



Universidad de Valladolid



**ESCUELA DE INGENIERÍAS
INDUSTRIALES**

UNIVERSIDAD DE VALLADOLID

ESCUELA DE INGENIERIAS INDUSTRIALES

Grado en Ingeniería Electrónica y Automática Industrial

Prototipo de robot de rehabilitación de mano tipo “agarre terminal”

Autor:

Blanco Lahuerta, José Luis

Tutor:

**Fraile Marinero, Juan Carlos
Dpto. Ingeniería de Sistemas y
Automática**

Valladolid, Julio 2017.

Agradecimientos

Ahora que he llegado hasta aquí me gustaría agradecer a varias personas.

Primero a mi familia. A mi padre, cuya maestría con el látigo me ha hecho llegar tan lejos. A mi madre, la que ahora que voy a ser ingeniero en automática comprenda que no tiene razón respecto a la frecuencia. Y a mi hermana, que espero que me quite la banda de mejor estudiante de la casa pronto.

También a mis amigos, especialmente a Joel y Jorge, que me han hecho desconectar cuando más lo necesitaba, aunque a veces demasiado.

A mis compañeros de facultad, que sin ellos la carrera se me hubiese hecho mucho más larga y con los que he tenido momentos como 'a la cafeterie' y el 'teorema de hasselhoff'.

Y por último a todas las personas que me han ayudado a llegar aquí.

A todos ellos:

Gracias.

En este proyecto se realiza el diseño, la fabricación del primer prototipo y un primer control básico de un robot de rehabilitación de mano, tipo agarre terminal. Su posible uso se enmarca en el ámbito de la ayuda en la rehabilitación de pacientes con deficiencias neuromotoras tras haber sufrido un ataque cerebrovascular (ictus), principalmente para tratar problemas de espasticidad. Este proyecto tiene una parte de diseño mecánico y una parte de diseño de software.

Índice

Capítulo 1.- Introducción y objetivos.....	11
1.1.- Introducción al problema.....	11
1.2.- Estado del arte	12
1.3.-Cinemática de la mano.	16
1.4.- Objetivos del proyecto.....	20
Capítulo 2: Elementos mecánicos y de control utilizados.....	21
2.1.- Motores utilizados.....	21
2.2.-Elementos de control	23
2.2.1.- Opciones disponibles.....	23
2.2.2.- Control y programación desde Matlab.....	26
Capítulo 3.-Diseño mecánico del prototipo	31
3.1.- Consideraciones generales	31
3.2.- Primer prototipo.....	32
3.3.-Segundo prototipo.....	33
3.4.-Tercer prototipo	38
Capítulo 4.- Programación y control del prototipo.	55
4.1.- Consideraciones generales	55
4.2 Software desarrollado.....	55
4.2.1 Estructura del software	55
4.2.2.- Interfaz gráfica	61
4.3 Primeras pruebas.....	72
5.- Conclusiones.....	75
6.- Bibliografía	77
Anexo I: Planos de las piezas realizadas en matlab.....	79
Anexo II: Actuador AX-12:	89

Ilustración 1: Mano con espasticidad.....	11
Ilustración 2: Hand of Hope	12
Ilustración 3: Interfaz gráfica de Hand of Hope	13
Ilustración 4: Amadeo.....	13
Ilustración 5: Detalle de la parte de los dedos del robot Amadeo	14
Ilustración 6: Robot HEXORR	14
Ilustración 7: Proyecto SCRIPT	15
Ilustración 8: Esquema óseo de la mano.....	16
Ilustración 9: Ángulos de rotación fisiológicos.....	18
Ilustración 10: Actuador Dynamixel AX-12	21
Ilustración 11: Medidas del actuador AX-12.....	23
Ilustración 12: Controlador CM-5.....	24
Ilustración 13: Esquema de montaje de las partes.....	24
Ilustración 14: USB2Dynamixel	25
Ilustración 15: SMPS2Dynamixel.....	26
Ilustración 16: Pinza completa	31
Ilustración 17: Pinza simple	31
Ilustración 18: Primer prototipo	32
Ilustración 19: Ejemplo de la utilización de cables	33
Ilustración 20: Primera opción de posicionamiento.....	34
Ilustración 21: Segunda opción de posicionamiento	34
Ilustración 22: Tercera opción de posicionamiento	35
Ilustración 23: Segundo prototipo sin colocar	36
Ilustración 24: Segundo prototipo visto desde arriba	36
Ilustración 25: Segundo prototipo de perfil	36
Ilustración 26: Pieza que permite la sujeción.....	37
Ilustración 27: Tercer prototipo de perfil.....	38
Ilustración 28: Tercer prototipo visto desde arriba	38
Ilustración 29: Tercer prototipo visto desde el frente	39
Ilustración 30: Pieza_1_barra_brazo.....	40
Ilustración 31: Pieza_2_unión	40
Ilustración 32: Pieza_3_palma	41
Ilustración 33: Pieza_4_soporte	42
Ilustración 34: Pieza_5_bobina	42
Ilustración 35: Pieza_6_indice_parte1.....	43
Ilustración 36: Pieza_7_indice_parte2.....	43
Ilustración 37: Pieza_8_dedos_parte1	44
Ilustración 38: Pieza_9_dedos_parte2	44
Ilustración 39: Pieza 1 con soportes	46
Ilustración 40: Pieza 2 con soportes	46
Ilustración 41: Pieza 4 con soportes	46
Ilustración 42: Resultado impresión Pieza 1	47
Ilustración 43: Resultado impresión Pieza 2	47
Ilustración 44: Resultado impresión Pieza 3	47

Ilustración 45: Resultado impresión Pieza 4	48
Ilustración 46: Resultado impresión Pieza 4	48
Ilustración 47: Resultado impresión Pieza 4 montad	48
Ilustración 48: Resultado impresión Pieza 5	49
Ilustración 49: Resultado impresión Pieza 6	49
Ilustración 50: Resultado impresión Pieza 7	49
Ilustración 51: Resultado impresión Pieza 9	50
Ilustración 52: Pieza 1 modificada	50
Ilustración 53: Pieza 2 modificada	51
Ilustración 54: Pieza 4 modificada modelo 1	51
Ilustración 55: Pieza 4 modificada modelo 2	52
Ilustración 56: Prototipo 3 montado visto de perfil	52
Ilustración 57: Prototipo 3 montado visto desde abajo	53
Ilustración 58: Prototipo 3 montado visto desde arriba.....	53
Ilustración 59: Montaje completo 1	54
Ilustración 60: Montaje completo 2	54
Ilustración 61: Bloques del programa	56
Ilustración 62: Subprogramas para la identificación de usuario.....	56
Ilustración 63: Subdivisión de ejercicios de rehabilitación.....	57
Ilustración 64: Diagrama de flujo de la primera rutina de control	59
<i>Ilustración 65: Almacenamiento de datos.</i>	61
Ilustración 66: Estructura de la interfaz gráfica	62
Ilustración 67: Ventana nuevo usuario	63
Ilustración 68: Ejemplo de creación de directorio	63
Ilustración 69: Ejemplo de creación de archivo de datos	64
Ilustración 70: Contenido del archivo de texto	64
Ilustración 71: Ventana de antiguo usuario.....	64
Ilustración 72: Ventana de elección de actividad.....	65
Ilustración 73: Ventana de pinza completa	66
Ilustración 74: Gráfica de datos pinza completa.....	67
Ilustración 75: Paso 1 de la búsqueda.....	68
Ilustración 76: Paso 2 de la búsqueda.....	68
Ilustración 77: Paso 3 de la búsqueda.....	68
Ilustración 78: Ventana pinza simple	69
Ilustración 79: Gráfica de datos pinza simple	69
Ilustración 80: Ventana calibración pinza simple.....	70
Ilustración 81: Ventana calibración pinza completa	71
Ilustración 82: Ventana guardar gráfica.....	72
Ilustración 83: Datos prueba sin fuerza	73
Ilustración 84: Resultados prueba 2	74

Tabla 1: Longitud media y dimensiones de la mano del personal de vuelo de USAF (cm).....	17
Tabla 2: Varias dimensiones de la mano del personal de vuelo de la USAF (cm).....	17
Tabla 3: Medidas medias de las falanges de los de dos de la mano del personal de vuelo de la USAF (cm).....	18
Tabla 4: Datos de la población española diciembre de 1996 (cm)	18
Tabla 5: Ángulos de giro de las articulaciones de la mano.....	19
Tabla 6: Características del actuador AX-12.....	22
Tabla 7: Direcciones en el área EEPROM del actuador AX-12.....	28
Tabla 8: Direcciones en el área RAM del actuador AX-12.....	29

Capítulo 1.- Introducción y objetivos.

El problema planteado es la fabricación y programación de una herramienta para asistir a la rehabilitación de personas con problemas psicomotrices tras haber sufrido un ataque cerebrovascular (ictus), principalmente espasticidad.

Esta herramienta debe estar situada en la mano y debe asistir al fisioterapeuta.

1.1.- Introducción al problema.

Para abordar el problema primero hay que tener una base teórica de conocimientos para que la solución sea la apropiada.

Por ello primero se va a definir teóricamente que es un ataque cerebrovascular y sus consecuencias.

Un ataque cerebrovascular es la alteración del riego sanguíneo cerebral, ya sea en el propio cerebro o en partes del sistema nervioso como el cerebelo o el tronco del encéfalo. Esto produce una afectación, de manera permanente o transitoria, de una región del cerebro. Esta alteración puede ser isquémicos (falta de sangre) o hemorrágicos (derrame). Esto produce daños o secuelas neurológicas.

Entre las posibles secuelas de un ictus se encuentra el objetivo de este proyecto, la espasticidad, esta secuela es un trastorno del sistema nervioso que produce que algunos músculos están siempre contraídos. Esto produce una tensión y rigidez en los músculos. Lo que dificulta la movilidad del individuo.

Otra consecuencia de esta rigidez muscular es la posición en la que se encuentran la muñeca y los dedos de la mano de los pacientes, como se puede ver en la ilustración 1.



Ilustración 1: Mano con espasticidad

Entre las características de este tipo de dolencia se encuentra que los pacientes son incapaces de realizar el movimiento de apertura de la mano, mientras que si se le abre la mano con ayuda externa, inconscientemente el paciente volverá a cerrar la mano para volver al estado inicial. La fuerza ejercida es continua y es diferente en cada dedo.

1.2.- Estado del arte

Actualmente existen varios prototipos realizados tanto por empresas privadas como por equipos de investigación, a continuación se verán algunos ejemplos que se pueden encontrar.

La empresa Rehab-Robotics ha creado en colaboración con la Universidad Politécnica de Hong Kong y expertos en rehabilitación el servicio de terapia para rehabilitación neuromuscular 'Hand of Hope'. El robot es el que se puede observar en la ilustración 2.



Ilustración 2: Hand of Hope

Este sistema ayuda en la apertura y cierre de la mano en pacientes con problemas de movilidad en la mano contando con la ayuda de pequeños videojuegos interactivos como se puede ver en la ilustración 3.



Ilustración 3: Interfaz gráfica de Hand of Hope

El movimiento se ejerce con unos actuadores lineales y elementos mecánicos para transmitir ese movimiento a la apertura y cierre de la mano. Un inconveniente de este sistema es que está anclado a un soporte que se posiciona en la mesa, esto puede impedir algunos movimientos que se puedan querer hacer con el brazo.

Otro ejemplo es el dispositivo diseñado por la empresa austriaca Tyromotion, Amadeo. Como en el ejemplo anterior, este aparato tiene una componente visual mediante videojuegos.

Como se puede observar en la ilustración 4 el dispositivo se compone de una plataforma para el apoyo del brazo unida a una mesa en la que se posiciona una pantalla.



Ilustración 4: Amadeo



Ilustración 5: Detalle de la parte de los dedos del robot Amadeo

Este dispositivo se especializa en la rehabilitación de los dedos como se puede observar en la ilustración 5 dejando un poco al lado los movimientos de apertura y cierre de la mano.

Este diseño tiene la ventaja de poderse adaptar al tamaño de la mano del paciente pudiendo llegar a una gran variedad de pacientes con el mismo dispositivo.

El siguiente ejemplo es HEXORR, ilustración 6, un robot diseñado para la rehabilitación de la mano para su recuperación de la movilidad y de la fuerza:

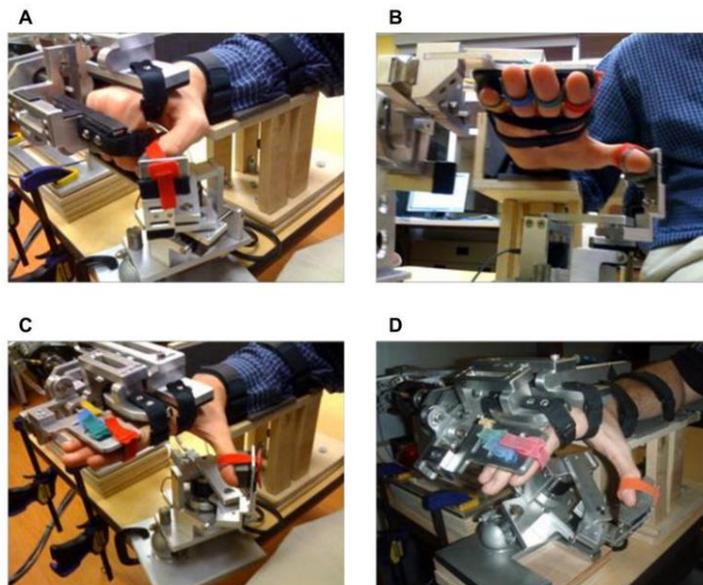


Ilustración 6: Robot HEXORR

Este prototipo mantiene los dedos juntos para realizar movimientos de pinza completa por lo que en último término se necesitarían otros ejercicios de rehabilitación para una recuperación más precisa del movimiento de los dedos por separado.

Como los robots vistos anteriormente reduce la movilidad del paciente al estar fijado a la mesa.

El volumen de dicho prototipo es elevado por lo que también puede resultar un problema.

El último ejemplo es el proyecto SCRIPT realizado por investigadores de University at Hertfordshire, dicho dispositivo aparece en la ilustración 7.



Ilustración 7: Proyecto SCRIPT

Este prototipo está diseñado para que el paciente pueda utilizarlo en casa y realice los ejercicios de rehabilitación con la ayuda de unos videojuegos. En este prototipo toda la fuerza es ejercida por el paciente, por lo que el prototipo ayuda a la recuperación de la fuerza en los dedos.

Este prototipo a diferencia de los anteriores tiene mayor movilidad en el espacio por lo que puede permitir al paciente realizar más movimientos.

Todos los robots que se han visto no ofrecen mucha libertad de movimientos pero una característica que tienen en común la mayoría de ellos es la utilización de una interfaz gráfica en forma de videojuegos.

1.3.-Cinemática de la mano.

Para poder realizar un buen prototipo primero se va a realizar un estudio cinemático de la mano.

El modelo óseo de la mano es el que se muestra en la ilustración 8.

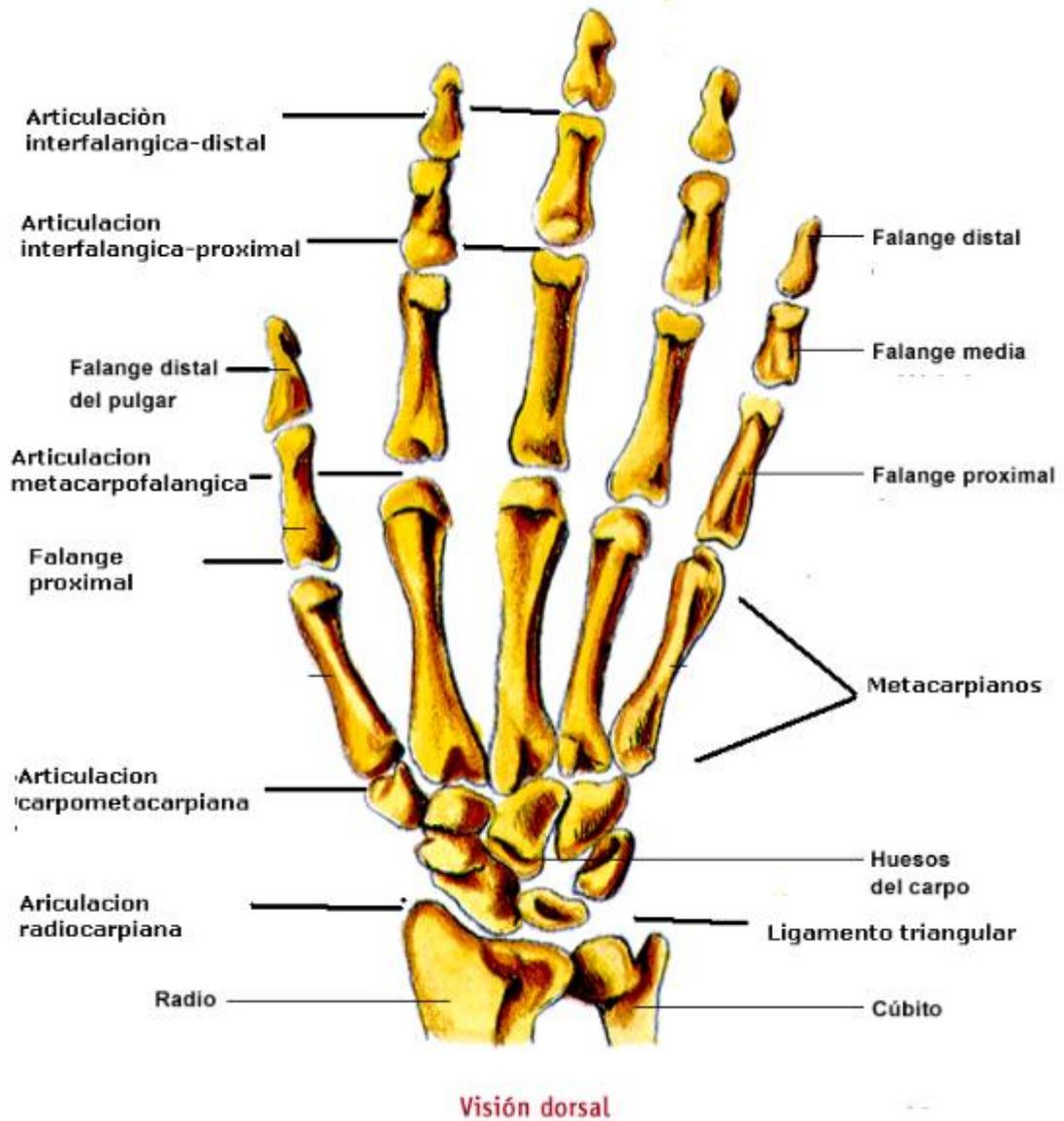


Ilustración 8: Esquema óseo de la mano

Una vez visto el modelo óseo se van a ver unas tablas con los valores de las medidas de las diferentes secciones de la mano.

Las tablas 1, 2 y 3 expresan la media de diferentes medidas de la mano tomadas del personal de vuelo de USAF (Fuerza Aérea de los Estados Unidos).

	Longitud (punta del dedo a la base)	Longitud (punta del dedo a la mano)
Hombres	Media	Media
Pulgar	5.87	12.7
Índice	7.53	18.52
Medio	8.57	19.52
Anular	8.0	18.72
Meñique	6.14	16.61
Mujeres		
Pulgar	5.37	11.05
Índice	6.9	16.67
Medio	7.79	17.65
Anular	7.31	16.76
Meñique	5.46	14.64

Tabla 1: Longitud media y dimensiones de la mano del personal de vuelo de USAF (cm)

Medida		Longitud media
Longitud de la mano	Hombres	19.72
	Mujeres	17.93
Anchura de la mano	Hombres	8.96
	Mujeres	7.71
Circunferencia de la mano	Hombres	21.59
	Mujeres	18.71
Espesor de la mano	Hombres	3.29
	Mujeres	2.76
Profundidad de la mano	Hombres	6.19
	Mujeres	5.17

Tabla 2: Varias dimensiones de la mano del personal de vuelo de la USAF (cm)

	I1	I2	I3	M1	M2	M3	A1	A2	A3	MN1	MN2	MN3
Mano derecha hombres	2.32	2.37	2.65	2.6	2.78	2.8	2.29	2.56	2.76	1.96	1.92	5.51
Mano izquierda hombres	2.32	2.39	2.61	2.6	2.82	2.75	2.30	2.59	2.78	1.95	1.98	2.49
Mano derecha mujeres	2.23	2.24	2.45	2.44	2.55	2.56	2.12	2.34	2.52	1.79	1.74	2.26
Mano izquierda mujeres	2.20	2.24	2.35	2.24	2.43	2.53	2.13	2.36	2.49	1.77	1.77	2.26

Tabla 3: Medidas medias de las falanges de los de dos de la mano del personal de vuelo de la USAF (cm)

En la tabla 4 se ven los datos obtenidos de un estudio de diferentes parámetros de la mano entre la población española.

	Longitud de la mano	Anchura de la palma	Longitud del dedo índice
Hombres	182.94	85.29	72.00
Mujeres	172.99	77.65	68.46

Tabla 4: Datos de la población española diciembre de 1996 (cm)

Antes de poner los datos de los ángulos de rotación hay que explicar las diferentes medidas, para ello nos ayudamos de la ilustración 9.

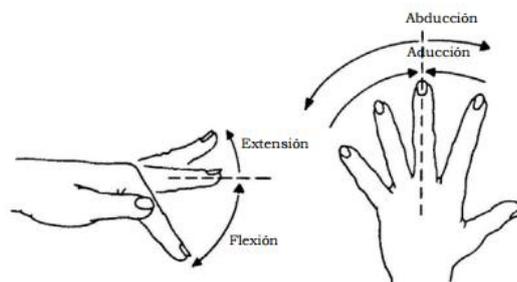


Ilustración 9: Ángulos de rotación fisiológicos

Ahora se mostrarán los ángulos de giro de las articulaciones de los dedos en la tabla 5.

Dedo	Flexión	Extensión	Abducción
Pulgar			
Carpometacarpiana	50°-90°	15°	45°-60°
Metacarpofalángica	75°-80°	0°	5°
Interfalángica	75°-80°	5°-10°	5°
Índice			
Carpometacarpiana	5°	0°	0°
Metacarpofalángica	90°	30°-40°	60°
Interfalángica proximal	110°	0°	0°
Interfalángica distal	80°-90°	5°	0°
Medio			
Carpometacarpiana	5°	0°	0°
Metacarpofalángica	90°	30°-40°	45°
Interfalángica proximal	110°	0°	0°
Interfalángica distal	80°-90°	5°	0°
Anular			
Carpometacarpiana	10°	0°	0°
Metacarpofalángica	90°	30°-40°	45°
Interfalángica proximal	120°	0°	0°
Interfalángica distal	80°-90°	5°	0°
Meñique			
Carpometacarpiana	15°	0°	0°
Metacarpofalángica	90°	30°-40°	50°
Interfalángica proximal	135°	0°	0°
Interfalángica distal	90°	5°	0°

Tabla 5: Ángulos de giro de las articulaciones de la mano

Para una buena rehabilitación el ángulo de la muñeca deberá ser de 60°, esto facilita el movimiento de los dedos y posiciona la mano en una posición en la que los movimientos de agarre se suelen realizar.

1.4.- Objetivos del proyecto

Una vez visto tanto los distintos ejemplos que se pueden encontrar en la actualidad y la cinemática de la mano nos ponemos los siguientes objetivos:

1. Diseño y construcción de un prototipo de robot de rehabilitación con las siguientes características:
 - Debe ser un prototipo que se pueda quitar y poner de forma fácil y en un espacio de tiempo reducido. También se debe tener en cuenta que durante los ejercicios se puede tener que quitar el prototipo para realizar ejercicios de estiramiento tanto en los dedos como en la muñeca.
 - Debe tener un diseño cómodo para el paciente. Este prototipo va a ejercer fuerza en contra de la reacción del paciente, si el diseño no es el correcto puede provocar daños tanto en la piel como en las articulaciones.
 - El peso no debe ser muy elevado. La mayoría de los ejercicios son con la mano sin apoyar en ninguna superficie, por lo que el prototipo no debe suponer una carga demasiado grande al paciente.
 - En la medida de lo posible debe ser adaptable para varios tamaños de mano y dedos, esta adaptabilidad debe ser lo más rápida posible y a ser posible sin intercambio de piezas. Esto dotará al prototipo de una capacidad de ser utilizado por una gran variedad de pacientes.
2. Desarrollo de un conjunto de programas para control del prototipo, realización de terapias básicas relacionadas con apertura y cierre de los dedos de la mano.
3. Los programas deben tener interfaces gráficos intuitivos y de fácil manejo.
4. El prototipo tiene que devolver algún tipo de feedback al fisioterapeuta para que se pueda utilizar en siguientes sesiones como elemento de estudio por parte del fisioterapeuta para ofrecer una evolución en la recuperación del paciente.

Capítulo 2: Elementos mecánicos y de control utilizados

2.1.- Motores utilizados

Una vez planteados los objetivos generales del proyecto vamos a empezar con el diseño del prototipo.

El primer aspecto que se va a tener en cuenta es el tipo de motorización que se va a utilizar ya que eso condicionará el tipo de piezas y la colocación de los elementos. Las características que deben tener los motores son las siguientes:

- Un bajo peso para que el peso del prototipo no suponga un impedimento al movimiento del paciente.
- Potencia suficiente para poder mover los dedos del paciente incluso cuando realice una fuerza considerable.
- Debe ser controlado fácilmente ya sea en la programación como en el uso por parte del fisioterapeuta.
- El sistema tiene que incluir algún tipo de sensorización y sea capaz de proporcionar datos al fisioterapeuta ya sea mediante el motor o mediante sensores externos.

La escuela puso a disposición para la realización de pruebas los actuadores Dynamixel AX-12, dicho actuador se puede ver en la ilustración 10.



Ilustración 10: Actuador Dynamixel AX-12

Este actuador es fabricado por la empresa ROBOTIS, esta empresa se caracteriza por la fabricación de motores y robots de uso educacional y personal.

Las características de este actuador son las que aparecen en la tabla 6.

Peso	55g	
Dimensiones	32mm x 50mm x 40mm	
Coeficiente de reducción	1/254	
Voltaje de uso	7V	10V
Par soportado (kgf*cm)	12	16.5
Velocidad de giro (s/60°)	0.269	0.196
Resolución	0.35° (1024 pasos)	
Rango de operación	300°	
Máxima corriente	900mA	
Temperatura de operación	-5°C ~ +85°C	
Tipo de comunicación	Asíncrona en serie en modo half-duplex	
Velocidad de comunicación	7343 bps ~ 1 Mbps	
Enlace físico	TTL	
Material	Plástico de ingeniería	

Tabla 6: Características del actuador AX-12

Uno de los objetivos del proyecto era poner a disposición del fisioterapeuta algún tipo de feedback que le dé información. Los actuadores AX-12 pueden dar los valores de las siguientes variables:

- La posición actual en la que se encuentra.
- La velocidad actual.
- El par que está ejerciendo el motor.
- La temperatura.

Las características técnicas que más nos interesan de estos actuadores son las siguientes:

- Peso de 55g, lo que ayudará al que el prototipo no pese demasiado.
- Gran precisión con una resolución de 0.35° (1024 pasos). Lo que nos permite una mayor precisión al ejercer el movimiento.
- Un par soportado de 12 ~ 16.5 kgf*cm esto ayuda a que se puedan mover los dedos aunque el paciente ejerza una fuerza elevada.
- Se pueden controlar mediante una conexión USB desde un portátil, esto facilita el poder controlar los actuadores desde una interfaz o programa desde un portátil.

Las medidas de estos actuadores son las que aparecen en la ilustración 11.

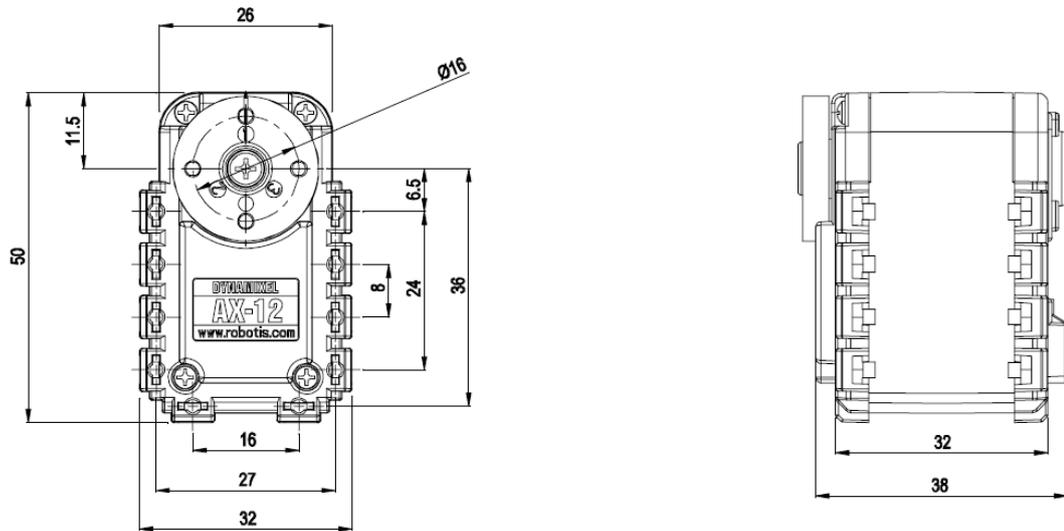


Ilustración 11: Medidas del actuador AX-12

2.2.-Elementos de control

2.2.1.- Opciones disponibles.

Los actuadores utilizados han sido los actuadores dynamixel AX-12, antes se explicaron sus características físicas, ahora se explicará en más detalle cómo se programan y se controlan dichos actuadores.

Para poder programar y controlar los actuadores se nos brindan dos opciones:

1. Mediante el CM-5, cuyo diseño se observa en la ilustración 12, que es un controlador específico de Bioloid que permite transferir los programas realizados por el usuario, con el interfaz de Bioloid, a los elementos Dynamixel. CM-5 dispone de una batería para su alimentación eléctrica. Una vez transferido un programa desde el ordenador al CM-5, este puede gobernar los movimientos de los actuadores sin necesidad de permanecer conectado en todo momento al PC. Además dispone en la zona frontal de varios botones para guiar el movimiento, para dar comienzo al mismo y para la transferencia del programa desde el PC. La placa interna de CM-5 se basa en el microcontrolador Atmega128 de Atmel (16MHz).



Ilustración 12: Controlador CM-5

2. Mediante Matlab.

Matlab es una herramienta de software matemático que ofrece un entorno de desarrollo integrado (IDE) con un lenguaje de programación propio (lenguaje M).

Los elementos físicos que se necesitan para poder programar los actuadores desde MATLAB son los que aparecen en la ilustración 13:

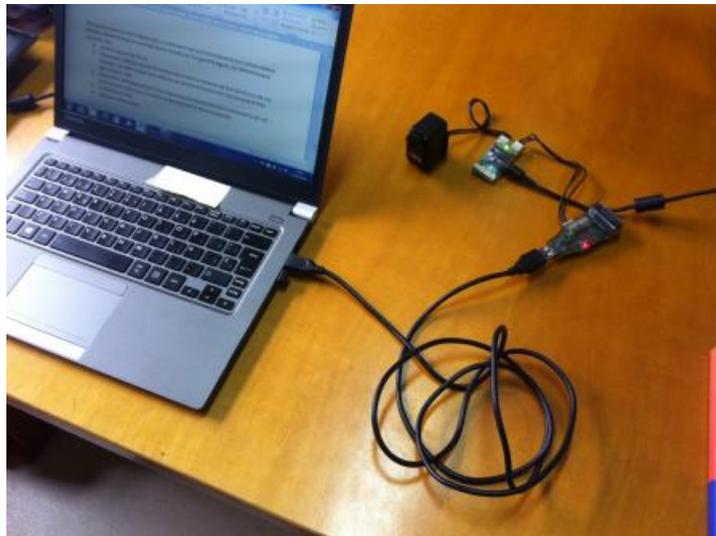


Ilustración 13: Esquema de montaje de las partes

- Un ordenador desde el que realizar la programación.
- Dispositivo USB2Dynamixel. Este dispositivo tiene un selector de función en uno de sus laterales. Se debe colocar este selector de función en la posición más cercana al lado del conector USB.
- Dispositivo SMPS2Dynamixel. Este dispositivo convierte la tensión y la corriente de red en la tensión continua y corriente necesarias para el servo dynamixel.
- 2 cables de 3 hilos.
- Cable de alimentación.
- El servo dynamixel AX-12.

A continuación se explica en más detalle los diferentes componentes.

USB2Dynamixel tiene el siguiente aspecto y estructura que se ve en la ilustración 14.

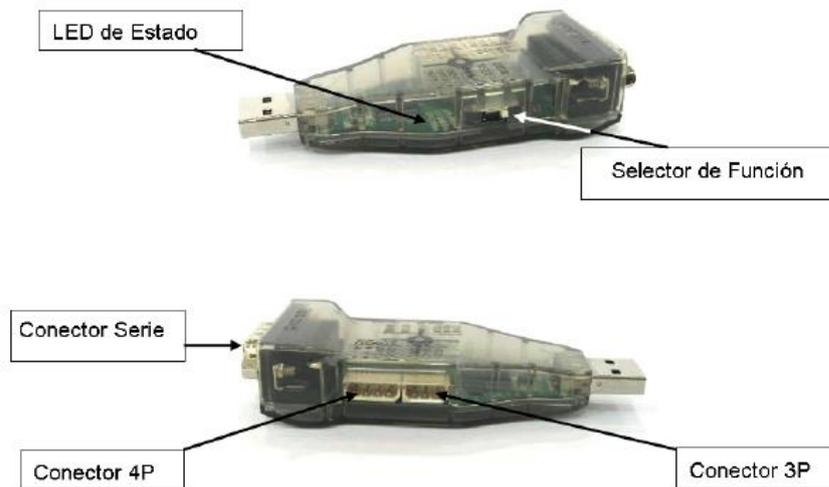


Ilustración 14: USB2Dynamixel

- LED de Estado: Muestra el estado de la alimentación PWD, TxD (escritura de datos) y RxD (lectura de datos)
- Selector de Función: para el tipo de salida de datos según el tipo de comunicación: TTL (3P), RS-232(Serie) o RS-485 (4P).
- Conector 3P: para conectar a la serie de Dynamixel AX usando niveles TTL.
- Conector 4P: para conectar a la serie de Dynamixel DX o RX usando niveles RS-485.
- Conector Serie: transforma el puerto USB en un puerto serie usando niveles RS-232.

Para conectar el USB2Dynamixel a los actuadores AX-12 primero se pone el selector en la posición TTL. Luego, se conecta el cable de 3 pines entre ambos dispositivos (cualquiera de los 2 conectores del actuador es válido). Pueden conectarse varios Dynamixel en serie, como se ha visto antes.

La única pega o dificultad es la cuestión de la alimentación. El USB2Dynamixel no es capaz de suministrar la corriente que necesitan los servos para moverse (hay que recordar que está unido al puerto USB del PC); sólo se comunica con ellos a través del cable de datos enviando los mensajes con las órdenes y las solicitudes de datos en niveles TTL.

El USB2Dynamixel es el interfaz de comunicaciones PC – Servomotores, y como tal deberá hacer una conversión entre las instrucciones escritas en los ficheros .m de Matlab y los paquetes de datos admisibles por los servos.

Para que el dispositivo USB2Dynamixel pueda funcionar correctamente como interfaz, deben instalarse los drivers correspondientes. Estos archivos pueden descargarse de la Web www.robotis.com gratuitamente. Una vez instalado el software, se dispondrá de 2 librerías ('dynamixel.dll' y 'dynamixel.h') en las cuales están descritas las funciones necesarias para manejar los servos.

SMPS2Dynamixel tiene el aspecto que se observa en la ilustración 15.



Ilustración 15: SMPS2Dynamixel

2.2.2.- Control y programación desde Matlab.

Tras ver las diferentes opciones se ha decidido elegir Matlab ya que se tiene experiencia previa programando en dicha plataforma.

Para este caso en que se deseen controlar los actuadores y sensores directamente con el PC sin hacer uso del CM-5, es necesario un mecanismo de alimentación de los mismos. Este dispositivo SMPS2Dynamixel, junto con el cable de alimentación, es el encargado de realizar dicha función. Dispone de 2 conectores de 3 pines (TTL) y dos conectores de 4 pines (RS-485), permitiendo así su unión con diferentes familias de Dynamixel.

Los pasos que se tienen que seguir para poder controlar dichos actuadores son los siguientes:

1. Instalar los drivers asociados al dispositivo.
2. Cambiar la latencia del puerto USB del ordenador de 16ms a 1ms.
3. Para saber si los cambios se han realizado correctamente se puede descargar el software “dynamixel configurator”. Es un programa sencillo con el que se puede ver si los motores se mueven como es debido.
4. Se tiene que tener el programa MATLAB instalado junto a un compilador de lenguaje C. Tras varias pruebas se consiguió que funcionase con MATLAB 2014b y Microsoft Visual Studio 2013.
5. Descargar e instalar la “robotics toolbox” desarrollada por Peter Corke.
6. Bajar el SDK para “DXL communication 1.0”, y dependiendo del ordenador, bajaremos la versión de 64 bits o de 32 bits.

Una vez finalizados estos pasos se realizaron unas pruebas de movimiento de los actuadores desde MATLAB.

Las pruebas fueron las siguientes:

- Se probó que la conexión de los actuadores con el ordenador fuera válida y que los pasos seguidos anteriormente fuesen correctos. Esta prueba consistió en el movimiento de un extremo a otro de los actuadores un número fijo de veces. Esta prueba fue satisfactoria.
- Se creó un programa que enseñaba por pantalla datos actuales de los actuadores como la posición y el par.

Para poder realizar dichos programas se utilizaron las funciones siguientes:

- `dxl_initialize()`
- `dxl_terminate()`
- `calllib('dynamixel','dxl_write_word',ID,dirección,valor)`
- `lectura=calllib('dynamixel','dxl_read_word',ID,dirección)`

La función `dxl_initialize()` inicializa la comunicación entre el controlador y los servos y devuelve un estado de “preparado”.

La función `dxl_terminate()` finaliza el proceso de comunicación.

La función `calllib('dynamixel','dxl_write_word',ID,dirección,valor)` modificar posiciones de memoria de 16 bits mientras que la función `lectura=calllib('dynamixel','dxl_read_word',ID,dirección)` realiza la lectura de una posición de memoria. Los diferentes campos de estas funciones son los siguientes:

- **ID:** identificador del motor/sensor al cual se quiere acceder. Si bien el identificador de los actuadores está comprendido entre 0 y 253, puede usarse como parámetro ID=254 (equivalente a BROADCAST_ID), para enviar la instrucción a todos los servos conectados.
- **dirección:** dirección de memoria en la que se desea escribir o de la que se desea leer.
- **valor:** en las escrituras en memoria este parámetro determina la magnitud a almacenar en la correspondiente posición de memoria indicada.

Las diferentes direcciones de memoria se pueden observar en la tabla 7 para las direcciones en el área EEPROM y en la tabla 8 para direcciones en el área RAM.

Address	Item	Access	Initial Value
0(0X00)	Model Number(L)	RD	12(0x0C)
1(0X01)	Model Number(H)	RD	0(0x00)
2(0X02)	Version of Firmware	RD	?
3(0X03)	ID	RD,WR	1(0x01)
4(0X04)	Baud Rate	RD,WR	1(0x01)
5(0X05)	Return Delay Time	RD,WR	250(0xFA)
6(0X06)	CW Angle Limit(L)	RD,WR	0(0x00)
7(0X07)	CW Angle Limit(H)	RD,WR	0(0x00)
8(0X08)	CCW Angle Limit(L)	RD,WR	255(0xFF)
9(0X09)	CCW Angle Limit(H)	RD,WR	3(0x03)
10(0x0A)	(Reserved)	-	0(0x00)
11(0X0B)	the Highest Limit Temperature	RD,WR	85(0x55)
12(0X0C)	the Lowest Limit Voltage	RD,WR	60(0X3C)
13(0X0D)	the Highest Limit Voltage	RD,WR	190(0xBE)
14(0X0E)	Max Torque(L)	RD,WR	255(0XFF)
15(0X0F)	Max Torque(H)	RD,WR	3(0x03)
16(0X10)	Status Return Level	RD,WR	2(0x02)
17(0X11)	Alarm LED	RD,WR	4(0x04)
18(0X12)	Alarm Shutdown	RD,WR	4(0x04)
19(0X13)	(Reserved)	RD,WR	0(0x00)
20(0X14)	Down Calibration(L)	RD	?
21(0X15)	Down Calibration(H)	RD	?
22(0X16)	Up Calibration(L)	RD	?
23(0X17)	Up Calibration(H)	RD	?

Tabla 7: Direcciones en el área EEPROM del actuador AX-12

RAM Area	24(0X18)	Torque Enable	RD,WR	0(0x00)
	25(0X19)	LED	RD,WR	0(0x00)
	26(0X1A)	CW Compliance Margin	RD,WR	0(0x00)
	27(0X1B)	CCW Compliance Margin	RD,WR	0(0x00)
	28(0X1C)	CW Compliance Slope	RD,WR	32(0x20)
	29(0X1D)	CCW Compliance Slope	RD,WR	32(0x20)
	30(0X1E)	Goal Position(L)	RD,WR	[Addr36]value
	31(0X1F)	Goal Position(H)	RD,WR	[Addr37]value
	32(0X20)	Moving Speed(L)	RD,WR	0
	33(0X21)	Moving Speed(H)	RD,WR	0
	34(0X22)	Torque Limit(L)	RD,WR	[Addr14] value
	35(0X23)	Torque Limit(H)	RD,WR	[Addr15] value
	36(0X24)	Present Position(L)	RD	?
	37(0X25)	Present Position(H)	RD	?
	38(0X26)	Present Speed(L)	RD	?
	39(0X27)	Present Speed(H)	RD	?
	40(0X28)	Present Load(L)	RD	?
	41(0X29)	Present Load(H)	RD	?
	42(0X2A)	Present Voltage	RD	?
	43(0X2B)	Present Temperature	RD	?
	44(0X2C)	Registered Instruction	RD,WR	0(0x00)
	45(0X2D)	(Reserved)	-	0(0x00)
	46(0x2E)	Moving	RD	0(0x00)
	47(0x2F)	Lock	RD,WR	0(0x00)
48(0x30)	Punch(L)	RD,WR	32(0x20)	
49(0x31)	Punch(H)	RD,WR	0(0x00)	

Tabla 8: Direcciones en el área RAM del actuador AX-12

Se van a utilizar las direcciones de memoria del área RAM, sobre todo las direcciones referentes a la posición y al par.

Capítulo 3.-Diseño mecánico del prototipo

3.1.- Consideraciones generales

Para la fabricación del prototipo se van a tener las siguientes consideraciones:

- Tiene que poder realizar los siguientes ejercicios de rehabilitación:
 - Movimiento de pinza completa. Este movimiento se realiza con los todos los dedos cerrándose y abriéndose al mismo tiempo como se ve en la ilustración 16.

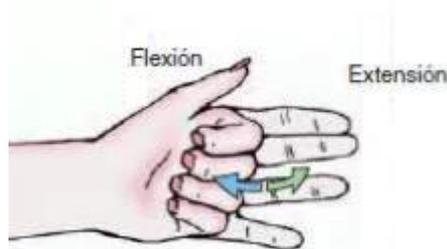


Ilustración 16: Pinza completa

- Movimiento de pinza simple. Este ejercicio se realiza solo con el dedo índice y el pulgar como se puede ver en la ilustración 17. En nuestro caso el resto de dedos estarán flexionados.



Ilustración 17: Pinza simple

- Tiene que ofrecer libertad de movimientos para permitir que también se realicen movimientos con el brazo, esto conlleva a que el prototipo tiene que tener un peso bajo para que el paciente no tenga que realizar un esfuerzo para levantarlo.
- Tiene que ser cómodo para el paciente y que se pueda quitar y poner en un tiempo reducido.

3.2.- Primer prototipo

Para un primer prototipo se utilizaron piezas diseñadas para ser acopladas a los actuadores AX-12 por parte de BIOLOID. En la ilustración 18 se puede ver dicho prototipo.

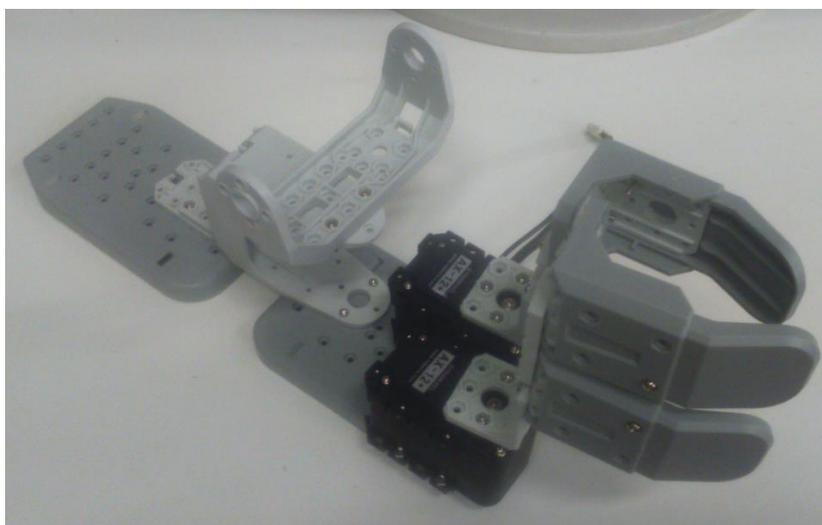


Ilustración 18: Primer prototipo

En este prototipo se puede observar un soporte para el brazo, dos piezas donde apoyar los dedos y otro soporte para apoyar el pulgar.

En este prototipo se han utilizado dos motores, uno para el movimiento del pulgar y otro para el movimiento del resto de los dedos, esto nos permite realizar un movimiento de cierre y apertura de la mano mediante una pinza completa.

Este prototipo se descartó ya que el momento generado por los dedos en la parte superior del soporte hacía que los soportes se doblaran y los motores eran incapaces de realizar los movimientos de cierre y apertura.

De todas formas este prototipo sirvió para comprobar que el lenguaje de programación funcionaba, solo que el prototipo físico no podía cumplir con las especificaciones.

3.3.-Segundo prototipo

Una vez vistos los problemas del primer prototipo se llegó a la conclusión de que la fuerza que se debía ejercer para que los dedos se movieran tenía que ser lineal con la dirección del dedo y que el punto de aplicación de dicha fuerza tenía que ser en la punta del dedo para que así no existiesen momentos de fuerza indeseables que pusiesen una mayor presión al aparato físico.

Con estas condiciones se ha llegado a la conclusión de la utilización de cables como elemento transmisor de la fuerza, la posición de los cables será parecida a la de la ilustración 19.

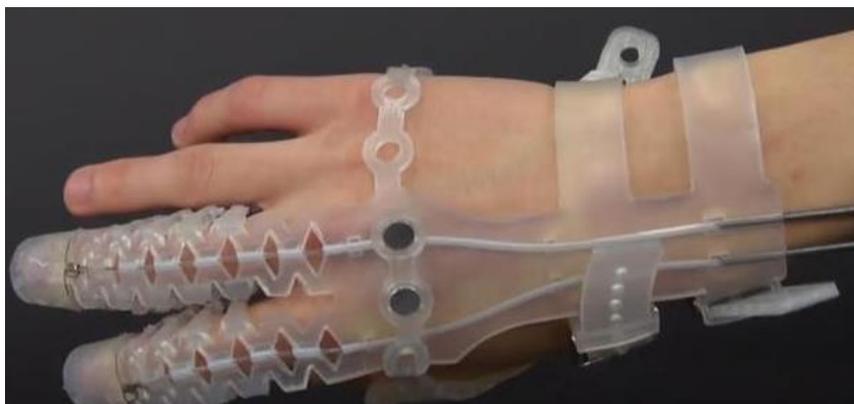


Ilustración 19: Ejemplo de la utilización de cables

La utilización de cables como elemento transmisor de la fuerza nos permite también que los movimientos de apertura de la mano, que son movimientos que esperamos que el paciente realice por sí solo, estén libres de impedimentos. Esto también facilitara la labor del fisioterapeuta al poder sin restricciones abrir la mano del paciente en cualquier momento.

También se ha elegido el cable como elemento transmisor ya que nos permite que la posición de las articulaciones de los dedos no sea de importancia, ya que la fuerza la ejercemos directamente en la punta de los dedos, esto nos permite que el prototipo pueda ser utilizado por una variedad de personas con el tamaño de la mano diferente.

Una vez elegidos los motores y método de transmisión de la fuerza se tienen que elegir el resto de elementos auxiliares de soporte y sujeción a la mano.

En los objetivos se ha propuesto que los ejercicios en los que se va a ayudar en la rehabilitación son movimientos de pinza, tanto pinza completa (utilizando los cuatro dedos) como la pinza simple (solo el dedo índice). Se ha elegido dejar el dedo gordo estático por la complejidad de control de este dedo al poseer otro grado de libertad y poder hacer movimientos circulares y por su poca movilidad en los ejercicios de pinza que realiza la mano.

Por ello se han elegido dos actuadores, uno para que controle el movimiento del dedo índice y otro para que controle el movimiento del resto de los 3 dedos mientras que el dedo gordo se mantiene estático.

Estos dos actuadores se pueden situar en diferentes posiciones:

- Encima de la palma de la mano (ilustración 20), esto nos permite los siguientes puntos:
 - Los actuadores están muy cerca de los dedos y por ello hay menos puntos de rozamiento en la transmisión de la fuerza.
 - Añade peso a la mano, lo que puede llegar a ser un inconveniente para la realización de algunos ejercicios. Esto se puede compensar con un peso bajo del resto de las piezas y el peso reducido de dichos actuadores.



Ilustración 20: Primera opción de posicionamiento

- Encima del brazo (ilustración 21):
 - Desplazan el peso de los actuadores al brazo, lo que disminuye el peso en la planta de la mano.
 - Se tiene que llevar los cables por una distancia mayor, lo que provocará más rozamiento y un mayor esfuerzo a realizar por los motores.



Ilustración 21: Segunda opción de posicionamiento

- A los laterales del brazo (ilustración 22):
 - Como en el caso anterior, disminuye el peso en la palma de la mano, pero hay que realizar giros con los cables para canalizarlos que aumentaran el rozamiento y la disminución de la fuerza ejercida por los actuadores.



Ilustración 22: Tercera opción de posicionamiento

También existe la posibilidad de anclar los actuadores a un soporte exterior a la estructura del brazo, esto se ha desestimado desde el principio ya que uno de los objetivos es que el paciente pueda realizar ejercicios en los que se necesite una movilidad en los 3 ejes del espacio, y este tipo de soportes obligan al paciente a realizar los ejercicios siempre en contacto con una mesa por el elevado peso de dichos soportes.

Tras todas las opciones planteadas se ha decidido optar por la opción de ubicar los actuadores en la palma de la mano, las razones son por la mayor proximidad a los dedos y porque se ha considerado que el prototipo no tendrá un peso elevado, lo que no será ningún inconveniente a la hora de realizar ejercicios de rehabilitación.

Para la colocación de los actuadores se tiene que fabricar un soporte para que se sujeten a la mano. También se fabricarán piezas que se anclaran a los dedos que permitan canalizar los cables hasta la punta de los dedos.

En un principio se realizó un primer prototipo en el que la base de la palma se realizó en madera y las distintas piezas de los dedos en PVC.

Los resultados son los que se observan en las ilustraciones 23, 24 y 25.



Ilustración 23: Segundo prototipo sin colocar



Ilustración 24: Segundo prototipo visto desde arriba



Ilustración 25: Segundo prototipo de perfil

Los actuadores están sujetos a la placa de madera mediante unos soportes, ilustración 26, que son piezas fabricadas por BIOLOID:



Ilustración 26: Pieza que permite la sujeción

Dicha pieza permite la unión de los actuadores a una superficie plana. En la placa de madera se realizaron los correspondientes agujeros para luego poder unir la placa con la pieza mediante tornillos y tuercas.

Tanto las piezas de los dedos como la palma se han unido a la mano mediante tiras de velcro.

Tras realizar pruebas con dicho prototipo con la utilización de unos programas sencillos de control de los actuadores se comprobó la funcionalidad de dicho prototipo, pero se observaron algunos problemas:

- La sujeción del cable al actuador para la transmisión de la fuerza no tenía ningún problema mientras no se ejerciese mucha fuerza con los dedos, pero una vez que se ejercía una fuerza suficiente, el cable se salía del enrollado. Esto provocaba que la longitud del cable fuera de la bobina cambiase y hacia que el programa no funcionase.
- El prototipo tenía poca robustez y estabilidad, esto provocaba que las piezas se moviesen más de lo deseado.

3.4.-Tercer prototipo

Para este tercer prototipo se plantean los siguientes objetivos:

- Arreglar el sistema de bobinado del cable para que no de problemas de que se salga el cable y de problemas de longitud.
- Diseñar un conjunto de piezas que dé más robustez y sujeción al conjunto.

La primera versión que se diseñó se puede observar en las ilustraciones 27, 28 y 29.

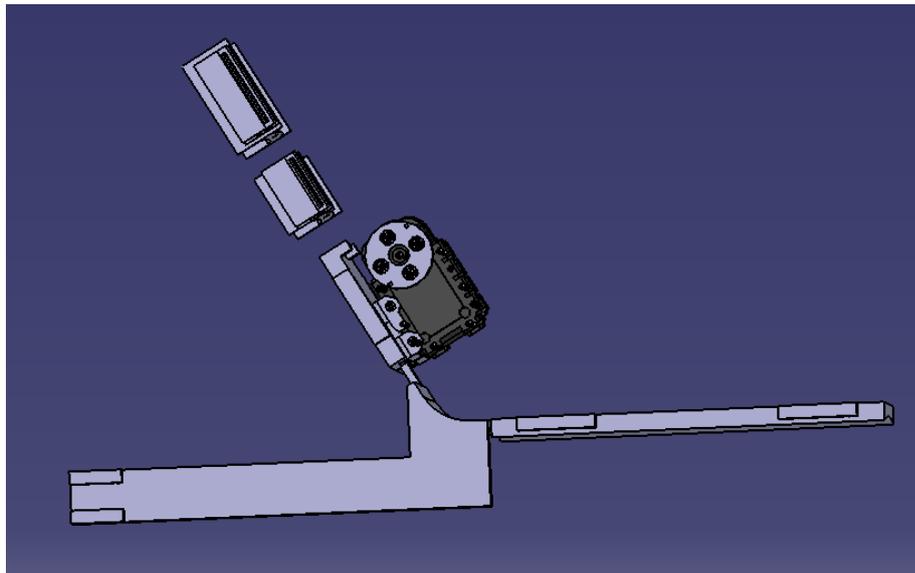


Ilustración 27: Tercer prototipo de perfil

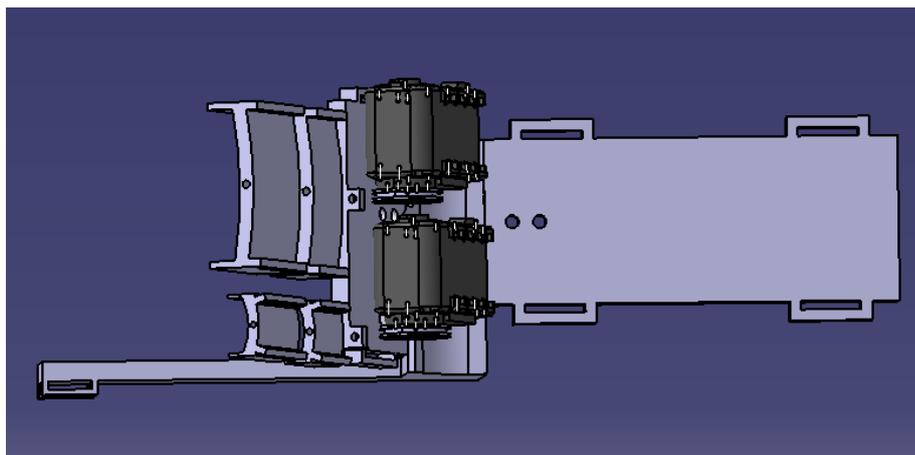


Ilustración 28: Tercer prototipo visto desde arriba

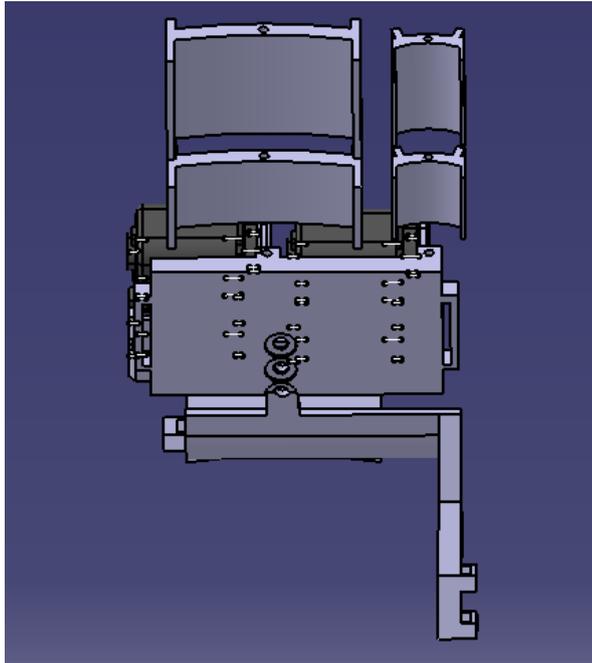


Ilustración 29: Tercer prototipo visto desde el frente

Para el diseño de este prototipo se ha decidido que el dedo pulgar no realice ningún tipo de movimiento por la complejidad de poder controlar su movimiento al tener dos grados de libertad.

Dicho prototipo se ha diseñado creando las distintas piezas mediante un programa informático de diseño, fabricación e ingeniería asistida por computadora comercial realizado por Dassault Systèmes, CATIA.

Este programa nos permite el diseño de una manera sencilla de las piezas que vamos a fabricar más adelante mediante impresión 3D.

El conjunto de piezas se puede dividir en dos grupos, las piezas que sirven para el soporte y la sujeción de las piezas al brazo y la mano y las piezas que sirven como transmisoras de la fuerza ejercida por los motores.

A continuación se explicaran en más detalle las diferentes piezas y su desarrollo.

Para el tema de la robustez y estabilidad del exoesqueleto se ha diseñado las siguientes piezas:

- Una barra que se sujetará al brazo mediante unas cintas de velcro para una mayor estabilidad del prototipo (ilustración 30).

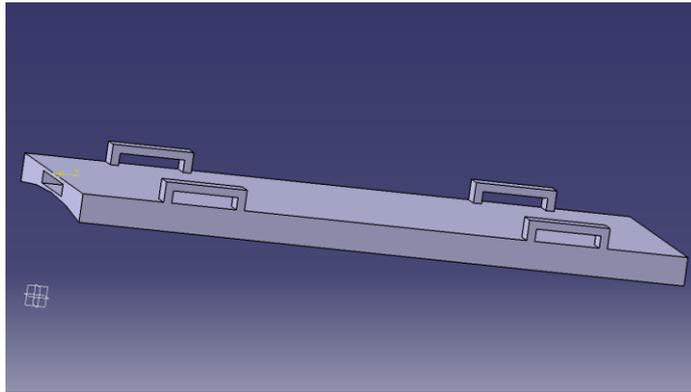


Ilustración 30: Pieza_1_barra_brazo

- A esta pieza se le ha nombrado Pieza_1_barra_brazo con extensión CATIA Part.
- Tiene también una superficie curva para una mejor sujeción al brazo.
- Una pieza que servirá de unión entre la palma y el brazo (ilustración 31).

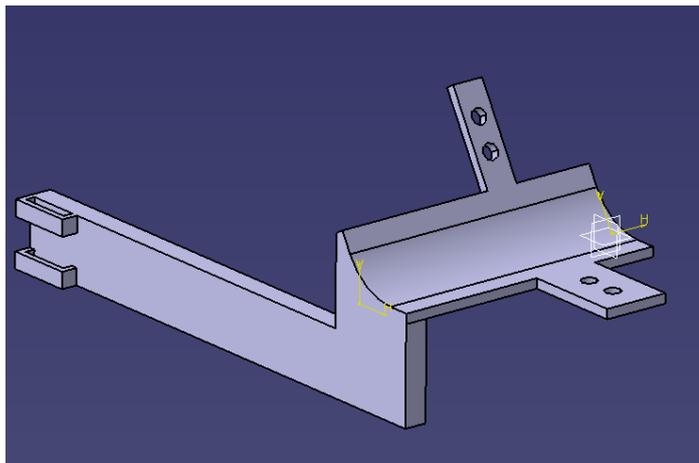


Ilustración 31: Pieza_2_unión

Esta pieza permitirá que la palma esté en una posición de 60°, lo que ayuda a un buen estiramiento de los músculos y tendones de la mano y muñeca facilitando los movimientos de apertura y cierre de la mano.

- Esta pieza tiene el nombre Pieza_2_union con la extensión CATIA Part.
 - Esta pieza también tiene una barra lateral para sujetar el pulgar.
 - Cuenta también con agujeros a diferentes alturas para regular la altura de las otras piezas a las que va sujeta.
- Una pieza que se situará en la mano y que servirá de plataforma para los actuadores (ilustración 32).

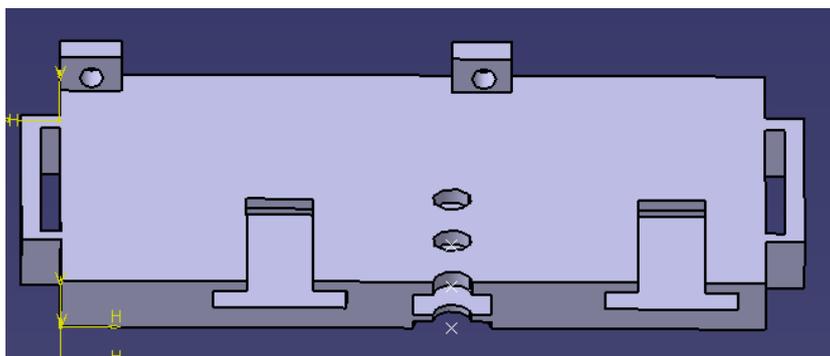


Ilustración 32: Pieza_3_palma

En esta pieza se pueden observar varias cosas:

- Las dos bandas laterales sirven para la colocación de unas cintas de velcro para la sujeción de la pieza en la mano.
 - Unas guías donde irán montados los actuadores, esto permite al fisioterapeuta poder montar y desmontar los actuadores con mayor simplicidad.
 - Los prismas de la parte superior con un agujero sirven como guías para los cables.
 - Los agujeros en la superficie de la pieza es para poder regular la sujeción de la pieza con la Pieza_2_union.
 - Esta pieza se llama Pieza_3_palma con la extensión CATIA Part.
- Una pieza que servirá como base para los actuadores y que permitirá su sujeción a la pieza de la palma (ilustración 33).

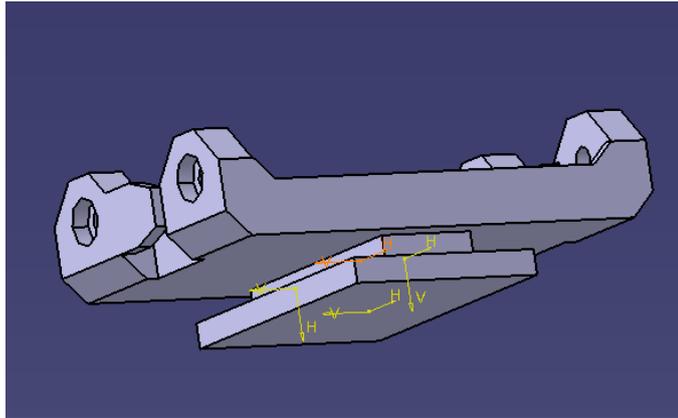


Ilustración 33: Pieza _4_soporte

- Esta pieza se une mediante 4 tornillos a los actuadores.
- Esta pieza se ha creado mediante el modelo de una pieza creada por Bioloid a la que se le ha unido un sistema de guía.
- Esta pieza se llamará Pieza_4_soporte con la extensión CATIA Part.

Las piezas que van a servir como transmisoras de fuerza son las siguientes:

- Una pieza que servirá como bobina para enrollar el cable al eje de los motores (ilustración 34).

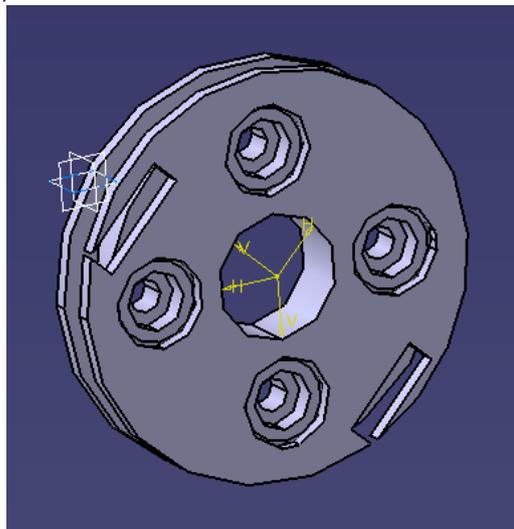


Ilustración 34: Pieza_5_bobina

- Se han diseñado dos ranuras en la superficie exterior de la bobina para que pase por ahí el extremo y tope del cable.
- A esta pieza se le ha nombrado Pieza_5_bobinacable con la extensión CATIA Part.

Las siguientes piezas serán las piezas que se sujetan a los dedos:

- Para el dedo índice (ilustración 35 y 36).

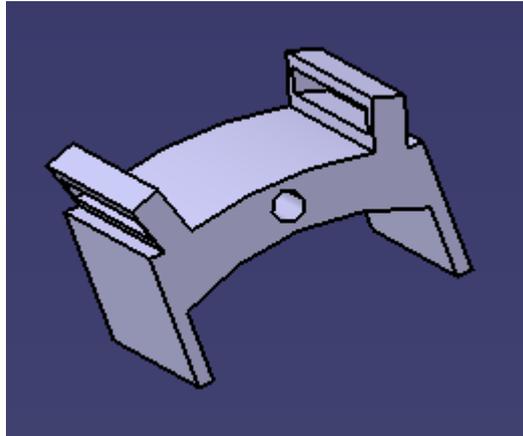


Ilustración 35: Pieza_6_indice_parte1

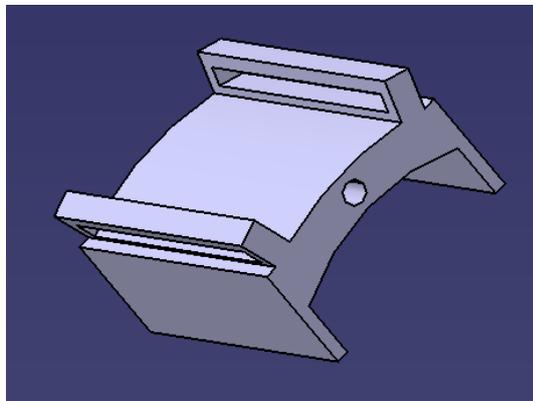


Ilustración 36: Pieza_7_indice_parte2

- Estas piezas tienen el mismo diseño pero sus longitudes son diferentes, la primera estará situada en la falange proximal y la segunda en las falanges media y distal.
 - Tiene en los laterales guías en las que se situaran las cintas de velcro para la sujeción a los dedos.
 - Se han llamado Pieza_6_indice_parte1 y Pieza_7_indice_parte2 respectivamente con la extensión CATIA Part.
-
- Para el resto de los dedos (ilustraciones 37 y 38).

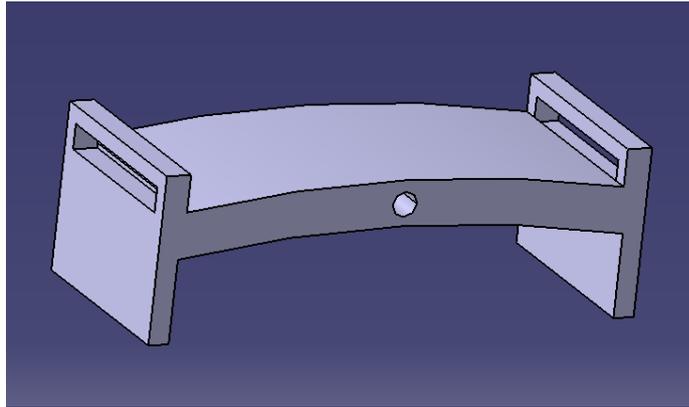


Ilustración 37: Pieza_8_dedos_parte1

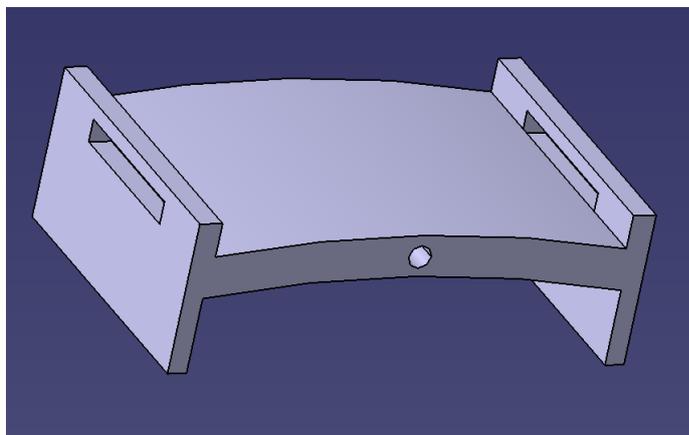


Ilustración 38: Pieza_9_dedos_parte2

- Como en las piezas anteriores ambas piezas tienen un diseño similar pero su longitud es distinta, la primera se sitúa en la falange proximal y la otra en las falanges media y distal.
- También tienen las guías para las cintas de velcro.
- Se han llamado Pieza_8_dedos_parte1 y Pieza_9_dedos_parte2 con la extensión CATIA Part.

En resumen la distribución de las piezas va a ser la siguiente:

- Una superficie situada en la parte exterior de la palma donde se posicionaran los actuadores AX-12 y que sujeta el pulgar.
- Dos piezas que se situaran en la falange proximal, una pieza en el dedo índice y otra en las falanges proximales de los otros tres dedos.
- Dos piezas que deberán estar situadas en los otros dos falanges de los dedos, se ha decidido que la última articulación de los dedos se

mantendrá inmóvil para ayudar en los ejercicios de pinza. Estas piezas correspondientes a los dedos que no son el índice deben estar unidas.

- Un soporte para que la muñeca este en una posición de 60°.
- Dos cables que unan los actuadores con los dos conjuntos de piezas. Se ha decidido utilizar cables ya que lo que nos interesa es que el paciente pase de la contracción de los músculos a poder abrir la mano. Por eso siempre es deseable que la persona pueda abrir la mano sin ningún impedimento. Los cables se tensionan para abrir la mano del paciente pero si el paciente consigue abrir la mano más de lo esperado puede hacerlo.
- Una barra atada al brazo para una mayor estabilidad del prototipo.

Una vez diseñadas las piezas se procederá a la impresión en 3D de dichas piezas.

La impresión 3D es un grupo de tecnologías de fabricación por adición donde un objeto tridimensional es creado mediante la superposición de capas sucesivas de material.

Para poder imprimir las piezas en una impresora 3D los archivos previamente realizados en Catia se tienen que guardar en formato stl.

Al empezar a imprimir las piezas se vio que determinadas piezas no se podían imprimir tal cual habían sido creadas ya que tenían partes en el aire que la impresora 3D no podía imprimir, dichas piezas son la Pieza_1_barra_brazo, la Pieza_2_union y la pieza Pieza_4_union.

Esto nos llevó a la búsqueda de una solución. Entre las soluciones a este problema se planteó dividir las piezas mediante cortes y luego unirlos con pegamento o la utilización de soportes que permitirían a la impresora imprimir dichas partes flotantes y que luego dichos soportes se removerían.

El programa utilizado es el Meshmixer, este programa nos permite generar automáticamente los soportes para dicha pieza, este programa tiene muchas características variables que nos permiten crear los soportes como queramos.

El resultado de la generación de dichos soportes aparece en las ilustraciones 39, 40 y 41.

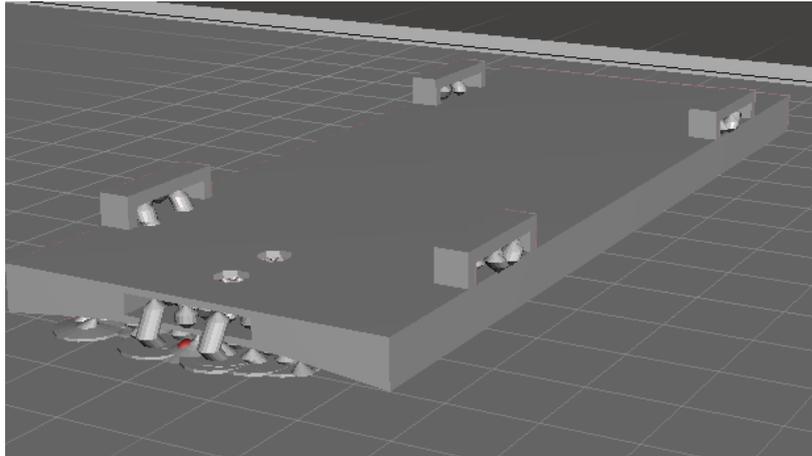


Ilustración 39: Pieza 1 con soportes

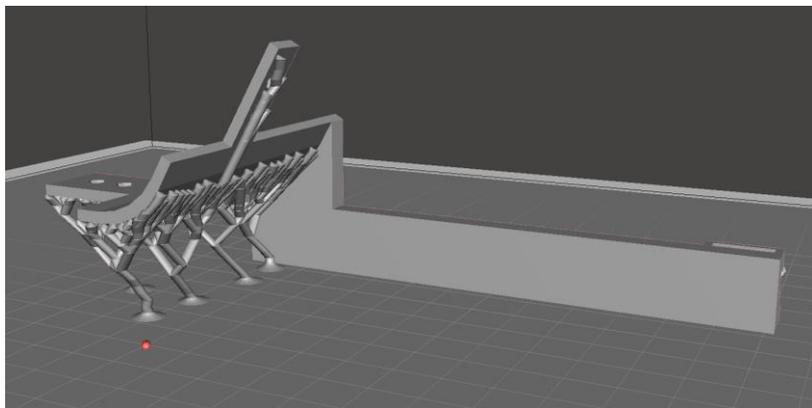


Ilustración 40: Pieza 2 con soportes

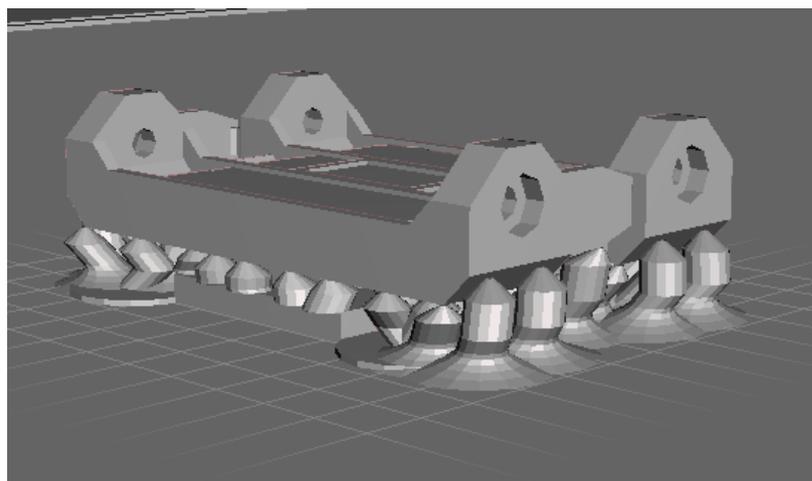


Ilustración 41: Pieza 4 con soportes

Tras la realización de dichas piezas el resultado ya impreso es el siguiente:

- Pieza 1 (ilustración 42).



Ilustración 42: Resultado impresión Pieza 1

- Pieza 2 (ilustración 43).

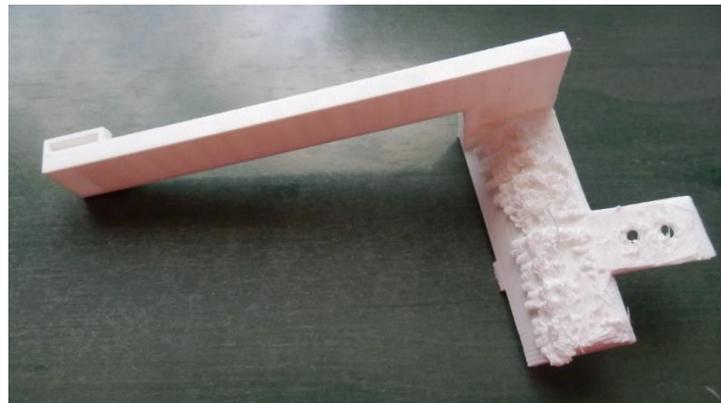


Ilustración 43: Resultado impresión Pieza 2

- Pieza 3 (ilustración 44).

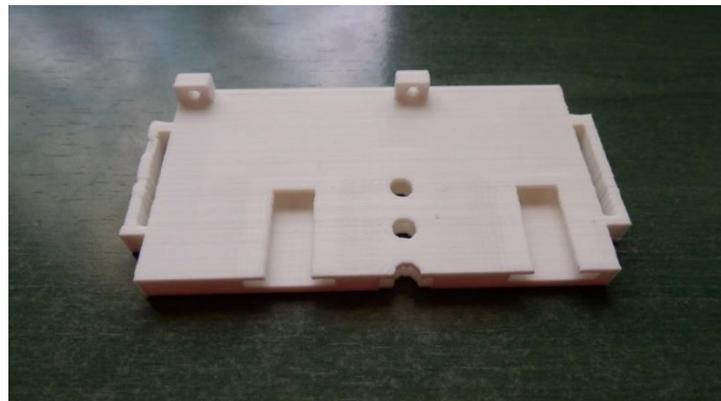


Ilustración 44: Resultado impresión Pieza 3

- Pieza 4 (ilustración 45).



Ilustración 45: Resultado impresión Pieza 4

- Pieza 5 (ilustraciones 46 y 47).



Ilustración 46: Resultado impresión Pieza 4

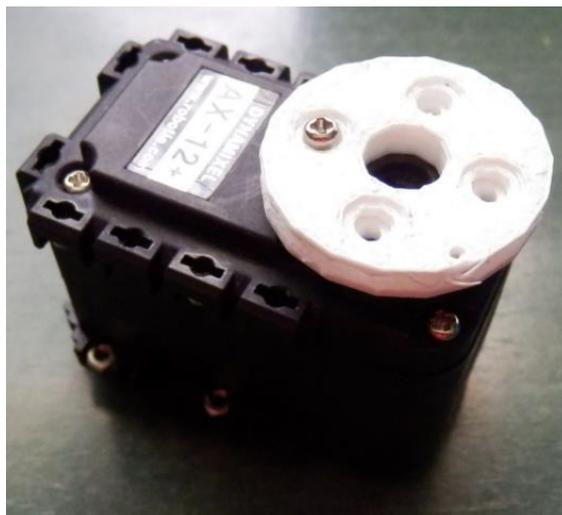


Ilustración 47: Resultado impresión Pieza 4 montad

- Pieza 6 (ilustración 48).

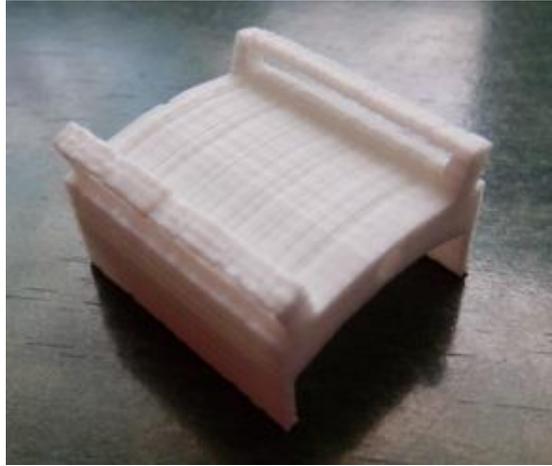


Ilustración 48: Resultado impresión Pieza 5

- Pieza 7 (ilustración 49).

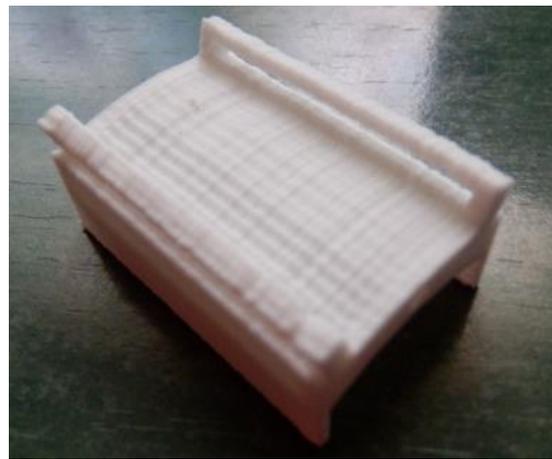


Ilustración 49: Resultado impresión Pieza 6

- Pieza 8 (ilustración 50).

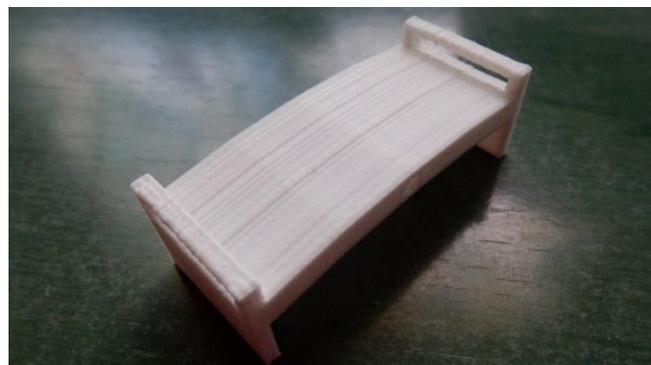


Ilustración 50: Resultado impresión Pieza 7

- Pieza 9 (ilustración 51).

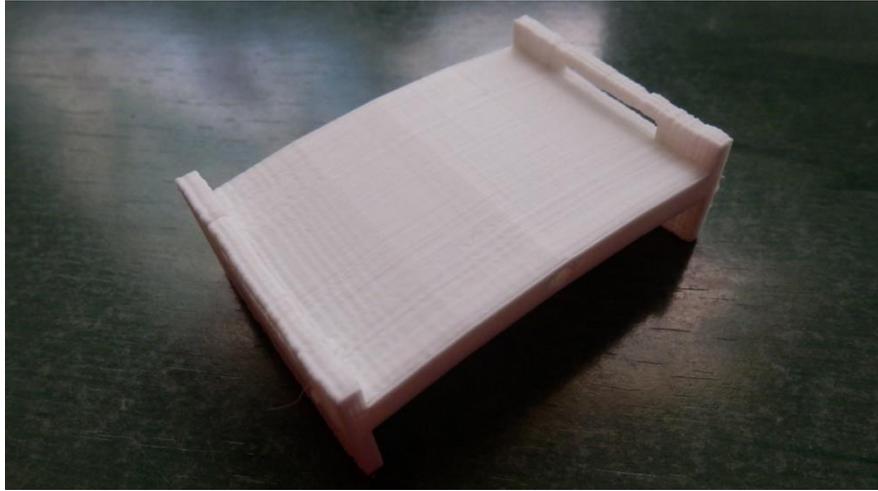


Ilustración 51: Resultado impresión Pieza 9

Las piezas que requerían soportes tuvieron defectos de fabricación ya que la densidad y número de los soportes hacía imposible su eliminación, por eso se realizaron algunos cambios tanto en las piezas como en los soportes.

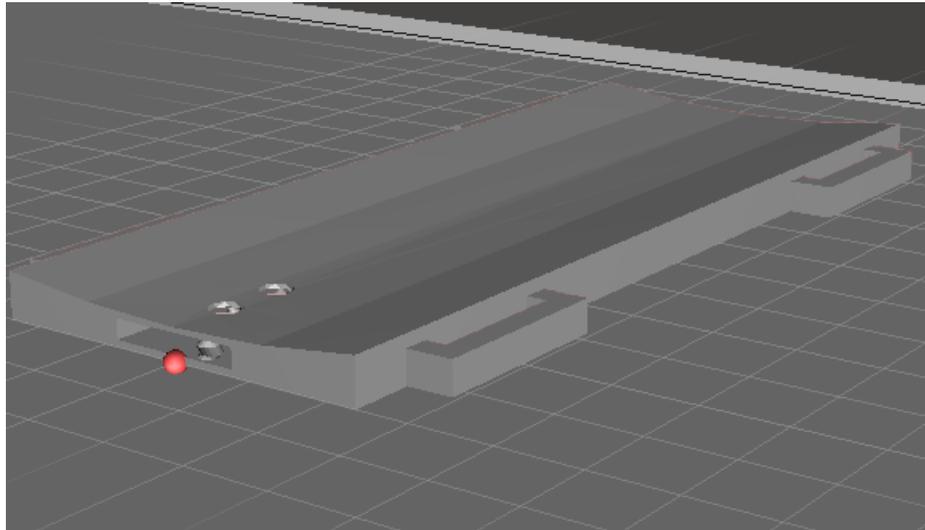


Ilustración 52: Pieza 1 modificada

En la pieza número 1 (ilustración 52), que es la pieza del brazo las guías se situaron en los laterales, gracias a ello la pieza se puede imprimir por la cara plana, lo que facilita mucho su impresión y reduce a casi 0 el número de soportes necesarios.

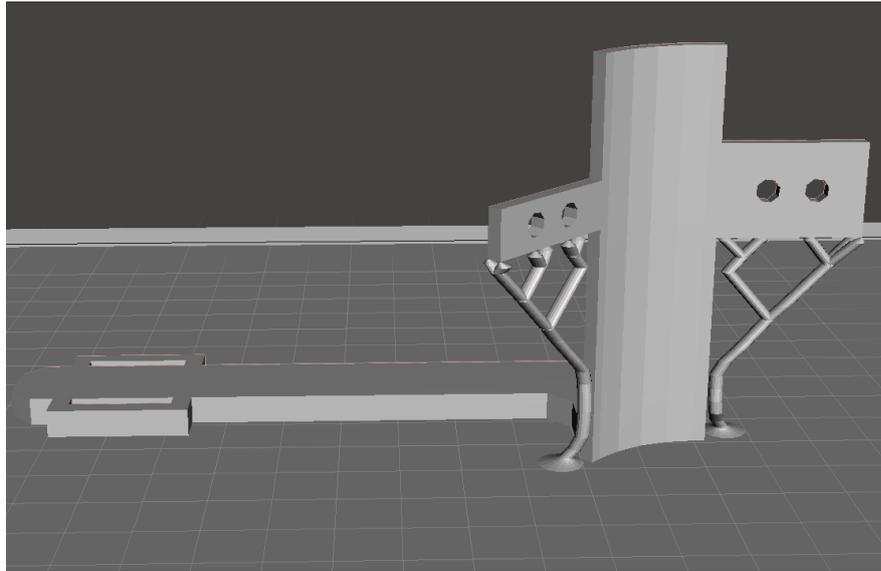


Ilustración 53: Pieza 2 modificada

En la pieza número 2 (ilustración 53) también se han colocado las guías en los laterales, lo que nos permite realizar la impresión sobre una cara lisa. También se ha reducido el ángulo de la pieza de 60° a 45° , lo que deja la mano en una posición mucho más cómoda de cara al paciente. También se han reducido el número de soportes.

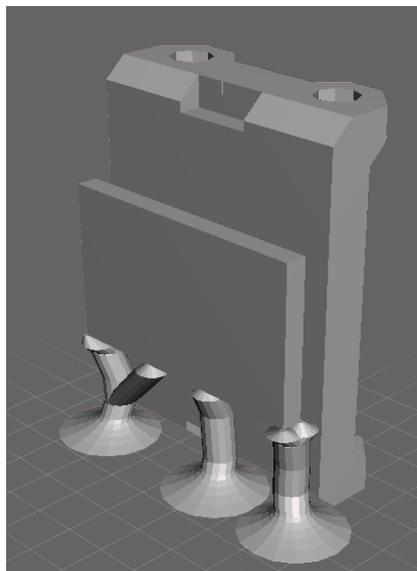


Ilustración 54: Pieza 4 modificada modelo 1

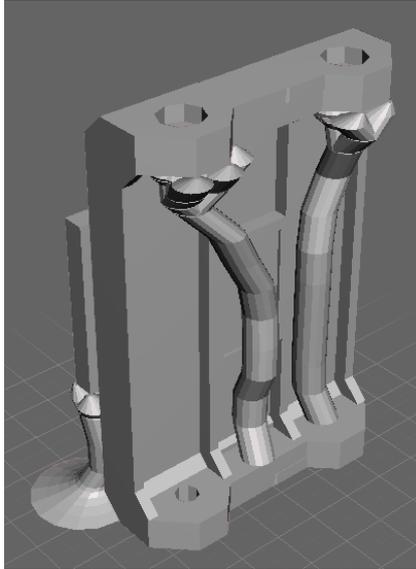


Ilustración 55: Pieza 4 modificada modelo 2

En la pieza de los soportes de los actuadores (ilustraciones 54 y 55) se han eliminado las pestañas para una simplificación de la pieza y así una mejor impresión. También se ha colocado en una posición que nos permite reducir significativamente el número de soportes.

Una vez impresas las piezas y montadas el resultado se puede observar en las ilustraciones 56, 57 y 58.

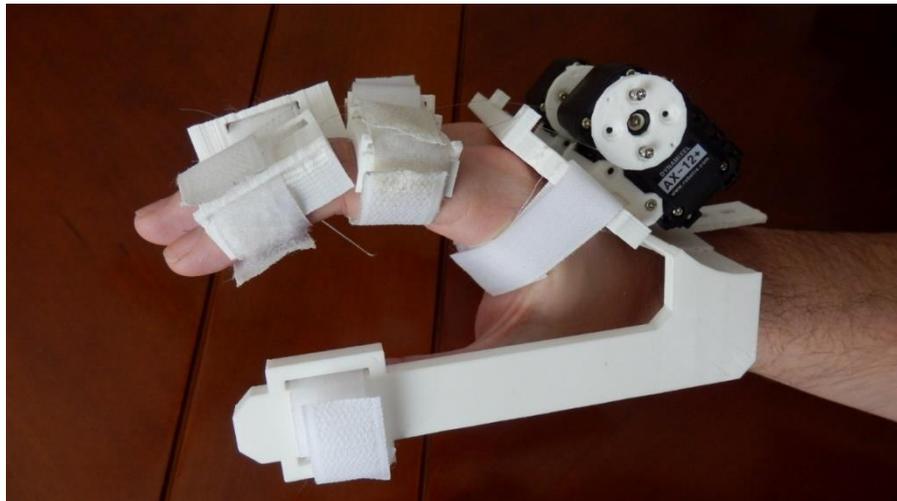


Ilustración 56: Prototipo 3 montado visto de perfil



Ilustración 57: Prototipo 3 montado visto desde abajo

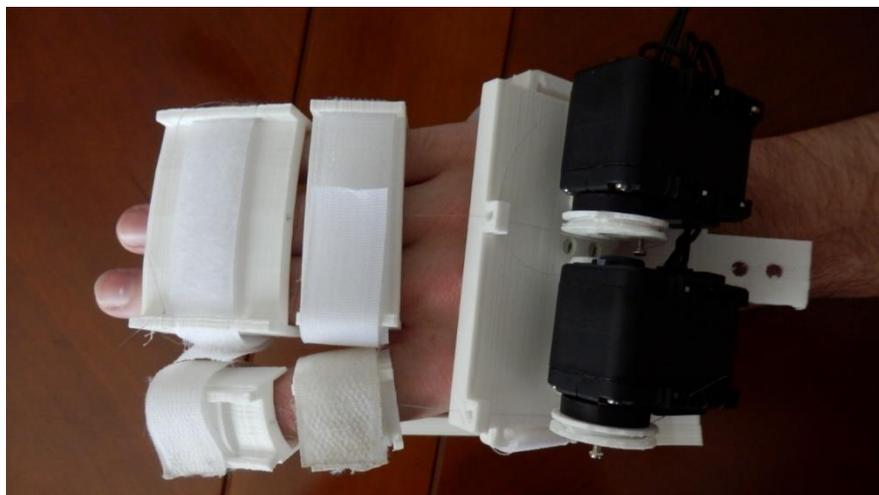


Ilustración 58: Prototipo 3 montado visto desde arriba

Como se puede observar las piezas van sujetas con cintas de velcro a la mano y a los dedos. Una vez que se tiene el prototipo se pueden realizar diferentes pruebas con la programación.

El montaje completo junto a los elementos de control se ve en las ilustraciones 59 y 60.

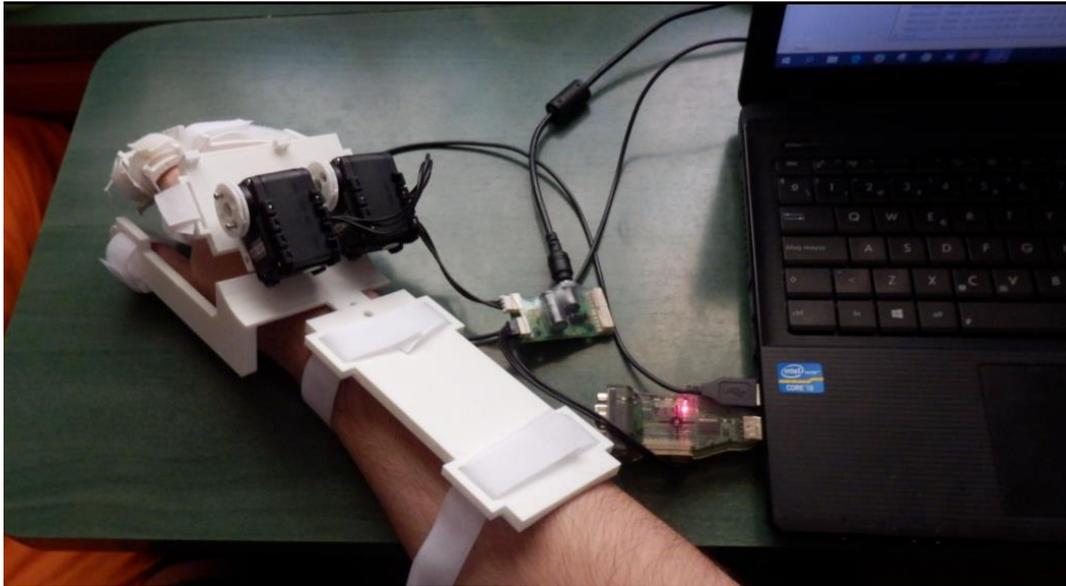


Ilustración 59: Montaje completo 1

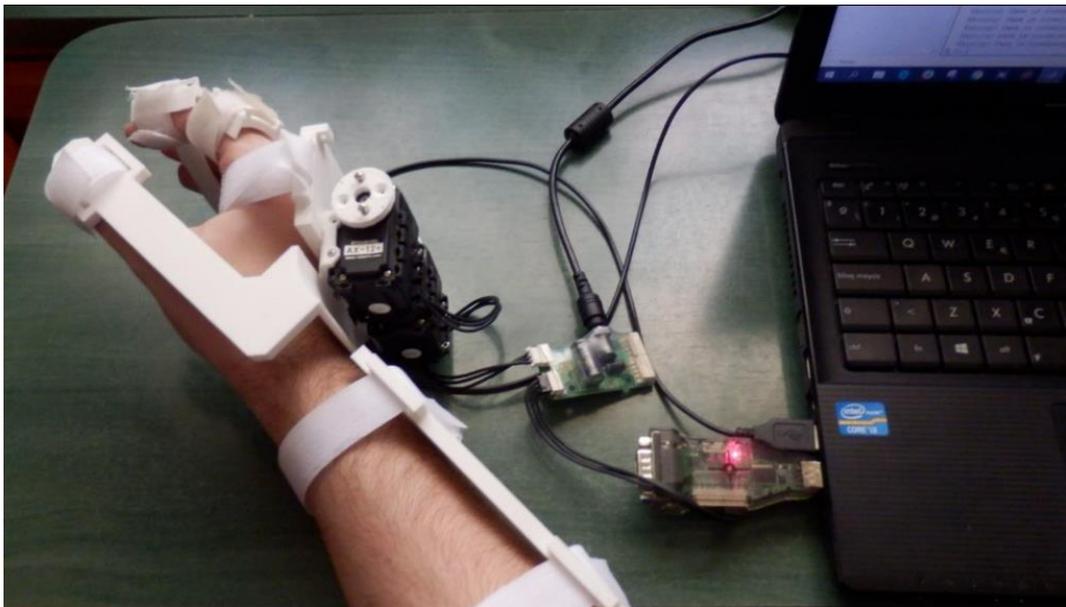


Ilustración 60: Montaje completo 2

Capítulo 4.- Programación y control del prototipo.

4.1.- Consideraciones generales

Como se planteó en los objetivos, las tareas que se quieren programar para el prototipo de robot de rehabilitación son: el ejercicio de pinza completa (utilización de todos los dedos), y el ejercicio de pinza simple (utilización de solo el dedo índice y el pulgar).

El programa que se desarrolle debe ser intuitivo, de fácil manejo, y debe proporcionar información al fisioterapeuta. La programación se ha realizado en MATLAB, con ayuda de una toolbox para poder controlar los motores AX-12.

Los pacientes que han sufrido un ictus tienen la mano cerrada y realizan fuerza con los dedos para evitar una apertura de la mano. Se ha elegido la variable “par motor” como variable a analizar, ya que refleja la fuerza que tiene que proporcionar el motor para superar la fuerza que está ejerciendo el paciente, y así conseguir que el robot pueda abrir la mano del paciente. Con esto se quiere conseguir que el fisioterapeuta sepa en qué momento el paciente está realizando una mayor o menor fuerza, y en función de esta, se propone al paciente la realización de ejercicios de mayor o menor intensidad.

4.2 Software desarrollado

Como se ha dicho en el apartado anterior el programa tiene que ser intuitivo y fácil de aprender, por ello se ha utilizado una interfaz gráfica.

La explicación del software se hará desde el punto de vista de la “estructura del programa”, y desde el punto de vista de la “interfaz gráfica” desarrollada.

4.2.1 Estructura del software

La estructura del programa se puede dividir en tres bloques principales como se indica en la ilustración 61.

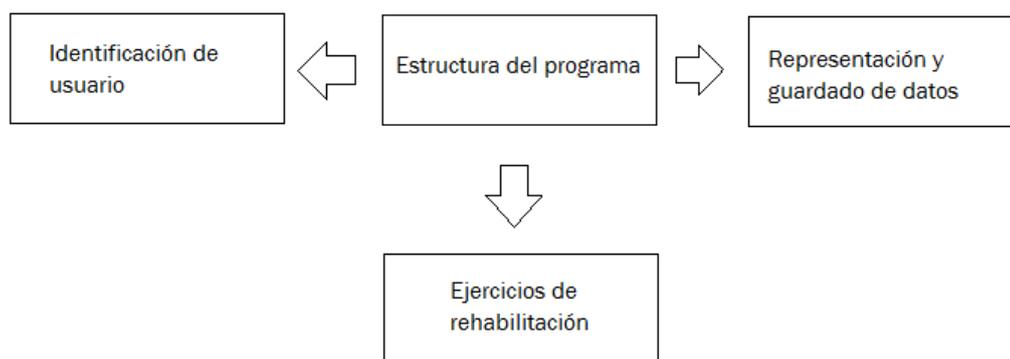


Ilustración 61: Bloques del programa

- **Identificación de usuario:**

El prototipo se espera que vaya a ser utilizado por una variedad de pacientes, por lo que su identificación es necesaria. Para ello se crearon varios subprogramas que creaban una estructura de árbol de directorios en los que se iban guardando los datos obtenidos por cada paciente en su propia carpeta. Para esta estructura se han diseñado tres subprogramas (ilustración 62).

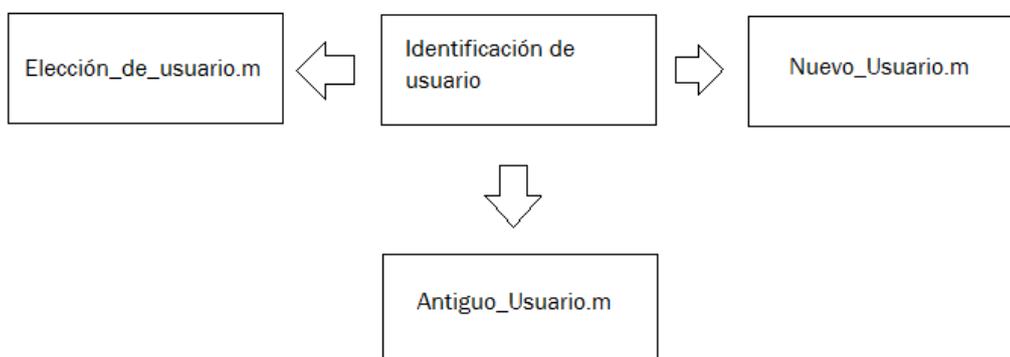


Ilustración 62: Subprogramas para la identificación de usuario

Elección_de_usuario.m, en este subprograma se elige entre si el usuario es nuevo o antiguo.

Nuevo_usuario.m, en este subprograma se introducen los datos del paciente como el nombre completo, la edad y el género para poder crear su directorio en la carpeta 'Pacientes'.

Con los datos obtenidos se crea un directorio dentro del directorio 'Pacientes' con el nombre completo del paciente. Dentro de este directorio se crea un archivo txt con el nombre 'datos' con todos los datos del paciente.

Antiguo_Usuario.m, en este subprograma se recoge el nombre completo del paciente.

Crea una cadena de caracteres llamada 'nombre completo' uniendo los datos recogidos anteriormente mediante la función 'strcat' y busca si existe un directorio con el ese nombre en el directorio 'Pacientes'. Si no existe, por haberse cometido un error o por no existir un paciente con dicho nombre, la ventana se cerrara y se abrirá para que el fisioterapeuta pueda introducir otra vez los datos. Si el directorio existe, guarda la dirección.

- **Ejercicios de rehabilitación:**

Este bloque se puede dividir como se ve en la ilustración 63.

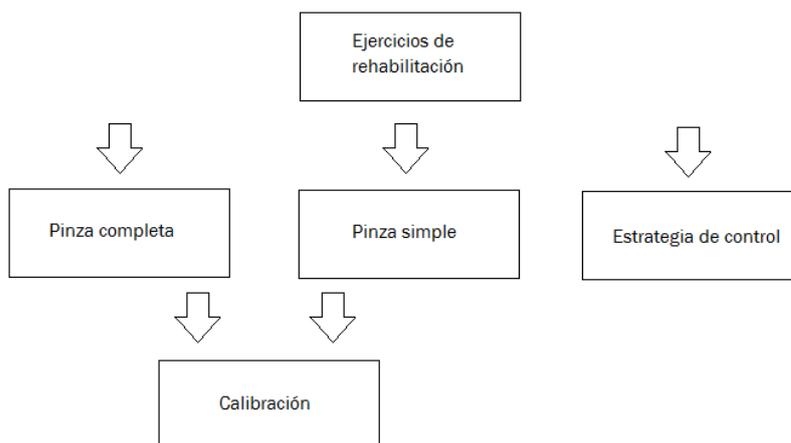


Ilustración 63: Subdivisión de ejercicios de rehabilitación

En este bloque se realizan las distintas tareas de rehabilitación, se ha decidido que el dedo pulgar este inmóvil ya que al poseer dos grados de libertad el control de su movimiento es demasiado complejo. A continuación se explicarán con detalle los distintos sub-bloques:

Pinza completa:

Este sub-bloque está situado en el subprograma Pinza_Completa.m y realiza dos tareas:

- Un ejercicio de apertura y cierre de mano utilizando todos los dedos en sincronía.
- Un ejercicio de agarre.

El control de ambas tareas se explicará en más detalle en el apartado de estrategia de control.

Pinza simple:

Este sub-bloque está situado en el subprograma PinzaSimple.m y realiza la tarea de pinza simple, apertura y cierre del dedo índice. Al principio del programa el motor que controla el resto de los dedos hará que se cierren y se quedará en esa posición.

Calibración:

Para este apartado se han diseñado dos subprogramas, uno para la calibración del ejercicio de pinza completa y otro para el ejercicio de pinza simple, llamados CalibracionCompleta.m y CalibracionSimple.m respectivamente.

En esta calibración se definen los extremos entre los que se moverán los dedos, estos extremos serán los puntos de apertura y cierre entre los que se moverán los dedos. Esta elección se debe a que se prefiere que sea el fisioterapeuta el que elija los extremos del movimiento de los dedos en vez de dejar unos extremos predefinidos.

El fisioterapeuta podrá abrir y cerrar los dedos y guardar los puntos de máxima apertura y cierre que considere oportunos.

En la calibración de la pinza completa se pueden controlar ambos motores por separado para poner los dedos en la misma posición y que el movimiento sea uniforme.

Estrategia de control:

Se han diseñado dos rutinas de control para las tareas:

- La primera para controlar los ejercicios de pinza completa y pinza simple, el control es el mismo pero con la diferencia de que el primero se mueven los dos actuadores y en el segundo solo el actuador del dedo índice. El diagrama de flujo de esta rutina es el que se observa en la ilustración 64.

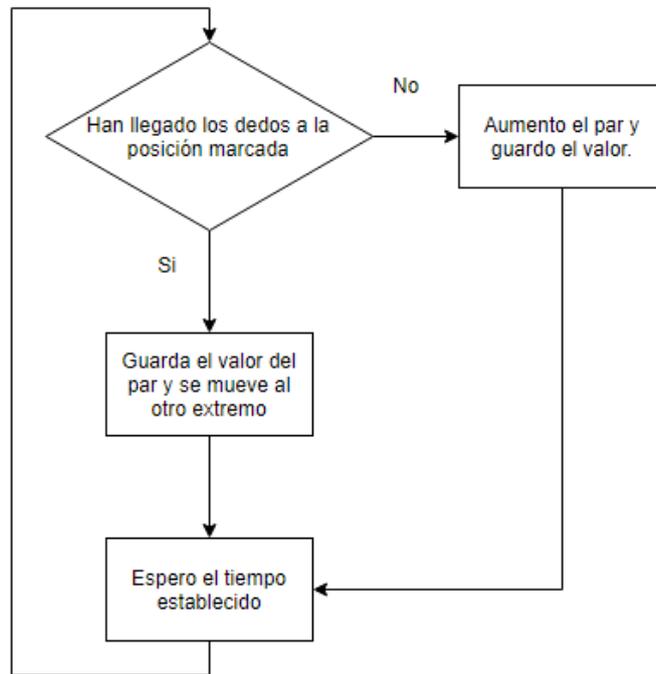


Ilustración 64: Diagrama de flujo de la primera rutina de control

Para conseguir que el actuador pudiese mover el dedo si se ejercía una fuerza elevada se tenía que subir el par del motor, pero no sabía cuanta cantidad de par se tenía que utilizar para que el actuador pudiese mover el dedo ya que dependía de la persona y la situación. Para ello modifique el programa de la siguiente manera: El motor girará entre dos extremos ya predefinidos a una velocidad y aun par mínimos.

Se puso un tiempo en el cual el actuador tenía que moverse entre los dos extremos. Una vez cumplido ese tiempo, se comprobaba la posición del actuador, si la posición estaba en las cercanías del otro extremo, el programa seguía sin cambios, en caso contrario, significaba que el motor no tiene suficiente fuerza, por ello se aumenta el par. Esto continúa hasta que se cumple con la restricción del tiempo.

Cada cierto número de iteraciones el valor del par baja, esto se ha diseñado para que si el valor del par aumenta por un aumento brusco de la fuerza que ejerce el paciente el par vuelva a su valor normal al cabo del tiempo.

Los valores del par de los dos actuadores se regulan por separado, lo que implica que si el paciente está realizando fuerza solo con el dedo índice, solo el actuador del dedo índice subirá el par.

Estos valores se van almacenando en una serie de valores llamadas 'serie' y 'serie2'. Más tarde estos datos serán guardados.

- La segunda es el control realizado para el ejercicio de agarre, en este ejercicio lo que se quiere conseguir es que el prototipo consiga ayudar al paciente a realizar ejercicios de agarre de objetos:

Se realizan un número de iteraciones, en este caso 5, para medir cuanta fuerza está realizando el paciente.

Una vez realizadas dichas iteraciones, se va cerrando la mano y se va comprobando la posición de los actuadores para comprobar en qué momento se realiza el contacto con el objeto. Mientras se va cerrando la mano se realizan intentos de apertura de la mano a un par muy bajo, si se abre, eso indica que la mano ya ha agarrado el objeto y el motor deja de cerrar la mano.

Una vez agarrado el objeto, se empieza a intentar abrir la mano con un par un poco inferior al obtenido en las iteraciones de prueba, con esto se quiere conseguir que el paciente solo sea ayudado en caso de que la fuerza que esté ejerciendo sea menor a la normal.

Con este tipo de ejercicio se quiere incurrir en que el paciente vaya haciendo cada vez menos fuerza cuando quiera dejar de agarrar un objeto y se llegue al momento en el que no haga ninguna fuerza de agarre o haga fuerza de apertura.

Estas rutinas de control se utilizan en Pinza_completa.m y PinzasSimple.m.

- **Representación y guardado de datos:**

Como se ha dicho anteriormente el valor que se desea utilizar para representar y guardar es el valor del par. Tanto sus unidades de medida como su valor son irrelevantes. Esto se debe a que el fisioterapeuta solo debe tener en cuenta los cambios en la misma.

Cada vez que los actuadores llegan a un extremo el valor del par se almacena en un vector de datos. También, cada vez que se aumenta el par se guarda ese nuevo valor.

Al mismo tiempo que el valor del par se almacena también se representa en una ventana en una gráfica para que el fisioterapeuta tenga una representación de la fuerza que está ejerciendo el paciente en cada momento.

Los ejes de dicha representación son 'numero de iteraciones' para el eje x y 'fuerza ejercida' para el eje y.

Los valores del par se guardaran en una carpeta con el nombre de la actividad y que se guarda en un directorio con el nombre del día como se puede ver en la ilustración 65.

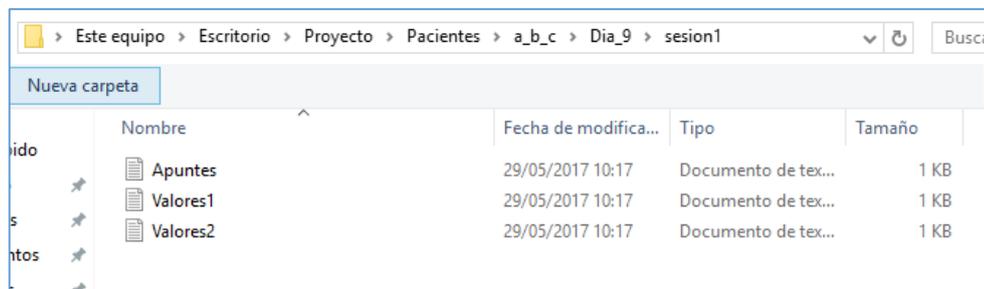


Ilustración 65: Almacenamiento de datos.

Estas acciones se realizan en los subprogramas `guardargrafica.m`, `Pinza_completa.m` y `PinzasSimple.m`.

4.2.2.- Interfaz gráfica

Tras la realización de estas pruebas y verse que las tareas funcionaban se llegó a la conclusión de que la utilización de MATLAB desde consola no era la mejor interfaz de cara al fisioterapeuta. Por ello se ha considerado la realización de una interfaz gráfica con unos controles sencillos e intuitivos para que el fisioterapeuta pueda controlar el programa sin complicaciones.

Para la elaboración de la interfaz gráfica del programa se ha utilizado la herramienta GUIDE de Matlab.

Las GUI (también conocidas como interfaces gráficas de usuario o interfaces de usuario) permiten un control sencillo (con uso de ratón) de las aplicaciones de software, lo cual elimina la necesidad de aprender un lenguaje y escribir comandos a fin de ejecutar una aplicación.

Las apps de MATLAB son programas autónomos de MATLAB con un frontal gráfico de usuario GUI que automatizan una tarea o un cálculo. Por lo general, la GUI incluye controles tales como menús, barras de herramientas, botones y controles deslizantes. Muchos productos de MATLAB, como Curve Fitting Toolbox, Signal Processing Toolbox y Control System Toolbox, incluyen apps con interfaces de usuario personalizadas. También es posible crear apps personalizadas propias, incluidas las interfaces de usuario correspondientes, para que otras personas las utilicen.

Los objetivos para esta interfaz son los siguientes:

- Tiene que ser intuitiva.
- Se tiene que poder elegir entre las diferentes tareas con sencillez.
- Tiene que realizar las tareas programadas.
- Tiene que realizar un feedback de algún tipo, que es uno de los objetivos marcados en el proyecto.

La estructura de la interfaz gráfica se ve en la ilustración 66.

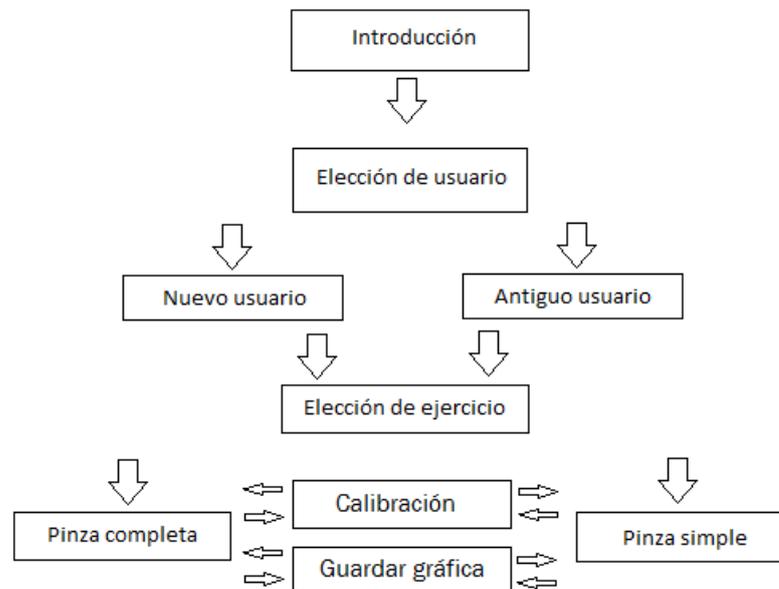


Ilustración 66: Estructura de la interfaz gráfica

Las primeras ventanas que se crearon fueron entonces las de 'Antiguo usuario' y 'Nuevo usuario', a continuación se explicarán en detalle tanto la ventana cómo las tareas que hace.

- **Nuevo usuario:**

Se ha creado esta primero ya que para la tarea de antiguo usuario se necesita un directorio ya creado. Lo que se pide para esta tarea es la creación de un directorio con los datos del paciente. Para la creación de los directorios se tienen que recoger los datos del paciente, la herramienta GUI nos permite la creación de unos cuadros en los que los datos que se escriban en ellos se pueden guardar. La ventana que se ha diseñado es la que se observa en la ilustración 67. Y su nombre es Nuevo_Usuario.fig.

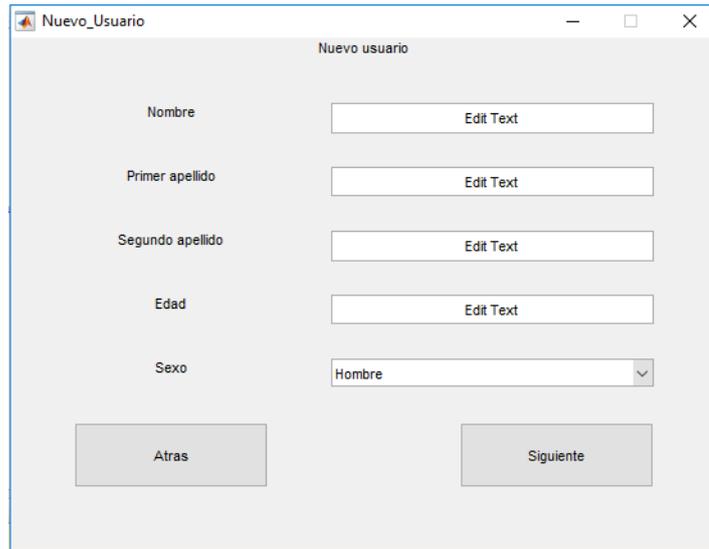


Ilustración 67: Ventana nuevo usuario

Los datos que se recogen son el nombre completo con los apellidos, su edad y el género de la persona.

El botón 'Atrás' nos lleva a otra ventana sin guardar los datos, esto evita que si se realiza algún error en la introducción de los datos estos no se guarden.

El botón 'Siguiete' crea un directorio dentro del directorio 'Pacientes' con el nombre completo del paciente, el nombre completo del paciente se consigue al unir los strings (cadenas de caracteres) del nombre, primer apellido y segundo apellido mediante la función 'strcat' como se puede ver en la ilustración 68.

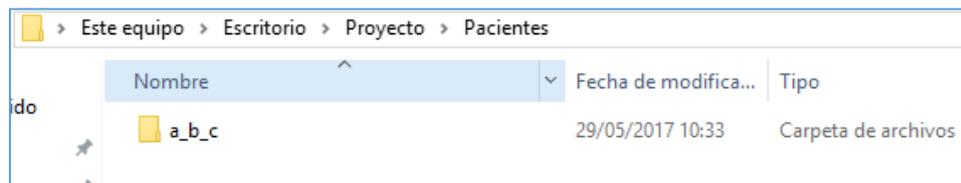


Ilustración 68: Ejemplo de creación de directorio

En este caso a modo de prueba el nombre introducido es a, b, c. Dentro de este directorio se crea un archivo txt con el nombre 'datos' con todos los datos del paciente como se puede ver en la ilustración 69 y 70.

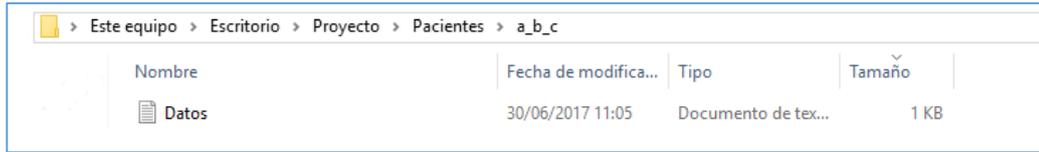


Ilustración 69: Ejemplo de creación de archivo de datos

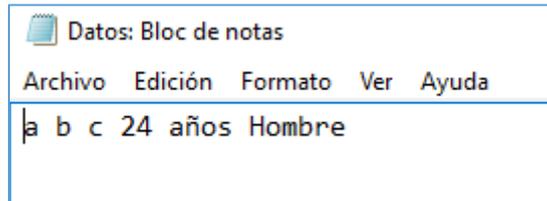


Ilustración 70: Contenido del archivo de texto

Una vez realizado lo anterior, cierra la ventana de 'Nuevo usuario' y abre la ventana siguiente.

- **Antiguo usuario:**

Esta tarea tiene que acceder al directorio ya creado el paciente. La ventana es la siguiente (ilustración 71) y su nombre es Antiguo_Usuario.fig.

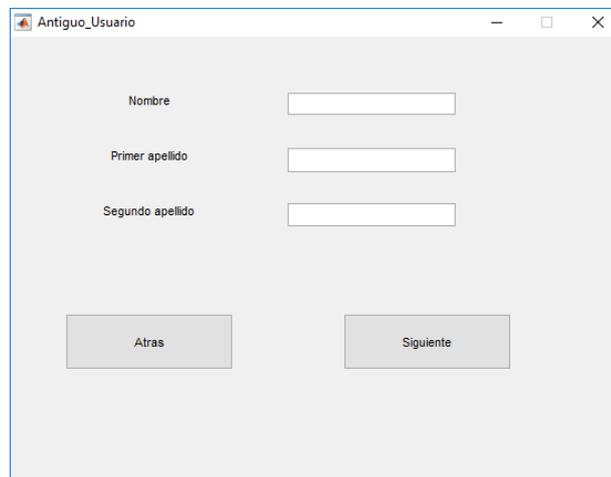


Ilustración 71: Ventana de antiguo usuario

En la ventana, como en la anterior, se introducen los datos del paciente y se van guardando. El botón 'Atrás' cierra esta ventana y nos devuelve a la ventana anterior. El botón 'Siguiete' crea una cadena de caracteres llamada 'nombre completo' uniendo los datos recogidos anteriormente mediante la función 'strcat' y busca si existe un directorio con el ese nombre en el

directorio 'Pacientes'. Si no existe, por haberse cometido un error o por no existir un paciente con dicho nombre, la ventana se cerrara y se abrirá para que el fisioterapeuta pueda introducir otra vez los datos. Si el directorio existe, guarda la dirección, cierra la ventana 'Antiguo usuario' y abre la siguiente ventana.

Una vez creadas las ventanas que nos permitirán elegir entre un usuario nuevo o uno antiguo tenemos que crear las ventanas que nos permitirán la elección de la actividad que se quiere realizar, ya sea pinza completa o pinza simple.

- **Actividades:**

En esta ventana (ilustración 72), a la que se accede con el botón siguiente tanto desde la ventana de nuevo usuario como la de antiguo usuario, nos permitirá elegir que tarea queremos realizar, pinza simple o pinza completa. El fichero se llama Actividades.fig.

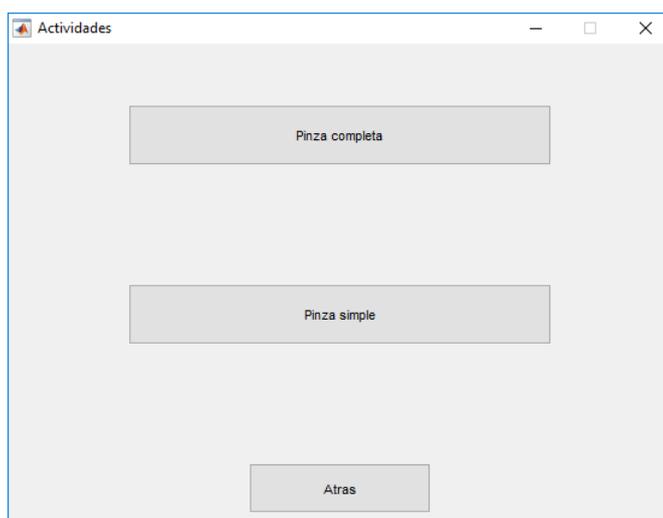


Ilustración 72: Ventana de elección de actividad

En esta ventana se puede elegir entre pinza completa y pinza simple y se puede volver a la ventana anterior.

- **Pinza completa:**

En esta ventana se podrán elegir entre varios ejercicios que se pueden hacer realizando movimientos de pinza completa. La ventana se puede ver en la ilustración 73 y su nombre es Pinza_Completa.fig.

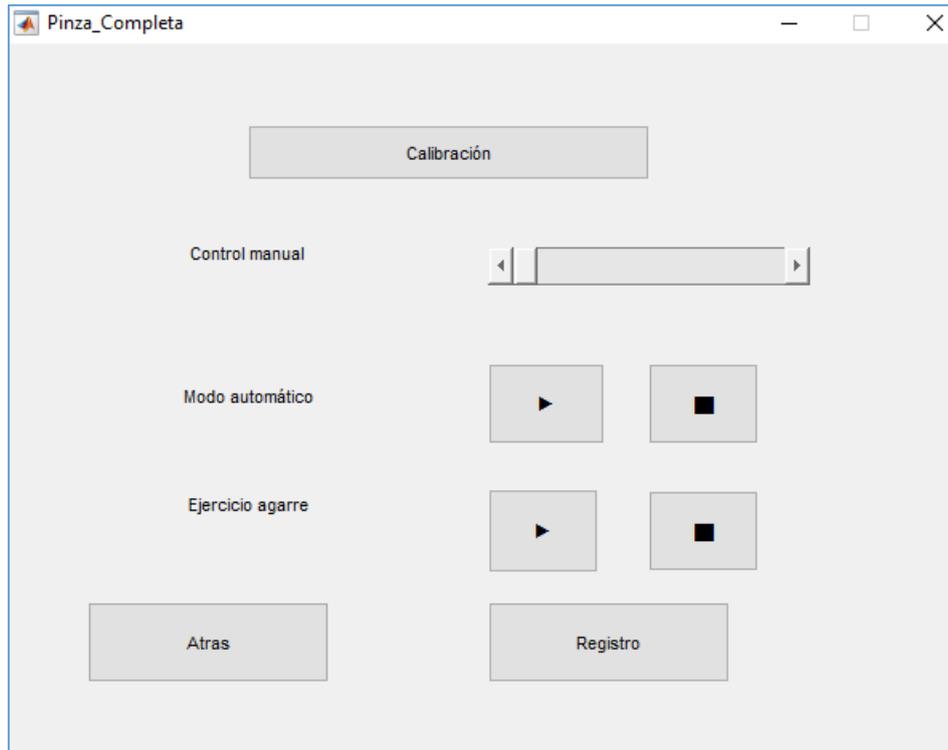


Ilustración 73: Ventana de pinza completa

En dicha ventana se pueden ver las siguientes características:

- **Calibración:** En este botón se accederá a una ventana nueva en la que podremos hacer los diferentes ajustes iniciales, esta ventana se explicara más adelante.
- **Control manual:** Con esta barra horizontal se podrá controlar la apertura y cierre de la mano manualmente, por si existiese algún problema. Esta barra está dividida en diferentes pasos en los que se podrá posicionar los actuadores. Estos pasos se consiguen mediante el uso de la función 'switch' en la que se pone la posición del intervalo a la que se quiere llegar en cada caso.
- **Modo automático:** En este modo los dedos se abrirán y cerrarán automáticamente hasta que se pulse el botón de stop. En este modo el botón de play pone en funcionamiento el programa mientras que el botón de stop para el programa y da paso al guardado de los datos obtenidos del par. Al dar al botón play se abre la ventana de 'Guardar gráfica'. Cuando se ejecuta el modo automático se van mostrando por pantalla los datos que se van recogiendo en una gráfica como se ve en la ilustración 74.

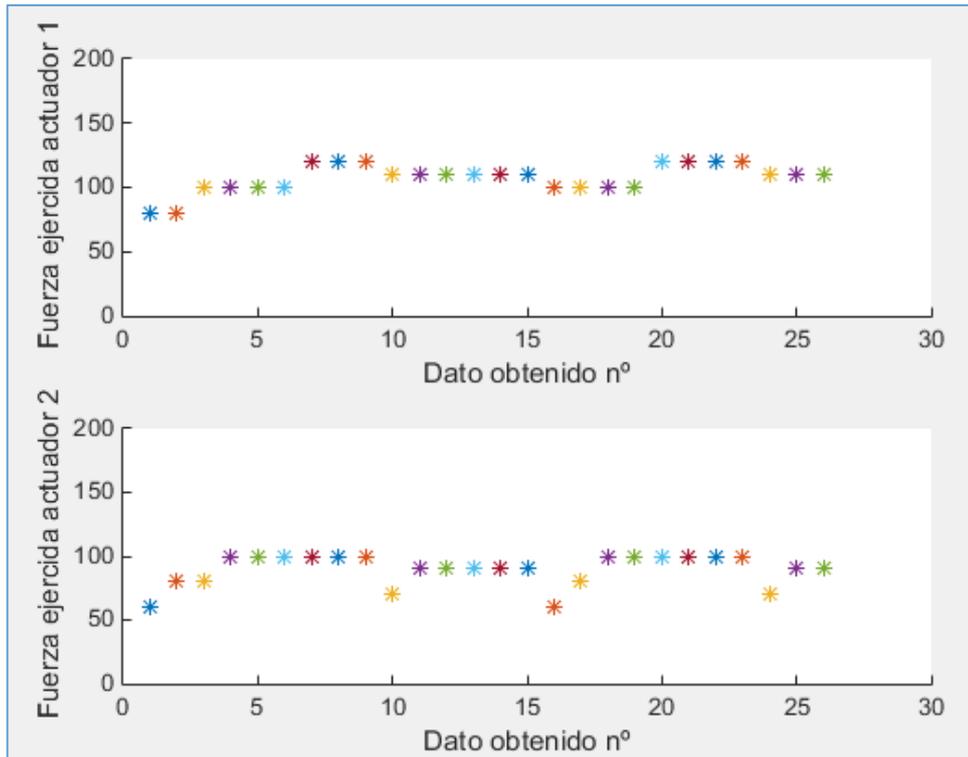


Ilustración 74: Gráfica de datos pinza completa

- Ejercicio de agarre: En este ejercicio lo que se quiere conseguir es que el prototipo consiga ayudar al paciente a realizar ejercicios de agarre de objetos. Con el botón 'stop' se para el programa, en este programa no se guarda ningún dato.
- El botón 'Atrás' nos lleva de vuelta a la ventana de 'Elección de usuario'.
- El botón de 'Registro' nos permite visualizar los datos guardados anteriormente en distintas sesiones de rehabilitación. Nos permite abrir solamente una gráfica a la vez, es decir, los valores de un actuador. Cuando se pulsa el botón se abre una ventana de Windows en la que tendremos que buscar los valores que nos interesen, buscando el nombre del paciente en la carpeta 'Pacientes' y eligiendo el día y la sesión, por ejemplo los pasos son los siguientes, ilustraciones 75, 76 y 77.

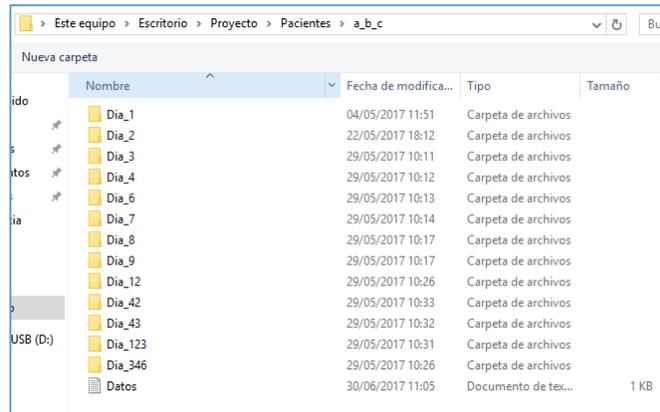


Ilustración 75: Paso 1 de la búsqueda

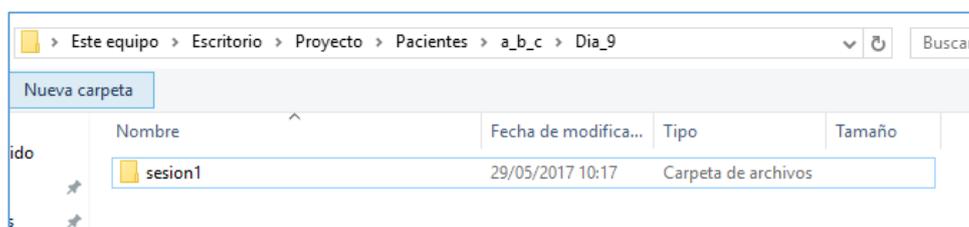


Ilustración 76: Paso 2 de la búsqueda

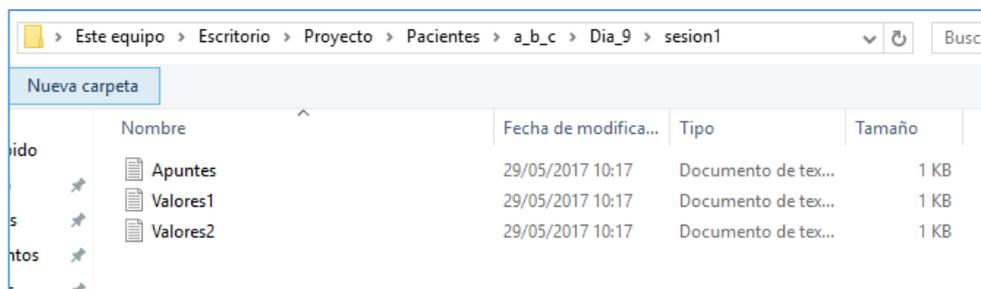


Ilustración 77: Paso 3 de la búsqueda

Donde se elegirá los valores del actuador que se quiera, actuador 1 dedo índice y actuador 2 el resto de los dedos. Esta búsqueda se realiza con la función 'uigetfile' que nos permite coger un fichero de una dirección. Una vez se coge ese fichero se abre y se recorre buscando los datos y luego se representan dichos datos en una gráfica.

- **Pinza simple:**

Esta tarea realiza el movimiento de pinza simple, es decir, solo realiza el movimiento el dedo índice. La ventana es la que se observa en la ilustración 79 y su nombre de fichero es PinzaSimple.fig.

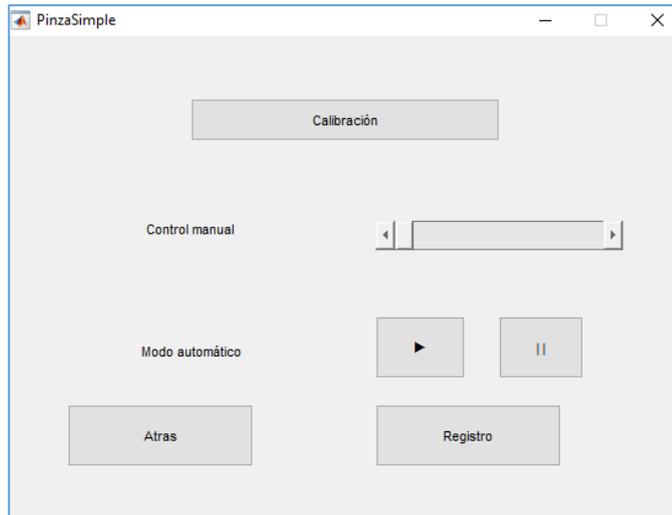


Ilustración 78: Ventana pinza simple

El botón de calibración abre la ventana de calibración, que se explicará cuando se explique su ventana correspondiente.

El control manual es el mismo caso que en el de la pinza completa, es una barra con pasos con la cual se puede abrir y cerrar el dedo índice en este caso por pasos manualmente.

El modo automático es también como el anterior, solo que en este caso solo se utiliza el actuador del dedo índice y solo se guardan los datos del par de dicho actuador.

Solo ve representan los datos del par del actuador del dedo índice, como se puede ver en la ilustración 77.

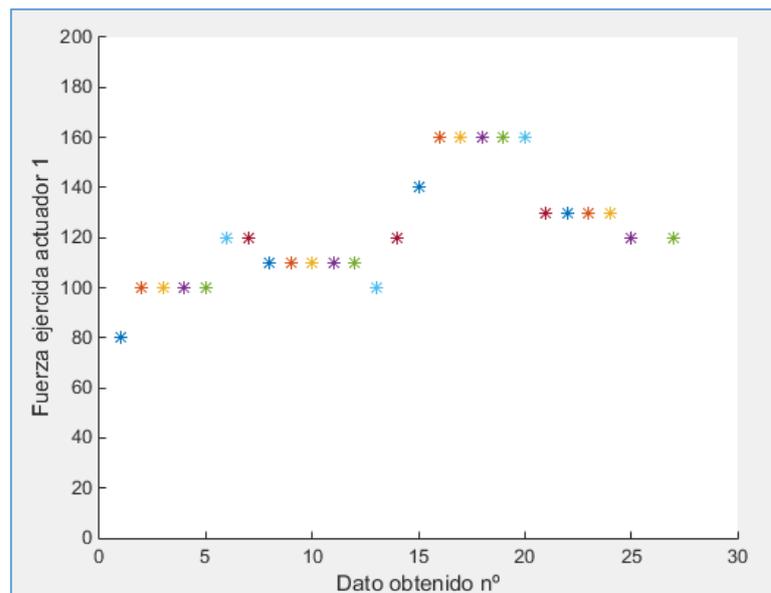


Ilustración 79: Gráfica de datos pinza simple

Los valores de los ejes se definen de antemano de poner en marcha la tarea en modo continuo. El botón de 'Registro' y 'Atrás' realizan las mismas funciones que en el caso de la pinza completa.

Una vez vistas las ventanas de las actividades ahora se van a explicar las ventanas auxiliares que nos permitirán tanto calibrar los actuadores como guardar los datos obtenidos anteriormente:

- **Calibración:**

En dicha ventana se definen los extremos entre los que se moverán los dedos para que así el fisioterapeuta pueda elegir el intervalo que mejor se ajuste a lo deseado.

A esta ventana se accede desde el botón 'Calibración' tanto desde la ventana de 'Pinza Completa' como de 'Pinza Simple', pero son ventanas diferentes. Aunque las funciones son prácticamente las mismas tienen pequeños detalles que se explicaran a continuación.

Calibración simple:

Es la calibración para el movimiento de pinza simple y su ventana se ve en la ilustración 80 y nombre de fichero es CalibracionSimple.fig.

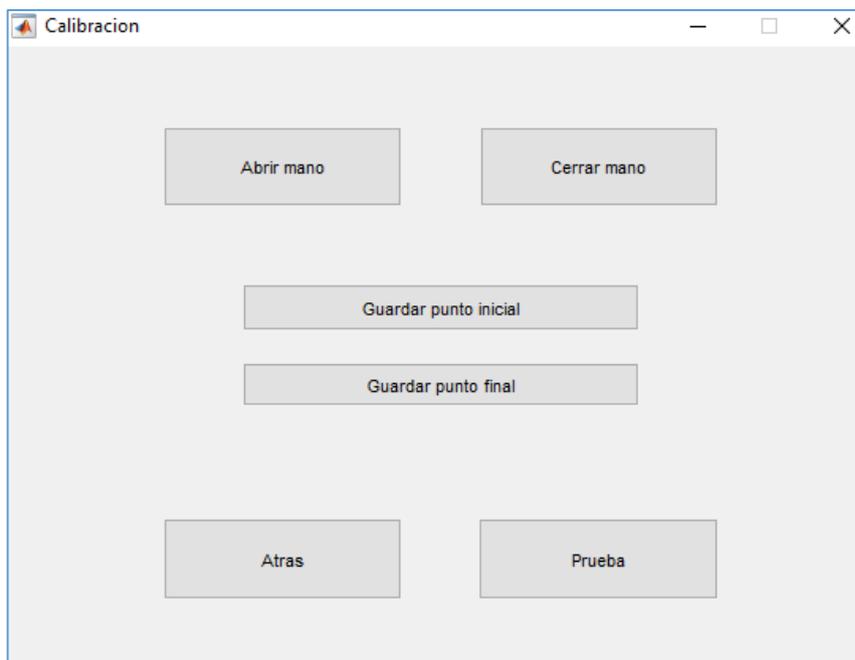


Ilustración 80: Ventana calibración pinza simple

Tanto los botones 'abrir mano' como 'cerrar mano' realizan los movimientos de los dos actuadores simultáneamente ya que aunque lo único que nos interesa es el dedo índice la posición del resto de los dedos tiene que ser

cómoda. Una vez que se han alcanzado tanto el punto inicial como el punto final del movimiento estos se guardan en variables globales para que puedan ser utilizadas en los ejercicios de pinza simple.

El botón de 'Prueba' realiza un movimiento de apertura y cierre de mano para comprobar que los extremos son los deseados. El botón 'Atrás' nos devuelve a la ventana del ejercicio.

Calibración completa:

La ventana es la que aparece en la ilustración 81 y su nombre es CalibracionCompleta.fig.

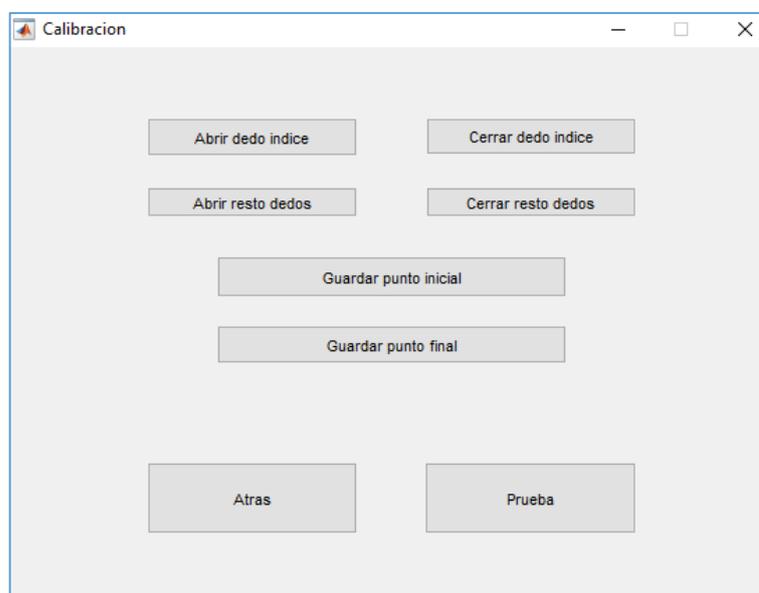


Ilustración 81: Ventana calibración pinza completa

La única diferencia de dicha ventana es que la calibración de los dos actuadores va separada ya que lo que nos interesa es que el movimiento sea sincronizado y que los extremos de movimiento de todos los dedos sea el mismo.

Guardar gráfica:

Cuando se pulsa el botón 'stop' tanto del ejercicio de pinza completa como pinza simple los datos obtenidos del par van a ser guardados mediante la ventana de la ilustración 82 y su nombre es guardargrafica.fig.

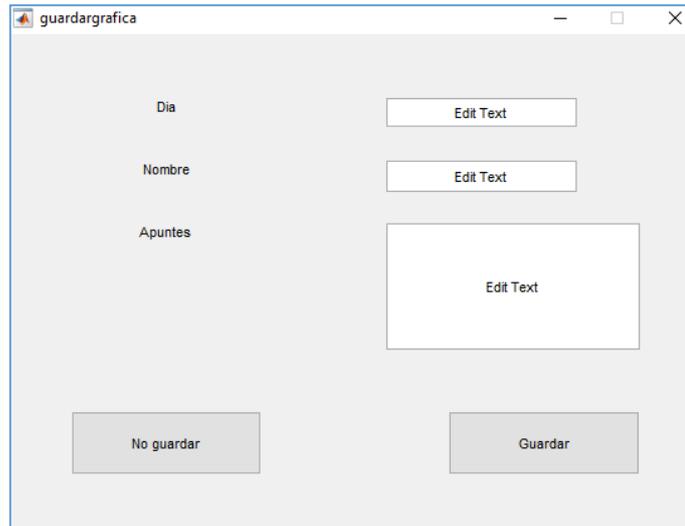


Ilustración 82: Ventana guardar gráfica

En esta ventana se pueden ver varios campos a completar como el día de la sesión de la rehabilitación, el nombre que se va a poner a dicho ejercicio y un campo en el que se pueden poner detalles del ejercicio.

Si se pulsa el botón 'no guardar' los datos obtenidos no se guardan y se vuelve a la ventana anterior.

Si se pulsa el botón 'Guardar' se guardan los valores obtenidos en archivos txt con el nombre 'Valores1' para el actuador 1, 'Valores2' para el actuador 2 y apuntes para los detalles que se hayan querido guardar de la sesión.

4.3 Primeras pruebas

Una vez realizada la interfaz gráfica se han realizado un conjunto de pruebas para comprobar que el programa funciona correctamente.

La primera prueba ha consistido en no ejercer ningún tipo de fuerza con los dedos en el programa de pinza completa, al no hacer fuerza con los dedos, el prototipo ha abierto los dedos y ha dejado la mano abierta.

Los datos obtenidos son los que se observan en la ilustración 83.

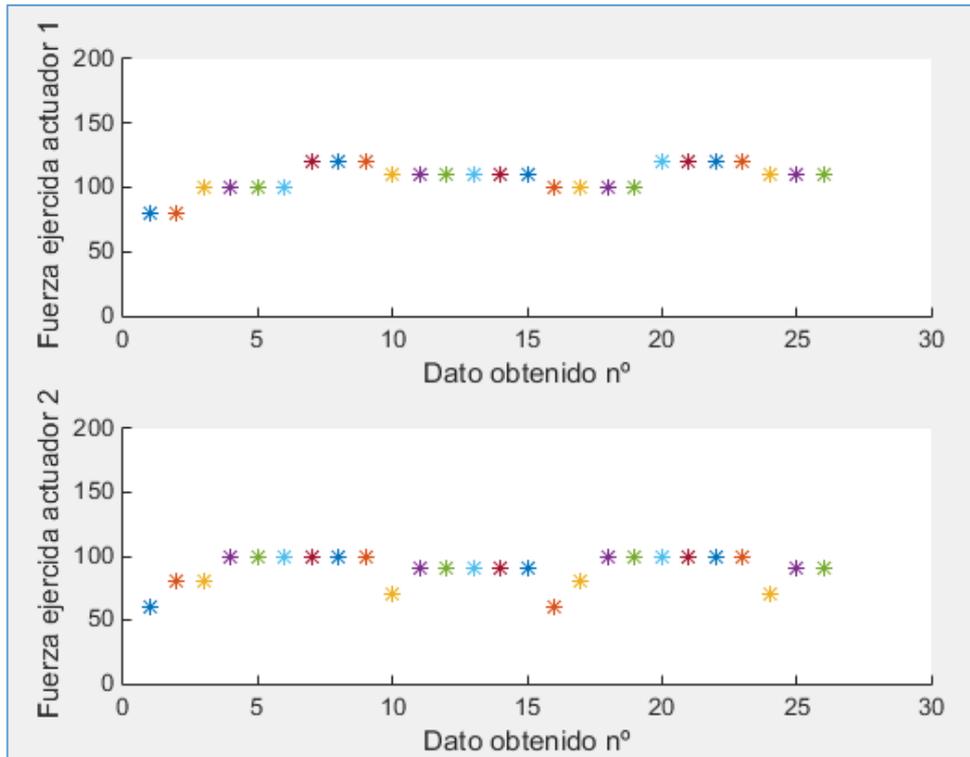


Ilustración 83: Datos prueba sin fuerza

Como se puede observar al principio existe una subida del par correspondiente a la resistencia mecánica del prototipo pero en las siguientes iteraciones no se produce ninguna subida elevada del par.

También se pueden observar las bajadas periódicas para nivelar el valor del par en caso de una subida. Al no haber fuerza, tras esta bajada, los valores del par vuelven a los valores normales.

Esta prueba también se ha realizado en el programa de pinza simple cuyo resultado se puede observar en la ilustración 84.

En la segunda prueba se ha ejercido fuerza en el programa de pinza completa primero con todos los dedos, luego con solo el dedo índice y luego con todos menos el índice. El resultado se ve en la ilustración

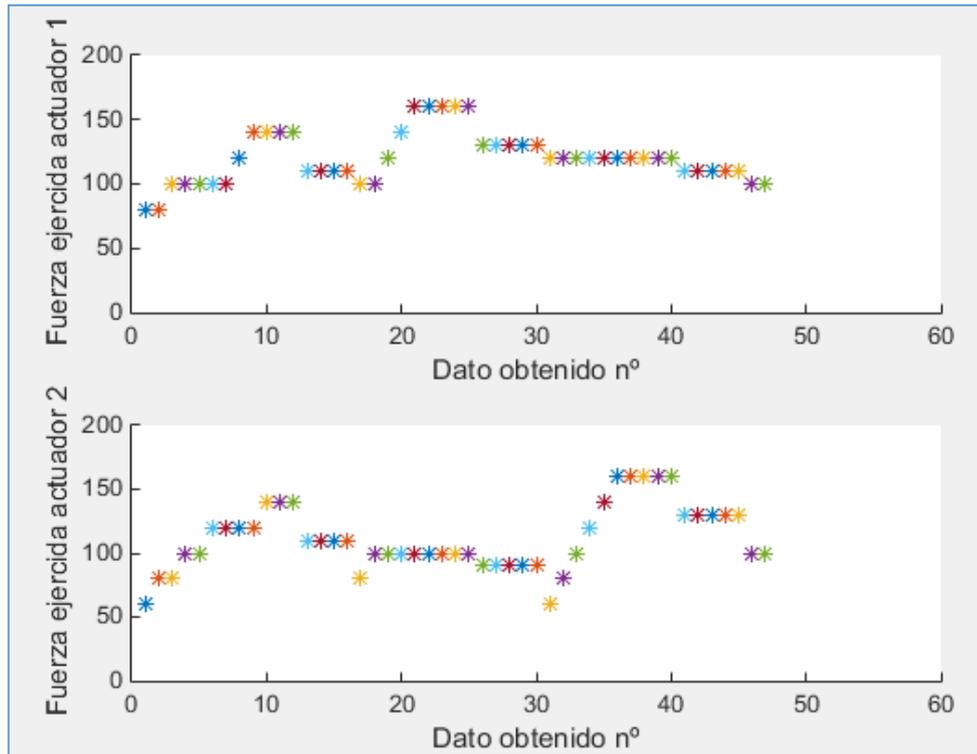


Ilustración 84: Resultados prueba 2

Como se puede observar al principio se produce un aumento en el valor en ambos actuadores, luego un aumento solo en el actuador 1 y luego un aumento solo en el actuador 2.

También se ha realizado la prueba con el programa de pinza simple y el resultado se puede apreciar en la ilustración 84.

5.- Conclusiones

Al comienzo del proyecto se marcaron unos objetivos, estos eran hacer un prototipo de un robot de rehabilitación que realizase una serie de ejercicios, pinza simple y completa, tuviese un diseño cómodo para el paciente y fuese fácil de aprender a utilizarlo.

Para ello se ha elegido un prototipo cuyas piezas han sido realizadas mediante impresión 3D y cuyo elemento transmisor de fuerza es un cable.

Esta ha sido una primera aproximación para intentar alcanzar los objetivos. Si bien se buscaba cumplir los objetivos, nunca se ha perdido de vista que es una primera aproximación, por lo que el prototipo no es perfecto, pero si nos ha permitido poder avanzar en una posible línea de trabajo de robots de rehabilitación de mano: robots guiados por cables.

De todas formas hay varios puntos cuyo enfoque creo que ha sido acertado:

- El uso de cables. Aunque su utilización no haya sido optima, ni mucho menos, creo que puede ser un buen elemento transmisor de fuerza. En un futuro, cuando los cables puedan variar su longitud por si mismos al encogerse o expandirse, creo que podrán ser utilizados en este tipo de dispositivos.
- La utilización de una interfaz gráfica. El programa tiene que ser lo más sencillo e intuitivo posible para el paciente y para el fisioterapeuta.

En conclusión, el trabajo que he realizado ha sido la primera aproximación al problema planteado de diseño de dispositivos mecatrónicos para rehabilitación de mano, que espero que sirva para que futuros desarrollos puedan seguir con la idea planteada.

El prototipo se puede mejorar de muchas maneras. A continuación voy a citar algunas posibles ideas:

- Mejora del modelo mecánico, ya sea por un mejor diseño de las piezas que lo componen, como por la utilización de distintos materiales. Las piezas de transmisión de fuerza situadas en los dedos pueden tener un diseño mucho más cómodo para el paciente.
- Mejora del programa informático. Se pueden añadir más ejercicios y se pueden guardar más tipos de datos. Más tarde estos datos se pueden utilizar para una base de datos para comparar distintos pacientes con dolencias similares.

6.- Bibliografía

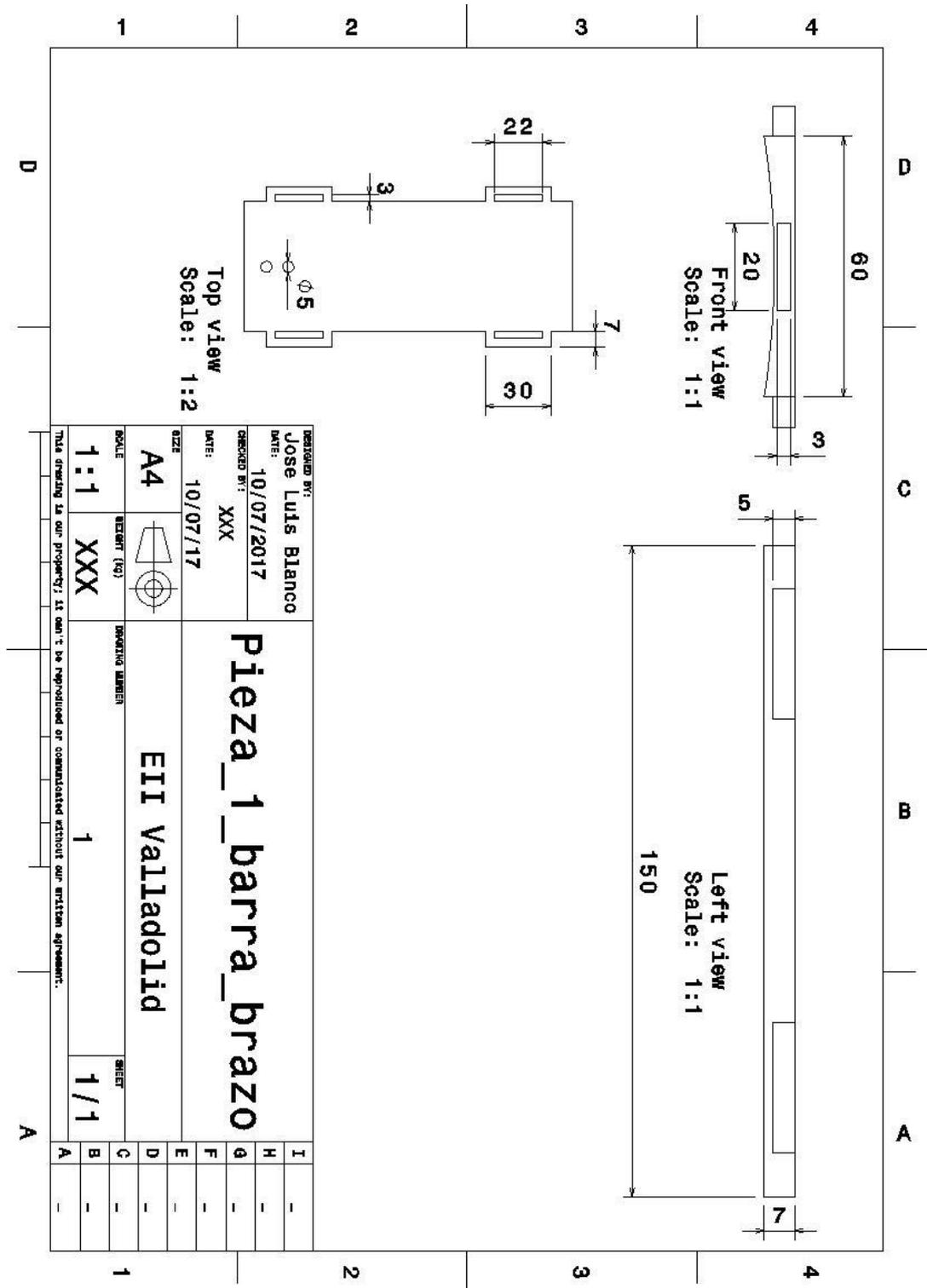
- Chen Chen F., Favetto A., Mousavi M., Ambrosio E.P., Appendino S., Battezzato A., Manfredi D.,Pescarmona F., Bona B. (2011). Human Hand: Kinematics, Statics and Dynamics. In: International Conference on Environmental Systems, Portland, Oregon, 17 - 21 Luglio 2011.
- Fraile Marinero, J.C. “Control del motor Dynamixel desde un ordenador PC con matlab”. Apuntes asignatura ‘Taller de robótica’. Universidad de Valladolid 2015.
- Schabowsky, C. N., Godfrey, S. B., Holley, R. J., & Lum, P. S. (2010). Development and pilot testing of HEXORR: hand exoskeleton rehabilitation robot. Journal of neuroengineering and rehabilitation, 7(1), 36.

Webs de información del estado del arte consultadas durante mes de noviembre de 2016.

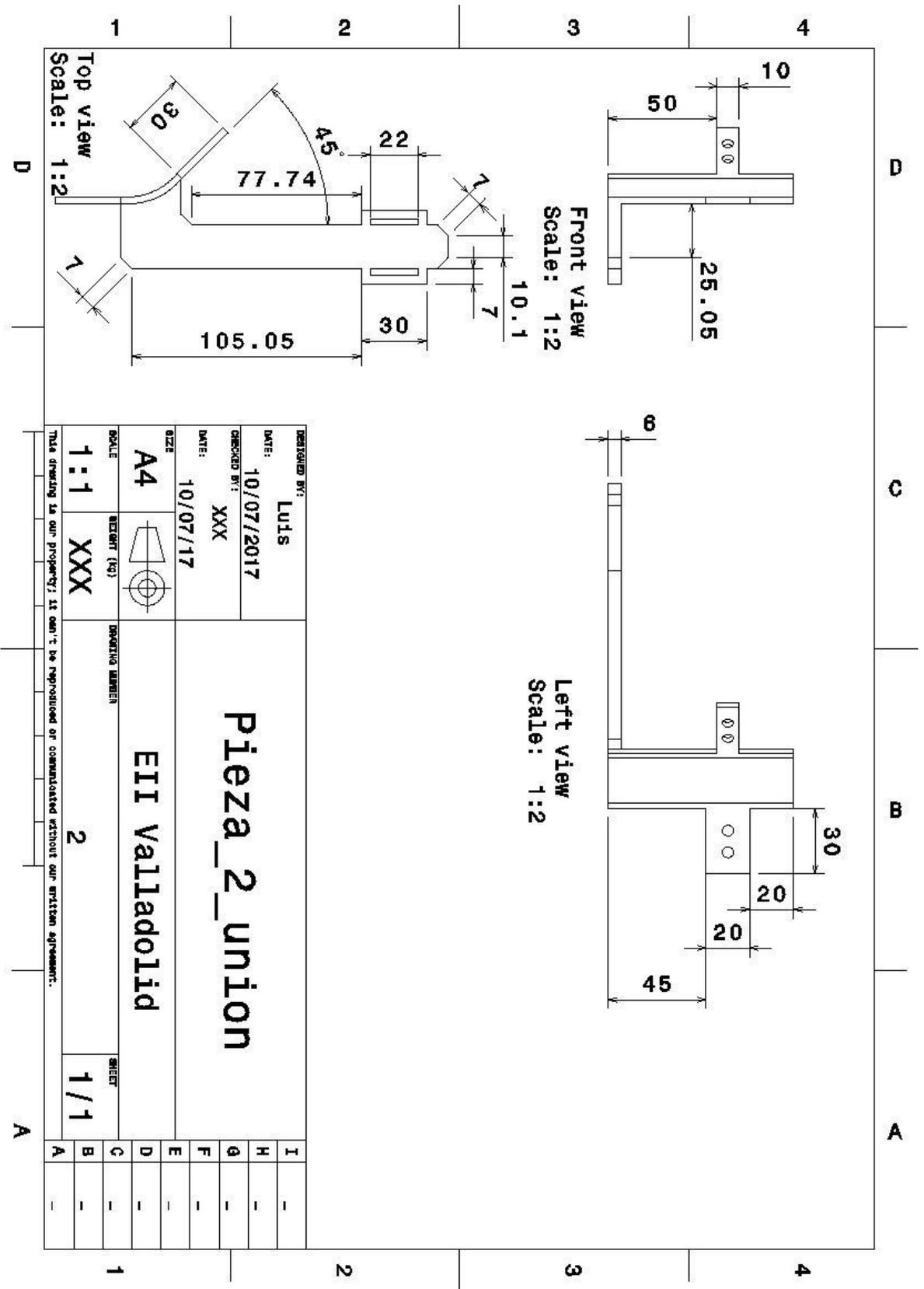
- <http://www.rehab-robotics.com/hoh/index.html>
- <http://tyromotion.com/en/products/amadeo>
- <http://rehabilitationrobotics.net/cms3/>
- [http://support.robotis.com/en/.](http://support.robotis.com/en/)
- http://support.robotis.com/en/software/dynamixel_sdk/usb2dynamixel/usb2dxl_windows.htm
- http://petercorke.com/Robotics_Toolbox.html

Anexo I: Planos de las piezas realizadas en matlab

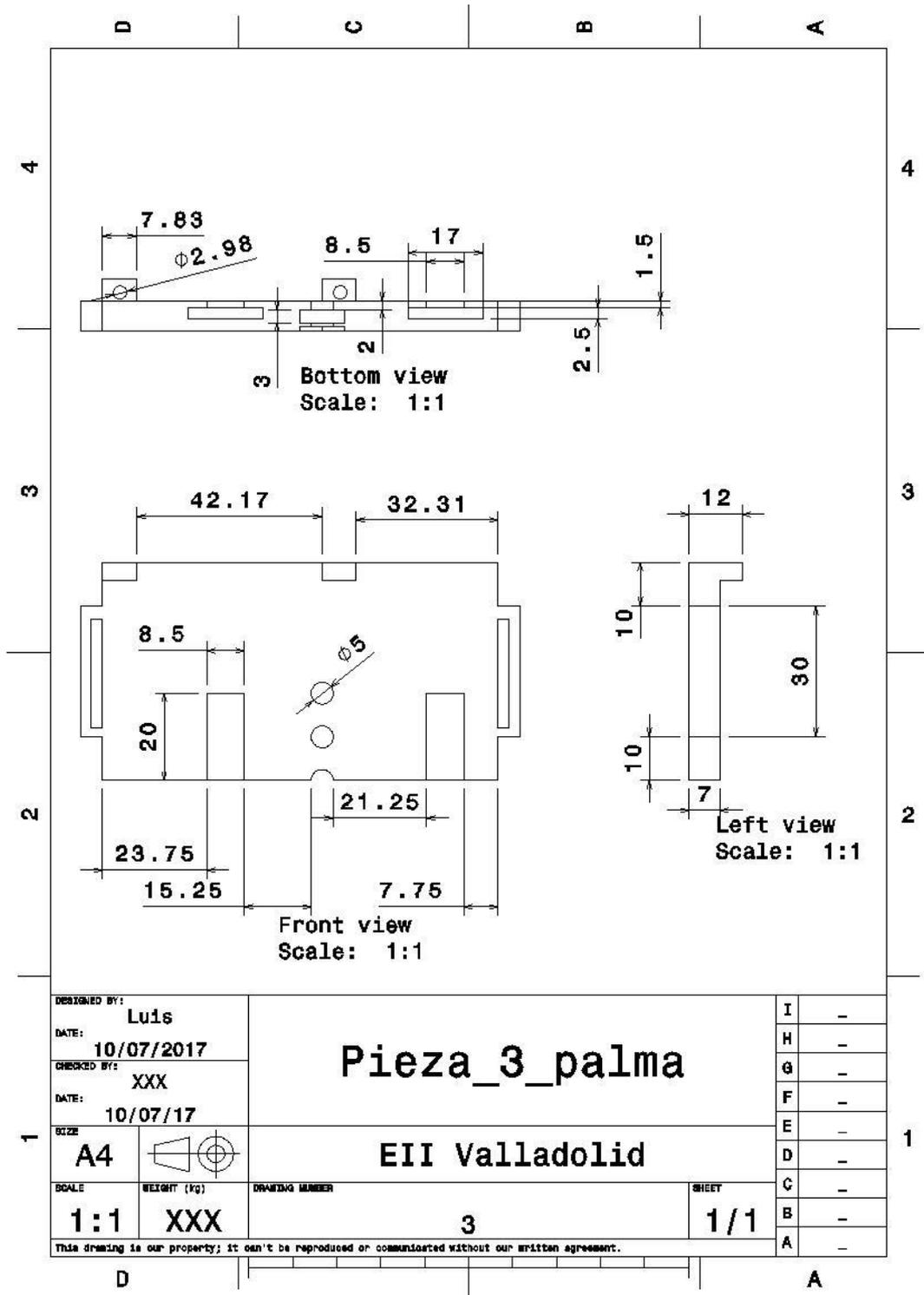
Pieza 1:



Pieza 2:

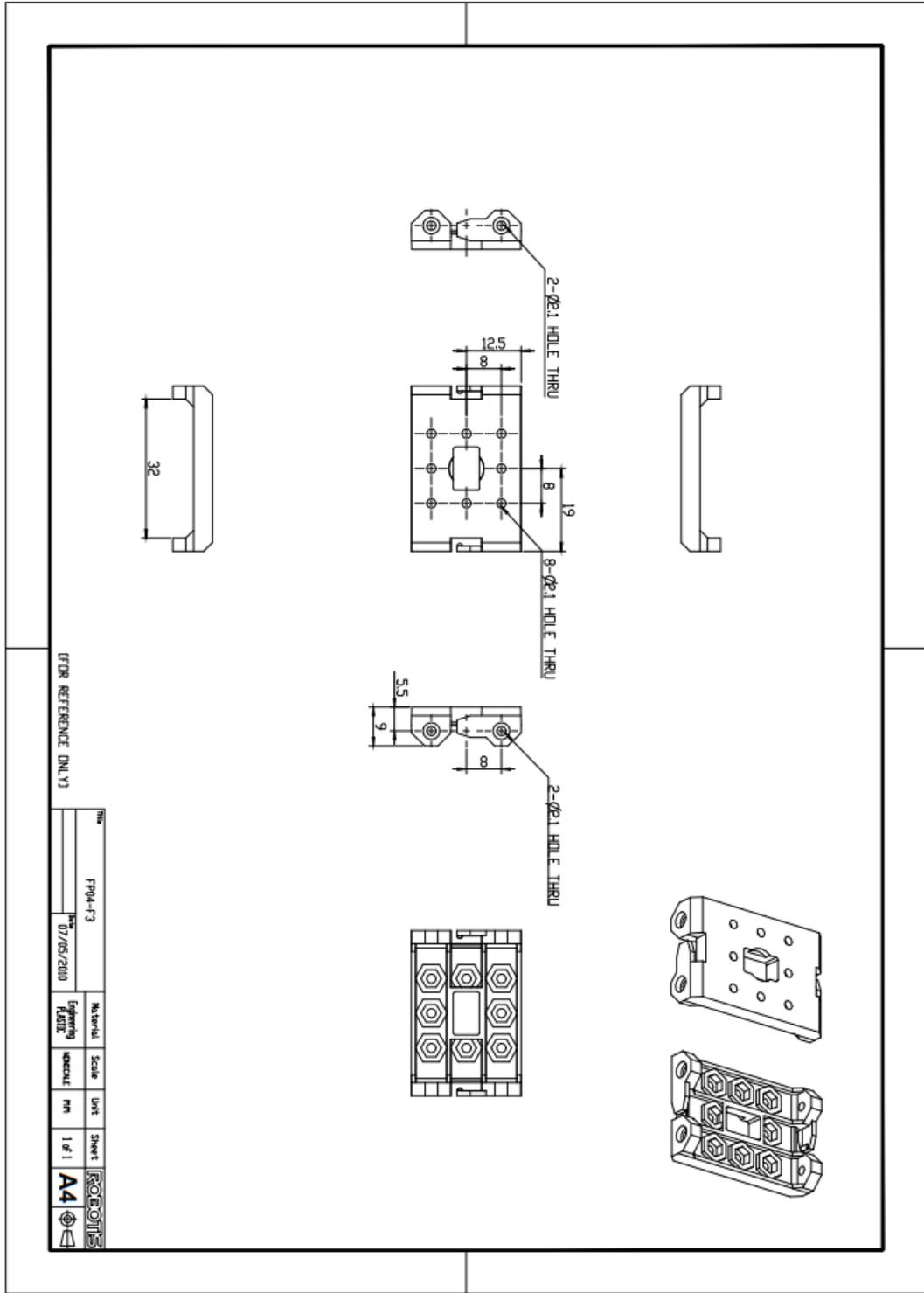


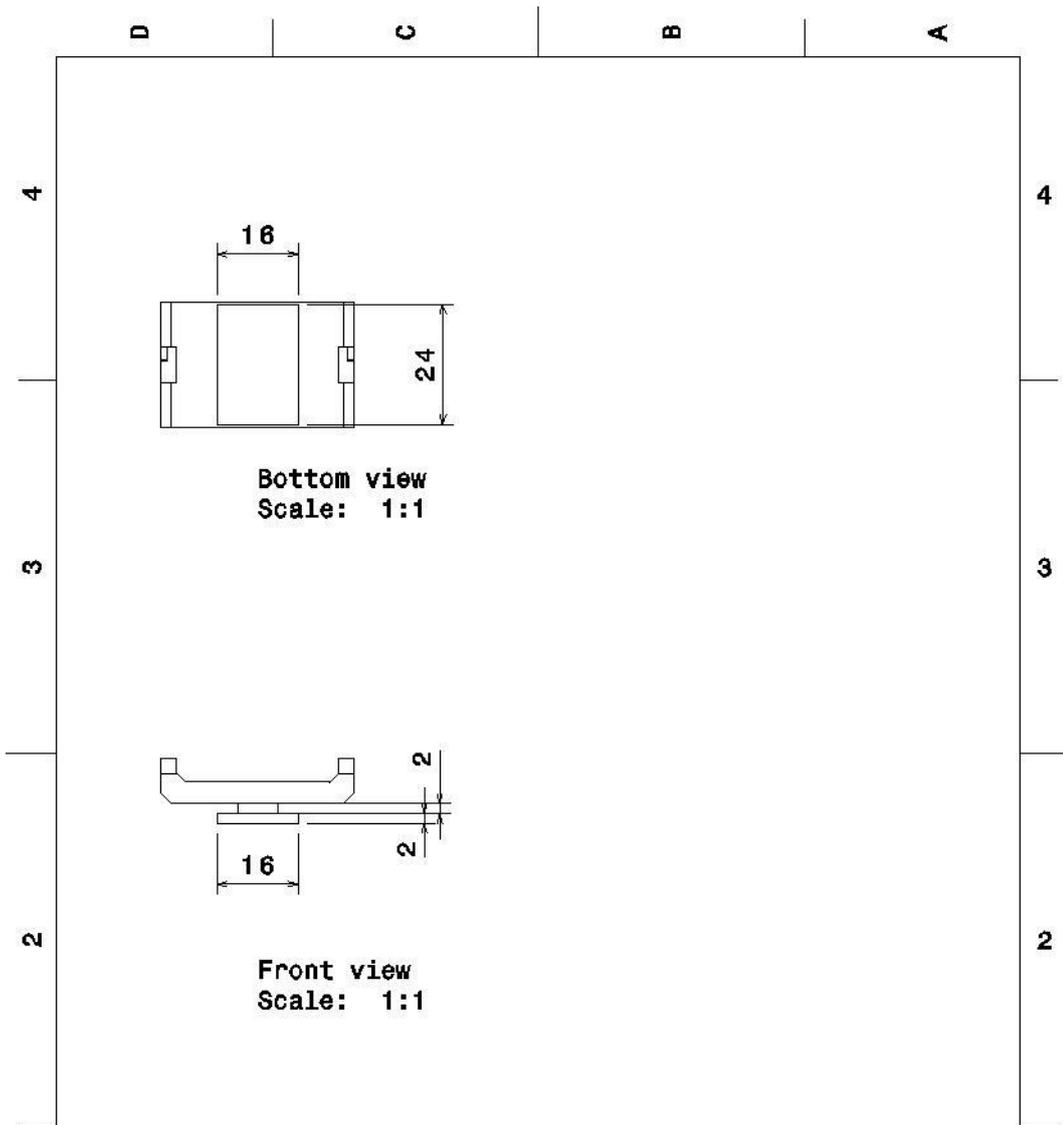
Pieza 3:



Pieza 4:

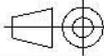
Se utilizan dos planos para dicha pieza al ser una unión entre dos piezas:



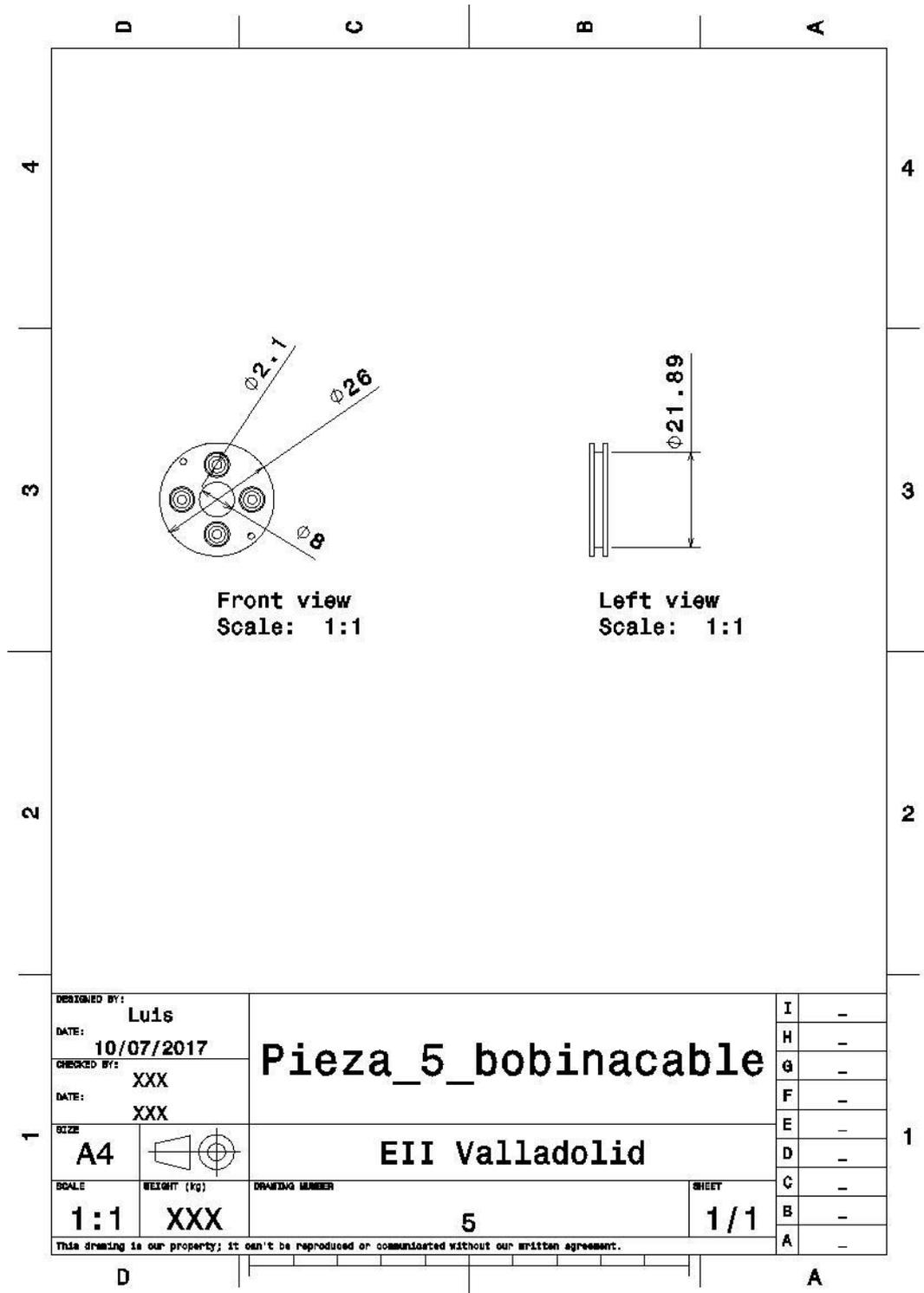


Bottom view
Scale: 1:1

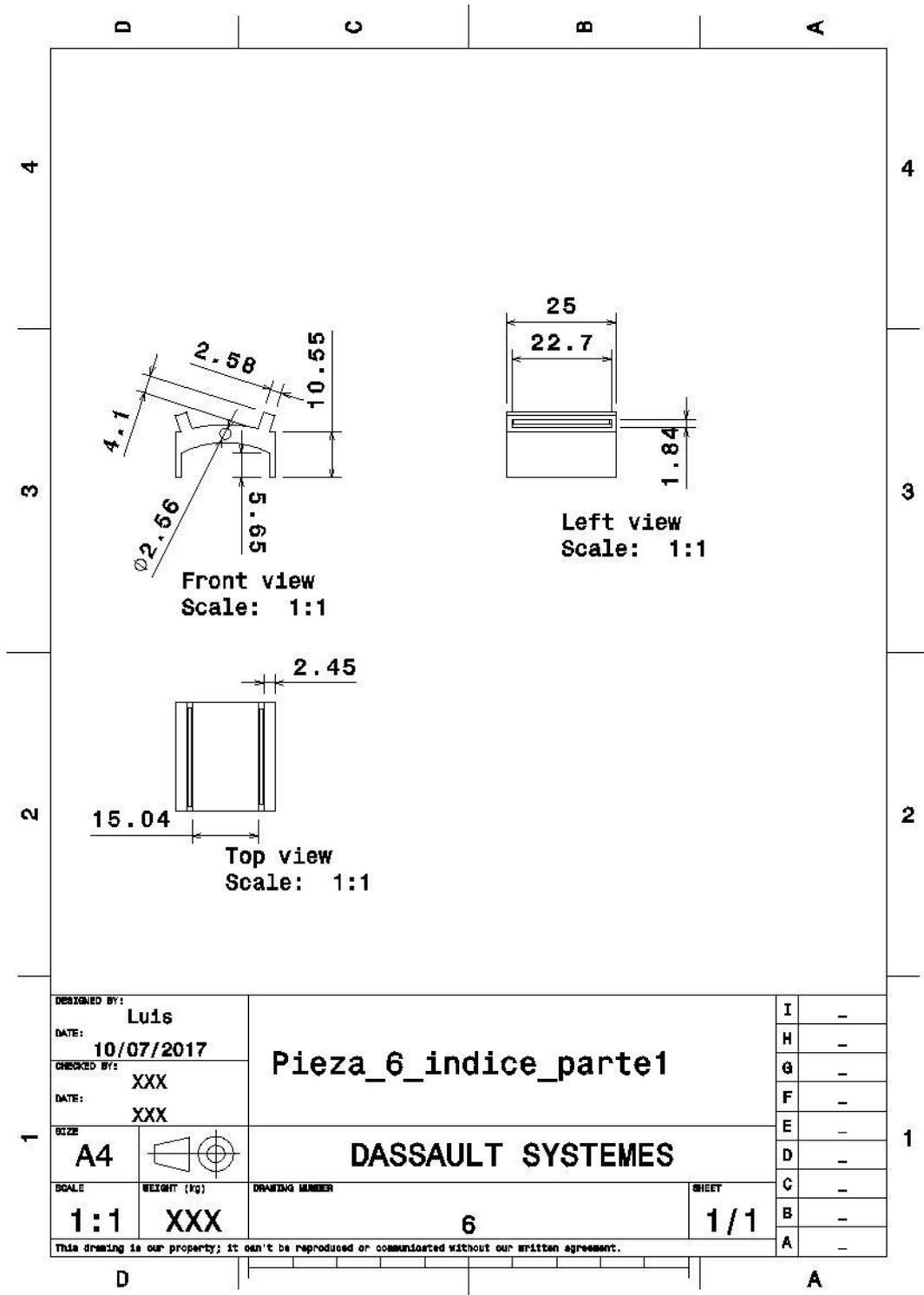
Front view
Scale: 1:1

DESIGNED BY: Luis	<h1>Pieza_4_soporte</h1>		I	-	
DATE: 10/07/2017			H	-	
CHECKED BY: XXX			G	-	
DATE: XXX			F	-	
SIZE: A4	<h2>EII Valladolid</h2>		E	-	
			D	-	
SCALE: 1:1	WEIGHT (kg): XXX	DRAWING NUMBER: 4	C	-	
		SHEET: 1/1	B	-	
This drawing is our property; it can't be reproduced or communicated without our written agreement.				A	-

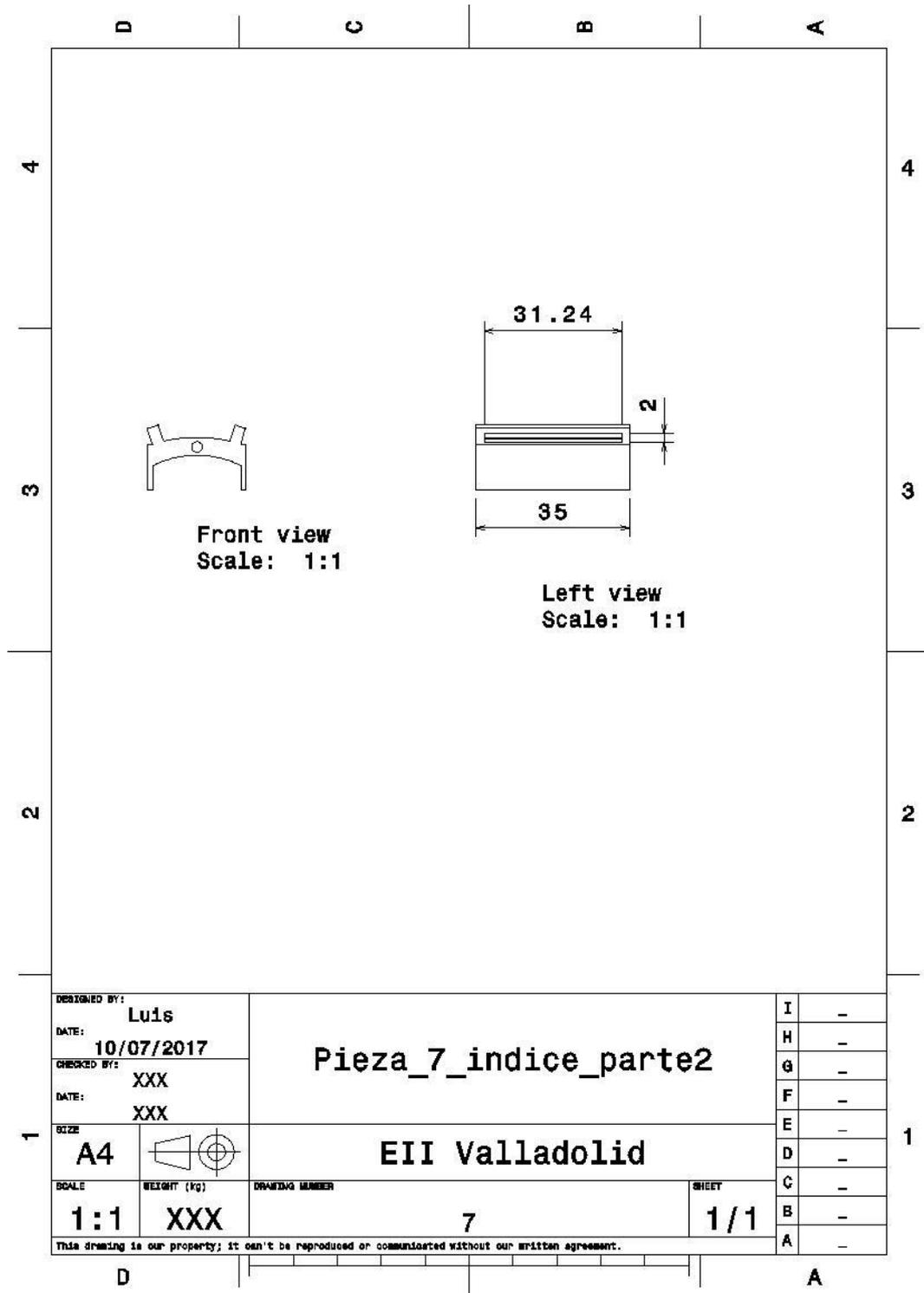
Pieza 5:



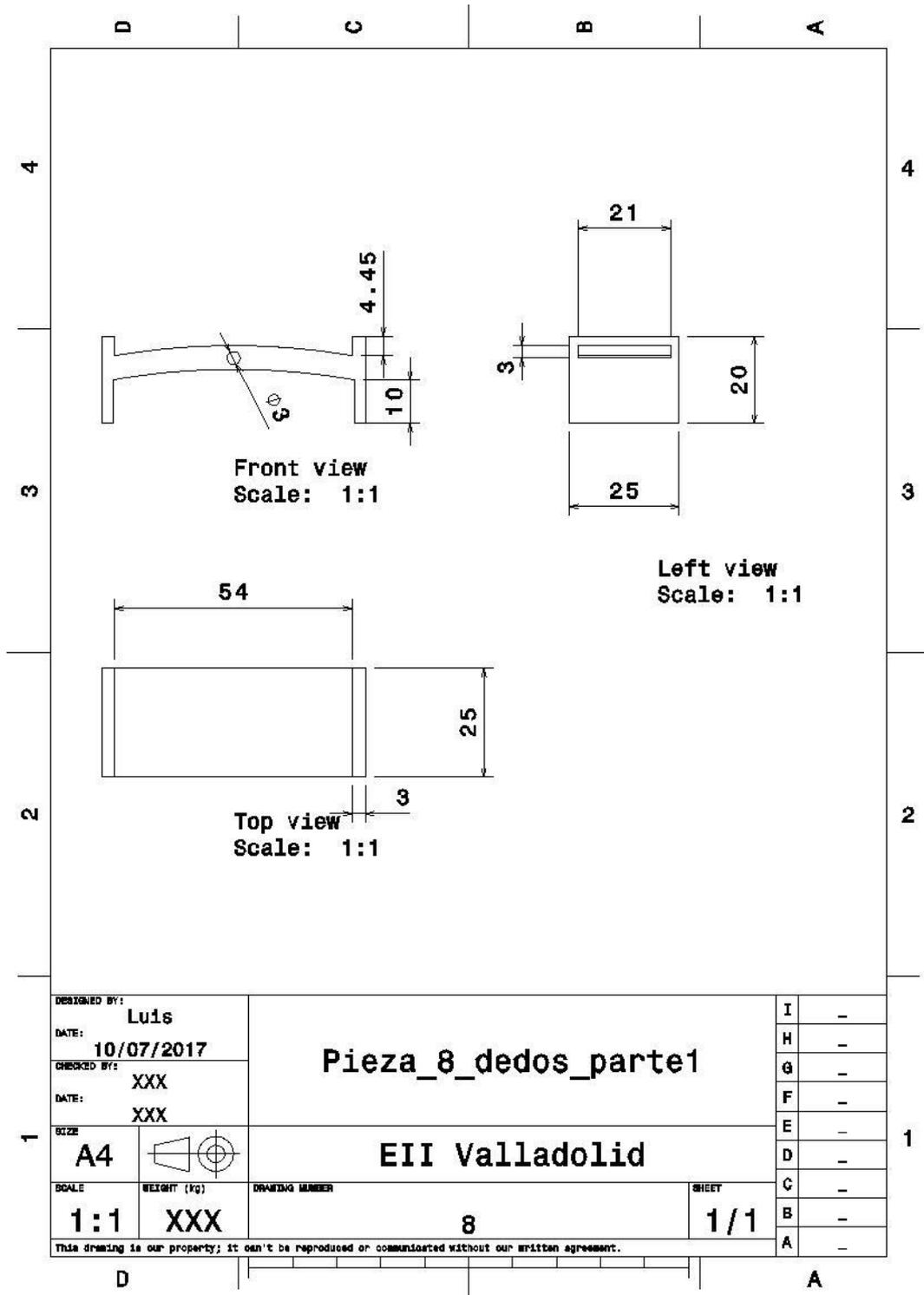
Pieza 6:



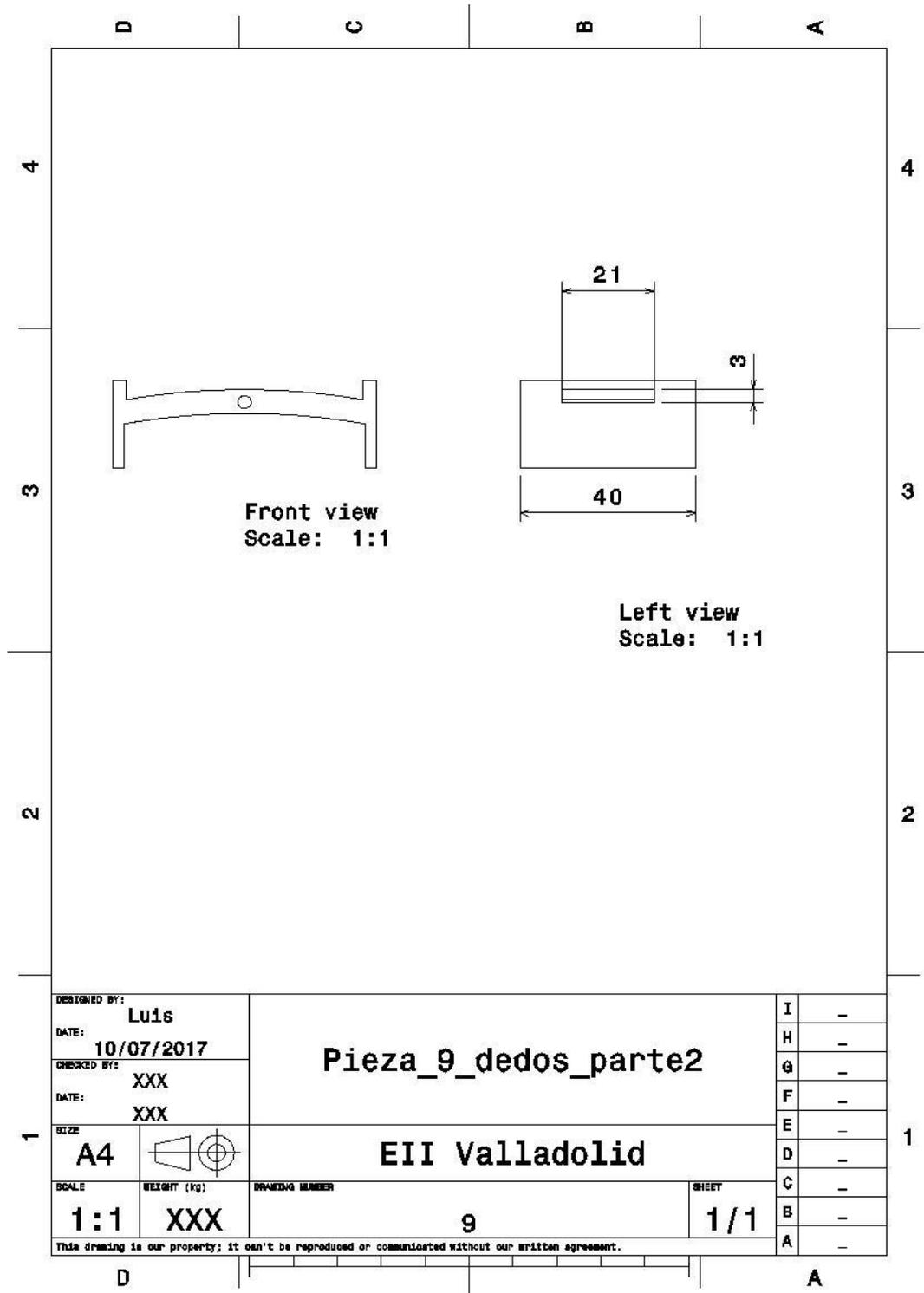
Pieza 7:



Pieza 8:



Pieza 9:



1. Dynamixel AX-12

1-1. Overview and Characteristics of AX-12

Dynamixel AX-12	The Dynamixel series robot actuator is a smart, modular actuator that incorporates a gear reducer, a precision DC motor and a control circuitry with networking functionality, all in a single package. Despite its compact size, it can produce high torque and is made with high quality materials to provide the necessary strength and structural resilience to withstand large external forces. It also has the ability to detect and act upon internal conditions such as changes in internal temperature or supply voltage. The Dynamixel series robot actuator has many advantages over similar products.
Precision Control	Position and speed can be controlled with a resolution of 1024 steps.
Compliance Driving	The degree of compliance can be adjusted and specified in controlling position.
Feedback	Feedback for angular position, angular velocity, and load torque are available.
Alarm System	The Dynamixel series robot actuator can alert the user when parameters deviate from user defined ranges (e.g. internal temperature, torque, voltage, etc) and can also handle the problem automatically (e.g. torque off)
Communication	Wiring is easy with daisy chain connection, and it support communication speeds up to 1M BPS.
Distributed Control	Position, velocity, compliance, and torque can be set with a single command packet, thus enabling the main processor to control many Dynamixel units even with very few resources.
Engineering Plastic	The main body of the unit is made with high quality engineering plastic which enables it to handle high torque loads.
Axis Bearing	A bearing is used at the final axis to ensure no efficiency degradation with high external loads.
Status LED	The LED can indicate the error status to the user.
Frames	A hinge frame and a side mount frame are included.

1-2. Main Specifications

	AX-12	
Weight (g)	55	
Gear Reduction Ratio	1/254	
Input Voltage (V)	at 7V	at 10V
Final Max Holding Torque(kgf.cm)	12	16.5
Sec/60degree	0.269	0.196

Resolution	0.35°
Operating Angle	300°, Endless Turn
Voltage	7V~10V (Recommended voltage: 9.6V)
Max. Current	900mA
Operate Temperature	-5℃ ~ +85℃
Command Signal	Digital Packet
Protocol Type	Half duplex Asynchronous Serial Communication (8bit,1stop,No Parity)
Link (Physical)	TTL Level Multi Drop (daisy chain type Connector)
ID	254 ID (0~253)
Communication Speed	7343bps ~ 1 Mbps
Feedback	Position, Temperature, Load, Input Voltage, etc.
Material	Engineering Plastic