

# UNIVERSIDAD NACIONAL DE ASUNCIÓN

## Facultad de Ingeniería Ingeniería Mecatrónica



### Mano Robótica

Cristian Acero Caicedo

Camila Caceres Kowalewski

Leila Gauna Nazer

Nicolas Silva Carmona

Sol Fernández Vargas

San Lorenzo - Paraguay

2025

---

## Índice general

<b>1. Capítulo 1. Introducción</b>	<b>4</b>
1.1. Repositorio . . . . .	4
1.2. Introducción . . . . .	4
1.3. Objetivo General . . . . .	5
1.4. Objetivos Específicos . . . . .	5
1.5. Problemática . . . . .	6
1.6. Justificación . . . . .	7
<b>2. Capítulo 2. Metodología del Proyecto</b>	<b>8</b>
2.1. Estado del Arte . . . . .	8
2.2. Cronograma . . . . .	9
2.3. Plan de Actividades . . . . .	9
2.4. Diagrama de base de datos . . . . .	10
2.5. Flujograma . . . . .	11
<b>3. Desarrollo de la Interfaz Gráfica para la Mano Robótica</b>	<b>12</b>
3.1. Funcionalidad general de la GUI . . . . .	12
3.2. Componentes implementados . . . . .	13
3.3. Descripción de componentes . . . . .	15
3.4. Diagrama eléctrico final . . . . .	17
3.5. Formato del Documento . . . . .	17
3.6. Conexión de Sensores . . . . .	18
3.7. Conexión con Base de Datos . . . . .	20
3.8. Presupuesto . . . . .	23
3.9. Horas Hombre . . . . .	24
3.10. Cambios y Ajustes Realizados . . . . .	24
3.11. Manual de Usuario . . . . .	26

3.12. Esquema general del proyecto . . . . .	29
--	----

---

**MANO ROBÓTICA**

---

**Índice de figuras**

2.1. Diagrama entidad–relación (ERD) simplificado para autenticación.	11
2.2. Interfaz gráfica de usuario para el control de la mano robótica. . .	11
3.1. Diagrama Electrico . . . . .	17
3.2. Esquema general del sistema del proyecto. . . . .	29
3.3. Diagrama de programa del sistema de mano robótica. . . . .	31
3.4. Mano robótica. . . . .	32

## **CAPÍTULO 1**

### **Introducción**

#### **1.1. Repositorio**

Repositorio de la mano robótica: [https://github.com/nazerdev/Robotic\\_hand](https://github.com/nazerdev/Robotic_hand)

#### **1.2. Introducción**

El presente proyecto consiste en el diseño e implementación de un prototipo de mano robótica controlada a través de un guante instrumentado con sensores flex y una interfaz gráfica de usuario (GUI). El sistema combina tres áreas principales: el diseño mecánico de la mano, la electrónica de control basada en un microcontrolador ESP32, y el desarrollo de la interfaz, que permite la interacción entre el usuario y el prototipo.

La GUI incorpora un módulo de inicio de sesión que garantiza un acceso seguro, además de controles manuales mediante sliders y botones para mover individualmente los dedos, guardar posiciones, reproducir secuencias, detener acciones y resetear el sistema. Asimismo, incluye la opción de activar el control automático utilizando directamente los datos de los sensores flex del guante.

Este enfoque integral permite evaluar la relación entre las señales captadas por los sensores y los movimientos de los servomotores, asegurando precisión mediante funciones de calibración. El proyecto se orienta tanto a fines académicos como a la exploración de aplicaciones futuras en ámbitos médicos, educati-

vos e industriales.

### **1.3. Objetivo General**

Diseñar e implementar una interfaz gráfica de usuario con sistema de autenticación que permita controlar y supervisar los movimientos de la mano robótica de manera manual, así como habilitar su funcionamiento automático mediante sensores flex, garantizando seguridad, usabilidad y confiabilidad en el proceso.

### **1.4. Objetivos Específicos**

- Diseñar e implementar un módulo de inicio de sesión (login) que asegure la autenticidad del usuario antes de acceder a la interfaz de control.
- Desarrollar controles deslizantes (sliders) que representen gráficamente el ángulo de flexión de cada dedo de la mano robótica.
- Incorporar botones de acción destinados a guardar posiciones, reproducir secuencias de movimientos, detener acciones en curso y resetear el sistema.
- Implementar un módulo de conmutación que permita alternar entre el control manual (a través de la interfaz) y el control automático (a través de los sensores flex).
- Configurar y optimizar la comunicación serial con el microcontrolador ESP32, garantizando la transmisión en tiempo real de datos entre la interfaz y la mano robótica.
- Validar experimentalmente la funcionalidad de la interfaz mediante pruebas con usuarios, comparando el desempeño entre el control manual y el control sensorizado.

## **1.5. Problemática**

El diseño de manos robóticas representa un área de gran interés en la ingeniería debido a la complejidad que implica replicar los movimientos naturales de la mano humana mediante sistemas electromecánicos. La coordinación de múltiples grados de libertad requiere de sensores, mecanismos y estrategias de control que trabajen de forma precisa y confiable.

En el ámbito académico, la mayoría de los prototipos existentes se limitan a demostraciones básicas de movimiento, careciendo de herramientas que permitan al usuario visualizar, ajustar y supervisar en tiempo real el funcionamiento del sistema. Esto dificulta tanto la validación de los sensores como la evaluación del desempeño de los servomotores.

A esta limitación se suma que los sensores flex utilizados para detectar la flexión de los dedos presentan un comportamiento no lineal, lo que ocasiona errores entre la señal captada y el movimiento real. Sin un mecanismo adecuado de calibración y supervisión, estos errores reducen la exactitud y dificultan el estudio del sistema.

En consecuencia, surge la necesidad de desarrollar un prototipo de mano robótica que incorpore no solo el diseño mecánico y electrónico, sino también una interfaz gráfica de usuario (GUI) que integre:

- Un módulo de inicio de sesión, para garantizar seguridad y control de accesos.
- Un sistema de control manual, con sliders y botones que permitan manipular los dedos de manera individual y global.
- La opción de control automático, directamente desde los sensores flex del guante.

De esta forma, el proyecto busca superar las limitaciones de control y supervisión

---

## MANO ROBÓTICA

---

de prototipos similares, ofreciendo un sistema más preciso, accesible y versátil para fines académicos, experimentales y tecnológicos.

### 1.6. Justificación

El presente proyecto abarca el diseño e implementación de un prototipo funcional de mano robótica, integrando tres componentes principales:

- **Sistema mecánico:** diseño y fabricación de la estructura de la mano utilizando impresión 3D, asegurando compatibilidad con los servomotores seleccionados.
- **Sistema electrónico de control:** implementación de un microcontrolador ESP32 encargado de procesar las señales provenientes de los sensores flex y accionar los servomotores de manera coordinada.
- **Interfaz gráfica de usuario (GUI):** desarrollo de una aplicación con módulo de inicio de sesión, control manual mediante sliders y botones, y la opción de habilitar el control automático a través de los sensores flex del guante.

Dentro de este alcance se incluyen la programación de funciones básicas de calibración, la validación experimental del prototipo y la documentación técnica del proceso de desarrollo.

De este modo, el alcance queda delimitado al ámbito académico y experimental, orientado a la validación de conceptos de control robótico, interacción humano-máquina y desarrollo de prototipos tecnológicos.



## **CAPÍTULO 2**

### **Metodología del Proyecto**

#### **2.1. Estado del Arte**

El desarrollo de manos y brazos robóticos ha sido objeto de investigación en las últimas décadas debido a la complejidad que implica replicar los movimientos naturales de la mano humana. Una de las aproximaciones más utilizadas es el empleo de guantes sensorizados con *flex sensors*, que permiten capturar la flexión de los dedos y transmitir esta información a un sistema electrónico encargado de accionar servomotores.

En [1] se presenta un sistema de control para un brazo robótico operado mediante gestos manuales. En este trabajo, un guante equipado con sensores flex transmite las señales a una tarjeta de control que, posteriormente, mueve un brazo robótico, demostrando la viabilidad del control inalámbrico basado en gestos.

De manera similar, en [2] se desarrolla un prototipo de mano robótica orientado a personas con discapacidad motora, empleando sensores flex y controladores basados en Arduino. Este proyecto evidencia la importancia de integrar mecánica, electrónica y control para lograr un movimiento coordinado.

Por su parte, la compañía Shadow Robot Company ha diseñado el *Shadow Teleoperation System*, un sistema de teleoperación que utiliza sensores ópticos y magnéticos para replicar con alta precisión los movimientos de la mano humana en un entorno robótico [3]. Aunque de mayor complejidad tecnológica, este ejemplo confirma el potencial del control remoto de manipuladores robóticos mediante

---

**MANO ROBÓTICA**

---

guantes sensorizados.

Finalmente, la literatura reciente destaca también el uso de interfaces gráficas de usuario (GUI) como complemento al control físico. Estas permiten monitorear señales en tiempo real, corregir errores de no linealidad y ofrecer una experiencia de usuario más intuitiva, aspecto que se busca integrar en el presente proyecto.

## **2.2. Cronograma**

El cronograma de trabajo se planifica en un horizonte de siete semanas, distribuidas de la siguiente manera:

<b>Semana</b>	<b>Actividad</b>
1	Definición de requerimientos funcionales y diseño preliminar de la interfaz.
2	Implementación del módulo de login y pruebas iniciales de acceso.
3	Programación de sliders para el control manual de los dedos.
4	Desarrollo de botones de acción: guardar, reproducir, detener y resetear.
5	Incorporación del botón para control automático mediante sensores flex y pruebas de comunicación con ESP32.
6	Integración de todos los módulos en una única interfaz y pruebas iniciales.
7	Validación con usuarios, pruebas comparativas entre control manual y automático, correcciones y documentación final.

## **2.3. Plan de Actividades**

El plan de actividades contempla las siguientes etapas:

1. **Análisis de requerimientos:** definición de las funciones principales de la GUI y de los parámetros de control de los dedos.

---

## MANO ROBÓTICA

---

2. **Diseño visual:** elaboración de la estructura de la interfaz, disposición de sliders, botones e indicadores de estado.
3. **Implementación del login:** programación del sistema de autenticación para acceso seguro.
4. **Desarrollo del control manual:** integración de sliders y botones con el microcontrolador ESP32.
5. **Incorporación del control automático:** habilitación del modo sensorizado mediante los flex sensors.
6. **Pruebas de validación:** ejecución de ensayos para verificar precisión, usabilidad y correspondencia de movimientos.
7. **Documentación:** redacción del manual de usuario y de la memoria técnica del proyecto.

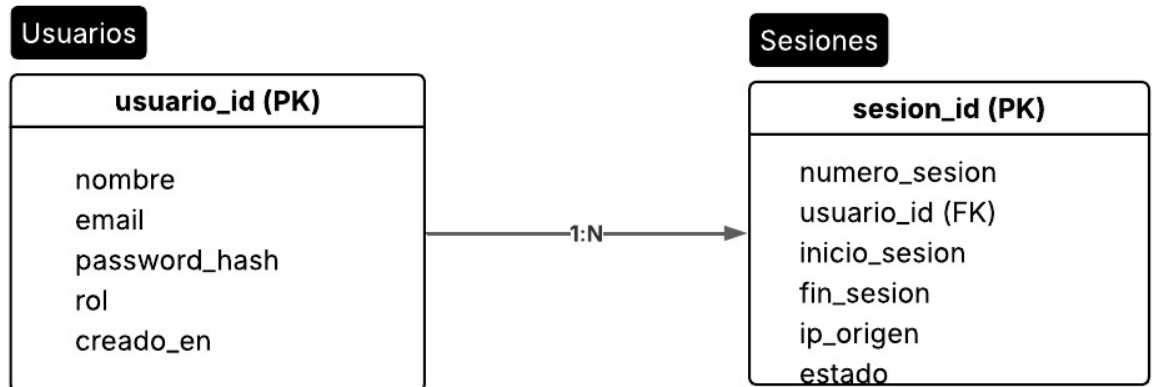
### 2.4. Diagrama de base de datos

Como se muestra en el diagrama entidad–relación [4], la base de datos diseñada para la interfaz de la mano robótica se estructura en torno a tres componentes principales: la tabla de **Usuarios**, que almacena la información básica de quienes acceden al sistema; la tabla de **Sesiones**, que registra cada ingreso y salida de los usuarios; y la tabla de **IntentosLogin**, destinada a conservar los intentos fallidos de autenticación. De esta manera, se establece un modelo que garantiza seguridad, trazabilidad y un control organizado de los accesos al sistema.

## MANO ROBÓTICA

PK: clave primaria

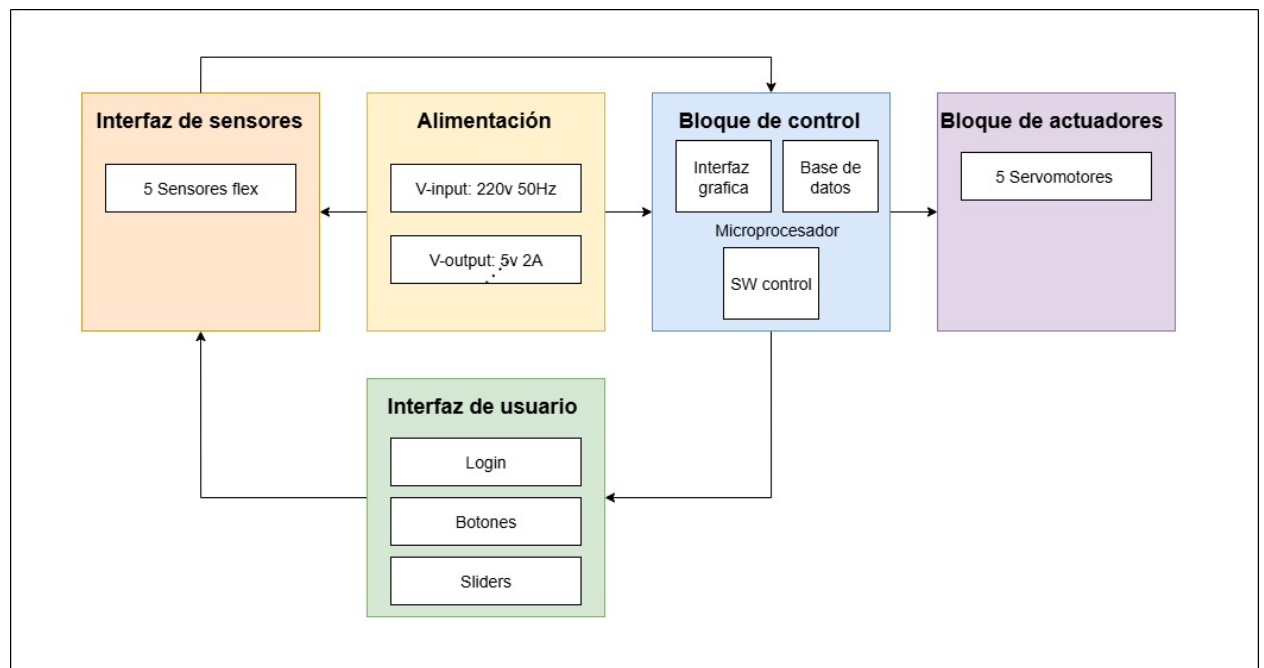
FK: clave foranea



**Figura 2.1:** Diagrama entidad-relación (ERD) simplificado para autenticación.

## 2.5. Flujograma

En el flujograma [5] se ve la Interfaz grafica de usuario para el control de la mano robotica



**Figura 2.2:** Interfaz gráfica de usuario para el control de la mano robótica.

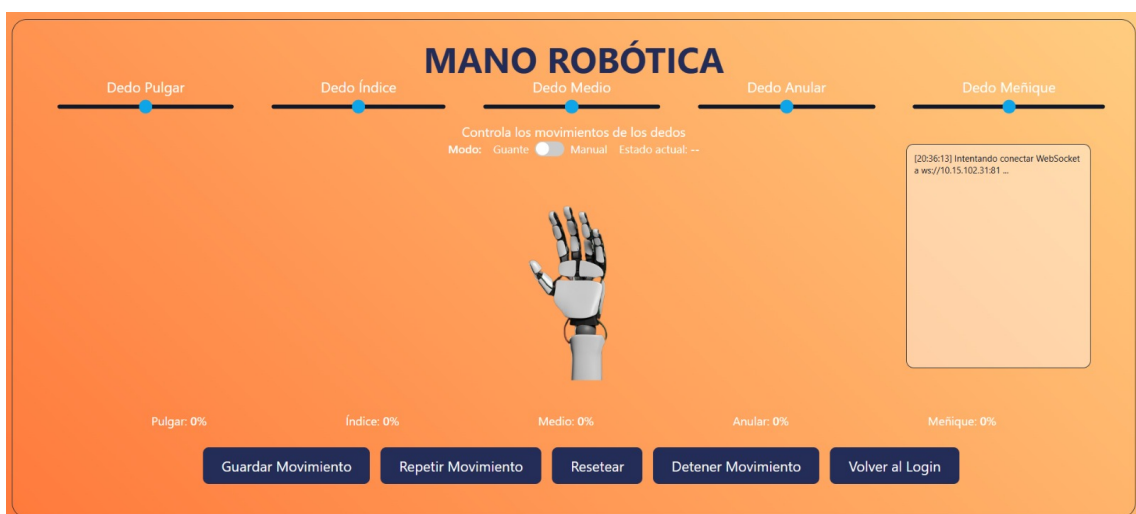
## CAPÍTULO 3

### Desarrollo de la Interfaz Gráfica para la Mano Robótica

En este capítulo se detalla el desarrollo de la interfaz gráfica de usuario (GUI) diseñada para el control de la mano robótica, siguiendo los requerimientos funcionales establecidos en la rúbrica de evaluación del segundo parcial. El trabajo integra programación, diseño de interfaz, comunicación con hardware, documentación técnica y elaboración de diagramas eléctricos.

#### 3.1. Funcionalidad general de la GUI

La interfaz gráfica fue desarrollada en Python utilizando Tkinter como toolkit principal. El objetivo de la interfaz es permitir el control individual de los cinco dedos de la mano robótica mediante deslizadores (sliders) que representan el ángulo de apertura entre  $0^\circ$  y  $180^\circ$ . Cada cambio en los sliders actualiza en tiempo real el valor visualizado, facilitando la operación del usuario.

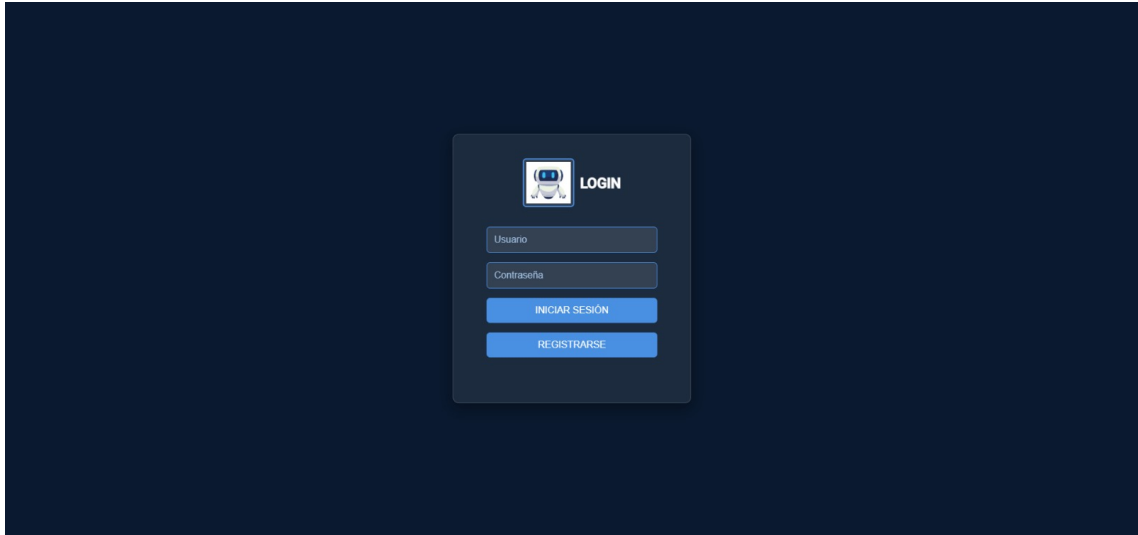


## MANO ROBÓTICA

---

La interfaz incluye:

- Inicio de sesión con campo de cédula y contraseña.



- Control independiente de los dedos: Pulgar, Índice, Medio, Anular y Meñique.
- Visualización gráfica de la mano robótica mediante una imagen de referencia.
- Botones de acción: Guardar movimiento, Repetir movimiento, Resetear y Detener.
- Compatibilidad con conexión WiFi al microcontrolador.

El diseño se centró en un estilo moderno, con colores oscuros y elementos minimalistas, permitiendo una experiencia de usuario clara e intuitiva.

### 3.2. Componentes implementados

Los componentes utilizados en el desarrollo del sistema completo (mano robótica + interfaz) son los siguientes:

---

## MANO ROBÓTICA

---

- **Computadora personal:** ejecuta la interfaz gráfica desarrollada en Python (Tkinter), desde donde el usuario visualiza y controla la mano robótica.
- **Microcontrolador ESP32-S3-WROOM-1:** actúa como núcleo de control del sistema. Recibe los datos de los sensores flex a través del ADC externo, procesa la información y genera las señales PWM para los servomotores.
- **ADC externo ADS1115:** convertidor analógico–digital de 16 bits, encargado de muestrear las señales analógicas provenientes de los sensores flex resistivos y enviarlas al ESP32 mediante el bus I<sup>2</sup>C.
- **Sensores flex resistivos 2,2” (x5):** colocados en un guante, uno por cada dedo. Cambian su resistencia en función de la curvatura, permitiendo medir el grado de flexión de cada dedo de la mano humana.
- **Servomotores MG996R (x5):** actuadores que generan el movimiento de los dedos de la mano robótica. Cada servo corresponde a un dedo y recibe una señal PWM desde el ESP32.
- **Nivel lógico NBXB0108:** conversor de nivel lógico bidireccional que adapta las señales de 3,3 V del ESP32 a 5 V, necesarias para el correcto comando de los servomotores.
- **Regulador LDO LM1117-3.3:** regulador lineal de baja caída que provee 3,3 V estables para alimentar el ESP32 y la lógica de 3,3 V a partir de una alimentación de 5 V.
- **Fuente de alimentación de 5 V:** utilizada para la alimentación de los servomotores y del resto de la electrónica de potencia del sistema.
- **Guante instrumentado:** soporte físico donde se montan los sensores flex. Permite capturar de manera natural los gestos de la mano del usuario.

- **Interfaz gráfica en Python:** software que permite ajustar parámetros, visualizar estados y, en futuras etapas, supervisar en tiempo real la lectura de sensores y la posición de los dedos.

### 3.3. Descripción de componentes

A continuación se describen brevemente los componentes principales que intervienen en la interacción entre la interfaz gráfica, los sensores y la mano robótica:

- **Sensores flex resistivos:** son tiras que varían su resistencia eléctrica al doblarse. Cada sensor se coloca sobre un dedo del guante, de modo que la flexión del dedo produce una variación medible de tensión en el divisor de voltaje. Esta señal es leída por el ADC externo.
- **ADC ADS1115:** es el encargado de convertir las señales analógicas de los sensores flex en datos digitales de alta resolución (16 bits). Se comunica con el ESP32 mediante el bus I<sup>2</sup>C, permitiendo la lectura simultánea de varios canales.
- **Microcontrolador ESP32-S3:** recibe los valores digitalizados de los sensores, aplica las funciones de calibración y mapeo de flexión, y genera las señales PWM para controlar los servomotores. Además, gestiona la comunicación inalámbrica con la interfaz gráfica mediante WiFi.
- **Servomotores MG996R:** convierten las señales PWM en movimiento mecánico de los dedos de la mano robótica. El ángulo de giro del servo se asocia al grado de flexión del dedo captado por el sensor correspondiente.
- **Nivel lógico NBXB0108:** adapta los niveles de tensión entre la lógica de 3,3 V del ESP32 y los 5 V requeridos por los servos, garantizando compatibilidad eléctrica y protección del microcontrolador.



---

**MANO ROBÓTICA**

---

- **Regulador LDO LM1117-3.3:** suministra una tensión estable de 3,3 V para el microcontrolador y el resto de la lógica de baja tensión, a partir de la línea principal de 5 V.
- **Guante con sensores:** permite que el usuario realice movimientos naturales, mientras el sistema mide la flexión real de cada dedo. Es el elemento de entrada física del sistema.
- **Interfaz gráfica de usuario:** representa la capa de interacción visual. A través de ella se pueden observar estados, ajustar parámetros y, en etapas posteriores, comparar el movimiento captado por los sensores con el movimiento ejecutado por la mano robótica.

### 3.4. Diagrama eléctrico final

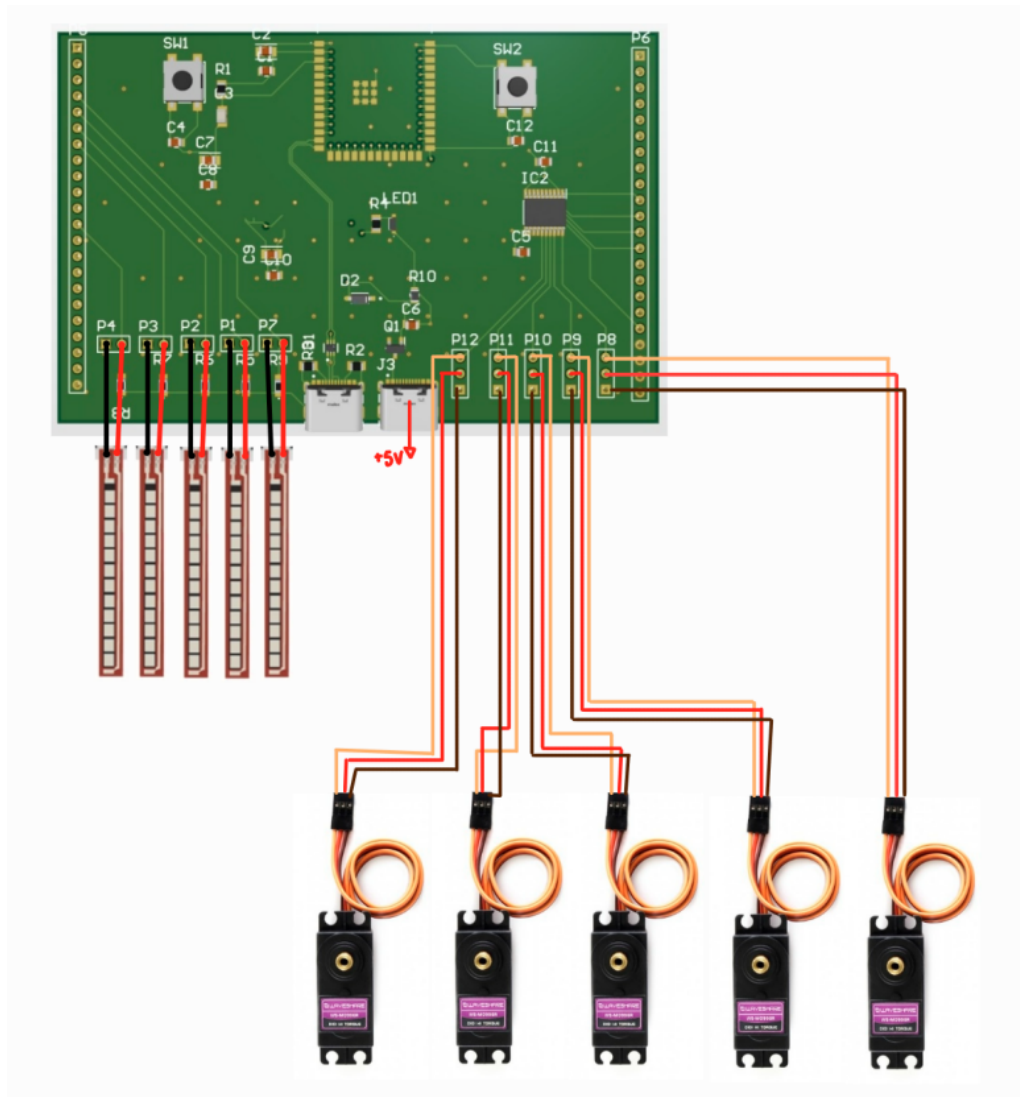


Figura 3.1: Diagrama Electrico

### 3.5. Formato del Documento

El presente informe fue desarrollado siguiendo las normas de presentación establecidas para trabajos académicos de la Facultad de Ingeniería de la Universidad Nacional de Asunción. Para ello se empleó el lenguaje de composición tipográfica  $\text{\LaTeX}$ , asegurando una estructura clara, homogénea y profesional en todo el documento.

El formato del informe incluye:

- Uso de tipografía homogénea y estilo académico.
- Numeración consistente de capítulos, secciones, tablas y figuras.
- Inclusión de leyendas descriptivas en cada figura y tabla.
- Referencias cruzadas internas mediante comandos de  $\text{\LaTeX}$ .
- Márgenes y espaciados uniformes de acuerdo al estilo institucional.
- Organización en capítulos: introducción, metodología, desarrollo del sistema, diagramas, presupuesto y manual de usuario.

El uso de  $\text{\LaTeX}$  permitió mantener la coherencia visual, facilitar la inclusión de imágenes técnicas, diagramas eléctricos y tablas, y garantizar que el documento cumpla con los estándares formales solicitados en la rúbrica de evaluación.

### **3.6. Conexión de Sensores**

Los sensores flex resistivos utilizados en el guante instrumentado constituyen la entrada principal del sistema. Cada uno de los cinco sensores corresponde a un dedo de la mano del usuario y permite medir su grado de flexión mediante la variación de resistencia cuando el sensor se curva.

Para obtener una lectura precisa y estable, cada sensor flex se conecta en configuración de divisor de voltaje, donde el cambio de resistencia produce una variación proporcional de tensión que es posteriormente digitalizada por el ADC ADS1115.

#### **Esquema de conexión**

La conexión general de cada sensor se realizó siguiendo los siguientes puntos:

---

## MANO ROBÓTICA

---

- Cada sensor flex se conectó en serie con una resistencia fija de precisión, formando un divisor de voltaje.
- El nodo central del divisor se conectó a uno de los canales analógicos del ADC ADS1115.
- El ADC se comunicó con el microcontrolador ESP32-S3 mediante el bus I2C (pines SDA y SCL).
- La alimentación de los sensores y del ADC provino de una línea estable de 5V.
- Se compartió una referencia de tierra común entre el guante sensorizado, el ADC y el ESP32, garantizando lecturas consistentes.

### Canales utilizados en el ADC

Cada uno de los cinco sensores flex fue conectado a los siguientes canales:

- Sensor del pulgar: **AIN0**
- Sensor del índice: **AIN1**
- Sensor del medio: **AIN2**
- Sensor del anular: **AIN3**
- Sensor del meñique: **AIN0** (segundo módulo ADS1115 en caso de expansión)

### Procesamiento en el microcontrolador

Las señales digitalizadas fueron enviadas al ESP32-S3, donde se aplicaron:

- Filtrado simple para reducir ruido.

---

## MANO ROBÓTICA

---

- Calibración basada en los valores mínimo y máximo de cada usuario.
- Mapeo lineal del rango medido a un ángulo de servo entre 0° y 180°.

Este proceso permitió que los movimientos reales del usuario pudieran ser replicados con precisión por los servomotores de la mano robótica.

### 3.7. Conexión con Base de Datos

Para la gestión de usuarios, almacenamiento de movimientos y registro de eventos del sistema se empleó un backend desarrollado con el framework **Django**. Este entorno permitió implementar un sistema robusto, escalable y con autenticación integrada, facilitando la comunicación entre la interfaz gráfica y la base de datos.

#### 1. Arquitectura general

El sistema se organiza en dos partes principales:

- **Frontend (GUI):** interfaz gráfica utilizada por el usuario final.
- **Backend Django:** servidor encargado de procesar datos, autenticar usuarios y gestionar la base de datos.

Django actúa como intermediario entre la GUI y el almacenamiento permanente, permitiendo mantener la integridad de la información y centralizar la lógica de negocio.

#### 2. Base de datos utilizada

El proyecto utiliza una base de datos **SQLite** durante el desarrollo, debido a su simplicidad y compatibilidad nativa con Django. No obstante, la estructura fue diseñada para migrar fácilmente a **MySQL o PostgreSQL** en caso de ser necesario.

### 3. Modelos implementados

Siguiendo el ORM de Django, se crearon modelos que representan las entidades necesarias para el sistema:

- **Usuario:** almacena credenciales, roles y fechas de acceso.
- **Movimiento:** guarda las posiciones de cada dedo en formato estructurado.
- **LogEvento:** registra actividad de la interfaz, como movimientos ejecutados o errores de conexión.

Un ejemplo de modelo es el siguiente:

```
class Movimiento(models.Model):  
    pulgar = models.IntegerField()  
    indice = models.IntegerField()  
    medio = models.IntegerField()  
    anular = models.IntegerField()  
    menique = models.IntegerField()  
    fecha = models.DateTimeField(auto_now_add=True)  
    usuario = models.ForeignKey(User, on_delete=models.CASCADE)
```

### 4. API y comunicación con la interfaz

Django expone los datos mediante:

- **Vistas HTTP (views.py):** para guardar movimientos.
- **EndPoints JSON:** para enviar valores de los dedos.
- **Autenticación integrada:** para validar cédula y contraseña.

La interfaz gráfica se comunica con el backend enviando solicitudes HTTP o WebSocket, dependiendo del comando requerido.

## **5. Flujo de funcionamiento**

El funcionamiento general es el siguiente:

1. El usuario inicia sesión desde la interfaz.
2. Django valida las credenciales mediante su sistema de autenticación.
3. La GUI solicita o envía datos de movimientos al servidor.
4. Django guarda, actualiza o recupera la información desde la base de datos.
5. El ESP32 recibe los comandos procesados para ejecutar los movimientos.

## **6. Ventajas del uso de Django**

- Seguridad en el manejo de credenciales.
- ORM que evita errores de SQL manual.
- Facilidad para crear APIs con JSON.
- Migración sencilla entre distintos motores de base de datos.
- Registro automático de actividades.

El uso de Django permitió un sistema estructurado, mantenible y escalable, cumpliendo con los requerimientos de autenticación, almacenamiento y comunicación definidos en la rúbrica del proyecto.

**MANO ROBÓTICA**

### 3.8. Presupuesto

Descripción	Cantidad	Precio (USD)	Total (USD)
Thin Film Resistors - SMD 0805 5.1Kohm 0.1 % 10ppm AEC-Q200	10	0.336	3.36
Multilayer Ceramic Capacitors MLCC - SMD/SMT 0.1 $\mu$ F 50VDC 10 % 0805 X7R AEC-Q200	10	0.13	1.3
Translation - Voltage Levels Dual supply translating transceiver	2	0.99	1.98
Multiprotocol Modules SMD ESP32-S3, 8 MB SPI flash	2	3.2	6.4
LDO Voltage Regulators 800mA A926-LM1117MP-3.3NOPB	2	0.95	1.9
MOSFETs 5.0A 29mOhm 30V 2.5V gate capable	2	0.34	0.68
Single Color LEDs Blue	2	0.02	0.04
Schottky Diodes Vr/40V Io/1A T/R	2	0.23	0.46
USB Connectors Type-C SMT 16kt Type C Rec.	3	0.79	2.37
Tactile Switches Tact Switch Wsbple SMT 2.5mm	3	0.57	1.71
ESD Protection / TVS Diodes Low Cap	2	0.44	0.88
Thick Film Resistors - SMD 0805 Anti-sulfur 0.5W 68Kohm AEC-Q200	7	0.28	1.96
Thick Film Resistors - SMD 10K OHM 1 %	4	0.38	1.52
MLCC SMD/SMT 16V 1 $\mu$ F X7R 0805 10 % AEC-Q200	2	0.24	0.48
MLCC SMD/SMT 10 $\mu$ F 25VDC 10 % 0805 X5R	5	0.7	3.5
Servomotores MG996R	6	4.5	27
Sensor flex de 2.2"	5	9.38	47
Diseño de PCB realizado en JLCPCB	5	1.4	7
Stencil para el PCB	1	7	7
<b>TOTAL GENERAL</b>			<b>117 + 97 = 214</b>

Costos de impresiones 3D y materiales para el montaje



### MANO ROBÓTICA

Descripción	Cantidad	Precio (Gs)	Total (Gs)
Nylon	1	4000	14000
Impresiones 3D	2	14000	28000
Tornillos	27	76	2052

#### b. Materiales y herramientas prestadas/existentes

Componente	Descripción	Cantidad
Flux en pasta	Flux para soldar, facilita una soldadura uniforme	1
Estaño en pasta SN67PB37 Tipo 4	Aleación 67 % Sn y 37 % Pb	1
Malla de cobre	Cinta trenzada absorbente por capilaridad	1
Impresiones 3D	Material para la estructura	24

### 3.9. Horas Hombre

El desarrollo del proyecto requirió trabajo coordinado entre los integrantes, abarcando tareas de diseño, programación, ensamblaje, pruebas y documentación. En la Tabla 3.1 se detallan las horas hombre estimadas para cada actividad principal.

Actividad	Horas Hombre
Diseño de la interfaz gráfica (GUI)	12 h
Implementación del módulo de login	6 h
Programación de sliders y control manual	10 h
Integración con ESP32 y comunicación	8 h
Diseño del diagrama eléctrico y conexiones	6 h
Montaje físico de la mano robótica	10 h
Pruebas, validación y calibración	8 h
Documentación y redacción del informe	12 h
<b>Total estimado</b>	<b>72 h</b>

**Tabla 3.1:** Horas hombre estimadas para el desarrollo del proyecto.

### 3.10. Cambios y Ajustes Realizados

Durante el desarrollo del proyecto se identificaron diversas mejoras necesarias tanto en la interfaz gráfica como en la parte electrónica y mecánica del sis-

---

## MANO ROBÓTICA

---

tema. A continuación, se enumeran los principales cambios y ajustes implementados durante el proceso:

- **Optimización del diseño de la interfaz gráfica:** se ajustó la distribución de los sliders, botones y elementos visuales para mejorar la usabilidad y permitir un control más intuitivo de los dedos de la mano robótica.
- **Correcciones en el módulo de inicio de sesión:** se corrigieron problemas en la validación de credenciales y se ajustó el flujo de acceso para asegurar un funcionamiento consistente del sistema de autenticación.
- **Ajustes en la comunicación entre la GUI y el ESP32:** se modificaron los formatos de datos enviados para mejorar la estabilidad y reducir retrasos durante la transmisión de comandos.
- **Revisión del diagrama eléctrico:** se actualizaron las conexiones entre el conversor de nivel lógico, el ADC y los servomotores, con el fin de asegurar compatibilidad de tensiones y evitar sobrecargas.
- **Mejoras en el guante sensorizado:** se ajustó la disposición de los sensores flex y su calibración para reducir errores no lineales en la medición de flexión de dedos.
- **Ajustes en la estructura mecánica de la mano:** se reforzaron algunas uniones y se realizaron pequeñas modificaciones para mejorar el acoplamiento entre servomotores y piezas impresas.
- **Corrección de errores en las funciones de control:** se modificaron los rangos de movimiento enviados a los servomotores para evitar saturaciones y asegurar un movimiento fluido y consistente.
- **Actualización del manual de usuario:** se revisaron las instrucciones y se

añadieron pasos adicionales para facilitar la comprensión del funcionamiento de la interfaz.

### **3.11. Manual de Usuario**

El presente manual describe el uso de la interfaz gráfica desarrollada para controlar la mano robótica mediante dos modos de operación: **modo manual** y **modo guante**. La interfaz permite mover cada dedo individualmente, guardar movimientos, ejecutar secuencias y monitorear el estado de conexión del sistema.

#### **1. Inicio de la aplicación**

- Abra el programa *Control Mano Robótica* desde su computadora.
- Espere a que cargue la ventana de inicio de sesión.

#### **2. Acceso al sistema**

- Ingrese su número de cédula en el campo correspondiente.
- Introduzca su contraseña.
- Seleccione el botón **Iniciar Sesión**.
- Una vez autenticado, la interfaz principal se abrirá automáticamente.

#### **3. Interfaz principal**

Al acceder, el usuario visualizará:

- Los cinco **sliders** superiores correspondientes a cada dedo: Pulgar, Índice, Medio, Anular y Meñique.

---

## MANO ROBÓTICA

---

- Un selector de **modo de operación**: Guante o Manual.
- Una sección de **estado actual** del sistema.
- Una zona de **registro de eventos** donde se muestran mensajes (por ejemplo, intentos de conexión WebSocket).
- Una imagen ilustrativa de la mano robótica.
- Los botones de acción en la parte inferior.

### 4. Modos de operación

#### **Modo Manual:**

- Permite mover cada dedo individualmente usando los sliders.
- Cada slider controla el ángulo del servo entre 0 % y 100 %.
- Los valores se envían al microcontrolador en tiempo real.

#### **Modo Guante:**

- La interfaz deja de usar los sliders.
- Los movimientos son controlados por los sensores flex del guante.
- La mano robótica replica directamente la flexión del usuario.

### 5. Funciones de los botones

En la parte inferior de la interfaz se encuentran los siguientes controles:

- **Guardar Movimiento:** almacena la posición actual de los cinco dedos para su uso posterior.
- **Repetir Movimiento:** ejecuta la última secuencia guardada, moviendo los servomotores automáticamente.

---

## MANO ROBÓTICA

---

- **Resetear:** envía todos los dedos a la posición 0 % (completamente extendidos).
- **Detener Movimiento:** cancela cualquier secuencia o acción en curso.
- **Volver al Login:** cierra la sesión y retorna a la ventana inicial.

### 6. Visualización de porcentajes

Debajo de cada slider se muestra el porcentaje de apertura del dedo seleccionado. Este porcentaje se actualiza conforme el usuario mueve el control.

### 7. Registro y estado del sistema

La interfaz incluye una ventana lateral donde se muestran mensajes como:

- Estado de conexión del WebSocket.
- Comandos enviados al microcontrolador.
- Avisos y errores de la comunicación.

Esto permite al usuario monitorear la conexión y diagnosticar posibles fallas.

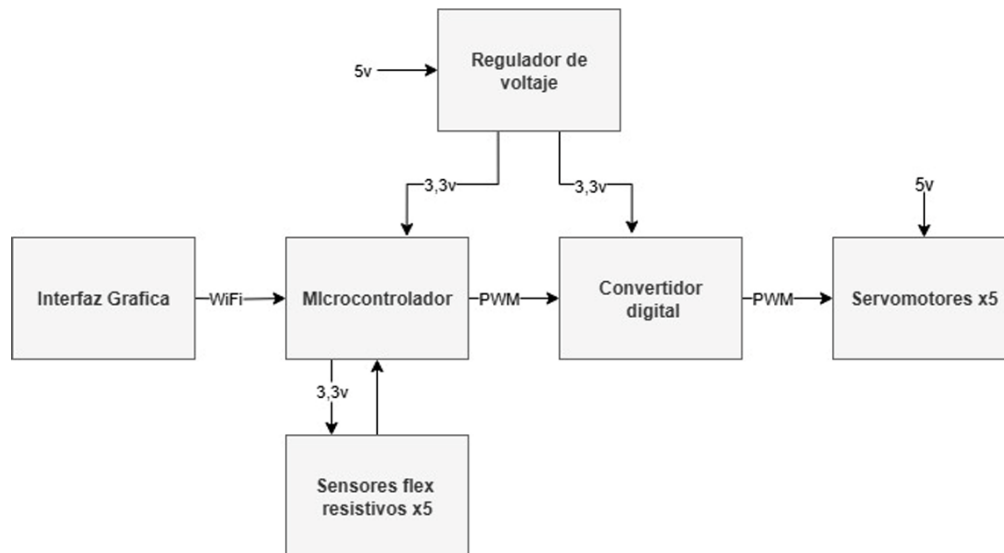
### 8. Finalización

- Finalice su sesión presionando **Volver al Login**.
- Cierre la aplicación si no se utilizará nuevamente.
- Puede apagar el ESP32 o desconectar la alimentación del sistema.

Este manual proporciona una guía clara y completa para operar la interfaz gráfica de la mano robótica, permitiendo un uso intuitivo tanto en modo manual como con el guante sensorizado.

### 3.12. Esquema general del proyecto

En la Figura 3.2 se presenta el esquema general del sistema de mano robótica, donde se visualizan los bloques principales: la interfaz gráfica, el microcontrolador, el conversor de nivel, el ADC, los sensores flex y los servomotores.



**Figura 3.2:** Esquema general del sistema del proyecto.

#### a. Computador / Interfaz gráfica

Tiene como función recibir parámetros del usuario y enviar comandos al microcontrolador.

**Software utilizado:** Interfaz gráfica desarrollada en Python (Tkinter).

**Comunicación:** WiFi hacia el microcontrolador ESP32.

#### b. Microcontrolador ESP32-S3-WROOM-1-N8

Tiene como función ser el núcleo del sistema: leer los sensores flex, procesar los datos, ejecutar el algoritmo de mapeo de flexión, generar señales PWM para los servomotores y comunicarse con la PC.

**Voltaje de operación:** 3.3V.

**Interfaces:**

- I<sup>2</sup>C para el ADC ADS1115.
- PWM para los 5 servomotores MG996R.
- WiFi para comunicación con la interfaz gráfica.

**c. ADC Externo – ADS1115IDGSR**

Realiza el muestreo analógico de los 5 sensores flex resistivos.

**Características principales:**

- Resolución: 16 bits.
- Comunicación: I<sup>2</sup>C.
- Alimentación: 5V.

**d. Sensores Flex resistivos 2.2” (x5)**

Miden la flexión de cada dedo mediante cambio de resistencia al momento de curvarse.

**Alimentación:** 5V.

**e. Actuadores – Servomotores MG996R (x5)**

Función: mover los dedos de la mano impresa, reproduciendo la flexión detectada por los sensores del guante.

**Alimentación:** 5V.

**Control:** Señal PWM proveniente del ESP32.

#### f. Regulador de voltaje – LM1117MPX-3.3/NOPB

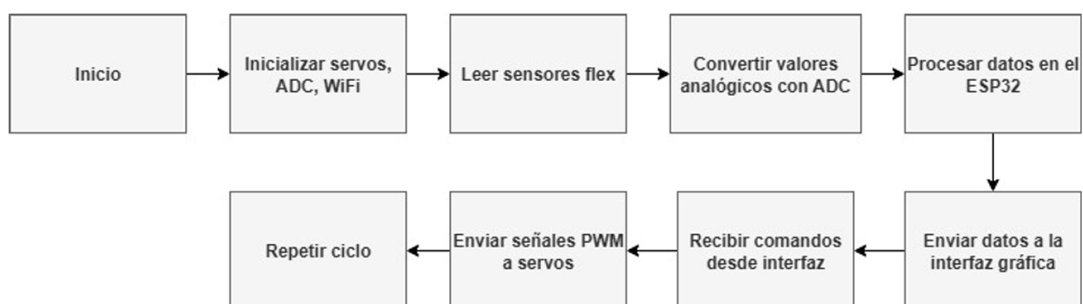
El sistema emplea como regulador principal el LM1117MPX-3.3/NOPB, un regulador lineal de baja caída (LDO) con salida fija de 3.3V. Se utiliza para alimentar al microcontrolador ESP32-S3 y a la lógica de 3.3V del sistema.

##### Características:

- Tipo: Regulador LDO con salida fija a 3.3V.
- Conexión: VIN = 5V.
- GND común para todo el sistema.
- VOUT = 3.3V al ESP32 y a dispositivos lógicos.
- Corriente de salida: hasta 800 mA.

#### Diagrama de programa

En la Figura 3.4 se presenta el flujo de ejecución que lleva a cabo el microcontrolador ESP32 para recibir los datos de los sensores, procesarlos y controlar los servomotores.

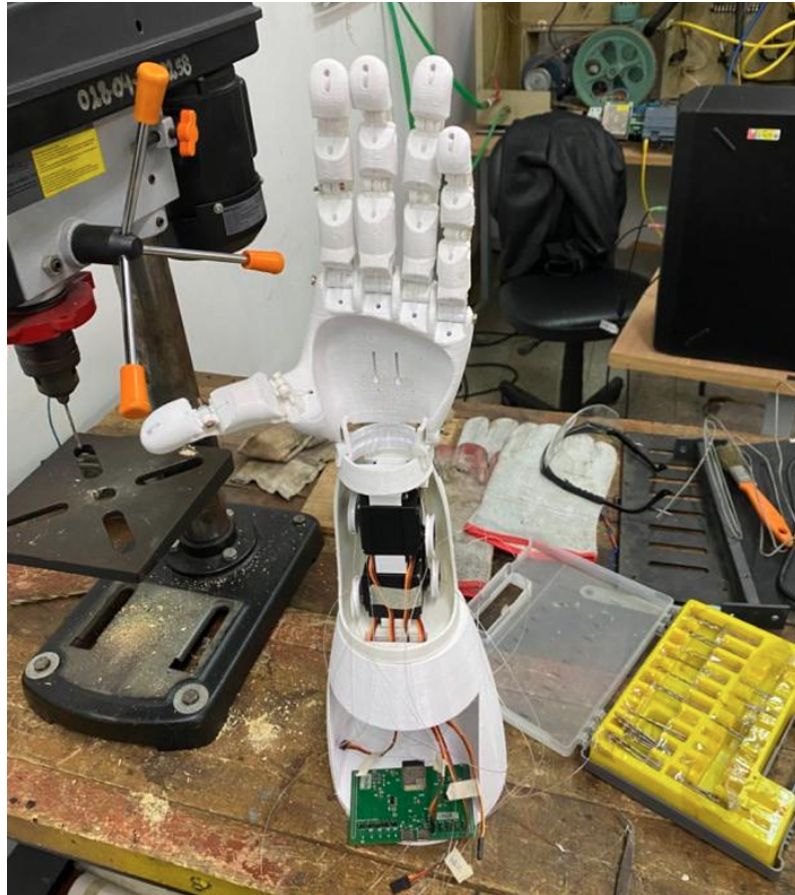


**Figura 3.3:** Diagrama de programa del sistema de mano robótica.



### MANO ROBÓTICA

---



**Figura 3.4:** Mano robótica.

**MANO ROBÓTICA**

---

**BIBLIOGRAFÍA**

- [1] J. Martínez Becerra, “Control de brazo robótico mediante gestos manuales,” 2023.
- [2] J. Reynoso Villaverde, “Diseño y control de una mano robótica para función motora de personas discapacitadas,” Tesis de maestría, Universidad Nacional de Ingeniería, Huancayo, Perú, 2021.
- [3] Shadow Robot Company, “Shadow teleoperation system - technical specification,” 2022. [En línea]. Disponible: <https://shadowrobot.com/shadow-hand-glove/>
- [4] S. Fernandez, “Diagrama de base de datos de la interfaz de la mano robótica,” [https://lucid.app/lucidchart/0a230cbb-447b-4ef2-974f-b25350947e46/edit?viewport\\_loc=472%2C-672%2C2218%2C1010%2C0\\_0&invitationId=inv\\_61c5bfd7-0af0-4efd-9a78-7f8f8e487b90](https://lucid.app/lucidchart/0a230cbb-447b-4ef2-974f-b25350947e46/edit?viewport_loc=472%2C-672%2C2218%2C1010%2C0_0&invitationId=inv_61c5bfd7-0af0-4efd-9a78-7f8f8e487b90), 2025.
- [5] L. Gauna, “Flujograma del sistema de interfaz de la mano robótica,” [https://drive.google.com/file/d/11w\\_s33sXaeoxKkzNcSnfKoEcZBbJr6\\_y/view?usp=sharing](https://drive.google.com/file/d/11w_s33sXaeoxKkzNcSnfKoEcZBbJr6_y/view?usp=sharing), 2025.