

HMI para Supervisión de Línea de Empaquetado Asistida por Cobot

HMI for Supervision of a Cobot-Assisted Packaging Line

Kevin Aníbal Chuchuca Minga

Universidad Estatal de la Península de Santa Elena.

Facultad de Sistemas y Telecomunicaciones. La Libertad – Santa Elena, Ecuador.

ORCID: 0009-0005-2553-5006

Email: kchuchuca9@gmail.com

Linier Ignacio Ramos Rambay

Universidad Estatal de la Península de Santa Elena.

Facultad de Sistemas y Telecomunicaciones. La Libertad – Santa Elena, Ecuador.

ORCID: 0009-0003-6844-6180

Email: dwill177.wr@gmail.com

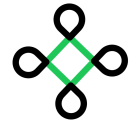
Este artículo presenta el desarrollo e implementación de una interfaz hombre-máquina (HMI) modular para supervisar variables críticas en procesos de empaquetado con robótica colaborativa, en el contexto de la producción industrial ecuatoriana. El objetivo principal es diseñar una HMI de bajo costo que facilite la configuración, el monitoreo y el control en tiempo real de un proceso de empaquetado de confitería formado por un cobot, comunicado con protocolo Modbus TCP/IP, banda transportadoras, sensores de presencia, etc. La metodología incluye el análisis de protocolos de comunicación, el diseño de la interfaz en Python sobre una Raspberry Pi 4 y la integración de componentes físicos como el cobot EC63, sensores, actuadores y una pantalla táctil de 7 pulgadas. La validación se realiza en un entorno controlado, donde se evalúan estabilidad de la comunicación, latencias, respuesta ante eventos críticos y operatividad del sistema en condiciones similares a planta. Entre los resultados se muestran una comunicación estable con latencias promedio en el orden de 1.3 ms, sin pérdidas ni retransmisiones, lo que permite una respuesta en tiempo real adecuada, la interfaz permite visualizar variables, alarmas, estados y conteos, facilitando la interacción del operador y contribuyendo a la seguridad y a la productividad del proceso de empaquetado. Además esta propuesta se orienta a los Objetivos de Desarrollo Sostenible (ODS) al promover infraestructura industrial resiliente y accesible (ODS 9), mejorar la eficiencia productiva y las condiciones de trabajo (ODS 8), favorecer prácticas de producción más sostenibles (ODS 12) y fortalecer las capacidades técnicas en instituciones educativas y PYMEs (ODS 4) en la región.

Palabras clave: Interfaz Hombre-Máquina (HMI), robot colaborativo, línea de empaquetado, supervisión de procesos industriales, objetivos de desarrollo sostenible.



Esta obra está bajo una licencia Creative Commons de tipo (CC-BY-NC-SA).

E-mail: editorial@ecosur.gopsapp.com



Abstract

This article presents the development and implementation of a modular human-machine interface (HMI) for supervising critical variables in collaborative-robot packaging processes in the context of Ecuadorian industrial production. The main objective is to design a low-cost HMI that facilitates configuration, monitoring, and real-time control of a confectionery packaging process composed of a cobot communicating via the Modbus TCP/IP protocol, conveyor belts, presence sensors, and auxiliary devices. The methodology includes the analysis of industrial communication protocols, the design of the interface in Python on a Raspberry Pi 4, and the integration of physical components such as the EC63 cobot, sensors, actuators, and a 7-inch touchscreen. Validation is carried out in a controlled environment, where communication stability, latency, response to critical events, and system operability under plant-like conditions are evaluated. The results show stable communication with average latencies on the order of 1.3 ms, with no losses or retransmissions, enabling adequate real-time response. The interface allows the visualization of process variables, alarms, states, and counts, facilitating operator interaction and contributing to both safety and productivity in the packaging process. In addition, the proposed solution is aligned with the Sustainable Development Goals (SDGs) by promoting resilient and accessible industrial infrastructure (SDG 9), improving productive efficiency and working conditions (SDG 8), supporting more sustainable production practices (SDG 12), and strengthening technical capacities in educational institutions and SMEs (SDG 4) in the region.

Keywords: Human-Machine Interface (HMI), collaborative robot, packaging line, industrial process supervision, Sustainable Development Goals (SDGs).

1 Introducción

La incorporación de tecnologías digitales en la industria manufacturera ha sido un factor clave para mejorar la eficiencia, la calidad y la seguridad en los procesos productivos. En este contexto, la denominada Industria 4.0 ha promovido la integración de sistemas ciberfísicos, la recopilación masiva de datos y la automatización avanzada, permitiendo la supervisión en tiempo real y la toma de decisiones basada en datos [1]. Más recientemente, la Industria 5.0 ha añadido un énfasis explícito en la colaboración humano-máquina, la personalización y la sostenibilidad, buscando procesos productivos más flexibles y centrados en las personas.

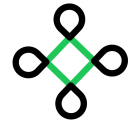
Uno de los componentes esenciales en la automatización moderna son las interfaces hombre-máquina (HMI), que facilitan la interacción entre operadores y sistemas automatizados. Las HMI permiten visualizar variables críticas, controlar procesos y diagnosticar errores, integrando pantallas táctiles, representaciones gráficas y software especializado [2]. La evolución de estas interfaces ha llevado a la adopción de estándares como la norma ISA-101, que orienta el diseño, la usabilidad y la gestión del ciclo de vida de las HMI, con el objetivo de mejorar la seguridad y la eficacia operativa [3, 4].

En paralelo, las tecnologías de comunicación industrial han evolucionado hacia protocolos como OPC UA, PROFINET o EtherCAT, que permiten interoperabilidad entre dispositivos heterogéneos en redes de alta velocidad [5, 6]. Sin embargo, en países en desarrollo como Ecuador, la adopción de estas tecnologías sigue condicionada por el costo de licencias, la complejidad de integración y las limitaciones de infraestructura. En este escenario, soluciones basadas en protocolos abiertos y hardware de bajo costo, como Raspberry Pi y plataformas de código abierto, constituyen una alternativa realista para democratizar la automatización y hacerla accesible a PYMEs e instituciones educativas [7].



Esta obra está bajo una licencia Creative Commons de tipo (CC-BY-NC-SA).

E-mail: editorial@ecosur.gopsapp.com



El uso de robots colaborativos (cobots) ha transformado la automatización industrial en tareas repetitivas como el empaquetado. Estos robots, diseñados para operar próximos a personas, ofrecen ventajas en términos de flexibilidad, reconfiguración rápida y seguridad funcional, gracias a sensores y algoritmos que permiten detectar la presencia humana [8, 9]. No obstante, su integración en procesos industriales tradicionales plantea retos en el diseño de interfaces de supervisión que combinen, de manera coherente, las variables del robot (posiciones, estados de colisión, modos de operación) con las variables del proceso (velocidad de línea, conteo de piezas, alarmas de equipos auxiliares).

En el contexto ecuatoriano, la adopción de cobots y de sistemas de supervisión en tiempo real sigue siendo incipiente en varios sectores. La curva de aprendizaje asociada a los sistemas robóticos y la dependencia de interfaces básicas, como los *teach pendants*, limitan la capacidad de supervisión integrada, dificultan la trazabilidad de variables operativas y retrasan la gestión de alarmas en celdas de trabajo donde conviven equipos convencionales y robots [10, 11]. Esta brecha tecnológica impacta la competitividad de las empresas y limita el avance hacia esquemas de producción más sostenibles e inclusivos, vinculados a los Objetivos de Desarrollo Sostenible (ODS), en particular el ODS 8 (trabajo decente y crecimiento económico), el ODS 9 (industria, innovación e infraestructura) y el ODS 12 (producción y consumo responsables).

En respuesta a estas necesidades, este trabajo propone el desarrollo de una HMI modular y de bajo costo, capaz de supervisar variables críticas en una celda de empaquetado con cobots, basada en Raspberry Pi 4 y Python, y comunicada mediante el protocolo Modbus TCP/IP. La solución está pensada para ser replicable en PYMEs y entornos educativos, de manera que contribuya al fortalecimiento de capacidades técnicas (ODS 4) y al mismo tiempo facilite la incorporación de prácticas de supervisión más seguras y eficientes.

Las contribuciones principales de este artículo son:

- Diseñar una arquitectura HMI modular, basada en hardware y software de bajo costo, que integra un cobot EC63, sensores y actuadores mediante Modbus TCP/IP en una celda de empaquetado.
- Implementar y validar un sistema de supervisión que proporciona lectura y escritura de registros críticos, gestión de alarmas y visualización de estados en tiempo real, con latencias en el orden de milisegundos.
- Evaluar experimentalmente la estabilidad de la comunicación en red (TCP/IP y Modbus/TCP), cuantificando latencias, pérdidas de paquetes y retransmisiones.
- Analizar el potencial de la solución para impulsar procesos de modernización industrial en PYMEs ecuatorianas y entornos académicos, alineando la propuesta con los ODS 4, 8, 9 y 12.

El resto del artículo se organiza de la siguiente forma: en la Sección 2 se presenta el marco teórico y los trabajos relacionados; en la Sección 3 se describen los materiales y métodos empleados; en la Sección 4 se discuten los resultados experimentales; en la Sección 5 se analizan los resultados en relación con la literatura y los ODS; finalmente, en la Sección 6 se presentan las conclusiones y las líneas de trabajo futuro.

2 Marco teórico y trabajos relacionados

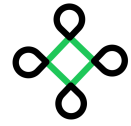
2.1 Robótica colaborativa

En la industria, los robots colaborativos han adquirido relevancia por su capacidad de trabajar de forma segura en espacios compartidos con personas. El cobot EC63, empleado en este estudio, se caracteriza por



Esta obra está bajo una licencia Creative Commons de tipo (CC-BY-NC-SA).

E-mail: editorial@ecosur.gopsapp.com



su tamaño compacto, su peso reducido y su facilidad de reprogramación, lo que favorece su uso en tareas de empaquetado y manipulación donde se requiere flexibilidad [12, 8]. Plataformas de software como ROS (Robot Operating System) facilitan la integración de distintos robots en un mismo entorno, así como la interoperabilidad con otros sistemas [12]. La programación de cobots con interfaces intuitivas y limitaciones de velocidad o par de seguridad contribuye a reducir el riesgo de accidentes en la interacción humano-robot [8, 9].

2.2 Interfaces hombre-máquina (HMI)

Las HMI son herramientas necesarias para la supervisión y el control de procesos industriales, ya que combinan hardware y software para presentar la información de forma accesible al operador [13]. Además de mostrar variables de proceso en tiempo real, las HMI permiten gestionar alarmas, diagnosticar errores y ejecutar acciones manuales sobre equipos. La norma ISA-101 propone una estructura jerárquica de pantallas y criterios de diseño gráfico orientados a mejorar la interpretación de la información y la respuesta ante condiciones anómalas [3, 4]. En el contexto de los ODS, HMI bien diseñadas contribuyen a reducir errores humanos, mejorar la seguridad y optimizar el uso de energía y materias primas, lo que se relaciona con el ODS 8 y el ODS 12.

2.3 Protocolos de comunicación industrial

La comunicación fiable entre dispositivos es crítica en sistemas automatizados. El protocolo Modbus, en sus variantes RTU, ASCII y TCP/IP, se ha consolidado como uno de los estándares en la industria por su simplicidad y amplia adopción [14, 15]. En el esquema maestro-esclavo, el maestro inicia las solicitudes y los esclavos responden a ellas. Modbus TCP/IP permite transportar el protocolo sobre redes Ethernet, manteniendo la estructura de mensajes original y facilitando la integración en redes locales existentes [16]. La literatura señala que, en condiciones adecuadas, la latencia en las respuestas de los esclavos suele ser menor a 10 ms, valor que se considera aceptable para muchas aplicaciones de supervisión en tiempo real [17].

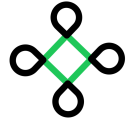
2.4 Sistemas de supervisión y control en celdas de empaquetado

La supervisión de celdas de empaquetado con robots industriales o cobots suele realizarse mediante SCADA o HMI propietarias integradas a controladores PLC. En trabajos recientes se han explorado arquitecturas basadas en hardware de bajo costo y software abierto para reducir costes e incrementar la flexibilidad, lo que resulta especialmente atractivo para PYMES y centros de formación. La integración de estas soluciones con objetivos de eficiencia energética, reducción de desperdicios y mejora de las condiciones de trabajo permite vincularlas directamente con los ODS 8, 9 y 12, así como con iniciativas de formación técnica alineadas con el ODS 4.



Esta obra está bajo una licencia Creative Commons de tipo (CC-BY-NC-SA).

E-mail: editorial@ecosur.gopsapp.com



3 Materiales y métodos

3.1 Entorno de pruebas y arquitectura física

El entorno de pruebas se configuró en un laboratorio de control de procesos, con una topología de red basada en un *switch* Ethernet que conecta la Raspberry Pi 4, el cobot EC63, sensores y otros dispositivos, conformando una red LAN segmentada [18, 19, 20]. El objetivo fue reproducir las condiciones de una celda de empaquetado representativa de una línea de producción, pero en un entorno controlado que permitiera instrumentar la comunicación de forma detallada. La disposición general de la celda de trabajo y los principales módulos físicos se muestran en la Figura 1, mientras que la arquitectura de comunicaciones y accionamiento se resume en la Figura 2.

La propuesta incluye componentes físicos como el cobot EC63, un *gripper* DH Robotics, sensores capacitivos, motores de corriente continua, variadores de frecuencia, relés, fusibles, botones de parada de emergencia, una pantalla táctil de 7 pulgadas y la propia Raspberry Pi 4. La Tabla 1 resume las características técnicas más relevantes. El prototipo del módulo HMI implementado sobre Raspberry Pi 4 puede apreciarse en la Figura 3.

Table 1: Especificaciones de componentes principales

Componente	Especificación
COBOT EC63	Carga útil 3 kg, precisión $\pm 0,02$ mm, IP54, RS485 Modbus RTU [18]
Gripper DH Robotics	Apertura rápida 0,2 s, fuerza ajustable, estructura compacta [19]
Sensor capacitivo LJC18A3-B-Z/BX	Detección hasta 10 mm, salida NPN, alimentación 6–36 V DC [21]
Motor DC S9D90-90CH	Potencia 90 W, 90 V DC, velocidad 3000 rpm [20]
Variador de frecuencia	Regulación continua, compatible con 24 V y 90 V [22]
Relé MK2P-I	24 V DC, 8 pines, 5 A / 250 V AC [23]
Pantalla táctil 7"	Resolución 1024x600, panel capacitivo, HDMI/USB [24]
Raspberry Pi 4	Procesador ARM Cortex-A72, 4 núcleos, 2–8 GB RAM [25]



Esta obra está bajo una licencia Creative Commons de tipo (CC-BY-NC-SA).

E-mail: editorial@ecosur.gopsapp.com

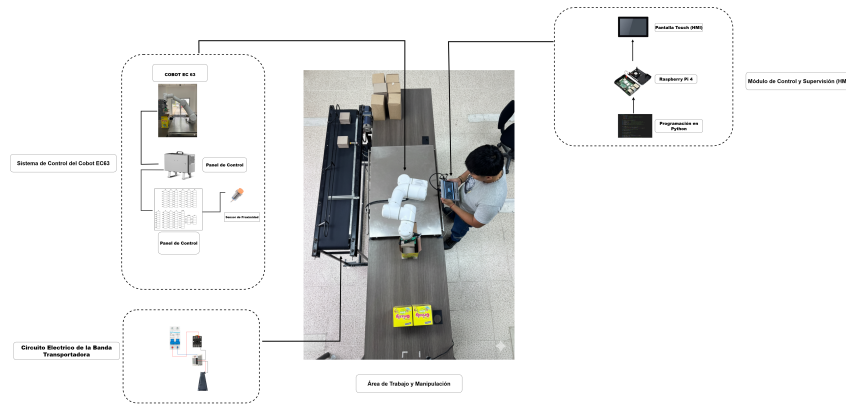
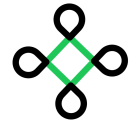


Figure 1: Disposición física de la celda de empaquetado con el cobot EC63, la banda transportadora y los módulos de control.

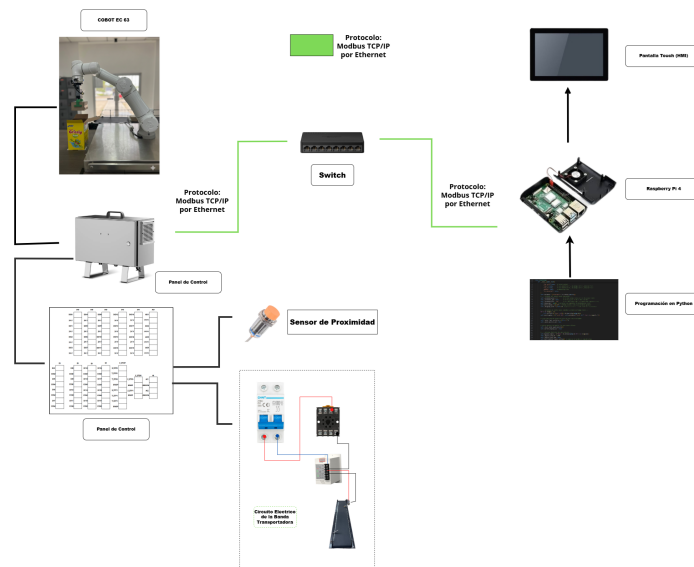


Figure 2: Arquitectura de comunicaciones y accionamiento entre el cobot, el panel de control, el sensor de proximidad y la Raspberry Pi 4 mediante Modbus TCP/IP.



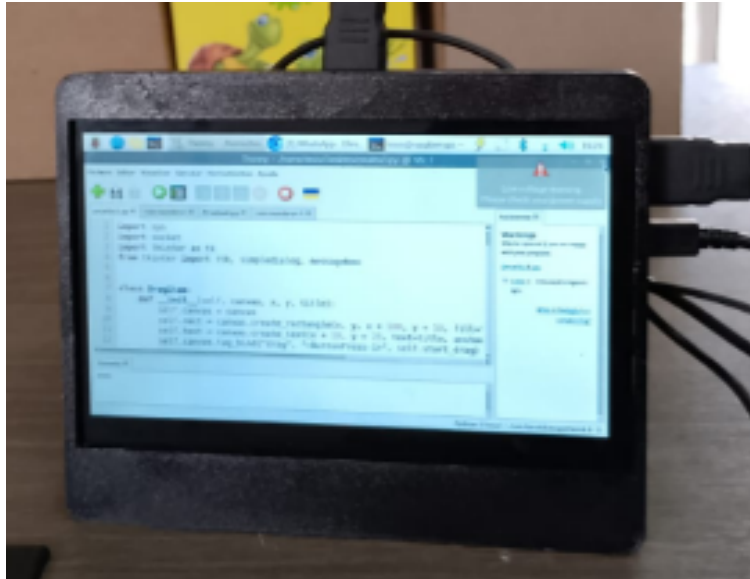
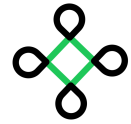


Figure 3: Prototipo del módulo HMI ejecutándose en una pantalla táctil de 7 pulgadas sobre Raspberry Pi 4.

3.2 Arquitectura lógica y módulos de software

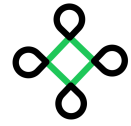
La Raspberry Pi 4 actúa como nodo maestro Modbus TCP/IP y ejecuta el software de supervisión desarrollado en Python, estructurado en los siguientes módulos:

- **Módulo de conexión:** establece y mantiene la comunicación con el cobot y otros dispositivos mediante Modbus TCP/IP, verifica la conectividad y gestiona reconexiones automáticas en caso de fallo [26].
- **Módulo de variables:** gestiona la lectura y escritura de registros específicos en los dispositivos, mediante solicitudes cíclicas y operaciones de control. La estructura de los registros incluye posiciones articulares, estados de entradas/salidas digitales y banderas de colisión.
- **Módulo de alarmas:** monitoriza registros asociados a condiciones críticas (colisión, sobrevelocidad, fallo de sensores) y activa alarmas visuales y sonoras en la interfaz. Se mantiene un registro histórico de eventos para análisis posterior.
- **Módulo de interfaz gráfica:** presenta la información en una pantalla táctil de 7 pulgadas, organizada en pestañas que agrupan visión general del sistema, variables de proceso, alarmas y parámetros de configuración. La interfaz está pensada para operadores con formación técnica básica, lo que favorece el alineamiento con el ODS 4.



Esta obra está bajo una licencia Creative Commons de tipo (CC-BY-NC-SA).

E-mail: editorial@ecosur.gopsapp.com



El flujo general de información desde la programación en Python hasta el proceso de empaquetado y la visualización en la pantalla HMI se ilustra en la Figura 4, mientras que la Figura 5 resume la arquitectura de módulos de control y supervisión implementados.

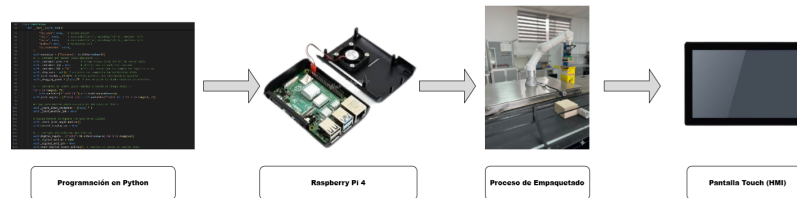


Figure 4: Flujo lógico desde la programación en Python hacia la Raspberry Pi 4, el proceso de empaquetado y la visualización en la pantalla HMI.

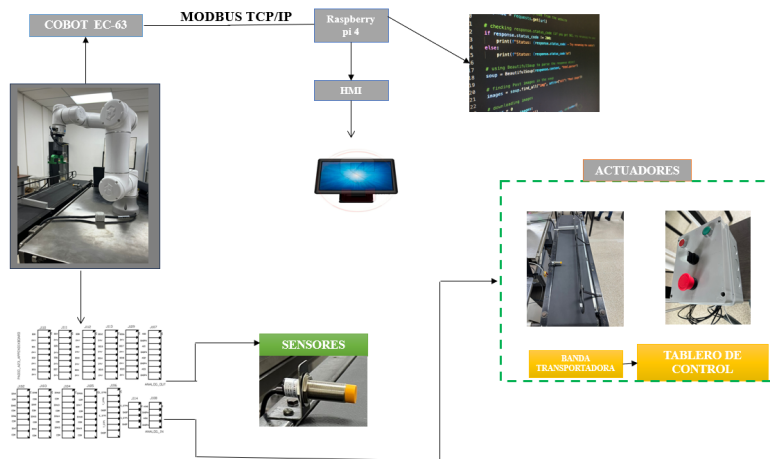


Figure 5: Arquitectura de módulos de control y supervisión para la celda de empaquetado asistida por cobot.

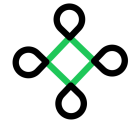
3.3 Protocolo experimental y métricas

Para validar la comunicación y el funcionamiento del sistema se realizaron pruebas de transmisión de datos capturando tramas Modbus/TCP con Wireshark. El protocolo experimental incluyó:



Esta obra está bajo una licencia Creative Commons de tipo (CC-BY-NC-SA).

E-mail: editorial@ecosur.gopsapp.com



- 50 pruebas de tráfico con lecturas continuas de registros Modbus mediante funciones FC03 (*Read Holding Registers*).
- Captura de aproximadamente 500 paquetes por prueba, registrando tiempos entre solicitud y respuesta.
- Cálculo de métricas de latencia mínima, máxima y promedio, así como verificación de pérdidas de paquetes y retransmisiones.

Adicionalmente, se verificó la respuesta del sistema ante eventos simulados, como colisiones y fallos en sensores, forzando valores en registros específicos y comprobando la activación de alarmas y el registro de eventos en la HMI.

Las métricas seleccionadas se relacionan directamente con requisitos típicos de supervisión en tiempo real: latencia de comunicación menor a 10 ms, ausencia de pérdidas de paquetes y capacidad de detección oportuna de eventos críticos. Desde la perspectiva de los ODS, estas métricas están asociadas a la eficiencia operativa (ODS 8 y ODS 9) y a la reducción de rechazos o retrabajos (ODS 12).

4 Resultados

4.1 Requerimientos del sistema HMI

Los requerimientos funcionales definidos en la fase de diseño se resumen en la Tabla 2. Todos ellos fueron verificados experimentalmente en el entorno de laboratorio.

Table 2: Requerimientos funcionales del sistema HMI-COBOT

Requerimiento	Descripción
R1	Comunicación mediante el protocolo Modbus TCP/IP para el intercambio confiable de datos.
R2	Monitorización y control en tiempo real del cobot EC63 desde la HMI.
R3	Visualización de alarmas, estados de conexión y conteo de piezas.
R4	Implementación con herramientas de bajo costo (Python) y hardware accesible.
R5	Latencia de comunicación menor a 10 ms en condiciones nominales.

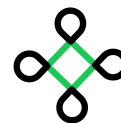
Durante las pruebas, la HMI mostró correctamente las variables de posición, el conteo de piezas y el estado de conexión, y permitió la detección y gestión de alarmas simuladas.

4.2 Interfaz HMI para supervisión y operación

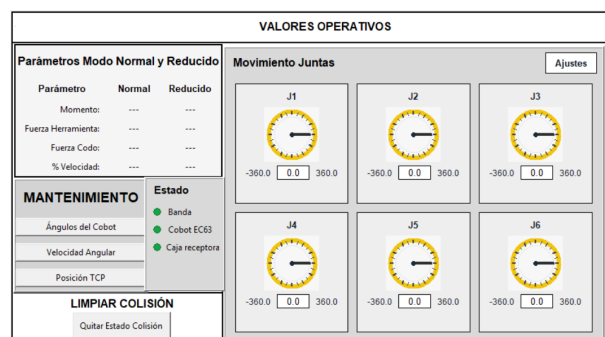
La Figura 6 muestra las dos pantallas principales desarrolladas para la HMI. En la subfigura 6a se presenta la pantalla de *Valores operativos*, donde se supervisan los parámetros del modo normal y reducido, así como los ángulos de las juntas del cobot. Esta vista se utiliza principalmente durante las tareas de mantenimiento y ajuste fino del robot.



Esta obra está bajo una licencia Creative Commons de tipo (CC-BY-NC-SA).
E-mail: editorial@ecosur.gopsapp.com



En la subfigura 6b se observa la pantalla del *Proceso de empaquetado*, que integra el estado de conexión, el control básico de arranque y parada, los archivos de programa y el panel de alarmas, junto con los contadores de producción. Esta pantalla es la que utiliza el operador durante la operación normal de la celda.



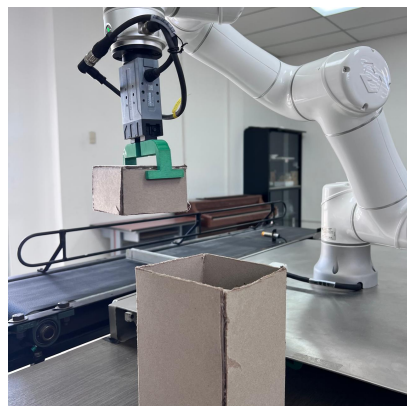
(a) Pantalla de valores operativos y movimiento de juntas del cobot.



(b) Pantalla principal del proceso de empaquetado con contadores y panel de alarmas.

Figure 6: Pantallas principales de la HMI desarrollada para la celda de empaquetado asistida por cobot.

La Figura 7 presenta una secuencia fotográfica de la operación del cobot EC63 sobre la banda transportadora. Se observa la aproximación al producto, la extracción de la caja interna y el traslado hacia el área de descarga. Estas imágenes permiten relacionar visualmente la secuencia física del proceso con las variables que se supervisan desde la HMI, reforzando el carácter didáctico de la solución para actividades de formación.



(a) Aproximación y sujeción del producto dentro de la caja.



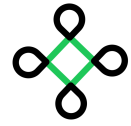
(b) Extracción y elevación de la caja interna.



(c) Traslado del paquete hacia el área de descarga.

Figure 7: Secuencia de manipulación de cajas en la celda de empaquetado mediante el cobot EC63 y el *gripper* DH Robotics.





4.3 Estabilidad y desempeño de la comunicación en red

Para verificar la disponibilidad de la red se realizaron pruebas de comunicación hacia las direcciones IP del cobot EC63 y de la Raspberry Pi 4. Se obtuvieron tiempos de respuesta entre 2 y 6 ms sin pérdida de paquetes (0%), lo que confirma la estabilidad de la comunicación IP en la LAN de pruebas.

El análisis de las sesiones TCP se resume en la Tabla 3. La duración típica de las sesiones se situó alrededor de 27.7 s, con latencias medias en torno a 1.3 ms y sin retransmisiones ni pérdidas observadas.

Table 3: Resultados de análisis a nivel de transporte (TCP/IP)

Métrica	Conexiones / frames	Observaciones
Duración de sesiones TCP	Varios rangos (por ejemplo, 52179–52277)	Conexiones breves asociadas a ciclos de consulta.
Latencia mínima	Frames 240–243	0,638 ms.
Latencia máxima	Frames 200–205	3,659 ms.
Latencia promedio	Todas las sesiones	Aproximadamente 1,3 ms.
Pérdida de paquetes	–	0%, sin pérdidas detectadas.
Retransmisiones	–	No se observaron retransmisiones.

4.4 Análisis de transacciones Modbus/TCP

Se analizaron transacciones FC03 entre la HMI (cliente) y el cobot EC63 (servidor) para distintas direcciones de registro. En la Tabla 4 se muestran ejemplos de solicitudes y en la Tabla 5 las respuestas correspondientes.

Table 4: Ejemplos de solicitudes Modbus/TCP (FC03)

Registros	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	192.168.1.100:53896 → 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

Table 5: Ejemplos de respuestas Modbus/TCP (FC03)

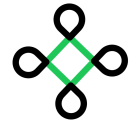
Registros	Dirección (IP:Puerto)	Rol	Bytes	Tipo	Latencia (ms)
200–205	192.168.1.201:502 → 192.168.1.100:53878	Servidor	21	Respuesta	5,972
240–243	192.168.1.201:502 → 192.168.1.100:53875	Servidor	17	Respuesta	5,051
200–205	192.168.1.201:502 → 192.168.1.100:53869	Servidor	21	Respuesta	8,120
200–205	192.168.1.201:502 → 192.168.1.100:53896	Servidor	21	Respuesta	0,780
250–253	192.168.1.201:502 → 192.168.1.100:53889	Servidor	17	Respuesta	9,711

En un conjunto de pruebas continuas, la duración agrupada de las transacciones fue de aproximadamente



Esta obra está bajo una licencia Creative Commons de tipo (CC-BY-NC-SA).

E-mail: editorial@ecosur.gopsapp.com



27.7 s, con latencias entre 0.638 ms y 3.659 ms y un promedio cercano a 1.3 ms, sin errores Modbus ni retransmisiones. Estos valores son compatibles con los requerimientos definidos en la Tabla 2.

4.5 Gestión de eventos críticos y alarmas

El módulo de alarmas registró eventos simulados relacionados con colisiones y sobrevelocidad en distintos ejes del cobot. La Tabla 6 ilustra ejemplos de eventos registrados.

Table 6: Ejemplo de eventos en el módulo de alarmas

Fecha	Hora	ID	Estado	Variable	Comentario
2024-05-01	14:35:22	AL001	ON	Colisión	Detectada en eje 3
2024-05-01	14:36:10	AL002	ON	Sobrevelocidad	Eje 2
2024-05-01	14:37:05	AL003	OFF	–	–

La HMI presentó las alarmas con cambio de color de estado, texto descriptivo y registro de *timestamp*. La activación fue inmediata tras el cambio del registro asociado en el cobot, lo que indica que la latencia de comunicación no compromete la detección de eventos críticos.

5 Discusión

Los resultados obtenidos muestran que la HMI modular desarrollada sobre Raspberry Pi 4 y Python cumple los requisitos básicos de supervisión en tiempo real en una celda de empaquetado con robótica colaborativa. Las latencias promedio en el orden de 1.3 ms y la ausencia de pérdidas de paquetes o retransmisiones en las pruebas realizadas indican que la arquitectura de comunicaciones es adecuada para aplicaciones de monitorización y control de nivel HMI.

Desde el punto de vista técnico, la solución confirma que una arquitectura basada en hardware de bajo costo y protocolos abiertos puede integrarse de manera efectiva con un cobot comercial, sin necesidad de plataformas propietarias adicionales. Esto resulta relevante para entornos donde el presupuesto es limitado y donde la adopción de tecnologías de Industria 4.0 ha sido lenta por motivos económicos y de capacitación.

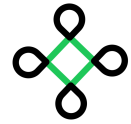
En relación con los Objetivos de Desarrollo Sostenible, la propuesta presenta varios puntos de alineamiento:

- **ODS 9 (Industria, innovación e infraestructura):** la arquitectura HMI-cobot se puede replicar en PYMEs y plantas con infraestructura existente basada en Ethernet, favoreciendo la modernización gradual de celdas de producción mediante estándares abiertos.
- **ODS 8 (Trabajo decente y crecimiento económico):** una supervisión más clara de variables, alarmas y conteos contribuye a reducir tiempos muertos y errores de operación; además, una interfaz más intuitiva permite que operadores con diferente nivel de experiencia interactúen de manera más segura con el cobot.



Esta obra está bajo una licencia Creative Commons de tipo (CC-BY-NC-SA).

E-mail: editorial@ecosur.gopsapp.com



- **ODS 12 (Producción y consumo responsables):** al mejorar la trazabilidad y el conteo de piezas, la HMI ayuda a identificar paradas y rechazos, facilitando acciones de mejora para reducir desperdicios y retrabajos.
- **ODS 4 (Educación de calidad):** la solución, basada en tecnologías abiertas, puede emplearse como plataforma didáctica en instituciones de educación técnica y universitaria, contribuyendo a la formación de talento en automatización y robótica colaborativa.

No obstante, el sistema fue probado en un entorno de laboratorio controlado, sin interferencias significativas de tráfico adicional en la red ni condiciones prolongadas de operación. Tampoco se evaluaron métricas de usabilidad con usuarios reales ni se comparó la HMI propuesta con soluciones comerciales de SCADA o con otros protocolos como OPC UA. Estas limitaciones acotan el alcance de las conclusiones y señalan la necesidad de estudios adicionales.

6 Conclusiones y trabajos futuros

La investigación presentada describe el desarrollo, la implementación y la validación de una interfaz hombre-máquina modular para supervisar y controlar un sistema de robótica colaborativa en un proceso de empaquetado, utilizando el protocolo Modbus TCP/IP y una plataforma basada en Raspberry Pi 4. En las condiciones de ensayo, la solución cumple los requisitos de estabilidad y latencia definidos: las sesiones TCP se mantienen estables durante ventanas continuas del orden de 27.7 s, sin pérdidas de paquetes ni retransmisiones, y con latencias promedio alrededor de 1.3 ms. Estos resultados indican que la red Ethernet y la configuración Modbus/TCP empleadas proporcionan un canal de comunicación adecuado para supervisión en tiempo real.

La HMI desarrollada, estructurada en módulos de conexión, variables, alarmas y configuración, facilita la visualización y modificación de variables críticas, así como la detección y gestión de eventos anómalos. El uso de Python y bibliotecas abiertas, junto con hardware accesible como Raspberry Pi y pantallas táctiles, contribuye a reducir costos y a simplificar el mantenimiento del sistema, lo que es especialmente relevante para PYMEs y centros educativos en países en desarrollo.

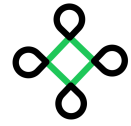
En términos de contribución a los ODS, la arquitectura propuesta apoya:

- La modernización de la infraestructura industrial mediante tecnologías accesibles (ODS 9).
- La mejora gradual de la productividad y la seguridad de los trabajadores al disponer de una supervisión más clara y centralizada (ODS 8).
- La identificación de ineficiencias y la reducción potencial de desperdicios a través del monitoreo de variables y conteos de producción (ODS 12).
- La formación de talento humano en automatización y robótica colaborativa mediante una plataforma basada en tecnologías abiertas (ODS 4).

Como trabajos futuros se plantea:



Esta obra está bajo una licencia Creative Commons de tipo (CC-BY-NC-SA).
E-mail: editorial@ecosur.gopsapp.com



- Validar el sistema en un entorno industrial real con múltiples dispositivos Modbus, analizando la escalabilidad de la solución y el impacto sobre indicadores de proceso.
- Incorporar mecanismos de seguridad en la comunicación, como segmentación de red mediante VLAN o ACL, autenticación y, cuando sea posible, encapsulación de Modbus/TCP en túneles TLS o VPN.
- Integrar paneles de indicadores clave de desempeño (KPI) y módulos de historización para registrar métricas como latencia, tasa de alarmas, disponibilidad y eficiencia global del equipo (OEE), facilitando la toma de decisiones y el mantenimiento predictivo.
- Realizar estudios de usabilidad con operadores y estudiantes, orientados a mejorar la ergonomía de la HMI y a cuantificar su impacto en la carga cognitiva y en la reducción de errores de operación.

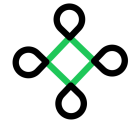
Estas líneas de trabajo buscan consolidar la propuesta como una solución replicable, económica y adaptable, capaz de contribuir de manera concreta a los procesos de modernización industrial y a las metas asociadas a los Objetivos de Desarrollo Sostenible.

References

- [1] M. Šverko and T. Galinac Grbac, “Automated hmi design as a custom feature in industrial scada systems,” *Procedia Computer Science*, vol. 232, no. 3, pp. 1789–1798, 2024.
- [2] P. Zhang, “Chapter 13 – human–machine interfaces,” in *Advanced Industrial Control Technology*. William Andrew, 2010, pp. 527–555.
- [3] A. A. Group, “Isa-101 lifecycle standard and hmi design,” <https://www.arcweb.com/...>, 2023.
- [4] ISA, “Isa-101 standards for hmi,” <https://www.isa.org/...>, 2025.
- [5] J. A. Matute Caguana, “Monitoreo y operación de un sistema industrial servo–posicionamiento con hmi python + opc-ua,” <http://dspace.ups.edu.ec/handle/123456789/25734>, 2023.
- [6] J. H. Hwang Cárdenas and J. A. Iñiguez Avila, “Sistema inalámbrico + hmi python para monitoreo de motor trifásico,” <http://dspace.ups.edu.ec/handle/123456789/25739>, 2023.
- [7] S. R. Torres Gualsaqui, “Implementación de un sistema hmi con tecnologías abiertas para control industrial,” <http://repositorio.utn.edu.ec/handle/123456789/11651>, 2021.
- [8] I. Gracia Ruiz, “Manipulación robótica colaborativa con ros,” 2019.
- [9] C. Cobots, “¿qué es un cobot y sus ventajas?” <https://cadecobots.com/que-es-un-cobot/>, online.
- [10] J. Pelegrí, “Aplicaciones robóticas para un empaquetado más eficiente,” <https://www.universal-robots.com/es/blog/aplicaciones-roboticas-para-un-empaquetado-mas-eficiente/>, s.f.
- [11] F. Group, “Partner autorizado de la marca kuka en ecuador,” <https://fainca-group.com/2021/12/06/partner-autorizado-de-la-marca-kuka-en-ecuador/>, 2021.



Esta obra está bajo una licencia Creative Commons de tipo (CC-BY-NC-SA).
E-mail: editorial@ecosur.gopsapp.com



- [12] C. Cobots, “Empaquetadora y paletizadora automática,” <https://cadecobots.com/...>, online.
- [13] Sicma21, “¿qué es un hmi y cómo funciona?” <https://www.sicma21.com/...>, 2021.
- [14] Modbus-IDA, “Modbus tcp/ip messaging guide v1.0b,” <https://scadahacker.com/...>, 2006.
- [15] A. F. Castro, “Protocolo modbus,” <https://cuervaenergia.com/...>, 2024.
- [16] Logicbus, “Comunicación modbus tcp/ip,” <https://www.logicbus.com.mx/...>, 2024.
- [17] D. S.A., “Modbus rtu protocolo industrial,” <https://www.ditel.es/...>, 2024.
- [18] E. Robots, “Ec63 cobot - especificaciones,” <https://es.eliterobots.com/cobots/ec63>, 2025.
- [19] D. Robotics, “Pge product data,” <https://en.dh-robotics.com/product/pge>.
- [20] S. Motor, “Product literature,” <https://spgmotor.net/product-literature/>, 2024.
- [21] A. Electronics, “Sensor capacitivo ljc18a3,” <https://avelectronics.cc/...>, 2024.
- [22] U. ECUADOR, “Controlador velocidad dc 24-90v,” <https://www.ubuy.ec/...>, 2024.
- [23] Roboticsec, “Relé 24v 30a 1 canal,” <https://roboticsec.com/...>, 2023.
- [24] R. P. Chile, “Pantalla touch 7” para raspberry pi,” <https://raspberrypi.cl/...>, 2025.
- [25] J. Pastor, “Revisión raspberry pi 4,” <https://www.xataka.com/...>, 2019.
- [26] G. Cuadrado, “¿qué es visual studio code?” <https://openwebinars.net/...>, 2022.



Esta obra está bajo una licencia Creative Commons de tipo (CC-BY-NC-SA).

E-mail: editorial@ecosur.gopsapp.com