



# Manual de Uso del Guante de Rehabilitación

---

MATERIA

**INTEGRAORA**

PRESENTA:

**AYALA MEJIA ALDO GUSTAVO**

**CORONEL LIMIAS HECTOR**

**BARRETO OCAMPO SANTIAGO**

**OJEDA HERNÁNDEZ MARIO ALBERTO**

**MEDINA SALMERÓN MOISES JACOB**

**ORTEGA CABRERA EDGAR**

**RIVERA PAZ RAMIRO**

**GUTIERREZ ARAUJO FRANCISCO JAVIER**

ASESOR:

**DRA. NORMA ALEJANDRA LEDESMA URIBE**



## Descripción General

Este prototipo de guante tiene como objetivo monitorear la motricidad de la mano del paciente para apoyar procesos de rehabilitación física. Recoge datos como el ángulo de flexión de los dedos y la fuerza ejercida durante los ejercicios. Los datos se muestran en tiempo real en una interfaz de consola y se almacenan en Firebase para el seguimiento a través de una aplicación móvil o web.



## Componentes del Sistema

**Hardware**

- 1 Arduino UNO
- 4 Flexisensores:
  - 2 de 7 cm para dedos índice y medio (voltaje entre 3.00V y 3.68V)
  - 2 de 2.5 cm para dedos anular y meñique (voltaje entre 1.60V y 2.33V)
- 1 Servo motor
- Cableado y protoboard
- Guante base
- Conexión USB a PC

## Software

- Script en **Python** con visualización en consola usando Rich
- Código en **Arduino** para captura de datos y control del servo
- Base de datos en **Firebase Firestore**
- Plataforma web/móvil conectada a Firebase

## Funcionamiento General

1. El guante recoge el voltaje de los flexisensores.
2. El Arduino convierte este voltaje en ángulos y calcula la fuerza ejercida.
3. Los datos se envían por puerto serial a un script Python.
4. El script muestra la información en tiempo real y la almacena en Firebase.
5. El terapeuta y el paciente pueden acceder a esta información desde una app o página web.

## Instrucciones de Uso del Script Python

### 1. Requisitos

- Python 3.8+

- Librerías instaladas:

`pip install firebase-admin rich pyserial`

- Archivo de credenciales Firebase (``proyecto.json``)
- Arduino conectado al puerto ``COM5`` (ajustable en el código)

## 2. Ejecución

1. Abre una terminal.
2. Navega a la carpeta donde se encuentra tu script.
3. Ejecuta el script:

`python main.py`

4. Ingresa el ID del usuario (paciente).
5. Ingresa el número de sesión.

## 3. Interfaz de Usuario

- Visualización de los dedos (índice, medio, anular, meñique) con:
  - Ángulo (°)
  - Fuerza (N)
  - Fuerza del servo (N)
  - Velocidad del servo
- Comandos disponibles para controlar el servo:
  - L: Girar a la izquierda
  - R: Girar a la derecha
  - S: Detener servo



## Instrucciones para el Código Arduino

### 1. Conexión de Flexisensores

Dedo	Pin Arduino	Longitud
Índice	A0	7 cm
Medio	A1	7 cm
Anular	A2	2.5 cm
Meñique	A3	2.5 cm

## 2. Control del Servo

- Conectado al pin 9
- Comandos recibidos por puerto serial
- Velocidad y fuerza del servo impresas por consola

## 3. Lógica del Código

- Mide voltaje y calcula el ángulo de cada dedo
- Estima la fuerza ejercida
- Envía los datos por serial si hay cambios mayores a 1°
- Responde a comandos para mover el servo

## Almacenamiento en Firebase

 Base de datos: usuarios

 Estructura:

usuarios/

[USER\_ID]/

datos/ ← Sesión 1

Index/

Middle/

Ring/

Little/

datos2/ ← Sesión 2 (y así sucesivamente)

Datos por dedo:

- angle: Ángulo del dedo
- force: Fuerza ejercida por el dedo
- servoforce: Fuerza generada por el servo
- velocity: Velocidad del servo



## Visualización y Seguimiento

El sistema permite al terapeuta:

- Ver progreso por sesión
- Comparar valores entre sesiones
- Identificar mejoras o retrocesos

El paciente:

- Ve su evolución
- Se motiva al ver resultados

Ambos pueden acceder a través de la plataforma web o app móvil conectada a Firebase.

## **Mantenimiento y Ajustes**

- 🎬 Ajustar los valores de  $V_{\min}$  y  $V_{\max}$  en Arduino si se cambian flexisensores.
- 🎬 Verificar puerto COM en `arduino = serial.Serial('COM5', ...)` si cambia.
- 🎬 Revisar la conexión de Firebase si hay errores al subir dato

## **Advertencias**

- 🎬 No usar el guante durante más de 30 minutos seguidos sin descanso.
- 🎬 Verificar que no haya cables sueltos antes de iniciar una sesión.
- 🎬 Asegurar que el servo no tenga objetos que puedan obstruir su movimiento.