



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

GRADO EN INGENIERÍA EN INFORMÁTICA

**Sistema de soporte a la rehabilitación física de
niños afectados por el síndrome de Prader-Willi**

María Blanco González-Mohíno

Julio, 2023

**SISTEMA DE SOPORTE A LA REHABILITACIÓN FÍSICA DE NIÑOS
AFECTADOS POR EL SÍNDROME DE PRADER-WILLI**



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

Departamento de Tecnologías y Sistemas de Información

**GRADO EN INGENIERÍA EN INFORMÁTICA
COMPUTACIÓN**

**Sistema de soporte a la rehabilitación física de
niños afectados por el síndrome de Prader-Willi**

Autor: María Blanco González-Mohíno

Tutor académico: David Vallejo Fernández

Coautor académico: Lourdes Nieto Moreno-Cid

Julio, 2023

María Blanco González-Mohíno

Ciudad Real – Spain

E-mail: Maria.Blanco4@alu.uclm.es

Teléfono: 689615123

© 2023 María Blanco González-Mohíno

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

Se permite la copia, distribución y/o modificación de este documento bajo los términos de la Licencia de Documentación Libre GNU, versión 1.3 o cualquier versión posterior publicada por la *Free Software Foundation*; sin secciones invariantes. Una copia de esta licencia esta incluida en el apéndice titulado «GNU Free Documentation License».

Muchos de los nombres usados por las compañías para diferenciar sus productos y servicios son reclamados como marcas registradas. Allí donde estos nombres aparezcan en este documento, y cuando el autor haya sido informado de esas marcas registradas, los nombres estarán escritos en mayúsculas o como nombres propios.

TRIBUNAL:

Presidente:

Vocal:

Secretario:

FECHA DE DEFENSA:

CALIFICACIÓN:

PRESIDENTE

VOCAL

SECRETARIO

Fdo.:

Fdo.:

Fdo.:

Resumen

Las enfermedades raras afectan a un 7 % de la población. En ese 7 % co-existen más de 7.000 enfermedades diferentes con sus respectivos tratamientos. Esto, junto a la baja inversión realizada sobre cada una de estas enfermedades por parte de gobiernos y empresas privadas, hace que su diagnóstico y tratamiento sean difíciles de llevar a cabo de manera efectiva.

Dentro de la variabilidad de las enfermedades raras, encontramos el síndrome de Prader-Willi. Se trata de un trastorno genético que afecta, aproximadamente, a 1 de cada 25.000 nacidos en España. Este trastorno está caracterizado, en la mayoría de casos, por la carencia de un gen paterno en el cromosoma 15. Normalmente, el síndrome de Prader-Willi afecta en edades tempranas, generando una composición corporal anómala con baja musculatura corporal.

Las soluciones software aplicadas a esta problemática particular para dar soporte en términos de rehabilitación física son todavía menores. Si bien es cierto que existen algunas propuestas bastante utilizadas, no fueron creadas con este fin y apenas cuentan con elementos de gamificación que incentiven su uso. A esto se le suma la necesidad de elementos hardware como *Wii Remote* o el dispositivo *Microsoft Kinect*, usados para realizar el tracking del cuerpo del usuario y conocer así cómo está realizando los ejercicios de rehabilitación. En algunos casos, estos componentes son poco accesibles debido al coste o antigüedad, lo que dificulta aún más su uso. En este contexto, en el presente Trabajo Fin de Grado se propone crear una aplicación que no requiera hardware específico y que esté basada en realidad aumentada y en técnicas de gamificación, incentivando y guiando la realización de ejercicios físicos utilizados en entornos de rehabilitación para niños con el síndrome de Prader-Willi.

Este sistema hará uso de la cámara del dispositivo (típicamente un ordenador portátil estándar) y será diseñado con la escalabilidad en mente, facilitando el desarrollo de nuevas actividades en espacios de tiempo reducido. Las actividades desarrolladas se han diseñado para ayudar en la mejora de algunas capacidades del usuario como el equilibrio, la coordinación y la flexibilidad.

Abstract

Rare diseases affect around 7 % of the population. In this 7 % there are more than 7,000 different diseases with their respective treatments. This, together with the low investment made in each of these diseases by governments and private companies, makes their diagnosis and treatment difficult to carry out effectively.

Within the variability of rare diseases, we find Prader-Willi syndrome. It is a genetic disorder that affects approximately 1 in 25,000 births in Spain. This disorder is characterized, in the majority of cases, by the lack of a paternal gene on chromosome 15. It usually affects at an early age, generating an abnormal body composition with low body musculature.

The software solutions applied to this particular problem to provide support in terms of physical rehabilitation are still minor. Although it is true that there are some widely used proposals, they were not created for this purpose and hardly have any gamification elements to encourage their use. This is compounded by the need for hardware elements such as Wii Remotes or the Kinect device, used to track the body of the user and thus know how he/she is performing the rehabilitation exercises. In some cases, these components are not very accessible due to cost or age, which makes their use even more difficult. In this context, this Final Degree Project proposes to create an application that does not require specific hardware and is based on augmented reality and gamification techniques, encouraging and guiding the performance of physical exercises used in rehabilitation environments for children with Prader-Willi syndrome.

This system will make use of the camera of the device (typically a standard laptop computer) and will be designed in terms of scalability, facilitating the development of new activities in small time slots. The activities developed will be designed to help in the improvement of some user capabilities such as balance, coordination and flexibility.

Agradecimientos

Primero de todo me gustaría dar las gracias a las diferentes personas involucradas en este proyecto, Lourdes, Carlos y en especial David, tutor de este Trabajo Fin de Grado (TFG), muchas gracias por el apoyo, todo lo enseñado, paciencia y confianza depositada en mí durante este proyecto. Sé que no hubiese sido posible sin todo ese entendimiento ofrecido de tu parte.

También a mi madre, Mari Cruces, por ser mi modelo a seguir y por confiar en mí cuando ni yo misma lo hacía. Gracias por ser un pilar fundamental para poder obtener este resultado, por implicarte y probar todas y cada una de las veces que fuese necesario este proyecto.

Para finalizar, debo mencionar a todas las personas que me acompañaron durante esta etapa, las que se quedaron por el camino y las que llegaron al final de mi mano. Elena, MJ, Andrés, Pablo, José Alberto, Adrián, Laura, Celia, Mario, Rubén; todos y cada uno de vosotros habéis hecho posible que hoy esté aquí y, de una forma u otra, sé que estáis conmigo en todo momento.

Muchas gracias a todos, María

Faayalo zweegbe

Índice general

Resumen	V
Abstract	VII
Agradecimientos	IX
Índice general	XIII
Índice de cuadros	XVII
Índice de figuras	XIX
Índice de listados	XXIII
Listado de acrónimos	XXV
Glosario	XXVII
1. Introducción	1
1.1. Propuesta	3
1.2. Estructura del documento	4
2. Objetivos	7
2.1. Objetivo general	7
2.2. Objetivos específicos	7
3. Estado del arte	9
3.1. Síndrome de Prader-Willi	9
3.2. Mecanismos de gamificación	11
3.2.1. Fundamentos de la gamificación	13
3.2.2. Elementos de gamificación	14
3.3. Soluciones tecnológicas	14

3.3.1.	Aplicaciones relacionadas con rehabilitación física	14
3.3.2.	Aplicaciones sobre rehabilitación física en personas con enfermeda- des raras	19
3.3.3.	Comparativa de las aplicaciones y webs analizadas	21
3.4.	Tecnologías de creación de videojuegos	23
3.4.1.	Comparativa de las diferentes tecnologías	24
3.5.	MediaPipe	25
4.	Metodología	29
4.1.	Desarrollo iterativo e incremental	29
4.1.1.	Planificación de iteraciones	30
4.2.	Herramientas hardware y software	33
4.2.1.	Recursos software	33
4.2.2.	Recursos hardware	35
5.	Arquitectura	37
5.1.	Fundamentos	37
5.2.	Visión general de la arquitectura	38
5.2.1.	Módulo <i>tracker</i>	39
5.2.2.	Módulo de recolección de datos	40
5.2.3.	Módulo <i>stats</i>	41
5.2.4.	Módulo UI	43
5.2.5.	Módulo escenas	44
5.3.	Historias de usuario	50
6.	Resultados	55
6.1.	Aplicación gráfica para fisioterapia en niños con el Síndrome de Prader-Willi	55
6.1.1.	Iteración 2 - ‘Diagonales Superiores’	56
6.1.2.	Iteración 3 - Menú principal y menú de actividades	56
6.1.3.	Iteración 5 - Histórico	58
6.1.4.	Iteración 6 - ‘Squad’	58
6.1.5.	Iteración 7 - ‘Movilidad’	59
6.1.6.	Iteración 8 - Integración de los registros	60
6.1.7.	Iteración 9 - Tutorial	62
6.1.8.	Iteración 10 - Experimentación en un entorno real	63

7. Conclusiones y líneas de trabajo futuro	67
7.1. Consecución de objetivos	67
7.2. Adquisición de competencias	68
7.3. Líneas de trabajo futuras	69
A. Manual de usuario	73
A.1. Consideraciones previas	73
A.2. Instrucciones de uso	73
A.2.1. Calibración	75
A.2.2. Diagonales Superiores	76
A.2.3. Squad	76
A.2.4. Movilidad	77
A.2.5. Tutorial	79
B. Código	81
B.1. Creación de posiciones - Módulo tracking	81
B.2. Exergame 0 - Diagonales Superiores	82
B.3. Exergame 1 - Squad	83
B.4. Exergame 2 - Movilidad	83
Referencias	85

Índice de cuadros

2.1. Tabla asociación objetivo-identificador.	8
4.1. Tabla asociación identificador objetivo-iteración.	32

Índice de figuras

1.1. Tiempo de diagnóstico en años por comunidad autónoma a partir de la aparición de síntomas (n: número de casos). Círculos: valores atípicos que superan la longitud de la caja IQR en 1,5 unidades. Asteriscos: valores atípicos que superan la longitud de la caja IQR en 3 unidades. Imagen extraída de: [BLAMGM ⁺ 22].	2
3.1. Alteraciones faciales del Síndrome de Prader-Willi. Imagen extraída de: http://nuevoarcobaleno.org/detalle-nota.php?nota_id=287nota=15	10
3.2. Causas del SPW. Imagen extraída de: https://www.endocrinologiapediatrica.org/modules.php?name=articulos&idarticulo=485	11
3.3. <i>Framework The Octalysis Framework for Gamification & Behavioral Design</i> . Imagen extraída de: https://yukaichou.com/gamification-examples/octalysis-complete-gamification-framework/	12
3.4. Rehabilitación física analizada con Kinovea. Imagen extraída de: https://mundoentrenamiento.com/kinovea-analiza-tus-movimientos/	15
3.5. Rehabilitación física con Wii Sport y Wii Fit. Imágenes extraídas de: [Fra10] y [How09]	16
3.6. Rehabilitación física de la mano de Kinect Sports. Imagen extraída de: [Per14]	17
3.7. Uso del dispositivo desarrollado por EvolvRehab en los módulos <i>Body</i> . Imagen extraída de: https://evolvrehab.com/	18
3.8. Rehabilitación física de la mano en RehaMetrics. Imagen extraída de: [Reh16]	19
3.9. Aplicación RehApp en uso. Imagen extraída de: https://www.uco.es/rehapp/	20
3.10. Mujer utilizando MindMotion Pro. Imagen extraída de: https://healthiar.com/mindmotion-pro-for-neuro-rehabilitation	21
3.11. Comparativa de aplicaciones y webs analizadas.	22
3.12. Comparativa de las diferentes tecnologías expuestas siguiendo los criterios establecidos.	25
3.13. Grafo sobre detección de puntos faciales. Los cuadros transparentes representan nodos (<i>calculators</i>), los recuadros amarillos representan <i>inputs</i> y <i>outputs</i> externas al gráfico, y las líneas que entran por arriba y por abajo representan el flujo de paquetes. Imagen extraída de: [LTN ⁺ 19]	27
3.14. Landmarks de MediaPipe Pose.	28
4.1. Desarrollo iterativo e incremental. Imagen extraída de: [Gad23]	30

0. ÍNDICE DE FIGURAS

4.2. Diagrama Gantt semanal por iteración.	32
4.3. Esquema sobre las herramientas utilizadas.	35
5.1. Arquitectura del proyecto.	39
5.2. Diagrama UML sobre el histórico.	42
5.3. Diagrama de estados del módulo ‘Actividades’.	46
5.4. Historia de usuario. ID: HU_01.	50
5.5. Historia de usuario. ID: HU_02.	51
5.6. Historia de usuario. ID: HU_03.	51
5.7. Historia de usuario. ID: HU_04.	51
5.8. Historia de usuario. ID: HU_05.	52
5.9. Historia de usuario. ID: HU_06.	52
5.10. Historia de usuario. ID: HU_07.	52
5.11. Historia de usuario. ID: HU_08.	53
5.12. Historia de usuario. ID: HU_09.	53
6.1. Boceto sobre la actividad Diagonales Superiores.	56
6.2. Boceto sobre el menú principal.	57
6.3. Primer boceto sobre el menú actividades.	57
6.4. Modificador de la actividad diagonales.	58
6.5. Primer boceto sobre el módulo historial.	58
6.6. Gráfica sobre el <i>exergame</i> Diagonales Superiores.	59
6.7. Boceto sobre la actividad Squad.	59
6.8. Modificador de la actividad Squad.	60
6.9. Boceto sobre la actividad Movilidad.	60
6.10. Modificador de la actividad Movilidad.	60
6.11. Gráfica sobre el <i>exergame</i> Squad.	61
6.12. Gráfica sobre el <i>exergame</i> Movilidad.	61
6.13. Estadísticas sobre Squad y Movilidad, izquierda y derecha respectivamente.	62
6.14. Tutorial sobre la escena del histórico.	62
6.15. Realización de ‘Diagonales Superiores’ por parte de Manuel. En la imagen, el niño se moviliza en la dirección del elemento pero no lo alcanza.	63
6.16. Realización de ‘Squad’ por parte de Manuel. En la imagen, el jugador forma un ángulo de 150° aproximadamente.	64
6.17. Realización de ‘Movilidad’ por parte de Manuel.	65
A.1. Menú de inicio.	74

A.2. Modificación de actividad.	74
A.3. Inicio de actividad.	75
A.4. Calibración.	75
A.5. Actividad Diagonales Superiores.	76
A.6. Actividad Squad.	77
A.7. Actividad Movilidad.	77
A.8. Histórico de la actividad Diagonales Superiores.	78
A.9. Histórico de la actividad Squad.	78
A.10. Histórico de la actividad Movilidad.	79

Índice de listados

B.1. Bloque de código en el que se extrae la información requerida de MediaPipe. La creación de este diccionario genera una solución a la problemática de la obtención de la posición.	81
B.2. Bloque de código en el que se calcula la posición alcanzable de la mano izquierda sobre el eje x.	82
B.3. Fragmento en el que se calcula la posición máxima alcanzable de la mano sobre el eje y.	82
B.4. Método para el cálculo del ángulo flexor de la rodilla.	83
B.5. Bloque de código en el que se actualiza e instancia la posición actual del usuario en sprites.	83

Listado de acrónimos

EERR Enfermedades Raras

OMS Organización Mundial de la Salud

SPW Síndrome de Prader-Willi

TIC Tecnologías de la Información y las Comunicaciones

TFG Trabajo Fin de Grado

Glosario

Disfunción hipotálamo-hipofisaria	Es un trastorno poco frecuente en el cual la glándula de la hipófisis (glándula en la base del cerebro la cual genera hormonas que intervienen en las funciones biológicas básicas) no produce una o más hormonas, o bien la cantidad que produce es insuficiente.
Estado de metilación	La metilación es la unión de un grupo metilo (-CH ₃) a la citosina, una de las bases nitrogenadas del ADN, lo que desemboca en la inhibición de la expresión de algunos genes. El análisis del estado de metilación del ADN se realiza mediante técnicas moleculares que permiten detectar la presencia o ausencia de la metilación en ciertas regiones del genoma.
Enfermedad rara	Según la Organización Mundial de la Salud (OMS) se considera enfermedad rara aquella que afecta a menos de 1 de cada 2.000 personas.
Hiperfagia	Apetito excesivo o anormalmente aumentado.
Hipotonía grave	Disminución significativa del tono muscular, lo que conlleva debilidad generalizada en los músculos del cuerpo.
Periodo neonatal	4 primeras semanas de vida.

Capítulo 1

Introducción

ENTENDEMOS por enfermedad rara aquella que, con peligro de muerte o invalidez, está presente en menos de 5 por cada 10.000 habitantes [EuC99]. De acuerdo a esta definición, la OMS estima que las enfermedades raras están presentes en un 7 % de la población mundial. Esto significa que en torno 3 millones de españoles sufren algún tipo de enfermedad rara en este momento [enf19].

*Orphanet*¹, una de las mayores bases de datos europeas sobre enfermedades raras, utiliza esta definición para recoger un listado de más de 7.000 enfermedades distintas. En torno al 80 % de estas enfermedades tienen origen genético [BBP⁺18], por lo que su diagnóstico es complejo y muchas aún no tienen tratamiento específico.

Los síntomas de una enfermedad rara pueden estar presentes desde el nacimiento, incluso en la etapa de gestación, aunque sobre todo aparecen en los dos primeros años de vida del niño [Sil22]; aún así, según la Federación Española de Enfermedades Raras (FEDER²), un paciente espera de media 4 años hasta recibir un diagnóstico. En el 20 % de los casos transcurren 10 o más años hasta lograr el diagnóstico adecuado. En la figura 1.1 podemos apreciar la media por comunidad autónoma de estos diagnósticos. La línea dentro de la caja representa la mediana de cada comunidad autónoma, y la línea roja vertical, la mediana de España. Entonces, ¿por qué no se están detectando estas enfermedades en plazos más cortos y efectivos?

Dada la prevalencia de cada una de las enfermedades raras, el conocimiento sobre cada una de estas se encuentra fragmentado y la inversión en cada una, por ende, es ínfima. Para ello la Unión Europea desplegó una serie de planes presupuestarios orientados a realizar labores de inversión y desarrollo, *Programa Marco para la Investigación y la Innovación*, entre otros, en los que las Enfermedades Raras (EERR) se encuentran integradas. Dentro de estos planes presupuestarios hallamos proyectos más concretos como *Solve-RD* o *Screen-4Care*.

¹<https://www.orpha.net>

²<https://www.enfermedades-raras.org/enfermedades-raras/conoce-mas-sobre-er/enfermedades-raras-en-cifras>

1. INTRODUCCIÓN

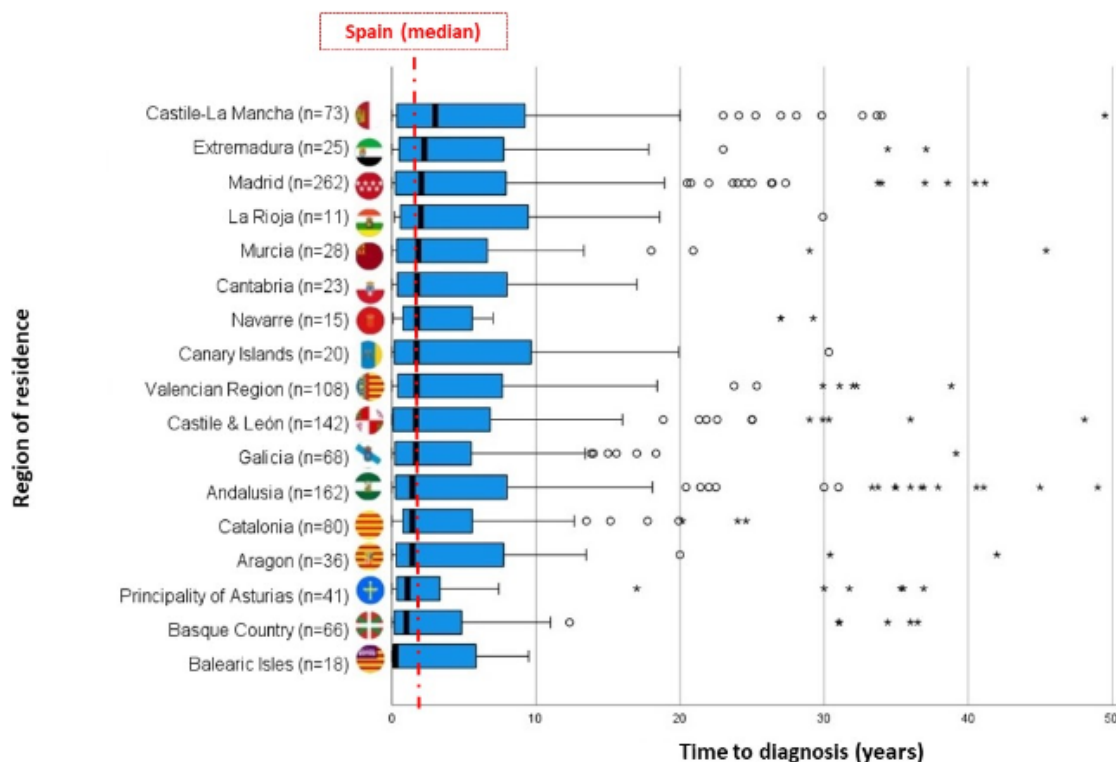


Figura 1.1: Tiempo de diagnóstico en años por comunidad autónoma a partir de la aparición de síntomas (n: número de casos). Círculos: valores atípicos que superan la longitud de la caja IQR en 1,5 unidades. Asteriscos: valores atípicos que superan la longitud de la caja IQR en 3 unidades. Imagen extraída de: [BLAMGM⁺22].

Dentro de estas enfermedades se encuentran algunas bastante conocidas como la Hemofilia A, una enfermedad hematológica hereditaria poco frecuente de origen genético y caracterizada por hemorragias espontáneas o prolongadas debido a un déficit del factor VIII. Esta enfermedad la sufren en torno a 3.000 personas en España, según datos recogidos por la Federación Española De Hemofilia³. Otra de las enfermedades raras más conocidas es la Narcolepsia, trastorno del sueño que genera somnolencia durante el día. Según la Asociación Española del Sueño (ASENARCO) hay en torno a 20 casos por 100.000 habitantes.

La lucha contra estas enfermedades se está enfocando de manera multidisciplinar, siendo la inversión en investigación científica una de las claves para entender las causas y el desarrollo de cada enfermedad y crear terapias y tratamientos efectivos. Estas investigaciones nos conducen a terapias mediante agentes físicos, como la hidroterapia o la electroterapia; y terapias manuales, como la *kinesioterapia* y diferentes fisioterapias [Pér19].

Los niños y niñas con enfermedades raras en edades tempranas suelen hacer uso de diferentes fisioterapias, aunque cada una de estas enfermedades es única y los beneficios y riesgos que conllevan pueden variar de una persona a otra; los resultados suelen ser positivos, como se han demostrado en diferentes estudios. Algunos de estos vienen de la mano de

³<https://fedhemo.com/>

la revista *Orphanet*, como en su estudio “*Exercise training in adults with Pompe disease: the effects on pain, fatigue, and functioning*”, en el que se evaluó los efectos del ejercicio aeróbico en personas con la enfermedad del Pompe, los resultados mostraron que este ejercicio mejoró la capacidad respiratoria y la calidad de vida en los participantes [FBK⁺15].

Junto a estas fisioterapias, muchos profesionales se han ayudado de la rama tecnológica de los *e-Health*. Este campo tiene su origen en la época de los 90, cuando comenzaron a surgir los primeros sistemas de información médica electrónica. El término *e-Health* se acuñó en 1999 por la OMS, que lo definió como “un término genérico para el uso de las Tecnologías de la Información y las Comunicaciones (TIC), para apoyar la salud y la atención médica”. Desde entonces, este campo ha evolucionado rápidamente y ha abarcado áreas como la telemedicina, la salud móvil, la informática médica y la inteligencia artificial, entre otras [Wor99].

Dentro de las diferentes ramas de los *e-Health* se encuentran los *Serious Games* o Juegos Serios, término acuñado por Clark C. Abt en 1970 en su libro *Serious Games* [Abt70]. Estos juegos serios se refieren a aquellos videojuegos que se utilizan en el sector educativo, científico, médico, ingeniería y política entre otros. Un área de estos juegos son los llamados **exergames**, también conocidos como juegos de ejercicio, juegos de vídeo que combinan los dos mundos. Suelen ser diseñados para jugarlos con consolas de videojuegos, dispositivos móviles o computadoras, y requieren que los usuarios realicen movimientos corporales para interactuar con el juego.

Algunos utilizan accesorios, como sensores de movimiento, bandas de resistencia o plataformas de equilibrio para mejorar la experiencia del usuario y hacer que el ejercicio sea más efectivo, como el popular *Kinect Sport* o *Wii Fit*, los cuales se desarrollaron pensando en un perfil de usuario bastante alejado de las labores de fisioterapia aplicada a enfermedades raras.

1.1 Propuesta

Dentro de este contexto se enmarca este TFG, en el que se busca diseñar y desarrollar un sistema *software* capaz de dar soporte en la realización de labores de fisioterapia en niños con el Síndrome de Prader-Willi, síndrome que afecta entorno a 1 de cada 25.000 nacidos en nuestro país [AES22]. Este sistema será probado por un niño que sufre este síndrome en un entorno escolar que cuenta con un aula de educación especial y aula de fisioterapia.

Este sistema *software* se basa en actividades realizadas en diferentes sesiones de fisioterapia, las cuales han sido adaptadas para su uso en computador junto con elementos de gamificación para aumentar la motivación a la hora de realizar estos ejercicios. Estas actividades han sido desarrolladas en coordinación con dos expertos en el síndrome, lo que hace que el *software* haya sido diseñado y desarrollado enfocado en esta enfermedad. Este enfoque ha permitido que las actividades aporten avances en ciertos aspectos fisiológicos del

1. INTRODUCCIÓN

usuario, como la mejora del equilibrio, mejoras en la coordinación viso-motriz y aumento de musculatura corporal, entre otras.

Se ha creado un entorno accesible para todos los centros educativos y de fisioterapia, el cual únicamente hace uso de un ordenador y una cámara. Esto hace el *software* accesible a la mayoría de usuarios o centros con bajos recursos. Para conseguir una mejor visualización de la herramienta se hará uso una televisión o pantalla como recurso *hardware*. Este es un *software* que hace uso de realidad aumentada por medio de la detección corporal del usuario para realizar interacciones [RB17]. Será accesible desde los principales sistemas operativos, Windows, Linux y MacOs.

Desde el punto de vista de la arquitectura, este software está basado en módulos, lo que proporciona mayor escalabilidad en el desarrollo de nuevas funcionalidades y actividades, y permite el almacenamiento de mediciones corporales durante los ejercicios para proporcionar estadísticas cuantitativas sobre la mejora del usuario.

1.2 Estructura del documento

En esta sección se aborda la estructura del documento siguiendo la normativa vigente.

Capítulo 1: Introducción

En este capítulo se realiza una pequeña introducción que explica la problemática que ha llevado a la realización de este TFG, también se realiza la estructura del documento, en la que se pretende dar una visión general de la distribución.

Capítulo 2: Objetivos

Se recoge el objetivo principal de este TFG y los objetivos específicos necesarios para su correcta resolución.

Capítulo 3: Estado del arte

Se incluye un análisis de aplicaciones similares al fin de este TFG, así como algunas plataformas utilizadas por fisioterapias en casos de enfermedades raras. Además, también se realizará una comparativa entre diferentes tecnologías de creación de videojuegos, y una breve introducción al *framework* MediaPipe.

Capítulo 4: Metodología

Abarca los métodos seguidos durante el desarrollo del proyecto. También se encuentran incluidas las diferentes iteraciones en las que se ha subdividido el trabajo y las herramientas hardware y software utilizadas.

Capítulo 5: Arquitectura

Se centra en la arquitectura general del proyecto, características en el diseño e implementación sobre las partes más relevantes. Además, se incluyen diferentes diagramas para un mayor entendimiento de la arquitectura.

Capítulo 6: Resultados

Se discuten los resultados más relevantes centrándose en el diseño del sistema junto con una pequeña prueba en un entorno de rehabilitación real.

Capítulo 7: Conclusiones y líneas de trabajo futuro

A lo largo de este capítulo se expone un análisis los objetivos alcanzados, principal y específicos, junto con las competencias adquiridas, por último, se exponen diferentes líneas de trabajo futuro.

Capítulo 2

Objetivos

EN este capítulo se discute el objetivo general de este TFG. Este objetivo se detalla, a su vez, en base a un objetivo principal y una serie de objetivos específicos necesarios para alcanzar el objetivo general. Esta división en objetivos específicos facilita la consecución del objetivo general y posibilita la construcción de manera incremental del sistema propuesto en el presente TFG.

2.1 Objetivo general

El **objetivo general** de este proyecto es el **diseño, desarrollo y validación de un sistema de soporte para la rehabilitación físico-cognitiva de niños afectados por síndrome de Prader-Willi**, con el propósito de generar un entorno atractivo y motivacional en el que los niños no encuentren limitaciones a la hora de realizar sus actividades.

Este sistema se centrará en las necesidades básicas, en términos de rehabilitación física y con la colaboración de expertos en el dominio, de personas en edades tempranas con este síndrome permitiendo realizar ejercicios usados frecuentemente durante las sesiones de rehabilitación.

A continuación, se introducen los objetivos específicos que se han marcado para el desarrollo del sistema en su totalidad.

2.2 Objetivos específicos

Para el **estudio detallado de la caracterización del Síndrome de Prader-Willi (SPW) y la selección de ejercicios físicos para niños que ya tienen adquirida la marcha** será requerida la investigación del síndrome por medio de fuentes fiables con información contrastada y expertos con experiencia, con el fin de poder realizar una caracterización lo más acorde con la realidad y seleccionar los ejercicios que representen mayor progresión psicomotora en el usuario.

Se desarrollarán mecánicas y dinámicas de gamificación. Además, se incluirán componentes que fomenten la motivación de los ejercicios en el usuario. Para poder elegir los elementos que mejor se adapten al sistema se realizará un **estudio, selección e implementación de mecanismos de gamificación**.

2. OBJETIVOS

Puesto que el perfil de usuario final está orientado a personas de temprana edad no familiarizadas con el uso de la informática, es vital el **diseño y desarrollo de mecanismos de interacción orientados a pacientes con SPW**. Para esto se creará un sistema intuitivo y visualmente sencillo que permita la interacción corporal del usuario con el entorno.

Para realizar el **diseño, desarrollo e integración de un módulo de personalización de ejercicios físicos**, se implementará una sección en la que el usuario pueda cambiar las diferentes características de cada ejercicio. Se gestionará todo lo referente a tiempos de juego, tiempo de aparición y configuraciones propias de cada uno de los *exergames*.

Se realizará una etapa de **experimentación del sistema desarrollado con un paciente real en coordinación con el experto clínico**. Esto implica la realización de pruebas con el usuario para determinar la eficacia y la usabilidad del sistema en un entorno real.

Actualmente se encuentra seleccionado un paciente real, con el SPW. Se observará cómo el paciente interactúa con él y cómo el sistema responde a las acciones del usuario. Se comprobará si los ejercicios resultan efectivos y podrían tener alguna implicación positiva en el desarrollo físico-cognitivo del paciente.

La tabla 2.1 muestra de manera resumida los objetivos específicos planteados, asociando un identificador a cada objetivo.

Identificador	Objetivo
OBJ_1	Estudio detallado de la caracterización del Síndrome de Prader-Willi (SPW) y selección de ejercicios físicos para niños que ya tienen adquirida la marcha.
OBJ_2	Estudio, selección e implementación de mecanismos de gamificación.
OBJ_3	Diseño y desarrollo de mecanismos de interacción orientados a pacientes con SPW.
OBJ_4	Diseño, desarrollo e integración de un módulo de personalización de ejercicios físicos.
OBJ_5	Experimentación del sistema desarrollado con un paciente real en coordinación con el experto clínico.

Cuadro 2.1: Tabla asociación objetivo-identificador.

Capítulo 3

Estado del arte

EN este capítulo se expondrá el contexto actual de este TFG, las distintas aplicaciones en uso para la realización de ejercicios físicos, y las diferentes soluciones software utilizadas en la actualidad en el contexto de enfermedades raras. Además, también se realizará un análisis de las diferentes herramientas elegidas para el desarrollo del proyecto software junto con su justificación de uso.

3.1 Síndrome de Prader-Willi

El síndrome de Prader-Willi o síndrome de Prader-Labhart-Willi (terminología en desuso) es una enfermedad genética rara que afecta directamente al neurodesarrollo del individuo. Los signos y síntomas de esta enfermedad fueron descritos por primera vez en 1956 por los doctores Andrea Prader, Alexis Labhart y Heinrich Willi, aunque su descripción inicial se atribuye a Dr. John Langdon Down a finales del siglo XIX [CHH⁺19].

Según Orphanet, afecta a 1 de cada 15.000-30.000 personas nacidas en todo el mundo [orp22], lo sufren ambos sexos por igual y no se ha reportado que tenga predominio en ninguna etnia o grupo humano en particular. Aun siendo una enfermedad genética, el porcentaje de repetición dentro de una familia es inferior al 1 % [AES22].

Descripción clínica del síndrome

Las personas con el SPW presenta un disfunción hipotalámica, la cual explica gran parte de los síntomas cómo la hiperfagia severa, la dificultades en la regulación térmica, altos umbrales para el dolor, apneas del sueño o déficits hormonales. En edades tempranas las anomalías hipotalámicas se traducen en una composición corporal atípica, con poco desarrollo muscular e incremento de masa grasa en muchos casos.

Frecuentemente la hipotonía se desarrolla en la etapa de gestación y suele manifestarse con disminución del movimiento del neonato, posición fetal anómala en el parto y mayor necesidad de parto asistido o cesárea. En edades tempranas hay disminución del movimiento, llanto débil, reflejos pobres e incluso una mala succión lo que conduce a dificultades de alimentación temprana y un lento aumento de peso.

3. ESTADO DEL ARTE

En la figura 3.1 podemos ver las alteraciones faciales más recurrentes que presentan las personas con el SPW. Suelen estar caracterizadas por tener una frente estrecha, labios delgados y con comisuras inclinadas hacia abajo en la mayoría de casos y ojos almendrados. También pueden presentar estrabismo, dorso nasal estrecho y, en muchas ocasiones, manos y pies pequeños.

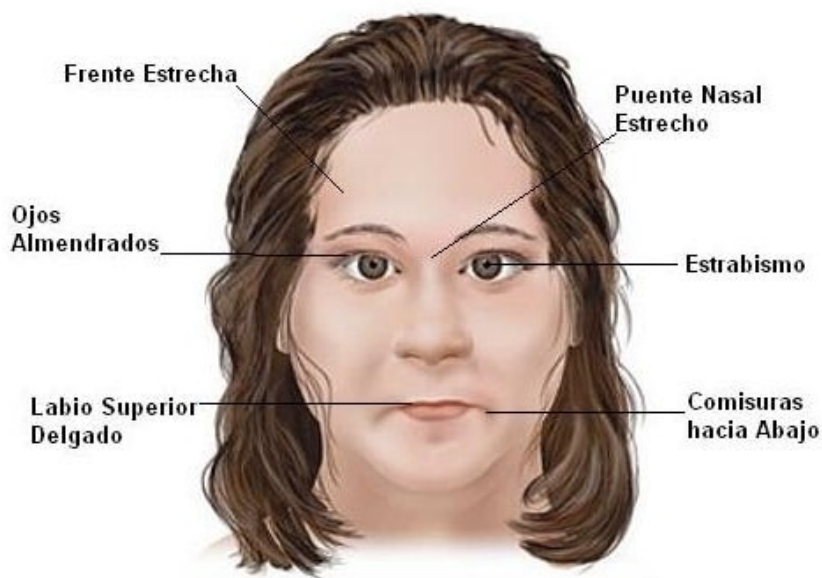


Figura 3.1: Alteraciones faciales del Síndrome de Prader-Willi. Imagen extraída de: http://nuevoarcobaleno.org/detalle-nota.php?nota_id=287nota=15

A estos síntomas se suele unir el retraso cognitivo, y el grado varía ampliamente de un niño a otro. También se asocia con problemas de aprendizaje, habla y desarrollo del lenguaje, agravados por problemas de comportamiento y psicológico.

Dentro de los problemas de comportamiento, las personas con el SPW son descritas como irritables, testaduras, inestables emocionalmente, con temperamento fuerte y ataques de rabia que pueden incluir cierto nivel de agresividad.

Posibles causas genéticas

Existen diferentes alteraciones genéticas que originan el SPW, las cuales, con un análisis del estado de metilación del ADN en las regiones críticas de Prader-Willi, llegan a un diagnóstico en más del 99 % de los casos.

La mayoría de alteraciones genéticas que originan el SPW tiene como causa común la pérdida o inactivación de genes paternos en la región 15q11-q13 de cromosoma 15. El tipo de alteraciones genéticas y frecuencia de las mismas son las siguientes:

- Deleción «de novo» en el cromosoma 15 paterno en el 70-75 % de los casos.
- Disomía uniparental del cromosoma 15 materno en el 20-25 % de los casos.

- Herencia de dos copias del cromosoma 15 de su madre y ninguna copia del cromosoma 15 del padre.
- Alteración de la impronta en el 2-5 %. Ciertos genes se expresan o silencian en función de la herencia parental.
- Reorganizaciones cromosómicas en el 1 %.

De forma visual se pueden observar estas posibles causas genéticas en la figura 3.2.

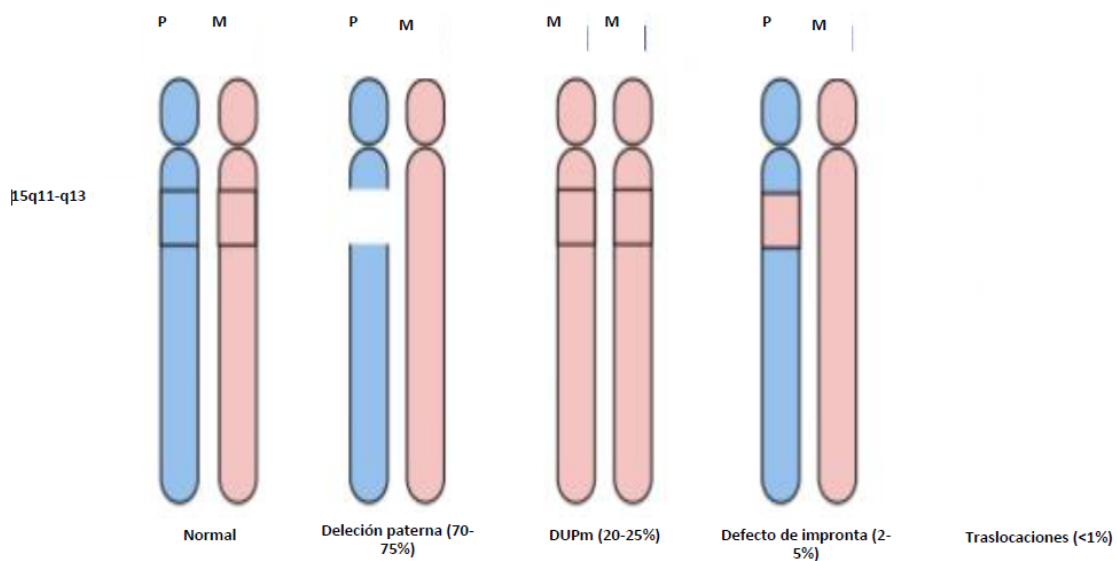


Figura 3.2: Causas del SPW. Imagen extraída de: <https://www.endocrinologiapediatrica.org/modules.php?name=articulos&idarticulo=485>

3.2 Mecanismos de gamificación

En 1984 Charles Coonradt observó el valor de añadir elementos usados en videojuegos al trabajo cotidiano, en su libro *The Game of Work* [Coo12] ofrece un análisis sobre el porqué la gente podría pagar por realizar su *hobby* favorito en vez de ir a un trabajo por el que son pagados. En él, se llega a 5 conclusiones sobre el porqué se prefieren los distintos *hobbies*: engloban metas bien definidas, mayor rastreo del progreso, mayor frecuencia de *feedback*, métodos de ejecución elegidos por el usuario y entrenamiento constante.

Más adelante esto sirvió de referencia a Yu-Kai en su viaje por los mundos de la gamificación. Chou observó que la mayoría de juegos no tienen elementos que los hagan más intuitivos sobre lo que hacer o mecanismos que aporten *feedback* sobre cuáles son los siguientes pasos u opciones posibles. Razón por la que se pregunta, en su libro *Actionable Gamification* [Cho15], cómo puede hacer que los juegos tengan más sentido, y sobre todo, cómo puede hacerlos más divertidos.

3. ESTADO DEL ARTE

Yu-Kai Chou describe la gamificación como el arte de hacer algo jugable y relata como el diseño enfocado a humanos, en el que se optimizan sentimientos y motivaciones en vez de la eficiencia, es el más utilizado por empresas dedicadas a desarrollo de videojuegos. Estas empresas han pasado decadas investigando como captar consumidores por medio de elementos de gamificación razón por la que Yu-Kai Chu elaboró un *framework* para analizar y construir estrategias sobre sistemas para captar más usuarios llamado *The Octalysis Framework for Gamification & Behavioral Design*¹.

Este *framework*, figura 3.3 consta de 8 *Core Drives* o unidades centrales, en el que para realizar un juego con buenas bases de gamificación se deberá focalizar el desarrollo en, como mínimo, una unidad. El *framework* se encuentra posicionado de forma que en la parte izquierda nos encontramos la parte más creativa, social (motivación extrínseca), mientras que la parte derecha se centra en factores más lógicos y analíticos (motivación intrínseca). Las unidades colocadas arriba son las que ofrecen motivaciones positivas, las que hacen al usuario sentirse bien, mientras que las colocadas abajo refuerzan emociones más negativas. Junto a estos ocho *cores* el autor añade uno más, la **sensación**, en relación al placer físico.



Figura 3.3: *Framework The Octalysis Framework for Gamification & Behavioral Design* . Imagen extraída de: <https://yukaichou.com/gamification-examples/octalysis-complete-gamification-framework/>

¹<https://yukaichou.com/gamification-examples/octalysis-complete-gamification-framework/>

3.2.1 Fundamentos de la gamificación

La gamificación engloba una serie de técnicas, mecánicas y elementos en el momento de diseñar un videojuego que lo hacen más atractivo, estos elementos añadidos intentan involucrar al usuario aumentando la motivación y el compromiso con el juego.

La **teoría conductista**, descrita por John Watson [Rib10] en 1913, ha aportado ciertos puntos de vista a la gamificación. Esta teoría se basa en el factor **estimulo-respuesta** ya que sostiene que el comportamiento humano puede ser moldeado en base a las diferentes consecuencias que conlleva la realización de una tarea. Según esta teoría las personas aprenden asociando un estímulo con una respuesta, en la que se refuerza o castiga un comportamiento para aumentar o disminuir su frecuencia.

De esta forma, la gamificación utiliza los principios conductistas para crear experiencias de juego atractivas y motivadoras al usuario, de manera que refuerza positivamente las acciones correctamente realizadas y negativamente las acciones erróneas.

De igual manera la **teoría cognitivista**, creada por Jean Piaget, sostiene que el aprendizaje no sólo depende de la experiencia, sino también de la forma en que se procesa y se organiza la información en el cerebro. Esta teoría se organiza en dos tipos de recompensas.

- **Recompensas intrínseca.** Aquellas en las que el usuario reacciona a la acción en sí misma, sin tener en cuenta las consecuencias.
- **Recompensas extrínseca.** El usuario actúa en función de una recompensa externa. Asimismo, estas recompensas externas se clasifican en 4 grupos (SAPS) [ZC11]:
 - **Status:** tablas de clasificación.
 - **Acceso:** permite acceder a partes bloqueadas para otros usuarios.
 - **Poder:** otorgan habilidades que otras personas no poseen, como poder saltarse ciertas partes de un videojuego sin repercusiones negativas.
 - **Stuff:** recompensas tangibles.

3.2.2 Elementos de gamificación

Los elementos de la gamificación se subdividen en dos categorías, **dinámicas**, efectos, motivaciones, y **mecánicas**, reglas que pueden generar una cierta “adicción”.

En cuanto a las dinámicas de juego se utilizan componentes como progresión y restricción de secciones de juego, por lo que el usuario se encuentra motivado para conseguir aumentar su nivel o desbloquear una nueva parte.

Las mecánicas más comunes son los retos o el *feedback* con premios por acciones completadas correctamente.

Componentes

Entre los elementos que se incluyen en el juego, tanto dinámicas como mecánicas más comunes, destacan aquellas que generan una motivación mayor en el usuario, las más recurrentes son las siguientes:

- **Insignias**
- **Desbloques**
- **Animaciones**
- **Niveles**
- **Puntos**
- **Tablas de clasificación**
- **Objetos virtuales**

3.3 Soluciones tecnológicas

Actualmente, existen muchas plataformas para promover la actividad física con un propósito similar a este Trabajo Fin de Grado, pero la mayoría requieren componentes hardware poco accesibles y tiene un software complejo para niños.

En esta sección, se presenta un análisis sobre aplicaciones y páginas web relacionadas con la rehabilitación física y aplicaciones sobre rehabilitación física en personas con enfermedades raras.

Por otro lado, se incluye una comparativa de las tecnologías de creación de videojuegos más usadas y se discute su uso en este proyecto.

3.3.1 Aplicaciones relacionadas con rehabilitación física

Las aplicaciones realizadas por visión por computador están actualmente en auge, en su gran mayoría, gracias a tecnologías de *machine learning*, en consecuencia cada vez es más común el uso de estas aplicaciones en contextos de rehabilitación física, como *Kinovea* o *Fitbod*; algunos *exergames* podrían englobarse dentro de esta agrupación, aunque, por lo



Figura 3.4: Rehabilitación física analizada con Kinovea. Imagen extraída de: <https://mundoentrenamiento.com/kinovea-analiza-tus-movimientos/>

general, utilizan elementos *hardware* para interactuar con el sistema, como *Wii Fit* o *Wii Sport*.

A continuación, se incluyen los juegos más relevantes de ambas categorías con el objetivo de dar una visión global del mercado.

Kinovea

Es un *software* gratuito de análisis de actividades, desarrollado en C++, está disponible en *Windows*, *Linux* y *MacOS*, fue diseñado especialmente para deportistas y médicos, aunque ha sido usado en numerosos estudios de rehabilitación física. Analiza el ejercicio en tiempo real y permite realizar diferentes mediciones de movimiento, como ángulos, distancias, velocidad y aceleración, entre otros.

Kinovea funciona segmentando las imágenes que recoge, de manera que separa la persona del medio en el que se encuentra detectando bordes y objetos. De esta forma correlaciona los patrones de movimiento de la persona que usa la aplicación con el movimiento óptimo que debería seguir para que el ejercicio fuese perfecto. También cuenta con herramientas de calibración y sincronización que permiten analizar movimientos de distintas partes del cuerpo de forma simultánea.

Este programa almacena las diferentes mediciones que recoge, lo que permite analizar la evolución técnica de los ejercicios realizados.

Un estudio realizado en 2022 de la mano de la Universidad Industrial de Santander, Colombia, utilizó Kinovea para realizar un estudio sobre la efectividad de una nueva muleta de manos libres para individuos con lesiones en el tercio distal del miembro inferior (pierna, tobillo y pie) [MVVBS22].

Wii Fit y Wii Sport

Wii Fit y Wii Sport son exergames creados por la compañía *Nintendo* en 2008 y 2007 respectivamente. Ambos se basan en una serie de mini-juegos pensados para ser controlados por elementos externos o periféricos.

Wii Sport utiliza dos “*Wiimotes*” o mandos Wii para simular el movimiento de las manos en la pantalla, a modo de puntero, y una barra de sensores de infrarrojos para recoger el movimientos de los mandos. El objetivo de Wii Sport es simular la realización de diferentes deportes por medio de los periféricos, en él se incluían deportes como bolos o béisbol.

Wii Fit por el contrario utilizaba una plataforma o *Wii Balance Board*, una tabla capaz de recoger la presión ejercida sobre ella, y un *Wiimote* para la navegación por el sistema. La finalidad era la realización de diferentes ejercicios físicos como yoga o boxeo.

Ambos contaban con un seguimiento del progreso a lo largo del tiempo, lo que dió pie a numerosos estudios sobre los beneficios de estos dos juegos, uno de los más relevantes fue “*Effects of Exergaming on Physical Fitness in Elementary School Students: A Pilot Study*” [TSF⁺13], el cual contó con 32 niños que participaron en un programa de 10 semanas. Como resultado los niños que participaron obtuvieron una mejora en cuanto a aptitud física, incluyendo consumo máximo de oxígeno, fuerza muscular y flexibilidad.



Figura 3.5: Rehabilitación física con Wii Sport y Wii Fit. Imágenes extraídas de: [Fra10] y [How09]

Kinect Sports

Kinect Sports es un *exergame* desarrollado por *Rare* y lanzado por *Microsoft Game Studios* para la videoconsola Xbox 360 en 2010.

Consiste en una serie de videojuegos que hacen uso del sensor Kinect, sensor con cámara integrada que permite obtener imágenes de la habitación, incluido objetos, proyectando un sensor infrarrojo. Este sensor cuenta con cámara de vídeo en color (RGB), emisor IR (infrarrojo), cámara de profundidad, micrófonos y motor de inclinación.

Kinect Sports utiliza este sensor para monitorizar los movimientos del cuerpo, detectándolo con gran precisión y simulando el movimiento en los distintos deportes que se encuentran incluidos en el videojuego, como atletismo o boxeo. Estos mini-juegos cuentan con mecanismos de gamificación que hacen la experiencia más atractiva al usuario.

En la actualidad, Kinect es un dispositivo de bajo coste fácilmente accesible, el cual, junto a los juegos desarrollados por *Rare* forman un ambiente idóneo para realizar ejercicio físico en espacios cerrados o actividades de rehabilitación en entornos motivacionales.



Figura 3.6: Rehabilitación física de la mano de Kinect Sports. Imagen extraída de: [Per14]

EvolvRehab

Conocidos por muchos como Virtual Rehab, es un *software* desarrollado por la empresa *VirtualWare*, un fabricante de productos *software* y *hardware* con certificado CE. Están enfocados en la rehabilitación y trabajan con diferentes centros hospitalarios.

Cuentan con múltiples secciones de rehabilitación diferenciando las diferentes áreas a tratar. EvolRehab² Body ofrece terapia para extremidades superiores e inferiores centrados en deterioros neuromotores. Ofrece tres módulos diferentes, evaluaciones, ejercicios y *exergames*, cada módulo es personalizable en función del paciente y tiene una interfaz intuitiva de fácil acceso. Cada una de los ejercicios y exergames recoge diferentes datos como flexiones y rotaciones de extremidades para presentarlos de forma gráfica en un módulo al que han llamado gestor en el menú principal.

Otra de sus secciones es EvolRehab Hand orientada al entrenamiento de las habilidades motoras finas a través de juegos. Incluye actividades para la flexión, extensión y abducción

²<https://evolvrehab.com/>



Figura 3.7: Uso del dispositivo desarrollado por EvolvRehab en los módulos *Body*. Imagen extraída de: <https://evolvrehab.com/>

de los dedos y la muñeca. Al igual que otras secciones se ofrece la personalización de cada uno de los ejercicios y *exergames* ya que se subdivide en los mismos módulos. Para hacer estas actividades más fáciles han desarrollado un soporte para el brazo de uso opcional.

Prevención caídas es una de estas secciones, es una rutina de ejercicios validada para personas mayores con riesgo de caídas que aumenta la resistencia, la fuerza muscular y mejora el equilibrio.

Por último, EvolvRehab ha desarrollado un dispositivo *hardware* que incorpora la cámara sensor 3D Azure Kinect, la cual utiliza IA para un rastreo preciso del cuerpo, figura 3.7. Este dispositivo permite una cercanía de hasta 1 metro. Está enfocado a personas con deterioro cognitivo o físico ya que dispone de un botón para encenderse y no necesita ningún tipo de configuración puesto que accede directamente al entorno.

RehaMetrics

RehaMetrics³ es un *software* de ejercicios de rehabilitación virtual orientado a profesionales sanitarios. Cuenta con certificado CE y es usado por más de 200 organizaciones sanitarias con resultados en más de 12.000 pacientes [Reh16].

El sistema cuenta con 220 ejercicios de rehabilitación tanto física como cognitiva, con opción a su personalización. El sistema está pensado para que el profesional pautе los ejercicios en función de las necesidades y desarrollo clínico del paciente y estos pueden completar los ejercicios tanto en el entorno clínico como en el medio doméstico. Una vez realizados el *software* genera un informe clínico, cuantificando los resultados obtenidos, este informe se pone a disposición del profesional para llevar un seguimiento del paciente.

³<https://rehametrics.com/>

Uno de los puntos fuertes del *software* es la incorporación de elementos de gamificación que incrementa la motivación del paciente y su participación en la realización del ejercicio.

Este sistema tiene presente la reducción de costes pues permite aprovechar los recursos disponibles, ordenadores, televisores, *tablets* que ya están presentes en el propio centro y que pueden utilizarse para la instalación del programa. También hace uso de una cámara o dispositivo de vídeo, como se muestra en la figura 3.8.



Figura 3.8: Rehabilitación física de la mano en RehaMetrics. Imagen extraída de: [Reh16]

3.3.2 Aplicaciones sobre rehabilitación física en personas con enfermedades raras

Es complejo encontrar una aplicación creada con el fin de ayudar en las tareas de rehabilitación en niños con el síndrome de Prader-Willi. Si bien es cierto, la mayoría de profesionales que trabajan con estos niños utilizan, a menudo, *exergames* o juegos de visión por computador para mantener la motivación e intentar hacer más llevadera estas actividades. Algunos de estos profesionales utilizan juegos fácilmente accesibles a día de hoy, como *Wii Sport*, un juego recurrente cuando son cuestionados los profesionales del sector.

Para el resto de enfermedades raras, se han ido desarrollando algunos *exergames* dirigidos a algunas de ellas. Debido a la gran cantidad de EERR, normalmente, estos juegos se desarrollan agrupando enfermedades en categorías nosológicas.

A continuación, se exponen diversas aplicaciones centradas en enfermedades raras.

RehApp

RehApp es una plataforma web y móvil de fisioterapia, esta aplicación fue desarrollada por la Universidad de Córdoba, y permite al paciente realizar ejercicios de rehabilitación desde casa sin necesidad de desplazarse a un centro especializado ni contar con la presencia de un profesional.

En ella el usuario puede comprobar la sesión programada por su especialista, estos ejercicios cuentan con una explicación detallada de la actividad a realizar. Para la realización del ejercicio se utiliza la cámara del teléfono móvil o computadora y se procede a la grabación de la ejecución de la actividad desde la aplicación.

Una vez realizada es cuando se extrae información de la grabación, por ejemplo, el ángulo de flexión de la rodilla.

Esta aplicación es utilizada en pacientes con enfermedades raras que afectan el sistema nervioso y muscular.

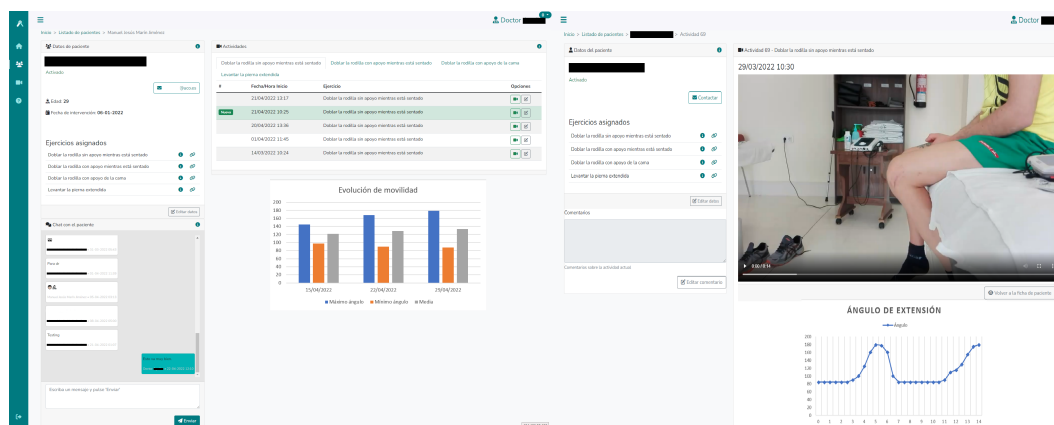


Figura 3.9: Aplicación RehApp en uso. Imagen extraída de: <https://www.uco.es/rehapp/>

MindMotion Pro

MindMotion Pro utiliza la Realidad Virtual como medio de realización de actividades; esta aplicación es usada en pacientes con enfermedades neurológicas raras, como el síndrome de Moebius o el síndrome de Charcot-Marie-Tooth, con el fin de mejorar la coordinación, movilidad y fuerza muscular.

Los pacientes utilizan una serie de sensores corporales para realizar diferentes ejercicios sobre zonas específicas, estos ejercicios son movimientos básicos, como la pulsación de un botón o mover objetos de un sitio a otro. Este tipo de ejercicios permite el desarrollo de planes de tratamiento individualizados que enfatizan en la mejora frente a la adquisición de nuevas habilidades.

Los terapeutas pueden modificar y adaptar diferentes sesiones a sus pacientes; puesto que MindMotion Pro está pensado para personas con enfermedades neurológicas es necesario usarse en compañía de un experto, esta es una de las razones por las que el sistema no recoge información sobre rangos de movilidad hoy en día.



Figura 3.10: Mujer utilizando MindMotion Pro. Imagen extraída de: <https://healthiar.com/mindmotion-pro-for-neuro-rehabilitation>

3.3.3 Comparativa de las aplicaciones y webs analizadas

El objetivo de este proyecto es la creación de un *software* con actividades físicas idóneas enfocadas en el SPW. Para la realización de comparativas entre aplicaciones y páginas webs se han seguido ciertos criterios.

- **Enfoque en EERR.** Dado la baja inversión en EERR, la creación de *exergames* con fines fisioterapéuticos para estas enfermedades es casi nula, por lo que, un desarrollo enfocado en personas con enfermedades raras contribuye a aumentar los beneficios en materia saludable de estas personas. Si la aplicación no se creó bajo este enfoque pero es usada en la actualidad por este tipo de pacientes, o hay estudios que avalan su efectividad en personas con EERR, se valorará de forma positiva.
- **Interfaz accesible.** La mayoría de sistemas *exergames* no fueron desarrollados para que personas con límites motores los utilicen. Es imprescindible contar con un entorno intuitivo y de fácil acceso.
- **Mecanismos de gamificación.** Será positiva la inclusión de mecanismos de gamificación para aumentar la motivación del usuario.

3. ESTADO DEL ARTE

- **Modificación activa de ejercicios.** La aplicación o página web permite, en caso de tratarse de un *exergame*, cambiar características del ejercicio como tiempos de juego o velocidad de realización.
- **Recolección de información.** La aplicación recoge datos en caso de tratarse de ejercicio físico, antes o después de su realización, y, estos datos son almacenado en algún tipo de base de datos o tablas *excel*.
- **Visionado de información.** Posibilidad de poder analizar los datos recogidos tras la realización del ejercicio.
- **No requiere hardware externo.** Dado que estar en posesión de un ordenador y una cámara es algo común, se valorará de forma positiva no tener que realizar desembolsos externos para tener que utilizar un hardware específico como en *Wii Sports*.

	SISTEMA SOPORTE SPW	KINOVEA	WII FIT Y WII SPORT	KINECT SPORTS	REHAPP	MIDMOTION PRO	EVOLVREHAB	REHABMETRICS
ENFOQUE EERR	✓	✗	✓	✗	✓	✓	✗	✗
INTERFAZ ACCESIBLE	✓	✗	✓	✓	✗	✓	✓	✓
MECANISMOS DE GAMIFICACIÓN	✓	✗	✓	✓	✗	✗	✓	✓
MODIFICACIÓN DE EJERCICIOS	✓	✗	✗	✗	✓	✓	✓	✓
RECOLECCIÓN DE INFORMACIÓN	✓	✓	✓	✓	✓	✗	✓	✓
VISIONADO DE INFORMACIÓN	✓	✗	✓	✓	✓	✗	✓	✓
SIN HARDWARE EXTERNO	✓	✓	✗	✗	✓	✗	✗	✓

Figura 3.11: Comparativa de aplicaciones y webs analizadas.

Todas las plataformas o aplicaciones enfocadas a EERR expuestas en la figura 3.11 recopilan información sobre los diferentes juegos de las que se componen con el fin de poder realizar una trazabilidad sobre cualquier ejercicio concreto, requisito fundamental en rehabilitación física en personas con enfermedades raras por lo que uno de los focos apuntará a la recolección de datos. Estas plataformas también prestan atención a incluir mecanismos de gamificación que hagan intuitivas estas interfaces y actividades, por lo que será un objetivo fundamental en este trabajo.

Por otro lado, en plataformas con un enfoque más hospitalario como EvolvRehab o RehabMetrics, muy utilizadas en España, se observa la importancia de poder seguir una trazabilidad de la evolución del paciente por medio de la recolección de datos durante o después de la ejecución del ejercicio; razón por la que la recolección de datos será uno de los puntos claves de este sistema.

3.4 Tecnologías de creación de videojuegos

Puesto que el sistema se basa en la rehabilitación física en niños con el SPW, las dinámicas de juego se encuentran limitadas, con lo cual, se han elegido varias mecánicas, como sistema de puntuación, animaciones y retroalimentación constantemente con objetos visuales. A continuación, se muestra una pequeña recopilación de las tecnologías más relevantes que permiten implementar estas mecánicas.

Unity

Unity⁴ es un motor de desarrollo de videojuegos creado por la empresa Unity Technologies. El término motor de videojuego, hace referencia a un software con una serie de rutinas de programación que permiten el diseño, creación y funcionamiento de un juego.

Está construido sobre una combinación de C# y JavaScript y cuenta con una interfaz de usuario intuitiva que permite a los usuarios diseñar y construir entornos de manera visual. También cuenta con soporte para Realidad Aumentada e Inteligencia Artificial.

Uno de los puntos fuertes de Unity es que engloba herramientas de *networking* para hacer tus juegos multijugador, permite monetizar tu juego y exportarlo a una gran cantidad de plataformas. Además, cuenta con una amplia comunidad de desarrolladores por lo que existe una gran comunidad de soporte.

Pygame

Pygame⁵ es un motor de videojuegos 2D escrito en *Python*, de código abierto, y compatible con los principales sistemas operativos, Windows, MacOS, y las principales distribuciones Linux.

Es sencillo y fácil de usar, desde su lanzamiento, en el año 2000, ha sido desarrollado y mantenido por una comunidad activa de desarrolladores de software libre y ha evolucionado para ser una de las bibliotecas más populares y versátiles para el desarrollo de juegos en *Python*.

Pygame cuenta con un motor de sonido integrado, que permite a los desarrolladores de juegos reproducir y manipular archivos de audio en diferentes formatos. También ofrece soporte para la reproducción de sonidos en segundo plano, lo que permite a los desarrolladores crear efectos de sonido y música que se ejecutan de manera independiente del resto del juego.

PyOpenGL

PyOpenGL⁶ es un *binding* a OpenGL, se necesita mayor conocimiento sobre programación de videojuegos. Se puede integrar con Pygame, ya que este no tiene aceleración gráfica,

⁴<https://unity.com>

⁵<https://www.pygame.org>

⁶<https://pyopengl.sourceforge.net>

para suplir esta necesidad.

PyOpenGL permite a los desarrolladores utilizar las capacidades gráficas de OpenGL y al igual que Pygame, cuenta con motor de sonido integrado.

Es muy antiguo, por lo que no hay gran comunidad que lo utilice y muchos de los archivos de documentación se encuentran obsoletos, a esto hay que añadir que su página web no resulta muy intuitiva, por lo que, trabajar con PyOpenGL puede resultar engorroso.

Panda3D

Panda3D⁷ es un motor 3D de código abierto y uso gratuito para el desarrollo de juegos 3D en tiempo real, visualizaciones y simulaciones. Desarrollado en C++ y Python puede ser utilizado por ambos lenguajes. Fue pensado para desarrollo de juegos comerciales, por ello su motor de juego hace hincapié en potencia, velocidad, integridad y tolerancia a errores.

Esta tolerancia a fallos quizá sea su característica estrella, el sistema no se bloquea si se encuentra un “bug”, por el contrario trata de rastrear y aislar el error.

Panda3D fue desarrollado por *Disney* para su juego multijugador masivo en línea *Toon-town Online*. Se lanzó como software gratuito en 2002.

Unreal engine

Unreal Engine es un motor de juego y código abierto desarrollado en C++ por Epic Games. Es una herramienta de creación de juegos 3D. Hoy en día se utiliza para desarrollar videojuegos, películas animadas y experiencias interactivas o basadas en realidad virtual.

Además, Unreal Engine cuenta con un completo conjunto de herramientas y características integradas que agilizan el flujo de trabajo de desarrollo. Esto incluye un editor visual intuitivo, soporte para animaciones complejas, herramientas de diseño de niveles, edición de materiales y efectos visuales, y un sistema de partículas altamente personalizable. Esto hace que el resultado obtenido con Unreal Engine cuenten con unos gráficos realistas de gran calidad.

3.4.1 Comparativa de las diferentes tecnologías

Los criterios en los que se enmarca esta comparativa son aquellas características que hacen posible el desarrollo del presente TFG.

- **Motor de físicas.** Simulan y calculan la física del juego, son los encargados de aportar realismo a las interacciones usuario-máquina, también se encargan de colisiones, movimiento y fuerzas, entre otros.
- **Motor de sonido.** Gestionan el audio, incluyendo música, efectos de sonido, diálogos y otros elementos. Si se trabaja adecuadamente puede funcionar como un mecanismo

⁷<https://www.panda3d.org>

de gamificación, aportando retroalimentación positiva o negativa dependiendo de la realización de la actividad.

- **Soporte de *tracking* corporal.** La tecnología debe de tener soporte para realizar *tracking* corporal en tiempo real.
- **Realidad Aumentada.** Se debe de permitir superponer elementos en imágenes en tiempo real.
- **Curva de aprendizaje.** Se entiende por curva de aprendizaje rápida cuando conocer y utilizar sus componentes resulta sencillo para los desarrolladores de videojuegos, incluso si no tienen mucha experiencia en programación o diseño de juegos. Una curva de aprendizaje rápida no siempre es traducida como calidad en el resultado final, por lo que, se buscará cierto equilibrio entre curva de aprendizaje y resultado final.

	UNITY	PYGAME	PYOPENGL	PANDA3D	UNREAL ENGINE
MOTOR DE FÍSICAS	✓	✓	✗	✓	✓
INTERFAZ ACCESIBLE	✓	✓	✗	✗	✓
MECANISMOS DE GAMIFICACIÓN	✓	✓	✗	✗	✓
MODIFICACIÓN DE EJERCICIOS	✓	✓	✗	✗	✓
RECOLECCIÓN DE INFORMACIÓN	✗	✓	✗	✗	✗
CURVA DE APRENDIZAJE	Alta	Media	Media	Media	Alta

Figura 3.12: Comparativa de las diferentes tecnologías expuestas siguiendo los criterios establecidos.

Aunque Unity y Unreal Engine se adaptan a los criterios establecidos, la curva de aprendizaje de ambos es bastante más elevada con respecto a Pygame.

Ambos son buenas alternativas para realizar juegos más complejos, con necesidad de mejores gráficos. Unreal Engine está diseñado para desarrollo 3D, por lo que se estaría invirtiendo tiempo y esfuerzo en realizar tareas no prioritarias en el proyecto.

3.5 MediaPipe

Desarrollada por Google y lanzada en 2019 con el fin de realizar soluciones software aplicando visión por computador y aprendizaje automático en tiempo real, MediaPipe⁸ es un

⁸<https://developers.google.com/mediapipe>

framework de trabajo que se centra en la construcción de *pipelines* de procesamiento de datos complejos para la captura, procesamiento, visualización y análisis de datos multimedia.

El pipeline de MediaPipe se compone de una serie de módulos de procesamiento de datos que se encadenan de forma secuencial para procesar los datos multimedia. Cada módulo se encarga de una tarea específica, como la detección de objetos, el seguimiento de la mano o el reconocimiento facial. La modularidad del pipeline proporciona a los desarrolladores la capacidad de crear soluciones de procesamiento de datos multimedia personalizados.

MediaPipe también utiliza técnicas de aprendizaje automático para mejorar el rendimiento y la precisión de los módulos de procesamiento de datos. Los algoritmos de aprendizaje automático se entrenan con conjuntos de datos de alta calidad para mejorar la detección y seguimiento de objetos, la segmentación de imágenes, etc. Esto también se utiliza para mejorar la capacidad de procesamiento en tiempo real y reducir el consumo de energía.+

Funcionamiento del *Framework*

Para dar comienzo al procesamiento de datos MediaPipe se tiene que iniciar el grafo MediaPipe, figura 3.13, el cual controla el flujo de paquetes a través de los diferentes nodos del grafo; una vez iniciado puede procesar un flujo de paquetes hasta que cada flujo se cierre o el grafo se cancele. Una vez su ciclo de vida terminado, este puede destruirse o iniciarse de nuevo.

Los nodos de estos grafos pueden ser de tres tipos, *open*, se llama una sola vez y todos los paquetes de entrada requeridos por el nodo están disponibles; *process*, varias veces ejecutado cuando un nuevo conjunto de *inputs* esta disponible; *closed*, se llama una vez al final para cerrar el grafo.

Los nodos, también llamados *calculators* consumen paquetes, estos nodos consumen $[0, n]$ *inputs* y genera $[0, m]$ *outputs* o flujos de salida.

La política del flujo de paquetes es determinada por *timestamps*, el *framework* garantiza que todos los paquetes tengan el mismo *timestamps*, este tiempo aumenta cada vez que se hace una llamada al nodo tipo *process*

Además, cada *calculator* puede definir un constructor y un destructor, útiles para crear y designar recursos independientes de los datos procesados.

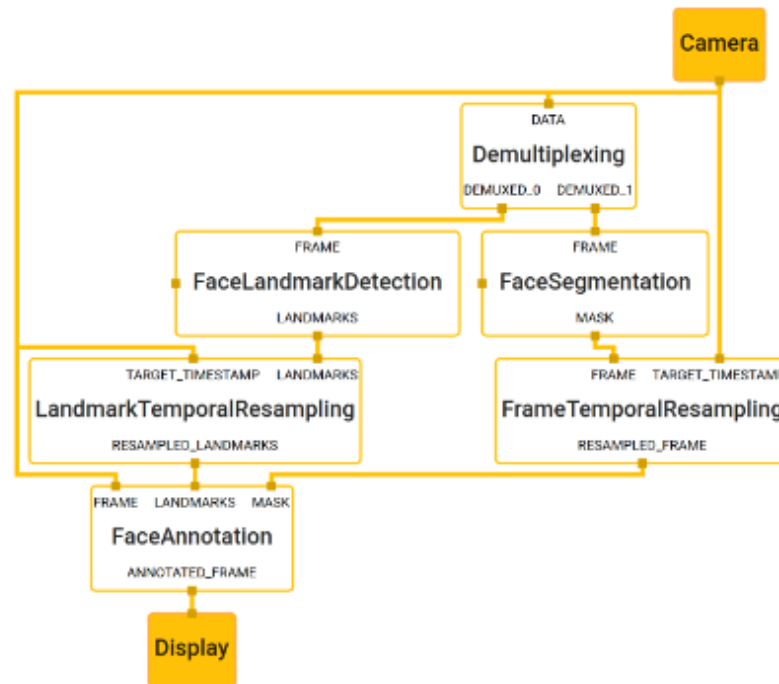


Figura 3.13: Grafo sobre detección de puntos faciales. Los cuadros transparentes representan nodos (*calculators*), los recuadros amarillos representan *inputs* y *outputs* externas al gráfico, y las líneas que entran por arriba y por abajo representan el flujo de paquetes. Imagen extraída de: [LTN⁺19]

Los gráficos de calculadora MediaPipe se utilizan a menudo para procesar flujos de vídeo o audio para aplicaciones interactivas. Normalmente, cada *calculator* se ejecuta en cuanto están disponibles todos sus paquetes de entrada para una determinada marca de tiempo.

El *framework* MediaPipe en Python permite el acceso directo a los componentes principales de MediaPipe C++, como *Timestamp*, *Packet* y *CalculatorGraph*. Se basa en la librería *pybind11*.

Las soluciones Python listas para usar, ocultan los detalles técnicos del *framework* y simplemente devuelven los resultados legibles de la inferencia del modelo a los que llaman.

Las distintas soluciones ofrecidas por MediaPipe varían desde detección facial, detección de objetos y detección de la pose corporal, entre otras. MediaPipe Pose⁹, es la solución encargada del reconocimiento de 33 *landmarks* o puntos de referencia del cuerpo humano a partir de imágenes o vídeos en 3D, estos puntos se pueden observar en la figura 3.14.

⁹<https://developers.google.com/mediapipe/solutions/vision/pose.landmarker/python>

3. ESTADO DEL ARTE

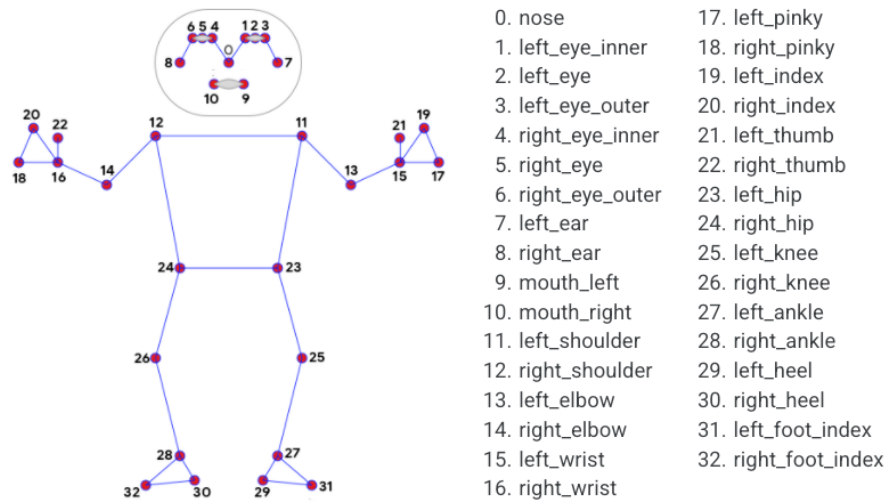


Figura 3.14: Landmarks de MediaPipe Pose.

Metodología

ENTENDEMOS como metodología el conjunto de técnicas y métodos organizativos que seguiremos para conseguir los objetivos marcados [MP15]. Dado que el proyecto ha de permitir la implementación de nuevos módulos y actividades se plantea el uso de un desarrollo iterativo e incremental, lo que nos permitirá abarcar todas las características a implementar en forma de iteraciones.

4.1 Desarrollo iterativo e incremental

Para el desarrollo de este sistema de soporte se ha llevado a cabo un desarrollo iterativo e incremental. Este tipo de desarrollo se basa en realizar diferentes iteraciones en las que el sistema incrementa una nueva funcionalidad y llevar a cabo mejoras de otras que han sido realizadas en iteraciones anteriores [GLB10].

Una de las principales razones por las que se ha elegido este tipo de desarrollo es porque permite realizar proyectos software escalables, requisito fundamental a la hora de desarrollar nuevas actividades. También se centra en la mejora continua, puesto que en cada una de las iteraciones se puede realizar mejoras sobre las funcionalidades desarrolladas en etapas anteriores. Este tipo de desarrollo permite adaptar partes ya desarrolladas a nuevas funcionalidades pertenecientes a la iteración actual, y realizar entregas del producto en cada una de las iteraciones; estas entregas proporcionan *feedback* con el que podremos realizar ciertas adaptaciones en la planificación si fuese necesario hacer cambios en las funcionalidades vigentes. El proyecto no se puede considerar acabado hasta que no se realice la última iteración.

Normalmente se suelen utilizar iteraciones cortas de 2 a 4 semanas, ya que se ha estudiado un aumento de eficiencia en objetivos marcados a corto plazo, lo que se traduce en una necesidad de *feedback* más regular en comparación con metodologías tradicionales. En este proyecto se han utilizado iteraciones de 1, 2 y 3 semanas, visibles en la figura 4.2, puesto que se necesita incluir nuevos módulos o funcionalidades con frecuencia para poder abordar los plazos marcados de entrega. Estos tiempos de iteraciones suelen estar sujetos a ajustes, ya que se acomodan a los equipos de trabajo en cada caso. Resulta obvio mencionar que en desarrollos llevados a cabo por una persona los tiempos suelen variar dependiendo de las capacidades del individuo, por lo que, en iteraciones en las que se realizan módulos o mejoras

en las que se necesite un aprendizaje previo se ha dado un mayor tiempo de iteración.

En la siguiente imagen se muestra el planteamiento seguido en cada una de las iteraciones del proyecto. Primero se desglosan las iteraciones, haciendo subbloques de las funcionalidades a realizar. Una vez desglosadas cada una de las funcionalidades, se realiza una fase de diseño en la que se evalúan los componentes a desarrollar y la lógica a seguir de acuerdo a cada iteración. Por último en la etapa de implementación se construyen las diferentes funcionalidades de la etapa de diseño. Esto permite obtener resultados tangibles de forma rápida y obtener retroalimentación temprana con cada iteración.

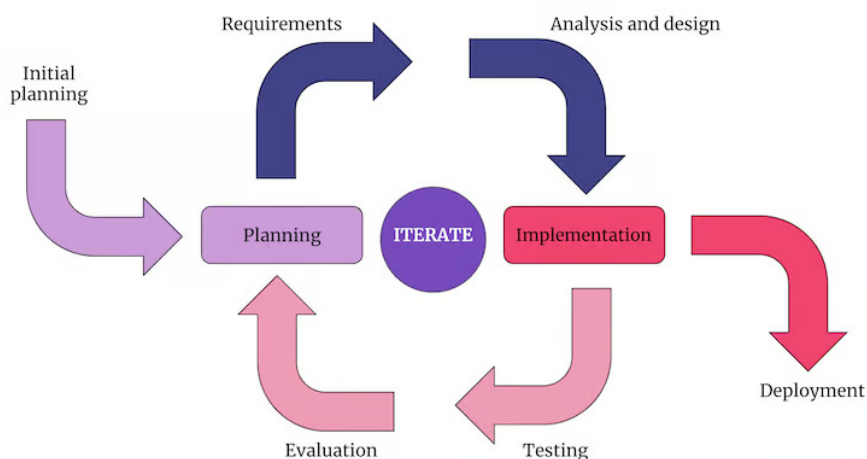


Figura 4.1: Desarrollo iterativo e incremental. Imagen extraída de: [Gad23]

En la siguiente sección se realiza el desglose de iteraciones llevadas a cabo para conseguir los diferentes objetivos marcados.

4.1.1 Planificación de iteraciones

En esta sección se presentan las subdivisiones que forman parte de este proyecto. Cada una de estas divisiones corresponde con una iteración del modelo de desarrollo iterativo e incremental.

- **Iteración 1: Estudio detallado de tecnologías.**

Primera iteración, estudio de los diferentes mecanismos de gamificación, MediaPipe y Pygame. Aparte de estos estudios se realizarán diferentes reuniones con los especialistas para poder sintetizar los aspectos generales de las sesiones de rehabilitación y poder concretar los diferentes *exergames* a desarrollar.

- **Iteración 2: Realización del primer *exergame* “Diagonales Superiores”.**

Exergame que incita el desarrollo de la motricidad gruesa superior y coordinación óculo-manual mediante estímulos en la pantalla. Esta actividad se realizará de forma independiente para conseguir la validación de un primer ejercicio de cara a tener un punto de partida para el correcto desarrollo de las demás iteraciones.

- **Iteración 3: Menú principal y menú de actividades.**

Desarrollo del menú principal, permite el acceso a los diferentes módulos de la aplicación y un menú de actividades, donde se encuentran los diferentes *exergames* y la personalización de estos.

Al inicio de esta iteración se concreta el diseño de la arquitectura del juego. De esta manera se parte de un *exergame* validado, lo que permite que el diseño experimente una menor variación.

- **Iteración 4: Almacenamiento de datos.**

Creación del módulo que permite la recolección de datos para el registro de resultados tras la realización de una actividad. Este módulo también será el encargado de facilitar datos para crear diferentes estadísticas y gráficas.

- **Iteración 5: Módulo historial.**

Módulo que permite la visualización de los registros dependiendo de una actividad específica, diferentes estadísticas sobre esa actividad y también se podrá visualizar en forma de tabla los datos recopilados sobre cada una de ellas.

- **Iteración 6: Realización de la actividad “Squad”.**

Ejercicio en el que se busca mejorar la fuerza muscular así como la coordinación del usuario mediante la realización de sentadillas. Además se añade la modificación de los ajustes de este *exergame* a la escena actividades.

- **Iteración 7: Realización de la actividad “Movilidad”.**

Actividad en la que se buscan mejoras en la movilidad, equilibrio y coordinación visomotriz evitando objetos que aparecen en pantalla. Durante esta iteración también se añaden los ajustes de modificación de esta actividad.

- **Iteración 8: Integración de los registros.**

Realización de elementos estadísticos y de visualización sobre los *exergames* realizados en las iteraciones 2 y 3. Se ha decidido realizar en una iteración aparte para agilizar el proceso de integración de ambos registros y centrarse en la implementación de los *exergames* en las iteraciones separadas.

- **Iteración 9: Módulo tutorial.**

Realización de un módulo “Tutorial” a modo de manual de instrucciones sobre la aplicación y cómo utilizar las diferentes funcionalidades que incorpora.

4. METODOLOGÍA

■ Iteración 10 - Experimentación

Realización de una prueba del sistema en un entorno real. En esta sesión el usuario probará el *software* junto al profesional para ofrecer retroalimentación real sobre usabilidad y cambios a realizar en él.

Tras la realización de estas iteraciones, los objetivos marcados en el capítulo 2 serán completados. Para una mayor comprensión del tiempo empleado en cada iteración se presenta el siguiente diagrama Gantt.

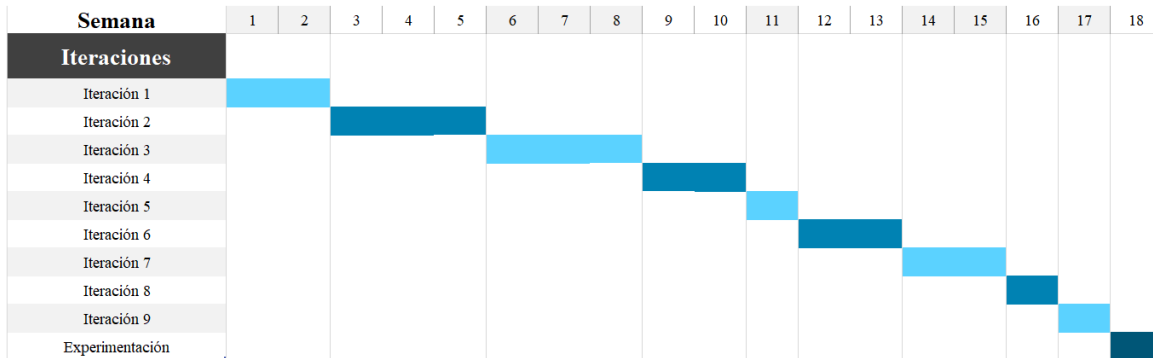


Figura 4.2: Diagrama Gantt semanal por iteración.

Por último, se presenta una tabla iteración/objetivo, cuadro 4.1, que muestra la asociación entre ambos, este identificador será usado en el capítulo 5.

Iteración	Objetivo identificador
Iteración 1	OBJ_1 y OBJ_2
Iteración 2	OBJ_3 y OBJ_2
Iteración 3	OBJ_3 y OBJ_4
Iteración 4	OBJ_3 y OBJ_4
Iteración 5	OBJ_4
Iteración 6	OBJ_3 y OBJ_2
Iteración 7	OBJ_3 y OBJ_2
Iteración 8	OBJ_4
Iteración 9	OBJ_4
Iteración 10	OBJ_5

Cuadro 4.1: Tabla asociación identificador objetivo-iteración.

4.2 Herramientas hardware y software

A continuación se detallarán las herramientas utilizadas durante el desarrollo del proyecto y el uso que se le ha dado a cada una de ellas. Se hará un repaso tanto de herramientas de gestión de proyectos, edición, lenguajes de programación y librerías, entre otros.

4.2.1 Recursos software

Sistemas operativos

Ubuntu 22.04.2 LTS

Sistema operativo principal utilizado durante el desarrollo del proyecto. Se ha elegido Ubuntu gracias a la libertad que ofrece, ya que es un sistema operativo de código abierto, y por la gran variedad de herramientas de desarrollo disponibles. Cabe puntualizar la menor propensión a errores con respecto a otros OS como Windows.

Windows 10

Sistema operativo utilizado para la realización de pruebas debido a que un requisito del proyecto es que este sea multiplataforma. También se ha utilizado este sistema operativo para el uso de herramientas sin versión para distribuciones Linux.

Lenguaje de programación

Python

Python en su versión 3.10.6. Python es un lenguaje de programación de alto nivel interpretado. Apoyado por diferentes librerías y *frameworks*, es el único lenguaje que ha sido necesario para el correcto desarrollo del proyecto.

Herramientas de desarrollo software

Mediapipe

Es un *framework* de Google multiplataforma para construir soluciones basadas en aprendizaje automático. Utilizado para realizar el *tracking* de las diferentes partes del cuerpo.

Pygame

Utilizada para la creación de la interfaz de usuario, creando elementos gráficos personalizados adaptados a los elementos de gamificación. Pygame es también la encargada de soportar todas las mecánicas de juego.

Matplotlib

Matplotlib es una librería para crear visualizaciones estáticas, animadas e interactivas en Python. Ha sido utilizada para la creación de gráficas en los módulos de registro.

4. METODOLOGÍA

Visual Studio Code

Visual Studio Code es un editor de código optimizado con soporte para operaciones de desarrollo como depuración, ejecución de tareas y control de versiones. Contiene herramientas muy útiles para el desarrollo de proyectos como comandos *git* integrados y funciones de *IntelliSense* que ofrecen sugerencias de código inteligentes y completado automático.

Gestión de proyectos

Mendeley

Mendeley es un software de gestión de referencias creado en 2007. Usable tanto en versión de escritorio como extensión web, permite extraer referencias bibliográficas directamente de la web en diferentes formatos.

GitHub

Es un servicio basado en la nube que permite el alojamiento de repositorios para almacenar el código fuente de un proyecto. Permite realizar un control de versiones de la mano de Git.

Lucidchart

Herramienta de diagramación basada en la web. Permite crear diagramas de flujo, organigramas, esquemas web y diseños UML, entre otros. Plataforma utilizada para realizar los principales diagramas de estados y diseños UML.

Herramientas de edición

Overleaf

Overleaf es servicio web para la edición y colaboración de documentos científicos y académicos, especialmente orientada a la escritura de documentos en LaTeX. Plataforma utilizada para crear la documentación del proyecto.

Figma

Herramienta de diseño de interfaces de usuario (UI) y experiencia de usuario (UX) basada en la nube. Se utiliza para crear y colaborar en el diseño de aplicaciones, sitios web y otros proyectos de diseño digital. Se ha utilizado para realizar los bocetos del proyecto.

Canva

Web de herramientas de diseño gráfico y creación de contenido visual simplificado, creada por *Word Inc.* y lanzada en 2012 al mercado. Ha resultado de gran utilidad para la creación de imágenes e iconos, tanto en elementos visuales de la aplicación como en la documentación.

Krita

Software de código abierto para dibujo y pintura digital. Está diseñado principalmente para profesionales del sector artístico. Utilizado como modo de soporte junto a Canva para la creación de elementos visuales.



Figura 4.3: Esquema sobre las herramientas utilizadas.

4.2.2 Recursos hardware

Computador

- **Procesador:** AMD Ryzen 7 4800H 2.90 GHz
- **GPU:** NVIDIA GeForce GTX 1650 Ti
- **RAM:** 16,0 GB DDR4
- **Cámara:** HP TrueVision HD
- **OS:** Windows 10 y Ubuntu 22.04.2

Capítulo 5

Arquitectura

ESTE capítulo detalla la arquitectura del proyecto. Se persigue que el lector conozca cómo se han abordado los aspectos de diseño y desarrollo desde una perspectiva técnica. Así, el capítulo se divide en dos grandes bloques: i) un primer apartado que explica las distintas reuniones realizadas con el fin de extraer diferentes historias de usuario, y ii) un segundo apartado sobre la arquitectura planteada, diseñada tras estas reuniones iniciales y considerando las diferentes historias de usuario elaboradas.

La arquitectura diseñada es modular y escalable. Por cada uno de los módulos existentes en la misma, se detallan particularidades más concretas desde el punto de vista del desarrollo.

5.1 Fundamentos

El punto de partida para el desarrollo de este proyecto fue la entrevista con **Lourdes Nieto Moreno-Cid**, profesional asignada al paciente con el síndrome de Prader-Willi durante el curso 2021/2022.

Tras estas entrevistas extrajeron diferentes historias de usuario para analizar y diseñar la arquitectura del sistema junto a dos *exergames*, ambos, variantes de ejercicios realizados por el paciente durante sus sesiones habituales. Estas historias de usuario se recogen en la sección 5.3, de forma que el lector pueda obtener más información acerca de ellas. Sin embargo, el foco del presente capítulo está en el diseño y desarrollo de la arquitectura, el cual se presenta directamente en la siguiente sección.

La primera actividad debe ser capaz de incitar al usuario a la captura de elementos que aparecen por pantalla. Estos, deben mostrarse por encima de los hombros del usuario, sin promover el movimiento de pies, de modo que contribuya a corregir patrones flexores. Se deben incluir elementos que inciten la motivación del usuario como animaciones y sonidos cuando se realice la actividad correctamente.

Se pueden hacer variaciones de este ejercicio para aumentar la complejidad, lo que se discutirá en ramas futuras de desarrollo.

5. ARQUITECTURA

Para cuantificar el progreso que realiza el usuario durante el uso del sistema se han de recoger distintas mediciones durante la ejecución del ejercicio, tiempo de reacción y capacidad de tocar los elementos que aparecen por pantalla por cada una de las partes del cuerpo.

El segundo ejercicio es la realización de sentadillas en un plazo delimitado de tiempo, se ha de priorizar la realización del ejercicio frente a su correcta resolución, por lo que el ángulo de flexión de la rodilla se debe poder modificar.

Para ambos ejercicios recogidos en esta entrevista el sistema debe permitir la realización de actividades 1 ó 2 días con unas 4 repeticiones por ejercicio a la semana. Esto hace requisito indispensable la usabilidad y obtener un buen rendimiento del sistema.

Una vez terminado el año escolar, Lourdes fue sustituida por Carlos Pérez Poblete. Se realizó una reunión con Carlos que sirvió para comentar lo desarrollado hasta el momento y puntualizar ciertos aspectos del sistema.

Durante esta entrevista salieron a relucir diferentes actividades que se realizan con el paciente en las sesiones de rehabilitación, por lo que se incluye una nueva actividad. En ella, el usuario debe poder esquivar elementos que aparecen en pantalla sin ocultar su cuerpo. Estos elementos deben caer de abajo a arriba de forma que el usuario ha de poder moverse en horizontal.

De cara a siguientes líneas de trabajo se desarrolló otro *exergame*. En él, el usuario tiene que poder superar diferentes alturas controladas por un elemento ascendente en la pantalla. Estos ejercicios pueden contar con variaciones sobre pies juntos o realización sobre una pierna.

Este debe ser un sistema mantenible y escalable, diseñado para poder aumentar y modificar los ejercicios y variantes de estos de forma sencilla.

Estos niños suelen desarrollar problemas de coordinación viso-manual, visomotriz y de equilibrio entre otros, por lo que los *exergames* han de contribuir a al desarrollo de la focalización, precisión y coordinación.

Realizadas las diferentes entrevistas se decidió dar el nombre de **jugador** al paciente y **profesional** al personal encargado de su rehabilitación.

5.2 Visión general de la arquitectura

Este proyecto hace uso de realidad aumentada para interaccionar con el sistema, de modo que utiliza las manos del usuario a modo de controladores. Se ha decidido crear una arquitectura basada en módulos, lo que permite obtener un sistema escalable de cara a incluir nuevos ejercicios. Esta arquitectura es clave para poder realizar un sistema de *tracking* independiente, lo que permite enfocarse en otros aspectos como la lógica y mecánicas de los ejercicios. Estos módulos se utilizan para estructurar y gestionar diferentes partes o etapas

del *software*. Todos los módulos que conforman la arquitectura planteada se muestran, de forma esquemática, en la figura 5.1.

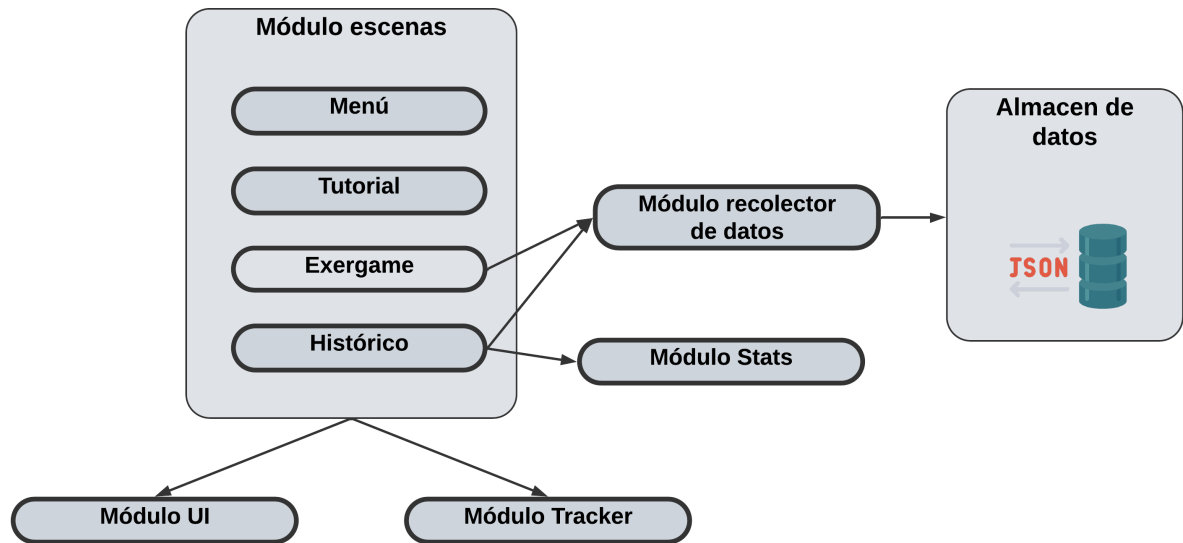


Figura 5.1: Arquitectura del proyecto.

5.2.1 Módulo *tracker*

El desarrollo de este módulo permite obtener la posición corporal en tiempo real y extraer información necesaria sobre esta, para ser utilizada por los diferentes módulos que componen el sistema.

Problema

Es necesario resolver el problema que plantea el *tracking* del control corporal, se debe buscar una solución a controlar la posición del cuerpo del usuario que contribuya a proporcionar una escena fluida. El jugador debe de ser capaz de interactuar con el sistema sin necesidad de componente hardware, permitiendo tanto navegar como interactuar con él.

Esta solución debe permitir la escalabilidad del sistema, facilitando la incorporación del control por gestos en nuevos módulos o *exergames* del sistema. La solución obtenida se debe desarrollar teniendo en cuenta la historia de usuario 5.4.

Solución

Este módulo se crea para poder transformar y almacenar los resultados obtenidos por MediaPipe. En él, se ha creado la clase *ObjectTracker*, la cual se compone de los datos procesados y no procesados obtenidos. Para procesar los diferentes puntos corporales se instancia MediaPipe Pose, cuyos valores por defecto sobre el mínimo de confianza del modelo para que la detección de una nueva pose se considere correcta (*min_detection_confidence*, 0.5) y la confianza mínima para que el seguimiento de la pose se considere correcto (*min_tracking_confidence*, 0.5), se han mantenido por defecto puesto que buscamos un equilibrio entre

5. ARQUITECTURA

latencia y detección. Una vez obtenidos los diferentes *landmarks* se realiza un cambio en su sistema de coordenadas a píxeles, los cuales son almacenados en forma de lista de diccionarios de acuerdo a los identificadores visibles en la figura 3.14. De esta forma se pueden obtener los diferentes puntos corporales sin necesidad de realizar ninguna operación.

Ventajas

La creación del módulo de seguimiento corporal aporta varias ventajas desde el punto de vista de la mantenibilidad y el desarrollo:

- Permite realizar y modificar la captura de *landmarks* de forma independiente al sistema.
- Gracias a crear un módulo común para toda la interacción de usuario se ha permitido agilizar el proceso de desarrollo, facilitando así la escalabilidad del sistema.
- Los parámetros seleccionados para instanciar MediaPipe permiten obtener un seguimiento bastante real con una escena fluida, sin que se aprecien errores en la detección de valores de confianza del modelo.

5.2.2 Módulo de recolección de datos

Este módulo ha sido desarrollado para recopilar los datos obtenidos tras la ejecución de cada una de las actividades. Permite almacenar y recuperar estos registros por otros módulos.

Problema

Para cuantificar el progreso del jugador es necesario almacenar información de las diferentes ejecuciones de actividades que se realicen. Para ello, el sistema debe ser capaz de realizar mediciones acordadas con el profesional durante los *exergames*.

El tipo de fichero a almacenar debe permitir la flexibilidad en la forma de almacenamiento, ha de ser de tipo clave-valor y se ha de poder identificar los datos correspondientes a cada una de las actividades.

Este problema hace referencia a la historia de usuario 5.6.

Solución

Para almacenar los datos se ha elegido el tipo de archivo JSON, debido a la flexibilidad y legibilidad de sus estructuras y compatibilidad de lenguajes, estos archivos han sido almacenados en una carpeta llamada *data*, que cuenta con un fichero por ejercicio.

Los archivos se han nombrado siguiendo un identificador único de cada actividad: 0 para la actividad 'Diagonales Superiores', 1 para 'Squad' y 2 para la actividad 'Movilidad'. Dentro de estos archivos, partimos de dos registros que son almacenados por todas las actividades,

tiempo de duración de la actividad y fecha en la que se realiza.

Para la actividad 0, se han almacenado los diferentes elementos alcanzados, tanto correctos como incorrectos, haciendo distinción entre izquierda y derecha, y tiempo de reacción haciendo distinción entre las diferentes manos y elementos.

Para la actividad 1 o ‘Squad’, se almacenan las sentadillas realizadas correcta e incorrectamente durante el tiempo indicado por el profesional, y una estimación del ángulo alcanzado durante su realización.

La actividad 2 o ‘Movilidad’ recoge los componentes alcanzados por el jugador y las que ha esquivado con éxito.

Estos registros se almacenan en una lista de diccionarios, en la que cada diccionario es una ejecución de la actividad asociada a ese identificador, de manera que los datos se almacenan de modo *campo: valor*. Esto permite utilizar el atributo *campo* como identificador a la hora de realizar operaciones con los datos guardados.

De esta manera, se cumplimentan 2 de los 3 requisitos de admisión de la historia de usuario 5.6 cumpliendo el tercero de estos gracias al módulo de historial.

Ventajas

La implementación realizada en el módulo de recolección de datos ofrece las siguientes ventajas:

- Permite recopilar información sobre cálculos realizados durante la ejecución de diferentes *exergames*.
- Se puede almacenar información sin una base de datos, lo que para archivos pequeños, como los resultantes, permite ahorrarse operaciones de conexión, lectura y escritura.
- Desde el punto de vista de la escalabilidad, esta solución permite la incorporación de ficheros de datos en función de la creación de nuevos *exergames*.
- De cara a la lectura de datos, permite acceder directamente al fichero donde se almacenan los registros de la actividad seleccionada, lo que acelerará el proceso de generación de gráficas y estadísticas.

5.2.3 Módulo stats

Módulo desarrollado para operar con los registros almacenados y generar elementos gráficos y estadísticos sobre los datos.

Problema

Para conocer el progreso del jugador es necesario mostrar información relativa a las ejecuciones de los *exergames*. Esta información puede ser mostrada en forma de tabla o gráfico

5. ARQUITECTURA

y es necesaria la generación de estadísticas por actividad. La información debe ajustarse a un espacio de muestra reducido, por lo que se deberá prestar especial atención a las gráficas y tablas seleccionadas.

Se debe cumplir con los requisitos de aceptación de las historias de usuario 5.6 y 5.7.

Solución

Gracias a la implementación del módulo recolector se han podido extraer datos ya estructurados a los que no se les ha tenido que aplicar ningún procedimiento de transformación. Este es usado por el módulo *stats*, el cual se ha creado para realizar operaciones sobre los datos dependiendo de cada actividad.

Este nuevo módulo genera un objeto dependiente de la actividad sobre la que queremos elaborar estadísticas y gráficas. En el diseño UML de la figura 5.2 podemos ver su composición.

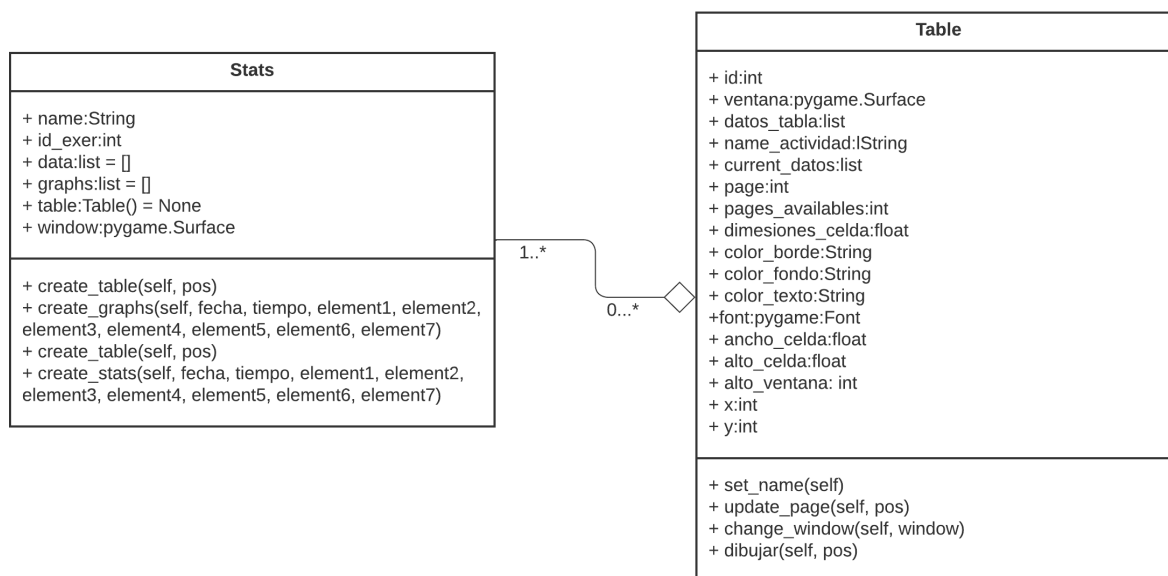


Figura 5.2: Diagrama UML sobre el histórico.

Estas gráficas han sido desarrolladas con la biblioteca *Matplotlib*, elegida por su amplia gama de opciones de visualización y personalización. Las más utilizadas en este proyecto son las llamadas *grouped bar chart*, seleccionadas por el reducido espacio de muestra y la cantidad de información que se puede introducir en ellas. Estas gráficas permiten observar la evolución del usuario en una primera visualización del componente.

Agrupar la información recogida en un mismo día y en un mismo tiempo de ejecución, eje x. Puesto que las actividades están pensadas para ser ejecutadas la misma cantidad veces en un mismo día, esta agrupación sirve como primera aproximación. Por otro lado, se ha utilizado un segundo eje y, para poder mostrar, además de los datos recogidos en forma de gráficos de barras, el tiempo de ejecución de cada ejercicio.

Para la actividad ‘*Squad*’, se ha creado, además del gráfico expuesto con anterioridad, un gráfico de puntos, pensado para mostrar la media de ángulos diaria alcanzada por el usuario.

De cara al muestreo de datos, dado que *pygame* no está diseñado para realizar de interfaces de usuario, se ha desarrollado una manera rudimentaria de dibujar celdas junto con el dato asociado. La tabla resultante ha sido creada con funciones propias de *pygame* en las que se realiza un cálculo sobre dónde debería posicionarse cada celda dentro de la escena.

Al lado de estos elementos gráficos encontramos estadísticas generadas dependiendo de cada *exergames*. Estas estadísticas son operaciones simples sobre los datos almacenados, como los días con mejores y peores ejecuciones en función del tiempo de ejecución junto con los resultados obtenidos esos días, o la media sobre todas las ejecuciones del ángulo flexor alcanzado en la actividad con identificador 1.

Ventajas

La implementación de este módulo ofrece ciertas ventajas:

- Permite la creación de gráficas, estadísticas y tablas de datos de dependiendo de la actividad seleccionada, lo que implica poder realizar un seguimiento real al usuario.
- Permite reutilizar componentes gráficos en todas las actividades, lo que ofrece mayor escalabilidad a la hora de añadir nuevos ejercicios.
- La implementación realizada permite cambiar de actividad sin la necesidad de generar nuevas instancias de una misma clase, lo que aumenta la mantenibilidad del código.

5.2.4 Módulo UI

Módulo en el que se encuentran los principales componentes gráficos usados para crear la interfaz del sistema.

Problema

Pygame no está diseñada para crear interfaces de usuario (UI), por lo que se deberán crear elementos personalizados que permitan navegar por el sistema e interactuar con él. Este módulo ha de poder ayudar a cumplimentar la historia de usuario 5.4.

Solución

Se decide crear un módulo para generar elementos propios de una interfaz usuario, esto permite tener control total de los elementos desarrollados.

En este módulo se encuentran los principales elementos *software* utilizados a lo largo del sistema, *ImageButton*, botón con imagen de fondo; *Button*, botón que cuenta con interacción en la superposición; *BackgroundText*, texto superpuesto en un fondo personalizable y

semitransparente; *DropDown*, desplegables, utilizados para elegir entre múltiples opciones; *InputNumberBox*, *input* de texto numérico.

Todos los componentes creados controlan eventos de manera personalizada.

Ventajas

La creación de un módulo UI implica ciertas ventajas así como inconvenientes:

- Se tiene control total sobre el diseño y funcionamiento de la interfaz, lo que permite adaptar los componentes a las necesidades específicas del sistema.
- Permite una mayor escalabilidad al tener control sobre el módulo de interfaz. Se pueden agregar fácilmente nuevas funcionalidades y características según sea necesario.
- La creación de componentes específicos de interfaz de usuario ofrecen mucha más coherencia visual.
- Existen una serie de inconvenientes a la hora de crear funcionalidades avanzadas ya que las bibliotecas y *frameworks* especializados en UI suelen ofrecer una amplia gama de funcionalidades que no son posibles de desarrollar con los recursos disponibles.

5.2.5 Módulo escenas

Módulo en el que se encuentran las escenas que componen el sistema, estas hacen uso de los módulos expuestos con anterioridad. Contienen los diferentes *exergames*, los cuales están implementados utilizando un enfoque de herencia, donde existe una clase padre que encapsula la funcionalidad común, y cada actividad específica se implementa como una clase hija. Este módulo también se compone por una escena en la que se muestran los componentes realizados por el módulo *stats*.

Problema

Es necesario la creación de componentes lógicos que interaccionen con los diferentes módulos y permitan al usuario la navegación con y por el sistema. Para esto, el usuario debe poder conocer su posición.

De cara a las actividades, se debe recibir instrucciones sobre cómo realizar cada una de ellas y poder modificar ajustes de juego. También se debe controlar los errores cometidos por el jugador durante su ejecución y gestionar los componentes de gamificación desarrollados.

Las actividades necesitan poder realizar tanto acotaciones de superficie, calcular ángulos formados durante la realización de movimientos corporales y poder ayudar a esquivar componentes que aparecen en pantalla.

Se debe mostrar la información recogida en el módulo *stats*, pudiendo alternar entre diferentes actividades.

Por último, se debe poder indicar al usuario cómo interaccionar con el sistema y cuáles son las secciones de este.

Este módulo ayuda a poder completar los criterios de admisión establecidos en las diferentes historias de usuario recogidas en la sección 5.3.

Solución

Las escenas agrupan los elementos lógicos y visuales que componen el sistema. Se ha utilizado este concepto para poder acceder a las diferentes secciones del entorno.

Todas hacen uso del módulo *tracker* para consultar la posición actual del usuario, ya que la principal forma de interacción es a partir de las manos, usadas a modo de controladores.

Escena menú

Se ha creado una escena menú para poder navegar por las diferentes secciones que componen el sistema. Para realizar interacciones, se puede utilizar tanto ratón como el control corporal. Para ello, se ha implementado un sistema de botones con temporizador de superposición a modo de *click* con las manos.

Para interaccionar con las manos, estas son integradas en un *sprite* de tipo *Source*, el cual se actualiza junto con la posición recopilada por el módulo *tracker*. Una vez superponemos nuestras manos a un botón, se puede acceder a la siguiente sección tras el paso de 2.5 segundos.

Colisiones entre elementos

Para poder interaccionar con el sistema se utilizan colisiones entre elementos. En *pygame* tenemos varias maneras de controlar una interacción entre dos elementos, o bien, podemos comparar la posición de ambos en cada *frame* y comprobar si sus posiciones los hacen coincidir, o bien, podemos utilizar funciones *pygame*¹ sobre *sprites* que permiten comprobar de manera automática si dos *sprites* o agrupaciones colisionan.

Hay diferentes tipos de colisiones. Se ha decidido usar las llamadas *groupcollide* y *spritecollideany*. Ambas utilizan la clase *Group()* de *pygame*, que contiene y gestiona múltiples objetos *sprite*.

Por un lado, *groupcollide* recoge dos grupos *sprite* y permite comprobar si hay alguna colisión entre elementos de distinto grupo, si dos elementos colisionan, devuelve la posición y elementos de colisión, por lo que resulta de utilidad a la hora de realizar animaciones. Por otro lado, *spritecollideany* comprueba si un único elemento colisiona con algún elemento de un grupo, devolviendo los mismo parámetros que la función anterior.

Escena actividades

Se ha decidido crear una escena actividades para organizar todos los elementos que tienen

¹<https://www.pygame.org/docs/ref/sprite.html>

relación en la ejecución de *exergames*. En la figura 5.3 se presenta el funcionamiento general de esta escena mediante un diagrama de estados, una representación visual para entender el acceso a las diferentes secciones y *exergames* que la componen. Se han creado diferentes tarjetas que dan acceso a las actividades para dar información sobre a que juego se está accediendo de forma visual. Estas funcionan mediante superposición de manos o *clicks* del ratón.

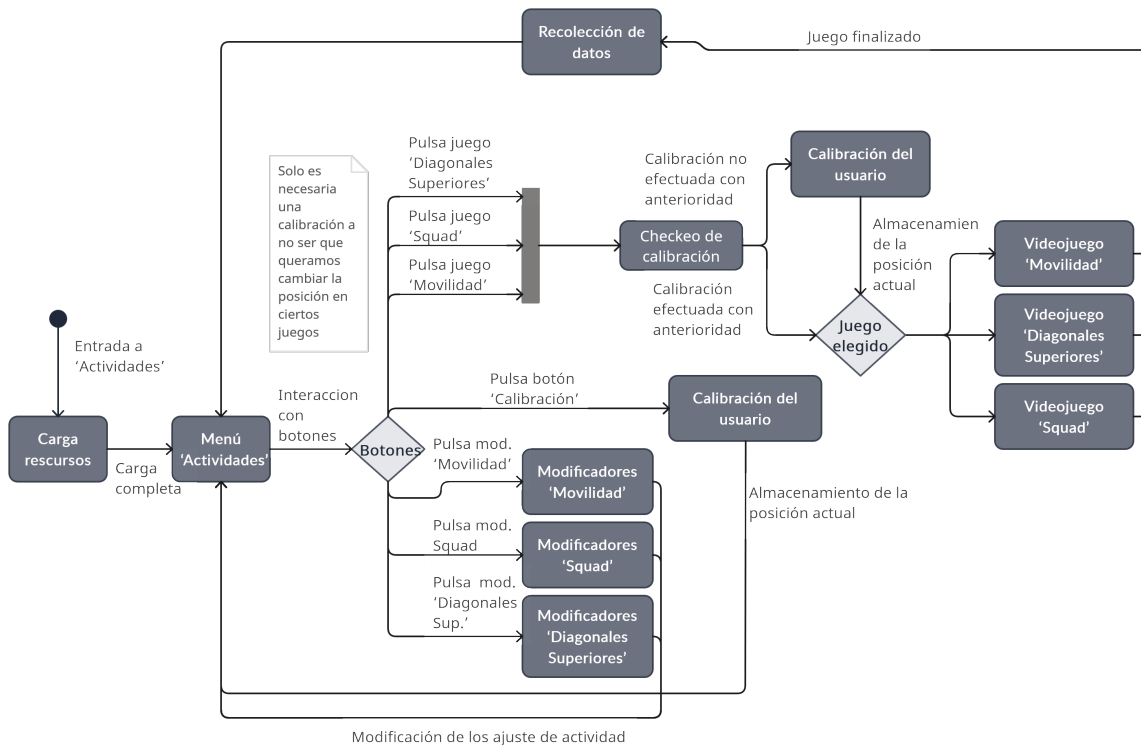


Figura 5.3: Diagrama de estados del módulo 'Actividades'.

Calibración

Se ha creado una escena para obtener la posición actual del usuario. Esta posición es usada para dar *feedback* sobre cómo debe posicionarse en la pantalla y crear partes del cuerpo inamovibles, lo que permite incluir elementos estáticos, si fuese necesario, en las diferentes actividades.

Modificadores de configuración

Para poder adaptar los ejercicios al usuario se han incluido una serie de modificadores dependientes de cada *exergame*. Estos modificadores cumplen los criterios de aceptación de la historia de usuario 5.8. Para su implementación, se han creado archivos de configuración en función de la actividad correspondiente, carpeta *config*, estos archivos han sido llamados *Exer_ID_config.json*, como se puede ver, son archivos JSON donde ID es el identificador único de actividad. Cada archivo está compuesto por un diccionario de elementos configurables *clave: valor*.

Para poder realizar esta configuración, se han incluido botones debajo de cada una de las actividades. Estos pueden ser utilizados por el ratón, ya que están orientados a la modificación por parte del profesional. Cada modificador contiene diferentes *InputNumberBox* para introducir diferentes ajustes de juego. Para poder almacenar correctamente esta configuración se comprueba la validez de los datos introducidos, esto es, si un modificador tiempo de ejecución es menor que un modificador de aparición de elementos, ambos se consideran no válidos y no se aplica configuración alguna.

Exergames

En la figura 5.3 podemos ver el proceso antes de entrar en un *exergame*.

Al inicio de todas las actividades se muestra una pequeña recreación sobre cómo realizar el ejercicio en forma de animación, lo que hace admisible la historia de usuario 5.5. Tras esto, se realiza una cuenta regresiva y se inicia la actividad. Para realizar este contador se ha creado una escena secundaria llamada *TimeDown*. Durante este inicio se recogen los datos de configuración instanciados en el archivo JSON *Exer_ID_config* y se preparan los diferentes recursos utilizados por el *exergame*.

Para que cada una de las actividades se centre en la lógica que implementan, se ha creado una clase *Exergame* que contiene todos los elementos comunes y de inicio de los ejercicios.

Estos elementos comunes son tiempo de ejecución, puntuación del usuario, botón para salir de la actividad y diferentes mensajes de error.

Para la actividad 0, '**Diagonales Superiores**' se ha usado un componente llamado *Sticker*, el cual incluye una *flag* sobre si el elemento es correcto o incorrecto. Para posicionar el elemento en la pantalla se crea un área de aparición de elementos, de manera que, con los puntos recogidos durante la calibración, se calcula la posición horizontal máxima alcanzable por mano, eje x, y la posición vertical máxima alcanzable. Con estas posiciones se instancia de forma aleatoria la posición del elemento a alcanzar, teniendo en cuenta los márgenes superiores en caso de que la posición vertical máxima los sobrepase. Una vez creada esta superficie, se posiciona aleatoriamente un elemento en pantalla. Con esto, se da por completada la historia de usuario 5.9.

En esta actividad se muestran *stickers* en las manos del usuario para indicar la superficie de colisión con el componente; si este impacta sobre otro, el sistema hace uso del motor de sonido junto con una animación indicando si ha sido correcta o incorrectamente realizada.

Para incluir estas animaciones se ha decidido crear una clase *Animation*, en la que es posible incorporar las llamadas *frame-based animations*, animaciones formadas a partir del visionado secuencial de imágenes. Cada vez que un elemento es alcanzado se crea una instancia de esta clase, la cual se muestra en la posición en la que se ha producido el impacto,

tupla (x,y).

Una de las dificultades para el usuario es la no movilidad de pies, los cuales tienen que permanecer estáticos en todo momento. Para esto, se utiliza la pose almacenada en la escena calibración, con esta pose se crea un rectángulo de posicionamiento en el que el sistema comprueba si ambos pies se encuentran dentro. Si alguno de estos saliese del rectángulo, el sistema detiene su ejecución para informar del error hasta que vuelvan a posicionarse dentro del mismo.

Se ha asociado una puntuación positiva cuando se acierta en la colisión de un elemento correcto; por consiguiente, cuando se colisiona con un elemento erróneo la puntuación total disminuye.

De cara a la siguiente actividad, realizar una sentadilla, es necesario calcular el ángulo flexor de la rodilla. Se ha tomado como referencia la parte derecha del cuerpo y se calcula el ángulo formado por tres puntos en 2D, cadera, rodilla y pie. Primero se computa la longitud formada por cada uno de los lados del triángulo y una vez obtenida, se el ángulo formado por la rodilla. Esto asegura la admisión de la historia de usuario 5.11.

Cuando el usuario realiza un ángulo igual o inferior al determinado en la configuración por el profesional, el sistema ofrece *feedback* sobre la correcta realización del ejercicio.

Con respecto a la puntuación, el usuario únicamente recibe puntuación positiva cuando se realiza correctamente una *squad*.

Durante la realización del ejercicio el sistema almacena el ángulo obtenido en todas y cada una de las realizaciones para obtener la flexión media de la rodilla alcanzada en esa ejecución de la actividad.

Por último, para la actividad '**Movilidad**' se desarrolla un nuevo componente llamado *Moving_Sprite*. Este cuenta con una imagen asociada, capaz de actualizar su posición en la pantalla en función de *frames*, esta velocidad de posicionamiento es modificable por el profesional. Las posiciones de estos elementos inician con el eje y a 0, por lo que todos se movilizan de arriba hacia abajo. La posición de aparición es aleatoria dentro de las dimensiones de la pantalla en el eje x. La generación de estos componentes también es aleatoria y el porcentaje de probabilidad de aparición es modificable por el profesional.

Estos elementos se agrupan en tipo *Group()* de *pygame*, por lo que, en cada *frame* todos los componentes dentro de este grupo actualizan su posición. Por otro lado, todos los pertenecientes al cuerpo se incluyen en un *Group()* *body* que se actualiza por cada detección de una nueva pose, por consiguiente, la realización de colisiones se efectúa de la misma manera que en el resto de soluciones, pudiendo así mostrar animaciones y reproducir sonidos cuando un elemento impacta sobre el cuerpo. Esto, cumplimenta la historia de usuario 5.12.

Control de sonido en los diferentes *exergames*

Se ha decidido incluir un total de 6 canciones diferentes para reproducir durante la ejecución de todos los *exergames* para motivar al usuario. Estas son elegidas de manera aleatoria durante la carga de cada uno de los ejercicios y se reproducen durante toda su ejecución.

Por otro lado, se han incluido diferentes sonidos de acierto y fallo en caso de realización correcta o incorrecta del ejercicio o las dinámicas que lo componen.

Para instanciar estos componentes se utiliza el módulo *pygame.mixer*, capaz de cargar y reproducir sonidos. Además, se compone de 8 diferentes canales con los que se puede reproducir sonidos en paralelo y modificar el volumen de cada uno de ellos.

Escena historial

Se ha creado esta escena, la cual hace uso del módulo *stats*, para permitir la visualización de los registros almacenados dependiendo de la actividad que se solicite.

Incluye dos secciones, accesibles mediante un botón, sección gráfica, muestra gráficos y estadísticas, y sección datos, muestra los registros en forma de tabla.

Escena tutorial

Se ha decidido desarrollar una escena tutorial para mostrar el funcionamiento del sistema. Funciona en base a fotos, en las que se explica, de forma sencilla, los componentes del sistema. En esta escena, las diferentes imágenes se almacenan en listas, cada elemento de la lista es una imagen, pudiendo acceder a las mismas con botones distribuidos a ambos lados de cada una.

Ventajas

- Las creación de escenas ha ayudado a dividir el sistema secciones, lo que facilita la gestión del flujo del juego y mejora la legibilidad del código.
- Permite la reusabilidad gracias a la reutilización de las mismas escenas en diferentes partes del sistema.
- Permite la modificación de parámetros escogidos por el profesional, lo que se traduce en poder ajustar el sistema al usuario. Promoviendo así la motivación del jugador ya que las actividades siempre serán un reto para él.
- Gracias al *feedback* visual y auditivo ofrecido cuando el jugador ejecuta alguna de las actividades, puede entender intuitivamente la realización correcta o incorrecta del ejercicio.
- Los diferentes elementos de gamificación se adaptan a cada uno de los juegos, lo que permite incluirlo en nuevas actividades sin tener que crear diseños personalizados para cada una.

5. ARQUITECTURA

- Los mensajes de error proporcionan retroalimentación cada vez que se realiza una posición incorrecta, lo que permite corregirla durante la ejecución del ejercicio.
- La aleatoriedad en la generación de componentes ayuda al dinamismo del juego, ya puede volver a realizarse sin conocer donde aparecerá el siguiente elemento a interactuar.
- La creación de la clase *Exergame* permite centrarse en la implementación de la lógica de cada uno de los *exergames*.

5.3 Historias de usuario

Se decidieron crear historias de usuario ya que permiten enfocarse en el valor que se brinda al usuario final. Cada historia de usuario describe un objetivo o resultado deseado, lo que ayuda a priorizar las características y funcionalidades más importantes. Cada una de estas historias cuenta con un identificador, rol (dependiendo si se trata de jugador o profesional), funcionalidad, valor y distintos criterios de aceptación. A este capítulo se ha hecho referencia en los diferentes módulos, los cuales han permitido hacer admisibles los criterios de aceptación de estas historias.

IDENTIFICADOR	HU_01
COMO	Jugador y Profesional
FUNCIONALIDAD	Quiero: poder recibir feedback visual sobre el posicionamiento
VALOR	Para: poder acceder a los diferentes módulos
ADMISIÓN	Cuando se acceda se debe poder retroalimentar al usuario con la posición
	Se ha de poder cambiar de módulo sin perder el posicionamiento del usuario

Figura 5.4: Historia de usuario. ID: HU_01.

IDENTIFICADOR	HU 02
COMO	Jugador
FUNCIONALIDAD	Quiero: poder recibir instrucciones claras sobre la actividad
VALOR	Para: poder realizar correctamente el <i>exergame</i>
ADMISIÓN	El jugador ha de poder realizar la actividad valiéndose de la información mostrada al inicio
	El jugador ha de poder encontrar fácilmente instrucciones sobre como se realiza la actividad.
	En caso del no entendimiento de las instrucciones por parte del jugador han de ser lo bastante claras para el profesional

Figura 5.5: Historia de usuario. ID: HU_02.

IDENTIFICADOR	HU 03
COMO	Profesional
FUNCIONALIDAD	Quiero: poder visualizar los datos recogidos
VALOR	Para: poder observar las mediciones realizadas durante la actividad
ADMISIÓN	Se ha de poder realizar las distintas mediciones acordadas con el profesional.
	Se han de poder almacenar estas mediciones una vez realizado un <i>exergame</i>
	Se han de poder mostrar estos datos en forma de tabla fácilmente accesible

Figura 5.6: Historia de usuario. ID: HU_03.

IDENTIFICADOR	HU 04
COMO	Profesional
FUNCIONALIDAD	Quiero: poder visualizar los datos de manera gráfica
VALOR	Para: poder obtener una visión intuitiva del progreso del jugador
ADMISIÓN	Cuando se acceda al módulo se deben visualizar en forma de gráfica el progreso del jugador
	Para cada actividad se ha de poder visualizar sus gráficas disponibles teniendo como principal objetivo el tiempo de respuesta del sistema

Figura 5.7: Historia de usuario. ID: HU_04.

5. ARQUITECTURA

IDENTIFICADOR	HU_05
COMO	Profesional
FUNCIONALIDAD	Quiero: poder modificar ajustes de los <i>exergames</i>
VALOR	Para: poder ajustar las actividades al progreso del usuario
ADMISIÓN	Se ha de poder cambiar los diferentes tiempos de juego de cada una de las actividades
	Se ha de poder cambiar ajustes propios de cada uno de los <i>exergames</i>
	En caso de discrepancia en los ajustes de juego se mantendrán los últimos ajustes válidos

Figura 5.8: Historia de usuario. ID: HU_05.

IDENTIFICADOR	HU_06
COMO	Jugador
FUNCIONALIDAD	Quiero: poder recibir feedback visual en caso de errores durante el <i>exergame</i>
VALOR	Para: poder terminar de realizar la actividad
ADMISIÓN	Se ha de poder mostrar al jugador si alguna parte de su cuerpo necesaria para una actividad no se encuentra en pantalla
	Se ha de poder mostrar mensajes al jugador si sale del espacio asignado para las diferentes partes del cuerpo en determinados juegos

Figura 5.9: Historia de usuario. ID: HU_06.

IDENTIFICADOR	HU_07
COMO	Jugador
FUNCIONALIDAD	Quiero: que el <i>exergame</i> ‘Diagonales Superiores’ los elementos estén acotados al rango de movimiento
VALOR	Para: poder realizar el movimiento corporal adecuadamente
ADMISIÓN	Se ha de poder controlar el posicionamiento de elementos en la pantalla
	Se ha de poder crear una superficie de aparición en la que los elementos puedan posicionarse

Figura 5.10: Historia de usuario. ID: HU_07.

IDENTIFICADOR	HU_08
COMO	Jugador
FUNCIONALIDAD	Quiero: que el <i>exergame</i> ‘Squad’ sea capaz de controlar el ángulo formado por la rodilla
VALOR	Para: controlar que el movimiento se realiza adecuadamente
ADMISIÓN	Se ha de poder controlar la posición de los elementos clave de los miembros inferiores
	Se ha de poder realizar una estimación del ángulo formado por la pierna en su momento de flexión

Figura 5.11: Historia de usuario. ID: HU_08.

IDENTIFICADOR	HU_09
COMO	Jugador
FUNCIONALIDAD	Quiero: que el <i>exergame</i> ‘Movilidad’ permita el movimiento controlado
VALOR	Para: que el usuario pueda realizar movimientos de coordinación visomotriz
ADMISIÓN	Se ha de poder permitir la movilidad por la pantalla al jugador
	El sistema ha de poder controlar colisiones cuerpo-elemento

Figura 5.12: Historia de usuario. ID: HU_09.

Capítulo 6

Resultados

ESTE capítulo se centra en los resultados finales obtenidos en cada una de las iteraciones, considerando especialmente aspectos visuales y de diseño de la interfaz de usuario, construida a partir de la arquitectura expuesta en el capítulo anterior. Asimismo, se abordan cuestiones relativas a la sesión de experimentación realizada en un entorno de rehabilitación real.

El código fuente se encuentra disponible para su consulta y evaluación en el siguiente repositorio GitHub:

<https://github.com/MariaBlancoGonzalez/PraderWilli-Rehab>

Se ha creado un vídeo sobre el funcionamiento, tanto *exergames* e historial, del sistema. Este vídeo se puede encontrar en el siguiente enlace:

<https://drive.google.com/file/d/1zCbBfCHxu9ake1YcFnMX-gFZ7e1qdTmM/view?usp=sharing>

6.1 Aplicación gráfica para fisioterapia en niños con el Síndrome de Prader-Willi

En esta sección se expondrán los resultados obtenidos durante la etapa de desarrollo. Para la parte visual del sistema se han creado bocetos gracias a la herramienta *Figma*, que serán expuestos durante el avance de la sección, siendo estos intercalados con imágenes reales del sistema.

Dado que la iteración 1 se enfoca en adquirir conocimientos previos al desarrollo del sistema, esta sección inicia en la iteración 2, donde se encuentran los primeros resultados sobre este desarrollo.

6.1.1 Iteración 2 - ‘Diagonales Superiores’

Primer *exergame* desarrollado, se puede ver su boceto en la figura 6.1. En ella el usuario utiliza sus manos para alcanzar elementos posicionados por encima de sus hombros. Se busca un diseño sencillo e intuitivo que permita al usuario focalizar su atención en los elementos que aparecen por pantalla. Este *exergame* fue desarrollado de forma independiente para conseguir una validación inicial de la actividad. De esta forma, se fueron realizando diferentes modificaciones para obtener el primer diseño válido y así, poder sentar unas bases en las que se apoyen las siguientes iteraciones. El diseño se compone de una barra superior compartida por todos los *exergames*. Esta, contiene el tiempo restante de ejecución y la puntuación alcanzada por el usuario.



Figura 6.1: Boceto sobre la actividad Diagonales Superiores.

6.1.2 Iteración 3 - Menú principal y menú de actividades

El boceto del menú principal se puede encontrar en la figura 6.2, este diseño busca ser intuitivo al usuario, utiliza la cámara a modo de *background* para dar sensación de estar en un entorno real, lo que permite al usuario entender el funcionamiento del menú. Desde aquí se puede acceder a las diferentes escenas que compone el sistema.

La escena actividades, en ella se pueden acceder a los diferentes *exergames*. También permite acceder a los elementos de configuración de cada uno de ellos. Desde esta, se puede calibrar el sistema de nuevo si así se desea (ver figura 6.3). Se busca un diseño intuitivo para el usuario y de fácil acceso a las actividades.

Una vez accedemos a un modificador, este se compone de diferentes elementos con los



Figura 6.2: Boceto sobre el menú principal.

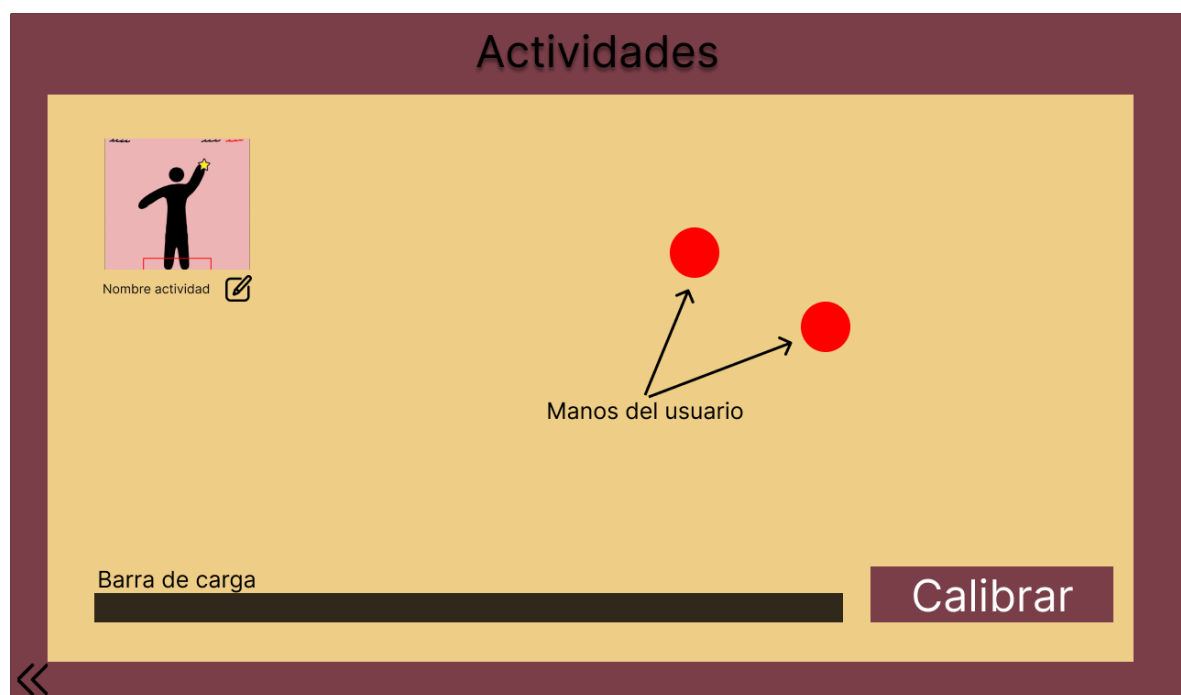


Figura 6.3: Primer boceto sobre el menú actividades.

que podemos interaccionar para cambiar la configuración a un *exergame* (ver figura 6.4). En el modificador de la actividad 'Diagonales Superiores', se puede configurar el tiempo de ejecución, la velocidad de aparición de los elementos, y el porcentaje de elementos trampa que pueden aparecer durante la ejecución.

6. RESULTADOS

Modificadores.

Tiempo de juego (segundos)	Tiempo en el que los elementos aparecen	Porcentaje de trampa (0, 1)
<input type="text"/>	<input type="text"/>	<input type="text"/>

Figura 6.4: Modificador de la actividad diagonales.

6.1.3 Iteración 5 - Histórico

El primer boceto sobre la escena histórico se puede ver en la figura 6.5. A continuación, se muestran las diferentes gráficas desarrolladas durante esta iteración para la primera actividad.

Historial

usuarios actividad

Gráficas

Estadística

Figura 6.5: Primer boceto sobre el módulo historial.

Sobre el *exergame* ‘Diagonales Superiores’ se han creado dos *grouped bar chart* separadas por manos, en ellas, se muestran los aciertos y errores de cada ejecución y el tiempo empleado en cada una de ellas.

6.1.4 Iteración 6 - ‘Squad’

Con respecto al segundo *exergame*, se ha incluido su boceto en la figura 6.7. En esta actividad el usuario realiza una sentadilla en un tiempo delimitado por el profesional. Se busca un diseño motivador, en el que el jugador sienta la realización de *squads* como parte de un reto. Este es un diseño simple, privando al usuario de elementos que puedan generar distracciones.

El modificador de la actividad ‘Squad’ permite ajustar el tiempo de ejecución, límite máximo de tiempo para ejecutar una sentadilla y ángulo que el jugador debería alcanzar para que una sentadilla se considere correcta.

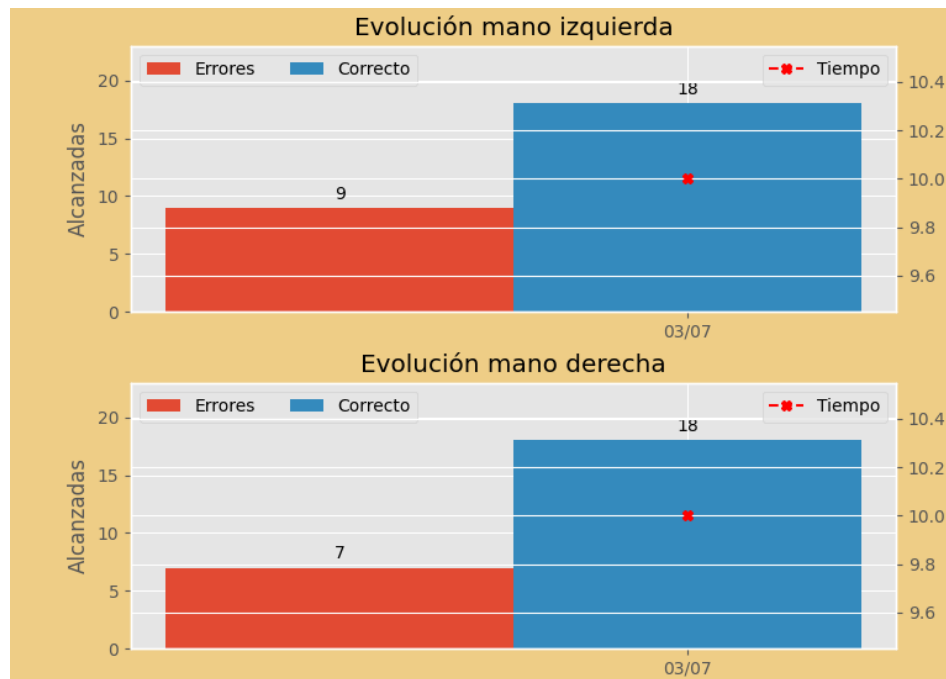


Figura 6.6: Gráfica sobre el *exergame* Diagonales Superiores.

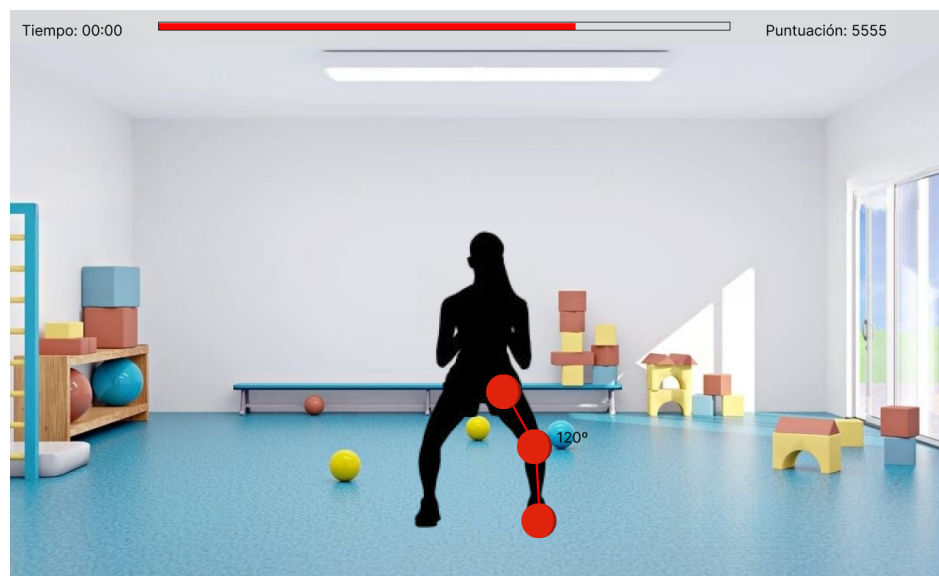


Figura 6.7: Boceto sobre la actividad Squad.

6.1.5 Iteración 7 - 'Movilidad'

Este es el último *exergame* desarrollado. En él, la función del usuario es esquivar los elementos que caen de arriba hacia abajo sin salirse de las dimensiones de la pantalla. Se busca un diseño atractivo hacia el jugador que motive la realización de la actividad sin incitar a realizar poses corporales anómalas que puedan dañar al jugador.

El modificador de la actividad 'Movilidad' permite ajustar el tiempo de ejecución, velocidad con la que caen las pelotas y el porcentaje de pelotas que pueden caer (ver figura 6.10).

6. RESULTADOS

Modificadores.		
Tiempo de juego (segundos)	Tiempo entre sentadillas	Ángulo (1, 360)
<input type="text"/>	<input type="text"/>	<input type="text"/>

Figura 6.8: Modificador de la actividad Squad.



Figura 6.9: Boceto sobre la actividad Movilidad.

Modificadores.		
Tiempo de juego (segundos)	Velocidad de las pelotas	Probabilidad bolas (1, 100)
<input type="text"/>	<input type="text"/>	<input type="text"/>

Figura 6.10: Modificador de la actividad Movilidad.

6.1.6 Iteración 8 - Integración de los registros

En esta iteración se implementan diferentes estadísticas y gráficas en el módulo historial sobre las actividades desarrolladas en las iteraciones 6 y 7.

Sobre la actividad ‘Squad’ encontramos dos gráficas diferentes. Mostrada en la parte superior de la figura 6.11, encontramos la gráfica de barras desarrollada en la iteración 5, la cual sigue la misma lógica que en su uso anterior. En la parte inferior, se muestran gráficamente los diferentes ángulos alcanzados en cada una de las ejecuciones.

En la gráfica sobre ‘Movilidad’, se muestran los datos recopilados durante la ejecución de este *exergame*, elementos impactados, esquivados y totales de las ejecuciones, además se

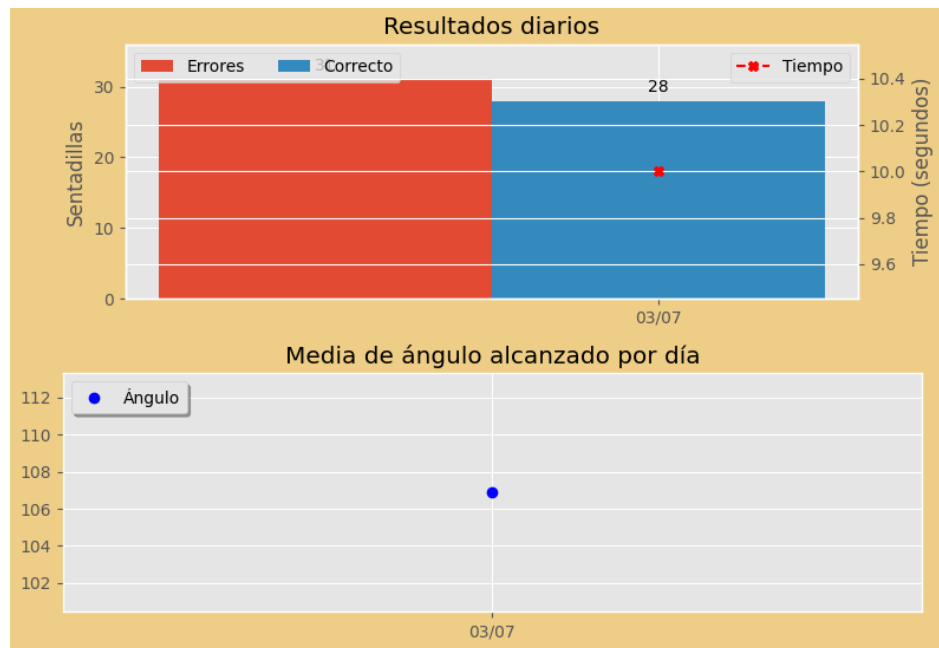


Figura 6.11: Gráfica sobre el *exergame* Squad.

muestra el tiempo empleado en cada ejecución.

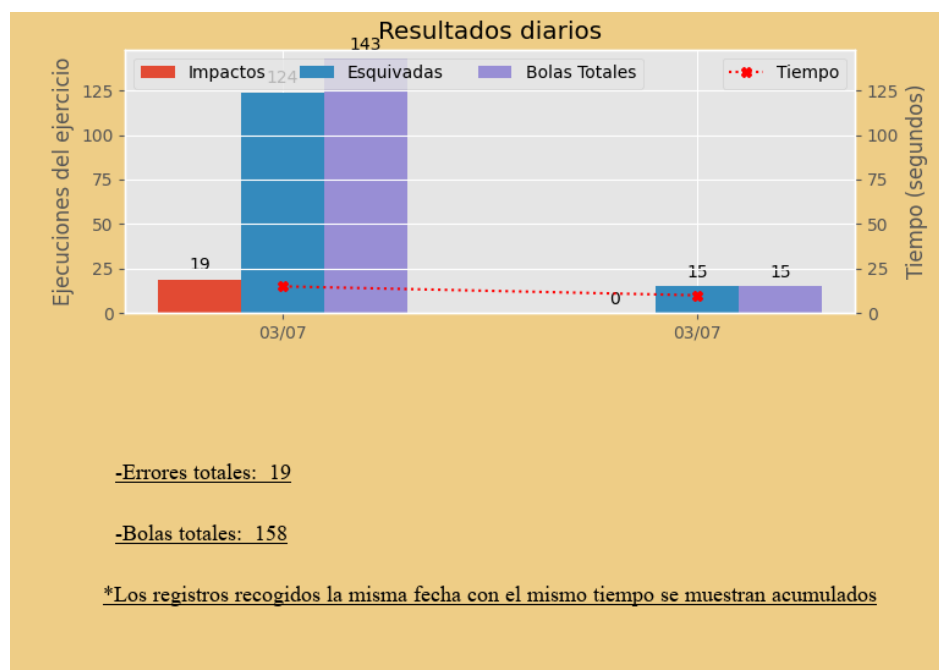


Figura 6.12: Gráfica sobre el *exergame* Movilidad.

6. RESULTADOS

Las estadísticas generadas, mostradas en la figura 6.13, se componen de diferentes valores computados gracias a las ejecuciones de los *exergames* 2 y 3. Estos valores recogen la peor y mejor ejecución de la actividad, y distintas medidas que varían en función del *exergame*.

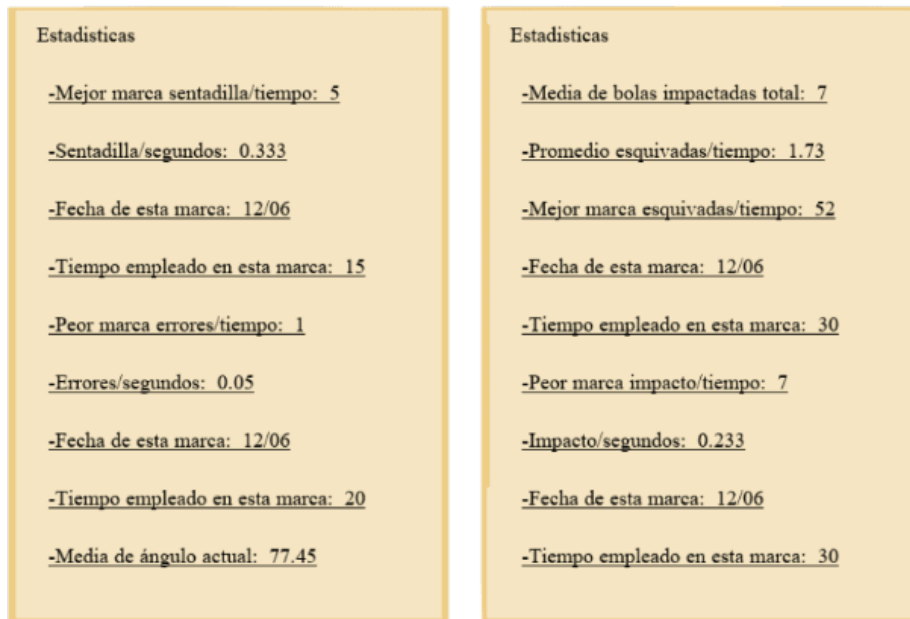


Figura 6.13: Estadísticas sobre Squad y Movilidad, izquierda y derecha respectivamente.

6.1.7 Iteración 9 - Tutorial

En el tutorial, mostrado en la figura 6.14, encontramos diferentes imágenes ordenadas según la escena a la que pertenecen. Para pasar estas imágenes se utilizan los botones flecha, y botones superiores para cambiar de imagen.

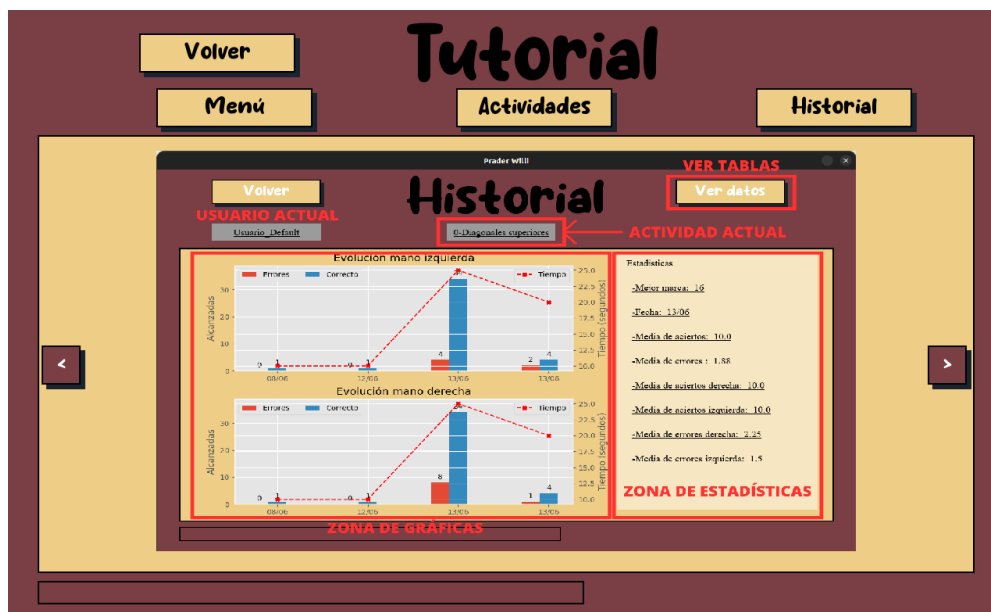


Figura 6.14: Tutorial sobre la escena del histórico.

6.1.8 Iteración 10 - Experimentación en un entorno real

Gracias a los profesionales Carlos y Lourdes se pudo probar el sistema en un entorno real. En esta prueba se contó con Manuel, un niño diagnosticado con el síndrome de Prader-Willi. Las sesiones de rehabilitación llevadas a cabo por Carlos tienen una duración de 40 minutos, tiempo en el que se realizaron dos ejecuciones de cada uno de los *exergames*.

Primero, se expuso el sistema y sus componentes al profesional, el cual expresó la facilidad con la que se puede acceder a las diferentes actividades. Por otro lado, puntualizó la idea de poder añadir distintos filtros a la zona de gráficas con el objetivo de poder llevar un mayor seguimiento del usuario. Las ideas que manifestó desembocaban en poder focalizar la información recogida en un rango de días, tanto estadísticas como gráficas.

En los siguientes minutos, se comentó las actividades que debería realizar el jugador. Posteriormente, se realizaron los diferentes *exergames* en el orden: ‘Diagonales Superiores’, ‘Squad’ y ‘Movilidad’.

Durante el desarrollo de la primera actividad, el jugador tuvo dificultades en el entendimiento de la aparición de los elementos en la pantalla y como resultado observaba en dirección hacia donde se deberían posicionar los componentes en la vida real. Gracias a las explicaciones durante la ejecución del ejercicio se consiguió que el jugador realizase la actividad correctamente. Una vez entendido el funcionamiento del sistema, la siguiente problemática fue la coordinación viso-manual, pues el movimiento de brazos en dirección a los elementos no se efectuaba con precisión en un principio. La ejecución de esta actividad se puede ver en la figura 6.15.



Figura 6.15: Realización de ‘Diagonales Superiores’ por parte de Manuel. En la imagen, el niño se moviliza en la dirección del elemento pero no lo alcanza.

6. RESULTADOS

Para la ejecución de la actividad ‘Squad’, el proceso de entendimiento fue más simple. Según el profesional, el jugador acató con más entusiasmo el ejercicio en comparación con el desarrollo habitual. Durante el proceso salió a relucir la falta de elementos visuales que incitasen la realización de la sentadilla. Por último, el profesional sugirió poder estudiar el tiempo empleado en de cada una de ellas. En la figura 6.16 se puede apreciar su ejecución.

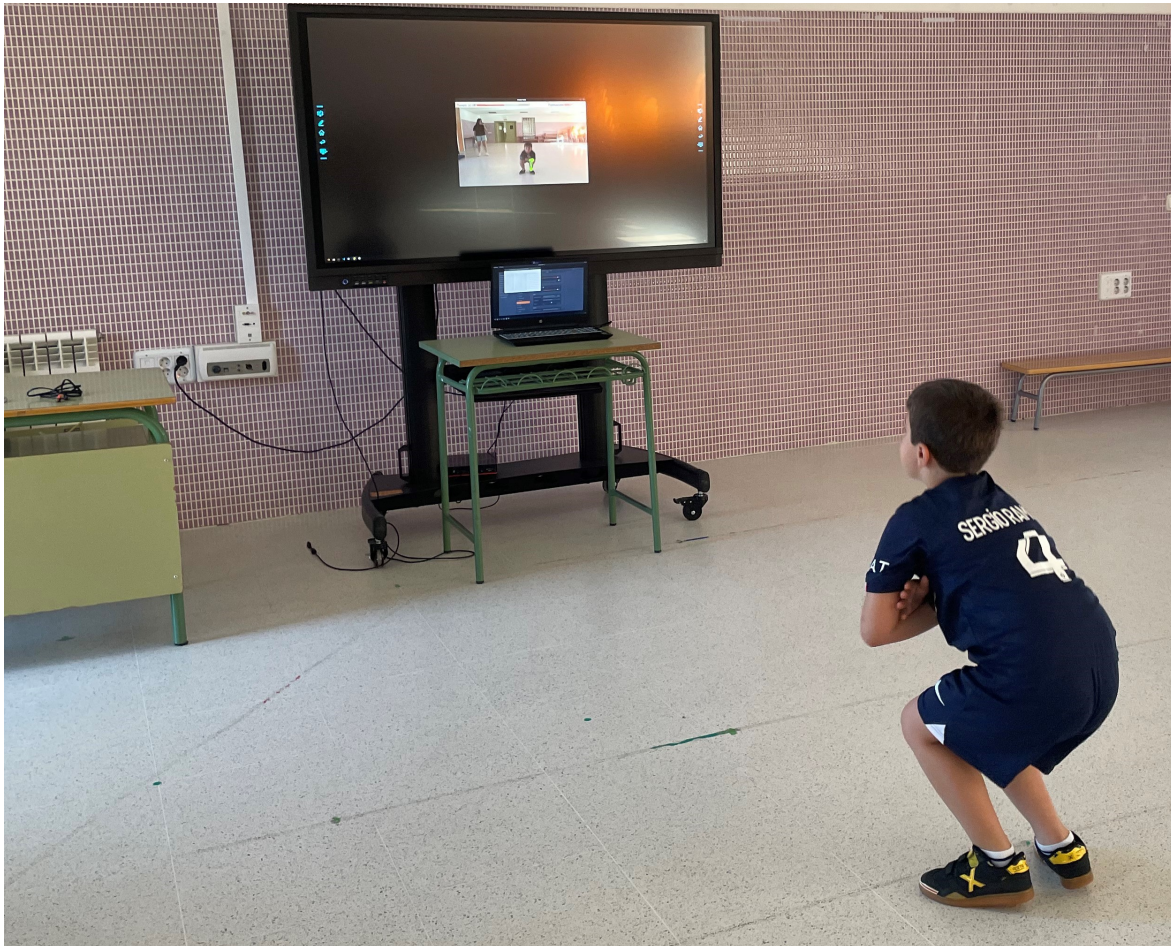


Figura 6.16: Realización de ‘Squad’ por parte de Manuel. En la imagen, el jugador forma un ángulo de 150° aproximadamente.

Por último se realizó el *exergame* ‘Movilidad’, fue el que más motivó al jugador, aunque necesito un tiempo para comprender el uso de realidad aumentada. Una vez entendido el funcionamiento de la actividad, el profesional instó al jugador a realizar diferentes posiciones, como intentar esquivar componentes manteniéndose sobre una pierna, lo que podría llevar a realizar modificaciones o incluir variantes del *exergame*. Se realizó una captura de la actividad en la figura 6.17.

Tras la realización de los *exergames* se comprobaron los resultados obtenidos. Con carácter general, el profesional mostró su agrado con el sistema y constató que podría utilizarse a modo de soporte en las sesiones de rehabilitación tras una serie de modificaciones remarcadas con anterioridad.

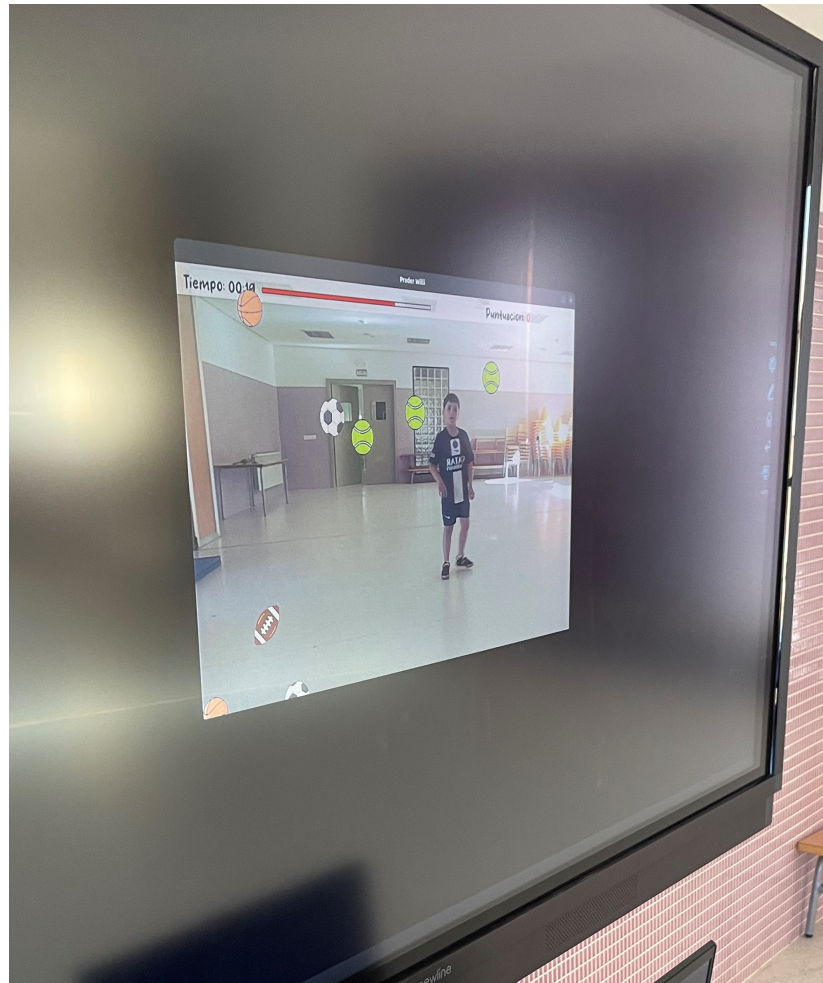


Figura 6.17: Realización de 'Movilidad' por parte de Manuel.

Capítulo 7

Conclusiones y líneas de trabajo futuro

EN este capítulo se analizan los resultados alcanzados tras la realización de este TFG. Se justificarán los diferentes objetivos marcados, competencias adquiridas y se trazarán futuras líneas de trabajo.

7.1 Consecución de objetivos

El **objetivo general** de este proyecto era el diseño, desarrollo y verificación de un sistema de soporte de la rehabilitación de niños con el síndrome de Prader-Willi. Como se ha expuesto en los capítulos anteriores este objetivo se ha cumplimentado con la creación de un sistema *software* con 3 *exergames* los cuales motivan al usuario a realizar estos ejercicios por medio de elementos de gamificación.

Para poder lograr el objetivo general se plantearon diferentes subobjetivos. Partiendo de la **realización del estudio del síndrome**, presente en el capítulo 3, se realiza una caracterización desde un enfoque técnico. En él, se exponen las causas y síntomas adheridos a esta enfermedad.

El **segundo objetivo parcial** hace referencia al estudio, selección e implementación de mecanismos de gamificación. Se ha logrado realizando una exposición sobre los mecanismos y fundamentos que lo componen, presente en el capítulo 3. Además, se han creado componentes y mecánicas reutilizables que facilitan el uso de la gamificación en los diferentes *exergames* y en la creación de nuevos.

Para cumplimentar el **tercer objetivo** y crear mecanismos de interacción sencillos y accesibles por usuarios con el SPW, se creó, haciendo uso de realidad aumentada, un módulo que permite al usuario hacer uso de su cuerpo para interactuar con el sistema, evitando así que su curva de aprendizaje sea elevada.

El **cuarto objetivo** se ha llevado a cabo por medio de la implementación de un área de modificación de ejercicios, estos elementos de modificación fueron validados por el profesional durante las entrevistas y tras la experimentación en un entorno real.

Por último, se ha llevado a cabo **una sesión de experimentación** como cumplimiento del **quinto subobjetivo**. Esta sesión, la cual se analiza en el capítulo 6, ha servido de referencia

para realizar ajustes y nuevas líneas de trabajo futuro en el proyecto.

7.2 Adquisición de competencias

A continuación se exponen las competencias propuestas al inicio de la realización del presente trabajo y se discuten cómo se han alcanzado:

[CM6] *Capacidad para desarrollar y evaluar sistemas interactivos y de presentación de información compleja y su aplicación a la resolución de problemas de diseño de interacción persona-computadora.*

El sistema es, principalmente, un sistema interactivo puesto que es capaz de hacer uso del cuerpo del usuario para controlarlo. Gracias al uso de realidad aumentada, es posible emplear interfaces naturales de usuario que facilitan la utilización del sistema. Por otra parte, se presenta información en pantalla de forma simple y concisa lo que hace el entendimiento de este más sencillo. Con respecto a la resolución de problemas de diseño se han creado componentes *software* orientados a cada uno de los problemas encontrados durante el desarrollo.

[CM4] *Capacidad para conocer los fundamentos, paradigmas y técnicas propias de los sistemas inteligentes y analizar, diseñar y construir sistemas, servicios y aplicaciones informáticas que utilicen dichas técnicas en cualquier ámbito de aplicación.*

Para poder realizar *tracking* corporal de manera efectiva se han tenido que conocer los fundamentos y paradigmas asociados a MediaPipe Pose, basado en modelos de aprendizaje automático, con el fin de construir el sistema basado en realidad aumentada. Si bien no se han desarrollado técnicas de Inteligencia Artificial, se ha tenido que comprender su funcionamiento para poder integrar en el sistema las herramientas que hacen uso de ellas.

Por otra parte, el sistema utiliza desarrollos *software* para abordar la rehabilitación de diferentes habilidades. Permite mejorar el equilibrio y la coordinación viso-manual, ayuda al fortalecimiento del tren inferior y favorece mejorar la coordinación visomotriz, fomentando la adquisición de su esquema corporal gracias a las actividades desarrolladas.

7.3 Líneas de trabajo futuras

En el desarrollo del proyecto siempre ha estado marcado por conseguir un incremento real en el progreso fisiológico del usuario. Para esto se ha construido el sistema desde el punto de vista de la escalabilidad, siendo posible la implementación de los *exergames* de manera rápida y efectiva. De tal manera, en esta sección se exponen diferentes mejoras posibles que representan líneas de trabajo futuras:

- **Validación del sistema por parte del profesional.** Se entiende cómo validación la realización de diferentes pruebas con un paciente y en un entorno real. Como objetivo de estas pruebas, se busca la comprobación de que el usuario muestra progreso tras el uso del sistema. Además, estas pruebas también permiten identificar posibles problemas y evaluar la usabilidad.
- **Ampliación del número de *exergames*.** Análisis, diseño e implementación de los *exergames* recogidos en las reuniones que no se incluyen en el alcance inicial de este trabajo. Esta línea de trabajo incluye la creación de variantes de las actividades desarrolladas y el diseño e implementación de ejercicios captados en reuniones como la realización de saltos en función de la capacidad del usuario.
- **Variedad de estadísticas y gráficas.** Esta mejora viene directamente de la mano del profesional. Aunque el sistema cuente con una serie de gráficas y estadísticas básicas estas pueden ser tremendamente mejoradas. Por ejemplo, se pueden añadir filtros sobre el espacio horario sobre el que se requieren estas estadísticas o datos. Para generar estadísticas más complejas será necesario recopilar más información durante la ejecución del ejercicio. El objetivo de esta línea de trabajo es la inclusión de filtros y mejora de la parte visual, gráfica y de visualización de datos junto con la creación de estadísticas avanzadas.
- **Módulo sobre recomendaciones.** Inclusión de un módulo sobre recomendaciones personalizadas en función de ejercicios y partes corporales. Este módulo se centra en informar al usuario que zona corporal está menos desarrollada. Este sugiere el área corporal a mejorar y que actividades debería ejecutar con más frecuencia. Un ejemplo de esto podría ser utilizar los tiempos de reacción recogidos para realizar mediciones sobre si los reflejos del usuario han disminuido o aumentado y proporcionar información al respecto.
- **Autenticación y base de datos.** Dado que el proyecto ha sido desarrollado en base a un único usuario no ha sido necesaria una base de datos para almacenar los registros y jugadores. Si este sistema se validase y se desarrollasen las diferentes líneas descritas con anterioridad, sería necesario el desarrollo de un módulo de autenticación junto con una base de datos para almacenar la información de los usuarios. Además, se necesitarán nuevos mecanismos de protección y privacidad de datos.

ANEXOS

Anexo A

Manual de usuario

A.1 Consideraciones previas

Para interactuar y realizar las actividades el usuario deberá moverse de su posición inicial, por este motivo se recomienda que el espacio donde se realice esté libre de objetos con los que se pueda impactar y a una distancia de, al menos, metro y medio de la cámara. Es necesario una buena iluminación en la sala, un exceso o escasez de luminosidad podría afectar a la detección corporal.

Se recomienda el uso de ropa clara para poder distinguir adecuadamente los componentes que aparecen en pantalla. Además, se aconseja el uso de una pantalla auxiliar, de mayor dimensión que un ordenador, para una mejor visualización del sistema.

A.2 Instrucciones de uso

Es posible acceder al entorno por cualquier dispositivo que incorpore como sistema operativo Windows, distribuciones Linux y MacOS. Debe poseer una cámara web, integrada o conectada como dispositivo externo.

Una vez se accede al sistema, es posible cambiar la cámara web a una diferente que este conectada desde antes de entrar a la aplicación.

Al acceder, el sistema comienza a reconocer los movimientos corporales y aparecerán dos círculos rojos en las manos del usuario a modo de puntero. Si la posición no es reconocida por el sistema estos puntos se encontrarán en la esquina superior izquierda.

Para navegar por el entorno se puede utilizar tanto la interacción corporal como el tradicional ratón. Para la interacción con el cuerpo, se debe comprender que los elementos en pantalla funcionan como un componente corporal con el que se puede interactuar. Para esto, se debe superponer el puntero al elemento, el cual iniciará una barra de carga situada en la parte inferior izquierda de la pantalla, una vez se complete esta barra se accede a la escena requerida.

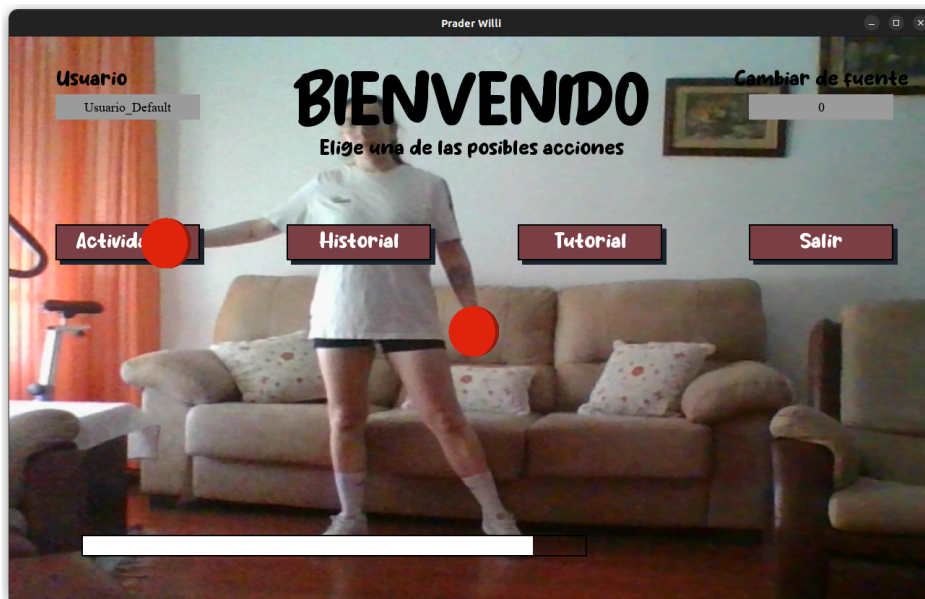


Figura A.1: Menú de inicio.

Cuando accedemos a la escena actividades el sistema simula las manos del usuario posicionando ambas en pantalla.

Podemos modificar los ajuste de juego de cada una de las actividades. Pensadas para modificarse utilizando el ratón y el teclado del dispositivo. De este modo, se incluirán nuevos ajustes tras pulsar el botón aplicar. Estos se almacenan una vez cerrada la aplicación, por lo que es opcional realizar modificaciones en ellos.



Figura A.2: Modificación de actividad.

Tras las modificaciones se puede acceder a los diferentes *exergames* de igual forma que en el menú principal.



Figura A.3: Inicio de actividad.

A.2.1 Calibración

Una vez iniciamos un juego por primera vez en una ejecución de la aplicación, el sistema calibra la posición del usuario. Esto sirve para poder crear elementos estáticos en algunas actividades y dar *feedback* visual sobre cómo posicionarse.



Figura A.4: Calibración.

A.2.2 Diagonales Superiores

Tras entrar al juego, a partir de su tarjeta de inicio, se calibra el sistema como se ha expuesto con anterioridad. Una vez iniciada la actividad, el objetivo del ejercicio es la mejora de la coordinación óculo-manual, el equilibrio y la corrección de patrones flexores del usuario.

En este *exergame* aparecen componentes por encima de los hombros del usuario. Este, debe de alcanzar los elementos correctos evitando las bombas que aparecen en pantalla. El objetivo es alcanzar la mayor cantidad de elementos para conseguir una puntuación elevada.

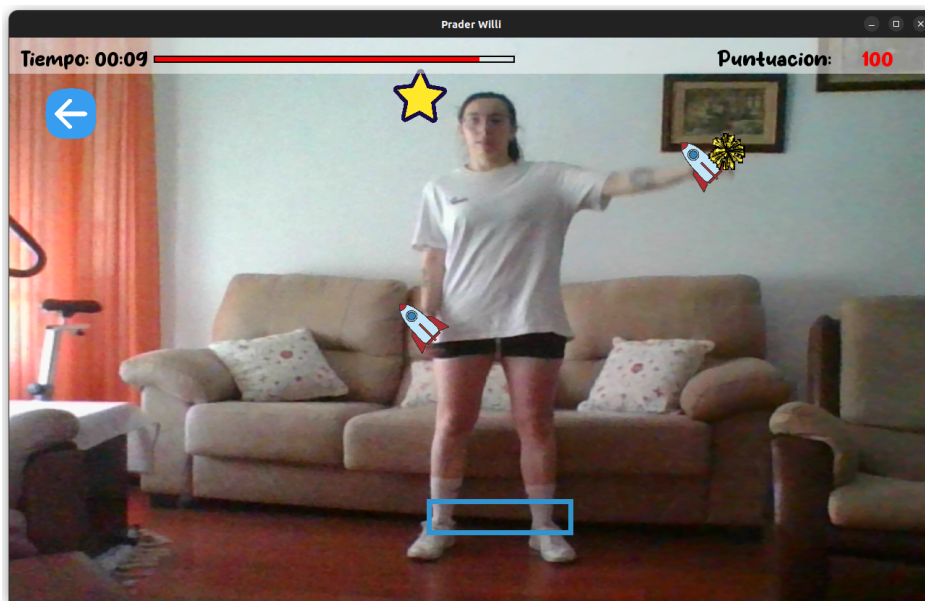


Figura A.5: Actividad Diagonales Superiores.

A.2.3 Squad

Si la calibración no ha sido realizada el sistema se calibrará. Una vez calibrado, el objetivo de esta actividad es mejorar el equilibrio y fortalecer la musculatura del usuario a través de la realización de sentadillas en un espacio de tiempo determinado por el profesional en los ajustes de juego.

Es importante ejecutar el ejercicio antes que efectuar una buena técnica pues el ángulo de flexión de la rodilla es adaptado al usuario.

Para realizar correctamente el ejercicio se incluye un pitido cuando es posible la realización de la sentadilla, si en ese espacio de tiempo no ha sido realizada se reproduce una alarma de fallo.

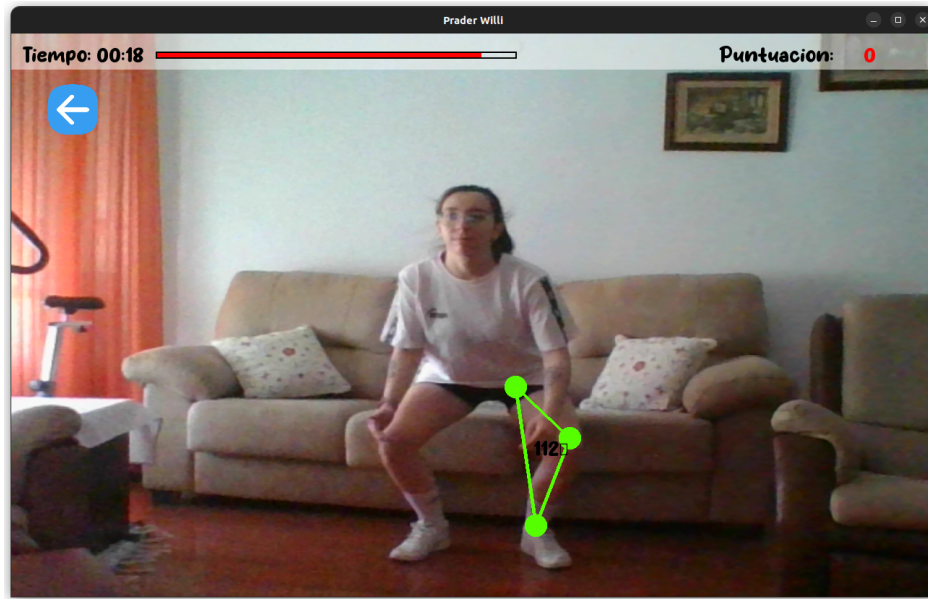


Figura A.6: Actividad Squad.

A.2.4 Movilidad

Si la calibración no ha sido realizada el sistema se calibrará. Una vez calibrado, empezará el ejercicio Movilidad. El objetivo de este ejercicio es poder mejorar la coordinación visomotriz y el equilibrio del usuario, tratando de evitar balones que caen en pantalla de forma aleatoria.

Estos componentes pueden impactar en cualquier parte del cuerpo del usuario, lo que resta puntuación. El objetivo es tratar de evitar el máximo número de elementos para poder conseguir la puntuación más alta.

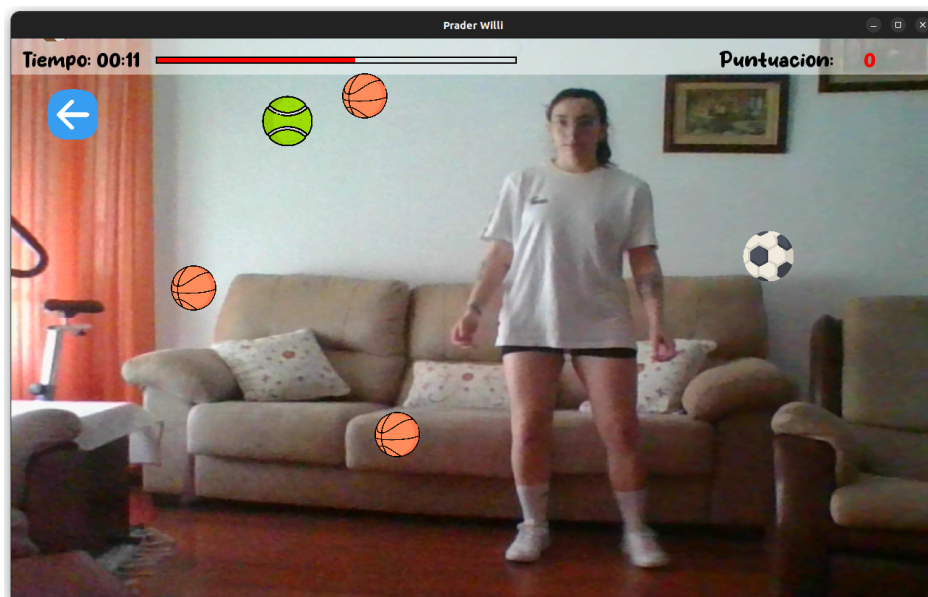


Figura A.7: Actividad Movilidad.

Histórico

El historial permite ver componentes gráficos y estadísticos. Estos se realizan en función de la actividad seleccionada, pudiendo cambiarla mediante el uso de un *DropDown* situado en la parte superior. Esta escena también incorpora un botón que da acceso a los datos recogidos en forma de tabla. A continuación, se recogen los diferentes componentes por cada una de las actividades.

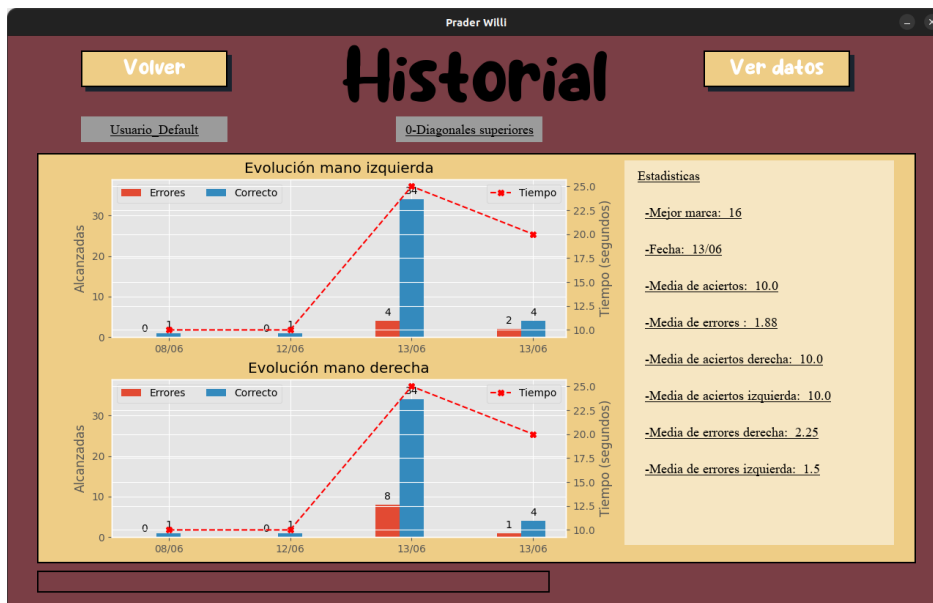


Figura A.8: Histórico de la actividad Diagonales Superiores.



Figura A.9: Histórico de la actividad Squad.



Figura A.10: Histórico de la actividad Movilidad.

A.2.5 Tutorial

Se ha incluido un tutorial al que se accede mediante un botón en el menú de inicio. Permite la visualización de distintas imágenes que explican la navegación, el acceso a los diferentes *exergames* y las distintas funcionalidades que contiene el sistema.

Estas imágenes se han dividido en grupos a los que se puede acceder por medio de botones situados por encima de estas.

Anexo B

Código

En este anexo se incluyen diferentes bloques y métodos de código clave para la correcta resolución de los diferentes *exergames* y módulo *tracking*.

B.1 Creación de posiciones - Módulo tracking

```
2 for landmark in self.results.pose_landmarks.landmark:
3     x = landmark.x * WIDTH
4     y = landmark.y * HEIGHT
5     visibility = landmark.visibility
6
7     landmark_dict = {
8         'x': x,
9         'y': y,
10        'visibility': visibility
11    }
12    self.landmark_process.append(landmark_dict)
```

Listado B.1: Bloque de código en el que se extrae la información requerida de MediaPipe. La creación de este diccionario genera una solución a la problemática de la obtención de la posición.

B.2 Exergame 0 - Diagonales Superiores

```
2 shoulder = (pose[12]['x'],pose[12]['y'])
3 elbow = (pose[14]['x'], pose[14]['y'])
4 _ , hand = get_hands_points(pose)

6 shoulder_to_elbow = (elbow[0] - shoulder[0], elbow[1] - shoulder[1])

8 shoulder_elbow_segment = euclidean_distance(
9     shoulder_to_elbow[0], shoulder_to_elbow[1]
10 )

12 elbow_to_hand = (hand[0] - elbow[0], hand[1] - elbow[1])
13 elbow_hand_segment = euclidean_distance(elbow_to_hand[0], elbow_to_hand[1])

15 hand_pos = (
16     shoulder[0] - (elbow_hand_segment + shoulder_elbow_segment),
17     shoulder[1],
18 )
```

Listado B.2: Bloque de código en el que se calcula la posición alcanzable de la mano izquierda sobre el eje x.

```
2 shoulder = (pose[11]['x'],pose[11]['y'])
3 elbow = (pose[13]['x'],pose[13]['y'])

5 hand, _ = get_hands_points(pose)

8 shoulder_to_elbow = (elbow[0] - shoulder[0], elbow[1] - shoulder[1])

10 shoulder_elbow_segment = euclidean_distance(
11     shoulder_to_elbow[0], shoulder_to_elbow[1]
12 )

14 elbow_to_hand = (hand[0] - elbow[0], hand[1] - elbow[1])
15 elbow_hand_segment = euclidean_distance(elbow_to_hand[0], elbow_to_hand[1])

17 margin_point = (
18     shoulder[0],
19     shoulder[1] - (elbow_hand_segment + shoulder_elbow_segment),
20 )
```

Listado B.3: Fragmento en el que se calcula la posición máxima alcanzable de la mano sobre el eje y.

B.3 Exergame 1 - Squad

```
1 def angle_calculate_by_points(p1, p2, p3):  
  
3     p1 = np.array([p1[0], p1[1]])  
4     p2 = np.array([p2[0], p2[1]])  
5     p3 = np.array([p3[0], p3[1]])  
  
7     l1 = np.linalg.norm(p2 - p3)  
8     l2 = np.linalg.norm(p1 - p3)  
9     l3 = np.linalg.norm(p1 - p2)  
  
11    return degrees(acos((l1**2 + l3**2 - l2**2) / (2 * l1 * l3)))
```

Listado B.4: Método para el cálculo del ángulo flexor de la rodilla.

B.4 Exergame 2 - Movilidad

```
2 for landmark in pose:  
3     x = int(landmark['x'])  
4     y = int(landmark['y'])  
  
6     pose_point = pygame.sprite.Sprite()  
7     pose_point.rect = pygame.Rect(x, y, 5, 5)  
8     pose_point.image = pygame.Surface(pose_point.rect.size)  
  
10    pose_point.rect.center = (x, y)  
11    pose_group.add(pose_point)
```

Listado B.5: Bloque de código en el que se actualiza e instancia la posición actual del usuario en sprites.

Referencias

- [Abt70] Clark C. Abt. *Serious Games*. Viking Press, 1970.
- [AES22] El síndrome Prader Willi – AESPW, 2022. Última revisión el 5 de julio de 2023. url: <https://www.sindromepraderwilli.org/el-sindrome-prader-will/>.
- [BBP⁺18] Correspondencia H Hugo Abarca Barriga, Hugo H Abarca Barriga, Miguel Chávez Pastor, Milana Trubnykova, Jorge E La Serna-Infantes, Julio A Poterico, y Acta Médica Peruana. Factores de riesgo en las enfermedades genéticas Risk factors in genetic diseases ARTÍCULO DE REVISIÓN. *Acta Med Peru*, 35:43–50, 2018.
- [BLAMGM⁺22] Juan Benito-Lozano, Greta Arias-Merino, Mario Gómez-Martínez, Alba Ancochea-Díaz, Aitor Aparicio-García, Manuel Posada de la Paz, y Verónica Alonso-Ferreira. Diagnostic Process in Rare Diseases: Determinants Associated with Diagnostic Delay. *International Journal of Environmental Research and Public Health*, 19(11), 2022. url: <https://www.mdpi.com/1660-4601/19/11/6456>.
- [CHH⁺19] Rossana Borges Cauich, Jesús Dzul Hernández, Montserrat Rodríguez Huchim, Guillermo Pérez Correa, Nina Méndez Domínguez, Rossana Borges Cauich, Jesús Dzul Hernández, Montserrat Rodríguez Huchim, Guillermo Pérez Correa, y Nina Méndez Domínguez. Manifestaciones clínicas asociadas al síndrome de Prader-Willi. *Revista biomédica*, 30:13–23, 2019. url: <http://www.scielo.org.mx/scielo.php?script=sci.arttext&pid=S2007-84472019000100013&lng=es&nrm=iso&tlng=es>
<http://www.scielo.org.mx/scielo.php?script=sci.abstract&pid=S2007-84472019000100013&lng=es&nrm=iso&tlng=es>.
- [Cho15] Y.K. Chou. *Actionable Gamification: Beyond Points, Badges, and Leaderboards*. Createspace Independent Publishing Platform, 2015. url: <https://books.google.es/books?id=jFWQrgEACAAJ>.

- [Coo12] C.A. Coonradt. *The Game of Work (pb)*. Gibbs Smith, 2012. url: <https://books.google.es/books?id=B0b3ugAACAAJ>.
- [enf19] Más de 2.5 millones de pacientes con enfermedades raras en España no cuentan con tratamientos específicos para su patología. 2019. url: <https://www.medicosypacientes.com/articulo/mas-de-25-millones-de-pacientes-con-enfermedades-raras-en-espana-no-cuentan-con>.
- [EuC99] Unit F4-Communicable, rare and emerging diseases PROGRAMME OF COMMUNITY ACTION ON RARE DISEASES of the European Parliament and of the Council. 1999.
- [FBK⁺15] Marein M. Favejee, Linda E.M. Van Den Berg, Michelle E. Kruijshaar, Stephan C.A. Wens, Stephan F.E. Praet, W. W.M. Pim Pijnappel, Pieter A. Van Doorn, Johannes B.J. Bussmann, y Ans T. Van Der Ploeg. Exercise training in adults with Pompe disease: the effects on pain, fatigue, and functioning. *Archives of physical medicine and rehabilitation*, 96:817–822, 5 2015. url: <https://pubmed.ncbi.nlm.nih.gov/25499687/>.
- [Fra10] Sam Frado. REHABILITACIÓN Y MEDICINA FÍSICA. Mirando al futuro.: Wii Balance Board podría remplazar a las plataformas que se utilizan en clínica, 10 2010. url: <http://www.rehabilitacionblog.com/2010/03/wii-balance-board-podria-reemplazar-las.html>.
- [Gad23] Suhasini Gadam. What is iterative and incremental development? Process, examples - LogRocket Blog, 1 2023. url: <https://blog.logrocket.com/product-management/what-is-iterative-incremental-development-process-examples/>.
- [GLB10] Oscar Tinoco Gomez, Pedro Pablo Rosales Lopez, y Julio Salas Bacalla. Criterios de selección de metodologías de desarrollo de software. *Industrial data*, 13(2):70–74, 2010.
- [How09] Jacinda Howard. For physical therapy, Federal Way nursing homes embrace Nintendo Wii | Federal Way Mirror, 9 2009. url: <https://www.federalwaymirror.com/life/for-physical-therapy-federal-way-nursing-homes-embrace-nintendo-wii/>.
- [LTN⁺19] Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, y Matthias

- Grundmann. MediaPipe: A Framework for Building Perception Pipelines, 2019.
- [MP15] Esteban Gabriel Maida y Julián Pacienza. Metodologías de desarrollo de software. 2015.
- [MVBBS22] José Luis Mejía-Villamizar, Diego Fernando Villegas-Bermúdez, y Heller Sánchez. Análisis de marcha de una persona con muleta de manos libres usando el programa Kinovea. *Revista UIS Ingenierías*, 21(3):1–8, may 2022. url: <https://revistas.uis.edu.co/index.php/revistauisingenierias/article/view/11163>.
- [orp22] Orphanet: Síndrome de Prader Willi, 2022. Última revisión el 5 de julio de 2023. url: [https://www.orpha.net/consor/cgi-bin/Disease_Search.php?lng=ES&data_id=139&Disease.Disease_Search_diseaseGroup=prader-willi&Disease.Disease_Search_diseaseType=Pat&Enfermedad\(es\)/](https://www.orpha.net/consor/cgi-bin/Disease_Search.php?lng=ES&data_id=139&Disease.Disease_Search_diseaseGroup=prader-willi&Disease.Disease_Search_diseaseType=Pat&Enfermedad(es)/).
- [Per14] Marta Perse. Terapia y Fisio para Ti: Nuevas tecnologías: Wii y Kinect, 2 2014. url: <http://terapiayfisioparati.blogspot.com/2014/02/nuevas-tecnologias-wii-y-kinect.html>.
- [Pér19] María Teresa Montañana Pérez. Papel de la fisioterapia en las enfermedades raras. 2019.
- [RB17] Camilo Rigueros Bello. La realidad aumentada: lo que debemos conocer. *Tecnología Investigación y Academia*, 5(2):257–261, nov. 2017. url: <https://revistas.udistrital.edu.co/index.php/tia/article/view/11278>.
- [Reh16] Software para fisioterapia, 2016. url: <https://rehametrics.com/software-fisioterapia/>.
- [Rib10] Emilio Ribes. John B. Watson: El conductismo y la fundación de una psicología científica. *Acta Comportamentalia: Revista Latina de Análisis del Comportamiento*, 3(3), jun. 2010. url: <https://www.revistas.unam.mx/index.php/acom/article/view/18327>.
- [Sil22] María Teresa García Silva. Qué son y cómo se presentan las Enfermedades Raras en niños, 2022. url: <https://www.guiainfantil.com/articulos/salud/enfermedades-infantiles/las-enfermedades-raras-caracteristicas-y-sintomas/>.
- [TSF⁺13] Stewart G Trost, Deborah Sundal, Gary D Foster, Michelle R Lent, y Deleen Vojta. Effects of exergaming on physical fitness in elementary school

students: a pilot study. *Journal of Pediatric Health Care*, 27(2):133–140, 2013. url: <https://www.researchgate.net/publication/229324094>.

- [Wor99] World Health Organization. E-Health: Use of the Internet and related technologies in healthcare. Technical report, World Health Organization, 1999. url: https://www.who.int/ehealth/resources/WHO_eHealth_report.pdf.
- [ZC11] Gabe Zichermann y Christopher Cunningham. *Gamification by Design: Implementing Game Mechanics in Web and Mobile Apps*. O'Reilly Media, Inc., 2011.

Este documento fue editado y tipografiado con \LaTeX empleando la clase **esi-tfg** (versión 0.20181017) que se puede encontrar en:
https://bitbucket.org/esi_atc/esi-tfg

