



Universidad de Valladolid



**ESCUELA DE INGENIERÍAS
INDUSTRIALES**

UNIVERSIDAD DE VALLADOLID

ESCUELA DE INGENIERIAS INDUSTRIALES

Grado en Ingeniería Electrónica Industrial y Automática

**GESTIÓN Y ANÁLISIS DE UNA BBDD CON EL
RESULTADO DE TERAPIAS DE REHABILITACIÓN
NEUROMOTORA.**

Autor:

Del Pozo Ituero, Fernando

Tutor:

Pérez Turiel, Javier

**Ingeniería en Sistemas y
Automática**

Valladolid, Septiembre de 2020.



AGRADECIMIENTOS

A Javier Pérez Turiel, por hacerme un mejor profesional y las oportunidades otorgadas.

Al equipo de CARTIF, en especial, a Alejandro Cuadrado y a Pablo Viñas, por aportarme un nuevo mundo de conocimientos y un inmejorable ambiente de trabajo.

A mi familia y a mi novia, por todo vuestro apoyo, hoy soy quien soy gracias a vosotros.



RESUMEN

En el presente Trabajo de Fin de Grado se desarrolla un sistema para optimizar el análisis clínico de terapias de rehabilitación del sistema robotizado Physiobot, aplicando métricas clínicas existentes y desarrollando otras nuevas, que permitan caracterizar el estado y evolución de un paciente. Para ello, se realiza la gestión y tratamiento de una BBDD de terapias de rehabilitación neuromotora, formada con datos de distintos pacientes y distintas sesiones de rehabilitación de cada uno de ellos. Finalmente, se obtienen las conclusiones de las métricas y del software desarrollado y se plantean posibles líneas futuras de trabajo.

PALABRAS CLAVES

Physiobot

Neurotecnología

Bases de datos

Rehabilitación

Métricas



ABSTRACT

In this Final Degree Project, a system is developed to optimize the clinical analysis of rehabilitation therapies of the Physiobot robotic system, applying existing clinical metrics and the development of new ones that allow characterizing the status and evolution of a patient. To do this, the management and treatment of a database of neuromotor rehabilitation therapies, consisting of data from different patients and different rehabilitation sessions of each of them, is carried out. Finally, the conclusions of the metrics and the software developed are obtained and possible future working lines are proposed.

KEY WORDS

Physiobot

Neurotechnology

Database

Rehabilitation

Metrics



ÍNDICE

1.	Introducción.....	1
1.1	Planteamiento	1
1.2	Problemas	1
1.3	Objetivos	2
2.	Marco Teórico.....	3
2.1	Estado del arte.....	3
2.1.1	Introducción.....	3
2.1.2	Taxonomía	3
2.1.3	Tecnología específica.....	4
2.1.4	Neuromodulación y Neuroestimulación	6
2.1.5	Robótica de rehabilitación y asistencia.....	7
2.2	CARTIF	16
2.3	Bases de datos	17
2.3.1	Historia de las bases de datos.....	17
2.3.2	Concepto básico.....	18
2.3.3	Modelo de datos	19
3.	Entorno de trabajo	23
3.1	Visual Studio	23
3.2	C#	23
3.3	R.....	24
3.4	SQL Server	25
3.5	Linq.....	25
3.6	MVC	26
4.	Desarrollo del software.....	29
4.1	Arquitectura del software.....	29
4.2	BBDD de PHYSIOBOT	30
4.2.1	TerapiaBM	31
4.2.2	PacientesBM.....	32
4.2.3	Muestra.....	32



4.2.4	Nom_Terapias	33
4.3	Vista.....	33
4.4	Ejes de referencia.....	35
4.5	Análisis de los datos asociados a una terapia	36
4.5.1	Trayectoria	36
4.5.2	Velocidad respecto al eje X	37
4.5.3	Velocidad respecto al eje Y	38
4.5.4	Fuerza respecto al eje X	39
4.5.5	Fuerza respecto al eje Y.....	41
4.6	Evolución de pacientes	41
4.7.1	Marcador.....	42
4.7.2	Porcentaje de error	43
4.7.3	Media y desviación.....	44
4.7.4	Facilidad de movimiento.....	45
4.7.5	Tiempo de reacción.....	46
4.7.6	Número de temblores	48
4.7.7	Simetría direccional respecto a X e Y	49
4.7.8	Simetría respecto a velocidad y ángulo.....	49
4.7.9	Evolución de fuerza.....	51
4.7.10	Generación de informe	51
4.8	Tratamiento de errores	53
4.8.1	Errores de selección de parámetros.....	54
4.8.2	Errores de tratamiento de datos.....	54
4.8.3	Errores de gestión de la BBDD.....	54
4.9	Documentación del código	55
5.	Implantación.....	59
5.1	Primeros resultados	59
5.2	Calibración	60
5.2.1	Calibración de la terapia activo-asistida	60
5.2.2	Calibración de la terapia pasivo-guiada	62
5.2.3	Calibración de la terapia AVD.....	64



Universidad de Valladolid

6	Evaluación de las métricas.....	67
7	Conclusiones	69
8	Referencias bibliográficas	71
	ANEXOS.....	77



ÍNDICE DE FIGURAS:

Figura 1: Estructura del anterior software..	1
Figura 2: Proyecto BioMot..	4
Figura 3: TMS y tDCS..	6
Figura 4: Detección de intención de movimiento y fatiga muscular a través de HD-EMG con un exoesqueleto.....	8
Figura 5: Exo-H2.....	8
Figura 6: ARMassist.....	9
Figura 7: Exoesqueleto WOTAS.....	9
Figura 8: RobHand.....	10
Figura 9: Lokomat.....	11
Figura 10: Armeo Power.....	12
Figura 11: Amadeo.....	12
Figura 12: Diego.....	13
Figura 13: MyPro.....	14
Figura 14: Physiobot.....	14
Figura 15: Terapia activo-asistida.....	15
Figura 16 Terapia pasivo-guiada.....	16
Figura 17: Terapia AVD de supermercado..	16
Figura 18: Robot Copito.....	17
Figura 19. Historia de las bases de datos.....	18
Figura 20: Conexión a BBDD.....	19
Figura 21: Diagrama E-R.....	20
Figura 22: Diagrama jerárquico.....	21
Figura 23: Tablas según el modelo relacional.....	22
Figura 24: Diagrama relacional.....	22
Figura 25: Arquitectura .NET Framework.....	24
Figura 26: Estructura de la operación de consulta con LINQ.....	26
Figura 27: MVC. Fuente: Campus MVC.....	27
Figura 28: BBDD de PHYSIOBOT.....	30



Universidad de Valladolid

Figura 29: Vista del inicio de las aplicaciones.	34
Figura 30: Vista de las aplicaciones graficando.	34
Figura 31: Referencias de PHYSIOBOT.	35
Figura 32: Diagrama de bloques de la fuerza del robot.	40
Figura 33: Portada del informe.	52
Figura 34: Página estándar del informe.	53
Figura 35: Errores de selección de parámetros.	54
Figura 36: Error por agotar el tiempo de espera con el servidor.	55
Figura 37: Documentación web.	56
Figura 38: Documentación de métodos.	56
Figura 39: Documentación de variables de entrada.	57



ÍNDICE DE ECUACIONES

Ecuación 1: Cálculo de la fuerza asistiva.....	40
Ecuación 2: Cálculo de la aceleración para un punto.....	40
Ecuación 3: Porcentaje de error cometido.....	43
Ecuación 4: Media de una variable.....	44
Ecuación 5: Media de los puntos realizados frente a los deseados.....	45
Ecuación 6: Desviación típica.....	45
Ecuación 7: Velocidad total en un punto.....	46
Ecuación 8: Cálculo de la dirección de las posiciones realizadas respecto a un eje.....	47
Ecuación 9: Cálculo de la dirección de las posiciones deseadas respecto a un eje.....	47
Ecuación 10: Producto de las direcciones respecto a un eje.....	47
Ecuación 11: Suma cuadrática de las direcciones en los ejes.....	48
Ecuación 12: cálculo de la velocidad del objeto de la pantalla.....	50
Ecuación 13: Cálculo del ángulo.....	50
Ecuación 14: Porcentaje de fuerza realizado respecto a un eje.....	51
Ecuación 15: Calibración respecto a Y para la BBDD de prueba.....	59
Ecuación 16: Calibración respecto a X para la BBDD de prueba.....	59
Ecuación 17: Calibración sin inversión de las posiciones deseadas respecto a X.....	61
Ecuación 18: Calibración sin inversión de las posiciones deseadas respecto a Y.....	61
Ecuación 19: Calibración con inversión de las posiciones deseadas respecto a X.....	63
Ecuación 20: Calibración con inversión de las posiciones deseadas respecto a Y.....	63
Ecuación 21: Escalado de las posiciones deseadas respecto a X.....	66
Ecuación 22: Escalado de las posiciones deseadas respecto a X.....	66



ÍNDICE DE TABLAS

Tabla 1: TerapiaBM. Fuente: Elaboración propia.	31
Tabla 2: PacientesBM. Fuente: Elaboración propia.....	32
Tabla 3: Muestra. Fuente: Elaboración propia.....	32
Tabla 4: Nom_Terapias. Fuente: Elaboración propia.	33



ÍNDICE DE GRÁFICAS

Gráfica 1: Métrica de Trayectoria.....	36
Gráfica 2: Métrica de trayectoria para una terapia de tipo 1	37
Gráfica 3: Métrica de la velocidad respecto al eje X.	38
Gráfica 4: Métrica de la velocidad respecto al eje Y.	38
Gráfica 5: Métrica de la fuerza respecto al eje X.....	39
Gráfica 6: Métrica de la fuerza respecto al eje Y.	41
Gráfica 7: Métrica de Marcador	42
Gráfica 8: Métrica de Marcador.....	43
Gráfica 9: Métrica de Media y Desviación.....	44
Gráfica 10: Facilidad de movimiento.....	46
Gráfica 11: Métrica de tiempo de reacción.....	47
Gráfica 12: Métrica de número de temblores.....	48
Gráfica 13: Métrica de simetría direccional.....	49
Gráfica 14: Métrica de simetría de velocidad y ángulo.....	50
Gráfica 15: Métrica de evolución de fuerza.	51
Gráfica 16: Trayectoria realizada con R.	59
Gráfica 17: Trayectoria con la calibración desarrollada en R.....	60
Gráfica 18: Trayectoria realizada y trayectoria deseada sin calibrar de la terapia tipo 1.....	61
Gráfica 19: Trayectoria realizada y trayectoria deseada calibrada de la terapia tipo 1.....	62
Gráfica 20: Trayectoria realizada y trayectoria deseada sin calibrar de la terapia tipo 2.....	63
Gráfica 21: Trayectoria realizada y trayectoria deseada calibrada de la terapia tipo 2.....	64
Gráfica 22: Trayectoria realizada y trayectoria deseada sin calibrar de la terapia tipo 3.....	64
Gráfica 23: Puntos realizados y puntos deseados respecto al eje X frente al tiempo en una terapia de tipo 3 sin calibrar.....	65
Gráfica 24:Puntos realizados y puntos deseados respecto al eje Y frente al tiempo en una terapia de tipo 3 sin calibrar.....	65



Universidad de Valladolid

Gráfica 25: Trayectoria realizada y trayectoria deseada calibrada de la terapia tipo 366



1. INTRODUCCIÓN

1.1 PLANTEAMIENTO

El presente Trabajo de Fin de Grado (TFG) plantea el desarrollo de un entorno software que realice el tratamiento de la información almacenada en una base de datos de terapias de rehabilitación neuromotora con la plataforma robotizada de neurorehabilitación PHYSIOBOT.

La Fundación CARTIF (en adelante CARTIF) ya desarrolló un software para realizar este tratamiento de datos, pero finalmente, debido al problema expuesto en el siguiente apartado, no se pudo implantar el programa informático y, por consiguiente, no se realizó la comprobación de la utilidad de las métricas ni la evolución de los pacientes.

El software inicial realizaba el tratamiento con el lenguaje de programación R, mostrando los resultados al usuario a través de una interfaz desarrollada en C#. En el propuesto en este TFG, se pretende desarrollar tanto el tratamiento como la interfaz en C#.

1.2 PROBLEMAS

El problema del programa que había inicialmente es que R y C# no trabajaban “en un mismo bloque”. Es decir, el programa se iniciaba con el ejecutable de C# y C# en algún proceso solicitaba a R información (R realizaba el tratamiento en sí de la base datos, ya que es un programa estadístico).

Para ello, había que referenciar donde se encontraba el script de R, la ruta donde se encontraba R instalado, y la ruta donde iba a guardar la información que iba a necesitar C# (véase la figura 1).

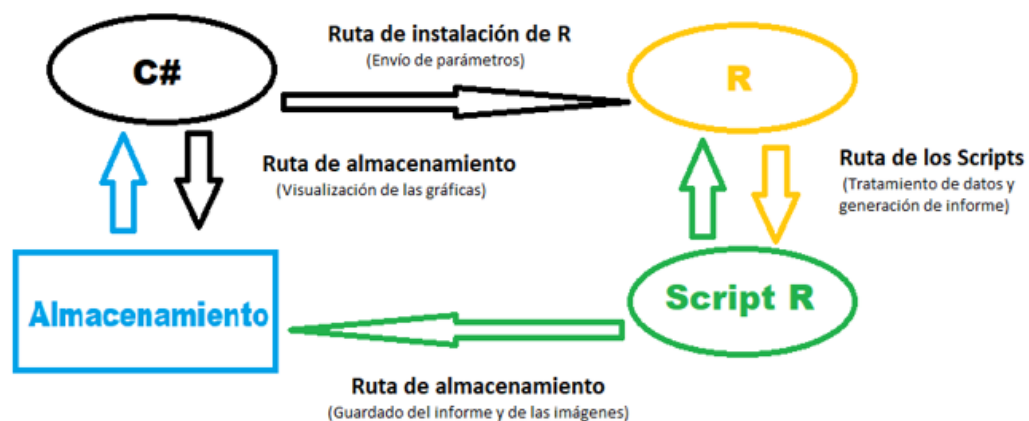


Figura 1: Estructura del anterior software. Fuente: Elaboración propia.



Esta comunicación se repetía varias veces, dando lugar a una gran variedad de rutas que necesariamente eran distintas entre sí (es decir, donde estaba R instalado era distinta a la ruta donde estaban sus scripts y a la ruta donde se guardaba la información).

Esta gran variedad de rutas hacía que implementar el programa en un determinado equipo fuera complejo, puesto que había que buscar dónde se utilizaban estas rutas y sustituirlas por las utilizadas en ese equipo. Además, en cada equipo es distinto, ya que depende de dónde se instale el programa R y el nombre de las unidades de almacenamiento (C, D, etc.). Esto implica que, una vez establecidas las rutas, el programa no se podía mover, puesto que se tendrían que referenciar de nuevo.

1.3 OBJETIVOS

El principal objetivo es realizar un entorno software que permita el tratamiento de la base de datos de terapias del proyecto PHYSIOBOT de CARTIF, más en concreto, la finalidad va a ser desarrollar e integrar dos aplicaciones software diferentes, siendo la primera la encargada de mostrar los gráficos más relevantes de los resultados de una terapia realizada con este robot cartesiano. Dichos gráficos mostrarán la trayectoria que ha realizado una persona con el robot frente a la trayectoria deseada, es decir, la que el paciente ve en pantalla. También tendremos la opción de observar la evolución de la fuerza y de la velocidad en los rangos de tiempo definidos por el terapeuta.

La segunda aplicación permitirá determinar la evolución relativa de un paciente. Para ello se han utilizado y desarrollado métricas que posibiliten ver en gráficos cómo ha evolucionado un paciente en un rango de terapias especificado por el terapeuta. Se tendrán las opciones tanto de ver una determinada métrica en el programa como la de elaborar un informe en formato PDF con las métricas seleccionadas por el terapeuta, y que, además, muestre los datos más relevantes, como son los valores máximos y mínimos, fechas en las que se han producido, etc.

El último objetivo es que los programas desarrollados permitan tratar las bases de datos de terapias de PHYSIOBOT de distintos hospitales para el uso de los terapeutas. Para ello, se utilizará una base de datos de terapias reales procedente de un hospital para realizar las calibraciones necesarias y sacar los resultados finales.

De estos resultados finales se obtendrán las conclusiones de la utilidad de las distintas métricas que se han implantado.



2. MARCO TEÓRICO

2.1 ESTADO DEL ARTE

2.1.1 Introducción

Se puede entender el movimiento humano como un conjunto de actuadores controlados por un sistema central que recibe múltiples entradas de sensores, siendo los actuadores el conjunto de nuestros músculos (sistema musculoesquelético), el sistema central sería nuestro sistema nervioso, y los sensores serían nuestros sentidos (sistema sensorial). Este sistema central se puede ver dañado por enfermedades, trastornos, etc., dando lugar a la pérdida o reducción de nuestra capacidad motora.

En este apartado se va a hablar de la situación actual de las neurotecnologías, entendiendo por neurotecnología a aquellas tecnologías dirigidas al estudio del sistema nervioso o a mejorar su función (Barrios et al 2017b). Se centrará sobre todo en España, aunque también se mencionará algún proyecto relevante de otros países. Además, se hará referencia a las métricas que utilizan para determinar la evolución de los pacientes.

2.1.2 Taxonomía

Según Barrios et al. (2017a) podemos distinguir conforme sus principios de diseño dos tipos de neurotecnologías, las basadas en modalidades físicas y las basadas en procesos bioquímicos. Dentro de las primeras podemos distinguir entre estos cuatro tipos de tecnologías: específicas, particularizadas, dirigidas al sistema sensorial y al sistema muscular y, finalmente, de soporte y complemento.

- Las **tecnologías específicas** desarrollan las interfaces neurales, destacando la interfaz cerebro-computador que consiste en la adquisición de ondas cerebrales para posteriormente ser tratadas por un computador.
- Las **tecnologías particularizadas** se pueden dividir en distintos grupos, como la neuroimagen, la neuromodulación o la neuroestimulación y la que más interesa en este TFG, la robótica de rehabilitación y asistencia.
- Las **tecnologías dirigidas al sistema sensorial y al sistema muscular** consisten en combinar las tecnologías mencionadas anteriormente con las desarrolladas para el sistema muscular o sensorial, como por ejemplo la estimulación cutánea.
- Las **tecnologías de soporte** son las tecnologías básicas sobre las que parten las anteriores tecnologías, como por ejemplo el almacenamiento de datos, la transmisión de señales, procesamiento de datos, etc.

2.1.3 Tecnología específica

De las interfaces cerebro-computador, el proyecto Brain2Motion de la Universidad Miguel Hernández de Elche (UMH), realizó un estudio sobre la decodificación de los parámetros cinemáticos del miembro superior, donde utilizaron métricas para tratar la información, como la precisión de la posición, el tiempo de seguimiento y la variabilidad del movimiento (Úbeda, Hortal, Iáñez, Pérez-Vidal y Azorín, 2015). También desarrollaron una interfaz de cerebro-computador (BCI, del inglés Brain-Computer Interface) capaz de controlar un exoesqueleto a través de la detección de la intención de movimiento, utilizando métricas como la precisión del sistema o la tasa positiva verdadera (TPR) para obtener los resultados (Hortal et al., 2015).

En el proyecto europeo de BioMot la UMH desarrolla una BCI que determina el nivel de concentración de la persona sobre la marcha (véase la figura 2), utilizando como métrica el promedio y la desviación de esta (Costa et al., 2016). También desarrollaron una BCI que detecta el cambio cerebral cuando aparece un obstáculo improvisado, utilizando para la evaluación de la evolución tasas de verdadero positivo, falso positivo y precisión (Salazar-Varas et al., 2015).

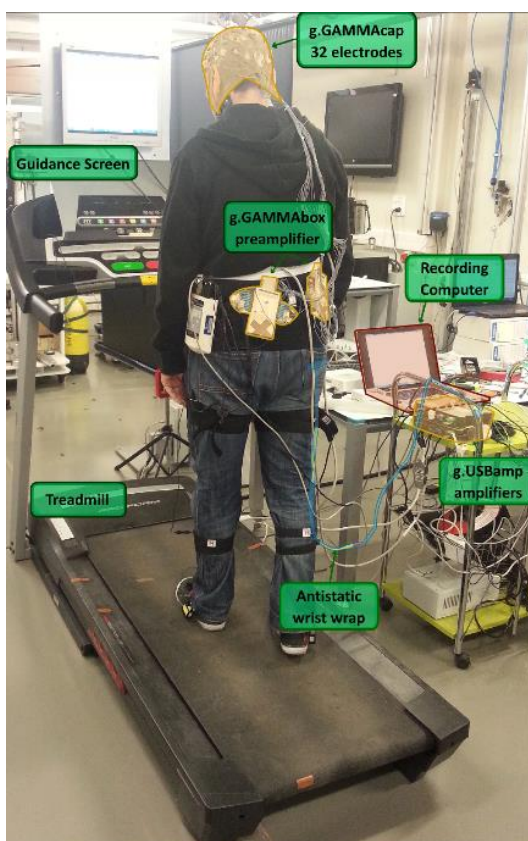


Figura 2: Proyecto BioMot. Fuente: Costa et al., 2016.

Según Barrios et al.(2017a), otra de las investigaciones relevantes es la que realiza el grupo UMA-BCI, perteneciente al grupo de investigación DIANA de la



Universidad de Valladolid

Universidad de Málaga, aportando a proyectos como BRAINS cuya finalidad es doble: por un lado identificar las tareas mentales más sencillas (Da Silva, Valero, Velasco y Ron, 2011) y por otro lado hacer uso de la realidad virtual y de sus ventajas como, la motivación de los sujetos, utilizando para determinar los resultados la métrica de porcentaje de error cometido (Ron-Angevin y Díaz-Estrella, 2009).

Otro proyecto en el que trabaja UMA-BCI es el de INCADI, que junto con BRAINS, tiene como objetivo el llegar a manejar una silla de ruedas con una BCI, dando lugar a un modelo probado en un entorno virtual con cuatro comandos de navegación que se basan en la clasificación de las tareas mentales (Velasco-Álvarez, Fernández-Rodríguez, Díaz-Estrella, Blanca-Mena y Ron-Angevin, 2018).

El último proyecto, LICOM, consiste en el desarrollo de un teclado virtual para mejorar los tiempos de escritura, siendo las métricas utilizadas el tiempo y el número de fallos cometidos (Da Silva-Sauer et al., 2016). Además, han realizado un estudio con resultados positivos para utilizar los sistemas BCI como tecnologías de asistencia para personas con demencia (Da Silva-Sauer, Torre-Luque, De La Silva y Fernández-Calvo, 2019).

El artículo de Barrios et al. (2017a) también menciona la investigación del Grupo de Ingeniería Biomédica de la Universidad de Valladolid (GIB-Uva) como las aplicaciones de ayuda doméstica o las nuevas metodologías del procesado de señales. Para el primer caso, se desarrolló una aplicación para la navegación web que obtuvo resultados positivos en personas con esclerosis múltiple, utilizando métricas cuantitativas (como la precisión) y métricas cualitativas de opinión del paciente (Martínez-Cagigal, Gomez-Pilar, Álvarez y Homero, 2017). Para el segundo caso, han realizado estudios sobre nuevos clasificadores (Martínez-Cagigal, Núñez y Homero, 2016).

El Brain-Computer Interface Lab de la Universidad de Granada (BCI Lab-UGR), también ha aportado a la investigación de los sistemas de BCI con estudios sobre la atención selectiva cuando hay diferentes fuentes de sonido (Minguillon, López-Gordo y Pelayo, 2016) con un análisis sobre el estado del arte de los sistemas de BCI (Minguillon, López-Gordo y Pelayo, 2017), entre otros.

Respecto al estado del arte de los sistemas de BCI, Barrios et al. (2017a) finalizan con la aportación del Instituto de Investigación en Ingeniería de Aragón y su desarrollo en una BCI, capaz de decodificar los intentos de movimiento del brazo en pacientes con un accidente cerebrovascular (ACV) (Antelis, Montesano, Ramos-Murguialday, Birbaumer y Mínguez, 2017).

A nivel internacional, se encuentra en Europa la aportación del Instituto de Psicología Médica y Neurobiología del Comportamiento, Universidad de

Tübingen, Alemania, el cual ha realizado varios estudios que demuestran la eficiencia de las BCI para la rehabilitación motora en pacientes que han sufrido un accidente cerebrovascular (Buch et al., 2008; Ramos-Murguialday et al., 2013; Lopez-Larranz, 2018).

2.1.4 Neuromodulación y Neuroestimulación

Estas tecnologías intentan fomentar la plasticidad neuronal modulando directamente el sistema nervioso para la rehabilitación de las funciones motoras. El principal problema de estas tecnologías es que aún no hay suficientes estudios que demuestren su eficacia en los pacientes.

Según Barrios et al. (2017a) destaca el Institut Guttmann en este área, el cual trabaja en distintas técnicas de terapias de estimulación cerebral no invasivas basadas en las técnicas de TMS (Transcranial Magnetic Stimulation) y tDCS (Transcranial Direct Current Stimulation) (véase la figura 3), como por ejemplo en terapias de rehabilitación en pacientes con una lesión medular para mejorar el patrón de marcha utilizando para determinar la evolución escalas específicas como la escala de Ashworth, la escala de Tardieu, y WISCI (Kumuru et al., 2016).

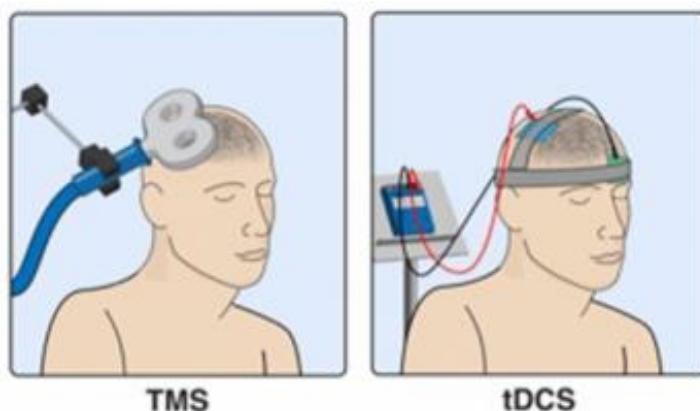


Figura 3: TMS y tDCS. Fuente: The brain simulator.

También han realizado estudios sobre cómo influyen estas técnicas en la espasticidad, utilizando como medidas para determinar la mejora las citadas anteriormente y añadiendo la escala analógica visual, la escala de frecuencia de espasmos de Penn, evaluación de Fugl-Mayer para los miembros superiores y la valoración de las actividades diarias (Gunduz, Kumuru y Pascual-Leone, 2014).

También, el UMH, trabaja en un sistema híbrido que combina la BCI y la tDCS para la rehabilitación de la marcha (Angulo, Rodríguez, Iáñez y Azorín, 2016).



2.1.5 Robótica de rehabilitación y asistencia

Este tipo de tecnología consiste en la interacción de un sistema robotizado con una persona. El desarrollo de robots para la asistencia de personas con diversidad funcional se encuentra en la actualidad en pleno auge. Estos dispositivos robóticos obtienen sus comandos de movimiento a través de la interacción física con el paciente. Un ejemplo conocido en la sociedad actual son los exoesqueletos robóticos, que permiten la movilidad de los miembros superiores e inferiores en personas con diversidad funcional y el aumento de su fuerza.

También hay que destacar los dispositivos robóticos destinados a la rehabilitación en clínicas, los cuales tienen como finalidad recuperar la movilidad y funcionalidad de los miembros superiores e inferiores dañados por algún trastorno o accidente.

En estas tecnologías, tanto para la detección de intención de movimiento como para registrar la actividad muscular y la fatiga, se puede utilizar el registro de la señal electromiográfica (EMG) que es la suma de la actividad eléctrica de las fibras musculares de un músculo.

El estudio de las señales EMG ha dado lugar a la técnica de electromiografía de alta resolución (HD-EMG) que permite obtener información sobre la posición de la articulación, cuánto dura la contracción y cuánta fuerza es capaz de ejercer (Barrios et al., 2017b).

El primer grupo español en utilizar y registrar esta técnica fue BIOART, de la universidad Politécnica de Cataluña (Mañanas, Rojas-Martínez y Alonso, 2016). Estudian la intención de movimiento en el miembro superior a través de esta técnica en pacientes con una lesión medular, utilizando para su estudio métodos estadísticos, como un análisis de varianza de medidas repetidas a los diferentes índices de rendimiento o la duración de la contracción (Jordanic, Rojas-Martínez, Mañanas y Alonso, 2016). La información resulta de interés a la hora de determinar la evolución muscular, así como para el control de los exoesqueletos (Barrios et al. ,2017b) (véase la figura 4).

La empresa Instead Technologies ha desarrollado el robot RoboTherapist3D para la rehabilitación de personas con diversidad funcional que permite realizar la terapia tanto de espaldas como sentado, algo muy significativo en las primeras fases de rehabilitación.



Figura 4: Detección de intención de movimiento y fatiga muscular a través de HD-EMG con un exoesqueleto. Fuente: Jordanic et al., 2016.

La empresa Technaid S.L ha desarrollado dos exoesqueletos (H1 y H2) para la rehabilitación de miembros inferiores, desatacando el segundo de ellos que actualmente se está comercializando (véase la figura 5). Se han realizado estudios sobre su efectividad teniendo buenos resultados, utilizando diversas pruebas para determinar la evolución, como el test de los 10 metros (10mWT), Test de los 6 minutos (6mWT) y el Test Time Up and Go (TUG) (Gil-Agudo et al., 2017)



Figura 5: Exo-H2. Fuente: Technaid.

El Centro tecnológico Tecnalía destaca por el desarrollo de ARMassist. Es un sistema robotizado diseñado para la rehabilitación de los miembros superiores tras un accidente cerebrovascular (véase la figura 6). Con él, se trabaja la pronosupinación, agarre y pinzamiento mediante terapias activas, asistidas y pasivas con la ayuda de la motivación de videojuegos. ARMassist puede instalarse en la vivienda del paciente y el terapeuta realizar el seguimiento de forma telemática, convirtiéndose en uno de los pocos dispositivos robotizados

de rehabilitación comerciales. Se ha hecho un estudio utilizando el lenguaje estadístico R (Tónic et al., 2017) en el que se observa la mejora de los pacientes.

Debido a su gran aceptación, TecNALIA ha lanzado la segunda generación: ARMassist 2.0. Esta nueva generación incorpora una nueva plataforma software que ayuda a los terapeutas en la evaluación de la evolución de los pacientes.



Figura 6: ARMassist. Fuente: Tónic et al., 2017.

También hay que mencionar los exoesqueletos destinados al control y supresión de los temblores en los miembros superiores del proyecto DRIFTS. De este proyecto destaca el exoesqueleto WOTAS (véase la figura 7), el cual se ha analizado la evolución en pacientes con Parkinson con métricas como el número de temblores, comparación entre la posición angular deseada y la realizada o la velocidad angular, obteniendo resultados muy favorables (Rocon et al., 2007).



Figura 7: Exoesqueleto WOTAS. Fuente: Rocon et al., 2007.



Universidad de Valladolid

El Instituto de las Tecnologías Avanzadas de la Producción (ITAP) de la Universidad de Valladolid, está desarrollando un exoesqueleto para la rehabilitación de mano en personas que han sufrido un accidente cerebrovascular, que ha recibido el nombre de RobHand (Cisnal et al., 2018) (véase la figura 8). Trabajan en el desarrollo de distintas terapias basadas en repetir movimientos a través de ejercicios programados o copiando los movimientos de una mano sana, y se realizan a través de juegos basados en puntuación para tener un mayor *feedback* con el paciente (Cisnal et al., 2019).



Figura 8: RobHand. Fuente: Cisnal et al., 2019.

Hasta aquí se ha presentado un análisis de las principales aportaciones españolas en la robótica para la rehabilitación y la asistencia. A continuación, se mencionarán las principales aportaciones y empresas internacionales de estos sistemas robotizados.

En Europa destaca la empresa de innovación Hocoma, la cual se ha extendido ya por todo el mundo convirtiéndose en líder mundial, Desarrolla productos tanto para la rehabilitación del miembro inferior (Erigo, Lokomat, Andago) como del superior (Plataforma Armeo).

Fue fundada en el año 1996 en Suiza por un ingeniero eléctrico, un ingeniero biomédico y un economista. En la actualidad Hocoma cuenta con 160 empleados en 27 sedes por todo el mundo. Su gran expansión se debe a la continua actualización de plataformas software (siendo la última HocoNet) así como la creación de nuevos dispositivos robotizados utilizando las últimas técnicas de rehabilitación. También trabajan continuamente en actualizaciones de imágenes y sonido, con la finalidad de aumentar la motivación y neuroestimulación del paciente.

El producto Lokomat (véase la figura 9) es un sistema robótico diseñado para la rehabilitación de la marcha en pacientes con lesiones neurológicas. Consta



Universidad de Valladolid

de un exoesqueleto para las extremidades inferiores que moviliza las articulaciones (cadera y rodillas) de forma coordinada para realizar el movimiento de caminar. También posee un sistema de suspensión de peso corporal que se ajusta en función de la dificultad de la terapia. Además, según las necesidades del paciente, se puede poner mayor o menor resistencia a la hora de caminar, todo mientras el sujeto ve en una pantalla la simulación de andar para así obtener un mayor *feedback*.

Se ha demostrado la eficacia y viabilidad de Lokomat en varios estudios (Riener, Lünenburger, Maier, Colombo y Dietz, 2010). La métrica que destaca y que utiliza este producto es la relación entre la fuerza que es capaz de hacer el paciente en las distintas articulaciones frente a la fuerza de apoyo que realiza el sistema robotizado.



Figura 9: Lokomat. Fuente: Hocoma.

Armeo es una plataforma que consta de distintos dispositivos robotizados (Armeo Spring, Armeo Power y Armeo Senso)

Armeo Power (véase la figura 10) es un exoesqueleto para la rehabilitación del miembro superior que trabaja en el espacio tridimensional y que interacciona con el paciente a través de videojuegos, permitiendo hacer ejercicios más intensivos y repetitivos disminuyendo la fatiga. Para la validación de este producto, el estudio de Calabrò et al. (2016) utiliza la escala Fugl Meyer, que valora el déficit y describe el margen de recuperación.



Figura 10: Armeo Power. Fuente: Hocoma.

Tyromotion es una empresa estadounidense que trabaja en el desarrollo de neurotecnología para la rehabilitación de miembros superiores, y ha exportado muchos de sus productos clínicos al continente europeo, como es el caso de Amadeo, un sistema robotizado para la rehabilitación de la mano muy similar a RobHand. Amadeo está formado por un conjunto de palancas en donde se posicionan los dedos (véase la figura 11); las palancas ofrecen una mayor o menor resistencia al movimiento en función de cómo se haya personalizado la terapia.

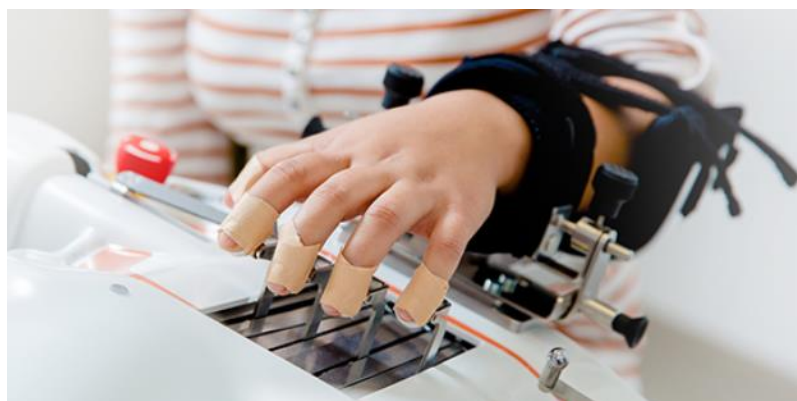


Figura 11: Amadeo. Fuente: Tyromotion.

Para determinar la evolución, Amadeo cuenta con distintos sensores que dan información sobre la velocidad, fuerza y señales EMG, dando lugar a métricas como la evolución de la fuerza, velocidad en los dedos, tiempos de ejecución o el rango del movimiento activo (ROM) (Gonzalo, 2014; Baldan et al., 2015).

Otro de los productos de Tyromotion para la rehabilitación es Pablo. Pablo consiste en un conjunto de dispositivos para realizar los ejercicios de terapia tradicionales para la rehabilitación, tanto del miembro superior como del

miembro inferior, empleando un método más dinámico a través de videojuegos y ejercicios de la vida cotidiana. Dependiendo del ejercicio realizado se obtienen distintas métricas. Algunas de ellas son el tiempo de ejecución, la fuerza en distintos agarres (lateral, pinzamiento, etc.) o la medición de la flexión extensión de las distintas partes del brazo (codo, hombro, muñeca).

Diego es un sistema robotizado diseñado por Tyromotion para la rehabilitación del miembro superior, que consiste en un juego de dos grúas que ayudan en función de la necesidad del paciente en mayor o menor medida. Además, utiliza gafas de realidad virtual para simular ejercicios de la vida cotidiana y conseguir un mayor número de repeticiones (véase la figura 12). Las métricas que utiliza Diego son algunas de las diseñadas en Pablo visto anteriormente, e incorpora otras nuevas como el rango de movimiento, velocidad horizontal, velocidad vertical, rotación del codo y del hombro y estabilidad del brazo (Knuth et al., 2017).



Figura 12: Diego. Fuente: Tyromotion.

Myomo es una empresa estadounidense que, colaborando con Harvard Medical School entre otros, ha desarrollado MyPro, un exoesqueleto que se utiliza en la vida cotidiana con la finalidad de ir rehabilitando el brazo en pacientes que han sufrido un accidente cerebrovascular. MyPro capta las señales EMG con métodos no invasivos y las procesa para detectar la intención de movimiento del paciente (véase la figura 13).



Figura 13: MyPro. Fuente: Myomo.

2.1.5.1 E2REBOT & PHYSIOBOT

Physiobot (véase la figura 14) es una plataforma robótica háptica diseñada por CARTIF para la rehabilitación neurológica en pacientes que han sufrido un accidente cerebrovascular, que impulsa el proceso de plasticidad neural y mejora la recuperación motora en los miembros superiores (Fraile et al., 2016).

Physiobot utiliza estrategias de control háptico basadas en el paradigma de control “Asistencia según sea necesario”, es decir, el robot es capaz de detectar cuándo el sujeto necesita ayuda para completar los movimientos de la terapia y en qué medida. Se puede dividir el robot en dos elementos fundamentales: la plataforma cartesiana y la pantalla.

La plataforma cartesiana es una estructura mecánica que, según Fraile et al. (2016), destaca por la reducción de los costos computacionales debido a una cinemática directa e inversa sencilla al tratarse de un diseño simple. Además, no es necesario añadir un modelo de compensación de gravedad, ya que por su estructura cartesiana estas fuerzas gravitatorias quedan estructuralmente compensadas. También hay que destacar que, debido a la similitud de la mecánica en ambos ejes, se puede lograr un tensor de inercia isotrópico.



Figura 14: Physiobot. Fuente: Villar et al. 2020.

Esta estructura mecánica cuenta con sensores de fuerza, velocidad, posición y par que nos permiten recopilar datos a tiempo real y almacenarlos en una base



de datos (BBDD) para posteriormente ser tratados. Esta obtención de datos de los sensores se realiza a una frecuencia de muestreo variable a través de un controlador FPGA (Field Programmable Gate Array).

La pantalla muestra al paciente los juegos virtuales, dando una realimentación al sujeto de la posición deseada o que debe alcanzar y la posición del robot. Además, cada juego tiene un marcador de puntuación que es mostrado por pantalla, con la finalidad de aumentar su motivación a la hora de realizar las terapias, reduciendo así el abandono de estas. Estos juegos virtuales basados en tareas de la vida cotidiana son necesarios para motivar al paciente y fomentar la reorganización neuronal, mejorando la recuperación motora perdida.

En la actualidad, se están desarrollando estos juegos en la realidad virtual con el fin de fomentar la inmersión del paciente y su motivación.

2.1.5.1.1 Tipo de terapias

Physiobot cuenta con tres tipos de terapias: Activo-asistida, pasivo-guiada y juegos AVD.

La terapia activo-asistida (Véase la figura 15) está formada por dos ítems: una mano y un ratón. La terapia consiste en que la mano, guiada por el *joystick*, consiga atrapar al ratón. Cuando la mano está muy cerca del ratón, esta se pondrá de color rojo y el sujeto aumentará la puntuación de su marcador. Cuando esto sucede, el ratón aumenta de velocidad y cambia de dirección, simulando la acción de escapar de la mano.

Esta terapia cuenta con cuatro niveles de dificultad distintos, que hacen que sea más fácil o difícil atrapar al ratón.

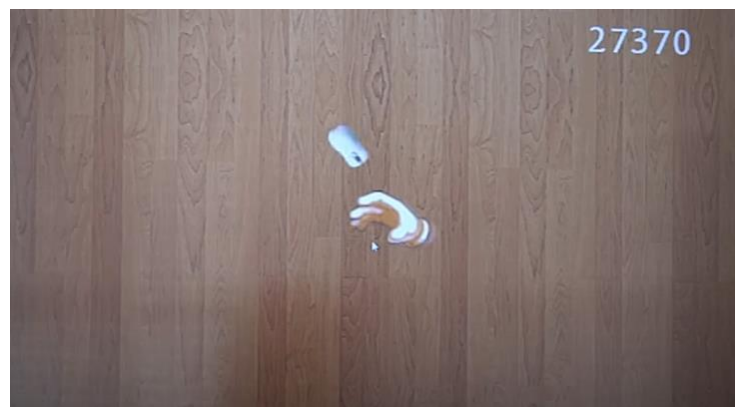


Figura 15: Terapia activo-asistida. Fuente: CARTIF.

La terapia pasivo-guiada consiste en grabar un recorrido para posteriormente seguirlo. Una mano de color blanca sigue la trayectoria grabada, y una vez que esta llegue al final del itinerario, la realizará en el sentido inverso. Para

aumentar la puntuación, el sujeto debe manipular la mano azul con el robot y seguir a la mano blanca (véase la figura 16).



Figura 16 Terapia pasivo-guiada Fuente: Zatarain, M., 2015.

Las terapias de tipo AVD son las denominadas Actividades de la Vida Diaria. Se basa en juegos que simulan actividades cotidianas, como por ejemplo, hacer la compra en un supermercado (véase la figura 17), ordenar el armario, colocar la cocina o cuidar la huerta, entre otras terapias (Fernández, 2018). También están las terapias más orientadas hacia niños, como la terapia basada en la serie de Pokémon.

Estas terapias son muy intuitivas y suelen consistir en realizar el desplazamiento de un objeto desde su origen al punto deseado, lo cual, según Fernández (2018), no requieren esfuerzos excesivos puesto que la libertad de movimiento es limitada.

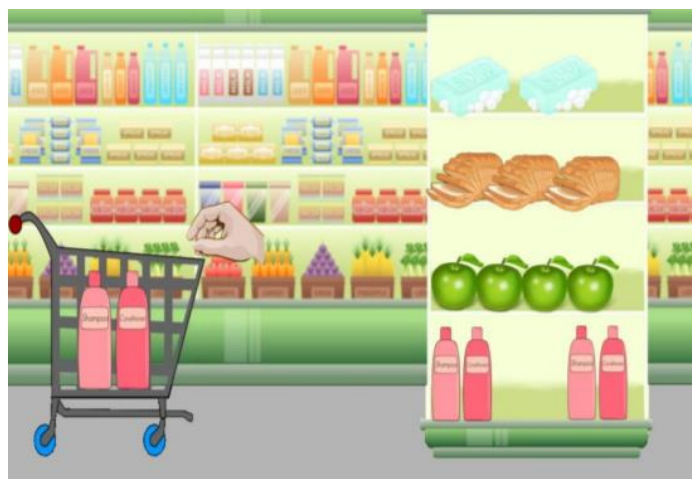


Figura 17: Terapia AVD de supermercado. Fuente: Fernández, 2018.

2.2 CARTIF

CARTIF es un centro tecnológico horizontal, privado y sin ánimo de lucro fundado en el año 1994 en la Escuela de Ingenieros Industriales de la UVa, en



Universidad de Valladolid

la que realizó su actividad hasta su traslado en 1999 al Parque Tecnológico de Boecillo, lugar en el que se encuentra actualmente.

Un centro tecnológico horizontal es aquel que no se especializa en un área concreta, sino que se desarrolla en distintos ambientes para afrontar los problemas desde diferentes perspectivas. Estos centros se caracterizan por su cooperación entre áreas, su agilidad y su innovación.

CARTIF se estructura en divisiones, y estas a su vez en áreas. El presente TFG se ha desarrollado en la División de Sistemas Industriales y Digitales en el área de Salud y Calidad de Vida.

En esta área, además del proyecto PHYSIOBOT, trabajan en la investigación e innovación de la robótica social (véase la figura 18) para el cuidado de las personas mayores, entre otros muchos proyectos.



Figura 18: Robot Copito. Fuente: El Día de Valladolid (2019).

2.3 BASES DE DATOS

2.3.1 Historia de las bases de datos

Toda la información que se genera día a día ha traído, a lo largo de la historia, el problema de cómo almacenarla y tratarla, teniendo ya sus orígenes en la antigüedad donde existían registros sobre los censos de población, de ganado etc. Pero la forma de consultar esta información era muy lenta, ya que no existía la ayuda de la computación y de la automática como se tiene ahora.

Ha sido mucha la evolución que se ha producido desde entonces (véase la figura 19), comenzando en 1884 con el tabulador magnético de Herman Hollerith, seguido en los años 50 de las cintas electromagnéticas, las cuales permitían almacenar la información en un orden secuencial, hasta llegar a la época de las revoluciones de las bases de datos. La primera de estas revoluciones ocurrió con la aparición de los discos duros, que dio lugar a las bases de datos jerárquicas; la segunda ocasionó las bases de datos de tipo relacional, la cual tuvo gran éxito en la industria, y la tercera revolución dio lugar a las bases de datos orientadas a objetos, de donde surge EXCEL.

En la actualidad, aunque las bases de datos tradicionales siguen siendo el elemento principal de las empresas, está cada vez en mayor auge el Big Data, el cual trata cantidades, velocidades y variedades de datos que las bases tradicionales no pueden.

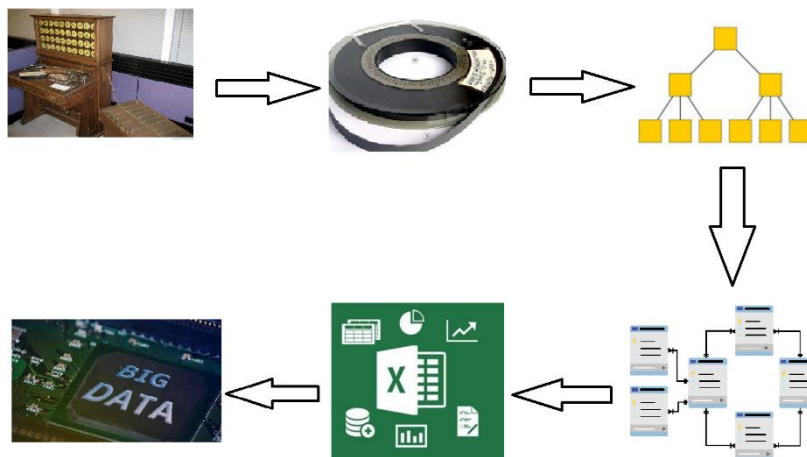


Figura 19. Historia de las bases de datos. Fuente: Elaboración propia.

2.3.2 Concepto básico

Una buena definición de base de datos (BBDD) es la que proporciona Date (2001):

Un sistema de bases de datos es básicamente un sistema computarizado para llevar registros. Es posible considerar la propia base de datos como una especie de armario electrónico para archivar, es decir, es un depósito o contenedor de una colección de archivos de datos computarizados. (p.2)

Entonces se puede concebir una BBDD como un conjunto de datos que están relacionados entre sí y pertenecen a un mismo contexto. Sobre ellas se pueden realizar operaciones de insertar más datos, consultarlos, modificarlos, ordenarlos o eliminarlos. Quien se encarga de proporcionar la forma para realizar estas operaciones es un Sistema de Gestor de Bases de Datos (SGBD) que, según Silberschatz et al. (2002), “El objetivo principal de un SGBD es proporcionar una forma de almacenar y recuperar la información de una base de datos de manera que sea tanto práctica como eficiente” (p.1)

Hay muchos sistemas comerciales de gestión de bases de datos, entre los que cabe destacar como los más utilizados: MySQL, ORACLE, SQL Server y Access. En el presente TFG se va a trabajar con SQL Server, ya que es el SGBD que más se ajusta a las necesidades de este, debido a que tiene un lenguaje intuitivo (razón por la que se descarta a MySQL), tiene una versión gratuita (descartando así ORACLE), y no tiene las limitaciones de búsqueda que se encuentran en Access. Además, es el entorno de trabajo que utiliza CARTIF.



La principal desventaja que tiene SQL Server es que su utilización requiere el uso de una gran cantidad de memoria RAM, por lo que es recomendable tener un equipo con un mínimo de 8GB de RAM. Además, la eficacia de SQL Server es bastante inferior a la de ORACLE.

Respecto a la arquitectura, hay que tener en cuenta que la BBDD y el software que se conecta a ella no suelen estar alojados en el mismo equipo. Es decir, una persona que utilice PHYSIOBOT en un hospital de Zamora se conecta a la BBDD de Boecillo.

Por consiguiente, se puede distinguir entre servidor, que en este caso sería la BBDD, y cliente, que será el usuario remoto (véase la figura 20).

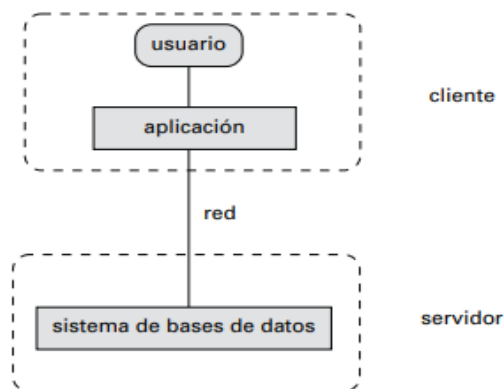


Figura 20: Conexión a BBDD. Fuente: Silberschatz et al., 2002.

2.3.3 Modelo de datos

Cuando se habla de un modelo de datos, se refiere a la estructura lógica de la BBDD, es decir, la forma en la que la información es utilizada y almacenada en ella. Otra definición del modelo de datos es la manera en la que se estructura la relación de los datos de un mismo ámbito.

Hay una gran variedad de modelos de datos, siendo los fundamentales tres: el modelo entidad-relación, el modelo relacional y el modelo jerárquico.

2.3.3.1 Modelo entidad-relación

Se habla de una entidad como un ítem que tiene distintas propiedades de interés y que se puede diferenciar de otras entidades, como lo serían una persona o un coche. Cada propiedad que se asigna a una entidad se denomina atributo. Una entidad puede tener varios atributos y así se define en la BBDD. En el caso de persona, por ejemplo, se le pueden asignar los atributos de DNI,

nombre, edad, etc. En el caso del coche sus atributos pueden ser el color, la marca, el modelo, la matrícula etc.

Es importante que se diferencien unas entidades de otras con los mismos atributos. Es por ello por lo que se añaden atributos que sean únicos para una entidad. En los ejemplos planteados anteriormente, dos personas pueden tener la misma marca de coche, el mismo color y el mismo modelo, por eso se añade el atributo matrícula que permite diferenciar los dos coches.

Una relación es la asociación entre dos entidades; la relación que hay entre persona y coche es la de poseer o tener.

Este modelo se puede representar gráficamente mediante un diagrama de E-R, en el cual los atributos están determinados por elipses, las relaciones por rombos y las entidades por cuadrados, como se ve en la figura 21.

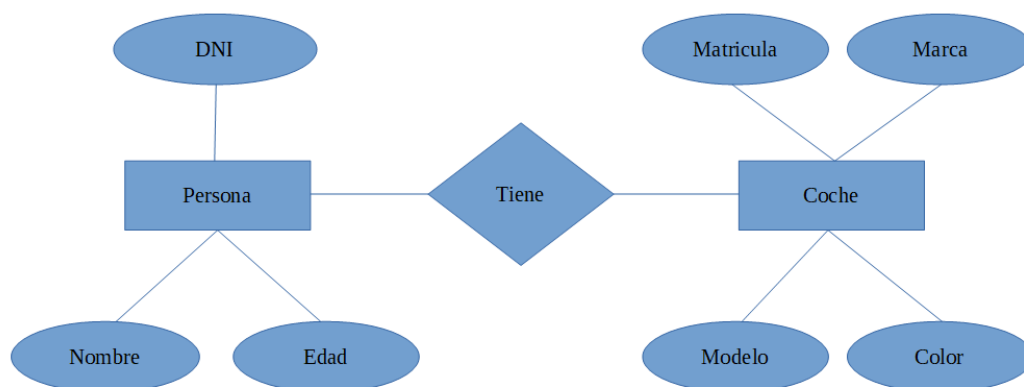


Figura 21: Diagrama E-R. Fuente: Elaboración propia.

2.3.3.2 Modelo jerárquico

Consiste en una estructura de árbol y sigue el modelo de padre/hijo, es decir, todos los hijos solo pueden tener un único padre, pero un padre puede tener varios hijos.

Supongamos como ejemplo una ciudad, donde cada ciudad tiene sus barrios, cada barrio solo tiene una ciudad pero tiene distintas calles (véase la figura 22)

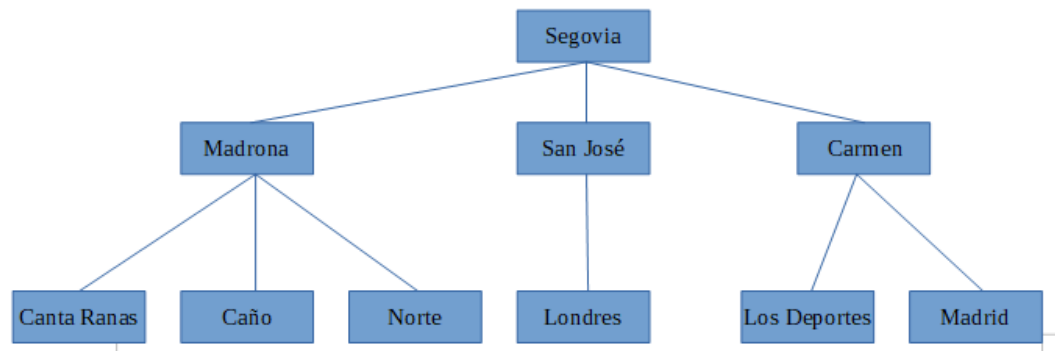


Figura 22: Diagrama jerárquico. Fuente: Elaboración propia.

2.3.3.3 Modelo relacional

Es el modelo más utilizado en la actualidad. Los datos se representan mediante tablas y estas se relacionan entre ellas. Es el modelo con el que se trabaja en el presente TFG.

Para realizar el diseño de una base de datos que utiliza el modelo relacional, es habitual empezar con el diagrama del modelo entidad-relación, y a partir de él pasar al modelo relacional.

Las tablas, relacionadas entre sí a partir de datos comunes (véase la figura 23), están divididas en filas y columnas. Cada tabla representaría una entidad, las columnas están asociadas a los atributos y las filas a cada uno de los ítems de dicha entidad. Estos ítems son tuplas con tantos elementos como atributos tiene una entidad.

Como se mencionó en el modelo de entidad-relación, es necesario disponer de un atributo que permita distinguir una tupla de otra. Según Quiroz (2003) "Para distinguir una tupla de otra, se recurre al concepto de "llave primaria", o sea un atributo o conjunto de atributos que permiten identificar unívocamente una tupla en una relación" (p.55)

La llave primaria nunca puede ser un valor nulo. Un atributo que cumple la condición que permite distinguir una tupla del resto se denomina llave candidata, pero solo un único atributo puede ser llave primaria.



<i>id-cliente</i>	<i>nombre-cliente</i>	<i>calle-cliente</i>	<i>ciudad-cliente</i>
19.283.746	González	Arenal	La Granja
01.928.374	Gómez	Carretas	Cerceda
67.789.901	López	Mayor	Peguerinos
18.273.609	Abril	Preciados	Valsain
32.112.312	Santos	Mayor	Peguerinos
33.666.999	Rupérez	Ramblas	León
01.928.374	Gómez	Carretas	Cerceda

(a) La tabla *cliente*

<i>número-cuenta</i>	<i>saldo</i>
C-101	500
C-215	700
C-102	400
C-305	350
C-201	900
C-217	750
C-222	700

(b) La tabla *cuenta*

<i>id-cliente</i>	<i>número-cuenta</i>
19.283.746	C-101
19.283.746	C-201
01.928.374	C-215
67.789.901	C-102
18.273.609	C-305
32.112.312	C-217
33.666.999	C-222
01.928.374	C-201

(b) La tabla *impositor*

Figura 23: Tablas según el modelo relacional. Fuente: Silberschatz et al., 2002.

Al conjunto de los valores permitidos en un atributo se le llama dominio. El diseño de la BBDD de PHYSIOBOT no es muy restrictivo respecto al dominio, ya que solo se especifica el tipo de dato, es decir, si se trata de un entero, de un carácter etc., No obstante, los dominios pueden ser tan restrictivos como se desee, estableciendo por ejemplo un número máximo de caracteres o teniendo solo dos posibles valores.

La mejor forma de comprender una base de datos relacional es a través del diagrama relacional, el cual nos presenta las tablas con los distintos atributos y la forma en la que se relacionan entre ellas (véase la figura 24).

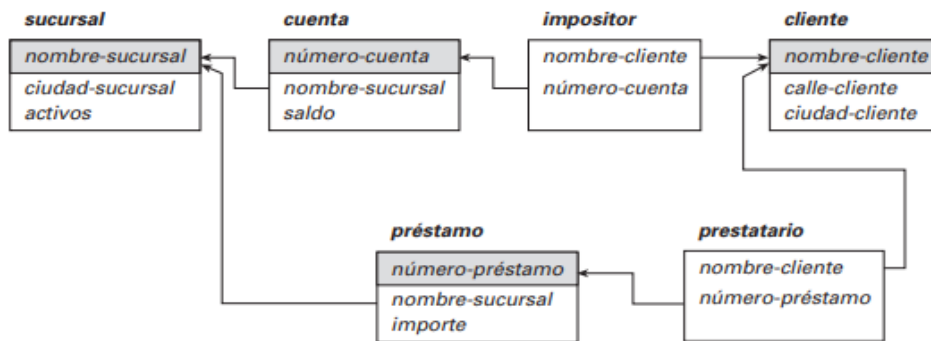


Figura 24: Diagrama relacional. Fuente: Silberschatz et al., 2002.



3. ENTORNO DE TRABAJO

3.1 VISUAL STUDIO

Microsoft Visual Studio es un entorno de desarrollo integrado (IDE), en el que Microsoft ha centralizado todas las herramientas para el desarrollo de aplicaciones.

Se trata de un entorno de programación muy utilizado a nivel mundial que se ha lanzado para los principales sistemas operativos (Windows, Linux y macOS). Cuenta con más de 20 años de antigüedad, durante los cuales se han lanzado 16 versiones, lo que hace de Visual Studio una herramienta muy versátil.

Es compatible con una gran variedad de lenguajes de programación como C#, C++, R, .NET, Python, Visual Basic, Java etc. y con modelos de desarrollo web como ASP.NET, MVC, etc.

Es una herramienta que permite desarrollar desde aplicaciones de escritorio o de dispositivo móvil, hasta plataformas web que sean compatibles con .NET. Incluso es utilizado como herramienta de analítica de datos (uso que se da en el presente TFG) y *big data*.

Los proyectos de Visual Studio están basados en soluciones, entendiéndose por solución al conjunto de aplicaciones que trabajan en el mismo proyecto.

En el presente TFG se han desarrollado dos soluciones, las cuales se han llamado TFG_ señales y Evolucion.

3.2 C#

C# es un lenguaje de programación diseñado por Microsoft que se considera el sucesor de C++. Se trata de un lenguaje orientado a objetos que utiliza una sintaxis muy similar a C/C++ y un modelo de objetos muy similar a Java (Plataforma .NET).

C# utiliza un modelo de propiedades, métodos y eventos para cada uno de los paquetes independientes que se crean para realizar una determinada función.

La ejecución de los programas de C# se realiza siguiendo una arquitectura. Primero los ficheros con el código fuente escrito en C# se compilan según el estándar internacional de *Common Language Infrastructure* (CLI), dando lugar a un ejecutable que se denomina ensamblador.

Cuando se ejecuta el ensamblador, este es cargado en el *Common Language Runtime* (CLR) (un sistema de ejecución virtual integrado en Windows) de .NET Framework, que junto con las bibliotecas de este dan lugar a las instrucciones en código máquina (véase la figura 25).

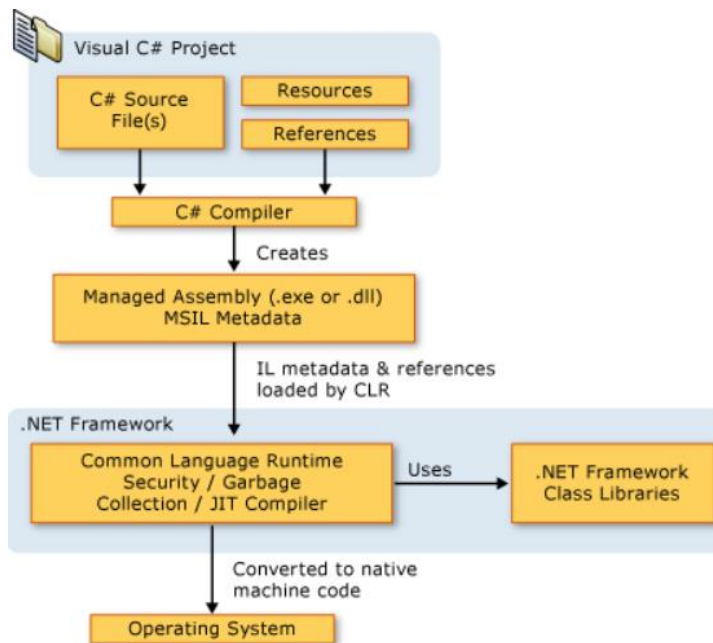


Figura 25: Arquitectura .NET Framework Fuente: Introducción al lenguaje C# y .NET Framework. Microsoft Docs.

3.3 R

R es un lenguaje de programación enfocado al análisis de datos, convirtiéndose en la actualidad en un referente mundial. Debido a su extensa comunidad y a su filosofía de *Open Source*, se desarrollan continuamente nuevos paquetes que son accesibles a todo el mundo de forma gratuita.

Es muy utilizado en la investigación científica debido a su gran cantidad de herramientas estadísticas (modelos lineales, no lineales, tests, series, etc.), la alta calidad de sus gráficos y su bajo consumo de recursos.

Los comandos de R se ejecutan directamente sin necesidad de ser compilados, siendo muy similar a Matlab. Muchos de estos comandos implican realizar operaciones sobre varios datos sin la necesidad de crear un bucle.

El principal inconveniente es que la interacción con los softwares de R se realiza mediante una consola, es decir, no dispone de una interfaz gráfica. Esto hace que su utilización se realice a través de comandos, y por consiguiente es necesario conocer su lenguaje.

Las métricas desarrolladas en PHYSIOBOT estaban escritas en lenguaje R (Zatarain, 2015), y como ya se ha explicado, no se implantaron en los distintos hospitales donde se encuentra dicho dispositivo.

R se utiliza en el presente TFG para comprobar el uso de las métricas antes de implantarlas en el lenguaje C#.



3.4 SQL SERVER

Como se mencionó en el apartado de BBDD, SQL Server es un SGBD desarrollado por Microsoft que se utiliza para la gestión de BBDD de tipo relacional.

Se trata de una plataforma que da servicio a aplicaciones a través de la red o en el mismo equipo. Una de sus principales ventajas es su entorno gráfico e intuitivo para la administración de la BBDD.

En resumen, se puede decir que SQL Server sirve para el almacenamiento y recuperación de datos de forma segura a través de aplicaciones.

La BBDD de PHYSIOBOT con la que se trabaja en este TFG está implantada con la versión de SQL Server 2012, mientras que nosotros vamos a utilizar SQL Server 2017 (versión gratuita). Para poder trabajar con esta versión es necesario hacer una migración de la BBDD de SQL Server 2012 a una BBDD local de SQL Server 2017.

En concreto se van a trabajar dos BBDD: la primera son datos de prueba realizados por los usuarios de CARTIF, utilizada para probar el software sin riesgo de perder datos de interés. La segunda, contiene los datos de las terapias realizadas con PHYSIOBOT por pacientes con ACV.

3.5 LINQ

Las BBDD que vamos a utilizar pueden contener millones de datos. Para realizar una métrica concreta no se necesita trabajar con todos ellos, sino solo con aquellos que sean de interés. A esta selección y obtención de datos se le denomina consulta.

Dependiendo del origen de datos (SQL, XML, etc.) las consultas tienen distintas estructuras, lo que obligaría a los programadores a aprenderlas. De este problema, junto con la dificultad de realizar las consultas de estos orígenes de datos que suele implicar el uso de bucles, lo que da lugar a una peor legibilidad del código, surge LINQ.

LINQ es una herramienta de .NET que utiliza el mismo patrón para realizar consultas, independientemente del origen de datos. Su sintaxis es muy similar a la de SQL dando mayor legibilidad al Código.

El principal inconveniente es que tiene un peor rendimiento que si se utilizan los bucles directamente. Las ventajas que nos ofrece LINQ son mucho mayores a los milisegundos que se pierden, ya que en este TFG el tiempo no es un factor crítico y las consultas son mucho más sencillas.

Para realizar una consulta, LINQ siempre utiliza la misma metodología: primero obtiene el origen de datos (en este caso SQL Server), a continuación se crea la consulta y finalmente se ejecuta (véase la figura 26).

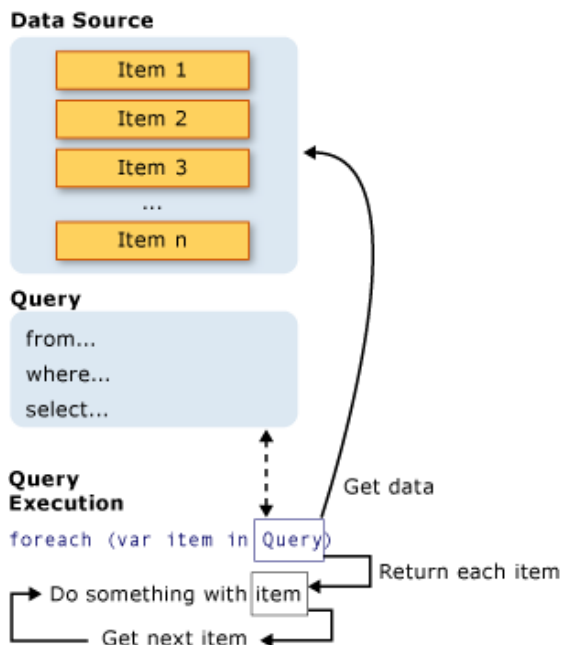


Figura 26: Estructura de la operación de consulta con LINQ. Fuente: Introducción a las consultas LINQ (C#). Microsoft Docs.

En la creación de la consulta se especifican tres cláusulas: la cláusula “*from*”, que especifica el origen de datos, la cláusula “*where*”, que especifica el filtro y la cláusula “*select*”, que especifica el tipo de dato que se selecciona.

Además, hay otros parámetros opcionales, como por ejemplo “*orderby*”, que indica cómo tienen que ir ordenados los datos.

La ejecución de la consulta no se realiza justo después de la creación, sino que esta se puede aplazar en el Código tanto como se desee.

3.6 MVC

MVC son las siglas de Modelo-Vista-Controlador (véase la figura 27), un estilo de arquitectura software que organiza los componentes que hay en el sistema y la relación que existe entre ellos. Esta estructura consta fundamentalmente de tres partes:

- El **Modelo** es la capa encargada de gestionar tanto el almacenamiento de datos como su recuperación. Según la solución planteada en el presente TFG, el modelo corresponde a la estructura y gestión de la BBDD de SQL Server 2017.

- El **Controlador** responde a los eventos, los cuales la mayoría los produce el usuario (un clic, cambiar un texto, etc.). El controlador es el encargado de solicitar la información al Modelo, tratarla y enviársela a la vista, actualizando la información que esta muestra por pantalla.
- La **Vista** es el componente que muestra al Modelo una vez que ha sido tratado por el controlador. Es decir, la vista se trata de una interfaz de usuario que solo muestra resultado, no realiza ninguna operación sobre los datos.

Para el desarrollo de la vista, en el presente TFG se ha hecho uso de la herramienta de Windows Forms de la plataforma .NET Framework, la cual nos da acceso a los elementos de la interfaz de Windows (botones, graficas, desplegables etc.).

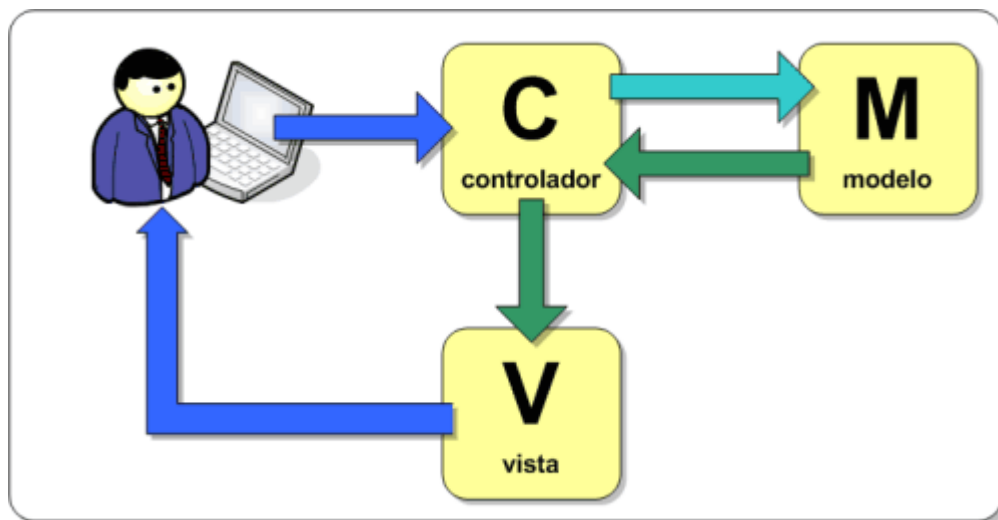


Figura 27: MVC. Fuente: Campus MVC.



4. DESARROLLO DEL SOFTWARE

4.1 ARQUITECTURA DEL SOFTWARE

Como ya se mencionó, el software se divide en dos soluciones: EvolucionPacientes y TFG_ñeales. Ambos siguen el estilo de arquitectura MVC, donde, en resumen, el modelo es la BBDD de PHYSIOBOT desarrollada por CARTIF, el controlador es el tratamiento de datos y las métricas desarrolladas por CARTIF y Zatarain (2015) y la interfaz es desarrollada con Windows Forms.

La vista es muy similar en ambas soluciones, con la finalidad de facilitar su aprendizaje. Además, se ha tratado de crear una vista con apariencia moderna para fomentar la motivación de los terapeutas y que sea lo más intuitiva posible.

El controlador de cada una de las soluciones se ha desarrollado siguiendo la misma estructura: un formulario, dos clases estáticas y una clase no estática. La diferencia entre una clase estática con una no estática es que la primera de ellas no puede crear instancias, es decir, no se pueden crear variables de clase que sirvan como almacenamiento de datos.

El **formulario** es el encargado de actuar ante los distintos eventos (dar a un botón, seleccionar un desplegable, etc.), ejecutando los métodos de las clases estáticas en función del evento. Además, en él están las constantes globales, como el nombre de la BBDD o la ruta de almacenamiento del informe,

La clase estática **CargaDatos** contiene los métodos que cargan la información en la vista en bruto, sin ningún tratamiento. Es utilizada para cargar los desplegables, las listas etc., ya sea con consultas breves a la BBDD, sin modificar los datos de origen, o con código fuente.

La clase estática **Metodos** es la que contiene todos los métodos que realizan las métricas. Estos métodos realizan el tratamiento de la BBDD, guarda la información en la clase no estática Series y muestran los resultados por pantalla o en un informe con formato PDF, según lo desee el terapeuta. Además, esta clase tiene métodos para mejorar la legibilidad del software, evitando repetir el mismo código, como el método GenerarCabecero que se repite por cada página de informe.

La clase no estática **Series** está formada por múltiples variables de clase para almacenar el resultado del tratamiento de las clases estáticas y, posteriormente, ser representado en una gráfica del programa o del informe. En el caso de ser el informe, se acudirá también a esta clase para mostrar los valores más significativos: primer valor, mínimo valor, máximo valor, etc.

4.2 BBDD DE PHYSIOBOT

El diagrama relacional de la BBDD de PHYSIOBOT (véase la figura 28) muestra las principales entidades con cada uno de sus atributos. El atributo que actúa como llave primaria es indicado con un símbolo de una llave de color amarilla en su lateral izquierdo.



Figura 28: BBDD de PHYSIOBOT. Fuente: Elaboración propia.

En este apartado se realiza un análisis de las entidades y sus respectivos atributos que van a ser utilizados para el análisis de la BBDD. Son cuatro las entidades con las que se trabaja: TerapiaBM, PacientesBM, Muestra y Nom_Terapias.



4.2.1 TerapiaBM

Esta tabla contiene las principales propiedades de una terapia (véase la tabla 1) como son por ejemplo la fecha, la dificultad, el marcador etc. Además, esta tabla es la unión entre el resto de las tablas debido a que contiene atributos comunes con ellas. Es decir, no podemos establecer una relación directa entre la tabla PacientesBM y Muestra, pero sí una relación indirecta a través de la tabla TerapiaBM, por sus atributos de IdPaciente (común entre PacientesBM y TerapiaBM) y IdTerapia (común entre TerapiasBM y Muestra).

	IdTerapia	Fecha	IdPaciente	IdExperimento	Dificultad	TipoTerapia	Completada	Observaciones	MarcadorTotal
1	1	2014-04-28 10:49:33.907	1	0	2	1	NULL	NULL	NULL
2	2	2017-03-14 10:46:33.753	1	0	1	3	NULL	NULL	NULL
3	1002	2017-03-14 11:00:43.560	1	0	1	3	NULL	NULL	387
4	1003	2017-03-14 11:03:17.527	1	0	2	1	1	La resolucion de la terapia fue de 800x450	960
5	1004	2017-03-14 11:05:34.147	1	0	1	1	1	La resolucion de la terapia fue de 800x450	30370
6	1005	2017-03-14 11:10:26.100	1	0	NULL	2	1	NULL	71
7	1006	2017-03-14 11:10:41.097	1	0	NULL	2	1	NULL	87
8	1007	2017-03-16 11:11:32.983	1	0	1	3	NULL	NULL	0
9	1008	2017-03-16 11:13:34.863	1	0	2	1	1	La resolucion de la terapia fue de 800x450	170
10	1009	2017-03-16 11:14:39.163	1	0	NULL	2	1	NULL	92
11	1010	2017-03-16 11:15:54.330	1	0	1	3	NULL	NULL	2487
12	1011	2017-03-16 12:31:52.137	1	0	NULL	2	1	NULL	128
13	1012	2017-03-16 12:34:04.533	1	0	NULL	2	1	NULL	107
14	1013	2017-03-16 12:36:12.327	1	0	2	1	1	La resolucion de la terapia fue de 1000x600	440

Tabla 1: TerapiaBM. Fuente: Elaboración propia.

Los atributos con los que se van a trabajar en esta terapia son:

- **IdTerapia:** es la llave primaria de la tabla. Se utiliza como identificador de cada una de las terapias. Cuando un paciente realiza una terapia se le asigna el próximo número entero que está libre.
- **Fecha:** la fecha nos indica el año, el mes, el día y la hora a la que se inició la terapia.
- **IdPaciente:** este atributo tiene una doble finalidad: por un lado, sirve como identificador de los distintos pacientes (llave primaria de la tabla PacientesBM). Por otro lado, es necesario asignar un número a los pacientes para mantener su anonimato, cumpliendo así la ley de protección de datos.
- **Dificultad:** en las terapias de tipo 1 (activo-asistida) tenemos tres niveles de dificultad, siendo el primer nivel el más sencillo y el tercero el más completo. Este atributo muestra el nivel con el que se realizó la terapia.
- **TipoTerapia:** indica el tipo de terapia que se ha utilizado, siendo la 1 para la terapia de tipo activo-asistida, la 2 es la terapia de tipo pasivo-guiada y de 3 a 11 están reservadas para las terapias de tipo AVD.
- **MarcadorTotal:** indica la puntuación final que ha hecho el paciente en la terapia.



4.2.2 PacientesBM

Esta tabla contiene todos los datos privados del paciente (véase la tabla 2) como su nombre, apellidos, edad, sexo, discapacidad etc.

	IdPaciente	Nombre	Apellidos	Edad	Discapacidad	Sexo
1	1	Prueba 1	Prueba 1	23	Brazo	Hombre

Tabla 2: PacientesBM. Fuente: Elaboración propia.

En nuestro software solo aparecerá el atributo IdPaciente para mantener el anonimato, pero el terapeuta con ese número puede acceder a la información de la persona que realiza la rehabilitación.

4.2.3 Muestra

Es la tabla que más información contiene. En ella, cada pocos milisegundos (véase la columna de Tiempo de la tabla 3), se guardan los datos registrados por los sensores y los obtenidos de la pantalla (véase la tabla 3), llegando a superar 10 millones de filas de información. Esta será la tabla más explotada en el análisis de datos de este TFG.

	IdMuestra	Tiempo	PosicionX	PosicionY	FuerzaX	FuerzaY	IdTerapia	FuerzaAsistiva	VelocidadX	VelocidadY	PuntoDeseadoX	PuntoDeseadoY
748	8748	00:00:15.4540649	412.00	170.00	11.00	5.00	1002	9.00	247.00	77.00	1022.00	235.00
749	8749	00:00:15.4720659	422.00	173.00	9.00	4.00	1002	9.00	251.00	77.00	1022.00	235.00
750	8750	00:00:15.4900670	422.00	173.00	9.00	4.00	1002	9.00	251.00	77.00	1022.00	235.00
751	8751	00:00:15.5080680	422.00	173.00	9.00	4.00	1002	9.00	251.00	77.00	1022.00	235.00
752	8752	00:00:15.5250690	433.00	176.00	10.00	0.00	1002	9.00	255.00	61.00	1022.00	235.00
753	8753	00:00:15.5430700	433.00	176.00	10.00	0.00	1002	9.00	255.00	61.00	1022.00	235.00
754	8754	00:00:15.5600710	443.00	178.00	7.00	0.00	1002	9.00	251.00	36.00	1022.00	235.00
755	8755	00:00:15.5790721	443.00	178.00	7.00	0.00	1002	9.00	251.00	36.00	1022.00	235.00
756	8756	00:00:15.5960730	448.00	179.00	7.00	4.00	1002	9.00	243.00	28.00	1022.00	235.00
757	8757	00:00:15.6140741	448.00	179.00	7.00	4.00	1002	9.00	243.00	28.00	1022.00	235.00

Tabla 3: Muestra. Fuente: Elaboración propia.

Los atributos con los que vamos a trabajar son:

- **Tiempo:** es el valor desde que se inició la terapia hasta que son leídos los sensores. Los intervalos no son constantes, pero aproximadamente cada 0,025 segundos los datos quedan registrados. Sigue el formato 00:00:00.000000, por lo que es necesario transformar los minutos y horas en segundos (multiplicando por 60 y 3600 respectivamente).
- **PosicionX:** es la posición del joystick en el eje x. Se mide en milímetros, por lo que a la hora de operar dividiremos entre mil para obtener la medida en metros.
- **PosicionY:** es la posición del joystick en el eje y. También se mide en milímetros, por lo que recibe el mismo tratamiento que la posición en x.
- **FuerzaX:** es el valor de la fuerza en el eje x. su valor viene expresado en función del tipo de terapia: para las terapias de tipo 1 viene expresado en Newtons, mientras que en el resto de los tipos viene expresado en



kg*mm/s², por lo que es necesario dividir entre mil para obtener el valor en Newtons.

- **FuerzaY:** es el valor de la fuerza en el eje y. Su valor viene expresado igual que la FuerzaX, por lo cual se le realiza el mismo tratamiento.
- **FuerzaAsistiva:** es el valor de una variable que permite el cálculo de a fuerza que ayuda al paciente.
- **VelocidadX:** es la velocidad del *joystick* en el eje x. Viene expresada en mm/s, por lo que es necesario dividirla por mil para obtener m/s.
- **VelocidadY:** es la velocidad del *joystick* en el eje y. Viene expresada igual que la VelocidadX, por lo cual recibe el mismo tratamiento.
- **PuntoDeseadoX:** es el punto que debe alcanzar el paciente con el *joystick* en el eje x y que se muestra en la pantalla. Viene expresado en milímetros, de modo que para el tratamiento de datos se divide por mil para obtener metros.
- **PuntoDeseadoY:** es el punto que debe alcanzar el paciente con el *joystick* en el eje y, y que se muestra en la pantalla. Viene expresado igual que el PuntoDeseadoX y recibe el mismo tratamiento que este.

4.2.4 Nom_Terapias

Debido a que se incorpora un nuevo software a PHYSIOBOT, es necesario una tabla actualizada que indique que tipo de terapia se está utilizando. Nom_Terapias permite relacionar el TipoTerapia utilizado en el tratamiento de las otras tablas con el nombre de la terapia que simboliza.

	IdNombres	TipoTerapia	Descripcion
1	1	1	Activo Asistido
2	2	2	Pasivo
3	3	3	AVD Biblioteca
4	4	4	AVD Telefono
5	5	5	AVD Imagenes
6	6	6	GRADIOR ACTIVO
7	7	7	GRADIOR PASIVO

Tabla 4: Nom_Terapias. Fuente: Elaboración propia.

El atributo **Descripcion** es el nombre de la terapia que estamos utilizando. A nivel de código se utiliza el atributo TipoTerapia, pero a nivel visual se utiliza Descripcion.

4.3 VISTA

Como ya se ha mencionado, el desarrollo de la vista se ha realizado con la herramienta de Windows Forms, la cual nos permite realizar el diseño implementando directamente las formas sin necesidad de realizar el código.

El código de la vista se genera automáticamente a medida que vamos situando los distintos elementos del formulario (botones, desplegados, imágenes, gráficas etc.) y modificando sus propiedades (nombre, ancho, altura, etc.). Este código se genera en el archivo con terminación .Designer.cs, el cual podemos ver en el anexo de los códigos de las soluciones.

La interfaz desarrollada pretende ser:

- **Intuitiva:** los elementos de la interfaz que interactúan con el usuario están bloqueados. Estos se van desbloqueando uno a uno a medida que el terapeuta completa el último elemento que se desbloquea. Para aprender el funcionamiento de las aplicaciones se ha creado una guía para usuarios (véase el ANEXO I: Guía de usuario).
- **Visual:** la interfaz pretende ser estética con la finalidad de motivar al usuario.
- **Actualizada:** utilización de los logotipos actualizados de CARTIF y PHYSIOBOT.

La interfaz tiene dos modos: El modo de inicio, que se ve nada más iniciar el programa (véase la figura 29), y el modo de graficar, que es cuando se tienen los datos rellenos y se realiza la gráfica (véase la figura 30). En la barra de herramientas, la aplicación se reconoce por la letra griega Φ .



Figura 29: Vista del inicio de las aplicaciones. Fuente: Elaboración propia.

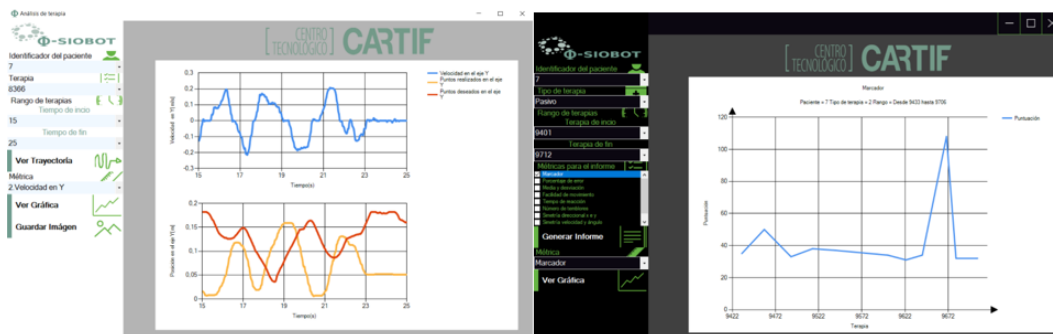


Figura 30: Vista de las aplicaciones graficando. Fuente: Elaboración propia.

4.4 EJES DE REFERENCIA

Es necesario establecer una referencia para realizar el tratamiento de datos y que esta sea la misma, tanto para las posiciones del *joystick* como para las posiciones de la pantalla.

Las posiciones del *joystick* se almacenan siempre de igual forma, siguiendo la misma referencia. El origen se encuentra en la esquina inferior izquierda de PHYSIOBOT, teniendo el eje X la dirección del lado más largo y el eje Y la del lado más corto (véase los ejes de color amarillo de la figura 29).

Para el caso de la pantalla, la posición del origen y la orientación de los ejes depende del tipo de terapia que se esté utilizando. Esto quiere decir que las posiciones deseadas se almacenan en función del tipo de terapia.

Para realizar el tratamiento de datos se establecen unos ejes de referencia para los puntos deseados (posiciones de la pantalla), que son independientes del tipo de terapia. Esta referencia creada tiene su origen en la esquina inferior de la pantalla, siendo su eje x el lado largo de la pantalla y su eje y su lado corto (véase los ejes de color rojo de la figura 31).

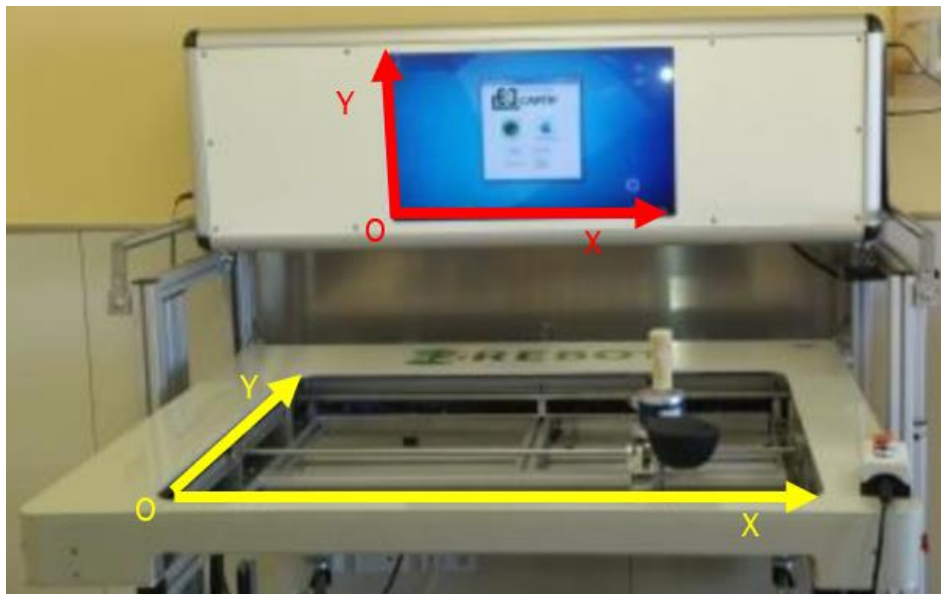


Figura 31: Referencias de PHYSIOBOT. Fuente: Elaboración propia.

La creación de estos ejes, que no coinciden con todos los ejes de los tipos de terapia, implica la modificación de los puntos deseados para que sigan la misma orientación y origen.

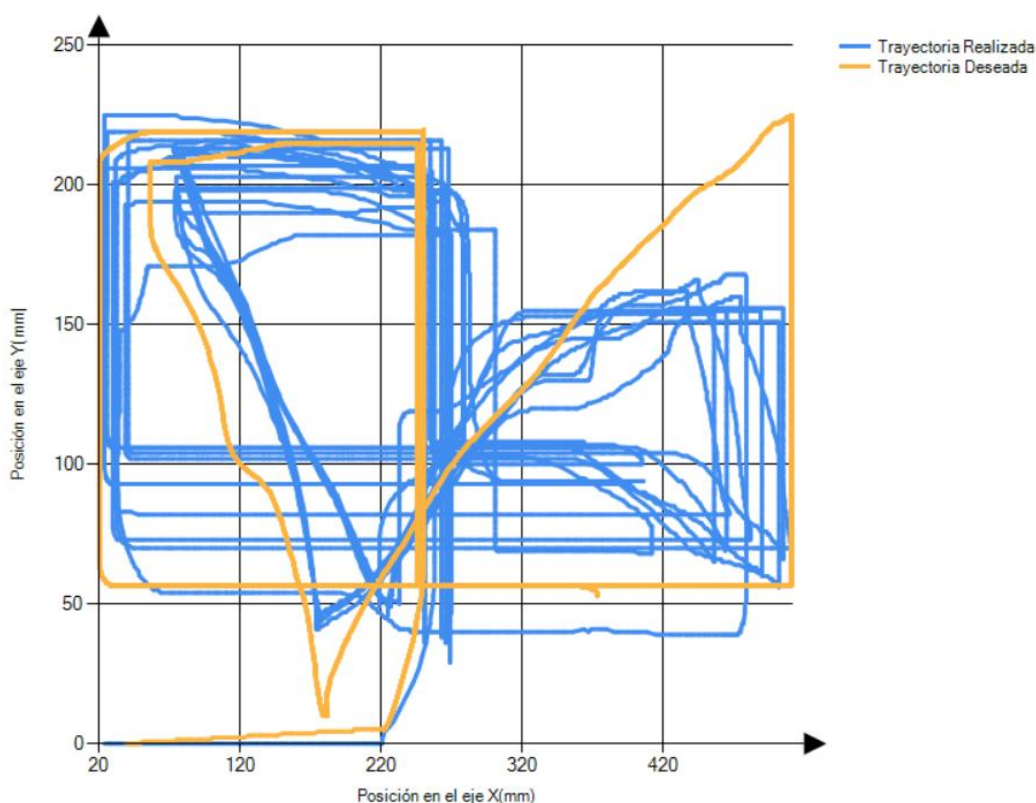
4.5 ANÁLISIS DE LOS DATOS ASOCIADOS A UNA TERAPIA

Es la aplicación del software que realiza los gráficos más característicos de una única terapia. Se ha elaborado un manual de usuario para aprender su funcionamiento (ANEXO I: Manual de usuario).

La aplicación consta de cinco métricas cuyo algoritmo se ha desarrollado y parametrizado en la clase Metodos de la solución TFG_Señales del código anejó. Estos gráficos son: Trayectoria, velocidad respecto a X, velocidad respecto a Y, Fuerza respecto a X y Fuerza respecto a Y.

4.5.1 Trayectoria

Esta métrica representa en una gráfica el itinerario realizado por el *joystick* (Trayectoria Realizada) y el recorrido que se visualiza en la pantalla (Trayectoria Deseada) (véase la gráfica 1).



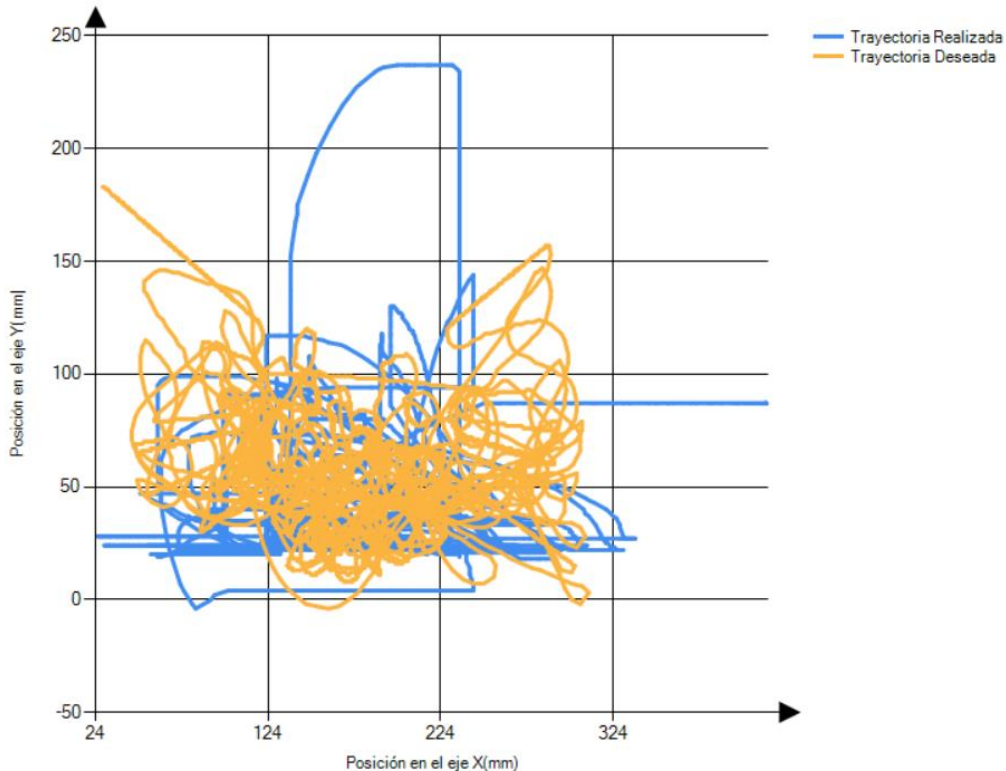
Gráfica 1: Métrica de Trayectoria. Fuente: Elaboración propia.

Para realizarla se hace una consulta de las posiciones realizadas y deseadas y se ordenan por el atributo *IdMuestra* (se ordenan cronológicamente). A continuación, se grafica punto a punto las posiciones que se irán uniendo con una línea recta entre ellas.



Esta métrica es de utilidad para las terapias de tipo 2 y 3, ya que se trata de terapias en las que se sigue un itinerario definido, por lo que a través de esta gráfica se puede observar cómo es capaz de seguir el paciente este recorrido.

En cambio, las terapias de tipo 1 no siguen un itinerario definido, sino que realizan una trayectoria aleatoria, llegando a ocupar casi por completo el plano XY sin poder sacar ninguna información clara (véase la gráfica 2).



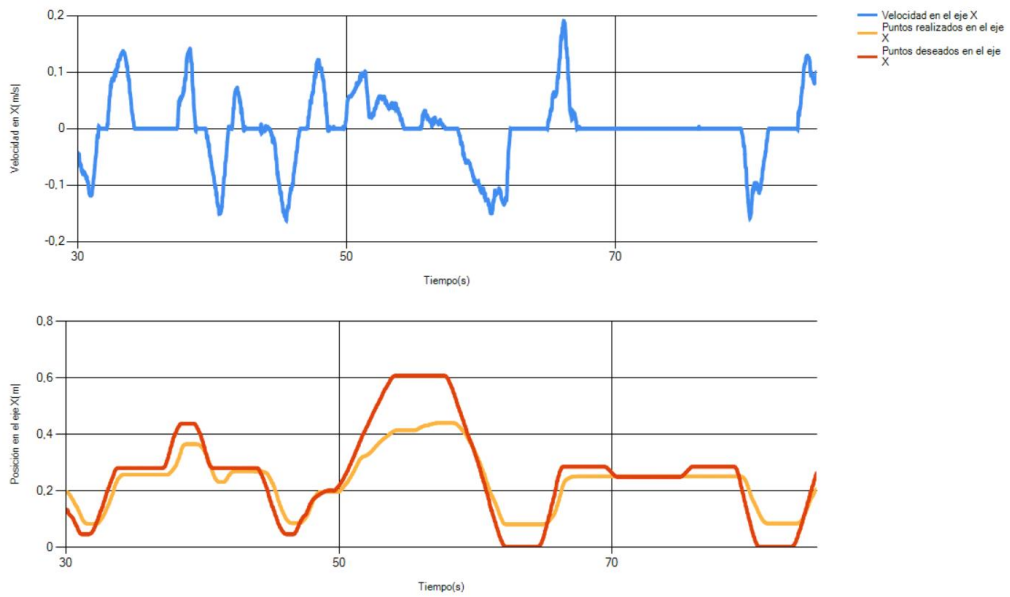
Gráfica 2: Métrica de trayectoria para una terapia de tipo 1. Fuente: Elaboración propia.

4.5.2 Velocidad respecto al eje X

Esta métrica muestra una gráfica de la velocidad del paciente respecto al eje X en un intervalo de tiempo. Además, junto a la gráfica de la velocidad, se añade otra gráfica de las posiciones realizadas y las posiciones deseadas respecto al eje X (véase la gráfica 3).

Para realizarla se hace una consulta a la BBDD del tiempo, las posiciones realizadas y deseadas y la velocidad, todo respecto x y ordenado por el atributo IdMuestra. A continuación, se representa la velocidad y posiciones frente al tiempo en gráficas distintas.

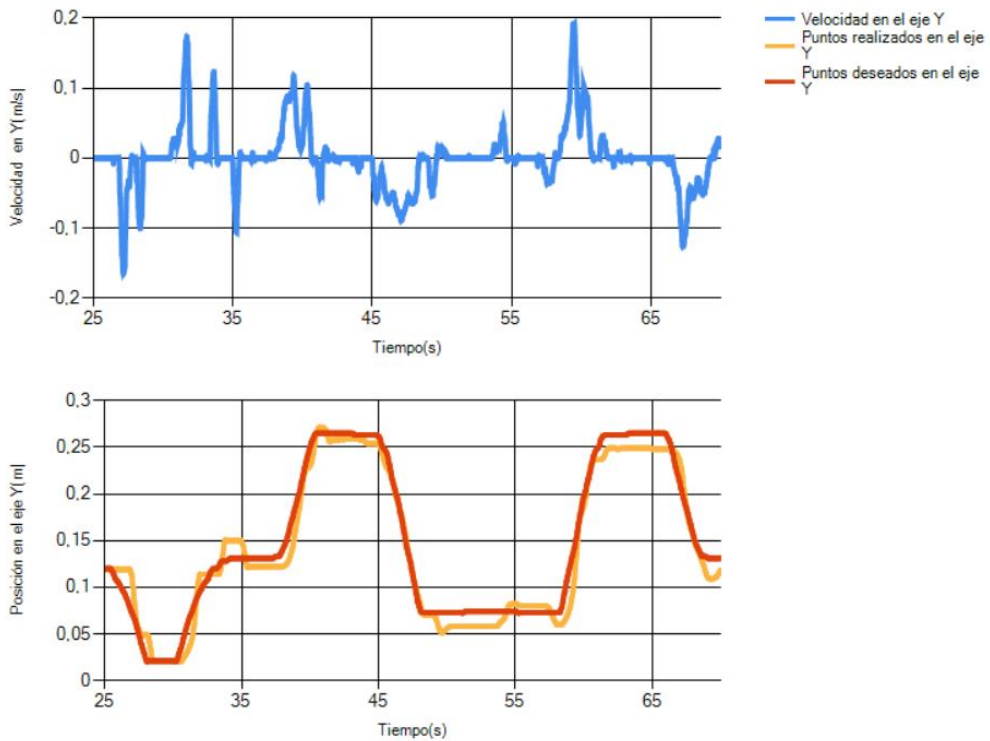
Hay que tener en cuenta que, si la velocidad es mayor que cero, el paciente se está desplazando a la derecha, y si es menor a la izquierda.



Gráfica 3: Métrica de la velocidad respecto al eje X. Fuente: Elaboración propia.

4.5.3 Velocidad respecto al eje Y

Esta métrica muestra dos gráficas: velocidad respecto al eje Y y posiciones realizadas y deseadas respecto al eje Y, ambas gráficas en un intervalo de tiempo que se especifica en la aplicación (véase la gráfica 4).

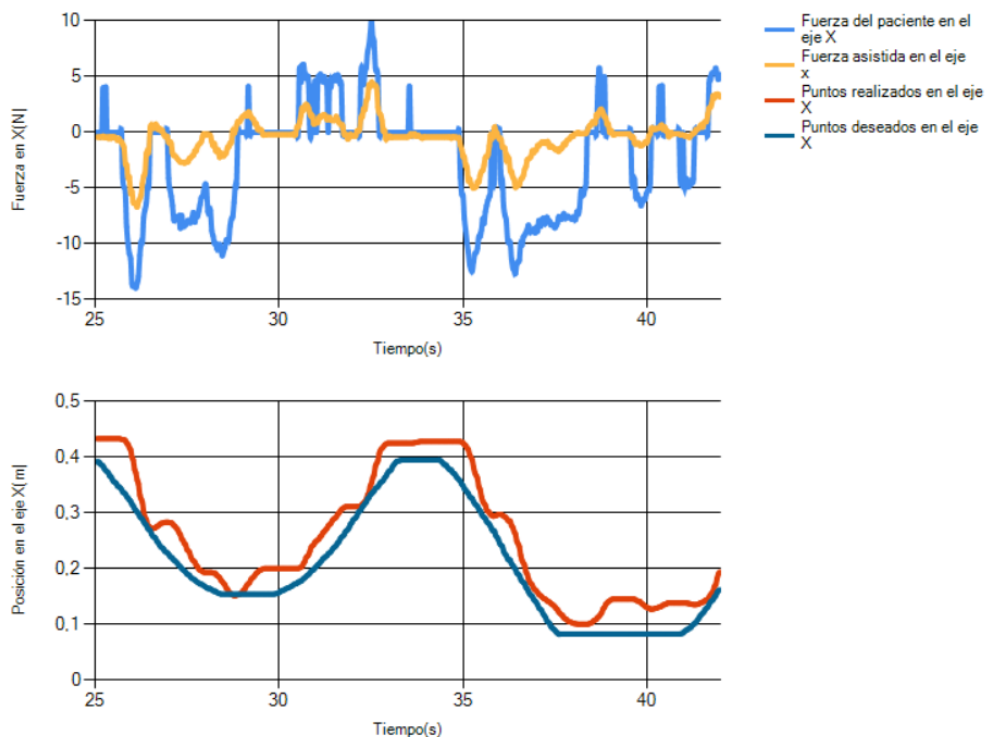


Gráfica 4: Métrica de la velocidad respecto al eje Y. Fuente: Elaboración propia.

La realización es idéntica a la del apartado anterior, pero referenciado al eje Y. Además, hay que tener en cuenta que velocidades mayores que cero implican desplazamiento hacia arriba, y velocidades negativas implican desplazamiento hacia abajo.

4.5.4 Fuerza respecto al eje X

Esta métrica realiza una gráfica que representa la fuerza que ha realizado el paciente y la fuerza con que le ha ayudado el robot (Fuerza asistida), ambas respecto al eje X en un intervalo de tiempo determinado. También se representan las posiciones respecto a X en otra gráfica (véase la gráfica 5).



Gráfica 5: Métrica de la fuerza respecto al eje X. Fuente: Elaboración propia.

Para graficar la fuerza del paciente solo es necesario consultar el atributo FuerzaX de la tabla Muestra y el tiempo. En cambio, la fuerza asistida es más compleja de graficar ya que no se obtiene directamente. Para calcularla hay que apoyarse en el diagrama de bloques de las fuerzas de PHYSIOBOT (véase la figura 32).

En él se observa varias variables cuyo significado, según Zatarain (2015), es:

- **Fuerza deseada:** es la fuerza asistiva que queremos calcular.
- **Masa:** es la masa del soporte (1kg).
- **Aceleración:** es la aceleración del joystick.
- **b:** constante de 18 kg/s.

- **Velocidad:** velocidad lineal del *joystick*.
- **K:** es el dato almacenado en la base de datos con el nombre de FuerzaAsistiva.
- **Pxf:** es el punto de la pantalla (punto deseado) respecto a un eje.
- **Px:** es la posición respecto a un eje del *joystick* (punto realizado).
- **Ks:** ganancia del sistema, adimensional y de valor 5.
- **Fuerza de consigna:** fuerza de consigna enviada a los motores.
- **Fuerza de compensación:** fuerza para compensar el error.
- **Error:** error entre la fuerza deseada y la medida.
- **Fuerza:** fuerza medida de los motores.

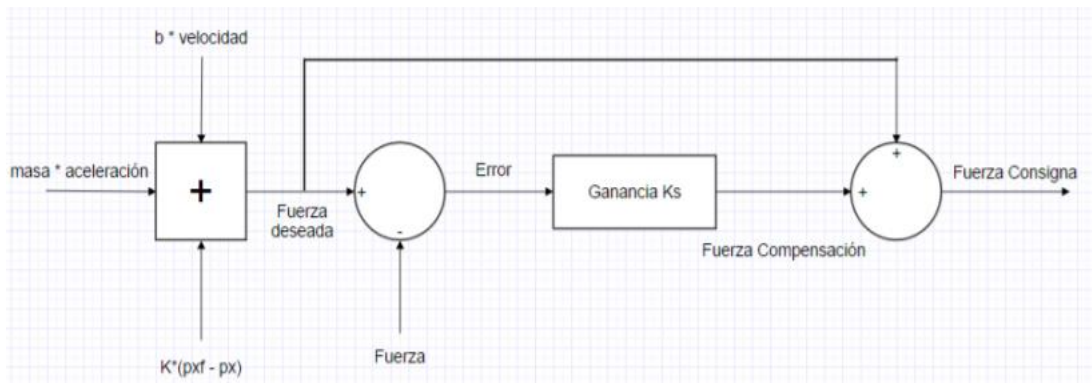


Figura 32: Diagrama de bloques de la fuerza del robot. Fuente: Zatarain (2015).

Del diagrama de bloques se quiere calcular la fuerza deseada. Para ello, obtenemos su ecuación del primer sumador (véase la ecuación 1).

$$Fuerza\ deseada = masa * aceleración + b * velocidad + k * (pxf - px)$$

Ecuación 1: Cálculo de la fuerza asistiva.

De esta ecuación, las constantes son conocidas y todas las variables se pueden obtener directamente con una consulta de la tabla IdMuestra, salvo la aceleración. Para el cálculo de la aceleración en un punto i , se emplea una nueva ecuación (véase la ecuación 2).

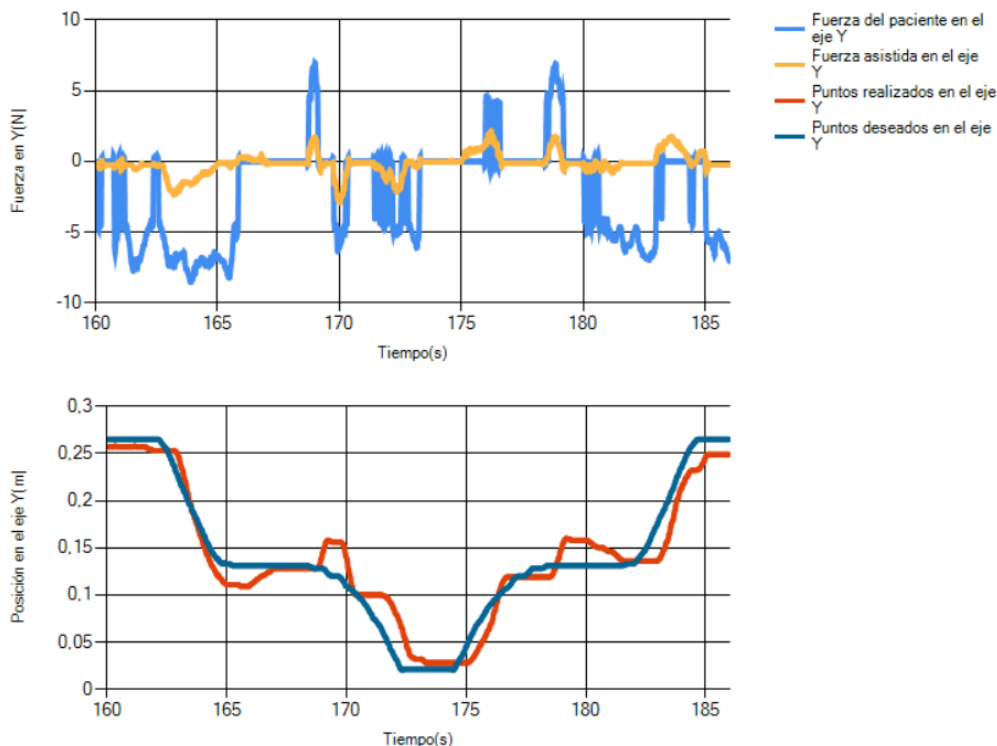
$$aceleración_i = \frac{velocidad_{i+1} - velocidad_i}{tiempo_{i+1} - tiempo_i}$$

Ecuación 2: Cálculo de la aceleración para un punto.

Teniendo todos los valores de la ecuación 1 se puede calcular y graficar la fuerza asistiva respecto al tiempo.

4.5.5 Fuerza respecto al eje Y

Esta métrica representa una gráfica de la fuerza asistida y la fuerza realizada por el paciente (ambas respecto al eje Y) en un intervalo de tiempo y otra gráfica con las posiciones, también respecto al eje Y, a lo largo del tiempo especificado.



Gráfica 6: Métrica de la fuerza respecto al eje Y. Fuente: Elaboración propia.

Para la realización de esta métrica se utiliza el mismo procedimiento que el utilizado en el apartado anterior, salvo que en vez de respecto al eje X, se utilizan los datos respecto al eje Y.

4.6 EVOLUCIÓN DE PACIENTES

Evolución de pacientes es el nombre de la segunda aplicación software. Esta realiza el tratamiento de los datos de varias terapias para tratar de determinar cómo evoluciona un paciente. Para aprender su funcionamiento se ha realizado una guía de su uso (ANEXO I: Manual de usuario).

Hay que tener en cuenta que las terapias siguen un orden secuencial independiente del paciente. Es decir, si se selecciona un paciente (por ejemplo, el 3) y se selecciona un rango de terapias (por ejemplo, de 1000 a 1020), no todas las terapias de ese rango están hechas por el mismo paciente y solo se tendrán en cuenta las realizadas por el paciente seleccionado (por ejemplo, la 1000, 1001, 1013, 1020).



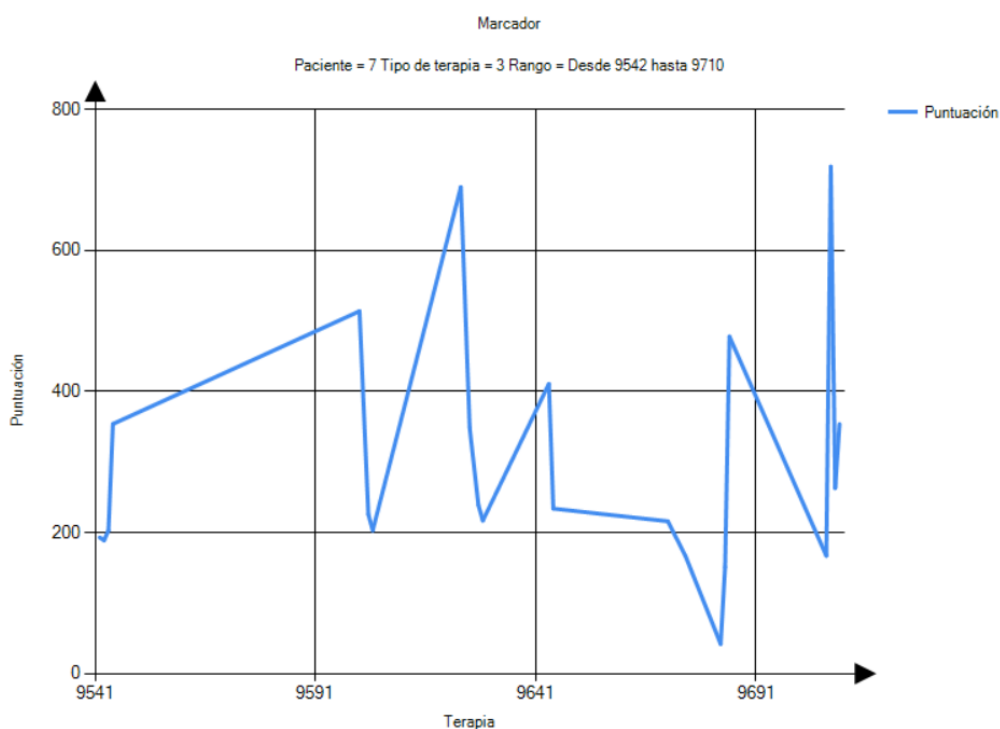
Además, cada tipo de terapia tiene que ser analizado por separado, ya que la puntuación o los puntos deseados no evolucionan igual para las terapias activo-asistida, pasivo-guiada y AVD.

Como última observación, no todas las métricas son recomendables para todos los tipos de terapia. Por ejemplo, las terapias de AVD suelen cambiar los puntos deseados de un extremo de la pantalla al otro extremo, sin “simular” un recorrido de un punto al otro, por lo que en la métrica de error cometido se obtendrán valores muy elevados.

En total hay desarrolladas en la aplicación 9 métricas: marcador, porcentaje de error, media y desviación, facilidad de movimiento, tiempo de reacción, número de temblores, simetría direccional en X y en Y, simetría de velocidad y ángulo y evolución de fuerza. Los algoritmos de estas métricas están desarrollados en la clase Metodos de la solución Evolucion del código anejo.

4.7.1 Marcador

Esta métrica representa en una gráfica la puntuación final del juego realizado en una terapia en el eje de ordenadas, mientras que en el de abscisas se muestra un rango de terapias especificado por el usuario (véase la gráfica 7).

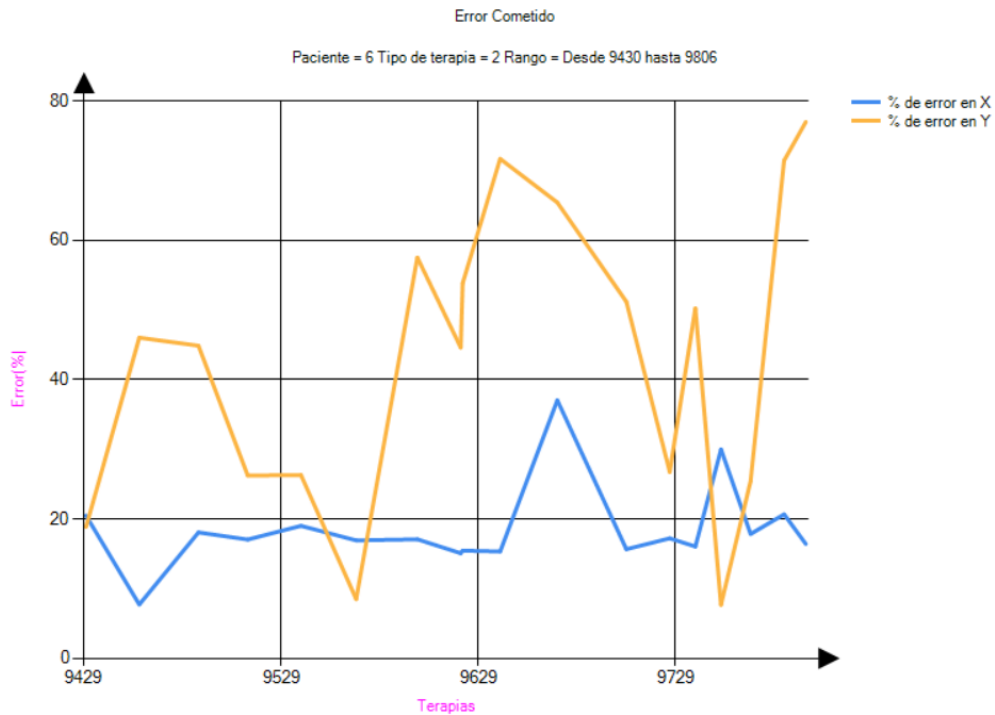


Gráfica 7: Métrica de Marcador. Fuente: Elaboración propia.

Para su realización se ejecuta una consulta del atributo MarcadorTotal de la tabla IdMuestra y del atributo IdTerapia, con la condición de que sean del mismo tipo de terapia (seleccionada por el usuario). A continuación, se grafican los datos obtenidos.

4.7.2 Porcentaje de error

En esta métrica se calcula el porcentaje de error cometido en las posiciones realizadas frente a las deseadas respecto a ambos ejes en el intervalo de terapias especificado por el usuario (véase la gráfica 8).



Gráfica 8: Métrica de Marcador. Fuente: Elaboración propia.

Para la realización de esta métrica se consultan las posiciones deseadas y realizadas del tipo de terapia y rango de terapias especificado por el usuario en la aplicación. A continuación, se calcula el error cometido en cada uno de los puntos (véase la evacuación 3).

$$Error_i = \frac{|pxf_i - puntoX|}{pxf_i + puntoX}$$

Ecuación 3: Porcentaje de error cometido.

Siendo:

- **i:** es el valor de IdMuestra del punto que vamos a calcular el error.
- **pxf:** es la posición deseada respecto a un eje.
- **puntoX:** punto realizado por el paciente con el robot respecto a un eje.
- **Error:** porcentaje de error cometido.

Cada vez que se calculan todos los porcentajes de errores de las posiciones muestreadas de una terapia, se realiza la media de todos ellos (véase la



ecuación 4), se almacena en la clase Series y se prosigue con la siguiente terapia. Cuando todos los valores han sido calculados y almacenados se grafican.

$$Media_k = \frac{1}{n} \sum_{\min(i \in k)}^{\max(i \in k)} \mu$$

Ecuación 4: Media de una variable.

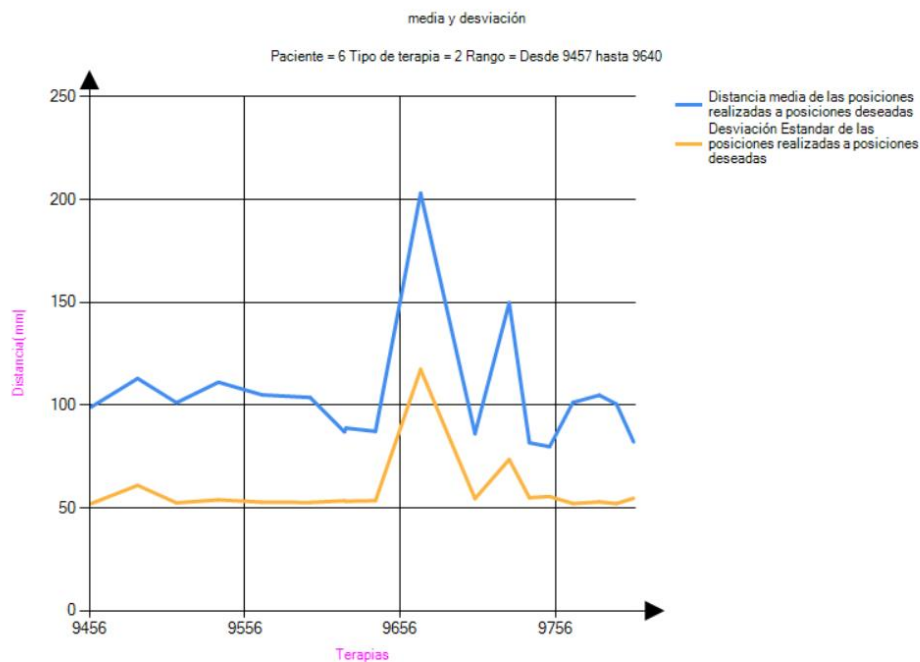
Siendo:

- **n**: es la diferencia entre el IdMuestra inicial y el IdMuestra final.
- **k**: es el valor de IdTerapia.
- **μ**: es la variable de la que se quiere hacer la media.

Esta terapia nos permite ver qué movimientos le cuesta más realizar al paciente, y es de utilidad tanto para las terapias de tipo 1 como las de tipo 2, pero no para las de tipo 3. Esto es porque las terapias de tipo 3 no siguen un itinerario, sino que cambian de posición de extremo a extremo de la pantalla dando lugar a errores muy elevados.

4.7.3 Media y desviación

En esta métrica se calcula y grafica la media de la distancia entre las posiciones realizadas y las deseadas y la desviación de estas en el intervalo de terapias especificado por el usuario (véase la gráfica 9).



Gráfica 9: Métrica de Media y Desviación. Fuente: Elaboración propia.



Para el cálculo de la media de un punto se realiza la suma cuadrática de la diferencia entre la posición realizada y la posición deseada en cada uno de los ejes (véase la ecuación 5).

$$M_i = \sqrt{(\text{punto}X_i - pxf_i)^2 + (\text{punto}Y_i - pyf_i)^2}$$

Ecuación 5: Media de los puntos realizados frente a los deseados.

Para el cálculo de la desviación se calcula la media de las distancias medias (M_k) (véase la ecuación 4) y a continuación se calcula la desviación típica (véase la ecuación 6)

$$\sigma_i = (M_i - M_k)^2$$

Ecuación 6: Desviación típica.

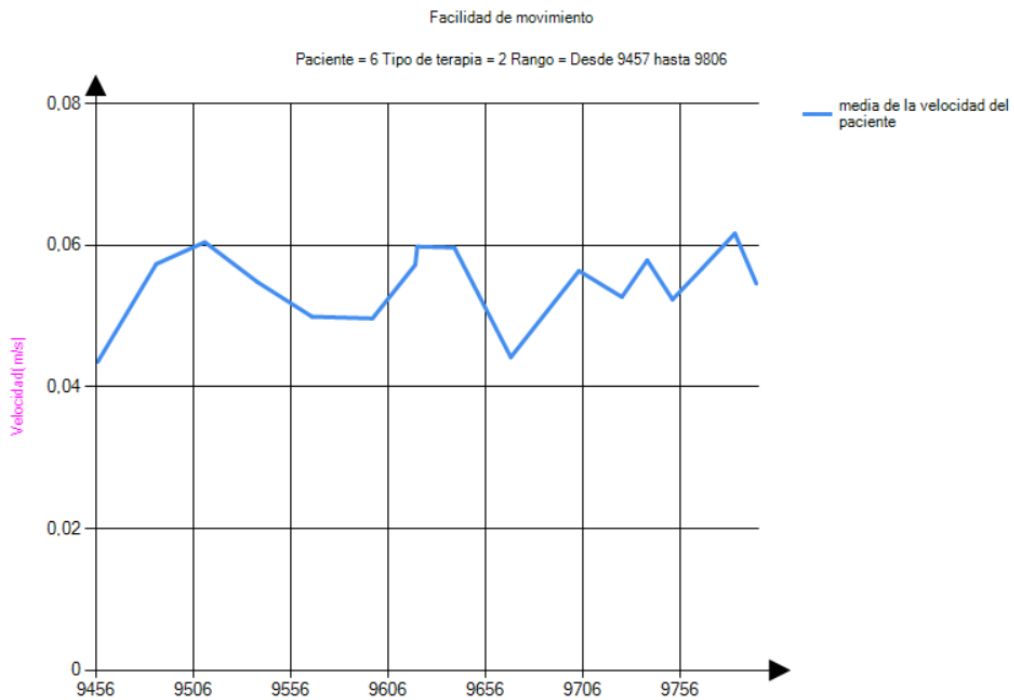
Finalmente, se realiza la media de las desviaciones típicas de todos los puntos de una terapia (ecuación 4), se obtiene σ_k y se almacenan los datos en la clase Series. Cuando se tienen los valores de todas las terapias se grafica.

Esta métrica es válida para todos los tipos de terapia, siempre y cuando se realicen independientemente, ya que en las terapias de tipo AVD tendremos mayores valores por las grandes distancias entre puntos deseados y realizados. Tampoco podrían analizarse conjuntamente las terapias de tipo 1 y tipo 2, ya que en las terapias de tipo 1 no siguen un recorrido predefinido, lo que hace necesario un tiempo de reacción que aumenta la distancia media.

4.7.4 Facilidad de movimiento

Esta métrica calcula y grafica la media de las velocidades obtenidas en ambos ejes de las terapias escogidas por el usuario (véase la gráfica 10), con la finalidad de determinar con qué facilidad se realizan los movimientos.

Para su realización se hace la suma cuadrática de las velocidades en los ejes (véase la ecuación 7) y a continuación se realiza la media de todas las velocidades de una terapia (véase la ecuación 4). Finalmente se grafican los datos obtenidos.



Gráfica 10: Facilidad de movimiento. Fuente: Elaboración propia.

$$V_i = \sqrt{V_y^2 + V_x^2}$$

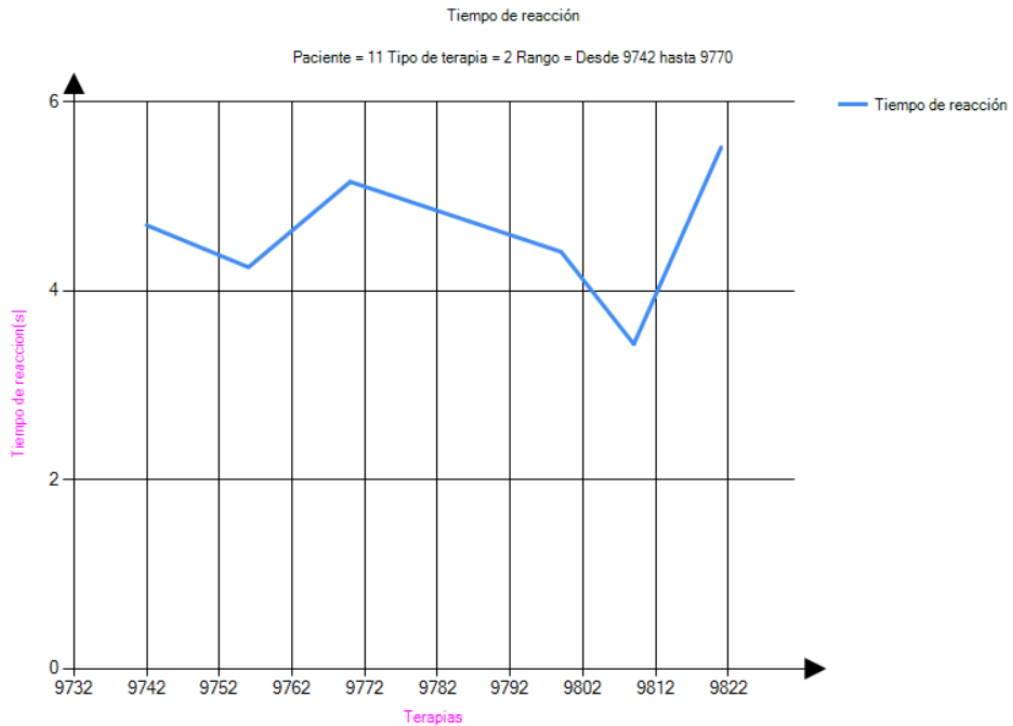
Ecuación 7: Velocidad total en un punto.

Esta métrica se puede utilizar para cualquier tipo de terapia, teniendo en cuenta que las terapias de tipo 3 tienen mayores velocidades al ser recorridos más largos y las de tipo 1 son recorridos cortos y aleatorios y por consiguiente sus velocidades son menores.

4.7.5 Tiempo de reacción

En esta métrica se calcula el tiempo que tarda una persona en reorientarse ante un cambio de sentido. Es decir, se mide el tiempo desde que se produce un cambio de dirección en la pantalla hasta que el paciente realiza este cambio de dirección con el robot (véase la gráfica 11)

Para la elaboración de esta métrica se calcula la dirección de las posiciones realizadas y deseadas respecto a ambos ejes (véanse las ecuaciones 8 y 9). A continuación, se multiplica la dirección de las posiciones realizadas y la dirección de las posiciones deseadas (véase la ecuación 10) de forma que se obtienen PDx y PDy.



Gráfica 11: Métrica de tiempo de reacción. Fuente: Elaboración propia.

$$Dpx_i = puntoX_{i+1} - puntoX_i$$

Ecuación 8: Cálculo de la dirección de las posiciones realizadas respecto a un eje.

$$Dpxf_i = pxf_{(i+1)} - pxf_i$$

Ecuación 9: Cálculo de la dirección de las posiciones deseadas respecto a un eje.

$$PDx_i = Dpx_i * Dpxf_i$$

Ecuación 10: Producto de las direcciones respecto a un eje.

Con PDx y PDy se puede saber cuánto tarda en reaccionar el paciente, ya que si uno de los dos es negativo significa que las posiciones realizadas y deseadas tienen una dirección distinta. De esta forma, cuando se detecta un cambio de positivo a negativo en el producto de las direcciones significa que ha habido un cambio de dirección. Si se alcanza el tiempo en el que se pasa de positivo a negativo y se le resta al tiempo que se consigue cuando pasa de negativo a positivo (momento en el que las direcciones se igualan), se obtiene el tiempo que ha tardado en reorientarse.

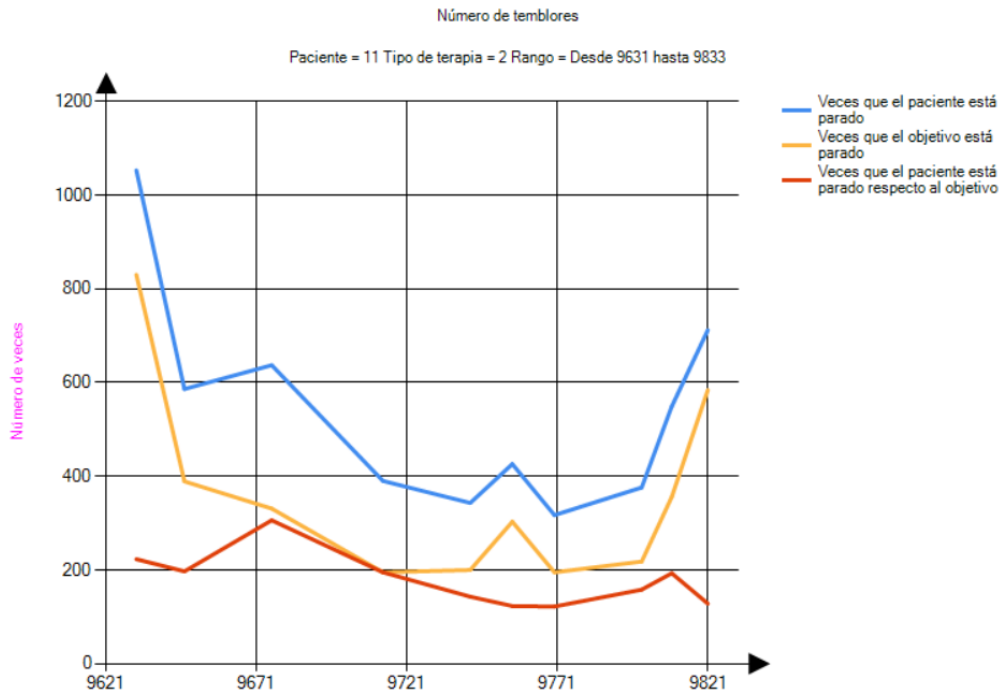
Esta métrica es válida para todas las terapias, teniendo en cuenta que las terapias de tipo 1 son cambios de sentido más “suaves” que las terapias de



tipo 3, y que las terapias de tipo 2 son recorridos definidos, por lo que se tarda mucho menos tiempo en reaccionar.

4.7.6 Número de temblores

En esta métrica se realiza una gráfica que muestra el número de veces que ha estado parado el objetivo de la pantalla y el robot. Además, se grafica la comparación entre estas dos (véase la gráfica 12)



Gráfica 12: Métrica de número de temblores. Fuente: Elaboración propia.

Para la elaboración de esta métrica se calcula la dirección de las posiciones realizadas y deseadas respecto a los ejes (véase las ecuaciones 8 y 9). A continuación, se realiza la suma cuadrática de las direcciones en los ejes (véase la ecuación 11).

$$DP_i = \sqrt{Dpx_i^2 + Dpy_i^2}$$

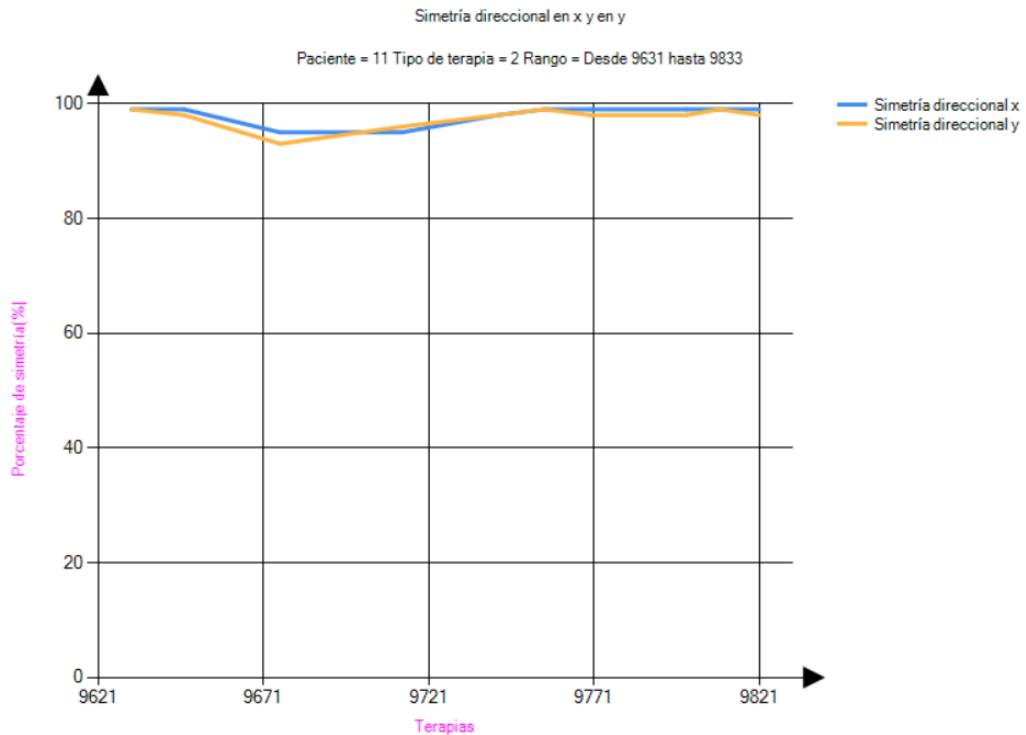
Ecuación 11: Suma cuadrática de las direcciones en los ejes.

Si en algún punto DP vale 0, significa que el robot (para el caso de las posiciones realizadas) o el objetivo (en el caso de las posiciones deseadas) está parado. Si en la posición anterior DP vale también 0, es que el robot o el objetivo siguen parados, por lo que en este caso no se cuenta nuevamente como parado. Finalmente, se realiza la comparación entre cada una de las terapias especificadas y se grafica.



4.7.7 Simetría direccional respecto a X e Y

En esta métrica se calcula y gráfica el rendimiento de predicción de cambio de dirección en una serie temporal con respecto a cada uno de los ejes (véase la gráfica 13). Es decir, esta métrica muestra la evolución de la capacidad del paciente de seguir los puntos previstos.



Gráfica 13: Métrica de simetría direccional. Fuente: Elaboración propia.

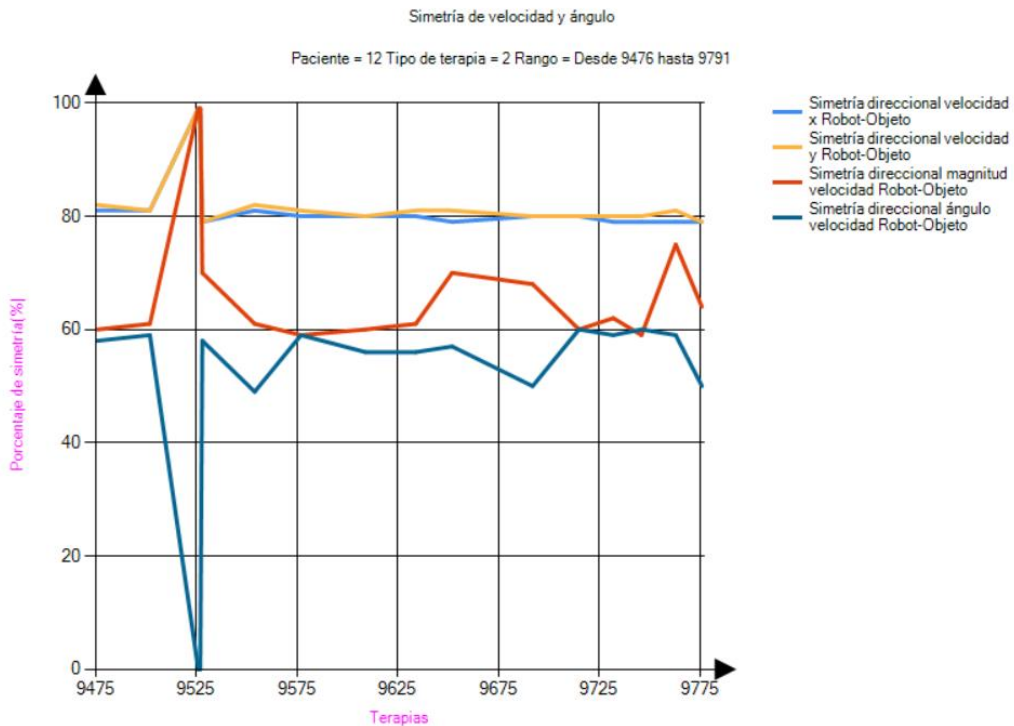
Para su cálculo se sigue el mismo procedimiento que la métrica de tiempo de reacción: se calcula la dirección de las posiciones realizadas (véase la ecuación 8) y las deseadas (véase la ecuación 9) con respecto a cada uno de los ejes, y a continuación se multiplican (véase la ecuación 10) obteniéndose PDx y PDy.

Se cuenta el número de veces que PDx y PDy es mayor que 0 (misma dirección entre posiciones realizadas y deseadas) y se obtiene el rendimiento direccional en X y en Y.

Esta métrica se puede utilizar en todos los tipos de terapia, aunque hay que tener en cuenta las consideraciones mencionadas en la métrica de tiempo de reacción (autoridad de la terapia tipo 1, recorrido definido de la terapia tipo 2, etc.).

4.7.8 Simetría respecto a velocidad y ángulo

En esta métrica se calcula el rendimiento de predicción de cambio de la velocidad respecto a cada uno de los ejes y de la magnitud, y el ángulo (véase la gráfica 14).



Gráfica 14: Métrica de simetría de velocidad y ángulo. Fuente: Elaboración propia.

Para la realización de esta métrica es necesario calcular la velocidad del objetivo de la pantalla con respecto a cada uno de los ejes (véase la ecuación 12), la magnitud de las velocidades (suma cuadrática de las velocidades con respecto a los ejes) y el ángulo de la trayectoria deseada y realizada (véase la ecuación 13).

$$V_{obj_i} = \frac{\text{punto}Xf_{i+1} - \text{punto}Xf_i}{\text{tiempo}_{i+1} - \text{tiempo}_i}$$

Ecuación 12: cálculo de la velocidad del objeto de la pantalla.

$$\text{Angulo} = \text{arcoseno} \left(\frac{Vy}{|V|} \right)$$

Ecuación 13: Cálculo del ángulo.

A continuación, se sigue el mismo procedimiento que la métrica simetría direccional (véase las ecuaciones 8,9 y 10), salvo que en vez de utilizar las posiciones realizadas y deseadas se emplean las velocidades del robot y del objetivo, y de la misma manera el ángulo.

Esta métrica se puede utilizar para todos los tipos de terapia, considerando que las velocidades en las terapias de tipo 3 son mayores y que los ángulos de la terapia de tipo 1 suelen ser muy pequeños, por lo que la variación también lo es.

4.7.9 Evolución de fuerza

En esta métrica se calcula la evolución de la fuerza que ejerce el paciente con respecto a cada uno de los ejes, teniendo en cuenta la fuerza realizada por el robot (véase la gráfica 15).



Gráfica 15: Métrica de evolución de fuerza. Fuente: Elaboración propia.

Para su realización se calcula la fuerza con respecto a cada eje en cada punto como se indica en el apartado 4.5.4 y 4.5.5 (véase las ecuaciones 1 y 2). A continuación, se calcula el porcentaje de fuerza realizada por el paciente en cada eje (véase la ecuación 14).

$$EvolucionF_k = \frac{FuerzaX_k}{FuerzaX_k + FuerzaAsistiva_k}$$

Ecuación 14: Porcentaje de fuerza realizado respecto a un eje.

Esta métrica se puede utilizar para todos los tipos de terapia, teniendo en cuenta que en las terapias de tipo 3 y 2 se trabaja con grandes recorridos, lo que da lugar a mayores aceleraciones, y por consiguiente mayores fuerzas.

4.7.10 Generación de informe

El programa desarrollado inicialmente realizaba un informe con R (Zatarain, 2015). En la aplicación desarrollada en el presente TFG se realiza con C#, modificando ligeramente el informe generado por la antigua aplicación (actualización de logotipos, mejora de la visualización de los datos relevantes,



Universidad de Valladolid

diferente estilo de gráficas, etc.) pero manteniendo tanto la estructura de presentación como las definiciones de las métricas (véase el ANEXO II).

Para la realización del informe se ha utilizado el NuGet iTextSharp, el cual permite crear y modificar un pdf con simples comandos en C#. Un NuGet es un paquete creado por un desarrollador que contiene código compilado y que se puede añadir con facilidad a los proyectos de .NET para incorporar nuevas herramientas.

Lo primero en crear en el informe es la portada, en la cual se muestran siempre el identificador del paciente seguido del logo de CARTIF y PHYSIOBOT, y en la parte inferior la fecha de creación del informe y el tipo de terapia escogido (véase la figura 33).

EVOLUCIÓN DEL PACIENTE 6



Fecha del Informe: 29/05/2020 13:10:21

El tipo de terapia es: 2

Figura 33: Portada del informe. Fuente: Elaboración propia.

El siguiente paso es la creación del cabecero de las métricas, para el cual se ha creado un método ya que siempre sigue la misma estructura: primero el logotipo de PHYSIOBOT y a continuación el título formado por el identificador del paciente, la métrica que se muestra y el tipo de terapia (véase la figura 34).

Para finalizar, se llaman a los métodos de las métricas seleccionadas por el usuario para incluir en el informe. Estos métodos tienen un parámetro booleano (`generarInforme`) que, al ser su valor verdadero, introduce la definición de la métrica, crea una tabla con los datos más relevantes de esta e incluye la gráfica realizada (véase la figura 34).



Paciente 4/ Marcador / Terapia tipo 2

En esta métrica vamos a ver la cantidad de puntos que consigue el paciente en el juego del ratón, cada vez que el paciente atrapa al ratón recibe puntos de forma que cuantas más veces haya atrapado al ratón, mayor puntuación tendrá, la puntuación puede variar bastante dependiendo de la dificultad.

El índice de esta métrica son los puntos del juego.

Valor	Puntuación	Número de terapia	Fecha
Primero	28	9384	25/05/2017 10:39:40
Último	54	9749	30/06/2017 9:53:56
Máximo	259	9659	21/06/2017 9:55:36
Mínimo	7	9657	21/06/2017 9:54:53

[media puntuación: 75,92308](#)

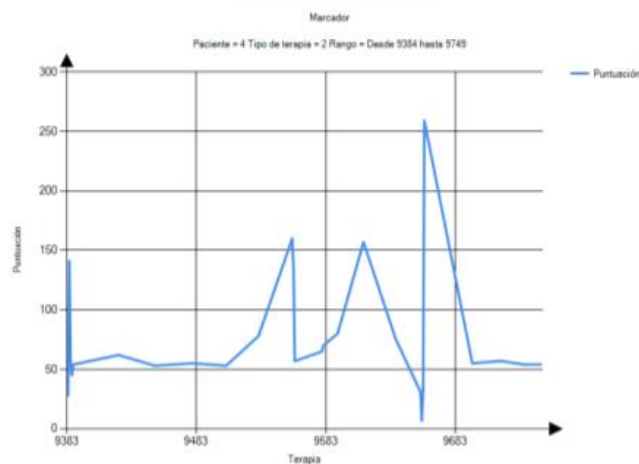


Figura 34: Página estándar del informe. Fuente: Elaboración propia.

En la antigua aplicación los datos relevantes se expresaban en forma de redacción. En la nueva se ha optado por implantar tablas, que dan mayor facilidad de lectura de datos y una mejor estética.

Un inconveniente de no utilizar R es que la calidad de las gráficas que se incluyen en el informe es menor, diferencia que se nota sobre todo cuando se quieren obtener datos concretos de la gráfica, además de tener una peor estética.

4.8 TRATAMIENTO DE ERRORES

Durante el uso del programa pueden surgir diferentes errores, algunos causados por el usuario, otros de tipo matemático, o simplemente que no hay conexión con la base de datos. Se ha desarrollado un protocolo para solucionar o informar al usuario de cada tipo de error.



4.8.1 Errores de selección de parámetros

Son errores que se producen cuando falta algunos de los campos por completar o que se han completado de forma errónea, como por ejemplo establecer una terapia de inicio mayor que la terapia de fin.

Cuando se producen este tipo de errores se avisa al usuario a través de un mensaje por pantalla indicándole qué parámetros son incorrectos o están incompletos (véase la figura 35).

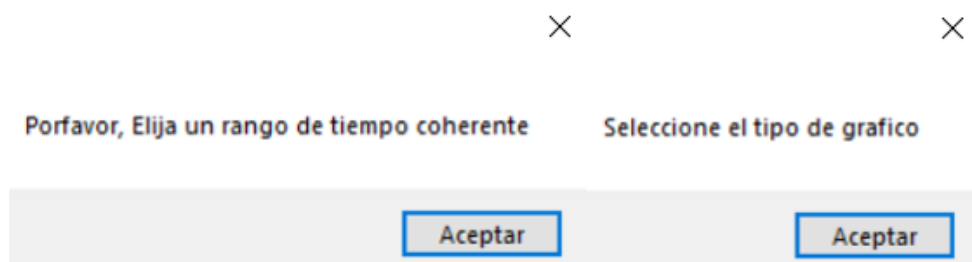


Figura 35: Errores de selección de parámetros. Fuente: Elaboración propia.

4.8.2 Errores de tratamiento de datos

Son errores que se producen en el cálculo de las métricas, como por ejemplo obtener como resultado de una división infinito. Suelen aparecer sobre todo en divisiones, por ello, una forma de prevenir este error es no tratar los datos que tienen como denominador 0 (debido a la gran cantidad de datos con la que se trabaja, descartar algunos datos no es relevante y su influencia en el resultado final es mínima).

Además, en las operaciones con fracciones se ha implantado un filtro para que las variables con valor infinito o número no definido tampoco se tengan en cuenta en el tratamiento de datos.

4.8.3 Errores de gestión de la BBDD

Este tipo de errores se producen cuando ha habido un error con el gestor de la base de datos, ya sea porque no se ha podido establecer conexión con la BBDD, porque no hay datos en ella o porque no se ha podido completar correctamente la consulta.

Cuando se producen este tipo de errores, se muestra al usuario un mensaje por pantalla mostrando el reporte del error, el cual es generado automáticamente por la app (véase la figura 36).



```
Error debido a: Se agotó el tiempo de espera de ejecución. El periodo de tiempo de espera transcurrió antes de la finalización de la operación o el servidor no responde.
Error debido a: .Net SqlClient Data Provider
Error debido a: en System.Data.SqlClient.SqlConnection.OnError(SqlException exception, Boolean breakConnection, Action`1 wrapCloseInAction)
en System.Data.SqlClient.SqlInternalConnection.OnError(SqlException exception, Boolean breakConnection, Action`1 wrapCloseInAction)
en System.Data.SqlClient.TdsParser.ThrowExceptionAndWarning(TdsParserStateObject stateObj, Boolean callerHasConnectionLock, Boolean asyncClose)
en System.Data.SqlClient.TdsParser.TryRun(RunBehavior runBehavior, SqlCommand cmdHandler, SqlDataReader dataStream, BulkCopySimpleResultSet bulkCopyHandler,
TdsParserStateObject stateObj, Boolean& dataReady)
en System.Data.SqlClient.SqlDataReader.TryConsumeMetaData()
en System.Data.SqlClient.SqlDataReader.get_MetaData()
en System.Data.SqlClient.SqlCommand.FinishExecuteReader(SqlDataReader ds, RunBehavior runBehavior, String resetOptionsString, Boolean isInternal, Boolean
forDescribeParameterEncryption, Boolean shouldCacheForAlwaysEncrypted)
en System.Data.SqlClient.SqlCommand.RunExecuteReaderTds(CommandBehavior cmdBehavior, RunBehavior runBehavior, Boolean returnStream, Boolean async, Int32 timeout,
Task& task, Boolean asyncWrite, Boolean inRetry, SqlDataReader ds, Boolean describeParameterEncryptionRequest)
en System.Data.SqlClient.SqlCommand.RunExecuteReader(CommandBehavior cmdBehavior, RunBehavior runBehavior, Boolean returnStream, String method,
TaskCompletionSource`1 completion, Int32 timeout, Task& task, Boolean& usedCache, Boolean asyncWrite, Boolean inRetry)
en System.Data.SqlClient.SqlCommand.RunExecuteReader(CommandBehavior cmdBehavior, RunBehavior runBehavior, Boolean returnStream, String method)
en System.Data.SqlClient.SqlCommand.ExecuteReader(CommandBehavior behavior, String method)
en System.Data.SqlClient.SqlCommand.ExecuteReader(CommandBehavior behavior)
en System.Data.Common.DbCommand.ExecuteReader()
en System.Data.Linq.SqlClient.SqlProvider.Execute(Expression query, QueryInfo queryInfo, IObjectReaderFactory factory, Object[] parentArgs, Object[] userArgs,
ICompiledSubQuery[] subQueries, Object lastResult)
en System.Data.Linq.SqlClient.SqlProvider.ExecuteAll(Expression query, QueryInfo[] queryInfos, IObjectReaderFactory factory, Object[] userArguments, ICompiledSubQuery[]
subQueries)
en System.Data.Linq.SqlClient.SqlProvider.System.Data.Linq.Provider.IProvider.Execute(Expression query)
en System.Data.Linq.DataQuery`1.System.Collections.Generic.IEnumerable`T>.GetEnumerator()
en System.Collections.Generic.List`1.ctor(IEnumerable`1 collection)
en System.Linq.Enumerable.ToList(TSource[] IEnumerable`1 source)
en Evolucion.Metodos.graficaError(Boolean GenerarGrafico, Document Informe, Int32 ValorTipoTerapia, ComboBox cbPaciente, ComboBox CbRangoTerapiaSuperior, ComboBox
CbRangoTerapiaInferior, Chart GrafTraectoria, bddDataContext bd, String Ruta) en D:\Escritorio\documentación
TFG\Programas\TFG_señales.v1\EvolucionPacientes\Evolucion\Evolucion\Metodos.cs:línea 193
Error debido a: Void OnError(System.Data.SqlClient.SqlException, Boolean, System.Action`1[System.Action])
```

Aceptar

Figura 36: Error por agotar el tiempo de espera con el servidor. Fuente: Elaboración propia.

4.9 DOCUMENTACIÓN DEL CÓDIGO

Se ha contemplado que las aplicaciones desarrolladas pueden sufrir modificaciones en un futuro, ya sea por modificaciones del robot, de la BBDD o de la misma aplicación. Además, también se ha tenido en cuenta que las métricas desarrolladas en C# pueden ser de interés para otros sistemas de rehabilitación.

Por ello, con la finalidad de facilitar la comprensión del código, se ha realizado una documentación detallada del algoritmo desarrollado. En concreto, se han realizado dos documentaciones. En primer lugar, se ha comentado en el mismo código tanto las funciones de las variables como el procesado de la información. En segundo lugar, se han elaborado dos sitios web describiendo la estructura, clases, métodos y variables de las aplicaciones.

Se han adjuntado los sitios web de TFG_Señales y Evolucion en las carpetas anejas de Help_TFG_Señales y Help_Evolucion respectivamente. Para abrir la ayuda web hay que acceder a la carpeta de la cual queramos obtener la documentación y ejecutar index.html.

Cuando se abre la documentación aparecen las principales clases, de las cuales algunas han sido creadas por el desarrollador, otras se generan automáticamente por Visual Studio y el resto son las procedentes de la BBDD de SQL Server (véase la figura 37).



A Sandcastle Documented Class Library

Evolucion

- bbddDataContext Class
- CargaDatos Class
- Experimento Class
- Formulario Class
- HR Class
- Metodos Class
- Muestra Class
- Nom_Terapias Class
- PacientesBM Class
- Pleth Class
- SAM Class
- Series Class
- Signals Class
- SpO2 Class
- TerapiaBM Class
- TPrueba Class

Evolucion Namespace

[Missing <summary> documentation for "N:Evolucion"]

Classes

Class	Description
bbddDataContext	
CargaDatos	clase que tiene los procesos para cargar los datos de los ComboBox de windows Forms, además de la lista de checkedListBox.
Experimento	
Formulario	Formulario principal que registra los eventos y escoge los metodos a ejecutar.
HR	
Metodos	La clase Metodos contiene los metedos encargados de tratar los datos para generar las distintas métricas, En estos metodos también se puede implementar las métricas desarrolladas en un informe, se genera también el cabecero de las métricas y se realiza la comprobación de la correcta selección de los ComboBox

Figura 37: Documentación web. Fuente: Elaboración propia.

Como se observa en la figura 37 no todas las clases tienen definición, ya que las pertenecientes a la BBDD carecen de ella. Por otro lado, las clases, métodos, objetos, etc. propios de Visual Studio tienen las definiciones oficiales de Microsoft. Finalmente, las clases realizadas por el desarrollador tienen las definiciones que le ha otorgado este.

A la izquierda de la ayuda tenemos todas las clases principales, y clicando en ellas podemos acceder a los métodos, funciones, etc. que tienen cada una de ellas (véase la figura 38).

Syntax

```
C# VB C++ F#
public static class Metodos
```

The Metodos type exposes the following members.

Methods

Name	Description
comprobarSeleccion	Función que comprueba que que los ComboBox de Paciente, Terapia y Tipo Terapia han sido correctamente seleccionados
generaCabecero	Función que se encarga de realizar el cabecero del pdf de cada una de las métricas representadas.
graficaDireccional	Es la métrica de Simetría direccional x e y, gráfica tanto en la interfaz como en el PDF, el valor es la simetría direccional
graficaError	Es la métrica del % de error cometido en los ejes,gráfica tanto en la interfaz como en el PDF, el valor es el error cometido
graficaEvolucionFuerza	Es la métrica de la evolución de la fuerza, gráfica tanto en la interfaz como en el PDF, el valor es el porcentaje de la evolución de la fuerza teniendo en cuenta la que realiza el paciente y la que realiza el robot
graficaMarcador	Es la métrica de Marcador, gráfica tanto en la interfaz como en el PDF, el valor es la puntuación
graficaMedia	Es la métrica de media y desviación, gráfica tanto en la interfaz como en el PDF, Los valores son la media y la desviación de los valores recorridos menos los deseados

Figura 38: Documentación de métodos. Fuente: Elaboración propia.



Como se ve en la figura 38, cada método creado por el desarrollador tiene su propia definición. Además, se muestra su declaración en la aplicación, la cual puede ser modificada a otros lenguajes de programación. Clicando en un método accederemos a las variables de entrada de cada método (véase la figura 39).

▲ Syntax

```
C# VB C++ F#  
  
public static void graficaEvolucionFuerza(  
    bool GenerarGrafico,  
    Document Informe,  
    int ValorTipoTerapia,  
    ComboBox cbPaciente,  
    ComboBox CbRangoTerapiaSuperior,  
    ComboBox cbRangoTerapiaInferior,  
    Chart GrafTrayectoria,  
    bdddDataContext bd,  
    string Ruta  
)
```

Parameters

- GenerarGrafico*
Type: [System.Boolean](#)
Determina si esta métrica se incluye en el PDF
- Informe*
Type: **Document**
Es el PDF que generamos con las métricas
- ValorTipoTerapia*
Type: [System.Int32](#)
Indica que tipo de terapia es la seleccionada
- cbPaciente*
Type: [System.Windows.Forms.ComboBox](#)
ComboBox en el que seleccionamos el id del paciente
- CbRangoTerapiaSuperior*
Type: [System.Windows.Forms.ComboBox](#)
ComboBox en el que seleccionamos la ultima terapia a representar
- cbRangoTerapiaInferior*
Type: [System.Windows.Forms.ComboBox](#)
ComboBox en el que seleccionamos la primera terapia a representar

Figura 39: Documentación de variables de entrada. Fuente: Elaboración propia.

Como se ve en la figura 39, cada variable de entrada tiene su definición o significado, su declaración en el método y el tipo de variable de que se trata.

Esta documentación permite extraer los métodos de las métricas para poder implantarlos en otros proyectos, guiándose por las variables de entrada de cada método.

5. IMPLANTACIÓN

5.1 PRIMEROS RESULTADOS

Una vez completada la aplicación software y comprobado su correcto funcionamiento, se eliminó la BBDD de prueba con la que se había trabajado y se implantó la BBDD de terapias de rehabilitación reales.

Como resultado de graficar la trayectoria de varias terapias, se observó que la posición deseada estaba desplazada en algunas de ellas. Más concretamente, dependiendo del tipo de terapia los datos de la posición deseada tenían una referencia de ejes y un escalado distinto.

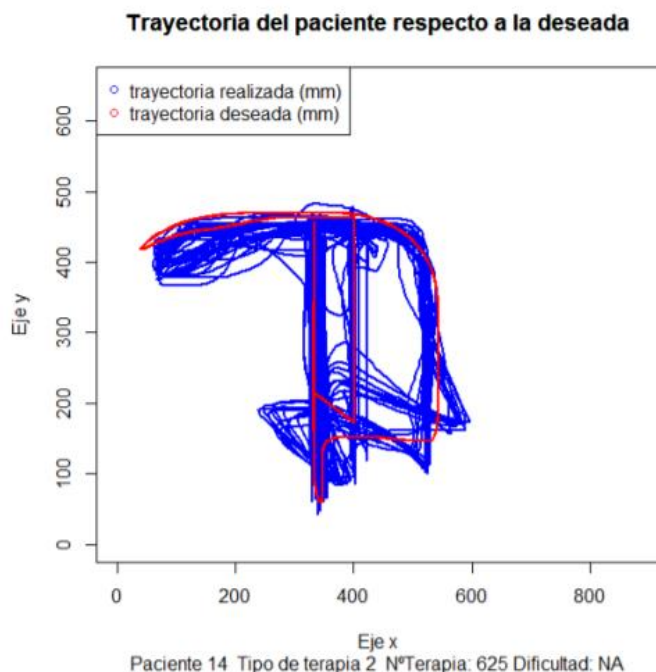
En el software inicial de CARTIF había un calibrado para las posiciones deseadas que se ajustó a la BBDD de prueba, para la cual sí funcionaba (véase la gráfica 16). Este calibrado se realizaba siguiendo las ecuaciones 15 y 16:

$$\text{PosiciónCalibradaY} = 600 - \text{PosicionDeseadaY}$$

Ecuación 15: Calibración respecto a Y para la BBDD de prueba.

$$\text{PosicionCalibradaX} = 1 + \text{PosicionDeseadaX}$$

Ecuación 16: Calibración respecto a X para la BBDD de prueba.

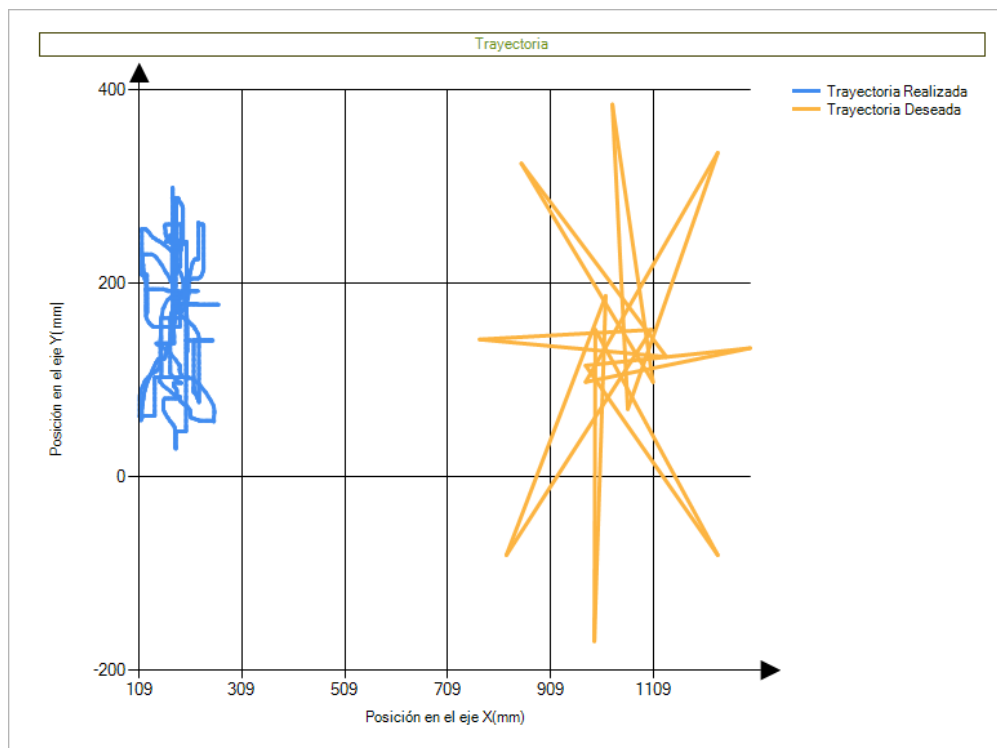


Gráfica 16: Trayectoria realizada con R. Fuente: Zatarain (2015).

En cambio, esta calibración para la BBDD real no sirve, ya que sigue habiendo errores de desviación, escalado e inversión (véase la gráfica 17). Estos errores



pueden ser distintos en las BBDD de PHYSIOBOT de los distintos hospitales y por esta razón se ha desarrollado un algoritmo que no utilice constantes para su calibración, sino que se realice a través de variables relativas a la posición realizada.



Gráfica 17: Trayectoria con la calibración desarrollada en R. Fuente: Elaboración propia.

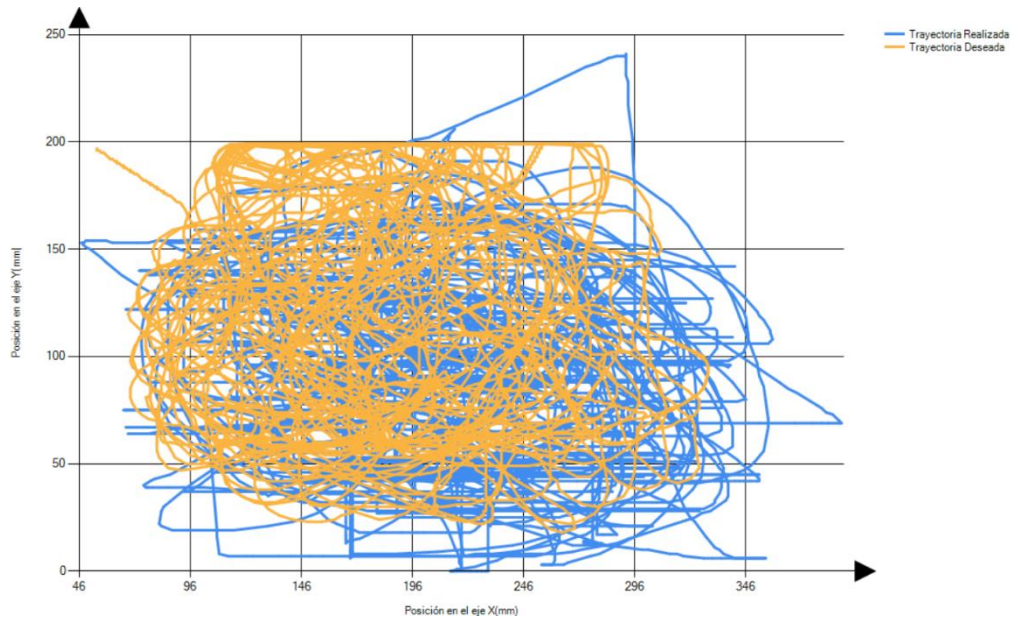
Para la realización de este algoritmo se eliminó la calibración implantada en R y se trabajó con los datos directamente almacenados en la BBDD real, estudiando las posiciones de cada terapia.

5.2 CALIBRACIÓN

Teniendo en cuenta que hay tres tipos de terapia (activo-asistida, pasivo-guiada y AVD), se estudia el error de cada uno de ellos para realizar la corrección y calibración de la posición deseada.

5.2.1 Calibración de la terapia activo-asistida

Se realiza la gráfica de la trayectoria realizada por el paciente frente a la trayectoria deseada (véase la gráfica 18) para determinar el error que tenemos.



Gráfica 18: Trayectoria realizada y trayectoria deseada sin calibrar de la terapia tipo 1. Fuente: Elaboración propia.

Se observa que no tenemos error de escalado, y estudiando las posiciones deseadas y las realizadas a lo largo del tiempo observamos que tampoco tenemos inversión del sentido de los ejes.

Respecto al error de distancia al origen, es difícil de determinar si tiene o no. En cambio, para la BBDD de prueba sí es necesario una calibración de desviación para este tipo de terapia. Por esta razón, a esta terapia se le va a aplicar una calibración para corregir este error.

Para ello, se han desarrollado ecuaciones de modificación de la posición deseada en función de la posición realizada, teniendo como resultado para cada uno de los ejes las ecuaciones 17 y 18:

$$P_{calibradoX} = PXo + escaladoX * (PuntoDeseadoX - minPdX)$$

Ecuación 17: Calibración sin inversión de las posiciones deseadas respecto a X.

$$P_{calibradoY} = PYo + escaladoY * (PuntoDeseadoY - minPdY)$$

Ecuación 18: Calibración sin inversión de las posiciones deseadas respecto a Y.

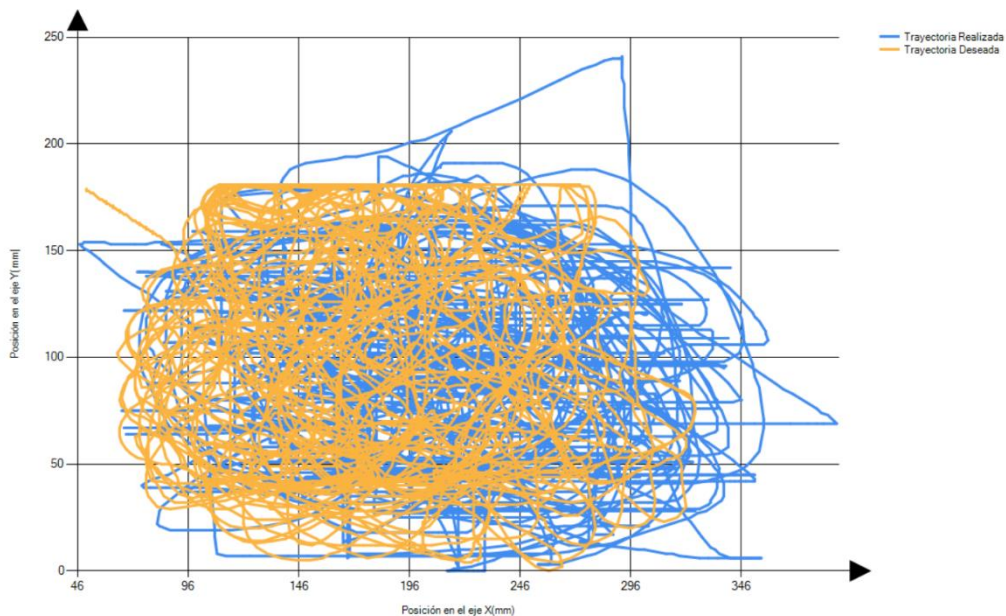
Donde para la ecuación 3:

- PcalibradoX**: es la coordenada calibrada en X del punto deseado.
- PXo**: es el mínimo valor respecto a X de los puntos realizados.
- escaladoX**: es el factor de escalado, en este tipo de terapia vale 1.
- PuntoDeseadoX**: coordenadas del punto deseado sin calibrar en el eje X.
- minPdX**: es el mínimo valor respecto a X de los puntos deseados.



En la ecuación 4, sus términos significan lo mismo que la ecuación 3 pero respecto al eje Y.

Como resultado de realizar esta calibración vemos un ligero desplazamiento de las posiciones deseadas que, en este caso, como los puntos deseados prácticamente coinciden con los puntos realizados la corrección es mínima (véase la gráfica 19).

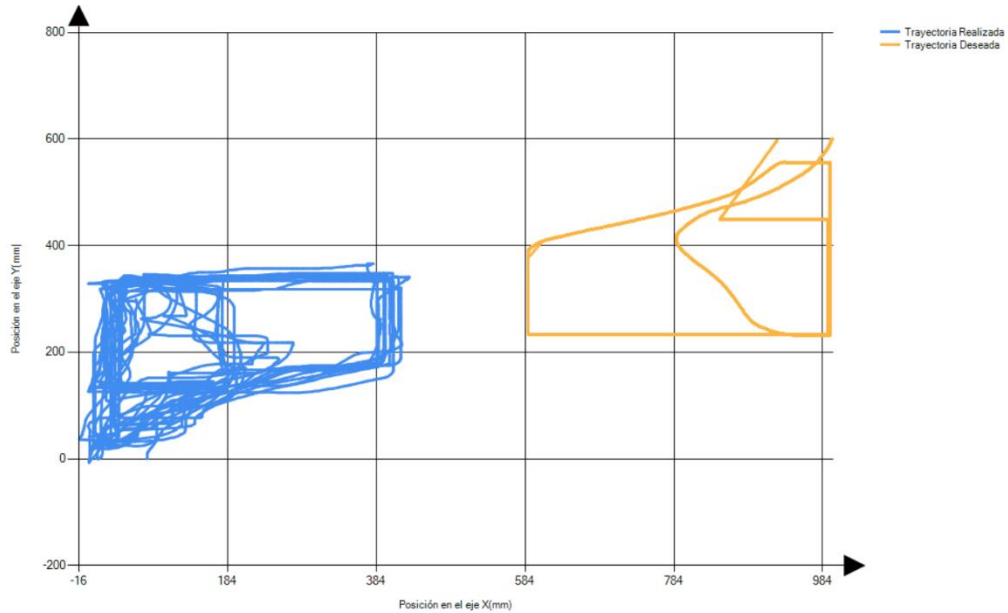


Gráfica 19: Trayectoria realizada y trayectoria deseada calibrada de la terapia tipo 1. Fuente: Elaboración propia.

5.2.2 Calibración de la terapia pasivo-guiada

Seguimos el mismo procedimiento que en el apartado anterior. Inicialmente, se grafica la trayectoria realizada y la trayectoria deseada para detectar los primeros errores (véase la gráfica 20).

Se puede ver a primera vista errores de distancia del origen y de inversión de los ejes, tanto del X como del Y. En cambio, no se aprecia que tengan distintos escalados, por lo que primero se corrigen estos dos errores y después se valora si hay diferente escalado.



Gráfica 20: Trayectoria realizada y trayectoria deseada sin calibrar de la terapia tipo 2. Fuente: Elaboración propia.

Para la corrección de estos errores se han desarrollado dos nuevas ecuaciones que tienen en cuenta la inversión de los ejes, y como las anteriores, se elaboran en función de las posiciones realizadas, teniendo como resultado las ecuaciones 19 y 20:

$$P_{calibradoX} = PXo - escaladoX * (PuntoDeseadoX - maxPdX)$$

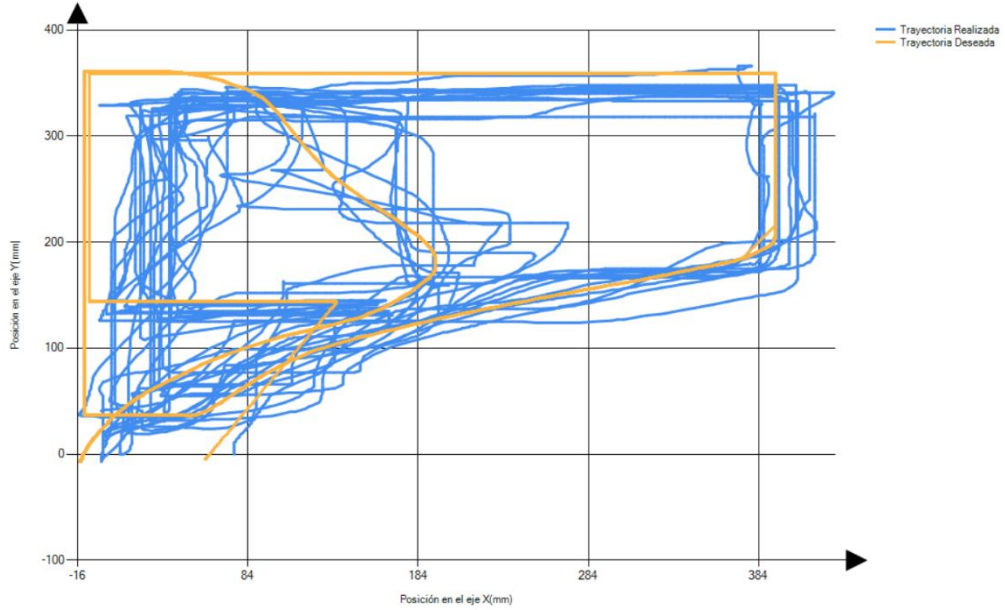
Ecuación 19: Calibración con inversión de las posiciones deseadas respecto a X.

$$P_{calibradoY} = PYo - escaladoY * (PuntoDeseadoY - maxPdY)$$

Ecuación 20: Calibración con inversión de las posiciones deseadas respecto a Y.

Las variables de estas ecuaciones mantienen el significado de las anteriores, siendo la nueva variable **maxPdX** (**maxPdY** si es respecto al eje Y) el valor máximo de los puntos deseados en el eje X (en el eje Y si es respecto a este eje).

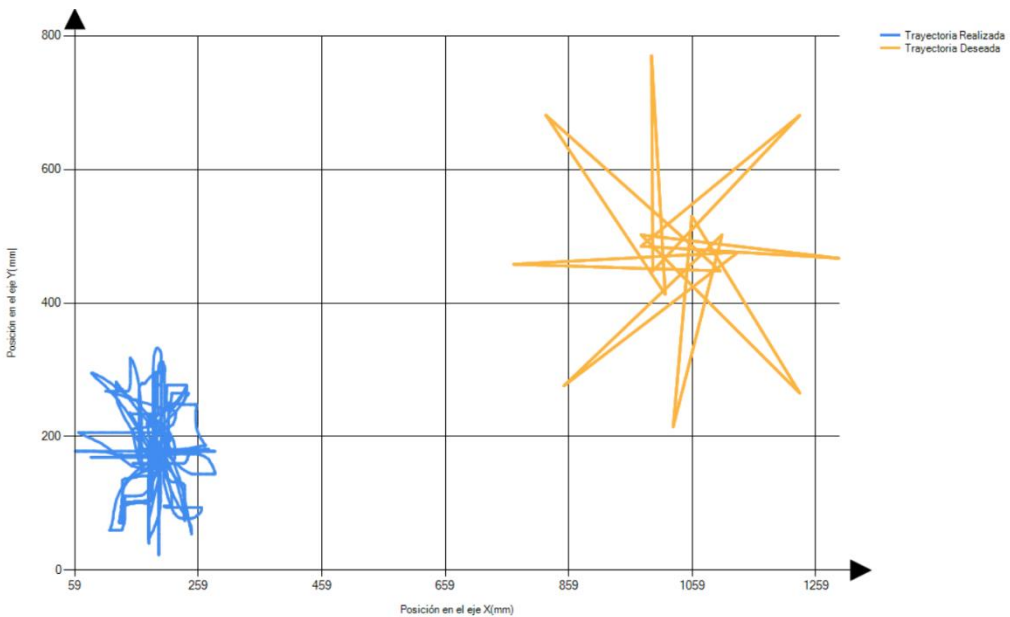
Como resultado de aplicar estas nuevas ecuaciones se consigue eliminar los errores mencionados anteriormente (véase la gráfica 21). Además, se observa que no hay distintas dimensiones, por lo que en las ecuaciones el escalado seguirá valiendo 1.



Gráfica 21: Trayectoria realizada y trayectoria deseada calibrada de la terapia tipo 2. Fuente: Elaboración propia.

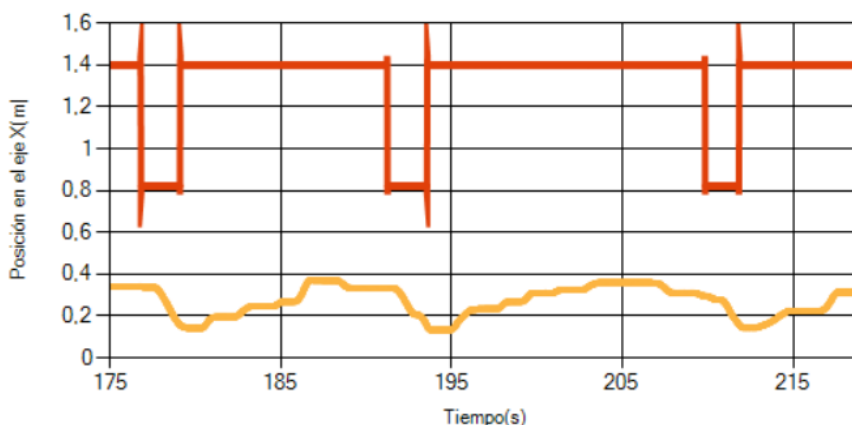
5.2.3 Calibración de la terapia AVD

Se repite el procedimiento. Se grafica las trayectorias de los puntos realizados y los deseados (véase la gráfica 22). Se observa que hay un error de desviación del origen, y aparentemente, de escalado. Al tratarse de una trayectoria deseada, casi simétrica (tanto del eje X como del eje Y), es necesario estudiar las posiciones deseadas y realizadas a lo largo del tiempo respecto a cada uno de los ejes.



Gráfica 22: Trayectoria realizada y trayectoria deseada sin calibrar de la terapia tipo 3. Fuente: Elaboración propia.

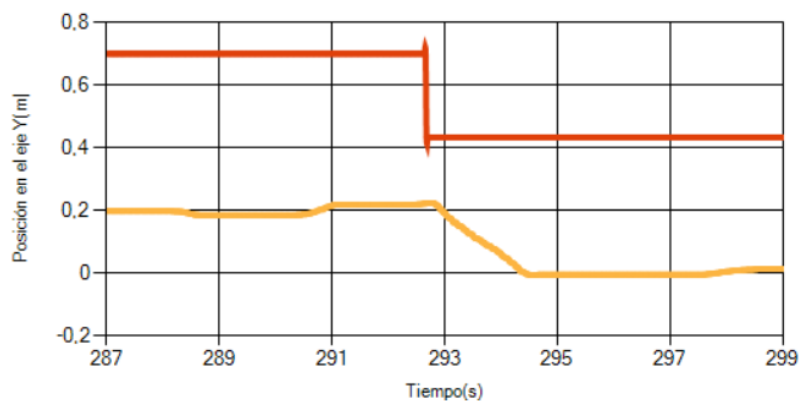
Se realiza la gráfica de las posiciones del eje X en el tiempo (véase la gráfica 23) y se observa que cuando las posiciones deseadas se desplazan hacia la izquierda (disminuye el valor X), las posiciones realizadas tienden a desplazarse también hacia la izquierda (aumenta el valor X). Es por ello que el eje X no se encuentra invertido.



Gráfica 23: Puntos realizados y puntos deseados respecto al eje X frente al tiempo en una terapia de tipo 3 sin calibrar. Fuente: Elaboración propia.

Además, se observa que hay distinto escalado, ya que las terapias de tipo AVD consisten en coger y arrastrar un ítem. Para coger este ítem, las coordenadas de X e Y del joystick tienen que ser muy similares a las posiciones que se deben alcanzar, lo cual como vemos en las gráficas 23 y 24 no sucede.

Se realiza también el estudio de las posiciones respecto a Y a lo largo del tiempo (véase la gráfica 24) y se observa que no hay inversión de ejes, ya que cuando las posiciones deseadas descienden, la posición realizada también lo hace.



Gráfica 24: Puntos realizados y puntos deseados respecto al eje Y frente al tiempo en una terapia de tipo 3 sin calibrar. Fuente: Elaboración propia.



Universidad de Valladolid

Es decir, se necesita una calibración sin inversión de ejes, con ajuste de desviación de datos y de escalado. Para ello se emplean las ecuaciones 3 y 4 ya desarrolladas, pero esta vez el escalado no vale 1, sino que se han desarrollado unas ecuaciones (véanse las ecuaciones 21 y 22) para calcular su valor en función de las posiciones realizadas:

$$\text{escaladoX} = (\max PX - PXo) / (\max PdX - \min PdX)$$

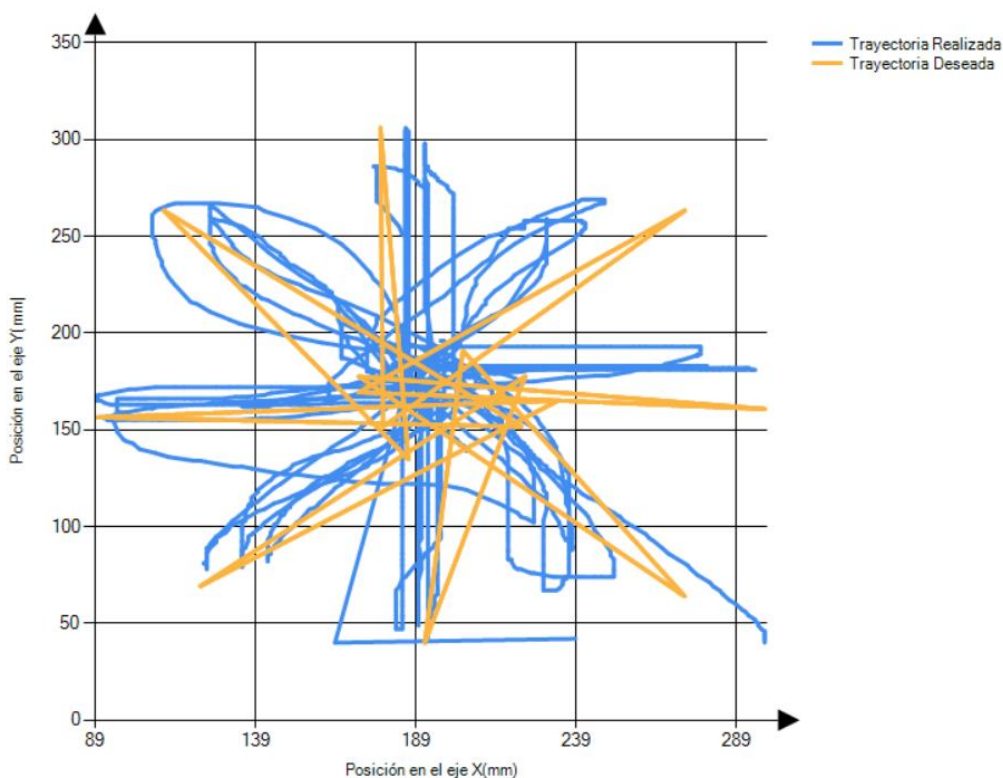
Ecuación 21: Escalado de las posiciones deseadas respecto a X.

$$\text{escaladoY} = (\max PY - PYo) / (\max PdY - \min PdY)$$

Ecuación 22: Escalado de las posiciones deseadas respecto a Y.

Donde los nuevos términos, **maxPX** y **maxPY**, simbolizan el máximo valor de las posiciones realizadas en el eje X y en el eje Y respectivamente.

Como resultado de aplicar las ecuaciones 17 y 18 para la desviación de las posiciones y las ecuaciones 21 y 22 para el escalado, obtenemos la trayectoria deseada ajustada a la trayectoria realizada (véase la gráfica 25).



Gráfica 25: Trayectoria realizada y trayectoria deseada calibrada de la terapia tipo 3. Fuente: Elaboración propia.



6 EVALUACIÓN DE LAS MÉTRICAS

En este apartado se van a evaluar las métricas que se han implantado, ya que cuando fueron desarrolladas por Zatarain (2015) no se comprobó su efectividad, solo su correcto funcionamiento.

Las métricas implantadas en la aplicación que realiza el estudio de una única terapia (trayectoria, velocidades respecto a los ejes y fuerzas respecto a los ejes) no necesitan de comprobación, ya que estas métricas plasman la información de la BBDD sin apenas modificar los datos, solo para calibrarlos o para obtener las medidas de fuerza.

Para comprobar la efectividad de los estadísticos en la aplicación que determina la evolución de los pacientes, se ha realizado un estudio con los datos de sujetos que han sufrido un accidente cerebrovascular y que han utilizado PHYSIOBOT.

Se trata de una población que ya ha utilizado PHYSIOBOT durante mucho tiempo, por lo que las métricas deberían mostrar estabilidad o una ligera mejora. En el Anexo II se adjuntan dos informes de estos pacientes para realizar la comprobación de los estadísticos.

El estadístico de marcador suele mantenerse en un rango, pero tiene muchos picos y es muy inestable, por lo que no es la mejor métrica para determinar la evolución de los pacientes.

La métrica de porcentaje de error por otro lado sí que es un buen estadístico, ya que no se produce ningún pico pronunciado y se ve la estabilidad de los pacientes. Hay en determinados pacientes en que el error en el eje Y de esta métrica si se producen picos. Esto es porque se ha hecho el análisis con terapias de tipo 2, por lo que, si se hace un recorrido en el que el paciente tiene que hacer muchos picos, este suele “recortarles” dando lugar a un mayor error.

El estadístico de media y desviación también es representativo de la estabilidad de los pacientes, aunque en él sí que se presentan picos más pronunciados que en el error, sobre todo en la métrica de la distancia media entre las posiciones realizadas y las deseadas.

Lo mismo sucede con el estadístico de facilidad de movimiento. En él se ve una ligera mejora de los pacientes, pero los valores oscilan entre un rango muy grande.

Respecto a los estadísticos de tiempo de reacción y número de temblores, se producen picos muy pronunciados que, aunque se puede apreciar la estabilidad de los pacientes, a causa de estos valores la información que se puede extraer es muy poca.



Universidad de Valladolid

Los estadísticos de las simetrías direccionales respecto a los ejes representan exactamente el estado de los pacientes y sin picos, siendo uno de los mejores estadísticos para analizar la evolución.

Lo mismo sucede con la simetría direccional de la velocidad y el ángulo. Se tratan de métricas muy representativas de los pacientes, en especial la simetría direccional de la velocidad respecto a los ejes.

Finalmente, en los estadísticos de evolución de fuerza respecto a los ejes sucede lo mismo que con las métricas de porcentaje de error, ya que representan el estado del paciente, pero se producen picos por que el usuario tiende a “atajar”.

En resumen, las métricas en las que hay una menor variabilidad y por tanto se aprecia mejor la evolución de los pacientes son: simetrías direccionales respecto a los ejes y simetría direccional de la velocidad y el ángulo. Seguido de estas, están las métricas en las que se sigue apreciando la evolución del paciente, aunque tienen mayor variabilidad. Estas son: porcentaje de error, marcador, facilidad de movimiento, media y desviación y evolución de fuerza.

Por otro lado, están las métricas de tiempo de reacción y número de temblores que tienen mayor variabilidad, y por consiguiente son en las que peor se aprecia la evolución de los pacientes. Sin embargo, de ellas se puede extraer otro tipo de información como la capacidad de atención del paciente. Por ejemplo, si en una terapia se observa que el paciente ha estado muchas veces parado, se puede interpretar que el sujeto se ha distraído en esa terapia.



7 CONCLUSIONES

Atendiendo a los objetivos planteados en el presente TFG, se van a presentar las conclusiones sobre estos:

Primero, se ha cumplido el objetivo principal de desarrollar un entorno software que permita el análisis de la BBDD de PHYSIOBOT. Más concretamente, la primera aplicación consigue analizar mediante representación de gráficas las principales características de una única terapia, además de poder almacenar dicha gráfica en formato .PNG.

La segunda aplicación permite determinar la evolución de los pacientes a través de las diversas métricas desarrolladas por CARTIF. También elabora informes representando las gráficas de estos estadísticos y mostrando sus datos más relevantes.

Se ha realizado la simulación de implantación de la aplicación con una BBDD de terapias reales, realizando la correspondiente calibración para futuras implantaciones.

Se ha comprobado la utilidad y efectividad de los estadísticos, obteniendo como conclusión que los mejores son las simetrías direccionales (tanto de posiciones, velocidades y ángulo), seguido del porcentaje de error, facilidad de movimiento, evolución de fuerza y distancia media. En el resto de los estadísticos se ha concluido que se obtiene menos información o son más sensibles.

Además, para posibles líneas futuras se ha realizado una documentación detallada del código, así como una guía de su funcionamiento. Estas posibles líneas de futuro y mejoras que puede tener el presente TFG son:

- **Elaborar un instalador.** Un ejecutable que permita descargar e instalar las dos aplicaciones software y realizar la conexión a la BBDD. Esto haría mucho más fácil su implantación.
- **Implantar el software en otra plataforma robótica.** El entorno software se ha desarrollado para PHYSIOBOT, pero sería interesante probar su utilidad en otros sistemas robotizados de rehabilitación.
- **Desarrollar nuevas métricas.** Crear o implantar nuevas métricas en las aplicaciones software que determinen mejor el estado y evolución del paciente que las ya desarrolladas.



8 REFERENCIAS BIBLIOGRÁFICAS

- Angulo, I. N., Rodríguez, M., Iáñez, E., Azorín, J. M. (2016). Efecto en la clasificación de imaginación motora a partir del EEG al aplicar tDCS en la corteza motora y el cerebelo. Libro de Actas CASEIB 2016. XXXIV Congreso Anual de la Sociedad Española de Ingeniería Biomédica. Valencia, 23 al 25 de noviembre de 2016, 249-252.
- Antelis, J.M., Montesano, L., Ramos-Murguialday, A., Birbaumer, N. y Minguez, J. (2017). Decoding Upper Limb Movement Attempt from EEG Measurements of the Contralateral aMotor Cortex in Chronic Stroke Patients. *IEEE Transactions on Biomedical Engineering*, 64(1), 99–111. <https://doi.org/10.1109/TBME.2016.2541084>
- Barrios, L. J., Hornero, R., Pérez-Turiel, J., Pons, J. L., Vidal, J. y Azorín, J. M. (2017a). Estado del Arte en Neurotecnologías para la Asistencia y la Rehabilitación en España: Tecnologías Fundamentales. *Revista Iberoamericana de Automática e Informática Industrial RIAI*, 14(4), 346-354. <https://doi.org/10.1016/j.riai.2017.06.003>
- Barrios, L. J., Minguillón, J., Perales, F. J., Ron-Angevin, R., Solé-Casals, J. y Mañanas, M. A. (2017b). Estado del arte en neurotecnologías para la asistencia y la rehabilitación en España: Tecnologías auxiliares, transferencia tecnológica y aplicación clínica. *Revista Iberoamericana de Automática e Informática industrial*, 14(4), 355-361. <https://doi.org/10.1016/j.riai.2017.06.004>
- Baldan, F., Binder, I., Istenic, R., Baba, A., Kiper, P. y Turolla, A. (2015). Surface EMG driven robotic hand rehabilitation: preliminary results from a feasibility study. *Physiotherapy*, 101(1) 1549. <https://doi.org/10.1016/j.physio.2015.03.1544>
- Villar, B., Viñas, P., Pérez-Turiel, J., Fraile, J.C. y Gordaliza, A., (2020). Influence on the user's emotional state of the graphic complexity level in virtual therapies based on a robot-assisted neuro-rehabilitation platform. *Computer Methods and Programs in Biomedicine*, 190, 1-12. <https://doi.org/10.1016/j.cmpb.2020.105359>
- Fernández, B. (2018). *Desarrollo de nuevas terapias y evaluación del estado emocional en la interacción física de una persona y el robot de rehabilitación neuromotora PHYSIOBOT (Trabajo Fin de Grado)*. Universidad de Valladolid, Valladolid. <http://uvadoc.uva.es/handle/10324/32514>
- Buch, E., Weber, C., Cohen, L. G., Braun, C., Dimyan, M. A., Ard, T. y Birbaumer, N. (2008). Think to move: A neuromagnetic brain-computer interface



- (BCI) system for chronic stroke. *Stroke*, 39(3), 910–917. <https://doi.org/10.1161/STROKEAHA.107.505313>
- Calabrò, R. S., Russo, M., Naro, A., Milardi, D., Balletta, T., Leo, A., Filoni, S. y Bramanti, P. (2016). Who may benefit from armeo power treatment? A neurophysiological approach to predict neurorehabilitation outcomes. *PM&R*, 8(10), 971-978. <https://doi.org/10.1016/j.pmrj.2016.02.004>
- Costa, A., Iáñez, E., Úbeda, A., Hortal, E., Del Ama, A., Gil-Agudo, A. y Azorín, J.M. (2016). Decoding the attentional demands of gait through EEG gamma band features. *PLoS One*, 11(4), 1–21. <https://doi.org/10.1371/journal.pone.0154136>
- Cisnal, A., Lobo, V., Moreno, V., Fraile, J. C., Alonso, R. y Turiel, J. P. (2018). Robhand, un exoesqueleto de mano para la rehabilitación neuromotora aplicando terapias activas y pasivas. *Actas de las XXXIX Jornadas de Automática. Badajoz, 5-7 de septiembre de 2018*, 34-41.
- Cisnal, A., Moreno, V., Pérez-Turiel, J., Alonso, R., Fraile-Marinero, J. C. y Lobo, V. (2019). Estrategia para el control háptico, basado en electromiografía, de un exoesqueleto de mano para neurorrehabilitación. 11º Simposio CEA de Bioingeniería. <http://dx.doi.org/10.4995/CEABioIng.2019.10038>
- Da Silva, L., Valero, L., Velasco, F. y Ron, R. (2011). Variables psicológicas en el control de interfaces cerebro-computadora. *Psicothema*, 23(4), 745-751.
- Da Silva-Sauer, L., Valero-Aguayo, L., de la Torre-Luque, A., Ron-Angevin, R., Varona-Moya, S. (2016). Concentration on performance with P300-based BCI systems: A matter of interface features. *Applied Ergonomics*, 52, 325-332. <https://doi.org/10.1016/j.apergo.2015.08.002>
- Da Silva-Sauer, L., Torre-Luque, A., De La Silva, J.S.C. y Fernández-Calvo, B. (2019). New perspectives for cognitive rehabilitation: Could brain-computer interface systems benefit people with dementia? *Psychology & Neuroscience*, 12 (1), 25–37. <https://doi.org/10.1037/pne0000154>
- Date, C. J. (2001). *Introducción a los sistemas de bases de datos*. Pearson Educación.
- Gonzalo, M. (2014). Evaluabilidad de los programas del robot amadeo en la rehabilitación de la mano del hemipléjico. *Revista Terapia Ocupacional Galicia*, 11(20), 1–22.
- Gil-Agudo, A., Pons, J.L., Alcobendas-Maestro, M., Lozano-Berrio, V., Benito-Penalva, J., García, S. y del-Ama, A.J. (2017). Estudio sobre la viabilidad clínica del exoesqueleto EXO-H2 para la rehabilitación de la marcha en



pacientes con lesión medular incompleta. *IX Congreso Iberoamericano de Tecnología de Apoyo a la Discapacidad*, 672-674.

Fraile, J. C., Perez-Turiel, J., Baeyens, E., Viñas, P., Alonso, R., Cuadrado, A., Franco-Martín, M., Parra, E., Ayuso, L., García-Bravo, F., Nieto, F., y Laurentiu, L. (2016). E2Rebot: a robotic platform for upper limb rehabilitation in patients with neuromotor disability. *Advances in Mechanical Engineering*, 8(8), 1-13. <https://doi.org/10.1177%2F1687814016659050>

Gunduz, A., Kumru, H. y Pascual-Leone, A. (2014). Outcomes in spasticity after repetitive transcranial magnetic and transcranial direct current stimulations. *Neural Regeneration Research*, 9(7), 712-718. <https://doi.org/10.4103/1673-5374.131574>

Hortal, E., Planelles, D., Resquin, F., Climent, J. M., Azorín, J. M. y Pons, J. L. (2015). Using a brain-machine interface to control a hybrid upper limb exoskeleton during rehabilitation of patients with neurological conditions. *Journal of neuroengineering and rehabilitation*, 12(92), 1-16. <https://doi.org/10.1186/s12984-015-0082-9>

Jordanic, M., Rojas-Martínez, M., Mañanas, M. A. y Alonso, J. F. (2016). Spatial distribution of HD-EMG improves identification of task and force in patients with incomplete spinal cord injury. *Journal of NeuroEngineering and Rehabilitation*, 13(41), 1-11. <https://doi.org/10.1186/s12984-016-0151-8>

Knuth, S., Passon, A., Dähne, F., Niedeggen, A., Schmehl, I. y Schauer, T. (2017). Adaptive arm weight support using a cable-driven robotic system. En J. Ibáñez, J. González-Vargas, J.M. Azorín, M. Akay y J.L. Pons (Eds.), *Converging Clinical and Engineering Research on Neurorehabilitation II* (pp. 1317-1321). Springer. https://doi.org/10.1007/978-3-319-46669-9_215

Kumru, H., Benito-Penalva, J., Valls-Sole, J., Murillo, N., Tormos, J.M., Flores, C. y Vidal, J. (2016). Placebo-controlled study of rTMS combined with Lokomat gait training for treatment in subjects with motor incomplete spinal cord injury. *Experimental Brain Research*, 234(12), 3447-3455. <https://doi.org/10.1007/s00221-016-4739-9>

López-Larraz, E., Sarasola-Sanz, A., Irastorza-Landa, N., Birbaumer, N. y Ramos-Murguialday, A. (2018). Brain-machine interfaces for rehabilitation in stroke: A review. *NeuroRehabilitation*, 43(1), 77-97. <https://doi.org/10.3233/NRE-172394>

Martínez-Cagigal, V., Gomez-Pilar J., Álvarez D. y Hornero, R. (2017). An Asynchronous P300-Based Brain-Computer Interface Web Browser for



- Severely Disabled People. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 25(8), 1332-1342. <https://doi.org/10.1109/TNSRE.2016.262338>
- Martínez-Cagigal, V., Núñez, P. y Hornero, R. (2016). Spectral Regression Kernel Discriminant Analysis for P300 Speller Based Brain-Computer Interfaces. En J. Ibáñez, J. González-Vargas, J.M. Azorín, M. Akay y J.L. Pons (Eds.), *Converging Clinical and Engineering Research on Neurorehabilitation II* (pp. 789-793). Springer. https://doi.org/10.1007/978-3-319-46669-9_129
- Mañanas M. A., Rojas-Martínez M. y Alonso J.F. (2016). Towards the application of HD-EMG decomposition in clinical practice. *Clinical Neurophysiology*, 127(6), 2534-2541. <https://doi.org/10.1016/j.clinph.2016.02.005>
- Minguillon, J., Lopez-Gordo, M. A. y Pelayo, F. (2016). Detection of attention in multi-talker scenarios: A fuzzy approach. *Expert Systems with Applications*, 64, 261-268. <https://doi.org/10.1016/j.eswa.2016.07.042>
- Minguillon, J., Lopez-Gordo, M. A. y Pelayo, F. (2017). Trends in EEG-BCI for daily-life: Requirements for artifact removal. *Biomedical Signal Processing and Control*, 31, 407-418. <https://doi.org/10.1016/j.bspc.2016.09.005>
- Quiroz, J. (2003). El modelo relacional de bases de datos. *Boletín de Política Informática* (6), 53-61.
- Ramos-Murguialday, A., Broetz, D., Rea, M., Läer, L., Yilmaz, O., Brasil, F. L. Liberati, G., Curado, M.R., García-Cossio, E., Vyziotis, A., Cho, W., Cohen, L.G. y Birbaumer, N. (2013). Brain-machine interface in chronic stroke rehabilitation: A controlled study. *Annals of Neurology*, 74(1), 101-108. <https://doi.org/10.1002/ana.23879>
- Riener, R., Lünenburger, L., Maier, I. C., Colombo, G. y Dietz, V. (2010). Locomotor training in subjects with sensori-motor deficits: an overview of the robotic gait orthosis lokomat. *Journal of Healthcare Engineering*, 1(2), 197-216. <https://doi.org/10.1260/2040-2295.1.2.197>
- Rocon, E., Belda-Lois, J.M., Ruiz, A.F., Manto, M., Moreno, J. C. y Pons, J.L. (2007). Design and Validation of a Rehabilitation Robotic Exoskeleton for Tremor Assessment and Suppression, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 15(3), 367-378. <https://doi.org/10.1109/TNSRE.2007.903917>



- Ron-Angevin, R. y Díaz-Estrella, A. (2009). Brain-computer interface: Changes in performance using virtual reality techniques. *Neuroscience letters*, 449(2), 123-127. <https://doi.org/10.1016/j.neulet.2008.10.099>
- Salazar-Varas, R. Costa, Á., Iáñez, E., Úbeda, A., Hortal, E. y Azorín, J.M. (2015). Analyzing EEG signals to detect unexpected obstacles during walking. *Journal of NeuroEngineering and Rehabilitation*, 12(101), 1-15. <https://doi.org/10.1186/s12984-015-0095-4>
- Silberschatz, A., Korth, H. F. y Sudarshan, S. (2002). *Fundamentos de bases de datos*. Mc Graw Hill Education.
- Tónic, T.J.D., Savić, A.M., Vidaković, A.S., Rodić, S.Z., Isaković, M.S., Rodríguez-de-Pablo, C. y Konstantinović, L.M. (2017). Sistema robótico ArmAssist versus terapia convencional combinada para la rehabilitación de la extremidad superior posterior al accidente cerebrovascular: un ensayo clínico aleatorizado. *BioMed research international*.
- Úbeda, A., Hortal, E., Iáñez, E., Pérez-Vidal, C. y Azorín, J.M. (2015). Assessing Movement Factors in Upper Limb Kinematics Decoding from EEG Signals. *PLoS One*, 10(5), 1-12. <https://doi.org/10.1371/journal.pone.0128456>
- Velasco-Álvarez, F., Fernández-Rodríguez, Á., Díaz-Estrella, A., Blanca-Mena, M. J. y Ron-Angevin, R. (2018). Control strategies of a brain-controlled wheelchair using two mental tasks. En P. Diez (Ed.), *Smart Wheelchairs and Brain-computer Interfaces. Mobile Assistive Technologies* (pp.345-368). Academic Press. <https://doi.org/10.1016/B978-0-12-812892-3.00014-5>
- Zatarain, M. (2015). *Gestión y análisis de una base de datos para una aplicación de ingeniería biomédica* (Trabajo de Fin de Grado). Universidad de Valladolid, Valladolid. <http://uvadoc.uva.es/handle/10324/14408>



ANEXOS

ANEXO I: Manual de usuario.....	I
ANEXO II: Informes de pacientes.....	VII

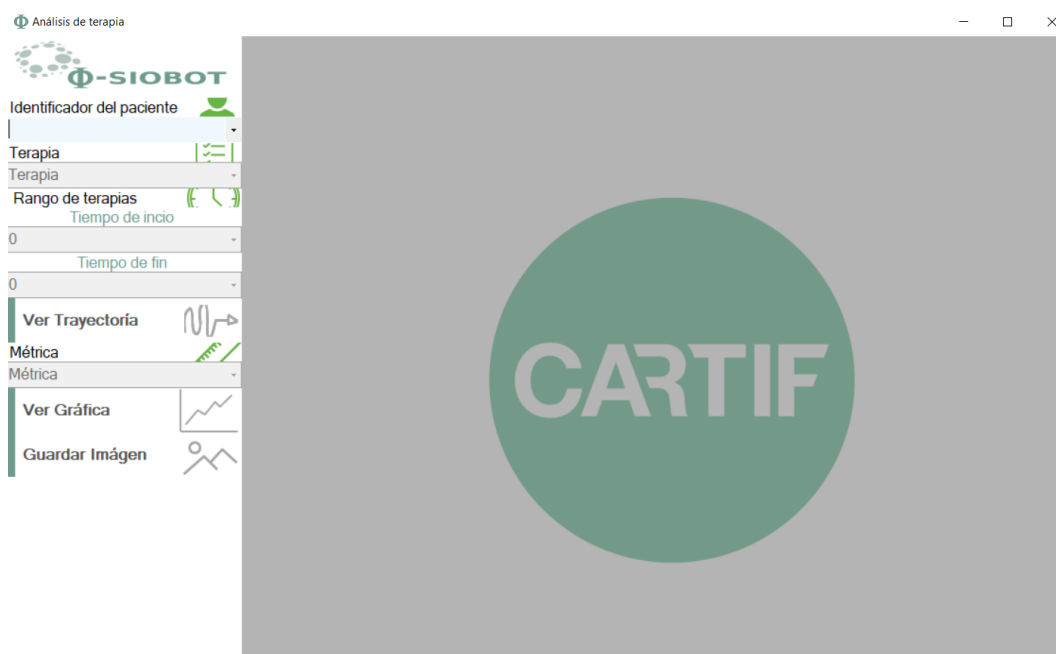


ANEXO I: MANUAL DE USUARIO

Para el uso de estos programas hay que enlazar una base de datos, ya que no hay ninguna vinculada inicialmente por dos razones. La primera de ellas es que una base de datos de prueba pesa alrededor de 1,5 GB, es decir, no se puede adjuntar como anexo. La segunda es que las bases de datos de CARTIF contienen datos personales, luego no se puede realizar una conexión directa con ellas para mantener el anonimato de los pacientes.

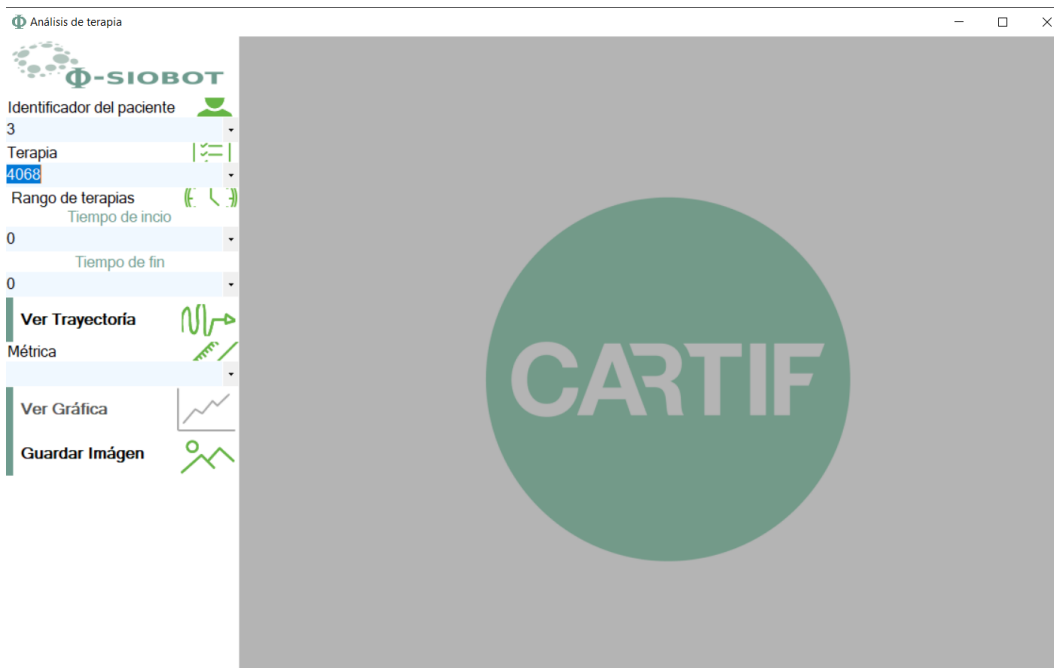
ANÁLISIS DE UNA TERAPIA:

Para iniciar el programa hay que acceder a la carpeta TFG_Señales y abrir el ejecutable (el archivo que acabe en .exe). Se abrirá la siguiente interfaz:

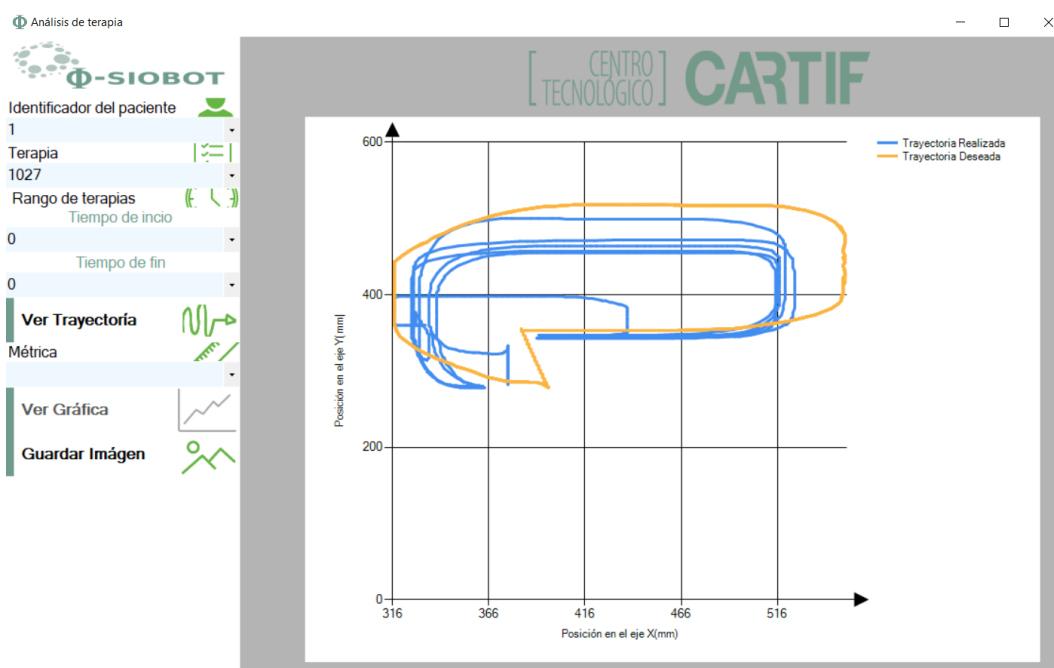


Todos los desplegables están bloqueados al iniciar al programa salvo el identificador del paciente. Se van desbloqueando el resto a medida que se rellenan los disponibles.

Se selecciona uno de los pacientes disponibles y a continuación se desbloqueará el desplegable de Terapia. Se selecciona una Terapia y todos los elementos se desbloquearán salvo el botón Ver Gráfica:

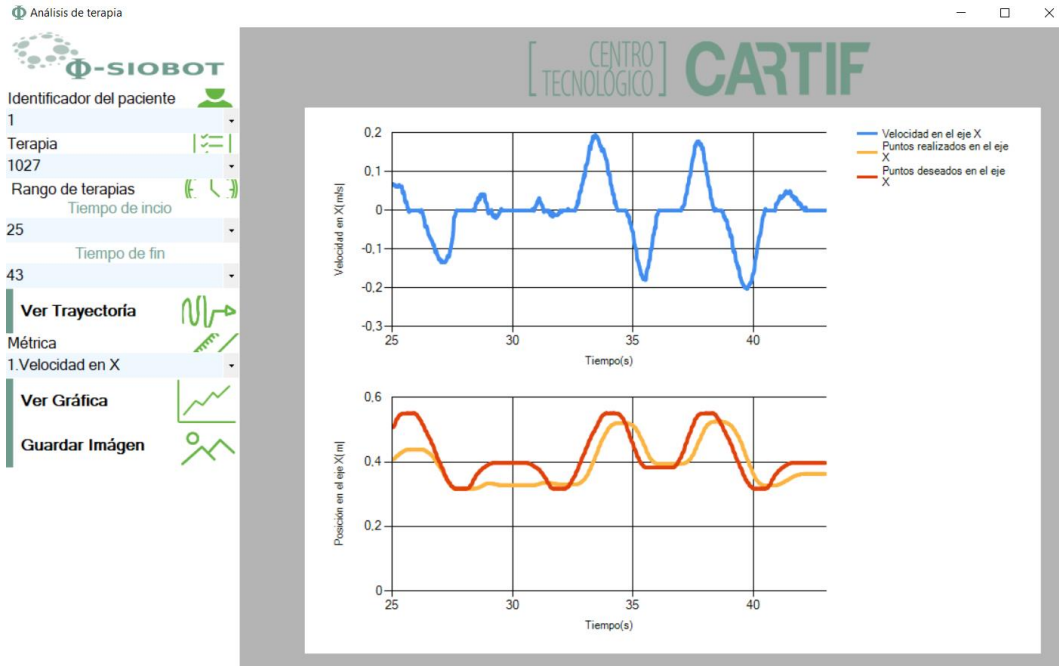


Dando al botón de Ver Trayectoria, se visualizará el itinerario realizado (Trayectoria realizada) y el mostrado en pantalla (Trayectoria deseada):



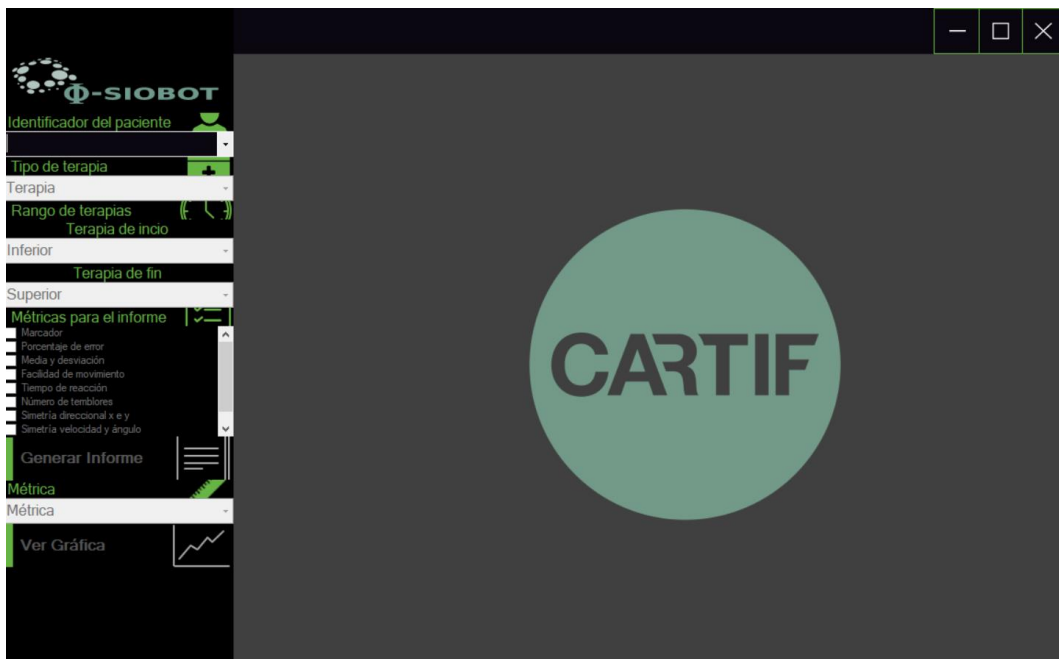
Si se da al botón Guardar Imagen, la gráfica que hay en la interfaz se guardará en formato PNG donde se haya abierto el ejecutable y a continuación se mostrará por pantalla.

Abriendo el desplegable Métrica y seleccionando una de ellas se desbloqueará el botón Ver Gráfica. Para poder utilizar las métricas hay que elegir un intervalo de tiempo, escribiendo el tiempo de inicio y el tiempo final con un rango coherente (el tiempo inicial tiene que ser menor que el tiempo final):



EVOLUCIÓN DE PACIENTES

Para iniciar el programa hay que acceder a la carpeta Evolucion y abrir el ejecutable (el archivo que acabe en .exe). Se abrirá la siguiente interfaz:

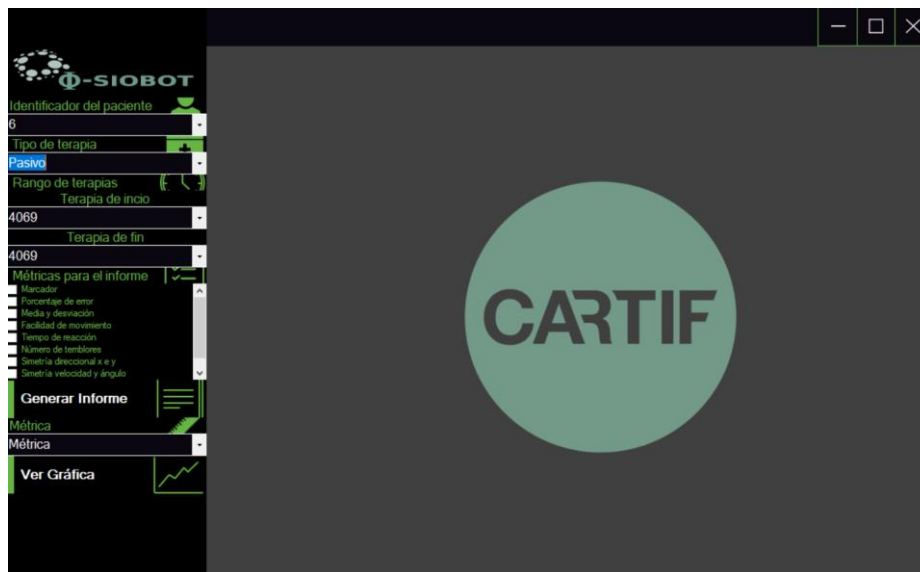


El procedimiento es el mismo que la aplicación de análisis de una terapia. Inicialmente solo está desbloqueado el identificador del paciente, el resto de los elementos se van desbloqueando a medida que completamos los disponibles,



Universidad de Valladolid

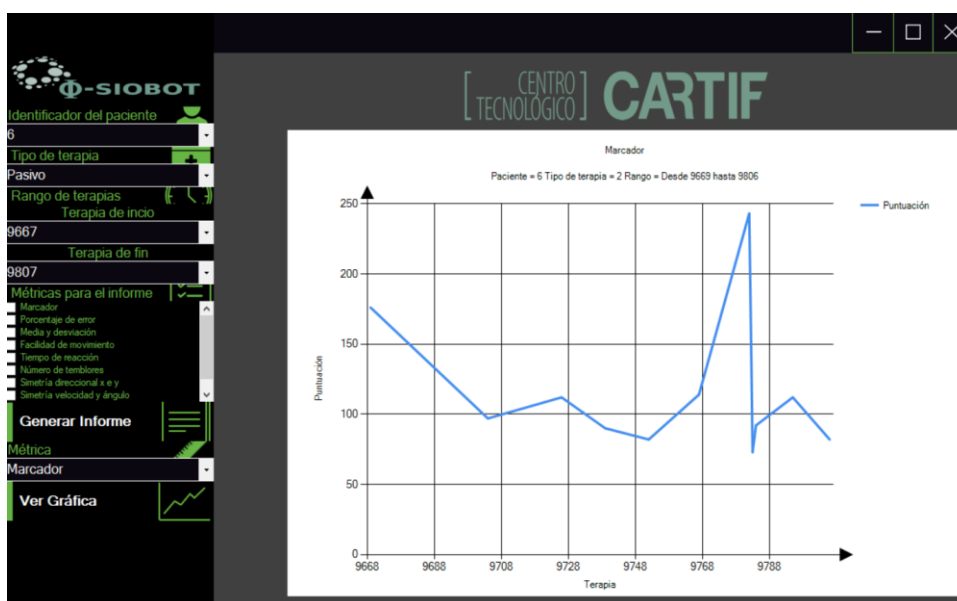
Se escoge uno de los pacientes, y a continuación se escoge el tipo de terapia que se quiere estudiar. Al seleccionar el tipo de terapia todos los elementos quedan desbloqueados:



A continuación, hay que seleccionar el rango de terapias para el que se quiere estudiar la evolución del paciente. La terapia de inicio siempre tiene que ser menor que la terapia de fin, de lo contrario la aplicación nos avisará con un mensaje de error.

Llegado a este punto, se tienen dos posibles opciones: ver la gráfica de una métrica en pantalla o realizar el informe de varias métricas con los datos más relevantes (Primer valor, último valor, valor más alto, valor más bajo, etc.).

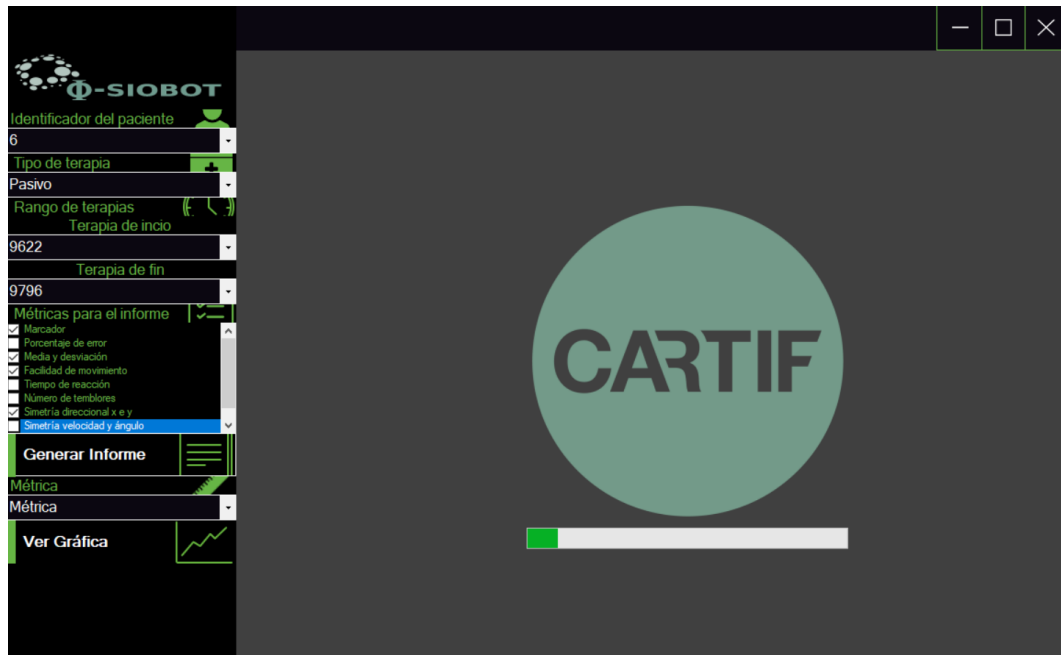
Para el primer caso, en la desplegable métrica seleccionamos la métrica que se quiera estudiar, y a continuación se da al botón ver gráfica:





Universidad de Valladolid

En caso de que se quiera realizar el informe, se seleccionan de la lista de métricas para el informe las que se quieran incluir en este, y a continuación se dará al botón Generar Informe:



El informe generado se guardará en el lugar donde se haya abierto el ejecutable con el nombre de Evolución_del_paciente_(identificador del paciente).pdf y a continuación se mostrará por pantalla (ANEXO II).



Universidad de Valladolid

ANEXO II: INFORMES DE PACIENTES

EVOLUCIÓN DEL PACIENTE 9



Fecha del Informe: 28/07/2020 20:17:32

El tipo de terapia es: 2



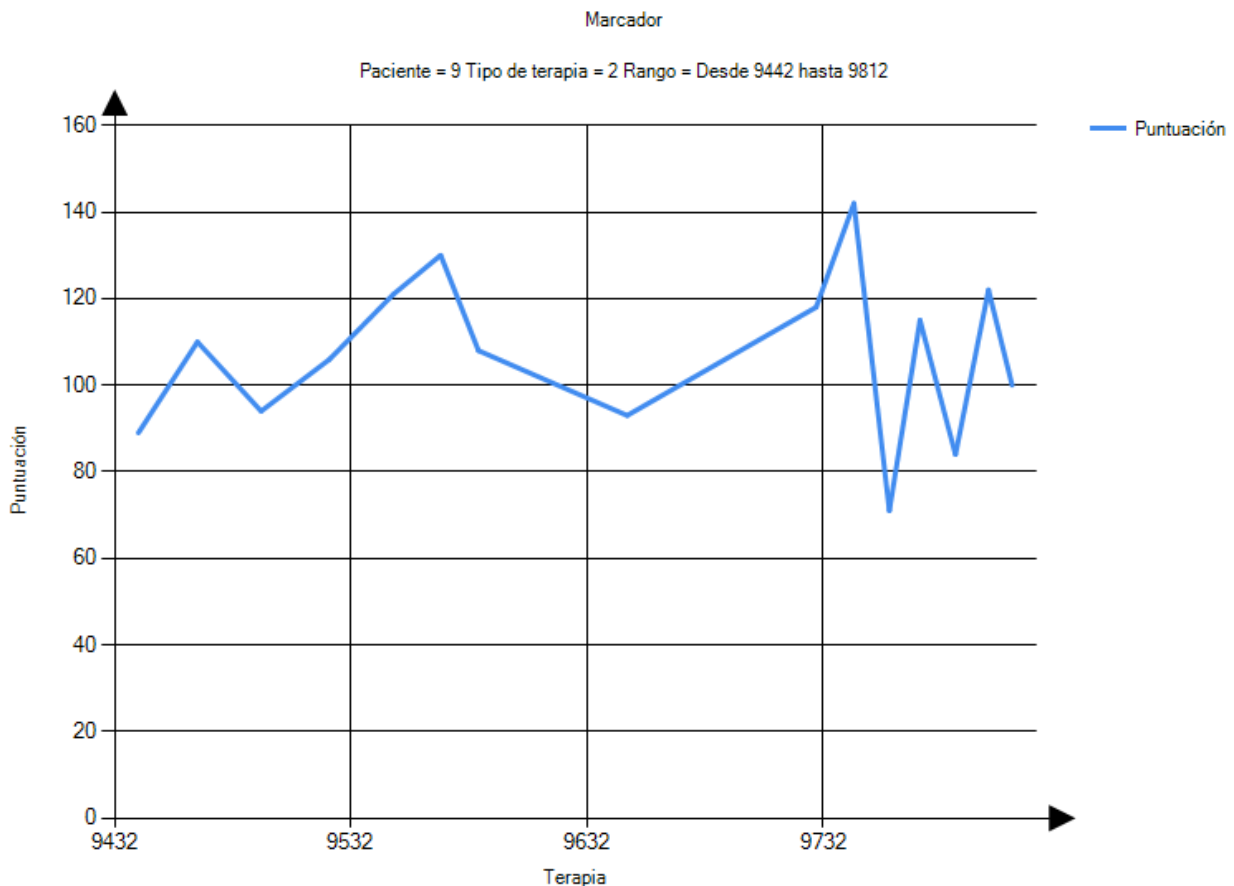
Paciente 9/ Marcador / Terapia tipo 2

En esta métrica se va a ver la cantidad de puntos que consigue el paciente en el juego del ratón. Cada vez que el paciente atrapa al ratón recibe puntos, de forma que cuantas más veces haya atrapado al ratón, mayor puntuación tendrá. La puntuación puede variar bastante dependiendo de la dificultad.

El índice de esta métrica son los puntos del juego.

Valor	Puntuación	Número de terapia	Fecha
Primero	89	9442	29/05/2017 12:50:00
Último	100	9812	10/07/2017 12:59:26
Máximo	142	9745	28/06/2017 12:53:32
Mínimo	71	9760	30/06/2017 13:03:59

Puntuación media: 106,8667





Paciente 9/ Porcentaje de error / Terapia tipo 2

Se va a visualizar la diferencia que hay en un punto entre x e y realizados por el paciente con respecto a los puntos deseados x e y, con el objetivo de poder observar qué movimiento realiza mejor. Para ello se hace la media del punto que ha movido el paciente menos el punto deseado, entre el punto deseado tanto en x como en y.

El índice es el error relativo en "x" y en "y", y se medirá en %.

Valor	Error en X	Número de terapia	Fecha
Primero	5,556301	9442	29/05/2017 12:50:00
Último	6,05217	9802	07/07/2017 12:58:30
Máximo	8,158521	9467	31/05/2017 12:50:24
Mínimo	4,823664	9760	30/06/2017 13:03:59

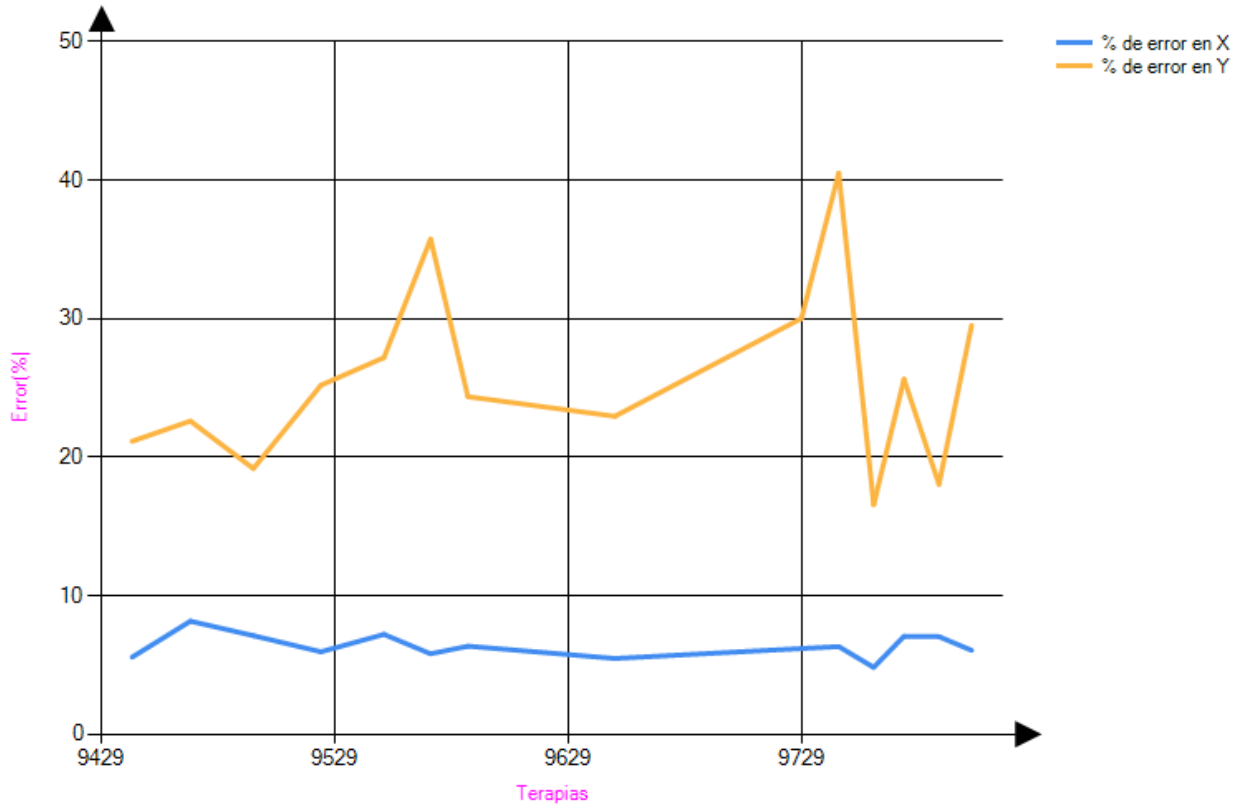
Media de Error en X: 6,363719

Valor	Error en Y	Número de terapia	Fecha
Primero	16,55438	9760	30/06/2017 13:03:59
Último	22,59971	9467	31/05/2017 12:50:24
Máximo	40,4965	9745	28/06/2017 12:53:32
Mínimo	16,55438	9760	30/06/2017 13:03:59

Media de Error en Y: 25,60978

Error Cometido

Paciente = 9 Tipo de terapia = 2 Rango = Desde 9442 hasta 9812





Paciente 9/ Media y desviación / Terapia tipo 2

Media: es la suma de la diferencia de los puntos geométricos en el plano en x y en y recorridos por el paciente con respecto a los puntos deseados, dividido entre el número de puntos recorridos.

Desviación típica: es el sumatorio de la raíz cuadrada del cuadrado de la diferencia de los puntos geométricos en el plano en x y en y recorridos por el paciente con respecto a los puntos deseados, dividido entre el número de puntos recorridos.

Los índices que tendremos serán la media y la desviación típica de los puntos de cada terapia.

Valor	media	Número de terapia	Fecha
Primero	79,84226	9442	29/05/2017 12:50:00
Último	104,4528	9802	07/07/2017 12:58:30
Máximo	123,5968	9745	28/06/2017 12:53:32
Mínimo	65,85947	9760	30/06/2017 13:03:59

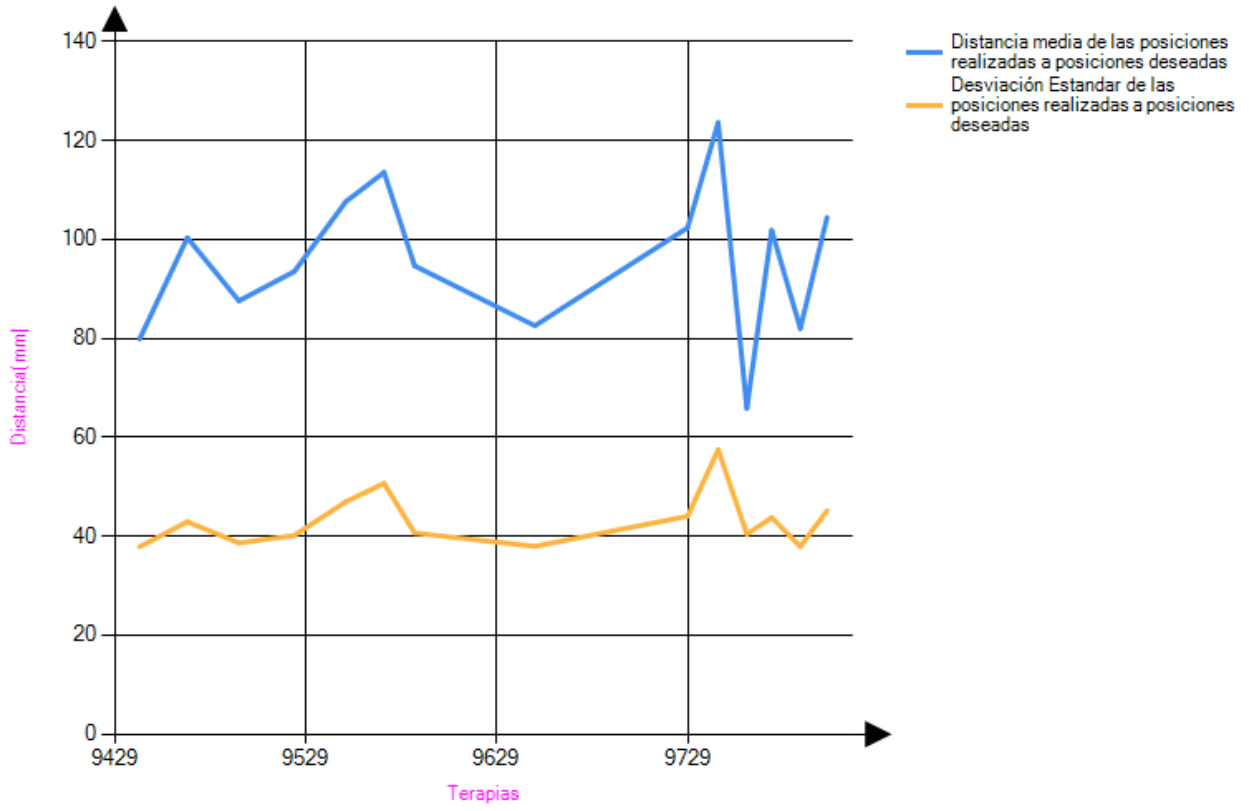
Media: 95,68968

Valor	Desviación	Número de terapia	Fecha
Primero	40,42295	9760	30/06/2017 13:03:59
Último	57,51307	9745	28/06/2017 12:53:32
Máximo	57,51307	9745	28/06/2017 12:53:32
Mínimo	37,86411	9442	29/05/2017 12:50:00

Media de desviación: 43,19748

Media y desviación

Paciente = 9 Tipo de terapia = 2 Rango = Desde 9442 hasta 9812



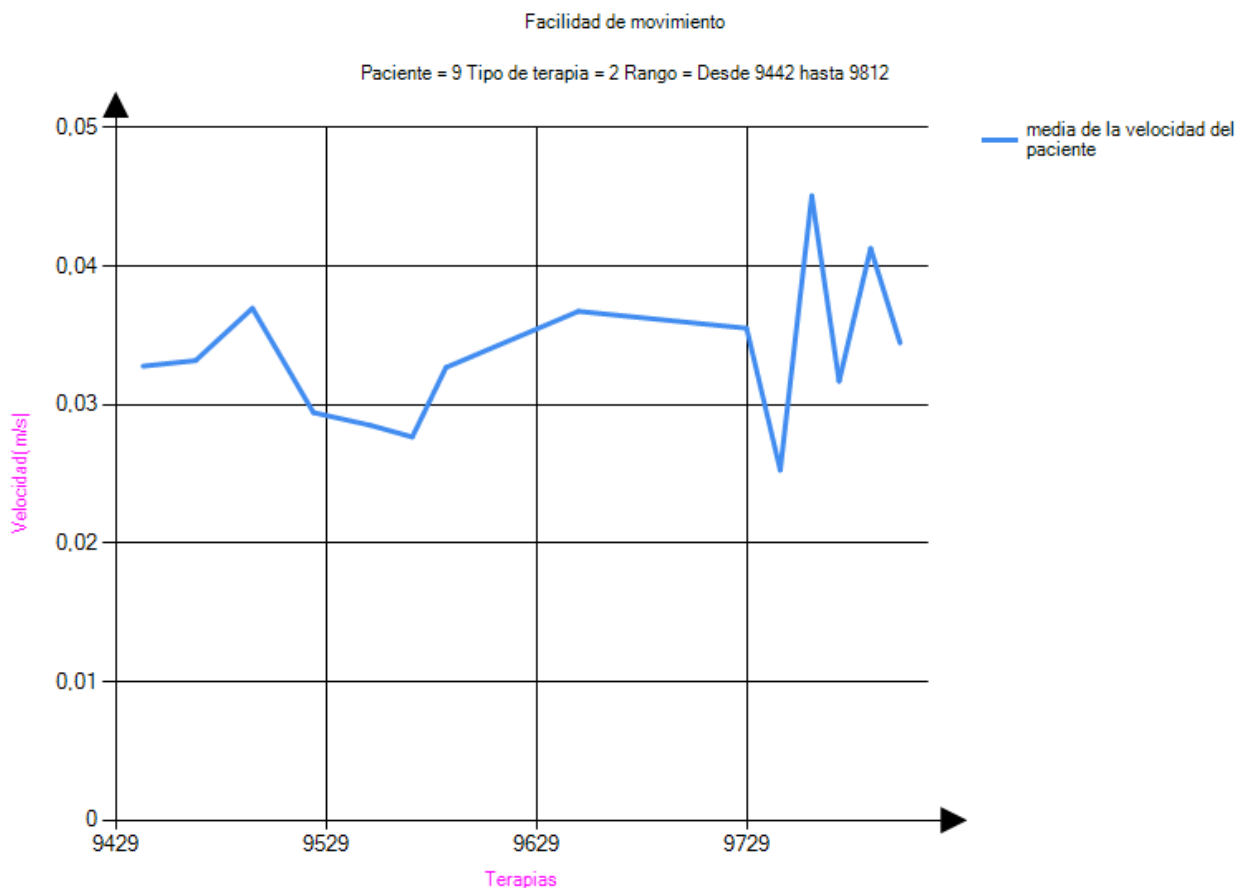
Paciente 9/ Facilidad de movimiento / Terapia tipo 2

Según varios estudios, la facilidad de movimiento se puede relacionar con a velocidad, de forma que cuanto más rápido tengas las capacidad de moverte, menos esfuerzo te costará realizar la tarea, y por lo tanto tendrás una mayor facilidad de movimiento.

El índice será la velocidad media medida en m/s.

Valor	Velocidad	Número de terapia	Fecha
Primero	0,03276835	9442	29/05/2017 12:50:00
Último	0,03444486	9802	07/07/2017 12:58:30
Máximo	0,04506231	9760	30/06/2017 13:03:59
Mínimo	0,02526712	9745	28/06/2017 12:53:32

Velocidad media: 0,03364996





Paciente 9/ Tiempo de reacción / Terapia tipo 2

Se denomina tiempo de reacción al tiempo que pasa desde que en la trayectoria deseada se produce un cambio de sentido hasta que en la trayectoria del robot se produce ese cambio de sentido también. Para ello se resta el tiempo en el que se produjo el cambio en el robot menos el tiempo en el que se produjo el cambio en el sentido deseado.

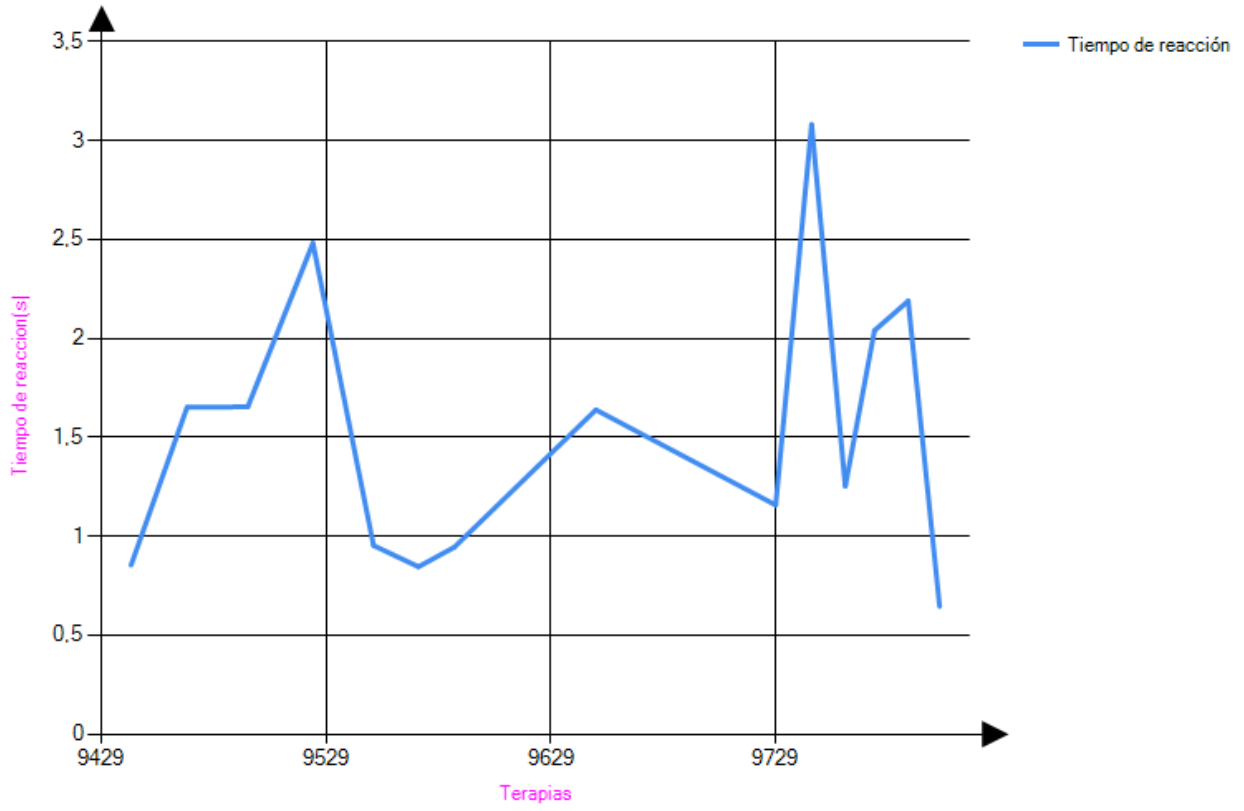
El índice que se obtiene será el tiempo de reacción y se medirá en segundos.

Valor	Tiempo de Reacción	Número de terapia	Fecha
Primero	0,8537401	9442	29/05/2017 12:50:00
Último	0,6451365	9802	07/07/2017 12:58:30
Máximo	3,079967	9745	28/06/2017 12:53:32
Mínimo	0,6451365	9802	07/07/2017 12:58:30

[Media del tiempo de reacción: 1,527664](#)

Tiempo de reacción

Paciente = 9 Tipo de terapia = 2 Rango = Desde 9442 hasta 9812



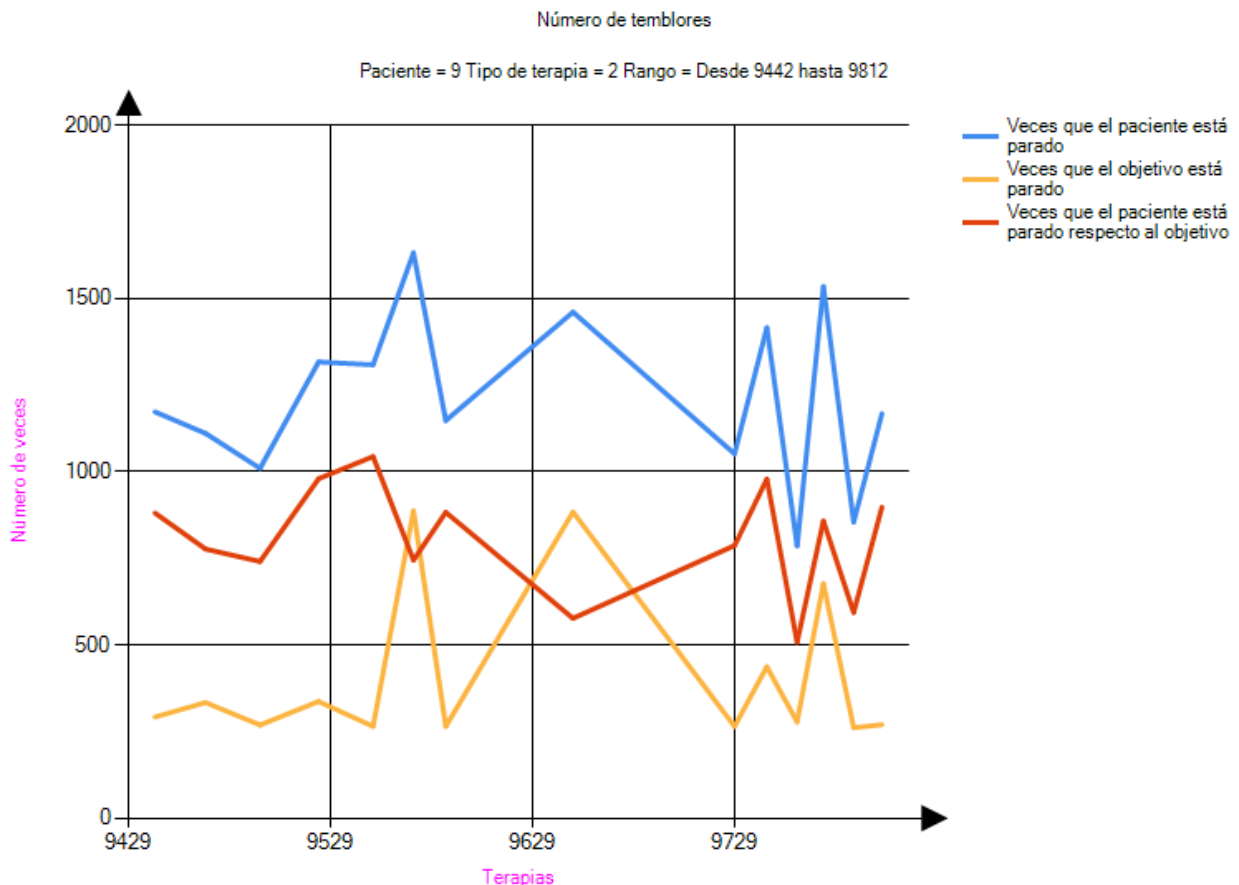
Paciente 9/ Número de temblores / Terapia tipo 2

Se observa cuántas veces ha estado parado el robot cuando era manejado por el fisioterapeuta o cuántas veces el ratón en la pantalla ha estado en una posición fija, y cuántas veces ha estado parado el paciente, así como la comparación de la mismas.

El índice es el número de veces que el robot está parado.

Valor	Relación Robot/Paciente	Número de terapia	Fecha
Primero	881	9442	29/05/2017 12:50:00
Último	898	9802	07/07/2017 12:58:30
Máximo	1044	9550	07/06/2017 12:55:44
Mínimo	508	9760	30/06/2017 13:03:59

Media del Número de temblores: 803,7143





Paciente 9/ Simetría direccional x e y / Terapia tipo 2

En el análisis estadístico de una serie temporal y en el procesamiento de la señal, la simetría direccional es una medida estadística de rendimiento de un modelo para predecir el cambio, positivo o negativo, de una serie temporal de un período de tiempo al siguiente.

El índice de la simetría direccionales es el porcentaje de simetría direccional(%).

Valor	Simetría direccional X	Terapia	Fecha
Primero	99	9442	29/05/2017 12:50:00
Último	99	9802	07/07/2017 12:58:30
Máximo	99	9802	07/07/2017 12:58:30
Mínimo	99	9442	29/05/2017 12:50:00

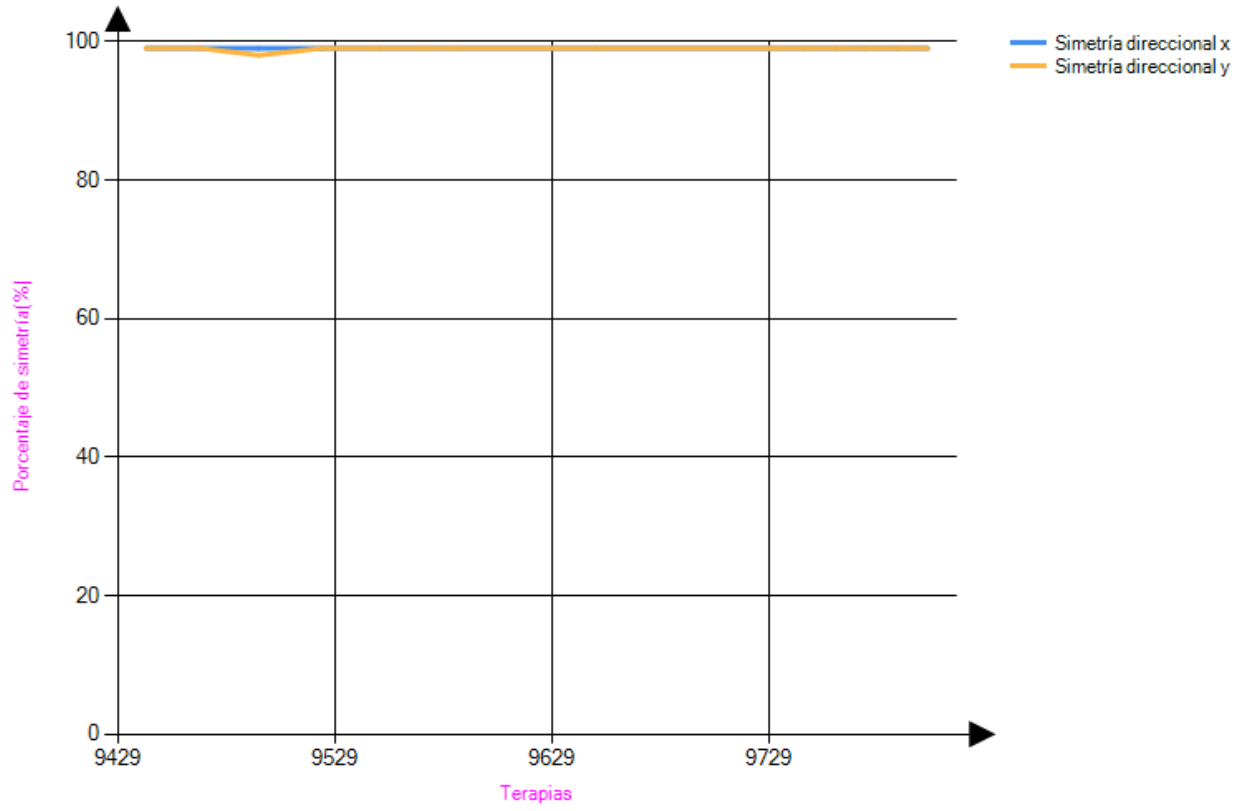
Media de la simetría direccional X: 99

Valor	Simetría direccional Y	Terapia	Fecha
Primero	99	9442	29/05/2017 12:50:00
Último	99	9802	07/07/2017 12:58:30
Máximo	99	9802	07/07/2017 12:58:30
Mínimo	98	9494	02/06/2017 12:35:21

Media de la simetría direccional Y: 98,92857

Simetría direccional en x y en y

Paciente = 9 Tipo de terapia = 2 Rango = Desde 9442 hasta 9812





Paciente 9/ Simetría velocidad y ángulo / Terapia tipo 2

Simetría direccional de la velocidad en x y en y, la magnitud de la velocidad es la semisuma al cuadrado de las velocidades x e y y el ángulo es el arcoseno de la velocidad con respecto a la magnitud y lo que se observa en la pantalla es que el porcentaje del ángulo sea el mismo del paciente y de la trayectoria objetivo.

Los índices son porcentajes de simetría(%).

Valor	Magnitud de velocidad	Terapia	Fecha
Primero	69	9442	29/05/2017 12:50:00
Último	67	9802	07/07/2017 12:58:30
Máximo	76	9570	09/06/2017 12:44:31
Mínimo	66	9494	02/06/2017 12:35:21

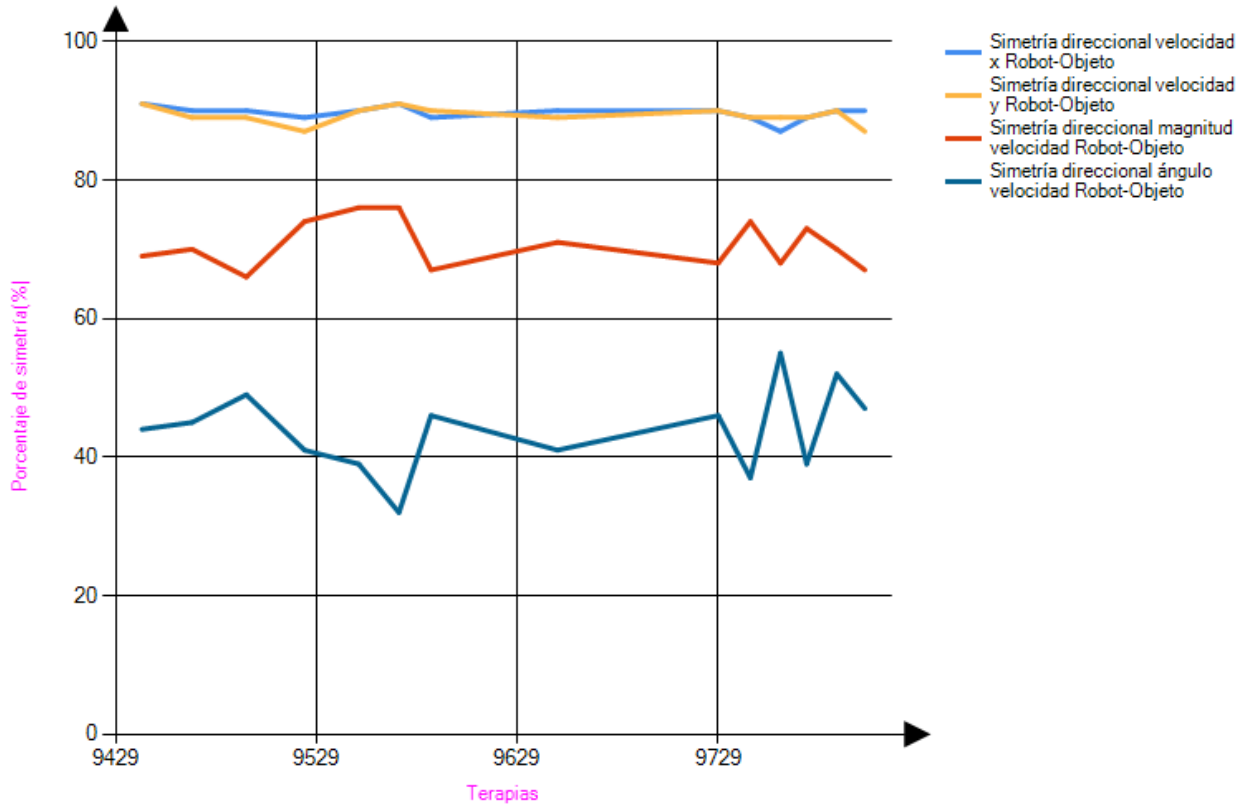
Media de la simetría de la velocidad: 70,64286

Valor	Simetría angular	Terapia	Fecha
Primero	49	9494	02/06/2017 12:35:21
Último	32	9570	09/06/2017 12:44:31
Máximo	55	9760	30/06/2017 13:03:59
Mínimo	32	9570	09/06/2017 12:44:31

Media de la simetría del ángulo: 43,78571

Simetría de velocidad y ángulo

Paciente = 9 Tipo de terapia = 2 Rango = Desde 9442 hasta 9812





Paciente 9/ Evolución de la fuerza / Terapia tipo 2

Esta medida va a marcar la evolución de la fuerza que tenga el paciente. Para ello lo que valoramos es no solo la fuerza que aplica el paciente, sino también la fuerza con la que le ayuda el robot. Para calcular la evolución vamos a tomar la fuerza que aplica el paciente y la vamos a dividir entre la suma de la fuerza que aplica el robot y la fuerza que aplica el paciente.

El índice será un porcentaje de la fuerza en x y en y que realiza el paciente con respecto a la fuerza que hace el paciente y la fuerza asistida que recibe.

Valor	Evolución en X	Terapia	Fecha
Primero	85,11382	9442	29/05/2017 12:50:00
Último	92,86902	9802	07/07/2017 12:58:30
Máximo	95,7106	9467	31/05/2017 12:50:24
Mínimo	85,11382	9442	29/05/2017 12:50:00

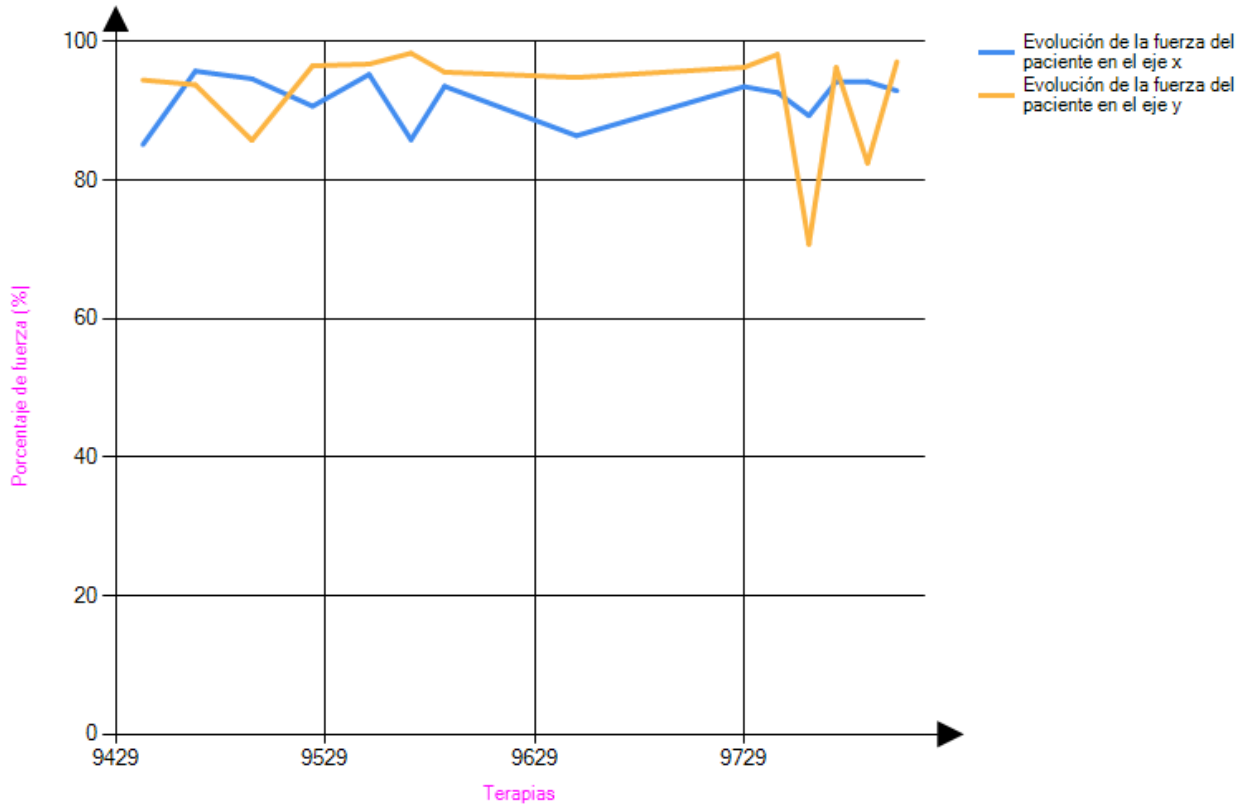
[Media de evolución en X: 91,68572](#)

Valor	Evolución en Y	Terapia	Fecha
Primero	94,41747	9442	29/05/2017 12:50:00
Último	93,73146	9467	31/05/2017 12:50:24
Máximo	98,29979	9570	09/06/2017 12:44:31
Mínimo	70,73007	9760	30/06/2017 13:03:59

[Media de evolución en Y: 92,61192](#)

Evolución de fuerza

Paciente = 9 Tipo de terapia = 2 Rango = Desde 9442 hasta 9812



EVOLUCIÓN DEL PACIENTE 12



Fecha del Informe: 28/07/2020 20:46:38

El tipo de terapia es: 2



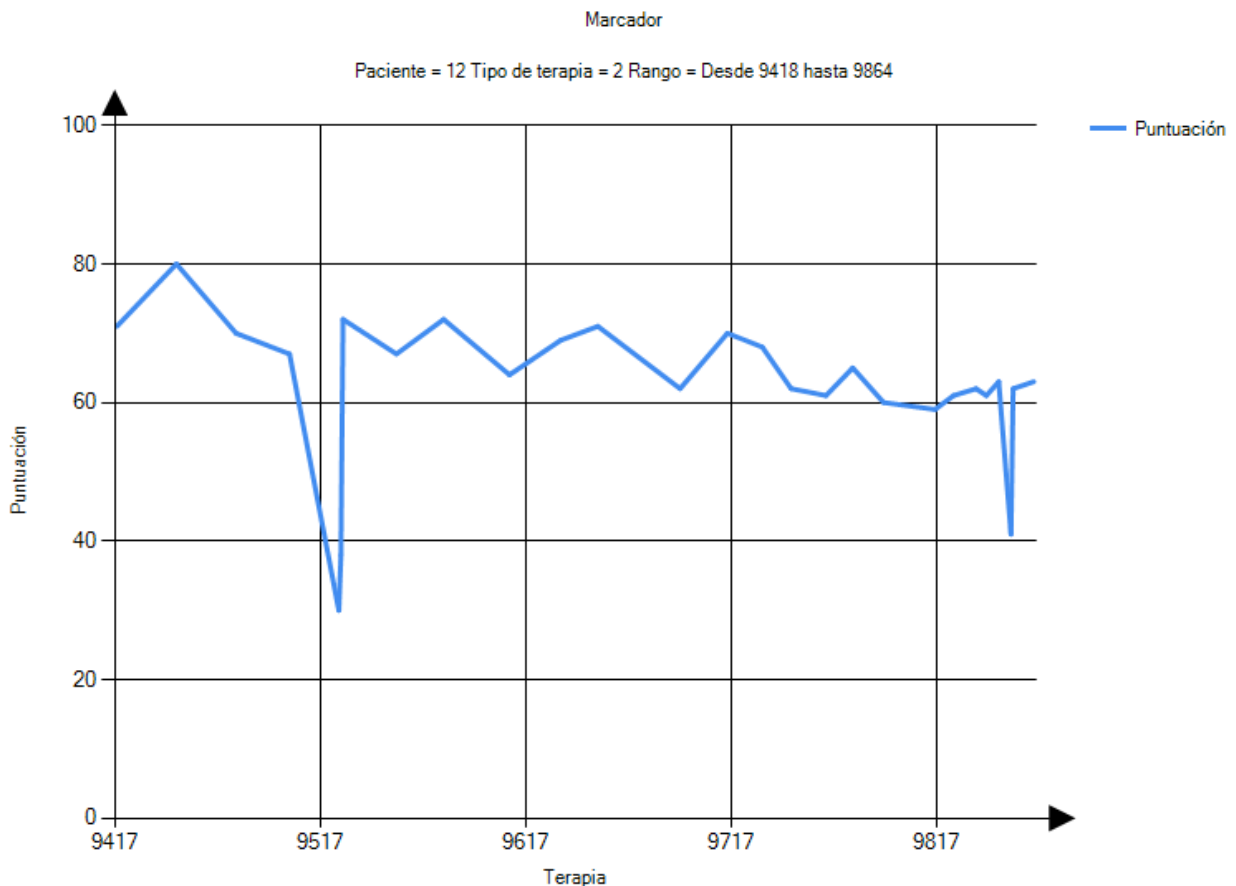
Paciente 12/ Marcador / Terapia tipo 2

En esta métrica se va a ver la cantidad de puntos que consigue el paciente en el juego del ratón. Cada vez que el paciente atrapa al ratón recibe puntos, de forma que cuantas más veces haya atrapado al ratón, mayor puntuación tendrá. La puntuación puede variar bastante dependiendo de la dificultad.

El índice de esta métrica son los puntos del juego.

Valor	Puntuación	Número de terapia	Fecha
Primero	71	9418	29/05/2017 9:09:22
Último	63	9864	24/07/2017 9:22:40
Máximo	80	9447	31/05/2017 9:19:48
Mínimo	30	9526	07/06/2017 9:15:49

Puntuación media: 62,62963





Paciente 12/ Porcentaje de error / Terapia tipo 2

Se va a visualizar la diferencia que hay en un punto entre x e y realizados por el paciente con respecto a los puntos deseados x e y, con el objetivo de poder observar qué movimiento realiza mejor. Para ello se hace la media del punto que ha movido el paciente menos el punto deseado, entre el punto deseado tanto en x como en y.

El índice es el error relativo en "x" y en "y", y se medirá en %.

Valor	Error en X	Número de terapia	Fecha
Primero	13,446	9418	29/05/2017 9:09:22
Último	12,76566	9854	21/07/2017 9:24:26
Máximo	14,85811	9836	13/07/2017 9:35:54
Mínimo	12,06489	9692	23/06/2017 9:25:42

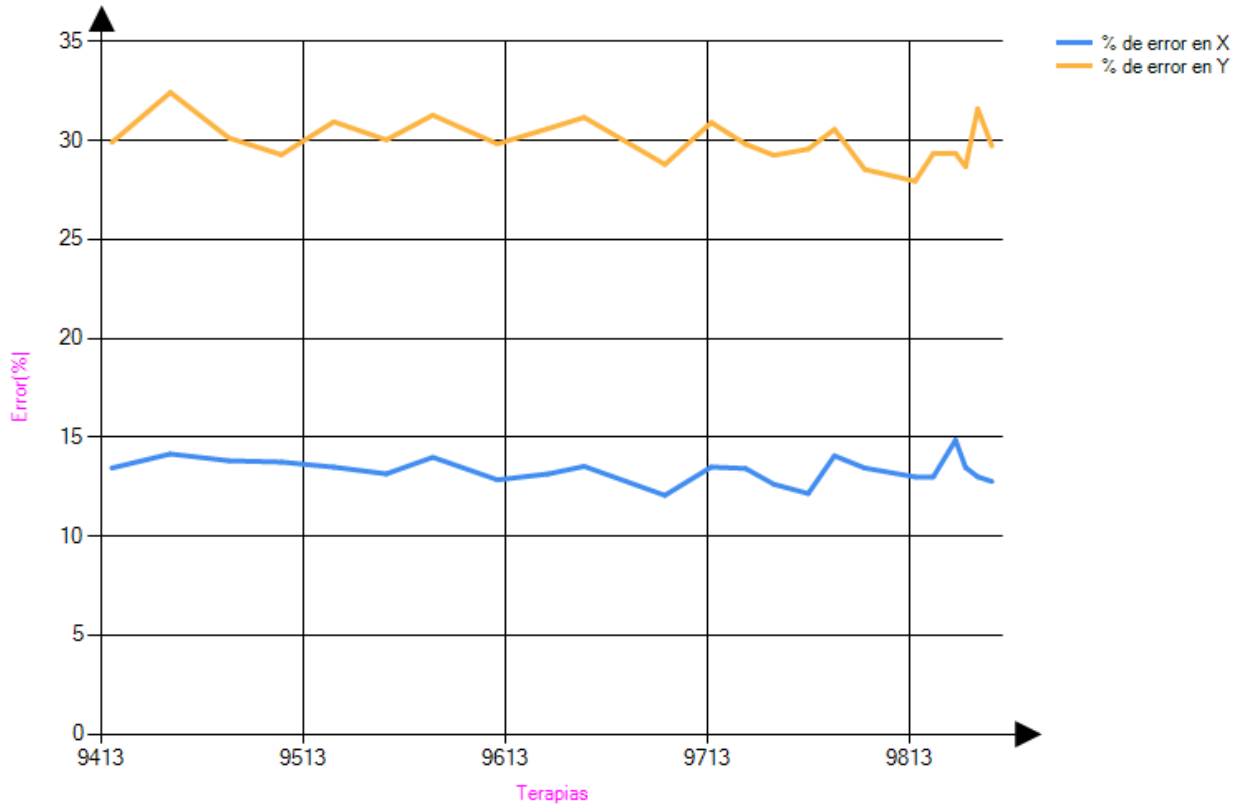
Media de Error en X: 13,33497

Valor	Error en Y	Número de terapia	Fecha
Primero	28,78361	9692	23/06/2017 9:25:42
Último	29,33972	9836	13/07/2017 9:35:54
Máximo	32,42147	9447	31/05/2017 9:19:48
Mínimo	27,93989	9816	10/07/2017 13:27:34

Media de Error en Y: 29,97852

Error Cometido

Paciente = 12 Tipo de terapia = 2 Rango = Desde 9418 hasta 9864





Paciente 12/ Media y desviación / Terapia tipo 2

Media: es la suma de la diferencia de los puntos geométricos en el plano en x y en y recorridos por el paciente con respecto a los puntos deseados, dividido entre el número de puntos recorridos.

Desviación típica: es el sumatorio de la raíz cuadrada del cuadrado de la diferencia de los puntos geométricos en el plano en x y en y recorridos por el paciente con respecto a los puntos deseados, dividido entre el número de puntos recorridos.

Los índices que tendremos serán la media y la desviación típica de los puntos de cada terapia.

Valor	media	Número de terapia	Fecha
Primero	76,39118	9418	29/05/2017 9:09:22
Último	90,22977	9854	21/07/2017 9:24:26
Máximo	90,69833	9528	07/06/2017 9:17:47
Mínimo	70,21412	9816	10/07/2017 13:27:34

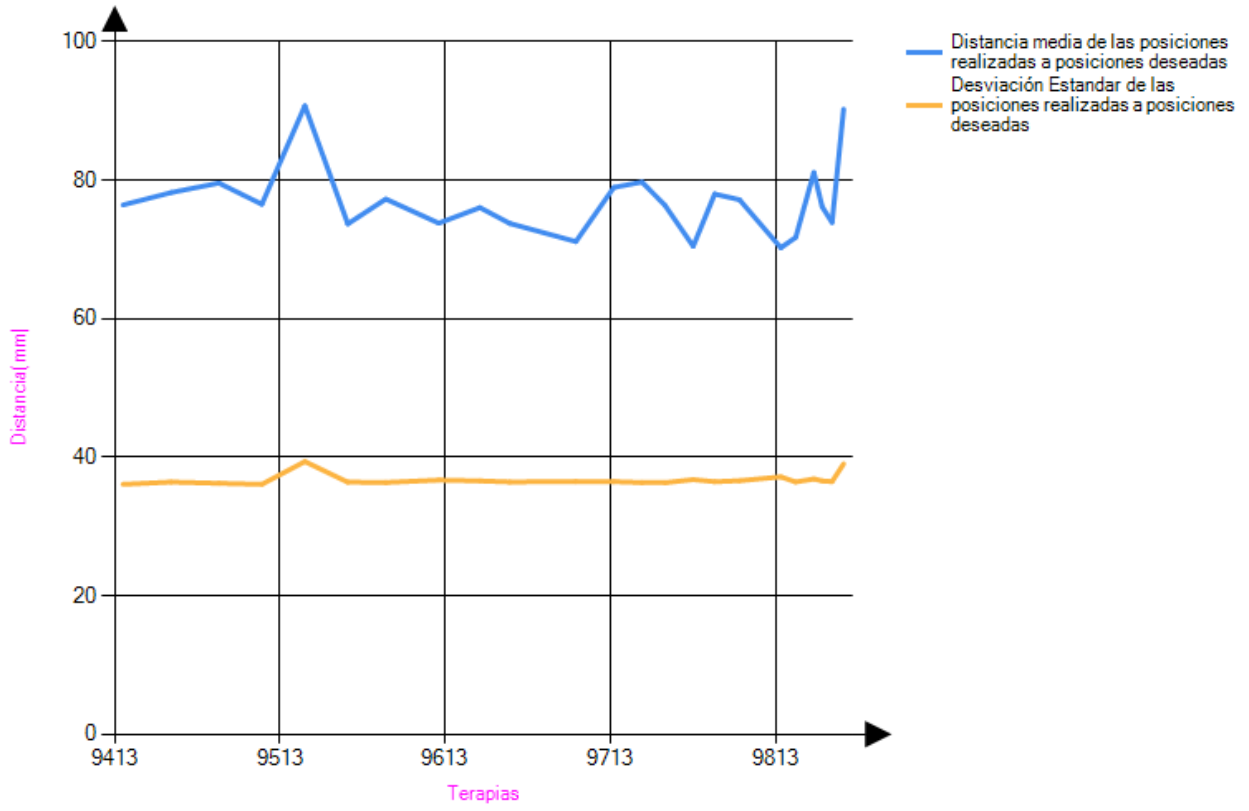
Media: 76,98038

Valor	Desviación	Número de terapia	Fecha
Primero	37,1474	9816	10/07/2017 13:27:34
Último	39,35684	9528	07/06/2017 9:17:47
Máximo	39,35684	9528	07/06/2017 9:17:47
Mínimo	36,06389	9418	29/05/2017 9:09:22

Media de desviación: 36,70627

Media y desviación

Paciente = 12 Tipo de terapia = 2 Rango = Desde 9418 hasta 9864





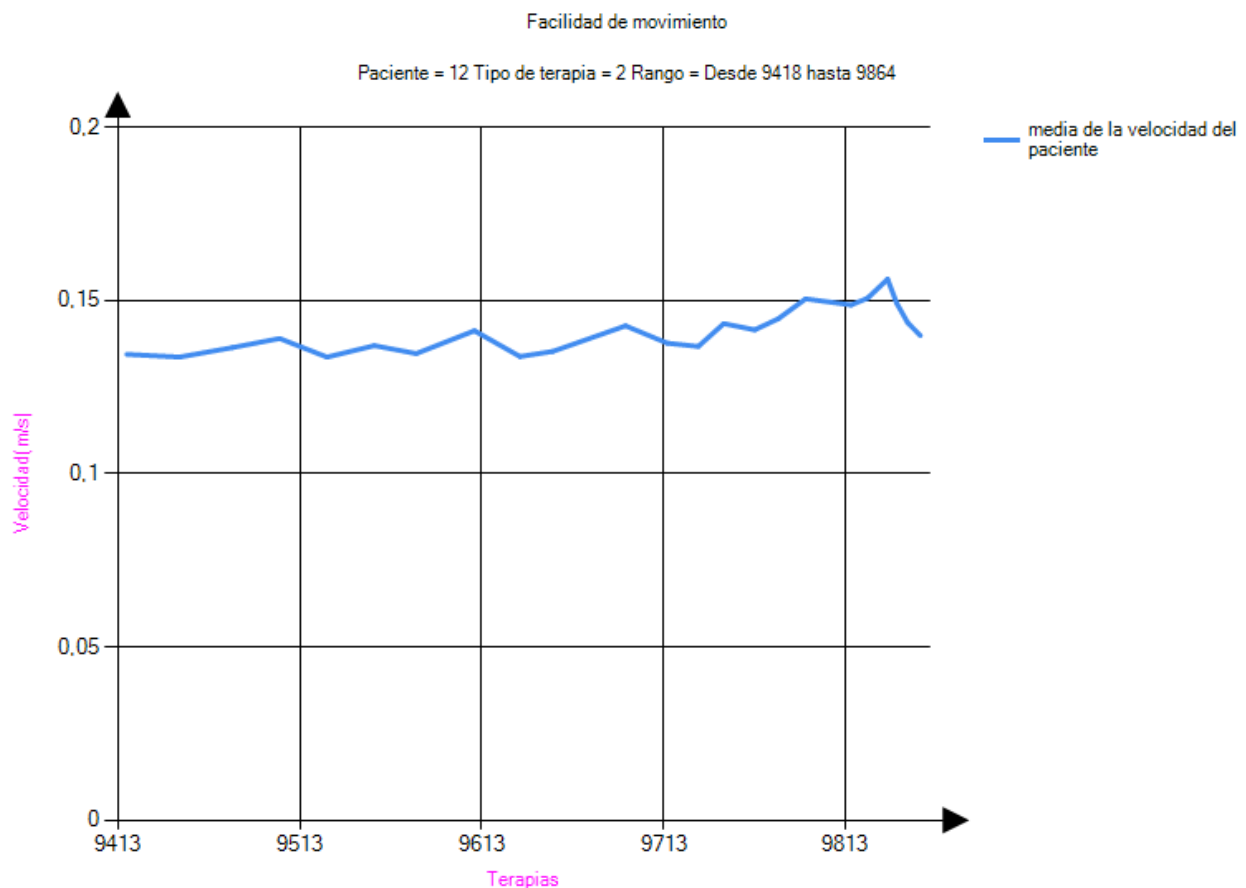
Paciente 12/ Facilidad de movimiento / Terapia tipo 2

Según varios estudios, la facilidad de movimiento se puede relacionar con a velocidad, de forma que cuanto más rápido tengas las capacidad de moverte, menos esfuerzo te costará realizar la tarea, y por lo tanto tendrás una mayor facilidad de movimiento.

El índice será la velocidad media medida en m/s.

Valor	Velocidad	Número de terapia	Fecha
Primero	0,1344818	9418	29/05/2017 9:09:22
Último	0,1398937	9854	21/07/2017 9:24:26
Máximo	0,1561858	9836	13/07/2017 9:35:54
Mínimo	0,1336903	9528	07/06/2017 9:17:47

Velocidad media: 0,1411047





Paciente 12/ Tiempo de reacción / Terapia tipo 2

Se denomina tiempo de reacción al tiempo que pasa desde que en la trayectoria deseada se produce un cambio de sentido hasta que en la trayectoria del robot se produce ese cambio de sentido también. Para ello se resta el tiempo en el que se produjo el cambio en el robot menos el tiempo en el que se produjo el cambio en el sentido deseado.

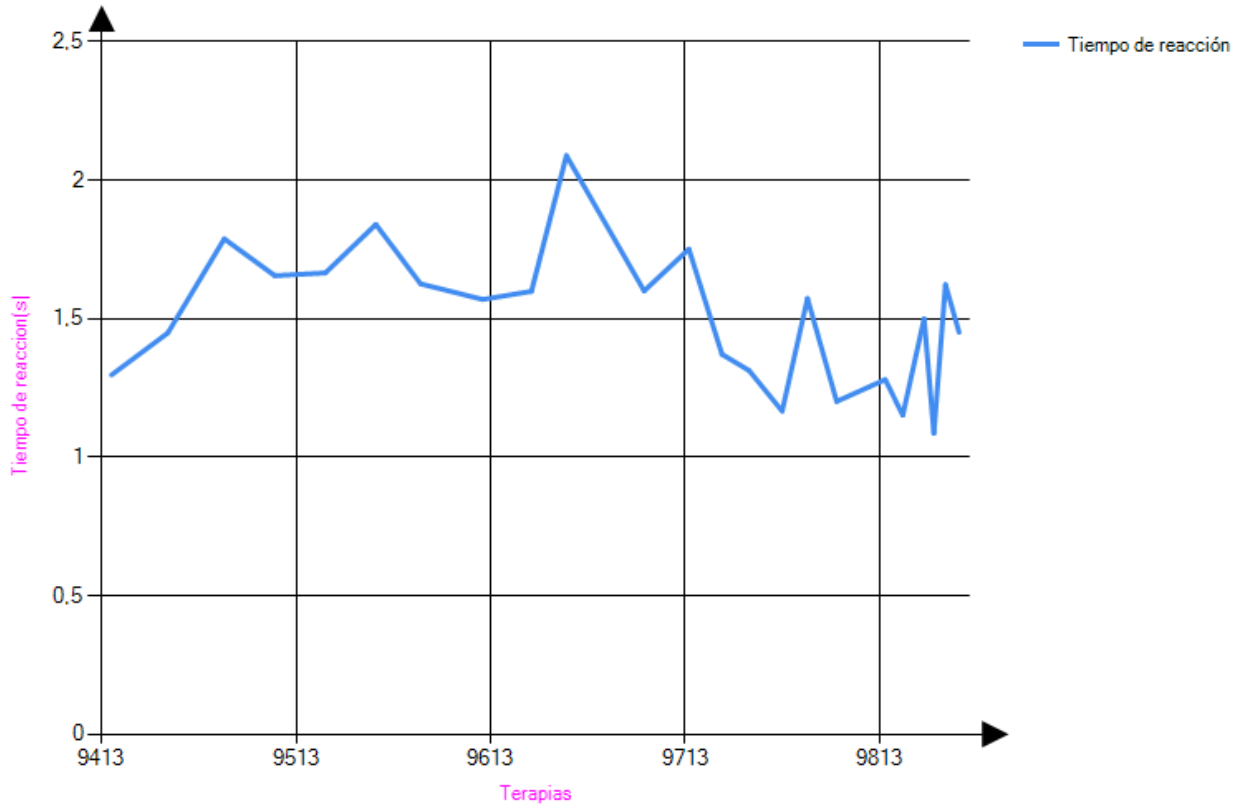
El índice que se obtiene será el tiempo de reacción y se medirá en segundos.

Valor	Tiempo de Reacción	Número de terapia	Fecha
Primero	1,29637	9418	29/05/2017 9:09:22
Último	1,449779	9854	21/07/2017 9:24:26
Máximo	2,087945	9652	21/06/2017 9:31:58
Mínimo	1,085685	9841	19/07/2017 9:27:17

[Media del tiempo de reacción: 1,505694](#)

Tiempo de reacción

Paciente = 12 Tipo de terapia = 2 Rango = Desde 9418 hasta 9864





Paciente 12/ Número de temblores / Terapia tipo 2

Se observa cuántas veces ha estado parado el robot cuando era manejado por el fisioterapeuta o cuántas veces el ratón en la pantalla ha estado en una posición fija, y cuántas veces ha estado parado el paciente, así como la comparación de la mismas.

El índice es el número de veces que el robot está parado.

Valor	Relación Robot/Paciente	Número de terapia	Fecha
Primero	127	9418	29/05/2017 9:09:22
Último	97	9854	21/07/2017 9:24:26
Máximo	185	9825	12/07/2017 10:23:35
Mínimo	-5	9527	07/06/2017 9:16:47

Media del Número de temblores: 106,0769





Paciente 12/ Simetría direccional x e y / Terapia tipo 2

En el análisis estadístico de una serie temporal y en el procesamiento de la señal, la simetría direccional es una medida estadística de rendimiento de un modelo para predecir el cambio, positivo o negativo, de una serie temporal de un período de tiempo al siguiente.

El índice de la simetría direccionales es el porcentaje de simetría direccional(%).

Valor	Simetría direccional X	Terapia	Fecha
Primero	98	9418	29/05/2017 9:09:22
Último	98	9854	21/07/2017 9:24:26
Máximo	98	9854	21/07/2017 9:24:26
Mínimo	97	9836	13/07/2017 9:35:54

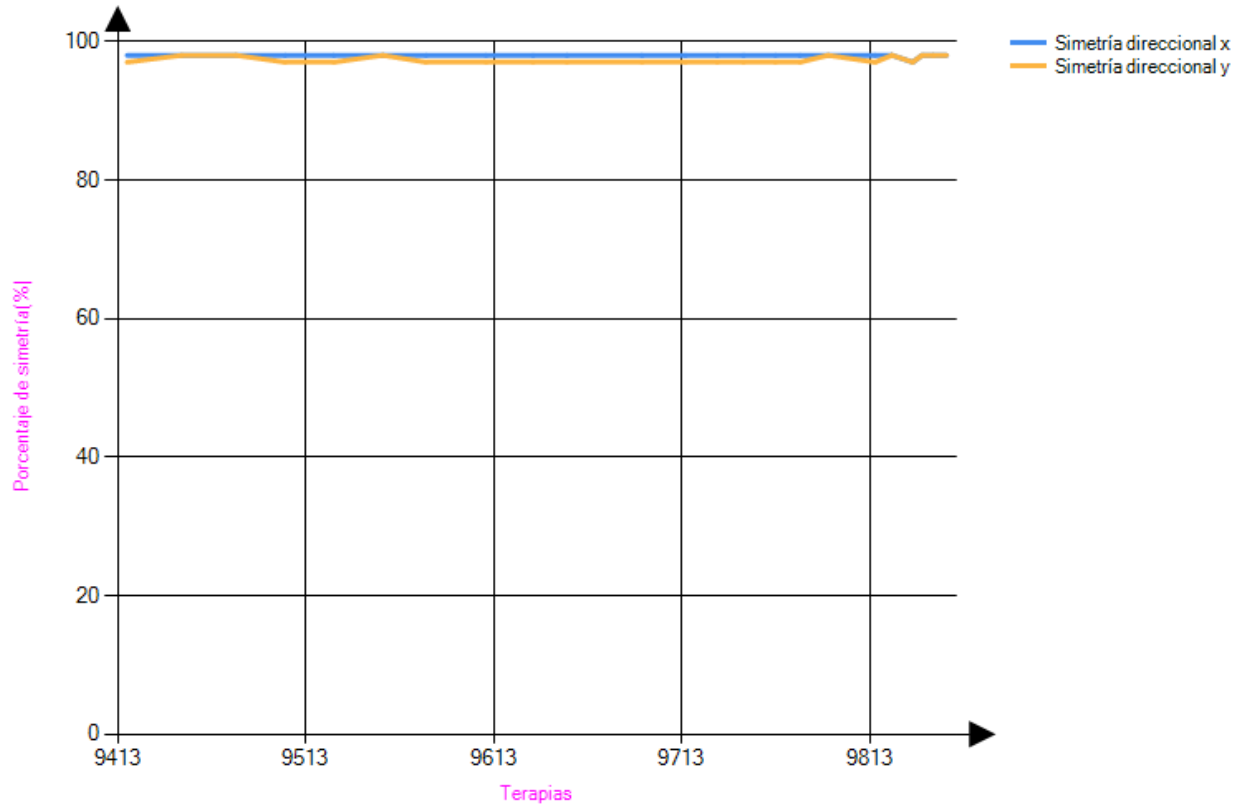
Media de la simetría direccional X: 97,95652

Valor	Simetría direccional Y	Terapia	Fecha
Primero	97	9836	13/07/2017 9:35:54
Último	98	9854	21/07/2017 9:24:26
Máximo	98	9854	21/07/2017 9:24:26
Mínimo	97	9836	13/07/2017 9:35:54

Media de la simetría direccional Y: 97,34782

Simetría direccional en x y en y

Paciente = 12 Tipo de terapia = 2 Rango = Desde 9418 hasta 9864





Paciente 12/ Simetría velocidad y ángulo / Terapia tipo 2

Simetría direccional de la velocidad en x y en y, la magnitud de la velocidad es la semisuma al cuadrado de las velocidades x e y y el ángulo es el arcoseno de la velocidad con respecto a la magnitud y lo que se observa en la pantalla es que el porcentaje del ángulo sea el mismo del paciente y de la trayectoria objetivo.

Los índices son porcentajes de simetría(%).

Valor	Magnitud de velocidad	Terapia	Fecha
Primero	69	9418	29/05/2017 9:09:22
Último	62	9854	21/07/2017 9:24:26
Máximo	69	9418	29/05/2017 9:09:22
Mínimo	59	9502	05/06/2017 9:16:03

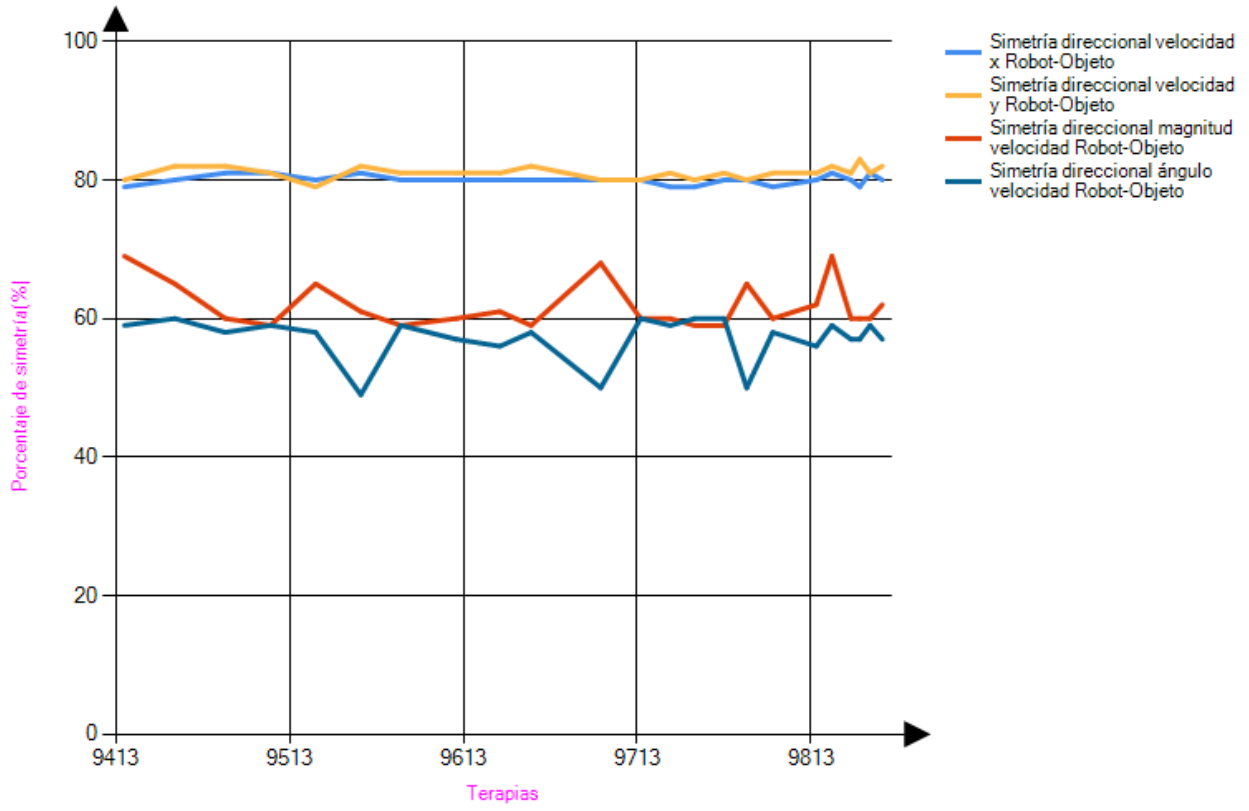
Media de la simetría de la velocidad: 61,82609

Valor	Simetría angular	Terapia	Fecha
Primero	59	9502	05/06/2017 9:16:03
Último	59	9418	29/05/2017 9:09:22
Máximo	60	9746	30/06/2017 9:33:10
Mínimo	49	9554	09/06/2017 9:18:07

Media de la simetría del ángulo: 57,17391

Simetría de velocidad y ángulo

Paciente = 12 Tipo de terapia = 2 Rango = Desde 9418 hasta 9864





Paciente 12/ Evolución de la fuerza / Terapia tipo 2

Esta medida va a marcar la evolución de la fuerza que tenga el paciente. Para ello lo que valoramos es no solo la fuerza que aplica el paciente, sino también la fuerza con la que le ayuda el robot. Para calcular la evolución vamos a tomar la fuerza que aplica el paciente y la vamos a dividir entre la suma de la fuerza que aplica el robot y la fuerza que aplica el paciente.

El índice será un porcentaje de la fuerza en x y en y que realiza el paciente con respecto a la fuerza que hace el paciente y la fuerza asistida que recibe.

Valor	Evolución en X	Terapia	Fecha
Primero	59,06144	9418	29/05/2017 9:09:22
Último	53,46383	9854	21/07/2017 9:24:26
Máximo	68,68685	9836	13/07/2017 9:35:54
Mínimo	49,5658	9609	16/06/2017 9:22:44

[Media de evolución en X: 59,78589](#)

Valor	Evolución en Y	Terapia	Fecha
Primero	61,7422	9609	16/06/2017 9:22:44
Último	62,9388	9836	13/07/2017 9:35:54
Máximo	88,57745	9447	31/05/2017 9:19:48
Mínimo	58,85397	9841	19/07/2017 9:27:17

[Media de evolución en Y: 70,81617](#)

Evolución de fuerza

Paciente = 12 Tipo de terapia = 2 Rango = Desde 9418 hasta 9864

