



UNIVERSIDAD
DE LA REPUBLICA
URUGUAY

PARKIBIP

Retroalimentación activa en la marcha de personas con
Enfermedad de Parkinson

Carlos Enrique Huerta Santana

Samuel Sainz Sanchez

Proyecto de grado de Ingeniería en Computación
Núcleo de Ingeniería Biomédica de las Facultades de Medicina e Ingeniería,
Facultad de Ingeniería
Universidad de la República

Montevideo – Uruguay

Noviembre de 2020

Versión – 1.1



UNIVERSIDAD
DE LA REPUBLICA
URUGUAY

PARKIBIP

Retroalimentación activa en la marcha de personas con
Enfermedad de Parkinson

Carlos Enrique Huerta Santana

Samuel Sainz Sanchez

Tesis de grado de Ingeniería en Computación
presentada al Programa de Ingeniería en
Computación, Núcleo de Ingeniería Biomédica
de las Facultades de Medicina e Ingeniería de la
Universidad de la República, como parte de los
requisitos necesarios para la obtención del título de
Ingeniero en Computación.

Director de tesis:

Prof. Franco Simini

Codirector:

Prof. Antonio Lopez

Montevideo – Uruguay

Noviembre de 2020

Versión – 1.1

Carlos Enrique Huerta Santana

Samuel Sainz Sanchez

PARKIBIP / Montevideo: Universidad de la República,
Núcleo de Ingeniería Biomédica de las Facultades de
Medicina e Ingeniería, Facultad de Ingeniería, 2020.

XIII, 270 p. 29, 7cm.

Director de tesis:

Franco Simini

Codirector:

Antonio Lopez

Tesis de grado de Ingeniería en Computación –
Universidad de la República, Programa de Ingeniería en
Computación, 2020.

Referencias bibliográficas: p. 197 – 207.

1. Análisis de la marcha, 2. Unidad de medición
inercial, 3. Sensores wereables, 4. Biorretroalimentación,
5. Enfermedad de Parkinson. I. Simini, Franco *et al.*
II. Universidad de la República, Programa de Posgrado en
Ingeniería en Computación. III. Título.

INTEGRANTES DEL TRIBUNAL DE DEFENSA DE TESIS DE GRADO

D.Sc. Prof. Humberto Prinzo

M.Sc. Prof. Raquel Sosa

M.Sc. Prof. Sebastián Pizard

Montevideo – Uruguay

Noviembre de 2020

Versión – 1.1

A quienes batallan día a día con
la enfermedad de Parkinson.

Agradecimientos

El presente trabajo fue realizado bajo el acompañamiento de los profesores Franco Simini y Antonio López Arredondo, pertenecientes al Núcleo de Ingeniería Biomédica de las Facultades de Medicina e Ingeniería, Universidad de la República - URUGUAY. Gracias por su amabilidad, dedicación e ideas.

Un trabajo de investigación es siempre fruto de ideas, proyectos y esfuerzos previos que corresponden a otras personas. En este caso los mas sinceros agradecimientos al D.Sc. Biomédicas Darío Santos y a la Lic. en Fisioterapia Macarena Vergara, que a su vez, formaron parte del equipo interdisciplinario del proyecto PARKIBIP.

Asimismo, se agradece al Prof. Ing. en Computación Sebastian Pizard, de la Facultad de Ingeniería, por la orientación y atención en la realización de la Revisión Sistemática Basada en Evidencia Científica.

El agradecimiento especial a los miembros de la Asociación Uruguaya de Parkinson -AUP-, encabezados por su presidente Ana Maria Martinez, por la colaboración en la investigación. Gracias por la predisposición, amabilidad y empeño en forjar una mejor calidad de vida para con los enfermos de Parkinson.

Finalmente, resta agradecer el apoyo vital que ofrecen las personas que nos estiman, sin el cual no tendríamos la fuerza y energía que nos anima a crecer como personas y como profesionales.

Los más sinceros agradecimientos.

*Our challenges don't define us.
Our actions do.*

- Michael J. Fox

RESUMEN

La alteración de la marcha es el síntoma primario de un enfermo de Parkinson, causando la pérdida de independencia en los sujetos afectados a raíz de caídas, temblores y la rigidez en el movimiento. Durante las sesiones terapéuticas los estímulos externos pueden mejorar las características de la marcha.

Este proyecto tiene como objetivo la implementación de PARKIBIP, un sistema de biorretroalimentación, que le permita al enfermo una rehabilitación personal orientada a prolongar el trabajo del terapeuta en su vida cotidiana.

Por lo tanto, se realizó una revisión sistemática basada en evidencia científica, que identifica todo el material relevante respecto al uso de dispositivos IMU para analizar la marcha de las personas. En base a la síntesis de estos resultados, se ingenió, diseñó y desarrolló un sistema portable capaz de procesar las mediciones arrojadas por los IMU en tiempo real. Así, detectar las fases de la marcha del paciente y estimularlo a través de sonido y/o vibración, según reglas clínicas pre-definidas por el equipo multidisciplinario de Ingeniería Biomédica.

Se construyó una herramienta que permite avanzar en el campo de la biomecánica hacia el entendimiento de cómo los estímulos externos favorecen la marcha de las personas con enfermedad de Parkinson. Las pruebas arrojaron excelentes resultados; en promedio se realizaron 20 sesiones, detectando la cantidad exacta de pasos para un 95 % de éstas. Además, PARKIBIP fue evaluado frente a un sistema modelo, obteniendo un error cuadrático medio (RSME) entre las velocidades instantáneas de 0.021.

En conclusión, PARKIBIP es una herramienta eficaz y de gran utilidad para el terapeuta y el paciente. Tanto para retroalimentar la marcha del sujeto afectado por la enfermedad de Parkinson de forma automática, así como también efectuar un análisis objetivo de la marcha del mismo.

Palabras claves:

Análisis de la marcha, Unidad de medición inercial, Sensores wereables, Biorretroalimentación, Enfermedad de Parkinson.

ABSTRACT

Gait disorders is the primary symptom of Parkinson's patients, causing loss of independence in affected subjects as a result of falls, tremors, and stiffness in movement. During therapeutic sessions external stimuli can improve gait characteristics.

The objective of this project is to implement PARKIBIP, an active bio-feedback system, which allows the patient a personal rehabilitation aimed at prolonging the work of the therapist in his daily life.

Therefore, a systematic review based on scientific evidence was carried out, which identifies all relevant material regarding the use of IMU devices to analyze people's gait. Based on the synthesis of these results, a portable system capable of processing the measurements produced by the IMUs in real time was devised, designed and developed. Thus, detect the phases of the patient's gait and stimulate it through sound and vibration, according to clinical rules pre-defined by the multidisciplinary Biomedical Engineering team.

A tool was built that allows progress in the field of biomechanics towards the understanding of how external stimuli favor the gait of people with Parkinson's disease. The tests yielded excellent results; On average, 20 sessions were carried out, detecting the exact number of steps for 95 % of them. Furthermore, PARKIBIP was evaluated against a model system, obtaining a mean squared error between the instantaneous speeds of 0.021.

In conclusion, PARKIBIP is an effective and useful tool for the therapist and the patient. Both to provide automatically feedback during the gait of the subject affected by Parkinson's disease, as well as to carry out an objective analysis of the patient's progress.

Keywords:

Gait analysis, Inertial measurement unit, Wearables sensors, Biofeedback, Parkinson disease.

Tabla de contenidos

1	Introducción	1
1.1	La enfermedad de Parkinson	1
1.2	Evaluación y tratamiento de la enfermedad de Parkinson	2
1.3	Oportunidades tecnológicas de mejorar el tratamiento	4
1.4	Especificación de PARKIBIP	6
1.5	Organización del documento	7
2	Fundamentos teóricos	8
2.1	Enfermedad de Parkinson	8
2.1.1	Epidemiología y Etiología	8
2.1.2	Sintomatología de la patología	11
2.1.3	Rehabilitación convencional	12
2.1.4	Nuevo paradigma de Rehabilitación	16
2.2	Sistemas clínicos de Biofeedback	18
2.3	Revisión Sistemática Basada en Evidencias	23
2.3.1	Paradigma basado en evidencias	23
2.3.2	Revisión sistemática de literatura	25
2.4	Análisis de la marcha	30
2.4.1	Identificación de fases de la marcha	31
2.4.2	Sensores de medición para análisis de la marcha	32
2.4.3	Unidad de Medición Inercial (IMU)	34
3	Revisión sistemática basada en evidencia	36
3.1	Planeamiento y diseño de Protocolo de Revisión	36
3.2	Estrategia de búsqueda	40
3.3	Criterios de inclusión/exclusión	41
3.4	Selección	42
3.5	Evaluación de calidad	43

3.6	Extracción	44
3.7	Síntesis: Resultados primarios	45
3.8	Limitaciones	59
3.9	Resultados secundarios	59
4	Proyecto: Decisiones de diseño	61
4.1	Selección tecnológica	61
4.2	Estudio de factibilidad	62
4.3	Benchmark: Elección del dispositivo IMU	63
4.4	Cantidad y ubicación del dispositivo	65
4.5	Granularidad de fases de la marcha	65
4.6	Filtro de Orientación	66
4.7	Detección de Velocidad Cero	68
4.8	Filtro de Kalman	70
4.9	Desarrollo de una aplicación móvil PARKIBIP	72
4.10	Sistema operativo y lenguaje utilizado	74
4.11	Interfaz y experiencia de usuario (UI & UX)	75
4.12	Especificación detallada y alcance del proyecto	76
5	Implementación	88
5.1	Arquitectura de PARKIBIP	88
5.2	Componentes de PARKIBIP	90
5.3	Interoperabilidad con dispositivos IMU	91
5.4	Administración de las conexiones	95
5.5	Algoritmo de empaquetado de datos PARKIBIP	100
5.6	Módulo de algoritmos numéricos y modelos de representación	101
5.7	Sistemas de coordenadas de referencias	104
5.8	Algoritmo Filtro de Orientación	106
5.9	Extracción de la aceleración física del sensor	108
5.10	Algoritmo de Detección de Velocidad Cero PARKIBIP	109
5.11	Algoritmo Filtro de Kalman PARKIBIP	114
5.12	Analizadores de datos	122
5.13	Configuración de algoritmos y parámetros	124
5.14	Módulo de Biofeedback	128
5.15	Caso de uso: Flujo automatizado de Sesión Activa PARKIBIP	131
5.16	Administración de Identidades	136

5.17	Administración de Terapeutas y Pacientes	143
5.18	Administración de Sesiones de rehabilitación	145
5.19	Diagrama de clases	147
5.20	SQLite: Almacenamiento de datos	150
6	Pruebas y resultados	153
6.1	Pruebas Estáticas: Peer Code Review	153
6.2	Pruebas funcionales: Unitarias e Integración	155
6.3	Prueba funcional: Filtro de Orientación PARKIBIP	160
6.4	Pruebas Exploratorias	162
6.5	Pruebas para-funcionales: Performance y Estrés	163
6.6	Asociación Uruguaya de Parkinson	169
6.7	Población y protocolo experimental	170
6.8	Toma de datos sin biofeedback (AUP)	171
6.9	Pruebas de PARKIBIP con biofeedback	175
6.10	Evaluación de PARKIBIP con sistema de referencia	175
6.11	Pruebas de PARKIBIP con biofeedback y referencias espacio-temporales	178
7	Gestión del proyecto	181
7.1	Actividades y cronología	181
7.2	Reporte de costos	184
7.3	Gestión de tareas	187
7.4	Administración bibliográfica	188
7.5	Gestión de archivos	190
7.6	Entorno de desarrollo	190
7.7	Versionado de código	191
7.8	Diagramas de diseño	192
7.9	Documentación	192
7.10	Otras herramientas de desarrollo	193
8	Conclusiones y trabajo futuro	194
	Referencias bibliográficas	197
	Glosario	213

Apéndices	214
Apéndice 1 PARKIBIP: Manual de usuario del sistema.	215
1.1 Instalación de PARKIBIP	215
1.2 Autenticación y autorización	216
1.3 Pantalla inicial y menú de navegación	217
1.4 Conexión de IMU	219
1.5 Sesión activa de rehabilitación PARKIBIP	222
1.6 Resumen general y detallado	223
1.7 Historial de sesiones	224
1.8 Usuario y Pacientes	225
1.9 Configuración de métodos y parámetros	226
Apéndice 2 Entidades de la base de datos relacional SQLite.	229
Apéndice 3 Pruebas funcionales: Matriz de casos de prueba	233
Apéndice 4 Pruebas Exploratorias: Tabla de sesiones exploratorias	242
Apéndice 5 Congreso de Ingeniería Biomédica SABI 2020.	248
Apéndice 6 Documento - Consentimiento informado al usuario de PARKIBIP	256
 Anexos	 258
Anexo 1 Escala Unificada de Clasificación de la Enfermedad de Par- kinson (MDS-UPDRS) Goetz et al. (2008)	259
1.1 Estado Mental, de comportamiento y estado de ánimo	259
1.2 Actividades de la vida diaria	260
1.3 Examen motor	263
1.4 Complicaciones del tratamiento (en la semana anterior)	267
1.5 Escala Schwab & England	269
1.6 Estadificación Modificada Hoehn & Yahr	270

Capítulo 1

Introducción

1.1. La enfermedad de Parkinson

La enfermedad de Parkinson (EP) es un trastorno neurodegenerativo multisistémico que afecta al sistema nervioso central (SNC) provocando la aparición de síntomas motores y no motores [Garrett \(2004\)](#). Causado por la deficiencia de un neurotransmisor, denominado DOPAMINA (DA), en el cual la pérdida progresiva de neuronas de dopamina -sito en el mesencéfalo- con la consecuente desaparición dopaminérgica de los ganglios basales, dan lugar a diversos trastornos (i.e. trastornos del movimiento y marcha, fragilidad cognitiva y emocional, alteraciones de la memoria, factores atencionales, entre otros). Es una enfermedad crónica -es decir, que persiste durante un extenso período de tiempo- y progresiva, lo que significa que sus síntomas empeoran con el tiempo [Garrett \(2004\)](#).

La alteración de la marcha es el síntoma primario del trastorno [Jankovic et al. \(2001\)](#), y junto con las caídas causa pérdida de independencia en los sujetos afectados [Ashburn et al. \(2001\)](#). Se caracteriza por deficiencias motoras, que incluyen temblor en las extremidades, aumento de la rigidez de las mismas, alteraciones de los parámetros espacio-temporales y fases de la marcha (p. ej. disminución de la longitud del paso y la velocidad de la marcha, variabilidad de zancada a zancada, ralentización del movimiento, marcha arrastrada o un inicio de marcha retrasado) [Rogers \(1996\)](#); [Stamatakis et al. \(Sept\)](#); [Hausdorff et al. \(1998\)](#); [Lord et al. \(2010\)](#). Asimismo, la interrupción paroxística de la zancada o una marcada reducción en el avance hacia adelante del pie, llamado congelación de la marcha o FOG (del inglés, Freezing of gait) es un evento

habitual, que afecta gravemente la calidad de vida, aumenta el riesgo de caídas y fracturas en los sujetos afectados [Perez-Lloret et al. \(2014\)](#); [Bloem et al. \(2004\)](#).

Al momento, es la segunda enfermedad neurodegenerativa más frecuente después de la enfermedad de Alzheimer y se estima una predominancia luego de los 60 años [Gómez-González et al. \(2019\)](#); [Chávez León et al. \(2013\)](#). En el año 2005, en las cinco naciones más pobladas de Europa occidental -Alemania, Francia, Reino Unido, Italia, y España- y las diez más pobladas del mundo -China, India, Estados Unidos, Indonesia, Brasil, Pakistan, Bangladesh, Rusia, Nigeria y Japón-, que representan 2/3 de la población mundial, el número de personas con EP mayores de 50 años estaba entre 4.1 y 4.6 millones -10 de cada 100 mil habitantes de estos países-. Se espera que este número se duplique para el 2030 [Dorsey et al. \(2007\)](#). A principios del año 2018, un estudio de prevalencia de la EP en Norte América -California, Minesota, Ontario y EE.UU.- para las personas con edad mayor o igual a los 45, estimó que existían alrededor de 680.000 individuos con la EP para el 2010 -equivalente a 477 casos cada 100.000 habitantes mayores de 45-, que ese número aumentaría a unos 930.000 aproximadamente para el 2020 -652 casos cada 100.000- y 1.238.000 para el 2030 -868 sujetos afectados cada 100.000- [Marras et al. \(2018\)](#). La tendencia creciente, destaca la importancia de optimizar el cuidado y el tratamiento para las personas con EP. Valga como ejemplo un estudio poblacional realizado en Uruguay (Migues, Canelones) entre los años 1993 y 1995 -única evidencia para Uruguay recuperada-, la prevalencia de la EP alcanzó los 136 casos cada 100.000 habitantes [R. and M. \(2012\)](#).

1.2. Evaluación y tratamiento de la enfermedad de Parkinson

En la actualidad, el tratamiento clínico de la EP se encuentra segmentado en farmacológico, quirúrgico y rehabilitador. Todas las terapias existentes para el manejo de los síntomas, basadas en el uso de fármacos dopaminérgicos, como la L-Dopa (levodopa) y agonistas de los receptores de dopamina, han demostrado una pérdida de eficacia a lo largo del tiempo. Por cierto, se han asociado con una variedad de efectos secundarios que incluyen DISCINESIA y fluctuaciones motoras, que pueden incluso empeorar las deficiencias motoras

propias de la enfermedad [Par \(2016\)](#).

De manera alternativa, para lidiar con la carencia de dopamina, la fisioterapia y la rehabilitación pueden ser efectivas para contrarrestar las deficiencias motoras.

En este sentido, la rehabilitación favorece a reinstaurar la marcha para lograr mayor autonomía del paciente, prolongar la funcionalidad, reducir la incapacidad motora y desarrollar estrategias para conservar la independencia en las ACTIVIDADES DE LA VIDA DIARIA (AVD) esenciales. La práctica por excelencia, es la aplicación de estímulos externos del tipo auditivos (i.e. metrónomos, golpes rítmicos) o visuales (i.e. líneas perpendiculares, aros marcados en el suelo) con el fin de mejorar la marcha del paciente. Aunque la evidencia científica en relación a la reeducación de la marcha mediante estímulos externos es limitada, ciertos investigadores han reportado que (i) el SISTEMA NERVIOSO CENTRAL (SNC) puede verse afectado a través de la aplicación de estímulos (i.e. táctiles, propioceptivos, auditivos, visuales, entre otros) [Moros et al. \(2000\)](#), (ii) es posible generar patrones oportunos del ciclo de la marcha [ME et al. \(1994\)](#); [Morris et al. \(1996\)](#), (iii) la propiedad adaptativa del encéfalo para con distintos contextos [Garcés Vieira and Suárez Escudero \(2014\)](#) y (iv) la recuperación de las características de la marcha y parámetros espacio-temporales mediante el entrenamiento con señales visuales y/o auditivas –de estos pacientes [Ostrosky \(2000\)](#); [Navarro et al. \(2016\)](#); [Mille et al. \(2012\)](#) [Moros et al. \(2000\)](#); [Dias et al. \(2017\)](#); [Almeida and Bhatt \(2012\)](#).

Los neurólogos solo pueden confiar en observaciones cualitativas en sesiones esporádicas, de corta duración y basadas en su experiencia. Lograr una evaluación clínica con resultados confiables puede ser muy complejo y poco práctico, en virtud de las siguientes limitaciones:

- Supervisión continua de los sujetos afectados por parte del personal médico.
- Auto-informes realizados por los mismos pacientes -afectado por el sesgo del recuerdo y probablemente sea subjetivo-.
- Presencia física de los pacientes a un entorno clínico donde se pueda obtener la evaluación -limitaciones de acceso (i.e condición del paciente, traslados), disponibilidad del médico o laboratorio-.
- Evaluación de escalas semi cuantitativa -intrínsecamente subjetivo, con

una significativa variabilidad entre los evaluadores en las puntuaciones [Parisi et al. \(2015\)](#)-.

- Personal especializado.
- Laboratorio de marcha especializado -equipo de alto costo, limitado y compartido entre patologías-.
- Mucho tiempo -difícil de lograr en sesiones clínicas-.

La cualidad progresiva de la enfermedad, conlleva a que las características motoras se vuelven cada vez más incapacitantes y las características no motoras menos tratables (por ejemplo, la demencia), acentuando la problemática de la enfermedad. Asimismo, los trastornos de la inestabilidad postural y las alteraciones de la marcha, no se traducen únicamente en alteraciones de aspectos cinemáticos como los mencionados; sino también en aspectos relacionados con su variabilidad: manifestaciones clínicas que de forma progresiva causan limitaciones en las funciones del paciente impactando directamente en su calidad de vida y restringiendo tanto su actividad como su participación social [Dillmann et al. \(2014\)](#); [Muñoz-Hellín et al. \(2013\)](#); [Fernández Del Olmo et al. \(2004\)](#); [Gómez-González et al. \(2019\)](#).

De este modo, contar con una evaluación clínica precisa y un seguimiento continuo, es fundamental para identificar una terapia efectiva; conforme a lograr una evaluación cuantitativa más objetiva de la gravedad de los síntomas de la EP, determinar con mayor precisión el grado de progreso de la enfermedad, ajustar la medicación en la atención clínica de rutina y reinstriuir la marcha en beneficio de una mejor calidad de vida para los sujetos afectados.

1.3. Oportunidades tecnológicas de mejorar el tratamiento

Los sistemas de análisis de movimiento ópticos basados en cámaras son distinguidos como el estándar de oro en la medición del movimiento [Stamatakis et al. \(Sept\)](#). Si bien son muy adecuados para medir los parámetros espacio-temporales de la marcha en términos de precisión y repetibilidad, requieren sesiones terapéuticas en un entorno de laboratorio, en compañía de personal especializado, equipo considerable y tiempo.

Los avances en los dispositivos de monitoreo WEARABLES están cambiando al sector médico desde un enfoque tradicional reactivo -atención de la salud en el nivel de crisis- a un enfoque de gestión proactiva de la salud [Castillo and Simini \(2019\)](#); [Alzugaray et al. \(2019\)](#); [Galnares et al. \(3 8\)](#); [Olivares et al. \(2019\)](#); que permite posponer, descubrir y abordar dichos problemas en una etapa temprana [Li and Guo \(2015\)](#). La implementación de sistemas basados en esta tecnología innovadora, apuntan a lograr una función terapéutica, tratando a los enfermos a través de la biorretroalimentación (del inglés, BIOFEEDBACK) [Frank et al. \(2010\)](#) [Nonnekes et al. \(2015\)](#); [Ginis et al. \(2016\)](#); [Lopez et al. \(2014\)](#); [Rochester et al. \(2010\)](#). Esto es, supervisando las funciones fisiológicas del organismo mediante la utilización de un sistema de retroalimentación, que informa al usuario del estado de la función que se desea controlar.

En el contexto de la salud, principalmente en la EP el biofeedback es fundamental, y se refiere a la provisión de estímulos externos durante o inmediatamente después de un acto motor (p. ej. cuando el sujeto levanta el pie del suelo) que brinden soporte al usuario, estimulen el compromiso cognitivo de los sujetos y potencien los efectos del ejercicio motor [Rochester et al. \(2010\)](#); [Nieuwboer et al. \(2007\)](#); [Rocha et al. \(2014\)](#); [Spaulding et al. \(2013\)](#). Combinando la naturaleza crónica y neurodegenerativa de la EP, el crecimiento continuo en avances tecnológicos wearables, y que la rehabilitación con ejercicios deben incorporarse a largo plazo en la rutina diaria para alcanzar la máxima eficacia [Tomlinson et al. \(2012\)](#); [Lamotte et al. \(2014\)](#), resulta evidente el GAP clínico y la necesidad de un sistema portable de biofeedback de la marcha en la EP. Cuyo propósito sea aumentar el control voluntario sobre los procesos fisiológicos que de otro modo estarían fuera de la conciencia y/o bajo un control menos voluntario; es decir, reeducar al paciente en rehabilitación a manipular eventos no detectados de forma voluntaria mediante el biofeedback, lograr una función terapéutica superando las limitaciones mencionadas.

La marcha de las personas es un proceso complejo y cíclico que requiere la sinergia de los músculos, los huesos y el sistema nervioso [Saunders et al. \(1953\)](#), principalmente dirigido a mantener la posición vertical y mantener el equilibrio durante condiciones estáticas y dinámicas [Ayyappa \(1997\)](#). Asimismo, implica la medición, la descripción y la evaluación de parámetros de marcha (p. ej. velocidad de la marcha, la amplitud de la zancada, la cadencia) que caracterizan la locomoción humana [Ribeiro and Santos \(2017\)](#). La correcta discriminación de las fases de marcha es considerado un punto de partida esencial para dis-

tinguir una marcha normal de una patológica o para la evaluación del estado de recuperación de la marcha durante tratamientos de recuperación [Taborri et al. \(2016\)](#).

1.4. Especificación de PARKIBIP

El presente estudio tiene como objetivo la implementación de PARKIBIP, un sistema que le permita a los pacientes con la enfermedad de Parkinson una rehabilitación personal. Durante las sesiones de fisioterapia los estímulos externos pueden mejorar las características de la marcha y PARKIBIP entonces busca emular estos estímulos en tiempo real para permitirle al enfermo una rehabilitación personal que prolongue el trabajo del fisioterapeuta en su vida cotidiana y mejore la calidad de vida de la persona afectada (i.e. ganar autonomía, estabilidad, independencia, seguridad).

La Unidad de Medición Inercial es un dispositivo que combina varios sensores como el acelerómetro, giroscopio, magnetómetro, entre otros. Estos dispositivos permiten realizar un análisis de la marcha en tiempo real y evaluación de los parámetros espacio-temporales en entornos de la vida real, interiores y exteriores, superando así las limitaciones típicas de las mediciones de laboratorio. Los IMU son más económicos y más prácticos que los sistemas GAIT ANALYSIS (GA) completos; solo se requiere una preparación relativamente rápida de los pacientes, ya que el sensor se coloca en el cuerpo por medio de una banda elástica y los datos se pueden transferir fácilmente a través de Bluetooth al software dedicado.

PARKIBIP es el resultado del trabajo de un equipo interdisciplinario a partir de la experiencia de fisioterapeutas que definen las reglas para estimular a los pacientes durante la marcha. Mediante un dispositivo electrónico wearable de bajo costo (Unidad de Medición Inercial), que combina múltiples sensores, se particiona el ciclo de marcha del usuario permitiendo identificar distintos eventos, tales como, el contacto inicial con el suelo o la fase de vuelo. Finalmente, con los eventos detectados, se aplican ciertos algoritmos matemáticos que calculan los parámetros espacio-temporales claves, y en base a un módulo clínico se estimula adecuadamente al usuario.

1.5. Organización del documento

El documento se organiza en secciones, en donde la Sección **Fundamentos teóricos** incluye la revisión de la literatura, los enfoques, las teorías o conceptos pertinentes en que se fundamenta la investigación, así como guiar al lector y ofrecer herramientas analíticas o interpretativas. Dada la complejidad del análisis, se detalla los materiales, procesos y métodos empleados para su resolución en la Sección **Proyecto: Decisiones de diseño y Revisión sistemática basada en evidencia**. Luego, se describe la solución propuesta en el apartado **Implementación**, definiendo la arquitectura y sus componentes, los algoritmos numéricos empleados, el flujo operativo del sistema PARKIBIP, entre otros conceptos relevantes. Finalmente en la Sección **Pruebas y resultados**, se presentan las pruebas de verificación del sistema, los principales hallazgos, el modo de recolección de datos y el protocolo experimental, así como las sesiones experimentales del proyecto. A modo de epílogo, se sintetizan las posturas expuestas durante el proyecto en el Apartado **Conclusiones y trabajo futuro**, se exponen las complejidades y limitaciones en el transcurso del proyecto. Asimismo, se efectúan contribuciones teóricas o metodológicas a la disciplina y recomendaciones para profundizar en el campo del biofeedback aplicado a la EP.

Capítulo 2

Fundamentos teóricos

2.1. Enfermedad de Parkinson

2.1.1. Epidemiología y Etiología

La ENFERMEDAD DE PARKINSON (EP) es un trastorno neurodegenerativo multisistémico y de presentación generalmente esporádica, que afecta al sistema nervioso central provocando la aparición de síntomas motores y no motores [Garrett \(2004\)](#). Es una enfermedad crónica -es decir, que persiste durante un extenso período de tiempo- y progresiva, lo que significa que sus síntomas empeoran con el tiempo. Su etiología desconocida conlleva a una patología aun mas compleja [García et al. \(2008\)](#); sin embargo, se cree que está relacionada con la acción e interacción de diversos factores genéticos y ambientales, que actúan como factores de susceptibilidad o precipitantes [Campdelacreu \(2014\)](#).

Fue descrita por primera vez por James Parkinson en el año 1817 [Parkinson \(2002\)](#), siendo actualmente la segunda enfermedad neurodegenerativa más frecuente después de la enfermedad de Alzheimer, y se estima una predominancia luego de los 60 años [Gómez-González et al. \(2019\)](#); [Chávez León et al. \(2013\)](#). Es fundamental evaluar y cuantificar el impacto de la enfermedad en la población, particularmente en Uruguay. De esta manera, se pudo constatar que:

- Alrededor del 3% de la población mundial mayor a los 65 años padece la enfermedad de Parkinson [Patel et al. \(2009\)](#), pudiendo alcanzar una prevalencia del 4% en pacientes mayores de 80 años [de Lau and Breteler \(2006\)](#)

- En el año 2005, en las cinco naciones de Europa occidental y las diez más pobladas del mundo, representando 2/3 de la población mundial, el número de personas mayores de 50 años con EP estaba entre 4.1 y 4.6 millones. Se espera que este número ascienda entre 8.7 y 9.3 millones de parkinsonianos para 2030 [Dorsey et al. \(2007\)](#)
- A principio del año 2018, un estudio de prevalencia de la EP en Norte América para las personas con edad mayor o igual a los 45 años, estimó que la prevalencia de la enfermedad era de 572 por 100.000 (intervalo de confianza del 95 %). Es decir, que habían unos 680.000 individuos en los EE. UU. con EP para el año 2010, que ese número aumentaría a unos 930.000 aproximadamente en 2020 y 1.238.000 para el año 2030 según las proyecciones de la población de la Oficina del Censo de los EE. UU [Marras et al. \(2018\)](#)

En cambio, no se encontró evidencia científica respecto a la prevalencia de la EP en Uruguay con datos actualizados y fiables. En el año 1993, se realizó un estudio de prevalencia de las enfermedades Neurodegenerativas donde se concluye que la prevalencia de la EP era de 1.36 casos cada 1000 habitantes [R. and M. \(2012\)](#). Sin embargo, el estudio resulta arbitrario debido a la metodología utilizada, donde en una primera fase consistió en una encuesta puerta a puerta -acotada a la ciudad de Migueles, Canelones-; adecuada para la época pero obsoleta en la actualidad. Además, como se pudo ver en los estudios de prevalencia en otras regiones, la población afectada por la EP varía año a año.

Por otro lado, el estudio reportado en [Dorsey et al. \(2018\)](#), estima la carga global de la enfermedad de Parkinson entre 1990-2016, y busca identificar tendencias asociadas al trastorno (GBL, del inglés Global Burden of Disease). Según los resultados alcanzados, en 2016, 6.1 millones de individuos en todo el mundo padecían la EP, de los cuales 2.9 millones (47.5 %) eran mujeres y 3.2 millones (52.5 %) eran hombres afectados. Un total de 2.1 millones (34.4 %) de estos individuos provenían de países con un índice socio-demográfico (SDI) alto, 3.1 millones (50.8 %) de países con SDI medio o medio alto, y un 0.9 millones (14.8 %) de países con SDI medio o bajo.

A nivel mundial, la enfermedad de Parkinson causó 211.296 muertes -93.512 mujeres y 117.784 hombres- y 3.2 millones de años perdidos debido a enfermedad, discapacidad o muerte prematura (AVAD, años de vida ajustados por discapacidad) -1.4 millones en mujeres; 1.8 millones en hombres- en 2016. Se

entiende por AVADs como la suma de los años de vida perdidos (YLL, siglas en ingles, Years of Life Lost) y los años vividos con la discapacidad (YLD, siglas en ingles, Years Lost due to Disability). La figura Fig. 2.1 permite visualizar las tasas de YLL y YLD según la edad; así, el aumento relacionado con la edad fue considerablemente más pronunciado para los YLL que para los YLD, lo que sugiere que la letalidad para la EP aumenta con la edad.

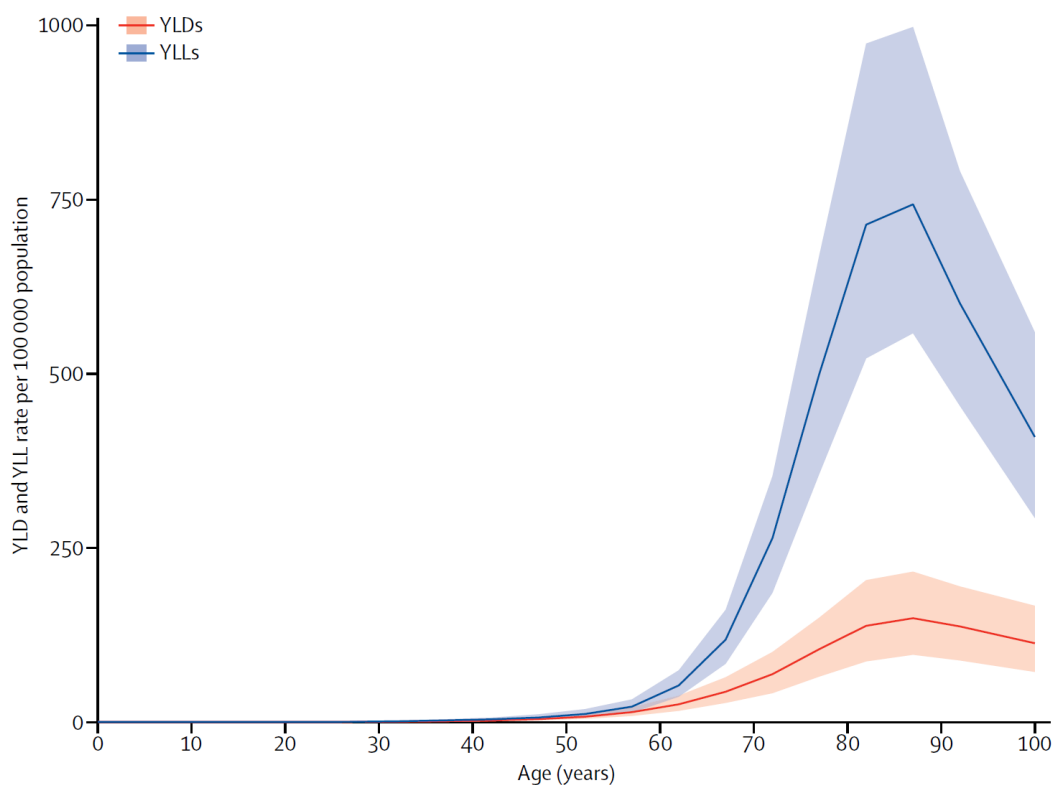


Figura 2.1: Tasas de años vividos con la discapacidad (YLD) y años de vida perdidos (YLL) cada 100.000 habitantes debido a la Enfermedad de Parkinson por edad, 2016. El sombreado indica intervalos de confianza del 95 %. [Dorsey et al. \(2018\)](#)

En resumen, algunos hallazgos logrados fueron que:

- El número de individuos con la EP en 2016 fue 2.4 veces mayor que a la de 1990. Dicho aumento no se explicó exclusivamente por un número cada vez mayor de personas adultas, ya que las tasas de prevalencia estandarizadas por edad aumentaron en un 21.7 % desde 1990 a 2016 en comparación con un aumento del 74.3 % para las tasas brutas
- El número de muertes fue 2.6 veces mayor y el número de AVADs fue 2.5 veces mayor en 2016 que en 1990. Similar a la prevalencia, los aumentos

no se explican solo por un número creciente de personas mayores, ya que las tasas estandarizadas por edad aumentaron en un 20 % tanto para las muertes como los AVADs

- La EP era poco frecuente antes de los 50 años de edad. La prevalencia en 2016 aumentó con la edad a partir de entonces y alcanzó un máximo entre los 85 y los 89 años (1.7 % para hombres; 1.2 % para mujeres) y disminuyó después de esa edad

La tabla Tab. 2.1 ilustra los indicadores de defunciones, de prevalencias y las tasas AVADs a nivel mundial y en América del sur, especialmente Uruguay.

Tabla 2.1: Defunciones, prevalencia y AVADs (años de vida ajustados por discapacidad) para la enfermedad de Parkinson en 2016, y cambio porcentual entre 1990 y 2016 en las tasas estandarizadas por edad y por ubicación. [Dorsey et al. \(2018\)](#)

	Decesos		Prevalencia		AVADs	
	En 100.000 (2016)	Tasa de cambio (1990–2016)	En 100.000 (2016)	Tasa de cambio (1990–2016)	Cantidad (2016)	Tasa de cambio (1990–2016)
Global	2,83	+19,5 %	81,2	+21,7 %	3.234.514	+22,1 %
América del sur	0,98	+3,1 %	28	+5,4 %	57.932	+4,3 %
Uruguay	8,38	+7,1 %	183,6	+10,9 %	3.775	+8,2 %

2.1.2. Sintomatología de la patología

La alteración de la marcha es el síntoma primario del trastorno [Jankovic et al. \(2001\)](#), y junto con las caídas causa pérdida de independencia en los sujetos afectados [Ashburn et al. \(2001\)](#). Se caracteriza por deficiencias motoras, que incluyen esencialmente [Rogers \(1996\)](#); [Stamatakis et al. \(Sept\)](#); [Hausdorff et al. \(1998\)](#); [Lord et al. \(2010\)](#):

- Temblor en las extremidades
- Aumento de la rigidez de las mismas
- Alteraciones de los parámetros espacio-temporales y fases de la marcha (p. ej. disminución de la longitud del paso y la velocidad de la marcha, variabilidad de zancada a zancada, fases de marcha)

- Ralentización del movimiento, marcha arrastrada o un inicio de marcha retrasado

Otro evento frecuente, es la congelación de la marcha (FOG, del inglés, Freezing of gait), dado por una interrupción paroxística de la zancada o una marcada reducción en el avance hacia adelante del pie, que afecta gravemente la calidad de vida, aumenta el riesgo de caídas y fracturas en los sujetos afectados [Perez-Lloret et al. \(2014\)](#); [Bloem et al. \(2004\)](#). En pacientes con EP, la investigación fisiopatológica de la FOG es bastante desafiante ya que se encuentra influenciada de manera crucial por una serie de factores cognitivos, atencionales, emocionales e incluso ecológicos [Nieuwboer and Giladi \(2008\)](#); [Barthel et al. \(2016\)](#).

La cualidad progresiva de la enfermedad, conlleva a que las características motoras se vuelven cada vez más incapacitantes y las características no motoras menos tratables (p. ej. la demencia), acentuando la problemática de la enfermedad. Asimismo, los trastornos de la inestabilidad postural y las alteraciones de la marcha, no se traducen únicamente en alteraciones de aspectos cinemáticos como los mencionados; sino también en aspectos relacionados con su variabilidad, manifestaciones clínicas que de forma progresiva causan limitaciones en las funciones del paciente que impactan directamente en su calidad de vida y restringiendo tanto su actividad como su participación social [Dillmann et al. \(2014\)](#); [Muñoz-Hellín et al. \(2013\)](#); [Fernández Del Olmo et al. \(2004\)](#); [Gómez-González et al. \(2019\)](#).

2.1.3. Rehabilitación convencional

En la actualidad, todas las terapias existentes para el manejo de los síntomas basadas en el uso de medicamentos dopaminérgicos, como la L-Dopa (levodopa) y agonistas de los receptores de DOPAMINA, han demostrado una pérdida de eficacia a lo largo del tiempo. A su vez, se han asociado con una variedad de efectos secundarios que incluyen DISCINESIA y fluctuaciones motoras, que pueden incluso empeorar las deficiencias motoras propias de la enfermedad.

Con el fin de lograr una evaluación cuantitativa más objetiva de la gravedad de los síntomas de la EP, determinar con mayor precisión el grado de progreso de la enfermedad, ajustar la medicación en la atención clínica de rutina o para su uso como medidas de resultados en ensayos clínicos, los neurólogos se apoyan en las escalas de evaluación semi cuantitativa, como lo es la Escala

Unificada de Clasificación de la Enfermedad de Parkinson -elaborada por la Sociedad de Trastornos del Movimiento- (MDS-UPDRS, por sus siglas en inglés) [Jellinger \(2002\)](#). La escala MDS-UPDRS consiste en diversas partes, que buscan evaluar los síntomas motores y no motores a través de encuestas respondidas por el paciente, y evaluaciones basadas en exámenes realizadas por un médico capacitado. La principal limitación de esta evaluación es ser intrínsecamente subjetiva, con una significativa variabilidad entre los evaluadores en las puntuaciones [Parisi et al. \(2015\)](#). Además, es una tarea que requiere mucho tiempo -difícil de lograr en sesiones clínicas-, y a la vez la presencia física de los pacientes en un entorno clínico donde se pueda obtener la evaluación. El Apéndice [Escala Unificada de Clasificación de la Enfermedad de Parkinson \(MDS-UPDRS\) Goetz et al. \(2008\)](#) permite visualizar la evaluación clínica mediante la escala MDS-UPDRS.

Fisioterapia y rehabilitación

A medida que la enfermedad progresa, la mayoría de las alteraciones del equilibrio y la marcha se vuelven resistentes a los tratamientos farmacológicos y quirúrgicos [Rubinstein et al. \(2002\)](#). En consecuencia, para sortear el déficit de dopamina, la fisioterapia y la rehabilitación (i.e. estiramiento, fortalecimiento muscular, movimientos de las extremidades con pelotas y materiales de juego, acondicionamiento en bicicleta) pueden ser efectivas para contrarrestar las deficiencias motoras [Fox et al. \(2008\)](#).

La finalidad de la terapia de rehabilitación, es mejorar o mantener la calidad de vida de los sujetos afectados conforme a aumentar la movilidad, mejorar el equilibrio, la coordinación, reeducar la marcha para lograr la autonomía del paciente. Si bien ésta terapia no detiene la progresión de la enfermedad, habilita a prolongar la funcionalidad y reducir la incapacidad motora perfeccionando la marcha, las destrezas propioceptivas y desarrollando estrategias para conservar la independencia en las ACTIVIDADES DE LA VIDA DIARIA (AVD) elementales.

De esta manera, la rehabilitación debe atender al paciente según sus necesidades, su discapacidad, el estadio evolutivo de la enfermedad y las expectativas del paciente -similar a todas las enfermedades crónicas y progresivas-. A modo de ejemplo, se describen las fases y actividades del tratamiento de acuerdo a su evolución y al protocolo clínico a seguir [Ministerio de S. de Salud Pública](#)

(2010):

1. Fase Inicial

- Prevenir y tratar la inestabilidad postural.
- Evaluar e identificar tempranamente los problemas relacionados con alteraciones del movimiento.
- Fomentar la participación del paciente con EP en programas diseñados para mejorar la condición física general. (Cardiovascular, músculo esquelética, y neuromuscular).
- Prevenir las deficiencias posturales, deficiencias en marcha y traslado.
- Tratar la debilidad muscular y/o la rigidez articular.
- Utilizar estrategias de movimientos a lo largo de la enfermedad.
- Informar y Educar en salud, acerca de la enfermedad a los pacientes, familiares y cuidadores.

2. Fase de Mantenimiento

- Tratar el deterioro músculo esquelético.
- Reeducar marcha y traslados. Prevención de caídas.
- Evaluar el medio en donde se desenvuelve el paciente (entrenamiento en el hogar y comunidad).
- Aplicar estrategias funcionales.
- Evaluar la prescripción de AT e instruir en su uso adecuado.

3. Fase Avanzada o tardía

- Asegurar el manejo adecuado de movimiento y posición.
- Evitar las caídas.
- Prevenir lesiones trofocirculatorias y posturales.
- Prevenir y tratar los problemas respiratorios.

Otra metodología muy empleada en la fisioterapia y rehabilitación, es la aplicación de incentivos o guías externas del tipo auditivo (i.e. a modo de

metrónomos o golpes rítmicos en el suelo) o visual (i.e. huellas o líneas perpendiculares marcadas en el suelo) para mejorar los trastornos de la marcha; es decir, REEDUCAR LA MARCHA. Si bien la evidencia científica existente respecto a la reeducación de la marcha es reducida, muchas investigaciones han concluido y reportado los siguientes resultados:

- Viabilidad de generación de patrones adecuados de la marcha. Según [ME et al. \(1994\)](#); [Morris et al. \(1996\)](#), es posible que los parkinsonianos en presencia de estimulación sensorial regular, generen patrones adecuados de la marcha, corroborando una mejoría en los parámetros espacio-temporales bajo el uso de estímulos externos.
- Factibilidad de incidencia sobre el SISTEMA NERVIOSO CENTRAL (SNC). Con fundamento en el principio de que todo acto motor es una elaboración del SNC -en respuesta a múltiple información sensitivo motora, simultánea y secuencial-, en efecto, se concluye que el segundo puede verse afectado o modificado positivamente mediante diversos estímulos táctiles, propioceptivos, auditivos, visuales, entre otros [Moros et al. \(2000\)](#).
- Capacidad adaptativa del encéfalo. Existe evidencia científica que comprueban las propiedades de flexibilidad y modificabilidad del cerebro en distintos contextos -durante la infancia y la adolescencia, durante la edad adulta, como también en situaciones de lesión cerebral-. Así, el encéfalo puede alterarse para adaptarse a diversas circunstancias [Garcés Vieira and Suárez Escudero \(2014\)](#).
- Estimulación visual sobre la marcha. El Neurólogo Británico James Purdon Martin, logró demostrar el efecto positivo de las señales visuales sobre la marcha en la EP en el año 1967. Comprobó cómo tan solo una serie de líneas de color dispuestas en el suelo -de forma perpendicular a la dirección del movimiento-, mejoraban las características de la marcha de estos pacientes [Ostrosky \(2000\)](#); [Navarro et al. \(2016\)](#); [Mille et al. \(2012\)](#); [Loyola \(2017\)](#).
- Resultados alentadores. Sidaway -entrenamiento con señales visuales- [Moros et al. \(2000\)](#), logro una mejoría duradera en la velocidad de la marcha y en la longitud del paso incrementando la estabilidad del control motor subyacente responsable de la marcha. La investigación de [Dias](#)

[et al. \(2017\)](#) -señales visuales y terapia convencional-, aumentó la velocidad, la longitud de zancada y la cadencia al caminar, mejoró el equilibrio y la independencia en las actividades funcionales. Las sesiones experimentales en [Almeida and Bhatt \(2012\)](#), lograron mejoras significativas de la longitud del paso, mejorando así la velocidad de la marcha -un incremento aproximado de 10 cm/s).

Conforme a lo mencionado, lograr una evaluación clínica con resultados confiables puede ser muy complejo y poco práctico, ya que requiere una supervisión continua de los sujetos afectados por parte del personal médico, o auto-informes realizados por los mismos pacientes (lo que probablemente sea subjetivo). El aumento de las disfunciones motoras a medida que avanza la enfermedad requiere evaluar cuantitativamente y controlar las alteraciones de la marcha con el tiempo. Asimismo, los neurólogos solo pueden confiar en observaciones cualitativas en sesiones esporádicas y basadas en su experiencia. De este modo, contar con una evaluación clínica precisa de la gravedad de estos síntomas y un seguimiento continuo (p. ej. un instrumento que actuara directamente sobre el desempeño de la marcha sería de utilidad mayor), es fundamental para identificar una terapia efectiva.

2.1.4. Nuevo paradigma de Rehabilitación

Con fundamento en las limitaciones clínicas mencionadas en el tratamiento convencional del trastorno para identificar, evaluar y monitorear la enfermedad de parkinson, la Ingeniería Biomédica ha realizado esfuerzos para desarrollar métodos objetivos y precisos utilizando sistemas de medición de movimiento. En general, el análisis instrumentado de la marcha (GAIT ANALYSIS (GA), por sus siglas en inglés) se emplea con frecuencia para obtener parámetros cinemáticos, cinéticos y espacio-temporales, y en efecto, lograr una imagen cuantitativa de la función de la marcha. Estos parámetros espacio-temporales son ampliamente utilizados en el contexto clínico, ya que describen objetivamente los principales eventos del ciclo de la marcha y reflejan la capacidad del paciente para cumplir una marcha adecuada (i.e la aceptación del peso, el soporte de una sola extremidad, la fase de vuelo de una extremidad ([Bugané et al. \(2012\)](#))). Específicamente, diversas investigaciones han reportado que las principales alteraciones en el GA para la EP son:

- Reducción de la longitud de la zancada, a menudo acompañada de una velocidad de caminata más baja [Sande de souza et al. \(2011\)](#).
- Intento de extender la fase de doble apoyo [Sofuwa et al. \(2005\)](#).
- Ausencia de “golpe de talón”, debido al típico soporte de pie plano [Sofuwa et al. \(2005\)](#).
- Problemas para cambiar de dirección durante el giro [Morris et al. \(2001\)](#); [Nutt et al. \(2011\)](#). Girar puede ser incluso más frecuente que caminar hacia adelante, especialmente en entornos confinados como las viviendas.
- Dificultad para iniciar la marcha, acortamiento de la longitud del paso, poca elevación de los pies del suelo con su consecuente arrastre.
- Variabilidad entre los pasos. Esta variación en la duración del ciclo es una característica importante de la marcha parkinsoniana y se considera un indicador del riesgo de sufrir caídas.

En estas condiciones, los sistemas instrumentados de análisis de la marcha son fundamentales. Los sistemas de análisis de movimiento ópticos basados en cámaras (3D-GA) son reconocidos como el estándar de oro en la medición del movimiento [Stamatakis et al. \(Sept\)](#). Por ejemplo, se utilizan para medir los parámetros de la marcha con el fin de cuantificar la gravedad de la EP [Roiz et al. \(2010\)](#), y así proporcionar información esencial sobre el estado funcional de un sujeto. Estos sistemas son muy adecuados para medir las características de la marcha en términos de precisión y repetibilidad, sin embargo, requieren ejecutar las pruebas en un entorno de laboratorio con equipos de alto costo, solo se pueden usar para evaluar segmentos cortos de la marcha, requiere personal especializado, equipo considerable y tiempo.

Análogamente, los recientes avances en la tecnología wearable, han llevado al desarrollo de dispositivos de medición que pueden evaluar el movimiento humano utilizando sensores conectados al cuerpo con una variedad de configuraciones diferentes (i.e. frecuencia cardiaca, acelerómetro, giroscopio). Los parámetros de la marcha medidos bajo estos instrumentos, son indicadores útiles para caracterizar la EP, así como para cuantificar la gravedad de la enfermedad en los sujetos [Dewey et al. \(2014\)](#).

Teniendo en cuenta la naturaleza crónica y neurodegenerativa de la EP, e integrando el crecimiento continuo en avances tecnológicos wearables, y que la

rehabilitación y la terapia con ejercicios deben incorporarse a largo plazo en la rutina diaria para alcanzar la máxima eficacia [Tomlinson et al. \(2012\)](#); [Lamotte et al. \(2014\)](#), comprueban la existencia de un importante GAP clínico. Combinar los sistemas instrumentados de análisis de la marcha con la terapia de biorretroalimentación, para construir un sistema final de biofeedback portable -uso cotidiano y en exteriores-, accesible y de bajo costo; cuyo objetivo sea aumentar el control voluntario sobre los procesos fisiológicos que de otro modo estarían fuera de la conciencia y/o bajo un control menos voluntario. Es decir, reeducar al paciente en rehabilitación a manipular eventos no detectados de forma voluntaria mediante la retroalimentación, lograr una función terapéutica superando las limitaciones mencionadas en [2.1.4](#).

2.2. Sistemas clínicos de Biofeedback

El objetivo del presente apartado es transmitir las tendencias y la evolución en el desarrollo de la biorretroalimentación (del inglés, BIOFEEDBACK) aplicada en la salud. Brindar al lector una perspectiva histórica que habilite a comprender los orígenes multifacéticos y la importancia de el biofeedback en el tratamiento de diversos trastornos.

La bioretroalimentación y su aplicación clínica poseen una larga historia. Si bien en la actualidad el concepto ha adquirido gran relevancia -fruto de los avances tecnológicos-, inició en los Estados Unidos a principio de la década de 1950 a partir de varias disciplinas clínicas y científicas, y continúa evolucionando [Mark Stephen and Frank \(2016\)](#). Dentro de los principales antecedentes y disciplinas por los cuales fue desarrollada se encuentran los siguientes:

- Condicionamiento instrumental de las respuestas del (SISTEMA NERVIOSO AUTÓNOMO (ANS))
- Psicofisiología
- Terapia y medicina conductual
- Investigación y estrategias de manejo del estrés
- Ingeniería biomédica
- Electromiografía de superficie (EMG), diagnóstico EMG, y control de unidades de un solo motor

- Conciencia, estados de alteración de la conciencia y la encefalografía (biofeedback de EEG, también llamada neuroretroalimentación)

Las teorías del aprendizaje, por ejemplo, han proporcionado modelos teóricos y evidencia científica de que las respuestas del sistema nervioso autónomo (ANS) pueden ser condicionadas de forma instrumental u operacional. En otro caso, los terapeutas conductuales han planteado los principios de aplicación de modelos de condicionamiento operantes y clásicos, así como modelos de aprendizaje observacional y modelos cognitivos (procesamiento de información). El área de ingeniería biomédica ha implementado instrumentos para la supervisión en tiempo real de la actividad fisiológica. A través de la tecnología, puede realizarse un procesamiento y posterior análisis de señales con gran rapidez, grabaciones de múltiples canales y pantallas fáciles de usar. Según [Mark Stephen and Associates \(1998\)](#), no existiría el biofeedback sin una instrumentación de alta calidad que mida de manera precisa y confiable los eventos fisiológicos. En este sentido, el biofeedback se ha convertido en una terapia fundamental en el campo de la medicina, ha demostrado tener diversas aplicaciones, como el manejo del estrés, la terapia de relajación, el manejo del dolor, entre otras aplicaciones que introduciremos. Establecer una definición integral y clara sobre el biofeedback aplicado, que contemple su proceso, alcance y objetivos, será una cuestión central para un mejor entendimiento y soporte a la propuesta de trabajo. Aunque existen diversas definiciones, se hará hincapié en el biofeedback desde el punto de vista de proceso y en la definición oficial acordada por la ASSOCIATION FOR APPLIED PSYCHOPHYSIOLOGY AND BIOFEEDBACK (AAPB):

“Como proceso, la biorretroalimentación es un grupo de procedimientos terapéuticos que utiliza instrumentos electrónicos y electromecánicos para medir, procesar y retroalimentar con precisión, a las personas y a sus terapeutas, información con propiedades educativas y de refuerzo, sobre sus actividades neuromusculares y autónomas, tanto normales como anormales, en forma de señales de retroalimentación analógica o binaria, auditiva y/o visual. Lo mejor se logra con un profesional de biofeedback competente, los objetivos son ayudar a las personas a desarrollar una mayor conciencia, confianza y un aumento en el control voluntario sobre los procesos fisiológicos que de otro modo estarían fuera de la conciencia y/o bajo un control menos voluntario, controlando primero la señal externa, y luego mediante el uso de cogniciones, sensaciones u otras señales para prevenir, detener o reducir los síntomas.” [Schwartz and Olson \(1995\)](#) .

“La biorretroalimentación es un proceso que le permite a un individuo aprender como cambiar la actividad fisiológica con el fin de mejorar la salud y el rendimiento. Los instrumentos precisos miden la actividad fisiológica, como las ondas cerebrales, la función cardíaca, la respiración, la actividad muscular y la temperatura de la piel. Estos instrumentos de forma rápida y precisa “retroalimentan” información al usuario. La presentación de esta información, a menudo junto a cambios de pensamiento, de comportamiento y emocionales, brindan soporte a los cambios fisiológicos deseados. Con el tiempo, estos cambios pueden perdurar sin el uso continuo de un instrumento.” [Oficial, AAPB\(2008\)](#)

En resumen, cuando los pacientes reciben visualizaciones electrónicas en tiempo real de sus eventos fisiológicos internos (i.e. usando sensores de medición, barras de luces, entre otros), se les puede educar a manipular eventos no detectados de forma voluntaria mediante la retroalimentación. Dependiendo del contexto de aplicación, ésta retroalimentación se puede dar de diferentes formas, por ejemplo, estímulos externos o internos al paciente.

El crecimiento continuo en avances tecnológicos y su combinación, como lo son los dispositivos WEARABLES, los teléfonos inteligentes con gran capacidad de computo, los sistemas IoT o de AI; son un campo de investigación emergente para la supervisión y entrenamiento de la salud en el sector médico. Estos sistemas, compuestos por múltiples tipos de sensores de medición pequeños

y de bajo costo, que interoperan con servicios clínicos de fondo, prometen cambiar el futuro de la atención médica de los pacientes y a permitir una rehabilitación personal.

En la actualidad, los contextos de aplicación de el biofeedback son amplios y diversos. La terapia con biofeedback en electromiografía (EMG) se ha utilizado en lesiones de la motoneurona superior -encargada de enviar las señales a toda los músculos de nuestro cuerpo para que estos se muevan- (i.e. hemiplejia debido a accidente cerebro-vascular o función muscular espástico en la parálisis cerebral), lesiones en la motoneurona inferior (i.e. parálisis facial de Bell), parálisis histérica y discinesias (i.e. tortícolis espasmódica y la enfermedad de Parkinson). Otra aplicación es lo que se denomina electroencefalografía, utilizada para la manipulación consciente de las ondas cerebrales en pacientes con ataques epilépticos. También en dolores de cabeza vasculares, a través de la temperatura de la piel y su relación con el flujo sanguíneo. A nivel cardiovascular e hipertensión, fue aplicada la pletismografía directa -mide los cambios de volumen en diferentes áreas de su cuerpo- [Basmajian \(1977\)](#). Por lo tanto, resulta evidente que el biofeedback tiene diferentes facetas en base a la misma idea; detectar con precisión un evento fisiológico y convertir la señal electrónica resultante en retroalimentación auditiva, visual, kinestésica u otra estímulo alternativo.

Con la finalidad de clarificar lo que antecede e introducir el problema en cuestión, se menciona brevemente una aplicación clásica de la ingeniería biomédica y el biofeedback.

Aplicación: Andador robótico multifunción.

Las alteraciones ortopédicas y neuromusculares impactan negativamente en la calidad de vida de las personas con discapacidad, resultando en limitaciones de movimiento, fuerza y control de la movilidad. Para muchas personas mayores o simplemente con debilitamiento de la marcha, el desempeño de actividades de la vida diaria (AVD) (i.e. estar de pie, agacharse o caminar) están restringidas hasta un cierto punto y pueden ser difíciles de lograr [Conger \(2011\)](#). El andador es un dispositivo ortopédico orientado a ganar autonomía, estabilidad, independencia y seguridad a la persona que tiene restringida la movilidad. Sin embargo, un andador convencional consiste en una estructura de metal pasiva con tres o mas puntos de contacto (férula de goma como con

bastones, ruedas o combinación de ambas) en el suelo, el cual el usuario coloca delante y ejerce fuerzas de movimiento para desplazarse. Esta asistencia pasiva para caminar es difícil y hasta imposibles de operar en terrenos irregulares -i.e. superficies de césped o adoquines-.

De esta manera, la ingeniería biomédica y los sistemas de biofeedback son utilizados en la navegación robótica como solución a los desafíos mencionados [Molano et al. \(2019\)](#); [Wang et al. \(2018\)](#). Las interfaces de control de usuario indirectas, reconocen el movimiento o la intención del usuario mediante el uso de sensores o sistemas de reconocimiento de imagen; y posteriormente, un modulo de transmisión de fuerza (motor) es responsable de planificar y ejecutar el movimiento del robot. La ilustración 2.2, permite visualizar el caso del andador robótico multifunción.

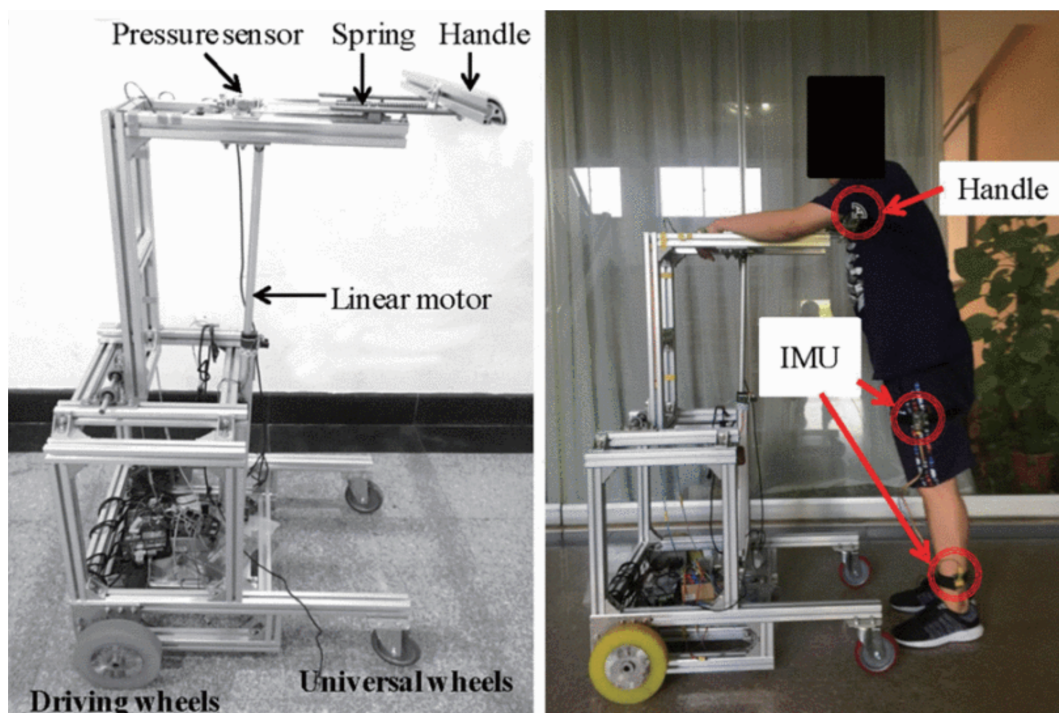


Figura 2.2: Andador robótico multifunción guiado por sensores de medición inercial y un modulo de transmisión de fuerza. [Wang et al. \(2018\)](#)

Un modulo de interfaz de usuario supervisa la progresión lineal en un grado de libertad utilizando sensores de medición. En base al principio bípedo, calcula el ángulo dinámico necesario para reflejar el movimiento cinemático, y así determinar las señales de control que serán enviadas a los controladores del motor. Finalmente, las señales recibidas son colocadas en un ciclo de retroalimentación que determinan los comandos del motor para controlar el andador

robótico multifunción [Modise et al. \(2016\)](#).

2.3. Revisión Sistemática Basada en Evidencias

Esta sección aborda el paradigma basado en evidencias, actualmente adoptado en muchas ciencias prácticas (i.e. medicina, educación, política social e ingeniería). Las REVISIONES SISTEMÁTICAS DE LA LITERATURA (SLR) son un aspecto esencial del paradigma basado en evidencias, y tienen como objetivo sintetizar los estudios de investigación asociados a una pregunta de investigación específica, de forma rigurosa, auditable y justa.

La presentación en cada subsección, describe el beneficio y el potencial de las revisiones sistemáticas de la literatura, razón por la cual será la metodología de investigación adoptada durante el proyecto PARKIBIP. Asimismo, procura especificar el flujo de actividades estándar a realizar en una SLR.

2.3.1. Paradigma basado en evidencias

Para lograr un mayor entendimiento se considera apropiado iniciar por el significado de EVIDENCIA en un marco científico y tecnológico, ya que será aplicado a prácticas de ingeniería de software en la realización de PARKIBIP.

En general, la evidencia se asocia al conocimiento. Normalmente se espera que el conocimiento se derive de la evidencia a través de algún proceso de interpretación (i.e. un modelo matemático o estadístico), y no basado en ilusiones. Por ejemplo, una ilustración menos científica ocurre cuando un jurado, en un juicio penal, tiene que considerar la evidencia de un conjunto de testigos para obtener un conocimiento razonable sobre lo sucedido. Asimismo, analizar dicho proceso por el cual se interpreta el primero para agregar o inferir el segundo es de suma importancia. Se detectan dos parámetros esenciales para producir evidencia sólida a partir de un proceso de revisión objetiva:

- Selección objetiva e imparcial de estudios relevantes

- Síntesis sistematizada de los resultados de los estudios previamente seleccionados

En resumen, la practica basada en evidencias (EBP, por sus siglas en ingles) busca utilizar un enfoque objetivo, riguroso y planificado para seleccionar estudios relevantes y realizar una sintesis de los resultados de esos estudios [Kitchenham \(2006\)](#). La rigurosidad metodológica hace que los resultados sean mas confiables, ya que es posible estudiar el procedimiento llevado a cabo para su obtención asi como también reproducirlo.

La ingeniería de software basada en evidencias realiza un fuerte énfasis en el rigor metodológico involucrando los siguientes cinco pasos [Kitchenham et al. \(2004\)](#); [Sebastián et al. \(2019\)](#):

1. Convertir un problema relevante o una necesidad de información en una pregunta que pueda ser respondida.
2. Buscar en la literatura la mejor evidencia para responder a esa pregunta.
3. Evaluar de forma critica la validez, el impacto y la aplicación de la evidencia.
4. Integrar la evidencia evaluada con la experiencia practica y los valores y circunstancias de los interesados.
5. Evaluar la efectividad y la eficiencia de este proceso para buscar maneras de mejorarlo.

Los primeros tres pasos son esencialmente el papel de la revisión sistemática de literatura, mientras que el cuarto, es el de la traducción del conocimiento. El quinto es garantizar que los procedimientos de investigación estén sujetos a un análisis constante.

Para brindar una definición concisa a la técnica mencionada, es necesario clasificar las investigaciones en estudios:

- **Primarios:** métodos utilizados para realizar estudios con el objetivo de obtener evidencia empírica sobre un tema de interés (i.e. encuestas, experimentos, estudios de caso, investigación y acción - del ingles, action research-).
- **Secundarios:** métodos que permiten recopilar de manera sistemática y rigurosa los estudios primarios relacionados con una pregunta de investigación específica, con el objetivo de sintetizar la evidencia disponible para responder a dicha pregunta (i.e. revisión sistemática de literatura).

De esta manera, una revisión sistemática de literatura o SLR es un estudio secundario generado a partir de estudios primarios, y se define como un método para identificar, evaluar e interpretar todas las investigaciones pertinentes a una determinada pregunta de investigación, área, temática o fenómeno de interés. Es la principal herramienta del paradigma basado en evidencias, siendo sistemáticas -porque se conducen siguiendo un conjunto de procedimientos bien delineados y reproducibles-, además tratan de evitar sesgos, son rigurosas y abiertas.

La ilustración Fig. 2.3 permite resumir mediante un esquema macro el flujo de información entrante y saliente relativo a una SLR; así como también el impacto futuro en políticas, estándares y prácticas.

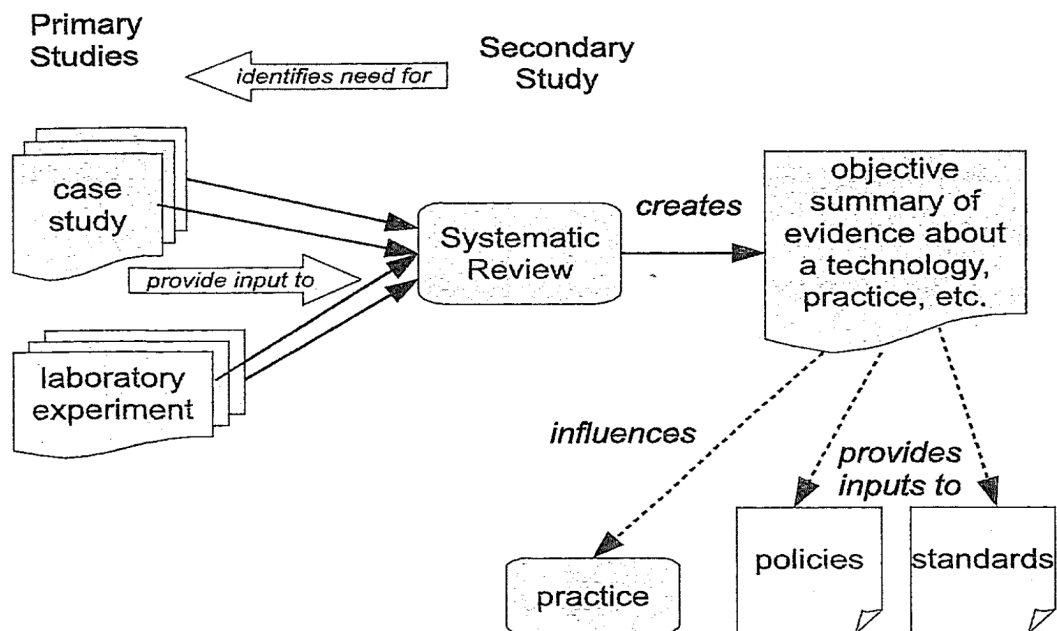


Figura 2.3: Contexto de una revisión sistemática de literatura Kitchenham (2006).

2.3.2. Revisión sistemática de literatura

En general, los investigadores ejecutan las revisiones para resumir la evidencia científica existente sobre algún fenómeno particular, de manera exhaustiva e imparcial. Según una encuesta reciente, los principales factores de motivación en la ingeniería de software son:

- Reunir conocimiento respecto a un campo o estudio en particular.
- Identificar recomendaciones para investigaciones futuras.

- Establecer el contexto de un tema o problema de estudio.
- Identificar las principales metodologías y técnicas de investigación usadas en un campo de investigación particular.

Realizar una revisión sistemática es un proceso extremadamente lento, que requiere suma atención en los detalles. Siempre es bueno comenzar por establecer la necesidad (i.e. ya existe otra que aborde el mismo tema) y por ende la motivación de ejecutar una revisión, para luego evaluar su factibilidad.

Asimismo, especificar los aspectos técnicos del proceso de revisión (planeamiento) es un factor fundamental para lograr resultados satisfactorios. Esto es, desarrollar un protocolo de revisión que permita realizar y documentar las estrategias y toma de decisiones de diseño del estudio. Su definición de antemano puede ayudar a reducir la probabilidad de sesgo, limitando la influencia de las expectativas del investigador (i.e. la selección de estudios primarios o la síntesis de los resultados). A modo de ejemplo se propone la Tabla 2.2, que sugiere los componentes a incluir en un protocolo de revisión.

Tabla 2.2: Componentes de un protocolo de revisión

Antecedentes y justificación de la revisión.
Preguntas de investigación.
Estrategia de búsqueda para estudios primarios. Debe incluir los términos de búsqueda y recursos donde se realizará la búsqueda. Los recursos incluyen librerías digitales, revistas específicas, y actas de congresos.
Criterios de selección de estudios. Usados para determinar cuáles estudios son incluidos o excluidos de la revisión sistemática.
Procedimientos de selección de estudios. El protocolo debe describir cómo se aplicará el criterio de selección, por ejemplo, cuántos asesores evaluarán cada estudio primario, y cómo se resolverán los desacuerdos entre los evaluadores.
Procedimientos y listas de verificación para evaluar la calidad.
Estrategia de extracción de datos.
Método o técnica de síntesis de los datos.
Limitaciones.
Estrategia de difusión.
Calendario del proyecto de revisión.

Dentro de la fase de planeamiento, otro ítem sumamente determinante, es establecer las preguntas de investigación que guiarán e incidirán en todo el estudio

(i.e. que estudios primarios incluir o excluir, decidir que datos deben extraerse y cómo sintetizar los datos para contestar las preguntas). Una pregunta es adecuada cuando [Kitchenham et al. \(2004\)](#):

- Es significativa e importante tanto para investigadores como para profesionales.
- Dará lugar a cambios o incrementará la confianza en las practicas actuales de la ingeniería de software.
- Identificará las discrepancias entre las creencias comúnmente aceptadas versus la realidad.

Posterior a la etapa de diseño o planeamiento, inicia la fase de ejecución de la revisión sistemática. Si bien no será objeto de aplicación en el proyecto PARKIBIP, y por lo tanto no será analizada con detenimiento, es importante mencionar la última etapa de Documentación y publicación del estudio. El esquema en la figura Fig. 2.4 expone un patrón organizado y sistemático a seguir en la realización de una SLR. De igual manera, incluye la clasificación en fases y el protocolo de revisión mencionados anteriormente.

En resumen, una SLR tiene ciertas etapas discretas contempladas en tres fases principales, cada una con sus actividades particulares:

1. Planificar la Revisión

- Especificar las preguntas de investigación.
- Desarrollar el protocolo de revisión.
- Evaluar el protocolo de revisión (recomendada).

2. Realizar la Revisión

- Identificar la investigación.
- Seleccionar los estudios primarios.
- Evaluar la calidad de los estudios.
- Extraer datos y monitorear.
- Sintetizar los datos.

3. Informar la Revisión

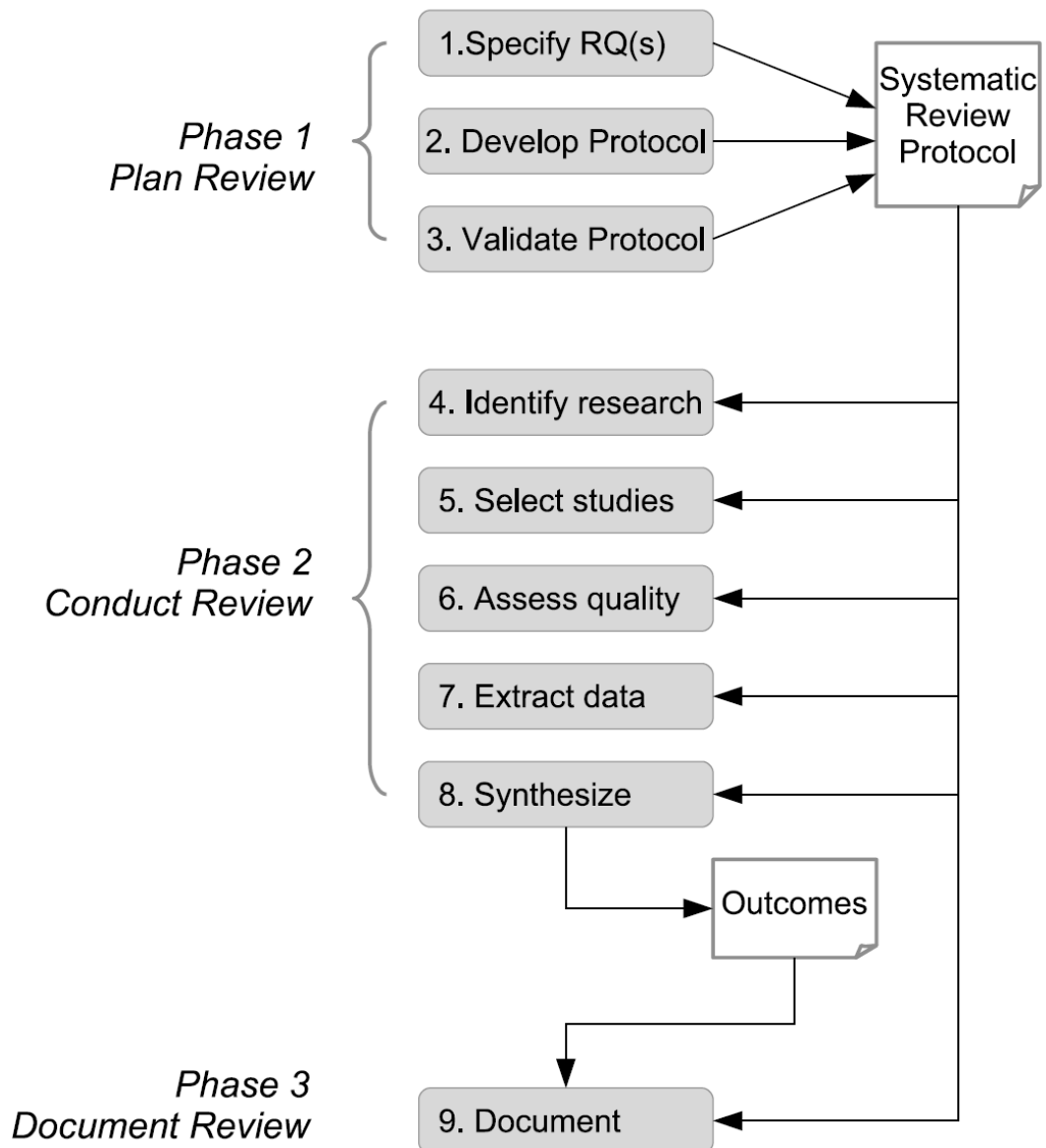


Figura 2.4: Flujo de actividades en el proceso de una SLR. [Kitchenham \(2006\)](#)

- Especificar los mecanismos de difusión.
- Formatear el informe principal.
- Evaluar el informe (recomendada).

Es relevante notar, que la lista de fases de una revisión sistemática es una guía sugerida y bien estructurada. Además, no es estrictamente secuencial y algunas pueden repetirse mas de una vez, involucrar iteraciones o de ser necesario volver a ejecutar etapas. En otras palabras, el proceso podrá tener variaciones

al caso típico y dependerá exclusivamente del contexto de aplicación, las necesidades y las limitaciones del proyecto, así como también las decisiones de los revisores.

A continuación, se hará una breve introducción a las principales actividades de la fase “Realizar una revisión sistemática”. Comprender el propósito y aplicación de cada sub-etapa, será sustancial para acompañar con éxito el desarrollo de la metodología en el proyecto PARKIBIP -Cap. **Revisión sistemática basada en evidencia-**.

- (I) Identificar la investigación. En esta etapa se establece y utiliza una estrategia de búsqueda para obtener una lista de todas las publicaciones relevantes para las preguntas de investigación. La estrategia de búsqueda puede ser automatizada mediante librerías digitales, búsquedas manuales, o mediante contactos con los investigadores; pero tiene que estar definida en el protocolo de la revisión y debe ser reportada luego de forma transparente y replicable. Una alternativa de búsqueda -a veces utilizada como complemento-, es el uso de las referencias en la búsqueda de nuevos estudios. Iniciando desde un estudio primario de interés, para luego emplear sus referencias para encontrar nuevos estudios relevantes (BACKWARD SNOWBALLING) o se buscan cuales estudios lo ubican en sus referencias (FORWARD SNOWBALLING). En caso de utilizar una búsqueda automatizada, se debe incluir una descripción de la cadena de búsqueda y los recursos utilizados.

- (II) Seleccionar los estudios primarios. En este paso se establecen los criterios de selección de artículos primarios, que determinarán aquellas publicaciones que serán incluidas -Crit. Inclusión, definen qué estudios se deben incluir como relevantes- o excluidas -Crit. Exclusión, se aplican sobre los estudios seleccionados o sobre la lista inicial para remover estudios irrelevantes- de la revisión. Los criterios de selección pretenden identificar los estudios primarios que proporcionan evidencia directa acerca de las preguntas de investigación. Según el procedimiento estipulado, el procedimiento de selección de artículos puede basarse en la lectura rápida de la combinación título, resumen, palabras claves de las publicaciones primarias -en general, se tienen muchos artículos, y es imposible leerlos por completo-.

- (III) Evaluación de calidad. Evaluar la calidad de un artículo es una actividad desafiante, y conlleva a establecer qué parámetros de calidad serán evaluados y el procedimiento para su aplicación. En general es adecuado definir una lista de chequeos (del inglés, checklist) en base a preguntas concisas y proceder a aplicarlos de tal forma que asegure la mejor calidad de la evaluación. La calidad se refiere a la medida en que el estudio minimiza el sesgo y maximiza tanto la validez interna como la externa.
- (IV) Extracción de datos. El objetivo de esta etapa es el diseño de formularios de extracción de datos para que los investigadores puedan registrar adecuadamente la información que se obtiene de los estudios primarios. El formulario debería incluir información asociada a la extracción, información general del estudio, (i.e. identificador del estudio, título y detalles de publicación), preguntas que habiliten a responder las preguntas de investigación, preguntas para evaluar la calidad de los estudios y resumen de los datos.
- (V) Síntesis de datos. La síntesis es el proceso en el cual se recolectan, integran y resumen los hallazgos incluidos en los estudios primarios. Sin importar el tipo de síntesis -descriptiva (o narrativa) y cuantitativa-, se debería comenzar con la creación de un resumen de los estudios incluidos, presentando intervenciones, número y característica de los participantes, resultados, entre otros. Una alternativa es realizar una síntesis temática y narrativa, en donde los datos se presentan tabulados, de tal forma que sean consistentes con las preguntas de investigación. Es relevante notar, que el resultado de la síntesis es el disparador para las conclusiones finales.

2.4. Análisis de la marcha

En los últimos años, el análisis de la marcha y la partición de sus fases ha sido un tema de investigación desafiante debido a su impacto en muchas aplicaciones relacionadas con tecnologías de la marcha. El paso o la marcha humana es un proceso complejo y cíclico que requiere la sinergia de los músculos del cuerpo, los huesos y el sistema nervioso [Saunders et al. \(1953\)](#), principalmente dirigido a mantener la posición vertical y mantener el equilibrio durante condiciones estáticas y dinámicas [Ayyappa \(1997\)](#).

El análisis de la marcha o GAIT ANALYSIS (GA) instrumentado provee mediciones confiables y objetivas sobre los patrones de locomoción y su variabilidad. Estas mediciones, pueden contribuir a la investigación de patologías y definición de programas de rehabilitación. GA está emergiendo como una herramienta efectiva para detectar enfermedades neurodegenerativas o para monitorear su progresión [Bertoli et al. \(2018\)](#). Específicamente, los parámetros de la marcha son indicadores útiles para discriminar enfermos de Parkinson así como para cuantificar la severidad de la enfermedad. Aprovechando los modelos cinemáticos, los sistemas pueden identificar ciertos patrones en las señales para detectar pasos y calcular parámetros; como la cadencia, la longitud de zancada, la velocidad de zancada, el rango de movimiento, los ángulos de las articulaciones, la velocidad máxima de oscilación, sentarse para pararse y pararse para sentarse, tiempos de transición, tiempo de giro, velocidad máxima de giro, etc. Estos parámetros han mostrado una correlación significativa con el diagnóstico de la enfermedad de Parkinson y la cuantificación de su severidad [Akbari et al. \(2017\)](#).

2.4.1. Identificación de fases de la marcha

El ciclo de la marcha se define como el período de tiempo que transcurre desde el contacto inicial de un pie con el piso, hasta la próxima ocurrencia del mismo evento en el mismo pie. Existen muchos modelos de partición de dicho ciclo, con diferentes niveles de granularidad, que dependen de los objetivos clínicos: el modelo que incluye dos grandes fases de la marcha, es decir, apoyo y vuelo, es en general el más adoptado según la revisión sistemática de [Taborri et al. \(2016\)](#); la cual recopila una gran variedad de estudios dentro del GA así como los distintos modelos empleados para distinguir el ciclo de la marcha en sus respectivas fases. La siguiente figura Fig. 2.5, condensa los distintos hallazgos y nomenclaturas empleadas en la literatura respecto a las granularidades del ciclo de la marcha.

Granularity	Gait Phases							
Two Phases	Stance						Swing	
Three Phases	First Rocker	Second Rocker				Swing		
Four Phases	Heel Strike		Flat Foot		Heel Off		Swing	
Five Phases	Heel Strike	Flat Foot	Heel Off	Toe Off		Swing		
Six Phases (a)	Initial Contact	Loading Response	Mid Stance	Terminal Stance	Pre Swing	Swing		
Six Phases (b)	Loading Response	Mid Stance	Terminal Stance	Pre Swing	Swing 1	Swing 2		
Seven Phases	Loading Response	Mid Stance	Terminal Stance	Pre Swing	Initial Swing	Mid Swing	Terminal Swing	
Eight Phases	Initial Contact	Loading Response	Mid Stance	Terminal Stance	Pre Swing	Initial Swing	Mid Swing	Terminal Swing
Gait [%]	0		60				100	

Figura 2.5: Nomenclatura para diferentes granuralidades del ciclo de marcha [Taborri et al. \(2016\)](#).

La correcta discriminación de las fases de la marcha es considerado un punto de partida para muchas aplicaciones, por ejemplo: (i) la evaluación del estado de recuperación de la marcha durante tratamientos de rehabilitación o luego de alguna intervención; (ii) la clasificación de las distintas actividades de la vida diaria o AVD; (iii) el control sinérgico de los dispositivos robóticos, para la recuperación de la movilidad de las extremidades inferiores; (iv) el entrenamiento atleta; y finalmente, (v) para distinguir una marcha normal de una patológica [Taborri et al. \(2016\)](#).

2.4.2. Sensores de medición para análisis de la marcha

Una variedad de sensores pueden ser utilizados para alimentar algoritmos que identifican las distintas fases de la marcha, clasificados principalmente como vestibles o no vestibles - del ingles WEARABLES y NON-WEARABLES respectivamente- . Dentro de los sensores vestibles, las plantillas de presión (en inglés “footswitches”) son consideradas un estándar [Taborri et al. \(2016\)](#); sin embargo, para resolver algunas limitaciones inherentes en estos dispositivos, las unidades de medición inercial se han vuelto muy populares en la última década. Buenos resultados han sido alcanzados utilizando también sensores de ultrasonido, electro-miografías (EMG) y electro-neurografías (ENN).

Los sensores no vestibles, como los sistemas optoelectrónicos o las plataformas de fuerza, se mantienen como los sistemas más precisos para analizar la marcha en ambientes interiores. A pesar de su utilidad, han tenido un uso muy limitado, especialmente en países en desarrollo, debido al alto costo que implica establecer un laboratorio especializado de marcha. La principal desventaja de este tipo de sistemas, es que no proveen una evaluación de los pacientes

durante sus actividades cotidianas, extrapolando las conclusiones de un estudio de corto tiempo que no refleja las condiciones reales del paciente. Además, la evaluación puede requerir mucho tiempo y puede ser difícil de tolerar por los sujetos, principalmente si se trata de condiciones patológicas [Kumar et al. \(2018\)](#).

Resulta fundamental para el proyecto PARKIBIP identificar el tipo apropiado de sensor que se utilizará para nutrir los algoritmos, que luego detectarán las distintas fases de la marcha en tiempo real. Es inherente al objetivo del proyecto que el tipo de sensor debe ser del tipo vestible o wearable, ya que el sistema debe ser portable y utilizable en ambientes exteriores. Una revisión sistemática basada en evidencias del año 2016 [Taborri et al. \(2016\)](#) identifica, selecciona y categoriza las distintas metodologías utilizadas para la detección de fases de la marcha, analizando ventajas y desventajas de cada una de las soluciones.

Los pedales (en inglés, “footswitches”) son dispositivos que emplean sensores capaces de detectar el contacto del pie con el piso durante un ciclo de marcha. Se emplean principalmente para medir parámetros temporales en la marcha. Estos sensores, que van colocados en la parte inferior del pie, operan como interruptores y se encuentran conectados por cables a algún tipo de terminal. Al ser de bajo costo y proveer una precisión muy alta, son usualmente utilizados para complementar la medición de otro tipo de sistema. Sin embargo, presentan varias desventajas claves para PARKIBIP, como por ejemplo: (i) el número de fases detectables es limitado, dado que las sub fases en la etapa de vuelo del pie no pueden ser discriminadas; (ii) el lugar en el que son dispuestos los sensores, en los pacientes con marcha patológica, afecta la precisión y la confiabilidad de medición [Aminian et al. \(2002\)](#); por ultimo, (iii) la conexión cableada afecta la oportunidad del sistema de ser un instrumento para las actividades de la vida diaria de los sujetos afectados.

Las plantillas de presión o “Foot Pressure Insoles”, presentan ventajas y desventajas similares a los pedales, ya que se basan en los mismos principios. Si bien existen plantillas que no requieren cableado, por ende no presentan dicha desventaja, éstas son de un costo elevado en comparación a los pedales u otras soluciones (p. ej. los sensores de aceleración, giroscopios o incluso las unidades de medición inercial).

El acelerómetro es la solución más utilizada para el análisis del paso ambulatorio. Con el objetivo de reconocer las fases del paso de un individuo, la

evidencia literaria existente describe diferentes combinaciones de éstos sensores, junto a los posibles lugares donde pueden ser ubicados. Los acelerómetros, entre otras unidades inerciales, son relativamente pequeños, de baja energía y larga autonomía, baratos, portables y altamente accesibles en la industria [Tong and Granat \(1999\)](#). En comparación con las soluciones basadas en pedales o plantillas ya mencionadas, el análisis de la aceleración ha permitido a los investigadores distinguir ciertas fases de la marcha, como lo es la sub-fase de etapa de vuelo (o en inglés, fase de Swing). Sin embargo, el empleo de acelerómetros para identificar las fases de la marcha, implica ciertas complejidades que resultan críticas: (i) la necesidad de compensar la gravedad durante el cómputo de la aceleración, de la parte del cuerpo analizada; (ii) el error acumulado en la integral doble aplicada al conjunto de datos sensados, para el cálculo de la posición; y, (iii) el procedimiento de calibración que permite ubicar de forma correcta el sensor respecto a un estándar de referencia trazable, y así confiar en la validez de sus mediciones.

La velocidad angular -como una variable para la partición de las fases del paso-, dato extraído desde el dispositivo giroscopio, ha adquirido una gran popularidad en las últimas décadas y es preferida sobre otras variables inerciales. La velocidad angular no se encuentra afectada por la gravedad, como tampoco la curva de sus ondas por el toque del talón con el suelo (a.k.a. “heel strike”). Además, no es necesario mantener un cuidado minucioso respecto a la posición en donde es ubicado el giroscopio dentro del segmento del cuerpo a medir. Aunque, una desventaja en su utilización es que, tal como la aceleración, la velocidad angular también se ve alterada por el error propagado, en caso de que se compute el ángulo del movimiento.

2.4.3. Unidad de Medición Inercial (IMU)

La Unidad de Medición Inercial, también conocida como IMU (Inertial Measurement Unit, por sus siglas en inglés), combina varios de los sensores mencionados, como por ejemplo el acelerómetro y el giroscopio; algunos también incluyen magnetómetro, barómetro, sensores de luz y/o temperatura, entre otros.

Por su parte, los dispositivos IMUs presentan una serie de ventajas que hace con que sea el tipo de dispositivo adecuado para el proyecto PARKIBIP. A continuación, se listan algunas de ellas [Zago et al. \(2018\)](#):

- La combinación de diferentes cantidades inerciales permite la evaluación de diferentes tipos de primer contacto con el piso, el cual representa un índice sumamente importante para analizar el estado de salud del sujeto, utilizando la estimación de la orientación del pie.
- La fusión de los datos de velocidad angular y aceleración lineal permiten compensar el error acumulado.
- Estos dispositivos permiten la evaluación en tiempo real de los parámetros espacio-temporales en entornos de la vida real, interiores y exteriores, superando así las limitaciones típicas de las mediciones de laboratorio.
- Los IMUs son más baratos y más prácticas que los sistemas de GA completos; solo se requiere una preparación relativamente rápida de los pacientes, ya que el sensor se coloca en el cuerpo por medio de un cinturón elástico.
- Los datos se pueden transferir fácilmente a través de Bluetooth al software dedicado.

Si bien el dispositivo tiene un gran potencial para el proyecto PARKIBIP, se deberán tener en cuenta ciertos factores que pueden afectar la señal de entrada y salida [Anwary et al. \(2018\)](#). Por ejemplo, durante la caminata, el movimiento de la ropa puede causar interferencias con la salida del acelerómetro. Puede haber vibración o ruido causado por el impulso si el sensor no es ajustado correctamente. Ajustar el sensor en un cinturón o cargarlo en el bolsillo también puede inducir interferencia de movimiento relativo. Para aumentar la confiabilidad y validez de la extracción automática de los parámetros para el análisis de la marcha, es fundamental elegir correctamente el lugar donde serán colocados los sensores y la forma en la que se coloquen.

Capítulo 3

Revisión sistemática basada en evidencia

En este proyecto se realizó una revisión sistemática basada en evidencia científica relacionada al uso de sensores IMU para analizar la marcha de las personas al caminar. Conforme a los requerimientos del proyecto, el objetivo de la revisión sistemática se centró en identificar todo el material relevante que permita estudiar la marcha de las personas con el uso de sensores, reducir o eliminar incertidumbres en la investigación, y sobre todo no reinventar la rueda. El conocimiento recopilado fue fundamental para la implementación de la solución PARKIBIP, para mejorar la marcha de los pacientes con la enfermedad de Parkinson.

3.1. Planeamiento y diseño de Protocolo de Revisión

Es esencial para alcanzar resultados satisfactorios establecer un plan, llamado protocolo de revisión, con todas las actividades a realizar y criterios a utilizar previo a la ejecución de la SLR. Desarrollar dicho plan, permite guiar y registrar las estrategias y toma de decisiones de diseño del estudio (es decir, es el documento de referencia).

A su vez, la actividad de planeación incluye definir las preguntas de investigación y desarrollar la estrategia de búsqueda, los criterios de inclusión/exclusión, el formulario de extracción de datos, ente otros.

De esta manera, a través de un proceso iterativo y mediante la participa-

ción activa de todos los miembros del equipo en el desarrollo, se estableció un protocolo de revisión en una hoja de cálculos tabulada. Resultó esencial para su construcción la realización de diversas pruebas piloto con el mismo que permitan encontrar inconsistencias en los procedimientos de recolección y agregación de los datos.

Para establecer el objetivo de la revisión sistemática, se plantean cuatro preguntas de investigación que guían la revisión:

- ¿Qué tipo de sensores IMU se utilizan para medir la marcha de las personas?
- ¿Cuántos sensores son necesarios para medir las características de la marcha?
- ¿Qué datos sobre la marcha se pueden obtener con los distintos sensores?
- ¿Qué técnicas o algoritmos se utilizan para analizar y procesar los datos?

La tabla Tab. 3.1 resume los componentes del protocolo, especifica los métodos que se utilizarán para llevar a cabo la SLR; su definición de antemano ayuda a reducir la probabilidad de sesgo en la investigación.

1. Contexto	
1.1. Objetivo	Investigar el uso de sensores IMU para analizar la marcha de las personas al caminar
1.2. Necesidad	Se quiere mejorar la marcha de las personas con enfermedad de Parkinson, por lo que se quiere relevar la forma de estudiar la marcha de las personas con el uso de sensores
1.3. Preguntas de investigación	
RQ1	¿Qué tipo de sensores IMU se utilizan para medir la marcha de las personas?
RQ1.1	¿Cuántos sensores son necesarios para medir la marcha?
RQ1.2	¿Qué datos sobre la marcha se pueden obtener con los distintos sensores?
RQ1.3	¿Qué técnicas o algoritmos se utilizan para analizar y procesar los datos?

2. Proceso de Búsqueda	
2.1. Estrategia	Búsqueda automática en título, abstract y keywords filtrado por disciplina
2.2. Snowballing	No
2.3. Términos	
Gait analysis	Gait analysis, step analysis, step detection, gait detection
Inertial measurement unit	IMU, Inertial measurement unit
Foot-ground contact	Foot-ground contact, foot ground detection, FGCD
2.4. Cadena de Búsqueda	TITLE-ABS-KEY ((“gait analysis” OR “step analysis” OR “step detection” OR “gait detection”) AND (“imu” OR “inertial measurement unit”) OR (“foot ground contact” OR “foot-ground contact” OR “foot ground detection” OR “FGCD”)) AND PUBYEAR > 2015 AND (LIMIT-TO (SUBJAREA , “ENGI”) OR LIMIT-TO (SUBJAREA , “COMP”) OR LIMIT-TO (SUBJAREA , “BIOC”) OR LIMIT-TO (SUBJAREA , “MEDI”))
2.5. Motores y Cadenas de Búsqueda	Portal Timbó, Scopus, ScienceDirect
2.6. Fuentes a considerar	N/A
2.7. Período a tener en cuenta	Debido a la evolución de las tecnologías, sensores y aplicaciones se consideran solamente los artículos con año de publicación 2016-2019
2.8. Procedimientos auxiliares	N/A
2.9. Evaluación del Proceso de Búsqueda	Muy bueno. Se logro obtener un numero adecuado de artículos seleccionados con posible buen aporte.
3. Proceso de Selección de Estudios Primarios	
3.1. Criterios de Inclusión	Que sea completo. Que se tenga acceso al artículo. Estudios en Ingles, Español y Portugués. Año de publicación > a 2015.

3.2. Criterios de Exclusión	Que no sean papers científicos o con el formato incorrecto. Que no traten la marcha de personas. Que usen sensores difíciles de acceder. Descartar estudios duplicados. Que usen sensores que no sean IMU. Que se enfoquen en la posición en la que se encuentra el sujeto y no en los parámetros de la marcha
3.3. Roles de los revisores	Ambos revisores realizarán las mismas tareas de forma equitativa. Selección, extracción y síntesis.
3.4. Cómo se evaluará el acuerdo entre revisores	Coefficiente de acuerdo Kappa
3.5. Cómo se resolverán diferencias	Discusión abierta en busca de un acuerdo. En caso de no llegar al acuerdo sera excluido.
4. Proceso de Evaluación de la Calidad de los Estudios	
4.1. Se evaluará la calidad de los estudios (justificar)	Si.
4.2. Lista de Verificación propuesta (del inglés, Checklist)	En esta etapa se utilizará la técnica de evaluación Scoring, la cual consiste en plantear una checklist y asignarle a cada ítem el valor 0 ó 1. Luego, se suma el puntaje obtenido para la checklist y se discute la exclusión del artículo.
4.3. Cómo se evaluará el acuerdo entre revisores	N/A
4.4. Cómo se resolverán diferencias	Discusión abierta en busca de un acuerdo. En caso de no llegar al acuerdo el artículo será excluido.
5. Proceso de Extracción de Datos	
5.1. Formulario de extracción	Se establece un formulario en formato tabular que indica los campos de interés que direccionan las preguntas de investigación.
5.2. Estrategia de extracción	División equitativa de artículos. Revisiones de a pares en base a 1 artículo aleatorio.

5.3. Consideraciones adicionales (datos calculados, subjetivos, etc.)	El proceso sera proactivo y abierto a la inclusión de características que se consideren relevantes. Además, se pretende obtener un listado de tipos y/o marcas de los IMU, para luego realizar una comparación.
6. Proceso de Síntesis de Datos	
6.1. Tipo de Síntesis	Se realiza una síntesis temática.
6.2. Forma de validación de la síntesis	N/A

Tabla 3.1: Componentes del protocolo de revisión de PARKIBIP.

Tal como se puede apreciar en la tabla, la misma se encuentra segmentada en distintas etapas, siendo consistente con las principales etapas de una SLR estándar.

3.2. Estrategia de búsqueda

Se planeó una estrategia de búsqueda automatizada, en función de la definición de una cadena de búsqueda que incluya la combinación de las palabras claves del problema de estudio, así como también sus alias (p. ej. Gait Analysis, IMU, etc.). Para ello, se seleccionaron los servicios de indexamiento y librerías digitales a través de portal Timbó (e.g, Scopus, IEEE, Springer Link), como motores de búsqueda principales.

Los términos de búsqueda utilizados se construyeron siguiendo los siguientes pasos:

1. Definir los términos principales
2. Identificar ortografías alternativas, sinónimos o términos relacionados para términos principales
3. Verificar las palabras claves en los documentos relevantes previamente identificados
4. Se usa el OR booleano para incorporar ortografías alternativas, sinónimos o términos relacionados
5. Se usa el booleano AND para vincular los términos principales

Es de gran importancia resaltar la codificación de la Cadena de búsqueda, la cual es imprescindible para ejecutar una búsqueda automatizada, y que de su buena formación dependerán los resultados futuros (i.e. una malformación de la cadena de búsqueda podrá alcanzar estudios irrelevantes, demasiados o una poca cantidad de los mismos).

TITLE-ABS-KEY (("gait analysis" OR "step analysis" OR "step detection" OR "gait detection") AND ("imu" OR "inertial measurement unit") OR ("foot ground contact" OR "foot-ground contact" OR "foot ground detection" OR "FGCD")) AND PUBYEAR > 2015 AND (LIMIT-TO (SUBJAREA , "ENGI") OR LIMIT-TO (SUBJAREA , "COMP") OR LIMIT-TO (SUBJAREA , "BIOC") OR LIMIT-TO (SUBJAREA , "MEDI"))

3.3. Criterios de inclusión/exclusión

Se formularon criterios de selección de artículos a incluir en la revisión, con el fin de identificar aquellos estudios que proporcionen información relevante a las preguntas de investigación y eliminar estudios arbitrarios. Dichos criterios fueron expresados como dos conjuntos, criterios de inclusión y de exclusión, ilustrados en la Tabla 3.2.

Tabla 3.2: Criterios de selección de estudios primarios

Criterios de inclusión	Criterios de exclusión
Estudios completos	Estudios arbitrarios -no científicos- o con el formato incorrecto
Con acceso al artículo completo	No traten la marcha de personas
Escritos en los idiomas inglés, español y portugués	Usen sensores que no sean IMU o difíciles de acceder
Publicados a partir del año 2015	Se descartan estudios duplicados
	Se enfoquen en la posición en la que se encuentra el sujeto y no en los parámetros de la marcha

Asimismo, a través de la cadena de búsqueda se limitan aquellos estudios pertenecientes a las áreas de conocimiento Ingeniería, y/o Computación, y/o Medicina, y/o Bio-ciencias.

3.4. Selección

El propósito de esta etapa es recopilar estudios primarios que proporcionan evidencia directa acerca de las preguntas de investigación. En primera instancia, los resultados primarios obtenidos fueron evaluados utilizando el título y el resumen del estudio para excluir aquellos que son claramente irrelevantes. Los documentos marginales o para los cuales la inclusión/exclusión es incierta, fueron mantenidos en el conjunto selección y la decisión final fue tomada durante la extracción de datos. Posteriormente, se aplicaron los criterios de selección establecidos iniciando la Selección de estudios relevantes para el proyecto.

Para llevar a cabo este proceso y facilitar la manipulación de los estudios primarios (i.e. búsqueda y filtrado de datos), resultado de la búsqueda automatizada, los mismos fueron tabulados en hojas de cálculos que incluyen la meta-data:

- Año (Year)
- Título del artículo (Title)
- Resumen (Abstract)
- Título de la fuente (Source title)
- Enlace (Link)
- Editorial (Publisher)
- Idioma del documento original (Language of Original Document)
- Tipo de documento (Document Type)
- Tipo de acceso (Access Type)
- Fuente del artículo (Source)
- Autores (Authors)
- Inclusión
- Motivo exclusión
- Observación del Artículo

El proceso de Selección se realizó en varias etapas, partiendo de 493 artículos que fueron arrojados por la búsqueda automatizada. Aplicando los criterios de selección establecidos, se conformo una lista de 165 estudios, que actuó como entrada de datos para la tarea “lectura rápida”. De esta manera, leyendo únicamente el título y el resumen de cada artículo se realizó una primera selección, recopilando alrededor de 85 artículos relevantes para PARKIBIP. A partir de este punto, cada estudio preseleccionado fue identificado unívocamente en el Universo con un identificador auto-incremental.

Luego, leyendo nuevamente el título y el resumen del artículo a evaluar, fueron valorados o puntuados en una escala del 1 al 3, siendo los de categoría 1 aquellos que en primera instancia parecieran ser muy útiles y los de categoría mas alta los que si bien aportaban valor no serían tan relevantes. En este sentido, fue difícil establecer la categoría de ciertos artículos en base al título y su resumen, por lo que en una tercera etapa se leyeron cerca de 40 artículos para establecer su categoría (principalmente las secciones de objetivos del artículo, metodología y resultados). Ambos miembros del equipo realizaron esta tarea para revisar y justificar los desacuerdos, de esta forma se alinearon criterios.

En resumen, el resultado de la fase de Selección se puede visualizar en la tabla 3.3.

Tabla 3.3: Resultados del proceso de Selección de estudios. primarios

Categoría o valoración	Resultado
1 (relevancia máxima)	30 estudios
2 (relevancia intermedia)	22 estudios
3 (relevancia mínima)	33 estudios

3.5. Evaluación de calidad

De la misma forma que se establecieron criterios de inclusión y exclusión para el proceso de selección, es importante establecer criterios para evaluar la calidad de los artículos seleccionados. Mediante la evaluación de estos criterios se descartan los estudios que no son válidos o son subjetivos. La evaluación de la calidad es una etapa relevante ya que la excelencia de los estudios primarios puede incidir en la calidad de los resultados obtenidos.

Para ésta tarea se utilizó la técnica de “scoring”, la cual consiste en puntuar los artículos según determinados criterios preestablecidos. Los criterios están

dados por el conjunto de interrogaciones:

- ¿El artículo se encuentra basado en investigación?
- ¿Es claro el objetivo del artículo?
- ¿Hay una descripción adecuada acerca del contexto en el cual se realizó la investigación?
- ¿Se diseñó la investigación de manera correcta para cumplir con el objetivo del artículo?
- ¿Hubo alguna replicación, por ejemplo múltiples casos de prueba?
- ¿Hay una descripción clara de los hallazgos?
- ¿Los hallazgos son de valor para la industria o para investigadores?

De esta manera, para cada artículo se respondieron dichas preguntas con una valoración de 1 si la respuesta era afirmativa o en caso contrario 0. Luego, el puntaje de calidad del artículo es la suma de todas las valoraciones, por lo que cada artículo obtuvo un puntaje de 0 a 7. Se excluyen aquellos artículos cuya puntuación fue menor a los 5 puntos.

3.6. Extracción

El objetivo en esta etapa del proceso de revisión, es la extracción del contenido necesario para direccionar las preguntas de investigación. Para ello, los estudios resultados del proceso de selección, fueron incluidos a la herramienta Mendeley para un mejor seguimiento. Conforme a mantener consistencia a través de los estudios y dentro del equipo de revisores, se utilizó una estrategia de extracción en base a formularios preestablecidos.

De esta manera, se realizó una extracción de datos que incluye detalles generales de la publicación (i.e. título, autores, universidad u organización), como también se encuentra basada en temas específicos: (i) Objetivo del estudio; (ii) Características de los sensores utilizados; (iii) Cantidad de sensores necesarios; (iv) Ubicación de los sensores; (v) Fases del paso analizadas; (vi) Algoritmos utilizados; (vii) Características de los sujetos -sanos o con alguna enfermedad-; (viii) Resultados y efectividad. La figura Fig. 3.1 ilustra y ejemplifica el resultado del proceso mencionado.

Generales	
Id	18
Título	Neurological diseases differentiation analysis using inertial measurement units
Autores	Luksys D., Kizlaitiene R., Kaubrys G., Pakulaite G., Jatuzis D., Kaladyte-Lokominiene R., Buneviciute R., Griskevicius J.
Abstract	Human gait is cyclic process, which is affected by various neurological disorders like Parkinson's disease (PD) or multiple sclerosis (MS). Instrumented gait analysis facilitate monitoring and diagnosing motor deficiencies; it may serve as a tool for clinicians evaluating the patients. This study aims at finding biomechanical parameters that allow separating pathological gaits via application of inertial sensors to capture the gait. The experiment involved 33 subjects divided in three groups PD, MS and healthy controls (CO). The analysis showed the statistically significant difference between stride time of the right leg in MS and CO groups, left stride time between PD and CO, stance time difference between PD and CO group, right and left leg. Hip flexion and extensions amplitudes difference was between CO and PD group, left hip flexion and extension.
Año de publicación	2017
Tipo Publicacion	Conference Paper
Universidad u Organizacion	Institute of Electrical and Electronics Engineers Inc.
Origen	Lithuania
Especificos	
Objetivo del estudio	This study aims at finding biomechanical parameters that allow separating pathological gaits via application of inertial sensors to capture the gait. Compare subjects with pathological and normal gait.
Características del sensor	Shimmer Research SR7. IMU has a tri - axial accelerometer (freescala MMA7361, accelerometer limit $\pm 1,5$ g), gyroscope (InvenSense 500 MEMS Gyro, angular velocity limit ± 500 o/s, sensibility 2 mV/o/s) and magnetometer (Honeywell HMC5843, input field boundaries $\neq -0,7 - 4,5$ Ga). Inertial sensor size 53 x 32 x 12 mm, weight ~ 30 g, powered with 450-mAh battery. The data from the sensors were received via a Bluetooth
Cantidad de sensores	6 devices
Ubicacion	Sensors were attached on each subjects' right and left thigh, shank and foot using elastic straps.
Características del paso analizad	The following metrics for the analyses were selected: right, left leg stride, swing and stance time's, right, left hip joint flexion, extension angles
Algoritmo utilizado	Filter algorithm: filtered using fifth order Butterworth filter (order 5, Fcutoff = 5 Hz). In order to reduce the integration drift, partially due to a continuous component, we performed also a high pass Butterworth filtering (order 1, F cutoff=0.5 Hz). Also have algorithm equation of calculation times.
Características de los sujetos (si son sanos o tienen alguna enfermedad)	The experiment involved 33 subjects divided in three groups PD, MS and healthy controls (CO). volunteers who were divided into three groups - 11 control (CO) subjects, 11 Parkinsonian (PD) subjects and 11 multiple sclerosis (MS) subjects. None of the participants had any other injuries or disease affecting movement or coordination other than PD or MS and all provided informed consent prior to participating in the study.
Resultados	A one - way ANOVA with a significance level of $\alpha = 0.05$ was used to test the null hypothesis that the mean of calculated are the same between the PD, MS and CO groups. Statistically significant difference was found between the right leg stride time, between PD, MS and CO groups ($F = 6.44$, $p = 0.0047$), left leg stride time was difference between PD and CO ($F = 7.69$, $p = 0.002$). Stance time right leg difference between PD and CO group ($F = 4.23$, $p = 0.0242$), left leg - between PD and CO ($F = 6.9$, $p = 0.0034$).Calculation of hip flexion and extension amplitudes, significant difference was detected between left hip extension ($F = 3.4$, $p = 0.0465$) and left hip flexion -between PD and CO ($F = 4.62$, $p = 0.0178$).
Efectividad	The study showed that neurological disease potentially could be separated using the IMU. The analysis of spatial and temporal gait parameters can be found in the specific parameter that describe the neurological disease.

Figura 3.1: Formulario de extracción empleado en PARKIBIP.

3.7. Síntesis: Resultados primarios

Teniendo en cuenta el tiempo planificado para la etapa de análisis de la literatura, se priorizó sintetizar el conjunto de investigaciones pertenecientes a la Categoría 1 (30) cuyo aporte tendrá gran relevancia para el proyecto PARKIBIP, tal como indica la tabla Tab.3.3. Con el objetivo de responder las preguntas planificadas se analizó: el año de publicación, el objetivo, las características de los sensores utilizados, la cantidad y ubicación de los sensores, las características del paso analizadas y los algoritmos computacionales empleados para dicho fin, las características de los sujetos -sanos o con alguna patología-.

Más allá del estado del arte respecto al uso de dispositivos IMU para analizar la marcha de las personas al caminar -objetivo central-, resulta útil para los productores y consumidores de la investigación conocer las diversas salidas para la investigación y el crecimiento del campo de estudio. Así, en primera

instancia se realizó una clasificación previa de los artículos en base a cuatro criterios:

1. Fecha de Publicación
2. Países involucrados en las investigaciones
3. Tipo de Publicación
4. Editorial

La búsqueda realizada se focalizó en los años 2016 al 2019. Esta decisión se debió a que los avances tecnológicos han avanzado considerablemente en los últimos años, por lo tanto, no es relevante para este estudio el análisis de artículos con más de 4 años de antigüedad. La figura Fig. 3.2 presenta la distribución de estudios primarios por año de publicación. La razón de que exista un único estudio publicado para el año 2019 es trivial -mismo año que la presente LSR-; sin embargo, se observa un buen número de estudios para los restantes años.

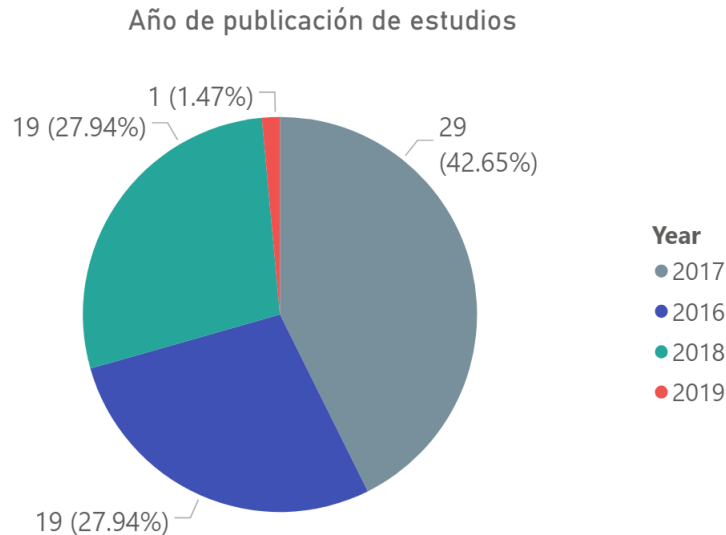


Figura 3.2: Distribución de estudios primarios por año.

Resulta importante identificar los países interesados en la materia, ya sea para compartir el conocimiento adquirido y trabajar colaborativamente, o simplemente Benchmarking. En la figura Fig. 3.3 se presenta la distribución de países involucrados en las investigaciones. Existe una gran prevalencia de investigaciones con origen Italiano (7) indicando un fuerte campo de interés para

dicho País. Asimismo, se puede apreciar una buena dispersión de estudios primarios (31) por Países (18) -aprox. 60 % de países distintos-, lo que infiere la importancia compartida del campo de estudio entre países y pocas investigaciones realizadas.

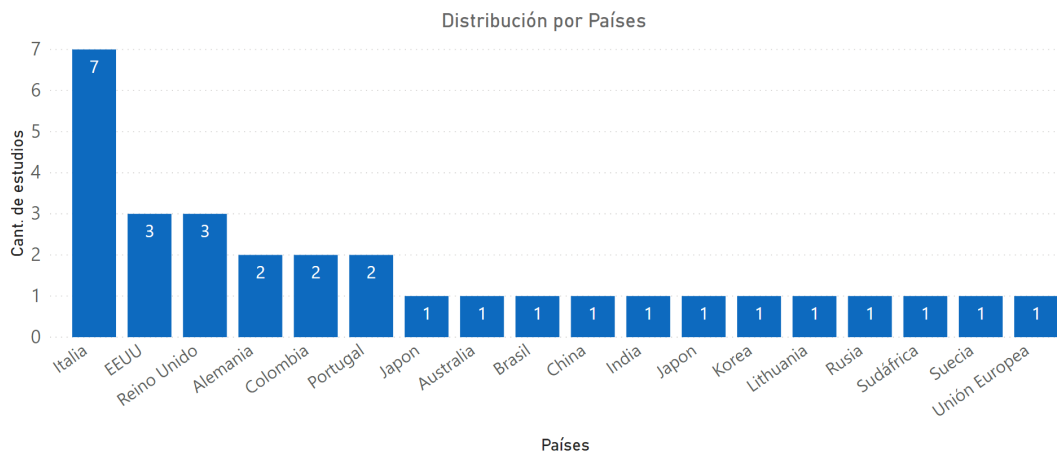


Figura 3.3: Gráfico de los países de origen de las investigaciones.

Por su parte los artículos analizados fueron publicados tanto en conferencias como en revistas de manera proporcional, con el análisis de una revisión sistemática respecto a las distintas particiones existentes del ciclo de la marcha, y la inclusión del capítulo de un libro. Además, dichas publicaciones fueron fragmentadas por casa editora, siendo la asociación profesional de ingeniería electrónica e ingeniería eléctrica (IEEE) la de mayor presencia con el 50 % de publicaciones aproximadamente. Las figuras Fig. 3.43.5.

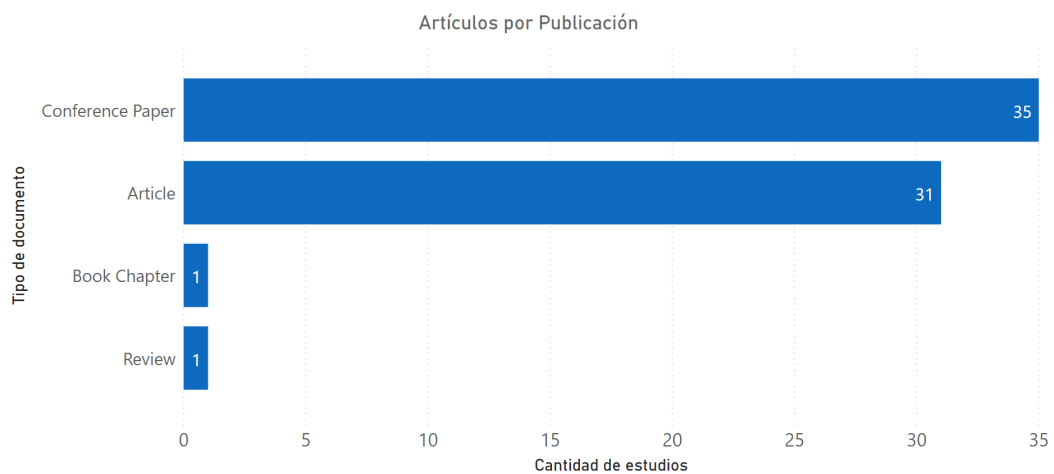


Figura 3.4: Artículos sobre análisis de la marcha por tipo de publicación.

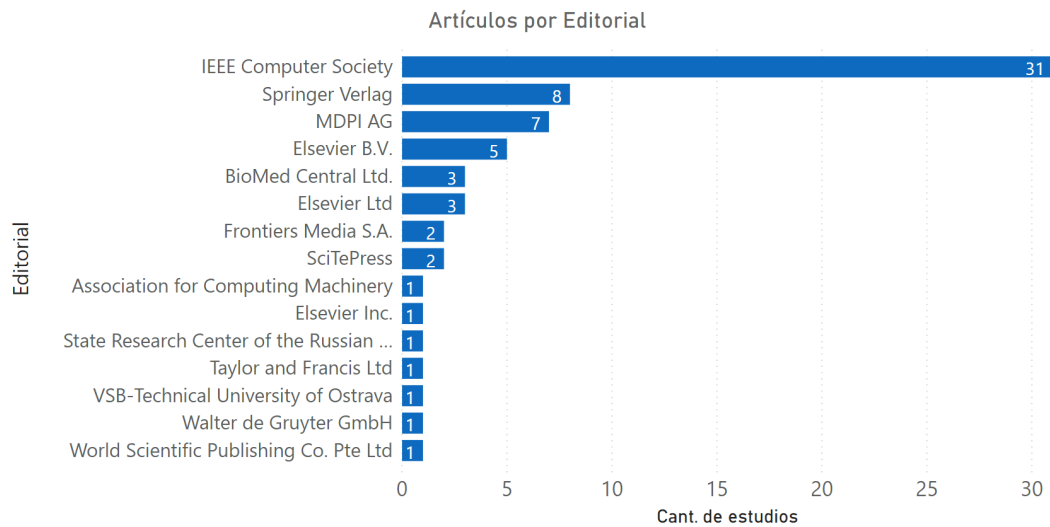


Figura 3.5: Artículos sobre análisis de la marcha por editorial.

Desde la perspectiva de la revisión sistemática realizada, es importante abordar las preguntas de investigación elaboradas en el objetivo con la finalidad de adquirir conocimiento, contemplar dificultades y riesgos, producir ventajas de la evidencia recolectada.

Ya mencionado, los dispositivos IMU pueden contar con diversos sensores de medición; en su forma más básica cuentan con un acelerómetro. En general, salvo en las investigaciones cuya información no fue especificada, la mayoría de los estudios usaron dispositivos IMU con 9/6 DOF (sus siglas del inglés, refieren a los grados de libertad, variables independientes) de diversas compañías; integrando la combinación de un acelerómetro y giroscopio tridimensionales, en algunos casos también un magnetómetro tridimensional. Es relevante notar que, todos los dispositivos empleados tienen como salida datos en la unidad correspondiente al sensor y sin procesar; de ahí la necesidad de algoritmos específicos al caso de uso.

Según el área de estudio particular dentro de la marcha de las personas, las investigaciones emplearon entre 1 y 7 dispositivos IMU. Aproximadamente el 86.5% de los estudios requirió dos o más dispositivos, en donde prevalece con el 50% el uso de dos dispositivos. La figura Fig. 3.4 sintetiza el número de dispositivos que fueron necesarios para realizar cada investigación.

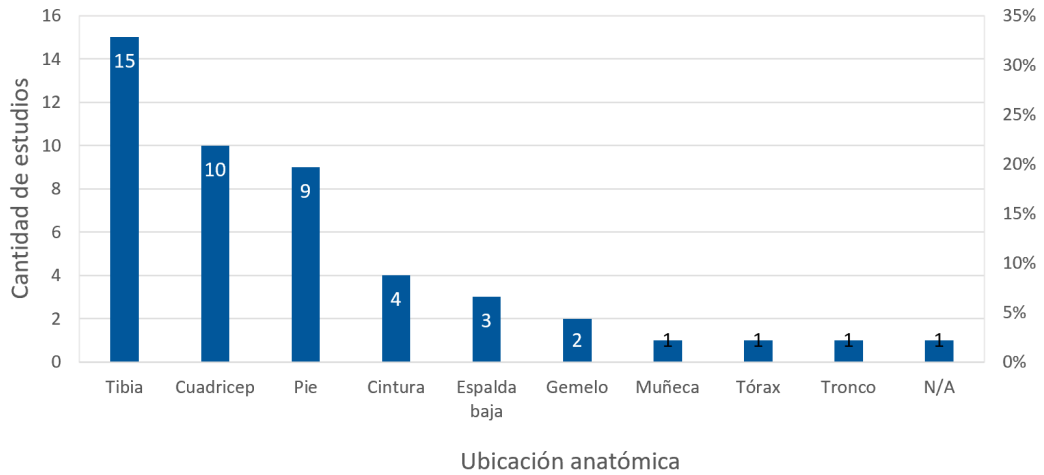
Tabla 3.4: Cantidad de dispositivos empleados para analizar la marcha de personas según los artículos contemplados.

Dispositivos IMU (en unidades)	Artículos (en unidades)
1	4
2	12
3	6
4	2
5	1
6	3
7	1
Total	28

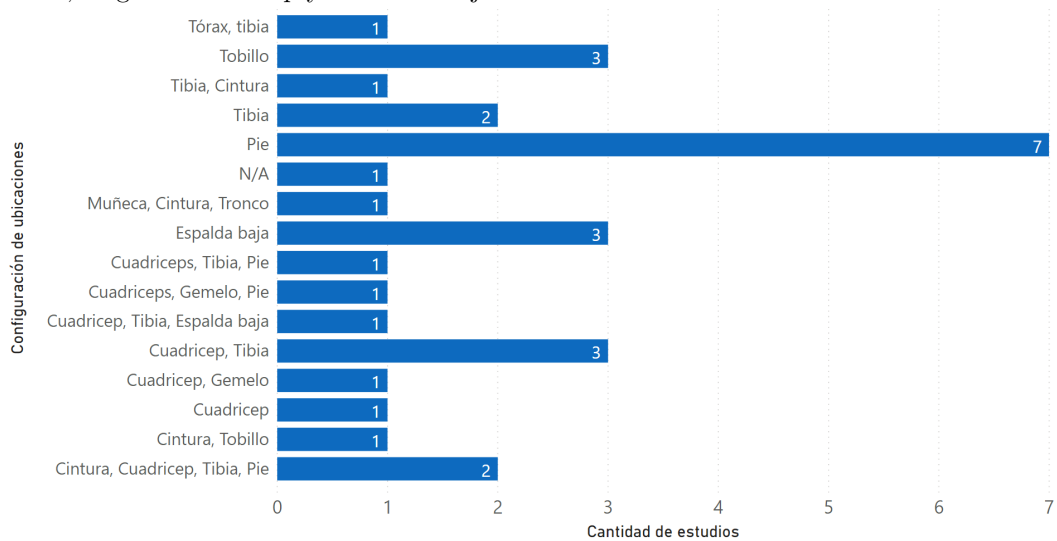
A priori, se podría concluir que empleando un único dispositivo inercial sería suficiente para analizar la marcha de las personas y elevar los costos y las complejidades con otros adicionales. Sin embargo, éste resultado subjetivo no es correcto; y como fue mencionado, la cantidad de dispositivos necesarios depende de las variables de estudio dentro del “gait analysis” -i.e. según partición del ciclo de la marcha, parámetro espacio-temporal particular (aceleración, velocidad, etc.), simetría entre pasos, rotación tobillo, entre otras-. Sumado a la anterior afirmación, se debe contemplar los algoritmos computacionales empleados, la ubicación del dispositivo, la cantidad de variables de la marcha a analizar para dicha ubicación, la efectividad reportada para esa variable de estudio. Por ejemplo, un único dispositivo ubicado en el centro de masa -a la altura de la pelvis- podría detectar si existe movimiento, en cambio, no se podría estudiar el ciclo de la marcha en 2 fases Stance (fase de apoyo) y Swing (fase de vuelo) por pie y por lo tanto limitando los parámetros espacio-temporales de los mismos.

Similar a lo mencionado, la ubicación anatómica del dispositivo en el cuerpo humano tiene gran relevancia y determina/limita el espectro de estudio. La figura Fig. 3.6 con sus respectivas sub-figuras, resume las distintas ubicaciones anatómicas empleadas para posicionar al menos un IMU, así como también sus distintas configuraciones que permiten el “gait analysis”. Las mismas, exhiben que si bien las posiciones anatómicas son variadas y aproximadamente el 33% localiza al menos un IMU en la Tibia; según las configuraciones, en su mayoría se localiza únicamente en el pie del sujeto en cuestión.

Combinando los resultados anteriores -cantidad y ubicación en el cuerpo-



(a) Ubicaciones anatómicas empleadas para posicionar al menos un IMU en el análisis de la marcha de personas. Según la evidencia, aproximadamente el 33% de estudios emplea la Tibia, luego el Cuádricep y el Pie del sujeto con un $\approx 20\%$.



(b) Configuraciones de ubicaciones anatómicas para posicionar los IMU que permiten analizar la marcha de un sujeto.

Figura 3.6: Ubicaciones de dispositivos IMU en el cuerpo de los sujetos para el análisis de la marcha, según los distintos estudios de la revisión sistemática.

sobre los dispositivos empleados, se puede afirmar que la mayoría de estudios empleó dos dispositivos IMU localizados en el pie del sujeto particular. Además, dicha ubicación permite abordar los problemas esenciales:

- Identificar el ciclo de la marcha por extremidad
- Estudiar diversos parámetros espacio-temporales por extremidad o de forma general

Por otro lado, resulta crucial para la investigación entender **de qué forma se procesan los datos, qué eventos se detectan** para distinguir las distintas etapas de la marcha y **qué tipo de parámetros se pueden obtener** con cada tipo de análisis. La tabla Tab. 3.5 enumera los datos mencionados para cada uno de los 30 artículos analizados.

ID	Nombre del artículo	Eventos detectados	Algoritmo utilizado	Parámetros obtenidos
1	Gait evaluation using inertial measurement units in subjects with Parkinson's disease	Heel Strike, Toe Off	Double Integration. High pass filter. Butterworth filter.	Cadence (steps/min). Mean velocity (m/s). Stride length (m). Stride duration (s). Step duration (%). Stance phase duration (%). Swing phase duration (%). Double support phase duration (%).
2	Towards a portable human gait analysis & monitoring system	Heel Strike, Toe Off	Double Differential and Integration, ZUPT	Steps. Kinematic angles in Sagittal, Transverse and Frontal planes.
3	Trainer in a pocket - Proof-of-concept of mobile, real-time, foot kinematics feedback for gait pattern normalization in individuals after stroke, incomplete spinal cord injury and elderly patients	Heel Strike, Toe Off	Kalman Filter - ZUPT	Stride length. Foot to ground angle. Stance time. Swing time
4	Validity of shoe-type inertial measurement units for Parkinson's disease patients during treadmill walking	Heel Strike, Toe Off	Double Integration. Butterworth filter.	Cadence. Step length. Step time
5	Estimation of spatio-temporal parameters of gait from magneto-inertial measurement units: Multicenter validation among Parkinson, mildly cognitively impaired and healthy older adults	Heel Strike, Toe Off	Custom method*	Stride duration. Step duration. Cadence

ID	Nombre del artículo	Eventos detectados	Algoritmo utilizado	Parámetros obtenidos
6	Optimal Foot Location for Placing Wearable IMU Sensors and Automatic Feature Extraction for Gait Analysis	Heel Strike, Toe Strike, Heel Off, Toe Off	Trapezoidal Double Integration. Butterworth filter.	Times of stance, swing, single and double support; stride length and stance phase; gait velocity, stride duration, cadence and step length; step number, moving distances, instant step speed and average speed; step counting; times of heel strike, toe strike, heel-off, and toe-off; stride length and duration; walking distance, time and speed
7	The clinical application of inertial measurement unit in identification of foot drop symptoms	Heel Strike, Toe Off	Custom method*	Cadence, step length, foot to ground angle
8	An Automatic gait feature extraction method for identifying gait asymmetry using wearable sensors	Heel Strike, Toe Strike, Heel Off, Toe Off	Trapezoidal Double Integration. Butterworth filter.	Stride length and stance phase; gait velocity, stride duration, cadence and step length; step number, moving distances, instant step speed and average speed; step counting; times of heel strike, toe strike, heel-off, and toe-off; stride length and duration; walking distance, time and speed
9	IMU-based gait analysis for rehabilitation assessment of patients with gait disorders	Heel Strike, Flat foot, Midstance, Heel off, Toe Off, Midswing, Heel Strike	Peak detection, ZUPT	Cadence, Symmetry
10	Real-time tool for human gait detection from lower trunk acceleration	Heel Strike, Flat foot, Toe Off, Midstance, HO	Double Integration, State Machine, Exponential Filter.	Cadence, step time
11	The application of inertial measurements unit for the clinical evaluation and assessment of gait events	Heel Strike, Toe Off	Sensor Fusion, State Machine. Butterworth low pass filter	Steps. Dorsiflexion angle
12	Validation of a new model-free signal processing method for gait feature extraction using inertial measurement units to diagnose and quantify the severity of Parkinson's disease	Heel Strike, Toe Off	Low pass and high pass filters Madgwick filter to calculate the orientation. Double Integration	Cadence, stride length, stride velocity, range of motion, joint angles, maximum swing velocity, sit to stand and stand to sit transition times, turn time, turn peak velocity, etc.

ID	Nombre del artículo	Eventos detectados	Algoritmo utilizado	Parámetros obtenidos
13	Validation of a step detection algorithm during straight walking and turning in Patients with Parkinson's disease and older adults using an inertial measurement unit at the lower back	Heel Strike, Toe Off	CWT - Continuous wavelet transform	Steps
14	L-DOPA and freezing of gait in Parkinson's disease: Objective assessment through a wearable wireless system	Heel Strike, Toe Off.FOG (Freezing of gait)	Proportional Integral Controller Filter. Custom method for orientations and events detection	Step velocity (centimeters per second) was defined as the distance covered by the leg in time unit; stride length (centimeters), the distance between two consecutive heel strike of the same foot; stride time (seconds), the time from initial contact of one foot to subsequent contact of the same foot; and finally, cadence (steps per minute) was defined as the number of steps per minute.
15	Assessing the accuracy of an algorithm for the estimation of spatial gait parameters using inertial measurement units: Application to healthy subject and hemiparetic stroke survivor	Heel Strike, Toe Off	Madgwick orientation Double Integration - ZUPT	cadence, walking speed, stride length, step length, and timing of the different stages of the gait cycle. More
16	The development and concurrent validity of a real-time algorithm for temporal gait analysis using inertial measurement units	Heel Strike, Toe Off	Noise-Zero Crossing algorithm NZC	GCT = Gait Cycle Time, SLS = Single Limb Support Time, DLS = Double Limb Support Time.
17	Neurological diseases differentiation analysis using inertial measurement units	Heel Strike, Toe Off, Midstance. Max. Flexion, min. extension.	Butterworth low and high pass filter.Double Integration	Right, left leg stride, swing and stance time's, right, left hip joint flexion, extension angles.
18	A wearable wireless system for gait analysis for early diagnosis of Alzheimer and Parkinson disease	Heel Strike, Toe Off	Low pass and high pass filter. Complementary Filter. Double pendulum model.	Stride number Mean stride length Standard deviation stride Standard deviation stance Mean swing Standard deviation swing Cadence Step time
19	Linear progression measurement and analysis of human gait for the development of a multifunctional robotic walker	Heel Strike, Toe Off	Low pass and high pass filters. Double Integration.	Steps, distance, velocity, time

ID	Nombre del artículo	Eventos detectados	Algoritmo utilizado	Parámetros obtenidos
20	Knee joint angle monitoring system based on inertial measurement units for human gait analysis	Knee joint angle	No specified	N/A. Just compare with other methods results.
21	Efficient techniques for gait-analysis: Comparing marker-less and imu-based tracking systems for monitoring rehabilitation processes	The oscillations of the medio-lateral center of mass (CoM).	Kalman Filter + ZUPT	Velocity, step or stride length and cadence.
22	A practical step length algorithm using lower limb angular velocities	Heel Strike, Toe Off, MSt and MSw	Inverted Pendulum Model algorithm	Step time, Gait velocity, Stride length
23	An IMU-to-Body Alignment Method Applied to Human Gait Analysis	Heel Strike, Toe Off	Joint Angles Calculation with Joint Coordinate System (JCS)	Knee flexion angle, ankle angle, hip flexion angle
24	Proposal of a short step and stride measurement system for uncontrolled environments	Loading response, Midstance, Terminal Stance, Preswing, Initial Swing, Midswing, Terminal Swing	Quaternion orientation with gravity removal. Trapezoidal Double Integration	Short step length, Stride length
25	Gait event detection in laboratory and real life settings: Accuracy of ankle and waist sensor based methods	Heel Strike, Toe Off	Custom method	stride time, step time and stance time
26	Technical challenges using magneto-inertial sensors for gait analysis	Not specified	Not specified	Not specified
27	Inertial BSN-Based Characterization and Automatic UPDRS Evaluation of the Gait Task of Parkinsonians	Heel Strike, Toe Off	Butterworth low pass filter.	Gait Cycle Time, Stance Time, Swing Time. Stride length, stride velocity, step length, step velocity.
28	Estimation of foot trajectory during human walking by a wearable inertial measurement unit mounted to the foot	Heel Strike, Midstance, Toe Off	Gravity removal. Double Integration	Step-by-step stride length
29	Motion Monitoring based on a Finite State Machine for Precise Indoor Localization	Heel Strike, Midstance, Toe Off	Orientation with euler angles. ZUPT - Kalman Filter.	Distance, stride length

ID	Nombre del artículo	Eventos detectados	Algoritmo utilizado	Parámetros obtenidos
30	Handling gait impairments of persons with Parkinson's disease by means of real-time biofeedback in a daily life environment	Not specified	Not specified	Distance, cadence, stride length, stride duration, gait speed

Respecto a los eventos detectados, la amplia mayoría de los artículos (89.2%) detecta los eventos de Heel Strike y Toe Off. Estos eventos son suficientes para dividir el ciclo de marcha en sus fases principales: Stance (apoyo) y Swing (vuelo). Un 60.7% de los artículos detecta únicamente estos dos eventos. Algunos estudios necesitan estudiar la marcha con mayor granularidad, otros necesitan analizar otros aspectos; a continuación se listan los diferentes eventos estudiados en todos los artículos:

- Heel Strike: Apoyo del talón
- Heel Off: Despegue del talón
- Toe Strike: Apoyo de los dedos
- Toe Off: Despegue de los dedos
- Midstance: Mayor apoyo del pie
- Knee max. flexion: Máxima flexión de la rodilla
- Knee min. extension: Extensión mínima de la rodilla
- FOG - Freezing of gait: Bloqueo de la marcha
- Pre swing: Etapa inicial en fase de vuelo del pie
- Mid swing: Etapa intermedia en fase de vuelo
- Terminal swing: Etapa final en fase de vuelo
- La oscilación del centro de masa medio-lateral

Como se puede observar en la tabla Tab. 3.5, se han utilizado una gran variedad de algoritmos para el estudio de la marcha. Algunos de estos estudios [20, 26, 30] no especificaron la forma con la que obtienen los resultados. Otros

[5, 7, 25] utilizaron métodos personalizados, basados en heurísticas y máquinas de estados.

Una primera etapa en el análisis de la marcha consiste en el cálculo de la orientación de cada dispositivo IMU. La orientación del dispositivo debe conocerse con un alto grado de precisión para que las mediciones de gravedad se puedan distinguir de la aceleración física del sensor. Incluso pequeños errores en la estimación de la orientación producirán errores extremadamente altos en la aceleración medida, lo que se traduce en errores aún mayores en las estimaciones de velocidad y posición. Para los artículos estudiados, algunos de los métodos utilizados son: la construcción de matrices de rotación en base a los ángulos de Euler, el método sugerido por Martin & Salaun [Martin and Salaün \(2010\)](#), la técnica propuesta por Mahony [Mahony et al. \(2006\)](#) y el filtro propuesto por S. Madgwick [Madgwick \(2010\)](#).

Para el análisis de la marcha mediante la detección de eventos, uno de los métodos más utilizados es el de **Doble Integración (DI)**. Este método fue utilizado por el 44.4% de los artículos que reportan el algoritmo utilizado [1, 4, 6, 8, 10, 12, 15, 17, 18, 19, 24, 28]. Es el método más simple entre los reportados, tanto con su implementación a través del método del Trapecio, como con el método de Simpson. Consiste en Integrar la aceleración para obtener la velocidad, para luego realizar una nueva integración y obtener el desplazamiento. Este método por si solo no es para nada suficiente, ya que, por un lado es necesario conocer las condiciones iniciales, y por otro, existe un error de integración que es incremental y acumulativo. Este error debe ser removido con algún otro método. Con este propósito se utilizan distintos tipos de filtros como: High Pass Filter, Low Pass Filter, Butterworth filter, etc.

Uno de los métodos utilizados para reducir el error acumulado es el **Zero Velocity Update (ZUPT)**, por sus siglas en inglés). Se basa principalmente en la detección de los intervalos de tiempo en los cuales el sistema está en una fase estacionaria, es decir, cuando el sistema tiene una posición y una orientación constante. El uso de actualizaciones de velocidad cero es especialmente atractivo para limitar el error en los sistemas donde los dispositivos que se encuentran montados a los pies del sujeto, ya que durante el ciclo de la marcha el pie vuelve a un estado estacionario.

Algunos de los estudios aplicaron ZUPTs a **Kalman Filter** para mejorar la solución de navegación. El filtro de Kalman no solo estima el error de velocidad en los intervalos ZUPT detectados, sino que también estima las correcciones

de posición, altitud y sesgo de los sensores inerciales debido a correlaciones cruzadas en la matriz de transición.

El resto de los métodos relevados para detectar los eventos de la marcha son variados. Algunos están basados en máquinas de estado, otros en detección de picos, otros utilizan el cálculo de ángulos articulares utilizando un sistema de coordenadas articulares, por mencionar algunas de estas variantes (ver en tabla Tab. 3.5).

Resulta interesante conocer los distintos parámetros espacio temporales que se obtienen en cada estudio mediante los algoritmos antes mencionados. Estos parámetros son el resultado final del análisis de la marcha. Los estudios tienen como objetivo analizar estos parámetros que caracterizan la marcha de los sujetos y seleccionan los algoritmos adecuados para obtener estos parámetros.

A continuación, se nombran los principales parámetros que se obtuvieron como resultado utilizando los distintos algoritmos:

Cadencia (pasos/min), Velocidad media (m/s), Velocidad instantánea (m/s), Longitud de zancada (m), Longitud de paso (m), Duración del paso (s), Duración de la zancada (s), Duración de fase de apoyo simple (%), Duración de fase de vuelo (%), Duración de fase de apoyo doble (%), Cantidad de pasos; Ángulos cinemáticos en planos Sagital, Transverso y Frontal; Distancia recorrida (m), Tiempo total (s), Ángulo de dorsiflexión; y más.

En cuanto a las características de los sujetos convocados para efectivizar las pruebas experimentales para cada investigación, en promedio se realizaron pruebas sobre unos 36 sujetos. Como se puede visualizar en la gráfica de la Fig. 3.7, la gran mayoría de los estudios optan por realizar las pruebas con sujetos de todas las edades, tanto jóvenes como adultos mayores; sin embargo, solamente cuatro estudios hicieron pruebas únicamente con jóvenes y otros cuatro estudios únicamente con adultos mayores.

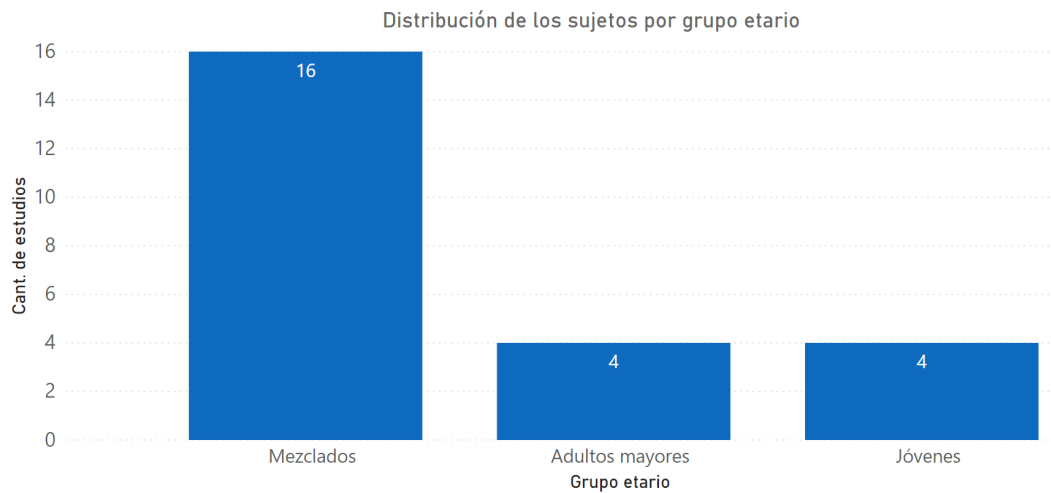


Figura 3.7: Resumen de la población de estudio por grupo etario.

Respecto a la condición de clínica de los sujetos participantes, se puede observar una ligera inclinación a realizar pruebas con personas completamente sanas (41.7%), aunque existe una proporción muy parecida de los estudios que se realizaron con sujetos enfermos (25%) o con ambas condiciones (33.3%) -ver gráfica en Fig. 3.8-.

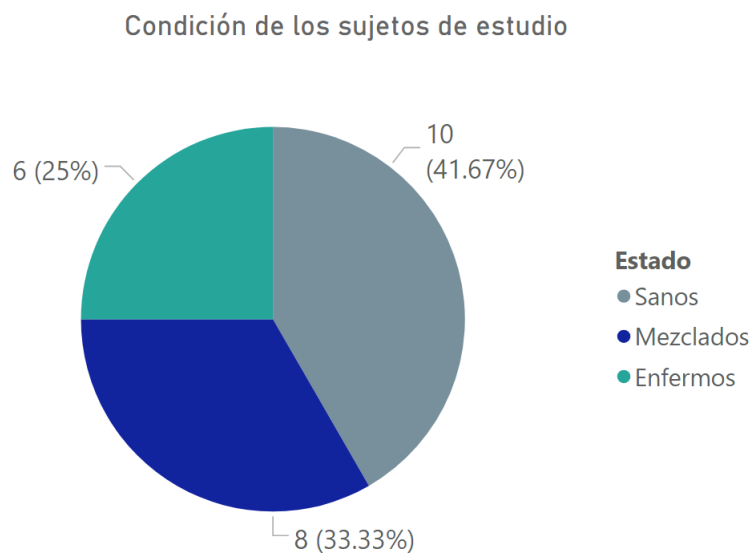


Figura 3.8: Resumen de condición sobre la totalidad de sujetos en cada estudio

Las métricas obtenidas respecto a los sujetos de las pruebas en las distintas investigaciones responden al contexto de investigación, en su totalidad son estudios de “gait analysis” con una finalidad médica, aunque no todos para

una alteración de la marcha. Por lo tanto, no necesariamente se requieren pruebas con sujetos enfermos.

3.8. Limitaciones

El presente estudio integra distintos conceptos que requieren un nivel medio a alto de entendimiento, como los son los algoritmos físico-matemáticos necesarios para detectar los eventos del ciclo de la marcha -HS, TO, Stance, Swing-, los métodos numéricos capaces de estimar los parámetros espacio-temporales (i.e. velocidad de la marcha), la biomecánica, y el manejo de múltiples señales de los distintos sensores del IMU. Se reconoce el no cumplimiento de estos requerimientos y un aprendizaje en el transcurso de la SLR.

En muchos casos no fue posible acceder al estudio completo. Este tipo de situaciones se da cuando la temática tratada tiene un alto impacto comercial y es necesario abonar una suma de dinero a cambio del material completo.

En general, los artículos analizados se enfocan en innovación, presentan una nueva combinación de técnicas para predecir eficazmente alguna de las fases del ciclo de la marcha o al menos un parámetro espacio-temporal. Si bien algunos investigadores proponen nuevos modelos, los esfuerzos se centran principalmente en encontrar y optimizar técnicas de estimación.

3.9. Resultados secundarios

Consecuentemente, se obtuvieron diversos resultados sumamente valiosos, que no fueron planificados al inicio del proyecto mediante preguntas de investigación. De esta manera, se listan los resultados adicionales de la revisión sistemática realizada:

1. Identificación de las compañías de soluciones de sensores. De gran utilidad para efectuar una amplia comparación de dispositivos existentes en el mercado y poder seleccionar el adecuado.
2. Posiciones oportunas del IMU en el pie. Las estimaciones resultantes de los algoritmos numéricos se ven afectadas por la localización del dispositivo en el cuerpo a sensor. De esta manera, se pudo analizar las distintas localizaciones que lograron mejores resultados y contemplarlas en la solución.

3. Particionamiento del ciclo de la marcha. Se constata la diversidad de granularidades de las fases de la marcha (i.e. dos, tres, cuatro, cinco, seis, siete y ocho) cada una con sus ventajas y desventajas; la cual permite definir las fases necesarias y suficientes a identificar mediante el sistema PARKIBIP.
4. Métodos de estimación de fases de la marcha. Existen diversos algoritmos innovadores para estimar las fases del ciclo de la marcha, que serán tenidos en cuenta para lograr una solución eficaz.
5. Métodos de estimación de parámetros espacio-temporales. Similar al punto previo, se detectaron variadas técnicas innovadoras para estimar los parámetros espacio-temporales de la marcha. En ambas ocasiones, no existe una regla de oro, y todo indica que el campo de estudio es emergente.
6. Requerimiento de un algoritmo eficaz de orientación. Trabajar con el movimiento de un cuerpo particular mediante dispositivos sujetos al mismo, requiere la identificación de la orientación de los dispositivos -por ende de la del cuerpo-, respecto a un sistema de referencia inercial (en general, el sistema tierra). Existen diferentes métodos para computar la orientación, basados en rotaciones clásicas hasta sistemas complejos como el gradiente descendente. Es sabido que, en los sistemas de gait analysis, un cuello de botella para estudiar parámetros de la marcha es lograr un resultado sumamente eficaz de la orientación del cuerpo respecto al sistema de referencia inercial. Será uno de los desafíos de PARKIBIP obtener tal resultado.
7. Conocimiento, desafíos y riesgos. El hecho de analizar detalladamente diversas investigaciones (aproximadamente 85) trae como consecuencia directa la adquisición de conocimiento, por consiguiente las dificultades y riesgos a contemplar y mitigar durante el transcurso del proyecto. Como por ejemplo la implementación de un filtro eficaz y eficiente de Orientación, frecuencias adecuadas por tipo de sensor (acelerómetro, giroscopio, magnetómetro), la necesidad de algoritmos de predicción para parámetros espacio-temporales (es decir, si bien se cumplen las leyes mecánicas, las mismas no son aplicables computacionalmente por el error cuadrático acumulativo).

Capítulo 4

Proyecto: Decisiones de diseño

La finalidad del presente capítulo es explicitar el proceso de investigación y toma de decisiones llevado a cabo durante el proyecto PARKIBIP, mediante los cuales se obtienen los resultados, y por tanto el cumplimiento -parcial o total- de los objetivos establecidos. A partir de las cualidades desafiantes del proyecto, las incertidumbres respecto al contexto clínico y de análisis -marcha de sujetos con EP-, y los objetivos preestablecidos; se optó por conducir durante todo el proyecto una investigación científica, rigurosa, orientada al entendimiento del contexto y a minimizar riesgos -SLR realizada-. De esta manera, todo aquel material, procedimiento o método arbitrario fue descartado mediante un proceso de evaluación de calidad.

4.1. Selección tecnológica

En primera instancia y como disparador, se elaboró una revisión de la evidencia científica existente vinculada a los avances tecnológicos y a su aplicación en la detección de las fases de la marcha en personas, con el fin de definir el tipo de dispositivo adecuado a emplear en PARKIBIP (i.e. plantillas inteligentes, unidad de medición inercial, etc.). La misma, se ejecutó en el portal en línea Timbó, el cual habilita a acceder a la última bibliografía y literatura científica-tecnológica mundial reportada.

Para seleccionar la tecnología a utilizar, se consideró una revisión sistemática realizada en el año 2016 [Taborri et al. \(2016\)](#), la cual identifica, selecciona y categoriza las distintas tecnologías para realizar la detección de las fases del paso, analizando ventajas y desventajas de cada solución. De esta manera, la

selección del tipo de sensor se realizó teniendo en cuenta características relevantes para el proyecto en cuestión, entre ellas, el costo, la accesibilidad, la efectividad y la portabilidad del dispositivo.

Bajo los criterios mencionados, el tipo de dispositivo elegido que mejor se adecuaba al proyecto fue un dispositivo IMU (del inglés Inertial Measurement Unit o Unidad de Medición Inercial). Estos dispositivos portables, de bajo costo y larga autonomía, han sido utilizados con buenos resultados en el área de análisis de movimientos dinámicos lineales y angulares en los últimos años. Se componen esencialmente de los sensores acelerómetro, giroscopio y magnetómetro, y en ciertos casos se extienden a incluir barómetro, sensores de luz, entre otros.

4.2. Estudio de factibilidad

Luego de seleccionar el tipo de dispositivo a utilizar, se tomo la decisión de realizar una **revisión sistemática de la literatura (SLR) respecto al uso de sensores IMU para analizar la marcha de las personas al caminar**. El objetivo de una revisión sistemática es buscar e identificar todo el material relevante relacionado con un tema determinado conforme a responder las preguntas de investigación, siguiendo una metodología objetiva, analítica y tan repetible como sea posible. De acuerdo con los procedimientos descritos en el capítulo de fundamentos, apartado **Revisión Sistemática Basada en Evidencias**.

Los principales factores de motivación que impulsaron a aplicar esta metodología fueron:

- Adquirir conocimiento respecto a la problemática. Por ejemplo el contexto clínico, el análisis de la marcha y sus fases, la aplicación de sensores IMU y sus desafíos, algoritmos numéricos reportados -matemáticos y computacionales-, entre otros
- Corroborar la propiedad innovadora del proyecto -no exista otra que aborde el mismo tema-
- Estudiar la marcha de las personas
- Reducir o eliminar incertidumbres en la investigación

- Identificar recomendaciones a seguir para PARKIBIP
- Identificar las principales metodologías y técnicas aplicadas

Si bien elaborar una SLR es un proyecto en sí mismo, el cual requiere mucho esfuerzo y dedicación, realizarla fue sumamente valiosa y se justifica plenamente tanto con los factores de motivación expuestos como con la obtención de sus resultados en la sección **Síntesis: Resultados primarios** de la SLR.

El capítulo **Revisión sistemática basada en evidencia**, describe el procedimiento completo de la SLR aplicada a la realidad de PARKIBIP, y una variación de sus etapas -de acuerdo al proyecto-. Además, presenta los resultados primarios que responden las preguntas de investigación planificadas, como también los resultados secundarios -no preestablecidos- derivados del análisis pormenorizado del conocimiento relevado.

Bajo el soporte fundamentado de la SLR elaborada, se toman las decisiones pertinentes con el fin de mitigar los riesgos y lograr construir PARKIBIP. Los resultados alcanzados actúan como guía de conclusiones a tomar, algunos puntos son: (i) Proveedores de HW, (ii) Cantidad de dispositivos IMU y ubicación en el cuerpo, (iii) Granularidad de fases a detectar como eventos, (iv) Posibles algoritmos numéricos, (v) Filtro de Orientación eficaz y eficiente.

4.3. Benchmark: Elección del dispositivo IMU

Durante el transcurso del estudio de factibilidad y adquisición de conocimiento llevado a cabo mediante la revisión sistemática, se analizaron los distintos proveedores de hardware especializados en soluciones de sensores embebidos. Recordando que, no se encuentra dentro de los objetivos la construcción de un dispositivo de HW -por ende no pertenece al alcance del Proyecto-, fue necesario seleccionar la tecnología (Sec.4.1) a emplear y posteriormente elegir el proveedor de hardware que mejor se adecua a Parkibip.

Existen en el mercado decenas de empresas que ofrecen estos dispositivos junto a sus distintas variaciones, por lo que surgió la necesidad de hacer una evaluación de mercado y elegir la más apropiada. La Tabla comparativa Tab.4.1 permite visualizar las principales características evaluadas.

A efecto de la etapa, se concluyó en la utilización de la solución de producto de dispositivos vestibles MetaMotionR (MMR). Además de ser un dispositivo liviano, accesible y de bajo costo; integra los principales sensores de lectura,

Tabla 4.1: Comparación de dispositivos IMU

	Arion	BTS G- Sensor	EXL S3	Inven- Sense	Isen	MMR	Opal APDM Inc	Reha- Gait (HA- SO- MED)	Senno- Gait Smar- tIn- soles- PRO	Shim- mer Sen- sor	Wave- share	Xsens
Acelerómetro (3-axis)	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI
Giroscopio (3-axis)	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI
Magnetómetro (3-axis)	NO	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI
Barómetro	SI	NO	NO	NO	SI	SI	NO	-	NO	NO	SI	NO
Altímetro	-	-	-	NO	NO	SI	-	-	NO	SI	NO	NO
Bluetooth	SI	SI (3.0)	SI (2.1)	NO (I2C, SPI)	NO	SI (LE)	NO	SI	SI	SI	NO	-
Wifi	NO	NO	NO	NO	SI	NO	NO	NO	SI	NO	NO	NO
Tiempo Real	SI	SI	SI	SI	SI	SI	<30ms	-	-	-		<2 ms
Runtime calibration	NO	NO	NO	SI	SI	SI	-	-	-	SI		-
Peso	-	37gr	22g	-	46g	8.5 g	<25g	-	-	23.6g	3g	-
Medidas (mm)	-	70 x40 x18	54 x33 x14	3 x3 x0.9	56 x38 x18	24 x17 x4	43 x39 x13	60 x15 x35	35-45	51 x34 x14	31 x16 x2.5	57
Batería	~7hs	~8hs	2 hs	-	~3.5 hs	2-14d	8- 16hrs	-	>48hrs	-	-	-
SDK Android	NO	NO	NO	SI	SI	SI	No	NO	NO	SI		SI
SDK iOS	NO	NO	NO	NO	NO	SI	No	NO	NO	NO		NO
Comunidad OpenSource	NO	NO	NO	SI	NO	SI	NO	NO	NO	NO		NO
Documentación	NO	NO	NO	SI	NO	SI	NO	NO	NO	SI		SI
Vibración	NO	NO	NO	NO	NO	SI	NO	NO	NO	NO		NO
Precio	€189	-	-	-	-	USD99	-	-	-	€359	USD16	€800

como lo son el acelerómetro (BMI160 6-Ejes), el giroscopio (BMI160 6-Ejes), el magnetómetro y la unidad vibratoria.

4.4. Cantidad y ubicación del dispositivo

Se cree pertinente emplear dos dispositivos IMU sincronizados, unidos con bandas elásticas de velcro a los tobillos -justo por encima de los maléolos- de los pacientes con la enfermedad de Parkinson tal como indica la Fig. 4.1. De esta manera, es posible estudiar el comportamiento y las distintas variables vinculadas a cada extremidad por separado, así como combinarlas para lograr otros resultados genéricos. De forma complementaria, la decisión logra un espectro o visión de análisis mas amplio, el cual habilita a escalar el alcance del proyecto en materia de variables de la marcha o patologías (i.e. rotaciones, asimetrías, etc.).



Figura 4.1: Banda elástica junto a la unidad de medición inercial (IMU).

4.5. Granularidad de fases de la marcha

Estandarizando los conceptos de fases de la marcha, tal como se sintetiza en [Taborri et al. \(2016\)](#), se pueden emplear distintas granularidades de las fases de la marcha (i.e. dos, tres, cuatro, cinco, seis, siete y ocho), tal como indica la Fig. 4.2.

El presente proyecto reconoce dos fases en el ciclo de la marcha, fase de apoyo y fase vuelo. En este sentido, los algoritmos matemáticos considerados

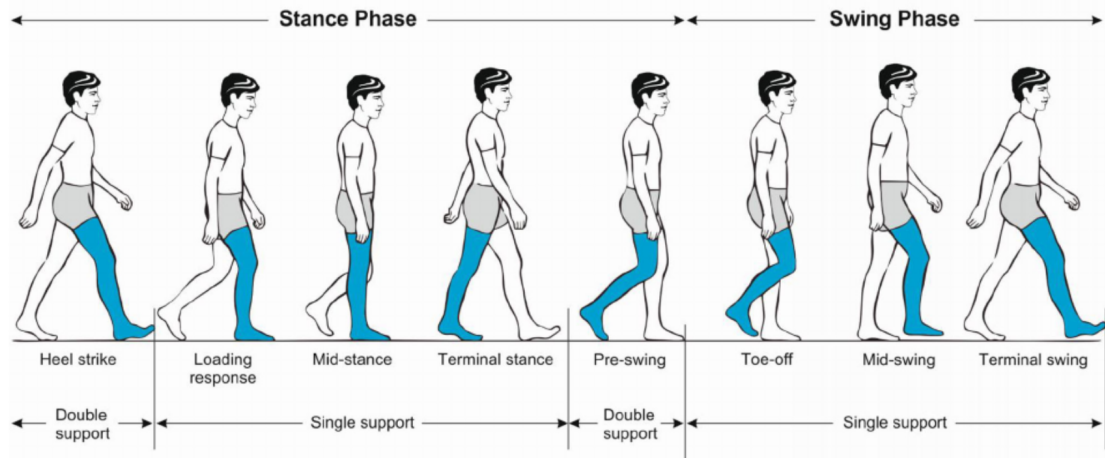


Figura 4.2: Partición del ciclo de la marcha normal en 8 fases. Tomado de “W. Pirker y R. Katzenschlager” [Pirker and Katzenschlager \(2016\)](#).

se enfocan en detectar eficazmente los eventos golpe de talón (HS, del inglés Heel Strike) y el despegue de los dedos del pie (TO, del inglés Toe Off).

4.6. Filtro de Orientación

La estimación precisa de la orientación de un cuerpo desempeña un rol crítico en una variedad de campos: aeroespacial, robótica, navegación y análisis del movimiento humano e interacción de máquinas. Por ejemplo, en la estimación de las fases de la marcha y el cálculo de parámetros espacio-temporales. Si bien una variedad de tecnologías permiten la medición de la orientación, los sistemas inerciales presentan la ventaja de ser completamente autónomos, de modo que la entidad de medición no está restringida ni en movimiento, ni a ningún entorno o ubicación específicos.

La función de un filtro de orientación es calcular una única estimación de orientación de un cuerpo relativo a un marco de referencia inercial; a través de la fusión óptima del giroscopio, acelerómetro y magnetómetro.

La principal limitante para el computo de la orientación es el error propagado a algoritmos al estimar la orientación. Para un mayor entendimiento, se mencionan ciertos inconvenientes:

- Un acelerómetro y un magnetómetro medirán los campos gravitacionales y magnéticos de la tierra, y proporcionarán una referencia absoluta de orientación. Sin embargo, es probable que estén sujetos a altos nive-

les de ruido (i.e. las aceleraciones debidas al movimiento corromperán la dirección de la gravedad medida). La integración doble de datos de aceleración, por ejemplo, daría como resultado cantidades muy elevadas de error residual, ya que la deriva se acumularía cuadráticamente.

- Un giroscopio (velocidades angulares) requiere conocer condiciones iniciales, y puede integrarse en el tiempo para calcular la orientación del sensor. La integración de los errores de medición del giroscopio conducirán a un error acumulado en la orientación calculada. Por lo tanto, los giroscopios por sí solos no pueden proporcionar una medida absoluta de orientación.
- Ciertos métodos requieren alta complejidad computacional o en otros casos no se pueden implementar. Adicional, pueden demandar frecuencias de muestreos de datos que superen el ancho de banda; por ejemplo, una frecuencia de muestreo entre 512 Hz y 30 kHz.

El problema se incrementa cuando deseamos calcular parámetros espaciotemporales, las estimaciones de posición y velocidad basadas en acelerómetros de sensores de bajo costo (IMU) son muy malas e inutilizables. Esto no se debe a que los acelerómetros en sí sean deficientes, sino a que la orientación del sensor debe conocerse con un alto grado de precisión para que las mediciones de gravedad se puedan distinguir de la aceleración física del sensor. Incluso pequeños errores en la estimación de la orientación producirán errores extremadamente altos en la aceleración medida, lo que se traduce en errores aún mayores en las estimaciones de velocidad y posición.

Por lo tanto, estimar la orientación de un cuerpo resulta en un verdadero desafío computacional y un cuello de botella en los sistemas de análisis de la marcha (alta dependencia en algoritmos). Sin embargo, es una necesidad para PARKIBIP, y se debe emplear un algoritmo eficaz y eficiente que se ajuste a la propuesta del presente proyecto, convirtiéndolo en un desafío aun mayor. Un algoritmo lento y costoso conllevará a retardos en las estimaciones de las fases de la marcha (por ende los estímulos asociados) y a consumos excesivos de los recursos del dispositivo inteligente (desgaste energético y rendimiento del dispositivo).

Los sistemas denominados AHRS (por sus siglas en inglés, Attitude and Heading Reference Systems) pueden proporcionar una medición completa de

la orientación relativa a la dirección de la gravedad y el campo magnético de la tierra. Existen diversas técnicas que logran buenas aproximaciones y fueron excelentes resultados -inputs para el presente hito- de la **Revisión sistemática basada en evidencia** para identificar el filtro de orientación a emplear. La construcción de matrices de rotación en base a los ángulos de Euler, el método sugerido por Martin & Salaun [Martin and Salaün \(2010\)](#), la técnica propuesta por Mahony [Mahony et al. \(2006\)](#) y el filtro propuesto por S. Madgwick [Madgwick \(2010\)](#).

Para el caso Mahony (muy empleado), combina las señales crudas del acelerómetro y el giroscopio mediante el uso de un algoritmo de estimación de orientación propuesto. Para eliminar la deriva del giroscopio y proporcionar la orientación del sensor en el espacio, se emplea un vector de corrección a partir de la orientación estimada previamente y el vector acelerómetro instantáneo. En cambio, la implementación de S. Madgwick incorpora la distorsión magnética (combinando el magnetómetro) y la compensación de deriva del giroscopio; reportándose en la literatura mejores resultados con éste último método.

A su vez, Madgwick proporciona ciertos beneficios adicionales: (i) computacionalmente económico; requiriendo 109 (IMU) o 277 (MARG) operaciones aritméticas escalares cada actualización de filtro, (ii) efectivo a bajas tasas de muestreo; p.ej. 10 Hz y (iii) contiene 1 (IMU) o 2 (MARG) parámetros ajustables a definir por las características del sistema. Por consiguiente, para estimar la orientación de los dispositivos, se decide aplicar la técnica de gradiente descendiente optimizado y derivado analíticamente sugerida por Madgwick, descrita en la FIG. 5.10 tomada de [Madgwick \(2010\)](#).

4.7. Detección de Velocidad Cero

Los dispositivos inerciales de bajo costo sufren de deriva en sus sensores, debido a la naturaleza integradora de los sistemas de navegación inercial (INS), cualquier desviación pequeña se acumulará y crecerá con el tiempo sin límites. Este suceso genera un error en la estimación de la posición, que crece proporcionalmente al cubo del tiempo de operación del sistema. De este modo, la navegación inercial solo es factible durante períodos de tiempo en el rango de unos pocos segundos. Sin embargo, el crecimiento del error cúbico puede reducirse imponiendo restricciones en la solución sobre la dinámica del sistema de navegación inercial [Skog \(2009\)](#).

En general, un tipo de información utilizado para este propósito es la detección de los intervalos de tiempo en los cuales el sistema está en una fase estacionaria, es decir, cuando el sistema tiene una posición y una orientación constante. Este método se lo conoce como ZUPT (por sus siglas en inglés). El uso de actualizaciones de velocidad cero es especialmente atractivo para reducir el error en los INS que se encuentran montados a los pies del sujeto, ya que durante el ciclo de la marcha el pie vuelve a un estado estacionario. La figura Fig. 4.3 permite visualizar un flujo del proceso -enfocado en ZERO VELOCITY DETECTION (del inglés Zero Velocity Detector) y abstrayendo otros componentes- empleado en PARKIBIP.

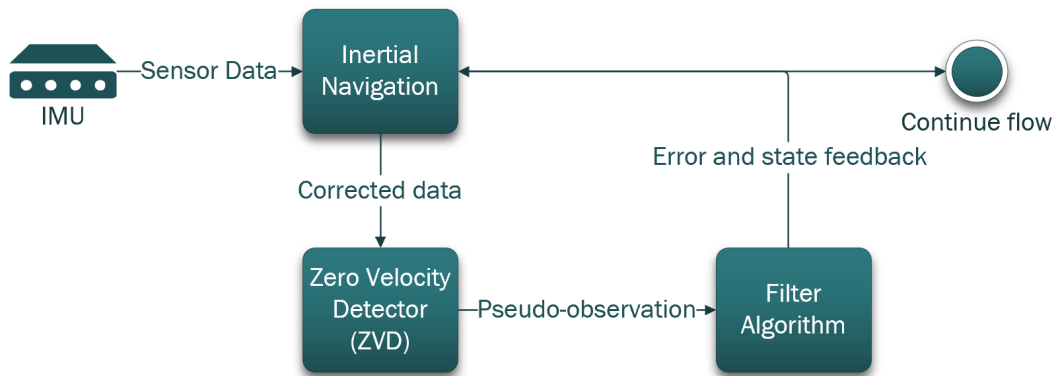


Figura 4.3: Flujo de detección de velocidad cero (ZVD) de PARKIBIP.

Cuando el ZVD decide que el IMU se encuentra estacionario, el sistema genera una pseudo-observación y emite un evento hacia el algoritmo de filtrado del detector. El filtro compara la pseudo-observación con la estimación de velocidad del INS y usa la diferencia para estimar los errores del INS. Las estimaciones de error se retroalimentan para reducir el error acumulativo.

Luego, detectados los momentos de cero velocidad, es trivial inferir que los algoritmos ZVD's permiten reconocer las transiciones de fases dentro del ciclo de la marcha. Por ejemplo, un ciclo binario Stance—Swing -fases de apoyo y vuelo-.

El problema de identificar los instantes de velocidad cero se formaliza como un problema de test binario de hipótesis - H_0 : hipótesis nula, H_1 : hipótesis alternativa-, y se analiza aplicando la teoría de detección estadística. Algunos ejemplos de algoritmos de ZVD son: (i) Acceleration moving variance detector -MV-, (ii) Acceleration magnitude detector -MAG-, (iii) Angular rate energy detector (ARE), Generalized Likelihood Ratio Test -GLRT-, entre otros.

Por lo tanto, se propone la utilización de algoritmos de ZVD para romper la deriva del error cúbico mediante la técnica de ZUPT e inferir el ciclo de la marcha del sujeto en deambulación. Además, se enfatiza en el requerimiento online del sistema PARKIBIP, acentuando la complejidad de la solución. Para ello, se construirá una combinación estandarizada de algoritmos óptimos en su función (e.g. Kalman Filter, Orientation Filter, ZVD, otros algoritmos auxiliares), cuya sinergia incremente la operativa de PARKIBIP y habilite a cumplir los propósitos de dicho INS.

4.8. Filtro de Kalman

Un método de filtrado lineal de gran utilidad y con variadas aplicaciones, es el KALMAN FILTER (KF, por sus siglas en inglés) en sus diversas variedades. En estadística y teoría de control, el KF, también conocido como estimación cuadrática lineal (LQE), es un algoritmo que utiliza una serie de mediciones observadas a lo largo del tiempo -que contienen ruido estadístico y otras inexactitudes-, y produce estimaciones de variables desconocidas que tienden a ser más precisas que los basados en una sola medición. Suponiendo fuentes de ruido distribuidas como variables aleatorias gaussianas, el filtro de Kalman proporciona la solución que minimiza el error cuadrático medio (MMSE) del problema lineal pre-establecido.

En otras palabras, KF es un algoritmo óptimo de estimación que permite hallar la mejor estimación de variables de interés -no medibles- a partir de la combinación de variables indirectas -medibles-. De esta manera, se logra extraer información respecto a algo que no es posible medir, desde algo que si lo es.

Con la finalidad de introducir a la temática se mencionan brevemente algunas de sus aplicaciones. Los filtros de Kalman o su variación extendida para sistemas no lineales, son de gran utilidad en la robótica dado que facilitan el problema de la planificación de un camino mínimo. Un clásico problema de la navegación robótica que puede ser resuelto con dicho algoritmo, es planificar el movimiento de un robot que viaja con una velocidad limitada, desde un determinado punto a otro en una superficie en presencia de obstáculos en movimiento [Nasir et al. \(2017\)](#). Otra aplicación es lo que se denomina control y navegación de vehículos, en particular aviones, naves espaciales y barcos posicionados dinámicamente [Mao et al. \(2007\)](#). Además, KF es un concep-

to ampliamente aplicado en el análisis de series de tiempo que se utiliza en campos como el procesamiento de señales y la econometría.

Para entrar en contexto con la utilidad de KF en PARKIBIP, se expone el problema de un sistema de navegación de un automóvil en movimiento, al estimar la posición del mismo dentro de un túnel. El automóvil integra dispositivos de medición a bordo, como lo son un IMU (acelerómetro y giroscopio) y un GPS (estima la posición en base a señales satelitales). Dentro del túnel, el sensor de GPS se ve afectado por el bloqueo de señal satelital (e.g. señal intermitente y de baja frecuencia, ruido en el receptor). Análogamente, realizar la doble integración de la señal del acelerómetro recibida desde el IMU -para obtener la posición-, se encuentra sujeta a desvíos incrementales debido al error acumulativo de integración respecto al tiempo. Por lo tanto, los dispositivos de medición a bordo no logran estimar la posición por si solos. En éste escenario, un KF puede ser usado para fusionar las mediciones de los sensores y hallar la solución óptima al problema de estimar la posición del vehículo en su sistema de navegación.

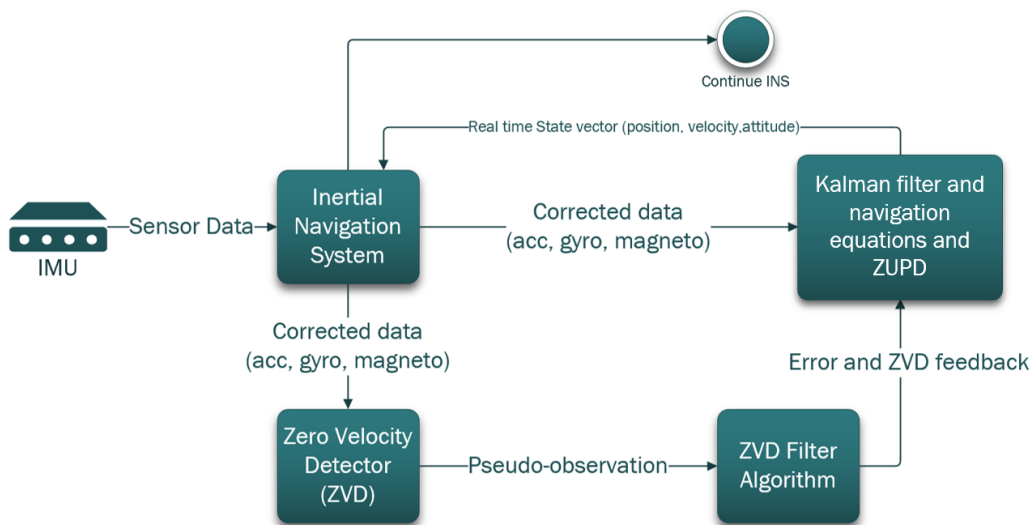


Figura 4.4: Flujo de filtro de Kalman para estimar las variables indirectas posición, velocidad, altitud de un IMU en movimiento en PARKIBIP. Además, integra la compensación de actualizaciones de velocidad cero (ZUPD) mediante la fusión de un algoritmo de detección de velocidad cero (ZVD).

En estas condiciones, PARKIBIP empleará el filtro de Kalman para predecir en tiempo real las variables indirectas velocidad, posición y altitud (variables no medibles por un IMU) a partir de las mediciones de acelerómetro,

giroscopio y magnetómetro. Asimismo, otra entrada al algoritmo, serán los momentos de cero velocidad detectados por el algoritmo de ZVD, que permitirán aplicar el ZUPT dentro del filtro de Kalman para compensar y mejorar aun mas la solución de navegación. El filtro de Kalman no solo estima el error de velocidad en los intervalos ZUPT detectados, sino que también estima las correcciones de posición, altitud, y ruido de los sensores inerciales debido a correlaciones cruzadas en la matriz de transición. La figura Fig. 4.4 brinda un modelo de flujo de aplicación que integra los conceptos mencionados.

4.9. Desarrollo de una aplicación móvil PARKIBIP

Dentro de las expectativas funcionales del proyecto PARKIBIP se encuentra la de brindar una herramienta tecnológica que le permita a los pacientes con la enfermedad de Parkinson una rehabilitación personal; que en la actualidad es inexistente. Asimismo, se espera que dicha herramienta sea un soporte para el terapeuta al momento de tomar decisiones y evaluar al sujeto afectado. De esta manera, se logra una evaluación clínica mas objetiva, que de otra forma se hace dificultosa debido a una serie de limitaciones estructurales, logísticas y financieras.

En la actualidad, los sistemas de análisis de movimiento ópticos basados en cámaras (3D-GA) son reconocidos como el estándar de oro en la medición del movimiento. Sin embargo, requieren ejecutar las pruebas en un entorno de laboratorio con equipos de alto costo, solo se pueden usar para evaluar segmentos cortos de la marcha, requiere personal especializado, equipo considerable -i.e. laboratorio de marcha-, tiempo y sobretodo una gran inversión.

Por consiguiente, se quiere implementar un instrumento portable, accesible y de bajo costo; cuyo usuario final sea un Terapeuta y/o un Parkinsoniano. Primero, mencionado en la sección **Selección tecnológica**, se optó por un dispositivo que cumpla los requerimientos antedichos y sea de gran amigabilidad para su portador (i.e. pequeño, cómodo, portable). Luego, se requiere un elemento de hardware que permita interoperar con el dispositivo, procesar información y presentar resultados al usuario.

En la actualidad, prácticamente todas las personas cuentan con dispositivos inteligentes personales con los cuales interactúan todo el día en sus tareas

cotidianas. Aquí, es donde nace una gran oportunidad para PARKIBIP de combinar la tecnología IMU con los teléfonos inteligentes, aprovechando todo su potencial:

- Permite una comunicación directa y personalizada hacia el usuario
- Ofrece atención continua (i.e. apps están disponibles 24x7)
- Usabilidad. Es una tecnología conocida, el desafío central es la amigabilidad en el diseño hacia el usuario.
- Acceso a periféricos del dispositivo. Por ejemplo el Bluetooth, requisito para enlazar a un IMU
- Dispositivo de procesamiento (variando el rendimiento según las características del mismo)

Finalmente, en consideración de las necesidades de PARKIBIP y los beneficios mencionados, se propone la implementación de una aplicación móvil adecuada para el usuario final (conceptos como usabilidad, rendimiento y automatización).

4.10. Sistema operativo y lenguaje utilizado

La implementación del Software PARKIBIP consistió principalmente en el desarrollo de una aplicación móvil. Una decisión clave para el proyecto fue decidir el sistema operativo a emplear, en el cual se desarrolló esta aplicación. Según la herramienta StatCounter (© 1999-2020 StatCounter), la cual permite consultar estadísticas globales sobre el uso de dispositivos móviles, los dos sistemas operativos dominantes en la industria son Android (70.68 %) y iOS (28.79 %), sumando un total del 99.47 % del mercado entre estas dos plataformas.

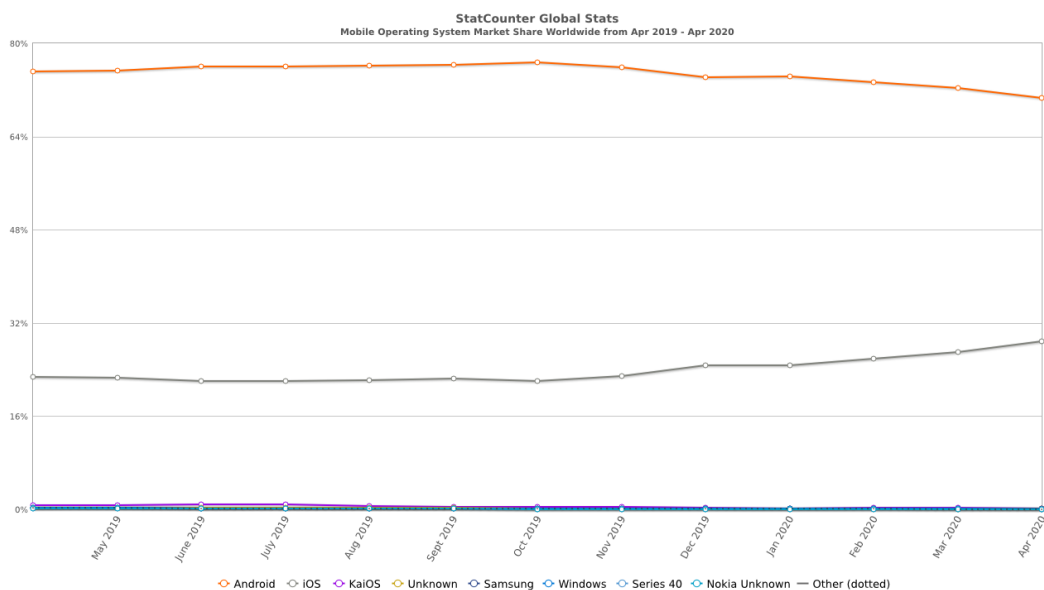


Figura 4.5: Evolución global de sistemas operativos en dispositivos móviles desde 04.2019 a 04.2020 [StatCounter \(2020\)](#)

Uruguay no es la excepción, en donde Android es aún más dominante con respecto al resto de los sistemas operativos. En la figura Fig. 4.6 se puede observar que el 85.41 % de los dispositivos móviles de Uruguay contienen un sistema operativo Android, tan solo un 14.39 % utiliza el sistema iOS y un 0.02 % ejecuta uno diferente.

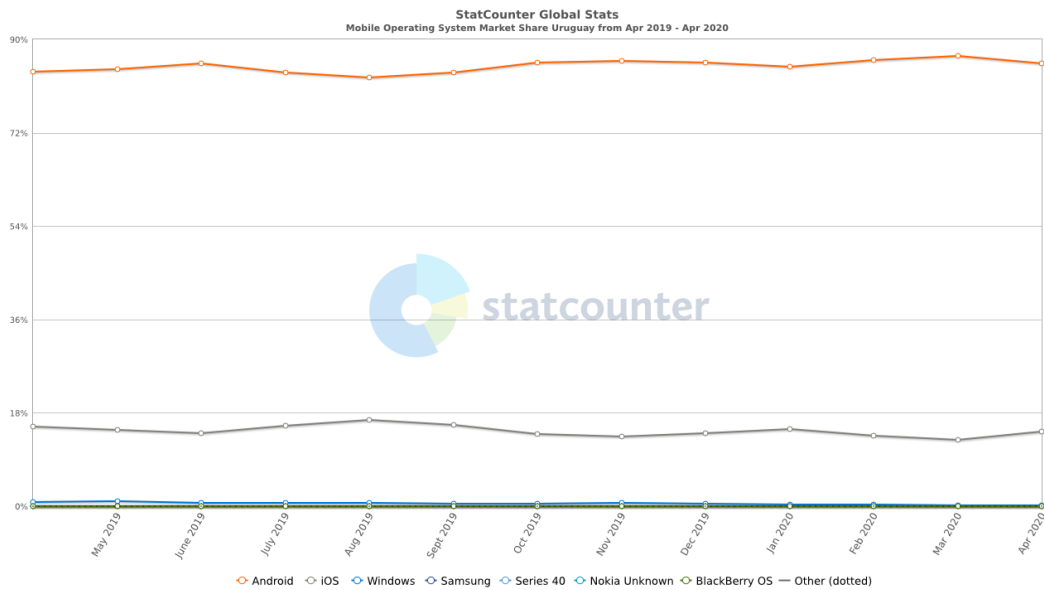


Figura 4.6: Sistemas operativos empleados en dispositivos móviles en Uruguay -periodo 04.2019 a 04.2020- StatCounter (2020)

Con el objetivo de realizar un sistema más accesible para los usuarios, se decidió desarrollar la aplicación para la plataforma Android. Sin embargo, existen diversas tecnologías para el desarrollo de aplicaciones multiplataforma -que luego pueden ser soportadas por ambos sistemas operativos- (p.ej: Flutter, Xamarin, Genexus, etc.). Además, la comunidad de soporte para el uso del dispositivo de medición seleccionado existe para los lenguajes nativos de Android -Java o Kotlin-, lo que facilita la implementación y la solución de inconvenientes durante el desarrollo de la aplicación móvil, sobre todo para la interoperabilidad de la misma con el dispositivo de medición inercial.

4.11. Interfaz y experiencia de usuario (UI & UX)

En la última década, la “experiencia de usuario” (UX) se convirtió en una palabra de gran importancia en el campo de la interacción humano-computadora (HCI) y el diseño de interacción. A medida que la tecnología maduró, los productos interactivos se volvieron no solo más útiles y utilizables, sino también más modernos y fascinantes [Hassenzahl and Tractinsky \(2006\)](#). Se puede decir entonces, que la experiencia de usuario no es nada más ni nada menos que cómo se siente el usuario al interactuar con un sistema de software.

En el área de desarrollo de aplicaciones móviles, la definición de la interfaz de usuario (UI) y la forma en que se va a interactuar con dicha aplicación es una etapa fundamental. El diseño de la interfaz de usuario debe tener en cuenta el público objetivo de dicha aplicación, para que el resultado final sea entendible y accesible para dicho público.

Por lo tanto, a fin de generar un buen diseño de la interfaz de usuario (UI) se deben seguir los lineamientos de diseño que la plataforma sugiere, ya que el usuario de ese dispositivo estará familiarizado con dichos lineamientos. En este sentido, la plataforma Android ofrece una guía de diseño llamada “Material Design”, utilizada por la gran mayoría de las aplicaciones disponibles en la plataforma. Material Design, es un conjunto de guías, componentes y herramientas que engloban las mejores practicas de diseño de UI para Android [LLC \(2020\)](#).

En PARKIBIP, se utilizó Material Design para diseñar tanto la interfaz de usuario como la experiencia de usuario. Para esto, se empleo la herramienta de diseño gráfico y prototipado Sketch (© 2020 Sketch B.V.). Sketch es un editor de vectores gráficos funcional en macOS, desarrollado por la empresa holandesa Boheamian Coding. Es utilizada para el diseño de UI y UX por más de un millón de usuarios en todo el mundo. Además, tiene funcionalidades para prototipado de aplicaciones y colaboración.

4.12. Especificación detallada y alcance del proyecto

El presente proyecto tiene como objetivo la implementación de PARKIBIP, un sistema que le brinde soporte a la toma de decisiones de un terapeuta y a su vez, le permita a los pacientes con la enfermedad de Parkinson una rehabilitación personal. Durante las sesiones de fisioterapia los estímulos externos pueden mejorar las características de la marcha. PARKIBIP entonces busca emular estos estímulos en tiempo real para permitirle al enfermo una rehabilitación personal que prolongue el trabajo del fisioterapeuta en su vida cotidiana. Es decir, reeducar al paciente en rehabilitación mediante feedback-estímulos adecuados y repetitivos- a manipular eventos fisiológicos no detectados de forma voluntaria, logrando una función terapéutica.

Por lo tanto, integrando los avances en dispositivos de análisis de la mar-

cha con los teléfonos inteligentes de uso cotidiano, haciendo uso de la evidencia adquirida a lo largo del tiempo en las terapias de Biofeedback para lograr construir un sistema portable -uso cotidiano y en exteriores-, accesible y de bajo costo; cuyo objetivo sea aumentar el control voluntario sobre los procesos fisiológicos. Mediante un dispositivo electrónico vestible de bajo costo (IMU - Unidad de Medición Inercial), que combina múltiples sensores (p. ej. acelerómetro, giroscopio, magnetómetro, entre otros), se particiona el ciclo de marcha del usuario permitiendo identificar distintos eventos, tales como, el contacto inicial con el suelo o la fase de vuelo. Finalmente, con los eventos detectados, se aplican ciertos algoritmos matemáticos que calculan los parámetros espacio temporales claves, y en base a un módulo clínico se estimula adecuadamente al usuario.

Definir el alcance del proyecto es una actividad central dentro del proceso de planeación del proyecto -también denominada ingeniería de requisitos-, que implica determinar y documentar una lista de objetivos finales, entregables y tareas específicas del proyecto. En otras palabras, es el trabajo a realizar, son las características y funcionalidades que caracterizan un producto, servicio o resultado; define los límites del proyecto, qué características se incluirán e implementarán dentro del mismo.

En software, es habitual definir el alcance de un proyecto mediante la especificación de requisitos de software (SRS, del inglés Software Requirements Specification), siendo una descripción de un sistema a desarrollar. La SRS establece requisitos funcionales y no funcionales y en general incluye un conjunto de historias de usuario o casos de uso que el sistema debe proporcionar.

Una herramienta fundamental para definir el alcance de un trabajo es la historia de usuario, que debe ser completada para cumplir con los requisitos de una funcionalidad del producto o un objetivo del usuario de forma que la misma le brinde un valor particular. Es la unidad de trabajo más pequeña en un marco de desarrollo ágil y se encuentra expresado desde la perspectiva del usuario del software. Es decir, una historia de usuario es una explicación general e informal de una funcionalidad del sistema, escrita desde la perspectiva del usuario final en lenguaje natural. Asimismo, una Épica es una unidad de alto nivel que contiene o desglosa un conjunto de tareas más pequeñas (historias de usuario).

Por lo tanto, se optó por construir la especificación de PARKIBIP en base a la definición de historias de usuarios, y por consiguiente su conjunto es el

alcance de la solución. Algunas de los beneficios adicionales contemplados para emplear esta metodología fueron:

- Una lista de tareas pendientes mantiene al equipo centrado en tareas que deben completarse, sin embargo un conjunto de historias lo mantiene centrado en solucionar problemas para el usuario final.
- Con el objetivo definido en historias, el equipo puede colaborar para decidir cómo implementar la mejor solución, dividir las tareas y cumplir con dicho objetivo.
- Las historias fomentan que el equipo piense de forma crítica y creativa sobre cómo lograr mejor un objetivo.
- Con cada historia el equipo de desarrollo realiza pequeños incrementos hacia el producto final y disfruta pequeños logros, lo que aumenta la motivación.

A continuación, se lista el conjunto de historias de usuario de PARKIBIP junto a sus actores, épica, descripción y nombre descriptivo. Cabe resaltar que el hecho de incluir la Épica y su desglose en historias, permite especificar de forma adicional el WBS (del inglés, Work Breakdown Structure) del proyecto PARKIBIP. La nomenclatura de especificación para cada historia de usuario es: Como < Rol > se Desea < Objetivo > Para < Beneficio >.

Épica	Usuario
Nombre	Autenticación y manejo de Sesión
Actores	Terapeuta — Paciente
Descripción	Como Terapeuta o Paciente previamente registrado se desea acceder al sistema PARKIBIP. Para esto ingresará sus credenciales de acceso y el sistema realizará la validación y autorización correspondiente.

Épica	Usuario
Nombre	Gestión de Roles y Permisos de Usuarios
Actores	PARKIBIP
Descripción	Como Sistema se quiere distinguir a los Usuarios por su Rol. PARKIBIP podrá determinar la información y las funcionalidades que le son permitidas según el Rol del usuario activo y sus autorizaciones asociadas.

Épica	Usuario
Nombre	Visualización de Usuario activo discriminado por Rol
Actores	Terapeuta — Paciente
Descripción	Como Terapeuta o Paciente se quiere visualizar su información de usuario en el Sistema.

Épica	IMU
Nombre	Escaneo de dispositivos Bluetooth
Actores	Terapeuta — Paciente
Descripción	Como Terapeuta o Paciente se desea identificar los dispositivos IMU en su radio cercano, y así iniciar las configuraciones pertinentes para su uso.

Épica	IMU
Nombre	Emparejamiento y sincronización de Dispositivo IMU
Actores	Terapeuta — Paciente
Descripción	Como Terapeuta o Paciente se quiere emparejar un dispositivo IMU previamente escaneado por PARKIBIP, conforme a iniciar sesiones de rehabilitación.

Épica	IMU
Nombre	Establecer sensor de color LED de luz
Actores	PARKIBIP
Descripción	Como Sistema se desea configurar el indicador LED de luz de un dispositivo previamente conectado. De esta manera, se logra distinguir visualmente -en la interfaz gráfica y en el cuerpo del paciente- el dispositivo conectado a cuál extremidad.

Épica	IMU
Nombre	Establecer la tasa de datos de salida y el rango de datos del Sensor Acelerómetro
Actores	PARKIBIP
Descripción	Como Sistema se quiere establecer la frecuencia en Hz de recolección de datos y el rango posible de datos para el sensor Acelerómetro de un IMU sincronizado.

Épica	IMU
Nombre	Establecer la tasa de datos de salida y el rango de datos del Sensor Giroscopio
Actores	PARKIBIP
Descripción	Como Sistema se quiere establecer la frecuencia en Hz de recolección de datos y el rango posible de datos para el sensor Giroscopio de un IMU sincronizado.

Épica	IMU
Nombre	Establecer el rango de datos, ruido, corriente media del Sensor Magnetómetro
Actores	PARKIBIP
Descripción	Como Sistema se quiere establecer la frecuencia en Hz de recolección de datos, el ruido y la corriente media para el sensor Magnetómetro de un IMU sincronizado.

Épica	IMU
Nombre	Iniciar recolección asincrónica de datos por sensor en un IMU
Actores	PARKIBIP
Descripción	Como Sistema se quiere iniciar la recolección de datos crudos por tipo de sensor para un IMU previamente conectado. De acuerdo a las necesidades numéricas, el sistema será capaz de iniciar cada sensor correspondiente para un procesamiento posterior de datos.

Épica	IMU
Nombre	Detener recolección asincrónica de datos por sensor en un IMU
Actores	PARKIBIP
Descripción	Como Sistema se quiere detener la recolección de datos crudos por tipo de sensor para un sensor activo en un IMU previamente conectado. De esta manera, se liberan los recursos asignados en el IMU y en el Host para una posterior recolección.

Épica	IMU
Nombre	Empaquetado de datos de múltiples sensores de un dispositivo IMU
Actores	PARKIBIP
Descripción	Como Sistema se quiere consolidar los diferentes datos recolectados (acelerómetro, giroscopio, magnetómetro) en un único paquete de datos con Timestamp de referencia. El requerimiento de manejar tantos threads como sensores activos, que a su vez tienen frecuencias distintas, incrementa la complejidad de empaquetado de datos. PARKIBIP debe implementar un algoritmo de empaquetado de datos consistente a las distintas frecuencias.

Épica	IMU
Nombre	Administrar conexiones por dispositivo IMU
Actores	PARKIBIP
Descripción	Como Sistema se desea gestionar las diferentes conexiones a los múltiples dispositivos conectados, y así intercambiar distintos comandos con el IMU adecuado.

Épica	IMU
Nombre	Establecer modulo de vibración IMU
Actores	PARKIBIP
Descripción	Como Sistema se quiere configurar el motor vibratorio integrado al IMU, establecer la duración e intensidad de vibración. Siendo el presente modulo uno de los estímulos externos base a emplear en PARKIBIP.

Épica	Configuración
Nombre	Visualización de parámetros y constantes Generales
Actores	Terapeuta — Paciente
Descripción	Como Terapeuta o Paciente se quiere visualizar parámetros generales de fabrica para evaluar dinámicamente su uso en una sesión de rehabilitación.

Épica	Configuración
Nombre	Visualización de parámetros y constantes específicos por Método numérico
Actores	Terapeuta — Paciente
Descripción	Como Terapeuta o Paciente se quiere visualizar información de parámetros específicos por algoritmos.

Épica	Configuración
Nombre	Modificación de parámetros y constantes
Actores	Terapeuta — Paciente
Descripción	Como Terapeuta o Paciente se quiere actualizar parámetros generales y específicos de algoritmos. Permitiendo afinar el comportamiento del algoritmo elegido según las necesidades clínicas del paciente en rehabilitación.

Épica	Configuración
Nombre	Restablecer valores de parámetros y constantes
Actores	Terapeuta — Paciente
Descripción	Como Terapeuta o Paciente se desea actualizar los parámetros generales y específicos de algoritmos a sus valores de fabrica elegidos como óptimos por los desarrolladores.

Épica	Configuración
Nombre	Deshacer cambios de parámetros y constantes
Actores	Terapeuta — Paciente
Descripción	Como Terapeuta o Paciente se desea deshacer cambios involuntarios de parámetros, retornando a la configuración previa.

Épica	Paciente
Nombre	Alta de Pacientes
Actores	Terapeuta
Descripción	Como Terapeuta se quiere dar de alta Pacientes para que puedan participar de sesiones de rehabilitación.

Épica	Paciente
Nombre	Visualización y listado de Pacientes
Actores	Terapeuta
Descripción	Como Terapeuta se desea visualizar la plantilla de Pacientes junto a su información básica.

Épica	Paciente
Nombre	Búsqueda y filtrado de Pacientes
Actores	Terapeuta
Descripción	Como Terapeuta se desea buscar de forma amigable un Paciente mediante filtros conforme a seleccionarlo como usuario activo de Sesiones de rehabilitación.

Épica	Paciente
Nombre	Selección de Paciente activo
Actores	Terapeuta
Descripción	Como Terapeuta, visualizando el listado de Pacientes y haciendo uso del buscador, podrá establecer un Paciente como Usuario activo. El Sistema reconocerá automáticamente al Usuario activo como el participante de una Sesión de rehabilitación.

Épica	Paciente
Nombre	Recuperación de último Estado de Sistema
Actores	PARKIBIP
Descripción	Como Sistema se quiere facilitar y automatizar toda característica de Parkibip de forma que se minimice la selección/digitación por parte del usuario de PARKIBIP. Para ello, el Sistema mantendrá distintos Estados de la aplicación y será capaz de almacenar y recuperar su último Estado (p. ej. Paciente activo, Usuario Logueado, Configuraciones, etc.)

Épica	Sesión
Nombre	Visualización e Historial de Sesiones
Actores	Terapeuta — Paciente
Descripción	Como Terapeuta o Paciente se desea visualizar el Historial de Sesiones de rehabilitación ordenadas de forma descendente por cada fecha de realización. El listado permitirá ver información básica de la Sesión y navegar a su información mas detallada.

Épica	Sesión
Nombre	Búsqueda y filtrado de Sesiones históricas
Actores	Terapeuta — Paciente
Descripción	Como Terapeuta o Paciente se desea buscar de forma amigable cada Sesión dentro del Historial de Sesiones mediante filtros, y así obtener información detallada según necesidad del usuario.

Épica	Sesión
Nombre	Selección de Sesión
Actores	Terapeuta — Paciente
Descripción	Como Terapeuta o Paciente se quiere obtener información particular respecto a una Sesión previamente realizada. Para ello, mediante el uso del buscador podrá filtrar las sesiones de rehabilitación e ingresar a una interfaz que presente información genérica y otra con un nivel de entendimiento técnico.

Épica	Sesión
Nombre	Resumen general de Sesión
Actores	Terapeuta — Paciente
Descripción	Como Terapeuta o Paciente se desea analizar y evaluar cada Sesión de rehabilitación mediante la visualización del comportamiento de los parámetros espaciotemporales de la marcha efectuada. Algunos parámetros son: duración, cadencia, velocidad promedio e instantánea, aceleración promedio e instantánea.

Épica	Sesión
Nombre	Resumen gráfico de Sesión
Actores	Terapeuta — Paciente
Descripción	Como Terapeuta o Paciente se desea analizar y evaluar cada Sesión de rehabilitación de forma gráfica respecto al tiempo (p. ej. velocidad, trayectoria, pasos).

Épica	Sesión activa
Nombre	Iniciar Sesión de Rehabilitación
Actores	Terapeuta — Paciente
Descripción	<p>Como Terapeuta o Paciente se quiere realizar una Sesión de rehabilitación para un Paciente activo. De esta forma, con las pre-condiciones funcionales cubiertas, el Sistema identificará las fases de la marcha y estimulará de acuerdo al protocolo pre-establecido.</p> <p>El usuario de PARKIBIP, únicamente tendrá que iniciar la Sesión, luego PARKIBIP automatizará por completo el flujo de proceso necesario para cumplir el requerimiento bajo el empleo de algoritmos numéricos particulares. Algunas de las pre-condiciones son los dispositivos conectados, paciente activo seleccionado.</p>

Épica	Sesión activa
Nombre	Detener Sesión para Paciente activo
Actores	Terapeuta — Paciente
Descripción	<p>Como Terapeuta o Paciente se desea detener una Sesión de rehabilitación previamente iniciada. El Sistema liberará recursos asignados -rendimiento y consumo energético de dispositivos- y presentara una interfaz con los resultados detallados de la Sesión efectuada.</p>

Épica	Sesión activa
Nombre	Partición del ciclo de la marcha
Actores	PARKIBIP
Descripción	<p>Como Sistema se quiere detectar eficazmente y en tiempo real los eventos asociados a la partición del ciclo de la marcha de cada Paciente (p. ej. HS, TO). Mediante la recolección de datos en tiempo real de los distintos sensores del IMU, y bajo el procesamiento de algoritmos numéricos particulares (existentes o propuestos) el sistema será capaz de detectar la partición del ciclo de la marcha.</p> <p>Para ello, serán requisitos los dispositivos conectados, los sensores asociados establecidos, los dispositivos posicionados en los tobillos del paciente mediante banda elástica y una Sesión de rehabilitación activa recolectando los datos de sensores inerciales.</p>

Épica	Sesión activa
Nombre	Identificación adecuada de estímulos en base a procesamiento de reglas
Actores	PARKIBIP
Descripción	Como Sistema se quiere automatizar la identificación adecuada de estímulos, bajo el soporte de la funcionalidad Partición del ciclo de la marcha, para lograr estimular al Paciente en rehabilitación cuando corresponda.

Épica	Sesión activa
Nombre	Estimular de forma vibratoria al Paciente de acuerdo a protocolo preestablecido
Actores	PARKIBIP
Descripción	Como Sistema se desea estimular mediante vibración el cuerpo del Paciente en rehabilitación, que se encuentra unido al dispositivo IMU durante una Sesión activa. El estímulo externo, será el feedback que reeduce la marcha del sujeto afectado. Para ello, PARKIBIP deberá implementar un Protocolo clínico que indique el instante de estimulación externa.

Épica	Sesión activa
Nombre	Estimular de forma sonora al Paciente de acuerdo a protocolo preestablecido
Actores	PARKIBIP
Descripción	Como Sistema se desea estimular de forma sonora al Paciente en rehabilitación durante una Sesión activa. Idéntico al estímulo vibratorio, PARKIBIP deberá emplear un Protocolo clínico que indique el instante de estimulación externa.

Épica	Sesión activa
Nombre	Computo de parámetros espacio-temporales de interés
Actores	PARKIBIP
Descripción	Como Sistema se desea realizar una estimación en tiempo real de los parámetros espacio-temporales de la marcha durante una sesión de rehabilitación. Mediante la recolección de datos en tiempo real de los distintos sensores del IMU, y bajo el procesamiento de algoritmos numéricos particulares (existentes o propuestos), PARKIBIP será capaz de estimar la duración, la cadencia, la velocidad, la aceleración y la cantidad de pasos en el transcurso de la Sesión. Logrando un resumen objetivo de la marcha del sujeto afectado y siendo el soporte a toma de decisiones posteriores, que de otra forma serían subjetivas.

Épica	Biofeedback
Nombre	Configuración de modo de estímulo: Vibratorio y/o sonoro
Actores	Terapeuta — Paciente
Descripción	Como Terapeuta o Paciente se quiere establecer el formato de estimulación externa al Paciente en rehabilitación; con el fin de maximizar resultados en el aprendizaje de una marcha adecuada.

Épica	Biofeedback
Nombre	Modulo de Biofeedback. Configuración de Protocolo de estímulo
Actores	Terapeuta
Descripción	Como Terapeuta se quiere establecer un formato condicional de estímulos, bajo el desarrollo de reglas clínicas de decisión. Según las necesidades del Paciente, el protocolo indica las necesidades de estímulos para dicho sujeto, indicando el instante temporal, característica disparadora, modo de estímulo. En otras palabras, el Protocolo PARKIBIP es análogo a una receta clínica.

Capítulo 5

Implementación

PARKIBIP es el resultado del trabajo de un equipo interdisciplinario, a partir de la experiencia de fisioterapeutas e ingenieros que definen las reglas para estimular a los pacientes durante la marcha de sujetos afectados por la patología de Parkinson.

PARKIBIP se basa en (i) la detección de las fases del ciclo de la marcha, (ii) el computo en tiempo real de los parámetros espacio-temporales de personas, y (iii) en base a reglas clínicas de decisión, procura estimular al paciente -en forma de vibración y/o sonora- para permitirle una rehabilitación personal que contribuya al trabajo del fisioterapeuta. En otras palabras, bajo la evidencia de las terapias de biofeedback y el trabajo de terapeutas durante sesiones de rehabilitación, se pretende reeducar la marcha del sujeto afectado mediante feedbacks adecuados.

Por lo tanto, para ejecutar un patrón de marcha más efectivo y estimular el proceso de aprendizaje motor se requiere identificar el ciclo de la marcha. Ya mencionado en el apartado **Granularidad de fases de la marcha**, PARKIBIP se orienta a estimar la fase de apoyo y fase de vuelo por cada extremidad. Para ello, empleando algoritmos de Zero Velocity Detection, Kalman Filter y Orientation Filter; PARKIBIP se enfoca en predecir eficazmente los eventos Heel Strike y Toe Off.

5.1. Arquitectura de PARKIBIP

Con el objetivo de enfrentar desafíos clínicos y tecnológicos específicos, PARKIBIP fue diseñado en capas. La solución tecnológica consiste en una ar-

arquitectura completamente portátil y wearable basada en Unidades de Medición Inercial (IMU), conectadas a través del protocolo Bluetooth a un teléfono inteligente Android, que funciona como una plataforma de procesamiento portátil. De esta manera, se diseñó e implemento la arquitectura de sistemas de la Fig. 5.1.

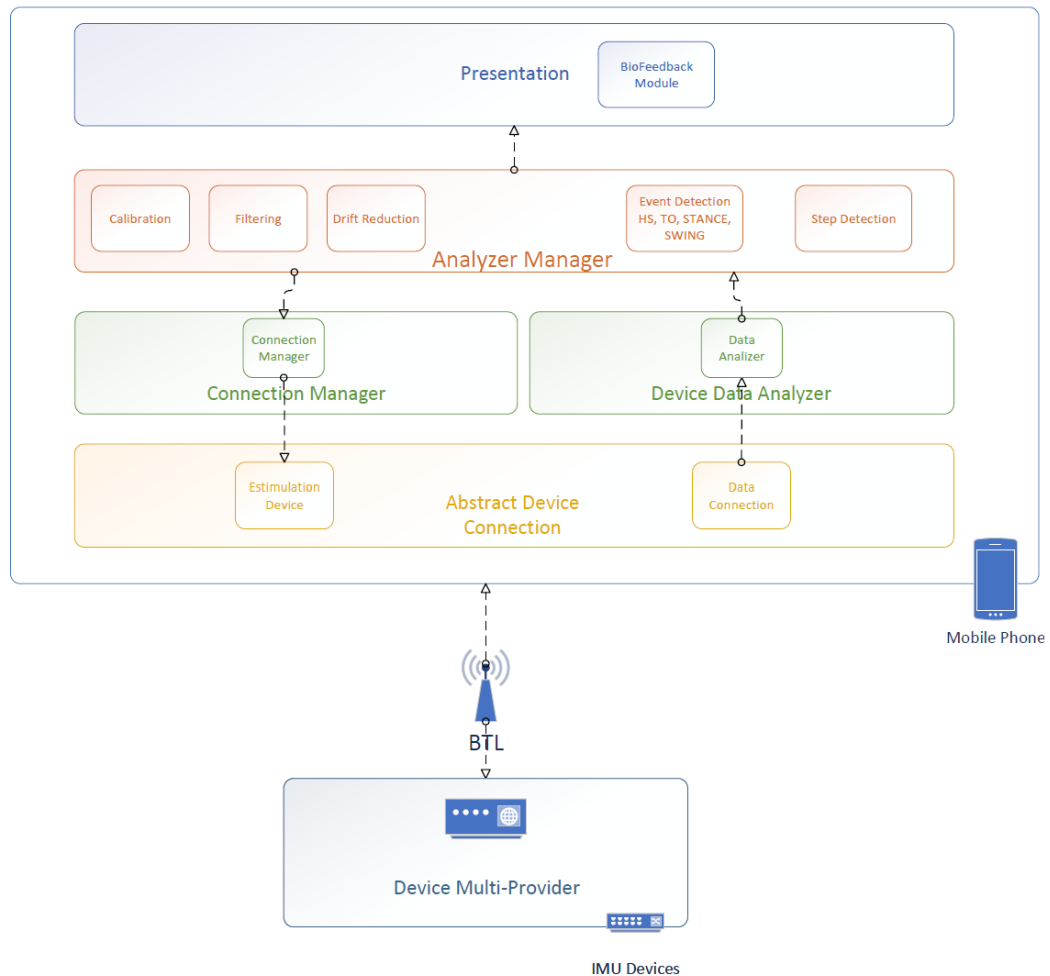


Figura 5.1: Arquitectura del sistema PARKIBIP. Se compone de seis capas bien definidas desde el controlador del hardware hasta la capa de presentación.

En el diseño se pueden identificar seis capas específicas, con responsabilidades bien delineadas. Primero, y externo al Software de la aplicación Android, los dispositivos IMU con conectividad Bluetooth compuestos por los distintos sensores de movimiento. Asimismo, se identifican tres módulos de bajo nivel, vinculados a la conectividad y búsqueda, interacción con los distintos dispositivos y recuperación de datos de los sensores empleados (“Abstract Device Connection”, “Connection manager”, “Device Data Analyzer”). Luego, se en-

cuentra un módulo de procesamiento de la información recopilada, vinculado al análisis y aplicación de algoritmos matemáticos que identifican los parámetros espacio temporales (“Analyzer Manager”). Finalmente, un módulo de presentación de resultados, compuesto por un controlador clínico capaz de tomar decisiones y estimular adecuadamente al usuario.

5.2. Componentes de PARKIBIP

El dispositivo seleccionado en **Benchmark: Elección del dispositivo IMU**, MetaMotionR (MMR), es un dispositivo portátil que ofrece un monitoreo continuo y en tiempo real de los datos de los sensores ambientales y de movimiento. Dentro de las características principales del MMR se encuentran:

- SDK (del inglés, Software Development Kit) en lenguaje JAVA, de código abierto para la adquisición de datos de sensores
- Batería recargable de Litio
- Bajo consumo de energía, el modo de suspensión admite hasta 6 meses de inactividad
- Tamaño diminuto, 27mm × 27mm x 4mm
- Ultraliviano: 5.6 gramos
- 8MB de memoria interna
- Protocolo de transferencia: Bluetooth Low Energy

Los sensores incluidos en el IMU son un giroscopio, un acelerómetro y un magnetómetro (todos de tres ejes); así como un sensor de presión barométrica, un sensor de luz ambiental y un sensor de humedad.

Se implementó una aplicación móvil Android en el lenguaje JAVA que se interconecta a dos dispositivos IMU a través del protocolo estándar Bluetooth Low Energy. El sistema se encuentra diseñado con el propósito de efectuar las siguientes actividades:

1. Escanear dispositivos IMU y enlazarlos a la aplicación móvil para ambos miembros inferiores

2. Calibrar los sensores de los IMU
3. Recuperar los datos acelerómetro, giroscopio y magnetómetro de los respectivos sensores
4. Estimar orientación del dispositivo respecto a una posición inercial mediante un método eficiente del gradiente descendiente
5. Identificar fase de la marcha en tiempo real mediante algoritmos numéricos específicos
6. Estimar parámetros espacio-temporales de la marcha bajo algoritmos numéricos particulares
7. Estimular al usuario de acuerdo a protocolo clínico

5.3. Interoperabilidad con dispositivos IMU

Para estudiar e implementar el modo de operación entre la aplicación Android y los sensores IMU, se optó, en una primera etapa, por desarrollar una prueba de concepto o POC (del inglés, Proof of Concept). Esta prueba de concepto, tenía principalmente los siguientes objetivos:

- Familiarizarse con el escaneo de dispositivos Bluetooth desde la plataforma Android.
- Comprender de qué forma se interactúa desde una aplicación Android con un dispositivo Bluetooth.
- Conocer y utilizar el SDK para interactuar con el dispositivo IMU (MMR).
- Realizar una conexión exitosa con el dispositivo.
- Generar comunicación bidireccional, accionando sobre el dispositivo MMR (e.g. encendiendo la luz led) y recuperando datos desde uno de los sensores (e.g. acelerómetro) del dispositivo.
- Empaquetar o fusionar los datos de acelerómetro, giroscopio y magnetómetro a una frecuencia determinada.

- Escalar el número de dispositivos inerciales conectados y operando en simultáneo.

El sistema operativo Android incluye soporte para el stack de red Bluetooth, permitiendo que un dispositivo intercambie datos de forma inalámbrica con otros dispositivos dentro del alcance del mencionado protocolo. Se proporciona acceso a la funcionalidad Bluetooth a través de un framework que es parte del kit de Android, habilitando a las aplicaciones a conectarse de forma inalámbrica a otros dispositivos Bluetooth. Esto es, funciones inalámbricas de punto a punto y multipunto.

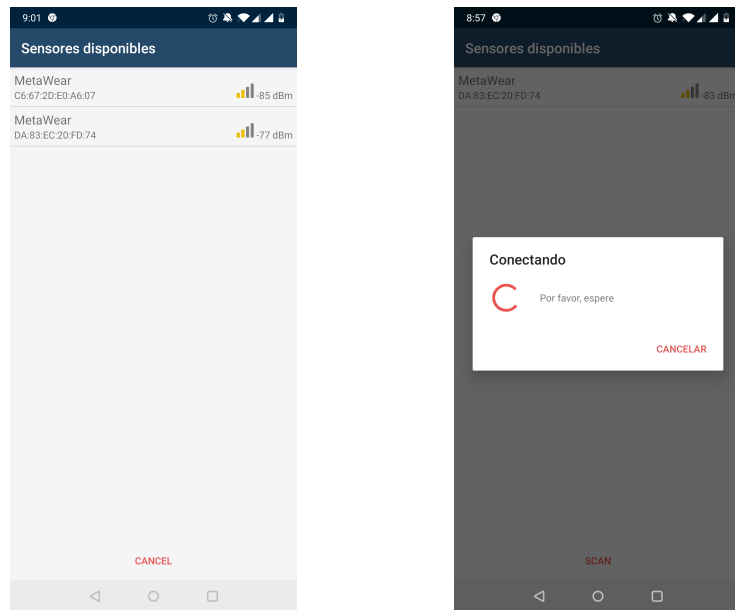
Para utilizar las funcionalidades de Bluetooth, se requiere solicitar autorización al usuario, para esto es necesario declarar los permisos pertinentes en el archivo de configuración de la aplicación, conocido como Manifiesto (en inglés, Manifest).

Uno de los grandes beneficios contemplados en la evaluación del dispositivo seleccionado -MMR del proveedor *MetaWear*-, es el SDK de código abierto para el *MetaWear*. Un kit de desarrollo de software contiene archivos de cabecera, bibliotecas, muestras, documentación y herramientas que simplifican el desarrollo -en este caso la integración y la comunicación con el hardware inercial-, que de otra forma sería muy complejo de efectuar. En general, es provisto por el proveedor del hardware conforme a comunicarse con el sistema embebido en un lenguaje de bajo nivel.

Por lo tanto, se desarrolló la integración con el hardware inercial, mediante la incorporación de la SDK de *MetaWear* al proyecto Android, a través del gestor de dependencias remotas GRADLE. Luego, fue necesario configurar e inicializar adecuadamente el servicio *BtleService*, con el fin de escanear y enlazar dispositivos existente dentro del rango Bluetooth -ver diseño en la Fig. 5.2-. Con los dispositivos enlazados correctamente, es factible comenzar a interactuar mediante diversas funciones de la interfaz.

A continuación, se listan los módulos y algunas de las funciones principales que se emplearon en PARKIBIP:

1. Bluetooth LE Connection: Gestión de la conexión entre aplicación Android y dispositivos IMU.
 - `connectAsync`: Establecimiento de conexión a un dispositivo.
 - `disconnectAsync`: Remover la conexión a un dispositivo.



(a) Escaneo de dispositivos IMU. Previa activación del módulo bluetooth, expone un listado de IMU en la cercanía. Para cada uno muestra su dirección MAC, el tipo de IMU y la intensidad de la señal. (b) Conexión en curso a un dispositivo IMU. Finalizada el proceso, el IMU es conectado exitosamente a PARKIBIP.

Figura 5.2: Proceso de escaneo previa a conexión de un IMU. Mediante el protocolo Bluetooth.

- `UnexpectedDisconnect`: Administración de eventos de desconexión inesperados.
- `readDeviceInformationAsync`: Obtener nivel de batería e información general del dispositivo.
- `tearDown`: Remueve los recursos asignados por el firmware del dispositivo.

2. *DataProducer*: Componentes que recuperan datos, ya sean de sensores o de funciones del firmware.

- *AsyncDataProducer*: Los productores de datos asincrónicos, cuando están activos, miden constantemente los datos en BACKGROUND y los envían cuando hay nuevos datos disponibles.
- `start`: Inicia un productor de datos previamente configurado (e.g. sensor acelerómetro).

- stop: Remueve un productor de datos sin remover la conexión al hardware.
3. *DataRoute*: Las rutas de datos proporcionan una forma simple y compacta para el acceso a las funciones avanzadas de *MetaWear*, como el registro *Log*, el procesamiento de datos y el manejo de eventos a bordo.
- addRouteAsync: creación de rutas de eventos, que definen cómo fluyen los datos desde un productor a diferentes puntos finales (a través de un *RouteComponent*).
 - Data: Los datos creados por los productores de datos están representados por la interfaz “Datos”, que encapsula atributos de clave-valor, como la hora en que fueron creados los datos (timestamp) y el valor de la muestra de datos.
 - stream: Permite el procesamiento del flujo de datos y como serán transformados los mismos. Establece un suscriptor al evento del productor de datos.
 - log: Habilita a grabar datos en la memoria flash incorporada y recuperarlos en un momento posterior.
 - state: Permite recuperar el vector de estados del dispositivo.
 - buffer: Los almacenes son unidades que graban la entrada más reciente en su estado interno, y posteriormente permiten acceder mediante el método de estado (*state*) perteneciente al módulo procesador de datos.
4. *Function Module*: Existen tantos módulos como sensores. Se listan funcionalidades genéricas a los módulos (Acelerómetro, giroscopio, magnetómetro, barómetro, luz ambiental, entre otros).
- start: Establece el uso de un módulo específico.
 - stop: Remueve el uso de un módulo específico.
 - configure: Configuraciones particulares a cada módulo.
 - odr: Tasa de datos de salida.
 - range: Rango de datos.

Es importante notar que, algunos de los nombres de los métodos disponibles finalizan con la palabra “Async”, y significa que toda comunicación es asincrónica con el dispositivo. Esto es un requisito a nivel del sistema operativo y es una característica esencial del sistema PARKIBIP, ya que el acceso a los datos del dispositivo no puede ser bloqueante.

De esta manera, PARKIBIP es un sistema multihilo (multithreading en inglés), teniendo la capacidad de ejecutar eficientemente múltiples hilos de ejecución y paralelizar diversas tareas en background, como por ejemplo los mensajes hacia o desde un dispositivo. Asimismo, esta tarea se complejiza aún mas, al escalar en número los dispositivos inerciales que puede gestionar el sistema. Entonces, se implementaron diversos hilos, para lograr:

- Gestionar múltiples conexiones a dispositivos inerciales.
- Interactuar con los dispositivos mediante el envío y recepción de mensajes apropiados.

Se resalta la necesidad de diseñar e implementar diferentes estrategias para el control de concurrencia sobre recursos compartidos de los diversos Threads en simultanea ejecución.

5.4. Administración de las conexiones

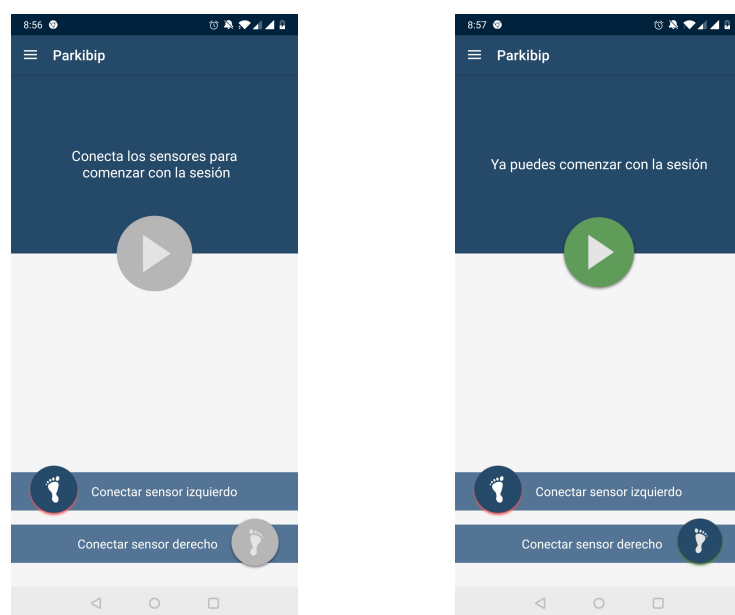
Como se resaltó en la sección anterior **Interoperabilidad con dispositivos IMU**, las conexiones entre la aplicación Android y un dispositivo Bluetooth se representan en una aplicación Android mediante la clase perteneciente al SDK de Android llamada *BluetoothDevice*. Esta clase representa un dispositivo remoto conectado por Bluetooth, permitiendo a la aplicación comunicarse con el dispositivo, operar sobre él y requerirle datos como el nombre, dirección, clase, estado, batería, etc.

Esta clase es realmente un fino WRAPPER delgado para una dirección de hardware Bluetooth. Los objetos de esta clase son inmutables. Las operaciones en esta clase se realizan en la dirección de hardware Bluetooth remota, utilizando la clase *BluetoothAdapter*.

Para obtener un dispositivo Bluetooth, se utiliza la clase *BluetoothAdapter*, con el método *getRemoteDevice(String)*, la cual crea una instancia de *BluetoothDevice* que representa un dispositivo de una dirección MAC conocida (que

se puede obtener a través del descubrimiento de dispositivos también utilizando *BluetoothAdapter*). Este descubrimiento se realiza en una pantalla dedicada dentro de la aplicación (en android representada con la clase *Activity*), la cual llamamos *ScannerActivity*.

En la pantalla inicial de la aplicación, llamada *MainActivity*, se le muestran al usuario botones para conectar cada uno de los dispositivos necesarios para el funcionamiento de PARKIBIP, uno para el pie izquierdo y otro para el pie derecho -ver Fig. 5.3-.



(a) Conexión exitosa del IMU asociado al pie izquierdo -botón coloreado de azul-. PARKIBIP identifica al Pie/IMU con un color en la aplicación, en la imagen rojo. Es el mismo color que comienza a parpadear en el IMU mediante su luz LED.

(b) Conexiones a los IMU exitosa. Ambos botones se encuentra en azul, las luces LED del IMU sincronizadas a los colores de los contornos de los botones, y el botón central es puesto en verde. PARKIBIP se encuentra listo para comenzar una sesión de rehabilitación.

Figura 5.3: Conexiones a los dispositivos IMU mediante el protocolo Bluetooth.

Al seleccionar uno de estos botones se le muestra al usuario la pantalla de escaneo y selección de dispositivos IMU. Es importante destacar que el descubrimiento de dispositivos Bluetooth lo realizamos en esta pantalla filtrando la clase específica de dispositivos que requiere Parkibip, para que solo se listen dispositivos Bluetooth del tipo IMU y no algún otro dispositivo que pueda estar cerca del Smartphone. Una vez que la conexión se establece, la pantalla principal muestra un indicador de que el sensor para ese pie se encuentra

conectado. Este indicador tiene un color específico que también se muestra a través de una luz Led en el sensor MMR (color rojo para el pie izquierdo y color verde para el pie derecho) que permite identificar qué sensor se debe colocar en cada pie.

En el listado de cada uno de los dispositivos se muestra información pertinente sobre los dispositivos encontrados, como lo es su dirección MAC, el tipo de dispositivo, la intensidad de la señal (importante para el buen funcionamiento de la aplicación), entre otras cosas.

Cuando el usuario selecciona un dispositivo, se realiza la conexión con el dispositivo y se obtiene una instancia de la clase *BluetoothDevice* que utilizaremos para obtener los datos del dispositivo mientras el usuario realice la actividad. Resulta importante entonces proveer un mecanismo para administrar estas conexiones, mantener las instancias, identificarlas, liberarlas cuando sea necesario y que se realice su re-conexión en caso de que sea interrumpida. Con todos estos objetivos diseñamos el sistema como se muestra en la figura Fig. 5.4

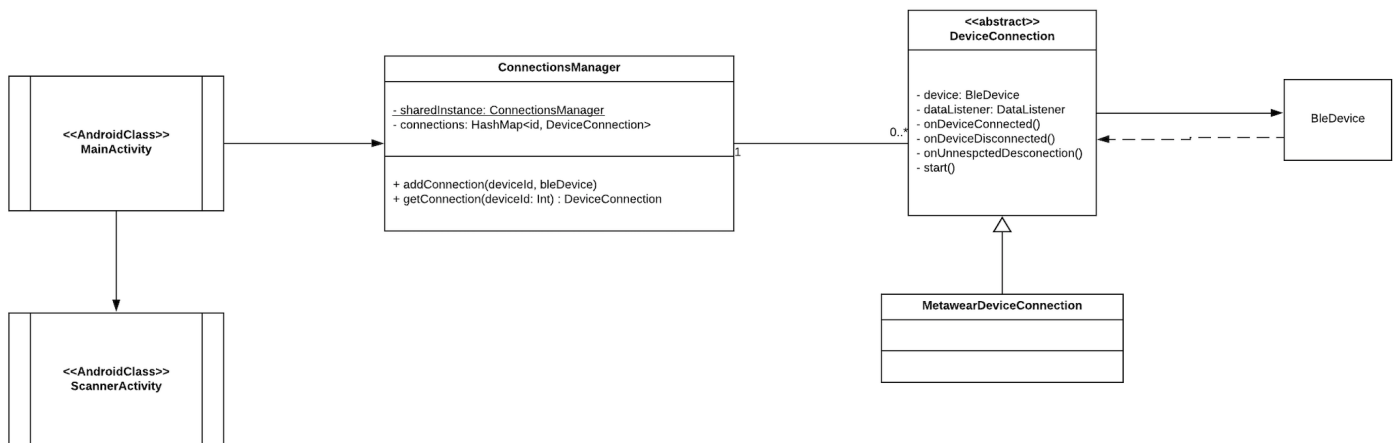


Figura 5.4: Diagrama de clases del módulo de la aplicación Android donde se administran las conexiones Bluetooth con los dispositivos IMU

A continuación se describen las principales características del diseño, la interacción entre las clases y sus roles:

- La pantalla principal representada con la clase *MainActivity* es encargada de crear y mostrar la pantalla de escaneo y selección de dispositivos (*ScannerActivity*).

- En la pantalla *ScannerActivity* el usuario selecciona el dispositivo y se crea la instancia de *BluetoothDevice*.
- La pantalla *MainActivity* recibe la instancia de *BluetoothDevice* y se la provee a *ConnectionsManager*.
- *ConnectionsManager* es un *Singleton* con la única responsabilidad de administrar las conexiones existentes, guardadas en un Diccionario Clave-Valor (utilizando como clave un identificador de tipo *String*).
- Cada conexión es mantenida por una instancia de la clase *DeviceConnection*, una clase abstracta que representa el concepto de conexión con un dispositivo IMU. Estos objetos tienen la capacidad de mantener la conexión activa, solicitarle los datos al sensor, terminar el flujo, reconectarlo en caso de que la conexión sea interrumpida, entre otras cosas.
- Para el dispositivo MetaMotionR se implementó una clase concreta de *DeviceConnection* llamada *MetaWearDeviceConnection*. La misma tiene una implementación concreta con las particularidades requeridas para comunicarse con este tipo de dispositivos, utilizando el SDK de MetaWear mencionado anteriormente.

El diagrama de comunicación de la Fig. 5.5 detalla las distintas interacciones para el caso de uso “Establecer conexión con dispositivo IMU”.

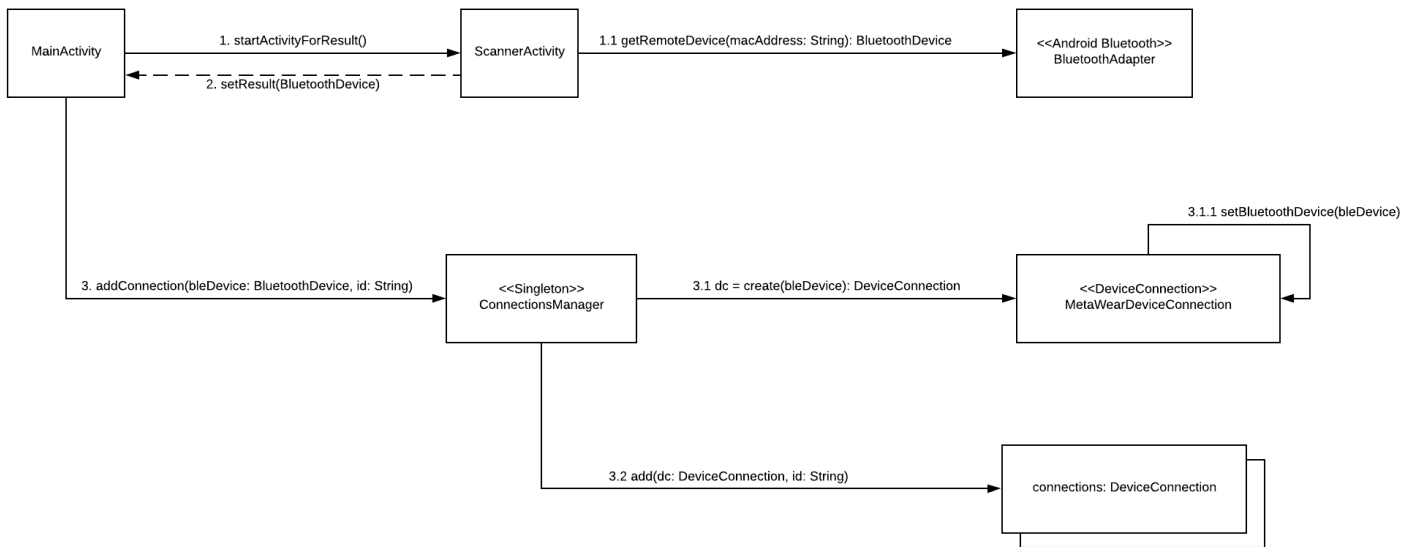


Figura 5.5: Diagrama de comunicación para el caso de uso “Establecer conexión con dispositivo IMU”

Este diseño nos provee de varias ventajas, a continuación nombramos algunas de éstas:

- Las responsabilidades de cada clase están claras.
- Cada clase tiene una única responsabilidad.
- Agregar un nuevo tipo de dispositivo IMU -ya sea de otro proveedor de hardware, así como otro modelo de MbienLab- implicaría simplemente implementar una nueva clase concreta de la clase abstracta *DeviceConnection*, lo que hace que el sistema sea escalable.
- Si en un futuro se quisiera utilizar más de dos dispositivos, este diseño funciona sin grandes cambios, ya que el manejador de conexiones *ConnectionManager* puede mantener tantas conexiones como sea necesario
- Este diseño soporta mantener conexiones con distintos tipos de dispositivos o de diferentes marcas de forma simultánea.

5.5. Algoritmo de empaquetado de datos PARKIBIP

Ya mencionado, la recopilación de datos de múltiples sensores de un dispositivo inercial, se realiza en diversos hilos de ejecución. Además, cada sensor opera con tasas de frecuencias eventualmente distintas -Hz-. Es fundamental combinar los datos recibidos -en hilos y frecuencias distintas- de manera adecuada, tal que los datos se mantengan consistentes.

Por lo tanto, se implementó un algoritmo de empaquetado de datos, con el objetivo de combinar valores de los sensores acelerómetro, giroscopio y magnetómetro en un único mensaje en cada instante de tiempo. Al fusionar varias fuentes de datos, es una restricción muestrear a la misma frecuencia, o al menos, múltiplos enteros de la frecuencia más rápida.

El algoritmo desarrollado opera de la siguiente manera:

1. Siempre se propagan los mensajes fusionados según la frecuencia mas alta
2. El muestreo de fuentes de datos en las frecuencias más bajas repetirá el último valor recibido
3. Se emplean dos colas de almacenamiento de mensajes, en modalidad de buffer FIFO (del inglés first-in, first-out), para las fuentes de datos cuyas frecuencias son mas bajas

A continuación, se presenta el diagrama dado por la figura Fig. 5.6, el cual ejemplifica el procedimiento del fusión de datos.

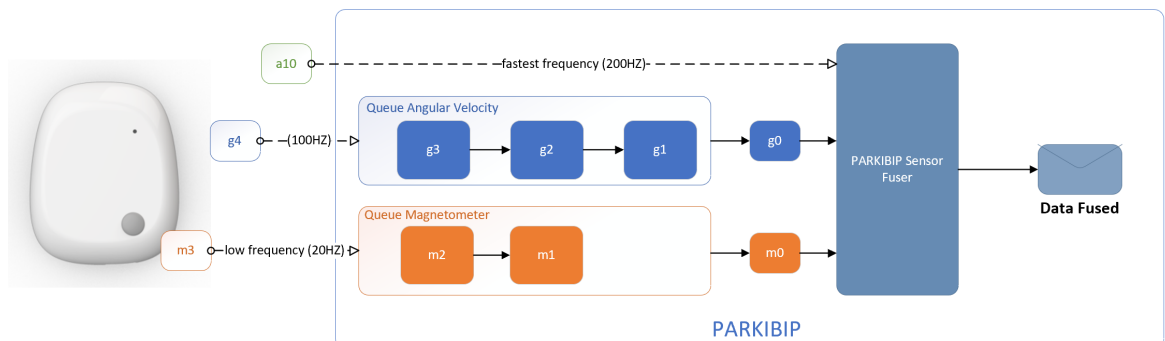


Figura 5.6: Algoritmo de empaquetado de datos PARKIBIP. Presenta un caso de uso de recopilación de datos de los sensores acelerómetro, giroscopio y magnetómetro en las frecuencias correspondientes 200HZ, 100HZ, 20HZ.

5.6. Módulo de algoritmos numéricos y modelos de representación

Puesto que PARKIBIP es un sistema físico-matemático complejo, el cual integra múltiples métodos numéricos y estadísticos; se optó por desarrollar un módulo destinado a la provisión de funciones independientes que aborden los problemas frecuentes. Este módulo, denominado “API Algorithm”, tiene la responsabilidad de encapsular funciones reutilizables y particulares a la solución, para que luego sean accesibles desde cualquier método que las requiera.

En primer lugar, se introducen los principales modelos de representación que fueron empleados y gestionados dentro de la Interfaz de Programación de Aplicaciones (API, Application Programming Interface) de algoritmos. Conforme a representar la orientación del dispositivo en tiempo real, es necesario comprender los conceptos ángulos de euler (del inglés, Euler angles) y cuaternión (del inglés, Quaternion). Ambos modelos proporcionan una forma de representar la orientación de un cuerpo tridimensional mediante una combinación de tres rotaciones sobre diferentes ejes. Para el caso de los ángulos de euler, la secuencia de rotaciones utilizadas para representar una orientación dada es (i) yaw -rotación sobre el eje z un ángulo ψ -, (ii) pitch -rotación alrededor del eje Y un ángulo θ -, (iii) roll -rotación alrededor del eje X un ángulo ϕ -. Por otro lado, un quaternion es un vector de cuatro componentes, un elemento real y tres elementos complejos. Por ejemplo, $q_i^b = (a, bi, cj, dk)$ definido como el vector unitario que codifica la rotación desde el marco inercial hacia el marco del sensor. Luego, ambas representaciones pueden ser utilizadas para construir una matriz de rotación $R_{3 \times 3}$ para realizar la rotación en una sola operación de multiplicación de matrices.

En vista de los modelos de representación introducidos y las necesidades operatorias del sistema, se implementaron las funciones necesarias para manipular y operar con éstos modelos de representación. En seguida, se listan algunas funciones desarrolladas en la API de algoritmos:

1. Medición

- `adjustMeasurementUnities(MeasurementModel measurementModel)`: Estandariza las unidades de las mediciones hacia los sistemas internacionales de unidades m/s^2 para aceleraciones y radianes para el caso de velocidades angulares.

- `buildMeasurementWithData(Data data)`: Convierte una instancia genérica de la clase `Data` retornada por el IMU a su correspondiente modelo en PARKIBIP (`MeasurementModel`).
- `buildMeasurement(Acceleration acceleration, AngularVelocity angularVelocity, MagneticField magneticField, Calendar timestamp)`: Crea una instancia lógica de `MeasurementModel` con el timestamp adecuado, a partir de los vectores tridimensionales acelerómetro, giroscopio y magnetómetro.

2. Filtro de Orientación

- `updateOrientation(MeasurementModel measure, boolean hasMg)`: Función responsable de ejecutar el filtro de orientación implementado y retornar el nuevo quaternion. El parámetro *hasMg*, indica la existencia de una observación del tipo magnetómetro.

3. Aceleración

- `removeGravityForces`: Método responsable de la extracción de la aceleración instantánea física del sensor.
- `getUserAcceleration(MeasurementModel measure)`: Función encargada de obtener la aceleración del usuario en el marco inercial, resultado de la transformación de marcos y remoción de fuerzas gravitatorias.
- `linearAccelerationMethod(MeasurementModel measure)`: Algoritmo responsable de computar la aceleración lineal instantánea del sensor.

4. Giroscopio

- `deg2rad(float degrees)`: Convierte el valor de grados a radianes para los datos del giroscopio.
- `rad2Deg(float radians)`: Convierte el valor de radianes a grados para los datos del giroscopio.

5. Orientación

- `quaternionProd(QuaternionModel q1, QuaternionModel q2)`: Función responsable de calcular el producto de dos cuaterniones.

- `vectorFromRotationMatrix(RotationMatrix r, MeasurementModel measure)`: Dada una medición en el marco del sensor, aplica la Rotación por parámetro, para trasladar el vector a un marco de referencia inercial.
- `vectorFromQuaternionProd(QuaternionModel q, MeasurementModel measure)`: Dada una medición en el marco del sensor, aplica la rotación dada por la orientación en formato quaternion, para trasladar el vector a un marco de referencia inercial.
- `convertToRotationMatrix(QuaternionModel q)`: Convierte un quaternion a su correspondiente matriz de rotación.
- `convertToEulerAngles(QuaternionModel q)`: Convierte un quaternion a sus correspondientes ángulos de euler.
- `getQuaternion()`: Retorna el ultimo quaternion calculado por el filtro de orientación.
- `dcm2q(double[][] rot)`: Convierte una matriz de cosenos de dirección al respectivo quaternion.
- `rt2b(RealVector attitudeVector)`: Calcula y retorna la matriz de rotación desde un marco inercial al marco del sensor, a partir de los ángulos de euler.

6. Calibración

- `calibrateQuaternion(MeasurementModel measurementModel)`: Método iterativo responsable de calibrar adecuadamente la orientación inicial del dispositivo IMU.
- `stopCalibrateQuaternion()`: Interrumpe un proceso de calibración retornando el ultimo quaternion computado.
- `hasToStopCalibrate()`: Chequea si es necesario interrumpir un procedimiento de calibración de orientación previamente iniciado. En caso de requerirse, retorna verdadero.
- `hasToCalibrate()`: Chequea si es necesario efectuar una calibración de orientación. En caso de requerirse, retorna verdadero.

7. Timestamp

- `getDiffSeconds(Calendar timestampOld, Calendar timestampNew)`:
Obtiene la diferencia entre dos marcas de tiempos del tipo `Calendar`.

Finalmente, para llevar adelante las distintas representaciones de matrices y vectores, así como el álgebra subyacente, se incluyó la librería de la compañía Apache, referida como “Apache Commons Math” [Commons \(2020\)](#). La misma proporciona diversas funciones ligeras en el lenguaje de programación JAVA, representaciones numéricas escalables (e.g. naturales, reales, precisión simple y doble); por ende, facilita su utilización y reutilización independientemente de la complejidad requerida.

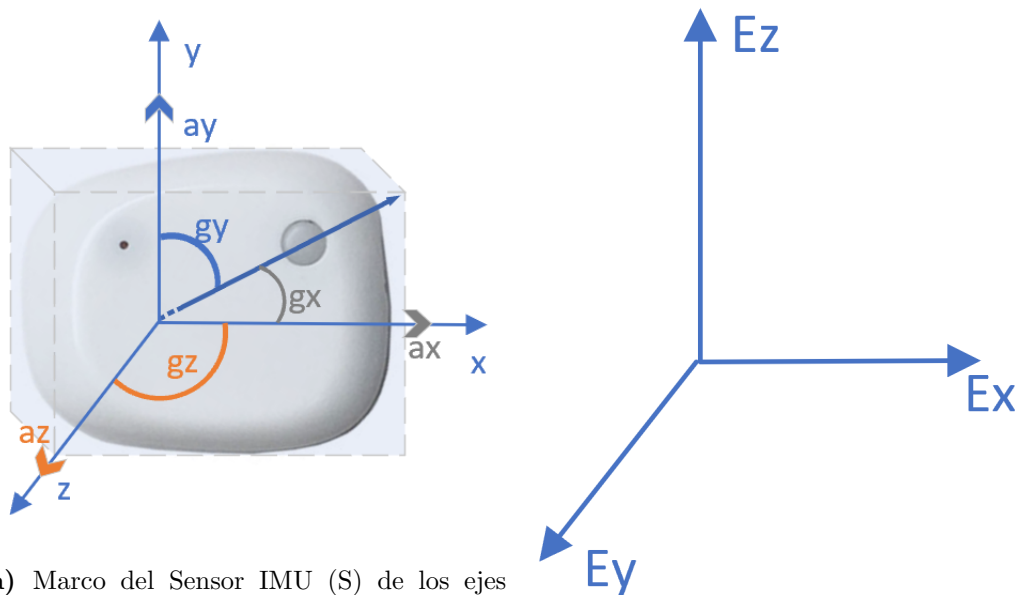
En este sentido, es importante notar la complejidad de PARKIBIP relativa a la precisión numérica requerida, es decir, operaciones matriciales complejas y con valores numéricos que descienden aproximadamente a 10^{-22} .

5.7. Sistemas de coordenadas de referencias

Para lograr analizar la marcha de personas, así como también comprender el significado de las mediciones, es necesario identificar los sistemas de coordenadas que intervienen en el proceso.

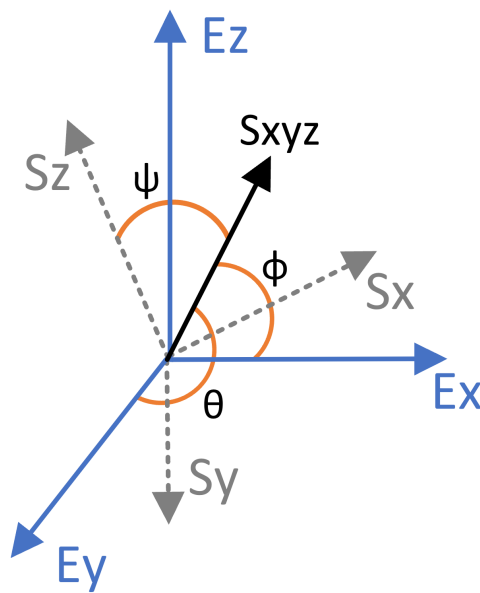
Puesto que los dispositivos inerciales IMU son colocados en los tobillos de los sujetos, existen tres sistemas de coordenadas: el marco de referencia Pie -que describe la rotación del pie-, el marco Sensor -que describe el movimiento del dispositivo inercial- y el marco fijo Inercial o Terrestre -ver figura Fig. 5.7-. Dado que el IMU se encuentra fijado al pie del sujeto mediante bandas elásticas de velcro, se asume que el dispositivo no se desliza ni se mueve durante la marcha. Por lo tanto, se considera que el sistema de coordenadas del pie es igual al sistema de coordenadas del Sensor.

El procedimiento consiste en recopilar las observaciones recibidas desde los distintos sensores, estimar la orientación de cada dispositivo en tiempo real; y luego transferir -mediante cuaterniones o rotaciones- el marco del sensor al marco inercial de la Tierra. Como consecuencia, es posible operar y realizar un análisis de la marcha desde un marco de referencia fijo, estandarizado y conocido.



(a) Marco del Sensor IMU (S) de los ejes del acelerómetro y giroscopio, según especificación.

(b) Marco inercial Tierra (E).



(c) La orientación del marco E se logra mediante una rotación, desde la alineación con el marco S, de un ángulo de $(\phi, \theta, \psi) = (\text{roll}, \text{yaw}, \text{pitch})$ alrededor del eje S_{xyz} .

Figura 5.7: Tres ejes de coordenadas dimensionales del IMU. Sistemas de coordenadas de referencia del Sensor ($S_X S_Y S_Z$) e Inercial ($E_X E_Y E_Z$), traslación del vector en S hacia E, mediante rotación de ángulos.

5.8. Algoritmo Filtro de Orientación

Para estimar la orientación de los dispositivos respecto a un sistema de referencia inercial, se decide aplicar la técnica de *gradiente descendiente optimizado y derivado analíticamente* sugerida por S. Madgwick -toma de decisión en **Filtro de Orientación**-, descrita en la figura FIG. 5.10 tomada de Madgwick (2010). El método, presenta grandes ventajas, logrando excelentes resultados en la estimación y siendo computacionalmente eficiente.

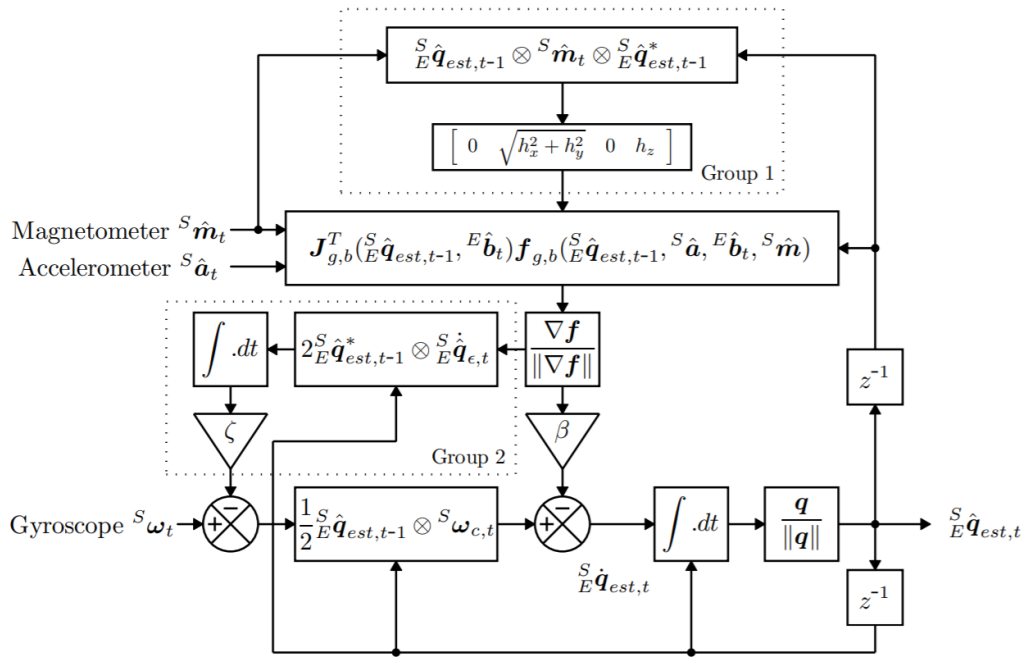


Figura 5.8: Filtro de orientación con distorsión magnética. Extraído de S. Madgwick Madgwick (2010).

En primer lugar, se realizó un relevamiento respecto a implementaciones de código abierto del algoritmo en el mismo lenguaje de desarrollo que PARKIBIP -de forma que se pueda reutilizar directamente-. No obstante, no se encontró una implementación oficial o alternativa en el lenguaje JAVA, y si se obtuvo un ejemplo en el lenguaje *Matlab* que sirvió como guía en el desarrollo y pruebas funcionales de PARKIBIP.

Por lo tanto, fue implementado en Android -JAVA- el flujo descrito en la figura Fig. 5.8 combinando los datos sensados por el acelerómetro, giroscopio y magnetómetro del dispositivo IMU.

Mencionado en las secciones **Módulo de algoritmos numéricos y modelos de representación** y **Sistemas de coordenadas de referencias**, existen varias formas

de representar la información de orientación de un cuerpo, como por ejemplo los ángulos de Euler o los cuaterniones. Un cuaternión es un número complejo de cuatro dimensiones que se puede utilizar para representar la orientación de un cuerpo o un marco de coordenadas en un espacio tridimensional. Si bien son menos intuitivos que los ángulos de Euler y las matemáticas pueden ser un poco más complejas, presentan ciertas ventajas -entre ellas, no padece “Gimbal lock”, que impide medir la orientación cuando el ángulo de inclinación se acerca a los +/- 90 grados-, y por lo tanto son empleados en la implementación de PARKIBIP.

Es crucial identificar adecuadamente los marcos de referencias y cómo los mismos son interpretados para obtener resultados esperados. Así, se utilizó un sistema de notación de superíndices y subíndices adoptado en Merat (1987), para denotar los marcos relativos de orientaciones y vectores. Un subíndice denota el marco que se está describiendo y un superíndice denota el marco al que se hace referencia. Por ejemplo, q_B^A describe la orientación del marco B relativo al marco A.

Como se puede apreciar en el modelo 5.8, las observaciones acelerómetro, giroscopio y magnetómetro en el sistema de coordenadas del sensor, actúan como entrada al algoritmo numérico. A partir de dichos datos y el estado del cuaternión previo, se computa la orientación completa del sensor respecto al marco de referencia inercial Tierra.

Tabla 5.1: Descripción, notación y tipo -entrada o salida- de los parámetros empleados por el algoritmo filtro de orientación.

Descripción	Parámetro	Notación
Magnetómetro en el marco del sensor instante t	Entrada	m_t^s
Acelerómetro en el marco del sensor instante t	Entrada	a_t^s
Giroscopio en el marco del sensor instante t	Entrada	g_t^s
Orientación de la Tierra relativa al sensor	Salida	${}^S_E \hat{q}_{est,t}$

Finalmente, con la estimación de la orientación completa, es posible rotar un vector de un marco de referencia a otro, particularmente las observaciones tridimensionales sensadas por el IMU. Logrando de esta manera, estandarizar los concepto para un posterior análisis. Un posible procedimiento de rotación de un vector en el marco del Sensor a uno inercial -empleando propiedades particulares de cuaterniones, e.g. el cuaternión conjugado es el inverso de los marcos $q_S^E = (q_E^S)^{-1}$ - es:

$$v_E = q_S^E \cdot v_S \cdot q_E^S$$

V_S : vector en el marco del Sensor

V_T : vector en el marco inercial

5.9. Extracción de la aceleración física del sensor

En general, las estimaciones de posición y velocidad basadas en acelerómetros de sensores de bajo costo son muy malas e inutilizables. Esto se debe a que la orientación del sensor debe conocerse con un alto grado de precisión para que las mediciones de gravedad se puedan distinguir de la aceleración física del sensor. Ya mencionado, pequeños errores en la estimación de la orientación producirán errores extremadamente altos en la aceleración medida, lo que se traduce en errores aún mayores en las estimaciones de velocidad y posición. Aunque en PARKIBIP el calculo de la posición y velocidad no dependerán únicamente del acelerómetro para mejorar su eficacia -también del giroscopio y magnetómetro-; es crucial eliminar la aceleración gravitatoria incluida en la aceleración sensada.

Entonces, para convertir la medición del acelerómetro en la aceleración física real del IMU -o aceleración del usuario-, es importante comprender exactamente lo que mide éste sensor. En este sentido, el acelerómetro mide tanto la aceleración física del sensor como la contribución de las fuerzas normales que evitan que el acelerómetro acelere hacia el centro de la Tierra. Para medir solamente las componentes de aceleración que son causadas por la aceleración física, las fuerzas normales debe ser removidas. Es decir:

Sea am_S la aceleración del movimiento sensada en el marco del Sensor, au_E es la aceleración real del dispositivo en el cuerpo -del usuario-, g_E el vector de componentes gravitatorias en el marco inercial, y R_E^S es la matriz de rotación desde el marco inercial a el marco del Sensor:

$$am_S = au_S - R_E^S \cdot \vec{g}, \quad \vec{g} = \begin{pmatrix} 0 \\ 0 \\ g \end{pmatrix}$$

Conforme a eliminar las componentes gravitatorias de la medición de aceleración y obtener la aceleración del usuario, se deduce que:

$$au_S = am_S + R_E^S \cdot \vec{g}$$

Obteniendo au_E , la aceleración del movimiento del usuario en el sistema de coordenadas inerciales:

$$au_E = R_S^E \cdot am_S + \vec{g}$$

Cabe resaltar que, la matriz de rotación (R_S^E) resulta de una transformación del cuaternión calculado con el algoritmo filtro de orientación (q_E^S) **Algoritmo Filtro de Orientación**. Alternativamente, se podría emplear el producto de cuaterniones para transformar un vector desde un marco a otro de referencias:

$$au_E = q_S^E \cdot am_S \cdot q_E^S + \vec{g}$$

Finalmente, se realiza la conversión desde unidades gravitacionales del acelerómetro (fuerza gravitacional g) a m/s^2 , empleando el coeficiente o factor de escala g . Esto habilita a, escalar los datos en el SISTEMA INTERNACIONAL DE UNIDADES (SI), como también a disminuir la pérdida de precisión derivada de la aritmética computacional.

5.10. Algoritmo de Detección de Velocidad Cero PARKIBIP

Durante la marcha de una persona, sus pies se encuentran periódicamente en una fase estacionaria, en la que todo el pie se halla en contacto con el suelo y el cuerpo se desplaza hacia adelante girando rígidamente la extremidad inferior sobre la articulación del tobillo. Dicha fase del ciclo de la marcha, es conocida por estacionaria o en inglés Stance. En contraposición, se encuentra la fase de vuelo o Swing. Asimismo, el ciclo Stance-Swing puede ser fragmentado conforme a aumentar la granularidad en dos nuevas fases, golpe de talón o HS (siglas en inglés, Heel Strike) y el despegue de los dedos del pie o TO (siglas en inglés, Toe Off). Frente a este problema, se emplea el algoritmo de detección de velocidad cero (ZVD, por sus siglas en inglés). Las figuras Fig. 2.5 y Fig. 4.2 permiten ilustrar estos conceptos.

A su vez, los sistemas de navegación inercial basados en sensores de ba-

jo costo padecen de errores de propagación asociados a la estimación de sus parámetros espacio-temporales -como ser la posición, que su error crece proporcionalmente al cubo del tiempo de operación del sistema-. Este problema intrínseco puede reducirse mediante la aplicación de un algoritmo de actualización de velocidad cero (ZVU, siglas en inglés).

Por consiguiente, en PARKIBIP, un algoritmo matemático de detección de velocidad cero tiene los siguientes propósitos:

- Identificar eficazmente los 4 eventos de fases de la marcha -HS, Stance, TO y Swing- a partir de un metodo de ZVD.
- Aplicar adecuadamente la técnica ZVU -basado en ZVD- para reducir el crecimiento de los errores en el sistema de navegación inercial.

Detectado un evento de velocidad cero, el sistema aplica actualizaciones fuertes (en ingles, hard), es decir, cuando el sistema impone una actualización de velocidad cero los parámetros espacio-temporales se ven automáticamente reajustados. Por ejemplo, la velocidad instantánea se establece en cero, la posición y la orientación se vuelven a inicializar.

Previo a que se utilicen las mencionadas compensaciones de velocidad cero, se requiere la identificación de los instantes de tiempo en los cuales los dispositivos inerciales se encuentran estacionarios. A continuación, se define una adaptación del problema en cuestión que soporte la ejecución en tiempo real del algoritmo, y se mencionan los métodos de detección de velocidad cero desarrollados e integrados al diseño de la solución de PARKIBIP.

Matemáticamente, se formaliza el problema de detección estacionaria como un problema de prueba de hipótesis binarias, donde el algoritmo puede elegir entre las dos hipótesis H_0 y H_1 , tal que:

$$\begin{cases} H_0 : \text{IMU en movimiento (hipótesis nula)} \\ H_1 : \text{IMU en Stance (hipótesis alternativa)} \end{cases}$$

Sea $y_k \in R^6$ el vector compuesto por las componentes tridimensionales aceleración ($y_k^a \in R^3$) y velocidad angular ($y_k^w \in R^3$), medidas por el acelerómetro y giroscopio respectivamente en el instante k :

$$\vec{y}_k = \begin{bmatrix} \vec{y}_k^a \\ \vec{y}_k^w \end{bmatrix}$$

Sea la ventana de tiempo **W** -WINDOWS-SIZE- pre-establecida y **configurable en PARKIBIP**, representando el número de observaciones incluidas en la decisión estadística. El objetivo del detector es determinar si en los instantes de tiempo n y $n+W-1$ el IMU está en movimiento o estacionario, a partir de una secuencia de medición z_n :

$$z_n^{\vec{}} = y_{k=n}^{\vec{n+W-1}}$$

Luego, la decisión estadística se toma de tal manera que la probabilidad de decidir si el IMU está estacionario cuando esto no ocurre debe mantenerse baja. Para ello, se emplea el teorema de Neyman-Pearson (NP), que indica cómo elegir entre las dos hipótesis para maximizar la probabilidad de acierto $P_D = Pr\{H_1|H_1\}$ (i.e. la probabilidad de decidir la hipótesis H_1 cuando la misma es cierta) dado un $P_{FA} = Pr\{H_1|H_0\}$ (i.e. la probabilidad de decidir la hipótesis H_1 cuando la hipótesis H_0 es verdadera).

De esta manera, se decidió desarrollar e integrar a la solución dos alternativas de algoritmos numéricos de detección de velocidad cero, que luego favorecerán a evaluar el desempeño de cada uno :

1. **Acceleration Moving Variance Detector (MVD)**. Se basa únicamente en los datos del acelerómetro y la decisión de aceptar H_1 se deduce de la siguiente manera:

$$\begin{cases} T_v(z_n^a) = \frac{1}{W} \sum_{k=n}^{n+W-1} \|y_k^a - \bar{y}_n^a\|^2 < \gamma_v \\ \bar{y}_n^a = \frac{1}{W} \sum_{k=n}^{n+W-1} y_k^a \\ \gamma_v : \text{umbral o threshold de aceptación} \end{cases}$$

2. **Generalized Likelihood Ratio Test (GLRT)**. Adquiere mayor información integrando al análisis, además de las aceleraciones observadas; las señales del giroscopio -velocidades angulares-. La aceptación de H_1

viene dada por la formula:

$$\left\{ \begin{array}{l} T(z_n) = \frac{1}{W} \sum_{k=n}^{n+W-1} \left(\frac{1}{\theta_a^2} \|y_k^a - g \frac{\overline{y_n^a}}{\|y_n^a\|}\|^2 + \frac{1}{\theta_w^2} \|y_k^w\|^2 \right) < \gamma \\ \overline{y_n^a} = \frac{1}{W} \sum_{k=n}^{n+W-1} y_k^a \\ \overline{y_n^w} = \frac{1}{W} \sum_{k=n}^{n+W-1} y_k^w \\ \theta_a^2 : \text{varianza del ruido del acelerómetro} \\ \theta_w^2 : \text{varianza del ruido del giroscopio} \\ \gamma : \text{umbral o threshold de aceptación} \end{array} \right.$$

Por lo tanto, se desarrolló en el lenguaje de programación JAVA, para Android, una clase abstracta -no instanciable- denominada “ZeroVelocityDetector”. Dicha clase define la estructura y las propiedades que compartirán las subclases derivadas que implementen a la clase abstracta, mediante los conceptos jerarquía y herencia. Luego, fueron realizadas las dos clases específicas “GLRatioTestDetector” y “MovingVarianceDetector”, las cuales implementan las decisiones estadísticas particulares citadas en [Algoritmo de Detección de Velocidad Cero PARKIBIP](#) (e.g. la ejecución del test $T(z_n)$), y mantienen el comportamiento de la superclase ZeroVelocityDetector (e.g. estructuras de datos, agregar una medición para efectuar un test, gestionar el resultado del test).

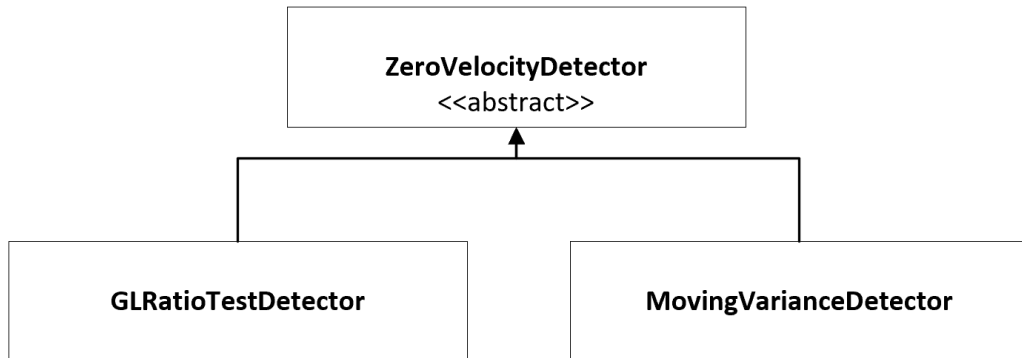


Figura 5.9: Componentes de detección de velocidad cero. Superclase abstracta ZeroVelocityDetector, subclases GLRatioTestDetector y MovingVarianceDetector.

Así pues, se emplearon dos estructuras de datos auxiliares. Primero, una cola FIFO del tamaño W implementada con la estructura de datos `LinkedList`, destinada a almacenar las mediciones recepcionadas por el detector. Segundo, un arreglo de estados booleanos del mismo tamaño, que permite almacenar e iterar sobre los resultados de los tests estadísticos.

Cuando el ZVD detecta un nuevo evento de medición, en orden de precedencia ocurren los siguientes pasos dentro del detector:

1. El sistema almacena la nueva medición en la cola FIFO
2. Chequea si la cantidad de elementos de la cola es igual al tamaño de la ventana W (caso base de primer procesamiento). Esto es, recopila los datos necesarios para el primer procesamiento, luego, a medida que llegan mediciones se agregan y se remueven elementos -el número de elementos es W -
3. En caso de que la cola tenga W observaciones, el detector procede a efectuar el test estadístico según el algoritmo ZVD seleccionado.
4. Posterior, el resultado del test es almacenado en el arreglo de resultados de tal forma que:
 - Si el test es positivo (H_1 , es estacionario), todos los elementos del arreglo booleano se convierten en 1.
 - Se obtiene y remueve el primer elemento del arreglo -primer test almacenado y actualizado $W-1$ veces dentro de la ventana W -.
 - Se desplazan un lugar hacia la izquierda los restantes tests del arreglo (en inglés, `left shift`), y el último elemento $W - 1$ es inicializado en 0 (i.e. H_0 o hipótesis nula).
5. Se despacha el resultado del test obtenido al listener u oyente del detector (en general, un analizador de datos), previamente establecido.
6. Finalizado el procesamiento, en caso de que la cola de mediciones posea W elementos, se remueve la primer medición y se continua iterando.

Con el fin de clarificar las etapas mencionadas, se propone el pseudocódigo **Pseudocódigo Zero Velocity Detector**, sin embargo, no se refinan los metodos auxiliares empleados (e.g. `makeTest`, `pushStatisticalTest`).

Algorithm 1 Pseudocódigo Zero Velocity Detector

```
1: Queue<MeasurementModel> measures;
2: Array<Boolean> zupdtValues;
3:  $W \leftarrow$  windowSize;
4: setupParameters();

5: function UPDATEDETECTOR(data)
6:   addQueue(measures,data);
7:   if sizeQueue(measures) ==  $W$  then  $\triangleright$  PB. Espera por una ventana
8:     boolean currentTest  $\leftarrow$  makeTest();  $\triangleright$  Ejec. del test particular
9:     int zValue  $\leftarrow$  pushStatisticalTest(currentTest);
10:    removeQueue(measures);
11:    dispatchEventDetected(zValue);
12:   end if
13: end function

14: function PUSHSTATISTICALTEST(test)  $\triangleright$  Actualizar la secuencia
15:   if isH1 (test) then
16:     setOnes(zupdtValues);  $\triangleright$  Establecer ventana en unos
17:   end if  $\triangleright$  shift data
18:    $z \leftarrow$  zupdtValues[0];  $\triangleright$  Shift 1 elemento a la izquierda
19:   for all zupdtValues - 1 do
20:     zupdtValues[i]  $\leftarrow$  zupdtValues[i+1];
21:   end for  $\triangleright$  Próximo test en 0
22:   zupdtValues[windowsSize-1] = 0; return z;
23: end function
```

5.11. Algoritmo Filtro de Kalman PARKIBIP

A modo introductorio, en el capítulo **Proyecto: Decisiones de diseño**, apartado **Filtro de Kalman**, se presentó un algoritmo óptimo de estimación denominado filtro de Kalman; el cual sirve para poder identificar el estado oculto -no medible- de un sistema dinámico lineal que está sometido a ruido blanco Gausseano -cuya función de densidad responde a una distribución normal-. Es decir, permite hallar la mejor estimación de variables de interés no medibles (e.g. velocidad), a partir de la combinación de variables indirectas que son medibles (e.g. aceleración). Asimismo, con la finalidad de introducir a la temática se mencionaron brevemente algunas de sus aplicaciones, siendo en ciertos casos, el problema de estimar la posición de un sistema de navegación

de un automóvil en movimiento similar al modelo empleado en PARKIBIP.

Análogo a lo sucedido con la estimación de la orientación en [Algoritmo Filtro de Orientación](#), no se encontró una librería de código abierto que se adecue al contexto de PARKIBIP, y por consiguiente pueda ser reutilizada.

Por lo tanto, se implementó un filtro de Kalman avanzado y multidimensional, en notación matricial. El propósito de dicho filtro es estimar las variables indirectas posición, velocidad y ángulos de Euler en cada instante de tiempo, a partir de las variables acelerómetro, giroscopio y magnetómetro sensadas por el dispositivo IMU. Para llevar adelante esta tarea, se requieren conocimientos y funciones avanzadas en álgebra lineal (e.g. operaciones matriciales) y física (e.g. leyes mecánicas, rotaciones y traslaciones). Todas las operaciones auxiliares requeridas fueron implementadas en la API de algoritmos en [Módulo de algoritmos numéricos y modelos de representación](#).

El filtro de Kalman es un método numérico Predictor-Corrector, explícito de un solo paso; es decir depende únicamente de la estimación del estado previo (x_{k-1}) para lograr predecir el siguiente estado (x_k). El procedimiento Predictor-Corrector comienza mediante una primera predicción de los valores de las variables -resultantes de la extrapolación de una función ajustada (e.g. leyes mecánicas)-, luego, el corrector, refina la aproximación inicial utilizando el valor predicho de la función y una función objetivo que minimiza el error cuadrático medio del problema en cuestión.

Dada la complejidad del análisis, se deducen las ecuaciones físico-matemáticas y la metodología numérica empleada, así como un pseudo-código de la resolución de Kalman Filter con actualizaciones de cero velocidad.

Sea el **vector de estado** \mathbf{x}_k en la iteración \mathbf{k} , dado por:

$$\begin{cases} \mathbf{x}_k = [P_k, V_k, O_k] = [position, velocity, orientation] \\ P_k = (p_0, p_1, p_2), p_i \in R \\ V_k = (v_0, v_1, v_2), v_i \in R \\ O_k = (roll, pitch, yaw), o_i \in R \end{cases}$$

Previo a introducir el procedimiento del algoritmo, es imprescindible presentar los **Coefficientes** que intervienen en el formulamiento del filtro de Kalman y los roles que cumplen:

F = Matriz Predicción o Transición

B = Matriz de Control

u = Vector de Control

P = Matriz de error de Covarianza de Estado

H = Matriz de Medición o relación entre mediciones y el vector de estado

R = Matriz de error de Covarianza Ruido de Medición

Q = Matriz de error de Covarianza Ruido del proceso

Aplicando las ecuaciones dadas por las **leyes mecánicas del movimiento** tanto para la posición como para la velocidad en función de la aceleración -medida por el sensor inercial- y el tiempo, se obtiene la siguiente formulación:

$$x_k = \begin{bmatrix} P_k \\ V_k \end{bmatrix} = \begin{bmatrix} P_{k-1} + V_{k-1}\Delta t + \frac{\Delta t^2}{2} \\ V_{k-1} + acc\Delta t \end{bmatrix}$$

Reordenando los términos en función del vector de estados:

$$x_k = \begin{pmatrix} 1 & \Delta t \\ 0 & 1 \end{pmatrix} X_{k-1} + \begin{pmatrix} \frac{\Delta t^2}{2} \\ \Delta t \end{pmatrix} acc \quad (5.1)$$

$$x_k = F X_{k-1} + B u \quad (5.2)$$

Definiendo los cambios de variables, hallamos los valores $\mathbf{F} = \begin{pmatrix} 1 & \Delta t \\ 0 & 1 \end{pmatrix}$, $\mathbf{B} = \begin{pmatrix} \frac{\Delta t^2}{2} \\ \Delta t \end{pmatrix}$ y $\mathbf{u}_{k-1} = acc$.

Finalmente, Se logra obtener un método iterativo para predecir las ecuaciones de navegación o transición de estados, que puede ejecutarse en tiempo real a medida que se recopilan los datos de medición.

Entonces, para el paso (i), se llama \hat{x}_k y \hat{P}_k las variables estimadas en la

predicción y sus ecuaciones vienen dadas por:

$$(i) \text{Predicción} \begin{cases} \hat{x}_k = FX_{k-1} + Bu_{k-1} + w_{k-1} \\ \hat{P}_k = FP_{k-1}F^t + Q \\ u_{k-1} = (accX, accY, accZ) \end{cases}$$

Luego, en (ii), se ejecuta la actualización de los valores estimados mediante la aplicación del factor de ganancia de Kalman. Primero, calcula el factor de ganancia K , corrige el estado con la medición, actualiza la matriz de Covarianza. Esto es:

$$(ii) \text{Corrección} \begin{cases} K_k = \frac{\hat{P}_k H^t}{H \cdot \hat{P}_k H^t + R} \\ y_k = H\hat{x}_k + v_k \\ x_k = \hat{x}_k + K_k y_k \\ P_k = (I - K_k \cdot H) \hat{P}_k \end{cases}$$

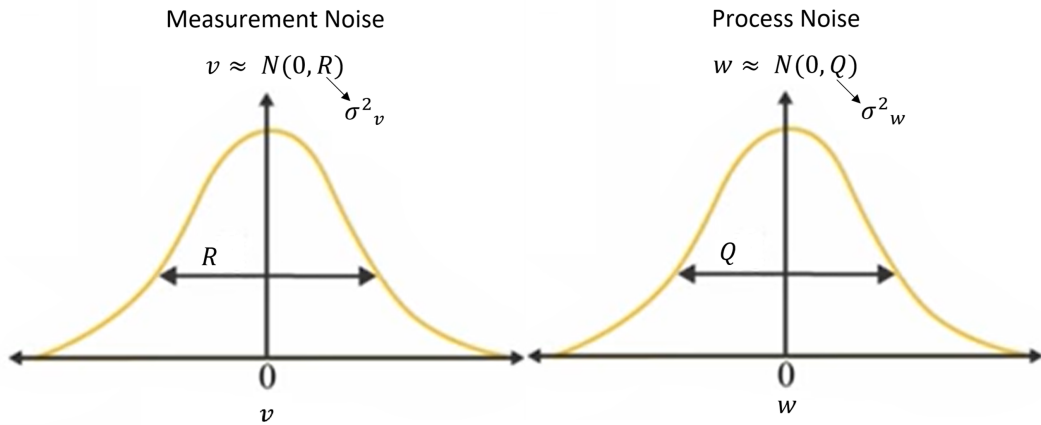


Figura 5.10: Distribuciones supuestas para los vectores de ruido estadísticos de Medición v_k y de Proceso w_k .

El sistema de ecuaciones debe contemplar los desvíos relacionados a los distintos ruidos estadísticos. Para esto, se asume que los vectores v_k y w_k de ruido de medición y proceso poseen una distribución gaussiana de media cero y covarianzas R y Q respectivamente. Las matrices Q y R en general se emplean como parámetros de ajuste para obtener el rendimiento deseado.

Una particularidad crucial del modelo empleado, es la integración del método Zero Velocity al filtro de Kalman. Esta tarea permite compensar la velocidad en la etapa de corrección (ii), llevando su valor a Cero en caso de que el algoritmo Zero Velocity Detector identifique un momento es-

tacionario. Así, se reducen considerablemente los desvíos incrementales debido al error acumulativo de integración respecto al tiempo.

Algorithm 2 Pseudocódigo de Kalman Filter con compensacion Zero Velocity

```

1: function INITKALMANFILTERZUPDTATE(A)
2:   initFilters();
3:   initVectors();
4: end function

5: function RUNKALMANFILTERZUPDTATE(measurek, zValuek)
                                     ▷ Prediction with previous  $x_h$ 
6:    $xhat \leftarrow$  NavigationEquations(measurek,  $x_h$ );
                                     ▷ Compute F and G
7:   stateMatrix(measurek);
       ▷ Predict state covariance matrix  $P = F * P * F' + G * Q * G'$ ;
8:    $P \leftarrow$  F.multiply(P).multiply(F.transpose()).add(G.multiply(Q).multiply(G.transpose()))
       ▷ Check covariance matrix P is symmetric
9:    $P \leftarrow$  P.add(P.transpose()).scalarMultiply(0.5d);
       ▷ Check if a zero velocity update should be done
10:  if zValue == true then
       ▷  $K = (P * H') / (H * P * H' + R)$ ;  $s = H * P * H' + R$ , solve  $s' * k' = H * P'$ 
11:    RealMatrix s  $\leftarrow$  H.multiply(P).multiply(H.transpose()).add(R);
12:    RealMatrix b  $\leftarrow$  H.multiply(P.transpose());
13:    RealMatrix      K  $\leftarrow$  new      CholeskyDecomposi-
        tion(s.transpose()).solve(b).transpose();
                                     ▷ Innovation  $z = -x$ 
14:    RealVector innovation  $\leftarrow$  H.operate(xhat).mapMultiply(-1);
                                     ▷ Correct  $xhat$  using the estimated perturbations
15:    RealVector dx  $\leftarrow$  K.operate(innovation);
16:     $x_h \leftarrow$  xhat.add(dx);
                                     ▷ Compensate current orientation
17:    compensateOrientation(dx);
                                     ▷ Compensate current covariance matrix P
18:     $P =$  Id.subtract(K.multiply(H)).multiply(P);
                                     ▷ Check covariance matrix P is symmetric
19:     $P \leftarrow$  P.add(P.transpose()).scalarMultiply(0.5d);
20:  else
21:     $x_h \leftarrow$  xhat;      ▷ Update new  $x_h$  if not correction is done
22:  end if
23: end function

```

Para concluir con la exposición de el filtro de Kalman integrado a un detector de cero velocidad, se expone el **pseudo-código** de la implementación

realizada en el lenguaje Java de Android dado por **Pseudocódigo de Kalman Filter con compensacion Zero Velocity**. El filtro se inicializa mediante la función “initKalmanFilterZUpdate”, luego con la recepción de cada medición se ejecuta la función “runKalmanFilterZUpdate”.

Los valores empleados para los coeficientes fueron elegidos acorde a la especificación del hardware y para optimizar el funcionamiento de la aplicación. A continuación, se exponen las configuraciones para el ruido de proceso, el ruido de medición y matrices de covarianza de estado inicial Q, R y P. Se supone que las tres matrices son matrices diagonales y todas las configuraciones se definen como desviaciones estándar:

- Período de muestreo del sistema:
 - T_s [seg] = $(1/f)$, f frecuencia configurable
- Ruido de proceso para modelar el ruido del acelerómetro (eje de coordenadas x, y, z) y otros errores del acelerómetro:
 - Ruido de Aceleración[m/s²]: $\sigma_{Acc} = 0.5^2$
- Ruido de proceso para modelar el ruido del giroscopio (eje de coordenadas x, y, z) y otros errores del giroscopio:
 - Ruido Angular[rad/s]: $\sigma_{Gyro} = (0.5 * \pi/180)^2$
- Elementos diagonales de la matriz de Covarianza P del estado inicial:
 - Varianza de Posición [m]: $\sigma_P = (1 \cdot 10^{-5})^2$
 - Varianza de Velocidad [m/s]: $\sigma_V = (1 \cdot 10^{-5})^2$
 - Varianza de Orientación [rad]: $\sigma_{Att} = (\pi/180 * 0.1)^2$
 - Desviación de Aceleración[m/s²]: $\sigma_A = (0.3)^2$
 - Desviación de Velocidad angular [rad/s]: $\sigma_G = (0. * \pi/180)^2$
- Covarianza de ruido de medición R:
 - Desviación de Velocidad σ_{VelR} [m/s] = 0.01

El vector de estados se inicializa mediante un **Procedimiento de Calibración**. El mismo consiste en recopilar las primeras N = 30 observaciones del

sistema, computar la media de las variables roll, pitch, yaw; luego establecer el vector de estados. La posición y la velocidad inicial comienza con el valor 0:

$$X_{initial} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & r\bar{o}ll & p\bar{i}tch & y\bar{a}w \end{pmatrix}$$

Los coeficientes estables del sistema dinámico P (inicial), Q, H, R, fueron establecidos de la siguiente manera:

$$P_{initial} = \begin{pmatrix} sigmaP & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & sigmaP & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & sigmaP & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & sigmaV & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & sigmaV & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & sigmaV & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & sigmaAtt & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & sigmaAtt \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$Q = \begin{pmatrix} sigmaAcc & 0 & 0 & 0 & 0 & 0 \\ 0 & sigmaAcc & 0 & 0 & 0 & 0 \\ 0 & 0 & sigmaAcc & 0 & 0 & 0 \\ 0 & 0 & 0 & sigmaGyro & 0 & 0 \\ 0 & 0 & 0 & 0 & sigmaGyro & 0 \\ 0 & 0 & 0 & 0 & 0 & sigmaGyro \end{pmatrix}$$

$$H = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

$$R = \begin{pmatrix} sigmaVelR & 0 & 0 \\ 0 & sigmaVelR & 0 \\ 0 & 0 & sigmaVelR \end{pmatrix}$$

Los restantes valores -P, F, B, u-, mutan según las observaciones recopiladas en cada instante de tiempo y la dinámica del sistema. Se resalta la implementación de un algoritmo iterativo, que reacciona con cada evento de medición.

En orden de precedencia ocurren los siguientes eventos de sistema:

1. Recopilación de mediciones: aceleración, velocidad angular, fuerzas magnéticas
2. Estandarización de unidades inerciales sobre las mediciones
3. Computo de la orientación instantánea en formato cuaternión -a partir de los datos medidos- mediante filtro eficiente de orientación
4. Con la orientación adquirida. Rotación a un marco de referencia inercial -tierra-, desde el marco del sensor
5. Ejecución de algoritmo Zero Velocity Detector
6. Ejecución del filtro de Kalman con los datos procesados -cuaternión, mediciones, zero velocity-

5.12. Analizadores de datos

Los analizadores de datos son las entidades que tienen por misión procesar los datos crudos que provienen desde los distintos sensores del IMU, utilizando los métodos numéricos ya mencionados (e.g detección de velocidad cero, filtrado de Kalman, entre otros). Son representados en el sistema mediante la clase *DataAnalyzer*. El diseño de clases de esta sección del sistema se puede ver en la figura Fig. 5.11.

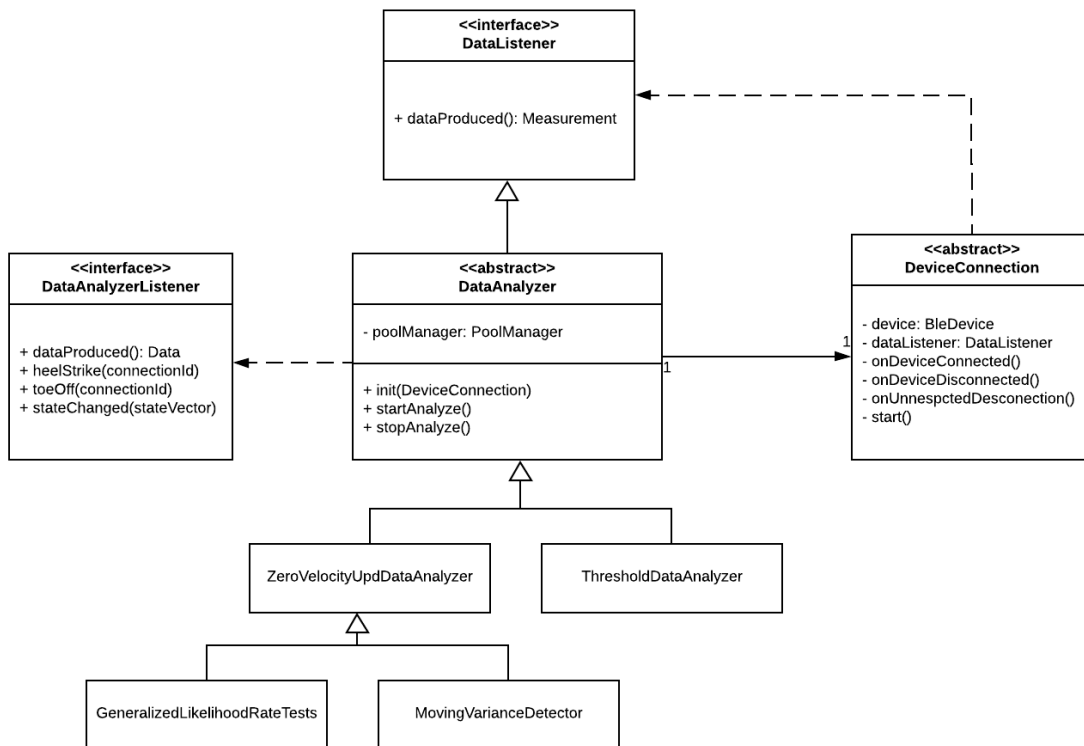


Figura 5.11: Diagrama de clases del módulo de la aplicación Android en donde se analizan y procesan los datos para detectar eventos relativos a la marcha del paciente.

Tal como se puede apreciar en la figura, el *DataAnalyzer* implementa la interfaz *DataListener*, y así, posee la capacidad de recibir los datos desde una instancia de *DeviceConnection* -entidad que representa una conexión con un dispositivo IMU-. Es decir, cada *DeviceConnection* tiene asociado un *DataAnalyzer*, definido como su *DataListener*. Cuando es invocado el método *startA-*

nalyze() correspondiente a un analizador, éste le comunica a su *DeviceConnection* que puede comenzar el flujo de datos. La instancia de *DeviceConnection* inicia el envío de datos de medición, representados con el modelo *Measurement*, a través de la interfaz *DataListener*, para que el analizador de datos pueda procesar las mediciones producidas por los sensores.

Existen diferentes *DataAnalyzers* en la aplicación. Cada uno implementa un método numérico particular para analizar los datos. Durante el diseño se utilizó el concepto de polimorfismo; es decir, el comportamiento de todos los *DataAnalyzers* se encuentra definido en ésta clase abstracta, sin embargo, cada instancia concreta implementa los métodos derivados de forma particular empleando diferentes métodos numéricos. Por ejemplo, *ZeroVelocityUpdateDataAnalyzer* implementa el método de detección de velocidad cero para procesar los datos e identificar los eventos Heel Strike y Toe Off.

El hecho de utilizar polimorfismo para los analizadores de datos hace que el sistema sea escalable y extensible, ya que implementar un nuevo método para analizar los datos y generar eventos, implica crear únicamente una clase concreta que extienda de la clase *DataAnalyzer*. Además, cada analizador desarrolla un único método de análisis y tiene asignada una única responsabilidad bien definida. Este forma de diseño, contribuye a mejorar la calidad del sistema, haciendo que sea más sencillo realizar pruebas sobre un método de análisis concreto y compararlo con el resto de los métodos; además de incrementar su mantenibilidad. Cuando un *DataAnalyzer* detecta eventos u obtiene nuevos datos como resultado, los envía a su *DataAnalyzerListener*. Esta interfaz está definida para ser implementada por la entidad que esté interesada en recibir el resultado del procesamiento, que en la implementación final de PARKIBIP esta entidad es *BiofeedbackModule*.

DataAnalyzerListener es un protocolo definido para ser implementados por las entidades que necesiten observar los resultados de un *DataAnalyzer*. Cuando un analizador de datos detecta cierto evento de la marcha, notifica a su *DataAnalyzerListener*. En PARKIBIP, este protocolo es implementado por *BiofeedbackModule*.

5.13. Configuración de algoritmos y parámetros

PARKIBIP es también una aplicación experimental. Es decir, si bien el objetivo principal es obtener un sistema que beneficie la marcha de los enfermos de Parkinson; pretende ser, en una primera instancia, una herramienta de investigación que permita encontrar la mejor forma de estimular a un paciente para que su marcha progrese. Por lo tanto, resulta crucial experimentar con los algoritmos desarrollados para estudiar la marcha, poder utilizar un método u otro, configurar éstos métodos numéricos de distintas formas, establecer distintas reglas para definir los estímulos, etc.

Ante ésta necesidad, se desarrolló un modulo de configuración. Este módulo consta de su propia interfaz gráfica, la cual puede ser encontrada dentro del menú principal de la aplicación. En la figura Fig. 5.12, se expone el diagrama de clases correspondiente a la porción del sistema dedicada a la parametrización.

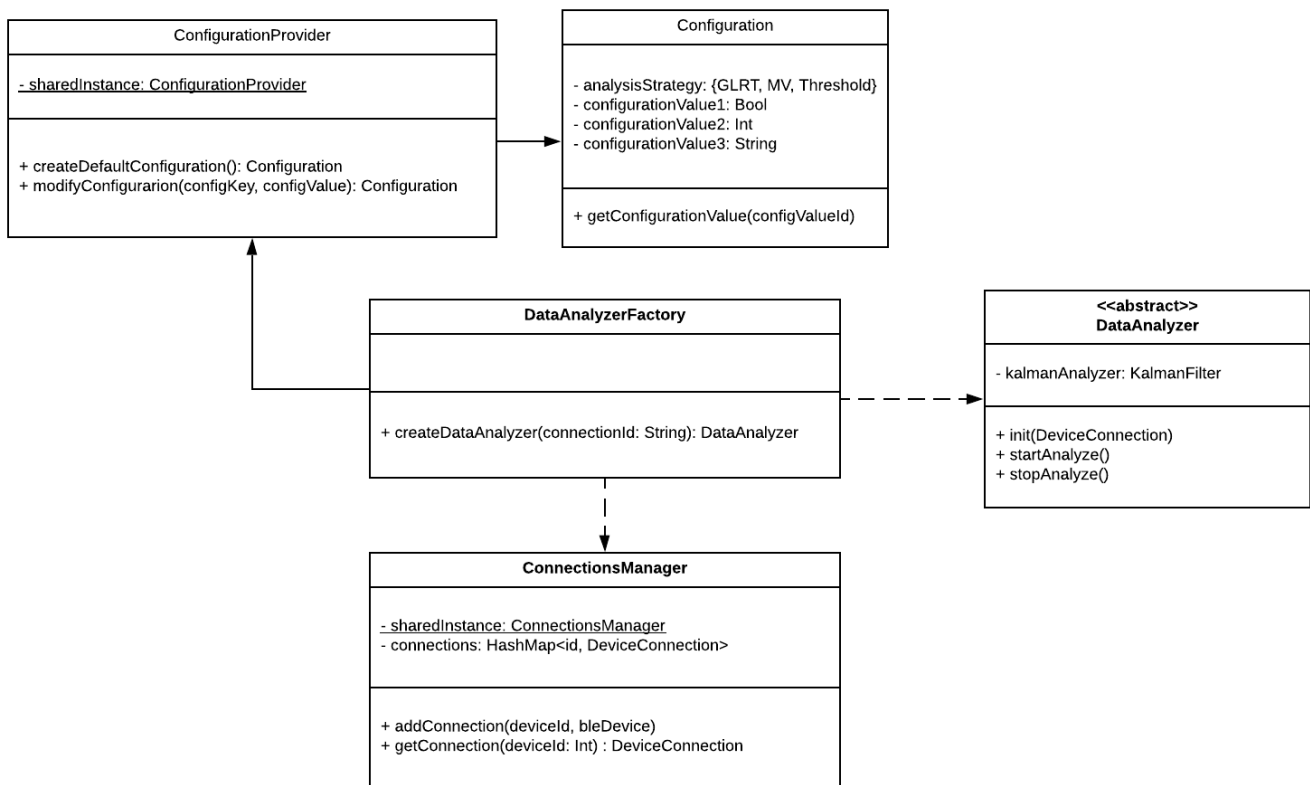


Figura 5.12: Diagrama de clases del módulo de la aplicación Android donde se administran las distintas configuraciones del sistema

Un estado de configuración está representado por una instancia de la clase *Configuration*. La misma clase, dispone del conjunto de atributos correspondientes a cada uno de los parámetros que pueden ser editados dentro del sistema. A su vez, *ConfigurationProvider* es la clase encargada de manejar éstas configuraciones, por ende, un objeto *Configuration* se mantiene inmutable mientras el proveedor no ejecute una transacción de modificación sobre el mismo. Cuando un valor es modificado a través de la pantalla de Configuraciones de la aplicación, esta *Activity* invoca al proveedor de configuraciones para solicitarle dicho cambio.

ConfigurationProvider implementa el patrón de diseño *Singleton*, es decir, una clase para la cual existe una única instancia, y una vez creada se mantiene en memoria durante todo el ciclo de vida de la aplicación Android. Este *Singleton* mantiene una referencia a la instancia de *Configuration* actual, y al ser global, puede ser accedido desde cualquier clase del sistema.

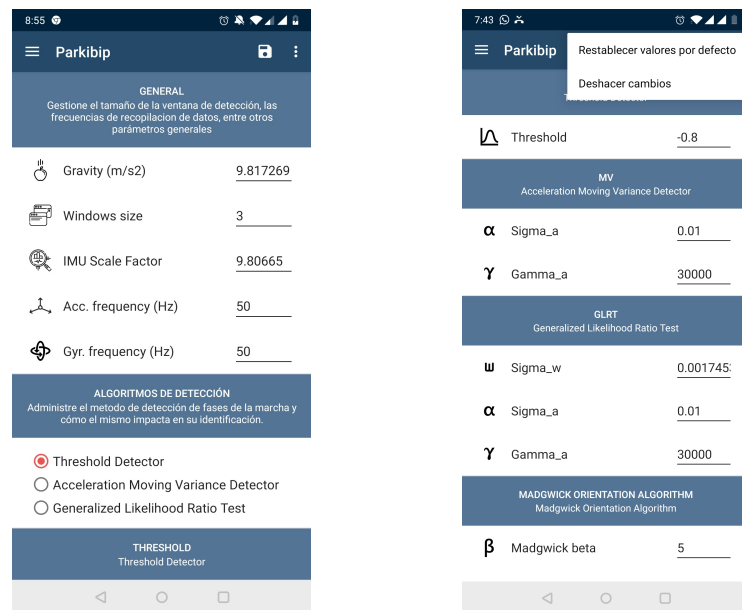
El rol principal del módulo de configuraciones, es el de proveer los parámetros necesarios para configurar los métodos numéricos pertenecientes a los analizadores de datos. Cuando se desea crear un analizador de datos - *DataAnalyzer*-, el método *createDataAnalyzer(DeviceConnection)* de la fábrica de datos *DataAnalyzerFactory* es invocado. De esta forma, éste patrón “factory”, puede crear instancias de *DataAnalyzer* vinculadas a la conexión con el dispositivo que le fue provista por parámetro, a partir de la configuración actual. Esto es, le solicita al proveedor *ConfigurationProvider* la configuración vigente y construye el *DataAnalyzer* adecuado empleando los parámetros de la misma. Esta configuración, indica cuál es el *DataAnalyzer* que debe ser creado, según el método de análisis seleccionado por el Usuario, así como sus parámetros requeridos para establecer éste método.

Dentro de los posibles parámetros que se pueden configurar en esta sección de la aplicación se encuentran:

1. Algoritmo de detección de fases de la marcha:
 - Threshold: Algoritmo básico de detección de eventos a partir de picos de aceleración.
 - Acceleration Moving Variance Detector: Algoritmo de detección de eventos a partir de la detección de momentos de velocidad cero -ver 5.10-.
 - Generalized Likelihood Ratio Test: Algoritmo de detección de eventos a partir de la detección de momentos velocidad cero -ver 5.10-.
2. Parámetros globales que afectan a todos los métodos implementados:
 - Gravity: Valor de la aceleración gravitatoria $\frac{m}{s^2}$.
 - Windows size: Tamaño de la ventana de mediciones contiguas a ser analizadas por los métodos para la detección de eventos.
 - Accelerometer frequency: Frecuencia configurada en el acelerómetro para recopilación de mediciones del sensor, expresada en la unidad Hz.
 - Gyroscope frequency: Frecuencia configurada en el giroscopio para enviar las mediciones, expresada en la unidad Hz.
3. Reglas de biofeedback, para decidir cuándo se debe estimular al Paciente.

4. Activación de tipos de estímulos: vibración y/o sonido.
5. Coeficientes particulares de cada método.

En la pantalla de configuración -ver Fig. 5.13-, representada con la clase *ConfigurationActivity*, se implementó adicionalmente la posibilidad de re-establecer la configuración por defecto. Esto permite que, luego de realizar ciertos experimentos al cambiar los parámetros, se pueda volver a la configuración de fábrica (i.e. considerados valores óptimos de PARKIBIP). Esta funcionalidad es de vital importancia si se quiere rápidamente re-establecer la configuración sin necesidad de obtenerla nuevamente, o de realizar una reinstalación de la aplicación. Además, durante un proceso de edición de parámetros, se le permite al usuario descartar los cambios realizados, retornando a su estado inicial. Estas funcionalidades se pueden ver con mayor detalle en la sección [Especificación detallada y alcance del proyecto](#).



(a) Configuración de parámetros y selección del algoritmo de detección. (b) Restauración de valores. PARKIBIP permite restablecer a valores de fábrica, así como descartar cambios efectuados.

Figura 5.13: Configuración de parámetros generales y particulares a los detectores. Además, selección del algoritmo de detección.

5.14. Módulo de Biofeedback

El módulo de *Biofeedback* cumple una función esencial para PARKIBIP. Con una visión global, es el encargado de capturar y procesar los resultados de los distintos *DataAnalyzer* -analizador creado para cada IMU-, los cuales procesan los datos, emiten eventos a dicho módulo -indicando la ocurrencia de un Heel Strike o un Toe Off- y además, notifican parámetros espacio-temporales como la posición, la velocidad y la orientación instantánea relativos a cada pie. Así, el módulo de Biofeedback transforma la información recibida según reglas pre-definidas dentro del mismo, para estimular al usuario, y luego calcula los datos de la sesión en tiempo real para ser mostrados en la pantalla.

La figura Fig. 5.14 presenta un diagrama del subsistema en el cual se encuentra el módulo de Biofeedback -representado con la clase *BiofeedbackModule*-, donde se puede visualizar cuáles son las clases con las que el mismo interactúa.

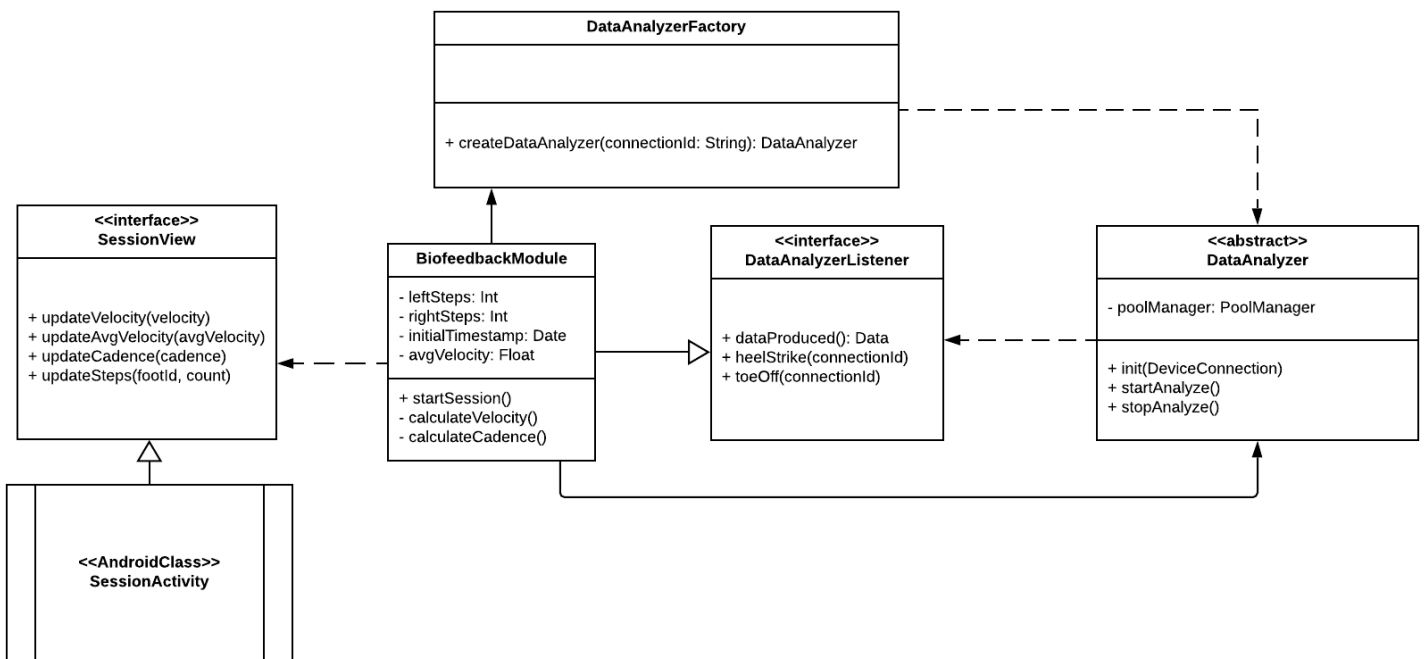


Figura 5.14: Diagrama de clases del módulo de la aplicación Android que establece las reglas para estimular y calcula los parámetros espacio-temporales de la sesión en tiempo real

Cuando es iniciada una sesión de fisioterapia con PARKIBIP, *BiofeedbackModule* es quien solicita al *DataAnalyzerFactory* la creación de los *DataAnaly-*

zer, se suscribe a los eventos como su *DataAnalyzerListener* manteniendo las referencias adecuadas. Para procesar los datos que produce cada *DataAnalyzer*, implementa la interfaz *DataAnalyzerListener*. Esta interfaz encapsula todos los métodos que utiliza el *DataAnalyzer* para notificar a las clases interesadas los datos que obtiene como resultado del procesamiento. Así, los *DataAnalyzer* notifican a su *DataAnalyzerListener* cada vez que se produce un evento del tipo Heel Strike o Toe Off, como también las actualizaciones del vector de estado del algoritmo filtro de Kalman, el cual contiene posición, velocidad y orientación.

BiofeedbackModule emplea los entradas mencionadas para calcular los siguientes datos en tiempo real:

- Velocidad instantánea para cada pie
- Velocidad promedio de la marcha
- Cantidad de pasos
- Cadencia (i.e pasos por minutos)
- Duración media del paso
- Aceleración instantánea para cada pie
- Aceleración promedio de la marcha

Este módulo, tiene además la capacidad de reaccionar a ciertas reglas pre-definidas y configurables conforme a estimular adecuadamente al paciente. Los niveles de estímulos para el paciente PARKIBIP son:

- Vibración en el dispositivo IMU utilizando el motor de vibración embebido en el dispositivo MetaMotionR.
- Sonido “Beep” emitido por la aplicación móvil, el cual es tan importante como para tener su referencia en el nombre del proyecto.

La figura Fig. 5.15 ilustra una sesión PARKIBIP en curso. Tal como se puede apreciar, el sistema reacciona en tiempo real a la marcha del paciente en rehabilitación, mediante el envío de estímulos externos -vibración y/o

sonido- y el cómputo de parámetros espacio-temporales. Así, medición a medición, la interfaz de usuario comienza a transitar por distintos valores fruto del procesamiento de PARKIBIP.

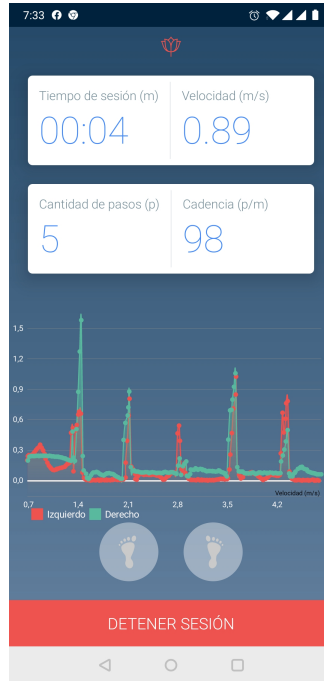


Figura 5.15: Pantalla de una Sesión Activa PARKIBIP. El sistema reacciona en tiempo real a la marcha del paciente en rehabilitación. Dispara estímulos externos -vibración y/o sonido- y computa los parámetros espacio-temporales.

Si bien las posibilidades son infinitas (i.e. debido a múltiples combinaciones de parámetros espacio-temporales), a continuación se detallan las principales reglas implementadas en PARKIBIP para decidir en qué momentos se producen dichos estímulos:

- **On Heel Strike:** Cada vez que se produce un Heel Strike. Esta es una regla básica pero sumamente importante, ya que el funcionamiento correcto de la misma indica que se identifican adecuadamente los pasos de la marcha.
- **On FoG:** Transcurrido un intervalo de tiempo pre-definido sin que ocurra un Heel Strike. Esta regla es útil para detectar un evento de FoG (Bloqueo de la marcha), y a través del estímulo buscar interrumpirlo.
- **On Instant Velocity Threshold:** Al atravesar umbrales de velocidad instantánea. En este caso se le notifica al paciente a través de estímulos

sobre la velocidad instantánea, que la misma es muy alta o muy baja y podría ocasionar, por ejemplo, una caída por mayor velocidad.

- **On Average Velocity Threshold:** Al atravesar umbrales de velocidad promedio. En este caso se le notifica al paciente mediante estímulos que el ritmo es demasiado bajo o alto.
- **On Dynamic Calculated Time:** Ocurre luego de cierto intervalo de tiempo calculado en forma dinámica a partir de una primera caminata de configuración. Por ejemplo, el paciente realiza una marcha inicial de 6 pasos, y en base a los datos recopilados se computa el tiempo promedio entre los mismos. Luego, el estímulo vibratorio y/o sonoro se realiza en un periodo constante definido por este tiempo calculado -simulando una terapia rítmica de rehabilitación-. Desde el punto de vista terapéutico, podría resultar que la caminata del paciente sea demasiado lenta o demasiado rápida, por lo que el tiempo promedio calculado a partir de su caminata inicial no sería un buen indicador para efectuar el estímulo. Sin embargo, podría ser de utilidad para evaluar al paciente en deambulación respecto al ritmo de marcha y su uniformidad.

Muchas de las grandes oportunidades de mejora que presenta el sistema, para ser implementadas a futuro, resultan de mejorar este módulo, ya sea implementando nuevas reglas o incrementando la inteligencia del mismo para decidir en qué momentos estimular. Además, se podrían crear nuevos tipos de estímulos, como por ejemplo el habla en lenguaje natural.

5.15. Caso de uso: Flujo automatizado de Sesión Activa PARKIBIP

En línea con los subprocesos desarrollados y mencionados con anterioridad, se propone el caso de uso de Sesión Activa, fundamental para lograr comprender el funcionamiento de fondo de PARKIBIP. De esta manera, se elaboró el diagrama Fig. 5.16, que modela el flujo de tareas e información mediante la notación estándar BPMN 2.0.

Como se puede apreciar en la figura, el modelo integra los distintos componentes (e.g. *DeviceConnection*, *DataAnalyzer*, *Services*, *Threads*, *Queues*, entre otros) y algoritmos numéricos desarrollados (e.g. filtro de Orientación, filtro de

kalman, ZVD); luego los combina conforme a proponer una sinergia de trabajo que aumente el rendimiento del sistema, minimice el retardo de transmisión de eventos (por ende, los estímulos) y mantenga las responsabilidades bien delineadas (i.e. bajo acoplamiento y alta cohesión entre componentes).

Para una mayor claridad y comprensión, el diagrama sintetiza el flujo a un único dispositivo inercial (en PARKIBIP, *DeviceConnection*), esto significa que, existirán tantos componentes *DataAnalyzers* con sus correspondientes *KalmanFilterThreads* como dispositivos IMU o *DeviceConnection* operando.

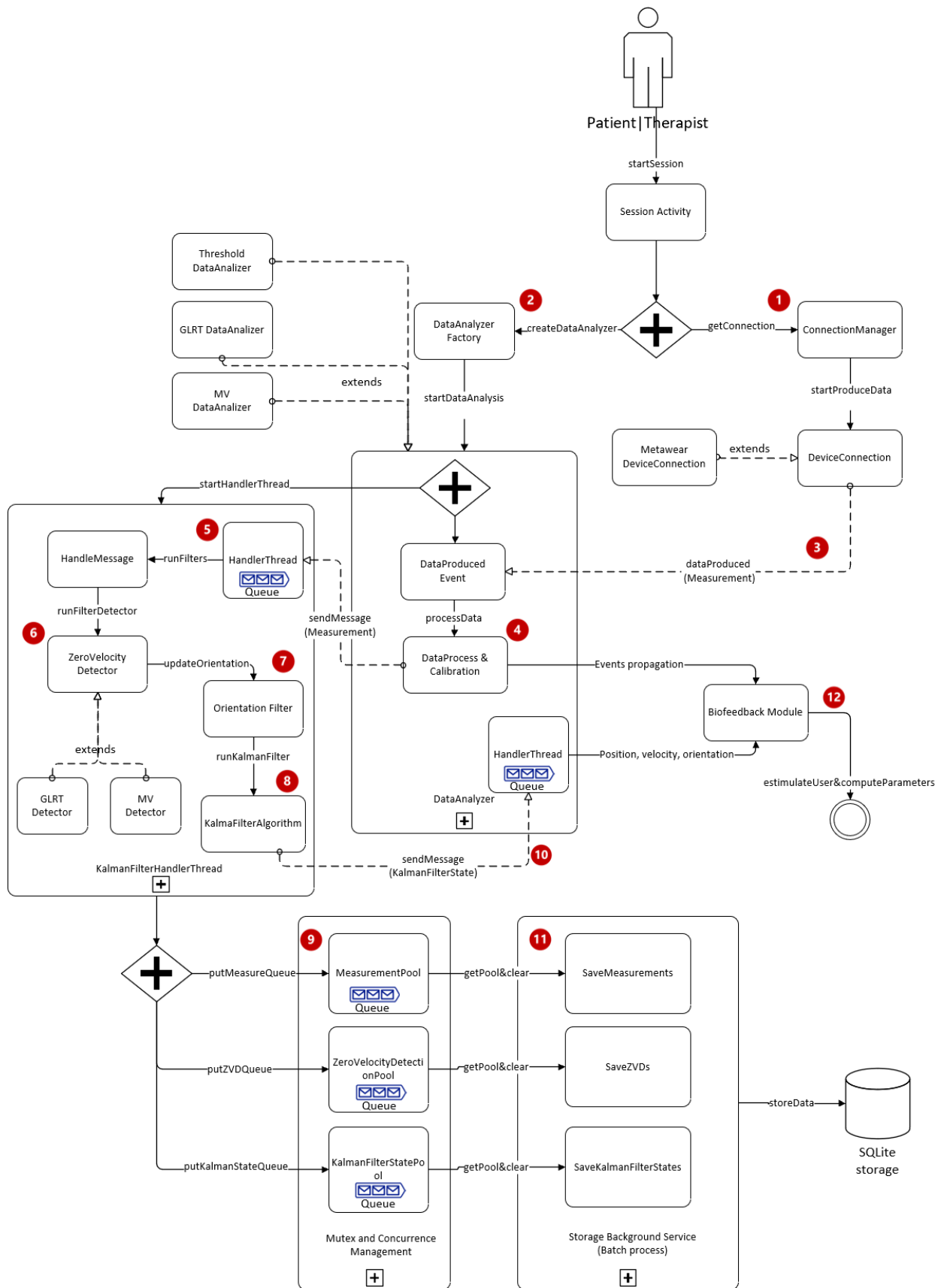


Figura 5.16: Flujo automatizado de Sesión Activa PARKIBIP. El modelo integra y combina los distintos componentes (e.g. *DeviceConnection*, *DataAnalyzer*, *Services*, *Threads*, *Queues*, entre otros) y algoritmos numéricos que fueron desarrollados (e.g. filtro de Orientación, filtro de kalman, ZVD) con el fin de cumplir los objetivos del Proyecto.

El Caso de Uso comienza cuando, un Terapeuta en una actividad de rehabilitación o un Paciente autónomo, inicia una Sesión previamente configurada en la aplicación PRKIBIP. En consecuencia, se disparan los eventos numerados en el diagrama Fig. 4.2, y descriptos como notas al pie en lo que sigue:

1. Mediante el manejador de conexiones -*ConnectionManager*-, se obtiene la instancia lógica del dispositivo IMU particular a iniciar, un *DeviceConnection* implementado por *MetawearDevice*, cuya conexión fue previamente establecida. Luego de que se hayan establecido todas las configuraciones de los sensor y habilitado los productores de datos asíncronos, el sistema procede a iniciar la producción de datos de los distintos sensores.
2. De forma paralela, PARKIBIP crea un analizador de datos o *DataAnalyzer* mediante la fábrica que encapsula dicha función -*DataAnalyzerFactory*-. El tipo de *DataAnalyzer* concreto a instanciar, dependerá de la selección vigente del mismo en el componente de configuraciones. Además, el sistema prepara e inicializa la recepción de eventos de datos de mediciones. En este punto, es importante resaltar que, el *DataAnalyzer* será el responsable de orquestrar el procesamiento de la información a través de la delegación de tareas a ciertos módulos específicos.
3. Cuando un sensor del IMU envía un evento de medición, éste es recibido mediante el componente *DeviceConnection*; que analiza la secuencia de símbolos y la transforma a una estructura legible en la aplicación -modelo *MeasurementModel*-. Luego, realiza un procesamiento previo del modelo de medición, ajustando las unidades al sistema internacional correspondiente y estandarizando la información. Finalmente, dispara un evento de producción de datos a su *DataAnalyzer* asociado.
4. Recibido un evento de medición en el *DataAnalyzer*, éste procede a ejecutar el flujo de procesamiento. En caso de que se requiera calibrar la orientación del dispositivo IMU, el sistema realiza el proceso de Calibración y descarta la medición. En otro caso, envía un evento de medición al manejador o *Handler* del componente *KalmanFilterHandlerThread*, así como también procesa la medición según el *DataAnalyzer* concre-

to elegido. El resultado del procesamiento se propaga como evento al módulo de *Biofeedback* de la aplicación.

5. Primero, es crucial comprender el comportamiento de *KalmanFilterHandlerThread*. Un Thread, representa un hilo de ejecución independiente al hilo principal de una aplicación; mientras que un *HandlerThread* -en Android-, es una extensión del Thread que integra el concepto de *Looper* y de *Handler*. Un *Looper* dentro de un Thread, tiene la capacidad de administrar adecuadamente los múltiples mensajes recibidos mediante su manejador -*Handler*- y su estructura de almacenamiento del tipo Cola FIFO (del inglés, Queue). Entonces, el evento de medición recepcionado por el *Handler* de *KalmanFilterHandlerThread*, es inicialmente puesto en la Cola del *Looper*, y luego atendido por el método *HandleMessage*.
6. La función *HandleMessage*, responsable del procesamiento del filtro de Kalman, comienza ejecutando el detector de cero velocidad -*ZeroVelocityDetector*-, dato requerido como parámetro por Kalman. El ZVD concreto a emplear (e.g. GLRT, MV), se obtiene de la configuración actual del sistema.
7. Luego, es vital la actualización de la orientación con cada medición entrante, no solo por ser pre-condición del algoritmo de Kalman, sino también para minimizar errores en la predicción de los parámetros espacio-temporales futuros. Por consiguiente, se corre el filtro de Orientación propuesto por S.Madwick en [Madgwick \(2010\)](#).
8. Junto a la medición recopilada y los parámetros orientación y momento de velocidad cero -*z_value*-, el sistema ejecuta el filtro de Kalman. Este método es iterativo a medida que su función de actualización es invocada, y arroja como resultado un vector de estados con las estimaciones: posición, velocidad y orientación. Además, con el parámetro *z_value*, el algoritmo compensa/corrigie las estimaciones según si el IMU se encuentra estacionario. Finalmente, el *HandlerThread* de *KalmanFilter* dispara un evento de mensajes hacia el *Handler* del *DataAnalyzer*, encapsulando el vector de estados resultante del procesamiento. A su vez, en paralelo, envía el modelo de medición y los resultados de cada algoritmo numérico -*z_value*, vector de kalman- a sus respectivos conjuntos de datos (en inglés, Data Pool).

9. Cada unidad de *Data Pool*, gestiona el acceso concurrente mediante la mutua exclusión de sus estructuras y recursos (e.g. Colas de mensajes). Los datos son puestos en el pool desde el hilo de ejecución Kalman, y recuperados en batch por un servicio de sistema de fondo.
10. Análogo al punto 5, el mensaje con el vector de estados es recibido por el *Handler* del *DataAnalyzer* y puesto en la Cola del *Looper*. En esta ocasión, el *DataAnalyzer* actúa como pasamanos de información, propagando el estado hacia el modulo de *Biofeedback*.
11. Un servicio en Android, es un componente de una aplicación que puede realizar operaciones de larga ejecución en segundo plano. En PARKIBIP, cumple la responsabilidad del almacenamiento de datos en la base de datos SQL de la aplicación móvil. Con una frecuencia configurable y repetitiva, el servicio obtiene los datos a almacenar desde los Pooles de datos -en batch-, los vacía, y luego almacena los elementos en la base de datos.
12. Concluyendo el proceso, el módulo de *Biofeedback*, responsable de disparar los eventos de estimulación al Paciente, ejecuta las reglas clínicas apropiadas.

5.16. Administración de Identidades

El sistema PARKIBIP fue diseñado específicamente para administrar dos niveles de usuarios, el Terapeuta y el Paciente en rehabilitación. En consecuencia, resulta imprescindible contar con un módulo particular, responsable de gestionar todas las funcionalidades requeridas para llevar adelante la gestión de identidades, proveer seguridad, y especialmente preservar la privacidad del Paciente.

Así, en el presente apartado, se cubren detalles de la solución en cuanto a la administración de identidades, los protocolos que se deben usar y el flujo de autenticación requerido.

PARKIBIP decidió utilizar Auth0 [Auth0 \(2020b\)](#), como su proveedor de identidad como servicio (Identity as a Service, IDaaS, por sus siglas en inglés). Un IDaaS es un servicio basado en la nube para la gestión de identidades y

accesos, a menudo también incluyen el inicio de sesión único (en inglés Single Sign-on, SSO), identidad federada, administración de contraseñas, entre otros. El razonamiento detrás de esta decisión fue la de integrar un servicio de terceras partes dedicado en la materia, y no implementar/mantener todos los aspectos complejos relativos a la seguridad, requeridos en ambientes productivos para datos sensibles. También, fue considerado y preparado en el diseño de la solución, la posibilidad de reemplazar fácilmente el módulo IDaaS por un nuevo micro-servicio (e.g. Agestic). Auth0 es una solución flexible y sencilla para agregar servicios de autenticación y autorización a las aplicaciones. De esta manera, el equipo y la organización pueden evitar el costo, el tiempo y el riesgo que conlleva la creación de su propia solución para autenticar, autorizar, discriminar y obtener información de los usuarios.

En definitiva, la aplicación Android PARKIBIP fue integrada a el proveedor de IDaaS Auth0 mediante el SDK del segundo, desarrollada en el lenguaje JAVA para Android y diseñada según la figura Fig. 5.17.



Figura 5.17: Proceso de autenticación y autorización del Usuario. Se requieren los campos nombre de usuario y contraseña.

En términos generales, para efectuar esta tarea se realizaron las configuraciones:

1. Primero, fue creado un Auth0 tenant. Aquí es donde se configura el uso de Auth0, y donde los activos y recursos, como aplicaciones, conexiones

y perfiles de usuario, se definen, administran y almacenan.

2. Luego, se registró una aplicación nativa o móvil (en inglés, Native/Mobile App) que indica el tipo de aplicación que empleará los servicios en el *Dashboard* de Auth0. Registrado el nuevo cliente, se obtuvo un ID de cliente único, necesario para efectuar llamados a las distintas funciones de la API de Auth0. Otro dato crucial, es el secreto del cliente (del inglés, Client Secret), análogo a una contraseña de aplicación que debe mantenerse confidencial en todo momento.
3. Además, fue configurada la conexión que indica cómo los usuarios de PARKIBIP iniciarán sesión. Para ello, se usó la base de datos en la nube de Auth0.
4. Finalmente, se implementaron las llamadas Reglas (Rules, en inglés). Estas, son funciones escritas en los lenguajes de programación JavaScript o C#, ejecutadas en el mismo servidor de Auth0 justo después de una autenticación exitosa y antes de que el control regrese a la aplicación que realizó la invocación. Por ejemplo, fueron usadas para adjuntar en el Token información codificada del usuario.

Otro aspecto importante, fue la manipulación del protocolo estándar abierto (RFC 7519) *JSON Web Token* (JWT). El token JWT define una forma compacta y autónoma de transmitir información de forma segura entre las partes involucradas en el intercambio, como un objeto JSON. Dicha información puede ser verificada y confiada, ya que la misma está firmada digitalmente. Los JWT se pueden firmar usando el *Client Secret* con el algoritmo de cifrado HMAC, o mediante el par de claves pública/privada empleando los algoritmos criptográficos RSA o ECDSA. En PARKIBIP se empleó el cifrado RS256 como algoritmo de firma digital para los *JsonWebTokens*, en donde se firmará con la clave de firma privada y se puede verificar utilizando la clave de firma pública. Emplear JWT presenta variadas ventajas en comparación con los tokens web simples, y es usado con los siguientes objetivos:

- Autenticación: cuando un usuario inicia sesión con éxito mediante sus credenciales, se devuelve un token de identificación *-id.token-* del tipo JWT.

- Autorización: una vez que un usuario ha iniciado sesión correctamente, PARKIBIP solicita acceso a rutas, servicios o recursos (por ejemplo, API) en nombre de ese usuario. Para hacerlo, en cada solicitud, debe pasar un Token de acceso *-access_token-*.
- Intercambio de información: Los JWT son una buena forma de transmitir información de forma segura entre las partes *-firma digital-*.

La figura Fig. 5.18 presenta un ejemplo de carga útil (del inglés Payload) encapsulada en el *id_token* durante el intercambio de información. Se aprecian metadatos del usuario, información descriptiva y transaccional del mismo. Valga como ejemplo el nombre, email, rol o tipo de usuario en el sistema, fecha de alta, entre otros. Por ende, el módulo de identidad de PARKIBIP, se encarga de interpretar el JSON conforme a acceder a los atributos de interés.

Asimismo, mediante el decodificador, se puede apreciar el algoritmo criptográfico empleado y el tipo de token generado. Véase la figura Fig. 5.19.

```

HEADER: ALGORITHM & TOKEN TYPE

{
  "alg": "RS256",
  "typ": "JWT",
  "kid":
  "M0RCMUE3NjVFNkM2RTAxRENBRUY5NEFDZCMjJERENFRUY0QkE1OA
  "
}

```

Figura 5.19: Ejemplo de encabezado de un token JWT. Especifica el algoritmo de cifrado y el tipo de token generado.

A continuación, se mencionan otras tareas de desarrollo relativas al módulo de identidad:

- Inclusión de la SDK de Auth0 mediante la dependencia *'com.auth0.android:auth0:1.+'*. La librería encapsula la comunicación con las API de autenticación y administración de Auth0.
- Administración de propiedades de acceso: *com_auth0_domain*, *com_auth0_client_id*.

```
PAYLOAD: DATA

{
  "https://com.fing.parkibip/created_at": {
    "created_at": "2020-06-05T01:07:38.322Z"
  },
  "https://com.fing.parkibip/user_metadata": {
    "IdUser": 3,
    "Role": "Terapeuta"
  },
  "https://com.fing.parkibip/app_metadata": {
    "IdUser": 3,
    "Role": "Terapeuta"
  },
  "nickname": "samuelsainz7",
  "name": "Samuel Sainz",
  "picture":
  "https://s.gravatar.com/avatar/1ca4aa48ef30236c3311c29e
  bcf954de?
  s=480&r=pg&d=https%3A%2F%2Fcdn.auth0.com%2Favatars%2Fsa
  .png",
  "updated_at": "2020-11-20T01:55:00.669Z",
  "email": "samuelsainz7@gmail.com",
  "email_verified": false,
  "iss": "https://chuerta.auth0.com/",
  "sub": "auth0|5ed99ada0793080013205d00",
  "aud": "H99jeR1t17FcXJB5uNE4N7XdP71PK93",
  "iat": 1605837300,
  "exp": 1605873300
}
```

Figura 5.18: Ejemplo de carga útil (del inglés Payload) durante el intercambio de información embebida en el token de identificación. Se aprecian metadatos del usuario, información descriptiva y transaccional (e.g. nombre, email, rol o tipo de usuario en el sistema, etc.)

- *Callback:* Un *callback*, es una URL en la aplicación en donde Auth0 redirige al usuario después de que el mismo se ha autenticado.
- *Logout URL:* Una URL de cierre de sesión, es una URL en la aplicación a la que Auth0 puede retornar luego de que el usuario haya cerrado la sesión del servidor de autorización. En PARKIBIP se definió: *fing://parkibip.auth0.com/android/parkibipapp/callback*.
- *Función Login:* Método responsable de realizar el flujo de autenticación y distinción por niveles de usuario (Terapeuta, Paciente).

- Función Logout: Método encargado de administrar el cierre de la sesión activa en la aplicación.
- Administración de credenciales: Componente responsable de administrar los tipos de tokens de Auth0, por ejemplo, token para actualización -*refresh_token*-, token de identidad -*id_token*-, token de acceso -*access_token*-.

El flujo de autenticación empleado a través de Auth0, se basa en *OAuth 2.0*, utilizando una clave de prueba para intercambio de código (del inglés, Proof Key for Code Exchange, conocido por PKCE) La ilustración Fig. 5.20, tomada de Auth0 (2020a) y adaptada al proyecto, refleja el flujo de intercambios de mensajes durante el proceso de autenticación que efectúa PARKIBIP.

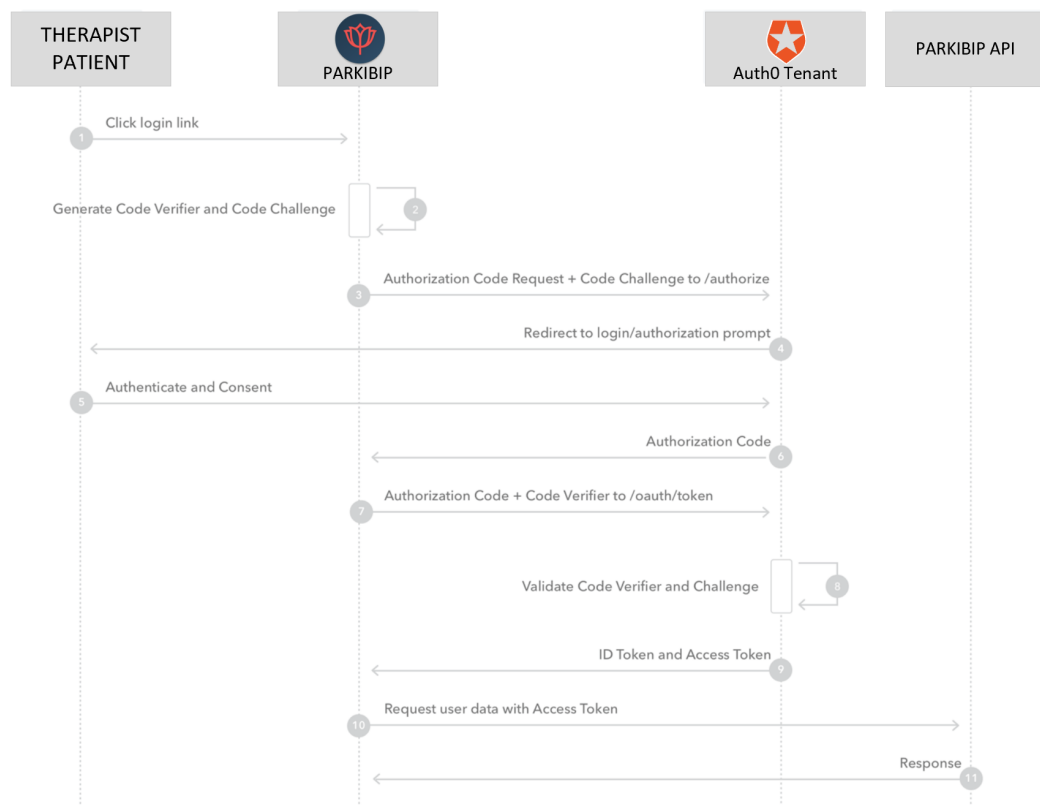


Figura 5.20: Flujo de autenticación y autorización entre PARKIBIP y Auth0 Tenant, a través del protocolo clave de prueba para intercambio de código (PKCE). Cita a Auth0 (2020a) y adaptación a PARKIBIP

En resumen, el procedimiento de autenticación e implícito al usuario sigue el proceso:

1. El Paciente o Terapeuta intenta ingresar al sistema, para ello, hace click en Iniciar Sesión dentro de la aplicación.
2. Mediante la invocación a la función LogIn de el SDK de Auth0, se crea un verificador de código *-code_verifier-* con cifrado aleatorio y a partir de éste se genera un desafío de código *-code_challenge-*.
3. El SDK en Android redirige al usuario al Servidor de autorización de Auth0 invocando el endpoint */authorize* de su API, adjuntando el *code_challenge*.
4. El servidor de Auth0 redirige al usuario a la solicitud de inicio de sesión y autorización.
5. Luego, el usuario se autentica utilizando una de las opciones de inicio de sesión pre-establecidas *-usuario y contraseña-*. En caso de ser el primer inicio de sesión, se despliega una página de consentimiento con los permisos que la aplicación le otorgará a Auth0.
6. El servidor de autorización, almacena el *code_challenge*, luego, redirige al usuario a la aplicación con un código.
7. El SDK envía el código recibido junto a el *code_verifier* -creado en el paso 2- al servidor de autorización de Auth0, endpoint */oauth/token*.
8. El servidor de autorización de Auth0 verifica los datos *code_challenge* y *code_verifier* enviados.
9. Finalmente, Auth0 responde con los tres tokens necesarios: *id_token*, *access_token*, *refresh_token*.

Así, cada vez que un usuario intenta autenticarse, Auth0 verifica la identidad y envía la información requerida a PARKIBIP. Para concluir, la imagen Fig. 5.21, brinda un ejemplo de token de identificación *-id_token-*, resultado de ejecuta el flujo de proceso autenticación y autorización.

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiaWF0IjoxNTE2MjM5MDIyfQ.SflKxwRJSMeKKF2QT4fwpMeJf36P0k6yJV_adQssw5c
```

Figura 5.21: Ejemplo de token de identidad *-id_token-* con formato JWT, obtenido como resultado del flujo 5.20

5.17. Administración de Terapeutas y Pacientes

PARKIBIP tiene como propósito la ejecución de sesiones de rehabilitación para un Paciente y coordinadas por un Terapeuta, por consiguiente es imprescindible la administración de Terapeutas y Pacientes. Es necesario llevar el registro de los mismos para asociar las diferentes sesiones de terapia. Mantener los datos de los pacientes, entre otras cosas, permitió el desarrollo de funcionalidades como ser la visualización del listado de los pacientes para un Terapeuta particular, ver sus sesiones, y luego medir el progreso de cada paciente. Dentro de las historias de usuario del sistema, se desea que, tanto los pacientes como los terapeutas, puedan observar la información vinculada a las distintas sesiones, mantener su registro y compararlas para analizar el progreso entre cada sesión.

La figura Fig. 5.22 presenta el diagrama de clases con el diseño de esta sección del sistema. La entidad *User* -Usuario-, representa tanto a los pacientes como a los terapeutas y mantiene la información descriptiva de éstos, como lo es el nombre, el apellido y un identificador único auto-generado en el sistema. Luego, utilizando el concepto de herencia, se extiende la clase *Patient* -Paciente- que además integra datos particulares que se desean conocer para éste nivel de usuario: edad, documento, observaciones relevantes (e.g. “rigidez en pierna derecha”, “episodios de FoG” o cualquier tipo de observación sobre el paciente que sea relevante). Similar, se tiene una entidad *Therapist* -Terapeuta- que hereda de *User*, y además adjunta la información específica de un terapeuta, por ejemplo el centro médico en el cual trabaja. A su vez, *UsersManager* es una componente que tiene el conocimiento para administrar

los niveles de usuarios e identificar cuál es el paciente activo para las futuras sesiones. A través de *UsersManager* se pueden obtener los distintos usuarios, agregar un nuevo paciente, recuperar un paciente particular a través de su identificador, etc.

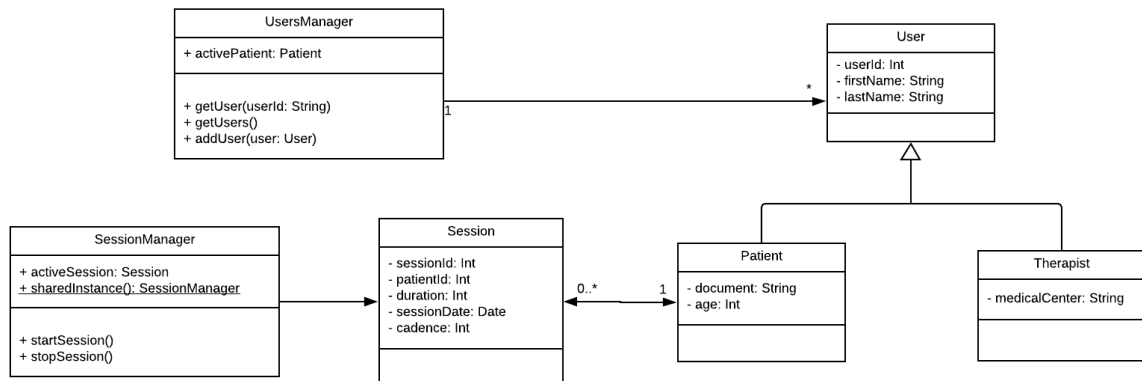


Figura 5.22: Diagrama de clases del módulo de la aplicación Android donde se representan los terapeutas, pacientes y las sesiones realizadas.

Dentro del menú de navegación de la aplicación, para visualizar el listado de pacientes almacenados, se puede consultar el acceso directo “Perfiles”. La figura Fig. 5.23 muestra dicha pantalla. Con el fin de trabajar la amigabilidad del sistema, se le proporciona al usuario la capacidad de filtrar los pacientes utilizando la funcionalidad de búsqueda, accesible a través del icono de búsqueda en la barra superior de navegación de la pantalla.

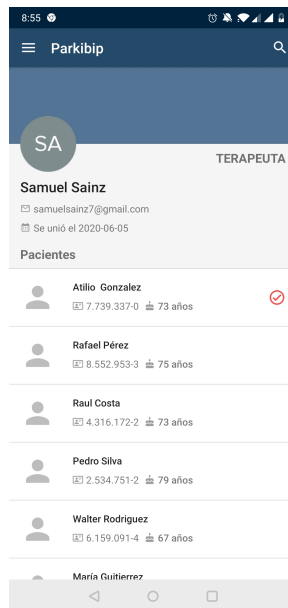


Figura 5.23: Visualización de perfiles de usuarios. Funcionalidades: (i) Ver información descriptiva del usuario logueado, (ii) Ver el listado de pacientes -en caso de ser un terapeuta-, (iii) Seleccionar el paciente activo para futuras sesiones de rehabilitación.

5.18. Administración de Sesiones de rehabilitación

PARKIBIP es un sistema rehabilitador en base a sesiones clínicas, por lo tanto, cada paciente tendrá tantas sesiones PARKIBIP -representadas con la clase *Session*- como sesiones de rehabilitación. Para cada una de estas sesiones, se conoce el identificador del paciente que realizó la sesión, la fecha de realización, la duración, así como los parámetros espacio-temporales resultantes de su procesamiento: cadencia, cantidad de pasos, velocidad promedio, duración -ver figura Fig. 5.24-.

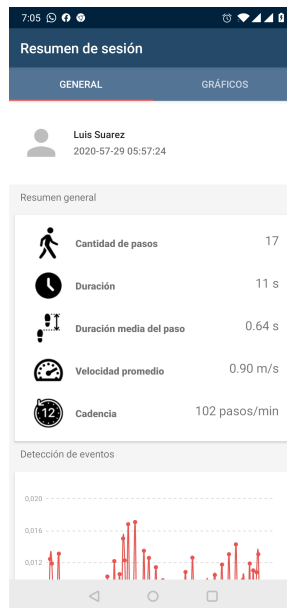


Figura 5.24: Resumen general y detallado de una sesión. Se presenta el comportamiento de la marcha en términos de valores de parámetros espacio-temporales y de forma gráfica.

Por su parte, el componente *SessionsManager* es la clase encargada de administrar las sesiones realizadas, tiene la responsabilidad de crear, listar, remover sesiones, y sobretodo identificar una sesión viva. Durante una sesión PARKIBIP, es el *SessionsManager* quien mantiene la instancia de sesión activa, útil para guardar en la sesión en la base de datos finalizada la sesión, combinarla con las mediciones y con los resultados del post-procesamiento.

Una vez que la sesión es completada, se almacena en la base de datos el resumen genérico y detallado de la misma, generando una pantalla específica en la aplicación conforme a medir los resultados logrados. Además, dicha sesión se puede encontrar en el historial de sesiones de la aplicación, a través del menú de navegación de la aplicación, acceso “Historial de sesiones”. La figura Fig. 5.25 visualiza dicha pantalla.

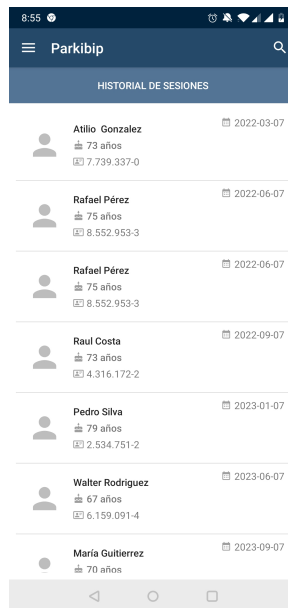


Figura 5.25: Historial de sesiones. Ordenadas cronológicamente con su correspondiente fecha de realización y paciente evaluado. Permite acceder a su resumen mediante su selección.

Cada una de las sesiones listadas, presenta la información del paciente que realizó la sesión. Adicionalmente, el usuario puede filtrar las sesiones digitando el nombre del paciente en la barra de búsqueda situada en la parte superior de la pantalla. Respecto a las sesiones, también se conocen las mediciones y los resultados de los procesamientos realizados en el transcurso de toda la sesión. En la pantalla donde se resume la sesión, se pueden ver las gráficas detalladas en función del tiempo: velocidad, trayectoria y aceleración.

5.19. Diagrama de clases

El objetivo de esta sección es, presentar el diseño de la aplicación Android mediante un diagrama de clases simplificado, ilustrado en la figura Fig. 5.26. Si bien, por simplicidad, se omiten algunas entidades del sistema, en el siguiente diagrama se pueden ver los elementos fundamentales de la aplicación móvil. El diagrama brinda una visión general de cómo está diseñado el sistema, así como sus responsabilidades y la interacción entre ellos.

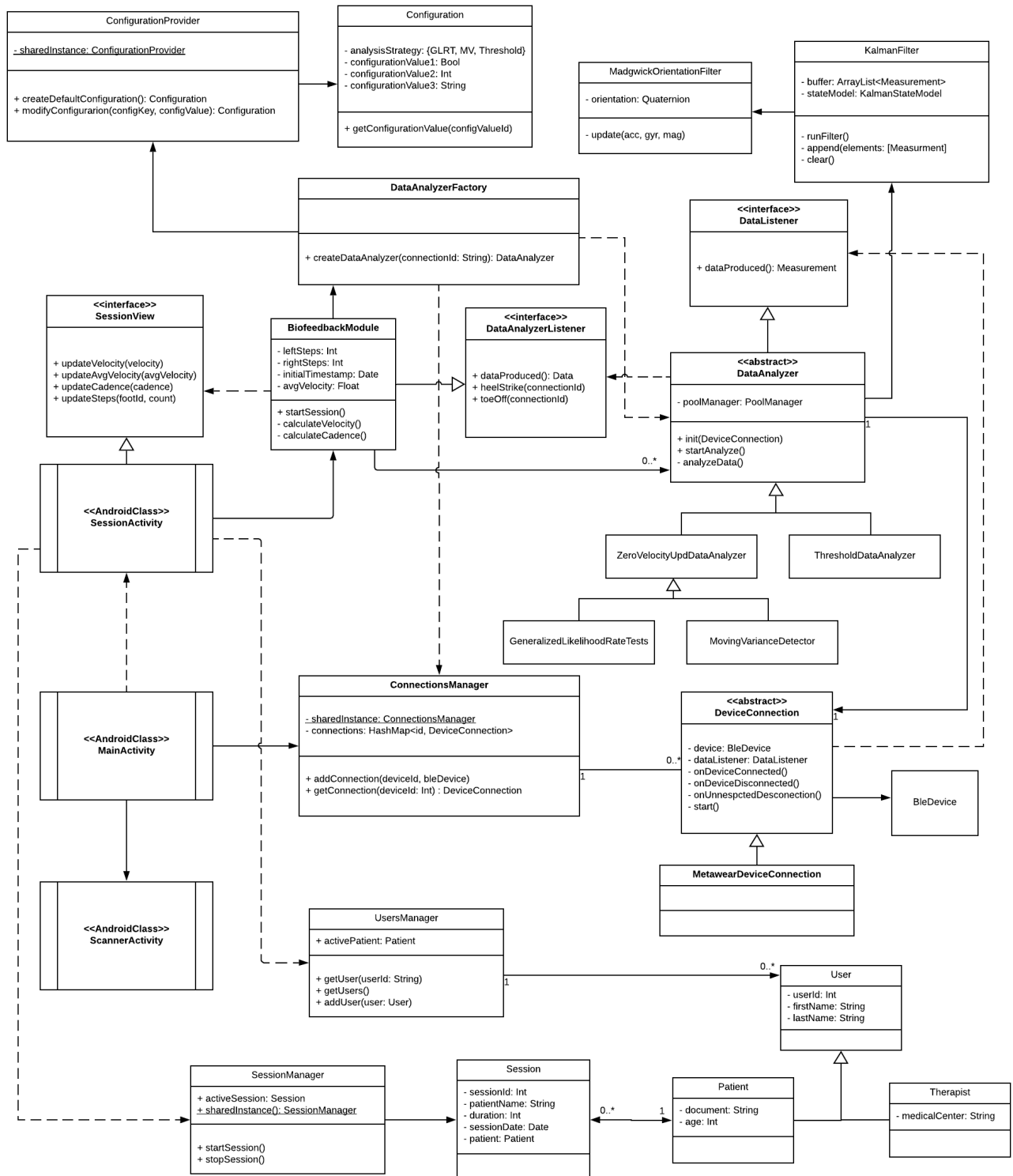


Figura 5.26: Diagrama de clases de la aplicación Android PARKIBIP. Se muestran en este diagrama los componentes esenciales del sistema y se omiten varios por simplicidad.

Para diseñar PARKIBIP se tuvieron en cuenta varios principios de diseño que favorecen las cualidades del sistema como mantenibilidad, escalabilidad, calidad, entre otras. Como se puede ver en el diagrama, existe una alta cohesión de los componentes y un bajo acoplamiento entre éstos. Se intentó no repetir código, mantener responsabilidades únicas, abstraer comportamiento mediante interfaces y clases abstractas. En PARKIBIP se tuvieron en cuenta los principios de diseño SOLID. SOLID es un acrónimo que resume los cinco principios de diseño de diagramas de clases [Chebanyuk and Markov \(2016\)](#):

- **Single Responsibility:** Responsabilidad única
- **Open-Closed:** Abierto-Cerrado
- **Liskov Substitution:** Principio de Sustitución de Liskov
- **Interface Segregation:** Principio de Segregación de Interfaces
- **Dependency Inversion:** Inversión de la Dependencia

El principio de **responsabilidad única** es quizás el más simple de identificar observando el diagrama de clases, ya que es fácil notar que cada clase del sistema tiene una única función u objetivo. Para mencionar un ejemplo, se puede ver que *ConfigurationProvider* es encargado de proveer las configuraciones, administrarlas, modificarlas, etc; pero la configuración en sí está dada por la clase *Configuration*. *DataAnalyzerFactory* es una clase que consulta a *ConfigurationProvider* por una instancia de *Configuration* para poder crear la instancia de *DataAnalyzer* que corresponda. Como se puede ver, cada uno de los actores tiene una única responsabilidad y esto está reflejado en **todas** las clases del sistema.

Por otro lado, se tuvo en cuenta también el principio **Open-Closed**, que consiste en lo siguiente: “Las entidades de software (clases, módulos, funciones, etc.) deben estar abiertas para su extensión, pero cerradas para modificaciones”. Esto se ve reflejado en el diseño de PARKIBIP de forma clara en la implementación de las conexiones para cada dispositivo, donde se define el comportamiento en la clase *DeviceConnection* y se extiende el mismo para el dispositivo *MetaMotionR* en la clase *MetaWearDeviceConnection*. Esto permite que el sistema sea fácilmente escalable o extensible a otros proveedores de

hardware, que sea mantenible y que además se pueda reutilizar código. También se aplicó este principio en la implementación de los algoritmos de análisis de los datos para detección de eventos, en la definición de usuarios, entre otros.

El principio de sustitución de Liskov complementa el principio de Open-Closed, y lo vemos aplicado en PARKIBIP en los mismos elementos que éste. El principio define que los objetos de una superclase serán reemplazables por objetos de sus subclasses sin interrumpir la aplicación. Eso requiere que los objetos de sus subclasses se comporten de la misma manera que los objetos de su superclase.

El principio de **segregación de interfaces** dice que “No se debe obligar a los clientes a depender de interfaces que no utilicen”. En PARKIBIP esto se cumplió complementando el principio de responsabilidad única, ya que se declararon las interfaces encapsulando una única funcionalidad por lo que ningún componente implementa una interfaz con funcionalidades que no utiliza.

Dependency Inversion refiere a lo que se conoce como inyección de dependencias. Es decir, si cierta clase depende de otro componente, esta dependencia debe ser inyectada en su constructor. Esto permite que las clases sean más testeables a la hora de realizar tests unitarios, donde es necesario aislar su funcionalidad para no depender de otros componentes. La inyección de dependencia nos permite este aislamiento, ya que para los tests unitarios podemos construir nuestra clase inyectando dependencias simuladas. Esto lo utilizamos repetidamente en PARKIBIP inyectando el Contexto de la aplicación Android. Context es una interfaz con información global sobre el entorno de la aplicación. Esta es una clase abstracta cuya implementación es proporcionada por el sistema Android. Permite el acceso a clases y recursos específicos de la aplicación, así como llamadas ascendentes para operaciones a nivel de la aplicación, como actividades de lanzamiento, transmisión y recepción de Intents, etc. [Google, LLC. \(2020\)](#).

5.20. SQLite: Almacenamiento de datos

Con el fin de gestionar el principal volumen de información, resultante de las sucesivas mediciones de datos obtenidos por los diversos sensores, se optó por implementar una base de datos que actúe local al dispositivo móvil. Almacenar los datos maestros (e.g. algoritmos de detección, parámetros, configuraciones y meta-data), como también los transaccionales (e.g. mediciones, estados de

Kalman Filter o actualizaciones de Zero Velocity) en una base de datos es ideal en PARKIBIP para lograr:

- Estructurar la información.
- Facilitar el procesamiento de la información.
- Aumentar o nivelar el rendimiento del sistema -computo sincrónico y asincrónico-, en particular dispositivos con baja capacidad de computo.
- Accesibilidad a datos repetitivos.
- Centralizar e integrar los datos provenientes de múltiples fuentes - dispositivos IMU y sensores inerciales-.
- Gestionar un elevado volumen de datos -en memoria sería inviable-. Por ejemplo, frecuencias altas desde 600 Hz.

Se desarrolló una base de datos en el lenguaje estándar SQL, en Android denominado SQLite. Este motor de base de datos liviano de SQL, fue diseñado específicamente para mejorar el rendimiento de sistemas con poca capacidad de procesamiento y almacenamiento. Las APIs requeridas para utilizar una base de datos en Android están disponibles en el paquete *android.database.sqlite*.

Uno de los principios fundamentales de las bases de datos SQL es la especificación, mediante el modelado de un esquema entidad-relación: una declaración formal de la manera en la que la base de datos está organizada. El diagrama Fig. 5.27, permite visualizar las principales entidades conceptuales y sus relaciones, que luego brindan soporte al sistema PARKIBIP.

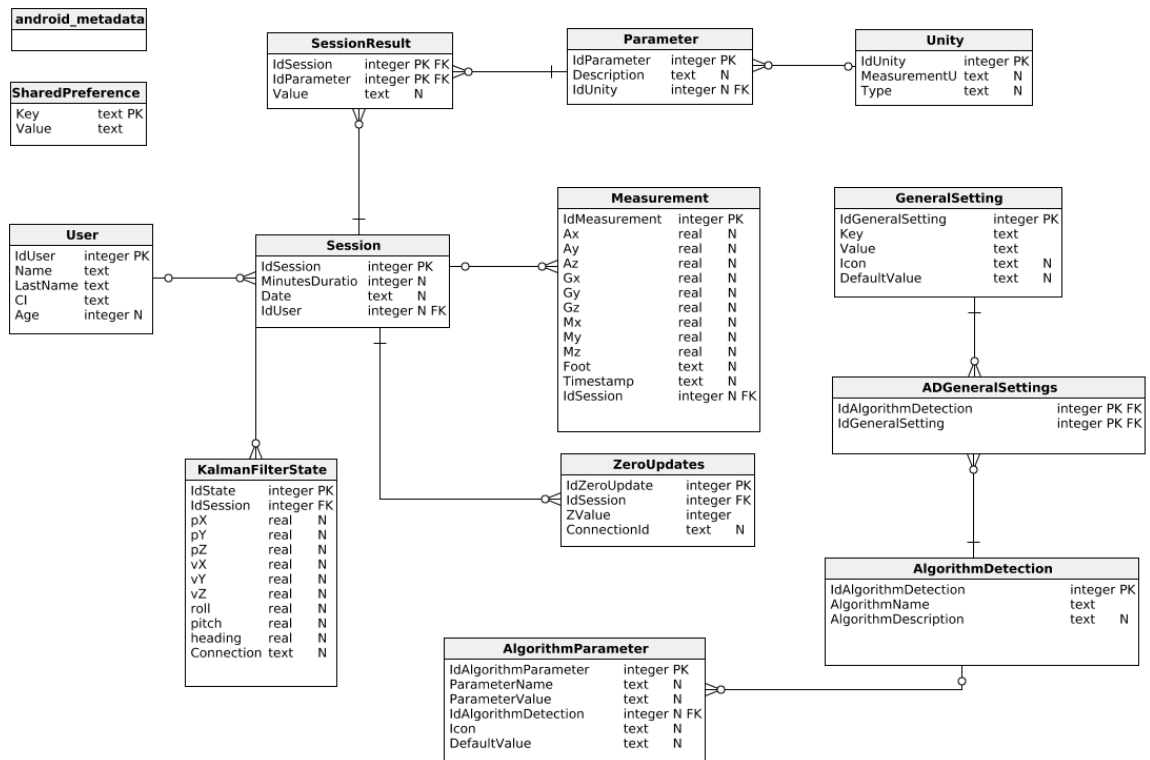


Figura 5.27: Modelo conceptual entidad-relación de PARKIBIP en Android.

Así, en el lenguaje Java de Android, resulta útil crear una clase complementaria, denominada clase de contratos *-ParkibipContract-*, que indica explícitamente el diseño del esquema de forma sistemática y auto-documentada. Asimismo, facilita y estandariza el acceso a los datos estructurados mediante el flujo de actividades.

Por otro lado, de forma paralela, se empleó el Software “DB Browser for SQLite”, siendo la herramienta principal para la gestión de bases de datos en el mencionado lenguaje. Durante el desarrollo, fue provechoso conectarse a la instancia para chequear consistencias, manipular los registros, crear tablas y relaciones, generar restricciones entre los datos, entre otras tareas.

En el apéndice [Entidades de la base de datos relacional SQLite](#) se presentan las estructuras de las entidades contempladas en el modelo conceptual.

Capítulo 6

Pruebas y resultados

Este apartado, refiere a las pruebas realizadas al sistema PARKIBIP, conformado por la aplicación móvil y ambos dispositivos IMU.

Por un lado pruebas estáticas, exploratorias, funcionales y para-funcionales. El propósito fue aumentar la confiabilidad del mismo a través de su evaluación de calidad; mediante abordajes que verifiquen la legibilidad del código, la escalabilidad del sistema, la correcta interoperabilidad entre componentes y, sobre todo, la detección temprana de incidentes que puedan generar una falla posterior (e.g. omitir eventos de desconexión de los IMU, ocasiona el quiebre de la aplicación).

Por otro lado, se evaluó a PARKIBIP en funcionamiento a través de pruebas de usabilidad y eficiencia. El objetivo de estas pruebas es comparar la performance del algoritmo final -resultado de la combinación de métodos- frente a datos conocidos, y estudiar en qué casos particulares el algoritmo tiene comportamientos diferentes (e.g. en términos de error relativo y tiempos de ejecución)

6.1. Pruebas Estáticas: Peer Code Review

En primer lugar, se empleó durante todo el proceso de desarrollo una técnica de verificación estática, denominada revisión de código por pares (del inglés, Peer Code Review). Esta práctica sistemática de desarrollo consiste en chequear manualmente el código y sus cambios de a pares, a fin de encontrar problemas, errores y vulnerabilidades, tan pronto como sea posible. Al ser una verificación estática, se realiza sobre el código sin ejecutarlo. Acorde con la

herramienta Gitlab, empleada en PARKIBIP para gestionar el repositorio y controlar el versionado de código (ver [Versionado de código](#)), fue de gran utilidad su herramienta integrada de revisión de código ligero. El procedimiento establecido fue, con cada modificación de código eventualmente finalizada, se le solicita al otro miembro del equipo la inclusión de dichos cambios, mediante el comando *Pull Request*. Luego, el revisor asignado y orientado a las oportunidades de mejora y a la crítica constructiva; comienza a verificar el cumplimiento de los estándares de codificación, identificar posibles errores en el código fuente, detectar posibles problemas de diseño en la nueva solución. La herramienta, habilita a realizar comentarios línea a línea del código, obtener una vista previa de los cambios en comparación con el código vigente del repositorio -para ver qué se está proponiendo-, mantener una discusión activa del código, entre otras cosas. A efectos de comprender lo antedicho, se propone la figura Fig. 6.1 como ejemplo de revisión. En la misma, se puede visualizar con color verde las nuevas modificaciones y con rojo las eliminaciones de código, además de un comentario del revisor asignado.


81	83	
82		-
83	84	<code>public void sendKalmanFilterMessage(MeasurementModel measurementModel){</code>
84	85	<code> Message msg = Message.obtain();</code>
85	86	<code> Bundle data = new Bundle();</code>
86		- <code> data.putParcelable(kalmanFilterThread.KEY_MEASURE,measurementModel);</code>
	87	+ <code> data.putParcelable(KEY_MEASURE,measurementModel);</code>
87	88	<code> msg.setData(data);</code>


Carlos Enrique Huerta Santana @carlos.huerta · right now Maintainer

Seria apropiado migrar la clave del Handler "KEY_MEASURE" a una constante global.

Figura 6.1: Ejemplo de revisión de código entre pares sobre la petición de modificación hacia la rama principal de desarrollo -*pull request*-. Con color verde se visualizan los nuevos cambios y con rojo las eliminaciones de código, además de un comentario del revisor asignado.

De forma adicional, la figura Fig. 6.2, muestra un resultado de un proceso de verificación y validación estática con la estrategia *Code Review* con la herramienta GitLab. Se observa la aprobación y posterior mezclado de código por parte del revisor bajo el indicador *Merged*, la cantidad y descripción de cambios incluidos, así como también el origen y el destino de los cambios.

Merged Opened 3 months ago by  Carlos Enrique Huerta Santana Edit Report abuse

Maintainer





Feature/integration fixes


Overview 0 Commits 2 Changes 9



- fix: getUserAcceleration
- fix: Madwick beta to 5
- Split updateOrientation from linnearAccelerationMethod
- New DataFuser with concurrent management and Magnetomer


MetawearConnection

- Add code to manage Magnetometer
- create method produceDataWithMagnetometer()
- Split startProducerData code to configureModules() and produceData()




 Request to merge `feature/integration-f...`  into `deve1op`  

 Approval is optional

 Merged by  Samuel Axl Sainz Sanchez 3 months ago Revert Cherry-pick

The changes were merged into `deve1op` with `bed8e589` 

You can delete the source branch now Delete source branch

 1  0 



Oldest first  Show all activity 

Figura 6.2: Resultado de un proceso de verificación y validación estática con la estrategia *Code Review* en la herramienta GitLab

6.2. Pruebas funcionales: Unitarias e Integración

Por otro lado, para realizar las pruebas funcionales, se siguió una estrategia de Testing Planificado, a partir del diseño de casos de prueba y su posterior ejecución. Según la etapa particular del proceso de desarrollo, fueron implementadas diferentes pruebas:

- Pruebas Unitarias: Las pruebas unitarias tienen por objetivo aislar cada componente individual del sistema, luego de su desarrollo, y demostrar que su funcionalidad y estructura es correcta.
- Pruebas de Integración: Las pruebas de integración tienen como propósito verificar la correcta interoperabilidad entre los distintos componentes del sistema, una vez que han sido probados mediante las pruebas unitarias. La idea por detrás, es comprobar que dichos componentes interactúan correctamente a través de sus interfaces, tanto internas como externas, cubren la funcionalidad establecida y se ajustan a los requisitos no funcionales

Así, el objetivo de esta actividad fue detectar defectos, problemas o anomalías, inyectadas en el software PARKIBIP ya sea por omisión o por error. Para llevar a cabo esta tarea, se utilizó el entorno de trabajo (en inglés, framework) JUnit para Android, versión 4.12. JUnit brinda un marco de ejecución para las pruebas, corre fácilmente los conjuntos de prueba definidos, presenta una API que facilitan la comparación y el manejo de errores, y el armado de cada prueba. Por consiguiente, se desarrolló en el lenguaje Android los llamados *Test Suites*, representando un conjunto de pruebas relacionadas que comparten el mismo *Fixture* (i.e. pre-condiciones o estado necesario para la prueba). Dichas pruebas del *Test Suite* son independientes y representadas mediante una clase, bajo la anotación *@Test* para indicar que es una prueba. La modalidad utilizada al probar fue la de caja negra, en la cual se comprueba el correcto funcionamiento de los componentes, analizando únicamente las entradas/salidas y verificando que el resultado es el esperado. Para ello, las anotaciones de JUnit utilizadas fueron:

- *@Suite*: Esta anotación permite agrupar algunos casos de prueba unitarios y ejecutarlos juntos
- *@Test*: Etiqueta un método para que sea considerado como caso de prueba
- *@Before*: Indica que un método particular se ejecuta previo a la prueba. En general, se emplea para inicializar el contexto del test (e.g. probar el filtro de Kalman requiere conocer previamente la orientación dada por al filtro de Orientación)

- *@After*: Indica que un método particular se ejecuta luego de una prueba. En caso de asignarse recursos con la anotación *@Before*, este método permite liberarlos después de que se ejecute la prueba (i.e. libera el contexto)
- *assert*: Función que verifica el comportamiento o estado de la unidad bajo prueba

Con el propósito de ejemplificar y presentar los casos de prueba esenciales en PARKIBIP, se sintetizan algunas *Suites* que encapsulan ciertas pruebas -por convención, todas las clases de prueba finalizan con la palabra “Test”-:

- *Quaternion.suite*: Contiene el conjunto de casos de prueba relacionados a la representación de la orientación en el formato de quaterion. Las clases desarrolladas fueron las siguientes:
 - *CalibrationQuaternionTest*. Con la anotación *@Before*, se define el conjunto de observaciones de prueba - tomadas de un archivo auxiliar- y el valor esperado del quaternion para dichas mediciones. Luego, se ejecuta la prueba comenzando por la estandarización al SISTEMA INTERNACIONAL DE UNIDADES (SI). Se recorren 20 mediciones, aplicando la función de calibración -perteneciente a la librería de algoritmos implementados, desde ahora *APIAlgorithm*-. Finalmente, se compara el resultado esperado frente cada una de las 4 componentes del vector quaternion obtenido, con el comando de igualdad *assertEquals*
 - *QuaternionToRotationMatrixTest*. Dados un quaternion y la matriz de rotacion -conocidos-, definidos en *@Before*, se corre la prueba invocando la función *ConvertToRotationMatrix()* de la *APIAlgorithm*. Las entradas de la matriz obtenida y esperada son evaluadas una a una con *assertEquals*.
 - *QuaternionToEulerAngleTest*. Procedimiento análogo a *QuaternionToRotationMatrixTest*, con la distinción que, en vez de una matriz es un vector.
 - *QuaternionToConjugateTest*. Similar a los casos previos, dados un quaternion y su conjugado -conocidos-, establecidos en *@Before*, se ejecuta la prueba invocando la función *ConvertToConjugate()* de la

APIAlgorithm. Luego, las entradas de los vectores de quaternions conjugados son comparadas mediante *assertEquals*.

- *QuaternionTestSuite*. Mediante su definición, le permite al entorno de desarrollo identificar y correr todas las pruebas pertenecientes al módulo o *Suite*
- *Orientation.suite*: Contiene el conjunto de casos de prueba relacionados al calculo de la orientación mediante el filtro de orientación en 5.8. Las clases desarrolladas fueron:
 - *UpdateOrientationTest*. Dado el quaternion esperado, que representa la orientación del sensor relativa al marco Tierra, y un conjunto de 21 mediciones, ambos definidos en *@Before*. Se recorren 20 mediciones, aplicando la función de calibración *calibrateQuaternion* de la *APIAlgorithm*, se inicializa el filtro de orientación para dicho quaternion, y luego se actualiza el filtro de orientación con la ultima medición, *updateOrientation(measure)*. Finalmente, se comparan las componentes de los vectores esperado y obtenido.
 - *OrientationTestSuite*. Mediante su definición, le permite al entorno de desarrollo identificar y correr todas las pruebas pertenecientes a la *Suite*.
- *ZeroVelocityDetector.suite*: Todas las pruebas asociadas al calculo de momentos de cero velocidad. Las clases desarrolladas fueron:
 - *MVZeroVelocityDetectorTest*: Primero, el test establece el algoritmo de detección de velocidad cero *MovingVarianceDetector*, bajo el soporte del controlador de configuraciones *PARKIBIP ConfigurationProvider*. Luego, el test almacena en memoria el conjunto de valores de cero velocidad de prueba -valores esperados- y el conjunto de mediciones de prueba -tomadas de un archivo auxiliar- (todo en *@Before*). Se ejecuta el test iterando sobre las mediciones y llamando a la función *zeroVelocityDetector.update(measure)* del detector particular. Para concluir, itera cada resultado y los compara contra su correspondiente valor esperado.
 - *GlrtZeroVelocityDetectorTest*. Idéntico al caso previo, con la salvedad que el detector elegido es *GLRTDetector*.

- *ZeroVelocityDetectorTestSuite*. Mediante su definición, le permite al entorno de desarrollo identificar y correr todas las pruebas pertenecientes a la *Suite ZeroVelocityDetector*
- *KalmanFilter.suite*: Modulo de prueba complejo, combina diversos algoritmos para construir el vector de estados de Kalman. Las clases desarrolladas fueron:
 - *KalmanFilterAlgorithmTest*. El caso de prueba emplea un conjunto de mediciones inerciales de acceso libre, cuyo resultado velocidad promedio es conocido. Así, bajo la anotación *@Before*, el test levanta las mediciones en memoria -tomadas de un archivo auxiliar-, y establece: (i) el algoritmo GLRT -como algoritmo de cero velocidad-, (ii) el resultado esperado, (iii) un valor θ , como la desviación aceptable para la estimación, (iv) valores de frecuencias a utilizar. Luego, inicia la ejecución del test. Primero, calibra la orientación inicial del dispositivo, después, itera sobre cada medición del conjunto de datos realizando los pasos:
 - Ejecuta la función $z = \text{zeroVelocityDetector.update(measure)}$ del detector, para identificar momentos de cero velocidad en una ventana de tiempo
 - Actualiza el quaternion del filtro de Orientación, bajo el método $\text{updateOrientation(measure)}$
 - Ejecuta el filtro de Kalman llamando a la función $\text{state} = \text{kalmanFilterZupdt.runFilter(measure,z)}$
 - Cada vector de estado obtenido es puesto en una estructura de almacenamiento en memoria

Finalmente, calcula la velocidad promedio como la sumatoria de normas de las velocidades instantáneas tridimensionales sobre la cardinalidad del conjunto, para luego comparar la velocidad media esperada con la obtenida. Se utiliza el comando *assertEquals* con el valor θ , que indica la diferencia máxima entre las medidas para que el caso sea satisfactorio.

- *KalmanFilterAlgorithmAUPTest*. **Este caso de test es esencial para la aplicación, ya que integra y combina todos los módulos necesarios para el funcionamiento de PARKIBIP**

en productivo. Es decir, emplea datos reales de PARKIBIP a partir de experimentos en la Asociación Uruguaya de Parkinson (AUP), y el flujo de comunicación entre los componentes finales. Su ejecución es análoga al caso previo, con la salvedad en los datos de prueba y frecuencias empleadas.

- *KalmanFilterTestSuite*. Mediante su definición, le permite al entorno de desarrollo identificar y correr todas las pruebas pertenecientes a la *Suite*.

Viendo el flujo de realización de pruebas, se puede apreciar la aplicación de una estrategia de integración ascendente (del inglés, Bottom-Up). Es decir, se construyen primero los componentes menos dependientes y luego se van integrando para construir otros de mayor jerarquía. Para el caso, primero lo vinculado a Quaternions y el filtro de orientación, siguiendo por los detectores de velocidad cero, para finalizar con el filtro de Kalman.

Por otra parte, con el propósito de complementar las actividades del proceso de pruebas funcionales, el apéndice **Pruebas funcionales: Matriz de casos de prueba** describe los 70 casos diseñados y ejecutados, a través de una matriz de casos de prueba. Cada uno incluyendo un identificador, su escenario, caso de test, pre-condiciones, prioridad, etapa, resultado esperado, entre otros.

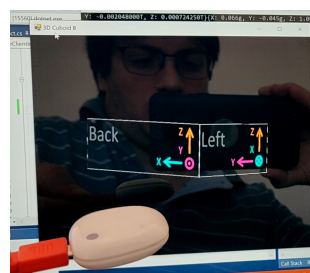
6.3. Prueba funcional: Filtro de Orientación PARKIBIP

A efectos de probar adecuadamente el filtro de Orientación, con datos reales, provenientes del IMU de PARKIBIP; se desarrolló una aplicación gráfica de escritorio, a modo de prueba de concepto (siglas en inglés, PoC, Proof of Concept). Esta, PoC se encarga de rotar un cubo 3D en tiempo real, según las mediciones que va recopilando del IMU conectado.

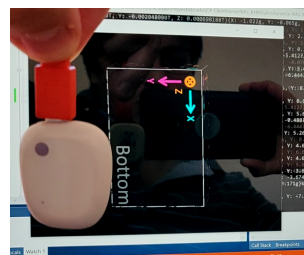
De esta manera, se desarrolló en el lenguaje .NET, un sistema Cliente y Servidor. El Servidor, integrado a un IMU, recopila los datos inerciales - acelerómetro, giroscopio, magnetómetro-, y se los envía a el Cliente. Al recibir los datos, el Cliente, ejecuta el filtro de Orientación obteniendo en cada instante un quaternion. A su vez, el Cliente presenta una interfaz gráfica con un cubo en 3 dimensiones. Cuando el Cliente finaliza el cálculo de un quaternion, dispara

un evento con dicho resultado que desencadena la rotación del cubo según los valores estimados.

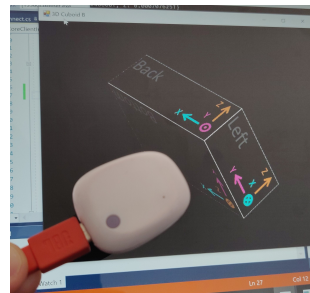
Entonces, la funcionalidad de la PoC, es emular en 3D y en tiempo real, la misma orientación que el dispositivo IMU, como si fuera un espejo. Así, es fácil corroborar la correctitud del cómputo de la orientación observando las semejanzas del IMU y del cubo. Las sub-figuras de la figura Fig. 6.3, emulan en tiempo real las rotaciones del IMU mediante el cubo 3D. Además, se adjunta una grabación -video- circular de este proceso ¹.



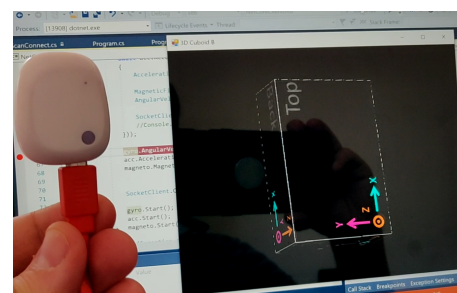
(a) Orientación regular del IMU-Cubo3D en el marco terrestre.



(b) Orientación inferior del IMU-Cubo3D en el marco terrestre.



(c) Orientación inclinada del IMU-Cubo3D en el marco terrestre.



(d) Orientación superior del IMU-Cubo3D en el marco terrestre.

Figura 6.3: Demostración gráfica de la orientación del IMU en tiempo real utilizando un cubo 3D mediante aplicación de escritorio .NET

Debido a las complejidades numéricas de PARKIBIP, dicha PoC se propuso con la idea de:

- Facilitar las pruebas en un entorno de desarrollo mas simple, ágil y sin la necesidad de un teléfono inteligente
- Probar diversas frecuencias y configuraciones sobre el IMU que impacten en la orientación.

¹Las grabaciones circulares se encuentran disponibles en el sitio [PARKIBIP-grabaciones-circulares](#)

- Probar el método de consolidación de datos o Data Fuser
- Identificar y solucionar un error “indescifrable” por los integrantes del proyecto en la aplicación móvil Android. El error fue detectado al graficar en tiempo real la velocidad instantánea del usuario en función del tiempo, en el cual se observó un patrón incorrecto. Aplicando ingeniería inversa, se evaluó primero el filtro de Kalman -sin errores-, luego las detecciones de cero velocidad -sin errores-, y por último la aceleración del Usuario, en donde se identificó el problema. Cada medición tridimensional de aceleración, recopilada del sensor acelerómetro, es rotada con un quaternion desde el marco Sensor al marco Tierra -previo cálculo del quaternion con el filtro de orientación-. Con la orientación -quaternion-, la aceleración era rotada al marco Tierra y las componentes gravitatorias eran extraídas, logrando así la aceleración del usuario. El error de la orientación, era propagado hacia la aceleración del usuario, y ésta a la velocidad. Este comportamiento ocurría únicamente con las mediciones del IMU de PARKIBIP

Finalmente, luego de diversas pruebas, se detectó que el factor β del filtro de Orientación era muy pequeño -0.1-, y por ende la convergencia en la estimación era muy lenta. Así, fueron probados distintos valores, y el valor óptimo alcanzado para PARKIBIP fue $\beta = 5$.

Por lo tanto, la PoC se justifica plenamente con la solución del error y el entendimiento general de los IMU.

6.4. Pruebas Exploratorias

De forma complementaria a las pruebas funcionales, se llevó a cabo la metodología de Testing Exploratorio. Esta estrategia, le permite al explorador navegar sobre el sistema para identificar desvíos o riesgos. En otras palabras, se invierte menos esfuerzo en la planificación temprana de pruebas, y el diseño, aprendizaje y ejecución se desarrollan en simultáneo.

Se considera oportuno emplear este tipo de técnica, combinada con las pruebas funcionales, a los sistemas distribuidos como lo es PARKIBIP. Esto se fundamenta en las eventuales complejidades para desarrollar pruebas funcionales en/con los sistemas de terceras partes -externos-. Asimismo, fueron de

utilidad para simular flujos de usabilidad que podrían seguir los usuarios de la aplicación y probarlos

De acuerdo con la metodología de pruebas exploratorias basadas en sesiones, fueron definidas las misiones exploratorias particulares a investigar, conforme a adquirir experiencia y confiabilidad en el sistema. Entonces para cada Sesión se registro:

- Misión: Representa el objetivo de la sesión de forma concisa y clara
- Áreas: Entornos de ejecución de la prueba. Por ejemplo, un sistema operativo
- Duración de la Sesión: Se consideran los tipos: Sesión corta -30 a 45 minutos-, Sesión media -45 a 90 minutos-, y Sesión larga -90 a 120 minutos-
- Nota de prueba: Luego de ejecutar el test, se registran las observaciones sobre las decisiones tomadas
- Análisis y reporte de errores (en inglés, Bug): Se registran los identificadores y la descripción de los defectos encontrados

En el apéndice [Pruebas Exploratorias: Tabla de sesiones exploratorias](#) se puede ver un resumen de las sesiones exploratorias ejecutadas en esta etapa (32 sesiones), así como también los errores encontrados a través de éstas.

6.5. Pruebas para-funcionales: Performance y Estrés

En las secciones anteriores se detallaron las actividades de pruebas realizadas para probar las distintas funcionalidades de PARKIBIP. Sin embargo, la calidad de un sistema no está dada únicamente por sus funcionalidades, sino también por su rendimiento. PARKIBIP es un sistema en tiempo real, por lo que una degradación en el rendimiento puede implicar demoras en el procesamiento que terminen produciendo un mal funcionamiento del sistema (e.g. estimulando con un retraso que sea perceptible, anulando todas las posibilidades de reeducar la marcha adecuadamente).

En una primer etapa se identificaron los puntos en donde podrían existir fallas en el rendimiento. Para esto, se plantearon las siguientes preguntas:

- ¿Es correcta la utilización de hilos de procesamiento (en inglés, Threads) para recepción y procesamiento de datos desde los sensores?
- ¿Se realiza un uso adecuado de hilos en paralelo para los accesos a la base de datos?
- ¿Existe una degradación del sistema al procesar los datos provenientes de dos IMU en paralelo? Si existe, ¿se puede minimizar?
- ¿Cuál es la frecuencia máxima de procesamiento de datos provenientes de los sensores para la que no existe degradación en el rendimiento?
- ¿Se están realizando operaciones “pesadas” en el hilo principal causando que la interacción con la aplicación resulte lenta o que no responda adecuadamente?

Para aplicaciones Android, existe una herramienta de Android Studio -IDE utilizado para el desarrollo de la aplicación- llamada *Profiler*. Esta herramienta permite realizar diagnósticos en tiempo real sobre la aplicación Android durante la ejecución de la misma. El objetivo principal de esta herramienta o conjunto de herramientas es el diagnóstico, optimización y solución de problemas de desempeño en aplicaciones de la plataforma.

Para PARKIBIP, se utilizó Profiler con el objetivo de monitorear el correcto uso de los recursos durante ciertas pruebas de carga y, de esta forma, contestar a las preguntas que se plantearon anteriormente. Se listan a continuación los recursos analizados:

- Unidad de Procesamiento Central (CPU, Central Processing Unit)
 - Utilización de la CPU
 - Utilización de los distintos Threads de PARKIBIP
- Memoria
 - Utilización de la memoria RAM
 - Seguimiento de las asignaciones de memoria
- Utilización de la batería

CPU

Es importante conocer el uso que se está realizando del procesador. Dado que PARKIBIP es un sistema en tiempo en real, demoras en el procesamiento podrían causar un retraso en todo el flujo de la sesión activa -detallado en la sección [Caso de uso: Flujo automatizado de Sesión Activa PARKIBIP](#)-.

Se debe considerar también que PARKIBIP es un sistema multi-hilos, como se explicó a lo largo de todo el capítulo [Implementación](#). Las aplicaciones Android tienen por defecto un hilo principal, el cual debe ser utilizado para tareas de renderizado de la interfaz o responder a interacciones del usuario (e.g. ante la acción de *scrolling*, es decir, deslizar el contenido de la pantalla). Una buena práctica es no ejecutar tareas con alto requerimiento del procesador en este hilo, y dedicarlo únicamente a las actualizaciones de interfaz gráfica. Un mal uso del hilo principal puede resultar en que la aplicación no responda a las interacciones del usuario, concluyendo en una muy mala experiencia para éste. En el caso de PARKIBIP, además, podría implicar una demora al actualizar los parámetros que se muestran en pantalla, generando que éstos sean inconsistentes con lo que está realizando el paciente.

Para analizar el uso de la CPU y los distintos hilos de ejecución, *Profiler* provee una herramienta extremadamente útil que despliega gráficas en tiempo real indicando el porcentaje de uso de la CPU en cada instante de tiempo. Este análisis se realizó en 3 escenarios posibles, en un dispositivo OnePlus 5T -API 28-, con dos dispositivos IMU y modificando la frecuencia de datos para cada escenario:

- Escenario 1: Frecuencia de datos de 100Hz
- Escenario 2: Frecuencia de datos de 200Hz
- Escenario 3: Frecuencia de datos de 400Hz

Para los tres escenarios el uso de la CPU no varió significativamente. En la figura [Fig. 6.4](#) se muestra el uso de la CPU durante una sesión donde se están procesando los datos en tiempo real, bajo las condiciones del escenario 3 -2 dispositivos IMU enviando datos a 400Hz-. En el escenario 1, el pico máximo de utilización fue de 46 %. En el escenario 2, el pico máximo de utilización fue de 52 %. Luego, el pico máximo de utilización de la CPU en el escenario 3 fue de 56 %.

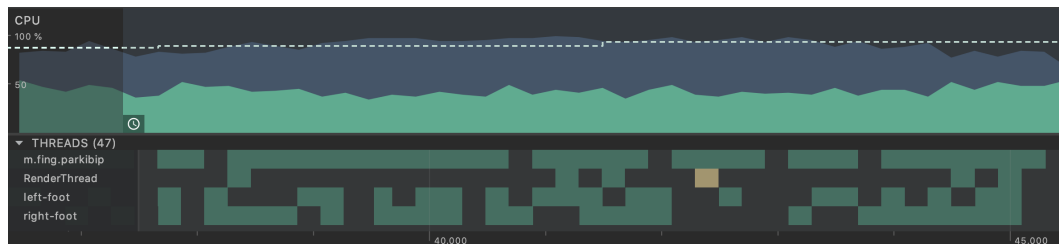


Figura 6.4: Porcentaje de utilización de la Unidad Central de Procesamiento (CPU) de PARKIBIP analizando datos de dos dispositivos a 400Hz -en verde-. Pico máximo de 56% de uso. En la parte inferior se puede ver la utilización de la CPU para los distintos hilos de ejecución. En gris se puede ver el uso de la CPU de otras aplicaciones del sistema. Esta gráfica se obtiene utilizando la herramienta *Profiler* de Android Studio.

Como se puede observar en la figura, los hilos “left-foot” y “right-foot”, dedicados al procesamiento de los datos para cada dispositivo, están quitando una carga considerable al hilo principal. Además, estos no están realizando un uso extensivo de la CPU, ya que realizan procesamiento en intervalos cortos y eficientes.

Adicionalmente, se puede utilizar esta herramienta para investigar el uso que se está realizando de cada hilo, es decir, cuáles son las instrucciones que se están ejecutando en cada hilo y cuáles son aquellas que están demandando más tiempo a este recurso. La figura 6.5 muestra las instrucciones que están siendo ejecutadas en un instante de tiempo determinado, durante una sesión de terapia activa.

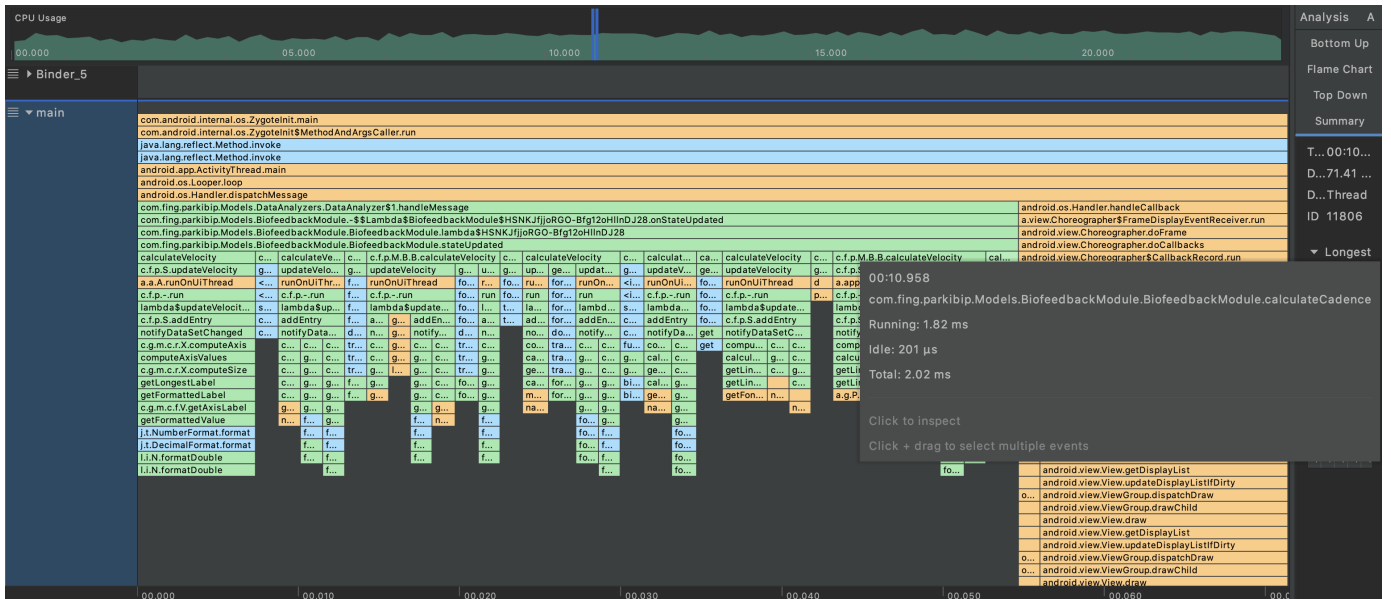


Figura 6.5: Ejemplo de instrucciones ejecutadas en un instante de tiempo dado por PARKIBIP en un hilo específico. En este caso estas instrucciones se ejecutan en el hilo principal. Es posible seleccionar una de estas instrucciones para ver más detalles. Esta gráfica se obtiene utilizando la herramienta *Profiler* de Android Studio.

La herramienta permite, esencialmente, indagar en los momentos específicos en los que un hilo está siendo sobrecargado. Así, detectar cuáles son las instrucciones que se están ejecutando y qué tiempo de procesamiento están necesitando para completarse.

Cabe destacar, que durante el análisis de uso de CPU, se detectó que en el escenario 3 la aplicación respondía de forma lenta a las interacciones del usuario. Esto se debía a que se estaba actualizando la interfaz gráfica de la pantalla de sesión activa de forma instantánea por cada dato recibido, resultando en un uso exhaustivo del hilo principal. Para corregir la problemática, se modificó la implementación para que las actualizaciones ocurran cada f datos, siendo f un número configurable. Si bien los parámetros se calculan elemento a elemento, las actualizaciones en pantalla se realizan cada f elementos. La selección de dicho número se realiza en función de la frecuencia de datos que provee el sensor, de forma que se mantenga una percepción de tiempo real sin saturar el hilo principal.

Memoria

Otro aspecto importante es, el uso de la memoria que realiza la aplicación. Es importante que la aplicación no esté haciendo un mal uso de este recurso, ya que puede derivar en una degradación de la aplicación e incluso de todo el *smartphone*. Para ésto, se monitorean las reservas de memoria (a.k.a. *allocs*) que realiza la aplicación en tiempo real, utilizando también la herramienta *Profiler*.

Los aspectos fundamentales a evaluar en cuanto al uso de la memoria son:

- Si la memoria que ya no se utiliza está siendo liberada adecuadamente
- Si no se está usando memoria de forma innecesaria

Para verificar el primer punto, se supervisa la memoria con *Profiler* durante la apertura de una pantalla en la aplicación. Se verifica que, una vez que se cierra dicha pantalla, la memoria ocupada al abrirla vuelve a liberarse. Esto se realiza para cada una de las pantallas o *Activities*. Luego, se estudió detalladamente en qué se está utilizando la memoria en cada momento, con el objetivo de evitar el uso innecesario de la misma. Se detectó que las listas de mediciones mantenidas en memoria estaban ocupando grandes cantidades de almacenamiento, por lo que se disminuyeron los intervalos de tiempo para realizar el guardado en la base de datos y de esta forma evitar mantener tantas mediciones en memoria. La figura Fig. 6.6 muestra la gráfica de uso de memoria durante el procesamiento en tiempo real. El pico máximo de memoria utilizada fue de 62MB.

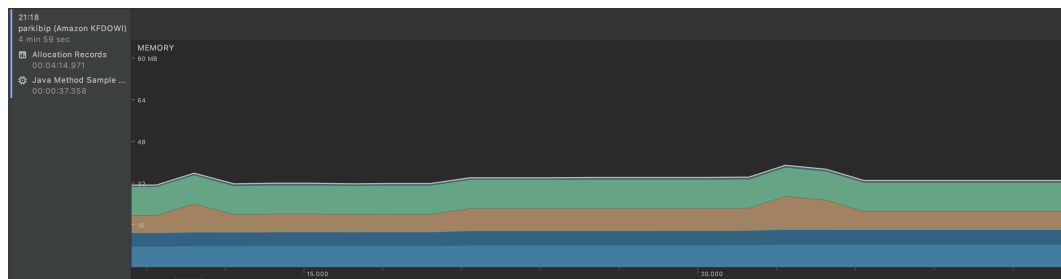


Figura 6.6: Ejemplo de uso de memoria utilizada por PARKIBIP durante el procesamiento de los datos. Esta gráfica se obtiene utilizando la herramienta *Profiler* de Android Studio.

Uso de Batería

Analizando el uso de la batería no se detectó ningún tipo de anomalía. El uso de la batería, es categorizado por *Profiler* como Liviano, Medio o Pesado. Para PARKIBIP, en el flujo con mayor procesamiento -sesión activa en curso-, el uso de la batería fue de nivel Medio, y en el resto de los flujos fue Liviano. El diagrama de la figura Fig. 6.7 expone el uso de la batería durante una sesión activa.

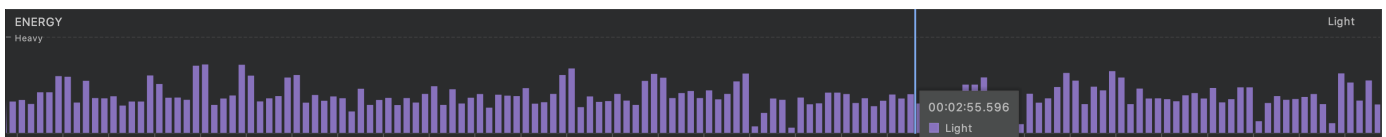


Figura 6.7: Porcentaje de uso de batería en tiempo real durante una sesión activa de PARKIBIP. En promedio el uso es de la batería es de nivel Medio en esta etapa.

6.6. Asociación Uruguaya de Parkinson

En virtud de las necesidades de recolección de datos espacio-temporales -relativa a los sujetos con la EP para su posterior análisis-, realizar un estudio observacional de los participantes en diferentes sesiones -para un análisis cualitativo- y evaluar el uso del sistema en tiempo real; se contactó a la ASOCIACIÓN URUGUAYA DE PARKINSON (AUP) con el fin de cubrir dichos requerimientos.

La AUP es una organización sin fines de lucro, presidida por la Sra. Ana Ma. Martinez, orientada a (i) contribuir en la investigación y la elaboración de material teórico y práctico sobre calidad de vida e intervención en grupos terapéuticos para personas con EP y sus familiares, (ii) contribuir al desarrollo de calidad de vida. En la misma, se realizan diversas actividades grupales con los sujetos afectados en reuniones semanales -presenciales y no presenciales-, tales como clases de yoga, canto, estiramiento, trabajos cognitivos y de memoria, terapias con voluntarios especializados (i.e. fisioterapeutas), etc.

Con la autorización adecuada de la presidente de la AUP y la posterior redacción de un consentimiento informado -anexo del presente informe, **Documento - Consentimiento informado al usuario de PARKIBIP-**, procedimiento mediante el cual se garantiza que el sujeto ha expresado voluntariamente su intención de participar en la investigación y comprendido la información que

se le ha dado acerca de los objetivos y los beneficios de la misma; se procedió a establecer sesiones guiadas de análisis y recopilación de datos.

6.7. Población y protocolo experimental

Quince voluntarios reclutados de la AUP con la enfermedad de Parkinson participaron en este estudio demostrando absoluto interés e ilusión en la investigación. El procedimiento para la extracción de las características de la marcha fue realizado bajo el acompañamiento de un Lic. en Fisioterapia, luego de que el protocolo y el consentimiento fueran aprobados.

Durante el mes de Diciembre del 2019, un total de 8 parkinsonianos masculinos y 7 femeninos emplearon el sistema PARKIBIP -15 participantes-, con una edad promedio de 74.5 años (mínima: 57, máxima: 85), marcha independiente y de distintas zonas barriales de Montevideo-Uruguay. Todos los sujetos realizaron una caminata lineal en un suelo plano -sin alteraciones- y con marcha normal hacia adelante. Los datos del acelerómetro, el giroscopio y el magnetómetro de los dispositivos IMU conectados, ubicados en los talones de los pies del participante, fueron registrados. La distancia recorrida por participante fue de 10 metros, medida a través de marcas en el piso, totalizando un total de 150 metros. La figura Fig. 6.8 expone una fotografía tomada durante una de las sesiones desarrolladas con voluntarios de AUP, además se adjunta una grabación de una sesión PARKIBIP ².

Asimismo, para cada parkinsoniano voluntario se efectuó un cuestionario, con el objetivo de registrar información relativa a la patología (i.e. tipo de parkinson), información personal (i.e. nombre completo, teléfono de contacto, continuidad en la participación), si asistían o asistieron al programa PREN-PAR -Programa de Educación y Rehabilitación en la Enfermedad de Parkinson para pacientes, familiares y cuidadores-, si frecuentaban el centro Hospital de Clínicas, entre otras.

²Las grabaciones circulares se encuentran disponibles en el sitio [PARKIBIP-grabaciones-circulares](#)



Figura 6.8: Foto tomada durante una de las sesiones desarrolladas con voluntarios de AUP. Diciembre, 2019.

6.8. Toma de datos sin biofeedback (AUP)

Con el fin de presentar los resultados adquiridos, el análisis se organiza según el orden cronológico de ciertas actividades:

1. En el mes de Diciembre de 2019, se evalúa la recolección de datos inerciales de enfermos de Parkinson, con el sistema PARKIBIP (i.e. integrantes de la AUP). Dichos datos fueron esenciales para trabajar durante el proyecto.
2. El proyecto PARKIBIP fue presentado y evaluado en funcionamiento por voluntarios en el congreso y pre-congreso latinoamericano de Ingeniería Biomédica SABI 2020 (por sus siglas, Sociedad Argentina de Bioingeniería), paper científico mediante -marzo del 2020-.
3. Luego, se propone evaluar el comportamiento de PARKIBIP con un sistema de referencia y obtener métricas comparativas.
4. Para finalizar, se analiza la marcha de sujetos sin afecciones, y se realizan reflexiones generales del uso de PARKIBIP.

Las pruebas iniciales consistieron en recuperar, consolidar y almacenar en

una base de datos local al dispositivo, las mediciones inerciales de parkinsonianos con PARKIBIP. Los propósitos de dichas pruebas fueron: (i) evaluar el funcionamiento de los sensores IMU en la aplicación, (ii) verificar el almacenamiento de datos inerciales asíncronos, (iii) validar el rendimiento de PARKIBIP (e.g. cuánto soporta la misma, en términos de fallas), (iv) por último y no menos importante, aproximarse y comprender la enfermedad. Tal como se menciona en [Población y protocolo experimental](#), para llevar a cabo los objetivos, se realizaron diversas sesiones de caminatas con integrantes de la asociación.

Primero y principal, la operación de PARKIBIP fue exitosa y todos los objetivos fueron cumplidos satisfactoriamente. El sistema, soporto adecuadamente 15 sesiones con sujetos diferentes, recopilando y procesando un total de 29.705 muestras de las extremidades inferiores -pies-. No se encontraron anomalías en las pruebas.

A continuación, se analizan algunas consecuencias logradas y se ejemplifica para la marcha de un sujeto particular. La persona se encuentra inicialmente en reposo, camina 10 metros y vuelve al reposo.

La figura Fig. 6.9 muestra el comportamiento de la aceleración del dispositivo en sus tres componentes, a_X -aceleración en x-, a_Y , -aceleración en y- a_Z -aceleración en z-, en las unidades gravitacionales (g) en el sistema internacional. Primero, el acelerómetro mide tanto la aceleración del usuario como la gravitatoria, la componente a_Y dada por la línea negra resalta la afectación de las fuerzas gravitatorias -iniciando aproximadamente en $-1g$ - y así, ese eje se encuentra proyectado en $-1g$. Segundo, se puede apreciar claramente un patrón de comportamiento repetitivo para la marcha, inicia con el pie estacionario -todas las aceleraciones aproximadamente en 0-, presenta un pequeño pico de aceleración vinculado a la fase *midswing* del ciclo, luego, otro pico mas acentuado desde el *midswing* al *HS* (i.e. recordando, heel strike), finalizando con el pie estacionario nuevamente -stance-. Asimismo, es claro un bloqueo de la marcha (i.e. FoG) entre las medidas 610-730, luego dos pasos, y nuevamente un bloqueo para las muestras 840-960.

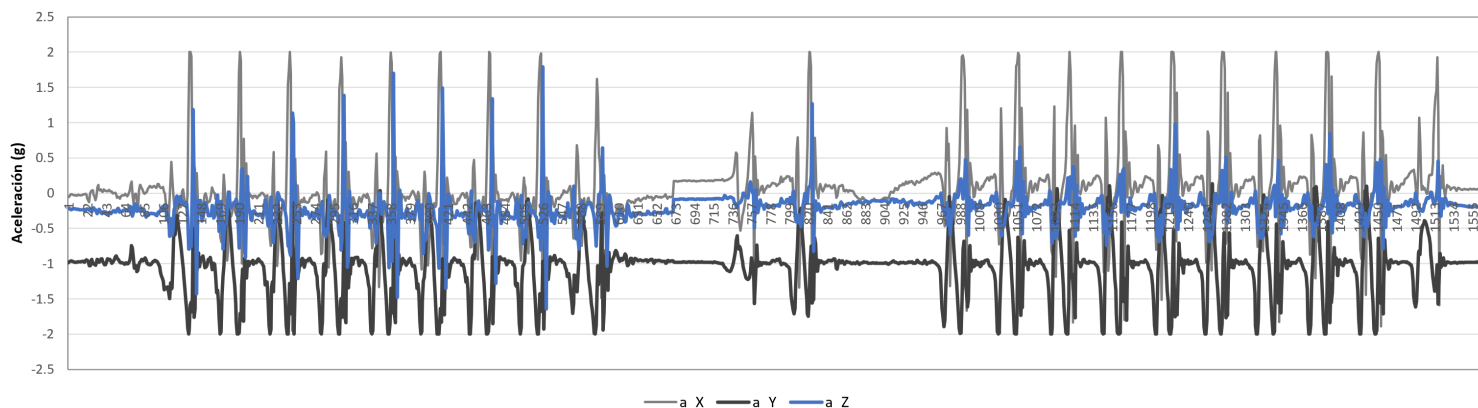


Figura 6.9: Aceleración tridimensional de un parkinsoniano en el marco del sensor. Representadas por las componentes a_X -aceleración en x-, a_Y -aceleración en y- a_Z -aceleración en z-; en las unidades gravitacionales en el sistema internacional.

Por otro lado, la figura Fig. 6.10 muestra el resultado de dos métodos que computan la aceleración lineal del usuario en un instante dado, fruto de la rotación de las componentes de la aceleración desde el marco Sensor al marco Tierra -obtenidas del IMU, ver 6.9-, la extracción de las fuerzas gravitatorias, y luego el cálculo de la norma del vector aceleración resultante. El primer procedimiento, línea azul de la gráfica, emplea el filtro de Orientación -útil en la rotación- únicamente con los datos del acelerómetro y del giroscopio. El segundo método, dado por la línea gris de la gráfica, integra además, el sensor magnetómetro. Así, se observan las consecuencias:

- Al integrar datos de fuerzas magnéticas, la calibración de la orientación del sensor, previa al procesamiento, es instantánea. Evaluando la gráfica, se observa que el método que no emplea magnetómetro tarda aproximadamente 100 observaciones en estabilizarse a cero. Se puede concluir que, integrar datos de fuerzas magnéticas resulta en un cálculo más eficiente de orientación.
- Asimismo, la gráfica expone un patrón de comportamiento repetitivo, en donde el pico más alto representa un paso del sujeto y los momentos cercanos a cero, momentos estacionarios.

A su vez, fueron evaluados dos aspectos fundamentales de la marcha del sujeto. Por un lado el ciclo conformado por sus fases Stance/Swing, por otro lado la velocidad instantánea adquirida en m/s . La figura Fig. 6.11, superpone ambos resultados en dicha gráfica. Primero, la línea superior bordo indica los estados

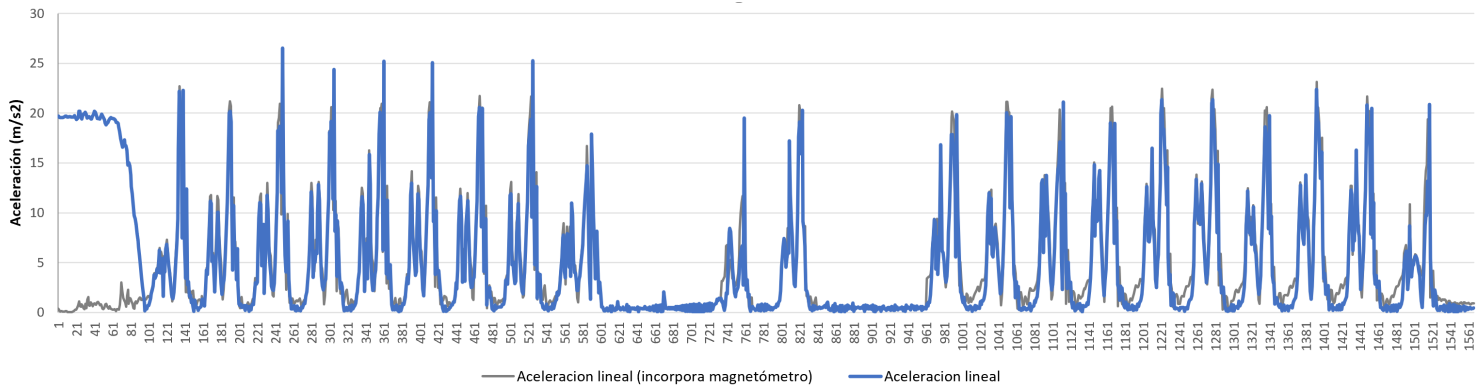


Figura 6.10: Aceleración lineal de un parkinsoniano en el marco de referencia Tierra. Representada en las unidad en el sistema internacional m/s .

estacionarios o de velocidad cero -detectados con el método MV de ZVD-, mientras que la línea inferior del mismo color, la transición al movimiento. Así, se detectan las fases HS, como la transición de 0 a 0.5 -en la gráfica-, y la fase TO, como la transición inversa. Segundo, dada por la línea azul, se expone la velocidad del usuario, en la unidad del sistema internacional m/s . Se aprecia que ambos resultados son consistentes, ya que cuando la velocidad es cero, el pie del usuario se encuentra estacionario. De manera complementaria, se reconocen ciertos falsos positivos, previos al valor 581. Esto, se debe a la fecha temprana de las pruebas, y que no fueron recopilados datos del magnetómetro en esta etapa -útil para la eficiencia del filtro de orientación-.

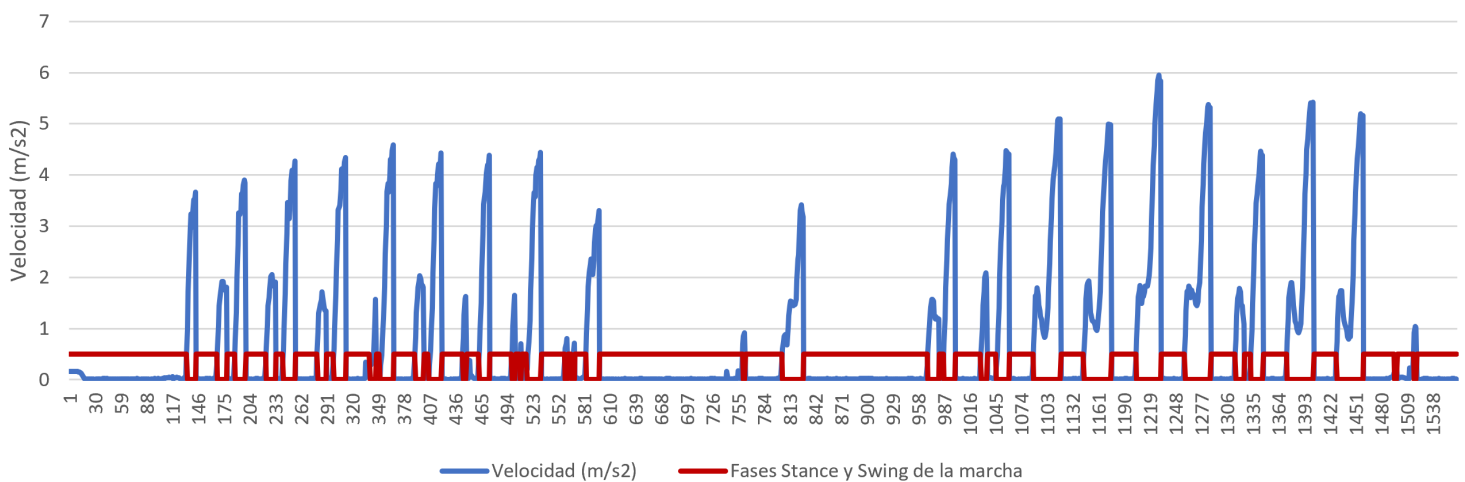


Figura 6.11: Velocidad del usuario en m/s^2 y detecciones de velocidad cero. La línea bordó indica las transiciones de desde el estado estacionario -nivel superior- al estado en movimiento -nivel inferior-.

Finalmente, se obtuvo una velocidad media de 0.73 m/s para dicho sujeto.

Hay que mencionar además que, se lograron mejores resultados en la rotación del vector tridimensional en el marco de referencia del Sensor, hacia el marco de coordenada inercial Tierra, con el factor de convergencia β igual a 5. En PARKIBIP y con el IMU Metawear, factores de β bajos conllevaban a errores incrementales en el cómputo del *quaternion*. Este error, luego impactaba negativamente en los parámetros espacio-temporales.

6.9. Pruebas de PARKIBIP con biofeedback

Seguido a las pruebas con la AUP, PARKIBIP fue aceptado e invitado a exponer como orador en el congreso SABI 2020, mediante la elaboración de un **paper de investigación**. El mismo, se puede hallar en el apéndice **Congreso de Ingeniería Biomédica SABI 2020**. Durante el transcurso del congreso, PARKIBIP fue evaluado en operativo mediante voluntarios vinculados a la ingeniería biomédica. Tres participantes emplearon PARKIBIP por medio de marchas lineales en una superficie plana y un estímulo de vibración configurado en cada pisada (un ingeniero biomédico de origen peruano, un médico argentino y un profesor de educación física uruguayo). El sistema actuó correctamente, estimulando al usuario con cada pisada. A su vez, el feedback recibido por los participantes fue positivo y alentador.

6.10. Evaluación de PARKIBIP con sistema de referencia

Conforme a evaluar el sistema frente a algún sistema de referencia, fue propuesto el laboratorio de marcha y su sistema VICON 3D del Hospital de Clínicas. Sin embargo, se presentaron ciertas limitantes del contexto: (i) pandemia de Covid-19 de por medio, (ii) complejidad en el acceso al laboratorio especializado, (iii) dificultad en la obtención de pacientes habilitados, (iv) en general, los pacientes son mayores, y pertenecen a la población de riesgo. Así, PARKIBIP fue evaluado usando datos inerciales *Open Source* en el lenguaje de desarrollo Matlab, propiedad de *OpenShoe* [OpenShoe \(2020\)](#).

Se procesaron 12.239 observaciones, tomadas del IMU MicroStrain 3DX-GX2 con una frecuencia de 250Hz. El IMU fue montado en la suela del zapato

del lado derecho del usuario, y el mismo camino simulando una trayectoria de circuito cerrado. Se tienen como valores esperados, la posición inicial igual a la final, la velocidad media de 1.94 m/s.

Se efectuaron las siguientes comparaciones con datos de *OpenShoe*:

- Comportamiento entre los métodos de ZVD PARKIBIP (i.e. MV versus GLRT)
- Velocidad media PARKIBIP versus velocidad media Openshoe. Cálculo del error cuadrático medio (en inglés, root-mean-square deviation, RMSD)
- Trayectoria PARKIBIP versus trayectoria Openshoe

La prueba inicial consintió en correr los algoritmos de detección de velocidad cero ZVD-, a través de los dos métodos implementados -MV y GLRT- con los datos ya mencionados. Las figuras Fig. 6.12 y 6.13, muestran la aceleración lineal del usuario en la unidad gravitacional junto a las detecciones de cero velocidad para ambos métodos. Como se puede observar la primer figura empleando el método MV -Fig. 6.12-, presenta un mayor número de detección o transiciones entre estados, que explican varios falsos positivos. Sin embargo, la segunda figura empleando el método GLRT -6.13- fue mucho mas precisa con las detecciones. **Como resultado, se puede decir que GLRT es mas preciso que MV, ya que contempla las variaciones en la velocidad angular.**

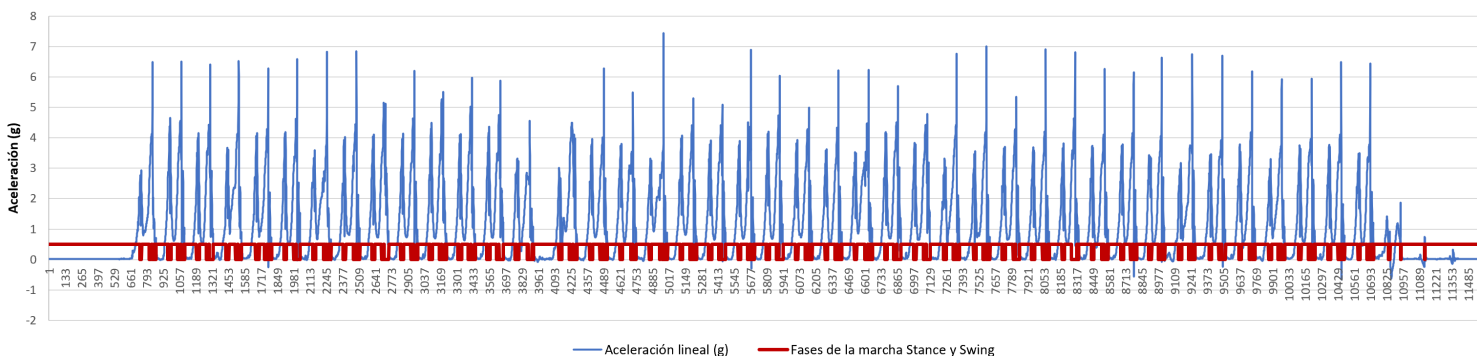


Figura 6.12: Aceleración lineal del usuario en m/s^2 y detecciones de velocidad cero con el método MV. La línea bordo indica las transiciones de desde el estado estacionario -nivel superior- al estado en movimiento -nivel inferior-.

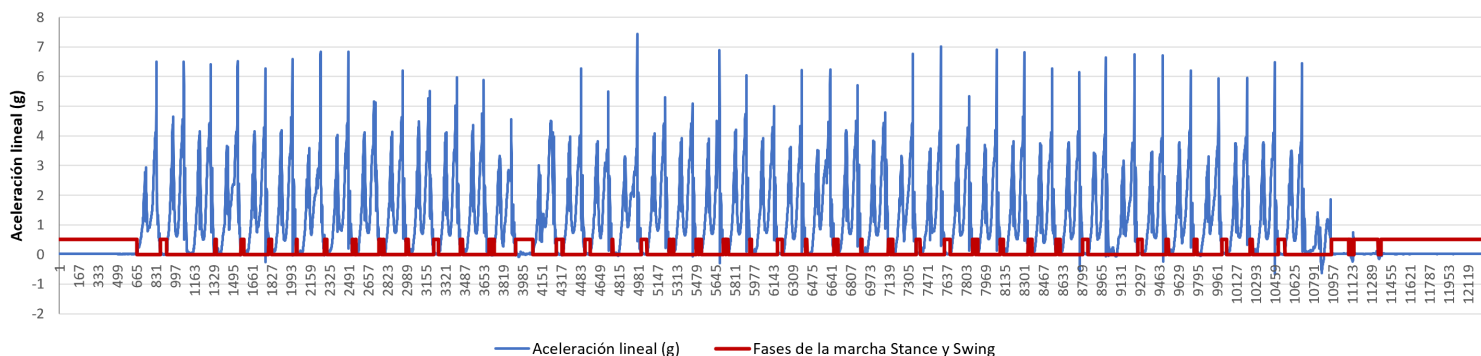


Figura 6.13: Aceleración lineal del usuario en m/s^2 y detecciones de velocidad cero con el método GLRT. La línea bordo indica las transiciones de desde el estado estacionario -nivel superior- al estado en movimiento -nivel inferior-.

Además, fueron comparados los parámetros espacio-temporales velocidad media y trayectoria frente a los valores esperados por Openshoe. La imagen Fig. 6.14, exhibe la velocidad del usuario resultado del procesamiento del filtro de Kalman PARKIBIP, en m/s . El valor esperado, según la marcha del usuario, es de $1.94 m/s$. Así, **la velocidad media lograda por PARKIBIP fue de $1.81m/s$, indicando una diferencia de $0.13 m/s$** . En cambio Openshoe alcanzo un valor de $1.75m/s$. Como consecuencia del análisis, **el error cuadrático medio (RSME) entre las velocidades instantáneas de PARKIBIP y Openshoe fue de 0.0214**.

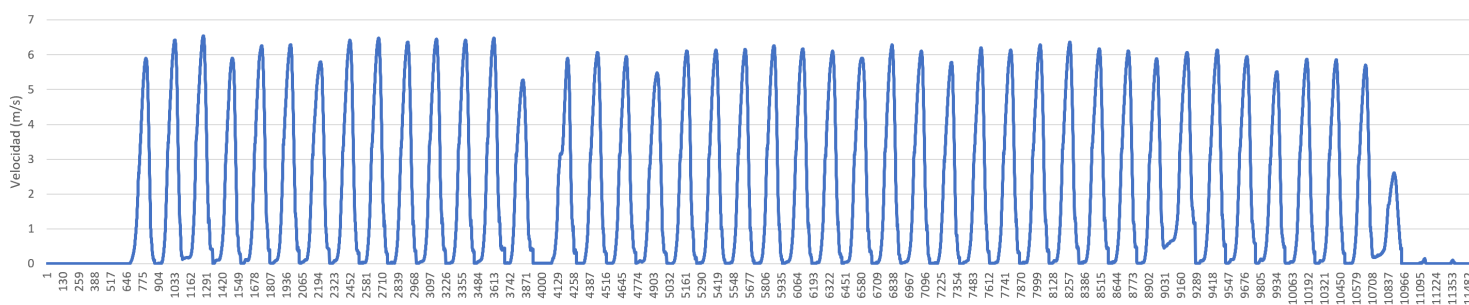


Figura 6.14: Velocidad del usuario calculada por el filtro de Kalman PARKIBIP, en m/s . Realizado con datos de prueba open source de Openshoe. La velocidad media lograda por PARKIBIP fue de $1.81m/s$ versus el valor esperado $1.94 m/s$.

Finalizando las comparaciones con los datos públicos de Openshoe, se presenta una gráfica de la trayectoria en las dimensiones x e y, en la unidad metros. La figura Fig. 6.15, dibuja la la trayectoria recorrida por el sujeto en deambulación, iniciando en el posición $(x, y) = (0, 0)$ dado por el punto naranja, y culminando en el punto $(x, y) = (-0.048, 0.103)$. Lo que indica la cercanía

respecto al valor esperado.

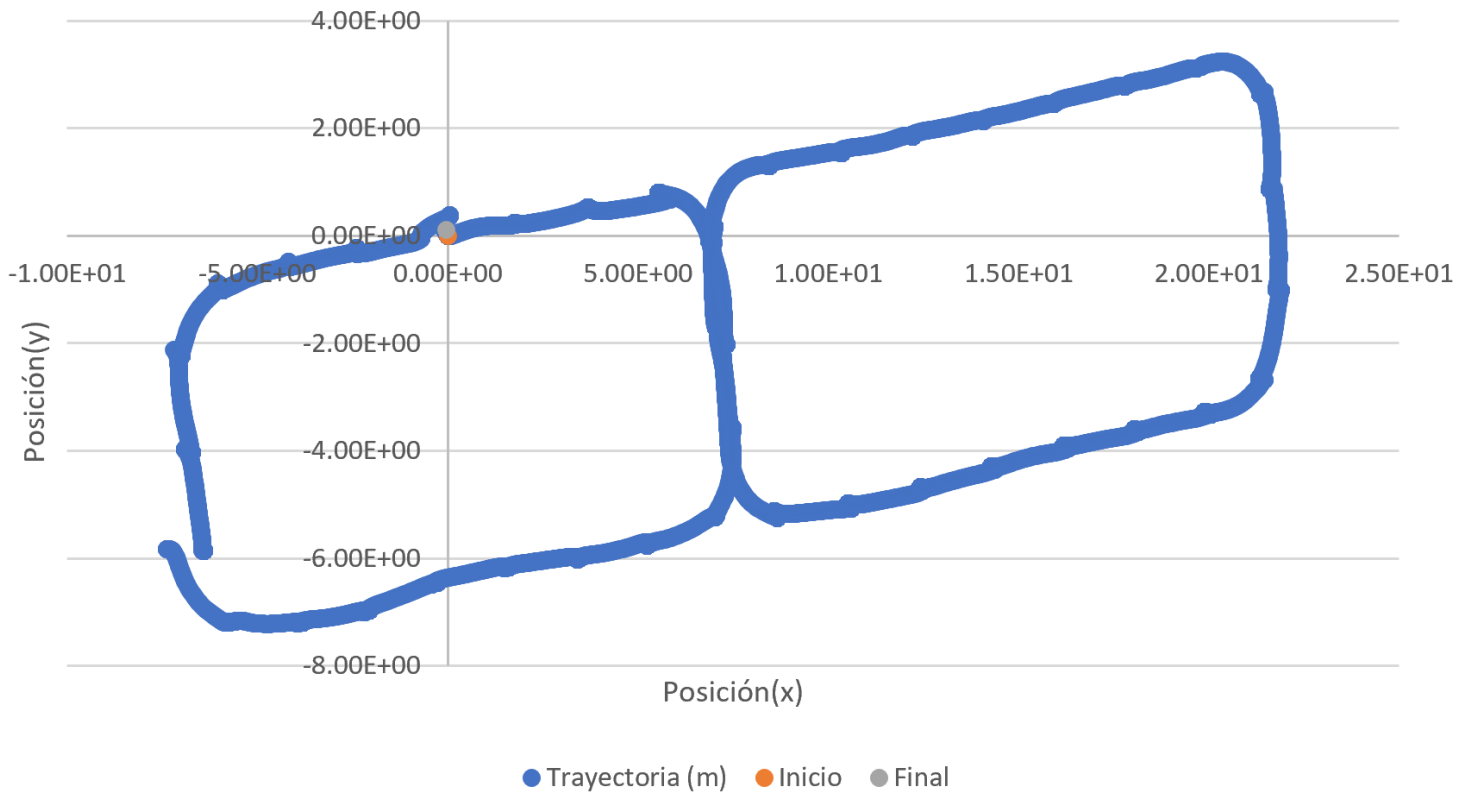


Figura 6.15: Gráfico de la trayectoria en las dimensiones x e y, en el sistema internacional de unidades -metros-. El sujeto inicia la marcha en la posición $(x, y) = (0, 0)$ dado por el punto naranja, y culminando en el punto $(x, y) = (-0.048, 0.103)$. Es decir, un ciclo cerrado.

6.11. Pruebas de PARKIBIP con biofeedback y referencias espacio-temporales

En líneas generales, se realizan ciertas reflexiones respecto a la marcha de sujetos sin afecciones en la misma. Los participantes evaluados mediante pruebas experimentales fueron los mismos creadores del sistema y allegados; con una edad promedio de 28 años. Así, sobre la finalización del proyecto, se llevaron adelante 20 marchas aleatorias sobre una superficie plana que fueron registradas en una planilla Excel. La configuración establecida fue estimular al usuario de forma vibratoria y sonora según la regla seleccionada, para mejorar la calidad de la validación.

Primero, para cada marcha, se analizó la aceleración lineal instantánea y media del usuario -rotada al marco terrestre y extraída la gravedad-. Todas las aceleraciones registradas presentaron un patrón de comportamiento adecuado, siendo acotadas superiormente por una aceleración instantánea de 25 m/s^2 y con un movimiento cíclico compuesto por los períodos *Stance* y *Swing*. La aceleración promedio, contempladas todas las marchas, ascendió a 2.88 m/s^2 , dentro de lo esperado.

En términos de estímulos, seleccionado el evento PARKIBIP *onHeelStrike* -toque de talón-, para identificar el comienzo de la fase de *Stance*, fueron contabilizados la cantidad de estímulos correspondientes a las pisadas. En general, los tres métodos detectores de velocidad cero presentaron resultados satisfactorios en la identificación de las fases de a marcha, con las siguientes particularidades:

- *Threshold*: En caso detectar momentos estacionarios, lo hace una única vez, mediante picos de aceleración. Si bien es un beneficio, ya que detecta adecuadamente el cambio de fase de la marcha, por otro lado no continúa el monitoreo del estado actual. Es decir, registra transiciones entre estados. Asimismo, el método es predispuesto a perder fases si el umbral es alto o a detectar mayores transiciones si el mismo es bajo.
- *MV*: Presenta excelentes resultados en la detección de momentos estacionarios, mediante la supervisión continua del estado activo. Si bien el método desarrollado no falla en la estimación del estado *stance*, en ocasiones presenta falsos-positivos, detectando un estado estacionario cuando en realidad estaba en movimiento. Por ende, emite unos pocos eventos incorrectos.
- *GLRT*: Similar a *MV*, al incorporar el sensor giroscopio, GLRT ajusta las detecciones de los estados *stance/swing*, logrando mejores resultados. Al ser preciso, mínimos movimientos angulares del tobillo del paciente generan transiciones de estados.

Luego, bajo el soporte de la funcionalidad que registra la duración de la sesión, la Cadencia (i.e pasos por minutos) y la duración media del paso fueron calculadas apropiadamente, según el método y los eventuales falsos-positivos. Otra prueba realizada con éxito fue el evento PARKIBIP *onFoG*. Esto es, transcurrido un intervalo de tres segundos en el que el pie se encontraba en una fase

estacionaria, se emitía un estímulo vibratorio/sonoro. En todas las ocasiones, el sistema actuó apropiadamente.

Además, se analizaron las velocidades instantáneas y medias de las marchas. Puesto que PARKIBIP no fue evaluado frente a otro sistema en tiempo real con el mismo sujeto, el parámetro espacio-temporal velocidad no fue validado con exactitud. Sin embargo, el sistema presentó valores de velocidad media acotados entre $[0.57, 1.92]$ m/s , siendo consistente con la velocidad de una marcha normal. A su vez, se probaron las reglas para estimular *onInstantVelocityThreshold* y *onAverageVelocityThreshold*. Ambas, cumplieron el objetivo de estimular al sujeto superado cierto umbral predefinido.

Para finalizar, cabe señalar que, aunque el parámetro espacio-temporal posición instantánea fue obtenido, el método que calcula la trayectoria no fue desarrollado. Esto se debió a la priorización de tareas, alcance del proyecto y tiempo disponible.

Capítulo 7

Gestión del proyecto

A lo largo del proyecto se establecieron procesos y herramientas que permiten el desarrollo del mismo con efectividad. Desde la gestión y planificación de tareas hasta herramientas de desarrollo, versionado de código, gestión de la documentación, almacenamiento de archivos compartidos, diagramas, etc. En este capítulo se detallan todas las herramientas y procesos de trabajo utilizados; las actividades del proyecto organizadas en sus distintas etapas; y por último, un reporte de los costos que implicó el desarrollo del proyecto.

7.1. Actividades y cronología

Durante el transcurso del proyecto, se realizaron diversas actividades para alcanzar los resultados obtenidos. Estas actividades se organizaron en etapas, las cuales se llevaron adelante con un orden específico planificado con anterioridad. En esta sección se detallan estas actividades, la duración de las mismas y cómo fueron organizadas.

Las actividades principales llevadas a cabo durante el proyecto fueron:

- Relevamiento inicial: Comprender la problemática, conocer el equipo de trabajo, definir el objetivo del proyecto y el alcance a grandes rasgos.
- Estado del arte: Investigación sobre los avances en el área de GAIT ANALYSIS (GA). Revisión sistemática de la literatura (SLR).
- Compra de dispositivos IMU y coordinación de envío desde E.E.U.U.
- Estudio de tecnología Android.

- PoC: Prueba de conceptos para mitigación de riesgos técnicos.
- Desarrollo de la aplicación:
 - Análisis: Alcance, especificación detallada y casos de uso
 - Diseño: Diagrama de clases. Interfaz y experiencia de usuario.
 - Implementación de la aplicación Android.
 - Revisión de código entre pares.
 - Pruebas del sistema: exploratorias, funcionales y no funcionales.
- SABI 2020: Congreso de bioingeniería. Marzo 2020, Piriápolis.
 - Desarrollo de un artículo científico para REVISTA ARGENTINA DE BIOINGENIERÍA.
 - Presentación y exposición de PARKIBIP en congreso (Fig. 7.1)
- Análisis de resultados
- Gestión del proyecto
 - Planificación
 - Gestión de tareas
 - Reuniones con tutores
 - Reuniones con fisioterapeutas
 - Coordinación para visita a sesiones de fisioterapia
 - Gestión con Asociación Uruguaya de Parkinson para pruebas con integrantes voluntarios
- Documentación del proyecto



Figura 7.1: Exposición de PARKIBIP en Congreso de Bioingeniería y Jornadas de Ingeniería Clínica SABI 2020. De izquierda a derecha: Carlos Huerta y Samuel Sainz. Marzo 2020. Argentino Hotel, Piriápolis, Uruguay.

El proyecto comenzó en Setiembre de 2018 y se finalizó en Diciembre de 2020. En el diagrama de Gantt de la figura Fig. 7.2 se muestra el orden cronológico en el que se realizaron las actividades. Este diagrama resulta útil para entender la duración de las tareas en comparación con la duración del proyecto final; así como para poder comparar la duración de una tarea con respecto a la otra. Además, permite ver visualmente como se organizaron las actividades del proyecto de principio a fin.

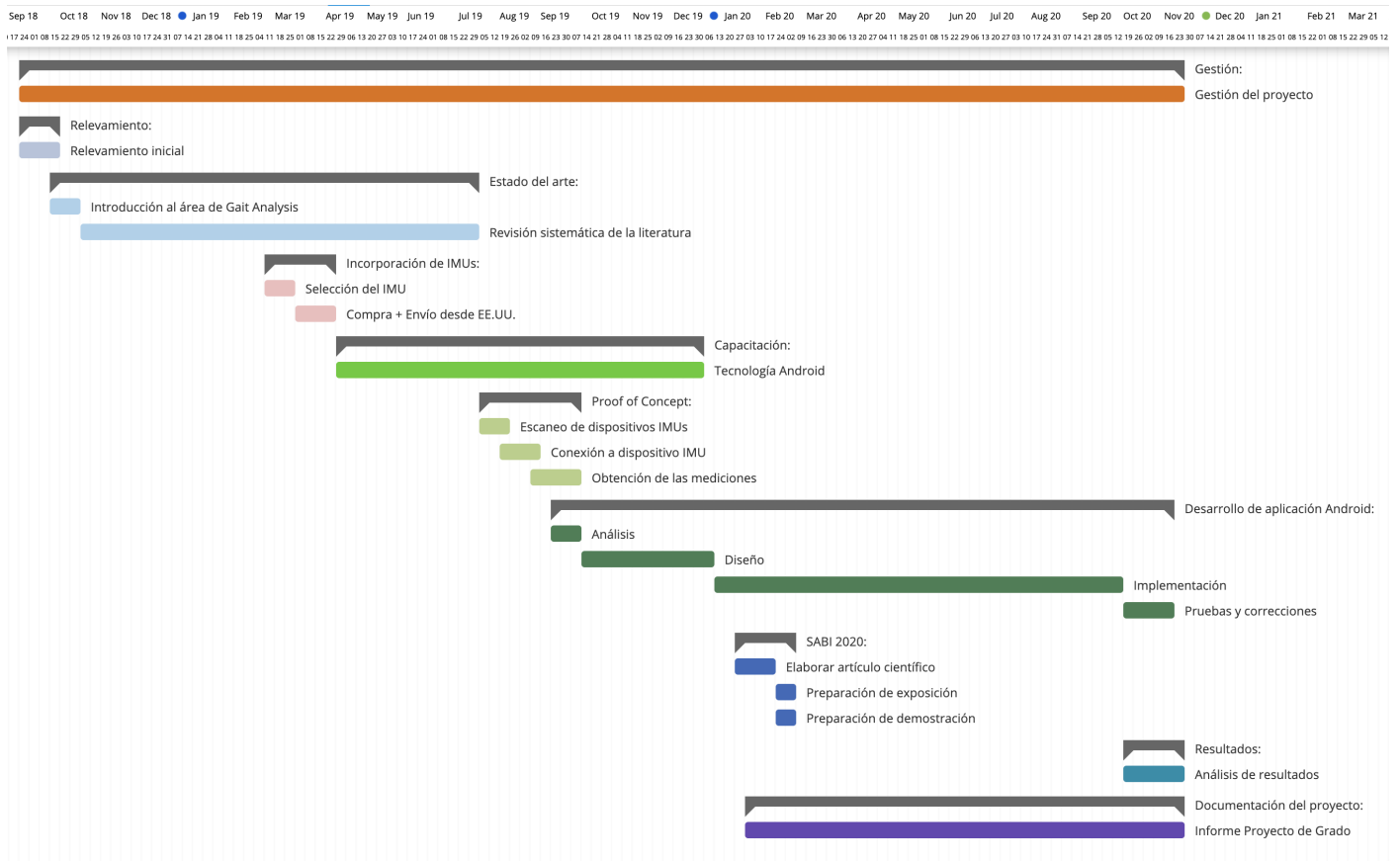


Figura 7.2: Diagrama de Gantt. Expone las actividades realizadas durante el proyecto PARKIBIP desde el inicio del proyecto en Setiembre de 2018 hasta el final del proyecto en Noviembre de 2020.

7.2. Reporte de costos

Con el objetivo de determinar los costos del proyecto, en una primera instancia se deben evaluar las inversiones de horas-persona. Las horas-personas son la unidad de medida que se emplea en gestión de proyectos para medir los esfuerzos necesarios para completar una tarea. Una hora-hombre equivale al trabajo completado en una hora de esfuerzo ininterrumpido por un trabajador.

La tabla Tab. 7.1 resume la inversión de horas en el proyecto PARKIBIP mediante los campos: (i) costo hora-persona de cada uno de los integrantes -usando la media de la industria-, (ii) el total de horas invertidas en el proyecto y (iii) el costo total para cada persona del equipo. Considerando todos los elementos, en total se invirtieron 2.334 horas calificadas del equipo multidisciplinario, con un costo total que asciende a los USD 111.195.

Tabla 7.1: Resumen de horas dedicadas por participante del proyecto con su respectivo costo, teniendo en cuenta el rol de la persona.

Persona	Rol	Horas dedicadas	Costo hora-persona (USD)	Costo total (USD)
Est. Ing. en Computación 1	Ingeniero	1.084	45	48.780
Est. Ing. en Computación 2	Ingeniero	1.063	45	47.835
Est. de Fisioterapia	Fisioterapeuta	50	45	2.250
Tutor de Proyecto de Grado	PRODUCT OWNER	137	90	12.330

Se analizó, además, la evolución de la dedicación horaria a lo largo del proyecto a partir de las actividades detalladas en la sección **Actividades y cronología**. La figura Fig. 7.3 expone una serie temporal en meses con dicha evolución en horas-persona. Estudiando el gráfico, se puede apreciar: (i) una línea de tendencia ascendente desde el inicio del proyecto hacia su finalización, (ii) momentos de bajo rendimiento, en general, para los meses enero e inicio de febrero (e.g. debido a ocio, parciales o exámenes), (iii) picos máximos en los meses de marzo y noviembre de 2020 -correspondiente a SABI 2020 y a la finalización del proyecto-.

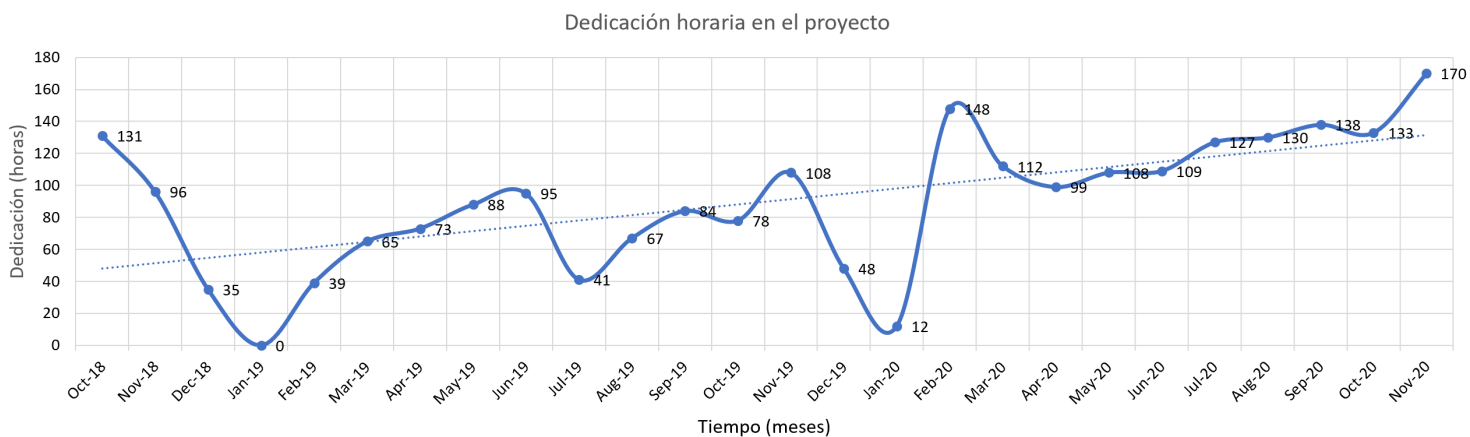


Figura 7.3: Evolución en meses de la dedicación horaria en el proyecto.

Por otro lado, para determinar de qué forma se invirtió el esfuerzo en las distintas actividades del proyecto, fueron establecidos seis centros de costo correspondientes a las principales actividades. Así, con el propósito de facilitar la comprensión de los diferentes gastos requeridos por PARKIBIP, se propone la figura Fig. 7.4, la cual presenta una serie temporal en meses con la distribución horaria por centro de costo.

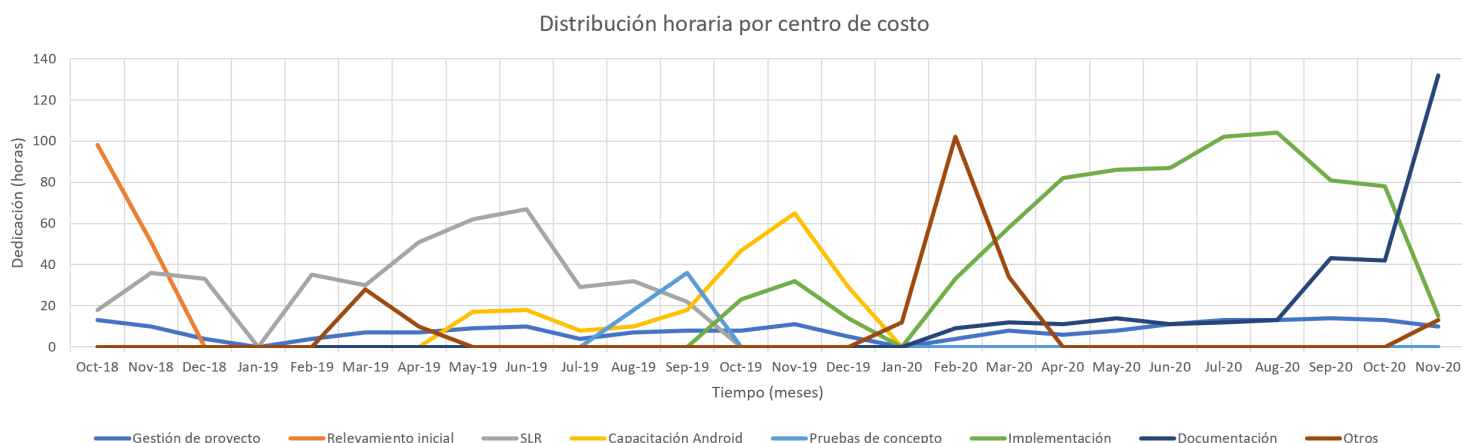


Figura 7.4: Serie temporal en meses con la distribución horaria por centro de costo.

La tabla Tab. 7.2 resume la información de la dedicación horaria por centro de costo, mostrando: (i) la cantidad de horas para cada centro de costos, (ii) el porcentaje del total de horas que fue dedicado a cada actividad.

Tabla 7.2: Resumen de horas dedicadas por actividad del proyecto, correspondientes a los distintos centros de costo.

Centro de costo	Horas dedicadas	Porcentaje sobre total de horas
Gestión de proyecto	207	8.88 %
Relevamiento inicial	149	6.39 %
Revisión Sistemática (SLR)	415	17.81 %
Capacitación Android	212	9.10 %
Pruebas de concepto	54	2.32 %
Implementación	795	34.12 %
Documentación	299	12.83 %
Otros (actividades menores)	199	8.54 %

Por otra parte, en adición al costo calculado según la dedicación del equipo, se deben contemplar también otras expensas como:

- Inscripción a congreso SABI 2020: 90 USD (45 USD por cada estudiante)
- Viáticos de viaje a congreso: 360 USD (180 USD por estudiante)
- Dispositivos IMU: 190 USD (95 USD cada uno)
- Bandas para fijación de IMU: 10 USD
- Envío de los IMU desde EE.UU: 35 USD

Estos gastos adicionales suman un monto de USD 685. Por lo tanto, en base a la estimación de horas y costos realizadas, **el proyecto presenta un costo total aproximado de USD 111.880.**

7.3. Gestión de tareas

La organización del BACKLOG de tareas es importante para la gestión del proyecto. Trello (© 2020 Trello, Help Scout) es una herramienta simple y gratuita para administrar las actividades de un proyecto. Se basa principalmente en un tablero de tarjetas, en el que cada tarjeta representa una tarea a ser realizada en el proyecto. La herramienta permite crear listas -en forma de columnas-, en donde las tarjetas pueden ser agregadas, y así organizar el trabajo según cualidades definidas por los administradores del proyecto. Trello, tiene como principal ventaja la capacidad de brindar fácilmente una visión sobre el estado del proyecto: el trabajo que fue realizado, el que está en curso y el que se encuentra pendiente; así como las personas involucradas en el desarrollo de cada tarea, entre otras propiedades.

Conforme a lo mencionado, el tablero de PARKIBIP fue dividido en 7 listas ordenadas de tareas:

- Tareas relacionadas a la gestión del proyecto
- Tareas referidas al desarrollo del informe
- Tareas de implementación del sistema de software
- Tareas en curso
- Pendientes de evaluación de calidad
- Tareas finalizadas

Asimismo, Trello permite agregar etiquetas a las tarjetas -de diversos colores-, así distinguirlas fácilmente dentro del tablero. En PARKIBIP, las etiquetas se emplearon para diferenciar el tipo de tarea: planificación, análisis, diseño, implementación, testing o error de sistema (bug).

Con el fin de estandarizar la gestión del proyecto, se estableció la información que debe ser agregada en la creación de cada tarjeta o actividad:

- Nombre y descripción de la tarea
- Persona asignada a la tarea -condición obligatoria para la ejecución de actividad-
- Etiquetas que establezcan el tipo de tarea
- Utilizar la funcionalidad de “checklist” si la tarea lo amerita
- En caso que corresponda agregar información extra sobre el resultado, emplear comentarios

La figura Fig. 7.5 resume la metodología empleada para la gestión del trabajo.

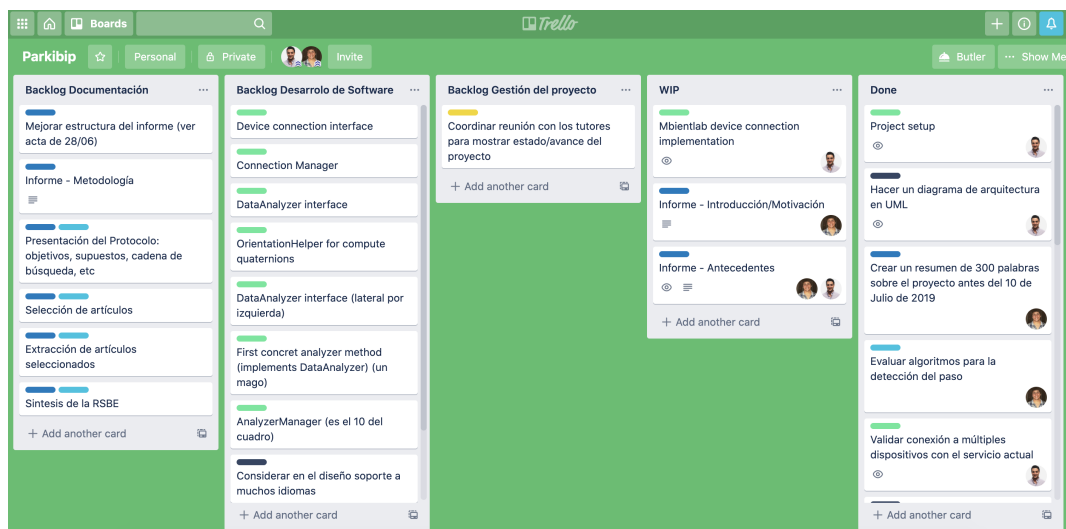


Figura 7.5: Herramienta de gestión de proyecto.

7.4. Administración bibliográfica

Resulta fundamental para analizar la literatura recopilada en el área de estudio y para desarrollar el estado del arte del proyecto, emplear una herramienta que brinde facilidades para la administración de las referencias bibliográficas. El gestor clásico Mendeley (© 2009-2013 Mendeley Ltd., Elsevier), fue la herramienta elegida por el equipo de trabajo para dicho fin. Toda la bibliografía seleccionada como relevante se agregó a la plataforma en un proyecto compartido por los investigadores. En una SLR, un gestor bibliográfico es de gran utilidad para administrar el inmenso volumen de estudios científicos. Así,

la herramienta Mendeley fue ampliamente utilizada en las etapas de selección y extracción de artículos de la SLR. A continuación, se destacan las principales funcionalidades de la herramienta utilizadas en el proyecto:

- Creación de un proyecto compartido para el almacenamiento de los artículos en la nube
- Centralizar la evidencia recopilada para una mejor trazabilidad y organización
- Extensión del explorador de internet para la importación de artículos al proyecto
- Generación de citas de los artículos en formato “bibtex” utilizado en este documento
- Subrayar, efectuar notas y comentarios con el fin de destacar sentencias dentro del artículo -visibles para el resto de los integrantes del proyecto-

La ilustración Fig. 7.6 permite visualizar la herramienta aplicada a el proyecto PARKIBIP, su organización configurable (i.e. datos básicos: autores, título, año de publicación, fuente) y algunas funcionalidades. Se administraron y analizaron aproximadamente 60 estudios en la librería -no se contabilizan los artículos indagados en otra herramienta-.

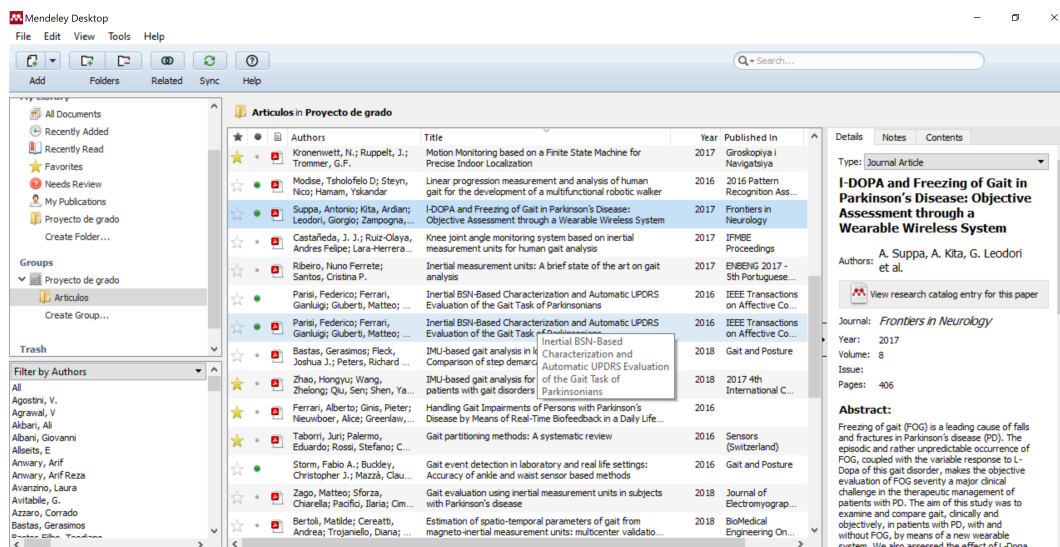


Figura 7.6: Gestor de referencias bibliográficas de PARKIBIP.

7.5. Gestión de archivos

En el transcurso del desarrollo del proyecto se generan y utilizan una gran variedad de archivos, tales como notas, actas de reuniones, documentos, planillas de cálculo, cuadros comparativos, imágenes, formularios, presentaciones, entre otros. Para gestionar estos archivos se utilizó la herramienta Google Drive (© Google LLC). Google Drive es un servicio de alojamiento de archivos en la nube que permite no solo almacenar documentos sino que también crear éste tipo de documentos (p. ej. documentos de texto, planillas de cálculo y presentaciones); para luego trabajar de manera colaborativa y sincronizada, es decir, con varias personas trabajando en un mismo documento y al mismo tiempo. Las unidades de almacenamiento pueden ser compartidas entre diferentes usuarios.

7.6. Entorno de desarrollo

Android Studio fue el IDE (del inglés “Integrated Development Environment”) que se utilizó para desarrollar la aplicación PARKIBIP. Es el IDE oficial por excelencia para el desarrollo de aplicaciones Android, basado en IntelliJ IDEA (© 2000-2020 JetBrains). Android Studio -desarrollado sobre el editor de código y herramientas de desarrollo de IntelliJ-, ofrece todas las funcionalidades necesarias para construir aplicaciones Android, tales como:

- Sistema de emuladores de dispositivos
- “Build system” flexible basado en Gradle
- Un entorno unificado para el desarrollo de aplicaciones para cualquier dispositivo
- Integración con herramientas de versionado de código
- Plantillas de código útiles para desarrollar patrones comunes
- Herramientas de Testing, etc.

7.7. Versionado de código

Un sistema de control de versiones (a.k.a. VCS, “Version Control System”), permite realizar un seguimiento de los cambios interactivos en la base de código [Blischak et al. \(2016\)](#). De esta forma, se puede experimentar nuevas ideas con la opción de revertir esos cambios hacia una versión anterior. A través de un VCS se pueden adjuntar mensajes que describen cada una de las versiones, así como trabajar en distintas versiones de forma paralela, por lo que facilita la colaboración. Además, un colaborador puede aplicar ciertos cambios al código y otro puede rápidamente incorporar esos cambios. El VCS más popular actualmente es Git [Blischak et al. \(2016\)](#) y fue el elegido para el versionado de código en PARKIBIP. Git permite crear fácilmente ramas -versiones del código, esencialmente-. Esto significa que se puede configurar de manera segura una rama para cierto desarrollo, una rama para la codificación de proyectos experimentales, una rama para producción y una rama para cualquier otra cosa que sea necesaria. Al finalizar el desarrollo de cierto código, se puede fusionar una rama con otra, por ejemplo la de desarrollo a la de producción.

En PARKIBIP se utilizó una rama de desarrollo llamada “develop” y una rama de producción llamada “master” como se muestra el esquema en la figura [Fig. 7.7](#). Luego, para cada nueva funcionalidad se creó una rama con un nombre que describe la misma y un prefijo “feature/” -indicando que está creado con el propósito de desarrollar una funcionalidad-.

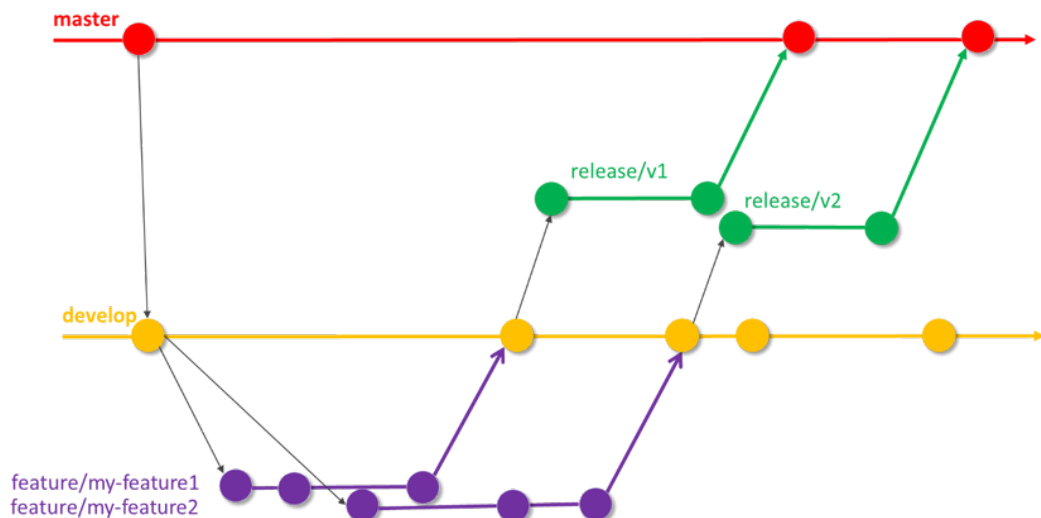


Figura 7.7: Modelo de ramas para flujo de trabajo en git

Una vez terminado el trabajo de desarrollo en esa rama, se crea un MERGE REQUEST contra la rama de desarrollo, utilizando la herramienta Gitlab disponible gratuitamente para alumnos de la Facultad de Ingeniería. Dicha solicitud puede ser revisada por otro integrante del equipo, es decir, realizar comentarios, sugerencias o correcciones sobre los cambios que se están solicitando fusionar. Una vez que el otro integrante aprueba la fusión, la rama puede fusionarse para que la nueva funcionalidad esté disponible en la rama de desarrollo.

De esta forma, es posible organizar las versiones del código de una forma estratégica, sencilla y efectiva. Cuando se quiere incorporar una corrección sobre una funcionalidad ya desarrollada, el proceso es el mismo pero el nombre de la nueva rama tiene el prefijo “fix/”.

7.8. Diagramas de diseño

Para elaborar la documentación de diseño -arquitectura, modelado de procesos-, diagramas de componentes, diagramas de comunicación, diagramas de clase, etc. se utilizó la herramienta Microsoft Visio (© Microsoft 2020). Esta herramienta permite diseñar de manera sencilla e intuitiva diagramas de flujo, organigramas, diseños de ingeniería y todo tipo de diagramas en general.

Fue utilizado principalmente para la elaboración de diagramas de flujo de actividades y datos, conocido como la notación BPMN 2.0 (Business Process Model and Notation), es decir, representaciones gráficas de flujos o procesos de negocio. De esta manera, el modelo permite visualizar de forma clara los eventos, actividades, actores, asociaciones y relaciones, tareas manuales o de sistemas, “timers” disparadores de acciones, y demás. Permite centralizar en un diagrama integrado todas las funcionalidades o historias de usuario del sistema.

7.9. Documentación

Para la documentación del proyecto se utilizó la herramienta Overleaf (© 2020 Overleaf), un editor de LaTeX en línea que permite escribir documentos científicos de forma colaborativa; es decir, que varios colaboradores puedan trabajar en el mismo documento al mismo tiempo. Es utilizada actualmen-

te por cerca de cinco millones de investigadores, estudiantes y profesores en instituciones educativas (p.ej: Harvard, Oxford, entre otras), laboratorios y la industria en general. La herramienta se accesible a través de la web y permite, entre otras cosas:

- Chat con personas que se encuentran trabajando en el mismo documento
- Agregar comentarios a partes específicas del texto
- Responder o resolver comentarios sobre el texto de otros colaboradores
- Disponer de un historial de modificaciones

7.10. Otras herramientas de desarrollo

Como toda investigación y experimentación, siempre es necesario la realización de Pruebas de Concepto (POC, del ingles Proof of concept) que permitan experimentar funcionalidades de una manera rápida y temprana, como también reducir incertidumbres.

Por lo tanto, fueron utilizadas diversas herramientas, como por ejemplo VisualStudio y el lenguaje .NET, Python, Bosch Development Desktop 2.0, etcétera. Principalmente para las actividades:

- Envío y recepción de eventos hacia/desde los dispositivos IMU.
- Recuperación de datos medidos por los sensores.
- Computo de algoritmos numéricos.
- Implementación de otros métodos o técnicas (i.e. algoritmo de fusionado y filtrado de datos, computo de orientación).

Capítulo 8

Conclusiones y trabajo futuro

En líneas generales, este proyecto abarcó todas las fases de una investigación científica -incluso la publicación de un artículo científico-, así como todas las etapas de elaboración de un sistema de software: relevamiento, prototipado, análisis, definición, diseño, implementación, pruebas y liberación.

A través de una revisión sistemática de la literatura, se formó el estado del arte en el área de “Estudio de la marcha de las personas a través de Unidades de Medición Inercial”. Como consecuencia, se recolectó evidencia empírica y se generó conocimiento sobre las distintas técnicas aplicadas a esta problemática en el GAIT ANALYSIS (GA). Asimismo, se recolectaron datos de distintos estudios publicados en todo el mundo y se estableció un contexto en el área, que sirva como fuente de conocimiento para proyectos innovadores como PARKIBIP.

Idear, diseñar y desarrollar un sistema wearable como PARKIBIP para reeducar la marcha de enfermos de Parkinson, es un avance trascendental en el área de la rehabilitación de la marcha. Se construyó una herramienta que **acompaña al enfermo en su vida cotidiana, que es accesible para la persona y que complementa el trabajo de rehabilitación del terapeuta.**

Diseñando, implementando y combinando diversos métodos físico-matemáticos para el procesamiento en tiempo real de las mediciones, se constató que existe una inmensa cantidad de posibilidades para crear y combinar algoritmos que permitan mejorar, aún más, la precisión de los parámetros espacio-temporales extraídos por PARKIBIP. A su vez, se aprecia una complejidad inherente en el desarrollo de estos métodos, que deben ser adecuados para un análisis en tiempo real de los datos (i.e medición a medición).

En cuanto a la detección de las fases de la marcha y a la aplicación del biofeedback, objetivo principal del proyecto, se obtuvieron excelentes resultados. Sin embargo, existe una oportunidad de mejora considerable en la extracción de los parámetros espacio-temporales, como lo es la trayectoria. La acumulación del error en el cálculo de la posición, aunque fuera contrarrestada con las actualizaciones de velocidad cero, generó falta de precisión en el cálculo de la trayectoria como para considerarla un buen resultado. A causa de la priorización de tareas y al tiempo acotado en el proyecto -ampliamente extendido a lo planificado-, no se alcanzó ahondar en la corrección de este parámetro.

En virtud del contexto actual generado por la pandemia global de COVID-19, no fueron realizadas sesiones experimentales con enfermos de Parkinson, ya que la gran mayoría de ellos pertenece a la población de riesgo. Estas pruebas, serán de crucial importancia para futuros avances de PARKIBIP. Además, se sugiere la evaluación de los parámetros espacio-temporales estimados por el sistema creado, frente a un sistema de análisis de marcha estándar de oro.

Con PARKIBIP, se provee una herramienta que permite avanzar en el campo de la biomecánica hacia el entendimiento de *cómo los estímulos externos favorecen la marcha de las personas* con enfermedad de Parkinson. El sistema es capaz de detectar las fases de la marcha en tiempo real, estimular al paciente a través de reglas pre-configuradas y recolectar datos objetivos sobre las características de la marcha de las personas.

Por lo tanto, es una herramienta de gran utilidad para el terapeuta, quién puede realizar un análisis objetivo respecto a la marcha de cada paciente, y luego evaluar su progreso en el transcurso de la rehabilitación. Además, se puede dar continuidad y seguimiento al trabajo de rehabilitación sin la obligación de concurrir a un centro clínico o tener acceso a un costoso y especializado laboratorio de marcha.

Los algoritmos utilizados, si bien logran buenos resultados, son un punto de partida para futuros desarrollos. El avance a pasos agigantados en tecnologías móviles y wearables, genera un extenso terreno fértil para trabajar en el progreso de dichos métodos.

Existen numerosas posibilidades en cuanto a la capacidad que se le puede dar al módulo que establece las reglas de BIOFEEDBACK; y es aquí donde vemos la mayor de las oportunidades de mejora. Se propone la posibilidad de desarrollar nuevas reglas clínicas o de utilizar inteligencia artificial para establecerlas, según las características de la marcha del sujeto que se está evaluando.

Integrar la aplicación con un servidor capaz de procesar los datos de los pacientes y generar métricas sobre todas sus sesiones, sería una herramienta extremadamente útil para la evaluación clínica. Se podría, además, adjuntar la información de las sesiones a la historia clínica electrónica de la persona para avanzar en la centralización de la información de los pacientes.

Sería deseable, realizar sesiones experimentales con enfermos de Parkinson previamente seleccionados, planificar dichas sesiones con el fin de comparar para cada paciente sus diversas sesiones.

Por último, pero no menos importante, a partir de la experiencia PARKI-BIP, se puede afirmar que el trabajo multidisciplinario es clave para avances que generen valor en campos como la medicina, acortando la brecha generada a partir de la alta especialización de los profesionales en una única disciplina.

La enfermedad de Parkinson es un trastorno extremadamente difícil para las personas que la sufren, así como para su núcleo familiar. El hecho de emplear conceptos de los que su aplicación fue puesta en tela de juicio en una temprana etapa de la carrera de Ing. en Computación (e.g. álgebra de matrices o las leyes de Newton), para aportar un pequeño grano de arena a la mejora de su calidad de vida, es -además de gratificante- una lección enriquecedora sobre la aplicación de la Ingeniería.

Referencias bibliográficas

- (2016). Inertial BSN-Based Characterization and Automatic UPDRS Evaluation of the Gait Task of Parkinsonians. *IEEE Transactions on Affective Computing*, 7(3):258–271.
- Akbari, A., Dewey, R. B., and Jafari, R. (2017). Validation of a new model-free signal processing method for gait feature extraction using inertial measurement units to diagnose and quantify the severity of parkinsons disease. In *2017 26th International Conference on Computer Communication and Networks (ICCCN)*. IEEE.
- Almeida, Q. and Bhatt, H. (2012). A manipulation of visual feedback during gait training in parkinson’s disease. *Parkinson’s disease*, 2012:508720.
- Alzugaray, M., Vico, S., Vega, C., Rey, G., and Simini, F. (2019). Sepepe seguimiento perinatal personalizado. Montevideo. Poster - Ingeniería de Muestra.
- Aminian, K., Najafi, B., Büla, C., Leyvraz, P.-F., and Robert, P. (2002). Spatio-temporal parameters of gait measured by an ambulatory system using miniature gyroscopes. *Journal of Biomechanics*, 35(5):689 – 699.
- Anwary, A. R., Yu, H., and Vassallo, M. (2018). Optimal foot location for placing wearable imu sensors and automatic feature extraction for gait analysis. *IEEE Sensors Journal*, 18(6):2555–2567.
- Ashburn, A., Stack, E., Pickering, R. M., and Ward, C. D. (Jan. 2001). A community-dwelling sample of people with parkinson’s disease: characteristics of fallers and non-fallers. *Age Ageing*, 30:47–52.
- Auth0 (2020 (accessed October 3, 2020)a). *Authorization Code Flow with Proof Key for Code Exchange (PKCE)*.

- Auth0 (2020 (accessed October 3, 2020)b). *Secure access for everyone. But not just anyone.*
- Ayyappa, E. (1997). Normal human locomotion. Part 1: Basic concepts and terminology. *Prosthetics Orthot*, 9:10–17.
- Barthel, C., Mallia, E., Debû, B., Bloem, B., and Ferraye, M. (2016). The practicalities of assessing freezing of gait. *Journal of Parkinson's Disease*, 6:1–8.
- Basmajian, J. (1977). Motor learning and control: a working hypothesis. *Archives of Physical Medicine and Rehabilitation*, 58(1):38–41. cited By 93.
- Bertoli, M., Cereatti, A., Trojaniello, D., Avanzino, L., Pelosin, E., Din, S. D., Rochester, L., Ginis, P., Bekkers, E. M. J., Mirelman, A., Hausdorff, J. M., and Croce, U. D. (2018). Estimation of spatio-temporal parameters of gait from magneto-inertial measurement units: multicenter validation among parkinson, mildly cognitively impaired and healthy older adults. *BioMedical Engineering OnLine*, 17(1).
- Blischak, J. D., Davenport, E. R., and Wilson, G. (2016). A quick introduction to version control with git and github. *PLoS Computational Biology*, 12(1):1.
- Bloem, B. R., Hausdorff, J. M., Visser, J. E., and Giladi, N. (2004). Falls and freezing of gait in parkinson's disease: A review of two interconnected, episodic phenomena. *Movement Disorders*, 19(8):871–884.
- Bugané, F., Benedetti, M., Casadio, G., Attala, S., Biagi, F., Manca, M., and Leardini, A. (2012). Estimation of spatial-temporal gait parameters in level walking based on a single accelerometer: Validation on normal subjects by standard gait analysis. *Computer Methods and Programs in Biomedicine*, 108(1):129 – 137.
- Campdelacreu, J. (2014). Enfermedad de parkinson y enfermedad de alzheimer: factores de riesgo ambientales. *Neurología*, 29(9):541 – 549.
- Castillo, H. and Simini, F. (2019). SIMIC: Sistema Informático de Monitoreo de la Insuficiencia Cardíaca. Montevideo. Poster - Ingeniería de Muestra.
- Chebanyuk, E. and Markov, K. (2016). An approach to class diagrams verification according to solid design principles. In *2016 4th International*

- Conference on Model-Driven Engineering and Software Development (MODELSWARD)*, pages 435–441.
- Chávez León, E., Ontiveros Uribe, M. P., and Carrillo Ruiz, J. D. (2013). La enfermedad de Parkinson: neurología para psiquiatras. *Salud mental*, 36:315 – 324.
- Commons, A. (2020 (accessed Novemer 14, 2020)). *Commons Math: The Apache Commons Mathematics Library*.
- Conger, S. A. (2011). Physical activity assessment in wheelchair users. In *Graduate School at Trace: Tennessee Research and Creative Exchange*.
- de Lau, L. M. L. and Breteler, M. M. B. (2006). Epidemiology of parkinson’s disease. *The Lancet. Neurology*, 5(6):525 – 535.
- Dewey, D., Miocinovic, S., Bernstein, I., Khemani, P., Dewey, Jr, R., Query, R., and Chitnis, S. (2014). Automated gait and balance parameters diagnose and correlate with severity in parkinson disease. *Journal of the neurological sciences*, 345.
- Dias, N. P., Fraga, D., Cacho, E. W. A., and Oberg, T. D. (2017). Treino de marcha com pistas visuais no paciente com doença de parkinson. *Fisioterapia em Movimento*, 18:43–51.
- Dillmann, U., Holzoffer, C., Johann, Y., Bechtel, S., Gräber, S., Massing, C., Spiegel, J., Behnke, S., Bürmann, J., and Louis, A. K. (2014). Principal component analysis of gait in parkinson’s disease: Relevance of gait velocity. *Gait & Posture*, 39(3):882 – 887.
- Dorsey, E., Elbaz, A., Nichols, E., Abd-Allah, F., Abdelalim, A., Adsuar, J., Ansha, M., Brayne, C., Choi, J.-Y., Collado-Mateo, D., Dahodwala, N., Do, P. H., Edessa, D., Endres, M., Fereshtehnejad, S.-M., Foreman, K., Gankpe, F., Gupta, R., Hankey, G., and Murray, C. (2018). Global, regional, and national burden of parkinson’s disease, 1990–2016: a systematic analysis for the global burden of disease study 2016. *The Lancet Neurology*, 17.
- Dorsey, E. R., Constantinescu, R., Thompson, J. P., Biglan, K. M., Holloway, R. G., Kieburtz, K., Marshall, F. J., Ravina, B. M., Schifitto, G., Siderowf,

- A., and Tanner, C. M. (2007). Projected number of people with parkinson disease in the most populous nations, 2005 through 2030. *Neurology*, 68(5):384–386.
- Fernández Del Olmo, M., Arias, P., and J, C. (2004). Facilitación de la actividad motora por estímulos sensoriales en la enfermedad de parkinson. *Revista de neurologia*, page 841.
- Fox, C., Rammig, L., Ciucci, M., McFarland, D., Farley, B., Goodwin, V., Richards, S., Taylor, R., Taylor, A., and Campbell, J. (2008). The effectiveness of exercise interventions for people with parkinson’s disease: a systematic review and meta-analysis. *Mov. Disord.*, 23:632–710.
- Frank, D. L., Khorshid, L., Kiffer, J. F., Moravec, C. S., and McKee, M. G. (2010). Biofeedback in medicine: Who, when, why and how?
- Galnares, M., Low, R., Nesmachnow, S., and Simini, F. (2018 June 3-8). Medical reasoning oriented orthosis to ease clinical practice and record keeping. Prague. World Congress on Medical Physics and Biomedical Engineering (WC2018).
- Garcés Vieira, M. and Suárez Escudero, J. (2014). Neuroplasticity: Biochemical and neurophysiological aspects. *CES Medicina*, 28:119 – 132.
- García, S., Suárez, S. S., Dávalo, E. M., and Castillo, J. L. (2008). Perspectiva histórica y aspectos epidemiológicos de la enfermedad de parkinson. *Medicina Interna de Mexico*, 24(1):28 – 37.
- Garrett, A. (2004). Biology of Parkinson’s disease: pathogenesis and pathophysiology of a multisystem neurodegenerative disorder. *Dialogues in clinical neuroscience*, 6(3):259–80.
- Ginis, P., Nieuwboer, A., Dorfman, M., Ferrari, A., Gazit, E., Canning, C. G., Rocchi, L., Chiari, L., Hausdorff, J. M., and Mirelman, A. (2016). Feasibility and effects of home-based smartphone-delivered automated feedback training for gait in people with Parkinson’s disease: A pilot randomized controlled trial. *Parkinsonism & Related Disorders*, 22:28–34.
- Goetz, C., Tilley, B., Shaftman, S., Stebbins, G., Fahn, S., Martinez-Martin, P., Poewe, W., Sampaio, C., Stern, M., Dodel, R., Dubois, B., Holloway,

- R., Jankovic, J., Kulisevsky, J., Lang, A., Lees, A., Leurgans, S., Lewitt, P., Nyenhuis, D., and Lapelle, N. (2008). Movement disorder society-sponsored revision of the unified parkinson’s disease rating scale (mds-updrs): Scale presentation and clinimetric testing results. *Movement disorders : official journal of the Movement Disorder Society*, 23:2129–70.
- Google, LLC. (2020). *Android Context*.
- Gómez-González, J., Martín-Casas, P., and de-la Cuerda, R. C. (2019). Efectos de los estímulos auditivos en la fase de iniciación de la marcha y de giro en pacientes con enfermedad de parkinson. *Neurología*, 34(6):396 – 407.
- Hassenzahl, M. and Tractinsky, N. (2006). User experience - a research agenda. *Behaviour & Information Technology*, 25(2):91–97.
- Hausdorff, J. M., Cudkowicz, M. E., Firtion, R., Wei, J. Y., and Goldberger, A. L. (May 1998). Gait variability and basal ganglia disorders: stride-to-stride variations of gait cycle timing in parkinson’s disease and huntington’s disease. *Mov. Disord.*, 13(3):428–37.
- Jankovic, J., Nutt, J., and Sudarsky, L. (Jan. 2001). Classification diagnosis and etiology of gait disorders. *Adv. Neurol*, 87:119–33.
- Jellinger, K. A. (2002). Recent developments in the pathology of parkinson’s disease. In Jellinger, K. A., Schmidt, R., and Windisch, M., editors, *Ageing and Dementia Current and Future Concepts*, pages 347–384, Vienna. Springer Vienna.
- Kitchenham, B. (2006). Evidence-based software engineering and systematic literature reviews. volume 51, page 3.
- Kitchenham, B., Dyba, T., and Jorgensen, M. (2004). Evidence-based software engineering. page 273–281.
- Kumar, S., Gopinath, K., Rocchi, L., Sukumar, P. T., Kulkarni, S., and Sampath, J. (2018). Towards a portable human gait analysis monitoring system. In *2018 International Conference on Signals and Systems (ICSigSys)*, pages 174–180.

- Lamotte, G., Rafferty, M., Prodoehl, J., Kohrt, W., Comella, C., Simuni, T., and Corcos, D. (2014). Effects of endurance exercise training on the motor and non-motor features of parkinson’s disease: A review. *Journal of Parkinson’s disease*, 5.
- Li, Y. and Guo, Y. (2015). Wiki-health: from quantified self to self-understanding. *future gener. comput. syst. J. Neurol.*, 56:333–359.
- LLC, G. (2020). Material Design by google.
- Lopez, W. O. C., Higuera, C. A. E., Fonoff, E. T., de Oliveira Souza, C., Albicker, U., and Martinez, J. A. E. (2014). Listenmee® and Listenmee® smartphone application: Synchronizing walking to rhythmic auditory cues to improve gait in Parkinson’s disease. *Human Movement Science*, 37:147–156.
- Lord, S., Baker, K., Nieuwboer, A., Burn, D., and Rochester, L. (Oct. 2010). Gait variability in parkinson’s disease: an indicator of non-dopaminergic contributors to gait dysfunction? *J. Neurol.*, 258(4):566–572.
- Loyola, I. P. (2017). Evaluación de los efectos de un programa individualizado de entrenamiento de la marcha en pacientes con enfermedad de parkinson y su impacto en la calidad de vida. Master’s thesis, Departamento y Cátedra de Rehabilitación y Medicina Física, Universidad de la República.
- Madgwick, S. O. (2010). An efficient orientation filter for inertial and inertial/magnetic sensor arrays.
- Mahony, R., Hamel, T., and Pflimlin, J.-M. (2006). Complementary filter design on the special orthogonal group $so(3)$. pages 1477 – 1484.
- Mao, G., Drake, S., and Anderson, B. (2007). Design of an extended kalman filter for uav localization. pages 224 – 229.
- Mark Stephen, S. and Associates (1998). *Definitions of biofeedback and applied Psychophysiology and Biofeedback*. The Guilford Press.
- Mark Stephen, S. and Frank, A. (2016). *Definitions of biofeedback*. . New York. The Guilford Press.

- Marras, C., , Beck, J. C., Bower, J. H., Roberts, E., Ritz, B., Ross, G. W., Abbott, R. D., Savica, R., Eeden, S. K. V. D., Willis, A. W., and Tanner, C. (2018). Prevalence of parkinson’s disease across north america. *npj Parkinsons Disease*, 4(1).
- Martin, P. and Salaün, E. (2010). Design and implementation of a low-cost aided attitude and heading reference system. *Control Engineering Practice*, 18:712–722.
- ME, M., R, I., TA, M., and JJ, S. (1994). The pathogenesis of gait hypokinesia in parkinson’s disease. *Brain : a journal of neurology*, 117:5.
- Merat, F. (1987). Introduction to robotics: Mechanics and control. *Robotics and Automation, IEEE Journal of*, 3:166 – 166.
- Mille, M.-L., Creath, R., Prettyman, M., Hilliard, M., Martinez, K., Mackinnon, C., and Rogers, M. (2012). Posture and locomotion coupling: A target for rehabilitation interventions in persons with parkinson’s disease. *Parkinson’s disease*, 2012:754186.
- Ministerio de S. de Salud Pública, S. (2010). Guía clínica enfermedad de parkinson. *Santiago Minsal-Cl*, 1:1–46.
- Modise, T. D., Steyn, N., and Hamam, Y. (2016). Linear progression measurement and analysis of human gait for the development of a multifunctional robotic walker. In *2016 Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference (PRASA-RobMech)*, pages 1–6. IEEE.
- Molano, C., Chen, L., and Fu, L. (2019). Robotic walker with high maneuverability through deep learning for sensor fusion. In *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pages 1024–1029.
- Moros, J., Ballero, F., Jáuregui, S., and Carroza, M. (2000). Rehabilitación en el ictus rehabilitation in the stroke. *ANALES Sis San Navarra*.
- Morris, M., Ianseck, R., Matyas, T., and Summers, J. (1996). Stride length regulation in parkinson’s disease. normalization strategies and underlying mechanisms. *Brain : a journal of neurology*, 119 (Pt 2):551–68.

- Morris, M. E., Huxham, F., McGinley, J., Dodd, K., and Iansek, R. (2001). The biomechanics and motor control of gait in parkinson disease. *Clinical Biomechanics*, 16(6):459 – 470.
- Muñoz-Hellín, E., de-la Cuerda, R. C., and Miangolarra-Page, J. C. (2013). Guías visuales como herramienta terapéutica en la enfermedad de parkinson. una revisión sistemática. *Revista Española de Geriátría y Gerontología*, 48(4):190 – 197.
- Nasir, N., Zakaria, M., Razali, S., and Abu, M. (2017). Autonomous mobile robot localization using kalman filter. *MATEC Web of Conferences*, 90:01069.
- Navarro, G. P., Pérez, S. A., and García, I. G. M. (2016). Effects of sensory cueing in virtual motor rehabilitation. a review. *Journal of Biomedical Informatics*, 60:49 – 57.
- Nieuwboer, A., Kwakkel, G., Rochester, L., Jones, D., van Wegen, E., Willems, A. M., Chavret, F., Hetherington, V., Baker, K., and Lim, I. (2007). Cueing training in the home improves gait-related mobility in Parkinson’s disease: the RESCUE trial. *Journal of Neurology, Neurosurgery & Psychiatry*, 78(2):134–140.
- Nieuwboer, P. A. and Giladi, N. (2008). The challenge of evaluating freezing of gait in patients with parkinson’s disease. *British Journal of Neurosurgery*, 22(sup1):S16–S18.
- Nonnekes, J., Snijders, A., Nutt, J., Deuschl, G., Giladi, N., and Bloem, B. (2015). Freezing of gait: a practical approach to management. *Lancet Neurol*, 14(7):768–78.
- Nutt, J., Bloem, B., Giladi, N., Hallett, M., Horak, F., and Nieuwboer, A. (2011). Freezing of gait: Moving forward on a mysterious clinical phenomenon. *Lancet neurology*, 10:734–44.
- Olivares, R., Silvera, G., Alvarez-Rocha, P., Taramasco, C., Ormaechea, G., and Simini, F. (2019). High-level language to specify an adaptive heart failure follow up strategy. SABI2020.
- OpenShoe (2020 (accessed Novemer 14, 2020)). *The OpenShoe Matlab Implementation*.

- Ostrosky, F. (2000). Características neuropsicológicas de la enfermedad de parkinson. *Revista de Neurología*, 30:788.
- Parisi, F., Ferrari, G., Giuberti, M., Contin, L., Cimolin, V., Azzaro, C., Albani, G., and Mauro, A. (2015). Body-sensor-network-based kinematic characterization and comparative outlook of updrs scoring in leg agility, sit-to-stand, and gait tasks in parkinson’s disease. *IEEE journal of biomedical and health informatics*.
- Parkinson, J. (2002). NEUROPSYCHIATRY CLASSICS An Essay on the Shaking Palsy Member of the Royal College of Surgeons PREFACE. Technical Report 2.
- Patel, S., Lorincz, K., Hughes, R., Huggins, N., Growdon, J., Standaert, D., Akay, M., Dy, J., Welsh, M., and Bonato, P. (2009). Monitoring motor fluctuations in patients with parkinson’s disease using wearable sensors. *IEEE transactions on information technology in biomedicine : a publication of the IEEE Engineering in Medicine and Biology Society*, 13(6):864—873.
- Perez-Lloret, S., Negre-Pages, L., Damier, P., Delval, A., Derkinderen, P., Destée, A., Meissner, W. G., Schelosky, L., Tison, F., and Rascol, O. (2014). Prevalence, Determinants, and Effect on Quality of Life of Freezing of Gait in Parkinson Disease. *JAMA Neurology*, 71(7):884–890.
- Pirker, W. and Katzenschlager, R. (2016). Gait disorders in adults and the elderly: A clinical guide. *Wiener klinische Wochenschrift*, 129.
- R., A. and M., M. (2012). Programa de Educación y Rehabilitación en la Enfermedad de Parkinson. Technical report.
- Ribeiro, N. F. and Santos, C. P. (2017). Inertial measurement units: A brief state of the art on gait analysis. *ENBENG 2017 - 5th Portuguese Meeting on Bioengineering, Proceedings*, pages 1–4.
- Rocha, P. A., Porfírio, G. M., Ferraz, H. B., and Trevisani, V. F. (2014). Effects of external cues on gait parameters of Parkinson’s disease patients: A systematic review. *Clinical Neurology and Neurosurgery*, 124:127–134.
- Rochester, L., Baker, K., Hetherington, V., Jones, D., Willems, A.-M., Kwakkel, G., Van Wegen, E., Lim, I., and Nieuwboer, A. (2010). Evidence for

- motor learning in Parkinson's disease: Acquisition, automaticity and retention of cued gait performance after training with external rhythmical cues. *Brain Research*, 1319:103–111.
- Rogers, W. (Nov. 1996). Disorders of posture balance and gait in parkinson's disease. *Clin. Geriatr. Med.*, 12(4):825–45.
- Roiz, R. d. M., Cacho, E. W. A., Pazinato, M. M., Reis, J. G. A., Cliquet Jr, A., and Barasnevicus-Quagliato, E. M. (2010). Gait analysis comparing Parkinson's disease with healthy elderly subjects. *Arquivos de Neuro-Psiquiatria*, 68:81 – 86.
- Rubinstein, T., Giladi, N., and Hausdorff, J. (2002). The power of cueing to circumvent dopamine deficits: A review of physical therapy treatment of gait disturbances in parkinson's disease. *Movement disorders : official journal of the Movement Disorder Society*, 17:1148–60.
- Sande de souza, L., Dionisio, V., and Almeida, G. L. (2011). Multi-joint movements with reversal in parkinson's disease: Kinematics and electromyography. *Journal of electromyography and kinesiology : official journal of the International Society of Electrophysiological Kinesiology*, 21:376–83.
- Saunders, J. B., Inman, V. T., and Eberhard, H. D. (1953). The major determinants in normal and pathological gait. *The Journal of bone and joint surgery. American volume*, 35-A(3):543–58.
- Schwartz, M. S. and Olson, R. P. (1995). *A historical perspective on the field of biofeedback and applied psychophysiology*. . New York. The Guilford Press.
- Sebastián, P., Fernando, A., Vanessa, C., Silvana, M., Rodrigo, G., Joaquín, L., and Diego, V. (2019). Conceptos de ingeniería de software basada en evidencias.
- Skog, I. (2009). *Low-Cost Navigation Systems A Study of Four Problems*. PhD thesis, KTH, School of Electrical Engineering - Stockholm, Sweden . Doctoral Thesis in Signal Processing.
- Sofuwa, O., Nieuwboer, A., Desloovere, K., Willems, A.-M., Chavret, F., and Jonkers, I. (2005). Quantitative gait analysis in parkinson's disease: Comparison with a healthy control group. *Archives of physical medicine and rehabilitation*, 86:1007–13.

- Spaulding, S. J., Barber, B., Colby, M., Cormack, B., Mick, T., and Jenkins, M. E. (2013). Cueing and Gait Improvement Among People With Parkinson's Disease: A Meta-Analysis. *Archives of Physical Medicine and Rehabilitation*, 94(3):562–570.
- Stamatakis, J., Crémers, J., Maquet, D., Macq, B. M., and Garraux, G. (30 Aug.-3 Sept). Gait feature extraction in parkinson's disease using low-cost accelerometers. In *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, page 7900–7903. The organization, IEEE. An optional note.
- StatCounter (2020). StatCounter © 1999-2020.
- Taborri, J., Palermo, E., Rossi, S., and Cappa, P. (2016). Gait partitioning methods: A systematic review.
- Tomlinson, C., Patel, S., Meek, C., Clarke, C., Stowe, R., Shah, L., Sackley, C., Deane, K., Herd, C., Wheatley, K., and Ives, N. (2012). Physiotherapy versus placebo or no intervention in parkinson's disease. *Cochrane database of systematic reviews (Online)*, 7:CD002817.
- Tong, K. and Granat, M. H. (1999). A practical gait analysis system using gyroscopes. *Medical Engineering and Physics*, 21(2):87 – 94.
- Wang, L., Yi, J., and Liu, T. (2018). Gait adaptable human-robot interaction system and its application to a robotic walker. In *2018 IEEE 8th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*, pages 707–712.
- Zago, M., Sforza, C., Pacifici, I., Cimolin, V., Camerota, F., Celletti, C., Condoluci, C., Pandis], M. F. D., and Galli, M. (2018). Gait evaluation using inertial measurement units in subjects with parkinson's disease. *Journal of Electromyography and Kinesiology*, 42:44 – 48.

Glosario

Un glosario incluye una lista de términos y su explicación sucinta. El objetivo de este apartado es permitirle al lector especializado en el área, aunque no necesariamente en la temática, comprender con mayor facilidad ciertos términos utilizados en el documento.

.NET .NET Framework (pronunciado como "dot net") es un marco de software desarrollado por Microsoft que se ejecuta principalmente en Microsoft Windows. **160**

AHRS Los Attitude and Heading Reference Systems (AHRS) o Sistemas de Referencia de Actitud y Rumbo, son sensores tridimensionales que proporcionan información acerca del rumbo, la actitud, y la guiñada de una aeronave. Este tipo de sistemas están específicamente diseñados para reemplazar a los antiguos instrumentos de control giroscópicos, y proporcionar una mejor precisión y fiabilidad. **67**

API API es una sigla que procede de la lengua inglesa y que alude a la expresión Application Programming Interface (cuya traducción es Interfaz de Programación de Aplicaciones). El concepto hace referencia a los procesos, las funciones y los métodos que brinda una determinada biblioteca de programación a modo de capa de abstracción para que sea empleada por otro programa informático. **101**

Actividades de la vida diaria Las actividades de la vida diaria, también conocidas como áreas de ocupación, son todas aquellas tareas y rutinas típicas que una persona realiza diariamente y que le permiten vivir de forma autónoma e integrada en la sociedad, cumpliendo así su rol dentro de ella. **3, 13**

Asociación Uruguaya de Parkinson (AUP) La Asociación Uruguaya de Parkinson es una organización uruguaya que tiene como objetivo con-

tribuir a la investigación y la elaboración de material teórico y práctico sobre calidad de vida e intervención en grupos terapéuticos para personas con enfermedad de parkinson y sus familiares. 169

Association for Applied Psychophysiology and Biofeedback La Asociación de Psicofisiología Aplicada y Biofeedback (AAPB) se fundó en 1969 como Biofeedback Research Society. Los objetivos de la asociación son promover una comprensión científica del biofeedback y avanzar en los métodos utilizados en la práctica y la aplicación. AAPB es una organización sin fines de lucro. 19

Background Se dice que una instrucción de una aplicación se está ejecutando en background cuando se está ejecutando en un hilo de procesamiento diferente al hilo principal. Este tipo de ejecución permite ejecutar de forma asíncrona o en paralelo las tareas para no sobrecargar al hilo principal. 93

Backlog La palabra inglesa Backlog significa “acumulación de algo, especialmente trabajo incompleto o cosas de las que debemos ocuparnos”. Para la Ingeniería de Software, el backlog es una lista de todo el trabajo pendiente, ordenado por prioridad. 187

Backward snowballing Es una estrategia utilizada como técnica de búsqueda dentro de una REVISIONES SISTEMÁTICAS DE LA LITERATURA (SLR). La técnica consiste en partir de un estudio primario relevante encontrado previamente y utilizar sus referencias para encontrar nuevos estudios relevantes. 29

Biofeedback El biofeedback es, en resumidas cuentas, una técnica basada en un sistema de sensores gracias a los cuales el paciente es consciente en tiempo real de varios parámetros fisiológicos que describen el funcionamiento de su cuerpo. El individuo es informado en tiempo real acerca de lo que ocurre en varias partes de su cuerpo, aprende a asociar esos fenómenos con ciertas sensaciones y, con un poco de práctica, va siendo más capaz de regular ciertas funciones corporales. 5, 18, 195

Discinesia Discinesia, en medicina, es un término utilizado para designar la presencia de movimientos anormales e involuntarios. 2, 12

Dopamina Las neuronas utilizan un químico cerebral, llamado dopamina, para ayudar a controlar el movimiento muscular. Es un neurotransmisor producido en una amplia variedad de animales, incluidos tanto vertebrados como invertebrados. Cuando se presenta el mal de Parkinson, las neuronas que producen dopamina mueren lentamente. Sin la dopamina, las células que controlan el movimiento no pueden enviar mensajes apropiados a los músculos. [1](#), [12](#)

Enfermedad de Parkinson La enfermedad de Parkinson (EP) es un trastorno neurodegenerativo multisistémico que afecta al sistema nervioso central (SNC) provocando la aparición de síntomas motores y no motores [8](#)

Evidencia Una evidencia (del latín, *evidens*, 'visible, evidente, manifiesto') es un conocimiento que se nos aparece intuitivamente de tal manera que podemos afirmar la validez de su contenido, como verdadero, con certeza, sin sombra de duda. En un sentido más restringido se denomina evidencia a cualquier conocimiento o prueba que corrobora la verdad de una proposición. [23](#)

Forward snowballing Es una estrategia utilizada como técnica de búsqueda dentro de una REVISIONES SISTEMÁTICAS DE LA LITERATURA (SLR). La técnica consiste en partir de un estudio primario relevante encontrado previamente y se buscan que estudios lo ubiquen en sus referencias. [29](#)

Gait analysis El Gait Analysis o Análisis de la Marcha es el estudio sistemático de la locomoción animal, más específicamente el estudio del movimiento humano, utilizando el ojo y el cerebro de los observadores, aumentado por instrumentación para medir los movimientos corporales, la mecánica corporal y la actividad de los músculos. El análisis de la marcha se utiliza para evaluar y tratar a personas con afecciones que afectan su capacidad para caminar. También se usa comúnmente en biomecánica deportiva para ayudar a los atletas a correr de manera más eficiente e identificar problemas relacionados con la postura o el movimiento en personas con lesiones. [6](#), [16](#), [31](#), [181](#), [194](#)

Gradle Gradle es un sistema de automatización de desarrollo de software que construye sobre los conceptos de Apache Ant. y Apache Maven. [92](#)

IDE Un entorno de desarrollo integrado o entorno de desarrollo interactivo, en inglés Integrated Development Environment (IDE), es una aplicación informática que proporciona servicios integrales para facilitarle al desarrollador o programador el desarrollo de software. Normalmente, un IDE consiste de un editor de código fuente, herramientas de construcción automáticas y un depurador. [164](#), [190](#)

Kalman filter En estadística y teoría de control, el filtrado de Kalman, también conocido como estimación cuadrática lineal (LQE), es un algoritmo que utiliza una serie de mediciones observadas a lo largo del tiempo, que contienen ruido estadístico y otras inexactitudes, y produce estimaciones de variables desconocidas que tienden a ser más precisos que los basados en una sola medición, mediante la estimación de una distribución de probabilidad conjunta sobre las variables para cada período de tiempo. El filtro lleva el nombre de Rudolf E. Kálmán, uno de los principales desarrolladores de su teoría. [70](#)

Merge Request En herramientas de versionado de código, un Merge Request es la acción de validar un código que se va a unir de una rama a otra. En este proceso de validación pueden entrar los factores que queramos: Builds (validaciones automáticas), asignación de código a tareas, validaciones manuales por parte del equipo, despliegues, etc. [192](#)

Non-Wearables Se dice que un dispositivo es Non-Wearable, cuando un dispositivo no es Wearable. Los dispositivos Non-Wearables no cumplen con las condiciones para ser “ponibles”, ya sea por su tamaño, peso, requerimiento de conexión, etc. [32](#)

PoC Prueba de concepto (POC), también conocida como prueba de principio, es la realización de un determinado método o idea para demostrar su viabilidad, o una demostración en principio con el objetivo de verificar que algún concepto o teoría tiene potencial práctico. Una prueba de concepto suele ser pequeña y puede estar completa o no. [91](#)

Product Owner El product owner en el desarrollo de software representa la voz de los clientes y crea una visión del producto junto con los interesados. Cada decisión se toma teniendo en cuenta la visión del producto. [185](#)

Reeducar la marcha La marcha es seguramente la tarea funcional más corriente. Esta actividad necesita fuerza, movilidad, propiocepción, coordinación y equilibrio. En el momento de la reeducación de la marcha, el fisioterapeuta debe calcular y analizar cada uno de los componentes de la marcha y así define las necesidades propias del paciente. El entrenamiento de la marcha se refiere a la ayuda al paciente para reaprender a caminar con seguridad y eficientemente. 15

SI El Sistema Internacional de Unidades (abreviado SI, del francés *Système international d'unités*) es un sistema constituido por siete unidades básicas: metro, kilogramo, segundo, kelvin, amperio, mol y candela, que definen a las correspondientes magnitudes físicas fundamentales y que han sido elegidas por convención. 109, 157

Sistema nervioso autónomo (ANS) El sistema nervioso autónomo es la parte del sistema nervioso que controla las acciones involuntarias, tales como los latidos cardíacos y el ensanchamiento o estrechamiento de los vasos sanguíneos. 18

Sistema nervioso central (SNC) En el sistema nervioso central se llevan a cabo los procesos mentales necesarios para comprender la información que se reciben desde el exterior. Asimismo, es el sistema encargado de transmitir ciertos impulsos hacia los nervios y los músculos, por lo que dirige sus movimientos. Se vale de las neuronas (sensoriales y motoras) del encéfalo y la médula espinal para provocar las respuestas precisas a los estímulos que el cuerpo recibe. 3, 15

Systematic literature review La revisión sistemática de la literatura, o revisión sistemática, es un método para identificar, evaluar y resumir el estado del arte de un tema específico en la literatura. En una revisión sistemática, el objetivo es construir una visión general de una pregunta específica y darle un resumen justo de la literatura. 23, 209, 210

Wearables Los Wearables son dispositivos tecnológicos con microprocesadores que, por su tamaño y diseño, se pueden llevar puestos. La palabra proviene del inglés, y hace referencia a accesorios tecnológicos que una persona puede llevar puestos, “wear” es el verbo, y “wearable” es el adjetivo que en español se traduciría como ponible. 5, 20, 32

Wrapper En ciencias de la computación, un Wrapper es cualquier entidad que encapsula (envuelve) otro elemento. Los contenedores se utilizan para dos propósitos principales: convertir datos a un formato compatible u ocultar la complejidad de la entidad subyacente mediante la abstracción. Los ejemplos incluyen envoltorios de objetos, envoltorios de funciones y envoltorios de controladores. [95](#)

ZVD Zero Velocity Detection o detección de velocidad cero refiere a la detección mediante algoritmos numéricos del estado estacionario de un sensor o dispositivo IMU. [69](#)

APÉNDICES

Apéndice 1

PARKIBIP: Manual de usuario del sistema

1.1. Instalación de PARKIBIP

Una vez que se descarga la aplicación PARKIBIP en el teléfono, la instalación de la misma se realiza de manera automática. Así, luego de un proceso de instalación satisfactorio, es preciso ubicar el icono ejecutable de la aplicación descargada, para asegurarse que se encuentra instalada correctamente -ver Fig. 1.1-.

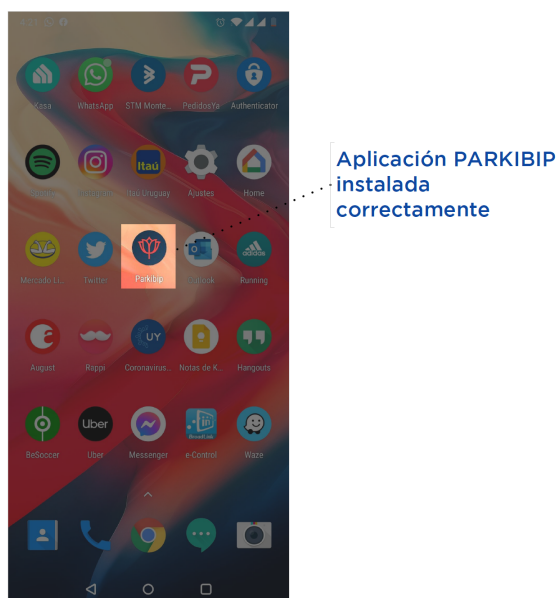


Figura 1.1: Icono de la aplicación PARKIBIP en el menú de aplicaciones.

Al ubicar el icono de la aplicación PARKIBIP en el menú de aplicaciones, presionar sobre la aplicación para que la misma se inicie. Conforme a lograr una mejor experiencia de usuario (UX), de manera instantánea se despliega una pantalla de bienvenida, denominada *splash screen*. Dicha pantalla, actúa como transición entre el menú de aplicaciones y PARKIBIP -mientras se cargan recursos-, así se logra una experiencia más vistosa e intuitiva -ver Fig. 1.2-.



Figura 1.2: Pantalla de bienvenida a PARKIBIP. Mientras la aplicación se prepara para su ejecución -carga recursos en memoria-, actúa como transición entre el menú de aplicaciones y PARKIBIP.

1.2. Autenticación y autorización

En caso de que sea la primera vez que inicie una sesión en la aplicación, mediante un formulario se deberán introducir las credenciales de acceso. Para ello, con un usuario previamente registrado en el sistema, le serán solicitados los campos usuario y contraseña tal como se aprecia en la figura Fig. ???. Luego, se dispara el proceso de autenticación y autorizaciones y el sistema mantendrá iniciada la sesión a partir de la generación de un token válido por un plazo de 60 días. Se resalta que, por cuestiones de privacidad, el sistema no almacena las credenciales del usuario y en cambio emplea un token.



Figura 1.3: Proceso de autenticación y autorización del Usuario. Se requieren los campos nombre de usuario y contraseña.

1.3. Pantalla inicial y menú de navegación

Una vez iniciada la sesión, mediante el proceso de autenticación y autorizaciones, se muestra la pantalla inicial de PARKIBIP -ver Fig. 1.4-. Siendo ésta una de las principales, ya que es el punto de partida para la ejecución de sesiones de rehabilitación.

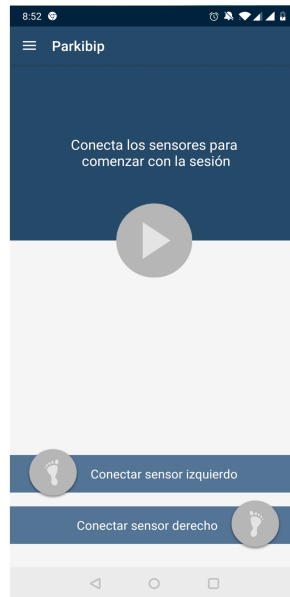


Figura 1.4: Pantalla inicial de PARKIBIP. Dicha pantalla, permite configurar dispositivos IMU e iniciar una sesión de rehabilitación PARKIBIP.

A su vez, la pantalla de inicio, posee un componente moderno de interfaz de usuario. Se trata de un panel lateral que contiene diversas opciones de navegación, y que permanece oculto por defecto en PARKIBIP, tal como se aprecia en la figura previa. Para acceder a esta funcionalidad, basta con deslizar horizontalmente el dedo desde el lateral izquierdo al móvil. Dentro de las opciones de navegación -ver Fig. 1.5-, se encuentran: (i) Inicio, (ii) Historial de sesiones, (iii) Perfiles de usuarios, (iv) Configuración, (v) Cierre de sesión.

Asimismo, para el Usuario logueado, el sistema distingue su Rol Terapeuta/Paciente, auto-genera una imagen y adjunta el email.

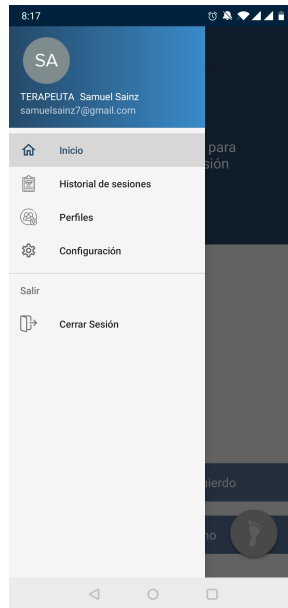


Figura 1.5: Panel lateral de navegación. Contiene diversas opciones para navegar a través de los distintos módulos de PARKIBIP.

1.4. Conexión de IMU

El objetivo de PARKIBIP, es realizar sesiones de rehabilitación mediante dispositivos IMU. Entonces, es necesario configurarlos adecuadamente. Para ello, se requiere que el modulo de bluetooth del móvil este activado.

Ya con bluetooth y en la pantalla inicial, presionando uno de los botones inferiores -con icono de Pie- comienza el proceso de configuración de un dispositivo para dicho pie. Automáticamente el sistema comienza un escaneo de dispositivos IMU en la cercanía, y despliega un listado de dispositivos encontrados -tal como se aprecia en al Fig. 1.6-. Para cada IMU se muestra su identificador MAC, el tipo de dispositivo y la intensidad de la señal del mismo.



Figura 1.6: Escaneo de dispositivos IMU. Previa activación del modulo bluetooth, expone un listado de IMU en la cercania. Para cada uno muestra su identificador MAC, el tipo de IMU y la intensidad de la señal.

Al seleccionar un IMU dentro de la lista resultante del escaneo, PARKIBIP establece las configuraciones necesarias y se conecta al dispositivo inercial. La figura Fig. 1.7 expone el éxito de la conexión pintando el botón con el color azul.

Sumamente importante, PARKIBIP identifica al Pie/IMU con un color en la aplicación, en la imagen rojo, y es el mismo color que comienza a parpadear en el IMU mediante una luz LED.

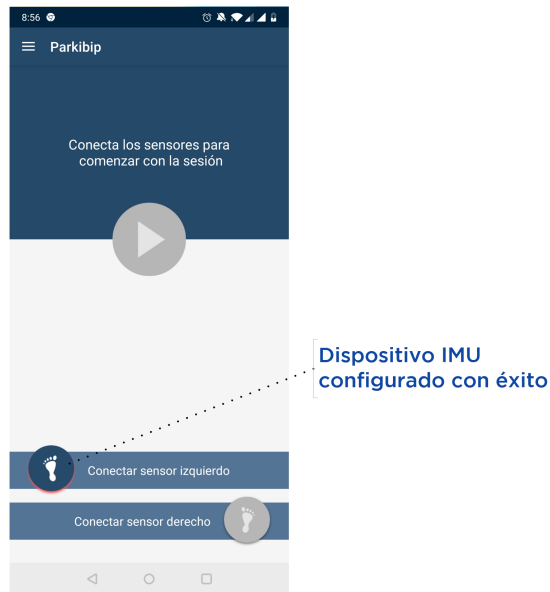


Figura 1.7: Conexión exitosa del IMU asociado al pie izquierdo -botón coloreado de azul-. PARKIBIP identifica al Pie/IMU con un color en la aplicación, en la imagen rojo. Es el mismo color que comienza a parpadear en el IMU mediante su luz LED.

Configurados ambos dispositivos IMU, el sistema habilita el botón que inicia una sesión de rehabilitación PARKIBIP -coloreado con Verde-.

En la figura Fig. 1.8, se aprecia que ambos botones se encuentra en azul, las luces LED del IMU sincronizadas a los colores de los contornos de los botones, y el botón central es puesto en verde.



Figura 1.8: Configuración de los IMU exitosa. Ambos botones se encuentra en azul, las luces LED del IMU sincronizadas a los colores de los contornos de los botones, y el botón central es puesto en verde. PARKIBIP se encuentra listo para comenzar una sesión de rehabilitación.

1.5. Sesión activa de rehabilitación PARKIBIP

Configurados los dispositivos IMU y presionando el botón que comienza una sesión de rehabilitación, PARKIBIP redirige al usuario a la pantalla esencial del sistema: Sesión Activa. La figura Fig. 1.9 permite visualizar un ejemplo -aleatorio- en tiempo real de sesión PARKIBIP.



Figura 1.9: Pantalla de una Sesión Activa PARKIBIP. El sistema reacciona en tiempo real a la marcha del paciente en rehabilitación. Dispara estímulos externos -vibración y/o sonido- y computa los parámetros espacio-temporales.

Tal como se puede apreciar, el sistema reacciona en tiempo real a la marcha del paciente en rehabilitación, mediante el envío de estímulos externos -vibración y/o sonido- y el cómputo de parámetros espacio-temporales. Así, medición a medición, la interfaz de usuario comienza a transitar por distintos valores fruto del procesamiento de PARKIBIP.

1.6. Resumen general y detallado

Finalizada una sesión de rehabilitación PARKIBIP, se podrá evaluar empíricamente la marcha del paciente en cuestión. Así, el sistema redirige al usuario a un síntesis, denominada “Resumen general” y “Resumen detallado” -ver Fig. 1.10-.

Dentro de la vista general, el sistema presenta el comportamiento de la marcha en términos de valores de parámetros espacio-temporales. Luego, en la vista detallada, se propone un análisis gráfico del comportamiento.

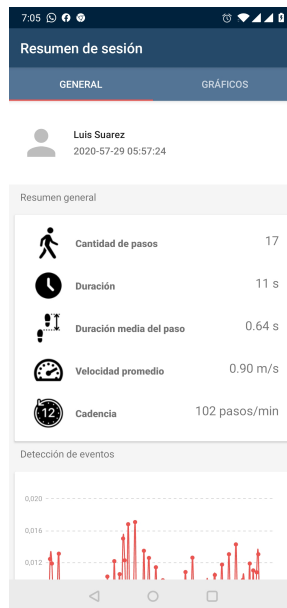


Figura 1.10: Resumen general y detallado de una sesión. Se presenta el comportamiento de la marcha en términos de valores de parámetros espacio-temporales y de forma gráfica.

1.7. Historial de sesiones

Desde el panel lateral y mediante la opción de navegación “Historial de sesiones”, es posible acceder a las distintas sesiones de rehabilitación. Como se puede apreciar en la figura Fig. 1.11, se muestra un listado de sesiones ordenadas cronológicamente con su correspondiente fecha y paciente evaluado. Además, en la barra lateral, es posible buscar y filtrar las distintas sesiones por nombre/documento del paciente o fecha de realización. En caso de querer mas información, se puede seleccionar una sesión y navegar a su resumen.

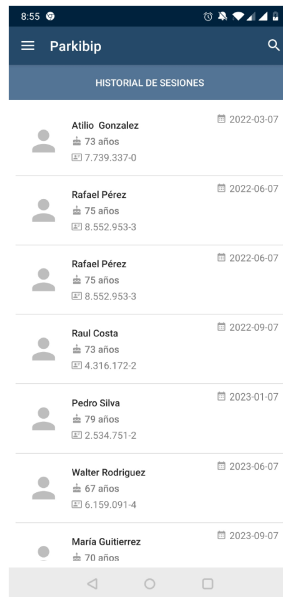


Figura 1.11: Historial de sesiones. Ordenadas cronológicamente con su correspondiente fecha de realización y paciente evaluado. Permite acceder a su resumen mediante su selección.

1.8. Usuario y Pacientes

A partir de la opción “Perfiles”, en el panel de navegación, es posible visualizar: (i) Información descriptiva del usuario que inicio sesión (i.e. logueado), (ii) Listado de pacientes -en caso de ser un terapeuta-, (iii) Paciente activo para futuras sesiones de rehabilitación.

Dentro de las características del Usuario logueado, el sistema distingue el Rol Terapeuta/Paciente.

Por otro lado, el sistema habilita la selección de un paciente dentro del listado para efectuarle sesiones de rehabilitación con PARKIBIP.

Por último, para lograr mayor celeridad, mediante el menú superior es posible buscar y filtrar el listado de paciente.

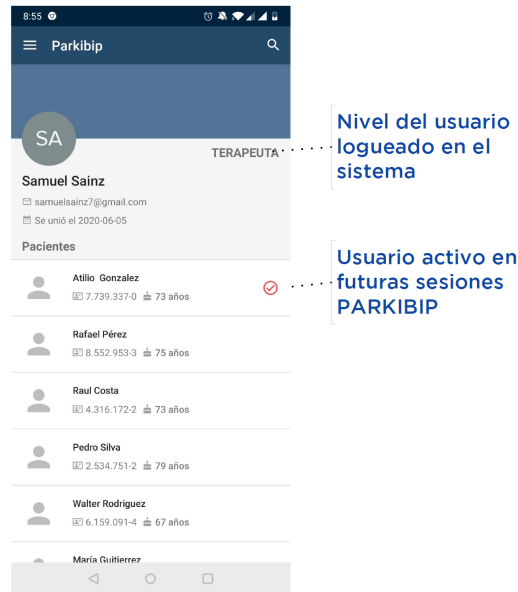


Figura 1.12: Visualización de perfiles de usuarios. Funcionalidades: (i) Ver información descriptiva del usuario logueado, (ii) Ver el listado de pacientes -en caso de ser un terapeuta-, (iii) Seleccionar el paciente activo para futuras sesiones de rehabilitación.

1.9. Configuración de métodos y parámetros

PARKIBIP es una aplicación de ajuste experimental. Es decir, el Terapeuta que evalúa al paciente -previa capacitación técnica-, podrá ajustar los parámetros según las necesidades del paciente en cuestión.

Por lo tanto, PARKIBIP, habilita a modificar los parámetros globales y calibrados por defecto -ver Fig. 1.13-.



Figura 1.13: Configuración de parámetros y selección del algoritmo de detección. PARKIBIP habilita al usuario a modificar parámetros generales y específicos de los métodos.

Además, podrá experimentar distintos algoritmos de detección de fases de la marcha mediante su selección en formato de grupo de opciones, así como también configurar sus distintos parámetros particulares -ver Fig. 1.14-.



Figura 1.14: Restauración de valores. PARKIBIP permite restablecer a valores de fábrica, así como descartar cambios efectuados.

Presionando el botón guardar, del menú superior, los cambios son establecidos.

Por ultimo, pero no menos importante, PARKIBIP viene calibrado con las mejores configuraciones, así que se habilitan las funciones en el menú superior: (i) restablecer a valores de fabrica y (ii) descartar cambios.

Apéndice 2

Entidades de la base de datos relacional SQLite

A continuación se presentan las distintas entidades de la base de datos con su correspondiente estructura:

Tabla 2.1: User

Column name	Type	Properties
IdUser	integer	PK
Name	text	
LastName	text	
CI	text	
Age	integer	null

Tabla 2.2: Session

Column name	Type	Properties
IdSession	integer	PK
MinutesDuration	integer	null
Date	text	null
IdUser	integer	null

Tabla 2.3: Measurement

Column name	Type	Properties
IdMeasurement	integer	PK
Ax	real	null
Ay	real	null
Az	real	null
Gx	real	null
Gy	real	null
Gz	real	null
Mx	real	null
My	real	null
Mz	real	null
Foot	text	null
Timestamp	text	null
IdSession	integer	null

Tabla 2.4: KalmanFilterState

Column name	Type	Properties
IdState	integer	PK
IdSession	integer	
pX	real	null
pY	real	null
pZ	real	null
vX	real	null
vY	real	null
vZ	real	null
roll	real	null
pitch	real	null
heading	real	null
ConnectionId	text	null

Tabla 2.5: ZeroUpdates

Column name	Type	Properties
IdZeroUpdate	integer	PK
IdSession	integer	
ZValue	integer	
ConnectionId	text	null

Tabla 2.6: SessionResult

Column name	Type	Properties
IdSession	integer	PK
IdParameter	integer	PK
Value	text	null

Tabla 2.7: GeneralSetting

Column name	Type	Properties
IdGeneralSetting	integer	PK
Key	text	
Value	text	
Icon	text	null
DefaultValue	text	null

Tabla 2.8: AlgorithmDetection

Column name	Type	Properties
IdAlgorithmDetection	integer	PK
AlgorithmName	text	
AlgorithmDescription	text	null

Tabla 2.9: AlgorithmParameter

Column name	Type	Properties
IdAlgorithmParameter	integer	PK
ParameterName	text	null
ParameterValue	text	null
IdAlgorithmDetection	integer	null
Icon	text	null
DefaultValue	text	null

Tabla 2.10: Parameter

Column name	Type	Properties
IdParameter	integer	PK
Description	text	null
IdUnity	integer	null

Tabla 2.11: Unity

Column name	Type	Properties
IdUnity	integer	PK
MeasurementUnity	text	null
Type	text	null

Apéndice 3

Pruebas funcionales: Matriz de casos de prueba

De acuerdo con la etapa de planeación, fueron diseñados los casos de prueba en la herramienta tabular Excel, mediante una matriz de casos de prueba con la definición de los atributos: identificador, escenario, caso de test, pre-condición, prioridad, etapa, resultado esperado, entre otros. La tabla Tab.3.1 muestra ciertas pruebas funcionales que se ejecutaron en PARKIBIP.

Tabla 3.1: Casos de prueba funcionales del sistema PARKIBIP, definidos por los campos identificador del caso, modulo del sistema, escenario de prueba, caso dentro del escenario, etapa, pre-condición y prioridad.

Id_caso_test	Módulo	Escenario	CASO	Pre-condición	Prioridad
TC_AUTH-001	Autenticación	Verificar login con Auth0	En la pantalla de inicio a la aplicación, se ingresa un usuario y contraseña validos	Usuario y contraseña validos en PARKIBIP	BAJA
TC_AUTH-002	Autenticación	Verificar login con Auth0	En la pantalla de inicio a la aplicación, se ingresa un usuario y contraseña inválidos	Usuario y contraseña inválidos para PARKIBIP	BAJA
TC_AUTH-003	Autenticación	Verificar logout con Auth0	Dentro de una sesión de aplicación y en el menú de navegación, se ejecuta el botón cerrar sesión	Sesión vigente en el sistema	BAJA

Tabla 3.1: Casos de prueba funcionales del sistema PARKIBIP, definidos por los campos identificador del caso, modulo del sistema, escenario de prueba, caso dentro del escenario, etapa, pre-condición y prioridad.

Id_caso_test	Módulo	Escenario	CASO	Pre-condición	Prioridad
TC_AUTH-004	Autenticación	Chequear descifrado de token JWT	En caso de un inicio de sesión satisfactorio, el proveedor de <i>Idaas -Auth0-</i> en el paso final del flujo de autenticación un token del tipo JWT. PARKIBIP interpreta el mismo, para adquirir información del usuario. Se desea chequear dicha interpretación.	Usuario y contraseña validos en PARKIBIP. Proceso de validación de ingreso en curso	BAJA
TC_AUTH-005	Autenticación	Distinguir niveles de Usuarios con Auth	Previo a finalizar el flujo de autenticación y autorización entre PARKIBIP-Auth0, se desea verificar los roles del usuario, para una posterior distinción dentro de la aplicación. Roles: Terapeuta/Paciente	Usuario y contraseña validos en PARKIBIP. Proceso de validación de ingreso en curso	BAJA
TC_IMU_001	IMU	Búsqueda bluetooth de dispositivos IMU	En la pantalla inicial, se oprime el botón cuyo icono es un pie	Sin una conexión establecida. Bluetooth activado	ALTA
TC_IMU_002	IMU	Búsqueda bluetooth de dispositivos IMU	En la pantalla inicial, se oprime el botón cuyo icono es un pie. Se repite el proceso con el otro pie	Sin una conexión establecida. Bluetooth activado	ALTA
TC_IMU_003	IMU	Establecer conexión a dispositivo IMU	Como resultado de una búsqueda de dispositivos, el sistema lista los hallados. Se pretende probar la conexión a un único dispositivo inercial	Sin una conexión establecida. Bluetooth activado	ALTA
TC_IMU_004	IMU	Establecer conexión a dispositivo IMU	Como resultado de una búsqueda de dispositivos, el sistema lista los hallados. Se pretende probar la conexión a múltiples dispositivos inerciales	Sin una conexión establecida. Bluetooth activado	ALTA
TC_IMU_006	IMU	Re-conexión a un dispositivo IMU	Luego de una interrupción involuntaria e inesperada, se desea recuperar la conexión con el dispositivo.	Conexión establecida	MEDIA

Tabla 3.1: Casos de prueba funcionales del sistema PARKIBIP, definidos por los campos identificador del caso, modulo del sistema, escenario de prueba, caso dentro del escenario, etapa, pre-condición y prioridad.

Id_caso_test	Módulo	Escenario	CASO	Pre-condición	Prioridad
TC_IMU_007	IMU	Verificar Productores de datos inerciales	Se desea probar la recuperación de datos de aceleración del sensor acelerómetro	Conexión establecida. Sensor acelerómetro configurado en la placa del IMU, con cierta frecuencia de salida de datos	ALTA
TC_IMU_008	IMU	Verificar Productores de datos inerciales	Se desea probar la recuperación de datos de velocidad angular del sensor giroscopio	Conexión establecida. Sensor giroscopio configurado en la placa del IMU, con cierta frecuencia de salida de datos y rango de valores establecido	ALTA
TC_IMU_009	IMU	Verificar Productores de datos inerciales	Se desea probar la recuperación de datos de fuerzas magnéticas del sensor magnetómetro	Conexión establecida. Sensor magnetómetro configurado en la placa del IMU, con cierta frecuencia de salida de datos	ALTA
TC_IMU_010	IMU	Verificar Productores de datos inerciales	Luego de recopilar información fragmentada, se desea integrar la recuperación de datos desde múltiples sensores inerciales	Conexión establecida. Sensor acelerómetro, magnetómetro, giroscopio configurados en la placa del IMU.	ALTA
TC_IMU_011	IMU	Motor de vibración	Vibración local al IMU	Conexión establecida. Motor de vibración establecido, mediante la potencia y tiempo de vibración.	ALTA
TC_QUAT_001	Quaternion	Chequear calibración inicial de quaternion	Estimación inicial de la Orientación en base a las primeras medidas. Se desea probar la correctitud numérica	Un conjunto de mediciones	ALTA
TC_QUAT_002	Quaternion	Validar conversión de representaciones	Conversión de un quaternion estimado a matriz de rotación. Se desea probar la correctitud numérica	Quaternion estimado	ALTA

Tabla 3.1: Casos de prueba funcionales del sistema PARKIBIP, definidos por los campos identificador del caso, modulo del sistema, escenario de prueba, caso dentro del escenario, etapa, pre-condición y prioridad.

Id_caso_test	Módulo	Escenario	CASO	Pre-condición	Prioridad
TC_QUAT_-003	Quaternion	Validar conversión de representaciones	Obtención del quaternion inverso o conjugado a el mismo. Se desea probar la correctitud numérica	Quaternion estimado	ALTA
TC_QUAT_-004	Quaternion	Validar conversión de representaciones	Transformación de quaternion a sus respectivos ángulos de euler. Se desea probar la correctitud numérica	Quaternion estimado	ALTA
TC_MEASU-RE_001	Medición	Validar conversión de grados a radianes	Conversión de las unidades grados a radianes. Útil para transformar las velocidades angulares según el proveedor del IMU. Se desea probar la correctitud numérica	Medición velocidad angular en grados	ALTA
TC_MEASU-RE_002	Medición	Verificar conversión a SI de unidades	Conversión de las unidades de las observaciones, a unidades estándar de PARKIBIP. Se desea probar la correctitud numérica	Mediciones aceleración, velocidad angular, fuerzas magnéticas tridimensionales.	ALTA
TC_CON-FIG_001	Configuration	Verificar modificación de parámetros	Dentro de la pantalla de Configuración, se pretende editar los valores de los parámetros, almacenarlos y luego recuperar el estado modificado.		MEDIA
TC_CON-FIG_002	Configuration	Verificar selección de algoritmo numérico	Dentro de la pantalla de Configuración, se pretende seleccionar dentro de un listado un método particular y luego que el sistema lo recupere para futuros procesamientos.		MEDIA
TC_CON-FIG_003	Configuration	Verificar recuperación de estado previo	Dentro de la pantalla de Configuración, se pretende editar los valores de los parámetros sin almacenarlos, luego deshacer los cambios y recuperar el estado previo.	Edición de parámetros en curso	BAJA
TC_CON-FIG_004	Configuration	Verificar recuperación de estado de fabrica	Dentro de la pantalla de Configuración, se pretende restablecer los valores por defecto.	Modificación previa de parámetros	BAJA

Tabla 3.1: Casos de prueba funcionales del sistema PARKIBIP, definidos por los campos identificador del caso, modulo del sistema, escenario de prueba, caso dentro del escenario, etapa, pre-condición y prioridad.

Id_caso_test	Módulo	Escenario	CASO	Pre-condición	Prioridad
TC_BIO-FEED-BACK_001	Biofeedback	Verificar ejecución de reglas de detección	Se desea validar la detección de fases de la marcha, en base a condiciones lógicas sobre los datos resultantes de los diversos algoritmos.	Conjunto de datos de ZVD. Conjunto de estados de Kalman. Datos inerciales.	ALTA
TC_BIO-FEED-BACK_002	Biofeedback	Verificar correctitud de parámetros espacio-temporales	Calculo de velocidad instantánea y media. Se desea probar la correctitud numérica	Conjunto de estados de Kalman	ALTA
TC_BIO-FEED-BACK_003	Biofeedback	Verificar correctitud de parámetros espacio-temporales	Calculo de cadencia. Se desea probar la correctitud numérica	Numero de pasos. Duración.	ALTA
TC_BIO-FEED-BACK_004	Biofeedback	Verificar correctitud de parámetros espacio-temporales	Calculo de aceleración media. Se desea probar la correctitud numérica	Mediciones de aceleración	ALTA
TC_BIO-FEED-BACK_005	Biofeedback	Verificar correctitud de parámetros espacio-temporales	Calculo de numero de pasos. Se desea probar la correctitud numérica	Conjunto de valores de ZVD	ALTA
TC_BIO-FEED-BACK_006	Biofeedback	Verificar correctitud de parámetros espacio-temporales	Distancia recorrida. Se desea probar la correctitud numérica	Conjunto de estados de Kalman. Duración	ALTA
TC_FUSER_-001	DataFuser	Verificar el fusionado con fuentes de datos	A partir de 2 productores de datos configurados -acc,gyro-. se desea construir un único paquete de observaciones en un instante dado.	Vectores tridimensionales aceleración y velocidad angular, además del timestamp correspondiente	ALTA
TC_FUSER_-002	DataFuser	Verificar el fusionado con fuentes de datos	A partir de los productores de datos configurados -acelerómetro, giroscopio, magnetómetro, se desea construir un único paquete de observaciones en un instante dado. Se desea probar la correctitud numérica	Mediciones aceleración, velocidad angular, fuerzas magnéticas tridimensionales, además del timestamp correspondiente	ALTA
TC_ZVD_001	ZVD	Chequear despachos de eventos de ZVD	Probar el funcionamiento de la cola circular de despacho de eventos de ZVD	Recepción iterativa de resultados de ZVD	ALTA

Tabla 3.1: Casos de prueba funcionales del sistema PARKIBIP, definidos por los campos identificador del caso, modulo del sistema, escenario de prueba, caso dentro del escenario, etapa, pre-condición y prioridad.

Id_caso_test	Módulo	Escenario	CASO	Pre-condición	Prioridad
TC_ZVD_002	ZVD	Verificar Resultado del test estadístico	Chequeo de la correctitud numérica del algoritmo numérico MV	Conjunto de datos de aceleración. Configuraciones requeridas	ALTA
TC_ZVD_003	ZVD	Verificar Resultado del test estadístico	Chequeo de la correctitud numérica del algoritmo numérico GLRT	Conjunto de datos de aceleración y giroscopio Configuraciones requeridas	ALTA
TC_ZVD_004	ZVD	Validar valores estacionarios con datos Open Source de MV	Se desea probar el algoritmo numérico MV mediante datos de prueba reales y de acceso libre	Conjunto de datos inerciales Open Source	ALTA
TC_ZVD_005	ZVD	Validar valores estacionarios con datos Open Source de GLRT	Se desea probar el algoritmo numérico GLRT mediante datos de prueba reales y de acceso libre	Conjunto de datos inerciales Open Source	ALTA
TC_ZVD_006	ZVD	Validar valores estacionarios con Metawear de MV	Se desea probar el algoritmo numérico MV mediante datos recibidos por el IMU de PARKIBIP	Caso de test 10 y 15	ALTA
TC_ZVD_007	ZVD	Validar valores estacionarios con Metawear de GLRT	Se desea probar el algoritmo numérico GLRT mediante datos recibidos por el IMU de PARKIBIP	Caso de test 10 y 15	ALTA
TC_HANDLER_001	HandlerThread	Verificar envío de datos desde DataAnalyzer a KalmanThread	Verificación de el envío de datos de medición entre hilos distintos de ejecución. Desde un DataAnalyzer a un KalmanFilter	Medición en un DataAnalyzer	ALTA
TC_HANDLER_002	HandlerThread	Verificar envío de datos desde KalmanThread a DataAnalyzer	Verificación de el envío de datos de medición entre hilos distintos de ejecución. Desde un KalmanFilter a un DataAnalyzer	Estado de Kalman en KalmanThread	ALTA
TC_ORIENTATION_001	Orientation Filter	Verificar el computo de la Orientación del sensor	Se desea probar el filtro de orientación mediante datos recibidos por el IMU de PARKIBIP	Rotación gráfica en tiempo real	ALTA
TC_ACCELERATION_001	Acceleration	Verificar la extracción de la aceleración del usuario	Se desea probar el método de rotación al marco inercial y remoción de componentes gravitatorias.	Caso de test 12	ALTA

Tabla 3.1: Casos de prueba funcionales del sistema PARKIBIP, definidos por los campos identificador del caso, modulo del sistema, escenario de prueba, caso dentro del escenario, etapa, pre-condición y prioridad.

Id_caso_test	Módulo	Escenario	CASO	Pre-condición	Prioridad
TC_KAL-MAN_001	KalmanFilter	Verificar Calibración inicial de Filtro de Orientación para Kalman	Se desea probar la integración del algoritmo de calibración al filtro de Kalman	Caso de test 17	ALTA
TC_KAL-MAN_002	KalmanFilter	Verificar detecciones de cero velocidad (ZVD) para Kalman	Se desea probar la integración del algoritmo de cero velocidad MV al filtro de Kalman	Caso de test 40	ALTA
TC_KAL-MAN_003	KalmanFilter	Verificar detecciones de cero velocidad (ZVD) para Kalman	Se desea probar la integración del algoritmo de cero velocidad GLRT al filtro de Kalman	Caso de test 41	ALTA
TC_KAL-MAN_004	KalmanFilter	Verificar la actualización del Filtro de Orientación para Kalman	Se desea probar la integración del algoritmo filtro de orientación	Caso de test 44	ALTA
TC_KAL-MAN_005	KalmanFilter	Verificar la obtención de la aceleración del sensor para Kalman	Se desea probar la integración del método de obtención de la aceleración del usuario	Caso de test 45	ALTA
TC_KAL-MAN_006	KalmanFilter	Verificar Matriz de Covarianza P y Q	Chequeo de la correctitud numérica del calculo de las matrices de covarianza	Caso de test 15	ALTA
TC_KAL-MAN_007	KalmanFilter	Verificar Innovation Z	Chequeo de la correctitud numérica del calculo del vector de invocación.	Caso de test 15	ALTA
TC_KAL-MAN_008	KalmanFilter	Verificar algoritmo Predictor -ecuaciones mecánicas-	Chequeo de la correctitud numérica del computo de las ecuaciones mecánicas.	Caso de test 15	ALTA
TC_KAL-MAN_009	KalmanFilter	Verificar algoritmo Corrector sin Compensación de velocidad	Chequeo de la correctitud numérica del método Corrector del filtro de Kalman	Caso de test 15	ALTA
TC_KAL-MAN_010	KalmanFilter	Verificar Compensación de velocidad en Kalman Filter	Chequeo de la correctitud numérica del método Corrector del filtro de Kalman, aplicando la compensación de ZVD	Caso de test 15. Caso 40/41	ALTA
TC_KAL-MAN_011	KalmanFilter	Verificar estados de Kalman Filter con datos Open Source de marcha	Se desea validar el comportamiento de la rotación inercial bajo una gravedad negativa	Conjunto de datos inerciales Open Source	ALTA

Tabla 3.1: Casos de prueba funcionales del sistema PARKIBIP, definidos por los campos identificador del caso, modulo del sistema, escenario de prueba, caso dentro del escenario, etapa, pre-condición y prioridad.

Id_caso_test	Módulo	Escenario	CASO	Pre-condición	Prioridad
TC_KAL-MAN_012	KalmanFilter	Verificar estados de Kalman Filter con datos Open Source de marcha	Verificación de la correctitud numérica del método que calcula la velocidad promedio	Conjunto de datos inerciales Open Source	ALTA
TC_KAL-MAN_013	KalmanFilter	Verificar estados de Kalman Filter con datos Open Source de marcha	Verificación de la correctitud numérica del método que calcula la trayectoria	Conjunto de datos inerciales Open Source	ALTA
TC_KAL-MAN_015	KalmanFilter	Verificar estados de Kalman Filter con datos Open Source de marcha	Verificación de la correctitud numérica del método que calcula la velocidad instantánea	Conjunto de datos inerciales OpenSource	ALTA
TC_KAL-MAN_016	KalmanFilter	Verificar estados de Kalman Filter con datos AUP y metawear	Pruebas experimentales con el factor Beta = 0.1 recomendado por el creador del filtro de orientación	Conjunto de mediciones inerciales	ALTA
TC_KAL-MAN_017	KalmanFilter	Verificar estados de Kalman Filter con datos AUP y metawear	Pruebas experimentales con el factor Beta = 1,3,5,6		ALTA
TC_KAL-MAN_018	KalmanFilter	Verificar estados de Kalman Filter con datos AUP y metawear	Se desea validar el comportamiento de la rotación inercial bajo una gravedad negativa para el sensor de PARKIBIP	Quaternion estimado. Medición inercial aceleración	ALTA
TC_KAL-MAN_019	KalmanFilter	Verificar estados de Kalman Filter con datos AUP y metawear	Se desea validar el comportamiento de la rotación inercial bajo una gravedad positiva para el sensor de PARKIBIP	Quaternion estimado. Medición inercial aceleración	ALTA
TC_KAL-MAN_020	KalmanFilter	Verificar estados de Kalman Filter con datos AUP y metawear	Verificación de la correctitud numérica del computo de los vectores de estados (posición, velocidad, orientación), resultados del filtro de Kalman	Vector de estado Kalman, ZVD y mediciones asociadas.	ALTA
TC_KAL-MAN_021	KalmanFilter	Verificar estados de Kalman Filter con datos AUP y metawear	Verificación de la correctitud numérica del método que aproxima la trayectoria en base a la posición y duración de la sesión de rehabilitación, empleando el sensor IMU de PARKIBIP.	Vector de estado Kalman y datos de sesión activa	ALTA

Tabla 3.1: Casos de prueba funcionales del sistema PARKIBIP, definidos por los campos identificador del caso, modulo del sistema, escenario de prueba, caso dentro del escenario, etapa, pre-condición y prioridad.

Id_caso_test	Módulo	Escenario	CASO	Pre-condición	Prioridad
TC_KAL-MAN_023	KalmanFilter	Verificar estados de Kalman Filter con datos 2 dispositivos	Integración de Kalman Filter para aproximar la trayectoria en base a la posición y duración de la sesión de rehabilitación, combinando ambos dispositivos IMU de PARKIBIP.	Vector de estado Kalman y datos de sesión activa	ALTA
TC_KAL-MAN_024	KalmanFilter	Verificar estados de Kalman Filter con datos 2 dispositivos	Integración de Kalman Filter para aproximar la velocidad promedio en base a la instantánea de la sesión de rehabilitación, combinando ambos dispositivos IMU de PARKIBIP.	Vector de estado Kalman y datos de sesión activa	ALTA
TC_KAL-MAN_025	KalmanFilter	Verificar estados de Kalman Filter con datos 2 dispositivos	Integración de Kalman Filter para aproximar la velocidad promedio en base a la instantánea de la sesión de rehabilitación, empleando ambos dispositivos IMU de PARKIBIP.	Vector de estado Kalman y datos de sesión activa	ALTA
TC_POOL_-001	PoolStorage	Chequear almacenamiento de datos	Se desea verificar el almacenamiento adecuado de mediciones bajo el soporte de un servicio recursivo de fondo. Su procesamiento es en batch	Medición inercial. Servicio de fondo iniciado	ALTA
TC_POOL_-002	PoolStorage	Chequear almacenamiento de datos	Se desea verificar el almacenamiento adecuado de cero velocidades, bajo el soporte de un servicio recursivo de fondo. Su procesamiento es en batch	Medición inercial. Servicio de fondo iniciado	ALTA
TC_POOL_-003	PoolStorage	Chequear almacenamiento de datos	Se desea verificar el almacenamiento adecuado de los vectores de estado de Kalman, bajo el soporte de un servicio recursivo de fondo. Su procesamiento es en batch	Medición inercial. Servicio de fondo iniciado	ALTA

Apéndice 4

Pruebas Exploratorias: Tabla de sesiones exploratorias

A continuación, se muestra un resumen de las sesiones exploratorias ejecutadas en PARKIBIP. Se pueden ver también los diferentes bugs encontrados a partir de estas sesiones.

Tabla 4.1: Testing exploratorio basado en sesiones en PARKIBIP. Para cada sesión -TE_SXXX- se registraron los campos: identificador, misión, áreas, duración de la sesión, notas de prueba, análisis y reporte de errores.

Sesión	Misión	Áreas	Duración	Notas de prueba	Análisis y reporte de error (Bug)
TE_S001	Validar que un usuario registrado pueda ingresar al sistema y se interpreten los datos del token <i>JWT</i>	OS — OnePlus 5t API 28. Funcionalidad — Login	Corta	Se ingresó correctamente a PARKIBIP con el usuario <i>chuerta</i> .	Bug.001. Al descifrar el token, no se encontraron los meta-datos del usuario (e.g. nivel de usuario)
TE_S002	Usabilidad: Verificar validación dinámica de campos en Login.	OS — OnePlus 5t API 28. Funcionalidad — Login	Corta	En la pantalla de inicio de sesión, se digitó el usuario y la contraseña	Bug.001: El mensaje de error ocupa todo el <i>layout</i> y posee fondo gris el cual no sigue el estilo de la aplicación.
TE_S003	Verificar la detección de velocidades cero con MV y el sensor quieto	OS — Nexus 5X API 24. Funcionalidad — ZVD	Larga	Se inicia la recolección de datos con el sensor estacionario.	Bug.001: Al principio los resultados fueron los esperados -true-. Luego de cierto tiempo, comenzó con detecciones periódicas.
TE_S004	Verificar la detección de velocidades cero con MV y el sensor quieto	OS — Nexus 5X API 24. Funcionalidad — ZVD	Larga	Se inicia la recolección de datos con el sensor estacionario. El detector no identificó movimiento	

Tabla 4.1: Testing exploratorio basado en sesiones en PARKIBIP. Para cada sesión -TE_SXXX- se registraron los campos: identificador, misión, áreas, duración de la sesión, notas de prueba, análisis y reporte de errores.

Sesión	Misión	Áreas	Duración	Notas de prueba	Análisis y reporte de error (Bug)
TE_S005	Verificar la detección de velocidades cero con MV	OS — Nexus 5X API 24. Funcionalidad — ZVD	Larga	Se inicia la recolección de datos y se realiza una caminata.	Bug_001: No detecta eventos estacionarios. Bug_002: Comportamiento inadecuado de la cola circular. Bug_003: Luego de cierto tiempo la aplicación se cae.
TE_S006	Verificar la detección de velocidades cero con GLRT y el sensor estacionario	OS — Nexus 5X API 24. Funcionalidad — ZVD	Larga	Se inicia la recolección de datos con el sensor estacionario.	Bug_001: Alternancia de resultados. Enseguida se cae la aplicación.
TE_S007	Verificar la detección de velocidades cero con GLRT	OS — Nexus 5X API 24. Funcionalidad — ZVD	Larga	Se inicia la recolección de datos y se realiza una caminata.	Bug_001: Detección continua de eventos de velocidad cero, afinar umbral estadístico Bug_002: Raro. La condición de tamaño de la ventana de medición se supera y crece exponencialmente
TE_S008	Calibración de Orientación	OS — Oneplus 5t API 28. Funcionalidad — Orientation	Larga	Se inicia la recolección de datos y se llama a calibrar el sensor. El IMU en estado estacionario	Bug_001: Mantiene el quaternion inicial. La calibración no se esta ejecutando, por ende no lleva al sistema de referencias esperado.
TE_S009	Verificar quaternion calculado por el filtro de Orientación con datos públicos	OS — Oneplus 5t API 28. Funcionalidad — Orientation	Media	Se inicia la recolección de datos y se ejecuta la actualización del filtro de orientación. Los valores son tabulados y comparados con el resultado esperado. Sin inconvenientes	
TE_S010	Chequear la consolidación de datos a distintas frecuencias	OS — Oneplus 5t API 28. Funcionalidad — Data Fuser	Media	Se inician los productos acelerómetro, giroscopio, magnetómetro. Luego se escribe en consola el resultado de la fusión de datos	Bug_001: Se pierden datos de magnetómetro -frecuencia mas baja?-.
TE_S011	Revisión de usabilidad: Usuarios	OS — Oneplus 5t API 28. Funcionalidad — Usuarios	Corta	Se accede a la pantalla de Pacientes.	Bug_001: El buscador y/filtrado no funciona

Tabla 4.1: Testing exploratorio basado en sesiones en PARKIBIP. Para cada sesión -TE_SXXX- se registraron los campos: identificador, misión, áreas, duración de la sesión, notas de prueba, análisis y reporte de errores.

Sesión	Misión	Áreas	Duración	Notas de prueba	Análisis y reporte de error (Bug)
TE_S012	Revisión de usabilidad: Inicio	OS — Oneplus 5t API 28. Funcionalidad — Inicio	Corta	Se accede a la pantalla de Pacientes y se configuran los dispositivos IMU.	Bug-001: No se establecen los colores adecuados en los botones, cuyo icono es un pie. Quedan en grises.
TE_S013	Revisión de Usabilidad: Configuraciones	OS — Oneplus 5t API 28. Funcionalidad — Configuración	Media	Se accedió a la sección de configuraciones en búsqueda de defectos de pantalla y estilos.	Bug-001: El sistema no permitía cambiar la selección de un algoritmo Bug-002: El texto de la unidad del parámetro frecuencia de la aceleración, superpone el campo de edición. Bug-003: Los márgenes laterales de las tarjetas de configuración son inadecuados
TE_S014	Modificación de Configuraciones	OS — Oneplus 5t API 28 Funcionalidad — Configuración	Corta	Se accedió a la sección de configuraciones en búsqueda de inconvenientes en la modificación de parámetros. Se editaron parámetros y luego se guardaron. Seguido fueron restablecidos a los valores por defecto.	Bug-001: No todos los parámetros se guardaron adecuadamente
TE_S015	Deshacer cambios en curso de Configuraciones	OS — Oneplus 5t API 28. Funcionalidad — Configuración	Corta	Se accedió a la sección de configuraciones. Se editaron parámetros y luego se presiono el botón deshacer cambios	
TE_S016	Detener una sesión activa PARKIBIP	OS — Nexus 5X API 24. Funcionalidad — Sesión	Larga	Se inicio una sesión de rehabilitación, luego de cierto tiempo se finalizo. El sistema tendría que acceder al resumen de la sesión, pero no redirección a dicha pantalla	Bug-001: La sesión no se guarda correctamente. Luego el resumen no tiene sesión. Bug-002: Se detectan muchos datos -1714- en el Pool del servicio de fondo que almacena los datos. Problema de concurrencia

Tabla 4.1: Testing exploratorio basado en sesiones en PARKIBIP. Para cada sesión -TE_SXXX- se registraron los campos: identificador, misión, áreas, duración de la sesión, notas de prueba, análisis y reporte de errores.

Sesión	Misión	Áreas	Duración	Notas de prueba	Análisis y reporte de error (Bug)
TE_S017	Validar el calculo de la aceleración lineal del Usuario	OS — Nexus 5X API 24. Funcionalidad — Aceleración lineal	Larga	Se detecta un comportamiento extraño e inadecuado. Se grafican los datos de aceleración y la misma incrementa de forma lineal, luego baja a 0. Se repite el comportamiento	Bug_001: Aceleración instantánea del usuario incorrecta. Sugiere defecto en la rotación del vector, y/o en la eliminación de la gravedad , y/o en el computo del filtro de orientación
TE_S018	Validar la rotación del vector aceleración. Problema aceleración lineal	OS — Nexus 5X API 24. Funcionalidad — Aceleración lineal	Larga	Se cargan datos de prueba con el fin de probar la rotación del vector aceleración. Se tabulan en Excel los vectores resultantes de las rotaciones y los sensados.	Bug_001: Defecto en la rotación del vector. Se debe probar la gravedad y la orientación del filtro
TE_S019	Validar la rotación del vector aceleración. Problema aceleración lineal	OS — Oneplus 5t API 28. Funcionalidad — Aceleración lineal	Larga	Se cargan datos de prueba con el fin de probar la rotación del vector aceleración. Se chequea la gravedad negativa y positiva para el sensor, luego se tabulan en Excel los vectores resultantes de las rotaciones y los sensados.	Bug_001: Continua el error en la rotación del vector.
TE_S020	Validar la rotación del vector aceleración. Problema aceleración lineal	OS — Oneplus 5t API 28. Funcionalidad — Aceleración lineal	Larga	Se cargan datos de prueba con el fin de probar la rotación del vector aceleración. Se chequea el quaternion obtenido por Madwick para el sensor de PARKIBIP, luego se tabulan en Excel los vectores resultantes de las rotaciones y los sensados.	Bug_001: Por alguna razón, el filtro de Orientación computa mal el quaternion, con el sensor de PARKIBIP. Luego la rotación se hace mal.
TE_S021	Verificar quaternion calculado por el filtro de Orientación con Datos de PARKIBIP	OS — Windows 10 Funcionalidad — Orientación	Larga	Se diseña una aplicación gráfica en el lenguaje c#, con un cubo en 3D. Se conectan los IMU y se prueban las rotaciones en tiempo real. Se prueban los valores de frecuencias altas y bajas (20-200Hz)	Bug_001: El cubo no rota adecuadamente, se tranca.

Tabla 4.1: Testing exploratorio basado en sesiones en PARKIBIP. Para cada sesión -TE_SXXX- se registraron los campos: identificador, misión, áreas, duración de la sesión, notas de prueba, análisis y reporte de errores.

Sesión	Misión	Áreas	Duración	Notas de prueba	Análisis y reporte de error (Bug)
TE_S022	Verificar quaternion calculado por el filtro de Orientación con Datos de PARKIBIP	OS — Windows 10 Funcionalidad — Orientación	Larga	Se diseña una aplicación gráfica en el lenguaje c#, con un cubo en 3D. Se conectan los IMU y se prueban las rotaciones en tiempo real. Se prueba con el valor de convergencia BETA=0.1, con una frecuencia de 100HZ	Bug-001: El cubo no rota adecuadamente, se tranca.
TE_S023	Verificar quaternion calculado por el filtro de Orientación con Datos de PARKIBIP	OS — Windows 10 Funcionalidad — Orientación	Larga	Se diseña una aplicación gráfica en el lenguaje c#, con un cubo en 3D. Se conectan los IMU y se prueban las rotaciones en tiempo real. Se prueba con el valor de convergencia BETA=0.5,1,3,5,6, con una frecuencia de 100HZ. Se logran buenos resultados con Beta igual a 5.	
TE_S024	Revisión de Usabilidad: Sesión activa	OS — OnePlus 5t API 28 Funcionalidad — Sesión	Larga	Se inicia una sesión de rehabilitación y se observan los resultados	Bug-001: La gráfica en tiempo real no despliega valores. Bug-002: Su vista es demasiado grande y los colores no se corresponden con los de la aplicación
TE_S025	Revisión de Usabilidad: Sesión activa	OS — OnePlus 5t API 28. Funcionalidad — Sesión	Larga	Se inicia una sesión de rehabilitación y se observan los resultados	Bug-001: Actualizar la UI por cada medición mata el UI thread
TE_S026	Revisión de Usabilidad: Sesión activa con actualización cada cierta cantidad de datos	OS — OnePlus 5t API 28. Funcionalidad — Sesión	Media	Se inicia una sesión de rehabilitación y se observan los resultados	
TE_S027	Verificar funcionamiento de Kalman con datos de prueba	OS — Nexus 5X API 24. Funcionalidad — Kalman	Larga	Se levantan los datos en memoria y se itera sobre las mediciones. Se gráfica la velocidad. Problema raro	Bug-001: Valores inadecuados de velocidad instantánea, conllevan a valores de velocidad promedio altos -9.5-

Tabla 4.1: Testing exploratorio basado en sesiones en PARKIBIP. Para cada sesión -TE_SXXX- se registraron los campos: identificador, misión, áreas, duración de la sesión, notas de prueba, análisis y reporte de errores.

Sesión	Misión	Áreas	Duración	Notas de prueba	Análisis y reporte de error (Bug)
TE_S028	Verificar funcionamiento de Kalman con datos de prueba. Verificación analítica de vectores y matrices	OS — Nexus 5X API 24. Funcionalidad — Kalman	Larga	Se levantan los datos en memoria y se itera sobre las mediciones. Se chequean las primeras iteraciones para matrices y vectores de forma analítica. Se compara con el resultado del filtro.	Bug_001: Valores inadecuados de velocidad instantánea, conllevan a valores de velocidad promedio altos -9.5- Bug_002: No se está compensando la orientación
TE_S029	Verificar funcionamiento de Kalman con datos de prueba.	OS — OnePlus 5t API 28 Funcionalidad — Kalman	Larga	Se levantan los datos en memoria y se itera sobre las mediciones. Se chequean las primeras iteraciones. Sin embargo continua el problema al graficar la velocidad.	Bug_001: Valores inadecuados de velocidad instantánea, conllevan a valores de velocidad promedio altos -9.5-
TE_S030	Verificar funcionamiento de Kalman con datos de prueba.	OS — OnePlus 5t API 28 Funcionalidad — Kalman	Larga	Se levantan los datos en memoria y se itera sobre las mediciones. Se chequean las primeras iteraciones.	Bug_001: El método de orientación requiere radianes para la velocidad angular. Lo aplica sin tener en cuenta la unidad que recibe. Chequear unidades antes.
TE_S031	Verificar funcionamiento de Kalman con datos de prueba.	OS — Nexus 5X API 24. Funcionalidad — Kalman	Corta	Se levantan los datos en memoria y se itera sobre las mediciones. Se chequean las primeras iteraciones.	
TE_S032	Revisión de Usabilidad: Resumen de sesión	OS — OnePlus 5t API 28 Funcionalidad — Resumen de sesión	Media	Se selecciona una sesión existente en el sistema para visualizar el resumen.	Bug_001: No se grafica adecuadamente en el resumen general. Bug_002: Los ejes de la gráfica no están alineados Bug_003: El título de la vista es incorrecto. Debería decir General Bug_004: El azul de la barra, no es el azul de PARKIBIP.

Apéndice 5

Congreso de Ingeniería Biomédica SABI 2020

PARKIBIP: IMU-Based Feedback App for Parkinson Disease Gait Rehabilitation

Carlos Huerta¹, Samuel Sainz¹, Macarena Vergara¹, Dario Santos¹, Franco Simini¹

¹NIB - Núcleo de Ingeniería Biomédica, Universidad de la República. Montevideo, Uruguay

Abstract— Gait disorders is the primary symptom of a Parkinson's patient causing loss of independence in affected subjects, due to falls, tremors and rigidity in movement. This study aims to implement PARKIBIP - a biofeedback system - that allows the patient a personal rehabilitation aimed at prolonging the work of the physiotherapist in their daily lives. Therefore, a systematic review based on scientific evidence was carried out, which identifies all the relevant material regarding the use of IMU sensors to analyze the progress of people. According to the synthesis of the results, PARKIBIP system was implemented. Finally, the main results of the implemented system are presented, starting with its components design and initials tests performed.

Keywords— Gait analysis, Inertial measurement unit, Wearables sensors, Biofeedback, Parkinson.

Resumen— La alteración de la marcha es el síntoma primario de un enfermo de Parkinson causando la pérdida de independencia en los sujetos afectados, a raíz de caídas, temblores y la rigidez en el movimiento. El presente estudio tiene como objetivo la implementación de PARKIBIP -un sistema de biorretroalimentación-, que le permita al enfermo una rehabilitación personal orientada a prolongar el trabajo del fisioterapeuta en su vida cotidiana. Por lo tanto, se realizó una revisión sistemática basada en evidencia científica, que identifica todo el material relevante respecto al uso de sensores IMU para analizar la marcha de las personas. Conforme a la síntesis de resultados arrojados, se procedió a implementar el sistema PARKIBIP. Finalmente, se presentan los principales resultados del sistema implementado, comenzando por sus componentes de diseño y las primeras pruebas efectuadas.

Palabras clave— Análisis de la marcha, Unidad de medición inercial, Sensores wereables, Biorretroalimentación, Parkinson.

I. INTRODUCCIÓN

I-A. Antecedentes patológicos y características de la patología

LA enfermedad de Parkinson (EP) es un trastorno neurodegenerativo multisistémico que afecta al sistema nervioso central provocando la aparición de síntomas motores y no motores [1]. Es una enfermedad crónica -es decir, que persiste durante un extenso período de tiempo- y progresiva, lo que significa que sus síntomas empeoran con el tiempo. La alteración de la marcha es el síntoma primario del trastorno [2], y junto con las caídas causa pérdida de independencia en los sujetos afectados [3]. Se caracteriza por deficiencias motoras, que incluyen temblor en las extremidades, aumento de la rigidez de las mismas, alteraciones de los parámetros espacio-temporales y fases de la marcha (p. ej. disminución de la longitud del paso y la velocidad de la marcha, variabilidad de zancada a zancada, fases de marcha) [4]–[7]. Los neurólogos solo pueden confiar en observaciones cualitativas en sesiones esporádicas y basadas en su experiencia. De este modo, contar con una evaluación clínica precisa de la gravedad de estos síntomas y un seguimiento continuo, es fundamental para identificar una terapia efectiva.

I-B. Antecedentes en el área de estudio

Los avances en los dispositivos de monitoreo vestibles -del inglés, Wearable- están cambiando al sector médico desde un enfoque tradicional reactivo -atención de la salud en el nivel de crisis- a un enfoque de gestión proactiva de la salud [8]–[11]; que permite posponer, descubrir y abordar dichos problemas en una etapa temprana [12]. La implementación de sistemas basados en esta tecnología innovadora, apuntan a lograr una función terapéutica, tratando las deficiencias motoras a través de la biorretroalimentación [13] [14]–[17]. Esta técnica, se emplea para supervisar las funciones fisiológicas del organismo mediante la utilización de un sistema de retroalimentación, que informa al usuario del estado de la función que se desea controlar.

I-C. Identificación del problema de investigación

En el contexto de la salud, principalmente en la EP la retroalimentación activa (de largo plazo en los pacientes durante su vida) es fundamental, y se refiere a la provisión de estímulos externos durante o inmediatamente después de un acto motor (p. ej. cuando el sujeto levanta el pie del suelo) que brinden soporte al usuario, estimulen el compromiso cognitivo de los sujetos y potencien los efectos del ejercicio motor [17]–[20]. La marcha de las personas es un proceso complejo y cíclico que requiere la sinergia de los músculos, los huesos y el sistema nervioso [21], principalmente dirigido a mantener la

posición vertical y mantener el equilibrio durante condiciones estáticas y dinámicas [22]. Asimismo, implica la medición, la descripción y la evaluación de parámetros de marcha (p. ej. velocidad de la marcha, la amplitud de la zancada, la cadencia) que caracterizan la locomoción humana [23].

I-D. Propuesta del trabajo

El presente estudio tiene como objetivo la implementación de PARKIBIP, un sistema que le permita a los pacientes con la enfermedad de Parkinson una rehabilitación personal. Durante las sesiones de fisioterapia los estímulos externos pueden mejorar las características de la marcha y PARKIBIP entonces busca emular estos estímulos en tiempo real para permitirle al enfermo una rehabilitación personal que prolongue el trabajo del fisioterapeuta en su vida cotidiana. PARKIBIP es el resultado del trabajo de un equipo interdisciplinario a partir de la experiencia de fisioterapeutas que definen las reglas para estimular a los pacientes durante la marcha. Mediante un dispositivo electrónico vestible de bajo costo (IMU - Unidad de Medición Inercial), que combina múltiples sensores (p. ej. acelerómetro, giroscopio, magnetómetro, entre otros), se particiona el ciclo de marcha del usuario permitiendo identificar distintos eventos, tales como, el contacto inicial con el suelo o la fase de vuelo. Finalmente, con los eventos detectados, se aplican ciertos algoritmos matemáticos que calculan los parámetros espacio temporales claves, y en base a un módulo clínico se estimula adecuadamente al usuario.

II. MATERIALES Y MÉTODOS

En los últimos años, la partición del ciclo de la marcha se ha vuelto un tema de investigación desafiante debido a su impacto en distintas aplicaciones relacionadas a la tecnología del movimiento de las personas [24]. Una variedad de sensores puede ser usados para alimentar algoritmos que detectan las distintas fases del paso, los mismos se clasifican principalmente como “vestibles” o “no-vestibles”. Dentro de los sensores vestibles, las plantillas inteligentes con sensores de presión son consideradas un estándar de oro; sin embargo, para superar algunas limitaciones inherentes en éstas, los dispositivos conocidos como Unidades de Medición Inercial se han vuelto muy populares en los últimos años. Por otro lado, los sensores no-vestibles como los sistemas óptico-electrónicos o plataformas de fuerza, siguen siendo los sistemas más precisos para analizar la marcha de personas en un ambiente interior [25].

II-A. Elección del sensor

Para seleccionar el tipo de sensor, se consideró una revisión sistemática realizada en el año 2016 [25], la cual identifica, selecciona y categoriza las distintas tecnologías para realizar la detección de las fases del paso, analizando ventajas y desventajas de cada solución. De esta manera, la selección del tipo de sensor se realizó teniendo en cuenta características relevantes para el proyecto en cuestión, entre ellas, el costo, la accesibilidad, la efectividad y la portabilidad del sensor.

Bajo los criterios mencionados, el tipo de dispositivo elegido que mejor se adecúa al proyecto fue un sensor IMU (del inglés Inertial Measurement Unit o Unidad de Medición Inercial). Estos dispositivos portables, de bajo costo y larga autonomía,

han sido utilizados con buenos resultados en el área de análisis de movimientos dinámicos lineales y angulares en los últimos años. Se componen esencialmente de los sensores acelerómetro, giroscopio y magnetómetro, y en ciertos casos se extienden a incluir barómetro, sensores de luz, entre otros. Existen en el mercado decenas de empresas que ofrecen estos dispositivos junto a sus distintas variaciones, por lo que surgió la necesidad de hacer una evaluación de mercado y elegir la más apropiada. La Tabla comparativa I permite visualizar las principales características evaluadas:

II-B. Revisión sistemática basada en evidencias

En este proyecto se realizó una revisión sistemática basada en evidencia científica relacionada al uso de sensores IMU para analizar la marcha de las personas al caminar. Conforme a los requerimientos del proyecto, el objetivo de la revisión sistemática se centró en identificar todo el material relevante que permita estudiar la marcha de las personas con el uso de sensores, reducir o eliminar incertidumbres en la investigación, y sobre todo no reinventar la rueda. El conocimiento recopilado fue fundamental para la implementación de la solución PARKIBIP, para mejorar la marcha de los pacientes con la enfermedad de Parkinson.

Para establecer el objetivo de la revisión sistemática, se plantean cuatro preguntas de investigación que guían la revisión:

- ¿Qué tipo de sensores IMU se utilizan para medir la marcha de las personas?
- ¿Cuántos sensores son necesarios para medir las características de la marcha?
- ¿Qué datos sobre la marcha se pueden obtener con los distintos sensores?
- ¿Qué técnicas o algoritmos se utilizan para analizar los datos?

II-B1. Estrategia de búsqueda: Se planeó una estrategia de búsqueda automatizada, en función de la definición de una cadena de búsqueda que incluya la combinación de las palabras claves del problema de estudio (p. ej. gait analysis, etc.). Para ello, se seleccionó el servicio de indexamiento y librería digital Scopus, como motor de búsqueda principal.

II-B2. Criterios de inclusión/exclusión: Se formularon criterios de selección de artículos a incluir en la revisión, con el fin de identificar aquellos estudios que proporcionen información relevante a las preguntas de investigación. Dichos criterios fueron expresados como dos conjuntos, criterios de inclusión y de exclusión, ilustrados en la Tabla II.

II-B3. Selección: Los resultados primarios obtenidos fueron evaluados utilizando el título y el resumen del estudio. Posteriormente, se aplicaron los criterios de selección establecidos iniciando la Selección de estudios relevantes para el proyecto.

El proceso de Selección se realizó en varias etapas, partiendo de 165 artículos. En una primera etapa, leyendo únicamente el título y el resumen de cada artículo se realizó una primera selección, dejando unos 85 artículos.

Luego, leyendo nuevamente el título y resumen de cada artículo, se puntuaron del 1 al 3 el valor de los artículos, siendo los de categoría 1 aquellos que en primera instancia parecían ser muy útiles y categoría 3 los que si bien aportaban

Tabla I: Comparación de dispositivos IMU

	Arion	BTS@G-Sensor	EXL S3	InvenSense	Isen	mbientLab Meta-MotionR	Opal, APDM Inc	RehaGait (HA-SO-MED)	SennoGait SmartInsolesPRO	Shimmer Sensor	Waveshare	Xsens
Acelerómetro (3-axis)	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI
Giroscopio (3-axis)	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI
Magnetómetro (3-axis)	NO	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI
Barómetro	SI	NO	NO	NO	SI	SI	NO	-	NO	NO	SI	NO
Altímetro	-	-	-	NO	NO	SI	-	-	NO	SI	NO	NO
Bluetooth	SI	SI (3.0)	SI (2.1)	NO (I2C, SPI)	NO	SI (low energy)	NO	SI	SI	SI	NO	-
Wifi	NO	NO	NO	NO	SI	NO	NO	NO	SI	NO	NO	NO
Tiempo Real	SI	SI	SI	SI	SI	SI	<30ms	-	-	-	-	<2 ms
Runtime calibration	NO	NO	NO	SI	SI	SI	-	-	-	SI	-	-
Peso	-	37gr	22g	-	46g	8.5 g	<25g	-	-	23.6g	3g	-
Medidas(mm)	-	70x40x18	54x33x14	3x3x0.9	56x38x18	24x17x4	43x39x13	60x15x35	35-45	51x34x14	31x16x2.5	57
Batería	~7hs	~8hs	2 hs	-	~3.5 hs	2-14d	8-16hrs	-	>48hrs	-	-	-
API en android	NO	NO	NO	SI	SI	SI	No	NO	NO	SI	-	SI
API en iOS	NO	NO	NO	NO	NO	SI	No	NO	NO	NO	-	NO
Comunidad/Open Sourcing	NO	NO	NO	SI	NO	SI	NO	NO	NO	NO	-	NO
Documentación	NO	NO	NO	SI	NO	SI	NO	NO	NO	SI	-	SI
Vibración	NO	NO	NO	NO	NO	SI	NO	NO	NO	NO	-	NO
Precio	€189	-	-	-	-	USD99	-	-	-	€359	USD16	€800

Tabla II: Criterios de selección de estudios primarios

Criterios de inclusión	Criterios de exclusión
Estudios completos	Estudios arbitrarios -no científicos- o con el formato incorrecto
Con acceso al artículo completo	No traten la marcha de personas
Escritos en los idiomas inglés y español	Usen sensores que no sean IMUs o difíciles de acceder
Publicados a partir del año 2015	Se descartan estudios duplicados
	Se enfoquen en la posición en la que se encuentra el sujeto y no en los parámetros de la marcha

valor no parecían ser tan útiles. Para muchos artículos fue difícil establecer su categoría leyendo únicamente el título y su resumen, por lo que en una tercera etapa se leyeron cerca de 40 artículos para establecer su categoría (principalmente las secciones de objetivos del artículo, metodología y resultados). Ambos miembros del equipo realizaron esta tarea para revisar y justificar los desacuerdos, de esta forma se alinearon criterios.

El resultado de la fase de selección abarca:

- 33 artículos en categoría 1
- 22 artículos en categoría 2
- 30 artículos en categoría 3

II-B4. Extracción: El objetivo en esta etapa del proceso de revisión, es la extracción del contenido necesario para direccionar las preguntas de investigación. Para ello, los estudios resultados del proceso de selección, fueron incluidos a la herramienta Mendeley para un mejor seguimiento. Conforme a mantener consistencia a través de los estudios y dentro del equipo de revisores, se utilizó una estrategia de extracción en base a formularios preestablecidos. De esta manera, se realizó una extracción de datos basada en temas principales: (i) Objetivo del estudio; (ii) Características de los sensores

utilizados; (iii) Cantidad de sensores necesarios; (iv) Ubicación de los sensores; (v) Fases del paso analizadas; (vi) Algoritmos utilizados; (vii) Características de los sujetos -sanos o con alguna enfermedad-; (viii) Resultados y efectividad.

II-C. Implementación PARKIBIP

PARKIBIP se basa en el computo en tiempo real de los parámetros espacio temporales de la marcha de personas, y en base a reglas clínicas de decisión, procura estimular al paciente -en forma de vibración- para permitirle una rehabilitación personal que contribuya al trabajo del fisioterapeuta. Por lo tanto, para ejecutar un patrón de marcha más efectivo y estimular el proceso de aprendizaje motor se requiere identificar el ciclo de la marcha. Estandarizando los conceptos de fases de la marcha, tal como se sintetiza en [25], se pueden emplear distintas granularidades de las fases de la marcha (i.e. dos, tres, cuatro, cinco, seis, siete y ocho), tal como indica la Fig. 1.

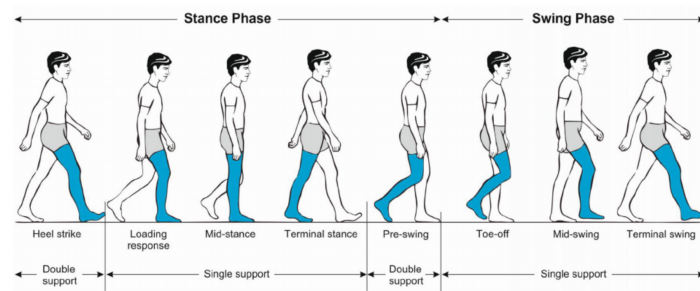


Figura 1: Partición del ciclo de la marcha normal en 8 fases. Tomado de “W. Pirker y R. Katzenschlager” [26].

El presente estudio reconoce dos fases en el ciclo de la marcha, fase de apoyo y fase vuelo. En este sentido, los

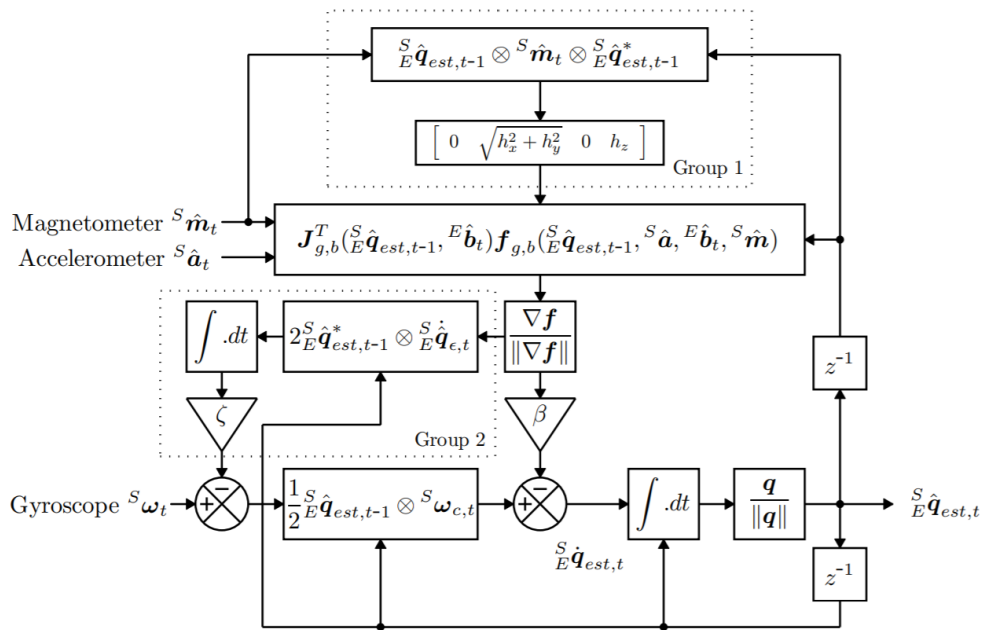


Figura 2: Filtro de orientación con distorsión magnética. Extraído de S. Madgwick [27].

algoritmos matemáticos considerados se enfocan en detectar eficazmente los eventos golpe de balón y el despegue de los dedos del pie.

A efecto de la etapa II-A -elección del sensor IMU-, se concluyó en la utilización de la solución de producto de dispositivos vestibles MetaMotionR. Además de ser un dispositivo liviano, accesible y de bajo costo; integra los principales sensores de lectura, como lo son el acelerómetro (BMI160 6-Ejes), el giroscopio (BMI160 6-Ejes), el magnetómetro y la unidad vibratoria. Se emplearon dos dispositivos sincronizados unidos con bandas elásticas de velcro a los tobillos -justo por encima de los maléolos- de los pacientes con la enfermedad de Parkinson tal como indica la Fig. 3.



Figura 3: Banda elástica junto a la unidad de medición inercial (IMU)

Se implementó una aplicación móvil Android que se interconecta a dos dispositivos IMU a través del protocolo estándar Bluetooth. El sistema se encuentra diseñado tal que habilita a efectuar las siguientes actividades:

1. Escanear dispositivos IMUs y enlazarlos a la aplicación móvil para ambos miembros inferiores
2. Calibrar los sensores de los IMUs

3. Recuperar los datos acelerómetro, giroscopio y magnetómetro de los respectivos sensores a 50Hz
4. Estimar orientación del dispositivo respecto a una posición inercial
5. Identificar fase de la marcha en cada momento y estimar parámetros de la marcha
6. Estimular al usuario de acuerdo a protocolo

La estimación precisa de la orientación del cuerpo -asociado al dispositivo- desempeña un rol fundamental para el cálculo de parámetros espacio temporales. Si bien existen diversos algoritmos que logran buenas aproximaciones, la principal limitante es el error propagado a algoritmos al estimar la orientación. Un acelerómetro y un magnetómetro medirán los campos gravitacionales y magnéticos de la tierra, y proporcionarán una referencia absoluta de orientación. Sin embargo, es probable que estén sujetos a altos niveles de ruido (i.e. las aceleraciones debidas al movimiento corromperán la dirección de gravedad medida). La función de un filtro de orientación es calcular una única estimación de orientación a través de la fusión óptima del giroscopio, acelerómetro y magnetómetro. Para estimar la orientación de los dispositivos, aplicamos la técnica Madgwick a los datos recopilados, descrita en la FIG. 2 tomada de [27].

PARKIBIP procura estimar durante cada sesión los parámetros longitud del paso, cadencia, distancia recorrida, velocidad de la marcha y tiempo de apoyo por pie; sobre un grupo de Parkinsonianos con marcha independiente.

III. RESULTADOS

Con el objetivo de enfrentar desafíos clínicos y tecnológicos específicos, PARKIBIP fue diseñado en capas, de forma tal que se cumplan los siguientes puntos:

1. Interopere con dispositivos IMU independientemente del fabricante de los mismos
2. Calculo automático y en tiempo real de las fases de la marcha

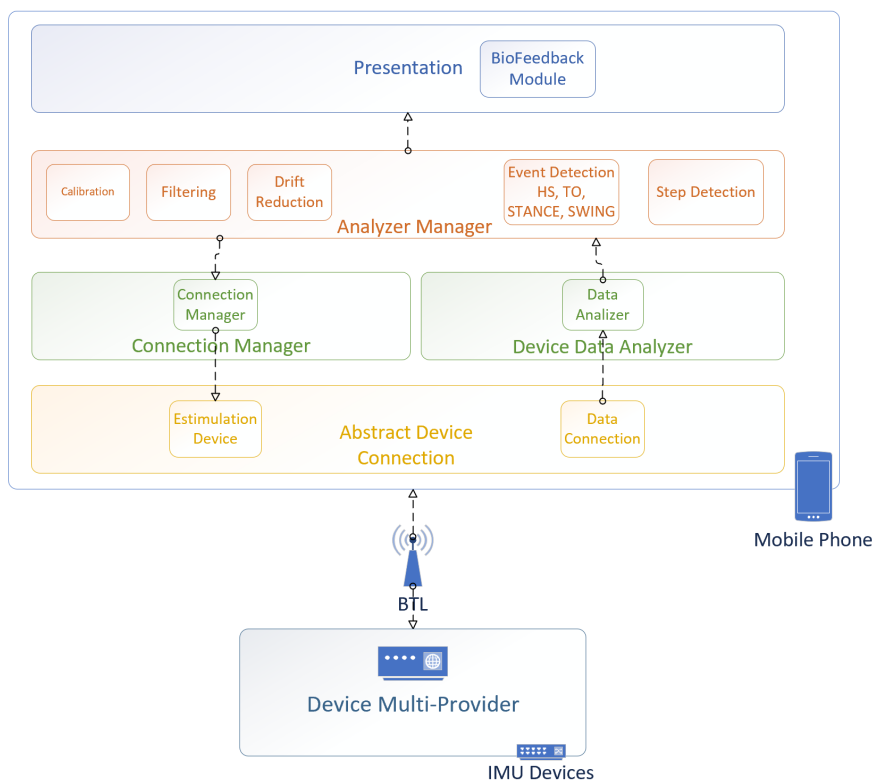


Figura 4: Arquitectura de sistemas de PARKIBIP.

3. Estimule de manera vibratoria al usuario en base a reglas clínicas predefinidas
4. Estandarice el flujo de datos dentro de la aplicación, y entre la misma y los dispositivos conectados
5. Su usabilidad escale a ambientes abiertos, contemplando su posible aplicación en la vida diaria del paciente

De esta manera, y a partir de los principios mencionados, se diseñó e implementó la arquitectura de sistemas de la Fig. 4.

En el diseño se pueden identificar seis capas específicas, con responsabilidades bien delineadas. Primero, y externo a PARKIBIP, los dispositivos IMU con conectividad Bluetooth compuestos por los distintos sensores de movimiento. Asimismo, se identifican tres módulos de bajo nivel, vinculados a la conectividad y búsqueda, interacción con los distintos dispositivos y recuperación de datos de los sensores empleados (“Abstract Device Connection”, “Connection manager”, “Device Data Analyzer”). Luego, se encuentra un módulo de procesamiento de la información recopilada, vinculado al análisis y aplicación de algoritmos matemáticos que identifican los parámetros espacio temporales (“Analyzer Manager”). Finalmente, un módulo de presentación de resultados, compuesto por un controlador clínico capaz de tomar decisiones y estimular adecuadamente al usuario.

Al momento, el sistema se encuentra en fase alfa, mejorando el módulo clínico -junto a un equipo interdisciplinario- y la precisión del sistema. Las primeras pruebas del sistema han tomado lugar en un ambiente cerrado de desarrollo y con sujetos sanos.

IV. DISCUSIÓN

En la línea de investigación e implementación planteada, se logró exitosamente adquirir conocimiento actualizado relevan-

te a la temática del proyecto, y desarrollar un sistema orientado a mejorar la marcha de los pacientes con la enfermedad de Parkinson.

A partir de los resultados -en la primera versión del sistema-, los datos indican que es posible estimular adecuadamente al sujeto en rehabilitación con un fin clínico. Para ello, es fundamental iniciar la fase de prueba con enfermos de Parkinson, mejorar el módulo clínico y evaluar los resultados en sesiones de fisioterapia. Asimismo, es clave resaltar y considerar los diversos algoritmos físico matemáticos existentes y los que se encuentran en continuo desarrollo, respecto al proceso de navegación humana. En este punto se encontró una gran limitante, si bien existe evidencia científica de algoritmos computacionales propuestos, en muchos casos no fue posible acceder a su implementación o explicación causa de líneas comerciales.

En este sentido, se utilizó un método sistemático que extrae características automáticas para la evaluación de la marcha de los pacientes. En base a los datos de los distintos sensores y al cómputo de la orientación expresada en la notación “quaternion”, es posible aplicar algoritmos computacionales que estimen las fases del ciclo de la marcha. Como primera aplicación, se empleó la técnica de detección picos -en determinado periodo de tiempo-, evaluada a los resultados de una función; la cual permite detectar eventos de golpe de talón y el despegue de los dedos del pie. Con los eventos detectados, se pueden analizar distintos parámetros (i.e. velocidad, zancada, cadencia, otros) y tomar decisiones en base a un módulo de reglas clínicas. Los primeros resultados muestran que es posible extraer características de la marcha de forma automática y ser usadas con un fin clínico y fuera de un laboratorio de marcha. El proyecto en curso, presenta el potencial de evaluar

la marcha de pacientes con la enfermedad de Parkinson, y permitirle al enfermo una rehabilitación personal que prolongue el trabajo del fisioterapeuta en su vida cotidiana.

En este apartado, se resalta la fase alfa del proyecto de investigación y los futuros pasos de pruebas, comparación frente a un laboratorio de marcha en actividad y evaluación de resultados.

V. CONCLUSIONES

En este trabajo se estudió el uso de sensores IMU para analizar la marcha de las personas al caminar mediante una revisión sistemática. Los resultados de dicha investigación fueron de gran utilidad para el avance en la implementación de un sistema de biorretroalimentación que permita al enfermo de Parkinson una rehabilitación personal.

Por lo tanto, se diseñó una aplicación móvil Android, que obtiene y procesa información en tiempo real. El sistema se interconecta a dos dispositivos IMU, colocados en el tobillo izquierdo y derecho de los pacientes, que permiten recopilar datos de acelerómetro, giroscopio y magnetómetro con el propósito de inferir la orientación y el movimiento de los cuerpos asociados -pies-.

En trabajos futuros, se espera concluir un periodo de evaluación de distintos algoritmos de detección de eventos y computo de parámetros de la marcha, con el fin de mejorar la precisión de los estímulos que favorezcan la rehabilitación de los pacientes. Asimismo, extender el número de pruebas concretadas con enfermos de Parkinson, que permitan validar el sistema implementado. Finalmente, pero no menor, se espera que las investigaciones realizadas contribuyan favorablemente a la comunidad biomédica.

AGRADECIMIENTOS

El presente trabajo fue realizado bajo el acompañamiento del Sr. XXXXXXXXXX, pertenecientes al Núcleo de Ingeniería Biomédica de las Facultades de Medicina e Ingeniería, Universidad de la República-URUGUAY. Los más sinceros agradecimientos.

REFERENCIAS

- [1] A. Garrett, "Biology of Parkinson's disease: pathogenesis and pathophysiology of a multisystem neurodegenerative disorder," *Dialogues in clinical neuroscience*, vol. 6, no. 3, pp. 259–80, 2004.
- [2] J. Jankovic, J. Nutt, and L. Sudarsky, "Classification diagnosis and etiology of gait disorders," *Adv. Neurol.*, vol. 87, pp. 119–33, Jan. 2001.
- [3] A. Ashburn, E. Stack, R. M. Pickering, and C. D. Ward, "A community-dwelling sample of people with parkinson's disease: characteristics of fallers and non-fallers," *Age Ageing*, vol. 30, pp. 47–52, Jan. 2001.
- [4] W. Rogers, "Disorders of posture balance and gait in parkinson's disease," *Clin. Geriatr. Med.*, vol. 12, no. 4, pp. 825–45, Nov. 1996.
- [5] J. Stamatakis, J. Crémers, D. Maquet, B. M. Macq, and G. Garraux, "Gait feature extraction in parkinson's disease using low-cost accelerometers," in *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, The organization. IEEE, 8–30 Aug.-3 Sept, p. 7900–7903, an optional note.
- [6] J. M. Hausdorff, M. E. Cudkowicz, R. Firtion, J. Y. Wei, and A. L. Goldberger, "Gait variability and basal ganglia disorders: stride-to-stride variations of gait cycle timing in parkinson's disease and huntington's disease," *Mov. Disord.*, vol. 13, no. 3, pp. 428–37, May 1998.
- [7] S. Lord, K. Baker, A. Nieuwboer, D. Burn, and L. Rochester, "Gait variability in parkinson's disease: an indicator of non-dopaminergic contributors to gait dysfunction?" *J. Neurol.*, vol. 258, no. 4, pp. 566–572, Oct. 2010.
- [8] H. Castillo and F. Simini, "SIMIC: Sistema Informático de Monitoreo de la Insuficiencia Cardíaca." Montevideo: Poster - Ingeniería de Muestra, 2019.
- [9] M. Alzugaray, S. Vico, C. Vega, G. Rey, and F. Simini, "Sepepe seguimiento perinatal personalizado." Montevideo: Poster - Ingeniería de Muestra, 2019.
- [10] M. Galnares, R. Low, S. Nesmachnow, and F. Simini, "Medical reasoning oriented orthosis to ease clinical practice and record keeping." Prague: World Congress on Medical Physics and Biomedical Engineering (WC2018), 2018 June 3–8.
- [11] R. Olivares, G. Silvera, P. Alvarez-Rocha, C. Taramasco, G. Ormaechea, and F. Simini, "High-level language to specify an adaptive heart failure follow up strategy." SABI2020, 2019.
- [12] Y. Li and Y. Guo, "Wiki-health: from quantified self to self-understanding. future gener. comput. syst." *J. Neurol.*, vol. 56, p. 333–359, 2015.
- [13] D. L. Frank, L. Khorshid, J. F. Kiffer, C. S. Moravec, and M. G. McKee, "Biofeedback in medicine: Who, when, why and how?" pp. 85–91, jun 2010.
- [14] J. Nonnekes, A. Snijders, J. Nutt, G. Deuschl, N. Giladi, and B. Bloem, "Freezing of gait: a practical approach to management." *Lancet Neurol.*, vol. 14, no. 7, pp. 768–78, 2015.
- [15] P. Ginis, A. Nieuwboer, M. Dorfman, A. Ferrari, E. Gazit, C. G. Canning, L. Rocchi, L. Chiari, J. M. Hausdorff, and A. Mirelman, "Feasibility and effects of home-based smartphone-delivered automated feedback training for gait in people with Parkinson's disease: A pilot randomized controlled trial," *Parkinsonism & Related Disorders*, vol. 22, pp. 28–34, jan 2016. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/26777408> <http://www.ncbi.nlm.nih.gov/pubmed/26777408>
- [16] W. O. C. Lopez, C. A. E. Higuera, E. T. Fonoff, C. de Oliveira Souza, U. Albicker, and J. A. E. Martinez, "Listenmee@ and Listenmee@ smartphone application: Synchronizing walking to rhythmic auditory cues to improve gait in Parkinson's disease," *Human Movement Science*, vol. 37, pp. 147–156, oct 2014. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/25215623> <https://linkinghub.elsevier.com/retrieve/pii/S0167945714001237>
- [17] L. Rochester, K. Baker, V. Hetherington, D. Jones, A.-M. Willems, G. Kwakkel, E. Van Wegen, I. Lim, and A. Nieuwboer, "Evidence for motor learning in Parkinson's disease: Acquisition, automaticity and retention of cued gait performance after training with external rhythmical cues," *Brain Research*, vol. 1319, pp. 103–111, mar 2010. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/20064492> <https://linkinghub.elsevier.com/retrieve/pii/S0006899310000041>
- [18] A. Nieuwboer, G. Kwakkel, L. Rochester, D. Jones, E. van Wegen, A. M. Willems, F. Chavret, V. Hetherington, K. Baker, and I. Lim, "Cueing training in the home improves gait-related mobility in Parkinson's disease: the RESCUE trial," *Journal of Neurology, Neurosurgery & Psychiatry*, vol. 78, no. 2, pp. 134–140, feb 2007. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/17229744> <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC2077658> <http://jnnp.bmj.com/cgi/doi/10.1136/jnnp.200X.097923>
- [19] P. A. Rocha, G. M. Porfirio, H. B. Ferraz, and V. F. Trevisani, "Effects of external cues on gait parameters of Parkinson's disease patients: A systematic review," *Clinical Neurology and Neurosurgery*, vol. 124, pp. 127–134, sep 2014. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/25043443> <https://linkinghub.elsevier.com/retrieve/pii/S0303846714002339>
- [20] S. J. Spaulding, B. Barber, M. Colby, B. Cormack, T. Mick, and M. E. Jenkins, "Cueing and Gait Improvement Among People With Parkinson's Disease: A Meta-Analysis," *Archives of Physical Medicine and Rehabilitation*, vol. 94, no. 3, pp. 562–570, mar 2013. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/23127307> <https://linkinghub.elsevier.com/retrieve/pii/S0003999312010842>
- [21] J. B. Saunders, V. T. Inman, and H. D. Eberhard, "The major determinants in normal and pathological gait." *The Journal of bone and joint surgery. American volume*, vol. 35-A, no. 3, pp. 543–58, jul 1953. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/13069544>
- [22] E. Ayyappa, "Normal human locomotion. Part 1: Basic concepts and terminology," *Prosthetics Orthot.*, vol. 9, p. 10–17, 1997.
- [23] N. F. Ribeiro and C. P. Santos, "Inertial measurement units: A brief state of the art on gait analysis," *ENBENG 2017 - 5th Portuguese Meeting on Bioengineering, Proceedings*, pp. 1–4, 2017.
- [24] C. Castagneri, V. Agostini, G. Balestra, M. Knaflitz, Carlone, and G. Massazza, "Emg asymmetry index in cyclic movements," 10 2018, pp. 223–226.
- [25] J. Taborri, E. Palermo, S. Rossi, and P. Cappa, "Gait partitioning methods: A systematic review," 2016.
- [26] W. Pirker and R. Katzenschlager, "Gait disorders in adults and the elderly: A clinical guide," *Wiener klinische Wochenschrift*, vol. 129, 10 2016.

- [27] S. O. Madgwick, "An efficient orientation filter for inertial and inertial/magnetic sensor arrays," 2010.

Apéndice 6

Documento - Consentimiento informado al usuario de PARKIBIP

PARKIBIP: Retroalimentación activa en la marcha de personas con Enfermedad de Parkinson.

Proyecto de investigación llevado a cabo por Samuel Sainz, Carlos Huerta, Lic. en fisioterapia Macarena Vergara, Lic. en fisioterapia Darío Santos, Franco Simini.

Dicha investigación tiene como objetivo analizar el efecto inmediato de un estímulo externo en las variables espacio temporales de la marcha de individuos con enfermedad de Parkinson. Procedimiento: El proyecto consta de una fase de recolección de información de la marcha de pacientes con Parkinson, y posteriormente una fase de prueba que se llevarán a cabo en el Laboratorio de Marcha del Hospital de Clínicas. Se utilizará un dispositivo IMU (Inertial Measurement Unit) colocado en el tobillo de los pacientes, desarrollado por el equipo interdisciplinario en el Núcleo de Ingeniería Biomédica (NIB) de la facultad de Medicina e Ingeniería y la cátedra de rehabilitación y medicina física. Este dispositivo, recibe datos de aceleración, velocidad angular y magnetómetro de cada IMU, los procesa en tiempo real determinando las fases de la marcha y sus parámetros espacio-temporales (longitud del paso, cadencia, velocidad). De esta forma, PARKIBIP emite un estímulo mecánico periódico con una latencia ajustable al tiempo del paso basal (marcha habitual) a la que Ud. camina.

Se hará un registro de su marcha en tres ocasiones dentro de una misma sesión:

- Caminando a una velocidad auto-seleccionada (como camina habitualmente)
- Caminando a una velocidad auto-seleccionada con el sensor inactivo
- Caminando a una velocidad auto-seleccionada, con PARKIBIP activo y un estímulo mecánico programado con la velocidad media de la marcha auto-seleccionada.

Además de utilizar PARKIBIP, se le colocarán marcadores (esferas de 2 cm de diámetro) que serán pegadas con cinta adhesiva sobre la piel en la proximidad de las articulaciones de sus piernas. Durante su marcha las esferas serán filmadas mediante cámaras infrarrojas, que solamente captan los marcadores y no su cuerpo, por lo que se preserva el anonimato de su identidad. Este procedimiento es inocuo y no presenta riesgo para su salud.

Tenga presente que su participación es voluntaria. Usted no está obligado a formar parte de este proyecto, y en caso que opte por no participar su decisión será respetada. Asimismo, aún cuando usted decida involucrarse, podrá en cualquier momento y sin ninguna consecuencia desistir.

Los datos que se recaben son confidenciales, por lo que únicamente los conocerá el grupo de investigadores.

Su participación es gratuita, si usted decide participar no recibirá ningún beneficio económico.

ANEXOS

Anexo 1

Escala Unificada de Clasificación de la Enfermedad de Parkinson (MDS-UPDRS) Goetz et al. (2008)

1.1. Estado Mental, de comportamiento y estado de ánimo

0	Ninguna
1	Leve, olvido con recuerdo parcial de algunos hechos sin otras dificultades
2	Pérdida moderada de memoria, con desorientación y moderada dificultad en el tratamiento de problemas complejos
3	Pérdida grave de memoria con desorientación temporal y a menudo espacial, grave dificultad con los problemas
4	Pérdida grave de memoria con orientación sólo para personas, incapaz de juzgar o resolver problemas

Tabla 1: Deterioro intelectual

0	Ninguno
1	Sueño Intenso
2	Alucinaciones «benignas» con retención de las mismas
3	Alucinaciones más frecuentes sin retención, pueden interferir con la actividad Diaria
4	Alucinaciones persistentes, ilusiones o psicosis floridas

Tabla 2: Trastornos del pensamiento

0	No presente
1	Periodos de tristeza o culpabilidad superiores a lo normal, nunca presentes durante más de unos días o una semana
2	Depresión persistente durante más de una semana
3	Síntomas vegetativos (insomnio, anorexia, abulia, pérdida de peso)
4	Síntomas vegetativos con tendencias suicidas

Tabla 3: Depresión

0	Normal
1	Menos afirmativo, más pasivo
2	Pérdida de iniciativa o desinterés en actitudes electivas
3	Pérdida de iniciativa o desinterés en la rutina diaria
4	Abandono, pérdida completa de motivación

Tabla 4: Motivación, iniciativa

1.2. Actividades de la vida diaria

0	Normal
1	Levemente afectado, sin dificultad para ser entendido
2	Moderadamente afectado, ocasionalmente debe pedírsele que repita las cosas
3	Gravemente afectado, se le pide frecuentemente que repita las cosas
4	Ininteligible la mayor parte del tiempo

Tabla 5: Lenguaje

0	Normal
1	Leve
2	Moderada excesiva salivación, babeo nocturno
3	Marcado exceso de salivación con algo de babeo
4	Marcado babeo, requiere constante de pañuelo

Tabla 6: Salivación

0	Normal
1	Levemente pequeña o lenta
2	Todas las palabras pequeñas pero legibles
3	Gravemente afectada, no son legibles todas las palabras
4	Mayoritariamente ilegibles

Tabla 7: Escritura

0	Normal
1	Lento y poco hábil, pero se vale solo
2	Puede cortar la mayoría de alimentos, para algunos necesita ayuda
3	Le deben cortar la comida, pero puede alimentarse solo
4	Necesita que lo alimenten

Tabla 8: Cortar alimentos/manejar utensilios

0	Normal
1	Lento, pero sin ayuda
2	Ocasionalmente necesita ayuda
3	Necesita considerable ayuda, aunque puede hacer algunas cosas solo
4	Necesita ayuda completa

Tabla 9: Vestir

0	Normal
1	Lento, pero sin ayuda
2	Necesita ayuda con la ducha o el baño, o es muy lento en el cuidado de la higiene
3	Necesita ayuda para lavarse, cepillarse los dientes, ir al baño
4	Necesita ayuda completa

Tabla 10: Higiene

0	Normal
1	Lento pero sin ayuda
2	Puede volverse o ajustar las sábanas pero con gran dificultad
3	No puede volverse o ajustarse las sábanas solo
4	Necesita ayuda completa

Tabla 11: Volverse en la cama/ajustar las sábanas

0	Ninguna
1	Rara
2	Ocasionales, menos de una por día
3	Cae en promedio una vez por día
4	Más de una por día

Tabla 12: Caídas sin relación con el freezing

0	Normal
1	Raro, puede haber duda
2	Caídas ocasionales por freezing
3	Frecuente freezing, caídas ocasionales
4	Frecuentes caídas por freezing

Tabla 13: Freezing al caminar

0	Normal
1	Leve dificultad, arrastra las piernas o disminuye el balanceo de los brazos
2	Dificultad moderada, sin requerir ayuda
3	Afectación grave, que requiere asistencia
4	No puede andar, incluso con ayuda

Tabla 14: Andar

0	Ausente
1	Leve e infrecuente, no molesta al paciente
2	Moderado, molesto para el paciente
3	Grave, interfiere con muchas actividades
4	Marcado, interfiere con muchas actividades

Tabla 15: Temblor

0	Ninguna
1	Ocasionalmente tiene insensibilidad, hormigueo y leve dolor
2	Frecuente pero no estresante
3	Sensación de dolor frecuente
4	Dolor insoportable

Tabla 16: Molestias sensoriales relacionadas con el parkinsonismo

1.3. Examen motor

0	Normal
1	Leve pérdida de expresión, dicción, volumen
2	Monótono, mal articulado pero comprensible
3	Marcada dificultad, difícil de entender
4	Ininteligible

Tabla 17: Lenguaje

0	Normal
1	Leve hipomimia
2	Leve pero definida disminución anormal de la expresión
3	Moderada hipomimia, labios separados parte del tiempo
4	Cara fija, labios separados 1/2 cm o más, con pérdida completa de expresión

Tabla 18: Expresión facial

0	Ausente
1	Leve e infrecuente
2	Necesita ayuda con la ducha o el baño, o es muy lento en el cuidado de la higiene
3	Necesita ayuda para lavarse, cepillarse los dientes, ir al baño
4	Necesita ayuda completa

Tabla 19: Temblor en reposo

0	Ausente
1	Leve, presente con acción
2	Moderado, presente con acción
3	Moderado, presente con acción y manteniendo la postura
4	Marcado, interfiere con la alimentación

Tabla 20: Temblor postural o de acción en manos

0	Ausente
1	Leve o sólo con actividad
2	Leve/moderada
3	Marcada, en todo el rango de movimiento
4	Grave

Tabla 21: Rigidez

0	Normal
1	Leve lentitud y/o reducción en amplitud
2	Dificultad moderada
3	Dificultad grave
4	Apenas puede realizarlos

Tabla 22: Tocarse la punta de los dedos (Finger tapping, el paciente golpea el pulgar con el índice en rápida sucesión y con la mayor amplitud posible; realizar con cada mano por separado)

0	Normal
1	Leve lentitud y/o reducción en amplitud
2	Dificultad moderada
3	Dificultad grave
4	Apenas puede realizarlos

Tabla 23: Movimientos de la mano (abrir y cerrar las manos en rápida sucesión)

0	Normal
1	Leve lentitud y/o reducción en amplitud
2	Dificultad moderada
3	Dificultad grave
4	Apenas puede realizarlos

Tabla 24: Agilidad en la pierna (movimientos con el talón sobre el suelo, la amplitud debería ser de 3 pulgadas)

0	Normal
1	Lento, puede necesitar más de un intento
2	Se empuja hacia arriba con los brazos o la silla
3	Tiende a caer hacia atrás, puede necesitar muchos intentos pero puede levantarse sin ayuda
4	Incapaz de levantarse sin ayuda

Tabla 25: Levantarse de una silla (con brazos cruzados)

0	Normal erecto
1	Levemente inclinado, podría ser normal para una persona mayor
2	Anormal: inclinado, puede que hacia algún lado
3	Grave inclinación con escoliosis
4	Marcada flexión con postura muy anormal

Tabla 26: Postura

0	Normal
1	Anda lentamente
2	Anda con dificultad, con poca o sin ayuda, algún balanceo, pasos cortos o propulsión
3	Afectación grave, necesita ayuda frecuente
4	No puede andar

Tabla 27: Marcha

0	Normal
1	Se recupera sin ayuda
2	Caería si no se sostiene
3	Se cae espontáneamente
4	Imposible mantenerse de pie

Tabla 28: Estabilidad postural (test de retropulsión)

0	Nada
1	Mínima lentitud, podría ser normal
2	Leve lentitud y escasez de movimientos, definitivamente anormales, o disminuye la amplitud de movimientos
3	Moderada lentitud, escasez de movimientos, o disminuye la amplitud
4	Marcada lentitud, escasez de movimientos, o disminuye la amplitud de movimientos

Tabla 29: Bradicinesia/hipocinesia

1.4. Complicaciones del tratamiento (en la semana anterior)

Discinesias.

0	No hay
1	1 % a 25 % del día
2	26 % a 50 % del día
3	51 % a 75 % del día
4	76 % a 100 % del día

Tabla 30: Duración: ¿En qué proporción de las horas de vigilia del día están presentes las discinesias? (información por anamnesis)

0	No incapacitantes
1	Discretamente incapacitantes
2	Moderadamente incapacitantes
3	Muy incapacitantes
4	Totalmente invalidantes

Tabla 31: Incapacidad: ¿Qué grado de incapacidad producen las discinesias? (información por anamnesis; puede modificarse por la exploración)

0	No hay discinesias dolorosas
1	Leve
2	Moderado
3	Intenso
4	Marcado

Tabla 32: Discinesias dolorosas: ¿Cuánto dolor producen las discinesias?

0	Si
1	No

Tabla 33: Presencia de distonía en las primeras horas de la mañana (información por anamnesis)

Fluctuaciones clínicas.

0	Si
1	No

Tabla 34: ¿Hay algunos periodos off predecibles que se instauran en un momento determinado tras una dosis de medicación?

0	Si
1	No

Tabla 35: ¿Hay algunos periodos off imprevisibles en relación con las tomas de medicación?

0	Si
1	No

Tabla 36: ¿Hay periodos off que instauran de forma súbita? (por ejemplo, en pocos segundos)

0	Ninguna
1	1 % al 25 % del día
2	26 % al 50 % del día
3	51 % al 75 % del día
4	76 % al 100 % del día

Tabla 37: Como promedio, ¿en qué proporción de las horas de vigilia del día está el paciente en off?

Otras complicaciones.

0	Si
1	No

Tabla 38: ¿Tiene el paciente anorexia, náuseas o vómitos?

0	Si
1	No

Tabla 39: ¿Tiene el paciente algún trastorno del sueño? (por ejemplo, insomnio o hipersomnia)

0	Si
1	No

Tabla 40: ¿Tiene el paciente ortostatismo sintomático?

1.5. Escala Schwab & England

%	Es posible seleccionar números entre los porcentajes establecido
100	Completamente independiente. Capaz de realizar cualquier tarea sin dificultad, lentitud o alteraciones
90	Completamente independiente. Puede tardar el doble de lo normal en realizar una tarea
80	Independiente en la mayoría de tareas. Tarda el doble. Consciente de su dificultad y enlentecimiento
70	No completamente independiente. En algunas tareas tarda tres o cuatro veces más de lo normal
60	Alguna dependencia. Puede hacer la mayoría de tareas, pero muy lentamente y con mucho esfuerzo
50	Más dependiente. Necesita ayuda en la mitad de las tareas cotidianas. Dificultad para todo
40	Muy dependiente. Sólo puede realizar algunas tareas sin ayuda. Con mucho esfuerzo puede realizar alguna tarea. Necesita mucha ayuda
30	Ninguna tarea solo. Grave invalidez
20	Totalmente dependiente. Puede ayudar algo en algunas actividades
10	Dependiente. Inválido
0	Las funciones vegetativas como tragar, vejiga e intestino no funcionan, postrado en cama

Tabla 41: Escala de AVD

1.6. Estadificación Modificada Hoehn & Yahr

Estadio 0	No hay signos de enfermedad
Estadio 1	Enfermedad unilateral
Estadio 1.5	Afectación axial y unilateral
Estadio 2	Enfermedad Bilateral, sin deterioro del equilibrio
Estadio 2.5	Enfermedad bilateral leve, con recuperación en la prueba de tracción
Estadio 3	Enfermedad bilateral leve a moderada, cierta inestabilidad postural; independencia física
Estadio 4	Discapacidad grave, aún puede caminar o estar de pie sin asistencia
Estadio 5	Destinado a la silla de ruedas o postrado en cama, a no ser que sea asistido

Tabla 42: Escala de estadios