



**Diseño de una interfaz usando Python para la identificación de  
asimetrías temporales en la marcha de personas con prótesis de  
miembro inferior**

Jeysson Alfonso Calderón Contreras

10441218173

**Universidad Antonio Nariño**

Programa Ingeniería Electrónica

Facultad de Ingeniería Mecánica, Electrónica y Biomédica

Bogotá-Colombia

2023

**Diseño de una interfaz usando Python para la identificación de  
asimetrías temporales en la marcha de personas con prótesis de  
miembro inferior**

Jeysson Alfonso Calderón Contreras

10441218173

Proyecto de grado presentado como requisito parcial para optar al título de:  
**Ingeniero Electrónico**

**Directores:**

UAN: Jazmín González Ocampo

Externo: Sebastián Jaramillo Isaza

**Universidad Antonio Nariño**

Programa Ingeniería Electrónica

Facultad de Ingeniería Biomédica, Electrónica y Mecánica

Bogotá-Colombia

2023

## NOTA DE ACEPTACIÓN

El trabajo de grado titulado

\_\_\_\_\_.

Cumple con los requisitos para optar

Al título de \_\_\_\_\_.

\_\_\_\_\_  
Firma del Tutor

\_\_\_\_\_  
Firma Jurado

\_\_\_\_\_  
Firma Jurado

Bogotá D.C, 26 de abril de 2023.

## Contenido

	<b>Pág.</b>
<b>Resumen</b>	<b>9</b>
<b>Abstract</b>	<b>10</b>
<b>Capítulo 1 – Introducción</b>	<b>11</b>
1.1    Introducción	11
1.2    Antecedentes	15
1.3    Planteamiento del problema	20
1.4    Justificación	24
1.5    Objetivos	28
1.5.1    Objetivo General	28
1.5.2    Objetivos Específicos	28
<b>Capítulo 2 – Marco teórico y legal</b>	<b>29</b>
2.1    Marco teórico	29
Biomecánica	29
Biomecánica de la marcha	30
Marcha normal	32
Marcha patológica	34
Prótesis de miembro inferior	35
Asimetrías temporales en pacientes con prótesis	37
Protocolos para el análisis de marcha	39
Unidad de medición inercial	41
Métodos de postratamiento de datos cinemáticos	42
Lenguaje de programación Python	43
Aplicaciones de Python en el campo Biomédico	44
2.2    Marco legal	46
<b>Capítulo 3 – Metodología</b>	<b>48</b>
3.1    Etapa 1: Identificación de las variables y parámetros	49
3.2    Etapa 2: Diseño de la interfaz gráfica en Python	50
3.3    Etapa 3: Pruebas de funcionamiento de la interfaz	51
3.4    Etapa 4: Validación de la usabilidad de la interfaz	53

<b>Capítulo 4 – Resultados</b>	<b>56</b>
4.1 Etapa 1: Identificación de las variables y parámetros	56
4.2 Etapa 2: Diseño de la interfaz gráfica en Python	59
4.3 Etapa 3: Pruebas de funcionamiento de la interfaz	67
4.5 Etapa 4: Validación de la usabilidad de la interfaz	69
<b>Capítulo 5 – Conclusiones</b>	<b>73</b>
5.1 Conclusiones	73
5.2 Recomendaciones	73
5.3 Trabajos Futuros	74
<b>Bibliografía</b>	<b>76</b>
<b>Anexos</b>	<b>80</b>

## Lista de Figuras

<b>Figura 1</b> Movimiento del tobillo en el plano sagital y frontal .....	31
<b>Figura 2</b> Músculos que participan en la marcha protésica.....	32
<b>Figura 3</b> Subdivisiones de la fase de apoyo.....	33
<b>Figura 4</b> Subdivisiones de la fase de balanceo .....	33
<b>Figura 5</b> Prótesis transfemorales y transtibiales .....	36
<b>Figura 6</b> Ciclograma de marcha, en personas con amputación de miembro inferior y sin amputación.....	39
<b>Figura 7</b> Posicionamiento de marcadores en el protocolo de Davis .....	40
<b>Figura 8</b> Posición de marcadores en protocolo Hayes.....	41
<b>Figura 9</b> Diagrama fases de la metodología.....	55
<b>Figura 10</b> Set de marcadores implementado.....	57
<b>Figura 11</b> Ubicación de los sensores inerciales .....	57
<b>Figura 12</b> Base de datos de sensores.....	58
<b>Figura 13</b> Amplitud articular promedio de la rodilla izquierda.....	59
<b>Figura 14</b> Diagrama de flujo de interfaz gráfica.....	61
<b>Figura 15</b> Delimitación de la base de datos por medio de “;” .....	62
<b>Figura 16</b> Ángulos tomados de la base de datos .....	62
<b>Figura 17</b> Página 1 de la interfaz gráfica – “Análisis de marcha” .....	63
<b>Figura 18</b> Página 2 de la interfaz gráfica – “Análisis de filtros” .....	63
<b>Figura 19</b> Tabla implementada en la interfaz de Cálculo de métricas.....	66
<b>Figura 20</b> Interfaz gráfica - Análisis de marcha .....	67
<b>Figura 21</b> Interfaz gráfica – Análisis de filtros con filtro de media móvil .....	68
<b>Figura 22</b> Nivel de educación de los encuestados .....	69
<b>Figura 23</b> Conocimientos de ingeniería de los encuestados .....	70
<b>Figura 24</b> Resultado de primera pregunta planteada a los encuestados en Diagrama Circular .....	70
<b>Figura 25</b> Resultado de segunda pregunta planteada a los encuestados en Diagrama Circular .....	70
<b>Figura 26</b> Resultado de tercera pregunta planteada a los encuestados en Diagrama Circular .....	71
<b>Figura 27</b> Resultado de cuarta pregunta planteada a los encuestados en Diagrama Circular.....	71
<b>Figura 28</b> Resultado de quinta pregunta planteada a los encuestados en Diagrama Circular.....	71

## Lista de Tablas

<b>Tabla 1</b> Variables seleccionadas .....	55
<b>Tabla 2</b> Resultados en porcentaje de la encuesta .....	66

## ***Dedicatoria***

Agradezco a Dios por la oportunidad que me brindó de culminar mis estudios, a mi madre que estuvo confiando en mis capacidades y luchando día a día para lograr el objetivo, mi novia quien se esforzó y no me dejó rendir apoyándome en los momentos más difíciles y demás familiares por el apoyo en cada uno de mis pasos. A cada uno de los profesores por compartir su conocimiento y a todas las personas que estuvieron en esta etapa de mi vida. A la Universidad Antonio Nariño en especial a la Facultad Sur por abrirme sus puertas para ser parte de ella. Por último, a mis tutores Sebastián Jaramillo y Jazmín González por apoyarme y brindarme su conocimiento para dar cierre a este ciclo educativo.

## Resumen

Las amputaciones en extremidades inferiores provocan cambios irreversibles y modifican el ciclo de la marcha, conllevando con esto, a la disminución en la capacidad autónoma de locomoción en las personas que lo sufren. La marcha es un movimiento sincrónico de los miembros inferiores, cuando se produce una alteración en sus determinantes aparece el concepto de asimetría. Para lograr identificar estas asimetrías en condiciones de marcha natural, es posible utilizar las unidades de medición inercial (IMU) como sistema de captura de movimiento. Los IMU son pequeños dispositivos electrónicos capaces de cuantificar parámetros cinemáticos como la aceleración lineal, velocidad angular y la posición. El objetivo de este proyecto es construir una interfaz usando Python que permita el monitoreo e identificación directa de asimetrías temporales en la marcha de personas con prótesis de miembro inferior. Esto serviría como elemento de prevención de lesiones, apoyo en la mejora en la adherencia al dispositivo protésico y como indicador para intervención temprana o monitoreo continuo en pacientes con prótesis.

**Palabras clave:** *Cuantificación de asimetría, marcha protésica, interfaz, unidad de medición inercial.*

## **Abstract**

Lower limb amputations cause irreversible changes and modify the gait cycle, leading to a decrease in the autonomous locomotion capacity of affected individuals. Gait is a synchronous movement of the lower limbs, and any alteration in its determinants results in the concept of asymmetry. In order to identify these asymmetries in natural walking conditions, it is possible to use inertial measurement units (IMUs) as a motion capture system. IMUs are small electronic devices capable of quantifying kinematic parameters such as linear acceleration, angular velocity, and position. The objective of this project is to construct a Python interface that allows for direct monitoring and identification of temporal asymmetries in the gait of individuals with lower limb prostheses. This will serve as an element of injury prevention, support for improving adherence to the prosthetic device, and as an indicator for early intervention or continuous monitoring in patients with prostheses.

Keywords: Inertial measurement unit, interface, prosthetic gait, quantification of asymmetry.

## **Capítulo 1 – Introducción**

### **1.1 Introducción**

De acuerdo con la Organización Mundial de la Salud se estima que más de 1.000 millones de personas presentan una condición en salud discapacitante a nivel mundial, de los cuales se tiene un estimado que el 15% de estos presentan alguna alteración que puede afectar su autonomía (Organización Mundial de la Salud, 2011). De igual manera la Organización Mundial de la Salud afirma que, de este número, solo entre el 26% y el 55% reciben procesos de rehabilitación médica. Aunque las lagunas en la información son notorias, bajo este sentido, se puede afirmar que el acceso a una rehabilitación integral prolongada es una meta y un aspecto para identificar y trabajar en la actualidad (Organización Mundial de la Salud, 2015).

En este sentido, uno de los principales requerimientos en cuanto a las personas con discapacidad es la generación de autonomía. Según el estudio de Águila (2017) para el Instituto de Democracia y Derechos Humanos, este requerimiento debe fungir como factor rector de trabajo en el desarrollo de equidad entre las personas, tanto a nivel social, como laboral. Por lo tanto, se espera que los organismos gubernamentales responsables adopten estrategias mediante planes de acción, para mejorar la participación, y la capacidad en cuanto a procesos de educación, de capacitación e incursión laboral. Así mismo se mejora la recopilación de datos (la cual es precaria para algunas de las personas que no hacen parte de procesos de rehabilitación actual) y la investigación científica en países poco desarrollados o en vía de desarrollo.

Allí cobra relevancia factores acerca de la identificación de los tipos de discapacidad, el diagnóstico de la discapacidad, el tratamiento que se realice, los equipos interdisciplinarios relacionados y los pronósticos afines, así como los compromisos en torno al desempeño y realización, que hace referencia a las responsabilidades y acuerdos que se establecen entre el profesional de la salud y el paciente o su familia, en relación con los objetivos y metas a alcanzar en el proceso de rehabilitación y tratamiento de la discapacidad. Estos compromisos pueden incluir aspectos como la participación del paciente en el proceso de rehabilitación, el seguimiento del plan de tratamiento prescrito, la asistencia regular a las citas médicas y terapias, entre otros. En su estudio Vargas (2019) plantea que un acceso en el momento correcto y óptimo a diversos servicios de rehabilitación puede potencializar favorablemente el empoderamiento, el autocuidado, la autodeterminación y el mantenimiento de diferentes entornos saludables en los individuos, las familias y las comunidades.

Dentro de la población en condición de discapacidad, las personas con prótesis representan un porcentaje importante. En países latinoamericanos como Colombia, la mayoría de esta población padece condiciones de difícil acceso a dispositivos de asistencia y a procesos de rehabilitación integral.

De acuerdo con la Guía de Práctica Clínica del Ministerio de Salud de Colombia para el diagnóstico y tratamiento preoperatorio, intraoperatorio y postoperatorio de la persona amputada, la prescripción de la prótesis y la rehabilitación integral es necesario pensar en un conjunto de intervenciones sicosociales, el acompañamiento de la familia, la disminución de barreras para la integración social, el entrenamiento en marcha y la adaptación a las prótesis, el reacondicionamiento físico, los ejercicios de fortalecimiento,

las adaptaciones ergonómicas y por último en la independencia en las actividades de la vida diaria, movilidad y trasferencias (Ministerio de Salud de Colombia, 2015).

El concepto de la asimetría en la marcha es de vital importancia para la caracterización de la marcha protésica. Este tipo de marcha patológica se caracteriza por desfase del correcto andar de las personas. En este sentido, Arellano-González et al. (2019) afirman que los estudios a desarrollarse deben estar caracterizados por el foco inicial sobre los parámetros espacial, temporal, angular, de fuerza y otros, a partir de la determinación de estudio de movimientos disperejos entre los miembros inferiores. Alteraciones en dichos parámetros generan consigo desviaciones en torno a la velocidad de la marcha, la velocidad de la zancada, la longitud del paso, la longitud de la zancada, la cadencia, el ancho del paso, el ángulo del paso, las fases de la marcha y postura del cuerpo, las cuales dan una idea general del grado de calidad de la marcha de los pacientes.

La detección temprana de estas asimetrías temporales permitirá al médico tratante y al personal de salud, realizar intervenciones tempranas sobre la alineación del dispositivo protésico, impactando positivamente tanto la adherencia terapéutica del paciente al uso del dispositivo como los resultados del proceso de rehabilitación y de reeducación funcional.

Teniendo en cuenta esto, este proyecto busca contribuir al bienestar y calidad de vida de las personas con prótesis de miembro inferior, a través de la construcción de una interfaz que permita monitorear y detectar asimetrías temporales en su marcha. Esto permitirá prevenir lesiones, brindar al personal médico herramientas para la toma asistida de decisiones especialmente aquellas relacionadas con el esfuerzo terapéutico, además de generar información útil para el monitoreo, seguimiento y ajuste de los tratamientos de

rehabilitación integral. De esta manera, se espera contribuir a la generación de autonomía en esta población y fomentar la equidad social y laboral en su inclusión en la sociedad.

El documento se encuentra estructurado en varios apartados que buscan abordar el tema de la biomecánica de la marcha en pacientes con prótesis de miembro inferior, desde diferentes perspectivas. En la introducción, se plantea el contexto del estudio y se presentan los antecedentes que han motivado su realización. Posteriormente, se establece el planteamiento del problema y la justificación para su abordaje. Los objetivos del estudio se dividen en general y específicos, para orientar la investigación hacia metas claras.

El marco teórico se encuentra dividido en dos secciones, donde se aborda la biomecánica en general y la biomecánica de la marcha en particular, tanto en su versión normal como patológica. Posteriormente, se profundiza en el tema de las prótesis de miembro inferior, haciendo énfasis en las asimetrías temporales que se presentan en los pacientes que las utilizan. Se presentan los protocolos para el análisis de marcha y se introduce la unidad de medición inercial y los métodos de postratamiento de datos cinemáticos, con un enfoque en el lenguaje de programación Python y sus aplicaciones en el campo biomédico.

Finalmente, se incluyen el marco legal y la metodología que se ha seguido en el estudio, dividida en cinco etapas desde la identificación de las variables hasta la validación de la interfaz de programación en Python. Los resultados se presentan en un apartado independiente, seguido de las conclusiones y recomendaciones, la bibliografía y los anexos.

## 1.2 Antecedentes

De acuerdo con Molina (2021) se pueden llegar a identificar dos tipos de asimetrías más comunes, la espacial y la temporal y parámetros relacionados a estas que son necesarios en el desarrollo de los estudios para maximizar su comprensión. Entre estos se destacan el tiempo de apoyo, el tiempo de balanceo, el número de Froude (la relación entre la fuerza inercial con la fuerza gravitatoria), la longitud de paso, la longitud paso completo o zancada, así como la cadencia y la velocidad de marcha.

Con relación a este aspecto, podemos encontrar el estudio de Olivares et al. (2011) en donde se asevera que el desequilibrio de la marcha insinúa ser una causa de las enfermedades degenerativas de la cadera. En concordancia con Romero y Barrios (2017), las herramientas de medición y proyección son una opción adecuada para obtener desviaciones respecto del patrón de normalidad en la marcha de las personas, en especial aquellas con discapacidad, por lo que es necesario el desarrollo de análisis cada vez más rigurosos para cuantificar anomalías con el fin de determinar acciones correctivas que apunten a la mejora de la calidad de vida.

Para Ogihara et al. (2020) el estudio de la asimetría de la marcha como indicador de alteración se puede desarrollar mediante el uso de herramientas cinemáticas, las cuales pueden aportar con claridad el reflejo de la función física, el equilibrio y las actividades del día a día de las personas, encontrando una posibilidad grande en la importancia de utilizar la evaluación de la asimetría de la marcha cinemática en pacientes con afectaciones significativas. De acuerdo con el centro de Biomecánica de Apple (2021), en su estudio de evaluación de las aplicaciones de iPhone, las mediciones de la movilidad permiten a los investigadores evaluar la movilidad de usuarios, con las cuales se puede proporcionar

medidas objetivas que ayuden a reafirmar las pruebas funcionales especializadas y los cuestionarios clínicos.

Para Gómez et. al. (2018) la estimación de estas asimetrías es posible gracias a los sistemas de captura de movimiento, tanto con videografía con los sistemas MoCap como son los sistemas basados en Unidades de Medida Inercia (IMU). En el tiempo reciente y gracias a la mejora, portabilidad y disminución en el costo de sistemas de captura de movimiento, se ha visto un aumento en la cuantificación de este tipo de parámetros, buscando mejorar la adherencia de los pacientes a sus dispositivos protésicos.

En este sentido, son varios los estudios predecesores, entre lo que se pueden destacar aquellos generados por la línea de Investigación Ingeniería de Rehabilitación y Discapacidad, del Grupo de Investigación Bio-Ingeniería de la Facultad de Ingeniería Mecánica, Electrónica y Biomédica (FIMEB) de la Universidad Antonio Nariño (Universidad Antonio Nariño, 2022) y que sirven de base para proyectos e iniciativas como la que se busca desarrollar en la presente propuesta. Así como estudios de cuantificación en otras instituciones, tal es el caso de Silva (2015), quien, mediante resultados cinéticos, cinemáticos y de activación muscular de la simulación, analiza la diferencia en la comparación entre los resultados obtenidos de marcha normal y marcha con prótesis. De igual forma Vasco (2022) en su estudio desarrolla y evalúa un sistema alternativo para la generación de indicadores de simetría en sujetos con prótesis de miembro inferior utilizando sensores IMU, por su versatilidad y validez.

La determinación de asimetrías en la marcha es de gran importancia en el campo de la rehabilitación y el seguimiento a pacientes con prótesis o enfermedades degenerativas. La identificación y cuantificación de las asimetrías permite evaluar el patrón de normalidad en

la marcha de las personas y detectar anomalías que pueden llevar a la aparición de enfermedades degenerativas de la cadera, así como mejorar la calidad de vida de los pacientes.

Para la determinación de las asimetrías en la marcha, existen diversas tecnologías que se pueden utilizar. Una de ellas son los sistemas de captura de movimiento, como la videografía con los sistemas MoCap y los sistemas basados en Unidades de Medida Inercia (IMU). Estos sistemas permiten cuantificar de manera precisa los parámetros relacionados con la marcha, como el tiempo de apoyo, el tiempo de balanceo, la longitud de paso, la cadencia y la velocidad de marcha. Además, gracias a la mejora en la portabilidad y la disminución en el costo de estos sistemas, se ha visto un aumento en la cuantificación de este tipo de parámetros, buscando mejorar la adherencia de los pacientes a sus dispositivos protésicos.

Otra tecnología que se utiliza para la determinación de asimetrías en la marcha es el uso de herramientas cinemáticas, las cuales pueden aportar con claridad el reflejo de la función física, el equilibrio y las actividades del día a día de las personas. Además, las mediciones de la movilidad mediante el uso de aplicaciones de iPhone también permiten evaluar la movilidad de los usuarios, proporcionando medidas objetivas que ayuden a reafirmar las pruebas funcionales especializadas y los cuestionarios clínicos.

Así mismo el autor Peña et. al. (2018), indica la importancia de analizar el grado de aprovechamiento de energía mecánica, la estabilidad dinámica y las variables cinemáticas de interés clínico en la marcha de un paciente con amputación transfemoral que no realizó el proceso de rehabilitación, para este trabajo se implementó la reconstrucción 3D, que permitió la cuantificación de valores angulares para: cadera, rodilla y tobillo. (Peña, 2018).

Según un estudio realizado en la Universidad Nacional de Colombia, se realizó el análisis del proceso de marcha de un paciente con amputación trasfemoral por medio de la captura de movimiento en un laboratorio de marcha y con el Software OpenSim desarrollado por la universidad de Stanford, el cual permite simular la cinemática inversa, fuerzas de reacción tridimensionales y activaciones musculares, para con ello identificar las contribuciones de la prótesis al caminar y diferentes tareas realizadas por el cuerpo. (Silva, 2015)

Desde el punto de vista de la rehabilitación y el seguimiento a pacientes, la determinación de asimetrías en la marcha también puede ser evaluada mediante métodos convencionales como la observación directa y la evaluación subjetiva del terapeuta. Sin embargo, estos métodos pueden presentar ciertas limitaciones en cuanto a la precisión y objetividad de los resultados obtenidos.

Por otro lado, la inteligencia artificial se está convirtiendo en una herramienta cada vez más importante para el análisis de grandes conjuntos de datos médicos y la generación de modelos predictivos para la detección temprana de desequilibrios y enfermedades degenerativas. La capacidad de la inteligencia artificial para procesar grandes cantidades de información y encontrar patrones complejos en los datos es fundamental para el desarrollo de técnicas de diagnóstico temprano. Los algoritmos de aprendizaje automático pueden ser entrenados con grandes cantidades de datos para identificar patrones y características asociados con enfermedades específicas. Los avances en la inteligencia artificial están permitiendo a los médicos y los investigadores acceder a información y conocimientos que antes eran imposibles de obtener, lo que lleva a una mejora en la atención médica y en la calidad de vida de los pacientes.

Gracias a los avances en métodos computacionales y software de libre acceso, actualmente para la construcción de una interfaz para el procesamiento de variables cinemáticas de interés entre ellas las asimetrías, existen diversas alternativas como Python, Matlab, R entre otras. De estas, la herramienta Python, se presenta como un recurso de fácil acceso gracias a la licencia campus con la que cuenta la Universidad Antonio Nariño. Esta herramienta permitirá desarrollar una interfaz amigable y de fácil de uso.

### **1.3 Planteamiento del problema**

No es sencillo poder garantizar una óptima adecuación de las personas con discapacidad en su movimiento, sin antes realizar un análisis completo de la situación, con el fin de entender las problemáticas mediante el diagnóstico de los patrones de funcionamiento de los cuerpos del segmento de estudio. Sin olvidar, que los sistemas de captura de movimiento basados en IMU entregan datos crudos que deben ser procesados para poder cuantificar e identificar las asimetrías, en donde se hace ideal el poder contar con interfaces de procesamiento que permitan facilitar la interacción con el sistema de usuarios o técnicos o personal asistencial no especializados en el cálculo o procesamiento de datos.

Para el caso específico de las personas que tienen prótesis de miembro inferior, el ejercicio regularmente parte por el entendimiento de sus patrones de marcha. En este sentido, regularmente con ayuda de herramientas mecánicas se busca dar igualdad entre las personas consideradas sanas y aquellas con discapacidad, sin embargo, Broche et al. (2015) en su estudio evidencian diferencias notables en los patrones de marcha entre sujetos normales y pacientes amputados.

Generalmente los sistemas ópticos son poco eficientes respecto a los sistemas con sensores inerciales ya que tienden a caer en la falla de marcadores que necesariamente estén unidos al cuerpo, además de contar con un tiempo considerable de trabajo para una sesión experimental, espacios limitados de trabajo y restricciones puntuales de medición por condiciones tecnológicas.

Es importante determinar, que a pesar de todos los beneficios que pueda tener el uso de herramientas de medición y apoyo a las asimetrías, el acceso a estas herramientas, sobre todo las especializadas tiende a ser costoso para el público en general, repercutiendo con

mayor impacto, en la accesibilidad y asequibilidad a este tipo de recursos. Son varios los casos en que los pacientes y principalmente aquellos con prótesis de miembro inferior no son monitoreados constantemente, conllevando con ello al agravamiento de las patologías, al posible rechazo al dispositivo de asistencia y la posibilidad de generación de nuevos inconvenientes en la movilidad.

Esto conlleva a pensar en cierto abandono en los tratamientos, algo que puede derivarse según Franco-Giraldo (2014) por el favorecimiento del sistema de salud al mercado financiero transnacional, lo que a su vez conlleva a tener inequidad, ineficiencia y deterioro de las condiciones de la salud pública de los ciudadanos. En la Universidad de Antioquia un grupo de investigadores plantea que es prioritario el avance en un modelo integral de atención para que personas con amputación puedan acceder a los servicios de salud, incluyendo su seguimiento que pueda servir como punto de ancla de una reintegración social y laboral adecuada, sin discriminaciones y con condiciones que no repercutan en el agravamiento de sus lesiones (Castaño et al, 2022).

Uno de los desafíos es lograr una igualdad en los patrones de marcha entre personas sanas y aquellas con discapacidad, lo que implica un análisis completo de la situación y la identificación de las asimetrías. Además, se debe tener en cuenta que los sistemas de captura de movimiento basados en IMU entregan datos crudos que deben ser procesados, lo que hace ideal contar con interfaces de procesamiento que faciliten la interacción con el sistema para usuarios o personal asistencial no especializados en el cálculo o procesamiento de datos.

Sin embargo, el acceso a herramientas de medición y apoyo a las asimetrías puede ser costoso para el público en general, lo que puede llevar a un abandono en los tratamientos

y, en consecuencia, a la agravación de las patologías y a la posibilidad de generación de nuevos inconvenientes en la movilidad. Por lo tanto, se hace necesario buscar métodos de apoyo precisos y accesibles, que permitan tener herramientas óptimas de medición y disminuyan el sesgo de evaluación entre personas sanas y aquellas con discapacidad.

Es importante tener en cuenta que actualmente, el cálculo de las asimetrías de características temporales requiere de un postratamiento que debe ser realizado de acuerdo con los datos obtenidos a través de los sistemas de captura de movimiento. Este proceso generalmente requiere de un conocimiento especializado para su correcta implementación, lo que limita el acceso a la tecnología y a la información para personas poco capacitadas y para sistemas de salud o centros de atención y rehabilitación con recursos limitados. Por lo tanto, es importante avanzar en un modelo integral de atención que permita a las personas con amputación acceder a servicios de salud y seguimiento que puedan servir como punto de ancla de una reintegración social y laboral adecuada, sin discriminaciones y con condiciones que no repercutan en el agravamiento de sus lesiones.

Por ende, se hace ineludible la búsqueda de métodos de apoyo precisos y accesibles, con el fin de tener herramientas óptimas de medición que disminuyan el sesgo de evaluación entre las personas sanas y aquellas con discapacidad. En este contexto, se plantea la importancia de integrar tecnologías emergentes en salud, como la ingeniería biomédica, para el análisis de relevancia biomédica y para el sector salud, con el fin de desarrollar soluciones de bajo costo que permitan mejorar la calidad de vida de las personas con discapacidad y lograr una mayor adherencia al uso de prótesis.

Considerando los argumentos expuestos, se formula la siguiente pregunta de investigación: ¿Cómo se puede automatizar de forma eficiente y amigable el proceso de cuantificación de asimetrías en la marcha de pacientes con prótesis de miembro inferior?

## **1.4 Justificación**

La importancia de investigar las asimetrías en la marcha radica en el hecho de que estas pueden afectar significativamente la calidad de vida de las personas y su capacidad para realizar actividades diarias y deportivas. La marcha asimétrica puede generar desequilibrios musculares, dolor en las articulaciones y fatiga excesiva, lo que puede llevar a una disminución en la actividad física y, en última instancia, a una disminución de la calidad de vida.

Por esta razón, es esencial que se realicen estudios biomecánicos para determinar la asimetría en la marcha y cómo esta afecta la funcionalidad del individuo. Estos estudios pueden incluir análisis de los planos del cuerpo y varios segmentos en las posiciones dinámica y estática, así como cálculos y análisis de la amplitud de la articulación, el centro de masa y la variación de esta durante la marcha.

Teniendo en cuenta la desigualdad de oportunidades que sufren las personas en condición de discapacidad, se plantea la necesidad de una ayuda eficiente para que puedan tener procesos de rehabilitación apropiados que permitan adherirse de una manera correcta al uso de prótesis mediante métodos que faciliten un monitoreo más frecuente de los pacientes con prótesis y que propicien intervenciones tempranas y poco traumáticas que busquen corregir las situaciones que generan dichas asimetrías. Que vayan más allá del uso de sistemas basados en IMU, los cuales no entregan dicha información procesada, y en donde se hace necesaria la construcción de interfaces para facilitar el cálculo y la identificación de los parámetros asociados, que posibilite la estimación de asimetrías en la marcha protésica, que puede convertirse en una herramienta que por un lado facilite el

seguimiento de los pacientes, evite la demora en el diagnóstico de posibles asimetrías y que, por el otro, sirva a la masificación del uso de este tipo de tecnologías en nuestro país.

La implementación de tecnologías emergentes en salud es fundamental para mejorar la calidad de vida de las personas con discapacidad y garantizar un cubrimiento más amplio en salud. En este sentido, es necesario promover la investigación y el desarrollo de tecnologías que permitan monitorear de manera constante la marcha y el movimiento de los pacientes con prótesis, con el fin de detectar posibles asimetrías en la marcha y prevenir complicaciones en su salud.

El abandono del recurso terapéutico por parte del paciente con prótesis es otro de los grandes problemas que se presentan en la rehabilitación de las personas con discapacidad. Muchos pacientes, debido a la falta de apoyo y seguimiento constante, abandonan el uso de su prótesis y se ven obligados a renunciar a las actividades que les gustan o a limitar su movilidad. Es por ello que resulta importante la implementación de tecnologías que permitan un monitoreo constante de la marcha, así como la detección temprana de posibles asimetrías y la implementación de intervenciones tempranas y poco traumáticas para corregirlas.

Por otro lado, la falta de monitoreo constante de los pacientes con prótesis puede representar un riesgo para su salud. Las asimetrías en la marcha, por ejemplo, pueden generar desequilibrios musculares, dolor en las articulaciones y fatiga excesiva, lo que puede llevar a una disminución en la actividad física y, en última instancia, a una disminución de la calidad de vida. Es por ello por lo que la implementación de tecnologías que permitan un monitoreo constante de la marcha y la detección temprana de posibles asimetrías resulta fundamental para garantizar la salud y el bienestar de los pacientes con prótesis.

De igual manera, cabe resaltar que este proyecto se encuentra dentro de las líneas de interés del grupo de investigación Bioingeniería de la facultad FIMEB. Adicionalmente, este proyecto nace de la experiencia previa de un grupo de estudiantes del semillero de Bioingeniería y Robótica del programa de Ingeniería Biomédica de la Universidad Antonio Nariño (Universidad Antonio Nariño, 2022) que han generado cuantificaciones previas de asimetrías de la marcha usando información de capturas de movimiento en pacientes con prótesis de miembro inferior. Por lo que se pudiese categorizar el proyecto, dentro de un marco de continuidad, ya que se contribuye al avance del grupo, permitiendo tener una interfaz que a personas no cualificadas les sirve de soporte para el uso de sus apoyos sin necesidad de tener un programador de tiempo completo.

El uso del lenguaje de programación Python, brinda eficiencia, oportunidad de ejecutarse en diferente software y permite que se optimice el desarrollo. Adicionalmente, Python cuenta con una gran variedad de librerías que admiten el procesamiento de gran número de datos y teniendo en cuenta el objetivo general del proyecto el cual es el procesamiento de la información, se tomó la decisión desde la etapa de planteamiento del proyecto elaborar la interfaz gráfica en el lenguaje de programación Python con licencia de uso gratuito.

Finalmente, a nivel personal, los ingenieros electrónicos deben contar con habilidades y conocimientos en áreas relacionadas con la programación y el desarrollo de interfaces y aplicaciones. Esto se debe a que estas tecnologías emergentes están teniendo un impacto significativo en la calidad de vida de las personas, especialmente en el campo de la salud. Al desarrollar estas competencias, los ingenieros electrónicos pueden contribuir al desarrollo de soluciones innovadoras que ayuden a mejorar la calidad de vida de las personas

y a enfrentar los desafíos en materia de salud que enfrenta nuestra sociedad. Además, estas competencias pueden abrir nuevas oportunidades profesionales y de investigación en un mundo cada vez más digitalizado.

## **1.5 Objetivos**

### **1.5.1 Objetivo General**

Diseñar una interfaz digital utilizando el software Python que permita el procesamiento de información para la identificación de asimetrías temporales de la marcha en personas con prótesis de miembro inferior.

### **1.5.2 Objetivos Específicos**

1. Identificar los métodos de procesamiento de variables cinemáticas asociadas a la marcha y a la determinación de asimetrías temporales.
2. Desarrollar una interfaz digital usando Python que permita el procesamiento de información cinemática adquirida con un sistema de captura de movimiento basado en unidades de medida inercial para la determinación de asimetrías temporales.
3. Establecer el flujo de trabajo de la interfaz para la cuantificación de asimetrías temporales de la marcha en pacientes con prótesis de miembro inferior.
4. Evaluar la usabilidad de la interfaz utilizando encuestas para el levantamiento de la información métodos digitales de recopilación de información.

## Capítulo 2 – Marco teórico y legal

### 2.1 Marco teórico

#### Biomecánica

La biomecánica es uno de los conceptos que soportan el estudio de la marcha y las asimetrías. La disciplina que se ocupa del estudio de sistemas biológicos, como el cuerpo humano, es conocida como biomecánica (Navarro, 2016). En el contexto de este proyecto, resulta relevante abordar la biomecánica debido a que una de las áreas de estudio se enfoca en el movimiento de los miembros inferiores.

Además, se puede definir la biomecánica como la disciplina que analiza los movimientos del cuerpo humano y los aspectos fisiológicos y mecánicos implicados en el proceso del movimiento. Se conocen dos tipos de biomecánica (Navarro, 2016):

- a) La biomecánica estática analiza el equilibrio de los cuerpos que se encuentran en reposo o en movimiento
- b) La biomecánica dinámica se encarga de estudiar el movimiento de los cuerpos por la acción de las fuerzas que intervienen en los movimientos.

A su vez, la biomecánica dinámica se divide en:

- a) La cinemática, que estudia el movimiento donde se produce un tipo de aceleración.
- b) La cinética estudia las fuerzas que generan los movimientos.

En términos generales podemos ver que la biomecánica involucra conceptos de la física como fuerza, aceleración, movimiento o reposo.

Ahora bien, respecto a los pacientes con prótesis de miembros inferiores, en estos se debe procurar tener una armonía entre los movimientos corporales y los movimientos de las prótesis que lleva en su cuerpo (Ramírez, 2020).

Respecto a las prótesis por debajo de la rodilla, la más común es la PIE Sach, que es un dispositivo que se caracteriza por tener una articulación de tobillo fija con una quilla de cualquier material que está cubierta de un material flexible que se extiende hasta el extremo del pie y un talón de material blando; el dispositivo permite el movimiento por medio de la compresión de los materiales del cual está elaborada la prótesis (NYU, 2000).

### **Biomecánica de la marcha**

Para comenzar esta sección, es necesario abordar el primer concepto enmarcado en algunas nociones de anatomía fisiológica y mecánica del cuerpo humano. Al hablar de la marcha humana, es importante tener en cuenta los siguientes términos referentes a los planos de referencia respecto al cuerpo (Chao, 1986):

- a) Plano sagital: donde el cuerpo se divide en derecha e izquierda
- b) Plano frontal: el cuerpo se divide en parte trasera y frontal
- c) Plano transverso: en este término nos referimos al cuerpo en parte superior e inferior.

Adicionalmente, se describen algunos de los movimientos del cuerpo implícitos en dicho proceso (marcha). En el plano sagital se encuentra la flexión y extensión, específicamente la flexión dorsal y plantar que señala el movimiento de los tobillos. Por otro lado, se mencionan los movimientos de abducción y aducción que se presentan en el plano frontal y finalmente se habla de rotación externa e interna, las cuales se presentan en el plano transversal (Chao, 1986).

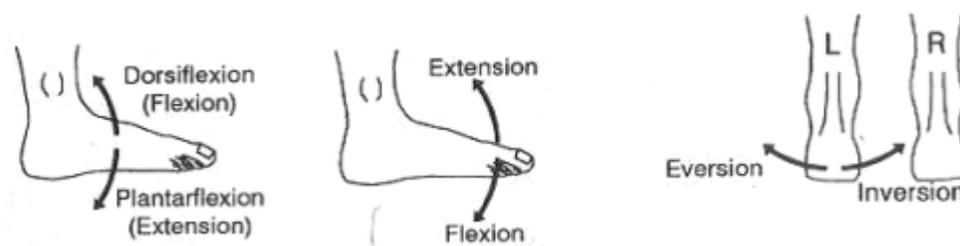
Con base en el desarrollo de los anteriores términos, se procede también a describir algunas de las articulaciones implicadas en el proceso de marcha, específicamente de las extremidades inferiores.

a) La articulación de cadera tiene que ver con la unión entre la parte superior e inferior del cuerpo, lo que constituye el sistema de locomoción y el mayor rango de movimiento está en el plano sagital (Whittle, 2007).

b) La articulación de rodilla se encarga de unir los huesos del fémur y la tibia (los huesos más largos del cuerpo); la rodilla es un factor primordial en la marcha ya que brinda estabilidad y avance al cuerpo. El movimiento de la rodilla se ocasiona en el plano sagital, además produce los movimientos de flexión y extensión (mayoritariamente) y en una menor proporción los movimientos de abducción y rotación (Whittle, 2007).

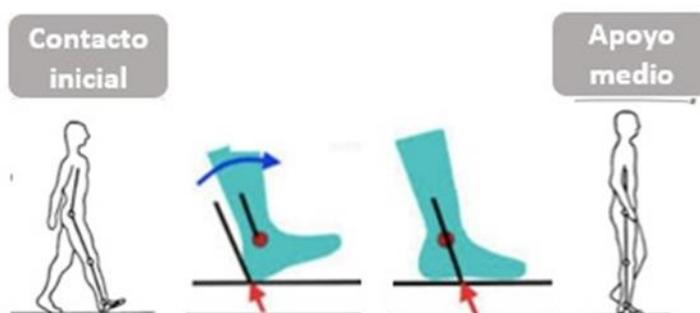
c) La articulación de tobillo involucra la unión de la tibia con el pie: esta articulación es de vital importancia para el movimiento del cuerpo ya que facilita la absorción del impacto del contacto del pie con la superficie. Su movimiento se presenta en el plano sagital durante la marcha (Whittle, 2007).

**Figura 1** Movimiento del tobillo en el plano sagital y frontal



*Nota:* Adaptado de Whittle M. (2007). Gait analysis: an introduction (4th ed.). Butterworth-Heinemann.

**Figura 2** Músculos que participan en la marcha protésica



*Nota:* La flecha indica el eje de acción de la fuerza vertical. Adaptado de Músculos que participan en la marcha – Mediprax

### **Marcha normal**

Según Whittle (2007) la marcha normal se define como un método de locomoción en el que se utilizan ambas piernas de forma alternada para proporcionar soporte y propulsión. Durante este proceso, al menos uno de los pies permanece en contacto con el suelo en todo momento, y cuando nos referimos a ciclo de marcha se define como el intervalo de tiempo en que ocurren los movimientos repetitivos al caminar.

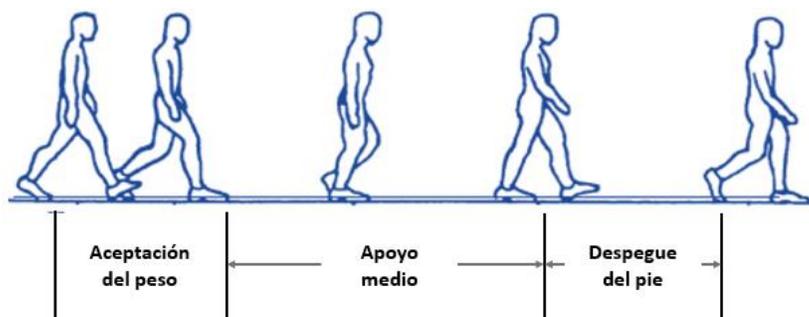
Vergara (2005) describe el proceso completo del ciclo de marcha, desde el momento en que un pie toca por primera vez la superficie hasta que ese mismo pie vuelve a tocar la misma superficie. Este proceso se representa como el 100% del ciclo de marcha, y se considera que ambas piernas tienen un comportamiento similar durante la marcha normal, lo que se conoce como simetría en los miembros. El ciclo de marcha se divide en etapas de balanceo y estancia.

Además de la marcha en su conjunto, hay subdivisiones específicas que ayudan a entender los diferentes movimientos. Estas subdivisiones incluyen el contacto inicial, el despegue del pie opuesto, la elevación del talón, el contacto inicial del pie opuesto, el despegue del pie, el avance del pie y la verticalidad de la tibia. Estas subdivisiones generan

períodos que se dividen en cuatro etapas durante la fase de estancia, cuando el pie está en contacto con el suelo, y tres etapas durante la fase de balanceo, cuando el pie se está moviendo hacia adelante.

**Fase de apoyo:** es el 60% aproximadamente del ciclo de marcha, sucede cuando una pierna se encuentra en contacto con el suelo. Esta se divide en 5 sub-fases, en las cuales se encuentran las siguientes: contacto inicial es decir el talón tiene un contacto en el suelo, respuesta o soporte de la carga, media del apoyo, elevación y arranque del talón.

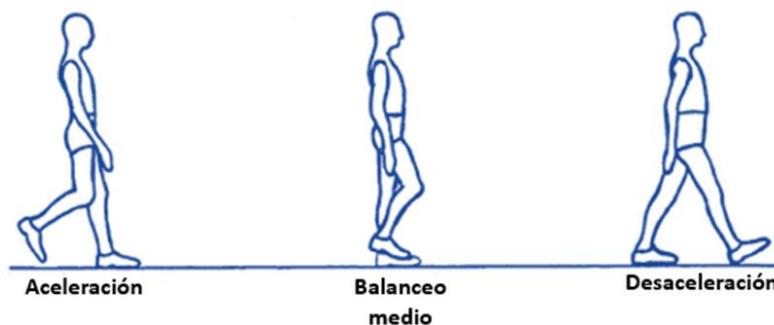
**Figura 3** Subdivisiones de la fase de apoyo



*Nota:* Adaptado de Músculos que participan en la marcha – Mediprax

**Fase de balanceo u oscilación:** se compone por el 40% restante del ciclo de marcha, hace referencia cuando uno de los pies se encuentra sin contacto con el suelo. Cuenta con tres fases, las cuales son: oscilación inicial, media y final.

**Figura 4** Subdivisiones de la fase de balanceo



*Nota:* Adaptado de Músculos que participan en la marcha – Mediprax

## **Marcha patológica**

Se define como “marcha patológica” a todas las anomalías que desarrollan patrones de movimiento o aplicación de fuerzas que no están delimitados dentro de los patrones definidos de normalidad (Herrera-Montero, et al., 2018).

Algunas de estas patologías se pueden identificar a simple vista, pero existen otras anomalías que deben ser diagnosticadas empleando sistemas de medición específicos. De acuerdo con Whittle (2007), existen cuatro elementos que determinan la forma en que una persona camina bien. Dichos aspectos son:

- a) Que cada pierna pueda soportar el peso del cuerpo de forma independiente.
- b) El balance del cuerpo debe ser mantenido tanto en forma estática como de manera dinámica con el apoyo de cada pierna.
- c) La pierna balanceada debe avanzar a otra posición donde se pueda iniciar la fase de soporte.
- d) La potencia generada debe ser suficiente para que los movimientos de las piernas puedan desarrollarse y poder mover el torso del cuerpo.

En relación con el uso de prótesis debido a la amputación de uno o ambos miembros inferiores, se produce una alteración en la marcha normal. En consecuencia, se considera la marcha protésica como una de las diferentes formas de marchas patológicas. Dicha marcha protésica puede ser afectada por algunas dificultades como lo son (Blasco, et al., 2022):

- a) Factores ambientales: tipo de terreno, intensidad de la iluminación a la que el paciente está expuesto.

- b) Factores internos del paciente: tiene que ver con el estado de ánimo del paciente (disposición para adaptarse a la prótesis, motivación personal, etc.)
- c) Factores físicos del paciente: estatura, peso, edad, sexo, estado físico.
- d) Factores específicos del amputado: tipo de amputación, tipo de prótesis y tipo de rehabilitación que se está llevando a cabo.

### **Prótesis de miembro inferior**

Estas prótesis están diseñadas para restaurar la función del miembro inferior y permitir al usuario caminar, correr y realizar actividades cotidianas con mayor facilidad y comodidad.

Existen diferentes tipos de prótesis de miembro inferior, que varían en función del nivel de amputación y las necesidades del usuario. A pesar de existir varios tipos de prótesis de miembro inferior, como sigue, las más comunes son las prótesis transtibiales y las prótesis transfemorales (Oliveira, 2011).

**Prótesis transtibiales:** También conocidas como prótesis de pierna por debajo de la rodilla, se utilizan para sustituir la parte de la pierna que se encuentra por debajo de la rodilla. Estas prótesis incluyen un socket o casquillo, una estructura de soporte, una rodilla artificial y un pie protésico (Oliveira, 2011).

**Prótesis transfemorales:** También conocidas como prótesis de pierna por encima de la rodilla, se utilizan para sustituir la parte de la pierna que se encuentra por encima de la rodilla. Estas prótesis incluyen un socket o casquillo, una estructura de soporte, una rodilla artificial y un pie protésico (Oliveira, 2011).

**Figura 5** Prótesis transfemorales y transtibiales



*Nota:* Adaptado de (Oliveira, 2011).

**Prótesis de pie:** Estas prótesis se utilizan para reemplazar el pie o los dedos del pie que se han amputado. Pueden incluir un tobillo y una estructura de soporte que se ajustan al socket o casquillo de la prótesis (Oliveira, 2011).

**Prótesis híbridas:** Estas prótesis combinan características de las prótesis transtibiales y transfemorales. Están diseñadas para proporcionar un mayor nivel de estabilidad y control en la rodilla, lo que puede ser beneficioso para personas con amputaciones más altas (Oliveira, 2011).

**Prótesis osteointegradas:** Estas prótesis se implantan directamente en el hueso del muñón del paciente. La prótesis se une al hueso mediante una técnica quirúrgica especial, lo que puede proporcionar una mayor estabilidad y reducir la presión en el muñón (Oliveira, 2011).

Las prótesis de miembro inferior están compuestas por varios componentes, como la estructura de soporte, la rodilla, el tobillo y el pie. Estos componentes se combinan para proporcionar estabilidad, amortiguación y un rango de movimiento adecuado durante la marcha y otras actividades.

En la actualidad, existen prótesis de miembro inferior avanzadas que utilizan tecnologías como la microelectrónica y la robótica para proporcionar un mayor nivel de control y adaptabilidad. Por ejemplo, algunas prótesis utilizan sensores y microprocesadores para ajustar automáticamente el ángulo y la velocidad de la articulación de la rodilla durante la marcha, lo que puede mejorar la eficiencia y reducir la fatiga (Díaz, 2016).

Sin embargo, es importante tener en cuenta que las prótesis de miembro inferior no son perfectas y pueden presentar algunas limitaciones. Por ejemplo, pueden requerir un período de adaptación y entrenamiento para aprender a usarlas de manera efectiva. Además, algunas prótesis pueden ser costosas y requieren mantenimiento y ajustes regulares.

### **Asimetrías temporales en pacientes con prótesis**

Para hablar de asimetrías es relevante hablar de marcha simétrica, definida como aquella en que el movimiento de ambas piernas (derecha e izquierda) es casi igual.

Por otro lado, para determinar la asimetría de la marcha se llevan a cabo estudios biomecánicos, los cuales se caracterizan por llevar a cabo análisis de los planos del cuerpo y varios segmentos en las posiciones dinámica y estática (Broche, et al. 2015).

Una de las formas de determinar la asimetría en la marcha involucra el cálculo junto con el análisis de la amplitud de la articulación para poder obtener una correlación entre cada lado, el centro de masa de la persona y la variación de esta durante la marcha.

Otra forma de calcular la asimetría tiene que ver con la relación entre el tiempo de apoyo y balanceo del lado afectado y no afectado (para la gráfica SR equivale a un ratio de simetría) (Broche, et al. 2015).

Fórmula para determinar la asimetría en la marcha:

$$SR_{Apoyo} = \frac{TIEMPO_{Apoyo\ afectado}}{TIEMPO_{Apoyo\ No\ afectado}} \quad (1)$$

$$SR_{Balanceo} = \frac{TIEMPO_{Balanceo\ afectado}}{TIEMPO_{Balanceo\ No\ afectado}} \quad (2)$$

*Nota:* Tomado de Molina, J (2021)

También es necesario hablar de algunos tipos de asimetrías como lo son:

a) Asimetría estructural: hablamos de asimetría estructural cuando se presenta una diferencia real en la medición desde la base del fémur hasta la articulación “tibioperonea astragalina”.

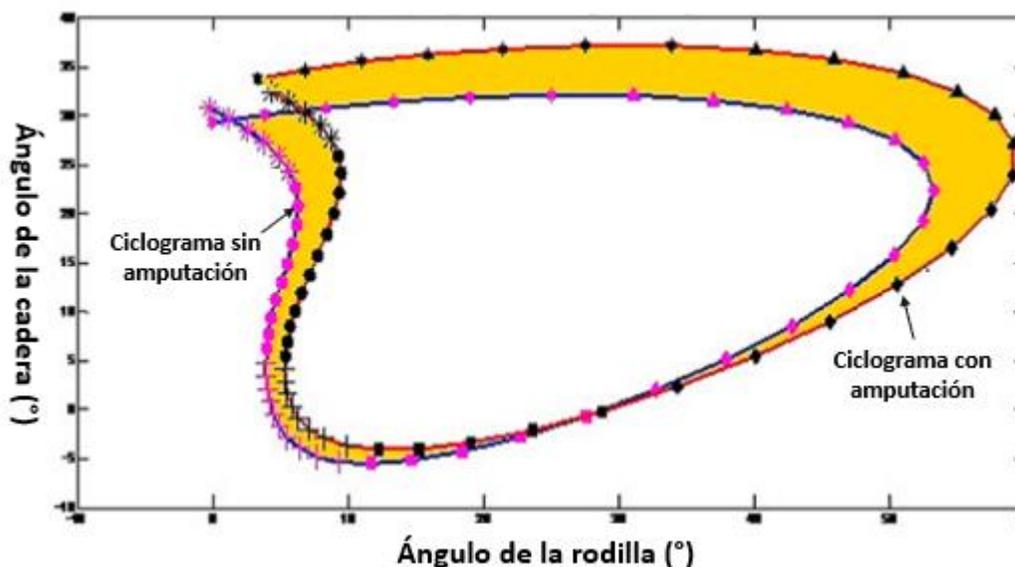
b) Asimetría funcional: Se trata de una condición que permite visualizar una discrepancia de longitud en las piernas que, en términos de medición, es real. La causa de esta asimetría se encuentra relacionada con una alteración postural. Es posible identificar distintas afecciones médicas que dan lugar a esta condición, tales como la escoliosis que conlleva una curvatura lateral de la columna vertebral, la contorsión lumbar que genera una torsión en la región lumbar de la columna, la descompensación de los aductores y abductores que desalinea la pelvis, la rotación de la pelvis debido a la anteroversión y retroversión de los huesos ilíacos, y, por último, la desviación hacia adentro o hacia afuera del retropié, que se conoce como valgismo o varismo (Institut Catalá del Peu, 2023).

Los orígenes de las asimetrías pueden ser:

- Idiopáticos: no hay causa conocida
- Congénitos: ausencia de peroné, pie zambo, luxación congénita de cadera
- Infecciosos: osteomielitis
- Traumáticos: acortamiento del cuerpo del hueso por fractura

- Otras causas como lo pueden ser: enfermedad de Perthes, radioterapia, cirugía.

**Figura 6** Ciclograma de marcha, en personas con amputación de miembro inferior y sin amputación.



*Nota:* Adaptado de Molina, J (2021)

### Protocolos para el análisis de marcha

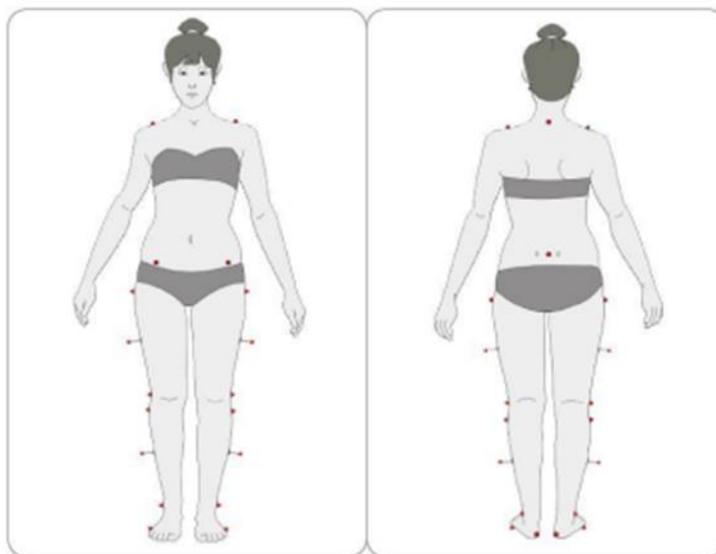
Para analizar el desarrollo de la marcha normal, se tienen en cuenta los siguientes protocolos:

- Protocolo de Davis: Dicho protocolo se caracteriza por integrar alrededor de 22 marcadores que facilitan el estudio de la actividad muscular (concretamente de la parte eléctrica) respecto a los músculos involucrados en el estudio. Algunos de los requerimientos de este protocolo tienen que ver con el peso y la altura del sujeto, además de algunas medidas corporales como lo son: el ancho de la pelvis, el largo de las piernas y los diámetros de los tobillos y las rodillas.

b) Protocolo de Hayes: En el análisis de la marcha, el protocolo de Hayes es considerado uno de los más completos ya que emplea alrededor de 15 marcadores que facilitan su parametrización; los datos que se necesitan para la aplicación de este protocolo son: peso y estatura el sujeto junto con las medidas de longitud de los miembros inferiores. Con la información recolectada previamente lo que se hace es calcular los centros articulares y los ángulos de movimientos (Bravo. 2016).

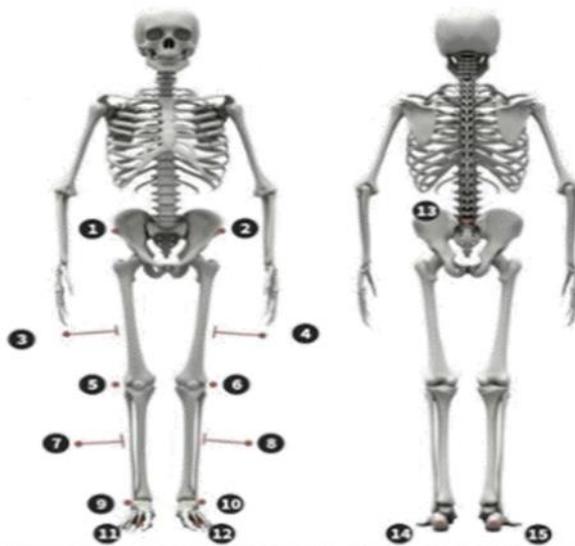
Las siguientes figuras muestran los puntos marcadores más importantes para el funcionamiento de cada una:

**Figura 7** Posicionamiento de marcadores en el protocolo de Davis



*Nota:* Tomado de Molina, J (2021)

**Figura 8** Posición de marcadores en protocolo Hayes



*Nota:* Tomado de STS Systems (2014)

### **Unidad de medición inercial**

Se entiende por unidad de medición inercial al dispositivo capaz de estimar y reportar estados dinámicos determinados como las velocidades angulares, así como las aceleraciones, la velocidad, la orientación y las fuerzas gravitacionales de un aparato. Estas unidades de medición inercial son generalmente empleadas para el manejo de aeronaves, incluyendo aquellas no tripuladas, además también son empleados en el manejo de naves espaciales que incluyen transbordadores, satélites y aterrizadores (Roldán, 2017).

Para el caso de la elaboración de prótesis, las unidades de medición inercial son pequeños dispositivos capaces de medir la aceleración (lineal) y la velocidad (para este caso, angular) siendo muy útiles en el campo de la salud para la medición, la cuantificación y la valoración (objetiva) del movimiento corporal humano (Roldán, 2017).

## **Métodos de postratamiento de datos cinemáticos**

Los métodos de post procesamiento de datos cinemáticos son una herramienta importante para el análisis de movimientos humanos en distintas aplicaciones, incluyendo la evaluación de la biomecánica de la marcha, el deporte y la rehabilitación.

Estos métodos se utilizan para analizar los datos cinemáticos recopilados a partir de diferentes sistemas de captura de movimiento, como los sistemas ópticos, inerciales y magnéticos. El post procesamiento de los datos implica una serie de procesos matemáticos y estadísticos que se aplican a los datos brutos para extraer información relevante sobre el movimiento humano. Entre los métodos de post procesamiento de datos cinemáticos más comunes se encuentran los siguientes (Caro Zapata, 2022):

**Filtrado de señales:** Este método se utiliza para eliminar el ruido de los datos capturados y reducir las fluctuaciones no deseadas. Los filtros más comunes son los filtros de paso bajo, que eliminan las frecuencias más altas que están por encima de un umbral establecido. Sin embargo, es importante tener en cuenta que el filtrado también puede eliminar información importante del movimiento, por lo que es necesario aplicarlo con cuidado (Caro Zapata, 2022).

**Interpolación:** Se utiliza para suavizar los datos y hacer que parezcan más continuos. Por ejemplo, si la tasa de muestreo de los datos es baja, se pueden utilizar técnicas de interpolación para rellenar los vacíos y suavizar las transiciones entre los datos. Sin embargo, es importante tener en cuenta que la interpolación también puede agregar error al movimiento si se utiliza de manera inapropiada (Caro Zapata, 2022).

**Segmentación:** Se utiliza para dividir el movimiento en fases más pequeñas y manejables. Esto puede ayudar a identificar patrones de movimiento específicos, así como

a analizar diferentes fases del movimiento con mayor detalle. Por ejemplo, en el análisis de la marcha, se pueden segmentar las fases de apoyo y balanceo para identificar patrones específicos en cada fase (Caro Zapata, 2022).

**Análisis de componentes principales:** Es una técnica avanzada de análisis de datos que se utiliza para identificar patrones subyacentes en los datos. En el contexto del análisis cinemático, el análisis de componentes principales puede ayudar a identificar patrones de movimiento específicos y reducir la complejidad de los datos. Por ejemplo, se puede utilizar para identificar los patrones de movimiento característicos de diferentes deportes o actividades físicas (Caro Zapata, 2022).

### **Lenguaje de programación Python**

Python es un lenguaje de programación creado en la década de los 90 por Guido van Rossum en Países Bajos como sucesor de un idioma denominado ABC (Challenger, Díaz y Becerra, 2014).

Es un lenguaje de programación que se caracteriza por tener estructuras de datos muy eficientes de un elevado nivel y un enfoque simple, pero súper efectivo para la programación orientada hacia los objetos.

La sintaxis de Python permite el desarrollo de aplicaciones tales como Instagram, Netflix y Spotify gracias a que este lenguaje soporta parcialmente la orientación a objetos, programación imperativa y funcional.

Algunos de los beneficios de Python son:

- Los desarrolladores pueden leer y comprender fácilmente los programas de Python debido a su sintaxis básica similar a la del inglés.

- Python permite que los desarrolladores sean más productivos, ya que pueden escribir un programa de Python con menos líneas de código en comparación con muchos otros lenguajes.
- Python cuenta con una gran biblioteca estándar que contiene códigos reutilizables para casi cualquier tarea. De esta manera, los desarrolladores no tienen que escribir el código desde cero.
- Los desarrolladores pueden utilizar Python fácilmente con otros lenguajes de programación conocidos, como Java, C y C++.
- Python se puede trasladar a través de diferentes sistemas operativos de computadora, como Windows, macOS, Linux y Unix (Amazon Web Services, 2023).

Ahora, respecto a la construcción de la interfaz para interfaz usando Python que permita la cuantificación e identificación de asimetrías temporales de la marcha en pacientes con prótesis de miembro inferior cabe aclarar que se escogió este lenguaje de programación teniendo en cuenta sus características en cuanto a las ventajas que ofrece respecto al soporte de operaciones funcionales.

### **Aplicaciones de Python en el campo Biomédico**

Algunas de las aplicaciones de Python en este campo incluyen:

**Análisis de datos:** Python se utiliza ampliamente para el análisis de datos biomédicos. Con las bibliotecas de análisis de datos como NumPy, Pandas y SciPy, los investigadores pueden procesar grandes conjuntos de datos y obtener información relevante para la investigación biomédica (Herrera-Hernández, et al., 2018).

**Machine learning:** Python es un lenguaje popular para la implementación de algoritmos de aprendizaje automático en aplicaciones biomédicas. Los algoritmos de

aprendizaje automático se utilizan para analizar grandes conjuntos de datos y encontrar patrones que puedan ayudar a identificar enfermedades o mejorar la eficacia de los tratamientos (Herrera-Hernández, et al., 2018).

Simulación de sistemas biológicos: Python se utiliza para simular sistemas biológicos complejos, como modelos de simulación de neuronas y sistemas de transporte de drogas. Estas simulaciones permiten a los investigadores comprender mejor los procesos biológicos y mejorar la eficacia de los tratamientos médicos (Higuera-González, et al., 2018).

Visualización de datos: Python también se utiliza para visualizar datos biomédicos. Con bibliotecas como Matplotlib y Seaborn, los investigadores pueden crear gráficos y visualizaciones que les ayuden a entender mejor los datos y comunicar sus hallazgos a otros investigadores y médicos (Higuera-González, et al., 2018).

En resumen, Python ofrece una amplia gama de herramientas para el análisis y la visualización de datos biomédicos, lo que lo convierte en una herramienta importante en el campo de la investigación y la atención médica.

## 2.2 Marco legal

En nuestro país se cuenta con disposiciones legales sobre el manejo de personas con discapacidad y amputaciones (dificultades motoras). Así, resulta imprescindible dar inicio con la Constitución Política de Colombia, ya que en su artículo 47 se establece que el gobierno debe desarrollar una estrategia cuyo propósito sea la prevención, recuperación e inclusión social de las personas que presenten discapacidades físicas, sensoriales o mentales. En consecuencia, se proveerá la asistencia especializada que requieran para satisfacer sus requerimientos.

Seguido a esto, tenemos otras disposiciones legales importantes tales como la ley 1145 de 2007 que contempla:

“Promover la formulación e implementación de políticas públicas sobre personas con discapacidad de manera coordinada entre las entidades públicas del orden nacional, regional y local, las organizaciones de personas con discapacidad y personas necesitadas, y la sociedad civil para promover y proteger a las personas con discapacidad en el marco de los derechos humanos.”

Otro referente jurídico para el desarrollo de este proyecto de investigación está relacionado con la ley 1448 de 2011 donde encontramos:

“Por la cual se dictan medidas de atención, asistencia y reparación integral a las víctimas del conflicto armado interno y se dictan otras disposiciones.” Con respecto a las personas con discapacidad, se menciona en el artículo 13 sobre trato diferenciado, el artículo 51 que es sobre medidas relativas a la educación, el artículo 123 sobre medidas de restitución por cuestiones de vivienda y artículo 136 sobre programas de rehabilitación.”

El propósito de la normativa "Por la cual se establecen disposiciones para la telesalud y parámetros para la práctica de la telemedicina en el país" (resolución 2654 de 2019) es unificar y actualizar las normas existentes en relación con la telemedicina y telesalud, con la finalidad de fomentar una mayor accesibilidad y calidad de los servicios de salud para la población que habita en las áreas más alejadas del país.

Adicionalmente, la resolución 8430 de 1993 de Colombia, la cual establece los requisitos técnicos y científicos que deben cumplir los laboratorios clínicos para garantizar la calidad de los servicios que ofrecen. La resolución establece normas para la infraestructura, los equipos, el personal, los procedimientos y los registros de los laboratorios clínicos. Se enfoca en la prevención de errores en el procesamiento de muestras y en la obtención de resultados confiables y precisos. Además, la resolución establece la obligación de los laboratorios clínicos de participar en programas de evaluación externa de la calidad para garantizar la continuidad de los servicios y la mejora continua de los mismos.

Y finalmente la normativa "por la cual se establecen disposiciones sobre la certificación de discapacidad y la creación del Registro de Localización y Caracterización de Personas con Discapacidad" (resolución 113 de 2020) tiene como finalidad la puesta en marcha de un certificado de discapacidad y un registro para ubicar a las personas que presenten discapacidad.

## Capítulo 3 – Metodología

El proyecto que se desarrolló consta de cuatro fases principales, diseñadas para dar cumplimiento a los objetivos establecidos en el presente documento. A continuación, se describe cada uno de los puntos:

En la primera fase, se realizó el estudio bibliográfico que permite identificar las variables cinemáticas relevantes en la determinación de las asimetrías temporales en pacientes con prótesis de miembro inferior.

La segunda fase implica la selección y desarrollo de las variables claves y la implementación de una interfaz de programación utilizando Python para cuantificar e identificar las asimetrías temporales de la marcha.

En la tercera fase, se llevó a cabo pruebas de funcionamiento, donde se desarrolló el Manual de Uso de la interfaz.

Por último, en la Cuarta fase, se realizó una validación inicial de la usabilidad de la interfaz con la ayuda de tres posibles usuarios no especializados o sin conocimientos técnicos en la materia.

Para lograr una correcta verificación del cumplimiento de hitos, así como de exhaustividad al momento de la ejecución del producto final, se tomó la decisión, que las cuatro (4) fases mencionadas fueron fundamentales para el diseño e implementación de ingeniería; Por tal razón, cada una tiene una serie de actividades asignadas para ser ejecutadas, con el fin de cumplir a cabalidad con el objetivo general del proyecto.

### **3.1 Etapa 1: Identificación de las variables y parámetros**

En la fase de Identificación de las variables, se realiza una revisión bibliográfica exhaustiva para identificar las variables cinemáticas que son relevantes para la cuantificación e identificación de asimetrías temporales de la marcha en pacientes con prótesis de miembro inferior.

Es importante tener en cuenta que las variables pueden cambiar, dependiendo del tipo de prótesis, la ubicación de la amputación, el nivel de amputación y el nivel de actividad del paciente. Algunas de las variables que se pueden considerar incluyen la duración del ciclo de la marcha, la longitud del paso, la velocidad de la marcha, la amplitud de la oscilación del cuerpo, la velocidad angular de la articulación de la cadera, la rodilla y el tobillo, el tiempo de contacto del pie, la altura de la elevación del pie y la coordinación intersegmental.

Una vez se identifican las variables relevantes, se determinó cuáles son las variables claves que se utilizan para cuantificar las asimetrías temporales de la marcha.

Para llevar a cabo esta etapa, se realizarán las siguientes actividades:

1.1. Revisión bibliográfica exhaustiva: Se llevó a cabo una revisión de la literatura especializada para identificar las variables y parámetros más utilizados en el análisis cinemático de la marcha en pacientes con prótesis de miembro inferior.

1.2. Selección de las variables y parámetros relevantes: Después de la revisión bibliográfica, se seleccionó las variables y parámetros relevantes que permitirán la cuantificación e identificación de asimetrías temporales de la marcha en pacientes con

prótesis de miembro inferior. Es importante tener en cuenta que estas variables pueden variar dependiendo del tipo de prótesis, la ubicación de la amputación, el nivel de amputación y el nivel de actividad del paciente.

El resultado de esta etapa es la identificación de las variables y parámetros claves que se utilizó en el análisis cinemático de la marcha en pacientes con prótesis de miembro inferior, que permitió la cuantificación e identificación de las asimetrías temporales de la marcha. Esta etapa contribuyo al alcance del objetivo general, que es construir una interfaz digital que permitió el procesamiento de información cinemática para la cuantificación e identificación de asimetrías temporales de la marcha en pacientes con prótesis de miembro inferior.

1.3. Determinación de las variables claves: Una vez seleccionadas las variables relevantes, se determinó cuáles son las variables claves que se usó para cuantificar las asimetrías temporales de la marcha en pacientes con prótesis de miembro inferior. Estas variables claves deben ser seleccionadas cuidadosamente para garantizar una evaluación precisa y eficiente de las asimetrías temporales de la marcha.

## **3.2 Etapa 2: Diseño de la interfaz gráfica en Python**

En la fase de diseño de la interfaz de programación en Python para la cuantificación e identificación de asimetrías temporales de la marcha en pacientes con prótesis de miembro inferior, se consideró los siguientes aspectos:

2.1. Definición de la interfaz de usuario: Se diseñó una interfaz que sea fácil de usar e intuitiva, teniendo en cuenta las necesidades de los usuarios. La interfaz incluyo las variables cinemáticas identificadas en el paso anterior y debe permitir al usuario ingresar y procesar los datos.

2.2. Selección de las herramientas y bibliotecas de Python: Se seleccionó las herramientas y bibliotecas de Python que se utilizaron para procesar y analizar los datos, teniendo en cuenta los requisitos de la interfaz de usuario y los objetivos específicos del proyecto.

2.3. Diseño del flujo de trabajo: Se diseñó el flujo de trabajo de la interfaz, definiendo cómo se ingresó los datos, cómo se procesaron y cómo se presentan los resultados.

El objetivo general de esta etapa fue el de elaborar una interfaz digital que permitió el procesamiento de información cinemática para la cuantificación e identificación de asimetrías temporales de la marcha en pacientes con prótesis de miembro inferior. El diseño de la interfaz de programación en Python permitió alcanzar el objetivo general, ya que los datos cinemáticos se procesaron y analizaron de manera eficiente y precisa.

Se obtuvo como resultado una interfaz de programación en Python bien diseñada y fácil de usar, que permite el procesamiento de los datos cinemáticos y la cuantificación e identificación de asimetrías temporales de la marcha en pacientes con prótesis de miembro inferior.

### **3.3 Etapa 3: Pruebas de funcionamiento de la interfaz**

En esta etapa, se hicieron pruebas de funcionamiento de la interfaz para validar su usabilidad y obtener retroalimentación de los usuarios. Para dar cumplimiento, se adelantó lo siguiente:

3.1. Seleccionar a los usuarios no especializados o con conocimientos técnicos limitados para que prueben la interfaz y proporcionen retroalimentación sobre su facilidad de uso y claridad.

3.2. Desarrollo de un procedimiento de prueba que incluya una serie de tareas que los usuarios deben completar utilizando la interfaz. Por ejemplo, se puede pedir a los usuarios que carguen un conjunto de datos de marcha, configuren las opciones de análisis y generen un informe de resultados.

3.3. Recopilación de los datos durante las pruebas. Incluye observaciones de los usuarios mientras utilizan la interfaz, preguntas que hacen sobre la funcionalidad de la interfaz y las respuestas que proporcionan a los cuestionarios o encuestas de satisfacción del usuario.

3.4. Se analizaron los datos recopilados para identificar problemas o áreas de mejora en la interfaz. Se usaron las herramientas de análisis de datos para identificar patrones y tendencias en los datos.

3.5. Se realizó los cambios en la interfaz basados en los resultados de las pruebas y la retroalimentación de los usuarios. Fue importante asegurarse que los cambios realizados no afecten negativamente a la funcionalidad de la interfaz o la precisión de los resultados.

Esta etapa contribuye al alcance del objetivo general, ya que la validación de la usabilidad de la interfaz es fundamental para asegurar que los resultados obtenidos a partir de la información procesada sean precisos y confiables.

Los resultados de esta etapa es la identificación de las áreas de mejora en la interfaz y realizar cambios para mejorar la usabilidad y la experiencia del usuario al utilizarla. Además, se obtiene la retroalimentación de los usuarios para continuar con la mejora de la interfaz y cumplir con el objetivo general y el objetivo específico del proyecto.

### **3.4 Etapa 4: Validación de la usabilidad de la interfaz**

La validación de la usabilidad de la interfaz fue esencial para asegurar que la interfaz es fácil manejo y satisfaga las necesidades de los usuarios. Cuando se mejora la usabilidad de la interfaz, se garantiza la calidad de la información obtenida y, por lo tanto, es eficacia de la identificación y cuantificación de asimetrías temporales de la marcha en pacientes con prótesis de miembro inferior.

En la fase de validación de la usabilidad de la interfaz, se seleccionaron usuarios que utilizaron la interfaz y proporcionaron comentarios útiles sobre su funcionalidad y facilidad de uso. Para ello, se siguieron los siguientes pasos:

4.1. Identificar el perfil de los usuarios: en este caso, se buscaron cinco (5) usuarios que no son especializados o no cuenten conocimientos técnicos en el área de la cuantificación e identificación de asimetrías temporales de la marcha en pacientes con prótesis de miembro inferior.

4.2. Seleccionar los usuarios: se seleccionaron los usuarios a través de diferentes medios, fueron ser voluntarios o conocidos que encajen en el perfil buscado. Fue importante asegurar que los usuarios seleccionados cumplieran como población a la que está dirigida la interfaz.

4.3. Realizar pruebas de usabilidad: se llevó a cabo pruebas de usabilidad con los usuarios seleccionados donde se evaluó la experiencia al utilizar la interfaz. Las pruebas se realizaron de diferentes formar, tales como la observación directa de los usuarios mientras utilizaron la interfaz, o a través de entrevistas estructuradas antes y después de la utilización de la interfaz.

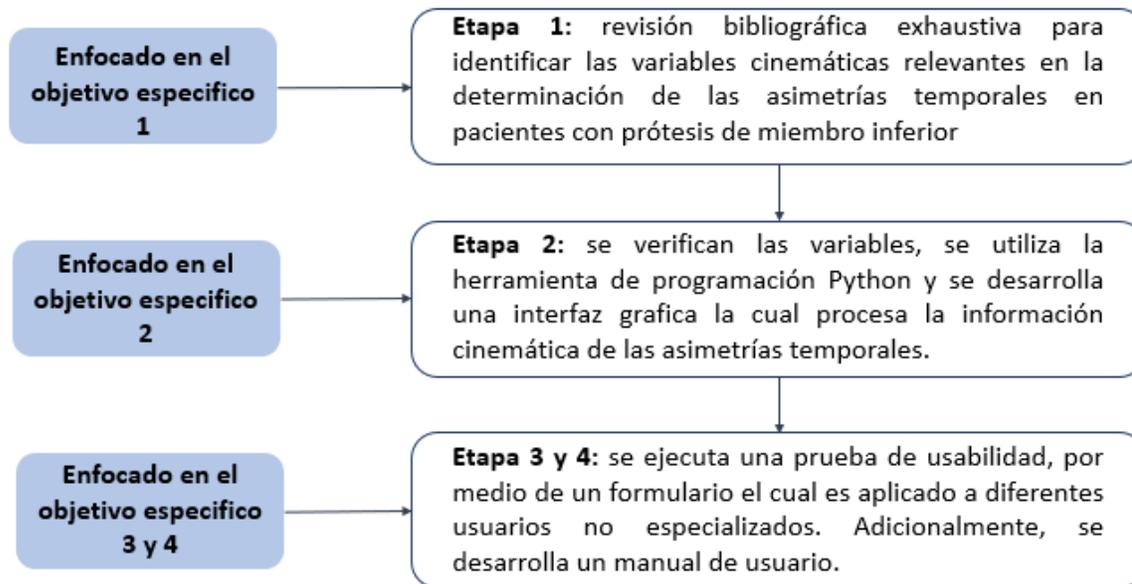
4.4. Registrar los comentarios y observaciones de los usuarios: Se recopiló los comentarios de los usuarios, los cuales permitieron utilizarlos para adelantar mejoras a la interfaz. Se usaron encuestas y/o cuestionarios para recopilar comentarios específicos sobre aspectos concretos de la interfaz.

4.5. Analizar los resultados y realizar mejoras: los resultados de las pruebas de usabilidad se analizaron cuidadosamente donde se pudo identificar las áreas en las que se necesita mejorar la interfaz. Con esta información, se realizaron los cambios necesarios para mejorar la funcionalidad y la facilidad de uso de la interfaz.

4.6. Repetir las pruebas de usabilidad: una vez realizados los cambios, se repitieron las pruebas de usabilidad con los mismos usuarios y con usuarios diferentes donde se constató que las mejoras en la interfaz surtieron efecto.

Los resultados esperados de la etapa de validación de la usabilidad de la interfaz incluyeron la identificación de posibles dificultades en el uso de la interfaz, y la recopilación de retroalimentación útil de los usuarios seleccionados. A partir de esta información, se mejoró la funcionalidad y facilidad de uso de la interfaz, que permitió dar cumplimiento al objetivo general del proyecto que es el de desarrollar una herramienta útil y accesible para la cuantificación e identificación de asimetrías temporales de la marcha en pacientes con prótesis de miembro inferior. Además, la repetición de las pruebas de usabilidad después de haber realizado mejoras en la interfaz permitió evaluar la efectividad de dichas mejoras garantizando que se lograran mejoras significativas en la experiencia del usuario.

A continuación, se sintetiza la información expuesta anteriormente en la metodología en la siguiente figura:

**Figura 9** Diagrama fases de la metodología

*Nota:* Elaboración propia.

## Capítulo 4 – Resultados

En cumplimiento a los objetivos propuestos, se desarrolla un algoritmo que permite procesar información cinemática para la cuantificación de asimetrías temporales en pacientes con prótesis en extremidades inferiores, para esto se aplican diferentes filtros como lo es Kalman, media móvil y se ejecutan análisis cinemáticos. Con ello se obtiene el cálculo de la asimetría deseada.

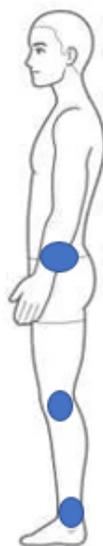
### 4.1 Etapa 1: Identificación de las variables y parámetros

La identificación de las variables se realiza a través de la base de datos obtenida en el trabajo de grado “Implementación de un Sistema Basado en Sensores Inerciales para la Caracterización Biomecánica de la Marcha Normal y de Pacientes con Amputación del Miembro Inferior.” realizado en la universidad. En el cual se realizaron los siguientes puntos más relevantes:

1. Búsqueda de personas que se encontraban con amputación de miembro inferior, y contaban con una prótesis por un periodo mínimo de 6 meses.
2. Se toma como base el protocolo Helen Hayes, se ubican 3 marcadores los cuales están ubicados en tres articulaciones: cadera, rodilla y tobillo (Figura 10). Adicionalmente, la ubicación de los sensores inerciales también fue tomada de la investigación acerca de la monitorización del ángulo de la articulación de la rodilla basado unidades de medidas inercial para la marcha realizada previamente en la Universidad Antonio Nariño, los sensores inerciales que fueron ubicados a partir de medidas antropométricas, dando como resultado el muslo y se ubica en la mitad de este, se ubica en la pierna a nivel de tibia y el sensor de la cadera al nivel de la columna en la vertebra lumbar L5. Figura (11)

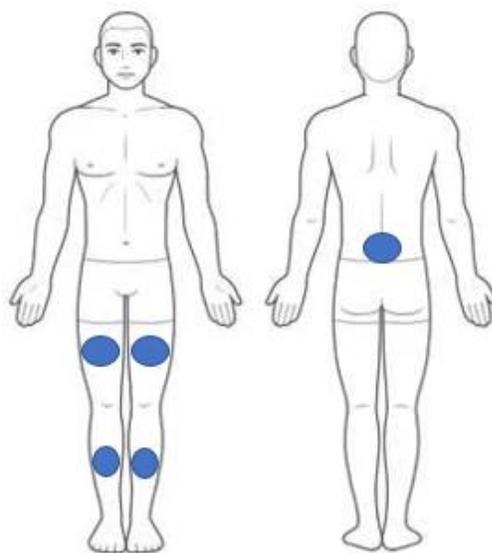
3. Con el fin de obtener el cálculo de los parámetros de marcha se ubicaron 5 sensores en las extremidades inferiores.

**Figura 10** Set de marcadores implementado



*Nota:* Elaboración propia.

**Figura 11** Ubicación de los sensores inerciales



*Nota:* Elaboración propia.

Dicho lo anterior, se idéntica que las variables cinemáticas para nuestro caso son los ángulos de Euler que se en los ejes X, Y y Z. Adicionalmente, el eje X es tomado como medida, ya que no permite inconvenientes de bloqueo de cardán o gimball lock. Por otra parte, se tiene la variable de amplitud articular. Tal cual como se muestra en el trabajo de grado “Implementación de un Sistema Basado en Sensores Inerciales para la Caracterización Biomecánica de la Marcha Normal y de Pacientes con Amputación del Miembro Inferior.” Sustentado en la universidad Antonio Nariño.

*Nota:* Elaboración propia

**Tabla 1** Variables seleccionadas

Variable	
<b>Angulo X</b>	Angulo de Euler en x
<b>Angulo Y</b>	Angulo de Euler en y
<b>Angulo Z</b>	Angulo de Euler en z
<b>Amplitud Articular</b>	Grado de flexión

*Nota:* Elaboración propia

**Figura 12** Base de datos de sensores

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	Persona	Toma	time	angX1	angY1	angZ1	angX2	angY2	angZ2	angX3	angY3	angZ3	angX4	angY4	angZ4	angX5	angY5	angZ5
2	angulos_Cristi	R1	25.52	92.46	-3.19	-26.85	-74.72	-10.68	24.37	108.39	-4.21	13.54	93.61	-2.08	-36.95	90.5	5.0	-21.49
3	angulos_Cristi	R1	25.56	92.47	-3.19	-26.88	-74.72	-10.68	24.42	108.37	-4.21	13.6	93.64	-2.08	-36.98	90.5	5.0	-21.52
4	angulos_Cristi	R1	0.0	92.48	-3.19	-26.91	-74.74	-10.68	24.48	108.37	-4.21	13.61	93.67	-2.08	-37.03	90.51	5.0	-21.55
5	angulos_Cristi	R1	0.04	92.48	-3.18	-26.95	-74.75	-10.68	24.54	108.38	-4.21	13.66	93.68	-2.07	-37.06	90.51	5.0	-21.58
6	angulos_Cristi	R1	0.08	92.49	-3.18	-26.98	-74.75	-10.68	24.6	108.4	-4.21	13.69	93.68	-2.06	-37.09	90.52	5.0	-21.61
7	angulos_Cristi	R1	0.12	92.5	-3.17	-27.01	-74.76	-10.68	24.66	108.41	-4.2	13.73	93.68	-2.06	-37.13	90.53	5.0	-21.64
8	angulos_Cristi	R1	0.16	92.52	-3.17	-27.04	-74.75	-10.67	24.72	108.41	-4.2	13.77	93.68	-2.06	-37.17	90.54	5.0	-21.66
9	angulos_Cristi	R1	0.2	92.53	-3.16	-27.07	-74.74	-10.68	24.78	108.41	-4.2	13.8	93.68	-2.05	-37.2	90.55	5.0	-21.69
10	angulos_Cristi	R1	0.24	92.55	-3.16	-27.1	-74.73	-10.69	24.84	108.41	-4.19	13.83	93.67	-2.05	-37.24	90.57	5.01	-21.72
11	angulos_Cristi	R1	0.28	92.56	-3.16	-27.12	-74.71	-10.69	24.89	108.42	-4.2	13.86	93.67	-2.04	-37.28	90.59	5.01	-21.74
12	angulos_Cristi	R1	0.32	92.56	-3.16	-27.14	-74.71	-10.68	24.95	108.43	-4.2	13.89	93.66	-2.04	-37.32	90.6	5.01	-21.77
13	angulos_Cristi	R1	0.36	92.57	-3.16	-27.17	-74.7	-10.69	25.0	108.44	-4.2	13.92	93.66	-2.03	-37.36	90.61	5.01	-21.79

*Nota:* Elaboración propia

La base de datos se encuentra estructurada de la siguiente manera: en la Columna A se encuentran los datos de las personas a las cuales se le tomaron las pruebas. Posteriormente, en la columna B nombrada como toma se tienen 4 datos de R1, R2, R3 y

R4 y calibración, los cuales hacen referencia a los sensores inerciales ubicados en el cuerpo de la persona.

La columna C, hace referencia al tiempo en el cual se tomaron las muestras de los datos. En las siguientes columnas se encuentran nombradas de la siguiente manera: angX1, angY1, angZ1, sucesivamente hasta angX5, angY5, angZ5, estos indican X1, Y1, Z1 y X2, Y2, Z2 son las tomas de los sensores sobre la pierna derecha que no cuenta con una prótesis, teniendo en cuenta eso X3, Y3, Z3 y X4, Y4, Z4 son los ángulos de los sensores sobre la pierna con prótesis. Los datos obtenidos en el ángulo X5, Y5, Z5 hace referencia al centro de gravedad el cual se ubicó en la cadera a nivel de la vertebra lumbar L5.

En la figura 13, tras el procesamiento de los datos obtenidos por los sensores inerciales, se logra graficar la amplitud articular promedio de la rodilla izquierda que es la que cuenta con afectación por amputación.

**Figura 13** Amplitud articular promedio de la rodilla izquierda



## 4.2 Etapa 2: Diseño de la interfaz gráfica en Python

Para el desarrollo de la interfaz se usó como lenguaje de programación base Python 3.8, usando el IDE de Microsoft Visual Studio Code (Anexo A). Para realizar todo el análisis

de la base de datos de los pacientes, se usaron las librerías *tkinter*, *matplotlib.pyplot*, *numpy* y *pandas*. Cada una de estas librerías tiene una función específica, como sigue:

**tkinter:** es una librería de Python que se utiliza para crear interfaces gráficas de usuario (GUI). Con tkinter, los programadores pueden crear ventanas, botones, menús, cuadros de diálogo y otros elementos de la interfaz gráfica de usuario. Tkinter es la biblioteca gráfica estándar de Python y viene preinstalada en la mayoría de las distribuciones de Python.

**matplotlib.pyplot:** Matplotlib es una librería de Python que se utiliza para crear gráficos y visualizaciones de datos. El módulo pyplot de matplotlib proporciona una interfaz para crear gráficos de alta calidad con una sintaxis simple y fácil de entender. Con pyplot, los programadores pueden crear gráficos de líneas, barras, dispersión, histogramas y otros tipos de visualizaciones.

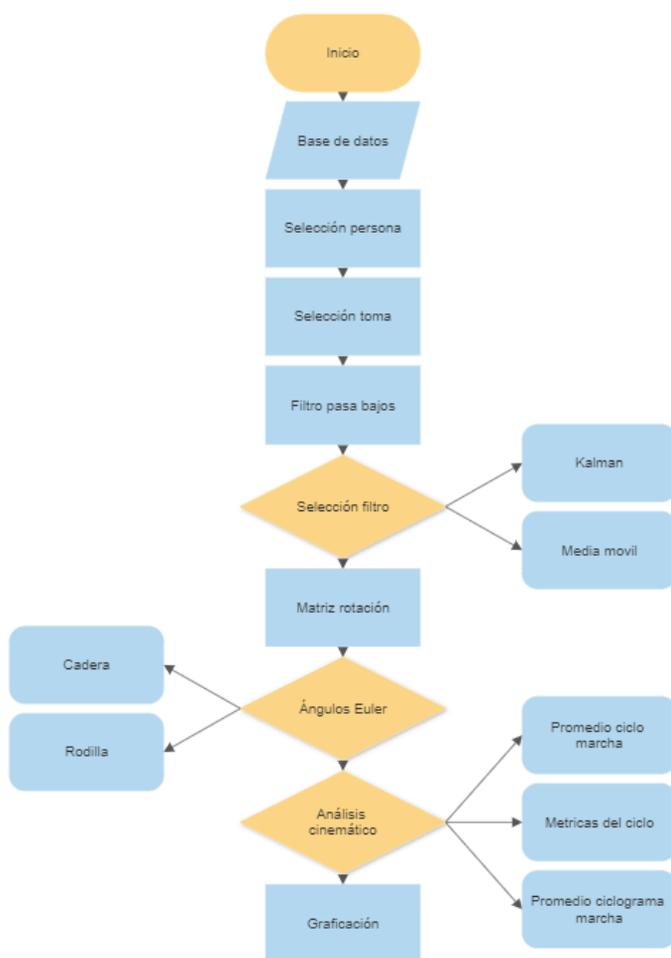
**numpy:** es una librería de Python que se utiliza para realizar operaciones numéricas y matemáticas en matrices y vectores. Con NumPy, los programadores pueden realizar operaciones complejas de manera rápida y eficiente en grandes conjuntos de datos. NumPy se utiliza a menudo en aplicaciones científicas y de análisis de datos.

**pandas:** es una librería de Python que se utiliza para manipular y analizar datos. Con pandas, los programadores pueden leer y escribir datos en una variedad de formatos, incluidos CSV, Excel y bases de datos SQL. Pandas proporciona una estructura de datos de alto rendimiento llamada DataFrame, que se utiliza para organizar y analizar datos en una tabla. Pandas también tiene funciones para realizar operaciones de agregación, filtrado y limpieza de datos.

En resumen, dentro del proyecto, tkinter se utilizó para crear interfaces gráficas de usuario, matplotlib.pyplot para crear gráficos y visualizaciones, numpy para realizar operaciones numéricas y matemáticas en matrices y vectores, y pandas para manipular y analizar los datos de los pacientes.

Con esto, se realizó el diagrama de flujo (Figura 14) el cual explica el funcionamiento de la interfaz desarrollada con Python.

**Figura 14** Diagrama de flujo de interfaz gráfica



*Nota:* Elaboración propia

Se tuvo en cuenta la base de datos de la persona de la cual realizaron las tomas con los sensores inerciales. Posterior a seleccionar las librerías para poder realizar el tratamiento de los datos se realizan los siguientes pasos:

Se toma la base de datos que se vaya a trabajar, la cual usualmente está determinada por un archivo con extensión .csv, el cual es delimitado por “;”, debido a que los datos suministrados por el trabajo de grado anterior realizado en la UAN se encontraban en un archivo .txt

**Figura 15** Delimitación de la base de datos por medio de “;”

```
df = pd.read_csv('Registros_de_sensores.csv',delimiter=";")
```

*Nota:* Elaboración propia

Se nombraron las variables durante el código, estas variables corresponden a los ángulos en x, y y z anteriormente descritos en la fase de “Identificación de variables y parámetros”.

**Figura 16** Ángulos tomados de la base de datos

```
angulos=['angX1', 'angY1', 'angZ1', 'angX2', 'angY2',
         'angZ2', 'angX3', 'angY3', 'angZ3', 'angX4',
         'angY4', 'angZ4']
```

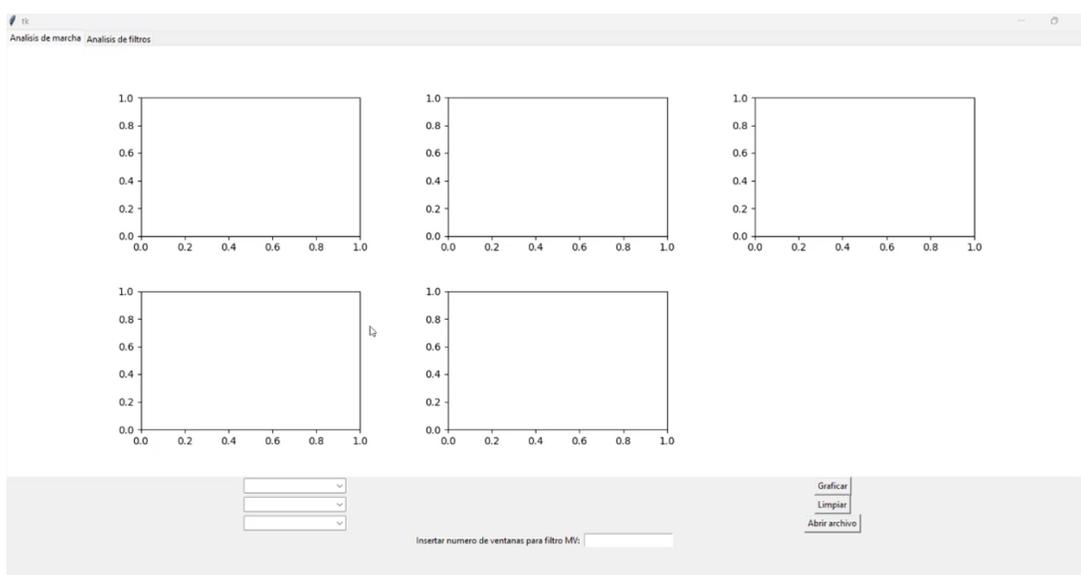
*Nota:* Elaboración propia

Se crearon diferentes clases, la cual estuvieron compuestas por diferentes subclases en la cual está el desarrollo de la interfaz.

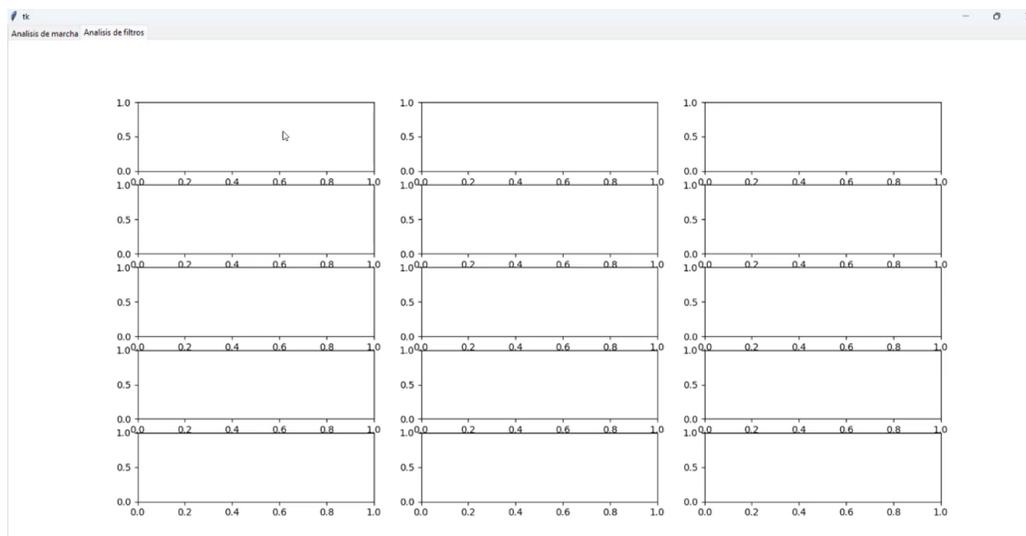
La primera clase que se definió son las clases de inicio, creación de las dos pestañas de las cual está compuesta la interfaz, la subclase consecutiva que se crea es la del botón archivo, el cual permite seleccionar la base de datos, dicha base de datos fue tomada de datos obtenidos por sensores inerciales en un trabajo realizado en la UAN.

Paso siguiente se crean las diferentes graficas de las cuales está compuesta la interfaz, se encuentran organizadas de la siguiente manera: dos graficas para la pierna izquierda y dos graficas para la pierna derecha del paciente, la gráfica del ciclograma. En la segunda página, se encuentran 15 gráficas de los filtros, se tienen 15 variables por los 5 sensores implementados en las tres dimensiones. Por último, se define el tamaño de cada una de estas y su ubicación en la ventana, como de cada uno de los botones que se encuentran.

**Figura 17** Página 1 de la interfaz gráfica – “Análisis de marcha”



**Figura 18** Página 2 de la interfaz gráfica – “Análisis de filtros”



*Nota:* Elaboración propia

Paso siguiente se realiza la selección y uso del filtro pasa bajos, dado que este permite procesar la señal y obtener resultados más precisos eliminando ruido, se usa este filtro cuando la frecuencia es menor a la frecuencia de corte.

Para este proyecto se definió las siguientes frecuencias en Hertz (Hz):

Frecuencia de corte: 10Hz y 5Hz

Frecuencia de muestreo ( $F_s$ ): 25Hz

Es importante mencionar que dicha adquisición de datos de los ángulos con los sensores inerciales fue realizada a la misma frecuencia de muestreo.

Posteriormente, se realizó el cálculo de las frecuencias de corte normaliza, en la cual se utilizó el factor de 0.7 multiplicado por la frecuencia de muestreo y luego de este se aplicó a cada una de las frecuencias de corte de la cadera y rodilla. Por último, se ejecutó el cálculo del filtro pasa bajo ButterWorth de orden dos, este permite una frecuencia máxima plana, es decir no se tiene ondulaciones en la banda de paso.

Posteriormente se realiza la definición de la clase del Filtro de Kalman, este permite estimar el estado de un sistema a partir de unos datos o mediciones obtenidos a través del

tiempo. Se divide en dos: fase de predicción y fase de actualización, son altamente utilizados en el campo de control y movimiento robótico.

Los datos de los sensores se pasan a radianes para el procesamiento, se define las matrices del modelo, para posteriormente inicializar el estado y la covarianza, se inicializa el vector de los ángulos de filtrado, que son los ángulos obtenidos en nuestra muestra. Finalmente, se ejecuta el filtro de Kalman en donde se toma la medición de los datos, se ejecuta el cálculo de la predicción del estado y la covarianza, luego se realiza la corrección o actualización del estado y covarianza y se limita el rango de los ángulos. El resultado de este se pasa a grados.

Filtro de media móvil es una solución para filtrar ruido de alta frecuencia, además las técnicas de promediado digital son actualmente empleadas como procedimientos comunes para aumentar la relación señal/ruido de las medidas obtenidas mediante sistemas de adquisición de datos.

Para este proyecto se implementó un filtro de media móvil de orden 10, el tamaño de la ventana estará definido por el usuario el cual será desde 11 hasta 61. Se define a través de matrices la posición de cada uno de los cinco sensores, se precisa el tiempo de la muestra el cual fue de 0.04 segundos. Para luego obtener el ángulo relativo de la rodilla (3) y ángulo relativo de la cadera (4), que se definen con la siguiente formula:

$$\theta_{Pierna} = (\theta_{Pierna} - \theta_{Muslo}) + 180 \quad (3)$$

$$\theta_{Cadera} = (\theta_{Cadera} - \theta_{Muslo}) \quad (4)$$

Matriz de rotación, se deben aplicar cuando se da el uso de ángulos de Euler, y pueden representar la rotación de un cuerpo en el espacio. Para el desarrollo, se realiza la definición de los ángulos de Euler de la rodilla y la cadera para ambas piernas, los cuales tendrá una dimensión de [1596, 3]. Los ángulos de Euler se encuentran en grado y son

pasados a radianes para su procesamiento. Posteriormente se inicializan las matrices de rotación para la cadera y la rodilla. Paso siguiente, se calculan los ángulos de Euler para cada conjunto de ángulos en la matriz de rotación en la cual se aplican las siguientes ecuaciones dependiendo de cada uno de sus ejes:

Eje x:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{bmatrix}$$

Eje y:

$$\begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix}$$

Eje z:

$$\begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Con ello se obtuvo el ciclo de marcha. Para luego realizar el análisis cinemático, en donde se encuentra el índice en el momento donde se produce el cambio de marcha. Por último, se realizó el cálculo total de pasos realizado por la persona, el cálculo de la duración de marcha y la cadencia que se tiene en la marcha.

Luego se calculan las métricas, de tiempos de apoyo, tiempos de balanceo, cadencia y velocidad de marcha. La cual se mostrará en una tabla, que se puede visualizar en la siguiente figura:

**Figura 19** Tabla implementada en la interfaz de Cálculo de métricas

Cadencia	98.4 pasos7min
AT apoyo	1.134
AT balanceo	0.453
Vel de marcha	0.594 metros/seg

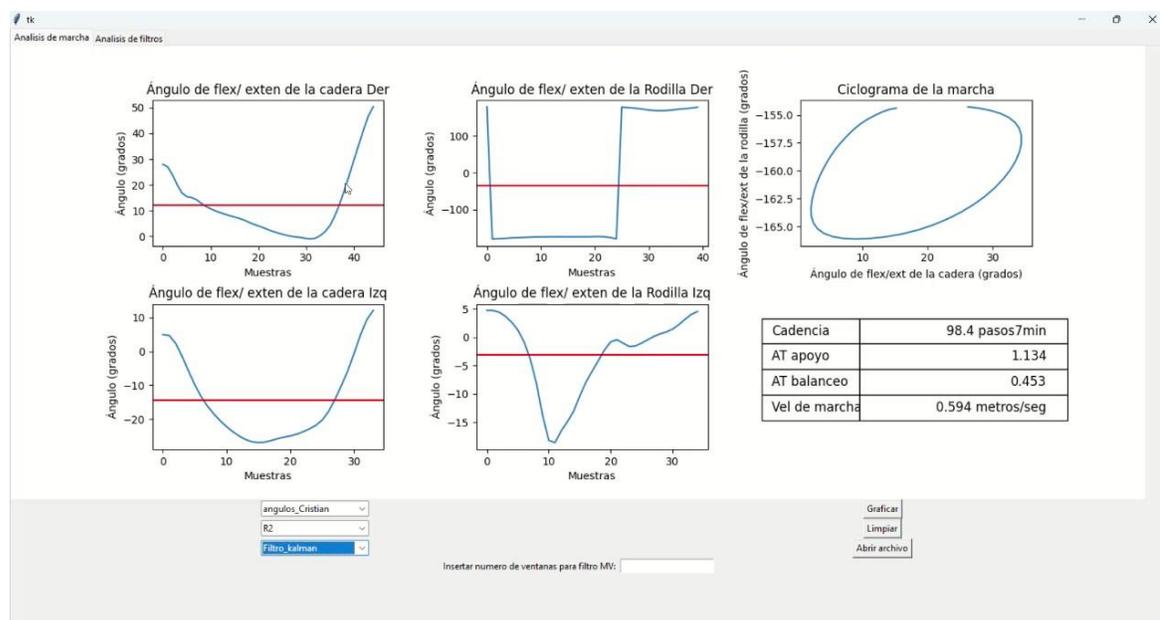
*Nota:* Elaboración propia

Las gráficas obtenidas en esta se pueden evidenciar en la Etapa 3.

### 4.3 Etapa 3: Pruebas de funcionamiento de la interfaz

La interfaz desarrollada consta de dos pestañas principales: 1) análisis de marcha, y 2) análisis de filtros. La pestaña inicial es la primera (Figura 20), en la que es posible cargar la base de datos de interés, es decir, se escoge el paciente y posteriormente se escoge cuál muestra de datos se quiere analizar. Al graficar, en esta pestaña se obtiene la representación cartesiana de los ángulos de extensión/flexión tanto del lado derecho, como del lado izquierdo, de la cadera y la rodilla. Adicionalmente, se obtiene el ciclograma de la marcha y una tabla de información adicional de la marcha (cadencia, tiempo de apoyo, tiempo de balanceo y velocidad de la marcha).

**Figura 20** Interfaz gráfica - Análisis de marcha

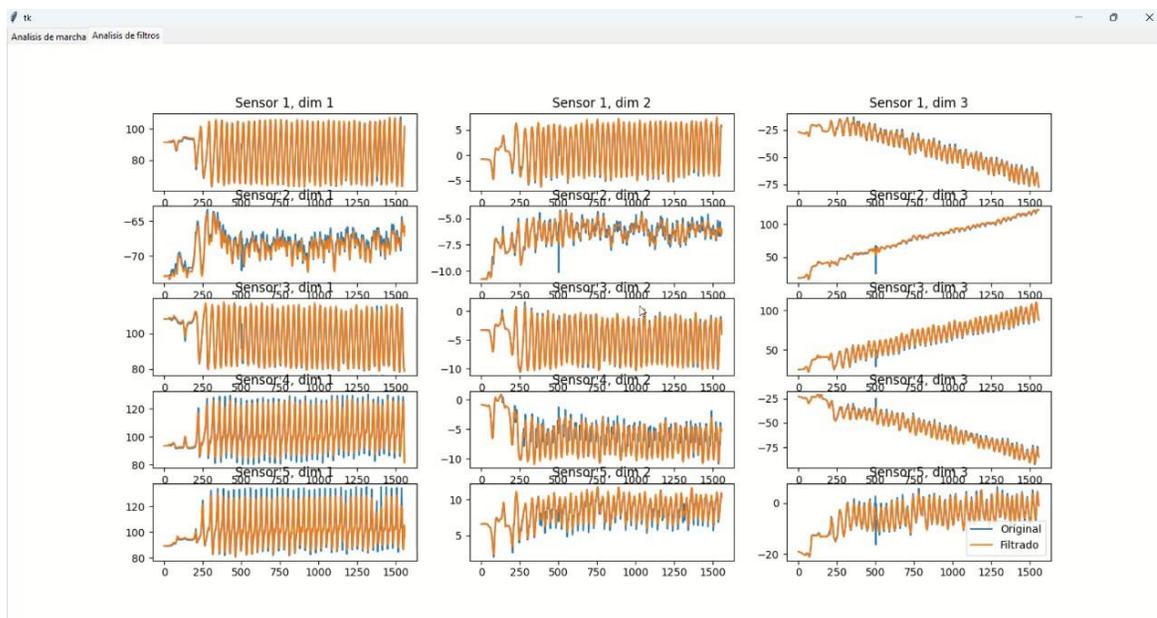


*Nota:* Elaboración propia

Para el uso de los filtros, el filtro Kalman se usa en automático, sin embargo, cuando se usa el filtro de media móvil, es necesario escoger el número de ventanas para el filtro, el

cual puede tomar un valor entre el siguiente listado: 1, 11, 21, 31, 41, 51 o 61 como valor máximo. Al seleccionar este filtro y el número de ventanas, es posible ver una variación en la pestaña de análisis de filtros (Figura 21). En donde el color azul en la gráfica representa la señal original, y el color naranja representa la señal ya filtrada.

**Figura 21** Interfaz gráfica – Análisis de filtros con filtro de media móvil



*Nota:* Elaboración propia

Una de las características importantes de la interfaz desarrollada es que permite limpiar la información que se está visualizando, en el caso de querer analizar un paciente completamente nuevo, y no realizar una comparación con los registros pasados. Mientras que, por otro lado, tiene también la posibilidad de cargar a la interfaz un archivo que ya se venía usando previamente, es decir, si se tiene un registro de muchos pacientes, y se quiere comparar entre ellos, es posible importar el archivo desde una extensión de Excel, y posteriormente seleccionar al paciente a analizar. Esta función permite que no se pierda información a lo largo de análisis a muchos pacientes.

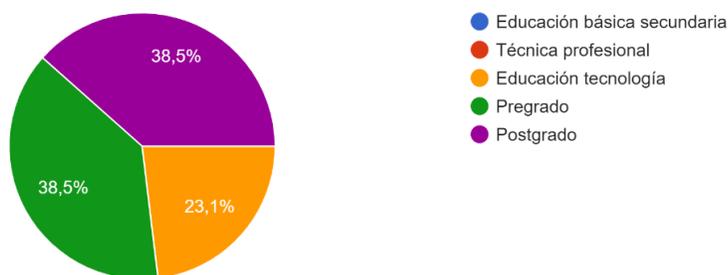
#### 4.4 Etapa 4: Validación de la usabilidad de la interfaz

Se realiza el diseño de una encuesta en la herramienta Google Forms, con el ánimo de conocer la aceptación, usabilidad y accesibilidad de la interfaz. La toma de datos se realizó de manera remota, donde se envía un video explicativo al encuestado y un manual en el cual se encuentra el paso a paso, de la descarga de la interfaz, el Software de Python y Visual Code y como interactuar para dar uso dicha interfaz (Anexo C). A continuación, se plantearon las siguientes preguntas para el formulario:

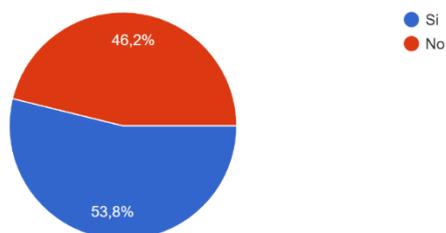
1. ¿Considera amigable y comprensible el uso de la interfaz?
2. ¿Considera que las gráficas de la interfaz son fáciles de interpretar?
3. ¿La presentación de los contenidos es visualmente atrayente?
4. ¿La interfaz cumple con el objetivo del tratamiento de datos?
5. ¿Considera que la interfaz es un desarrollo útil para futuros trabajos?

La encuesta se realizó a trece (13) personas, generando como resultado:

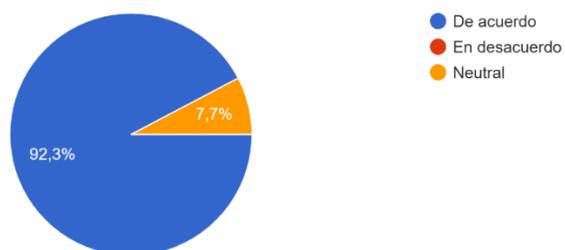
**Figura 22** Nivel de educación de los encuestados



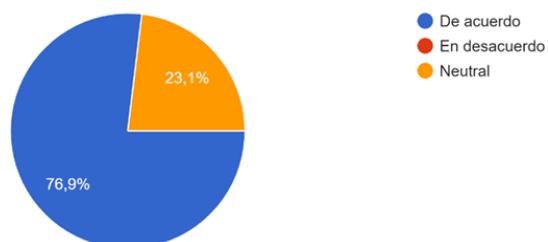
*Nota:* Elaboración propia

**Figura 23** Conocimientos de ingeniería de los encuestados

*Nota:* Elaboración propia

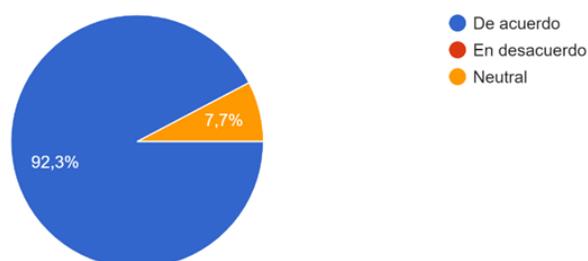
**Figura 24** Resultado de primera pregunta planteada a los encuestados en Diagrama Circular

*Nota:* Elaboración propia

**Figura 25** Resultado de segunda pregunta planteada a los encuestados en Diagrama Circular

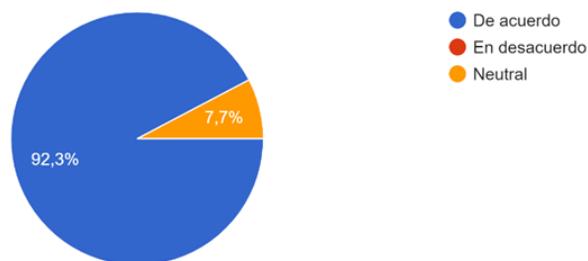
*Nota:* Elaboración propia

**Figura 26** Resultado de tercera pregunta planteada a los encuestados en Diagrama Circular



*Nota:* Elaboración propia

**Figura 27** Resultado de cuarta pregunta planteada a los encuestados en Diagrama Circular



*Nota:* Elaboración propia

**Figura 28** Resultado de quinta pregunta planteada a los encuestados en Diagrama Circular



*Nota:* Elaboración propia

**Tabla 2** Resultados en porcentaje de la encuesta

Preguntas	De acuerdo	En desacuerdo	Neutral
1.¿Considera amigable y comprensible el uso de la interaz?	92,30%	0,00%	7,70%
2.¿Considera que las gráficas de la interfaz son fáciles de interpretar?	76,90%	0,00%	23,10%
3.¿La presentación de los contenidos es visualmente atrayente?	92,30%	0,00%	7,70%
4.¿La interfaz cumple con el objetivo del tratamiento de datos?	92,30%	0,00%	7,70%
5.¿Considera que la interfaz es un desarrollo útil para futuros trabajos?	100%	0,00%	0,00%
<b>Promedio total</b>	<b>91%</b>	<b>0,00%</b>	<b>9%</b>

*Nota:* Elaboración propia

Tal como se evidencia en la tabla de resultados, el 91% de las personas encuestadas están de acuerdo y comprende desarrollo y beneficio de la interfaz del proyecto “Diseño de una interfaz usando Python para la identificación de asimetrías temporales en la marcha de personas con prótesis de miembro inferior.”. En estos mismos resultados se evidencia que, aunque se tiene un porcentaje nulo de desacuerdo, si se presenta el 9% restante en estado neutral. Por tal razón, se debe trabajar en la sensibilización de los conceptos de las personas con formación en áreas ajenas a las ingenierías.

## **Capítulo 5 – Conclusiones**

### **5.1 Conclusiones**

La principal conclusión de este proyecto es que se ha desarrollado una interfaz digital que permite procesar la información cinemática para cuantificar y detectar asimetrías temporales de la marcha en pacientes con prótesis de miembro inferior. Esta herramienta es un avance importante en el campo de la medicina y puede ayudar a mejorar la atención y el tratamiento de los pacientes con prótesis de miembro inferior.

La interfaz desarrollada en este proyecto es muy flexible y permite seleccionar las variables cinemáticas y los parámetros necesarios para detectar y cuantificar las asimetrías temporales en pacientes con prótesis de miembro inferior. Esto es importante, ya que diferentes pacientes pueden tener diferentes necesidades y requerir diferentes variables para una evaluación completa.

La interfaz ha sido validada mediante métodos digitales de recopilación de información, lo que ha demostrado que es una herramienta fácil de usar y que proporciona resultados precisos y fiables. Esto es importante, ya que la precisión y la fiabilidad son fundamentales en la evaluación y el tratamiento de los pacientes con prótesis de miembro inferior.

### **5.2 Recomendaciones**

Se sugiere compartir los resultados de este proyecto con la comunidad científica y médica para promover la adopción de esta herramienta en la práctica clínica. Esto es muy importante ya que compartir los resultados de este proyecto con la comunidad científica y médica se da a conocer la herramienta desarrollada y promover su adopción en la práctica

clínica. Esto puede mejorar la atención de los pacientes con prótesis de miembro inferior no sólo a nivel local, sino en todo el mundo.

Es importante mencionar, que para hacer uso óptimo de la interfaz gráfica se debe descargar programas como lo es Python, Visual Studio Code, Pycharm, para su óptimo funcionamiento.

### **5.3 Trabajos Futuros**

Por otro lado, en el desarrollo del proyecto se logró identificar una serie de oportunidades de mejora, que pueden ser tenidas en cuenta para proyectos futuros, se recomienda continuar mejorando la interfaz y ampliando su funcionalidad para incluir la cuantificación de otras variables cinemáticas y para otros tipos de prótesis. Es decir, la interfaz desarrollada en este proyecto es una herramienta muy útil, pero puede seguir mejorándose y ampliarse para incluir la cuantificación de otras variables cinemáticas, como la aceleración, la velocidad y la fuerza, y para otros tipos de prótesis, como prótesis de brazo o prótesis de cadera.

Aun así, la interfaz desarrollada en este proyecto puede ser utilizada por profesionales de la salud para la evaluación y seguimiento de pacientes con prótesis de miembro inferior, lo cual puede mejorar la atención y el tratamiento que reciben estos pacientes. Sin embargo, hacer que la interfaz se pueda llevar a una aplicación móvil mejoraría el uso de esta.

Finalmente, la interfaz desarrollada en este proyecto puede ser integrada con otras herramientas y tecnologías para mejorar la atención de los pacientes con prótesis de miembro inferior. Por ejemplo, la interfaz puede integrarse con sistemas de telemedicina para permitir a los profesionales de la salud realizar evaluaciones y seguimientos de los pacientes a distancia.



## Bibliografía

- Abdel-Malek, K., & Arora, J. (2013). Human motion simulation: Predictive dynamics. Elsevier Science.
- Accióncontraminas.gov.co. (2020). Estadísticas de víctimas. Recuperado el 23 de agosto de 2020.
- Adistambha, K., Ritz, C., & Burnett, I. (2008). Motion classification using dynamic time warping. En *Multimedia Signal Processing, 2008 IEEE 10th Workshop on* (pp. 622-627). IEEE.
- Alcántara, W., Flores, R., & Garmendia, F. (1999). Prevalencia y riesgo de amputación en pacientes con pie diabético. *Anales de la Facultad de Medicina*, 60(3), 14-8.
- Ayusawa, K., & Nakamura, Y. (2012). Fast inverse kinematics algorithm for large dof system with decomposed gradient computation based on recursive formulation of equilibrium. En *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on* (pp. 3447-3452).
- Blasco, M., Pardo, M., Lacassa, B. y López, A., (2022) Terapia ocupacional en la fase pre-protésica del amputado de miembro inferior, *Revista sanitaria de investigación*.
- Bravo, D. A., Rengifo, C. F., & Agredo, W. (2016). Comparación de dos Sistemas de Captura de Movimiento por medio de las Trayectorias Articulares de Marcha. *Revista mexicana de ingeniería biomédica*, 37(2), 149-160.
- Broche Vázquez, L., Torres Quezada, M., Díaz Novo, C., Pérez Bonne, P., & Sagaró Zamora, R. (2015). Influencia de la asimetría de la marcha en el comportamiento biomecánico de las articulaciones de cadera en pacientes con prótesis transfemorales. *Ingeniare. Revista chilena de ingeniería*, 23(2), 312-322

- Caro Zapata, S. (2022). Verificación de la integridad de los datos adquiridos por un sistema de medición de variables cinéticas, cinemáticas y fisiológicas, ajustado a una bicicleta horizontal en conexión con un juego serio.
- Chao, E.Y.S. (1986). Biomechanics of the Human Gait. In: Schmid-Schönbein, G.W., Woo, S.L.Y., Zweifach, B.W. (eds) *Frontiers in Biomechanics*. Springer, New York, NY.
- Fabrica, G., de., Peña, I., de., Silva, V., de., Ramos, V. (2018). Aprovechamiento de energía, cinemática y estabilidad en la marcha de un paciente con amputación transfemoral sin abordaje de rehabilitación, <https://doi.org/10.15446/revfacmed.v66n1.66724>
- Silva Castellanos, C. (2015). Modelamiento de la marcha humana con prótesis de miembro inferior mediante herramientas de simulación dinámica (“una aplicación en opensim”). <https://repositorio.unal.edu.co/handle/unal/53452>
- Challenger-Pérez, I., Díaz-Ricardo, Y., & Becerra-García, R. A. (2014). El lenguaje de programación Python. *Ciencias Holguín*, 20(2), 1-13.
- Díaz, L. (2016) Prótesis electrónicas: Una nueva esperanza para mejorar la calidad de vida de las personas, *Revista Informador Técnico*.
- Franco-Giraldo (2014). Sistemas de salud en condiciones de mercado: las reformas del último cuarto de siglo, *Rev. Fac. Nac. Salud Pública*; 32(1): 95-107
- Giraldo Castaño L, Pinto-Maquilón JK, Lugo-Agudelo LH, Velásquez-Correa JC, Pastor MP, Posada-Borrero AM, Patiño-Lugo DF, Plata-Contreras JA (2022). Ruta integral de atención en salud para personas con amputaciones de miembro inferior, para mejorar el funcionamiento y la calidad de vida. *Rev. Fac. Nac. Salud Pública*;40(1).
- Gómez, G. S. (1990). Guerra y política en la sociedad colombiana. *Análisis político*, (11), 7-27.

- Herrera-Hernández, A. R., & Sánchez-Uresti, A. (2018, October). Sistemas de captura y análisis de datos de la marcha. In *Memorias del Congreso Nacional de Ingeniería Biomédica* (Vol. 5, No. 1, pp. 362-365).
- Herrera-Montero, M., Gómez- Zoquez, A. P., & Infante- Ruiz, N. M. (2018). Análisis biomecánico de la marcha patológica: un estudio de caso. *Revista científica Especializada En Ciencias De La Cultura Física Y Del Deporte*, 15(35), 95–107
- Higuera-González, R. M., Martínez-García, A., & López-Castillo, V. (2018, November). Instbio: librería de Python para simular circuitos electrónicos de instrumentos biomédicos. In *Memorias del Congreso Nacional de Ingeniería Biomédica* (Vol. 5, No. 1, pp. 234-237).
- Meier III, R. (2014). *Amputee Rehabilitation, An Issue of Physical Medicine and Rehabilitation Clinics of North America*, E-Book, 25(1), Elsevier Health Sciences.
- Navarro, J. (2016). Definición de Biomecánica. *DefinicionABC*. Desde <https://www.definicionabc.com/ciencia/biomecanica.php>
- NYU, (2000) *Protesica Del Miembro Inferior*, New York University
- Oliveira, T. P. de ., Luz, S. C. T. da ., Szücs, A. P., Andrade, M. C. de ., Ávila, A. O. V., Tonon, J. J., & Rosa, F. J. B. de . la .. (2011). Análise do impacto mecânico nas próteses de um sujeito bi-amputado durante a marcha. *Fisioterapia E Pesquisa*, 18(1), 11–16.
- Ramirez, E (2020) *Metodología de prótesis personalizada: aplicación de caso amputación transmetatarsiana* [Tesis] Universidad de Chile
- Reiber, G., Boyko, J., & Smith, E. (1995). Lower extremity foot ulcers and amputations in diabetes. En M. Harris, C. Conie, & Stern (Eds.), *Diabetes in America* (2ª ed., pp.

409-428). US Department of Health and Human Services, Public Health Service, National Institutes of Health.

Roldán Jiménez, C. (2017). Estudio de la cinemática del miembro superior e inferior mediante sensores inerciales.

Stansbury, L. G., Lalliss, S. J., Branstetter, J. G., Bagg, M. R., & Holcomb, J. B. (2008). Amputations in US military personnel in the current conflicts in Afghanistan and Iraq. *Journal of orthopaedic trauma*, 22(1), 43-46.

Vergara, I. (2005). Análisis cinético de marcha para pacientes con prótesis de miembros inferiores [Tesis de pregrado]. Universidad de los Andes.

Waltz, K. (2007). El hombre, el Estado y la Guerra. *Revista Académica de Relaciones Internacionales*, (6), 1-26.

Whittle, M. (2000). *Gait analysis an introduction*. Butterworth-Heinemann.

Whittle M. (2007). *Gait analysis: an introduction (4th ed.)*. Butterworth-Heinemann.



```

        self.function_entry1 = ttk.Combobox(self.pestaañal, state="read
only",
                                           values=["R1", "R2", "R3", "R4
"])
        self.function_entry2 = ttk.Combobox(self.pestaañal, state="read
only",
                                           values=["Filtro_kalman", "F
iltro_Media_Movil"])
        # Crear un cuadro de texto para introducir la función
        self.tiempo = tk.Entry(self.pestaañal)
        label=tk.Label(self.pestaañal, text="Insertar numero de ventana
s para filtro MV: ")
        # Crear un botón para graficar la función
        self.plot_button = tk.Button(self.pestaañal, text='Graficar', c
ommand=self.Procesamiento)
        # Crear un botón para limpiar la función
        self.clear_button = tk.Button(self.pestaañal, text='Limpiar', c
ommand=self.limpiar_ventana)
        # Crear un botón para abrir nuevo archivo
        self.clear_button = tk.Button(self.pestaañal, text='Limpiar', c
ommand=self.limpiar_ventana)
        self.abrir_archivo_button = tk.Button(self.pestaañal, text="Ab
rir archivo", command=self.abrir_archivo)
        #Empaquetar elementos
        self.function_entry.grid(row=1, column=0)
        self.function_entry1.grid(row=2, column=0)
        self.function_entry2.grid(row=3, column=0)
        self.plot_button.grid(row=1, column=1)
        self.clear_button.grid(row=2, column=1)
        self.tiempo.grid(row=4, column=1, sticky="W")
        label.grid(row=4, column=0, sticky="E")
        self.abrir_archivo_button.grid(row=3, column=1)
        #####
###

        # Pestaña 2
        self.pestaañal2 = ttk.Frame(self.pestanas)
        self.pestanas.add(self.pestaañal2, text="Análisis de filtros")

        self.pestanas.pack(expand=1, fill="both")

    def abrir_archivo(self):
        archivo = filedialog.askopenfilename(title="Seleccionar archi
vo", filetypes=(("Archivos CSV", "*.csv"),))
        if archivo:

```

```

        # Leer el archivo CSV con pandas
        df = pd.read_csv(archivo,delimiter=";")
        # Mostrar el DataFrame en la consola
        print(df)

def limpiar_ventana(self):
    self.ax4.clear()
    self.canvas.draw()

def create_widgets(self):
    root.configure(bg="black") # Establecemos el color de fondo
en negro

    # Mostramos la caja con fondo y el botón en la mitad izquierd
a de la ventana
    #box_frame.pack(side="left", fill="both", expand=True, padx=1
0, pady=10)
    # Crear una figura de Matplotlib
    gridsize = (2, 3)
    self.fig = plt.figure(figsize=(15, 6))
    self.ax = plt.subplot2grid(gridsize, (0,0))
    self.ax1 = plt.subplot2grid(gridsize, (1,0))
    self.ax2 = plt.subplot2grid(gridsize, (0,1))
    self.ax3 = plt.subplot2grid(gridsize, (1,1))
    self.ax4 = plt.subplot2grid(gridsize, (0,2))
    self.ax5 = plt.subplot2grid(gridsize, (1,2))
    self.ax5.set_axis_off()
    self.ax5.set_facecolor('none')
    # Ajustar el espaciado horizontal y vertical entre los subplo
ts
    plt.subplots_adjust(wspace=0.4, hspace=0.4)
    # Crear un widget de Matplotlib para mostrar la figura
    self.canvas = FigureCanvasTkAgg(self.fig, master=self.pestaña
1)
    self.canvas.get_tk_widget().grid(row=0, column=0, columnspan=
2)

#####
###
    # Graficar los datos originales y los filtrados
    self.fig1, self.axs = plt.subplots(5, 3, figsize=(20, 20))
    # Crear un widget de Matplotlib para mostrar la figura
    self.canvas1 = FigureCanvasTkAgg(self.fig1, master=self.pesta
ña2)
    #self.canvas1.get_tk_widget().grid(row=0, column=0, columnspa
n=2)

```

```

self.canvas1.get_tk_widget().pack(expand=True, fill='both')

def FiltroKalman(self,z):

    # Definir las matrices del modelo
    A = np.eye(3)
    B = np.eye(3)
    C = np.eye(3)
    Q = np.diag([0.1, 0.1, 0.1])
    R = np.diag([0.01, 0.01, 0.01])

    # Inicializar el estado y la covarianza
    x_est = np.zeros((3,1))
    P_est = np.diag([1, 1, 1])

    # Definir la matriz de medición
    H = np.eye(3)

    # Inicializar el vector de ángulos filtrados
    self.angleskalman = np.zeros((len(z), 3))

    # Ejecutar el filtro de Kalman
    for k in range(len(z)):
        # Medición
        y = z[k,:].reshape((3,1))

        # Predicción del estado y la covarianza
        x_pred = A @ x_est
        P_pred = A @ P_est @ A.T + B @ Q @ B.T

        # Corrección del estado y la covarianza
        K = P_pred @ H.T @ np.linalg.inv(H @ P_pred @ H.T + R)
        x_est = x_pred + K @ (y - H @ x_pred)
        P_est = (np.eye(3) - K @ H) @ P_pred

        # Limitar los ángulos a un rango adecuado
        for i in range(3):
            if x_est[i] > np.pi:
                x_est[i] -= 2 * np.pi
            elif x_est[i] < -np.pi:
                x_est[i] += 2 * np.pi

        # Guardar los resultados
        self.angleskalman[k,:] = x_est.T

def FiltroMediaMovil(self,angles):

```

```

    # Aplicar filtro de media móvil
    window_size = int(self.tiempo.get()) # Tamaño de la ventana
del filtro
    order = 10 # Orden del filtro
    angles_filtered = np.zeros_like(angles)
    for i in range(15):
        angles_filtered[:, i] = savgol_filter(angles[:, i], window_size, order)
    return(angles_filtered)
def Procesamiento(self):
    # Obtener la función ingresada por el usuario
    function_str = self.function_entry.get()
    function_str1= self.function_entry1.get()
    filtros= self.function_entry2.get()
    # Evaluar la función en un rango de valores
    x=df[df["Persona"]==function_str]
    x=x[df["Toma"]==function_str1]
    x1=x.sort_values("time")

#####
#####
    #Filtro pasa bajos
    # Definir las frecuencias de corte (en Hz) del filtro pasa bajo
jo
    hip_cutoff = 10
    knee_cutoff = 5
    # Definir la frecuencia de muestreo (en Hz)
    fs = 25
    # Calcular las frecuencias de corte normalizadas
    nyq = 0.45 * fs
    hip_norm_cutoff = hip_cutoff / nyq
    knee_norm_cutoff = knee_cutoff / nyq
    # Calcular los coeficientes del filtro pasa bajo de Butterworth de segundo orden
    hip_b, hip_a = butter(2, hip_norm_cutoff, btype='low')
    knee_b, knee_a = butter(2, knee_norm_cutoff, btype='low')

    # Filtrar los ángulos de flexión/ extensión
    for column in angulos_cad:
        x1[column] = filtfilt(hip_b, hip_a,x1[column])

    for column in angulos:
        x1[column] = filtfilt(hip_b, hip_a,x1[column])
#####
#####

```

```

angles=x1.iloc[:,3:18].values
#angles=np.deg2rad(angles)
angles_filtered=np.zeros((len(angles),15))

#####Selección de filtro #####
#####
if filtros == "Filtro_kalman":
    anglesx=np.deg2rad(angles)
    for i in range(5):
        data=anglesx[:,3*i:3*(i+1)]
        self.FiltroKalman(data)
        angles_filtered[:,3*i:3*(i+1)] = self.angleskalman
    angles_filtered = np.rad2deg(angles_filtered)
elif filtros == "Filtro_Media_Movil":
    angles_filtered=self.FiltroMediaMovil(angles)

self.angles_org=angles
self.angles_filtred=angles_filtered

MusDer=angles_filtered[:,0:3]
MusIzq=angles_filtered[:,6:9]
PieDer=angles_filtered[:,3:6]
PieIzq=angles_filtered[:,9:12]
Cader=angles_filtered[:,12:15]
# Definir el tiempo de cada muestra (cada 0.04 segundos)
t = x1["time"]
#Ángulo relativo de la rodilla:
AngRodDer=(np.array(PieDer)-np.array(MusDer))+180
AngRodIzq=(np.array(PieIzq)-np.array(MusIzq))+180

#Ángulo relativo de la cadera:
AngCaderDer=(np.array(Cader)-np.array(MusDer))
AngCaderIzq=(np.array(Cader)-np.array(MusIzq))
##### Matriz de Rotacion #####
#####

# Definir los ángulos de Euler en grados para la cadera y la
rodilla
hip_angles = AngCaderDer # Ángulos de Euler en grados para l
a cadera (dimensiones: [1596, 3])
hip_angles_Izq = AngCaderIzq
knee_angles = AngRodDer
knee_angles_Izq = AngRodIzq
# Convertir los ángulos de Euler a radianes
hip_angles = np.radians(hip_angles)
hip_angles_Izq = np.radians(hip_angles_Izq)

```

```

knee_angles = np.radians(knee_angles)
knee_angles_Izq = np.radians(knee_angles_Izq)

# Inicializar las matrices de rotación para la cadera y la rodilla
R_hip = np.zeros((hip_angles.shape[0], 3, 3))
R_hip_izq = np.zeros((hip_angles_Izq.shape[0], 3, 3))
R_knee = np.zeros((knee_angles.shape[0], 3, 3))
R_knee_izq = np.zeros((knee_angles_Izq.shape[0], 3, 3))

# Calcular las matrices de rotación para cada conjunto de ángulos de Euler
for i in range(hip_angles.shape[0]):
    R_hip[i] = np.array([
        [np.cos(hip_angles[i, 1]) * np.cos(hip_angles[i, 2]),
        -
        np.cos(hip_angles[i, 1]) * np.sin(hip_angles[i, 2]), np.sin(hip_angles[i, 1])],
        [np.sin(hip_angles[i, 0]) * np.sin(hip_angles[i, 1])
        * np.cos(hip_angles[i, 2]) + np.cos(hip_angles[i, 0]) * np.sin(hip_angles[i, 2]), -
        np.sin(hip_angles[i, 0]) * np.sin(hip_angles[i, 1]) * np.sin(hip_angles[i, 2]) + np.cos(hip_angles[i, 0]) * np.cos(hip_angles[i, 2]), -
        np.sin(hip_angles[i, 0]) * np.cos(hip_angles[i, 1])],
        [-
        np.cos(hip_angles[i, 0]) * np.sin(hip_angles[i, 1]) * np.cos(hip_angles[i, 2]) + np.sin(hip_angles[i, 0]) * np.sin(hip_angles[i, 2]), np.cos(hip_angles[i, 0]) * np.sin(hip_angles[i, 1]) * np.sin(hip_angles[i, 2]) + np.sin(hip_angles[i, 0]) * np.cos(hip_angles[i, 2]), np.cos(hip_angles[i, 0]) * np.cos(hip_angles[i, 1])]]
    ])
    R_hip_izq[i] = np.array([
        [np.cos(hip_angles_Izq[i, 1]) * np.cos(hip_angles_Izq[i, 2]), -
        np.cos(hip_angles_Izq[i, 1]) * np.sin(hip_angles_Izq[i, 2]), np.sin(hip_angles_Izq[i, 1])],
        [np.sin(hip_angles_Izq[i, 0]) * np.sin(hip_angles_Izq[i, 1]) * np.cos(hip_angles_Izq[i, 2]) + np.cos(hip_angles_Izq[i, 0])
        * np.sin(hip_angles_Izq[i, 2]), -
        np.sin(hip_angles_Izq[i, 0]) * np.sin(hip_angles_Izq[i, 1]) * np.sin(hip_angles_Izq[i, 2]) + np.cos(hip_angles_Izq[i, 0]) * np.cos(hip_angles_Izq[i, 2]), -
        np.sin(hip_angles_Izq[i, 0]) * np.cos(hip_angles_Izq[i, 1])],
        [-
        np.cos(hip_angles_Izq[i, 0]) * np.sin(hip_angles_Izq[i, 1]) * np.cos(hip_angles_Izq[i, 2]) + np.sin(hip_angles_Izq[i, 0]) * np.sin(hip_angles_Izq[i, 2])]]
    ])

```

```

les_Izq[i, 2]), np.cos(hip_angles_Izq[i, 0]) * np.sin(hip_angles_Izq[
i, 1]) * np.sin(hip_angles_Izq[i, 2]) + np.sin(hip_angles_Izq[i, 0])
* np.cos(hip_angles_Izq[i, 2]), np.cos(hip_angles_Izq[i, 0]) * np.cos
(hip_angles_Izq[i, 1])]
    ])
    R_knee[i] = np.array([
        np.cos(knee_angles[i, 1]) * np.cos(knee_angles[i, 2])
    ), -
    np.cos(knee_angles[i, 1]) * np.sin(knee_angles[i, 2]), np.sin(knee_an
gles[i, 1])],
        [np.sin(knee_angles[i, 0]) * np.sin(knee_angles[i, 1]
) * np.cos(knee_angles[i, 2]) + np.cos(knee_angles[i, 0]) * np.sin(kn
ee_angles[i, 2]), -
    np.sin(knee_angles[i, 0]) * np.sin(knee_angles[i, 1]) * np.sin(knee_a
ngles[i, 2]) + np.cos(knee_angles[i, 0]) * np.cos(knee_angles[i, 2]),
    -np.sin(knee_angles[i, 0]) * np.cos(knee_angles[i, 1])],
        [-
    np.cos(knee_angles[i, 0]) * np.sin(knee_angles[i, 1]) * np.cos(knee_a
ngles[i, 2]) + np.sin(knee_angles[i, 0]) * np.sin(knee_angles[i, 2]),
    np.cos(knee_angles[i, 0]) * np.sin(knee_angles[i, 1]) * np.sin(knee_
angles[i, 2]) + np.sin(knee_angles[i, 0]) * np.cos(knee_angles[i, 2])
    ], np.cos(knee_angles[i, 0]) * np.cos(knee_angles[i, 1])]
    ])

    R_knee_izq[i] = np.array([
        np.cos(knee_angles_Izq[i, 1]) * np.cos(knee_angles_I
zq[i, 2]), -
    np.cos(knee_angles_Izq[i, 1]) * np.sin(knee_angles_Izq[i, 2]), np.sin
(knee_angles_Izq[i, 1])],
        [np.sin(knee_angles_Izq[i, 0]) * np.sin(knee_angles_I
zq[i, 1]) * np.cos(knee_angles_Izq[i, 2]) + np.cos(knee_angles_Izq[i,
0]) * np.sin(knee_angles_Izq[i, 2]), -
    np.sin(knee_angles_Izq[i, 0]) * np.sin(knee_angles_Izq[i, 1]) * np.si
n(knee_angles_Izq[i, 2]) + np.cos(knee_angles_Izq[i, 0]) * np.cos(kne
e_angles_Izq[i, 2]), -
    np.sin(knee_angles_Izq[i, 0]) * np.cos(knee_angles_Izq[i, 1])],
        [-
    np.cos(knee_angles_Izq[i, 0]) * np.sin(knee_angles_Izq[i, 1]) * np.co
s(knee_angles_Izq[i, 2]) + np.sin(knee_angles_Izq[i, 0]) * np.sin(kne
e_angles_Izq[i, 2]), np.cos(knee_angles_Izq[i, 0]) * np.sin(knee_angl
es_Izq[i, 1]) * np.sin(knee_angles_Izq[i, 2]) + np.sin(knee_angles_Iz
q[i, 0]) * np.cos(knee_angles_Izq[i, 2]), np.cos(knee_angles_Izq[i, 0
]) * np.cos(knee_angles_Izq[i, 1])]
    ])

```

```

#####
#####
# Crear matrices vacías para almacenar los ángulos de Euler d
e la cadera y la rodilla
hip_euler_angles = np.zeros_like(hip_angles)
hip_izq_euler_angles = np.zeros_like(hip_angles_Izq)
knee_euler_angles = np.zeros_like(knee_angles)
knee_izq_euler_angles = np.zeros_like(knee_angles_Izq)
# Calcular los ángulos de Euler para cada conjunto de ángulos
en las matrices de rotación
for i in range(len(R_hip)):
    # Matriz de rotación de la cadera derecha
    R_hip_i = R_hip[i]
    hip_euler_angles[i] = np.array([
        np.arctan2(R_hip_i[2, 1], R_hip_i[2, 2]),
        -np.arcsin(R_hip_i[2, 0]),
        np.arctan2(R_hip_i[1, 0], R_hip_i[0, 0])
    ])

    # Matriz de rotación de la cadera izquierda
    R_hip_izq_i = R_hip_izq[i]
    hip_izq_euler_angles[i] = np.array([
        np.arctan2(R_hip_izq_i[2, 1], R_hip_izq_i[2, 2]),
        -np.arcsin(R_hip_izq_i[2, 0]),
        np.arctan2(R_hip_izq_i[1, 0], R_hip_izq_i[0, 0])
    ])

    # Matriz de rotación de la rodilla derecha
    R_knee_i = R_knee[i]
    knee_euler_angles[i] = np.array([
        np.arctan2(R_knee_i[2, 1], R_knee_i[2, 2]),
        -np.arcsin(R_knee_i[2, 0]),
        np.arctan2(R_knee_i[1, 0], R_knee_i[0, 0])
    ])

    # Matriz de rotación de la rodilla izquierda
    R_knee_izq_i = R_knee_izq[i]
    knee_izq_euler_angles[i] = np.array([
        np.arctan2(R_knee_izq_i[2, 1], R_knee_izq_i[2, 2]),
        -np.arcsin(R_knee_izq_i[2, 0]),
        np.arctan2(R_knee_izq_i[1, 0], R_knee_izq_i[0, 0])
    ])

# Convertir los ángulos de Euler de radianes a grados
hip_euler_angles_deg = np.rad2deg(hip_euler_angles)
hip_izq_euler_angles_deg = np.rad2deg(hip_izq_euler_angles)
knee_euler_angles_deg = np.rad2deg(knee_euler_angles)
knee_izq_euler_angles_deg = np.rad2deg(knee_izq_euler_angles)

```

```

#####
#####

# Convertir los ángulos de Euler de radianes a grados
hip_euler_angles_deg = np.rad2deg(hip_euler_angles)
hip_izq_euler_angles_deg = np.rad2deg(hip_izq_euler_angles)
knee_euler_angles_deg = np.rad2deg(knee_euler_angles)
knee_izq_euler_angles_deg = np.rad2deg(knee_izq_euler_angles)
# Extraer los ángulos de Euler para cada momento del ciclo de
marcha
hip_flex_ext = hip_euler_angles_deg[:, 0]
hip_abd_add = hip_euler_angles_deg[:, 1]
hip_int_ext = hip_euler_angles_deg[:, 2]

hip_izq_flex_ext = hip_izq_euler_angles_deg[:, 0]
hip_izq_abd_add = hip_izq_euler_angles_deg[:, 1]
hip_izq_int_ext = hip_izq_euler_angles_deg[:, 2]

knee_flex_ext = knee_euler_angles_deg[:, 0]
knee_abd_add = knee_euler_angles_deg[:, 1]
knee_int_ext = knee_euler_angles_deg[:, 2]

knee_izq_flex_ext = knee_izq_euler_angles_deg[:, 0]
knee_izq_abd_add = knee_izq_euler_angles_deg[:, 1]
knee_izq_int_ext = knee_izq_euler_angles_deg[:, 2]
#####
# Encontrar los índices donde se produce el cambio de fase en
el ciclo de la marcha (aproximadamente cada 0.5 segundos)
self.phase_indices = np.where(np.diff(knee_izq_flex_ext > np.
mean(knee_izq_flex_ext)))[0]

# Contar el número de pasos
num_steps = len(self.phase_indices) / 2

# Calcular la duración de la marcha en segundos
total_time = max(t)

# Calcular la cadencia de la marcha
self.cadence = (60 * num_steps) / 25

hip_angles_deg=[hip_flex_ext,hip_izq_flex_ext,knee_flex_ext,k
nee_izq_flex_ext]
self.cycles_plot=[0,0,0,0]
self.mean_flex_ext=[0,0,0,0]

```

```

        for i in range(len(hip_angles_deg)):
            # Detectar los máximos locales de la señal de ángulo de l
a cadera
            peaks, _ = find_peaks(hip_angles_deg[i], prominence=20)
            # Segmentar los datos correspondientes a un ciclo de marc
ha
            self.cycles_plot[i] = hip_angles_deg[i][peaks[0]:peaks[1]
]
            # Promediar los ángulos de ese ciclo
            self.mean_flex_ext[i] = np.mean(self.cycles_plot[i])
            self.metricas(t,total_time)

##### Filtro de media movil #####
#####

from numpy import convolve

# Definir tamaño de la ventana de media móvil
window_size = 15

# Crear ventana de media móvil
window = np.ones(window_size)/window_size

# Aplicar filtro de media móvil a los ángulos de cadera y rod
illa
hip_angles_filt = convolve(hip_flex_ext, window, mode='valid'
)
knee_angles_filt = convolve(knee_flex_ext, window, mode='vali
d')
# Detectar los máximos locales de la señal de ángulo de la ro
dilla
peaks, _ = find_peaks(knee_angles_filt, prominence=10)

# Segmentar los datos correspondientes a un ciclo de marcha
self.cycle_knee_flex_ext = knee_angles_filt[peaks[0]:peaks[1]
]
self.cycle_hip_flex_ext = hip_angles_filt[peaks[0]:peaks[1]]
self.plot_function()

def metricas(self,t,total_time):
    # Calcular el tiempo de apoyo y tiempo de balanceo
    # Calcular el tiempo de apoyo y tiempo de balanceo
    apoyo_times = []
    balanceo_times = []
    t=t.values
    for i in range(0, len(self.phase_indices)-1, 2):

```

```

        if self.phase_indices[i] < len(t):
            start_apoyo = t[self.phase_indices[i]]
        else:
            start_apoyo = t[-1]
        if self.phase_indices[i+1] < len(t):
            end_apoyo = t[self.phase_indices[i+1]]
        else:
            end_apoyo = t[-1]
        apoyo_times.append(end_apoyo - start_apoyo)
        if i+2 < len(self.phase_indices):
            balanceo_times.append(t[self.phase_indices[i+2]] - en
d_apoyo)

# Calcular la velocidad de la marcha
longitud_paso = 1.0 # Longitud del paso en metros (ejemplo)
num_pasos = len(apoyo_times)
distancia_recorrida = num_pasos * longitud_paso
velocidad_marcha = distancia_recorrida / total_time
self.met=[str(self.cadence)+" pasos7min",str(round(np.mean(ap
oyo_times),3)),str(round(np.mean(balanceo_times),3)),str(round(veloci
dad_marcha,3))+ " metros/seg"]

def plot_function(self):
    # Graficar los datos segmentados y suavizados
    #####
    # Limpiar la figura y graficar la función
    self.ax.clear()
    self.ax1.clear()
    self.ax2.clear()
    self.ax3.clear()
    self.ax5.clear()

    self.ax.set_title('Ángulo de flex/ exten de la cadera Der')
    self.ax.set_xlabel('Muestras')
    self.ax.set_ylabel('Ángulo (grados)')
    self.ax1.set_title('Ángulo de flex/ exten de la cadera Izq')
    self.ax1.set_xlabel('Muestras')
    self.ax1.set_ylabel('Ángulo (grados)')
    self.ax2.set_title('Ángulo de flex/ exten de la Rodilla Der')
    self.ax2.set_xlabel('Muestras')
    self.ax2.set_ylabel('Ángulo (grados)')
    self.ax3.set_title('Ángulo de flex/ exten de la Rodilla Izq')
    self.ax3.set_xlabel('Muestras')
    self.ax3.set_ylabel('Ángulo (grados)')
    self.ax4.set_title('Ciclograma de la marcha')

```

```

self.ax4.set_xlabel('Ángulo de flex/ext de la cadera (grados)
')
self.ax4.set_ylabel('Ángulo de flex/ext de la rodilla (grados
)')

self.ax.plot(self.cycles_plot[0])
self.ax1.plot(self.cycles_plot[1])
self.ax2.plot(self.cycles_plot[2])
self.ax3.plot(self.cycles_plot[3])
self.ax.axhline(self.mean_flex_ext[0], color='red')
self.ax1.axhline(self.mean_flex_ext[1], color='red')
self.ax2.axhline(self.mean_flex_ext[2], color='red')
self.ax3.axhline(self.mean_flex_ext[3], color='red')
self.ax4.plot(self.cycle_hip_flex_ext,self.cycle_knee_flex_ex
t)

#####
# Agregar la tabla a la subtrama ax5
self.ax5.set_axis_off()
self.ax5.set_facecolor('none')
filas=["Cadencia","AT apoyo","AT balanceo","Vel de marcha"]
tabla = self.ax5.table(cellText=[[self.met[0]], [self.met[1]]
, [self.met[2]], [self.met[3]]], rowLabels=filas, colLabels=None, loc
='center',bbox=[0.3, 0.2, 0.9, 0.7])
tabla.auto_set_font_size(True)
tabla.set_fontsize(12)
self.canvas.draw()
# Actualizar el widget de Matplotlib

#####plot en segunda pestaña#####

for i in range(5):
    for j in range(3):
        self.axs[i, j].clear()
        idx = 3*i + j
        self.axs[i, j].plot(self.angles_org[:, idx], label='O
riginal')
        self.axs[i, j].plot(self.angles_filtred[:, idx], labe
l='Filtrado')
        self.axs[i, j].set_title('Sensor {}, dim {}'.format(i
+1, j+1))

plt.legend()
self.canvas1.draw()

root = tk.Tk()

```

```
app = Application(master=root)
app.mainloop()
```

**Anexo B.** Enlace a encuesta realizada en Google Forms.

<https://forms.gle/6M8LPkY9NtPnRryE8>

**Anexo C.** Manual de Uso de la interfaz grafica

		<b>Estudiante</b>
--	--	-------------------

	<b>MANUAL DE USO INTERFAZ GRÁFICA</b>	Jeysson Alfonso Calderon Contreras
---	---	---------------------------------------

## MANUAL DE USO DE INTERFAZ GRÁFICA

### Objetivo

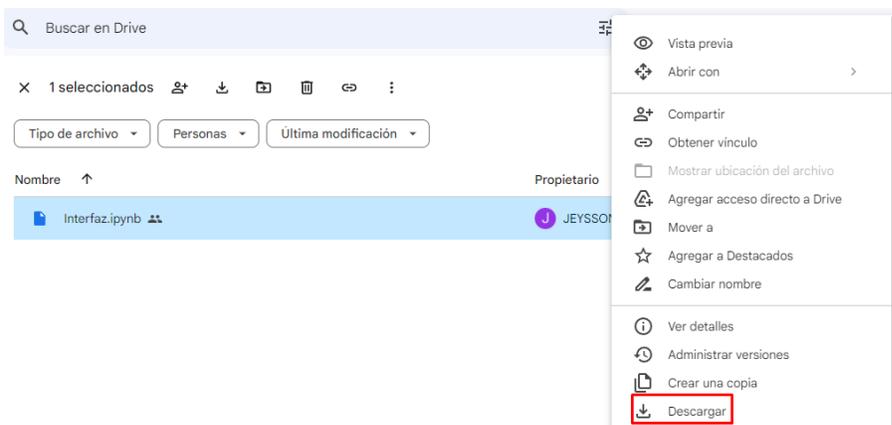
Permitir que el usuario pueda hacer un uso óptimo de la interfaz gráfica desarrollada, estableciendo los pasos que deberá seguir. El manual se desarrolla con el objeto de ser implementada en diferentes áreas.

### Desarrollo del manual de usuario

1. El usuario podrá descargar el código a implementar a partir de un enlace de Google Drive en el cual encontrará un bloc de notas con todo el programa

### Enlace:

[https://drive.google.com/drive/folders/1BsWVqjXjgeGkvH55ACzZ\\_2eBdHPKqsjN?usp=sharing](https://drive.google.com/drive/folders/1BsWVqjXjgeGkvH55ACzZ_2eBdHPKqsjN?usp=sharing)



Nota: es importante mencionar que para esto la persona debe tener acceso a su cuenta de Gmail con el fin de poder acceder al enlace.

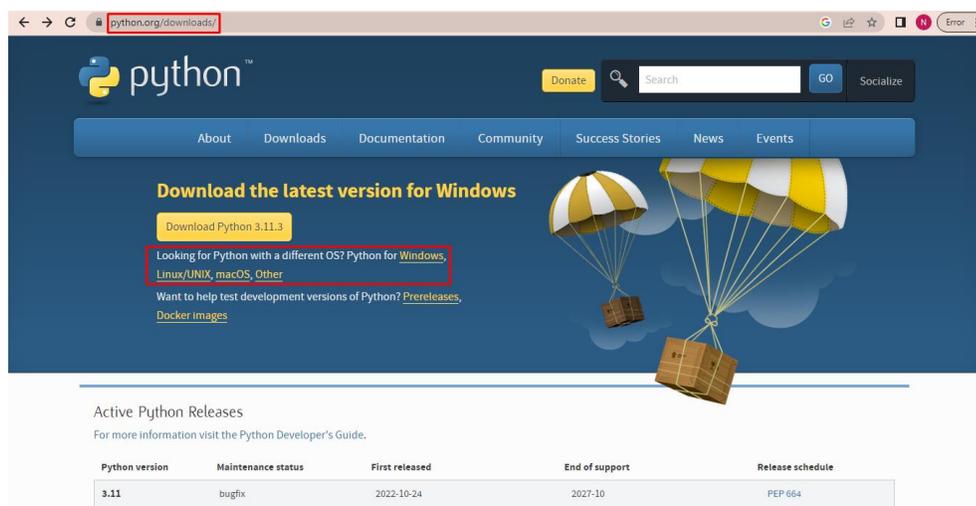
		<b>Estudiante</b>
--	--	-------------------

	<b>MANUAL DE USO INTERFAZ GRÁFICA</b>	Jeysson Alfonso Calderon Contreras
---	---	---------------------------------------

2. Se deben descargar ciertos aplicativos para el uso de la interfaz, para ello se debe tener el lenguaje de programación Python y Visual Studio Code

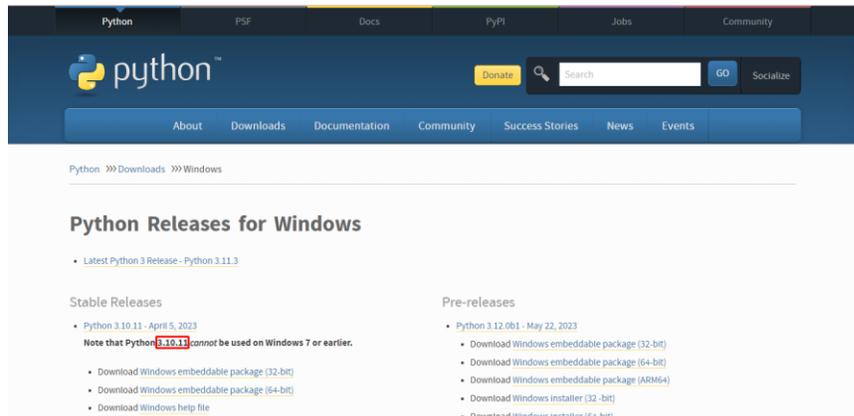
### **Paso a paso para descarga e instalación Python:**

Se debe ingresar al siguiente enlace: <https://www.python.org/downloads/> en donde se encuentra la página oficial de Python y las diferentes versiones están disponibles. Adicionalmente, nos permite escoger el sistema operativo Windows, Linux/UNIX, macOS, Other, dependiendo el equipo con el que cuente el usuario.

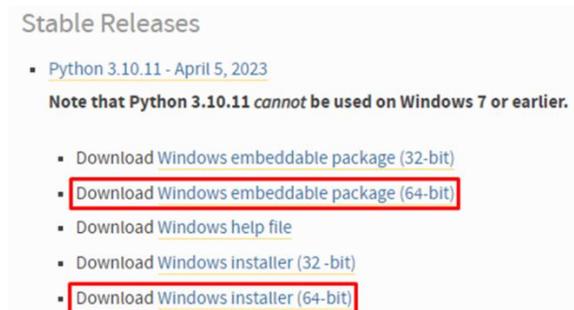


Se da clic en el sistema operativo en el cual se vaya a trabajar, que posteriormente va a dirigir a la siguiente página:

	<b>MANUAL DE USO INTERFAZ GRÁFICA</b>	<b>Estudiante</b>
		Jeysson Alfonso Calderon Contreras

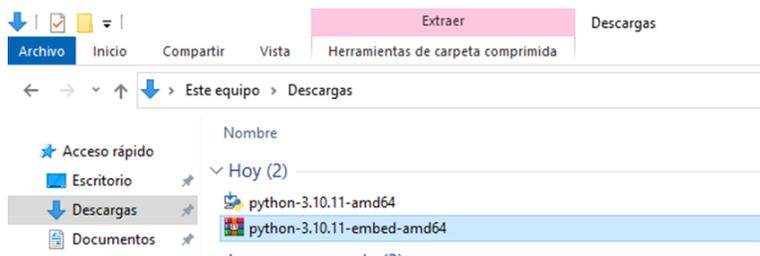


En esta página se encuentra cuáles son las versiones con las que cuenta Python, la interfaz gráfica fue desarrollada con Python v3.8. Sin embargo, es ejecutable en cualquiera otra versión. Se da clic sobre el instalador y el paquete integrable de 64-bits.

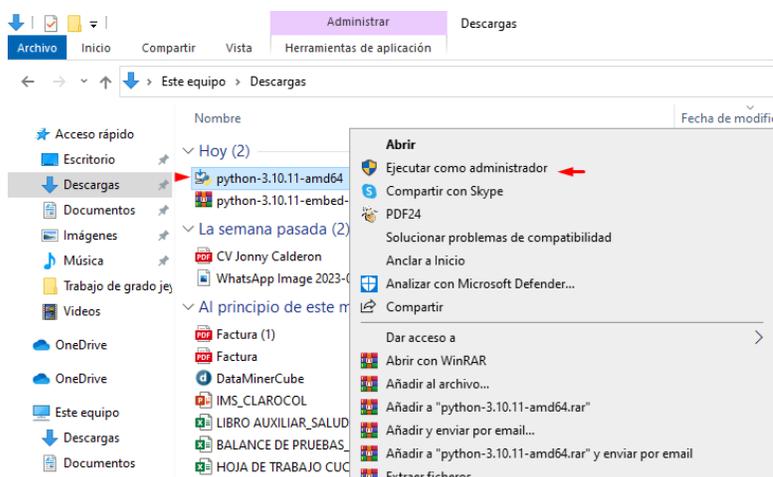


Se espera unos minutos hasta que se tenga la descarga lista, para luego dirigirse a donde se haya escogido guardar los archivos descargados, en este caso se encuentra en la carpeta de descargas.

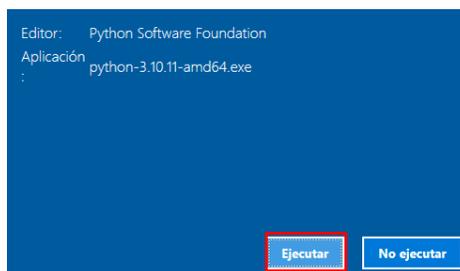
	<b>MANUAL DE USO INTERFAZ GRÁFICA</b>	<b>Estudiante</b>
		Jeysson Alfonso Calderon Contreras



Se da clic derecho sobre el archivo Python-3.13.11-amd65 y posteriormente se selecciona la opción de ejecutar como administrador:

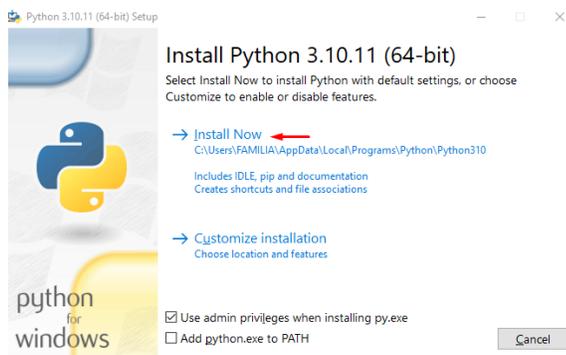


Se debe dar clic sobre ejecutar, con esto se permite que el programa pueda acceder al computador:

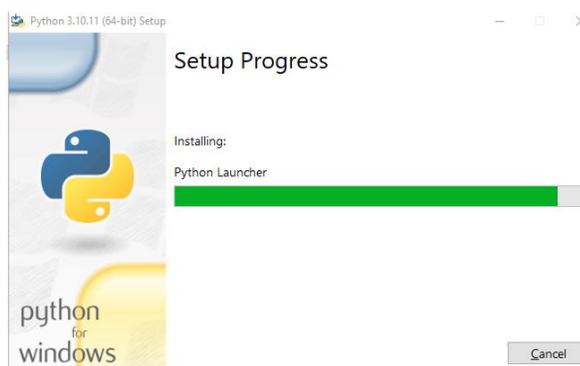


	<b>MANUAL DE USO INTERFAZ GRÁFICA</b>	<b>Estudiante</b>
		Jeysson Alfonso Calderon Contreras

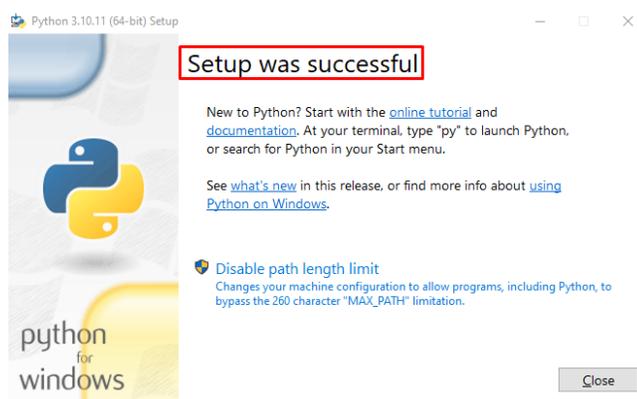
Una vez que se dé clic sobre ejecutar, se selecciona en la pantalla de descarga de Python la siguiente opción:



Luego se espera unos minutos mientras se instala Python en el computador:



Una vez instalado entregará un mensaje de instalación exitosa

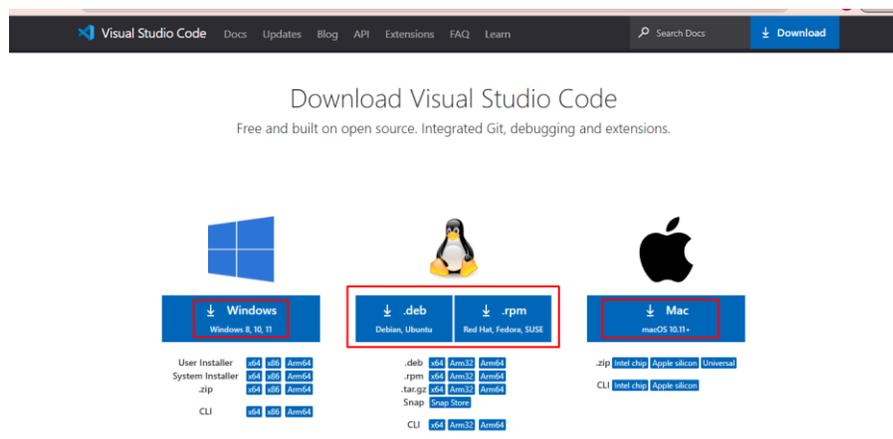


	<b>MANUAL DE USO INTERFAZ GRÁFICA</b>	<b>Estudiante</b>
		Jeysson Alfonso Calderon Contreras

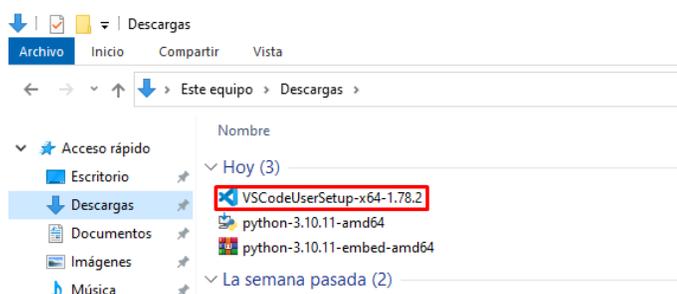
Se da clic el botón de close para continuar con la instalación del siguiente programa para iniciar hacer uso de la interfaz gráfica.

### Paso a paso descarga e instalación de Visual Studio Code:

Visual Studio Code es un editor de código fuente desarrollado por Microsoft para Windows, Linux, macOS y Web. Se debe ingresar al siguiente enlace: <https://code.visualstudio.com/download>, nuevamente dependiendo del sistema operativo en el que se va a trabajar se debe seleccionar en la página de Visual la opción correspondiente



Se espera unos minutos hasta que el navegador realice la descarga del programa. Una vez finalizada la descarga se debe dirigir a la carpeta en donde decidió guardar el archivo



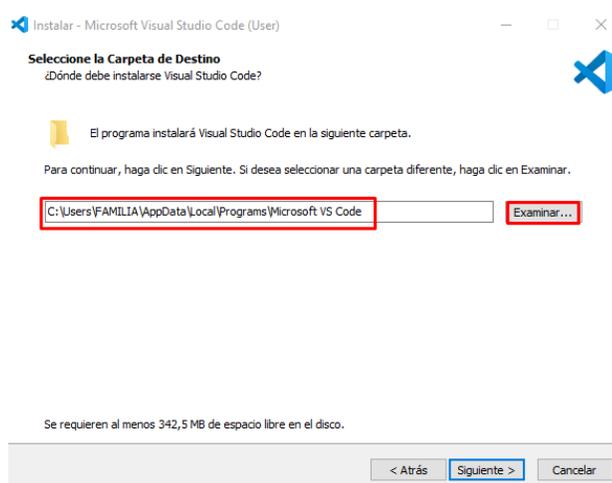
		<b>Estudiante</b>
--	--	-------------------

	<p align="center"><b>MANUAL DE USO INTERFAZ GRÁFICA</b></p>	<p align="center">Jeysson Alfonso Calderon Contreras</p>
---	---	--

Se da clic derecho sobre el archivo y nuevamente se ejecuta el programa en modo administrador. Se leen los términos y condiciones de la licencia de Visual para Microsoft y se aceptan las condiciones para posteriormente pulsar el botón de siguiente

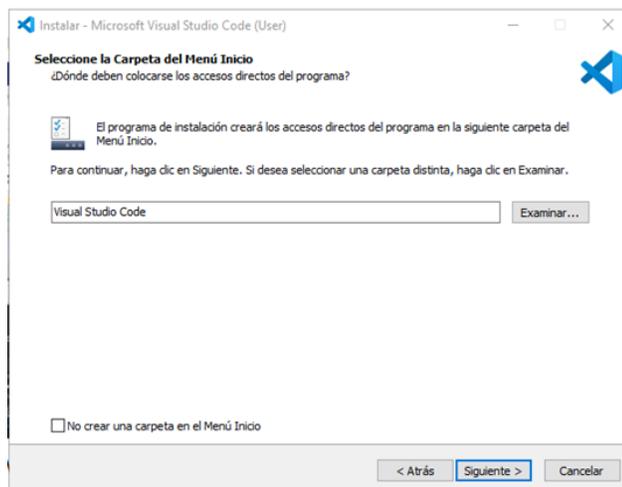


Ubica la carpeta en la cual decide guardar el programa, para este caso deja la carpeta preestablecida, si se quiere cambiar se da clic en el botón de examinar y se elige la correspondiente.

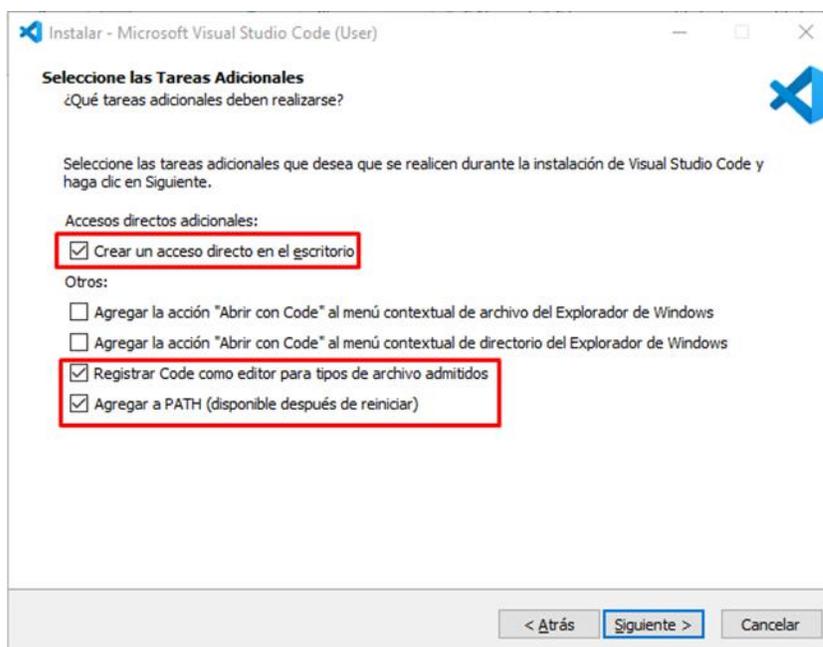


	<b>MANUAL DE USO INTERFAZ GRÁFICA</b>	<b>Estudiante</b>
		Jeysson Alfonso Calderon Contreras

En la siguiente pantalla se muestra el nombre del aplicativo, si decide cambiar el nombre lo puede realizar. Sin embargo, se sugiere dejar el nombre por defecto.

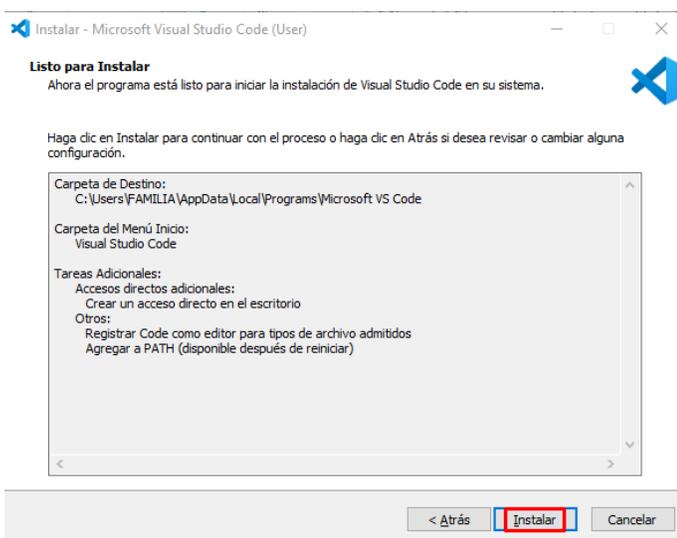


En la siguiente página se visualizan algunas opciones, en esta se debe seleccionar las siguientes:

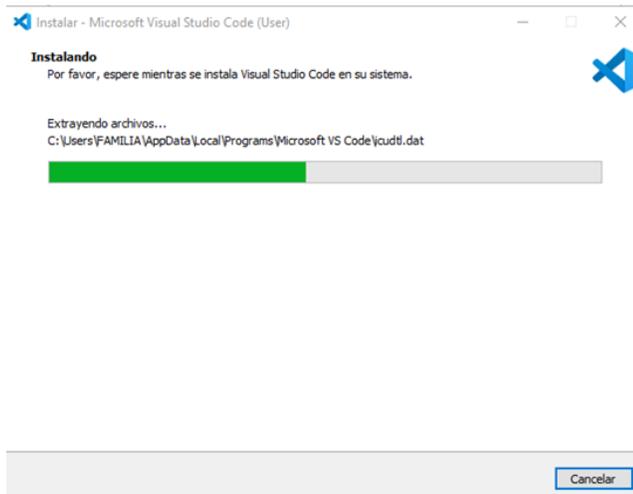


	<b>MANUAL DE USO INTERFAZ GRÁFICA</b>	<b>Estudiante</b>
		Jeysson Alfonso Calderon Contreras

Permite crear un acceso directo de Visual Studio Code al escritorio y logra que los archivos de programación puedan ser abiertos desde el computador por medio de Visual. Se da clic siguiente y en esta página esta la opción de iniciar con la instalación

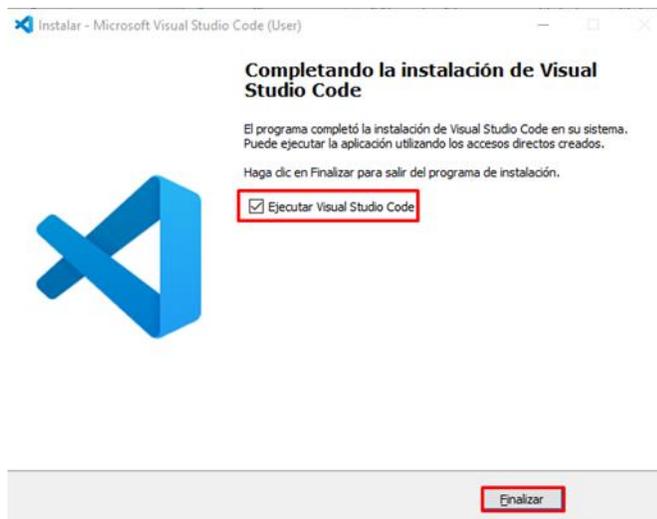


Se esperan aproximadamente tres minutos en el proceso de instalación

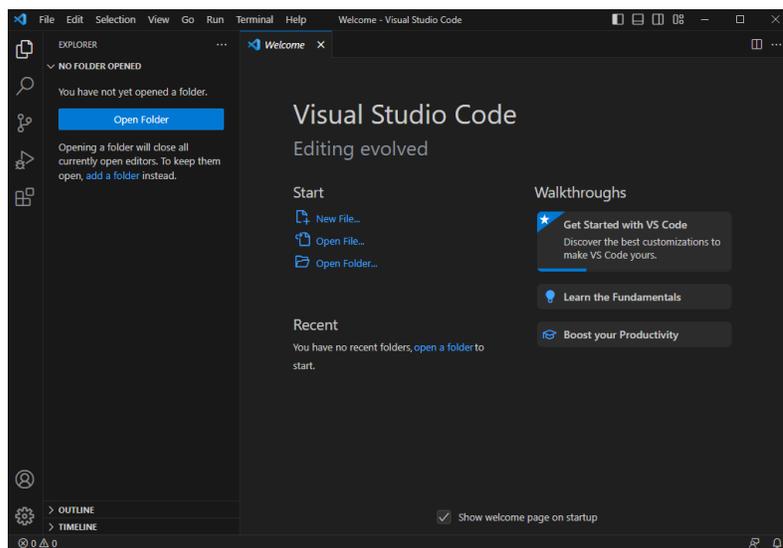


	<b>MANUAL DE USO INTERFAZ GRÁFICA</b>	<b>Estudiante</b>
		Jeysson Alfonso Calderon Contreras

Terminada la instalación de Visual Studio Code, se da clic en la opción de ejecutar Visual Studio Code y finalizar para terminar dicha instalación.

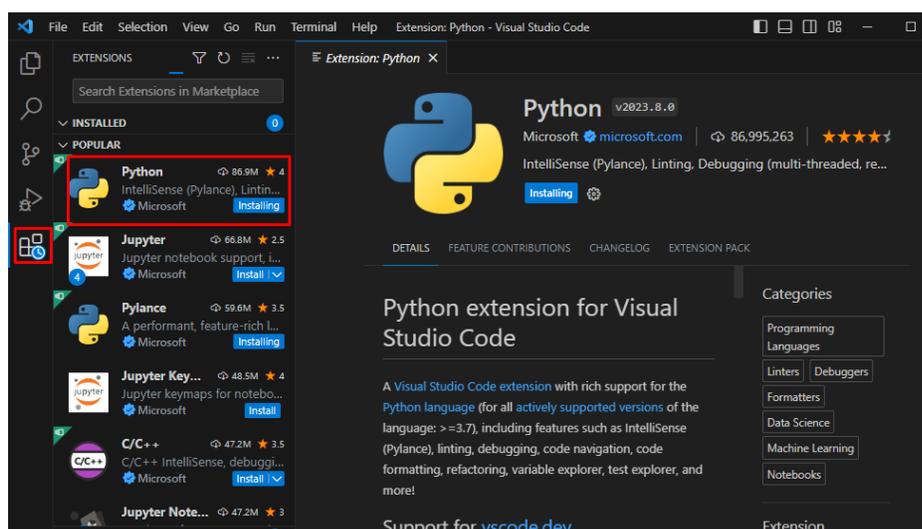


La interfaz gráfica de Visual Studio Code, se visualiza de la siguiente manera:

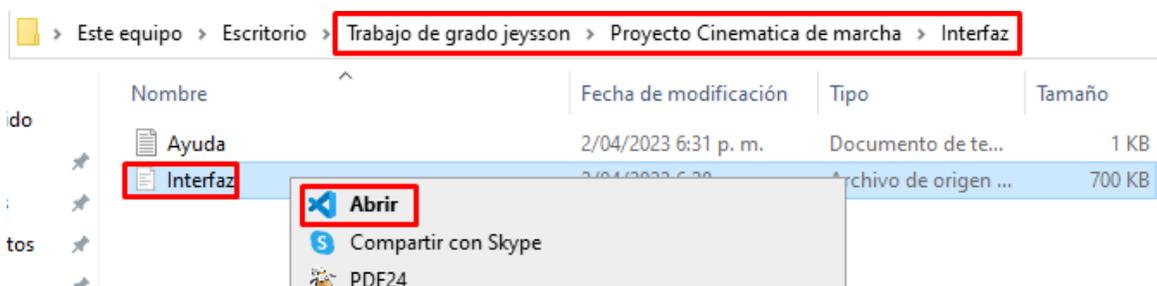


	<b>MANUAL DE USO INTERFAZ GRÁFICA</b>	<b>Estudiante</b>
		Jeysson Alfonso Calderon Contreras

Se da clic en la quinta (5) opción de la barra que se encuentra a la izquierda, este icono tiene el nombre de extensiones en el cual se puede descargar el plug de Python tal cual como se evidencia en la siguiente imagen

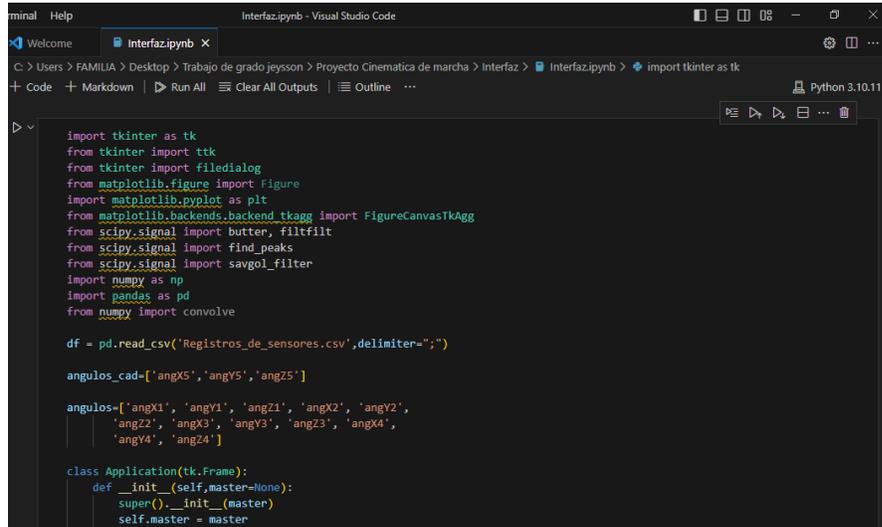


Para abrir la interfaz gráfica, se debe ubicar donde se guardó el archivo. En este caso el archivo tiene el nombre de “Interfaz”. Se da clic derecho sobre él y se selecciona la opción de abrir con Visual Studio Code.



Una vez el programa abra, se tendrá acceso al código desarrollado para el proyecto

	<b>MANUAL DE USO INTERFAZ GRÁFICA</b>	<b>Estudiante</b>
		Jeysson Alfonso Calderon Contreras



```
import tkinter as tk
from tkinter import ttk
from tkinter import filedialog
from matplotlib.figure import Figure
import matplotlib.pyplot as plt
from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg
from scipy.signal import butter, filtfilt
from scipy.signal import find_peaks
from scipy.signal import savgol_filter
import numpy as np
import pandas as pd
from numpy import convolve

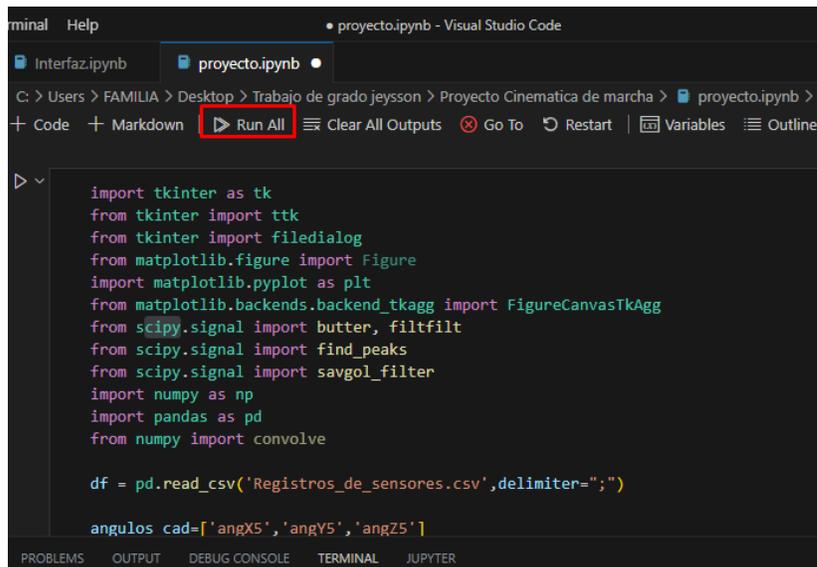
df = pd.read_csv('Registros_de_sensores.csv',delimiter=";")

angulos_cad=['angX5','angY5','angZ5']

angulos=['angX1', 'angY1', 'angZ1', 'angX2', 'angY2',
         'angZ2', 'angX3', 'angY3', 'angZ3', 'angX4',
         'angY4', 'angZ4']

class Application(tk.Frame):
    def __init__(self,master=None):
        super().__init__(master)
        self.master = master
```

Posteriormente, en esta parte de la interfaz se debe dar clic sobre Run All, el cual inicia la ejecución del programa



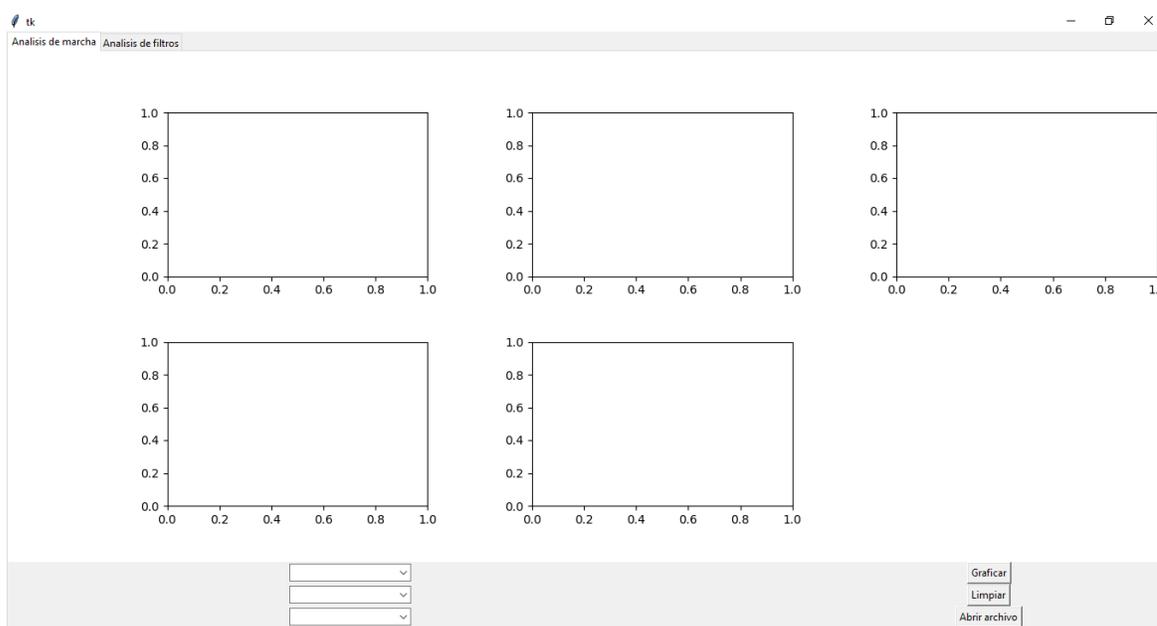
```
import tkinter as tk
from tkinter import ttk
from tkinter import filedialog
from matplotlib.figure import Figure
import matplotlib.pyplot as plt
from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg
from scipy.signal import butter, filtfilt
from scipy.signal import find_peaks
from scipy.signal import savgol_filter
import numpy as np
import pandas as pd
from numpy import convolve

df = pd.read_csv('Registros_de_sensores.csv',delimiter=";")

angulos_cad=['angX5','angY5','angZ5']
```

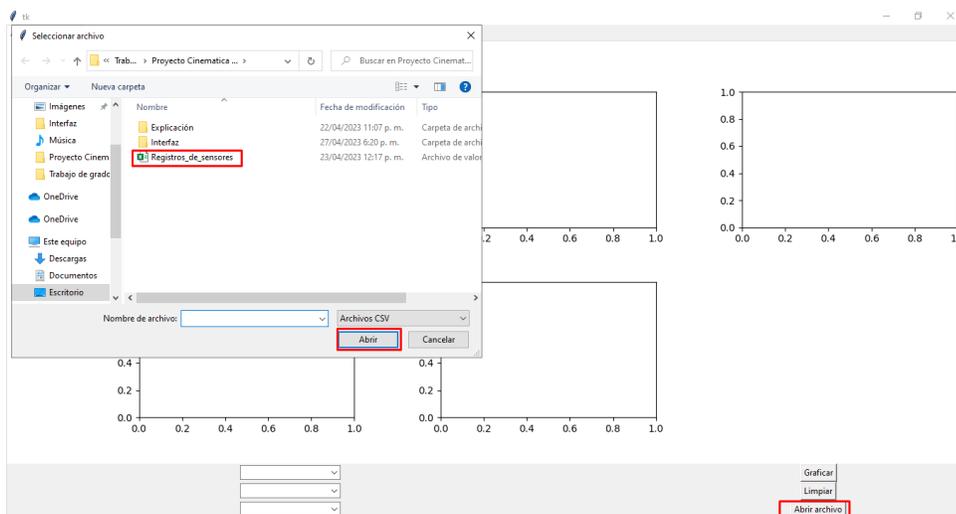
	<b>MANUAL DE USO INTERFAZ GRÁFICA</b>	<b>Estudiante</b>
		Jeysson Alfonso Calderon Contreras

Lista la ejecución, se abre una ventana en la cual se podrán visualizar dos páginas: en la primera página denominada análisis de marcha se puede encontrar cinco gráficas, las cuales van a corresponder a los sensores que se encuentren en la pierna izquierda y derecha de la persona y el ciclograma de marcha.



Adicionalmente, se pueden visualizar 3 botones; abrir archivo, en este se puede seleccionar la base de datos correspondientes que se quiera procesar. Este permite graficar los datos de los sensores inerciales que se encuentran en la base de datos y limpiar el cual limpia cada una de las gráficas que se encuentran en ellas.

	<h2 style="margin: 0;">MANUAL DE USO INTERFAZ GRÁFICA</h2>	<p><b>Estudiante</b> Jeysson Alfonso Calderon Contreras</p>
---	--	---



En la segunda página de la interfaz gráfica nombrado Análisis de Filtros, se evidencian las 15 graficas, en las cuales se tiene el procesamiento de los filtros en x, y y z para los datos ingresados.

