los movimientos corporales se podría considerar como el flujo de información más importante que recibe el sistema, como en el caso del terapeuta. Se reciben gestos para iniciar la ejecución del ejercicio, y una vez iniciado se recibe los datos del movimiento de la articulación a rehabilitar. Estos datos son suministrados por el sistema de seguimiento de articulaciones, que la aplicación del paciente utiliza para guiar al usuario durante su ejecución.
- **Comandos de voz:** puesto que el paciente es invidente es posible controlar la funcionalidad principal de la aplicación del paciente por medio de comandos de voz, todo ello gracias al sistema de interacción por audio, que procesará estos datos, como se puede observar en la Figura 5.20.

Este subsistema, una vez procesados los datos de las diferentes entradas, produce una serie de salidas o resultados. La principal salida de la aplicación del paciente serán los datos resultantes de las ejecuciones de los ejercicios. Además, durante la ejecución de un ejercicio el sistema produce una serie de salidas en forma de interacciones con el paciente, guiarlo durante la realización del ejercicio.

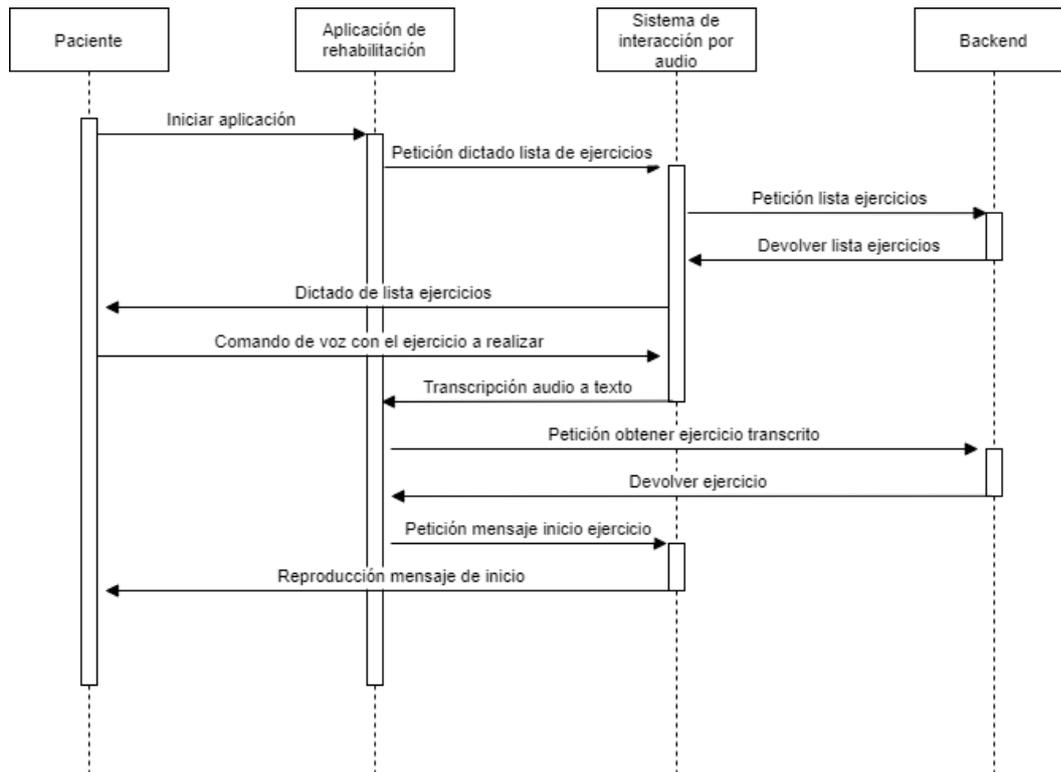


Figura 5.20: Diagrama de secuencia para iniciar ejercicio

### 5.6.2. Representación del ejercicio

Un ejercicio consiste en un serie de puntos de control que unidos forman un movimiento a reproducir, los cuales han sido creados por la aplicación del terapeuta. Estos puntos de control se crean durante la realización del ejercicio por parte del paciente. Al almacenar de forma persistente los puntos de control no se guarda la posición exacta en el espacio, sino que se guardan las diferencias entre los distintos puntos que representan el ejercicio que se ha creado. Debido a que se presupone que el espacio dimensional del terapeuta y del paciente será distinto. Por lo tanto, estos puntos de control deben ser dinámicos y debe existir la capacidad de reproducirlos en cualquier espacio dimensional. Esta independencia espacial se consigue con el cálculo de diferencias entre los puntos de control, como ya se ha explicado en la Sección 5.5.2.

$$\sum_{i=n}^{i=0} (x_{i+1} + x_i, y_{i+1} + y_i, z_{i+1} + z_i) \quad (5.2)$$

Listado 5.4: Algoritmo para recreación del ejercicio

```

1  Algoritmo recreacionEjercicio
2  posInicial <- []
3  diferencias <- []
4  puntosControl <- []
5
6  Para indice <- 0 Hasta indice < longitud(diferencias) Con Paso 1 Hacer
7      Si indice == 0 Entonces
8          puntosControl[indice] = suma(posInicial, diferencias[i])
9      SiNo
10         puntosControl[indice] = suma(puntosControl[i-1],
11             diferencias[i])
12     Fin Si
  
```

```

12   Fin Para
13
14   FinAlgoritmo

```

A la hora de realizar un ejercicio se debe aplicar el algoritmo (listado de código 5.4) que reproduce los puntos de control en el espacio del paciente siguiendo la Función 5.2, teniendo en cuenta que en el sistema un ejercicio es representado por conjunto de vectores de diferencias. Este algoritmo consiste en obtener el punto de inicio del paciente y a partir de éste sumar los distintos vectores. Con cada suma se obtendrá un nuevo punto que sumado el vector de diferencias correspondiente resultará otro punto y así sucesivamente hasta sumar consecutivamente todo. En la Figura 5.21 se pueden ver dos ejecuciones del mismo ejercicio. La trayectoria roja sería el ejercicio original con sus respectivos y sus coordenadas, en blanco se representa la trayectoria del movimiento del paciente. Como se puede observar las trayectorias de ambos ejercicios son las mismas, pero las coordenadas de los puntos de control son distintas, ya que cada ejecución tiene un espacio dimensional distinto. Por lo tanto, los puntos de control se crean en base al primer punto del paciente gracias al vector de diferencias comentado anteriormente.

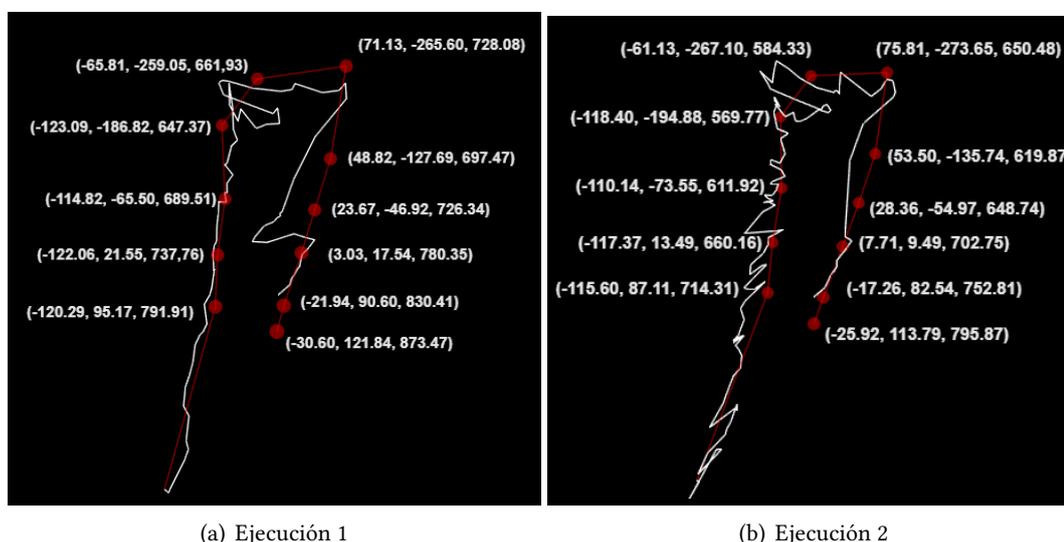


Figura 5.21: Ejecuciones de un ejercicio en distintos espacios dimensionales

### 5.6.3. Realización de ejercicios

El usuario de esta aplicación debe realizar una serie de movimientos con el objetivo de rehabilitar una articulación. Puesto que el usuario tiene una discapacidad visual, la aplicación es de gran importancia para que pueda rehabilitarse a distancia y de forma independiente. Teniendo en cuenta esto, se debe interactuar con el usuario de forma apropiada, es decir sin interacción visual alguna, ya que ésta no podrá ser percibida por el usuario.

El paciente tendrá disponible una lista de ejercicios, de los cuáles elegirá un ejercicio a realizar. Esta elección la podrá realizar mediante la voz, consiguiendo que el sistema sea accesible a personas con discapacidad visual. Una vez elegido el ejercicio a realizar el usuario indicará el inicio de la ejecución moviendo la cabeza, como se puede observar en la Figura 5.22. Este gesto que será detectado por el sistema para iniciar *tracking* de la articulación a rehabilitar y guiar al paciente hacia los distintos puntos de control para completar el ejercicio en cuestión. Este proceso será llevado a cabo con el algoritmo de ejecución de ejercicios. Una vez iniciado el ejercicio y recreados los puntos de control en el espacio dimensional del paciente el algoritmo realiza las siguientes acciones: i) guardar las coordenadas del primer punto; ii) si la variación media entre el punto actual y el punto de control a conseguir está dentro del umbral se pasará al siguiente punto de control y se actualizará el umbral

al valor por defecto; iii) se calcula la coordenada más alejada del punto de control actual, si es la primera vez que se calcula o la coordenada calculada es distinta a la anterior ésta se envía al sistema de guiado; iv) se comprueba si se ha completado el ejercicio, el criterio para completar un ejercicio es que se hayan completado todos los puntos de control, o que se hayan completado una cantidad representativa de los puntos de control y que la articulación se encuentre en la posición inicial; v) si no se ha completado el ejercicio se aumenta el umbral para conseguir el punto de control actual.

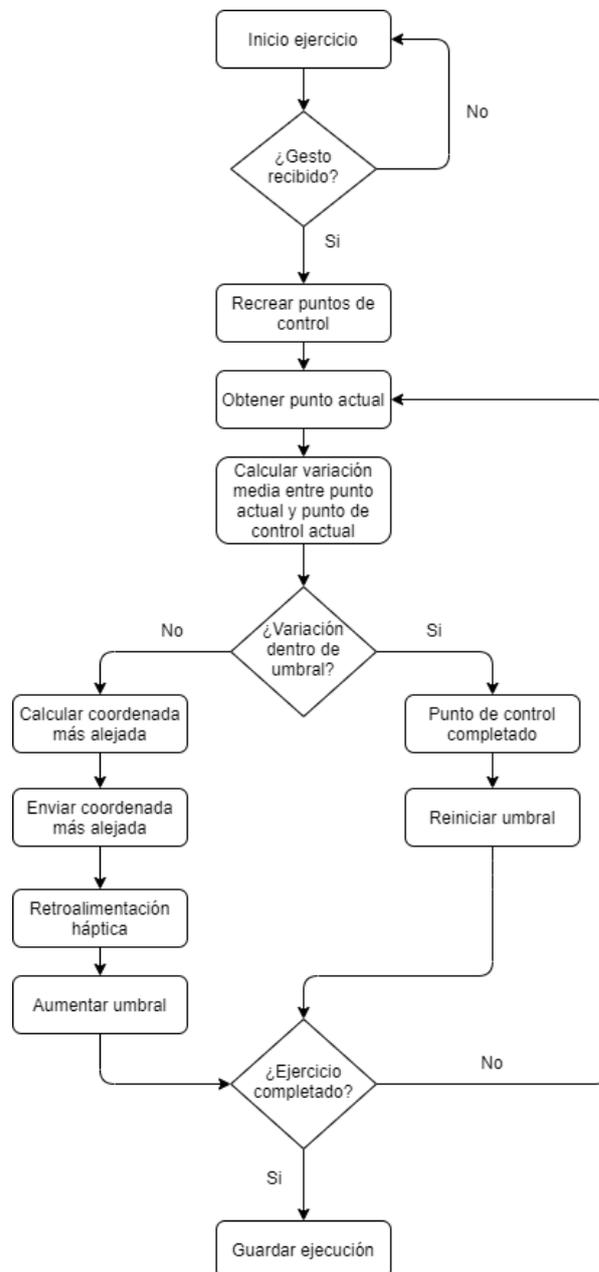
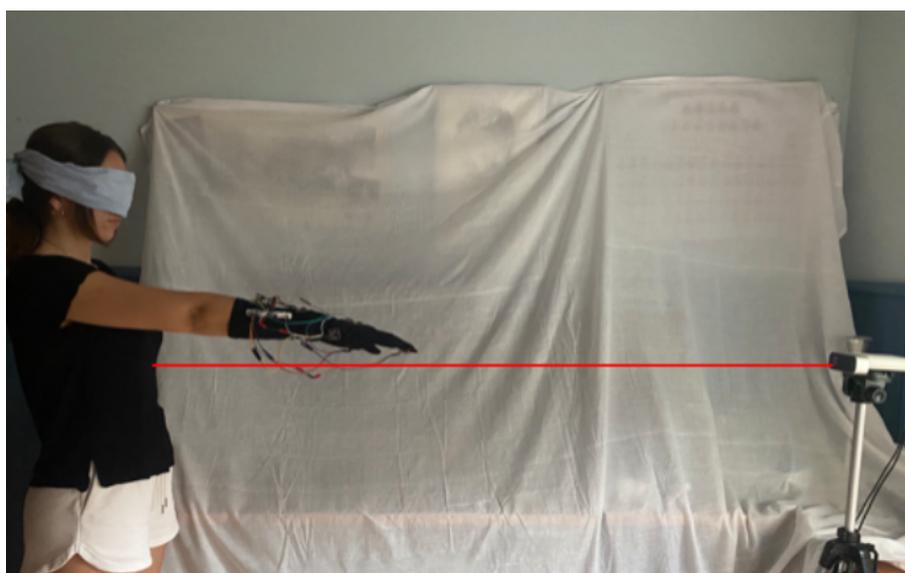


Figura 5.22: Diagrama de flujo ejecución ejercicio

En la Figura 5.23 se puede observar a dos pacientes realizando un ejercicio. Se puede apreciar como el espacio dimensional de ambos pacientes es distinto. La distancia entre el paciente y el dispositivo de seguimiento de articulaciones difiere entre ambos. Además, como cada paciente tiene una altura distinta, también afecta a los datos tridimensionales que capta el dispositivo. Por ello, se puede afirmar que la independencia espacial entre el ejercicio original y el espacio dimensional del paciente es una característica fundamental para que el sistema cumpla su función.



(a) Paciente 2



(b) Paciente 4

**Figura 5.23:** Pacientes realizando un ejercicio

## 5.7. BACKEND

Los sistemas que necesitan trabajar con datos persistentes requieren una arquitectura específica para este cometido. Para el presente proyecto, es necesario que tanto los ejercicios como las ejecuciones de éstos sean persistentes, para que se puedan acceder a ellos posteriormente. Por ello, se ha diseñado un subsistema que permita almacenar datos de forma persistente para las aplicaciones de rehabilitación.

En la Figura 5.24 se puede apreciar el diseño del *backend* del sistema, que esta formado de una base de datos y una *API Rest* que abstrae el acceso a los datos en forma de servicios. El *ORM* conectará con la base de datos para realizar las operaciones necesarias según lo requieran los distintos módulos.

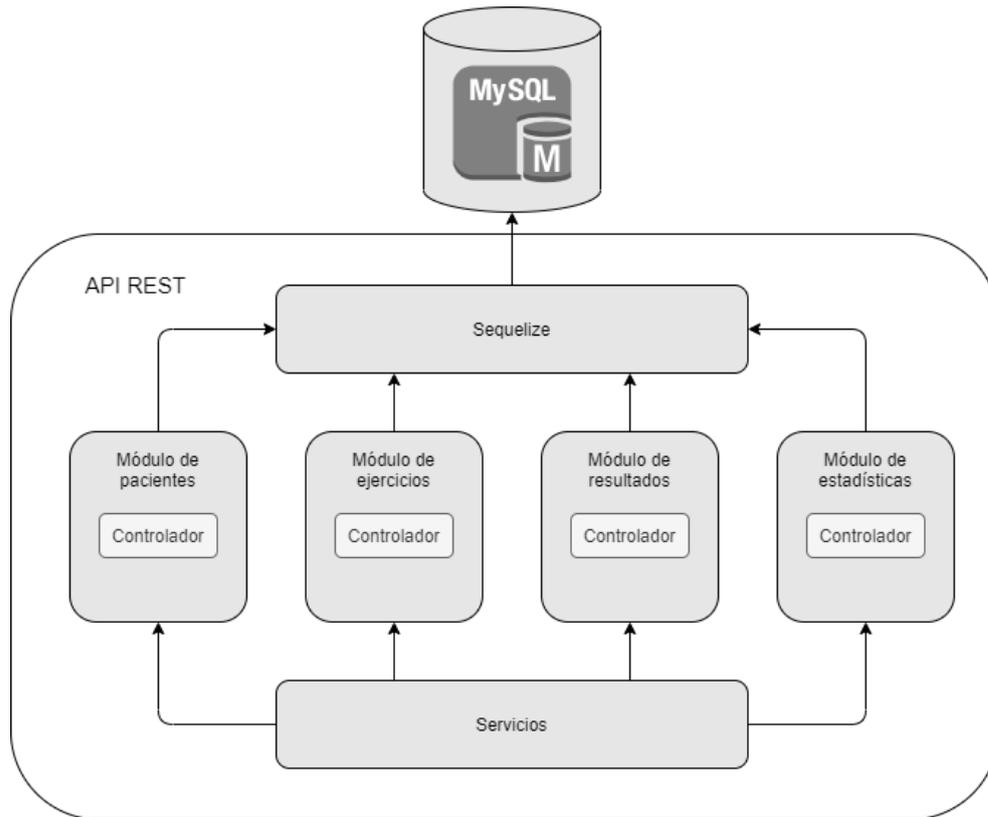


Figura 5.24: Visión general del *backend*

Cada uno de los módulos tiene su propio controlador implementados con el patrón *CRUD*<sup>8</sup>. Los servicios de la *API Rest* harán uso de los controladores de los módulos dependiendo la operación. El acceso a estos servicios se hace mediante peticiones *HTTP* a determinadas rutas de la *API Rest*.

### 5.7.1. Base de datos

La persistencia de datos del sistema se ha desarrollado con *MySQL* como sistema gestor de la base de datos. Se ha elegido este SGBD por su versatilidad, seguridad y rendimiento ofrecido, además, de ser de código abierto [32]. Una de las ventajas más relevantes para este proyecto es la posibilidad de guardar vectores espaciales con facilidad, funcionalidad muy importante para guardar de forma eficiente tanto los ejercicios creados como los resultados de la ejecución de los ejercicios.

La estructura de la base de datos diseñada se puede ver en la Figura 5.25. Tal y como se puede observar, se han definido un conjunto de cuatro tablas relacionadas entre sí que cubren los requisitos de persistencia de datos del proyecto. Los ejercicios se guardan en la tabla de ejercicios, en la cual se guardan los vectores de diferencias que representan los puntos de control, tal y como se explica en la Sección 5.5.2. La tabla de pacientes almacena la información relativa a los usuarios con este rol. La tabla de resultados se relaciona con la de pacientes, de tal forma que cada entrada en la tabla de resultados está relacionada con un paciente. A su vez, cada resultado también está relacionado con el ejercicio que se ha realizado para guardar ese resultado. De las ejecuciones de los ejercicios se guardarán una serie de estadísticas, que serán utilizadas para ver la evolución de los pacientes en la ejecución de los ejercicios.

Cada una de las tablas se mapea en un modelo utilizado por el *ORM* para ejecutar las operaciones sobre la base de datos. De tal forma, que en este modelo se definen todas las características de la tabla

<sup>8</sup>Se usa para referirse a las funciones básicas, Crear, Leer, Actualizar y Borrar, sobre bases de datos o la capa de persistencia en un *software*.

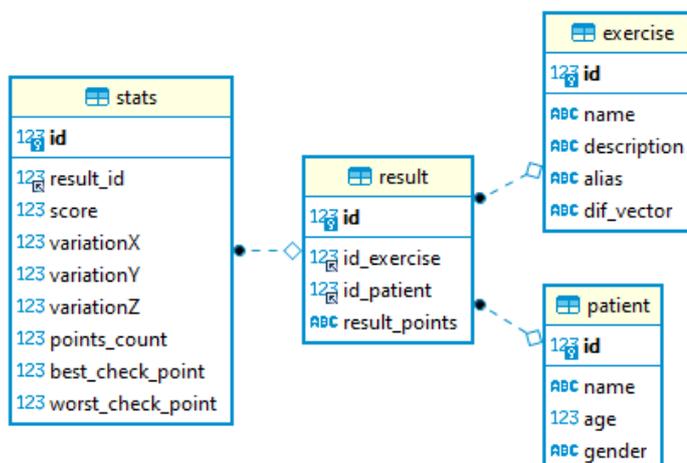


Figura 5.25: Esquema de la base de datos

y sus relaciones en forma de objeto, aplicando la técnica de mapeo-objeto relacional para convertir datos a objetos u entidades utilizables en la *API Rest*. Estos objetos serán utilizados para implementar las operaciones *CRUD* en los controladores. En el listado de código 5.5 se puede ver uno de los modelos creados.

Listado 5.5: Ejemplo del modelo *Patient*

```

1  const Sequelize = require('sequelize');
2  module.exports = function(sequelize, DataTypes) {
3    return sequelize.define('patient', {
4      id: {
5        autoIncrement: true,
6        type: DataTypes.BIGINT,
7        allowNull: false,
8        primaryKey: true
9      },
10     name: {
11       type: DataTypes.STRING(100),
12       allowNull: false
13     },
14     age: {
15       type: DataTypes.BIGINT,
16       allowNull: true
17     },
18     gender: {
19       type: DataTypes.STRING(100),
20       allowNull: true,
21       comment: "male/female"
22     }
23   }, {
24     sequelize,
25     tableName: 'patient',
26     timestamps: true,
27     indexes: [
28       {
29         name: "PRIMARY",
30         unique: true,
31         using: "BTREE",
32         fields: [
33           { name: "id" },
34         ]
35       },
  
```

```

36     ]
37   });
38 };

```

### 5.7.2. API Rest

Para acceder a la base de datos desde la aplicación de rehabilitación se ha implementado una capa de abstracción que permite realizar las operaciones sobre la base de datos con operaciones *HTTP*. De esta forma, se consigue dividir el sistema en una arquitectura cliente-servidor, en la cual las aplicaciones de rehabilitación son los clientes y el *backend*, formado por la *API Rest* y la base de datos, actúa como servidor. La *API Rest* está formada por una serie de servicios accesibles mediante distintos *endpoints*.

El cliente envía una petición *HTTP* a un *endpoint* para realizar una operación, el servicio ejecuta el controlador y éste ejecuta la operación sobre la base de datos utilizando el *ORM*, como se puede ver en la Figura 5.26. Las *endpoints* están divididos en tres grupos principales, según la entidad a la que se necesita acceso:

- Entidad paciente: para ejecutar operaciones *CRUD* relacionadas con pacientes se utiliza el *endpoint* */patient/*.
- Entidad ejercicio: las operaciones relativas a ejercicios siguen el patrón *CRUD* sobre el *endpoint* */exercise/*.
- Entidad resultados: los resultados siguen una estructura similar a las dos entidades anteriores, pero con un *endpoint* especial para obtener las estadísticas de un resultado, con la ruta */result/stats/*. El resto de operaciones *CRUD* se realizan sobre */result/*.

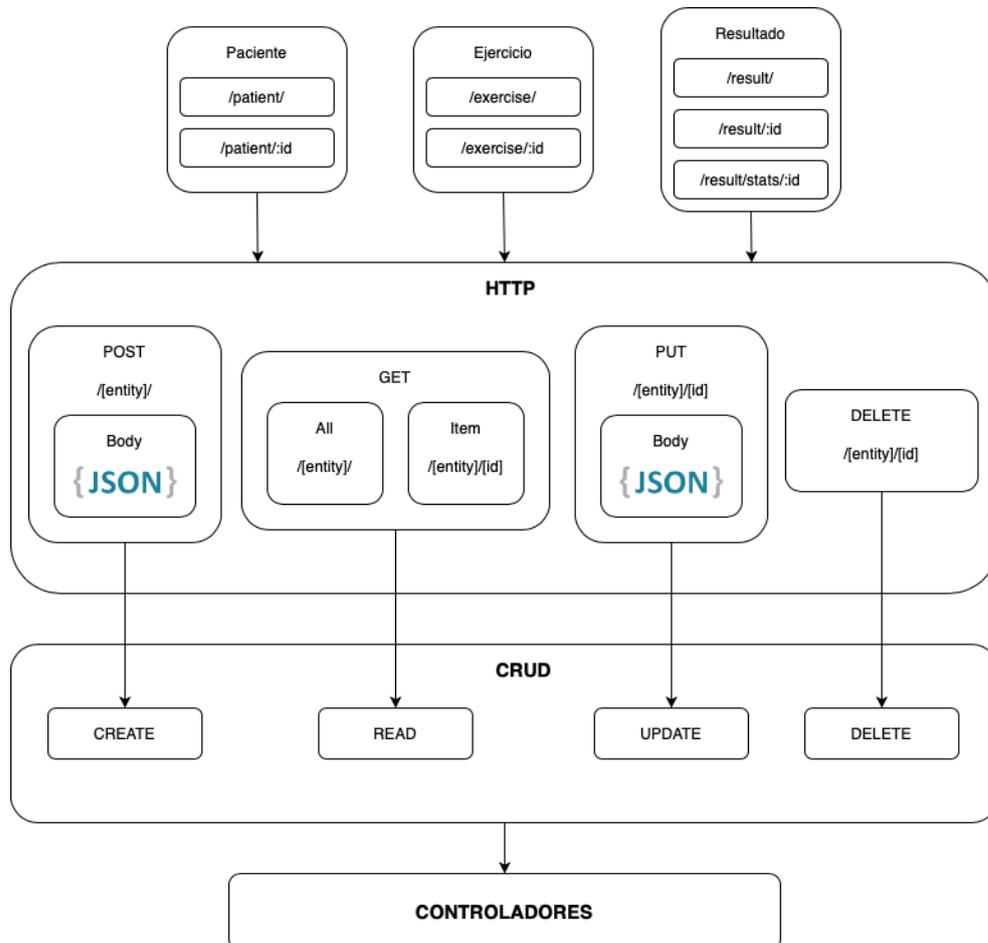


Figura 5.26: Estructura de las peticiones

# RESULTADOS

---

En este capítulo se van a describir los resultados obtenidos una vez finalizado el desarrollo del proyecto. En primer lugar, se va a exponer el sistema resultante y su funcionamiento. En segundo lugar, se van a mostrar los resultados de la experimentación de varias personas con el sistema.

La experimentación tiene como objetivo probar el sistema en un contexto realista, el cual estará compuesto por una persona con el rol de terapeuta que creará los ejercicios terapéuticos y varias personas que actuarán como pacientes invidentes. Éstos no tendrán la capacidad de ver durante la ejecución del ejercicio ni conocerán de antemano como es la trayectoria del movimiento. Con ello, se pretende demostrar que un paciente que nunca ha realizado un ejercicio particular, y tampoco ha recibido algún tipo de indicación por parte de un terapeuta, es capaz de recrearlo sin ayuda. De las sucesivas ejecuciones de los pacientes se obtendrán un número suficiente de datos para obtener conclusiones sobre el comportamiento del sistema. Además, con el objetivo de comprobar de forma objetiva si el uso del dispositivo de guiado en el proceso de rehabilitación mejora la precisión del recorrido de las articulaciones, se realizarán varias ejecuciones sin guiado para ser posteriormente comparadas con ejecuciones guiadas.

## 6.1. SISTEMA RESULTANTE

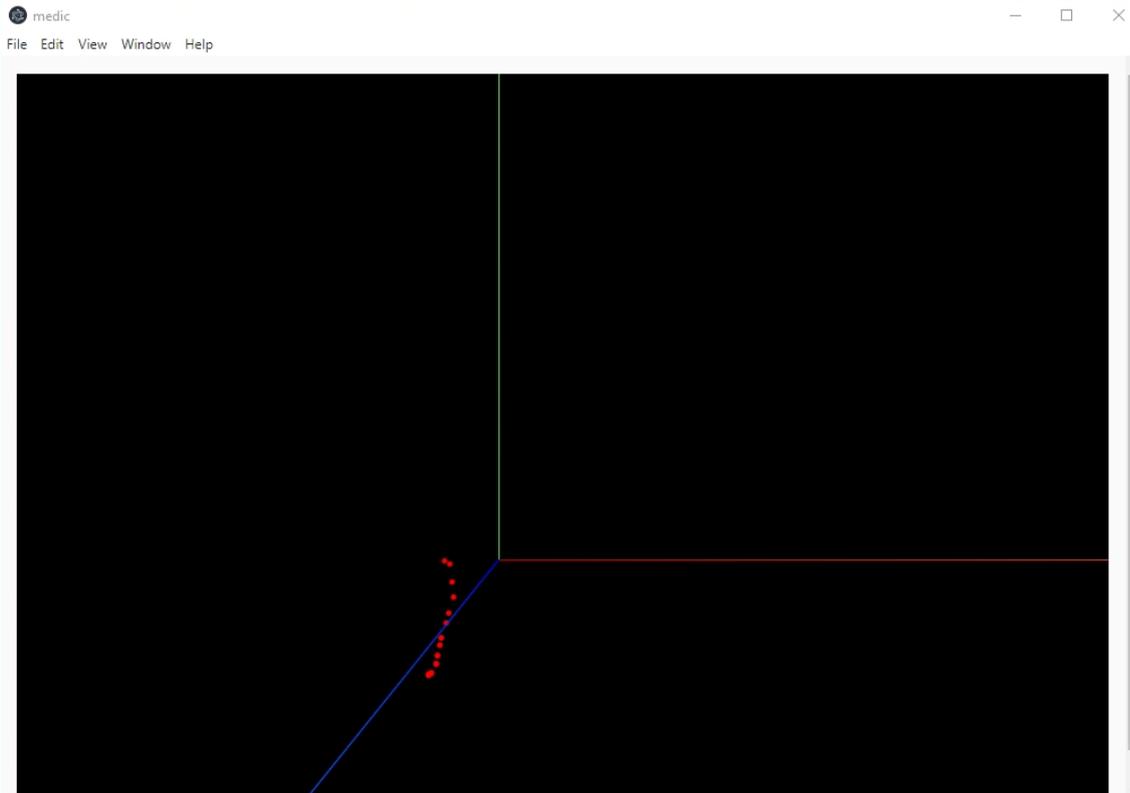
El sistema está compuesto por dos aplicaciones principales de rehabilitación y otros tres subsistemas necesarios para el funcionamiento de éstas, como se ha explicado en el Capítulo 5. En las siguientes sub-secciones se van a exponer los resultados del desarrollo de estas aplicaciones.

### 6.1.1. Aplicación del terapeuta

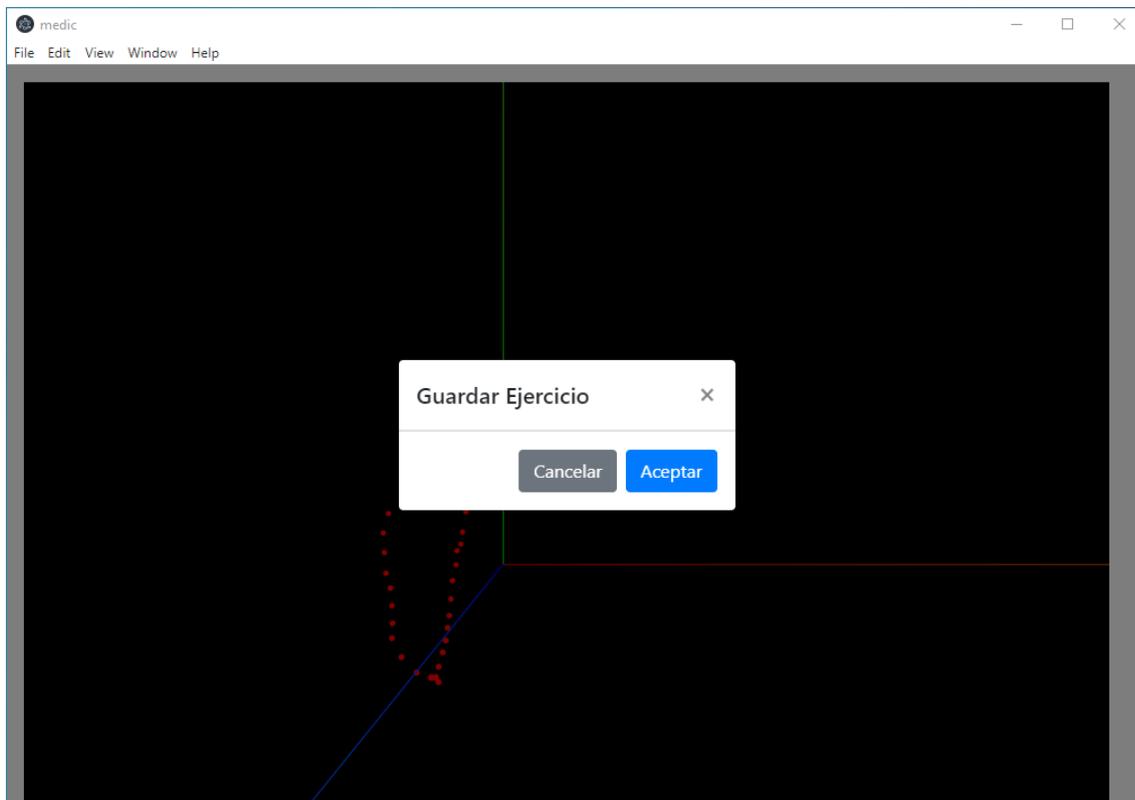
La aplicación del terapeuta se ha desarrollado siguiendo la arquitectura expuesta en la Sección 5.5. A continuación, se muestra el resultado del desarrollo de esta aplicación de rehabilitación. Esta aplicación tiene dos funcionalidades principales, la primera de ellas para crear ejercicios y la segunda de ellas para realizar un seguimiento de las distintas ejecuciones de los pacientes.

La creación de un ejercicio comienza con el gesto de mover la cabeza, una vez recibido este gesto en el sistema de seguimiento de articulaciones. La creación se considera acabada cuando la posición de la articulación es la misma que al inicio. En la Figura 6.1(a) se puede observar cómo se representa el ejercicio mientras se está creando. Esta pantalla tiene como objetivo que el terapeuta pueda ver cómo se está creando el ejercicio. Además, la creación está retroalimentada con audio, al comenzar, al crear un punto de control y al finalizar. Una vez finalizada la creación, se lanza un modal como se puede ver en la Figura 6.1(b) y el terapeuta decide si hacer persistente el ejercicio creado.

En la pantalla de seguimiento del paciente se puede interactuar de tal forma que se pueda observar la trayectoria del movimiento de la ejecución del paciente. Pudiendo observarla desde todas las perspectivas, como se puede observar Figura 6.2.



(a) Creación del ejercicio



(b) Guardado del ejercicio

**Figura 6.1:** Proceso de creación del ejercicio por el terapeuta

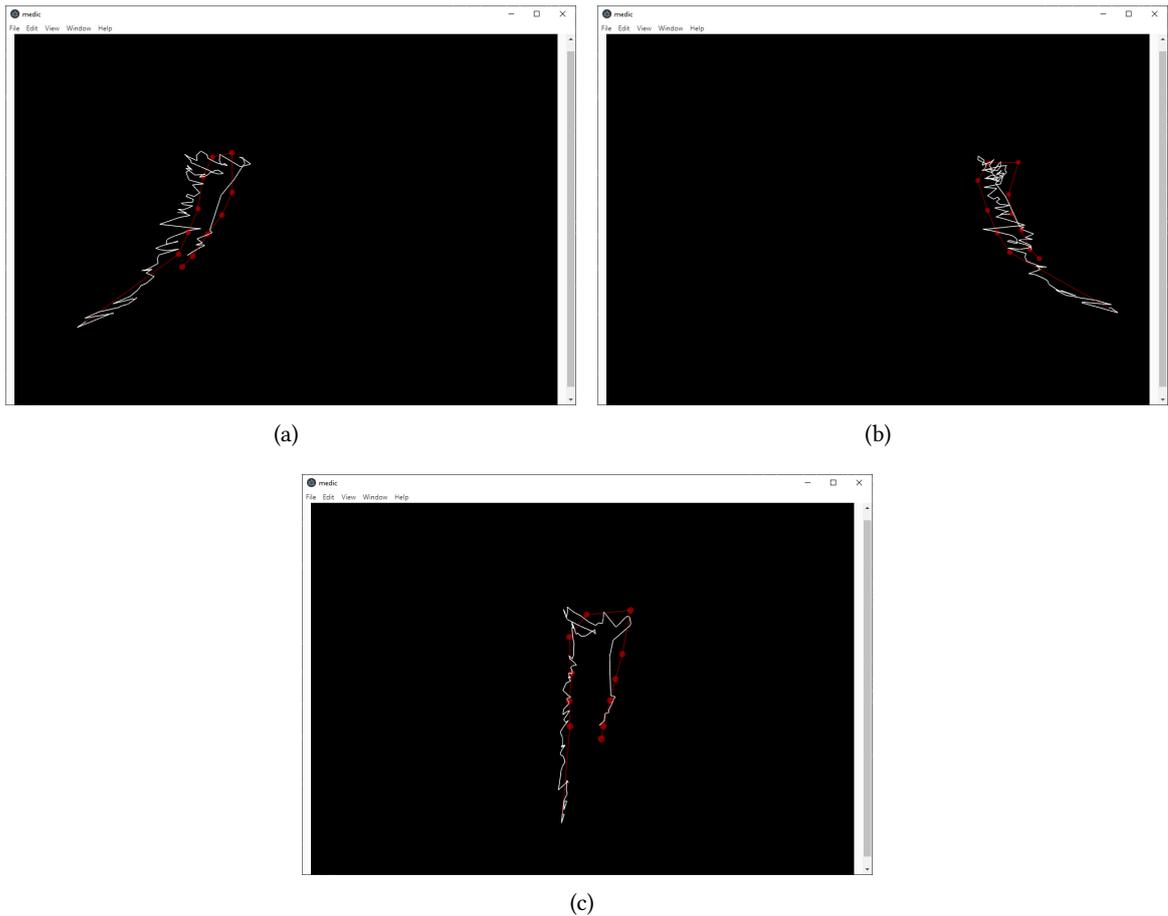


Figura 6.2: Ejecución ejercicio desde varias perspectivas

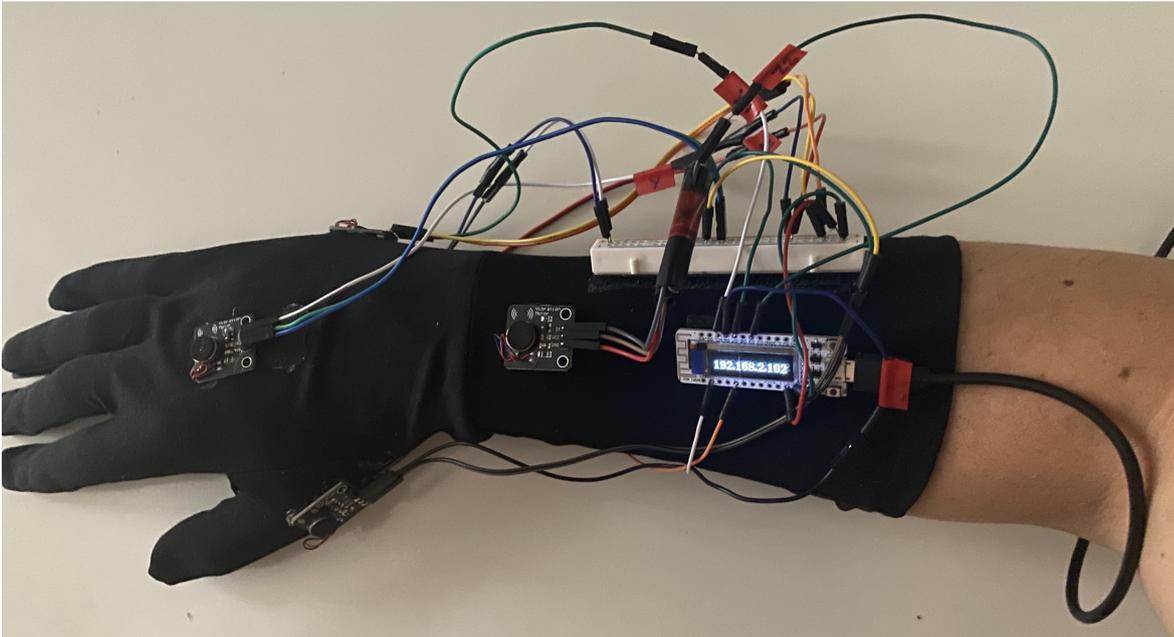
### 6.1.2. Aplicación del paciente

La aplicación del paciente se ha diseñado con interfaz gráfica para apreciar cómo funciona el sistema, ya que el usuario de esta aplicación no hará uso de esta interfaz al ser invidente. En la Sección 5.6 se han descrito las funcionalidades de esta aplicación de forma técnica. A continuación, se mostrará el resultado obtenido.



Figura 6.3: Configuración de red del sistema de guiado

En primer lugar, se debe de configurar el dispositivo de guiado, ya que como se ha descrito anteriormente la aplicación se conectará vía *WiFi* dentro de una red local. La configuración se puede realizar desde un *smartphone*, siguiendo los siguientes pasos: i) encender el dispositivo, el cual creará una red para su configuración (Figura 6.3); ii) con el *smartphone* conectaremos a esta red; iii) una vez conectados se configurará la red a la cual el dispositivo de guiado se conectará para interactuar con la aplicación. Cabe destacar, que esta configuración solo hay que realizarla la primera vez que el dispositivo se va a conectar a la red, ya que almacenará en memoria no volátil la información para conectarse posteriormente. Una vez configurado correctamente el dispositivo mostrará una dirección IP en su pantalla, como se muestra en la Figura 6.4, esta dirección será utilizada para que la aplicación pueda conectarse al dispositivo de forma inalámbrica.



**Figura 6.4:** Dispositivo de guiado conectado

Una vez configurado y colocado el dispositivo de guiado, se podrá comenzar a realizar el ejercicio. Al ejecutar la aplicación de rehabilitación del paciente se emitirán en forma de audio los ejercicios a realizar y el paciente mediante su voz elegirá el ejercicio a realizar. Una vez elegido, la grabación comenzará al realizar un gesto con la cabeza, similar al que realiza el terapeuta. En la Figura 6.5 se observa cómo se representa el ejercicio original con sus respectivos puntos de control unidos formando una trayectoria en el espacio. Además, se podrá ver en otro color los puntos que representan la trayectoria que sigue la ejecución del usuario. A la derecha de la ventana se puede ver una representación del dispositivo de guiado con los distintos actuadores, los cuales cambian de color según son utilizados en el dispositivo real. Se puede observar como el actuador encendido es el que indica que hay moverse hacia el eje *Y* positivo, porque como se puede ver en la trayectoria a realizar el punto de control a conseguir está situado en este eje y sentido.

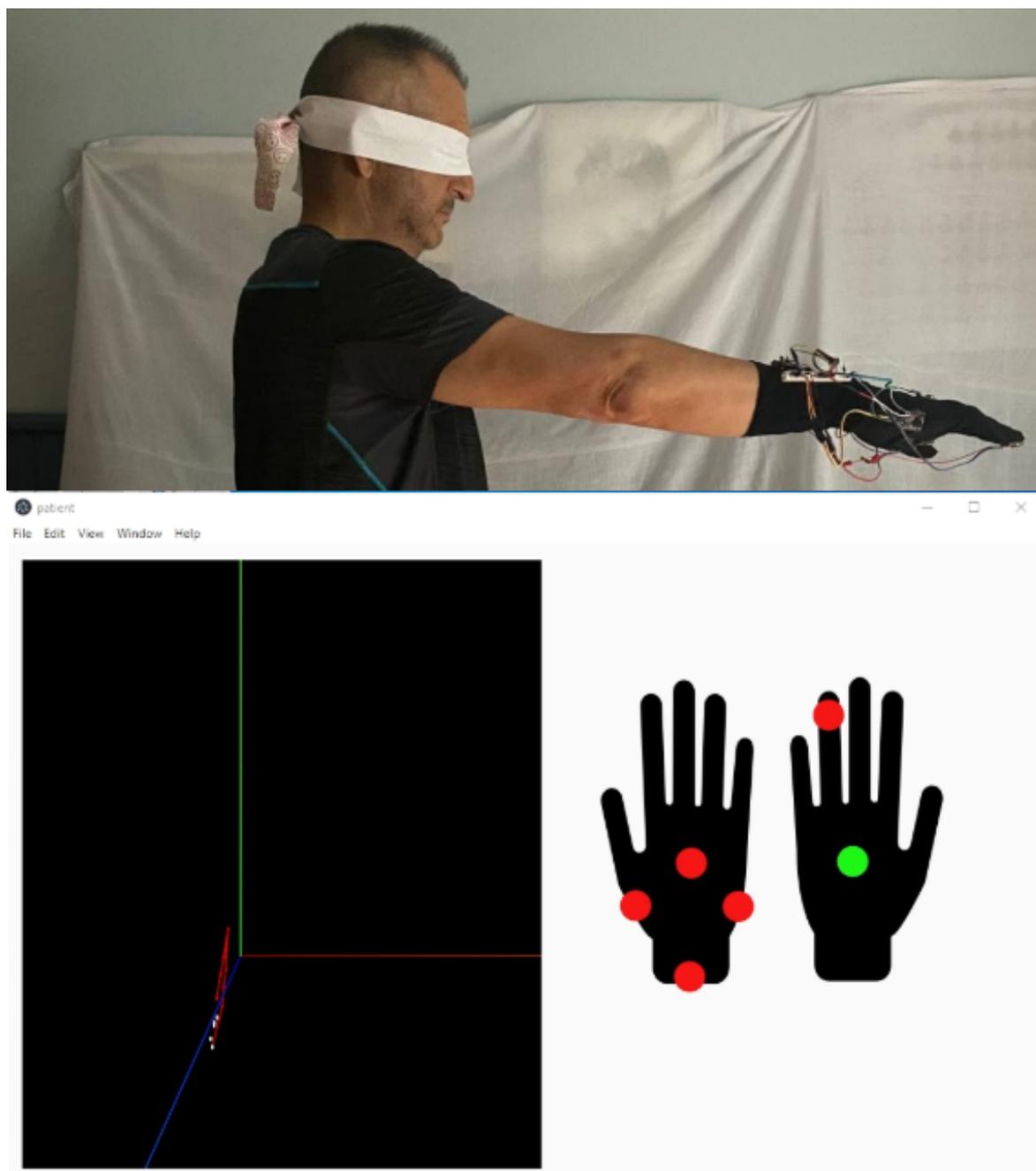


Figura 6.5: Interfaz del paciente

## 6.2. EXPERIMENTACIÓN

La experimentación ha consistido en crear dos ejercicios y la ejecución de estos dos ejercicios por parte de cinco personas. Debido a la pandemia, las pruebas se han restringido al entorno familiar del autor. Cada una de las personas que han formado parte de la experimentación ha realizado los ejercicios sin posibilidad de ver, para simular la discapacidad visual. Por lo tanto, se puede afirmar que los datos obtenidos son fiables y equivalentes a pacientes con discapacidad visual real.

Se han guardado ciertos datos considerados como relevantes de los ejercicios para realizar un estudio y sacar conclusiones del sistema desarrollado. Los datos calculados son los siguientes:

- **Mejor punto de control (Mejor PC):** se considera como mejor punto el conseguido de forma lo más cercana posible.

Paciente	Género	Edad	Articulación
1	Masculino	18	Brazo izquierdo
2	Masculino	53	Brazo derecho
3	Masculino	32	Brazo derecho
4	Femenino	17	Brazo derecho
5	Femenino	48	Brazo izquierdo

- **Peor punto de control (Peor PC):** de forma contraria, se considera como peor punto de control al conseguido de forma más lejana.
- **Puntuación:** para la puntuación se ha creado la Función 6.1 que relaciona la distancia media obtenida ( $d$ ) y los puntos ( $p$ ) que ha necesitado el paciente para completar el ejercicio. De tal forma, que a menor distancia media más puntuación y a menor número de puntos necesarios más puntuación. Con la división de ambos valores se obtienen resultados menores a cero, que dificultan la comprensión de la puntuación. Por ello, se ha decidido multiplicarlo por 1000, obteniendo así resultados más comprensibles.

$$F(d, p) = \left( \frac{1}{d + p} \right) * 1000 \quad (6.1)$$

- **Puntos:** puntos necesarios para completar el ejercicio. A más puntos utilizados mayor tiempo en completar el ejercicios y viceversa.
- **Variaciones de coordenadas:** se define como las variaciones medias de cada coordenada obtenidas al completar los distintos puntos de control del ejercicio, siguiendo la Función 6.2, la cual está formada por el valor original del punto de control ( $n$ ) y el valor de la coordenada con el cual se consigue el punto de control ( $n'$ ). El resultado se obtiene en porcentaje, por ello se multiplica por 100.

$$F(n, n') = \frac{n' - n}{n} * 100 \quad (6.2)$$

- **Distancia:** distancia espacial media calculada al conseguir los distintos puntos de control, utilizando la Función 6.3. Con esta función se calculará la distancia espacial entre un punto inicial ( $x_1, y_1, z_1$ ) y un punto final ( $x_2, y_2, z_2$ ).

$$distancia = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \quad (6.3)$$

### 6.2.1. Ejercicio 1

En las Tablas 6.1 y 6.2 se pueden observar los resultados obtenidos de la experimentación con el Ejercicio 1. La trayectoria de este ejercicio se puede ver en la Figura 6.8. Se puede apreciar como la trayectoria del movimiento que debe realizar el paciente varía entre ambas ejecuciones. Esto es debido a que los puntos de control se adaptan al espacio dimensional del paciente. De esta forma, se consigue la independencia espacial comentada anteriormente sin modificar significativamente el ejercicio terapéutico.

	Mejor PC	Peor PC	Puntuación	Puntos
Persona 1	2	6	3,91	164
Persona 2	2	11	3,49	170
Persona 3	2	6	4,43	130
Persona 4	8	4	4,25	143
Persona 5	9	4	4,53	136

**Tabla 6.1:** Resultados ejercicio 1

En la tabla 6.1 se puede ver las estadísticas obtenidas de las distintas ejecuciones del ejercicio 1 por parte de los cinco pacientes. En cuanto al punto de control conseguido con más cercanía se

puede observar que tanto el cuatro como el seis se repiten dos veces entre los cinco pacientes. La puntuación media obtenida ha sido de 4.12, siendo la Persona 2 la que menos puntuación ha obtenido con 3.49, y la Persona 5 la que más con 4.53. El promedio de puntos utilizados por los pacientes ha sido de 148.6, siendo la Persona 3 la que menos tiene y la que más es la Persona 1, coincidiendo con la peor puntuación obtenida. La Persona 3 ha obtenido una puntuación de 4.43, cercana a la máxima de 4.53. Se podría deducir que ha realizado el ejercicio más rápido que la Persona 5 pero ha completado los puntos de control más lejos. Teniendo en cuenta que el paciente no conoce la trayectoria del ejercicio a realizar los pacientes han obtenido una puntuación considerablemente buena, ya que se han completado todos los puntos de control con una media de puntos necesarios de 148.6. A menor puntos necesarios para completar un ejercicio más precisión en la trayectoria del movimiento, ya que se han conseguido los sucesivos puntos de control sin necesidad de demasiados cambios de dirección. Esto se se puede apreciar gráficamente en las dos ejecuciones con guiado de la Figura 6.8.

	Variación X	Variación Y	Variación Z	Distancia
Persona 1	24,57	46,82	8,20	91,63
Persona 2	25,93	47,58	11,86	116,70
Persona 3	26,47	47,71	9,06	95,52
Persona 4	28,13	50,54	8,41	92,26
Persona 5	29,52	50,69	6,63	84,84

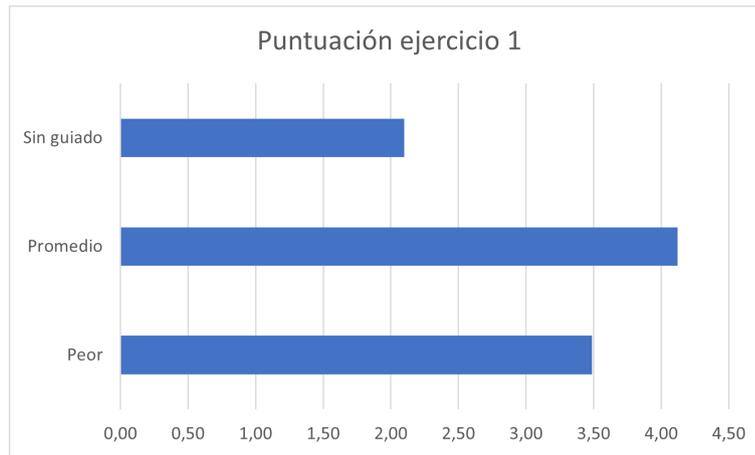
**Tabla 6.2:** Resultados ejercicio 1

En la Tabla 6.2 se pueden ver las distintas variaciones medias resultantes de la ejecución del ejercicio 1 y la distancia media obtenida. La coordenada que más ha variado es la Y, siguiendo por la X y la que menos la Z, resultando una variación media total de 28,14. En cuanto a la distancia media, se puede observar cómo la Persona 5 es la que menos distancia a los puntos de control ha obtenido. Este resultado en la distancia junto a la cantidad de puntos utilizados que está por debajo de la media, resulta en la mejor puntuación, como se ha comentado anteriormente. La variación media obtenida en las tres coordenadas se puede considerar un dato positivo, ya que en promedio un paciente se ha separado del punto de control a conseguir únicamente un 28,14 %.

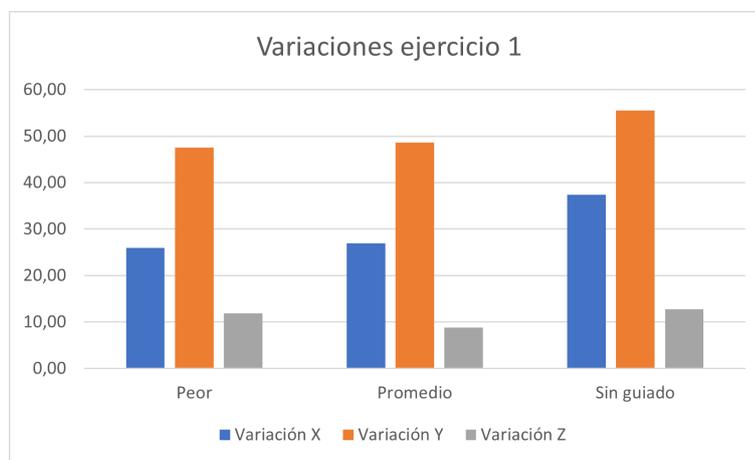
Ejecución	Puntuación	Variación total	Distancia	Puntos
Promedio con guiado	4,12	28,14	96,19	148,6
Peor con guiado	3,49	28,46	116,70	170
Sin guiado	2,1	37,4	135,3	340

**Tabla 6.3:** Comparación resultados ejercicio 1 sin guiado

En la tabla 6.3 se puede apreciar una comparación entre una ejecución sin guiado y las ejecuciones con guiado comentadas anteriormente. Como se puede ver en todos los parámetros medidos se obtiene un empeoramiento en la ejecución sin guiado respecto a las ejecuciones con guiado. La puntuación de la ejecución sin guiado se ha decrementado en un 49,1 % respecto al promedio con guiado y un 39,8 % respecto a la peor ejecución con guiado (ver Figura 6.6). La variación total de las tres coordenadas de la ejecución sin guiado se ha incrementado en un 25,2 % respecto al promedio con guiado y un 23,8 % respecto a la peor ejecución con guiado (ver Figura 6.7). Un incremento en la variación indica que los puntos de control se han conseguido de forma más imprecisa. Teniendo en cuenta los datos anteriores, se puede afirmar cómo ejecutar el ejercicio 1 con guiado supone una mejora sustancial con respecto al método tradicional sin guiado.

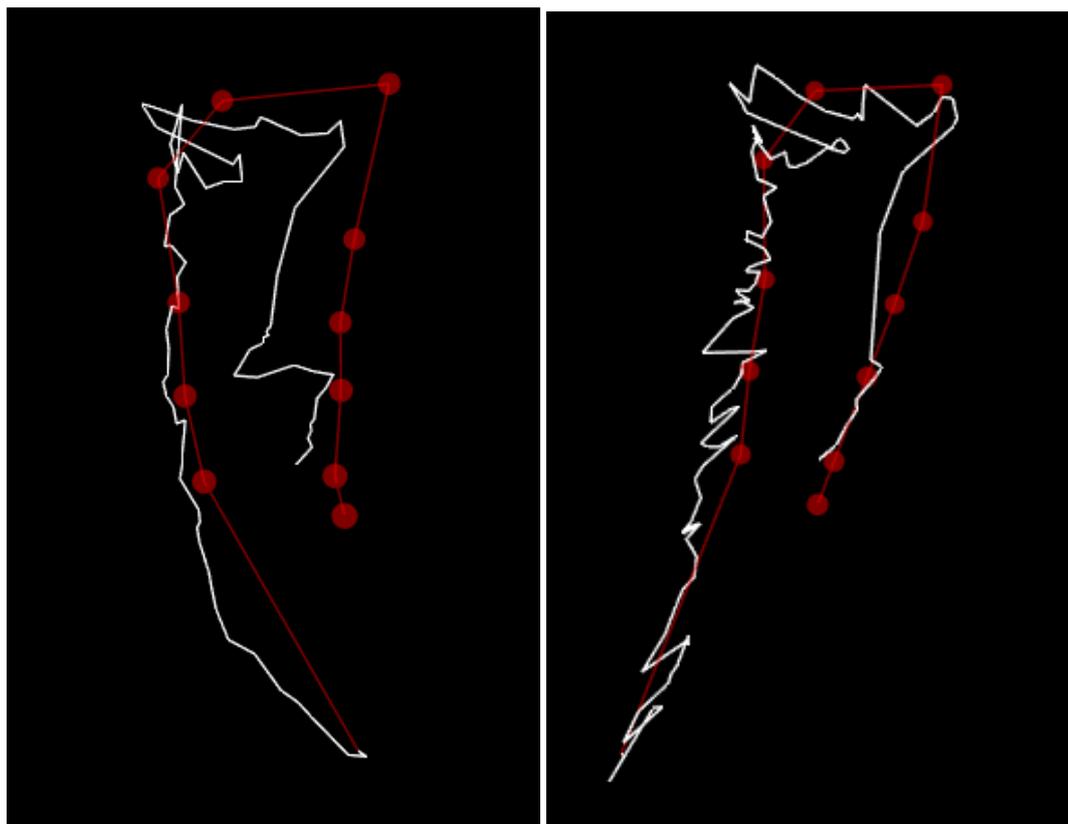


**Figura 6.6:** Puntuación ejercicio 1



**Figura 6.7:** Variaciones ejercicio 1

En la Figura 6.8 se pueden apreciar dos ejecuciones con guiado y un sin guiado. Se puede comprobar de manera gráfica lo que mostraban los datos anteriores, la ejecución sin guiado es sustancialmente peor que las ejecuciones con guiado en todos los parámetros medidos. Se observan un mayor número de cambios de dirección que dibujan una trayectoria que se asemeja menos al ejercicio original. Esto puede suponer que el ejercicio terapéutico no cumpla su función rehabilitadora, ya que se realiza de forma muy imprecisa.



(a) Ejecución con guiado

(b) Ejecución con guiado



(c) Ejecución sin guiado

**Figura 6.8:** Ejecuciones Ejercicio 1

### 6.2.2. Ejercicio 2

En las tablas 6.4 y 6.5 se pueden apreciar los resultados obtenidos de las distintas ejecuciones del Ejercicio 2 y la trayectoria de éste se puede observar en la Figura 6.11. Al igual que en el ejercicio 1, se puede apreciar como la trayectoria representa el mismo movimiento en ambas ejecuciones, pero los puntos de control se adaptan al espacio dimensional del paciente.

	Mejor PC	Peor PC	Puntuación	Puntos
Persona 1	7	2	6,46	106
Persona 2	3	4	4,00	136
Persona 3	2	6	3,88	154
Persona 4	8	6	5,05	79
Persona 5	3	1	5,54	104

**Tabla 6.4:** Resultados ejercicio 2

En la Tabla 6.4 se puede ver el mejor y peor punto de control de cada usuario. Entre ellos destaca el punto de control tres que se repite dos veces como mejor del conjunto y el seis como peor se repite dos veces también. La puntuación media obtenida de entre todos los pacientes ha sido 4,99, destacando la Persona 1 con una puntuación de 6,46, siendo la más alta, y la Persona 3 con la puntuación más baja, 3,88. La cantidad de puntos utilizados de media ha sido 115,8. La Persona 4 es el usuario que menos puntos ha empleado y la Persona 3 la que más, en consecuencia ha obtenido la peor puntuación para este ejercicio. La puntuación obtenida para este ejercicio indica que en promedio los pacientes han realizado el ejercicio con gran precisión, teniendo en cuenta los mismos supuestos que en el ejercicio 1. Además, los puntos necesarios para completar todos los puntos de control que representan el ejercicio han sido en promedio bajos, lo que indica que los pacientes no han tenido demasiadas dificultades para completar el ejercicio.

	Variación X	Variación Y	Variación Z	Distancia
Persona 1	7,66	47,96	3,99	48,91
Persona 2	24,02	52,33	10,14	113,70
Persona 3	28,55	56,69	9,16	103,44
Persona 4	17,69	62,03	9,14	119,14
Persona 5	10,93	56,76	4,69	76,35

**Tabla 6.5:** Resultados ejercicio 2

En la Tabla 6.5 se pueden ver las distintas variaciones medias resultantes de la ejecución del ejercicio 2 y la distancia media obtenida. La variación media total obtenida es de 26,78. La distancia media total es de 115,80, siendo la Persona 5 la que menos distancia ha necesitado para completar los distintos puntos de control y la Persona 2 la que más distancia. La variación en promedio de las tres coordenadas y la distancia espacial obtenidas indican también que la precisión a la hora de completar el ejercicio ha sido alta.

Ejecución	Puntuación	Variación total	Distancia	Puntos
Promedio con guiado	4,99	26,78	92,31	115,80
Peor con guiado	3,88	31,47	103,44	154
Sin guiado	2,42	34,24	104,42	340

**Tabla 6.6:** Comparación resultados ejercicio 2 sin guiado

En la tabla 6.6 se puede ver una comparación entre una ejecución sin guiado y las ejecuciones con guiado comentadas anteriormente. Como se puede apreciar en todos los parámetros medidos se obtiene un empeoramiento en la ejecución sin guiado respecto a las ejecuciones con guiado. La puntuación de la ejecución sin guiado se ha decrementado en un 51,5 % respecto al promedio con guiado y un 39,6 % respecto a la peor ejecución con guiado (ver Figura 6.9). La variación total de las tres coordenadas de la ejecución sin guiado se ha incrementado en un 27,8 % respecto al promedio con guiado y un 18,7 % respecto a la peor ejecución con guiado (ver Figura 6.10). Teniendo en cuenta los datos anteriores, se puede afirmar como ejecutar el ejercicio 2 con guiado supone una mejora sustancial a ejecutar sin guiado.

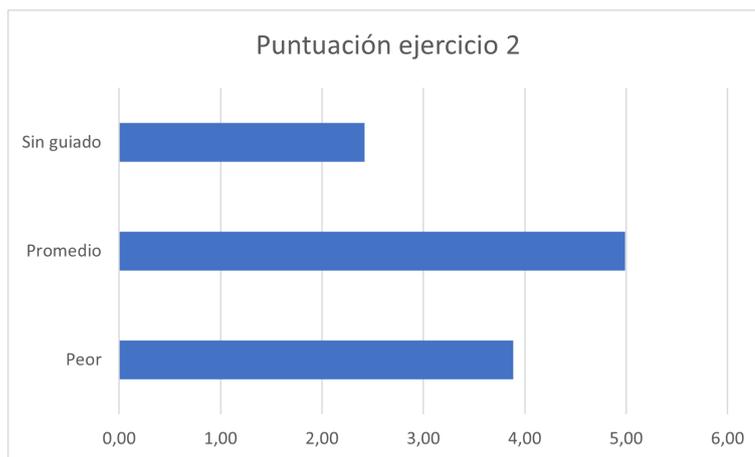


Figura 6.9: Puntuación ejercicio 2

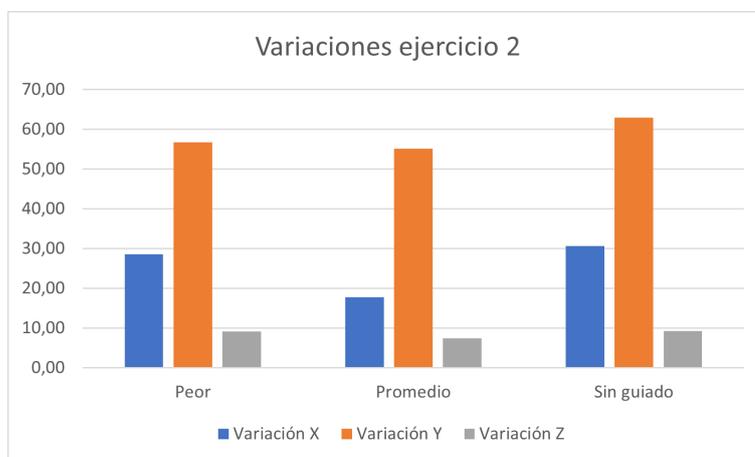
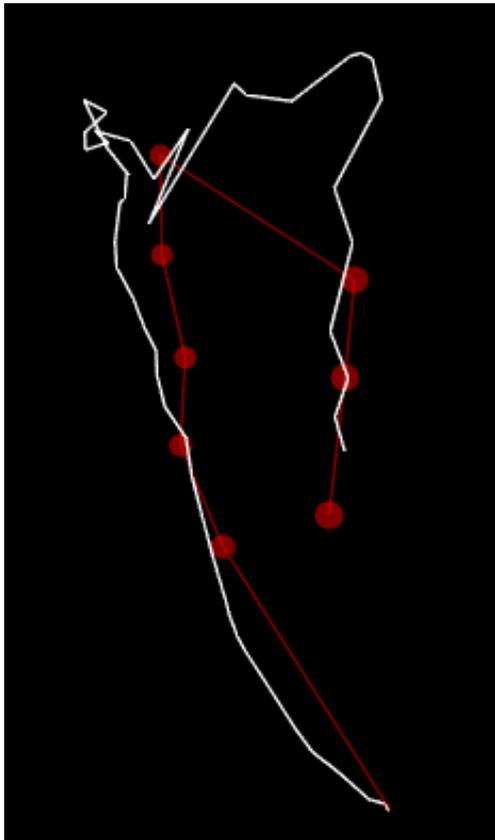
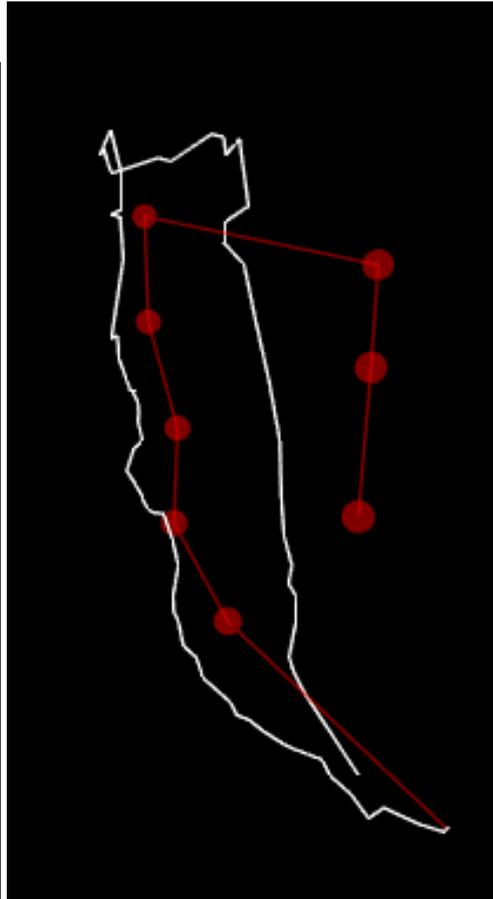


Figura 6.10: Variaciones ejercicio 2

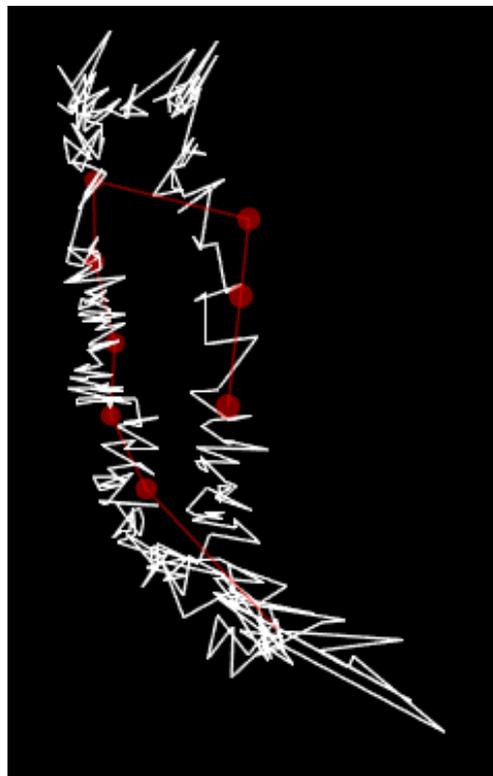
En la Figura 6.11 se pueden ver dos ejecuciones con guiado y un sin guiado. Al igual que en las ejecuciones del ejercicio 1, se puede apreciar de manera gráfica lo que mostraban los datos anteriores, la ejecución sin guiado es peor que las ejecuciones con guiado en todos los parámetros medidos. Se observan un mayor número de cambios de dirección que dibujan una trayectoria que se asemeja menos al ejercicio creado por el terapeuta. Esta imprecisión a la hora de realizar el ejercicio sin guiado puede suponer que éste no cumpla su función rehabilitadora.



(a) Ejecución con guiado



(b) Ejecución con guiado



(c) Ejecución sin guiado

**Figura 6.11:** Ejecuciones ejercicio 2

### 6.2.3. Valoración final

Una vez finalizada la experimentación se va a exponer de forma gráfica las estadísticas obtenidas de la ejecución de ambos ejercicios y compararlas con las ejecuciones sin guiado.

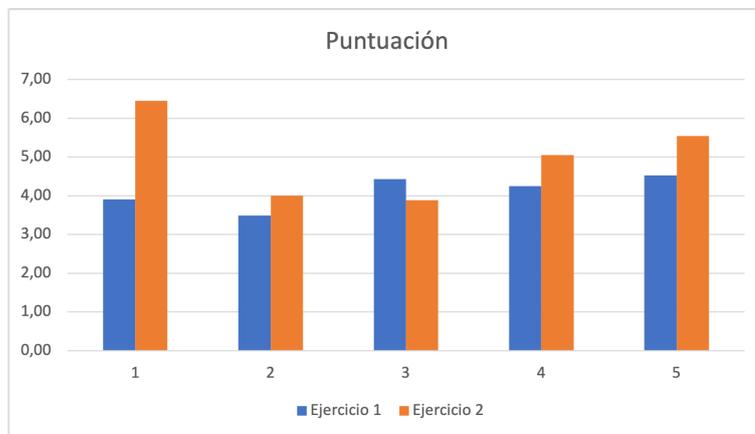


Figura 6.12: Puntuación

La comparación de puntuaciones entre ejercicios para todos los pacientes se puede ver en la Figura 6.13. Se aprecia como todos los pacientes han completado el ejercicio 2 con mejor puntuación que el ejercicio 1, a excepción de la Persona 3 que ha obtenido una puntuación un 12 % inferior. La mayor mejora entre ejercicios se puede apreciar en la Persona 1 que ha mejorado su puntuación un 65 % en el ejercicio 2 respecto al ejercicio 1. Una vez realizada la comparación entre ejercicios para cada uno de los pacientes se puede apreciar que en promedio se ha realizado mejor el ejercicio 2. Teniendo en cuenta que la puntuación es una relación entre la distancia espacial promedio y los puntos necesarios, se puede apreciar una alta precisión sin necesidad de estar demasiado tiempo para completar los puntos de control en el ejercicio 2.

Ejecución	Puntuación	Distancia	Puntos	Variación total
Sin guiado	2,42	104,42	308,00	34,74
Con guiado	4,55	94,25	132,20	27,46

Tabla 6.7: Resultados globales

En la tabla 6.7 se pueden ver las estadísticas globales de ejecuciones sin guiado y con guiado. Se puede observar como en promedio las ejecuciones con guiado superan a las ejecuciones sin guiado en todos los parámetros estudiados. La puntuación en promedio de las ejecuciones con guiado se ha incrementado un 88 % respecto a las ejecuciones sin guiado (ver Figura 6.13). La variación media total de las ejecuciones con guiado ha disminuido un 19,8 % respecto a las ejecuciones sin guiado (ver Figura 6.14). Un incremento en la puntuación junto a un decremento en la variación media total indica una mejora en la precisión y con ello en la trayectoria del movimiento.



**Figura 6.13:** Puntuación global



**Figura 6.14:** Variación media total

Una vez estudiados los datos obtenidos de la experimentación con el sistema resultante, se ha comprobado como las ejecuciones de los ejercicios terapéuticos han mejorado de forma muy notable al realizarlas con el sistema de guiado respecto a realizarlas sin ningún tipo de guiado en tiempo real. Teniendo en cuenta estos datos, se puede afirmar que se ha desarrollado una solución tecnológica en el campo de la rehabilitación articular que podría mejorar la eficiencia de los procesos de rehabilitación para pacientes invidentes. Esta mejora se debe a que utilizando este sistema se puede realizar una rehabilitación articular a distancia y asíncrona, es decir el terapeuta no tiene la necesidad de supervisar en tiempo real la ejecución de los terapéuticos pautados. El sistema tiene la capacidad de guiar al paciente sin conocer de antemano el ejercicio a realizar, por lo tanto, el terapeuta puede tener un número de pacientes mayor que en un proceso de rehabilitación convencional, ya que en una rehabilitación tradicional el terapeuta indica al paciente la trayectoria del movimiento mientras los está realizando.

# CONCLUSIONES

---

En este capítulo se realizará un juicio crítico y discusión sobre los resultados obtenidos. En primer lugar, se hará un breve resumen del contexto y la problemática a la cuál se pretende dar solución con este proyecto. A continuación, se revisarán los objetivos iniciales del trabajo para determinar si se han alcanzado una vez terminado el desarrollo del proyecto. Por último, se justificarán las competencias adquiridas de la tecnología específica de *ingeniería de computadores*, concluyendo con las posibles líneas de trabajo futuro relativas a este proyecto.

La rehabilitación física tradicional supone reunir a terapeuta y paciente en el mismo lugar físico, para realizar una serie de ejercicios terapéuticos pautados y supervisados por el terapeuta, puesto que el terapeuta tiene un tiempo limitado para estar con el paciente en el proceso de rehabilitación. Debido a esto, la rehabilitación podría ser de peor calidad para el paciente y el número de pacientes también está muy limitado. Este proceso se dificulta aún más cuando los pacientes son invidentes, ya que sus capacidades de orientación y movilidad están más limitadas, entonces el guiado en la ejecución de los ejercicios es más complejo. En un proceso de rehabilitación convencional para personas con este tipo de discapacidad, antes de realizar el ejercicio se les explica la trayectoria y ésta la deben recordar al realizarlo.

Este proyecto tiene como objetivo principal proponer una solución eficaz a estos problemas. Para ello se ha desarrollado un sistema formado por dos aplicaciones principales, una aplicación para el terapeuta y otra para el paciente. La aplicación del terapeuta tiene como finalidad principal la creación de los ejercicios terapéuticos y poder realizar un seguimiento de los pacientes. La aplicación del paciente le permitirá ejecutar los ejercicios terapéuticos para rehabilitarse a distancia. Los pacientes al ser invidentes el proceso de rehabilitación se realiza con un dispositivo de guiado diseñado específicamente para este proyecto, que guiará al paciente de forma háptica. Además, el paciente podrá hacer uso de comandos de voz para utilizar las funcionalidades de la aplicación. Los distintos subsistemas del proyecto se integran gracias al *backend*, con el cual se realizan todas las operaciones necesarias relacionados con datos. Este subsistema es de vital importancia para lograr la rehabilitación a distancia, ya que la aplicación del terapeuta y la del paciente se ejecutarán en ubicaciones distintas. Por lo tanto, se requiere una integración en la nube para que ambas aplicaciones realicen sus funciones de forma independiente.

Tal y como se muestra en el Capítulo 6 los pacientes son capaces de realizar los ejercicios terapéuticos sin información visual, únicamente con el dispositivo de guiado y con retroalimentación auditiva. Teniendo en cuenta que no conocen la trayectoria de antemano, se puede apreciar como los pacientes ejecutan los distintos ejercicios de una forma bastante precisa, ya que la variación media obtenida entre la trayectoria de los movimientos originales y sus ejecuciones se sitúa entorno a un 26 %. Las trayectorias se puede ver de forma gráfica en la Figura 6.8 para un ejercicio y en la Figura 6.11 para otro de los ejercicios.

Durante el desarrollo de este proyecto se han aplicado gran parte de los conocimientos adquiridos a lo largo del Grado en Ingeniería Informática. Éstos han sido utilizados tanto para el diseño y desarrollo

del sistema como para la gestión del proyecto. Las habilidades técnicas que más se han aplicado han sido las adquiridas en la intensificación cursada, como se comentará posteriormente. En cuanto a habilidades no técnicas empleadas destacan las relacionadas con la ejecución de la metodología de desarrollo iterativo-incremental para la gestión del proceso de desarrollo. Los conocimientos adquiridos más destacables han sido *tracking* corporal, creación de un prototipo *hardware* y manejo de datos tridimensionales.

## 7.1. OBJETIVOS ALCANZADOS

En esta sección se va a revisar si se han alcanzado los objetivos detallados en el Capítulo 2 tras el desarrollo del proyecto.

### **Desarrollar un sistema de definición de ejercicios por parte del rehabilitador**

Uno de los objetivos principales del sistema era que el terapeuta pudiera definir los ejercicios terapéuticos necesarios para que el paciente pudiera realizar el proceso de rehabilitación a distancia. Para ello, era necesario guardar la trayectoria del movimiento de tal forma que posteriormente se pudiera reproducir en el entorno del paciente. Este requisito fundamental se ha resuelto con un algoritmo que calcula las diferencias entre los puntos que representan el ejercicio.

### **Realizar un análisis del movimiento en tiempo real**

Para captar los movimientos corporales en la creación y ejecución de ejercicios terapéuticos es necesario realizar un *tracking* corporal. Para realizar este proceso es necesario utilizar un dispositivo que capte estos datos en tiempo real. El dispositivo elegido ha sido el *Azure Kinect DK*. Una vez obtenidos las coordenadas de los distintos puntos que representan el movimiento el sistema los procesa y realiza las operaciones necesarias según el rol del usuario.

### **Desarrollar un dispositivo de guiado que permita la correcta realización de ejercicios terapéuticos**

Los ejercicios definidos deben ser realizados por una persona con discapacidad visual. Por ello, se debe guiar al paciente sin implicar el sentido de la vista del mismo. Teniendo en cuenta esto, se ha diseñado un dispositivo en forma de guante con vibradores incorporados que permite guiar al paciente durante la ejecución de los ejercicios de forma háptica. La interacción consiste en emitir vibraciones perceptibles por el paciente que representen los distintos ejes de coordenadas en un espacio tridimensional. De esta forma, el paciente es guiado hacia los distintos puntos de control que representan el ejercicio original para conseguir una trayectoria lo más parecida posible.

### **Aplicación de técnicas de gamificación para mejorar la motivación del paciente**

La motivación del paciente al realizar el tratamiento de rehabilitación es primordial para lograr recuperarse. Por ello, es importante que el paciente sea consciente de cómo está realizando el ejercicio. La calidad de la ejecución depende de la proximidad al recorrido definido por el terapeuta sea capaz de conseguir el paciente al completar los distintos puntos de control. Un punto de control se considera completado cuando la variación entre el punto del paciente y el punto de control está por debajo de un umbral. Este umbral es variable, ya que si un paciente le cuesta demasiado completar un determinado punto de control porque no alcanza este umbral podría dejar de realizar el ejercicio. Por esta razón, el umbral va aumentando conforme el paciente intenta completar el punto de control objetivo y una vez completado el umbral se reinicia. Esta decisión de diseño es de gran importancia para que el paciente no pierda la motivación en el proceso de rehabilitación. Además, cada vez que se consigue un punto de control el paciente recibe retroalimentación acústica.

### **Desarrollar un sistema de seguimiento para ver los resultados de la ejecución de los distintos ejercicios**

El terapeuta debe ser consciente de cómo ha realizado el paciente las distintas ejecuciones de los ejercicios. Para ello, el terapeuta tendrá la posibilidad de visionar las trayectorias de estas ejecuciones comparadas con el ejercicio original. Además, de un conjunto de estadísticas que le permitirán realizar un seguimiento y tomar las decisiones pertinentes durante el proceso de rehabilitación.

## **7.2. JUSTIFICACIÓN DE COMPETENCIAS ADQUIRIDAS**

Durante el desarrollo del presente TFG se han puesto en práctica gran parte de los conocimientos adquiridos durante el grado, además de adquirir nuevos que han contribuido a la adquisición de competencias específicas de la intensificación de *Ingeniería de Computadores*.

**IC1:** [*Capacidad de diseñar y construir sistemas digitales, incluyendo computadores, sistemas basados en microprocesador y sistemas de comunicaciones.*]. Para el dispositivo de guiado se ha tenido que diseñar y construir un prototipo que fuese capaz de guiar de forma háptica al paciente en tiempo real. Durante este proceso se ha estudiado la colocación más adecuada para que la emisión de las vibraciones sean captadas y comprendidas por el paciente.

**IC4:** [*Capacidad de diseñar e implementar software de sistema y de comunicaciones.*]. El sistema resultante está formado por una serie de subsistemas. Cada uno de estos subsistemas es una pieza de *software* diseñada e implementada para que cumplan una serie de funcionalidades necesarias para que tanto el terapeuta como el paciente puedan llevar a cabo el proceso de rehabilitación. Además, para el correcto funcionamiento del sistema se han implementado dos tipos de comunicación, una asíncrona con *API Rest* para la comunicación con el *backend* y una síncrona con un *WebSocket* para comunicar la aplicación del paciente con el sistema de guiado.

**IC5:** [*Capacidad de analizar, evaluar y seleccionar las plataformas hardware y software más adecuadas para el soporte de aplicaciones empotradas y de tiempo real.*].

La implementación del *tracking* corporal en tiempo real requería el uso de un dispositivo especializado en seguimiento de los movimientos corporales. Tras un estudio de las opciones disponibles, como se ha comentado en la Sección 3.4, se ha optado por el dispositivo *Azure Kinect DK*. Para el dispositivo de guiado también era necesario hacer un estudio de las opciones disponibles. El microcontrolador que finalmente se ha elegido ha sido el *ESP8266* con pantalla *OLED* integrada.

## **7.3. TRABAJO FUTURO**

Para finalizar se van a explicar las líneas de trabajo futuro consideradas como más destacables relacionadas con este proyecto:

- El presente proyecto está centrado en la rehabilitación de lesiones de la articulación del brazo. Por esta razón el prototipo de dispositivo de guiado se ha realizado en forma de guante. A pesar de esto, se podrían diseñar nuevos prototipos ampliando así el espectro de articulaciones corporales a rehabilitar con el sistema. Para ello, se debería estudiar la colocación de los actuadores para que las vibraciones fuesen captadas y comprendidas por el paciente.
- Permitir que la aplicación del paciente pudiese retransmitir de forma síncrona las distintas ejecuciones en tiempo real. De tal forma, que el terapeuta pudiese ver al mismo tiempo como está realizando el paciente el ejercicio en cuestión. Además, se podría dar la posibilidad de que el terapeuta pudiera dictar las indicaciones en forma de audio que considere pertinentes. Este último aspecto conllevaría una mejora en el guiado del paciente durante la ejecución de los ejercicios terapéuticos.

- Configuración remota del sistema para el proceso de rehabilitación, ya que en el sistema resultante las funcionalidades principales se pueden controlar por voz, pero algunas características como la configuración del dispositivo de guiado se configura de forma convencional. Por lo tanto, una mejora sería dar la posibilidad de configurar remotamente el dispositivo.

# BIBLIOGRAFÍA

---

- [1] C. C. Abt, *Serious games*. University press of America, 1987.
- [2] (). Acerca de Azure Kinect DK, dirección: <https://docs.microsoft.com/es-es/azure/kinect-dk/about-azure-kinect-dk>.
- [3] J. A. Albert y col., «Evaluation of the pose tracking performance of the azure kinect and kinect v2 for gait analysis in comparison with a gold standard: A pilot study», *Sensors*, vol. 20, n.º 18, pág. 5104, 2020.
- [4] S. Arnab y col., «Enhancing learning in distributed virtual worlds through touch: a browser-based architecture for haptic interaction», en *Serious Games and Edutainment Applications*, Springer, 2011, págs. 149-167.
- [5] A. Asensio Vicente y col., «Sistema de ayuda a invidentes basado en cámaras de profundidad», 2016.
- [6] H. M. Ashour *et al.* (2020). Insights into the recent 2019 novel coronavirus (sars-cov-2) in light of past human coronavirus outbreaks. M. D. P. Institute, Ed., [Online]. Available: <https://www.mdpi.com/2076-0817/9/3/186>.
- [7] A. Baldominos, Y. Saez y C. G. del Pozo. (2015). An approach to physical rehabilitation using state-of-the-art virtual reality and motion tracking technologies, dirección: <https://www.sciencedirect.com/science/article/pii/S1877050915025922>.
- [8] G. Bardas y col., «3d tracking and classification system using a monocular camera», *Wireless Personal Communications*, vol. 92, n.º 1, págs. 63-85, 2017.
- [9] L. J. Barrios y col. (2017). Estado del Arte en Neurotecnologías para la Asistencia y la Rehabilitación en España: Tecnologías Auxiliares, Tránsito Tecnológica y Aplicación Clínica, dirección: <https://polipapers.upv.es/index.php/RIAI/article/view/9181>.
- [10] S. Bhagotra, A. K. Sharma y B. Raina, «Psycho-social adjustments and rehabilitation of the blind», *Social Medicine*, vol. 10, n.º 1, págs. 48-52, 2008.
- [11] A. Bhowmick y S. M. Hazarika, «An insight into assistive technology for the visually impaired and blind people: state-of-the-art and future trends», *Journal on Multimodal User Interfaces*, vol. 11, n.º 2, págs. 149-172, 2017.
- [12] I. Blohm y J. M. Leimeister, «Gamification», *Business & information systems engineering*, vol. 5, n.º 4, págs. 275-278, 2013.
- [13] L. Braille, «Method of writing words, music, and plain songs by means of dots, for use by the blind and arranged for them», *Paris, France: l'Institution Royale des Jeunes Aveugles*, 1829.
- [14] M. I. Bresler y P. A. Bahr, «A HISTORICAL OVERVIEW OF REHABILITATION ENGINEERING»,
- [15] F. Buttussi y L. Chittaro, «Smarter phones for healthier lifestyles: An adaptive fitness game», *IEEE Pervasive Computing*, vol. 9, n.º 4, págs. 51-57, 2010.
- [16] (). C++ addons for Node.js, dirección: <https://nodejs.org/api/addons.html>.
- [17] M. S. Cameirao y col., «The rehabilitation gaming system: a review», *Stud Health Technol Inform*, vol. 145, n.º 6, págs. 65-83, 2009.
- [18] S. Cano y col., «Toward a methodology for serious games design for children with auditory impairments», *IEEE Latin America Transactions*, vol. 14, n.º 5, págs. 2511-2521, 2016.

- [19] A. J. i Capó y col., «Motivational rehabilitation using serious games», *Virtual Archaeology Review*, vol. 4, n.º 9, págs. 167-173, 2013.
- [20] N. Caporusso, «A wearable Malossi alphabet interface for deafblind people», en *Proceedings of the working conference on Advanced visual interfaces*, 2008, págs. 445-448.
- [21] M. E. Castagnola, M. A. Bosio y G. A. Chiodi, «Juegos serios aplicados a niños con discapacidades», en *II Simposio Argentino sobre Tecnología y Sociedad (STS)-JAIIO 44 (Rosario, 2015)*, 2015.
- [22] R. Ceres, M. Á. Mañanas y J. M. Azorín, «Interfaces y sistemas en rehabilitación y compensación funcional para la autonomía personal y la terapia clínica», *Revista Iberoamericana de Automática e Informática Industrial RIAI*, vol. 8, n.º 2, págs. 5-15, 2011.
- [23] Y. Chen, «Olfactory display: development and application in virtual reality therapy», en *16th International Conference on Artificial Reality and Telexistence-Workshops (ICAT'06)*, IEEE, 2006, págs. 580-584.
- [24] J. Cid-Ruzafa y J. Damián-Moreno, «Valoración de la discapacidad física: el índice de Barthel», *Revista española de salud pública*, vol. 71, n.º 2, págs. 127-137, 1997.
- [25] S. Consolvo y col., «Flowers or a robot army? Encouraging awareness & activity with personal, mobile displays», en *Proceedings of the 10th international conference on Ubiquitous computing*, 2008, págs. 54-63.
- [26] J. Córdova-Guarachi, D. Aracena-Pizarro y J. Corrales-Muñoz. (2016). Sistema de monitoreo para pacientes con tratamientos de tendinitis del tendón rotuliano utilizando Kinect, dirección: [https://scielo.conicyt.cl/scielo.php?pid=S0718-33052016000200008&script=sci\\_arttext](https://scielo.conicyt.cl/scielo.php?pid=S0718-33052016000200008&script=sci_arttext).
- [27] Á. Csapó y col., «A survey of assistive technologies and applications for blind users on mobile platforms: a review and foundation for research», *Journal on Multimodal User Interfaces*, vol. 9, n.º 4, págs. 275-286, 2015.
- [28] E. H. M. Dantas y col., «La preponderancia de la disminución de la movilidad articular de la elasticidad muscular en la pérdida de la flexibilidad en el envejecimiento», 2002.
- [29] G. Delgado. (2009). Evaluación del cartilago articular con resonancia magnética, dirección: [https://scielo.conicyt.cl/scielo.php?pid=S0717-93082009000400006&script=sci\\_arttext&lng=e](https://scielo.conicyt.cl/scielo.php?pid=S0717-93082009000400006&script=sci_arttext&lng=e).
- [30] L. Dipietro, A. M. Sabatini y P. Dario, «A survey of glove-based systems and their applications», *Ieee transactions on systems, man, and cybernetics, part c (applications and reviews)*, vol. 38, n.º 4, págs. 461-482, 2008.
- [31] J. Dorado y col., «A computer-vision-based system for at-home rheumatoid arthritis rehabilitation», *International Journal of Distributed Sensor Networks*, vol. 15, n.º 9, pág. 1 550 147 719 875 649, 2019.
- [32] P. DuBois, *MySQL*. Pearson Education, 2008.
- [33] EcuRed. (). Rehabilitación integral para ciegos o débiles visuales, dirección: [https://www.ecured.cu/Rehabilitaci%C3%B3n\\_integral\\_para\\_ciegos\\_o\\_d%C3%A9biles\\_visuales](https://www.ecured.cu/Rehabilitaci%C3%B3n_integral_para_ciegos_o_d%C3%A9biles_visuales).
- [34] W. Elmannai y K. Elleithy, «Sensor-based assistive devices for visually-impaired people: current status, challenges, and future directions», *Sensors*, vol. 17, n.º 3, pág. 565, 2017.
- [35] B. de España. (2020). Escenarios macroeconómicos de referencia para la economía española tras el Covid-19, dirección: <https://repositorio.bde.es/handle/123456789/11721>.
- [36] espressif. (). ESP8266, dirección: <https://www.espressif.com/en/products/socs/esp8266/overview>.
- [37] R. Farcy y col., «Electronic travel aids and electronic orientation aids for blind people: Technical, rehabilitation and everyday life points of view», en *Conference & Workshop on Assistive Technologies for People with Vision & Hearing Impairments Technology for Inclusion*, vol. 12, 2006.
- [38] I. Fette y A. Melnikov, *The websocket protocol*, 2011.
- [39] J. F. F. Flores, «Using gamification to enhance second language learning», *Digital Education Review*, n.º 27, págs. 32-54, 2015.
- [40] F. J. Gallego-Durán y col., «Panorámica: serious games, gamification y mucho más», 2014.

- [41] J. L. González-Mora y col., «Development of a new space perception system for blind people, based on the creation of a virtual acoustic space», en *International Work-Conference on Artificial Neural Networks*, Springer, 1999, págs. 321-330.
- [42] R. W. Hill Jr y col., «Pedagogically structured game-based training: Development of the ELECT BiLAT simulation», UNIVERSITY OF SOUTHERN CALIFORNIA MARINA DEL REY CA INST FOR CREATIVE ..., inf. téc., 2006.
- [43] H. Kajimoto, Y. Kanno y S. Tachi, «Forehead electro-tactile display for vision substitution», en *Proc. EuroHaptics*, 2006.
- [44] W.-S. Kim y col., «A low cost kinect-based virtual rehabilitation system for inpatient rehabilitation of the upper limb in patients with subacute stroke: A randomized, double-blind, sham-controlled pilot trial», *Medicine*, vol. 97, n.º 25, 2018.
- [45] J. P. Kramer, P. Lindener y W. R. George, *Communication system for deaf, deaf-blind, or non-vocal individuals using instrumented glove*, sep. de 1991.
- [46] F. Laamarti, M. Eid y A. El Saddik, «An overview of serious games», *International Journal of Computer Games Technology*, vol. 2014, 2014.
- [47] E. Lachat y col., «Assessment and calibration of a RGB-D camera (Kinect v2 Sensor) towards a potential use for close-range 3D modeling», *Remote Sensing*, vol. 7, n.º 10, págs. 13 070-13 097, 2015.
- [48] O. Lahav, D. W. Schloerb y M. A. Srinivasan, «Rehabilitation program integrating virtual environment to improve orientation and mobility skills for people who are blind», *Computers & education*, vol. 80, págs. 1-14, 2015.
- [49] C. Larman, *Agile and iterative development: a manager's guide*. Addison-Wesley Professional, 2004.
- [50] S. Laxe y col. (2020). La rehabilitación en los tiempos del COVID-19. Elsevier, ed., dirección: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7151338/>.
- [51] B. Loreto Vergara, «Desarrollo de la Medicina Física y Rehabilitación como especialidad médica», 2014.
- [52] E. Lynch, S. Hillier y D. Cadilhac, «When should physical rehabilitation commence after stroke: a systematic review», *International Journal of Stroke*, vol. 9, n.º 4, págs. 468-478, 2014.
- [53] I. Marfisi-Schottman, S. George y F. Tarpin-Bernard, «Tools and methods for efficiently designing serious games», en *Proceedings of the 4th European Conference on Games Based Learning ECGBL*, 2010, págs. 226-234.
- [54] M. Masse, *REST API Design Rulebook: Designing Consistent RESTful Web Service Interfaces*. °Reilly Media, Inc.", 2011.
- [55] J. A. McKanna, H. Jimison y M. Pavel, «Divided attention in computer game play: analysis utilizing unobtrusive health monitoring», en *2009 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, IEEE, 2009, págs. 6247-6250.
- [56] Microsoft. (). Articulaciones para seguimiento de personas de Azure Kinect, dirección: <https://docs.microsoft.com/es-es/azure/kinect-dk/body-joints>.
- [57] —, (). Especificaciones de hardware de Azure Kinect DK, dirección: <https://docs.microsoft.com/es-es/azure/kinect-dk/hardware-specification>.
- [58] (). Motor V8, dirección: <https://v8.dev>.
- [59] I. L. Muñoz. (2018). Ejercicio terapéutico, Asepeyo, dirección: <https://salud.asepeyo.es/profesionales/rehabilitacion/ejercicio-terapeutico/>.
- [60] E. S. de Murieta y M. Supervia, «COVID-19 y cronicidad. Una oportunidad de reinventar los servicios de Medicina Física y Rehabilitación», *Rehabilitación*, 2020.
- [61] R. J. Nadolski y col., «EMERGO: A methodology and toolkit for developing serious games in higher education», *Simulation & Gaming*, vol. 39, n.º 3, págs. 338-352, 2008.
- [62] H. Nicolau y col., «UbiBraille: designing and evaluating a vibrotactile Braille-reading device», en *Proceedings of the 15th International ACM SIGACCESS Conference on Computers and Accessibility*, 2013, págs. 1-8.

- [63] P. Oesch *et al.* (). Effectiveness of exercise on work disability in patients with non-acute non-specific low back pain: Systematic review and meta-analysis of randomized controlled trials. M. J. Limited, Ed., [Online]. Available: <https://www.ingentaconnect.com/content/mjl/sreh/2010/00000042/00000003/art00001>.
- [64] S. Ohtsuka y col., «Body-Braille system for disabled people», en *International Conference on Computers for Handicapped Persons*, Springer, 2008, págs. 682-685.
- [65] W. H. Organization, P. H. A. of Canada, C. P. H. A. of Canada y col., *Preventing chronic diseases: a vital investment*. World Health Organization, 2005.
- [66] M. Orozco y col., «The role of haptics in games», *Haptics rendering and applications*, págs. 217-234, 2012.
- [67] N. Padilla-Zea y col., *Metodología para el diseño de videojuegos educativos sobre una arquitectura para el análisis del aprendizaje colaborativo*. Granada: Universidad de Granada, 2011.
- [68] M. Pedraza-Hueso *et al.* (2015). Rehabilitation using kinect-based games and virtual reality, [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1877050915036947>.
- [69] V. M. Peñeñory, A. F. Bacca y S. P. Cano, «Propuesta metodológica para el diseño de juegos serios para la rehabilitación psicomotriz de niños con discapacidad auditiva», *Campus Virtuales*, vol. 7, n.º 2, págs. 47-54, 2018.
- [70] T. Pillar, E. Gaspar y R. Dickstein, «Physical rehabilitation of the elderly blind patient», *International disability studies*, vol. 12, n.º 2, págs. 75-77, 1990.
- [71] D. C. Prado, «PhyReUP! Sistema basado en realidad aumentada y gamificación para la rehabilitación física», 2020.
- [72] C. M. Reed, «The implications of the Tadoma method of speechreading for spoken language processing», en *Proceeding of Fourth International Conference on Spoken Language Processing. ICSLP'96*, IEEE, vol. 3, 1996, págs. 1489-1492.
- [73] C. M. Reed, N. I. Durlach y L. D. Braida, «Research on tactile communication of speech: a review.», *ASHA monographs*, n.º 20, pág. 1, 1982.
- [74] N. F. Rodrigues y R. Thawonmas, «Guest Editorial: Special Issue on Serious Games for Health», *IEEE Transactions on Games*, vol. 12, n.º 4, págs. 337-340, 2020.
- [75] A. M. de la Salud, «Discapacidad: Proyecto de acción mundial de la OMS sobre discapacidad 2014-2021: Mejor salud para todas las personas con discapacidad: Informe de la Secretaria», inf. téc., 2014.
- [76] A. G. Santiago, J. M. C. Carral y M. Z. González, «De la "minusvalía" visual a la "discapacidad" visual», *Revista de investigación en educación*, vol. 3, págs. 33-50, 2006.
- [77] S. Scarle y col., «Complete motion control of a serious game against obesity in children», en *2011 Third International Conference on Games and Virtual Worlds for Serious Applications*, IEEE, 2011, págs. 178-179.
- [78] N. Shin y col., «Effects of game technology on elementary student learning in mathematics», *British journal of educational technology*, vol. 43, n.º 4, págs. 540-560, 2012.
- [79] J. M. Silva y A. El Saddik, «Exertion interfaces for computer videogames using smartphones as input controllers», *Multimedia systems*, vol. 19, n.º 3, págs. 289-302, 2013.
- [80] (). Sistemas de coordenadas de Azure Kinect DK, dirección: <https://docs.microsoft.com/es-es/azure/kinect-dk/coordinate-systems>.
- [81] R. Tapu, B. Mocanu y T. Zaharia, «Wearable assistive devices for visually impaired: A state of the art survey», *Pattern Recognition Letters*, vol. 137, págs. 37-52, 2020.
- [82] M. Tölgyessy y col., «Evaluation of the Azure Kinect and Its Comparison to Kinect V1 and Kinect V2», *Sensors*, vol. 21, n.º 2, pág. 413, 2021.
- [83] J. Warnecke, N. Devine y C. Olen, «Inpatient physical therapy rehabilitation provided for a patient with complete vision loss following a traumatic brain injury», *Brain Injury*, vol. 29, n.º 7-8, págs. 993-999, 2015.
- [84] K. Werbach y D. Hunter, *The gamification toolkit: dynamics, mechanics, and components for the win*. Wharton School Press, 2015.

- 
- [85] A. Whitehead y col., «Exergame effectiveness: what the numbers can tell us», en *Proceedings of the 5th ACM SIGGRAPH Symposium on Video Games*, 2010, págs. 55-62.
- [86] J. Yim y T. N. Graham, «Using games to increase exercise motivation», en *Proceedings of the 2007 conference on Future Play*, 2007, págs. 166-173.
- [87] G. Zichermann y C. Cunningham, *Gamification by design: Implementing game mechanics in web and mobile apps*. "O'Reilly Media, Inc.", 2011.



# **ANEXOS**



# REPOSITORIOS DEL SISTEMA DESARROLLADO

---

El sistema desarrollado se ha dividido en tres partes principales, cada una de ellas en un repositorio independiente, ya que conviene diferenciarlas.

## REPOSITORIO DE LA APLICACIÓN DE REHABILITACIÓN

La aplicación de rehabilitación esta dividida en dos roles principales, terapeuta y paciente. Se podría decir que cada rol tiene una aplicación diferente, pero ciertos módulos se comparten entre sí, por lo tanto, se ha decidido incluir ambos roles en el mismo repositorio para que se puedan reutilizar estos módulos de forma escalable.



**Figura A.1:** Código QR del repositorio de la aplicación rehabilitación

<https://github.com/mapecode/TFG-RehabilitationApp>

## REPOSITORIO DEL SISTEMA DE GUIADO

El sistema de guiado se ha desarrollado específicamente para el microcontrolador comentado en la Sección 3.4. Por ello, se ha creído conveniente incluir el código desarrollado para este sistema en un repositorio individual.



**Figura A.2:** Código QR del repositorio del sistema de guiado

<https://github.com/mapecode/TFG-Guide>

## REPOSITORIO DEL BACKEND

El *backend* del sistema requiere un repositorio independiente porque se ejecuta en la nube, por ello la creación de un repositorio específico para este sistema facilita el despliegue de éste en la máquina remota correspondiente.



**Figura A.3:** Código QR del repositorio del *backend*

<https://github.com/mapecode/TFG-Backend>