



**UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA  
FACSISEL  
INGENIERÍA EN ELECTRÓNICA Y AUTOMATIZACIÓN  
TRABAJO DE INTEGRACIÓN CURRICULAR**

**DESARROLLO DE UNA INTERFAZ HUMANO-MÁQUINA  
PARA EL PROCESO DE EMPAQUETADO UTILIZANDO  
COBOT  
Y COMUNICACIÓN MODBUS TCP/IP.**

**KEVIN ANIBAL CHUCHUCA MINGA**

**Dirigido por:**  
Ing. Carlos Alberto Saldaña Enderica, M.Sc.

La Libertad - 2025

# DEDICATORIA

Dedico este proyecto de titulación a mi madre, ya que gracias a su amor, apoyo y confianza, estoy donde estoy. Ella fue la principal fuente de motivación e inspiración para alcanzar esta meta. Gracias, madre, por ser mi mayor fuerza en este camino.

# AGRADECIMIENTO

Agradezco, en primer lugar a Dios, por brindarme la sabiduría, la fortaleza y la perseverancia necesarias para superar cada reto en mi vida universitaria. También expreso mi profundo agradecimiento a mis padres, quienes me apoyaron incondicionalmente y me dieron la fuerza para poder culminar mi carrera Universitaria.

## APROBACIÓN DEL TUTOR

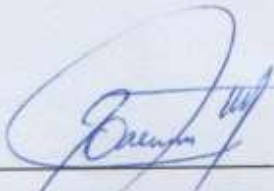
En mi calidad de tutor del trabajo de titulación denominado: "Desarrollo de una Interfaz Humano-Máquina para el proceso de empaquetado utilizando COBOT y comunicación Modbus TCP/IP.", elaborado por el estudiante Kevin Anibal Chuchuca Minga, de la carrera de Ingeniería en Electrónica y Automatización de la Universidad Estatal Península de Santa Elena, declaro que luego de haber orientado, estudiado y revisado, lo apruebo en todas sus partes y autorizo al estudiante que inicie los trámites legales correspondientes.



---

Ing. Carlos Alberto Saldaña Enderica, M.Sc.  
**Tutor**

## TRIBUNAL DE GRADO

X 

Ing. Ronald Humberto Rovira Jurado, Ph.D.  
DIRECTOR DE CARRERA



Ing. Carlos Alberto Saldaña Enderica, M.Sc.  
TUTOR



Ing. Luis Enrique Chuquimarca Jiménez, Mgtr.  
DOCENTE GUÍA UIC  
DOCENTE ESPECIALISTA



Ing. Corina Gonzabay De La A, Mgtr.  
SECRETARIA DEL TRIBUNAL

## Resumen

En este proyecto de titulación se desarrolló e implementó una Interfaz Hombre-Máquina (HMI) para supervisar y controlar, en tiempo real, un COBOT dentro de un entorno industrial de empaquetado utilizando el protocolo Modbus TCP/IP. Se integraron los dispositivos en una misma red industrial, la HMI ejecutada en Raspberry Pi 4 y el controlador EC63, con el fin de habilitar el control y la supervisión del proceso.

La solución integra funciones de gestión del cobot, monitoreo operativo y un sistema de alarmas en tiempo real, garantizando una operación segura y verificable. Las pruebas ICMP confirmaron el acceso y la estabilidad del sistema, registrando una pérdida del 0% y un tiempo medio de respuesta entre 2 y 3 ms. En la capa de aplicación (Modbus/TCP) se registró operación continua ( $\sim 27,7$  s) con 0.638–3.659 ms de latencia (promedio  $\approx 1,3$  ms), sin retransmisiones ni códigos de excepción y con lecturas uniformes. En TCP/IP, las sesiones fueron estables, sin fallas ni pérdidas; el tráfico paralelo no afectó el canal principal. Las transacciones FC03 respondieron en 0.4–6 ms, lo que permitió actualizaciones en milisegundos y confirmó la confiabilidad del intercambio. La validación se realizó en la Universidad Estatal Península de Santa Elena, en su laboratorio de procesos, en un entorno físico de empaquetado donde el COBOT ejecutó toma, traslado y colocación en caja; se evidenció que la comunicación no es un factor limitante. Además, al basarse en Raspberry Pi 4 y software abierto, la HMI es más económica y accesible que un desarrollo equivalente de HMI en PLC, pues reduce costos de hardware y licenciamiento, facilita el mantenimiento y el escalado, y no compromete el desempeño en tiempo real.

**Palabras clave:** HMI, Modbus, TCP/IP, COBOT, Latencia

## Abstract

This project develops and implements a Human-Machine Interface (HMI) to supervise and control, in real time, a COBOT operating in an industrial packaging cell using the Modbus TCP/IP protocol. Devices were integrated on a single industrial network—the HMI running on a Raspberry Pi 4 and the EC63 controller—to enable reliable control and supervision.

The solution integrates cobot management functions, operational monitoring, and a real-time alarm system, ensuring secure and verifiable operation. The ICMP tests confirmed system access and stability, recording 0% packet loss and an average response time between 2 and 3 ms. At the application layer (Modbus/TCP), the system sustained continuous operation for  $\approx 27,7$  s with 0.638–3.659 ms latency (mean  $\approx 1,3$  ms), no retransmissions or exception codes, and consistent register readings. At the transport layer (TCP/IP), sessions remained stable with no protocol errors or losses, and parallel traffic did not affect the main channel. FC03 (Read Holding Registers) transactions responded within 0.4–6 ms, enabling millisecond-level HMI updates and confirming data-exchange reliability. Validation in a physical packaging environment—pick, transfer, and placement into a box—showed that communication is not a limiting factor. Leveraging Raspberry Pi 4 and open-source software makes the HMI more economical and accessible than PLC-based HMI alternatives, reducing hardware and licensing costs and simplifying maintenance and scaling without compromising real-time performance.

**Keywords:** HMI, Modbus, TCP/IP, COBOTS, Latency

# Índice

Índice de figuras	8
Índice de cuadros	9
<b>1. Introducción</b>	<b>10</b>
1.1. Justificación	11
1.2. Panorama actual	12
1.3. Objetivos	14
1.3.1. Objetivo Principal	14
1.3.2. Objetivos Específicos	14
1.4. Fundamentos teóricos	15
1.4.1. Robótica Colaborativa	15
1.4.2. Sistema HMI	15
1.4.3. ISA101 HMI	16
1.4.4. Comunicación sockets	16
1.4.5. Comunicación Maestro/Esclavo mediante Modbus TCP/IP	16
1.4.6. Variantes de Modbus	17
1.4.7. Protocolo Modbus RTU	17
1.4.8. Protocolo Modbus ASCII	18
1.4.9. Protocolo Modbus TCP/IP	18
1.4.10. Robot colaborativo COBOT	18
1.5. Marco Contextual	19
<b>2. Métodos y diseño experimental</b>	<b>20</b>
2.1. Métodos de investigación	20
2.1.1. Investigación Documental:	20
2.1.2. Investigación Tecnológica:	20
2.1.3. Investigación Aplicada:	20
2.2. Descripción del proyecto	21
2.3. Componentes de la Propuesta	22
2.3.1. Componentes Físicos	22
2.3.2. COBOT EC63	22
2.3.3. Gripper DH Robotics	23
2.3.4. Disyuntor CHINT EBC C2	24
2.3.5. Fusibles 1A	24
2.3.6. Botón de parada de emergencia	25
2.3.7. Banda transportadora	25
2.3.8. Sensor Capacitivo LJC18A3-B-Z/BX	26
2.3.9. DC Motor S9D90-90CH SPG	27
2.3.10. Variador de Frecuencia	27
2.3.11. Relé de 24V	27
2.3.12. Relé Winters W4R3L 14 pines	28
2.3.13. Pantalla Touch de 7 pulgadas	29

2.3.14. Raspberry Pi 4 . . . . .	30
2.3.15. Componentes Lógicos . . . . .	30
2.3.16. Raspbian . . . . .	30
2.3.17. Programador Visual Studio Code . . . . .	31
2.3.18. Python . . . . .	31
2.3.19. Vnc Viewer . . . . .	31
2.3.20. Diseño experimental . . . . .	32
2.3.21. Diseño e Implementación del Hardware . . . . .	33
2.3.22. Diagrama Electrico del Proceso de Empaquetado . . . . .	35
2.3.23. Comunicación del Proceso de Empaquetado . . . . .	36
2.3.24. Diagrama Topológico del Proceso de Empaquetado . . . . .	37
2.4. Diseño de la HMI para el sistema de empaquetado . . . . .	38
2.4.1. Implementación de la Interfaz HMI . . . . .	38
2.4.2. Módulo de conexión . . . . .	39
2.4.3. Módulo de Variables . . . . .	40
2.4.4. Módulo de Alarmas . . . . .	41
2.4.5. Módulo de Mantenimiento . . . . .	42
2.4.6. Módulo de Parametros . . . . .	43
2.4.7. Consideraciones de Diseño Visual y Usabilidad . . . . .	44
<b>3. Resultados</b>	<b>45</b>
3.1. Requerimientos del Sistema HMI . . . . .	45
3.2. Restricciones del Sistema HMI . . . . .	46
3.3. Pruebas y resultados obtenidos en la red de comunicación conectada a los equipos mediante Modbus TCP/IP . . . . .	46
3.4. Análisis y resultados del tráfico de datos y latencias en la red Ethernet bajo protocolo Modbus TCP/IP para el sistema de monitoreo y control . . . . .	48
3.5. Resultados de análisis a nivel de aplicación (Modbus/TCP) . . . . .	50
3.6. Resultados de análisis a nivel de transporte (TCP/IP) . . . . .	51
3.7. Análisis de una petición enviada desde el HMI al COBOT EC63 . . . . .	53
<b>4. Conclusiones y Recomendaciones</b>	<b>55</b>
4.1. Conclusiones . . . . .	55
4.2. Recomendaciones . . . . .	56
<b>Referencias</b>	<b>57</b>
<b>A. Ventana de ejecución de la HMI para el proceso de empaquetado</b>	<b>63</b>
<b>B. Ventana de valores operativos de la HMI</b>	<b>64</b>
<b>C. Validación de el sistema HMI en un entorno de prueba que simule el proceso de empaquetado industrial</b>	<b>65</b>

# Índice de figuras

1.	Descripción de mi Proceso de Empaquetado . <i>Fuente: Autoría propia.</i> . . .	21
2.	Ilustración 3. GRIPPER DH ROBOTICS. . . . .	23
3.	Flujo de trabajo del sistema de Interfaz HMI [Autoría Propia] . . . . .	32
4.	Esquema de Pantalla táctil para Raspberry Pi 4 [Autoría Propia] . . . . .	33
5.	Conexiones del Proceso de Empaquetado . <i>Fuente: Autoría propia.</i> . . . .	35
6.	Comunicación del Proceso de Empaquetado <i>Fuente: Autoría propia.</i> . . . .	36
7.	Topología del Proceso de Empaquetado . <i>Fuente: Autoría propia.</i> . . . .	37
8.	Estructura del software Wireshark. <i>Fuente: Autoría propia.</i> . . . . .	48
9.	Datos capturados por el software Wireshark del transporte (TCP/IP). <i>Fuente: Autoría propia.</i> . . . . .	51
10.	Pantalla 1 . <i>Fuente: Autoría propia.</i> . . . . .	63
11.	Pantalla 2 . <i>Fuente: Autoría propia.</i> . . . . .	64
12.	Posición inicial del COBOT . <i>Fuente: Autoría propia.</i> . . . . .	65
13.	Aproximación al punto de toma . <i>Fuente: Autoría propia.</i> . . . . .	65
14.	Traslado de la pieza sobre la banda . <i>Fuente: Autoría propia.</i> . . . . .	66
15.	Aproximación al área de empaque . <i>Fuente: Autoría propia.</i> . . . . .	66
16.	Posicionamiento sobre la caja . <i>Fuente: Autoría propia.</i> . . . . .	66
17.	Alineación final . <i>Fuente: Autoría propia.</i> . . . . .	67
18.	Pieza asegurada . <i>Fuente: Autoría propia.</i> . . . . .	67
19.	Reporte Compilatio. <i>Fuente: Autoría propia.</i> . . . . .	68

## Índice de cuadros

1.	Especificaciones del COBOT EC63 . . . . .	22
2.	Características técnicas del Disyuntor CHINT EBC C2 . . . . .	24
3.	Características técnicas del Fusible 1A . . . . .	24
4.	Características técnicas del Botón de Parada de Emergencia . . . . .	25
5.	Características técnicas del Sensor Capacitivo LJC18A3-B-Z/BX . . . . .	26
6.	Especificaciones técnicas del Motor DC SPG S9D90-90CH . . . . .	27
7.	Características técnicas del Relé 24V DC . . . . .	28
8.	Características técnicas del Relé Winters W4R3L de 14 pines . . . . .	28
9.	Características técnicas de la Pantalla Táctil de 7 pulgadas . . . . .	29
10.	Características técnicas de la Raspberry Pi 4 . . . . .	30
11.	Resumen de funcionalidades por módulo de la HMI . . . . .	38
12.	Requerimientos funcionales del sistema HMI-COBOT . . . . .	45
13.	Restricciones técnicas del sistema HMI-COBOT . . . . .	46
14.	Ping a 192.168.1.210 — Detalle de respuestas . . . . .	46
15.	Ping a 192.168.1.210 — Estadísticas . . . . .	47
16.	Ping a 172.17.103.161 — Detalle de respuestas . . . . .	47
17.	Ping a 172.17.103.161 — Estadísticas . . . . .	47
18.	Resultados de análisis a nivel de aplicación (Modbus/TCP). . . . .	50
19.	Resultados de análisis a nivel de transporte (TCP/IP). . . . .	52
20.	Solicitudes Modbus/TCP (FC03) . . . . .	53
21.	Solicitudes Modbus/TCP (FC03) . . . . .	53
22.	Respuestas Modbus/TCP (FC03) . . . . .	54

# 1. Introducción

La industria 4.0 promueve la digitalización y la automatización, creando una gran cantidad de datos en procesos industriales. Mientras que la industria 5.0 intenta llevar todos los datos a que sean analizados y controlados por un operador, las herramientas que se utilizan para poder realizar el monitoreo y visualización de un proceso Industrial al operador, son las interfaces hombre - máquina (HMI) [1].

Las HMI facilitan la comunicación entre el operador y el equipo, permiten controlar procesos y diagnosticar errores, integrando diversos medios de interacción como pantallas, gráficos y software especializado [2]. Esto lleva al hecho de los COBOTS que se encuentran en el laboratorio de procesos de la Universidad Estatal Península de Santa Elena, no cuentan con un HMI para procesos específicos, en este caso el empaquetado de confitería, pero tienen un teach Pendal donde solo permite controlar el COBOT, no para poder controlar el proceso del empaquetado usando robots colaborativos.

El empaquetado industrial es un sistema altamente automatizado, los cuales emplean tecnologías avanzadas, como las interfaces HMI, a nivel mundial son utilizados con COBOTS, mientras que a nivel local no contamos con la tecnología de los COBOTS. Esta tecnología permite optimizar la producción, asegurar calidad, reducir la intervención manual y lograr procesos más rápidos, precisos y flexibles, consolidando al empaquetado [3]. Los COBOTS, eje de la industria 5.0, se integran con los operadores para elevar eficiencia y disminuir el esfuerzo físico; no obstante, su adopción enfrenta resistencia de empleados y modelos como el TAM (Technology Acceptance Model), no explican completamente la aceptación de estas soluciones [4].

## 1.1. Justificación

En Ecuador, la integración de COBOTS se mantiene aún en una fase temprana. Mientras que en países con un mayor nivel de industrialización estos robots colaborativos forman parte habitual de procesos como el empaquetado, el ensamblaje o la inspección, en el contexto ecuatoriano su implementación continúa siendo limitada. Actualmente, las marcas disponibles en el país son: Universal Robots (UR) y KUKA; no obstante, su nivel de adopción dentro del sector empresarial se mantiene reducido [5] [6].

En el Laboratorio de Control de Procesos de la Universidad Estatal de la Península de Santa Elena UPSE funcionan COBOTS fabricados por Elite Robots, incluyendo el EC63. Este se controla mediante un teach pendant, pero no permite ver el proceso de empaquetado ni la condición del sistema. Esto complica la supervisión en tiempo real, el rastreo de variables como la velocidad o el conteo y la detección de anomalías (colisiones, alarmas), lo que tiene un efecto perjudicial sobre la seguridad y la efectividad.

Mi propuesta para este trabajo de titulación se basa en desarrollar e implementar una HMI personalizada para el empaquetado de confitería con COBOTS, donde el operador disponga de dos ventanas de supervisión y control que integren sinóptico del proceso, indicadores, conteo, estados, alarmas y detección de colisiones. Esta propuesta guarda relación con el antecedente de empaquetado de platos de poliestireno con PLC y HMI, pero introduce innovación en la plataforma: Raspberry Pi 4 y Python comunicados por Modbus TCP/IP, esto permite compatibilidad con diversos equipos, bajo costo y facilidad de mantenimiento [7]. Por otra parte, el proyecto integrador de ESPOL respalda la viabilidad técnica de esta arquitectura al integrar PLC, Raspberry Pi 4 y Python en un entorno industrial real [8].

Con respecto al Software, el uso de Python junto a Bibliotecas de código abierto elimina la necesidad de licencias comerciales, lo que disminuye el costo de implementación y mantenimiento. Además, la amplia comunidad de desarrolladores públicos activos garantizan el apoyo y las actualizaciones continuas sin costo adicional, lo que con el tiempo promueve la sostenibilidad del proyecto [9]. Además la comunicación a través de Modbus TCP/IP facilita la compatibilidad entre distintas marcas [10].

La relevancia de este proyecto es que cualquier persona sin un conocimiento tan avanzado pueda monitorear el HMI, que permitiera al operador, estudiantes y maestros, una facilidad de manejo interno de supervisión y seguridad, además el enfoque de bajo costo que proporciona beneficios concretos para el laboratorio de procesos: inversiones iniciales económicas y bajos costos operativos. Esto garantiza una solución económica, escalable y de gran relevancia educativa, alineada con las exigencias de la Industria 5.0 y con las necesidades formativas de la Universidad.

## 1.2. Panorama actual

La automatización industrial incorpora PLC–HMI–dispositivos vía OPC UA/PROFINET para datos en tiempo real y supervisión continua. Desde 2023, interfaces en código Python impulsan la comunicación bidireccional y el monitoreo remoto [11].

Proyectos recientes son incorporados con soluciones de comunicación inalámbrica y control remoto a través de módulos SCALANCE y plataformas HMI las cuales son desarrolladas en Python para el control de motores trifásicos, estos demuestran una comunicación estable y de bajo recursos en entornos industriales [12]. En Ecuador, empresas como General Motors (GM OBB) han incorporado robots colaborativos para procesos de soldadura y pintura, llegando a lograr unidades semiautomatizadas a través de la comunicación con controladores lógicos programables [13].

De igual forma, se han integrado sistemas HMI basados en código abierto para Raspberry Pi, Python y Qt Designer que constituyen protocolos Modbus TCP/IP y RTU con PLC S7-1200 y variadores de frecuencia, permitiendo el monitoreo de variables como presión, temperatura y nivel. Estas estrategias sobresalen por su bajo costo, escalabilidad y mantenimiento simplificado [14]. Distintas investigaciones se han abordado en la automatización de sistemas de empaquetado a través de modelos PLC–HMI, coordinando sensores y actuadores neumáticos para perfeccionar el ciclo de producción y reducir errores [15].

El contexto de la formación técnica, la capacitación de operarios durante la interacción, mediante los COBOTS ha presentado mejoras en la transmisión de habilidades y una disminución considerable de fallas cuando se emplean procesos de tutorías estructuradas y guiadas [16]. En el entorno internacional, se ha demostrado que su incorporación de la robótica colaborativa, depende de su utilidad, facilidad de uso y seguridad, factores determinantes en su adopción industrial [17].

Bajo la perspectiva del monitoreo, se han planteado modelos de empaquetado humano–robot que combinan visualización por computadora y comunicación Modbus para realizar ejecuciones, prevenir colisiones y reducir los tiempos de ciclo [18]. De modo adicional, las interfaces cerebro–computadora (BCI) posibilitan controlar robots a través de señales neuronales, ampliando las funcionalidades de interacción y facilidad de acceso para los operarios [19].

Por último en el ámbito de la ergonomía, indica experimentar con 14 participantes, muestra los niveles de colaboración (FRb) estos disminuyen de una forma considerable, el desplazamiento y la velocidad del centro de presión, como las áreas y la variabilidad interindividual. En cambio, niveles intermedios implican una responsabilidad entre sus pocos riesgos biomecánicos y mayor control por parte del usuario [20].—

En HRC, un sistema con profundidad + RA (proyector–espejo y gafas AR) acorta tiempos, reduce inactividad y mejora seguridad/ergonomía; aún falta madurez en wearables como HoloLens para entornos industriales [21].

Se han desarrollado enfoques para conservar al humano en el bucle de control, analizando el estado emocional mediante EEG y adecuando en tiempo real el comportamiento del COBOT, lo que refuerza la confianza y la ergonomía del sistema [22]. En tareas de picking, el uso de cobots ha demostrado mejoras en tiempos de respuesta, reducción de costos y mayor seguridad frente a operaciones manuales [23].

Igualmente se ha demostrado que los gemelos digitales y los entornos virtuales inmersivos

permiten ayudar en la precisión operativa y la seguridad en procesos de entrenamiento y monitoreo industrial [24]. Nuevos estudios proponen estrategias de diseño multimodal centradas en el ser humano que integran visión por computadora y realidad aumentada para guiar al operario y disminuir el esfuerzo físico [25] [26].

La robótica colaborativa en la industria de servicios evidencia que la eficiencia y calidad del servicio son superiores, sin embargo, también conlleva retos vinculados a la percepción del cliente y la aceptación en el trabajo; esto resalta la necesidad de mejorar tanto el compromiso del trabajador como su experiencia [27].

En el ámbito de la seguridad HRI, los modelos basados en niveles de autonomía (LoA) y atención emocional, combinados con visión e inteligencia artificial, logran trayectorias más seguras en tareas pick-and-place y mejoran la confiabilidad del sistema [28]. Paralelamente, investigaciones sobre bienestar laboral asocian la complejidad de las tareas con indicadores fisiológicos de carga, como la frecuencia cardíaca o la conductancia de la piel [29]. Los avances en inteligencia artificial y visión incluyen modelos de deep learning para reconocimiento de objetos, voz y gestos que aumentan la naturalidad y precisión de la interacción con cobots [30].

Asimismo, el uso de métodos de videometría y cinemática inversa en robots colaborativos, como el FANUC CRX-10iA, ha permitido validar tareas de ensamblaje con mayor exactitud y respuesta rápida [31]. Bajo el enfoque de la Industria 5.0, la interacción humano-máquina se orienta a un modelo centrado en la persona. Se promueven la personalización, la sostenibilidad y el bienestar del trabajador, mediante la aplicación de inteligencia artificial, aprendizaje adaptativo y robótica cognitiva [32].

El AR-HMD mostró una adecuada capacidad de uso al integrarse en la carga y flujo de trabajo de un astillero de la Marina Australiana; aunque todavía queda pendiente realizar algunos ajustes (seguimiento, gestos, interfaces), se trata de una plataforma versátil que integra tecnología digital sin reducir el rendimiento, lo que eleva la calidad y la productividad [33]. Asimismo, se ha observado que los diseños lúdicos (playful) incrementan la confianza sentida y la confianza afectiva del operario hacia el cobot, lo que favorece la interacción en situaciones de cooperación [34].

En los módulos de ensamblaje se integró un cobot "social" con avatar (covatar), que colabora en la regulación del estado emocional del operario en tiempo real para sostener la fluidez durante las tareas tediosas. El sistema ajusta su intervención a través de un control global de sensibilidad, mientras modela continuamente las señales PAD (Placer-Activación-Dominancia). Se ha comprobado en ensayos controlados que la *Dominancia* predice el *flow* y la autoeficacia; también se ha verificado que la co-regulación reduce las circunstancias negativas (como el aburrimiento), lo que mejora tanto la productividad como el rendimiento sin perjudicar la eficacia [35].

El estudio de modelos paramétricos para establecer la viabilidad técnica, económica y social de implementar cobots en el proceso de fabricación. Los cobots son más rentables en situaciones con gastos laborales altos, ya que ayudan a disminuir los tiempos de procesamiento y respaldan decisiones sostenibles [36].

La protección en celdas colaborativas al delinear y combinar marcos de referencia basados en la ruta y egocéntricos. El método genera índices de riesgo numéricos y facilita el cambio de los parámetros de control. Los experimentos llevados a cabo con un brazo antropomórfico de 7 GDL muestran progresos [37].

## **1.3. Objetivos**

### **1.3.1. Objetivo Principal**

Diseñar una interfaz en HMI que posibilite a los operadores la configuración y el seguimiento en tiempo real de los sistemas de robótica colaborativa empleados en el proceso de empaquetado industrial, utilizando una red MODBUS TCP/IP.

### **1.3.2. Objetivos Específicos**

- Estudiar los protocolos de comunicación MODBUS TCP/IP necesarios para establecer una conexión en tiempo real entre la HMI y el sistema de robótica colaborativa, garantizando un intercambio de datos eficiente y seguro que permita el control y supervisión óptima del cobot.
- Desarrollar una interfaz HMI que incorpore medidas de seguridad y control, garantizando una operación segura del sistema de robótica colaborativa y previniendo posibles fallos durante la interacción con el cobot.
- Desarrollar un sistema de alarmas en tiempo real en la HMI para el proceso de empaquetado.
- Validar el sistema HMI en un entorno de prueba que simule el proceso de empaquetado industrial, asegurando la eficacia y fiabilidad del sistema en condiciones reales antes de su implementación definitiva.

## 1.4. Fundamentos teóricos

Este proyecto se fundamenta en las secciones siguientes, cuyos temas centrales tienen que ver con la interfaz entre humanos y máquinas (HMI), los protocolos de comunicación y la robótica colaborativa industrial Modbus TCP/IP no obstante los sistemas de monitoreo y administración en tiempo real. Estos elementos claves nos permite entender los fundamentos tecnológicos que respaldan la creación y puesta en marcha de sistemas automatizados de empaque con COBOTS.

Cada uno de estos temas tiene un papel muy importante en la comprensión de las características del ambiente industrial, en el que los COBOTS tienen interacciones efectivas y seguras con los seres humanos. Se examinará la arquitectura de los sistemas HMI, su función en el control y seguimiento del proceso de empaquetado y las ventajas que ofrece el empleo de protocolos estándar como Modbus TCP/IP, que aseguran una comunicación rápida, confiable y adaptable.

### 1.4.1. Robótica Colaborativa

En el ámbito industrial, los robots colaborativos y manipuladores han cobrado importancia y utilidad. Esto ocurre porque se sigue industrializando todos los trabajos manuales, con el objetivo de ahorrar dinero y tiempo, aumentar la productividad y mejorar las condiciones de trabajo. Por eso, en este contexto, saber programar distintos tipos de robots industriales ofrece una gran versatilidad. Por esa razón, nos enfocamos en el Sistema ROS (Robotic Operating System), porque tiene la capacidad de trabajar con robots producidos por distintas compañías en un mismo ambiente [38].

Un COBOT o robot colaborativo se puede ajustar en función de la carga de trabajo y puede trabajar sin parar y mantener la productividad las 24 horas de la jornada. Gracias a que un brazo robótico de empaquetado y paletizado puede ser reprogramado tantas veces como se requiera en cuestión de minutos, su interfaz intuitiva y su diseño ligero y compacto [39]. Su pequeño tamaño ahorra espacio y le permite encajar fácilmente en un espacio de trabajo con otras personas, garantizando seguridad y eficiencia en el empleo. Un cobot puede ser programado para que se detenga o funcione a un ritmo más lento si alguien entra en su área de trabajo, a fin de prevenir peligros [39].

### 1.4.2. Sistema HMI

Las interfaces hombre-máquina (HMI) es una herramienta que permite la comunicación entre operadores y sistemas de producción, combinando hardware y software para monitorear, controlar y automatizar procesos en tiempo real. Se suele utilizar en la industria a través de pantallas o paneles táctiles que ayudan a visualizar datos, detectar errores y controlar equipos centralizados. Estas interfaces mejoran la productividad, mejoran la seguridad y garantizan la calidad del producto en industrias como la manufacturera, de alimentos y bebidas, farmacéutica y de servicios públicos, que son esenciales para optimizar el rendimiento y garantizar la continuidad operativa [40].

### 1.4.3. ISA101 HMI

La norma ISA-101 se ha incorporado como estándar internacional para mejorar el diseño, la implementación y la gestión del ciclo de vida de las interfaces hombre-máquina (HMI) en todas las industrias. Su desarrollo fue una respuesta a las limitaciones de las HMI tradicionales, que a menudo dificultan que los operadores comprendan la situación, generando riesgo de errores, fatiga e incidentes en la fábrica. Desde esta perspectiva, el estándar se centra en proporcionar interfaces transparentes, seguras y eficientes que mejoren la capacidad del operador para interpretar información crítica y responder rápidamente a condiciones anormales [41].

Este estándar cubre de manera integral todas las etapas del ciclo de vida de la HMI, desde el diseño inicial hasta la operación, el mantenimiento y el desmantelamiento. Proporciona orientación sobre la preparación de documentos de requisitos, pruebas, capacitación y gestión de cambios, y es independiente de la plataforma o del proveedor, lo que le permite desarrollar guías de estilo comunes que se aplican en múltiples entornos. Además, incorpora factores humanos, ergonomía y principios de usabilidad para ayudar a reducir los costos de diseño y personalización. Además, las pantallas se dividen en cuatro niveles jerárquicos: desde descripciones generales de procesos hasta vistas de diagnóstico más detalladas [42].

Aprobado en 2015 como ANSI/ISA-101.01-2015, este estándar continúa siendo perfeccionado por grupos de trabajo que producen informes técnicos sobre guías de estilo, usabilidad, HMI móviles y aplicaciones de máquinas. Estas contribuciones brindan a las organizaciones una base sólida para desarrollar HMI potentes, intuitivas y con rendimiento optimizado. La implementación de ISA-101 permite a los equipos reducir los errores de los operadores y aumentar el conocimiento de la situación, mejorar la seguridad y el cumplimiento, mejorar la confiabilidad del sistema y la continuidad del negocio, y permitir la estandarización y escalabilidad en todas las plantas industriales [42].

### 1.4.4. Comunicación sockets

Un socket constituye en nodo de comunicación que permite la transmisión de datos entre operaciones locales o remotos dentro de una red. Esta tecnología, considerada estándar en los protocolos TCP/IP y compatible con la mayoría de sistemas operativos, facilita tanto las aplicaciones autónomas como las distribuidas, al proporcionar acceso a datos centralizados y mejorar la eficiencia en el uso de recursos. Su utilización más común se encuentra en el esquema cliente-servidor, donde el servidor enlaza una dirección para recibir solicitudes y responder a los clientes conectados, garantizando un flujo de comunicación seguro y eficiente [43].

### 1.4.5. Comunicación Maestro/Esclavo mediante Modbus TCP/IP

El protocolo Modbus TCP/IP, al simplificar la interconexión entre un gran número de sistemas y dispositivos, se ha establecido como el estándar para la comunicación industrial. Modbus TCP/IP ha actualizado el modo en que las industrias controlan y automatizan sus procesos utilizando la versatilidad y la cobertura de las redes Ethernet. Este procedimiento, sustentado en una arquitectura cliente-servidor, facilita la conexión de dispositivos como PLC, sensores y actuadores transmitan datos de forma segura y efectiva, independientemente de su fabricante [44].

El protocolo Modbus se basa en un esquema de comunicación maestro-esclavo. El maestro

es el único dispositivo autorizado para iniciar la comunicación, enviando solicitudes a uno o varios esclavos. Esta comunicación puede realizarse en modo unidifusión, cuando se dirige a un esclavo específico, o en difusión, al enviarse una instrucción idéntica a todos los dispositivos esclavos usando la dirección general. El maestro controla el flujo de información, garantizando orden y centralización en la red [45].

Por su parte, los dispositivos esclavos solo pueden contestar a las peticiones enviadas por el maestro, sin la posibilidad de iniciar comunicaciones por sí mismos. Respecto al desempeño, el tiempo de latencia de un esclavo generalmente es inferior a 10 ms en la mayoría de los casos de los intercambios, aunque puede alcanzar hasta 700 ms, por este motivo se sugiere un tiempo de espera cercano a 1 segundo. El protocolo Modbus comunica datos en forma de bits únicos y registros de 16 bits, cada dispositivo cuenta con una dirección única, lo que facilita la gestión y el procesamiento de datos en sistemas de automatización industrial [45].

#### **1.4.6. Variantes de Modbus**

”Variantes corresponde a versiones distintas o alternativas de un mismo elemento que, aunque tienen similitudes en su estructura, son diferentes en ciertos aspectos particulares. El sistema de comunicación más utilizado en la automatización industrial y el método más frecuente para conectar dispositivos electrónicos automatizados es Modbus. Este lenguaje permite que, desde los sensores más simples hasta los controladores más complejos, se conecten y envíen información, lo que favorece una operación más exacta y eficaz. El protocolo Modbus tiene múltiples versiones que se ajustan a diversas circunstancias y requerimientos específicos. Cada tipo proporciona características exclusivas que lo hacen apropiado para diferentes tecnologías de comunicación y configuraciones de red [46].

#### **1.4.7. Protocolo Modbus RTU**

Un protocolo se entiende como un conjunto de reglas y convenciones que permiten establecer un lenguaje común entre dispositivos que posibilitan la comunicación entre sistemas de software o entre dispositivos en una red. El protocolo Modbus RTU constituye un canal de comunicación entre dispositivos, que facilita la transferencia de información entre los ordenadores (PC) y los controladores lógicos programables. Este protocolo permite que los dispositivos electrónicos se comuniquen por medio de conexiones en serie. Funciona con una estructura maestro-esclavo, en la que un dispositivo de tipo maestro (como un sistema SCADA o un PLC) tiene la capacidad de establecer comunicación con varios dispositivos esclavos (por ejemplo, actuadores, sensores, variadores de frecuencia) mediante una conexión serial. Este protocolo ha sido diseñado de manera específica a fin de posibilitar la transmisión eficiente de datos entre dispositivos electrónicos por medio de enlaces serie, como RS-232 o RS-485. Intercambian información a través de secuencias de datos binarios, el aparato maestro, al desempeñarse como controlador lógico programable o cualquier otro dispositivo de control, comienza y supervisa la comunicación con los dispositivos esclavos (por ejemplo, sensores o actuadores) [47].

#### **1.4.8. Protocolo Modbus ASCII**

Esta es una variación del protocolo Modbus que utiliza ASCII (Código estándar americano para el intercambio de información) para transmitir datos en lugar de codificación binaria Modbus RTU. Esto quiere decir que cada byte de datos se transforma en un par de caracteres ASCII, lo cual provoca que el mensaje sea más fácil de leer para los humanos, pero provoca que la comunicación sea menos eficaz y más lenta en comparación con Modbus RTU. En sistemas que requieren integración con dispositivos antiguos solo ASCII. Es muy útil en entornos con mucho ruido porque su estructura de mensajes permiten identificar errores [48].

#### **1.4.9. Protocolo Modbus TCP/IP**

Es una versión de la familia MODBUS de protocolos de comunicación sencillos y neutrales para monitorear y controlar dispositivos de automatización. Comprende la utilización de mensajes MODBUS en una red local o entorno de Internet utilizando el protocolo TCP/IP. Actualmente, los protocolos más empleados conectan PLC, módulos de E/S y puertos Ethernet a diferentes buses de campo o redes de E/S simples. Se establece como estándar de automatización. Debido a que MODBUS es ampliamente conocido, hay poca información en este documento que no se pueda encontrar en otros lugares. A pesar de ello, se hizo un esfuerzo para determinar qué características de MODBUS son importantes para la compatibilidad del hardware de automatización [49].

#### **1.4.10. Robot colaborativo COBOT**

Un COBOT o robot colaborativo se puede adaptar a la carga de trabajo, funcionar sin parar y mantener la velocidad de producción las 24 horas del día. Además, gracias a la interfaz intuitiva y al diseño ligero y compacto, el brazo robótico de embalaje y paletizado se puede reprogramar tantas veces como sea requerido en solo unos minutos. Su pequeño tamaño facilita su integración en el puesto de trabajo con otras personas, garantizando seguridad y eficiencia en el trabajo. Cuando un humano interviene en el espacio de trabajo de un cobot, se puede programar para que trabaje más lento o incluso se detenga para evitar riesgos. [50].

## 1.5. Marco Contextual

El Laboratorio de Control de Procesos será el lugar donde se desarrolle la tesis. Este espacio dispone de los instrumentos requeridos para poner en marcha y probar robots colaborativos (COBOTS), redes de comunicación industrial y herramientas de monitoreo.

El laboratorio proporciona un entorno controlado donde se podrá establecer la interfaz entre el ser humano y la máquina para supervisar y cambiar en tiempo real los parámetros del proceso. El protocolo de comunicación Modbus TCP/IP se utilizará para garantizar la conexión entre los dispositivos y el HMI, lo cual permitirá un flujo de datos preciso y eficiente durante la automatización del empaquetado. Este procedimiento es factible en contextos industriales donde la manufactura y la producción automatizada son intensas.

Este tipo de sistema es especialmente relevante en fábricas y empresas que empacan alimentos, dulces, bebidas y productos farmacéuticos, dado que la exactitud y la eficiencia son elementos esenciales. La utilización de COBOTS junto con una interfaz HMI en estas áreas industriales permite mejorar la productividad, reducir los errores de los humanos y optimizar los recursos, lo que se adapta a la creciente necesidad de automatización en industrias en expansión.

La integración tiene la oportunidad de enfocarse en zonas con un desarrollo industrial estable, donde se promuevan los proyectos de Industria 4.0, como regiones industriales que dispongan de infraestructura tecnológica avanzada, sistemas de comunicación veloces y una mano de obra calificada que sea capaz de adaptarse a las nuevas tecnologías emergentes en materia de control y automatización en tiempo real.

Este proyecto será beneficioso para las empresas industriales de Ecuador y para la universidad. La universidad consolidará su investigación aplicada en control y automatización, brindando una alternativa de aprendizaje auténtica. Las compañías que posean robots colaborativos, aunque no cuenten con sistemas de interfaz entre humanos y máquinas que sean eficaces, aumentarán su eficacia y seguridad. Por otro lado, las empresas que no tengan automatización dispondrán de una vía para modernizarse y competir a escala global.

## **2. Métodos y diseño experimental**

El presente capítulo presenta los enfoques metodológicos utilizados para desarrollar la interfaz HMI, configurar las comunicaciones utilizando el protocolo Modbus TCP/IP e implementar control, monitoreo y alarmas en tiempo real para el sistema de robot cooperativo utilizado en el proceso de empaque. Además, se utilizan estrategias detalladas para validar la funcionalidad y eficacia del sistema desarrollado en un entorno de prueba controlado. .

### **2.1. Métodos de investigación**

#### **2.1.1. Investigación Documental:**

En el avance de este proyecto de tesis, la revisión de literatura será una parte fundamental. El propósito fundamental de este método será recopilar, analizar y condensar la información existente sobre las redes de comunicación industrial MODBUS TCP/IP, la robótica colaborativa y los sistemas de interfaz HMI que se utilizan en el empaquetado.

#### **2.1.2. Investigación Tecnológica:**

En este proyecto se aplica dicho enfoque para diseñar y validar una HMI que se comunica con el cobot EC63 mediante Modbus TCP/IP, demostrando su funcionamiento adecuado en un entorno de prueba industrial.

#### **2.1.3. Investigación Aplicada:**

Este enfoque se utiliza para aplicar conocimientos técnicos existentes con el fin de resolver un problema real del entorno industrial. En este proyecto, la investigación aplicada permite implementar una solución que mejora el proceso de empaquetado mediante la integración de una HMI y un cobot EC63, optimizando la interacción entre el operador y el sistema de empaquetado.

## 2.2. Descripción del proyecto

La finalidad de este proyecto es desarrollar una HMI intuitiva que permita a los operadores monitorear el proceso de empaquetado en tiempo real y controlar el COBOT, optimizar su rendimiento. El sistema se comunica mediante el protocolo Modbus TCP/IP, muy utilizado en la industria por su fiabilidad y flexibilidad, facilitando la integración con otros dispositivos y sistemas. La HMI muestra claramente parámetros clave del proceso de embalaje, como la cantidad de material detectado por los sensores, el progreso de la producción y advertencias o condiciones anormales. Además, se realizó un estudio técnico del protocolo Modbus TCP/IP, examinando la arquitectura cliente-servidor, direccionamiento IP y gestión de tramas, evaluando sus ventajas y limitaciones para proporcionar comunicaciones estables en tiempo real para el control COBOT en procesos industriales.

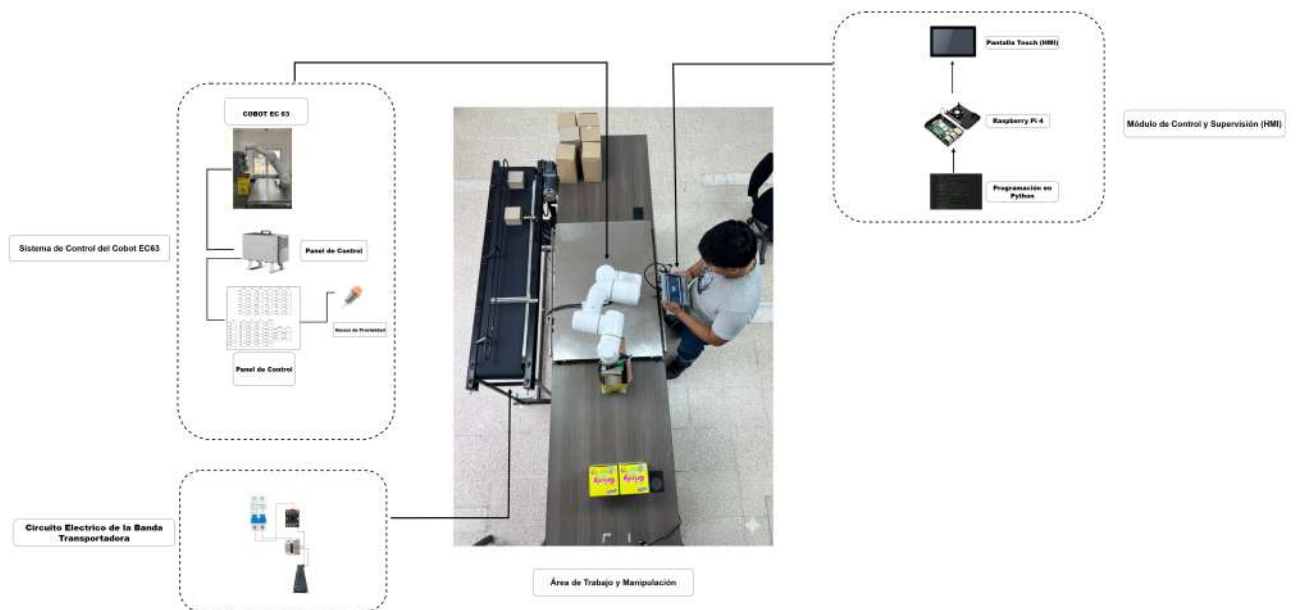


Figura 1: Descripción de mi Proceso de Empaquetado .  
*Fuente: Autoría propia.*

## 2.3. Componentes de la Propuesta

Este capítulo presenta la metodología utilizada para el diseño e implementación de un sistema automatizado de empaquetado, detallando los componentes físicos y lógicos involucrados en la propuesta.

### 2.3.1. Componentes Físicos

Los componentes físicos desempeñan un papel importante en la implementación práctica. A continuación, se encuentran detallados los componentes utilizados para desarrollar esta propuesta de investigación.

### 2.3.2. COBOT EC63

El Cobot EC63 es compacto, liviano y fácil de instalar en cualquier lugar. Se caracteriza por su precisión y flexibilidad, ideal para ensamblar, manipular y trabajar fácilmente en tablas de producción o líneas de producción, sus características técnicas se encuentran en la Tabla 1 [51].

Tabla 1: Especificaciones del COBOT EC63

<b>Parámetro</b>	<b>Especificación</b>
Carga útil	3 kg
Radio	624 mm
Precisión	$\pm 0,02$ mm
Eje	6
Grado de protección IP	IP54
Rango de temperatura	-10°C a 50°C
Humedad relativa	5 %–95 % (sin condensación)
Consumo típico	185 W
Conector de E/S de la herramienta	12 clavijas
Puertos de E/S de la herramienta	2 DI, 2 DO; 1 AI, 1 AO
Alimentación de E/S de la herramienta	24V/2A
Comunicación de E/S de la herramienta	RS485, Modbus RTU
Huella	$\varnothing$ 128 mm
Peso	13 kg

### 2.3.3. Gripper DH Robotics

El gripper de DH Robotics presenta un tamaño pequeño con una instalación flexible, el tamaño más delgado es de 18 mm con una estructura compacta. Además, admite al menos cinco métodos de instalación flexibles para satisfacer las necesidades de las tareas de sujeción, lo que ahorra espacio en el diseño [52].

El tiempo de apertura y cierre más rápido puede alcanzar 0,2 s / 0,2 s, lo que cumple con los requisitos de sujeción estable y de alta velocidad de la línea de producción. Con un diseño de controlador especial y una compensación de algoritmo de controlador, la fuerza de agarre se puede ajustar de forma continua, y la repetibilidad de la fuerza puede alcanzar 0,1 N [52].



Figura 2: Ilustración 3. GRIPPER DH ROBOTICS.

## Dispositivos de protección

### 2.3.4. Disyuntor CHINT EBC C2

El disyuntor CHINT EBC C2 protege sistemas eléctricos, monofásicos de sobrecarga, con capacidad de interrupción de 3000 A. Ofreciendo seguridad y confiabilidad en aplicaciones industriales [53]. A continuación, se detallan las especificaciones técnicas del Disyuntor CHINT EBC C2 .

Tabla 2: Características técnicas del Disyuntor CHINT EBC C2

Característica	Especificación
Modelo	CHINT EBC C2
Corriente nominal ( $I_n$ )	2 A
Número de polos	1 polo (1P)
Curva de disparo	Tipo C (5 a 10 veces la corriente nominal)
Tensión nominal ( $U_e$ )	240/415 V AC
Frecuencia nominal	50 Hz
Poder de corte ( $I_{cn}$ )	3 kA (3000 A)
Vida mecánica	10 000 ciclos
Vida eléctrica	4 000 ciclos
Montaje	Sobre carril DIN EN 60715 (35 mm)
Terminales de conexión	Cable o peine tipo pin
Conexión	Entrada por arriba o por abajo
Normas aplicables	IEC/EN 60898-1

### 2.3.5. Fusibles 1A

El fusible de acción rápida de 1A, que mide 5 mm x 20 mm, es un dispositivo que se utiliza frecuentemente en productos electrónicos y electrodomésticos para evitar daños ocasionados por sobrecorrientes. Su pronta reacción ante una sobrecarga de corriente lo convierte en el ideal para la protección de componentes delicados. Estos fusibles están diseñados para cortocircuitarse al instante si hay sobrecarga, brindando protección exacta en circuitos electrónicos, automotrices y de electrodomésticos [54].

La siguiente tabla presenta las especificaciones técnicas de un fusible estándar de 1 amperio:

Tabla 3: Características técnicas del Fusible 1A

Parámetro	Valor
Corriente nominal	1 A
Tensión nominal	250 V AC / 125 V AC
Tipo de fusión	Rápido (Fast-blow) / Lento (Time-delay)
Tamaño	5 × 20 mm o 6.3 × 32 mm
Material del cuerpo	Vidrio o cerámica
Aplicaciones típicas	Protección de circuitos electrónicos

### 2.3.6. Botón de parada de emergencia

Este pulsador es el que se utiliza típicamente en paradas de emergencia, al presionarlo el botón de emergencia está bloqueado, dejando el contacto abierto y devuelva solo su posición original, gire correctamente a la derecha. [55].

La siguiente tabla presenta las especificaciones técnicas de un botón de Parada de Emergencia:

Tabla 4: Características técnicas del Botón de Parada de Emergencia

Parámetro	Valor
Tipo de operación	Normalmente abierto (NO) / Cerrado (NC)
Tensión nominal	12V DC / 24V DC / 220V AC
Corriente máxima	10 A (dependiendo de la aplicación)
Protección	IP65 (resistente al polvo y agua)
Acción de activación	Presionar para detener, giro para reiniciar
Temperatura de operación	-25°C a +70°C
Aplicaciones típicas	Seguridad industrial, control de emergencia

## Dispositivos de Control

### 2.3.7. Banda transportadora

Las cintas transportadoras se utilizan ampliamente en diversas industrias, como en la fabricación, para el transporte continuo de materiales, en esta herramienta de transporte, se desplazan las cajas de confitería hacia el sistema de agarre para el empaquetado[56].

### 2.3.8. Sensor Capacitivo LJC18A3-B-Z/BX

Detecta objetos metálicos y no metales de hasta 5 mm, funciona desde 5 V, 12 V o 24 V y ofrece un rango más grande con mayor voltaje. Su salida NPN generalmente se abre y envía la señal GND después de encontrar el objeto, es duradero, fácil de integrar en Arduino, PIC o PLC y se usa en Automatización, calculando piezas y detectando niveles en líquidos. Las especificaciones del sensor capacitivo LJC18A3-B-Z/BX se pueden visualizar en la Tabla 2 [57].

Tabla 5: Características técnicas del Sensor Capacitivo LJC18A3-B-Z/BX

<b>Característica</b>	<b>Especificación</b>
Modelo	LJC18A3-B-Z/BX
Tipo de sensor	Capacitivo
Forma del cuerpo	Cilíndrico roscado
Diámetro del cuerpo	18 mm
Tensión de operación	6 – 36 V DC
Distancia de detección	10 mm (ajustable)
Tipo de salida	NPN, normalmente abierto (NO)
Frecuencia de respuesta	25 Hz
Indicador LED	Sí (indicador de detección)
Temperatura de operación	-25°C a 70°C
Protección	Contra inversión de polaridad y sobrecarga
Material del cuerpo	Plástico ABS
Montaje	Con tuerca de fijación en panel
Aplicaciones típicas	Detección de líquidos, papel, vidrio, plástico, madera

### 2.3.9. DC Motor S9D90-90CH SPG

La cabeza con el ángulo recto SC9HC60 envía el par y el motor acelera hasta 90 ° siendo compacto y eficiente para motores de 25 a 90 W. Es ideal en aplicaciones de transporte industrial y automático. [58].

A continuación, se presenta una tabla con las especificaciones técnicas del motor modelo SPG S9D90-90CH:

Tabla 6: Especificaciones técnicas del Motor DC SPG S9D90-90CH

Característica	Especificación
Modelo	SPG S9D90-90CH
Tipo de motor	Motor DC de imán permanente
Potencia nominal	90 W
Tensión nominal	90 V DC
Velocidad nominal	3000 rpm
Corriente nominal	1.2 A
Torque nominal	0.29 Nm
Dirección de rotación	Reversible (CW/CCW)
Peso	1.0 kg
Montaje	Brida cuadrada estándar
Normas aplicables	IEC 60034

### 2.3.10. Variador de Frecuencia

El controlador de velocidad ajustable para motores CC (24 V, 36 V y 90 V) , ofrece regulación continua, retroalimentación estable y alta precisión. Es duradero y efectivo , ideal para aplicaciones industriales, como: empaque, impresión, electrónica, textiles y salud . [59].

### 2.3.11. Relé de 24V

El relé MK2P-I de 8 pines permite conmutar de manera segura los circuitos CC y CA , separe la unidad de los circuitos controlados. Es eficiente, seguro y fácil de integrar en sistemas de automatización y control mediante enchufes estándar de 8 pines. [60].

La siguiente tabla muestra las especificaciones técnicas típicas de un relé electromecánico de 24V DC:

Tabla 7: Características técnicas del Relé 24V DC

<b>Característica</b>	<b>Especificación</b>
Tipo	Relé electromecánico
Voltaje de la bobina	24 V DC
Corriente nominal de la bobina	Aproximadamente 36 mA
Potencia de la bobina	0.9 W
Número de contactos	2 contactos conmutados (DPDT o 2CO)
Capacidad de conmutación	5 A / 250 V AC o 30 V DC
Tensión de aislamiento	1000 V AC (entre contactos y bobina)
Vida mecánica	10 millones de operaciones
Vida eléctrica	100,000 operaciones a carga nominal
Temperatura de operación	-25°C a 55°C
Montaje	Zócalo DIN o PCB
Normas aplicables	IEC 61810, RoHS

### 2.3.12. Relé Winters W4R3L 14 pines

El Relé Winters W4R3L de 220V es un dispositivo mecánico de 14 Pines con 4 Contactos NA y NC , capaz de procesar hasta 5A . Es ideal para automatización y control industrial , lo que permite la conversión de circuitos de baja potencia mediante el uso de señales de control en protección, motor y proceso [61].

En la siguiente tabla presenta las especificaciones técnicas de un : Relé Winters W4R3L 14 Pines

Tabla 8: Características técnicas del Relé Winters W4R3L de 14 pines

<b>Parámetro</b>	<b>Valor</b>
Modelo	Winters W4R3L
Número de pines	14 pines (generalmente 8 contactos)
Voltaje de operación	12V DC / 24V DC
Corriente nominal	10 A (para contactos de cambio)
Tipo de contactos	SPDT o DPDT
Frecuencia de conmutación	10 Hz
Tensión de aislamiento	250 V AC
Material de contactos	Plata o aleación de plata
Durabilidad	1 millón de ciclos de vida mecánica
Temperatura de operación	-40°C a +85°C
Aplicaciones típicas	Control de circuitos, control de motores, automatización industrial

### 2.3.13. Pantalla Touch de 7 pulgadas

La pantalla táctil es un instrumento ideal para incorporar a la Raspberry. Además de ser compatible con varias versiones de Raspberry y sistemas operativos, como Raspbian (la distribución nativa de la Raspberry), también es compatible con otros sistemas, incluyendo Ubuntu y Kali Linux. Su peso ligero y la portabilidad que brinda la pantalla son perfectos para incorporarla a trabajos industriales y personales [62].

La siguiente tabla presenta las especificaciones técnicas de una: Pantalla Táctil de 7 Pulgadas

Tabla 9: Características técnicas de la Pantalla Táctil de 7 pulgadas

<b>Parámetro</b>	<b>Valor</b>
Tamaño de pantalla	7 pulgadas (17.78 cm)
Resolución	1024x600 píxeles
Tipo de pantalla	TFT-LCD / IPS
Pantalla táctil	Capacitiva (multitáctil)
Interfaz de conexión	HDMI / VGA / USB / UART
Tensión de operación	5V DC
Consumo de energía	2-3 W
Ángulo de visión	170°
Aplicaciones típicas	Sistemas HMI, interfaces de usuario

### 2.3.14. Raspberry Pi 4

La Raspberry Pi 4 B presenta avances significativos en cuanto a conectividad y rendimiento para un mini PC de su tipo, caracterizado por sus capacidades multimedia avanzadas, con un procesador ARM Cortex-A72 de cuatro núcleos a 1.5 GHz, lo que nos ofrece una extensa variedad de aplicaciones tecnológicas. Desde estaciones de trabajo para programación hasta servidores de archivos, brindando un aumento significativo en la capacidad de respuesta y la velocidad en comparación con versiones previas. Nos brinda la oportunidad de incorporar sistemas de refrigeración para mejorar su desempeño bajo carga, lo cual la convierte en una opción valiosa para los desarrolladores, los educadores y los aficionados a la tecnología [63].

se presenta en la Tabla 12 las especificaciones técnicas de la Raspberry Pi 4:

Tabla 10: Características técnicas de la Raspberry Pi 4

Parámetro	Valor
Procesador	Broadcom BCM2711, Quad Core Cortex-A72 1.5 GHz
Memoria RAM	2GB, 4GB, 8GB LPDDR4-3200 SDRAM
Conectividad	2 x USB 3.0, 2 x USB 2.0, Gigabit Ethernet, HDMI
Almacenamiento	MicroSD (para SO), soporte para discos externos (USB)
Puertos de video	2 x Micro-HDMI (hasta 4K a 60Hz)
Tensión de operación	5V DC (USB-C)
Wi-Fi / Bluetooth	Wi-Fi 802.11ac, Bluetooth 5.0
Consumo de energía	3.4W (sin periféricos)

### 2.3.15. Componentes Lógicos

### 2.3.16. Raspbian

Raspbian es un sistema operativo que se fundamenta en Linux y que fue creado con el propósito específico de ser utilizado en la placa de desarrollo Raspberry Pi. Esta distribución se fundamenta en Debian y se ajusta a las características de la Raspberry Pi, que tiene recursos y una arquitectura limitados. Raspbian está optimizado para brindar una experiencia superior en los dispositivos. Esto se logra al ofrecer un entorno gráfico de escritorio y numerosas herramientas y aplicaciones que vienen preinstaladas, las cuales facilitan la realización de trabajos simples o proyectos complejos en áreas como el diseño de sistemas embebidos, la automatización y la programación [64].

### 2.2.2.4 Lenguaje de programación Thony

Debido a su sintaxis sencilla y su tipado dinámico (aunque fuerte), Python es uno de los lenguajes más sugeridos para tener un primer acercamiento al ámbito de la programación. Por estas razones, es mucho más accesible que otros lenguajes igualmente populares, como Java, C y C++, aunque este último puede ser muy complicado para los principiantes. En este sentido, Python tiene ciertas similitudes con Lua, que fue presentado en este sitio web hace dos años [65].

Thonny posee una variedad de cualidades fundamentales, tanto de Python como de los IDE. Se puede resaltar la incorporación de un depurador, que asiste al programador en la corrección de errores, y, por otro lado, brinda acceso a la consola nativa de Python. Otras funciones que se pueden resaltar son el inspector de objetos, la visualización de variables,

AST y más. En Fedora Magazine proporcionan una breve introducción, pero adecuada, acerca de cómo utilizar esta aplicación orientada sobre todo a principiantes [65].

### **2.3.17. Programador Visual Studio Code**

Visual Studio Code, creado por Microsoft, fue concebido para ser rápido, discreto y muy configurable. VS Code, a diferencia de otros editores de código, incluidos los IDE completos (entornos integrados de desarrollo), se centra en brindar una experiencia ágil para la edición y escritura de código sin renunciar a las funciones avanzadas. El autocompletado inteligente por medio de IntelliSense, la depuración integrada y el control de versiones usando Git son algunas de sus funcionalidades, las cuales hacen que VS Code sea una alternativa flexible para desarrolladores individuales y para grupos [66].

### **2.3.18. Python**

Python es un lenguaje de programación que tiene gran potencial y resulta fácil de aprender. Posee un sencillo pero eficaz sistema de programación orientado a objetos, así como estructuras de datos de alto nivel que son eficientes. La gramática elegante de Python, su tipo dinámico y su carácter interpretado hacen de este un lenguaje perfecto para la creación rápida de aplicaciones y scripting en numerosas áreas y para la mayoría de plataformas. Python 3.8 fue presentado en su versión beta durante el verano de junio del año 2019. Hay cuatro versiones beta anticipadas y su propósito fundamental es que la comunidad tenga la posibilidad de probar las nuevas funciones e informar sobre los errores que se hayan descubierto [67].

### **2.3.19. Vnc Viewer**

VNC, que significa "Virtual Network Computing" (Computación Virtual en Red, en español), es una aplicación de software libre que funciona con un modelo de cliente-servidor. Este posibilita que se vea y controle la pantalla de la computadora servidor desde una o más computadoras clientes, sin tener en cuenta los sistemas operativos que empleen. Por medio de esta aplicación, se puede manejar el equipo que lleva a cabo el servidor desde el aparato cliente.

Olivetti y Oracle fueron las empresas que desarrollaron este software, haciendo uso del protocolo RFB, el cual es fácil de usar. Después de que su laboratorio fue cerrado, establecieron RealVNC, donde siguen ofreciendo servicios VNC gratuitos y de pago, modificando las características para cumplir con los requerimientos empresariales. Con el VNC, es posible conectarse desde casi cualquier aparato a otro que esté ejecutando la aplicación del servidor, la cual tiene compatibilidad con sistemas como Windows, macOS, Linux, Raspberry Pi, Solaris, HP-UX y AIX. Se le conoce como visor de VNC al programa que permite visualizar la máquina servidor. Este programa está disponible no solo para los sistemas operativos indicados, sino también para dispositivos con Chrome, Android e iOS[68].

### 2.3.20. Diseño experimental

Se optó por el robot colaborativo COBOT EC 63 para diseñar la interfaz HMI, debido a su facilidad de integración en entornos industriales, adaptabilidad y precisión. Este cobot ha demostrado ser una herramienta apropiada para tareas repetitivas como el embalaje, ya que puede interactuar con los trabajadores humanos de manera segura. El sistema se compone de un programa escrito en Python que corre en una Raspberry Pi 4; esta última funciona como "Maestro", mientras que el COBOT lo hace como "esclavo". Este aparato se encarga de establecer la comunicación con el cobot a través del protocolo Modbus TCP/IP, lo que permite que se envíen órdenes y se consigan variables de estado en tiempo real. Con el objetivo de brindar una interacción sencilla con el sistema, se creó una interfaz HMI en una pantalla táctil, a través de la cual es posible que el operador controle y supervise el proceso de embalaje. Esta interfaz presenta datos operativos fundamentales del cobot, entre ellos: la situación actual, la localización de los ejes, las alertas en funcionamiento y los botones de parada de emergencia, inicio y detención.

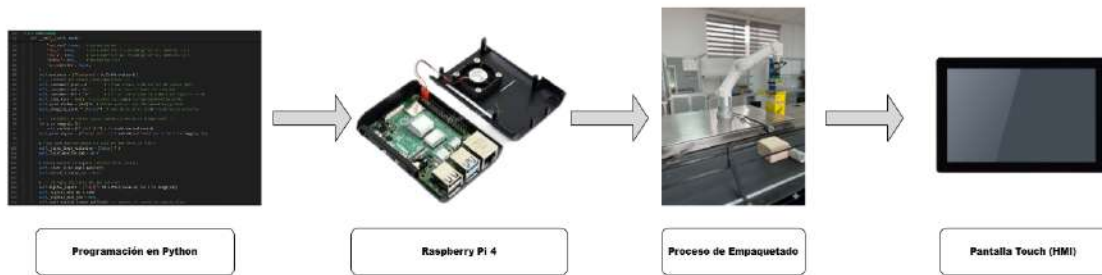


Figura 3: Flujo de trabajo del sistema de Interfaz HMI [Autoría Propia]

### 2.3.21. Diseño e Implementación del Hardware

En el inicio del proceso se encuentra la Raspberry Pi 4, que se encarga de procesar datos y poner en marcha el software de monitoreo, como se observa en el diagrama de la figura 4. Sobre esta, hay una pantalla táctil que permite al usuario interactuar directamente con el sistema, lo que facilita ver el estado del robot y enviar comandos de forma intuitiva.

La Raspberry Pi 4 se enlaza con el COBOT EC 63 a través de un cable Ethernet, que ejecuta las instrucciones recibidas y brinda información en tiempo real. Esta interacción se lleva a cabo mediante el protocolo industrial Modbus TCP/IP, que garantiza una transferencia de datos constante y fiable.

El cobot está situado en un área de trabajo dedicada a las actividades de empaquetado. En este contexto, su habilidad para adaptarse y su precisión permiten que el proceso se automatice sin poner en riesgo la seguridad del operador humano. Para simplificar el mantenimiento, garantizar la estabilidad del sistema en ambientes industriales y disminuir las interferencias, se ha dispuesto de manera cuidadosa la conexión lógica y física de estos elementos.

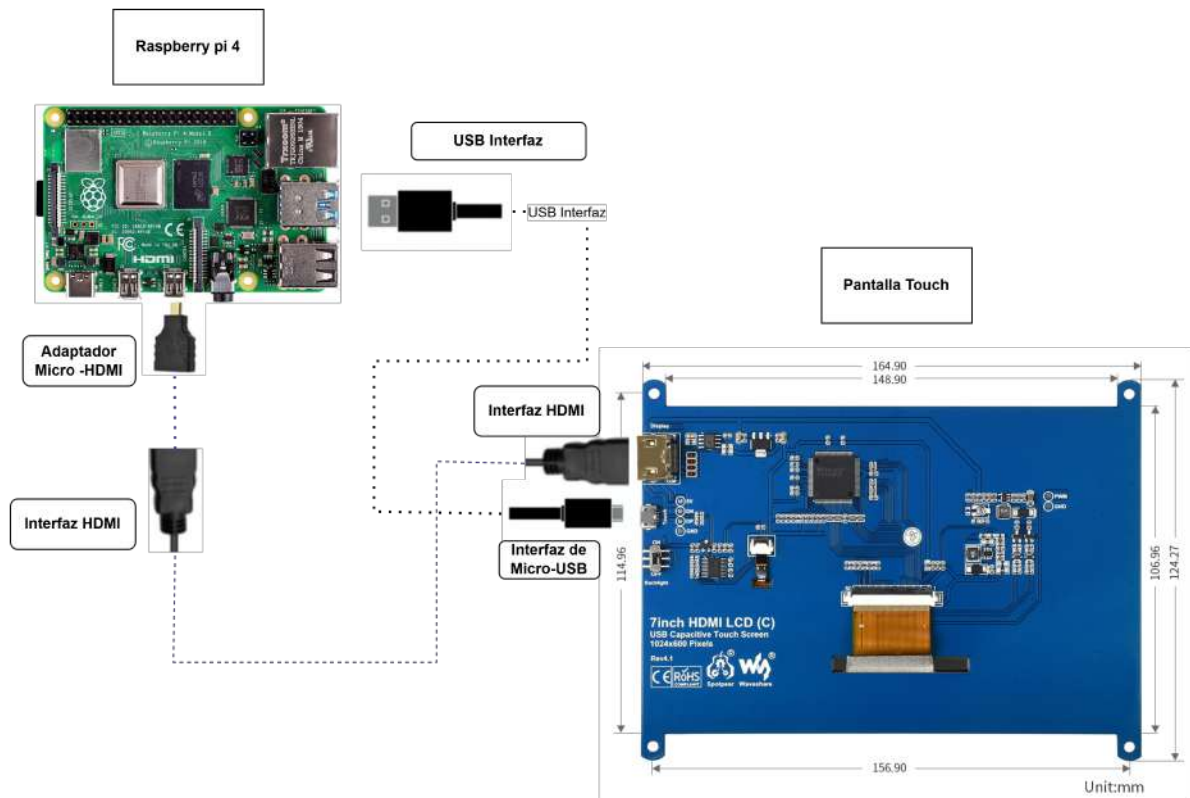


Figura 4: Esquema de Pantalla táctil para Raspberry Pi 4 [Autoría Propia]

En el sistema diseñado, la **Raspberry Pi 4** se conecta a la **pantalla táctil HDMI de 7 pulgadas** a través de dos conexiones primordiales:

- **HDMI (imagen):** Se emplea un adaptador micro-HDMI junto con un cable HDMI para transmitir la señal de video desde la Raspberry Pi hacia la pantalla.
- **USB (interacción táctil):** Se logra la interacción táctil debido a que se conecta la pantalla con la Raspberry Pi mediante un cable USB de tipo A a micro-USB..

Esta disposición facilita la visualización de la interfaz HMI creada en Python y potencia la interacción táctil del usuario, al mismo tiempo que se conecta con el sistema de vigilancia y control en tiempo real del **cobot EC63**, utilizando **Modbus TCP/IP**.

### 2.3.22. Diagrama Electrico del Proceso de Empaquetado

El sistema utiliza un COBOT EC63 y una cinta transportadora para empaquetar, con funciones de seguridad que incluyen un botón de parada de emergencia, control lógico mediante un controlador EC63 y protección eléctrica a través de disyuntores y relés. La combinación del control lógico (de voltaje bajo) y el control de potencia (120 VCA y 220 VCA) garantiza la seguridad y confiabilidad en el funcionamiento del dispositivo.

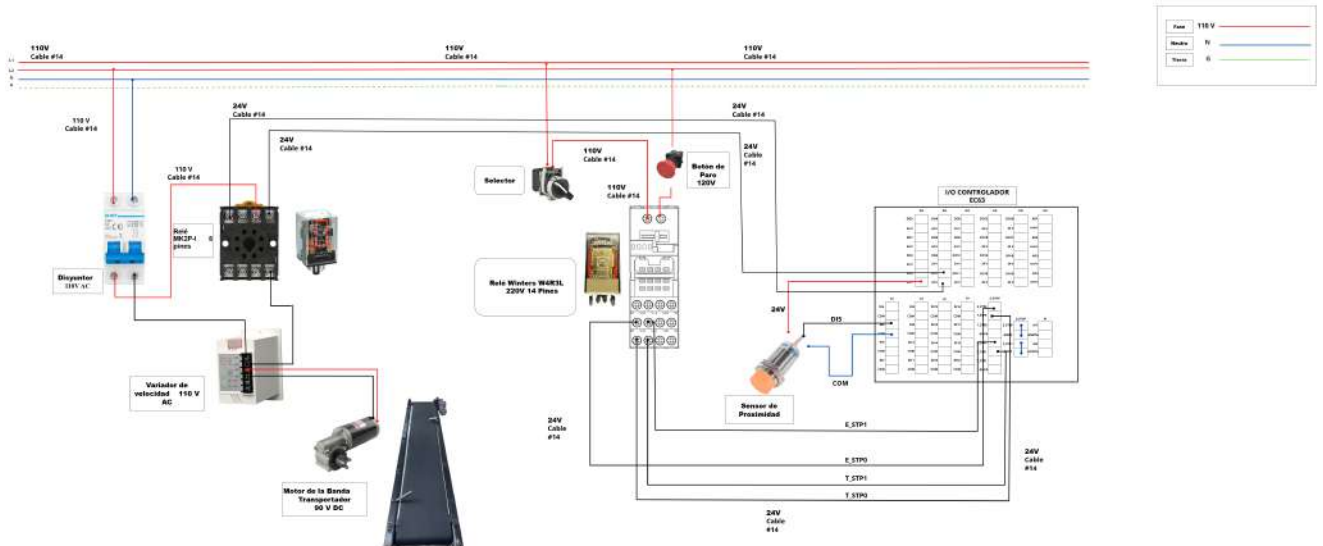


Figura 5: Conexiones del Proceso de Empaquetado .

*Fuente: Autoría propia.*

### 2.3.23. Comunicación del Proceso de Empaquetado

La Figura 6 ilustra cómo se comunica el proceso de empaquetado, que incluye un COBOT, una Raspberry Pi 4, sensores y una banda transportadora conectados a través del protocolo Modbus TCP/IP por medio de Ethernet. La función del robot y de otros dispositivos se coordinan a través de una interfaz táctil HMI y un programa creado en Python, los cuales emplean un módulo de entrada/salida (I/O) y un conmutador Ethernet.

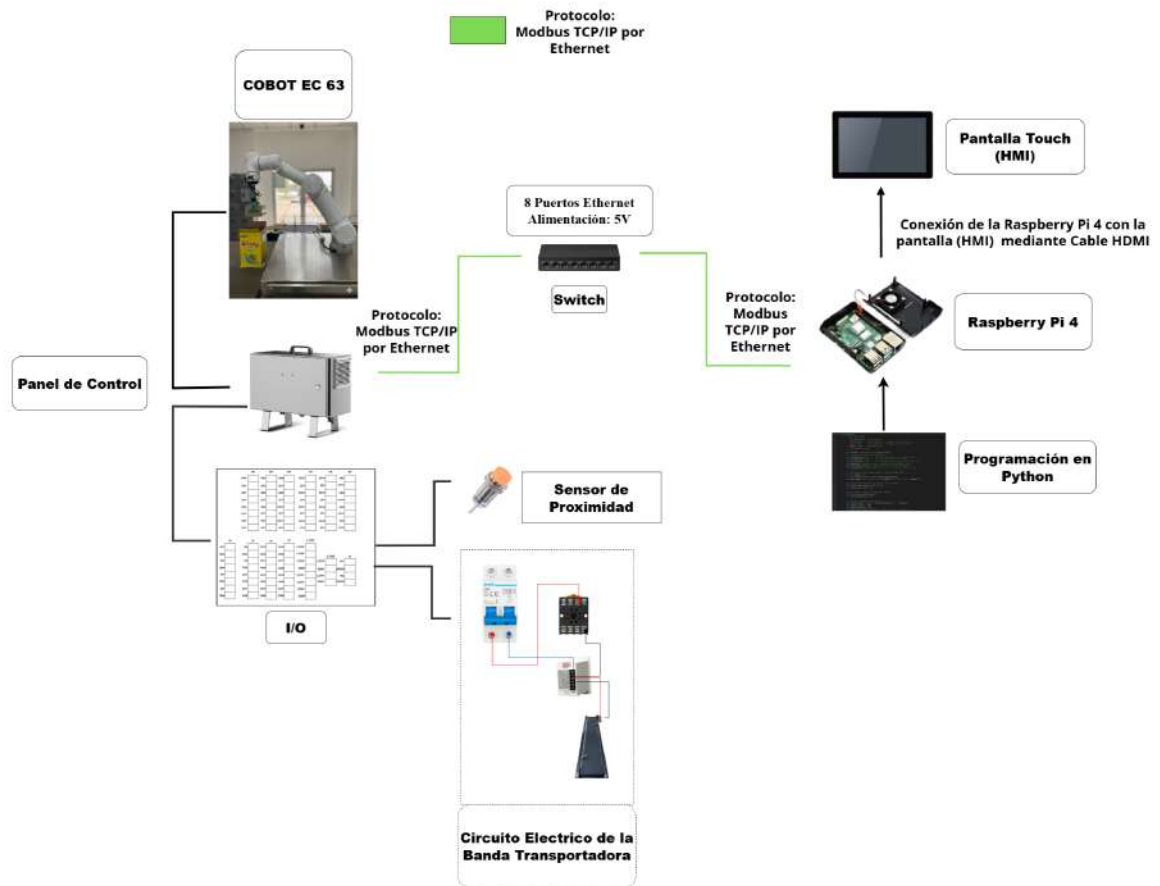


Figura 6: Comunicación del Proceso de Empaquetado *Fuente: Autoría propia.*

### 2.3.24. Diagrama Topológico del Proceso de Empaquetado

La topología del proceso de empaquetado se ilustra en la figura, donde el PC, la Raspberry Pi 4 (HMI) y el controlador del cobot EC63 están enlazados a través de un switch en una red Ethernet. El protocolo Modbus TCP/IP es el que se utiliza para la comunicación, lo que hace posible controlar y supervisar el sistema desde el PC usando VNC Viewer.

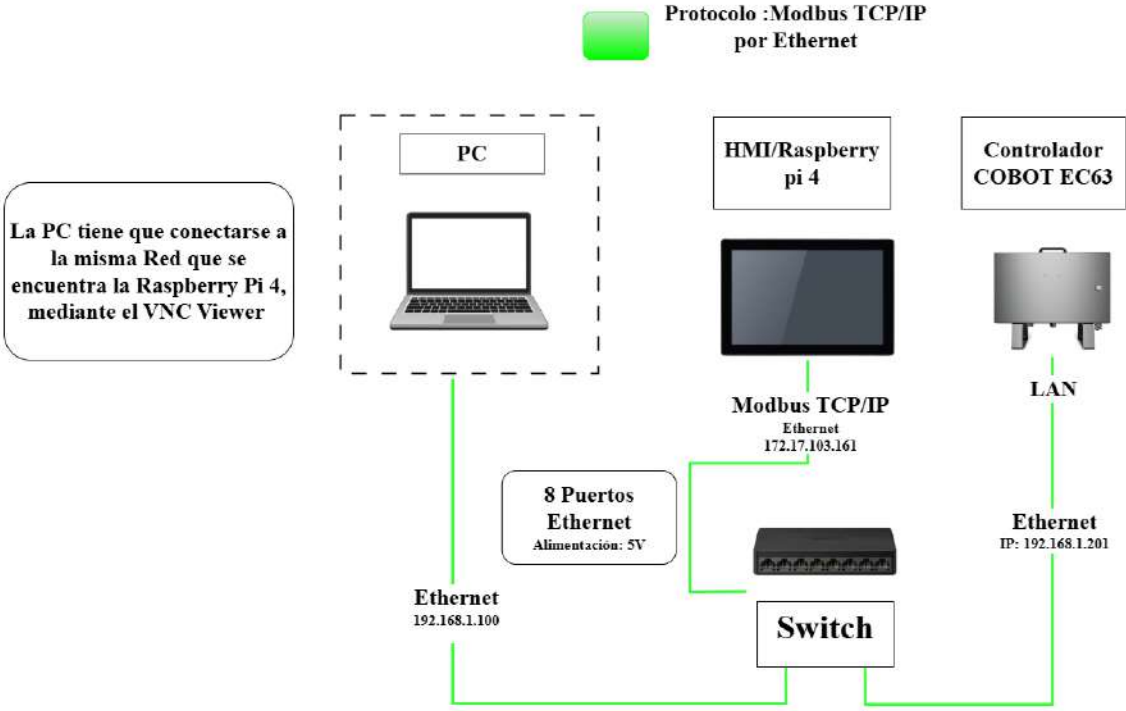


Figura 7: Topología del Proceso de Empaquetado .  
*Fuente: Autoría propia.*

## 2.4. Diseño de la HMI para el sistema de empaquetado

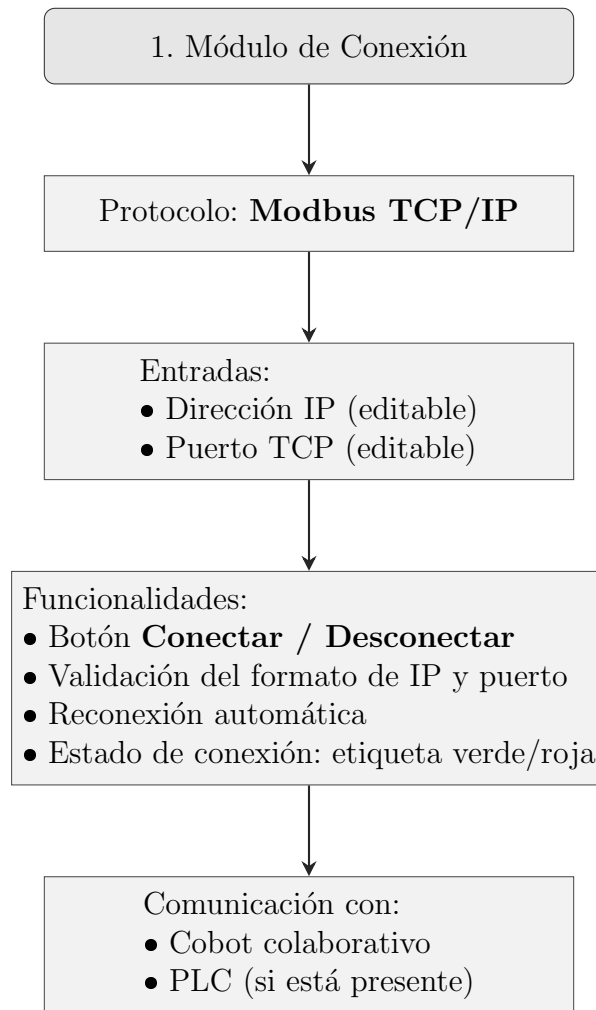
### 2.4.1. Implementación de la Interfaz HMI

EL HMI desarrollado es el centro visual y operativo del sistema, con el propósito en el enfoque del operador, priorizando claridad, seguridad y eficiencia. Este se organiza con módulos principales para su conexión, monitoreo en tiempo real, mantenimiento, acciones generales y gestión de alarmas. En lugar de que se visualice nada más una ventana única con exceso de información, se emplea una serie de pestañas, lo que facilita la navegación y reduce la carga cognitiva, además se incluyen indicadores permanentes de estado de red.

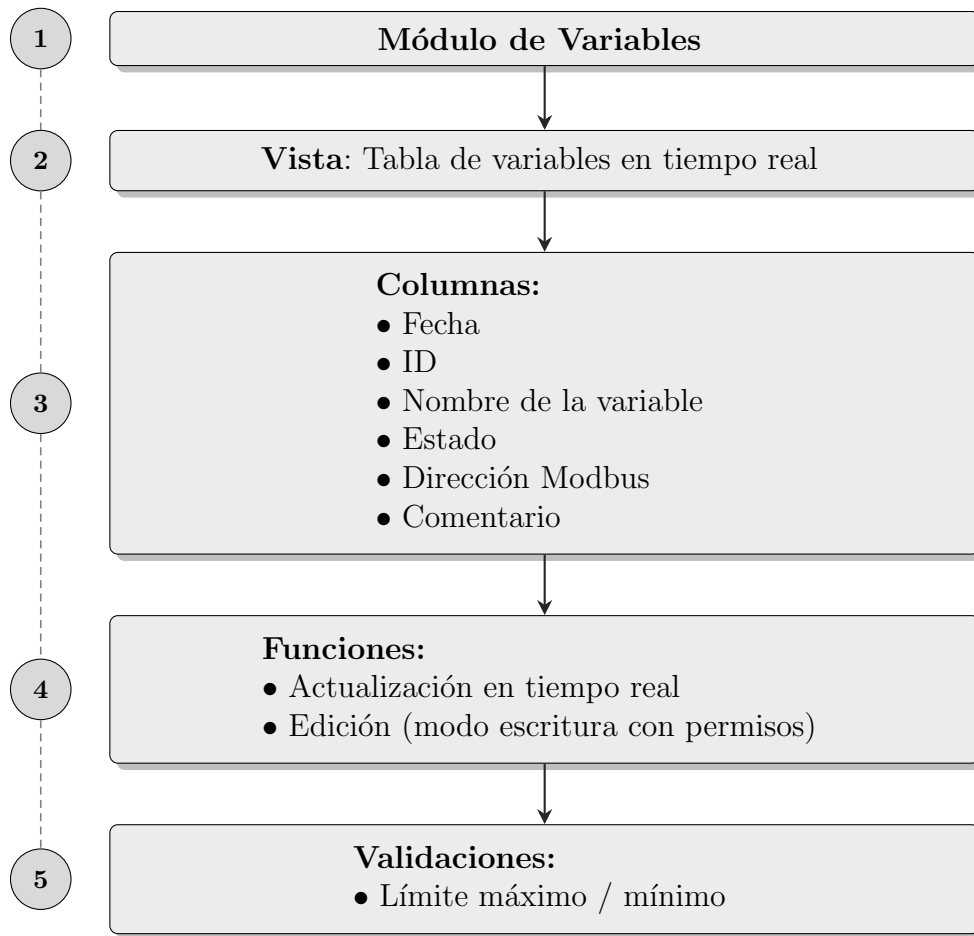
Tabla 11: Resumen de funcionalidades por módulo de la HMI

Módulo	Funcionalidad principal	Características destacadas
<b>Conexión</b>	A través de Modbus TCP/IP, establecer y comprobar la comunicación entre el cobot y el HMI.	IP establecida (192.168.1.201), puerto que se puede configurar, verificación de conexión, indicador de estado en tiempo real y reconexión.
<b>Variables</b>	Modificar y supervisar las variables del sistema en tiempo real.	Tabla dinámica que contiene observaciones, nombre, dirección, estado y valor. Facilita la redacción que valida rangos.
<b>Alarmas</b>	Detectar, describir y gestionar circunstancias críticas del sistema.	Cambio de color en el estado, activación en tiempo real, historial, botón para cerrar y prioridad de atención.
<b>Mantenimiento</b>	Ejecutar tareas de diagnóstico y control manuales para verificar el sistema.	Controles para limpiar registros, probar los actuadores y restablecer.
<b>Parámetros</b>	Determinar los valores operativos del sistema, según si se trata de un escenario de producción o de prueba.	Alteración de los límites, las unidades y el tiempo. Incorpora las validaciones y la asistencia contextual como parámetro.
<b>Diseño visual</b>	Garantizar que el entorno operativo sea legible, fácil de usar y claro.	Una jerarquía estructural, colores que contrastan, botones grandes, una tipografía fácil de leer y un sistema de navegación basado en tareas.

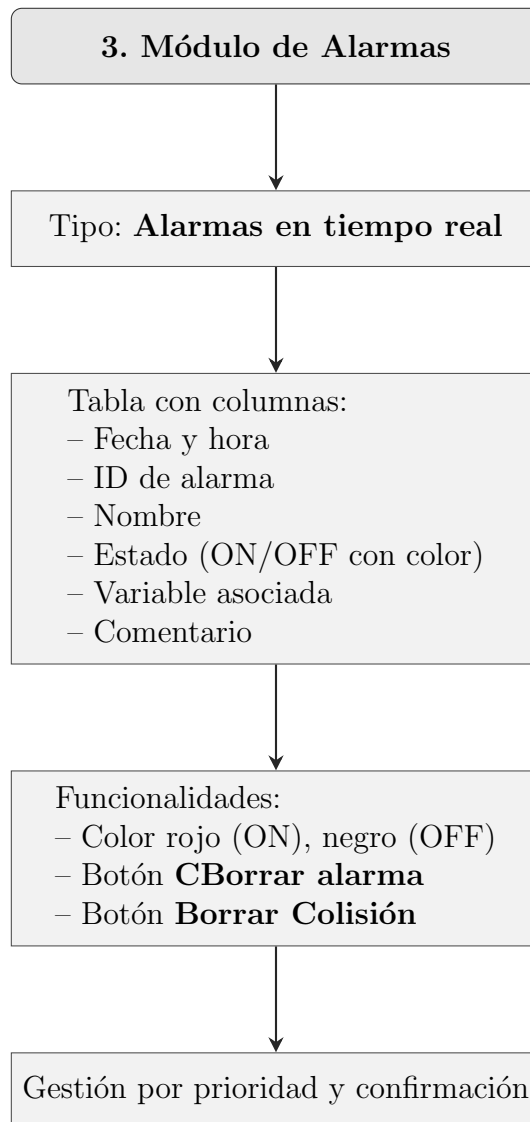
### 2.4.2. Módulo de conexión



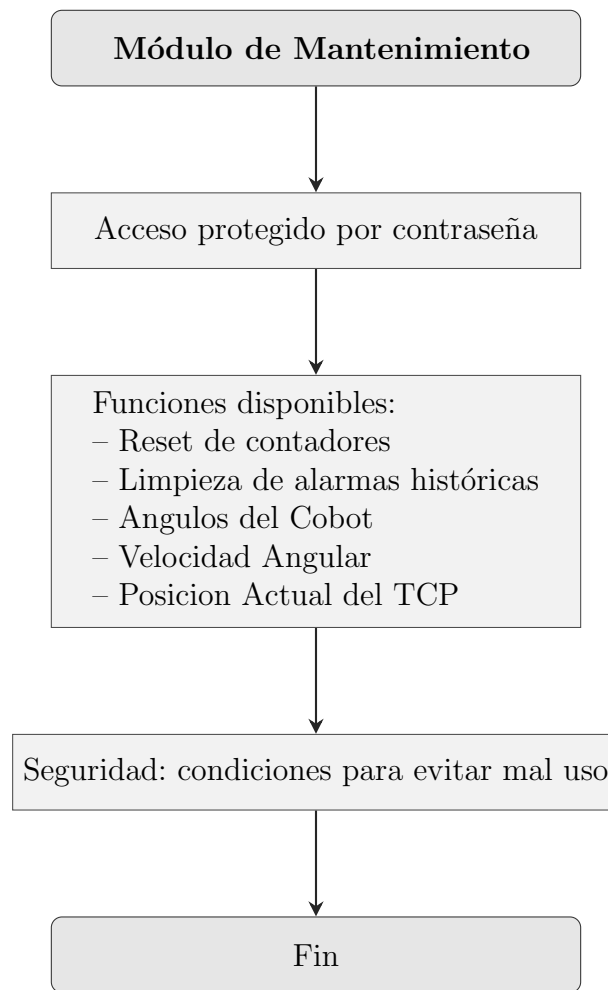
### 2.4.3. Módulo de Variables



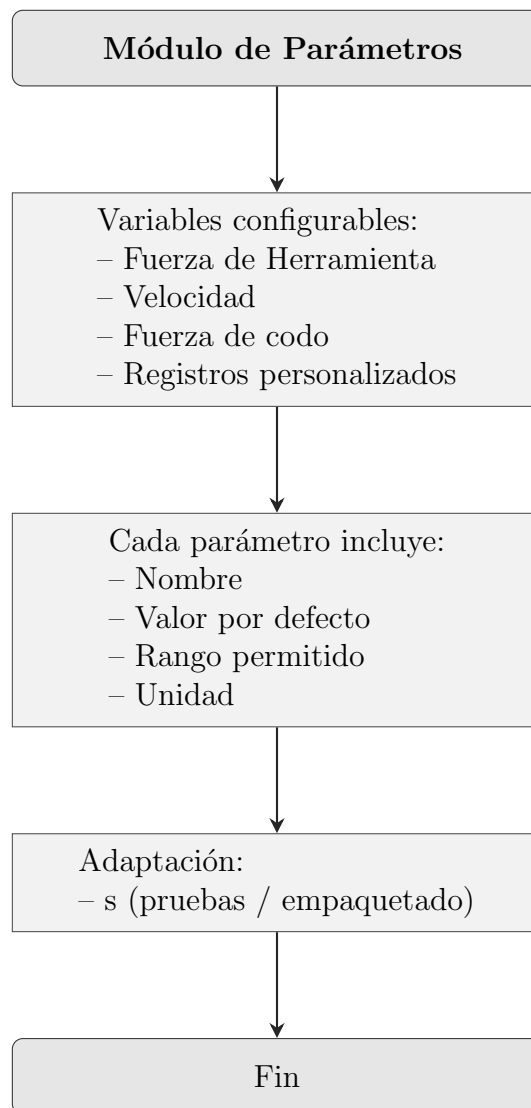
#### 2.4.4. Módulo de Alarmas



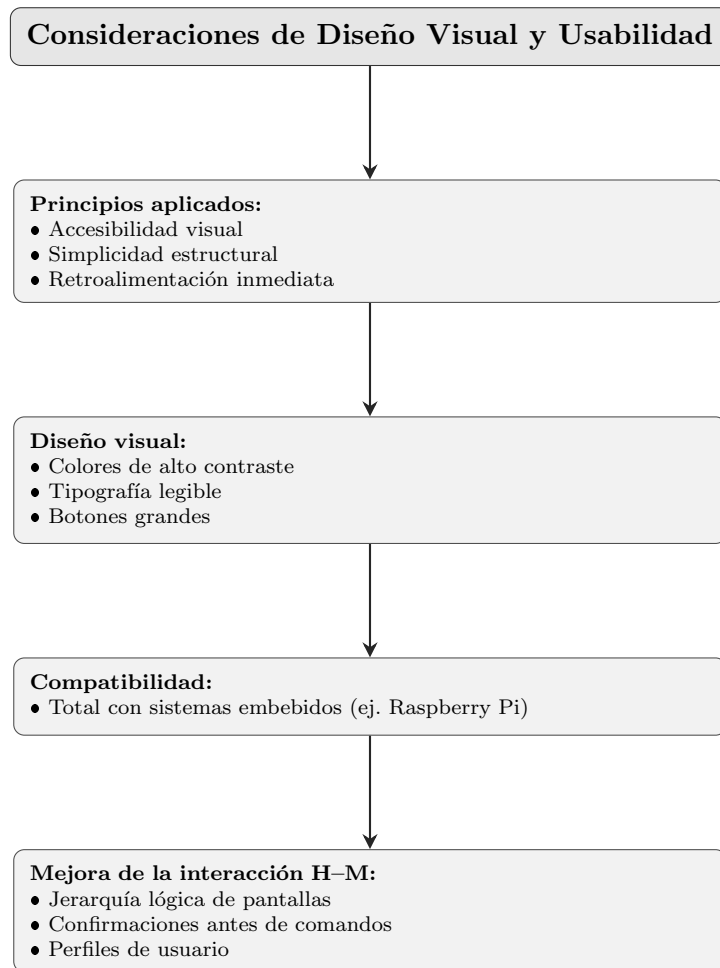
#### 2.4.5. Módulo de Mantenimiento



#### 2.4.6. Módulo de Parámetros



## 2.4.7. Consideraciones de Diseño Visual y Usabilidad



### 3. Resultados

En la sección anterior se detalló los componentes de la propuesta necesarios para el diseño e implementación de la interfaz hombre-máquina (HMI) desarrollada para la supervisión y control de un sistema colaborativo de empaquetado, haciendo uso del protocolo Modbus TCP/IP y una plataforma basada en Raspberry Pi 4. La HMI fue concebida con un enfoque modular, integrando funcionalidades clave como visualización de variables, monitoreo de alarmas, gestión de mantenimiento, y configuración remota de parámetros, con el objetivo de facilitar la interacción entre el operador humano y el sistema robotizado. En este capítulo se presentan los resultados obtenidos durante la fase de validación funcional, en la que se evaluó el comportamiento del sistema en un entorno de pruebas. Se analizaron aspectos como la estabilidad de la comunicación Modbus, la velocidad de actualización de las variables en la interfaz, la capacidad de respuesta ante eventos críticos, y la usabilidad general del sistema.

#### 3.1. Requerimientos del Sistema HMI

Estos requerimientos definen las características básicas que debe cumplir la interfaz de seguimiento y control:

Tabla 12: Requerimientos funcionales del sistema HMI-COBOT

Requerimiento	Descripción
R1	Comunicación mediante el protocolo <b>Modbus TCP/IP</b> para el intercambio confiable de datos.
R2	La HMI debe posibilitar la <b>monitorización y el control en tiempo real</b> del cobot EC63.
R3	La interfaz tiene que exhibir <b>alarma, estados de conexión y conteo de piezas.</b>
R4	El sistema debe estar desarrollado con costos bajos <b>Python</b> y hardware accesible.
R5	La comunicación tiene que sostener una <b>latencia menor a 10 ms.</b>

### 3.2. Restricciones del Sistema HMI

Se presentan a continuación las principales limitaciones consideradas en el diseño y validación del sistema HMI:

Tabla 13: Restricciones técnicas del sistema HMI-COBOT

Restricción	Descripción
RT1	Solo se utiliza un COBOT el <b>EC63</b> el cual se validó la comunicación mediante pruebas de laboratorio.
RT2	El desarrollo se limitó al uso del protocolo Modbus TCP/IP y Socket, sin implementar otro tipo de protocolos industriales como OPC UA, EtherCAT o Profinet..
RT3	El sistema fue ensayado solamente con <b>hardware de bajo costo</b> (Raspberry Pi y Touch).
RT4	La validación tuvo lugar en un <b>entorno simulado</b> , no en una instalación industrial.
RT5	Las pruebas de carga o congestión extrema no se incluyeron en el análisis de red, que solamente abarcó las capturas de Wireshark y las pruebas PING.
RT6	No se incluyó <b>visión artificial</b> ni algoritmos de aprendizaje automático.

### 3.3. Pruebas y resultados obtenidos en la red de comunicación conectada a los equipos mediante Modbus TCP/IP

Se utilizó el comando PING en el terminal de la PC, esta herramienta nos envía paquetes ICMP (Internet Control Message Protocol) a la dirección IP del equipo destinado, y mide su tiempo en milisegundos. En este caso, realizamos pruebas de PING hacia las direcciones 192.168.1.210 (COBOT EC63) y 172.17.103.161 (Raspberry Pi 4), correspondientes a los equipos configurados dentro de la red bajo nuestro protocolo Modbus TCP/IP.

Tabla 14: Ping a 192.168.1.210 — Detalle de respuestas

Paquete	IP	Bytes	Tiempo (ms)	TTL
1	192.168.1.210	32	3	64
2	192.168.1.210	32	2	64
3	192.168.1.210	32	2	64
4	192.168.1.210	32	2	64

Tabla 15: Ping a 192.168.1.210 — Estadísticas

<b>Métrica</b>	<b>Valor</b>
Enviados	4
Recibidos	4
Perdidos	0
Pérdida (%)	0%
Mínimo (ms)	2
Máximo (ms)	3
Media (ms)	2

Tabla 16: Ping a 172.17.103.161 — Detalle de respuestas

<b>Paquete</b>	<b>IP</b>	<b>Bytes</b>	<b>Tiempo (ms)</b>	<b>TTL</b>
1	172.17.103.161	32	2	64
2	172.17.103.161	32	2	64
3	172.17.103.161	32	6	64
4	172.17.103.161	32	5	64

Tabla 17: Ping a 172.17.103.161 — Estadísticas

<b>Métrica</b>	<b>Valor</b>
Enviados	4
Recibidos	4
Perdidos	0
Pérdida (%)	0%
Mínimo (ms)	2
Máximo (ms)	6
Media (ms)	3

En este caso, las pruebas de ping se realizaron con las direcciones de 192.168.1.210 y 172.17.161 correspondientes al equipo configurado en el protocolo Modbus TCP/IP con el objetivo de confirmar la disponibilidad, su estabilidad y respuesta de comunicación.

### 3.4. Análisis y resultados del tráfico de datos y latencias en la red Ethernet bajo protocolo Modbus TCP/IP para el sistema de monitoreo y control

En el análisis del tráfico de red se empleó el software Wireshark, el cual permitió interpretar la estructura de los protocolos involucrados cuando el HMI genera solicitudes bajo Modbus TCP/IP, para llevar a cabo estos resultados se tomaron 50 pruebas, el software cuenta de 3 ventanas en la Figura 5 se observa en interfaz principal.

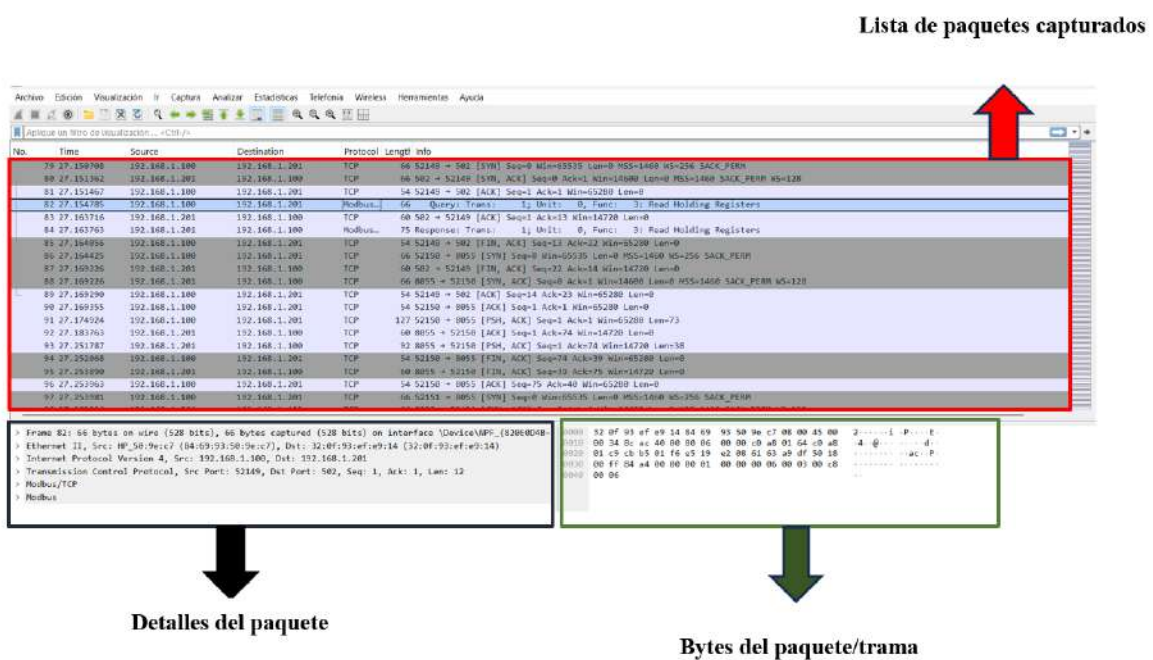


Figura 8: Estructura del software Wireshark.  
Fuente: Autoría propia.

En el panel principal de paquetes capturados de la red en tiempo real se muestra una tabla con las siguientes columnas:

- **No:** La cantidad de paquetes capturados en forma secuencial.
- **Time:** El tiempo en que se capturó el paquete.
- **Source:** Es la dirección IP de donde se obtuvo el paquete.
- **Destination:** Es la ubicación a la que se manda el paquete.
- **Protocol:** Es el protocolo de comunicación que se emplea para enviar el paquete.
- **Length:** Se refiere al tamaño del paquete.
- **Info:** Datos del paquete.

### 3.5. Resultados de análisis a nivel de aplicación (Modbus/TCP)

La información resumida en la Tabla 18 muestra un desempeño confiable, con una duración continua cercana a los 27.7 segundos, latencias comprendidas entre 0.638 ms y 3.659 ms, y un promedio de 1.3 ms en las transacciones registradas. No se evidenciaron pérdidas de respuesta ni códigos de excepción Modbus, lo que confirma la consistencia de los registros. Únicamente se detectaron variaciones en registros de carácter dinámico, mientras que no se observaron retransmisiones ni tráfico paralelo que interfiriera con la comunicación principal.

Métrica	Registros	Análisis con datos de registros
Duración de prueba continua	200–205, 240–243, 250–253	~27.7 s de lecturas continuas (desde la primera hasta la última lectura compartida).
Latencia mínima observada	240–243	<b>0.638 ms</b> (lectura de este bloque).
Latencia máxima observada	200–205	<b>3.659 ms</b> (lectura de este bloque).
Latencia promedio (5 lecturas)	200–205, 240–243, 250–253	≈ <b>1.3 ms</b> (promedio de los tiempos observados en los bloques analizados).
Tasa de pérdida de respuesta	200–205, 240–243, 250–253	<b>0 %</b> (todas las solicitudes tuvieron respuesta válida).
Errores Modbus (Exception Codes)	200–205, 240–243, 250–253	Ninguno (todas las respuestas fueron coherentes).
Consistencia de valores	200–205	Constantes en todas las lecturas: 200=2166, 201=57807, 202=6714, 203=58364, 204=8039, 205=6235.
Consistencia de valores	240–243	En todas las lecturas, los valores se mantienen constantes: 240=15, 241=60144, 242=0 y 243=0.
Variaciones de valores	250–253	250=0, 251=0, 252=0, 253=12 → Únicamente el 253 cambia (contando/estado dinámico).
Retransmisiones Modbus	200–205, 240–243, 250–253	No se ha observado ninguna.
Impacto de tráfico adicional	—	En la parte analizada no se registró tráfico distinto al Modbus/TCP.

Tabla 18: Resultados de análisis a nivel de aplicación (Modbus/TCP).

#### Desempeño de comunicaciones HMI–COBOT

Las lecturas de Modbus/TCP ocurrieron en una prueba continua de 27.7 s. La latencia observada se mantuvo de 0.638 ms a 3.659 ms con un promedio de 1.3 ms. No se registraron la pérdida de respuestas, retransmisiones ni códigos de excepción Modbus. Estos resultados muestran un canal de comunicación confiable con un retardo despreciable durante la exploración del ciclo HMI y COBOT, por lo que la comunicación no limita el rendimiento del sistema.

#### Consistencia de datos y correspondencia con la HMI

Durante la prueba, los registros 200–205 y 240–243 permanecieron constantes, confirmando la estabilidad de los parámetros/estados que se muestran en el HMI. Por otro lado, el registro 250–253 mostró solo variaciones en el 253, que consiste en su uso como un contador/estado dinámico en la pantalla. La ausencia de tráfico paralelo adicional en el segmento descarta la interrupción externa de la comunicación.

### 3.6. Resultados de análisis a nivel de transporte (TCP/IP)

Se tomo 50 pruebas para poder realizar el análisis del transporte TCP/IP ,empleando el software Wireshark:

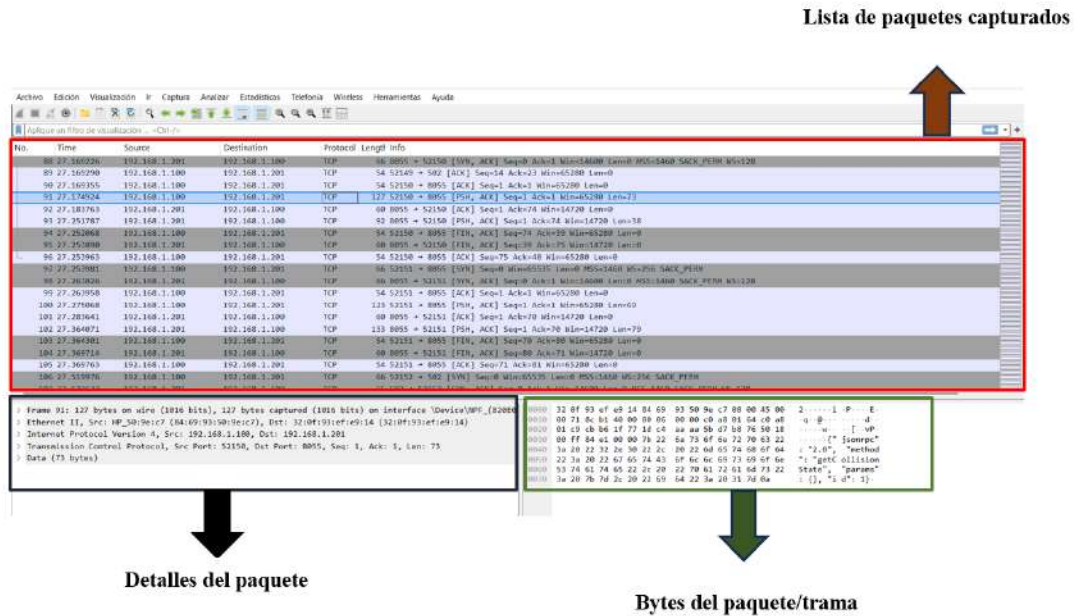


Figura 9: Datos capturados por el software Wireshark del transporte (TCP/IP).  
Fuente: Autoría propia.

La Tabla 19 muestra el rendimiento de la capa de transporte TCP se refleja considerando elementos como la duración de las conexiones, las latencias máximas y mínimas, las reconexiones, el tráfico adicional y las pérdidas potenciales. Los hallazgos muestran que las sesiones TCP no tuvieron errores de protocolo ni retransmisiones y se mantuvieron estables, a pesar de que se encontraron varias reconexiones relacionadas con el empleo de puertos efímeros y la presencia de tráfico paralelo en el puerto 8055. La comunicación principal de Modbus no fue afectada por este último.

Métrica	Conexiones / Frames	Análisis con datos observados
Duración de sesiones TCP	Varias (52179–52277, 49390–49396)	en respuesta a la petición Modbus. Sesiones de corta duración ( $\approx 27.7$ segundos en total de prueba), cada consulta activa y desactiva una conexión temporal hacia el puerto 502.
Latencia mínima observada	Frames 240–243	<b>0.638 ms</b> como respuesta a la petición de Modbus.
Latencia máxima observada	Frames 200–205	<b>3.659 ms</b> en la sección de lectura inicial.
Latencia promedio	Conexiones totales	$\approx$ <b>1.3 ms</b> , conservado en todas las sesiones revisadas.
Tasa de pérdida de paquetes	General	<b>0 %</b> , no se observó pérdida ni en las retransmisiones forzadas ni en las capturas.
Retransmisiones destacadas	—	no se notaron retransmisiones TCP en el transcurso de la captura.
Tiempo de reconexión tras fallo	Puertos efímeros (52186, 52198, 52270, 52277, etc.)	Las reconexiones se realizan inmediatamente al iniciar nuevas sesiones, sin interrupciones de larga duración.
Errores de protocolo	—	No se encontraron fallos TCP ni cierres de sesión inusuales.
Impacto de tráfico adicional	Puerto 8055 en modo paralelo	Hubo tráfico simultáneo en el puerto 8055, pero no afectó para nada al canal principal de Modbus/TCP.

Tabla 19: Resultados de análisis a nivel de transporte (TCP/IP).

## Desempeño de transporte (TCP/IP)

Hubo varias sesiones de TCP por debajo de la ventana de 27.7 s, correspondiente al sondeo de la HMI. Las latencias de ida y vuelta oscilaron entre 0.638 ms y 3.659 ms, son su promedio cerca de 1.3 ms según los tiempos que se muestran en el nivel de aplicación antibus/TCP. No se registró pérdida de paquetes ni retransmisiones y no se detectaron errores de protocolo. Las reconexiones inmediatas se identificaron usando puertos efímeros ( 52186, 52198, 52270, 52277), propias del cliente, sin descansos a largo plazo. Aunque se observó tráfico paralelo en el puerto 8055, no afectó la comunicación más importante de Modbus.

### Implicancias para la HMI del proceso de empaquetado

La capa de transporte tiene estabilidad y retardo despreciable en el tiempo de escaneo de la HMI y el ciclo del COBOT, por lo que la red no limita el refresco de pantallas ni la detección de eventos. La falta de pérdidas y retransmisiones garantiza la integridad de los contadores y estados utilizados en los KPIs (Key Performance Indicators) de producción.

### 3.7. Análisis de una petición enviada desde el HMI al COBOT EC63

Tabla 20: Solicitudes Modbus/TCP (FC03)

Registro	Dirección (IP:Puerto)	Rol	(bytes)	Tipo	Latencia (ms)
200–205	192.168.1.100:53878 → 192.168.1.201:502	Cliente	12	Solicitud	5,972
240–243	192.168.1.100:53875 → 192.168.1.201:502	Cliente	12	Solicitud	5,051
200–205	192.168.1.100:53869 → 192.168.1.201:502	Cliente	12	Solicitud	8,120
200–205	192.168.1.100:53883 → 192.168.1.201:502	Cliente	12	Solicitud	3,715
200–205	192.168.1.100:53896 → 192.168.1.201:502	Cliente	12	Solicitud	0,780
250–253	192.168.1.100:53889 → 192.168.1.201:502	Cliente	12	Solicitud	9,711
240–243	192.168.1.100:53888 → 192.168.1.201:502	Cliente	12	Solicitud	9,803
240–243	192.168.1.100:53901 → 192.168.1.201:502	Cliente	12	Solicitud	0,823
240	192.168.1.100:58417 → 192.168.1.201:502	Cliente	12	Solicitud	0,922
240	192.168.1.100:58417 → 192.168.1.201:502	Cliente	12	Solicitud	0,429

Tabla 21: Solicitudes Modbus/TCP (FC03)

Registro	Dirección (IP:Puerto)	Rol	(bytes)	Tipo	Observaciones
500	192.168.1.100	Cliente	12	Solicitud	colisión_sensibilidad (%)
500	192.168.1.100	Cliente	12	Solicitud	colisión_sensibilidad (%)
500	192.168.1.100	Cliente	12	Solicitud	colisión_sensibilidad (%)
500	192.168.1.100	Cliente	12	Solicitud	colisión_sensibilidad (%)
500	192.168.1.100	Cliente	12	Solicitud	colisión_sensibilidad (%)

#### Tabla de *Solicitud* (cliente → servidor)

Esta tabla evidencia que el sistema **interroga de forma cíclica** al controlador usando **FC03: Read Holding Registers**.

- **Evidencia principal:** Las direcciones (IP:puerto) y el rol *Cliente* indican que los tramas van desde el HMI hacia el controlador.
- **Cuáles son las consultas más comunes:**
  - **Reg. 200–205:** Posiciones de las articulaciones J1 hasta J6 (llamadas "ángulos J1–J6).
  - **Reg. 240–243:**entradas digitales (X) ON/OFF.
  - **Reg. 250–253:** salidas digitales (Y) ON/OFF.
  - **Reg. 500:** Colisión identificada .

Tabla 22: Respuestas Modbus/TCP (FC03)

Registro	Dirección (IP:Puerto)	Rol	(bytes)	Tipo	Latencia (ms)
200–205	192.168.1.201:502 → 192.168.1.100:53878	Servidor	21	Respuesta	5,972
240–243	192.168.1.201:502 → 192.168.1.100:53875	Servidor	17	Respuesta	5,051
200–205	192.168.1.201:502 → 192.168.1.100:53869	Servidor	21	Respuesta	8,120
200–205	192.168.1.201:502 → 192.168.1.100:53883	Servidor	21	Respuesta	3,715
240–243	192.168.1.201:502 → 192.168.1.100:53888	Servidor	17	Respuesta	9,803
250–253	192.168.1.201:502 → 192.168.1.100:53889	Servidor	17	Respuesta	9,711
200–205	192.168.1.201:502 → 192.168.1.100:53896	Servidor	21	Respuesta	0,780
240–243	192.168.1.201:502 → 192.168.1.100:53901	Servidor	17	Respuesta	0,823
240	192.168.1.201:502 → 192.168.1.100:58417	Servidor	11	Respuesta	0,922
240	192.168.1.201:502 → 192.168.1.100:58417	Servidor	11	Respuesta	0,429

### Tabla de *Respuesta* (servidor → cliente)

La tabla muestra que el controlador **devuelve los datos solicitados en tiempos de milisegundos**.

- **Evidencia principal:** rol *Servidor* y direcciones opuestas respecto a la petición.
- **Desempeño:** Se han registrado tiempos de latencia de alrededor de  $\approx 0,4\text{--}6$  ms, sin congestión importante.
- **Consistencia de tamaño:** Un mayor número de bytes implica más registros devueltos, lo que confirma el correcto funcionamiento del protocolo.
- **Lectura sencilla:** .<sup>El</sup> controlador responde casi de inmediato con los valores cuando el PC pregunta acerca de ángulos, entradas y salidas.”

## 4. Conclusiones y Recomendaciones

### 4.1. Conclusiones

La interfaz desarrollada demostró ser intuitiva, estable y funcional, permitiendo una visualización clara de las variables críticas del proceso y asegurando un control seguro y eficiente del COBOT.

Se estudió a detalle el protocolo MODBUS TCP/IP para garantizar una conexión fiable entre la HMI y el COBOT, logrando una comunicación eficiente y segura que aseguró un flujo constante de datos y un control adecuado del robot durante las pruebas.

Las herramientas implementadas permitieron anticipar y mitigar posibles fallos durante la interacción, garantizando un funcionamiento estable y seguro del sistema.

El sistema de alarmas en tiempo real permitió detectar oportunamente condiciones anormales y fallos de comunicación, mejorando la capacidad de respuesta del operador y asegurando la continuidad operativa durante las pruebas.

os resultados evidenciaron su eficacia, estabilidad y fiabilidad bajo condiciones controladas, confirmando su adecuado desempeño y su potencial para una futura implementación en un entorno industrial real.

## 4.2. Recomendaciones

De los resultados y la experiencia que se aprendió durante el desarrollo de esta investigación, Se han propuesto las siguientes sugerencias para trabajos futuros:

### Recomendaciones

- Integrar una cámara en la que podamos implementar visualización artificial para setear los puntos guardados para un proceso específico.
- Elevar la seguridad de la red mediante división por ACL/VLAN, uso de credenciales cambiantes y, cuando corresponda, encapsular Modbus/TCP dentro de túneles TLS/VPN.
- Implementar un control automatizado para la velocidad de la banda transportadora a través de la HMI desarrollada, con el fin de garantizar un flujo constante y una mayor eficiencia en el proceso de empaquetado.
- Elaborar manuales y guías rápidas para el entrenamiento; para mantener un enfoque de bajo costo, utilizar software libre, control de versiones con Git y paquetes de instalación replicables.

La tesis demuestra que una HMI de bajo costo basada en Modbus TCP/IP permite supervisar y controlar en tiempo real un cobot de empaquetado con comunicaciones estables y latencias de milisegundos, sin pérdidas ni retransmisiones. Estos resultados validan su uso en planta y abren líneas futuras de integración de nuevos procesos Industriales.

## Referencias

- [1] Mladen Šverko y Tihana Galinac Grbac, “Automated HMI design as a custom feature in industrial SCADA systems,” *Procedia Computer Science*, vol. 232(3), pp. 1789–1798, 2024. DOI: 10.1016/j.procs.2024.02.001.
- [2] Peng Zhang, “Chapter 13 – Human–Machine Interfaces,” en *Advanced Industrial Control Technology*, William Andrew Publishing, pp. 527–555, 2010. DOI: 10.1016/B978-1-4377-7807-6.10013-0.
- [3] Pat Flynn, “Innovative Solutions: Exploring the Benefits of Packaging Automation Systems,” Rapid Packaging Blog, 18 abril 2024. [En línea]. Disponible en: <https://blog.rapidpackaging.com/innovative-solutions-exploring-the-benefits-of-packaging-automation-systems/>. [Consultado el 21 de agosto de 2025].
- [4] S. Liao, R. Candell y M. Hany, “Research on the acceptance of collaborative robots for alleviating low acceptance: measures and methods to improve acceptance of cobots,” *Ergonomics*, vol. 66, no. 5, Artículo 103455, 2023. DOI: 10.1016/j.ergon.2023.103455.
- [5] J. Pelegrí, “Aplicaciones robóticas para un empaquetado más eficiente,” Blog de Universal Robots, s. f. [En línea]. Disponible en: <https://www.universal-robots.com/es/blog/aplicaciones-roboticas-para-un-empaquetado-mas-eficiente/>.
- [6] Fainca Group, “Partner autorizado de la marca KUKA en Ecuador,” Noticias Fainca Group, 6 de diciembre de 2021. [En línea]. Disponible en: <https://fainca-group.com/2021/12/06/partner-autorizado-de-la-marca-kuka-en-ecuador/>.
- [7] E. Flores-García, J. C. Quezada-Quezada, R. V. H. Calderón-Medina y M. Á. Castañeda-Bautista, “Propuesta de automatización del proceso industrial de empaquetado de platos de poliestireno expandido empleando PLC y HMI,” *Boletín Científico INVESTIGIUM de la Escuela Superior de Tizayuca*, vol. 3, no. 6, 2018. DOI: 10.29057/est.v3i6.2954. Disponible en: <https://repository.uaeh.edu.mx/revistas/index.php/investigium/article/download/2954/2976?inline=1>.
- [8] P. A. Arellano Jiménez y J. J. Mocha Astudillo, “Diseño e implementación de un sistema automatizado de supervisión de los indicadores de producción en el área de extrusión de la planta Plastigama – Wavin,” proyecto integrador (tesis de grado), Escuela Superior Politécnica del Litoral (ESPOL), Facultad de Ingeniería en Electricidad y Computación, Guayaquil, Ecuador, 2023. [En línea]. Disponible en: <https://dspace.espol.edu.ec/bitstream/123456789/60553/1/T-114064-INGE-2248.pdf>.
- [9] R. González Duque, *Python para todos*, Creative Commons Reconocimiento 2.5 España. [En línea]. Disponible en: <https://persoal.citius.usc.es/eva.cernadas/informaticaparacientificos/material/libros/Python%20para%20todos.pdf>.
- [10] Maple Systems, “Guía del protocolo Modbus,” Maple Systems, s. f. [En línea]. Disponible en: [https://maplesystems.com/modbus-protocol/?srsrtid=AfmB0oqiaOgh\\_c3IGkR4lnV0Z2huIa7-kt6H5LRh3plrwrEQM3SIxx0m](https://maplesystems.com/modbus-protocol/?srsrtid=AfmB0oqiaOgh_c3IGkR4lnV0Z2huIa7-kt6H5LRh3plrwrEQM3SIxx0m).

- [11] J. Andrés Matute Caguana, “Monitoreo y operación de un sistema industrial de servo posicionamiento: HMI basada en Python y OPC UA,” Jul-2023. [En línea]. Disponible en: <http://dspace.ups.edu.ec/handle/123456789/25734>.
- [12] Jae Hyun Hwang Cárdenas y Jaime Andrés Iñiguez Avila, “Sistema de comunicación inalámbrico y HMI basado en PYTHON para la operación y monitoreo remoto de un motor trifásico de inducción,” Jul-2023. [En línea]. Disponible en: <http://dspace.ups.edu.ec/handle/123456789/25739>.
- [13] Vistazo, “Ecuador avanza hacia la automatización, el uso de robots es más habitual en las industrias,” 28 noviembre 2023. [En línea]. Disponible en: <https://www.vistazo.com/enfoque/ecuador-avanza-automatizacion-uso-de-robots-es-habitual-en-industrias-EY6405215>.
- [14] Sergio Rodolfo Torres Gualsaqui, “Implementación de un sistema HMI mediante aplicaciones de código abierto para el control y monitoreo de un sistema dinámico real,” tesis de pregrado, Universidad Técnica del Norte, 11-Nov-2021. [En línea]. Disponible en: <http://repositorio.utn.edu.ec/handle/123456789/11651>.
- [15] E. Flores-García, J. C. Quezada-Quezada, I. Bautista-Mejía y R. V. H. Calderón-Medina, “Implementación de PLC-HMI para control y monitoreo en la automatización de una máquina empleadora en la industria refresquera para empaquetado de Tetra Briks,” *Boletín Científico INVESTIGIUM de la Escuela Superior de Tizayuca*, no. 9, pp. 12–23, 2019. Disponible en: <https://repository.uaeh.edu.mx/revistas/index.php/investigium/article/download/4016/6247/20426>.
- [16] A. K. Hansen, V. Villani, A. Pupa, and A. H. Lassen, “Introducing Novice Operators to Collaborative Robots: A Hands-On Approach for Learning and Training,” *IEEE Transactions on Automation Science and Engineering*, vol. 22, pp. 3933–3946, 2024. doi: 10.1109/TASE.2024.3403709. [Online]. Available: [https://vbn.aau.dk/ws/portalfiles/portal/768535668/Introducing\\_Novice\\_Operators\\_to\\_Collaborative\\_Robots\\_A\\_Hands-On\\_Approach\\_for\\_Learning\\_and\\_Training.pdf](https://vbn.aau.dk/ws/portalfiles/portal/768535668/Introducing_Novice_Operators_to_Collaborative_Robots_A_Hands-On_Approach_for_Learning_and_Training.pdf)
- [17] S. El Zaatari, M. Marei, W. Li, and Z. Usman, “Cobot programming for collaborative industrial tasks: An overview,” *Robotics and Autonomous Systems*, vol. 116, pp. 162–180, 2019. [Online]. Available: <https://doi.org/10.1016/j.robot.2019.03.003>
- [18] C. Brambilla, “Biomechanical monitoring of neurotypical and ASD workers during industrial human-robot collaborative assembly tasks,” *Applied Ergonomics*, vol. 121, 104557, 2025. DOI: 10.1016/j.apergo.2025.104557.
- [19] F. Vicentini, “Terminology in safety of collaborative robotics,” *Robotics and Computer-Integrated Manufacturing*, vol. 63, art. 101921, 2020. doi: 10.1016/j.rcim.2019.101921.
- [20] A. T. Eyam, W. M. Mohammed, and J. L. Martínez Lastra, “Emotion-driven analysis and control of human-robot interactions in collaborative applications,” *Sensors*, vol. 21, no. 14, p. 4626, 2021. doi: 10.3390/s21144626.
- [21] P. Fager, F. Sgarbossa, and M. Calzavara, “Cost modelling of onboard cobot-supported item sorting in a picking system,” *International Journal of Production Research*, vol. 59, no. 11, pp. 3260–3284, 2021. doi: 10.1080/00207543.2020.1854484.

- [22] S. Bermúdez i Badia, P. A. Silva, D. Branco, A. Pinto, C. Carvalho, P. Menezes, J. Almeida, and A. Pilacinski, “Virtual Reality for Safe Testing and Development in Collaborative Robotics: Challenges and Perspectives,” *Electronics*, vol. 11, no. 11, p. 1726, 2022. doi: 10.3390/electronics11111726.
- [23] G. Boschetti, M. Faccio, and I. Granata, “Human-Centered Design for Productivity and Safety in Collaborative Robots Cells: A New Methodological Approach,” *Electronics*, vol. 12, no. 1, p. 167, 2023. doi: 10.3390/electronics12010167.
- [24] M. Faccio, I. Granata, and R. Minto, “Task allocation model for human–robot collaboration with variable cobot speed,” *Journal of Intelligent Manufacturing*, vol. 35, pp. 793–806, 2024. doi: 10.1007/s10845-023-02073-9.
- [25] J. Habib, C. Prentice, P. Thaichon, and R. Gururajan, “Developing a cobotic model for employee and customer engagement,” *Tourism and Hospitality Industry 2024, Congress Proceedings*, vol. 27, pp. 195–203, 2024. doi: [https://www.researchgate.net/publication/382575514\\_DEVELOPING\\_A\\_COBOTIC\\_MODEL\\_FOR](https://www.researchgate.net/publication/382575514_DEVELOPING_A_COBOTIC_MODEL_FOR)
- [26] M. G. Antonelli, P. B. Zobel, C. Manes, E. Mattei, and N. Stamponi, “Emotional Intelligence for the Decision-Making Process of Trajectories in Collaborative Robotics,” *Machines*, vol. 12, no. 2, p. 113, 2024. doi: <https://www.mdpi.com/2075-1702/12/2/113>.
- [27] M. Capponi, R. Gervasi, L. Mastrogiacomo, and F. Franceschini, “Assembly complexity and physiological response in human-robot collaboration: Insights from a preliminary experimental analysis,” *Robotics and Computer-Integrated Manufacturing*, vol. 85, p. 102669, 2024. doi: ScienceDirect.
- [28] E. Mendez, O. Ochoa, D. Olivera-Guzman, V. H. Soto-Herrera, J. A. Luna-Sánchez, C. Lucas-Dophe, E. Lugo-del-Real, I. N. Ayala-Garcia, M. A. Perez, and A. González, “Integration of Deep Learning and Collaborative Robot for Assembly Tasks,” *Applied Sciences*, vol. 14, no. 2, p. 839, 2024. doi: <https://www.mdpi.com/2076-3417/14/2/839>.
- [29] M. Abbes and G. Poisson, “Geometric Approach for Inverse Kinematics of the FANUC CRX Collaborative Robot,” *Robotics*, vol. 13, p. 91, 2024. doi: <https://www.mdpi.com/2218-6581/13/6/91>.
- [30] V. O’Keeffe, R. Jang, K. Manning, R. Trott, S. Howard, A.-L. Hordacre, and J. Spoehr, “Forming a view: a human factors case study of augmented reality collaboration in assembly,” *Australian Industrial Transformation Institute, Flinders University, Adelaide, Australia, 202X*. [Online]. Available: [file:///C:/Users/User/Downloads/Forming%20a%20view%20a%20human%20factors%20case%20study%20of%20augmented%20reality%20collaboration%20in%20assembly%20\(1\).pdf](file:///C:/Users/User/Downloads/Forming%20a%20view%20a%20human%20factors%20case%20study%20of%20augmented%20reality%20collaboration%20in%20assembly%20(1).pdf)
- [31] R. Shah, A. S. A. Doss, and N. Lakshmaiva, “Advancements in AI-enhanced collaborative robotics: towards safer, smarter, and human-centric industrial automation,” *Journal of Industrial Automation*, 2025. doi: ScienceDirect.
- [32] M. Riar, M. Weber, J. Ebert, and B. Morschheuser, “Can Gamification Foster Trust-Building in Human–Robot Collaboration? An Experiment in Virtual Reality,” *Information Systems Frontiers*, early access, 2024. doi: 10.1007/s10796-024-10573-z.

- [33] F. Nunnari, D. Tsovaltzi, M. L. Nicora, S. Beyrodt, P. Prajod, L. Chehayeb, I. Brdar, A. Delle Fave, L. Negri, E. André, P. Gebhard, and M. Malosio, “Socially interactive industrial robots: a PAD model of flow for emotional co-regulation,” *Frontiers in Robotics and AI*, vol. 11, Art. no. 1418677, 2024. doi: 10.3389/frobt.2024.1418677.
- [34] N. Yuviler-Gavish, O. Muati, and B. H. Gabbay, “How performing several task segments simultaneously affects cognitive flexibility training in an assembly task,” *Interactive Learning Environments*, 2023. doi: 10.1080/10494820.2023.2170420.
- [35] D. Bibbo, G. Corvini, M. Schmid, S. Ranaldi y S. Conforto, “The Impact of Human-Robot Collaboration Levels on Postural Stability During Working Tasks Performed While Standing: Experimental Study,” *JMIR Human Factors*, vol. 12, e64892, 2025. doi: 10.2196/64892.
- [36] R. Calvo and P. Gil, “Evaluation of Collaborative Robot Sustainable Integration in Manufacturing Assembly by Using Process Time Savings,” *Materials*, vol. 15, no. 611, 2022. doi: 10.3390/ma15020611.
- [37] A. Borboni, R. Pagani, S. Sandrini, G. Carbone, and N. Pellegrini, “Role of Reference Frames for a Safe Human–Robot Interaction,” *Sensors*, vol. 23, p. 5762, 2023. doi: 10.3390/s23125762.
- [38] CADE Cobots, “Empaquetadora y paletizadora automática,” [En línea]. Disponible en: <https://cadecobots.com/aplicaciones-robots-colaborativos/empaquetadora-y-paletizadora-automatica/>.
- [39] I. Gracia Ruiz, “Manipulación robótica colaborativa mediante el Sistema Operativo Robótico,” Trabajo Fin de Grado, Escuela de Ingeniería y Arquitectura, Universidad de Zaragoza, Zaragoza, España, 2019. [En línea]. Disponible en: <https://zaguan.unizar.es/record/87329/files/TAZ-TFG-2019-4518.pdf>
- [40] Sicma21, “sicma21,” 11 octubre 2021. [En línea]. Disponible en: <https://www.sicma21.com/que-es-un-hmi-y-como-funciona/>.
- [41] ARC Advisory Group, “How ISA-101 Lifecycle Standard Improves Operator Effectiveness with Display Design and Lifecycle Management,” 2023. [En línea]. Disponible en: <https://www.arcweb.com/industry-best-practices/how-isa-101-lifecycle-standard-improves-operator-effectiveness-display>.
- [42] International Society of Automation (ISA), “ISA-101 Series of Standards: Enhancing Human-Machine Interface Design for Safer, Smarter Automation,” mayo 2025. [En línea]. Disponible en: <https://www.isa.org/standards-and-publications/isa-standards/isa-101-standards>.
- [43] IBM, “Cómo funcionan los sockets,” IBM i 7.5.0 Documentación. [En línea]. Disponible en: <https://www.ibm.com/docs/es/i/7.5.0?topic=programming-how-sockets-work>.
- [44] Modbus-IDA, “MODBUS Messaging on TCP/IP Implementation Guide V1.0b,” [En línea]. Disponible en: [https://scadahacker.com/library/Documents/ICS\\_Protocols/Modbus.org%20-%20Modbus%20Messaging%20on%20TCP-IP%20Implementation%20Guide%20v1.0b.pdf](https://scadahacker.com/library/Documents/ICS_Protocols/Modbus.org%20-%20Modbus%20Messaging%20on%20TCP-IP%20Implementation%20Guide%20v1.0b.pdf).

- [45] Modbus-IDA, “MODBUS Messaging on TCP/IP Implementation Guide V1.0b,” 2006. [Online]. Available: [https://scadahacker.com/library/Documents/ICS\\_Protocols/Modbus.org%20-%20Modbus%20Messaging%20on%20TCP-IP%20Implementation%20Guide%20v1.0b.pdf](https://scadahacker.com/library/Documents/ICS_Protocols/Modbus.org%20-%20Modbus%20Messaging%20on%20TCP-IP%20Implementation%20Guide%20v1.0b.pdf)
- [46] A. F. Castro, “Cuerva,” 12 julio 2024. [En línea]. Disponible en: <https://cuervaenenergia.com/es/comunidad/construccion-e-instalacion/protocolo-modbus/>.
- [47] Ditel – Diseños y Tecnología S.A., “Modbus RTU: El protocolo industrial de comunicación por excelencia,” 29 abril 2024. [En línea]. Disponible en: <https://www.ditel.es/modbus-rtu-el-protocolo-industrial-de-comunicacion-por-excelencia/>.
- [48] O. Weis, “Puerto serie virtual,” 3 octubre 2024. [En línea]. Disponible en: <https://www.virtual-serial-port.org/articles/modbus-ascii-guide/#what>.
- [49] Logicbus, “Protocolos de comunicación: MODBUS TCP/IP,” Nov. 30, 2024. [Online]. Available: <https://www.logicbus.com.mx/blog/modbus-tcp-ip/>
- [50] CADE Cobots, “¿Qué es un cobot y cuáles son sus ventajas?,” [Online]. Available: <https://cadecobots.com/que-es-un-cobot/>
- [51] Elite Robots, “EC63 Cobot,” EC Series – Elite Robots, [En línea]. Disponible en: <https://es.eliterobots.com/cobots/ec63>. Especificaciones principales: carga útil de 3 kg, alcance de 624 mm, repetibilidad de  $\pm 0.02$  mm, rating IP54, comunicación Modbus RTU entre otras; instalación flexible en mesa o línea de producción. [Consultado el 21 de agosto de 2025].
- [52] D. Robotics. [En línea]. Disponible en: <https://en.dh-robotics.com/product/pge>.
- [53] CEITSA Electromecatrónica, “Interruptor termomagnético Riel DIN, 2 polos, 16 A, curva C, 3 kA – Chint eB 2P C16 (186306),” [En línea]. Disponible en: <https://ceitsa.com.mx/tienda/interruptor-termomagnetico-2-polos-16-amperes-curva-c-3ka-eb-chint-eb-2p-c16r/>.
- [54] MAXITEC, “maxitec,” 2024. [En línea]. Disponible en: <https://www.maxitec.com.ec/steren-fusible-de-la-250vca-fame-1p/>
- [55] N. M. SAC, “naylampmechatronics,” 2023. [En línea]. Disponible en: <https://naylampmechatronics.com/accesorios-y-prototipado/660-pulsador-de-parada-de-emergencia-22mm.html>.
- [56] eurotransis.com, “¿Qué es una cinta transportadora? Principios de funcionamiento,” [En línea]. Disponible en: <https://eurotransis.com/que-es-una-cinta-transportadora-principios-de-funcionamiento/>. [Último acceso: 28 abril 2025].
- [57] A. Electronics, “avelectronics,” 2024. [En línea]. Disponible en: <https://avelectronics.cc/producto/sensor-proximidad-capacitivo-ljc18a3-b-z-bx/>.
- [58] S. motor, “spg,” 2024. [En línea]. Disponib..le en: <https://spgmotor.net/product-literature/>.

- [59] U. Co., “mercadolibre,” 2024. [En línea]. Disponible en: <https://www.ubuy.ec/es/productjp/1PXG0C40-speed-controller-dc-motor-speed-controller-24v-90v-variable-input-ac-110v-adjustable-small-high-prec>.
- [60] R. ECUADOR, “Roboticsec,” 2023. [En línea]. Disponible en: <https://roboticsec.com/producto/rele-24v-30a-1-canal/>
- [61] E. S. Luis, “electros1,” 2023. [En línea]. Disponible en: [https://electros1.mercadoshops.com.ar/MLA-1124108757-rele-4-inversores-na-nc-5amper-bobina-12vac-winters-\\_JM](https://electros1.mercadoshops.com.ar/MLA-1124108757-rele-4-inversores-na-nc-5amper-bobina-12vac-winters-_JM)
- [62] Raspberry Pi Chile, “Pantalla Touch 7” para Raspberry Pi,” [En línea]. Disponible en: <https://raspberrypi.cl/producto/pantalla-touch-7-para-raspberry-pi/>. [Último acceso: 28 abril 2025].
- [63] J. Pastor, 25 julio 2019. [En línea]. Disponible en: <https://www.xataka.com/ordenadores/raspberry-pi-4-analisis-caracteristicas-precio-especificaciones>.
- [64] “Raspberry Pi,” [En línea]. Disponible en: <https://www.raspberrypi.com/software/>.
- [65] E. Medina, “Thonny es un IDE sencillo que te ayudará a aprender Python,” MuyLinux, 19 de febrero de 2018. [En línea]. Disponible en: <https://www.muylinux.com/2018/02/19/thonny-ide-python/>.
- [66] G. C. Cuadrado, “openwebinars,” 22 julio 2022. [En línea]. Disponible en: <https://openwebinars.net/blog/que-es-visual-studio-code-y-que-ventajas-ofrece/>.
- [67] cursosaula21.com, “¿Qué es Python?,” [En línea]. Disponible en: <https://www.cursosaula21.com/que-es-python/>. [Último acceso: 28 abril 2025].
- [68] Geeknetic, “¿Qué es VNC y para qué sirve?,” [En línea]. Disponible en: <https://www.geeknetic.es/VNC/que-es-y-para-que-sirve/>. [Último acceso: 28 abril

## Anexos

### A. Ventana de ejecución de la HMI para el proceso de empaquetado



Figura 10: Pantalla 1 .  
Fuente: Autoría propia.

## B. Ventana de valores operativos de la HMI

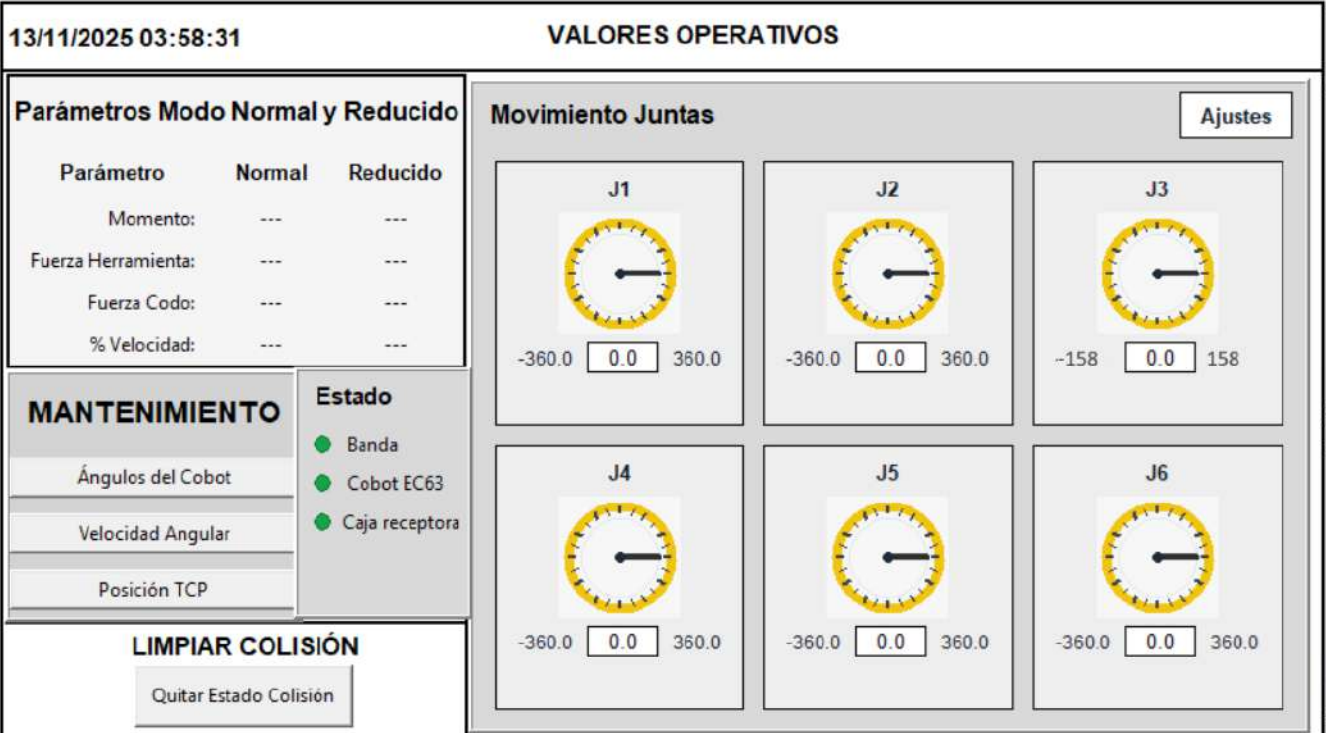


Figura 11: Pantalla 2 .  
Fuente: Autoría propia.

**C. Validación de el sistema HMI en un entorno de prueba que simule el proceso de empaquetado industrial**



Figura 12: Posición inicial del COBOT .  
*Fuente: Autoría propia.*

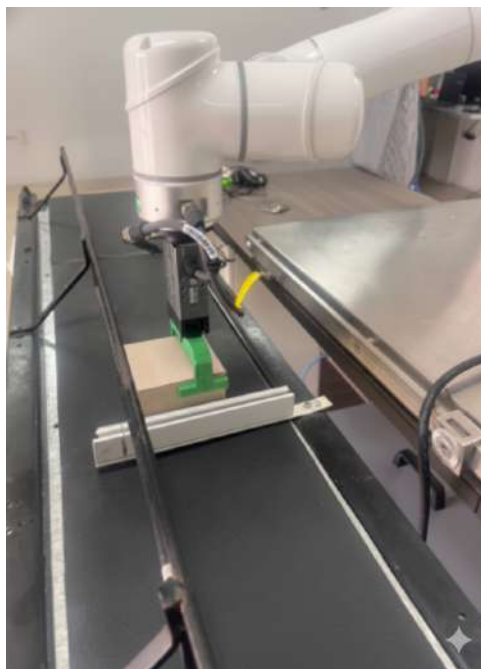


Figura 13: Aproximación al punto de toma .  
*Fuente: Autoría propia.*



Figura 14: Traslado de la pieza sobre la banda .  
*Fuente: Autoría propia.*



Figura 15: Aproximación al área de empaque .  
*Fuente: Autoría propia.*



Figura 16: Posicionamiento sobre la caja .  
*Fuente: Autoría propia.*



Figura 17: Alineación final .  
*Fuente: Autoría propia.*



Figura 18: Pieza asegurada .  
*Fuente: Autoría propia.*

# Reporte de coincidencias



Figura 19: Reporte Compilatio.  
Fuente: Autoría propia.