



UNIVERSIDAD DE VALLADOLID

E.T.S.I. TELECOMUNICACIÓN

**TRABAJO FIN DE MÁSTER**

**MÁSTER EN INGENIERÍA DE TELECOMUNICACIÓN**

**Desarrollo de herramienta software creada con Python para el análisis del ciclo y de los principales parámetros de la marcha a través de datos recogidos con sensores inerciales**

**Autor: Luis Eduardo Casado Vecino**

**Tutor: Mario Martínez Zarzuela**

Valladolid, 27 de septiembre de 2023

**TÍTULO:** **Desarrollo de herramienta software creada con Python para el análisis del ciclo y de los principales parámetros de la marcha a través de datos recogidos con sensores inerciales**

**AUTOR:** **Luis Eduardo Casado Vecino**

**TUTOR:** **Mario Martínez Zarzuela**

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

---

**TRIBUNAL**

---

**PRESIDENTE:** **Javier Manuel Aguiar Pérez**

**SECRETARIO:** **Jaime Gómez Gil**

**VOCAL:** **Miriam Antón Rodríguez**

**PRESIDENTE SUPLENTE:** **María Ángeles Pérez Juárez**

**SECRETARIO SUPLENTE:** **Jesús Poza Crespo**

**VOCAL SUPLENTE:** **Carlos Gómez Peña**

---

**FECHA:** **27 de septiembre de 2023**

**CALIFICACIÓN:**

---



## RESUMEN DE TFM

En este trabajo se ha desarrollado una herramienta en Python que permite visualizar los datos obtenidos por los sensores inerciales para el análisis de la marcha, de sus ciclos y de los parámetros que esta arroja. De esta forma la aplicación ayuda a visualizar y comprender los datos que la captura de movimiento con sensores inerciales proporciona. Las curvas de movimiento obtenidas, si la adquisición con sensores es correcta, pueden ayudar a una valoración más objetiva de los pacientes.

La aplicación, por tanto, permite visualizar los datos obtenidos en la captura de diferentes formas. Permite la visualización de los diferentes parámetros de cada pierna por separado, permitiendo ver la evolución del parámetro a través de toda la captura de movimiento o de cada ciclo en concreto o de todos los ciclos en conjunto (realizando la media). También permite la visualización de estos mismos datos para los diferentes parámetros, pero juntado ambas piernas en la misma gráfica. Por último, también permite la visualización de la media de los ciclos para todos los parámetros en conjunto.

Se ha utilizado Python para realizar la aplicación, ya que es de los lenguajes más conocidos y con más librerías para las ciencias de la computación. Para ello se ha usado una serie de librerías de Python que han ayudado al procesamiento y la visualización de los datos dentro de la aplicación. Para la implementación de la aplicación se ha utilizado el entorno de desarrollo Spyder de la distribución de Anaconda. Como ficheros de entrada a procesar, la aplicación permite usar tanto datos generados por una aplicación comercial (Xsens Awinda) como por una herramienta de software libre (OpenSense).

Palabras clave: Análisis de marcha, IMUs, Python, Xsens, OpenSense, Spyder



## ABSTRACT

In this work, a tool has been developed in Python that allows viewing the data obtained by inertial sensors for the analysis of gait. In this way, the application helps to visualize and understand the data that motion capture with inertial sensors provides. The movement curves obtained, if the acquisition with sensors is correct, can help a more objective assessment of patients.

The application, therefore, allows to visualize the data obtained in the capture of different shapes. It allows the visualization of the different parameters of each leg separately, allowing to see the evolution of the parameter through the entire motion capture or each specific cycle or of all the cycles as a whole (performing the average). It also allows the visualization of these same data for the different parameters but joined both legs on the same graph. Finally, it also allows the visualization of the average of the cycles to all parameters together.

Python has been used to make the application, since it is one of the best-known languages and with more libraries for computer science. For this, a series of libraries has been used of Python that have helped the processing and visualization of the data within the application. For the implementation of the application, the Spyder development environment of the Anaconda distribution. As input files to be processed, the application allows to use both Excel sheets and MOT files containing the motion capture data.

Key words: Gait analysis, IMUs, Python, analysis, Xsens, OpenSense, Spyder



## AGRADECIMIENTOS

A mi familia y muy especialmente a mis padres y a mi hermano, que siempre están ahí para apoyarme y aconsejarme con todo lo que hago.

A mis amigos por su amabilidad y apoyo.

A Mario que tanta paciencia ha tenido conmigo durante estos dos años.



## LISTA DE ACRONIMOS

Se listan una serie de términos que aparecerán por el documento de manera abreviada:

- MVC: Model View Controller
- FR: Functional Requirement
- NFR: Non Functional Requirement
- IDE: Integrated Development Environment
- IMU: Inertial Measurement Unit
- ASIS: Anterior Superior Iliac Spines
- PSIS: Posterior Superior Iliac Spines
- GT: Greater Trochanter
- VFX: Visual effects
- FPS: Frames Per Second
- MP: Megapíxels
- FoV: Field Of View
- AoV: Angle of View
- MB: Megabytes
- FBX: Formato de fichero para animación 3D
- XLSX: Fichero Excel
- SQL: Structured Query Language



## ÍNDICE

RESUMEN DE TFM.....	4
ABSTRACT.....	6
AGRADECIMIENTOS.....	8
LISTA DE ACRONIMOS.....	10
ÍNDICE.....	12
LISTA DE FIGURAS Y TABLAS.....	15
- FIGURAS.....	15
- TABLAS.....	15
1. INTRODUCCIÓN.....	18
1.1. CONTEXTUALIZACIÓN.....	18
1.2. OBJETIVOS.....	18
1.3. FASES DEL PROYECTO.....	19
1.4. ESTRUCTURA DEL DOCUMENTO.....	19
2. ANATOMÍA DE LA MARCHA.....	22
2.1. ANATOMIA DE LA PARTE INFERIOR DEL ESQUELETO.....	22
2.2. ANATOMIA DE LA PARTE INFERIOR DE LA MUSCULATURA.....	25
2.3. ANATOMIA DE LAS ARTICULACIONES.....	25
3. CICLO DE LA MARCHA.....	28
3.1. DEFINICIÓN.....	28
3.2. FASES Y EVENTOS.....	28
3.3. PARÁMETROS ESPACIO TEMPORALES.....	29
3.4. SEGMENTOS, PLANOS, EJES Y MOVIMIENTOS.....	30
4. SISTEMAS DE CAPTURA DE MOVIMIENTO.....	33
4.1. APLICACIONES.....	33
4.2. TIPOS DE SISTEMAS.....	33
4.3. VENTAJAS Y DESVENTAJAS.....	34
4.4. SISTEMA VICON.....	35
4.5. SISTEMAS XSENS.....	39
4.6. COMPARATIVA VICON VS. XSENS.....	43
5. MATERIALES Y MÉTODOS.....	45
5.1. PYTHON.....	45
5.2. PARADIGMA MODELO-VISTA-CONTROLADOR.....	47
5.3. HERRAMIENTA SOFTWARE DE DESARROLLO: ANACONDA - SPYDER.....	48

6.	CAPTURA DE MOVIMIENTO .....	52
6.1.	INTRODUCCIÓN .....	52
6.2.	DISEÑO DEL SISTEMA DE CAPTURA DE MOVIMIENTO.....	52
6.3.	PRUEBAS DE CAPTURA DE MOVIMIENTO .....	52
6.4.	PROTOCOLO DE CAPTURA DE MOVIMIENTO.....	53
6.5.	RESULTADOS OBTENIDOS DE LA CAPTURA.....	54
7.	DISEÑO DE LA APLICACIÓN .....	56
7.1.	REQUISITOS .....	56
7.2.	CASOS DE USO.....	58
8.	MANUAL DE USUARIO.....	61
8.1.	INTRODUCCION .....	61
8.2.	PRERREQUISITOS.....	61
8.3.	DESCARGA .....	61
8.4.	CONFIGURACIÓN PREVIA .....	62
8.5.	EJECUCIÓN.....	64
9.	CONCLUSIONES Y LÍNEAS FUTURAS .....	73
9.1.	CONCLUSIONES .....	73
9.2.	LÍNEAS FUTURAS .....	73
	REFERENCIAS.....	76



## LISTA DE FIGURAS Y TABLAS

### - FIGURAS

Figura 1. Ilustración de la pelvis (S, 2017).....	22
Figura 2. Ilustración del fémur (Serrano, 2023) .....	23
Figura 3. Ilustración de la rótula (Azucas, 2023) .....	23
Figura 4. Ilustración de la tibia y el peroné (Moreno, 2023).....	24
Figura 5. Ilustración del pie y tobillo (Huesos Del Pie Y El Tobillo, 2021) .....	24
Figura 6. Fases de la marcha (Pirker & Katzenschlager, 2016) .....	29
Figura 7. Segmentos de la parte inferior del cuerpo para la captura (Karatsidis et al., 2016) ...	30
Figura 8. Logo de VICON (Vicon   Award Winning Motion Capture Systems, 2023).....	35
Figura 9. Sistema simple VICON .....	38
Figura 10. Logo de XSENS .....	40
Figura 11. Características técnicas de cada hardware .....	41
Figura 12. Características técnicas de los sistemas software .....	43
Figura 13. Logo de Python (Welcome to Python.org, 2023).....	45
Figura 14. Logo de Anaconda (Welcome! — Anaconda Documentation, 2023) .....	49
Figura 15. Logo de Spyder (Welcome to Spyder’s Documentation — Spyder 5 Documentation, 2023) .....	49
Figura 16. Lista de ficheros descargables en Gitlab .....	62
Figura 17. Lugar en el código del fichero main.py donde se especifica la ruta .....	62
Figura 18. Ejemplo de lista de ficheros .....	65
Figura 19. Ejemplo de fichero mal seleccionado.....	65
Figura 20. Selección de fichero exitosa y lista de pestañas de fichero Excel .....	66
Figura 21. Selección del modo de visualización .....	66
Figura 22. Lista de parámetros por defecto del modo single para fichero Excel .....	67
Figura 23. Selección de tolerancia exitosa .....	67
Figura 24. Selección del modo duo .....	68
Figura 25. Visualización de gráficas para el modo “dúo” para ficheros Excel.....	68
Figura 26. Selección exitosa del modo "all" .....	69
Figura 27. Visualización de gráficas para el modo "all" para ficheros Excel .....	69
Figura 28. Visualización de gráfica de máximos y mínimos .....	70
Figura 29. Visualización de gráfica de los ciclos de la marcha .....	70
Figura 30. Visualización de gráfica de media y desviación estándar .....	71

### - TABLAS

Tabla 1. Movimientos, músculos implicados y periodos del ciclo (Malanga, 2005) .....	25
Tabla 2. Características técnicas de las cámaras Valkyrie (Vicon   Award Winning Motion Capture Systems, 2023) .....	36
Tabla 3. Características técnicas de las cámaras Vero (Vicon   Award Winning Motion Capture Systems, 2023) .....	36
Tabla 4. Características técnicas de las cámaras Vantage. (Vicon Motion Systems, 2016). .....	37
Tabla 5. Características técnicas de las cámaras Viper (VICON, 2019) .....	37
Tabla 6. Paquetes de Scipy (Signal Processing (scipy.signal) — SciPy v1.11.2 Manual, 2023)....	47
Tabla 7. Requisito funcional FRQ01 .....	56
Tabla 8. Requisito funcional FRQ02 .....	56
Tabla 9. Requisito funcional FRQ03 .....	56

Tabla 10. Requisito funcional FRQ04 .....	56
Tabla 11. Requisito funcional FRQ05 .....	56
Tabla 12. Requisito funcional FRQ06 .....	57
Tabla 13. Requisito funcional FRQ07 .....	57
Tabla 14. Requisito funcional FRQ08 .....	57
Tabla 15. Requisito funcional FRQ09 .....	57
Tabla 16. Requisito funcional FRQ10 .....	57
Tabla 17. Requisito funcional FRQ11 .....	57
Tabla 18. Requisito funcional FRQ12 .....	57
Tabla 19. Requisito funcional NFRQ01.....	58
Tabla 20. Requisito funcional NFRQ02.....	58
Tabla 21. Requisito funcional NFRQ03.....	58
Tabla 22. Caso de uso VisualizarGráficas .....	59
Tabla 23. Lista de los parámetros configurables por defecto .....	63



## 1. INTRODUCCIÓN

### 1.1. CONTEXTUALIZACIÓN

Con el avance de la tecnología y en este caso de las comunicaciones inalámbricas se ha conseguido llegar a un punto donde los sensores se utilizan cada vez para más casos de uso. Este movimiento se ha denominado sensorización. Los sensores más comunes son los táctiles, los de movimiento, los acelerómetros, los giroscopios o las brújulas, por decir algunos. Todos ellos suelen venir integrados en los smartphones, grandes precursores de esta sensorización.

Uno de los casos de uso de los sensores es la recolección de información de movimiento. Como ya se verá más adelante, existen muchos tipos de sensores capaces de generar información relevante al movimiento. Esto ha permitido desarrollar muchas aplicaciones que permiten capturar el movimiento, y en particular, la marcha humana. Con ello se puede estudiar los movimientos de una manera más precisa y poder detectar posibles afecciones del aparato locomotor o del sistema nervioso central. Por ello el desarrollo de este tipo de aplicaciones tienen una utilidad muy interesante en la ingeniería biomédica. También destaca el uso de la captura de movimiento para la industria del cine y de los videojuegos.

Uno de los tipos de sistemas más utilizados en la industria de la captura de movimiento son los que utilizan sensores inerciales o unidades de medición inercial (IMU). La captura de movimiento realizada para generar los datos usados en la aplicación creada en este proyecto usa este tipo de sensores.

Este proyecto se ha centrado en el estudio del movimiento de la parte inferior del cuerpo, es decir de la marcha. Esta se puede estudiar como la repetición de los mismos movimientos periódicamente, por lo que se puede dividir en ciclos donde se repiten los mismos movimientos. Para un mejor estudio de ella se ha generado una aplicación que permite ver la evolución de los movimientos que realizan las diferentes articulaciones del tren inferior (pelvis, rodilla y tobillo) sobre los diferentes planos y ejes. Todos ellos (movimientos, planos y ejes) se verán y explicarán en los próximos capítulos para una mejor comprensión de la aplicación.

La aplicación generada para el análisis de los distintos parámetros ha sido desarrollada sobre Python, uno de los lenguajes de programación más ampliamente utilizados. Se ha escogido este lenguaje por la utilidad y conveniencia de sus librerías. En este TFM se han utilizado algunas de las más conocidas, como Pandas o Numpy. También se ha utilizado el paradigma Modelo-Vista-Controlador para una mejor implementación y entendimiento de la aplicación.

### 1.2. OBJETIVOS

El objetivo principal del proyecto ha sido desarrollar una aplicación que permita la visualización de curvas para análisis de la marcha a partir de adquisiciones capturadas por sensores inerciales y con capacidad de leer los datos de ángulos articulares generados por una aplicación comercial y por una aplicación de software libre.

De esta manera, se perseguía también ayudar al equipo de investigación de la Universidad con su estudio sobre la captura de movimiento del cuerpo humano mediante sensores inerciales y con cámaras. Dentro del desarrollo del proyecto se plantearon varios diseños de diferentes aplicaciones hasta que surgió el finalmente elegido. Se trata de una aplicación que permita visualizar los diferentes movimientos que realizan las articulaciones del tren inferior en función del ciclo de la marcha para poder ver tanto su evolución durante la marcha como su evolución en el ciclo (realizando la media). Esta funcionalidad ya la tienen programas comerciales de pago

como los que se comentarán (XSENS), pero no estaba disponible en código abierto, por lo que es una herramienta interesante para el grupo de investigación.

Otro objetivo importante era poder aprender a desenvolverse en Python y sus librerías, ya que es un lenguaje muy demandado actualmente.

Por último, se destacan también otros objetivos como poder aprender sobre un tema tan interesante como la marcha y sus ciclos, fundamental en estudios de biomecánica, así como aprender a realizar una captura de movimiento mediante sensores inerciales y cámaras.

### 1.3. FASES DEL PROYECTO

La realización de este proyecto, por tanto, se ha realizado en las siguientes fases:

- **Fase de captura de movimiento:** La primera fase consistió en generar una gran cantidad de datos para luego poder usarlos. Esta captura de movimientos se realizó con varios estudiantes que también estaban realizando otros proyectos relacionados con la captura de movimiento, con otros sistemas u otras partes del cuerpo.
- **Fase de elección de la aplicación:** En la segunda fase se realizó un estudio de que aplicación sería la más interesante para llevar a cabo, escogiendo la que se ha comentado en la anterior sección.
- **Fase de documentación:** Una vez realizada la elección, se pasó a una fase de estudio tanto de Python como de sus librerías. Al ser un lenguaje de fácil comprensión y no muy diferente a otros conocidos, el estudio de Python y sus librerías fue corto. Durante este periodo también se produjo el estudio del paradigma MVC.
- **Fase de análisis:** Previo paso a la implementación de la aplicación se realizó un análisis preliminar para definir las funcionalidades y requisitos del programa. Esto se pueden ver en el capítulo dedicado a ello.
- **Fase de implementación:** En esta fase se produjo el desarrollo de la aplicación mediante la distribución de Anaconda y el IDE de Spyder.
- **Fase de testeo:** Por último, una vez realizada la aplicación se produjo la fase de pruebas hasta depurar todos los posibles fallos y errores contrarios a la funcionalidad y requisitos determinados en la fase de análisis. Las pruebas se realizaron en el mismo dispositivo donde se realizó la implementación.

### 1.4. ESTRUCTURA DEL DOCUMENTO

Este documento sobre el trabajo realizado está formado por nueve capítulos. En esta sección se resumirá cada uno de ellos

Este Capítulo 1 inicia, organiza y contextualiza el presente documento. Contiene en primer lugar la contextualización que relaciona el trabajo con la realidad tecnológica. Después se listan los objetivos del proyecto. Posteriormente se listan las fases que ha seguido el presente proyecto. Por último se realiza un pequeño resumen del contenido de los capítulos del documento.

En los siguientes capítulos se realiza un estudio sobre los conceptos claves de la marcha involucrados en la captura de movimiento para la contextualización de este proyecto. Para ello se ha dividido este estudio en el Capítulo 2: Anatomía de la marcha, el Capítulo 3: Ciclos de la marcha) y el Capítulo 4: Sistemas de captura de movimiento.

Por lo tanto, en el Capítulo 2 se ha realizado un estudio de la anatomía de la marcha. En primer lugar, se ha comentado la anatomía de esqueleto de la parte inferior, posteriormente la

anatomía de la musculatura de la parte inferior, y por último la anatomía de las articulaciones, todos ellos importantes para la generación del movimiento.

En el Capítulo 3, como se ha comentado antes se ha realizado un estudio del ciclo de la marcha, para ello este capítulo se divide en la definición de esta, la definición de sus fases, destacando las dos principales (estancia y swing), la definición de los parámetros temporales del ciclo y por último la definición de los segmentos, planos, ejes y movimientos más importantes que se utilizan en la captura de movimiento en relación con el ciclo de la marcha.

Por último, en estos capítulos de estudio de los conceptos claves de la marcha involucrados en la captura de movimiento se ha realizado, en el Capítulo 4, se realiza un extenso estudio sobre los sistemas comerciales más importantes tanto para la captura de movimiento con cámaras como es VICON, como para la captura de movimiento con sistemas inerciales, XSENS. Para ello se ha realizado previamente una contextualización previa de los tipos de sistemas de captura existentes.

El Capítulo 5, por su parte, se compone de los materiales y métodos usados en este proyecto para la creación de la aplicación. Para ello se ha realizado en primer lugar un resumen de Python y de sus librerías utilizadas en este proyecto. Posteriormente, se ha realizado un pequeño comentario del paradigma Modelo-Vista-Controlador, también usado en la aplicación. Por último, se comenta también el IDE y su distribución (Spyder y Anaconda, respectivamente) utilizado.

En el Capítulo 6 se ha comentado como se realizó la captura de movimiento necesaria para crear los datos de entrada para la aplicación. Para ello se ha explicado cómo se diseñó el sistema de captura de movimiento, comentando como se situaron los marcadores para la captura de movimiento con sensores inerciales y las cámaras para la captura con cámaras. Posteriormente se han explicado las diferentes pruebas que se realizaron para la captura de movimiento de la marcha y se ha adjuntado el protocolo que se realizó para realizar las pruebas. Por último, se comenta los tipos de ficheros generados.

El capítulo siguiente, Capítulo 7, se comentan el diseño previo a la implementación de la aplicación. Para ello se especifican tanto los requisitos funcionales como los requisitos no funcionales en primer lugar. Posteriormente se especifica el caso de uso único de esta aplicación.

El Capítulo 8 contiene el manual de usuario de la aplicación. Esta realizado con el objetivo de poder usar la aplicación a través de él. Se divide en la sección de prerequisites necesarios para poder ejecutarla, la sección de descarga, la sección de configuración previa y la sección de manejo de la aplicación propiamente dicha.

Por último, el Capítulo 9 contiene las conclusiones y líneas futuras posibles del proyecto. Después de ello hay un capítulo sin numerar de referencias para concluir con este documento.



## 2. ANATOMÍA DE LA MARCHA

Para realizar un análisis sobre la marcha es necesario entender la anatomía y funcionamiento del aparato locomotor. En este capítulo se realizará una breve descripción sobre los sistemas que intervienen en la marcha, su función en ella y en términos de análisis donde es más recomendable situar los sensores inerciales o (en el caso del sistema de captura de movimiento usado en este trabajo).

Tanto el esqueleto como la musculatura de la parte inferior del cuerpo son relevantes para el estudio de la marcha. El esqueleto proporciona estructura y soporte mientras que la musculatura proporciona la capacidad de producir fuerza.

### 2.1. ANATOMIA DE LA PARTE INFERIOR DEL ESQUELETO

La parte inferior del esqueleto se compone de los siguientes huesos, de arriba abajo, de la pelvis, el fémur, el menisco, la tibia, el peroné y, por último, los huesos del pie: tarsales, meta tarsales y falanges.

La pelvis se compone de dos partes de tres huesos fusionados cada una y conecta la parte inferior de la columna vertebral. Para el análisis de la marcha existen cuatro prominencias que son relevantes para la colocación de marcadores en algunos sistemas de captura de movimiento:

- Las espinas iliacas anterior superior (ASIS, en inglés)
- Las espinas iliacas posterior superior (PSIS, en inglés)

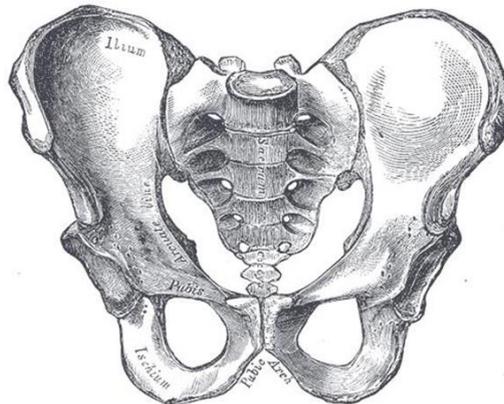


Figura 1. Ilustración de la pelvis (S, 2017)

La pelvis aporta estabilidad y soporte a la columna vertebral. Su función principal es establecer un enlace entre columna y los huesos de la parte inferior, por lo que permite la transferencia de fuerzas de manera ordenada durante el movimiento del cuerpo con la marcha (Arias, 2023).

El fémur es el hueso más grande, resistente y fuerte del cuerpo humano. Se sitúa en la parte superior de la pierna y su cabeza redondeada, en su parte superior, se inserta en la pelvis y rota libremente en ella. Por el otro lado, la parte inferior forma parte de la articulación de la rodilla. Para el análisis de la marcha existen 2 prominencias donde es recomendable situar los marcadores:

- El trocánter mayor (GT, en inglés)
- El cóndilo femoral



Figura 2. Ilustración del fémur (Serrano, 2023)

El fémur es el único hueso del muslo por lo que es un elemento clave para servir movimientos en la articulación coxofemoral y a la rodilla. Es por tanto un punto de llegada y partida de muchos músculos, por lo que es un elemento principal de la marcha. Participa también en la flexión y extensión de la cadera y de la rodilla (Junquera, 2023).

La rótula por su parte es un hueso triangular situado en el medio de la articulación de la rodilla. En términos del análisis de la marcha, no es necesario situar sensores sobre este hueso. Es el eje principal de la articulación de la rodilla, por lo que está involucrada en la flexión y extensión de esta. También sirve de apoyo al cuádriceps, soporta el roce del tendón rotuliano, da estabilidad y proporciona la fuerza para realizar los movimientos de la rodilla (Junquera, 2023)



Figura 3. Ilustración de la rótula (Azucas, 2023)

La tibia y el peroné se sitúan en la parte inferior de la pierna. De arriba abajo, La tibia forma parte de la zona inferior de la rodilla y termina en la parte superior de la articulación del tobillo. El peroné es más fino que la tibia y actúa como un acomodador para los músculos. Al igual que los anteriores huesos, para el análisis de la marcha son destacables estas partes:

- El maléolo medio
- El maléolo lateral
- La tuberosidad de la tibia
- La cabeza del peroné

La tibia y el peroné cumplen varias funciones esenciales para la marcha. Actúan de soporte de peso y estabilidad, ya que la tibia es el principal hueso de carga del cuerpo, por lo que es responsable de transmitir las fuerzas del peso corporal de la rodilla hasta el tobillo y el pie. El peroné por tanto contribuye a la estabilidad de la pierna (Triatlón, 2023).

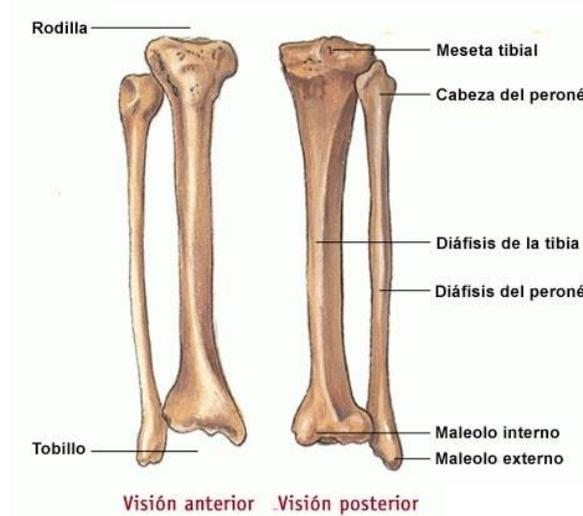


Figura 4. Ilustración de la tibia y el peroné (Moreno, 2023)

Ambos huesos también contribuyen a la inserción de músculos y tendones, como el cuádriceps y los isquiotibiales en la tibia y el peroneo largo y corto en el peroné. Por último también contribuyen a la movilidad articular de la rodilla y el tobillo permitiendo un amplio rango de movimientos (Triatlón, 2023).

El tobillo y el pie están formados por veintiséis huesos que, como ya se ha comentado previamente pueden ser tarsales, meta tarsales y falanges. Para el análisis de la marcha no se destaca ninguna prominencia donde situar sensores.



Figura 5. Ilustración del pie y tobillo (Huesos Del Pie Y El Tobillo, 2021)

La función del tobillo es la proporcionar estabilidad a la extremidad inferior. También permite realizar los movimientos de flexión y extensión del pie. Por otro lado, se sitúa el pie, siendo su función principal la de conexión del organismo con el medio que lo rodea, adaptándose a él, siendo la base del aparato locomotor (Viladot Voegeli, 2003).

## 2.2. ANATOMIA DE LA PARTE INFERIOR DE LA MUSCULATURA

Los músculos que componen las piernas suelen ser largos y estrechos. Cuando estos se contraen, inducen movimiento de la articulación. Generalmente actúan conjuntamente para conseguir movimiento coordinado. Estos se denominan "músculos sinérgicos", mientras que los "músculos antagonistas" trabajan al revés y se oponen al movimiento.

Los músculos suelen recibir nombre en referencia a su forma, localización, colocación, orientación y función. Se adjunta una tabla con los movimientos más importante de la marcha y los músculos implicados en ellos:

Actividad Muscular	Músculos	Periodo
<b>Amortiguamiento</b>	Cuádriceps Dorsiflexores	Carga de peso
<b>Estabilizadores</b>	Glúteo mayor, mediano y menor Tensor de la fascia lata Erector de la columna	Fase de apoyo
<b>Levantamiento del pie</b>	Flexor largo de los dedos Flexor largo del dedo gordo Gastrocnemio Peroneo largo y corto Soleo Tibial posterior	Descarga de peso
<b>Aceleradores</b>	Aductor Mayor y Largo Psoas iliaco Sartorio	Descarga de peso
<b>Controlador del pie</b>	Extensor largo de los dedos Extensor largo del dedo gordo Tibial anterior	Fase de balanceo
<b>Deceleradores</b>	Grácil Semimembranoso Semitendinoso Bíceps femoral	Fase de balanceo De balanceo medio hasta contacto inicial

Tabla 1. Movimientos, músculos implicados y periodos del ciclo (Malanga, 2005)

Los músculos que mueven el muslo, o la parte superior de la pierna se originan en la pelvis y se sitúan sobre el fémur, mientras que los músculos que mueven la parte inferior de la pierna se originan en la pelvis o el fémur y se sitúan sobre la tibia o peroné. Por último, los músculos que mueven el pie se originan en el fémur o en la parte inferior y se sitúan sobre el tobillo y los huesos del pie.

## 2.3. ANATOMIA DE LAS ARTICULACIONES

El movimiento del ser humano se suele definir usando un sistema de coordenadas cartesiano con planos perpendiculares. Este sistema difiere del sistema anatómico de cada articulación. La mayoría de los sistemas asumen cada segmento del cuerpo como una parte rígida donde estos se conectan con otros segmentos mediante las articulaciones.

El rango de movimiento disponible para cada articulación se limita por los ligamentos, más que por la forma de los huesos. Si los ligamentos se rompen, se aumentan los grados de libertad de la articulación. Cuando los músculos abarcan más de una articulación, el rango de movimiento

como de la articulación también depende de las posiciones de las otras articulaciones que el músculo cruza.

### 2.3.1. ARTICULACIÓN DE LA CADERA

La cadera conecta el fémur con la pelvis y mecánicamente se puede definir como una cabeza articular. Aunque el movimiento articular es multiplanar, se puede definir en 3 planos anatómicos, con los grados máximos de giro:

- Flexión 130° / Extensión 20°
- Aducción 30° / Abducción 50°
- Rotación Interna 40°/ Rotación Externa 50°

### 2.3.2. ARTICULACIÓN DE LA RODILLA

La rodilla es la articulación más grande del cuerpo humano y se compone de un mecanismo complejo. Anatómicamente consiste en dos articulaciones en una: la unión entre fémur y tibia y la unión entre el fémur y la rótula. Entre los dos huesos de la primera unión existe unas formaciones cartilaginosas, los meniscos, que mejoran la distribución de carga de la articulación.

El movimiento ocurre principalmente en el plano sagital. A medida que la flexión de la rodilla aumenta, la dureza en el plano frontal y transversal se reduce, por lo que se alcanza un máximo de movimiento y los otros planos cuando en el plano sagital existe una flexión de 90°. La rotura de ligamentos o tendones aumentan el movimiento en los demás planos. Los tipos de movimiento son los siguientes:

- Flexión 150° / Extensión 0°
- Valgus / Varus
- Rotación Interna 10°/ Rotación Externa 40° (Flexión de 90°)

### 2.3.3. ARTICULACIÓN DEL TOBILLO

Esta articulación conecta la parte inferior de la pierna con el pie, o lo que es lo mismo, la tibia y el peroné con los huesos del pie. Tiene forma elipsoidal y el ángulo de referencia para esta articulación es de 90°. El tobillo permite dorsiflexión y plantaflexión, por lo que los tipos de movimiento que permite esta articulación son los siguientes:

- Dorsiflexión 30° / PlantaFlexión 50°
- Eversión 15° / Inversión 30°
- Rotación del pie 20°

### 2.3.4. MOVIMIENTO DEL PIE

Para el análisis de la marcha el pie se simplifica bastante, ya que es un complejo sistema de huesos y músculos. Se entiende como un solo segmento conectado a la parte inferior del pie como mediante el tobillo.



### 3. CICLO DE LA MARCHA

#### 3.1. DEFINICIÓN

La marcha humana es el proceso por el cual el cuerpo humano puede transportarse usando movimientos coordinados de las extremidades inferiores. Este proceso requiere la coordinación del aparato locomotor y del sistema nervioso central como ya dijimos previamente para convertir el movimiento vertical en un movimiento hacia delante de la cabeza y el tronco. Muchos fallos en estos sistemas producen alteraciones en la marcha, por lo que estudiando este proceso podemos detectar estas afecciones.

La marcha es un proceso repetitivo donde cada extremidad realiza un mismo movimiento una y otra vez. Esta cualidad permite estudiar la marcha como la repetición de un mismo movimiento por lo que simplifica el proceso del estudio. El término que se utiliza para describir este movimiento repetitivo se denomina ciclo de la marcha.

#### 3.2. FASES Y EVENTOS

Por convenio, el momento que se utiliza para delimitar el inicio y por tanto el final del ciclo es cuando el pie golpea el suelo, tras la fase de balanceo. Una vez definido el inicio y el final del ciclo se definen las fases y eventos que este produce. Existen dos fases principales:

##### 3.2.1. FASE DE ESTANCIA

Se define como la parte donde alguna parte del pie está en contacto con el suelo. Empieza cuando se produce el primer contacto del pie con el suelo y se acaba cuando este despegue. Suele ocupar el primer 60% del tiempo del ciclo.

Si tomamos como referencia, por ejemplo, el ciclo de la pierna derecha, el despegue de la pierna izquierda ocurre sobre el 10% del ciclo referencia y el aterrizaje ocurre sobre el 50% del ciclo. Esto significa que durante dos periodos del tiempo del ciclo existe un apoyo doble donde ambas piernas están tocando el suelo con alguna parte del pie. Cada periodo de doble apoyo ocupa un 10% del ciclo.

El primer periodo de doble apoyo ocurre al inicio del ciclo, desde que el pie referencia apoya el pie hasta que el pie contrario despegue. Este periodo se caracteriza por una respuesta de carga y es un periodo de desaceleración cuando el impacto es absorbido y el peso del cuerpo se transfiere de una extremidad a la otra.

Después sigue un periodo de estancia simple del pie de referencia donde el pie contrario se encuentra en el aire y que ocupa un 40% del tiempo del ciclo. Este periodo de estancia simple puede ser dividido en estancia media y estancia final. La estancia media ocupa la primera parte de la estancia, del 10% al 30% del ciclo y se caracteriza por ser un periodo donde la progresión del cuerpo se realiza sobre la pierna referencia, por lo que la estabilidad de la parte inferior de la pierna es importante en esta fase. La estancia final ocupa, como su nombre indica, la parte final del ciclo, del 30 al 50% del ciclo. Comienza cuando el pie se pone de puntillas y acaba cuando el pie contrario toca el suelo. Esta fase se caracteriza porque el peso del cuerpo pasa por la pierna referencia.

Después se pasa al segundo periodo de apoyo doble, también llamado pre-balanceo. Comienza sobre la mitad del ciclo y termina con el despegue del pie de referencia. Es equivalente a la primera fase, pero, al contrario, por lo que la respuesta de carga se produce por la pierna contraria en un mismo intervalo de tiempo.

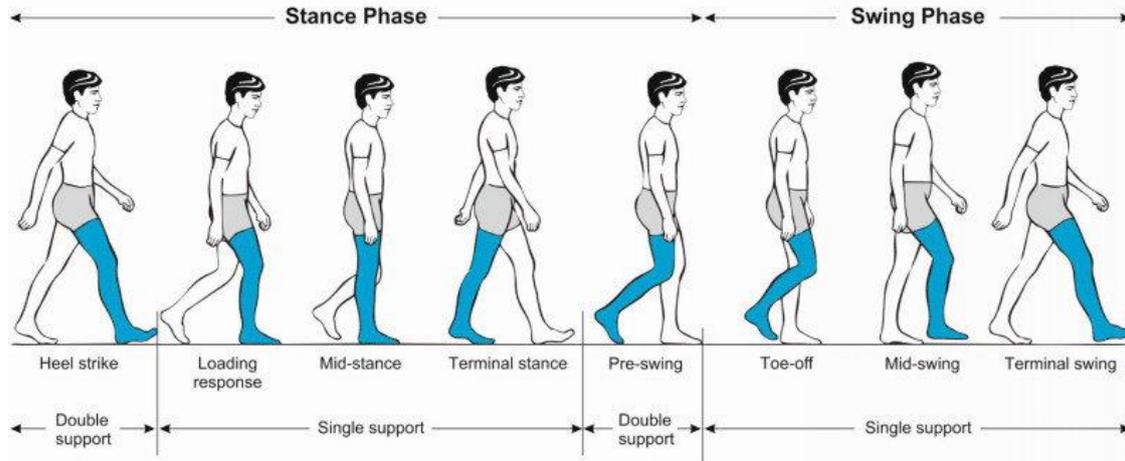


Figura 6. Fases de la marcha (Pirker & Katzenschlager, 2016)

### 3.2.2. FASE DE BALANCEO

Por otro lado, esta fase se define como la parte donde el pie no está en contacto con el suelo, por lo que es la fase que sigue a la anterior y finaliza el ciclo. Se suele dividir en tres fases:

El balanceo inicial ocurre del 60 al 73% del ciclo y es un periodo donde la extremidad avanza. Comienza con el inicio del balanceo y termina cuando el pie que se balancea se sitúa a la altura de la pierna contraria. El balanceo medio ocurre del 73% al 87% del ciclo y termina cuando la extremidad que balancea tiene la parte inferior vertical respecto al suelo. Por último, se encuentra el balanceo final que es la fase final del ciclo y es un periodo de desaceleración y preparación para el contacto con el suelo.

Todas estas fases ocurren para cada pierna, por lo que es necesario tener claro que el proceso se está repitiendo en cada pierna en diferente fase. Este proceso repetitivo dura menos de un segundo

### 3.3. PARÁMETROS ESPACIO TEMPORALES

Una vez definidos las partes del ciclo se pueden definir los parámetros espacio temporales que también ayudan a definir la caracterización de la marcha:

- **Longitud del paso:** Distancia longitudinal entre dos pasos expresada en metros (m). Distancia desde el punto de contacto del pie con el suelo y el lugar donde esa misma parte de ese mismo pie toca el suelo en el siguiente paso. La longitud paso derecho se define como la distancia media desde el tacón del paso izquierdo hasta el tacón del paso derecho mientras que la longitud del paso izquierdo se define como la distancia media desde el tacón derecho hasta el tacón izquierda del siguiente paso
- **Duración del paso:** El tiempo medio que se tarda en dar un paso, se mide en segundos
- **Longitud de la zancada:** Distancia cubierta durante un ciclo de la marcha completo y representa la suma tanto de paso derecho como del izquierdo. Se extiende desde el contacto inicial de una pierna hasta el contacto inicial siguiente de la misma pierna.
- **Anchura del paso:** Se define como la distancia medio-lateral entre los pies

- **Cadencia:** Es el número de pasos dados durante un determinado intervalo de tiempo
- **Velocidad de paso:** Distancia recorrida entre tiempo empleado. Suele expresarse en m/s para la marcha

### 3.4. SEGMENTOS, PLANOS, EJES Y MOVIMIENTOS

Una vez visto las fases del ciclo de la marcha y los parámetros más importantes que se puede medir en él, en esta sección se abordarán las diferentes partes en que se divide el cuerpo durante la captura de movimiento, los planos y ejes más destacados de movimiento y los ángulos que las articulaciones forman sobre ellos en cada ciclo.

#### 3.4.1. SEGMENTOS

Los segmentos son partes solidas o rígidas en las que se puede dividir el cuerpo para simplificar el análisis de la marcha. Para el estudio de la marcha simplificado se suele dividir la parte inferior del cuerpo en siete segmentos: la pelvis, las partes superiores de las piernas (muslos), las partes inferiores de las piernas y los pies.

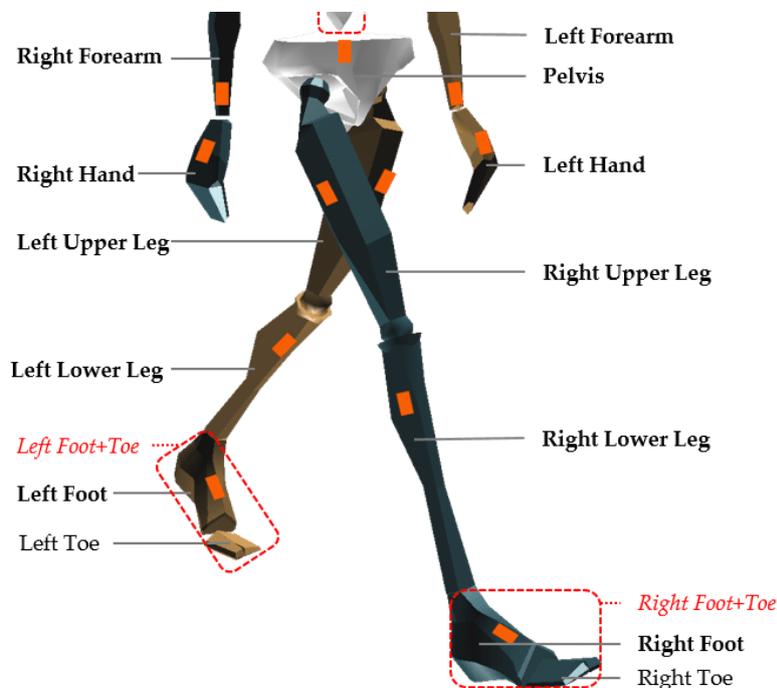


Figura 7. Segmentos de la parte inferior del cuerpo para la captura (Karatsidis et al., 2016)

#### 3.4.2. PLANOS Y EJES DE MOVIMIENTO

El movimiento de las extremidades es generalmente descrito usando los siguientes planos de referencia (Diferentes Ejes, Planos Y Movimientos Del Cuerpo, 2022):

- **Plano sagital:** Divide el cuerpo o segmento en dos lados izquierdo o derecho. El movimiento que producen las extremidades inferiores en este plano implica la inclinación anterior o posterior de la pelvis, los movimientos de flexión/extensión de la cadera y la rodilla y los movimientos plantares/dorsiflexión del tobillo.
- **Plano frontal:** Divide el cuerpo o segmento en parte frontal y parte trasera. El movimiento en este plano implica oblicuidad de la pelvis, la abducción/aducción de la cadera y varo/valgo de la rodilla.

- **Plano transversal:** Divide el cuerpo o segmento en parte superior o inferior. El movimiento de las extremidades inferiores en este plano incluye la rotación hacia delante / hacia atrás en la pelvis y la rotación interna y externa de la cadera y de la tibia y por último el giro del pie.

Y, por otra parte, los ejes son los siguientes (Diferentes Ejes, Planos Y Movimientos Del Cuerpo, 2022):

- **Eje anteroposterior:** Tiene una dirección de atrás hacia adelante por lo que se encuentra perpendicular al plano frontal.
- **Eje vertical:** Se dirige de arriba hacia abajo y es perpendicular al plano horizontal.
- **Eje transversal:** Por último, este eje se ubica de lado a lado y tiene como perpendicular el plano sagital

#### 3.4.3. MOVIMIENTOS DE LAS ARTICULACIONES

Ahora, después de ver los movimientos y los ejes, listaremos los movimientos más comunes que se desarrollan en un plano de los descritos y con relación a un eje de los vistos antes. Esto movimientos son los que la aplicación visualizará en función del ciclo (Diferentes Ejes, Planos Y Movimientos Del Cuerpo, 2022):

- **Flexión y extensión:** Se desarrolla en el plano sagital, por lo que es la modificación del ángulo formado por los huesos de una articulación. Son movimientos opuestos entre ellos. Si se realiza uno se debe realizar el otro para volver a la posición inicial.
- **Abducción y aducción:** Estos movimientos se desarrollan en el plano frontal, por lo que cuando se realiza una abducción el cuerpo se aleja de la zona media y con una aducción el cuerpo se acerca a la zona media.
- **Rotación:** Este tipo de movimiento se realiza en el plano horizontal, por lo que si una extremidad se aleja de la zona media rotando lateralmente se denomina rotación externa y, por otro lado, si un elemento se acerca, de llama rotación interna.
- **Circunducción:** Este movimiento es una suma de los anteriores movimientos



## 4. SISTEMAS DE CAPTURA DE MOVIMIENTO

La captura de movimiento consiste en registrar el movimiento de personas u objetos y puede realizarse con distintas tecnologías. La información se transfiere a un elemento de control, generalmente un software ejecutado en un servidor u ordenador que permite la visualización y análisis de esta información generada.

Se ha vuelto más popular en estos últimos años porque permite generar caracteres y movimientos en 3D con mayor naturalidad frente a los creados en su totalidad por ordenador. Este proceso requiere la captura del movimiento de la persona u objeto varias veces por segundo para que el movimiento generado en el software sea natural para el ojo humano (Teseo Noticias, 2023)

### 4.1. APLICACIONES

Como se puede prever, la captura de movimiento se puede usar en muchas aplicaciones y ámbitos como los siguientes (Captura De Movimiento | Introducción a La Captura De Movimiento | Adobe, 2023):

- **Creación de escenas para películas o series:** Este es el ejemplo más conocido. Se suele usar en la previsualización durante la preproducción. Los datos recogidos con la captura de movimientos ayudan a generar el movimiento y los efectos visuales (VFX).
- **Creación de escenas y movimiento en los video juegos:** Es la forma más efectiva para un desarrollador de generar una librería de movimientos.
- **Creación de movimientos realizando deporte y acciones cotidianas:** Es una forma efectiva para detectar enfermedades o lesiones y también sirve para ayudar con la rehabilitación. Esta es la aplicación donde incide este proyecto, ya que la captura de movimiento de la marcha ayuda a prevenir lesiones o detectar enfermedades como ya se ha comentado previamente.
- **Creación de entrenamientos de uso militar:** Permite mejorar el entrenamiento militar junto con la realidad virtual

### 4.2. TIPOS DE SISTEMAS

La captura de movimiento se puede realizar a través de diferentes sistemas. A continuación, veremos los tipos de sistemas más conocidos:

#### 4.2.1. SISTEMAS ÓPTICOS

Estos sistemas usan un conjunto de cámaras que captan el movimiento del sujeto y un software que interpreta el movimiento grabado por estas cámaras. De esta forma el actor, persona u objetos, se sitúa frente a las cámaras y estas registran los movimientos producidos.

La principal ventaja de estos sistemas frente a otros es su facilidad y comodidad ya que no requiere al actor llevar consigo elementos pegados al cuerpo. Por otro lado, su principal desventaja es que tiene menor precisión ya que los movimientos se pueden ver afectados por la falta de luz o contraste en la cámara.

Se destaca el sistema VICON como el más famoso ejemplo de este tipo de sistema, que se entrará más en detalle posteriormente.

#### 4.2.2. SISTEMAS INERCIALES

En este sistema el actor lleva sensores inerciales, como acelerómetros o giroscopios que generan y transmiten datos al equipo maestro. No es necesario el uso de cámaras para esta técnica. Como se puede deducir, esta es la forma con la que se han medido los diferentes movimientos generados y guardados en los ficheros que usa el software del proyecto.

Como principal ventaja de estos sistemas se destaca su gran precisión de captura frente a los sistemas ópticos. Como principal desventaja, en cambio, es su precio elevado y que son más incómodos al tener que usar sujeciones o trajes para situar los sensores inerciales.

Destacan los sistemas XSENS como los más famosos de este tipo. También se estudiará más adelante en el documento.

#### 4.2.3. SISTEMAS MAGNÉTICOS

En este tipo de sistemas se utilizan sensores magnéticos que miden el campo magnético generado por unos marcadores situados en el actor. Estos marcadores suelen ser bobinas capaces de generar un campo magnético. Los sensores magnéticos miden el campo magnético y son capaces de determinar la posición y orientación del actor a través de él.

Tiene las mismas ventajas que los sistemas inerciales en cuanto a precisión y dependen de la calidad de los sensores utilizados y de las necesidades de la aplicación a la hora de elegir un tipo de sistema u otro. Como desventajas, también compartidas con los sensores inerciales, son sistemas caros y los marcadores necesitan de algún tipo de sujeción en el actor (Yabukami, 2000)

#### 4.2.4. SISTEMAS BASADOS EN ULTRASONIDOS

Estos sistemas utilizan las ondas ultrasónicas para medir la distancia entre sensores y actores rastreados. Los transmisores de los sensores emiten señales ultrasónicas que rebotan en los actores y son detectados por los receptores y con esa información determinan la distancia.

Este sistema rebaja el coste, ya que los sensores son más sencillos y es adecuado para la captura en interiores. En cambio, son más limitados en términos de precisión (Three Sources, Three Receivers, Six Degrees of Freedom: An Ultrasonic Sensor for Pose Estimation & Motion Capture, 2015).

### 4.3. VENTAJAS Y DESVENTAJAS

A continuación, se listan las ventajas y las desventajas de esta tecnología. Se empieza comentando las ventajas (Captura De Movimiento | Introducción a La Captura De Movimiento | Adobe, 2023):

- Baja latencia de resultados que reduce el coste de la animación
- La cantidad de trabajo no depende de la complejidad o de la longitud
- Interacciones y movimientos realistas y complejos se pueden recrear fácilmente
- Grandes cantidades de datos animados

Por su parte, en cuanto a las desventajas existen las siguientes (Captura De Movimiento | Introducción a La Captura De Movimiento | Adobe, 2023):

- Software y hardware específico para obtener los datos
- Coste elevado del equipamiento y del software
- El campo de visión de la cámara o la distorsión magnética afectan a la calidad de la captura

- Si existen problemas en la captura, pocos sistemas permiten una visión para decidir si se debe realizar de nuevo la captura. Lo más fácil es rehacerla

Ahora se verán los sistemas más conocidos de cada uno de los tipos de captura de movimientos. Para los sistemas ópticos veremos el sistema óptico basado en cámaras de VICON ampliamente utilizado en la industria del cine, la animación, los videojuegos y la investigación. Para los sistemas inerciales, en cambio, se verán el sistema XSENS.

#### 4.4. SISTEMA VICON

##### 4.4.1. INTRODUCCIÓN

Es un sistema comercial que registra el movimiento humano gracias a la fotografía en 3D. Utiliza un conjunto de cámaras infrarrojas que son sensibles a la luz que envían unos marcadores reflectantes. Estos marcadores se sitúan en puntos estratégicos del cuerpo humano como las articulaciones o segmentos importantes (Bernad, 2021).



Figura 8. Logo de VICON (Vicon | Award Winning Motion Capture Systems, 2023)

Un sistema VICON es un sistema óptico avanzado de captura de movimientos. El sistema se compone de cámaras, el módulo hardware de control, el software para el análisis y la visualización de datos y el equipo para ejecutar dicho software.

##### 4.4.2. COMPONENTES Y CONFIGURACIÓN DEL SISTEMA

A continuación, se listan y se definen los principales componentes y modelos de los sistemas de captura de movimiento con cámaras de VICON:

- **Cámaras:** En cuanto a las cámaras VICON, se ha realizado un estudio sobre las especificaciones técnicas más importantes de los diferentes modelos que ofrece VICON. Las especificaciones técnicas se listan a continuación (Vicon | Award Winning Motion Capture Systems, 2023):
  - o Resolución: Cantidad de píxeles en la imagen capturada
  - o Velocidad de captura: Cantidad de *frames* por intervalo de tiempo
  - o Rango de captura: Espacio físico dentro del cual la cámara puede detectar y rastrear los puntos de referencia
  - o Precisión: Capacidad de la cámara para medir y registrar la posición y orientación de los puntos de referencia
  - o Sincronización: Capacidad de las cámaras de capturar movimientos de manera coordinada
  - o Sensibilidad: Capacidad de la cámara para capturar imágenes claras y nítidas

Por otro lado, los modelos más destacados son los siguientes

- o Valkyrie: Es la cámara de captura de movimiento más potente del mercado. Se ofrece con cuatro especificaciones diferentes con cuatro enfoques: resolución,

velocidad, valor y sin efecto estroboscópico (Vicon | Award Winning Motion Capture Systems, 2023).

Existen cuatro modelos con las siguientes características técnicas:

Modelo	VK26	VK16	VK8	VKX
Velocidad de captura (frames por segundos, fps)	150	240	500	380
Resolución (Megapíxeles, MP)	26.2	16.1	8.0	7.2
Campo de visión horizontal (ancho, H FoV)	72	72	72	66
Campo de visión horizontal (ancho, V FoV)	72	56	42	66
Campo de visión horizontal (estrecho, H FoV)	54	54	54	54
Campo de visión horizontal (estrecho, V FoV)	54	41	30	54
IP65	SI	SI	SI	NO

Tabla 2. Características técnicas de las cámaras Valkyrie (Vicon | Award Winning Motion Capture Systems, 2023)

- Vero: Una buena oferta de resolución/velocidad/precio. Con sensores integrados que controlan la posición de la cámara y la temperatura para facilitar un rendimiento óptimo en todo momento (Vicon | Award Winning Motion Capture Systems, 2023).

Modelo	v2.2	v1.3	V1.3 X
Velocidad de captura (frames por segundos, fps)	330	250	250
Resolución (Megapíxeles, MP)	2.2	1.3	1.3
Lentes estándares (milímetros)	6-12 Vari focal	6-12 Vari focal	-
Campo de visión horizontal (FoV) mínimos estándar (HxV)º	44.1 x 23.6 (tele)	27.2 x 21.8 (tele)	-
Campo de visión horizontal (FoV) mínimos ancho (HxV)º	98.1 x 50.1 (wide)	55.2 x 43.9 (wide)	79.0 x 67.6
Latencia (milisegundos)	3.6	3.4	3.4

Tabla 3. Características técnicas de las cámaras Vero (Vicon | Award Winning Motion Capture Systems, 2023)

- **Vantage:** Hermano mayor de la Vero. Proporciona alta resolución y velocidad para capturar el movimiento con gran precisión y detalle. Dependiendo de las lentes que usan los diferentes modelos estos tienen unas especificaciones diferentes (Vicon | Award Winning Motion Capture Systems, 2023).

Modelo	V16	V8	V5
Frames por segundos (FPS)	120 (resolución máxima) 2000 (resolución parcial)	260 (resolución máxima) 2000 (resolución parcial)	420 (resolución máxima)
Resolución (Megapíxeles, MP)	16	8	5
Lentes estándares (milímetros)	51	42	
Angulo de visión (AoV) <sup>º</sup>	54.22 x 54.22 (VICON 18 mm) 72.8 x 72.8 (VICON 12 mm)	61.36 x 47.15 (Ricoh Pentax 12.5 mm)	47.28 x 40.47 (Ricoh Pentax 12.5 mm) 65.54 x 56.93 (Ricoh Pentax 8.5 mm)
Latencia (milisegundos)			

Tabla 4. Características técnicas de las cámaras Vantage. (Vicon Motion Systems, 2016).

- **Viper:** Estas cámaras tienen las siguientes características

Modelo	Viper	Viper X
Frames por segundos (FPS)	240	240
Resolución (Megapíxeles, MP)	2.2	6.3
Rango (metros)	8	8
Lentes estándares (milímetros)	6.5	12.5
Campo de visión horizontal (FoV) mínimos estándar (HxV) <sup>º</sup>	81.8 x 49.4	50.2 x 50.2
Latencia (milisegundos)	3.2	3.2

Tabla 5. Características técnicas de las cámaras Viper (VICON, 2019)

- **Módulo hardware de control:** El sistema básico se compone de al menos un módulo hardware de control que proporciona energía y comunicación a los demás componentes del sistema. Las cámaras envían la información a este componente. Dependiendo de la cantidad de cámaras que envíen información se necesitarán más de un módulo hardware.

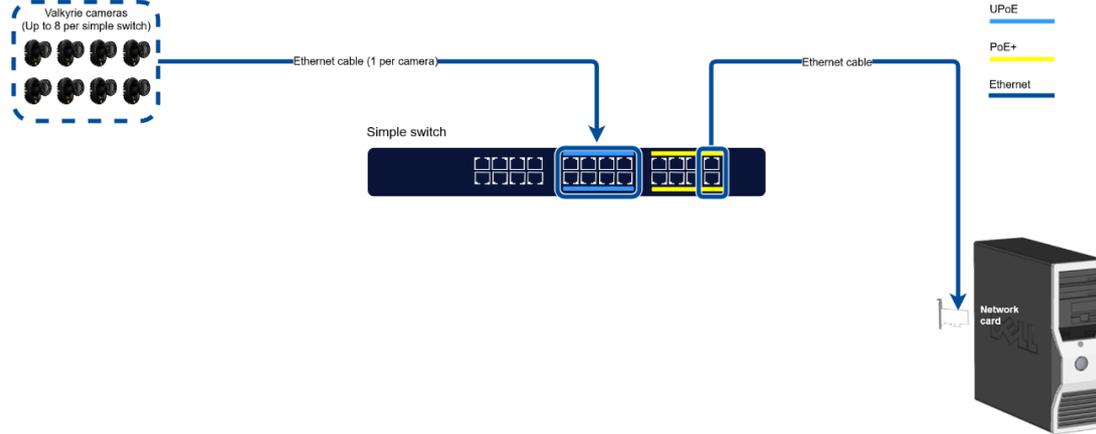


Figura 9. Sistema simple VICON

En la figura se puede apreciar el sistema VICON más sencillo, con 8 cámaras y un switch simple que actúa como módulo hardware de control. También se establecen las conexiones entre los componentes. Por último, tenemos el servidor (Vicon | Award Winning Motion Capture Systems, 2023).

- **Servidor:** El equipo que ejecuta el software de análisis y visualización de datos puede ser un servidor con la suficiente capacidad para procesar la cantidad de datos que el sistema genera. El hardware de control se conectará con él para enviar los datos generados por las cámaras. Cuantas más cámaras se conecten al sistema, mayor cantidad de datos generará el sistema y mayor capacidad de procesamiento se necesitará en el equipo (Vicon | Award Winning Motion Capture Systems, 2023).
- **Software:** Este sistema está diseñado bajo los principios de flexibilidad, escalabilidad y facilidad de integración. El software es flexible porque permite crear sistemas de diferente tamaño dependiendo de los componentes que se tenga. El software es escalable porque se pueden incorporar más componentes si así se desea. El software es fácil de integrar entre sí.

VICON contiene cuatro tipos diferentes de software según las necesidades de la aplicación (Vicon | Award Winning Motion Capture Systems, 2023):

- **Tracker:** Es un software de captura de movimientos que permite seguir todo tipo de movimientos en objetos, personas, robots, ... a través de la cámara. Permite capturar datos en tiempo real con una gran calidad, por lo que tiene su objetivo servir para proyectos de ingeniería. Combinado con las cámaras de última generación permite realizar seguimiento en entornos exteriores y a mayor cantidad de objetos. Permite analizar movimientos a mayores velocidades, por lo tanto, se generan datos con mayor calidad.
- **Capture.U:** Sirve para capturar movimientos y analizar datos en tiempo real con sensores inerciales. Está disponible tanto para MAC como para Windows.
- **Nexus:** Es el software de captura de movimientos y análisis de datos más conocido. Permite la configuración del sistema, capturar datos en tiempo real, poder procesar los datos y poder analizar los movimientos generados. Contiene una herramienta para la sincronización y la calibración del sistema. Es capaz de

sincronizar los datos inerciales con los datos ópticos, por lo que no deja de ser una combinación de los anteriores softwares.

- **Shogun**: Este software que sirve para dar otra dimensión a la captura de efectos visuales (VFX, en inglés). Es útil para la industria del entretenimiento para la generación de animaciones.

#### 4.4.3. APLICACIONES

Los principales usos de los sistemas VICON son los siguientes (Vicon | Award Winning Motion Capture Systems, 2023):

- **Ciencias de la biología**: El uso de los sistemas VICON es conocido y utilizado en el deporte, en la neurociencia, en el análisis de la marcha y en la ciencia animal. El software Nexus tiene herramienta optimizadas para ello. Las cámaras *Valkyrie* tienen un gran rendimiento como hemos dicho antes para este caso. *Capture.U* también tiene muchas aplicaciones enfocadas a las ciencias de la biología como un modelo de esqueleto o procesamiento en tiempo real. Marcas conocidas que han trabajado con VICON son Asics, Adidas, la universidad de Duke o la de Cardiff.
- **Efectos visuales**: También es conocido su uso para realizar efectos especiales como en producciones como la serie *The Mandalorian* de *Disney+*. Una prueba del éxito en este sector es que tiene un software dedicado para la realización de efectos visuales como se ha dicho previamente con *Shogun*. Algunos estudios conocidos que han usado los sistemas VICON para producir estos efectos son *Epic Games*, *Industrial Light & Magic*, *Framestore* o *Silverspoon*.
- **Entretenimiento**: Desde que se realizó la primera película en usar captura de movimiento en 1995, VICON ha estado involucrado en la industria del entretenimiento, ya sea para la realización de películas como la realización de videojuegos. Como se ha dicho previamente con los efectos visuales creados con *Shogun*, muchos estudios de producción tanto de películas como *DreamWorks* o *Disney* o de videojuegos como *Activision*, *Sony* o *Ubisoft* han usado los sistemas VICON para la creación de sus productos.
- **Ingeniería**: Muchas compañías de Ingeniería también han usado estos productos para generar sus productos, ya que VICON proporciona una solución robusta para ello. Ya sea para rastrear vehículos, evaluar diseños de productos, localización de herramientas o analizar el movimiento de edificios reales, compañías espaciales tan famosas como la *NASA*, la *ESA* o compañías automovilísticas como *FIAT*, *BMW* o *Toyota* han utilizado estos sistemas por comentar unos ejemplos.
- **Realidad virtual**: Esta tecnología aun en desarrollo también ha sido colonizada por VICON ya que los sistemas de seguimiento del movimiento en la realidad virtual han sido complejos y costosos. Con la experiencia de VICON estos sistemas son menos complejos, por lo que empresas pujantes en este sector como *Oculus* o *Samsung* han usado la tecnología de VICON para conseguir sus objetivos

#### 4.5. SISTEMAS XSENS

##### 4.5.1. INTRODUCCIÓN

Los sistemas XSENS representan la solución más conocida de la captura de movimientos inercial. Como se ha explicado previamente, estos sistemas usan sensores inerciales sujetos en el cuerpo del actor, como acelerómetros o giroscopios que mandan los datos a un sistema de control.



Figura 10. Logo de XSENS

Como características principales de este sistema se puede decir que permite la captura de movimiento en cualquier tipo de entorno, exterior o interior. También permite una rápida implementación ya que es un sistema sencillo de desplegar, ya que solo se necesita situar los sensores y calibrarlos. Permite obtener un gran detalle en los datos por lo que se puede obtener una gran precisión en variedad de movimientos (Home | Movella.com, 2023).

#### 4.5.2. COMPONENTES DEL SISTEMA

A continuación, se listan los componentes pertenecientes al sistema XSENS

- **Hardware (Sensores inerciales):** Existen tres posibilidades de sistemas hardware para XSENS (Home | Movella.com, 2023):
  - o MVN Awinda Starter: La solución más sencilla y económica que ofrece XSENS para la captura de movimiento. Se compone de 17 sensores inalámbricos, que se pueden situar sobre el cuerpo de actor mediante cintas y que envían sus datos al sistema de control. Los 17 sensores inerciales permiten cubrir todo el cuerpo y tienen una autonomía de 6 horas. Es un sistema robusto, preciso y se puede usar en cualquier entorno, ya sea interior o exterior ya que tiene inmunidad magnética. Está optimizado para una frecuencia de 60Hz, con un rango máximo de 25 metros de distancia entre sensores y sistema de control.
  - o MVN Awinda: Sistema similar al anterior en cuanto a número de sensores, a su autonomía, a su frecuencia de trabajo. Cuenta a mayores con un sensor de propagación que permite aumentar la distancia con el receptor de hasta más de 50 metros por lo que le hace susceptible de un mayor número de aplicaciones.
  - o MVN Link: Por su parte, este sistema es el más potente de los 3 que aquí se describen. Esto es así ya que, aunque contiene la misma cantidad de sensores que los anteriores sistemas, que permite capturar el movimiento del cuerpo entero, permite una frecuencia de captura de datos de 240Hz, 4 veces superior a los 60Hz utilizados en *Awinda*. Además, triplica en rango máximo al anterior con 150m de distancia y la batería de los sensores dura hasta 4 horas más que los de *Awinda*, llegando a las 10 horas de duración.

Features	MVN Awinda starter	MVN Awinda	MVN Link
Sensors	17 wireless sensors	17 wireless sensors (plus 1 prop sensor)	17 wireless sensors (plus 1 prop sensor)
Clothing	Full body strap set. This includes 3 t-shirts (sizes S, L, XXL) and a strap set	Full body strap set. This includes 3 t-shirts (sizes S, L, XXL) and a strap set	Full body Lycra suit available in 6 sizes (S-4XL)
Update rate	Up to 60 Hz	Up to 60 Hz	Up to 240 Hz
Range	25m	60m	150m
Battery life	6 hours	6 hours	10 hours

Figura 11. Características técnicas de cada hardware

Todos estos sistemas, como se ha comentado tienen diecisiete sensores inerciales que permiten cubrir la totalidad del cuerpo y proporcionan datos de gran calidad. Elegir cualquiera sobre el otro dependerá de las características de la aplicación y el presupuesto que se tenga. En la figura se muestra una tabla resumen de las características de estos sistemas.

- **Software:** XSENS también nos ofrece otras dos posibilidades en cuanto al apartado software (Home | Movella.com, 2023):
  - **MVN Animate:** Este software está destinado para la industria del entretenimiento, como la realización de películas, series, videojuegos. Ofrece herramientas para animar varias siluetas a la vez en tiempo real y en 3D. Permite poder conectarse con sistemas externos y varios formatos para la exportación de datos. Puede llegar a soportar flujos de datos superiores a 158 MB/min. XSENS *MVN Animate* ofrece tres posibilidades dependiendo de las necesidades del cliente al igual que se ha comentado antes con las posibilidades del hardware:
    - **MVN Record:** Es un software completamente libre que permite grabar ficheros de captura de movimientos de un actor. También permite reproducir ficheros en la aplicación. Tiene opción de poder conectarse con la nube de Xsens bajo suscripción, lo que permite usar las herramientas que esta tiene. Este software está optimizado si se tienen varios sistemas en varias ubicaciones.
    - **MVN Animate plus:** Software más avanzado que el anterior, tiene todas sus ventajas y a mayores permite hasta la captura de dos actores y la representación de los datos en tiempo real en softwares motores de juegos como *Unreal Engine, Unity, Maya*, etc. Este software permite que la aplicación remota de MVN lo pueda usar, permitiendo el control remoto.
    - **MVN Animate pro:** Por último, este software es la versión más avanzada de XSENS en este apartado. Contiene todas las herramientas del software anterior. Permite la captura de movimiento de hasta cuatro actores diferentes a la vez en tiempo real y la reproducción de movimiento de infinitos actores.

Otro apartado destacado de este software son las integraciones que permite, destinadas al mundo del entretenimiento. A continuación, se comentan las más destacadas:

- Unreal Engine: Este motor permite animar caracteres en tiempo real con los datos recibidos de la captura de movimiento. Se puede interactuar mediante el carácter creado con el entorno virtual. También permite exportar e importar archivos FBX.
- Unity: Utiliza un plugin creado por Xsens para el incluir los datos en tiempo real, lo que en conjunto permite ser una herramienta muy potente al igual que el motor anterior para interactuar con el entorno virtual. También soporta la importación de ficheros de captura de movimientos.
- *Motionbuilder, Maya, 3D Studio Max, Houdini, Cinema 4D, StretchSense, Pixotope, iClone, Blender, Notch, Mo-Sys StarTrackerVFX, StypeLand y Moveshelf* son otras opciones donde se puede integrar *MVN Animate* de XSENS.
- MVN Analyse: Este software en cambio, está destinado a la ciencia del deporte, a la rehabilitación, a la biomecánica, a la ergonomía. Por lo tanto, su punto clave es que la precisión es crucial para un estudio correcto. Este software permite grabar e investigar los datos de los sistemas hardware. Para ello, proporciona datos super precisos, permite la visualización en tiempo real de toda la información cinemática del cuerpo como articulaciones, segmentos, orientaciones y centros de masa. Además, proporciona un modelo biomecánico escalable y permite su calibración de varias formas. XSENS *MVN Analyse* ofrece tres posibilidades también dependiendo de las características de la aplicación similares a *MVN Animate*:
  - *MVN Record*
  - *MVN Analyse Plus*
  - *MVN Analyse Pro*

Escoger entre una suscripción u otra dependerá de las características que tenga la aplicación. Por ejemplo, dependiendo del número de actores, ya que si se usa un actor lo recomendable es usar la suscripción más simple, *MVN Record*, pero si se tienen más actores la cosa cambia y es recomendable usar *MVN Analyse*.

Dependiendo del lugar donde se quiera reprocesar los ficheros de captura de movimiento. Si se quiere realizar fuera de la nube, la mejor opción en este caso es *MVN Analyse Pro*, ya que es el único de las tres opciones que tiene motor de reprocesamiento integrado. La otra opción es realizarlo en la nube de XSENS, donde cualquiera de las suscripciones puede.

El tipo de formato que en el que se requieran los datos también es importante. No se pueden exportar los datos con las dos suscripciones más sencillas (*MVN Record* y *MVN Analyse Plus*) directamente. Se necesita de la nube de XSENS para exportar ficheros MVNX, FBX o BVHh. Por otro lado, la suscripción profesional permite la exportación en estos formatos y en XLSX y C3D a mayores.

Otra característica para tener en cuenta es si se quiere realizar la visualización en tiempo real. Las tres opciones proporcionan esta característica, pero *MVN Analyse Plus* y *Pro* permite además la visualización de articulaciones, de la aceleración y del centro de masas en tiempo real.

Ahora se pasará a ver las integraciones más famosas que tiene este software al igual que se hizo con el anterior software. *MVN Analyse* permite la integración en los softwares

biomecánicos y ergonómicos más importantes como *Visual3D*, *OpenSim*, *Unity*, *Delsys* y muchos más.

Features	MVN Awinda starter	MVN Awinda	MVN Link
Sensors	17 wireless sensors	17 wireless sensors (plus 1 prop sensor)	17 wireless sensors (plus 1 prop sensor)
Clothing	Full body strap set. This includes 3 t-shirts (sizes S, L, XXL) and a strap set	Full body strap set. This includes 3 t-shirts (sizes S, L, XXL) and a strap set	Full body Lycra suit available in 6 sizes (S-4XL)
Update rate	Up to 60 Hz	Up to 60 Hz	Up to 240 Hz
Range	25m	60m	150m
Battery life	6 hours	6 hours	10 hours

Figura 12. Características técnicas de los sistemas software

#### 4.6. COMPARATIVA VICON VS. XSENS

En este último apartado del estudio de la captura de movimiento se realizará una comparativa entre los sistemas que se han visto. Esta comparativa se realizará en términos de precisión y calidad de la captura, de usabilidad y portabilidad, de las aplicaciones y entornos de uso más adecuados y de costo y requisitos técnicos.

Por tanto, en términos de precisión y calidad de la captura se puede comentar que los sistemas XSENS son menos precisos que los sistemas Vicon. Los sistemas VICON son el *gold-standard* frente a los demás sistemas, entre ellos los XSENS. Aunque ciertos estudios, el error entre ambos sistemas puede ser bastante pequeño, inferior a 5º RMSE (Robert-Lachaine et al., 2016)

Las correlaciones cruzadas de diferentes parámetros de la marcha, es innovador que se pueda observar y analizar sistemáticamente los ángulos de las articulaciones en 3D en cada movimiento de manera independiente. Ambos sistemas tienen una concordancia bastante similar en términos de amplitud (Benjaminse, 2020).

También es notable que las medidas que los sistemas ofrecen en el plano sagital son bastante parecidas y precisas, pero, por otro lado, las medidas que ofrecen ambos sistemas para los otros planos frontal y transversal son más diferentes (Nijmeijer et al., 2023).

Por último, en la última parte de esta comparativa se va a tratar el tema de los entornos. Los sensores IMU al estar situados sobre el actor, permiten realizar mediciones en cualquier tipo de entorno donde los IMUs se puedan conectar con el sistema de control, por lo que se puede realizar perfectamente en exteriores y variando la localización. Por el otro lado, las cámaras VICON necesitan estar situadas en una misma posición, por lo que preferiblemente se deben realizar en interiores perfectamente adaptados.



## 5. MATERIALES Y MÉTODOS

En este apartado se describirá tanto el lenguaje como el paradigma utilizado para el desarrollo de la aplicación. Además, también se comentará las herramientas tanto hardware, como software, para poder probar y testear la aplicación.

### 5.1. PYTHON

El lenguaje utilizado para la realización de la aplicación ha sido Python debido a su gran variedad de librerías, recursos y comunidades expertas en el tratamiento de datos científicos como son los que generan los sensores inerciales. Por todas estas razones se ha preferido este lenguaje por encima de cualquier otro como puede ser Java o C++ (Mariño, 2023)



Figura 13. Logo de Python (Welcome to Python.org, 2023)

Python es uno de los lenguajes de programación más utilizados en estos momentos y su éxito entre los demás lenguajes de programación se debe a que es un lenguaje interpretado de código abierto, multiparadigma, orientado a objetos de alto nivel y sintácticamente fácil.

Al tratarse de un lenguaje interpretado, que se ejecuta de instrucción a instrucción no necesita de un compilador a mayores. Esto permite no tener que adaptar el código a una plataforma en concreto, pero disminuye la velocidad de ejecución.

Como se ha comentado, es un lenguaje sencillo de programar por su sintaxis sencilla y más cercana a la lógica humana. Resulta en un código más legible de desarrollar y la curva de aprendizaje es menos costosa

Por otro lado, como es un lenguaje de código abierto y propósito general, permite una capacidad de compartición de conocimiento y recursos mayor entre los usuarios de esta herramienta

Además, permite la integración de varios modelos de programación ya que como todo es un objeto, es un lenguaje principalmente orientado a objetos, pero tiene propiedades de otros paradigmas como imperativa, funcional, procedural o reflexiva.

Se destaca también la disponibilidad de un gran número de librerías, como SciPy o Numpy y entornos de desarrollo, como Anaconda o Spyder muy potentes para el análisis y procesado de datos científicos. Se verán más adelante en detalle. También es el lenguaje referencia en Educación por todas las ventajas comentadas previamente (De Los Santos, 2023).

#### 5.1.1. LIBRERIAS DE PYTHON UTILIZADAS

Como se ha comentado, una de las características principales de Python es la cantidad de librerías sobre ciencia de datos que existe, por tanto, para la realización de esta aplicación se han usado unas cuantas de ellas que comentaremos a continuación:

- **Pandas:** Poderosa librería de análisis y manipulación de datos. Sus elementos principales son el objeto Series, un array 1D etiquetado capaz de ser de cualquier tipo de dato y el objeto *DataFrame* que es un array 2D etiquetado capaz de tener columnas de diferentes

tipos (similar a una tabla en SQL). A partir de estos dos objetos, Pandas ofrece muchas funciones para la creación y edición de estos objetos (Intro to Data Structures — Pandas 2.1.0 Documentation, 2023).

- **Plotly:** Librería gráfica que crea gráficos interactivos. Con ella podemos crear gráficos de líneas, de puntos, de áreas, de barras, de errores, de burbujas y polares. También podemos crear histogramas, mapas de calor y ejes múltiples. De esta manera se pueden crear gráficos sencillos o gráficos más complejos, como estadísticos, científicos o financieros. Contiene los siguientes submódulos (Python API Reference for Plotly — 5.17.0 Documentation, 2023):
  - o plotly.express: Interfaz de alto nivel para la visualización de datos
  - o plotly.graph\_objects: Interfaz de bajo nivel para la visualización de datos
  - o plotly.subplots: Métodos de ayuda para la representación de varias gráficas en la misma figura
  - o plotly.figure\_factory: Métodos de ayuda para construir gráficas complejas
  - o plotly.io: Interfaz de bajo nivel para mostrar, leer y crear figuras
  - o plotly.colors: Escala de colores y funciones de utilidad
  - o plotly.data: Datos de ejemplo para demostraciones y testeo

Para esta aplicación se ha usado el submódulo `graph_objects` para crear los gráficos más personalizados (en comparación con el módulo `express`), el submódulo `subplots` para la división de los gráficos en las diferentes figuras y el submódulo `io` para indicar a los gráficos que se tienen que abrir en el navegador (Python API Reference for Plotly — 5.17.0 Documentation, 2023).

- **Numpy;** Esta librería es la fundamental para la computación científica en Python. Proporciona un objeto de array multidimensional, `ndarray`, varios objetos derivados, rutinas y métodos asociados a estos objetos como operaciones matemáticas, lógicas, etc. Las diferencias entre el Numpy array y las listas de Python son las siguientes (What Is NumPy? — NumPy v1.26 Manual, 2023):
  - o El Numpy array tiene tamaño fijo en la creación, mientras que las listas de Python pueden crecer dinámicamente
  - o El Numpy array debe tener todos los elementos del mismo tipo, y las listas de Python no
  - o El Numpy array facilita cálculos de matemática avanzada en un gran número de tipos de datos de una manera más eficiente y simple que las listas de Python
  - o El Numpy array cada vez tiene más uso en paquetes y librerías en Python por su eficiencia y simpleza

El Numpy array es más eficiente y rápido porque vectoriza el código, Esto evita la presencia de bucles, indexación, etc.. en el código, ya que se realiza internamente. Esto tiene la ventaja de que hace al código más conciso y fácil de leer, con menos bugs y más parecido a la notación matemática (What Is NumPy? — NumPy v1.26 Manual, 2023).

- **Scipy:** Librería que contiene algoritmos fundamentales para la computación científica en Python. Proporciona comandos y clases de alto nivel para manipular y visualizar datos. Se organiza en varios subpaquetes diferentes que cubren diferentes dominios (Signal Processing (`scipy.signal`) — SciPy v1.11.2 Manual, 2023):

cluster	<i>Clustering algorithms</i>
constants	<i>Physical and mathematical constants</i>
fttpack	<i>Fast Fourier Transform routines</i>
integrate	<i>Integration and ordinary differential equation solvers</i>
interpolate	<i>Interpolation and smoothing splines</i>
io	<i>Input and Output</i>
linalg	<i>Linear algebra</i>
ndimage	<i>N-dimensional image processing</i>
odr	<i>Orthogonal distance regression</i>
optimize	<i>Optimization and root-finding routines</i>
signal	<i>Signal processing</i>
sparse	<i>Sparse matrices and associated routines</i>
spatial	<i>Spatial data structures and algorithms</i>
special	<i>Special functions</i>
stats	<i>Statistical distributions and functions</i>

Tabla 6. Paquetes de Scipy (Signal Processing (scipy.signal) — SciPy v1.11.2 Manual, 2023)

Se comentará más en detalle el subpaquete *signal* ya que ha sido el único utilizado en la aplicación. Este paquete contiene funciones de filtros, funciones de diseño de filtros y algoritmos de interpolación para 1D y 2D. Para todo esto, se entiende que las funciones son *arrays* de números reales o complejos, preferiblemente Numpy *arrays* (Signal Processing (scipy.signal) — SciPy v1.11.2 Manual, 2023).

- **Seaborn:** Esta librería de visualización de datos está basada en matplotlib y en la estructura de datos de pandas. Proporciona un interfaz de alto nivel para visualizar gráficos estadísticos y de esta forma poder entender los datos de mejor manera. Su operabilidad recae sobre los *Dataframes* y series de Pandas. En la aplicación se ha usado para crear cierto tipo de gráficos en la aplicación (Seaborn, 2023).

## 5.2. PARADIGMA MODELO-VISTA-CONTROLADOR

En el desarrollo de software es muy importante modelar y estandarizar paradigmas que permitan implementar las ideas de una mejor manera. Existen muchos paradigmas que ayudan a resolver diferentes tipos de problemas, entre ellos está el paradigma Modelo-Vista-Controlador (MVC) (Selfa, 2006).

Como su nombre indica, se compone de tres elementos fundamentales cada uno con una función bien remarcada. El modelo es el componente que se encarga del dominio de los datos, la vista se encarga de la parte de la interfaz de usuario y el controlador se encarga de la parte de la lógica de la aplicación (Selfa, 2006).

El plano del modelo contiene objetos con tiempos de vida largos como datos que pueden ser tablas de una base de datos, ficheros o datos de sesión. El plano de la vista contiene objetos que permiten comunicar el sistema con el entorno, como puede ser pantallas, ventanas, menús o cualquier elemento gráfico. Por último, el plano de control contiene objetos que permiten realizar los casos de uso que filtran la información y la muestran al usuario (Selfa, 2006).

Esto permite imponer las siguientes reglas en el desarrollo:

- Un actor solo puede hablar con el plano de la vista

- Un objeto del plano de la vista solo puede hablar con los objetos de control y los actores
- Un objeto del plano de datos solo puede hablar con otro del plano de control
- Un objeto de control puede hablar con los objetos de las vistas y de los datos, pero no con los actores.

Como ventajas de este paradigma podemos destacar que la separación de los elementos de la aplicación en estos tres conceptos ayuda a la simplificación y entendimiento de esta por lo que simplifica el desarrollo de la aplicación, pudiendo separar este desarrollo en 3 partes y, por tanto, pudiendo agilizar el proceso de creación y mantenimiento. También permite la representación múltiple de la misma información al separar el plano de la vista del de la información, por lo que permite tener varias vistas de la misma información sin que esto afecte a los otros componentes. Además, permite la reutilización de código entre los mismos componentes (Selfa, 2006).

### 5.3. HERRAMIENTA SOFTWARE DE DESARROLLO: ANACONDA - SPYDER

Para el desarrollo de la aplicación se ha optado por el uso de Spyder, una herramienta perteneciente a la distribución de Anaconda. La distribución de Anaconda es una distribución de Python/R de ciencia de datos gratuita que contiene (Welcome! — Anaconda Documentation, 2023):

- **Conda** es un entorno y administrador de paquetes a través de la línea de comandos que ayuda a administrar más de 7500 paquetes de código abierto disponibles a través de esta distribución.
- **Anaconda Navigator** es una aplicación de escritorio construida sobre Conda que permite lanzar y administrar las demás aplicaciones de desarrollo
- 250 paquetes con instalación automática que función conjuntamente.
- Acceso al repositorio público de Anaconda con más de 8000 paquetes de código abierto de ciencia de datos y aprendizaje automático.

En el caso del presente trabajo se ha utilizado la aplicación de escritorio *Anaconda Navigator*, que como se ha dicho, funciona como un administrador de las diferentes aplicaciones de la distribución de Anaconda como se puede ver en la imagen ... entre la que destaca el entorno integrado de desarrollo, IDLE por sus siglas en inglés, de Spyder que es el que se ha utilizado para la realización de la aplicación (Welcome! — Anaconda Documentation, 2023)



Figura 14. Logo de Anaconda (Welcome! — Anaconda Documentation, 2023)



Figura 15. Logo de Spyder (Welcome to Spyder's Documentation — Spyder 5 Documentation, 2023)

Spyder, acrónimo de 'Scientific Python Development Environment' (Entorno de desarrollo científico de Python en castellano), es el IDE gratuito integrado en la aplicación *Anaconda Navigator* destinado a científicos, ingenieros y analistas de datos. Permite, entre más cosas, la edición, el testeo, la depuración de programas de Python (Welcome to Spyder's Documentation — Spyder 5 Documentation, 2023).

#### 5.3.1. COMPONENTES CLAVE DE SPYDER

Se puede destacar los siguientes componentes como los más importantes de Spyder (Welcome to Spyder's Documentation — Spyder 5 Documentation, 2023):

- El **editor**: Es el elemento más importante de Spyder. Permite la creación, apertura, edición de los ficheros fuente. Contiene una gran cantidad de herramientas de edición como el análisis en tiempo real del código, que permite el autocompletado, el resaltado de la sintaxis, la separación horizontal y vertical y el formateo automático.
- La **consola de Python**: Permite la ejecución de comandos y la interacción con los datos que existen dentro del intérprete de Python. La consola permite también el autocompletado, la importación de librerías, el uso de las variables dentro del propio explorador de variables y la creación de gráficos. Otra opción interesante es que permite tanto el uso como de *kernels* externos tanto locales como externos.
- El **explorador de Variables**: Permite manejar las variables y los objetos creados durante la ejecución del código de manera visual. Se compone de una tabla donde cada fila es una variable u objeto. Las columnas de izquierda a derecha son el nombre, el tipo el tamaño y el valor de la variable u objeto. Este elemento permite la creación, modificación y borrado de las variables u objetos si se desea. Es un elemento muy útil para usar en la depuración del código ya que permite ver el estado de las variables a lo largo de él.
- El **panel de gráficos**: Permite mostrar los gráficos y figuras creadas durante la ejecución del código o durante la ejecución de comandos en la consola de Python. En el panel solo aparecen las figuras que han sido creados con la consola que esta activa. Si se cambia de consola o se desactiva, los gráficos desaparecerán también. La barra superior que aparece en el panel permite interactuar con las figuras. Entre otras cosas se puede aplicar zoom, guardar como imagen, eliminar o seleccionar otra figura.
- El **depurador**: Permite depurar el código mediante la integración del depurador ipdb en la consola de Python. Este depurador permite ver y controlar los puntos de interrupción

y la ejecución del flujo desde la interfaz de Spyder en la barra de depuración. Se puede para el flujo, y reanudarlo de maneras diferentes.

- El **panel de ayuda**: Permite encontrar, mostrar y disponer de documentación de cualquier objeto con documentación. Permite acceder a la documentación directamente desde Spyder sin tener que pasar por Internet.



## 6. CAPTURA DE MOVIMIENTO

### 6.1. INTRODUCCIÓN

Como paso previo a la realización de la aplicación se realizó la captura de movimiento que luego ha generado los ficheros que actúan como datos de entrada de la aplicación y que pueden ser usados por la misma para una observación más visual de la captura del movimiento con la creación de las gráficas que se explicará posteriormente.

Para la obtención de datos se realizó la captura de movimiento sobre varios sujetos con buen estado de salud a través de sensores inerciales. Estos sensores se situaban sobre los puntos clave del cuerpo humano, como se explicará más adelante. Posteriormente los sujetos realizaban una serie de movimientos repetitivos mientras estos sensores mandaban los datos. Por último, estos datos se transformaban en ficheros que serán los que se use nuestra aplicación para su visualización.

### 6.2. DISEÑO DEL SISTEMA DE CAPTURA DE MOVIMIENTO

Al trabajar con un grupo de trabajo más amplio, el presente trabajo se ha centrado en la marcha humana, pero otros estudiantes estaban estudiando tanto el movimiento en la parte superior del cuerpo como la captura de movimiento a través de cámaras, por lo que la captura se ha realizado de manera conjunta.

Esto implica que en la sala de grabación también existían cámaras de captura de movimientos y que a los sujetos que han realizado las pruebas se les ha situado sensores tanto en la parte superior del cuerpo como en la inferior.

Como se puede ver en la imagen, el sujeto contiene sensores inerciales por las siguientes partes:

- Pie (uno en cada pie)
- Parte inferior de la pierna (uno por cada pierna)
- Parte superior de la pierna (uno por cada pierna)
- Espalda (a la altura de la pelvis)
- Espalda (a la altura del esternón)
- Parte superior del brazo (uno en cada brazo)
- Parte inferior del brazo (uno en cada brazo)

Para el análisis de la marcha nos interesa sobre todo tanto el de la pelvis como los cuatro sensores situados en la pierna. Los demás sensores, aunque son útiles tampoco representan datos importantes de cara la marcha.

### 6.3. PRUEBAS DE CAPTURA DE MOVIMIENTO

Para la captura se han utilizado varios movimientos que se han repetido hasta 2 veces cada uno. Para la parte de captura de movimiento de la parte inferior se ha diseñado los siguientes movimientos sencillos:

- Caminar hacia adelante: El sujeto caminará de frente tres veces ida y vuelta entre las dos marcas que se han situado en el suelo para ello.
- Caminar hacia atrás: El sujeto caminará hacia atrás tres veces ida y vuelta entre las dos mismas marcas que se han situado en el suelo para la prueba anterior.
- Caminar sobre una recta: El sujeto caminará tres veces ida y vuelta sobre una recta marchada en el suelo a unos 20º grados del plano sagital. Hay que tener especial cuidado con los cambios de sentido.

Posteriormente, se realizaron más pruebas que implican movimiento en la parte superior del cuerpo, pero al tratarse de un caso diferente al dominio de nuestra aplicación no se comentarán con tanto detalle.

#### 6.4. PROTOCOLO DE CAPTURA DE MOVIMIENTO

En este apartado se resumen como cada sujeto fue realizando la captura de movimiento:

Por orden, antes de la llegada de los sujetos:

- Comprobar la carga de los sensores, aplicar el *heading reset* y tenerlos listos para grabar.
- Comprobación de los formularios de consentimiento.
- Lanzar el sistema de video, y colocar la cámara en sus marcas.
- Asegurarse de que las marcas estén bien situadas y en buen estado.
- Comprobación de los materiales necesarios para la realización de los ejercicios

Por orden, a la llegada de los sujetos:

- Explicación de la prueba: grabación de movimientos con sensores y cámara. Se mostrará un ejemplo en la pantalla.
- Se grabarán doce movimientos: tres pruebas de pierna y nueve pruebas bimanuales.
- Entrega del guion de las pruebas y consentimiento de los sujetos a la grabación tanto de los movimientos como del video.
- Se rellenará una entrada del Excel con los datos del sujeto que llevará a cabo los ejercicios (ID del sujeto, nombre, mano dominante, pie dominante).
- Se vestirá al sujeto con los sensores. Se emplearán sensores de tipo:
  - o *TwynSens*
- La posición de los sensores *TwynSens* será con la entrada microUSB mirando hacia arriba tanto en tren superior como inferior con la cara externa transparente del sensor (o cara superior) hacia afuera del segmento corporal.
- Se realizará una grabación inicial del sujeto estando de pie y estirado sin realizar ningún movimiento.
- Posteriormente se llevarán a cabo los 10 ejercicios que se especifican a continuación 2 veces cada uno. En cada repetición se emplearán unos sensores diferentes.

Por orden, al realizar las pruebas:

- Tanto la grabación de sensores como de vídeo comenzará exactamente en el tercer golpe de sincronización.
- El encargado del video deberá llevar el tracking de las pruebas en todo momento, de tal manera que sólo se darán como válidas las pruebas en las que no se detecte *flicker*, desvanecimientos de la *bounding box*, o que el sujeto salga del plano de grabación.
- El encargado de los sensores deberá realizar un *heading reset* siempre antes de colocar los sensores en el sujeto y cerciorarse de la correcta colocación de los sensores. En el caso de los *TwynSens*, los sensores deberán ser encendidos en paralelo alineados con una línea recta como referencia (p.ej. el borde de la mesa) y con el puerto de carga microUSB mirando hacia la pantalla donde está el avatar. Esto es crucial ya que en estos sensores es el momento de encendido el que se toma como *Heading Reset*. Para el *Reset* por Software del sistema Unity3D, deberemos colocar el sujeto del mismo modo que vemos al avatar en su estado inicial, con las palmas de las manos apoyadas sobre sus

muslos y las piernas levemente separadas. Así, durante el *Reset SW* el sujeto mirará al frente donde está situada la cámara, pero siguiendo las líneas del suelo.

- La colocación del sujeto, de la silla y de la mesa debe seguir las marcas pertinentes.
- Se debe informar al sujeto de:
  - Los pasos deben ser lo más uniformes posibles.
  - La forma de caminar debe ser lo más natural posible.
  - La posición de inicio de todas las pruebas debe ser la misma.
  - Los cambios de dirección, giros o movimientos más bruscos deberán suavizarse y hacerse de una manera más cuidadosa.
  - Los movimientos bimanuales deben partir y finalizar de la posición inicial.

Por último, tras las pruebas:

- El encargado de los sensores quitará los mismos al sujeto.
- Se anotará en la sección de comentarios los sucesos anormales o situaciones a recalcar de la realización de las pruebas.
- Se deberá asegurar que todas las grabaciones en crudo de los sensores se encuentran en el directorio correspondiente con la etiquetación adecuada.

#### 6.5. RESULTADOS OBTENIDOS DE LA CAPTURA

El resultado de esta captura son múltiples ficheros, uno por cada movimiento que ha realizado cada sujeto, por lo que se ha generado una carpeta compartida para poder ir dejando todos los ficheros ordenados según sujeto y según movimiento.



## 7. DISEÑO DE LA APLICACIÓN

En esta sección se abordará la fase del desarrollo de la aplicación. Una vez escogidos todos los materiales y métodos que se han visto en el apartado anterior, se pueden pasar a la recolección de requisitos de la aplicación que permita configurar los casos de uso en lo que se debe basar la implementación de esta.

Por lo tanto, lo primero que se verá son los requisitos que deberá tener la aplicación realizada, divididos en requisitos funcionales y no funcionales y posteriormente se describirán todos los casos de uso de la aplicación.

### 7.1. REQUISITOS

Los requisitos de una aplicación software son, por tanto, descripciones de lo que debería hacer el sistema. Los requisitos funcionales son las descripciones de los servicios que la aplicación debe realizar y los no funcionales son descripciones de una característica que el sistema debe poseer o una restricción que el sistema debe respetar.

#### 7.1.1. FUNCIONALES

En este apartado se muestra la lista de requisitos funcionales de la aplicación:

<b>Identificador</b>	FRQ01
<b>Fecha última revisión</b>	20/06/2023
<b>Descripción</b>	El sistema deberá de listar los ficheros Excel y MOT que se encuentre en la carpeta que se configure como directorio raíz, asignándoles un número

*Tabla 7. Requisito funcional FRQ01*

<b>Identificador</b>	FRQ02
<b>Fecha última revisión</b>	20/06/2023
<b>Descripción</b>	El sistema ofrecerá al usuario escoger uno de estos ficheros si selecciona el número asignado al mismo

*Tabla 8. Requisito funcional FRQ02*

<b>Identificador</b>	FRQ03
<b>Fecha última revisión</b>	20/06/2023
<b>Descripción</b>	El sistema deberá de listar las hojas de cálculo que contengan el fichero Excel, si este es el elegido, asignándolas un número

*Tabla 9. Requisito funcional FRQ03*

<b>Identificador</b>	FRQ04
<b>Fecha última revisión</b>	20/06/2023
<b>Descripción</b>	El sistema ofrecerá al usuario escoger una de estas hojas de cálculo si selecciona el número asignado a la misma

*Tabla 10. Requisito funcional FRQ04*

<b>Identificador</b>	FRQ05
<b>Fecha última revisión</b>	20/06/2023
<b>Descripción</b>	El sistema ofrecerá al usuario escoger el modo de visualización

*Tabla 11. Requisito funcional FRQ05*

<b>Identificador</b>	FRQ06
<b>Fecha última revisión</b>	20/06/2023

<b>Descripción</b>	Si se escoge el modo solitario o dúo, el sistema listará la lista de parámetros del fichero MOT escogido o de la hoja de cálculo escogida del fichero Excel, asignándoles un número
--------------------	---

*Tabla 12. Requisito funcional FRQ06*

<b>Identificador</b>	FRQ07
<b>Fecha última revisión</b>	20/06/2023
<b>Descripción</b>	Si se escoge el modo solitario o dúo, el sistema ofrecerá al usuario escoger el parámetro si selecciona el número asignado al mismo

*Tabla 13. Requisito funcional FRQ07*

<b>Identificador</b>	FRQ08
<b>Fecha última revisión</b>	20/06/2023
<b>Descripción</b>	Si se escoge el modo solitario o dúo, el sistema ofrecerá al usuario introducir la tolerancia

*Tabla 14. Requisito funcional FRQ08*

<b>Identificador</b>	FRQ09
<b>Fecha última revisión</b>	20/06/2023
<b>Descripción</b>	Si el usuario completa con éxito el proceso de selección, el sistema deberá generar la gráfica de la media y desviación típica en el navegador

*Tabla 15. Requisito funcional FRQ09*

<b>Identificador</b>	FRQ10
<b>Fecha última revisión</b>	20/06/2023
<b>Descripción</b>	Si el usuario completa con éxito el proceso de selección, el sistema deberá generar la gráfica de los ciclos de la marcha en el navegador

*Tabla 16. Requisito funcional FRQ10*

<b>Identificador</b>	FRQ11
<b>Fecha última revisión</b>	20/06/2023
<b>Descripción</b>	Si el usuario completa con éxito el proceso de selección, el sistema deberá mostrar los parámetros de la marcha por pantalla

*Tabla 17. Requisito funcional FRQ11*

<b>Identificador</b>	FRQ12
<b>Fecha última revisión</b>	20/06/2023
<b>Descripción</b>	El sistema deberá volver al listado de los ficheros cuando se muestren las gráficas

*Tabla 18. Requisito funcional FRQ12*

### 7.1.2. NO FUNCIONALES

En este apartado se muestra la lista de requisitos no funcionales de la aplicación:

<b>Identificador</b>	NFRQ01
<b>Fecha última revisión</b>	20/06/2023
<b>Descripción</b>	La interfaz de usuario será eficaz y sencilla para que el usuario comprenda toda la información.

Tabla 19. Requisito funcional NFRQ01

<b>Identificador</b>	NFRQ02
<b>Fecha última revisión</b>	20/06/2023
<b>Descripción</b>	El sistema deberá informar de los errores que se produzcan cuando el usuario introduce un dato inválido.

Tabla 20. Requisito funcional NFRQ02

<b>Identificador</b>	NFRQ03
<b>Fecha última revisión</b>	20/06/2023
<b>Descripción</b>	El sistema deberá mostrar informar con una frase inicial del listado que se muestra.

Tabla 21. Requisito funcional NFRQ03

## 7.2. CASOS DE USO

Los Casos de Uso facilitan a comprender las funcionalidades de la aplicación de manera más granular por lo que se puede entenderla de mejor manera y se puede observar las acciones de los diferentes actores de una mejor manera.

### 7.2.1. VisualizarGráficas

Identificador	VisualizarGráficas	
<b>Descripción</b>	El usuario de la aplicación podrá visualizar la gráfica que desea	
<b>Actor</b>	Usuario	
<b>Precondición</b>	Haber ejecutado el programa y haber realizado la configuración necesaria	
<b>Escenario Principal de Éxito</b>	<b>Paso</b>	<b>Acción</b>
	1	El usuario ejecuta la aplicación
	2	El sistema muestra la lista de ficheros y una entrada para que el usuario pueda elegir el fichero
	3	El usuario escoge el fichero seleccionando el número correspondiente al fichero
	4	El sistema muestra la lista de modos y una entrada para que el usuario pueda elegir el modo que desee
	5	El usuario escoge el modo escribiéndolo por teclado
	6	El sistema muestra las gráficas correspondientes a ese modo y vuelve al paso 2
<b>Postcondición</b>	El usuario observa las diferentes gráficas que se generan según en modo escogido	
<b>Alternativas</b>	<b>Paso</b>	<b>Acción</b>
	3.1	Si el usuario escoge un fichero tipo Excel seleccionando el número correspondiente al fichero

	3.1.1	El sistema muestra la lista de pestañas y una entrada para que el usuario pueda elegir la pestaña que desee
	3.1.2	El usuario escoge la pestaña seleccionando el número correspondiente a la misma
	5.1	Si el usuario escoge el modo single o dúo:
	5.1.1	El sistema muestra la lista de parámetros disponibles para cada modo y una entrada para que el usuario pueda elegir el parámetro que desee
	5.1.2	El usuario escoge el parámetro seleccionando el número correspondiente al mismo
	5.1.3	El sistema muestra una entrada para que el usuario pueda incluir el porcentaje (de 0 a 99) de la tolerancia
	5.1.4	El usuario escoge la tolerancia que desee seleccionando el porcentaje correspondiente
	3.2, 5.2, 5.1.2.1 5.1.4.1	Si el usuario introduce cualquier valor que no esté permitido, es decir no pertenezca a la lista o este fuera del intervalo 0-99 para la tolerancia, el programa muestra un error y el caso de uso finaliza
	<b>Requisitos especiales</b>	Ninguno
	<b>Frecuencia esperada</b>	Alta

Tabla 22. Caso de uso VisualizarGráficas



## 8. MANUAL DE USUARIO

### 8.1. INTRODUCCION

En esta sección se procederá a explicar el uso de la aplicación al usuario final. La explicación abordará desde los prerequisites necesarios para poder ejecutar la aplicación, la correcta descarga de los ficheros de la aplicación, la configuración previa de ciertos ficheros y el manual de cómo usar la aplicación, con pantallazos y explicaciones de como trabajar con ella.

### 8.2. PRERREQUISITOS

Se pueden destacar los siguientes prerequisites:

- El primer requisito necesario es conceptual. Como es una aplicación bastante específica para la visualización de gráficas sobre parámetros de la marcha humana, se supone que el usuario final deberá tener conocimientos sobre la marcha humana. De esta forma podrá entender y estudiar los datos que se muestran por pantalla.
- Otro prerequisite importante es que el usuario también deberá conocer como poder editar ciertos parámetros clave del sistema, previos a la ejecución de este, para que la aplicación funcione correctamente. Se explicará a continuación en esta guía como poder acceder a ellos y modificarlos a su gusto.
- En cuanto a requisitos software, se enumeran a continuación las versiones tanto de Python como de las librerías utilizadas para el correcto funcionamiento de la aplicación:
  - **Python:** 3.9.7
  - **Pandas:** 2.0.1
  - **Plotly:** 2.0.1
  - **Numpy:** 1.20.3
  - **Scipy:** 1.7.1
  - **Seaborn:** 0.12.2

### 8.3. DESCARGA

Una vez se descarga el fichero comprimido del programa se debe situar sobre una carpeta conocida. Al descomprimir el fichero se crea una carpeta con el mismo nombre del fichero. La carpeta descomprimida cuenta con ficheros de prueba tanto EXCEL como MOT, pero si se quiere añadir más ficheros, estos se deben dejar dentro de la carpeta.

Nombre	Fecha	Tipo	Tamaño	Etiquetas
00.GaitBOTAS-001.xlsx	05/05/2022 21:49	Hoja de cálculo d...	16.952 KB	
ApplyToleranceController.py	17/03/2023 1:04	Python File	2 KB	
auxiliar.py	17/03/2023 0:48	Python File	4 KB	
ConfigGP.py	21/03/2023 20:51	Python File	1 KB	
ConfigRecord.py	21/03/2023 20:51	Python File	1 KB	
configuracion.txt	19/06/2023 19:48	Documento de te...	3 KB	
configuracion_old.txt	19/05/2023 0:49	Documento de te...	2 KB	
dfmot_m01.mot	18/05/2023 0:28	Archivo MOT	392 KB	
dfmot_m01_tab.mot	18/05/2023 0:29	Archivo MOT	396 KB	
dfmot_marcha_larga_S01_XZY.mot	15/06/2023 23:51	Archivo MOT	1.982 KB	
dfmot_marcha_larga_S01_ZXY.mot	15/06/2023 23:52	Archivo MOT	1.981 KB	
dfmot_marcha_larga_S02_ZXY.mot	19/06/2023 19:58	Archivo MOT	1.814 KB	
dfmot_marcha_larga_S02_ZXY.mot	19/06/2023 19:58	Archivo MOT	1.814 KB	
dfmot_marcha_larga_S03_ZXY.mot	19/06/2023 19:58	Archivo MOT	2.375 KB	
dfmot_marcha_larga_S03_ZXY.mot	19/06/2023 19:59	Archivo MOT	2.375 KB	
ExcelGP.py	17/03/2023 1:04	Python File	1 KB	
GaitCycleController.py	20/06/2023 21:23	Python File	9 KB	
ik_rawTwynSens19-05_05-08-27.AndarFre...	04/02/2023 18:20	Archivo MOT	868 KB	
ik_S40_A01_T01.mot	28/02/2023 17:32	Archivo MOT	873 KB	
ik_S41_A01_T02.mot	28/02/2023 17:35	Archivo MOT	891 KB	
main.py	31/08/2023 19:28	Python File	13 KB	
MotGP.py	17/03/2023 1:00	Python File	1 KB	
NOTAS.txt	09/05/2023 23:58	Documento de te...	2 KB	
PlotGraphController.py	19/06/2023 22:54	Python File	11 KB	
PlotParameterData.py	20/06/2023 21:25	Python File	11 KB	
ReadFolderController.py	17/06/2023 17:57	Python File	5 KB	
README.md	12/02/2023 11:37	Archivo MD	6 KB	
SheetGP.py	17/03/2023 1:02	Python File	2 KB	
views.py	17/06/2023 18:12	Python File	24 KB	

Figura 16. Lista de ficheros descargables en Gitlab

#### 8.4. CONFIGURACIÓN PREVIA

Una vez descargado los ficheros, hay que configurar la ruta del proyecto, para lo cual se debe abrir el fichero main.py. Se puede abrir mismamente con el bloc de notas o editor de texto que se prefiera. En la línea 294 como se puede observar en la captura se debe cambiar la ruta por defecto, por la ruta propia de la carpeta del proyecto del equipo donde se quiere ejecutar el programa.

```

291     pio.renderers.default='browser'
292
293     #Se establece el directorio del programa
294     ST_IN_FOLDER="C:/Users/Estudiante/Desktop/TFM_23/AnalysisParameters"
295
296     #Se establecen los tipos de ficheros que el programa puede leer
297     ext_excel = ".xlsx"
298     ext_mot = ".mot"

```

Figura 17. Lugar en el código del fichero main.py donde se especifica la ruta

Si se quiere visualizar ficheros a mayores de los que existen en la carpeta descargable por defecto, se deben incluir en la carpeta del proyecto antes de realizar la ejecución de este.

Por último, en esta sección, se comentará el fichero de configuración. Este fichero contiene una tabla con datos de los diferentes parámetros que el programa puede visualizar. Por defecto contiene los datos de los siguientes parámetros, correspondiente a las articulaciones importantes para la marcha:

Fichero MOT	Fichero Excel
<i>pelvis_tilt</i>	<i>Right Hip Abduction/Adduction</i>
<i>pelvis_list</i>	<i>Right Hip Internal/External Rotation</i>
<i>pelvis_rotation</i>	<i>Right Hip Flexion/Extension</i>
<i>pelvis_tx</i>	<i>Right Knee Abduction/Adduction</i>
<i>pelvis_ty</i>	<i>Right Knee Internal/External Rotation</i>
<i>pelvis_tz</i>	<i>Right Knee Flexion/Extension</i>
<i>hip_flexion_r</i>	<i>Right Ankle Abduction/Adduction</i>
<i>hip_adduction_r</i>	<i>Right Ankle Internal/External Rotation</i>
<i>Hip_rotation_r</i>	<i>Right Ankle Dorsiflexion/Plantarflexion</i>
<i>knee_angle_r</i>	<i>Left Hip Abduction/Adduction</i>
<i>hip_flexion_l</i>	<i>Left Hip Internal/External Rotation</i>
<i>hip_adduction_l</i>	<i>Left Hip Flexion/Extension</i>
<i>hip_rotation_l</i>	<i>Left Knee Abduction/Adduction</i>
<i>knee_angle_l</i>	<i>Left Knee Internal/External Rotation</i>
	<i>Left Knee Flexion/Extension</i>
	<i>Left Ankle Abduction/Adduction</i>
	<i>Left Ankle Internal/External Rotation</i>
	<i>Left Ankle Dorsiflexion/Plantarflexion</i>

Tabla 23. Lista de los parámetros configurables por defecto

Si se quiere incluir algún nuevo parámetro para visualizar de forma independiente se puede incluir en este fichero de configuración.

Para todos estos parámetros se deben configurar los siguientes campos:

- **FILE\_TYPE:** Contiene el tipo de fichero al que pertenece el parámetro. Dependiendo de si el fichero es tipo excel o tipo MOT se definirá como Sheet o Mot, respectivamente, contienen diferentes nombres para los tipos de parámetros y se procesan de manera distinta, por lo que es importante distinguirlos.
- **NUMBER:** Cada listado de parámetros para cada tipo de ficheros estas ordenados numéricamente, empezando por el índice 0. Si se requiere añadir un parámetro a mayores se tiene que respetar este orden y añadir la nueva línea al final de la lista con el siguiente número disponible. Si la última línea para FILE\_TYPE = Mot contiene el NUMBER = 11, la nueva línea deberá contener NUMBER = 12.
- **NAME:** Este campo contendrá el nombre del parámetro. Si se incluye como un nuevo parámetro, este campo debe coincidir con el nombre de la columna del fichero. Aunque se refieran a lo mismo, los nombres de las columnas de los distintos tipos de fichero cambian, como se ha comentado antes.
- **OFFSET\_MAX (findpeaks):** Este campo representa el nivel de offset que hay que añadir a las gráficas para que la función *find\_peaks* detecte los máximos, ya que, por defecto, esta función no detecta los máximos si están por debajo de 0. Por lo que la solución

propuesta en este trabajo ha sido incluir este parámetro que se puede cambiar externamente. Si se deja en 0, no se añade ningún offset.

- **PROMINENCE\_MAX (*findpeaks*):** Este campo representa el valor que debe tener un máximo desde su base para que la función *find\_peaks* pueda detectarlo. Este campo ayuda a quitarnos ciertos máximos que son solo ruido y alteran la correcta visualización del campo. Si se deja en 0, este campo no tiene relevancia en la detección de máximos, por tanto, pero entonces no ayuda a realizar el filtrado.
- **DISTANCE\_MAX (*find\_peaks*):** Este campo representa la distancia en muestras en la que deben estar los máximos para que la función *find\_peaks* pueda detectarlos. Este campo al igual que el anterior nos ayuda a filtrar ciertos máximos que son solo ruido y alteran la visualización óptima de las gráficas. Al igual que antes, si este campo se deja con valor 0, no realiza el filtrado.
- **DATA\_TYPE:** Este campo es un *boolean* que identifica los dos tipos de gráficas posibles. Las que tiene doble máximo (*TRUE*) y las que tienen un solo máximo (*FALSE*). Esto es importante porque el primer tipo hace necesario la actuación del siguiente parámetro.
- **VALUE\_MAX:** Este campo solo se aplica cuando se detectan dos máximos dentro del mismo ciclo y sirve para diferenciar uno de otro, por lo que actúa de umbral para separar el máximo absoluto del máximo relativo.
- **LEFT:** Este campo es otro *boolean* que identifica si el parámetro pertenece a la pierna izquierda (*TRUE*) o la pierna derecha (*FALSE*) y permite la correcta visualización del parámetro respetando el código de colores, por ejemplo

### 8.5. EJECUCIÓN

Una vez realizada la configuración se puede ejecutar correctamente la aplicación. En esta sección se explicará paso a paso las diferentes funcionalidades con capturas de pantalla a modo de guía.

La aplicación permite al usuario poder escoger de entre una lista de ficheros tipo Excel o tipo MOT generados en una captura de movimiento, y visualizar los parámetros vistos en la tabla o lo que el mismo usuario haya incluido en el fichero de configuración si vienen también como columna en el fichero elegido como se ha comentado antes.

Lo primera que hay que realizar es ejecutar el fichero `main.py`. Para ello se tiene que ejecutar el siguiente comando en la consola:

```
py main.py
```

Una vez se lanza el programa, este buscará y listará los ficheros Excel o MOT que existen en la carpeta que se ha configurado como raíz y ofrece al usuario poder elegir uno de ellos si introduce el número que tiene ese fichero en el listado y pulsa a la tecla *Intro*. Si introduce un número que no está en el listado o cualquier otro carácter y pulsamos la tecla *Intro*, el programa generará un error.

```

-----
[!] Ficheros:
[0] 00.GaitBOTAS-001.xlsx
[1] dfmot_m01.mot
[2] dfmot_m01_tab.mot
[3] dfmot_marcha_larga_S01_XZY.mot
[4] dfmot_marcha_larga_S01_ZXY.mot
[5] dfmot_marcha_larga_S02_XZY.mot
[6] dfmot_marcha_larga_S02_ZXY.mot
[7] dfmot_marcha_larga_S03_XZY.mot
[8] dfmot_marcha_larga_S03_ZXY.mot
[9] ik_rawTwynSens19-05_05-08-27.AndarFrenteYVuelta.recortado30s.mot
[10] ik_S40_A01_T01.mot
[11] ik_S41_A01_T02.mot

[!] Elige un num de fichero:
  
```

Figura 18. Ejemplo de lista de ficheros

```

-----
[!] Ficheros:
[0] 00.GaitBOTAS-001.xlsx
[1] dfmot_m01.mot
[2] dfmot_m01_tab.mot
[3] dfmot_marcha_larga_S01_XZY.mot
[4] dfmot_marcha_larga_S01_ZXY.mot
[5] dfmot_marcha_larga_S02_XZY.mot
[6] dfmot_marcha_larga_S02_ZXY.mot
[7] dfmot_marcha_larga_S03_XZY.mot
[8] dfmot_marcha_larga_S03_ZXY.mot
[9] ik_rawTwynSens19-05_05-08-27.AndarFrenteYVuelta.recortado30s.mot
[10] ik_S40_A01_T01.mot
[11] ik_S41_A01_T02.mot

[!] Elige un num de fichero: 99
[!] Error: No se ha introducido un valor válido
An exception has occurred, use %tb to see the full traceback.

SystemExit: -1
  
```

Figura 19. Ejemplo de fichero mal seleccionado

Si la elección de fichero fue exitosa, entonces si el fichero es Excel, se ofrecerá a mayores poder elegir el Sheet que queremos analizar de la misma forma que se seleccionó el fichero. Se listan los Sheets asignándolos un número. Esta aplicación solo tiene configurado el Sheet, por lo que, si se elige un Sheet que no es ese, se introduce un número que no aparece en el listado o cualquier otro carácter inválido y pulsamos la tecla *Intro*, el programa generará un error.

```

-----
[!] Ficheros:
[0] 00.GaitBOTAS-001.xlsx
[1] dfmot_m01.mot
[2] dfmot_m01_tab.mot
[3] dfmot_marcha_larga_S01_XZY.mot
[4] dfmot_marcha_larga_S01_ZXY.mot
[5] dfmot_marcha_larga_S02_XZY.mot
[6] dfmot_marcha_larga_S02_ZXY.mot
[7] dfmot_marcha_larga_S03_XZY.mot
[8] dfmot_marcha_larga_S03_ZXY.mot
[9] ik_rawTwynSens19-05_05-08-27.AndarFrenteYVuelta.recortado30s.mot
[10] ik_S40_A01_T01.mot
[11] ik_S41_A01_T02.mot

[!] Elige un num de fichero: 0
-----

[!] Sheets:
[0] General Information
[1] Markers
[2] Segment Orientation - Quat
[3] Segment Orientation - Euler
[4] Segment Position
[5] Segment Velocity
[6] Segment Acceleration
[7] Segment Angular Velocity
[8] Segment Angular Acceleration
[9] Joint Angles ZXY
[10] Joint Angles XZY
[11] Ergonomic Joint Angles ZXY
[12] Ergonomic Joint Angles XZY
[13] Center of Mass
[14] Sensor Free Acceleration
[15] Sensor Magnetic Field
[16] Sensor Orientation - Quat
[17] Sensor Orientation - Euler

[!] Elige un num de sheet:

```

Figura 20. Selección de fichero exitosa y lista de pestañas de fichero Excel

### 8.5.1. MODOS DE VISUALIZACIÓN

Tanto si selecciona el Sheet correcto como cualquier fichero tipo MOT, la siguiente pantalla que se muestra indica el primer pie que dio el paso (pie izquierdo o pie derecho), indica también el tipo de fichero elegido (MOT con etiquetas o MOT sin etiquetas o sheet) y, por último, da a elegir la forma de visualización.

```

-----
El primer paso se ha realizado con la izquierda
Mot sin etiquetas o sheet
-----
Seleccione el modo de visulización (single/duo/all):|

```

Figura 21. Selección del modo de visualización

Se explica a continuación cada modo de visualización:

- **Single:** En este modo solo se visualiza un parámetro de la tabla, por lo que según se escoja un tipo de fichero u otro se mostrará una columna u otra actualizada con los parámetros si se han incluido de manera correcta. En esta lista se diferencian los parámetros recogidos en la captura para la pierna izquierda y la pierna derecha. De la misma forma que las anteriores listas, se asigna un número a cada parámetro y si se quiere obtener las gráficas de un determinado parámetro, se debe introducir el número correctamente y pulsar la tecla *Intro*.

```

Seleccione el modo de visulización (single/duo/all):single
-----
-----
----- Has seleccionado el modo single! -----
-----
-----

[#] Sheet:
[0] Right Hip Abduction/Adduction
[1] Right Hip Internal/External Rotation
[2] Right Hip Flexion/Extension
[3] Right Knee Abduction/Adduction
[4] Right Knee Internal/External Rotation
[5] Right Knee Flexion/Extension
[6] Right Ankle Abduction/Adduction
[7] Right Ankle Internal/External Rotation
[8] Right Ankle Dorsiflexion/Plantarflexion
[9] Left Hip Abduction/Adduction
[10] Left Hip Internal/External Rotation
[11] Left Hip Flexion/Extension
[12] Left Knee Abduction/Adduction
[13] Left Knee Internal/External Rotation
[14] Left Knee Flexion/Extension
[15] Left Ankle Abduction/Adduction
[16] Left Ankle Internal/External Rotation
[17] Left Ankle Dorsiflexion/Plantarflexion

[!] Elige un num de columnas: |

```

Figura 22. Lista de parámetros por defecto del modo single para fichero Excel

Una vez hecho todo esto, se da la opción de escoger la tolerancia. Este parámetro debe ser un número entre 0 y 100. Esta tolerancia se aplica sobre la media de los máximos que tiene cada ciclo, por lo que actúa como un filtro que quita los ciclos con el máximo por encima o por debajo del intervalo que se genera con la tolerancia, es decir, filtra los ciclos menos representativos. Para una mejor interpretación de ello se ha incluido este umbral de tolerancia en las gráficas.

```

-----
-----
----- Seleccione la tolerancia en porcentaje respecto a la media de los máximos (0-100):20
-----
----- Tolerancia leída! -----
-----
-----

```

Figura 23. Selección de tolerancia exitosa

Para este modo se visualizan 3 gráficas como se puede apreciar en la captura, son las siguientes:

- Máximos y mínimos del parámetro
  - Media y desviación típica
  - Ciclos de la marcha
- **Duo:** En este modo se visualizan conjuntamente tanto el parámetro de la pierna izquierda como el de pierna derecha, por lo que el listado que se muestra es el siguiente:

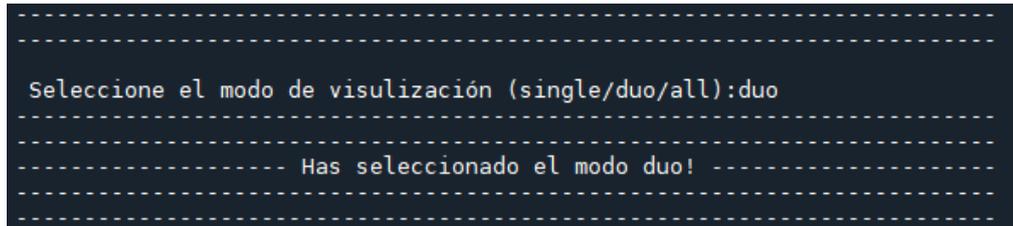


Figura 24. Selección del modo duo

Hay que tener en cuenta que no se incluyen en este listado ningún parámetro de la pelvis al no tener diferencia entre izquierda y derecha. Para este caso también se listan los parámetros como las anteriores listas, asignándoles un número. Si se escoge un número de la lista y se pulsa la tecla *Intro* y se introduce la tolerancia, esta vez compartida por las dos gráficas, se mostrarán las siguientes gráficas:

- Máximos y mínimos de ambas piernas
- Media y desviación típica de ambas piernas
- Ciclos de la marcha de una pierna
- Ciclos de la marcha de la otra pierna

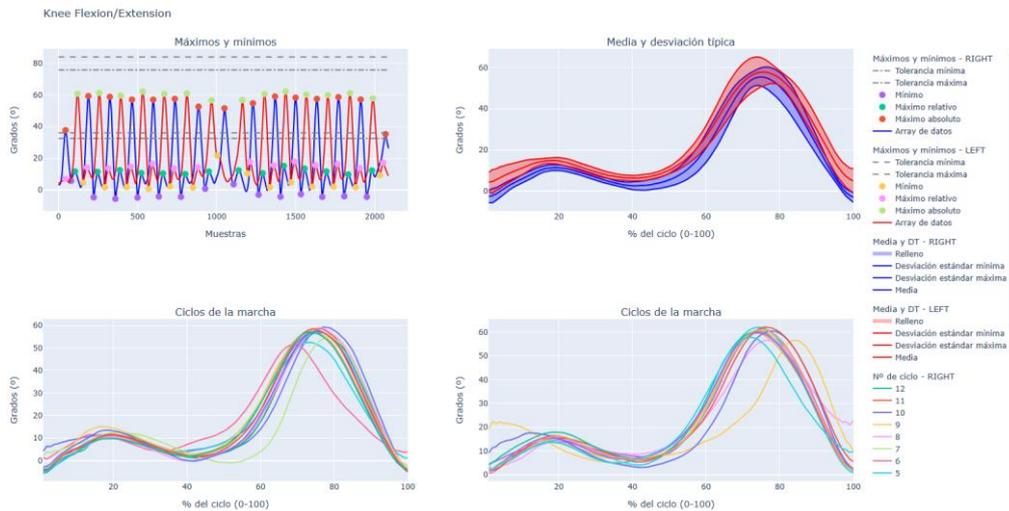


Figura 25. Visualización de gráficas para el modo "duo" para ficheros Excel

- **All:** En este modo de visualización se incluyen los principales parámetros tanto del fichero de tipo MOT, como del tipo excel, por lo que esta vez no se selecciona ningún parámetro, ni la tolerancia, que se establece por defecto. Se muestran las gráficas de la media y desviación típica de ambas piernas como se puede apreciar en las capturas.

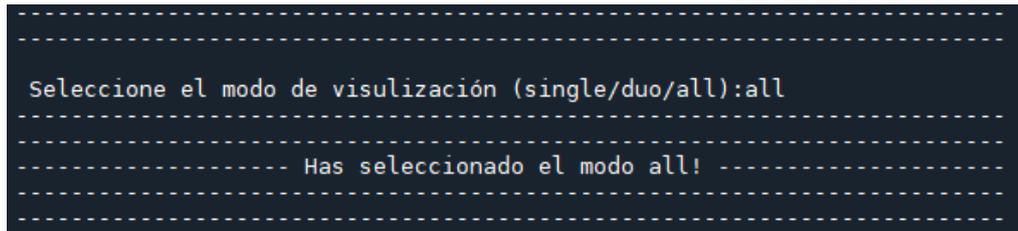


Figura 26. Selección exitosa del modo "all"

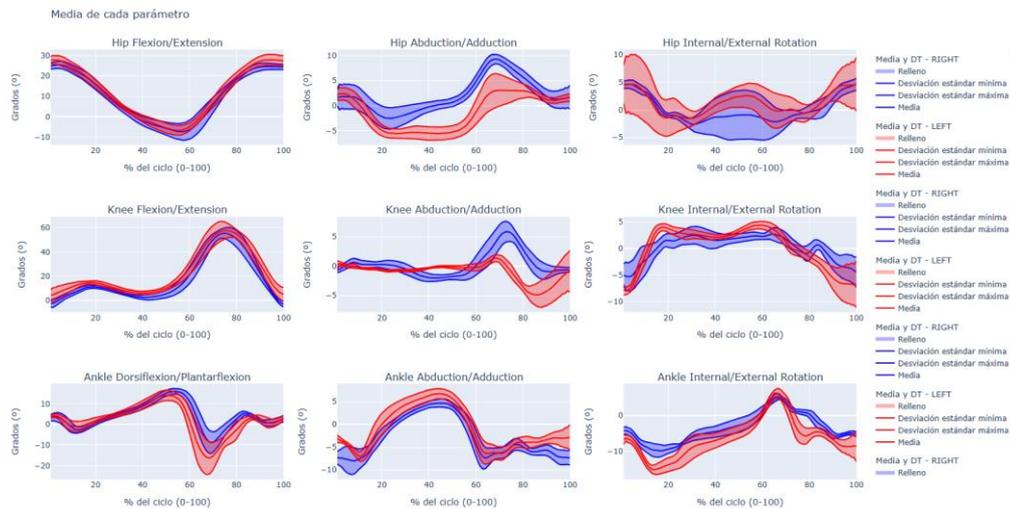


Figura 27. Visualización de gráficas para el modo "all" para ficheros Excel

### 8.5.2. GRÁFICAS

En este apartado se definen las gráficas que se pueden visualizar en el programa. Ya se ha comentado cuales se mostrarán en cada modo de representación, ahora se verá las gráficas en sí.

- **Gráfica de máximos y mínimos:** En esta gráfica se muestran todas las muestras de la columna seleccionada. En el eje de abscisas se tiene el índice mientras que en el eje de ordenadas se muestran los datos de la columna para ese índice. En ella además de muestran tanto los puntos donde se ha delimitado el inicio y el final de cada ciclo. Además, se muestran también los máximos absolutos y relativos y los mínimos. Como se puede apreciar en esta primera gráfica también se puede apreciar el rango que se genera con la tolerancia.

Right Knee Flexion/Extension: Gráfica de máximos y mínimos

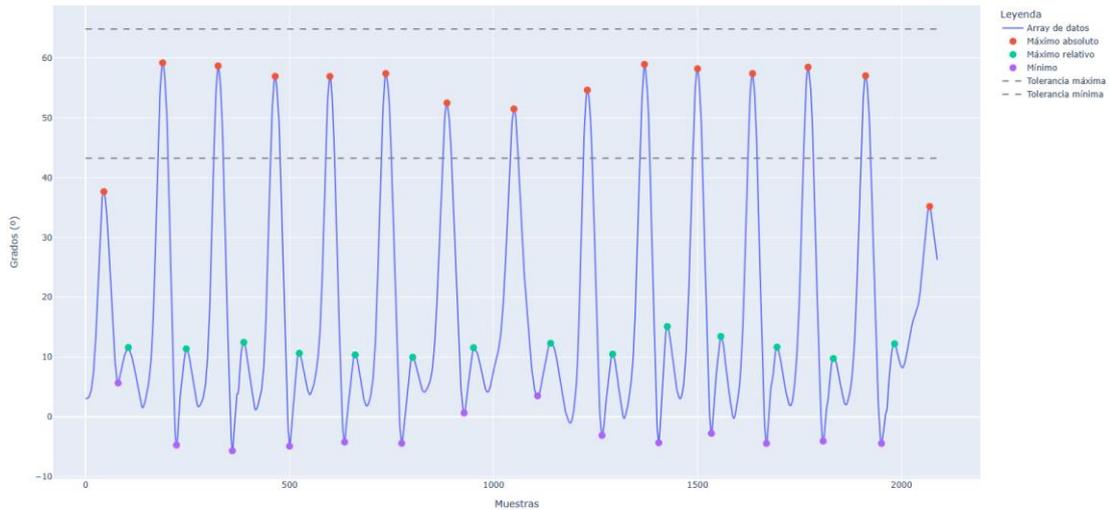


Figura 28. Visualización de gráfica de máximos y mínimos

Con esta gráfica podemos observar una visión general de como se ha capturado este parámetro para la captura de movimiento realizada. Además, también podemos observar cómo se generan los máximos y los mínimos y poder cambiar los parámetros del fichero de configuración al gusto para que las próximas gráficas se muestren correctamente.

Cuando se seleccionan el modo 'duo' se muestra el gráfico conjunto con los datos superpuestos de ambos parámetros

- **Gráfica de los ciclos de la marcha:** Esta gráfica muestra los diferentes ciclos superpuestos en los que el sistema ha dividido la marcha en diferentes colores. En este caso, el eje de abscisas contiene el porcentaje de 0 a 100 mientras que el eje de ordenadas vuelve a contener los datos de la columna. Solo se muestran los ciclos que tengan el máximo dentro del rango que se genera con la tolerancia.

Right Knee Flexion/Extension: Ciclos de la marcha

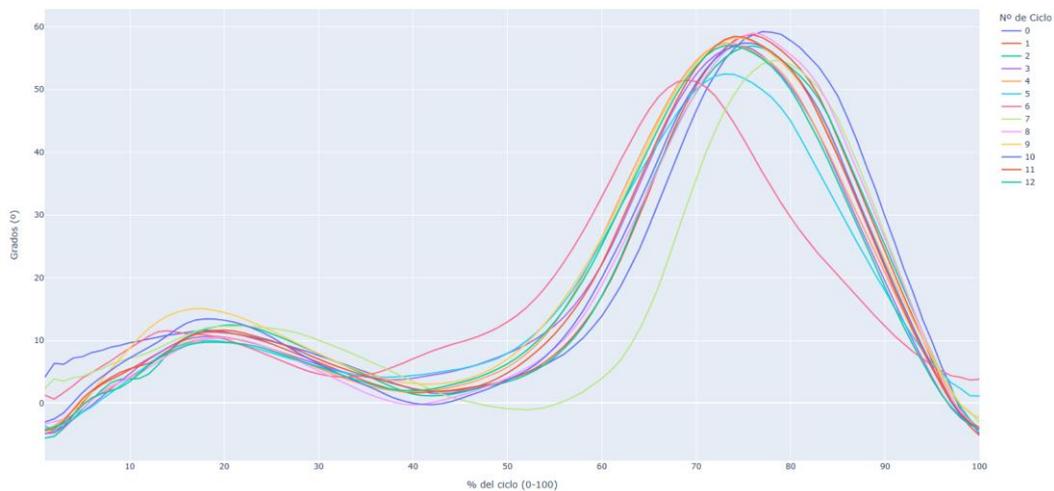


Figura 29. Visualización de gráfica de los ciclos de la marcha

Con esta gráfica podemos observar cómo se comporta el parámetro a través de los ciclos, porque al superponer los ciclos podemos ver la evolución de esta sobre ellos y poder predecir tendencias.

- **Gráfica de media y desviación estándar:** En este último tipo de gráfica se muestran la media y la desviación estándar de los diferentes ciclos que entran dentro de la tolerancia. Los ejes de abscisas y ordenadas son los mismos que la anterior gráfica. Se ha coloreado el rango que queda dentro de la desviación estándar.

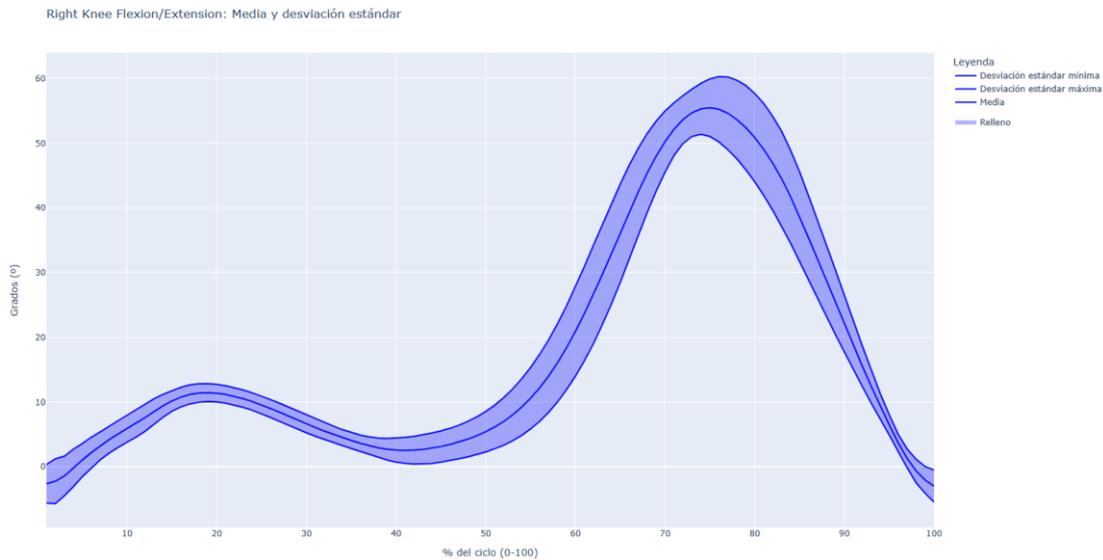


Figura 30. Visualización de gráfica de media y desviación estándar

Con esta gráfica, podemos ver cuál sería la media del parámetro para los ciclos analizados, y la desviación estándar. Cuantos más ciclos se capturen mayor calidad de datos se tendrá y, por tanto, unos gráficos más limpios y visibles.

Cuando se selecciona el modo 'duo' y el modo 'all' para esta gráfica también se superponen los datos de ambas piernas en una misma gráfica para ver la diferencia entre ellas



## 9. CONCLUSIONES Y LÍNEAS FUTURAS

Este último apartado se realizará un comentario sobre las conclusiones de la implementación de esta aplicación y de las acciones futuras que se pueden realizar sobre ella para realizar alguna que otra mejora.

### 9.1. CONCLUSIONES

Se ha conseguido el objetivo del trabajo, que era desarrollar una herramienta capaz de generar las gráficas, en función de los ciclos de la marcha. Las gráficas permiten estudiar uno o varios parámetros en varias formas como ya se ha visto. Esto permite poder ver las diferentes fases y los comportamientos de las diferentes articulaciones, por ejemplo.

Estas gráficas también se pueden generar con los programas software XSENS, pero en su versión de pago, por lo que es una herramienta muy útil de código abierto, para próximas investigaciones.

Sin duda, el mayor obstáculo de este trabajo ha sido encontrar una manera fiable de dividir los datos de la captura en los ciclos de la marcha. Se ha escogido la flexión/extensión de la rodilla porque era la gráfica más fiable y donde se distinguía mejor el cambio de ciclo, gracias a los mínimos de esta gráfica. Otro reto importante ha sido externalizar la configuración de las gráficas para que se puede manejar externamente el comportamiento del programa y de esta manera poder cambiar parámetros de la visualización para una mejor representación. El tercer reto más complicado ha sido aplicar el paradigma modelo-vista-controlador ya que supuso un cambio total la toda la implementación, pero significó un mayor entendimiento del código.

Con la realización de la aplicación se ha podido aprender a programar en Python, como ya se ha dicho antes, uno de los lenguajes más utilizados y famosos, así como conocer varias bibliotecas interesantes como *Numpy* o *Pandas* para el tratamiento de datos complejos en Python y *Plotly* o *Seaborn* para la creación de gráficas. También ha sido interesante programar basándose en el paradigma de Modelo-Vista-Controlador, un paradigma que no se ha visto durante este máster y ofrece un enfoque sencillo de entender.

Por otro lado, también se ha podido conocer mucha información sobre la marcha humana y sus ciclos, y a cómo usar la tecnología para ayudar a un mejor estudio de esta. Además, se ha podido aprender también a realizar una captura de movimiento en un laboratorio, necesaria para generar datos útiles para este trabajo. Dentro de la captura de movimiento se ha trabajado con sensores inerciales, una de las tantas opciones que existen, por lo que también se ha aprendido a trabajar con ellos.

### 9.2. LÍNEAS FUTURAS

Como líneas futuras o de mejora se pueden destacar varias, aunque la aplicación cumple con las especificaciones propuestas en un principio. Se destacan las siguientes líneas en forma de lista:

- Poder adaptar las gráficas también para los parámetros de la parte superior del cuerpo, ya que de esta manera se podría ver cómo se comportan en función de los ciclos de la marcha
- Poder adaptar más tipos de ficheros además de los Excel y MOT. De esta manera el programa puede obtener una mayor funcionalidad
- Permitir al usuario escoger entre otros idiomas además del español, como inglés
- Incluir más parámetros interesantes, como los que aparecen en las demás pestañas de los ficheros Excel utilizados

- Añadir los parámetros temporales en la salida de pantalla



## REFERENCIAS

Captura de movimiento | Introducción a la captura de movimiento | Adobe. (2023). <https://www.adobe.com/es/creativecloud/animation/discover/motion-capture.html>

Welcome! — Anaconda documentation. (2023). <https://docs.anaconda.com/>

Azucas, R. (2023, July 5). Rótula (patela). Kenhub. <https://www.kenhub.com/es/library/anatomia-es/rotula-patela>

Bernad, M. N. B. (2021, April 21). Análisis deportivo mediante el sistema VICON. Institut Català Del Peu. <https://www.institutcataladelpeu.com/analisi-deportivo-mediante-el-sistema-vicon/>

Benjaminse, A. (2020). A VALIDITY STUDY COMPARING XSENS WITH VICON. NMU Commons. <https://commons.nmu.edu/isbs/vol38/iss1/190/>

Junquera, R. (2023). Hueso fémur. [www.fisioterapia-online.com](http://www.fisioterapia-online.com). <https://www.fisioterapia-online.com/glosario/hueso-femur>

Junquera, R. (2023). Hueso rótula o patela. [www.fisioterapia-online.com](http://www.fisioterapia-online.com). <https://www.fisioterapia-online.com/glosario/hueso-rotula-o-patela#:~:text=%C2%BFQu%C3%A9%20funci%C3%B3n%20tiene%20la%20r%C3%B3tula,el%20roce%20del%20tend%C3%B3n%20rotuliano>

Diferentes ejes, planos y movimientos del cuerpo. (2022, December 22). Centro Impulso. <https://centroimpulso.es/ejes-y-ejercicio/>

Karatsidis, A., Bellusci, G., Schepers, H. M., De Zee, M., Andersen, M. S., & Veltink, P. H. (2016, December 31). Estimation of Ground Reaction Forces and Moments During Gait Using Only Inertial Motion Capture. Sensors; Multidisciplinary Digital Publishing Institute. <https://doi.org/10.3390/s17010075>

Three sources, three receivers, six degrees of freedom: An ultrasonic sensor for pose estimation & motion capture. (2015, November 1). IEEE Conference Publication | IEEE Xplore. <https://ieeexplore.ieee.org/abstract/document/7370689>

Malanga, G. (2005). Clinical Observation. <https://www.semanticscholar.org/paper/Clinical-Observation-Malanga-Delisa/37a22726bb0189632897fdbb9e81d1f22bb2ab60#citing-papers>

Mariño, M. (2023). Los 10 mejores lenguajes de programación en 2023. El Blog De Dinahosting. <https://dinahosting.com/blog/mejores-lenguajes-programacion-2023/>

S, J. (2017, October 3). Maniobras Exploratorias de Cadera y Pelvis para Ortopedas - Artículos de Ortopedia. Artículos De Ortopedia. <http://articulosdeortopedia.com/novedades/maniobras-exploratorias-de-cadera-y-pelvis-para-ortopedas/>

Huesos del pie y el tobillo. (2021, July 29). Mayo Clinic. <https://www.mayoclinic.org/es/diseases-conditions/broken-ankle/multimedia/foot-and-ankle-bones/img-20008997>

Home | Movella.com. (2023). <https://www.movella.com/>

What is NumPy? — NumPy v1.26 Manual. (2023). <https://numpy.org/doc/stable/user/whatisnumpy.html>

Intro to data structures — pandas 2.1.0 documentation. (2023). [https://pandas.pydata.org/docs/user\\_guide/dsintro.html#dsintro](https://pandas.pydata.org/docs/user_guide/dsintro.html#dsintro)

Pirker, W., & Katzenschlager, R. (2016). Gait disorders in adults and the elderly. *Wiener Klinische Wochenschrift*; Springer Science+Business Media. <https://doi.org/10.1007/s00508-016-1096-4>

Python API reference for plotly — 5.17.0 documentation. (2023). <https://plotly.com/python-api-reference/>

Welcome to Python.org. (2023). Python.org. <https://www.python.org/>

Robert-Lachaine, X., Mecheri, H., Larue, C., & Plamondon, A. (2016). Validation of inertial measurement units with an optoelectronic system for whole-body motion analysis. *Medical & Biological Engineering & Computing*, 55(4), 609–619. <https://doi.org/10.1007/s11517-016-1537-2>

Triatlón, R. P. (2023). *Tibia y peroné: guía completa* - Planeta Triatlón. Planeta Triatlón. <https://planetatriatlon.com/tibia-y-perone/>

Signal Processing (scipy.signal) — SciPy v1.11.2 Manual. (2023). <https://docs.scipy.org/doc/scipy/tutorial/signal.html>

Selfa, D. M., Carrillo, M., & Boone, M. del R. (2006). A Database and Web Application Based on MVC Architecture. *16th International Conference on Electronics, Communications and Computers (CONIELECOMP'06)*. <https://ieeexplore.ieee.org/document/1604744>

Serrano, C. (2023). Fémur. Kenhub. <https://www.kenhub.com/es/library/anatomia-es/femur-anatomia>

Welcome to Spyder's Documentation — Spyder 5 documentation. (2023). <https://docs.spyder-ide.org/current/index.html>

De Los Santos, P. R. (2023). Las 5 razones por las que todo el mundo quiere aprender Python. *Telefónica Tech*. <https://telefonicatech.com/blog/las-5-razones-por-las-que-todo-el-mundo-quiere-aprender-python>

Teseo Noticias (2023). ¿Qué es y cómo funciona la captura de movimiento? | Teseo Noticias. <https://teseo.es/noticias/que-es-y-como-funciona-la-captura-de-movimiento/>

Arias, E. T. (2023). *La pelvis: anatomía y biomecánica*. Esteban Tierra Arias. <https://estebantierratrainer.es/la-pelvis-anatomia-y-biomecanica/>

Vicon | Award Winning Motion Capture Systems. (2023, September 4). Vicon. <https://www.vicon.com/>

Vicon Motion Systems. (2016). *Vicon Vantage Reference*. <https://www.prophysics.ch/wp-content/uploads/2017/06/ViconVantageReference.pdf>

Viladot Voegeli, A (2003). Anatomía funcional y biomecánica del tobillo y el pie. *Revista Española De Reumatología*. <https://www.elsevier.es/es-revista-revista-espanola-reumatologia-29-articulo-anatomia-funcional-biomecanica-del-tobillo-13055077>

Yabukami, S., Kikuchi, H., Yamaguchi, M., & Arai, K. (2000). Motion capture system of magnetic markers using three-axial magnetic field sensor. *IEEE Transactions on Magnetics*, 36(5), 3646-3648.