|  |
| --- |
|  |

Especificación de requisitos de software

Proyecto: PLC4uni

Revisión 1



|  |  |  |
| --- | --- | --- |
|  |  | septiembre |

De la plantilla de formato del documento © & Coloriuris http://www.qualitatis.org

.Ficha del documento

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Revisión** | **Autor** | **Verificado dep. calidad.** |
| 18/9/2025 | Rev-1 | Henry Abdias Bautista Portes (Equipo de Desarrollo) | Ing. Carlos Pichardo (director del Proyecto) |

Documento validado por las partes en fecha: [18/09/2025]

|  |  |
| --- | --- |
| Por el cliente | Por la empresa suministradora |
|  |  |
| Ing. Carlos Pichardo (director del proyecto) | Equipo de Estudiantes de Mecatrónica – Proyecto PLC4uni). |

Contenido

[Ficha del documento 3](#_Toc33411057)

[Contenido 4](#_Toc33411058)

[1 Introducción 6](#_Toc33411059)

[1.1 Propósito 6](#_Toc33411060)

[1.2 Alcance 6](#_Toc33411061)

[1.3 Personal involucrado 6](#_Toc33411062)

[1.4 Definiciones, acrónimos y abreviaturas 6](#_Toc33411063)

[1.5 Referencias 6](#_Toc33411064)

[1.6 Resumen 6](#_Toc33411065)

[2 Descripción general 7](#_Toc33411066)

[2.1 Perspectiva del producto 7](#_Toc33411067)

[2.2 Funcionalidad del producto 7](#_Toc33411068)

[2.3 Características de los usuarios 7](#_Toc33411069)

[2.4 Restricciones 7](#_Toc33411070)

[2.5 Suposiciones y dependencias 7](#_Toc33411071)

[2.6 Evolución previsible del sistema 7](#_Toc33411072)

[3 Requisitos específicos 7](#_Toc33411073)

[3.1 Requisitos comunes de los interfaces 8](#_Toc33411074)

[3.1.1 Interfaces de usuario 8](#_Toc33411075)

[3.1.2 Interfaces de hardware 8](#_Toc33411076)

[3.1.3 Interfaces de software 8](#_Toc33411077)

[3.1.4 Interfaces de comunicación 8](#_Toc33411078)

[3.2 Requisitos funcionales 8](#_Toc33411079)

[3.2.1 Requisito funcional 1 9](#_Toc33411080)

[3.2.2 Requisito funcional 2 9](#_Toc33411081)

[3.2.3 Requisito funcional 3 9](#_Toc33411082)

[3.2.4 Requisito funcional n 9](#_Toc33411083)

[3.3 Requisitos no funcionales 9](#_Toc33411084)

[3.3.1 Requisitos de rendimiento 9](#_Toc33411085)

[3.3.2 Seguridad 9](#_Toc33411086)

[3.3.3 Fiabilidad 9](#_Toc33411087)

[3.3.4 Disponibilidad 9](#_Toc33411088)

[3.3.5 Mantenibilidad 10](#_Toc33411089)

[3.3.6 Portabilidad 10](#_Toc33411090)

[3.4 Otros requisitos 10](#_Toc33411091)

[4 Apéndices 10](#_Toc33411092)

# Introducción

La presente Especificación de Requisitos de Software (SRS) corresponde al proyecto **PLC4uni**, un controlador lógico programable de carácter docente que será diseñado y desarrollado por los estudiantes de la asignatura Mecatrónica, bajo la dirección del profesor de la materia. Este documento proporciona una visión global del proyecto, establece los objetivos, el alcance, el personal involucrado, las definiciones y abreviaturas necesarias, así como las referencias de apoyo y un resumen de la organización del documento.

## Propósito

* **Propósito del proyecto:** El proyecto PLC4uni tiene como propósito diseñar y construir un PLC educativo de bajo costo, basado en el microcontrolador ESP32, que permita cubrir la necesidad actual de la universidad de contar con equipos propios para la enseñanza de automatización y control. El objetivo es ofrecer una herramienta práctica que acerque a los estudiantes a los conceptos de programación, control y comunicación industrial en un entorno seguro y accesible.
* **Propósito del documento:** Este documento tiene la finalidad de establecer, de manera clara y verificable, los requisitos funcionales y no funcionales del sistema, servir de guía al equipo desarrollador durante todas las etapas del proyecto, y proporcionar una base de validación para el director del proyecto.

**Audiencia a la que va dirigido:** Está dirigido a:

* **Profesores de la universidad**, que usarán el PLC para impartir clases y diseñar prácticas.
* **Estudiantes en formación**, que lo utilizarán para aprender conceptos de automatización y control.
* **Equipo desarrollador y técnicos de laboratorio**, que requieren el documento como guía técnica de diseño, construcción y pruebas.

## Alcance

## El PLC4UNI se plantea como una herramienta formativa de bajo costo, destinada a la enseñanza de conceptos básicos de automatización. Contará con un entorno de programación introductorio en lenguajes como Ladder o FBD, ejecutará los programas de manera cíclica y permitirá trabajar con entradas y salidas digitales y analógicas. Asimismo, dispondrá de conectividad con computadoras a través de USB o Ethernet y funciones que faciliten la supervisión y la simulación de ejercicios prácticos.

## El desarrollo incluirá la definición de los requisitos principales, la creación del prototipo, su validación mediante ensayos en laboratorio y la preparación de materiales técnicos para su aplicación académica. Quedan excluidos los procesos de certificación industrial y la integración completa con sistemas propietarios de terceros, limitándose únicamente a usos educativos.

## Personal involucrado

|  |  |
| --- | --- |
| Nombre | Carlos Pichardo |
| Rol | Director del proyecto |
| Categoría profesional | Docente universitario |
| Responsabilidades | Supervisión, validación y aprobación del proyecto |
| Información de contacto | cpichardo@itla.edu.do |
| Aprobación |  |

|  |  |
| --- | --- |
| Nombre | Henry Abdias Bautista Portes |
| Rol | Integrante del equipo desarrollador |
| Categoría profesional | Estudiante |
| Responsabilidades | Desarrollo, diseño y documentación del proyecto |
| Información de contacto | 20231696@itla.edu.do |
| Aprobación |  |

## Definiciones, acrónimos y abreviaturas

**PLC (Controlador Lógico Programable / Programmable Logic Controller):** Equipo electrónico empleado para la automatización de procesos industriales mediante la programación de rutinas

**PLC4UNI:** Proyecto académico desarrollado por estudiantes de Mecatrónica del ITLA, consistente en un PLC con fines educativos.

**SRS (Software Requirements Specification / Especificación de Requisitos de Software):** Documento que detalla de manera exhaustiva los requerimientos funcionales y no funcionales que debe cumplir un software.

**ERS (Especificación de Requisitos del Sistema):** Documento que establece las necesidades técnicas de todo el sistema, considerando tanto hardware como software.

**ITLA (Instituto Tecnológico de Las Américas):** Centro de estudios superiores donde se lleva a cabo el desarrollo del proyecto.

**HW (Hardware):** Conjunto de elementos físicos del sistema, como la placa, el microcontrolador y los módulos de entradas/salidas.

**SW (Software):** Programación y código que permiten la operación, control y programación del PLC4UNI.

**I/O (Entradas y Salidas / Input/Output):** Señales analógicas o digitales que conectan el PLC con su entorno.

**DI (Entrada Digital / Digital Input):** Señal binaria que solo admite dos estados: 1 (activo) o 0 (inactivo).

**DO (Salida Digital / Digital Output):** Señal binaria de salida utilizada para accionar dispositivos externos en modo ON/OFF.

**AI (Entrada Analógica / Analog Input):** Señal de valor continuo empleada para representar magnitudes como voltaje, corriente o temperatura.

**AO (Salida Analógica / Analog Output):** Señal continua de salida que permite el control proporcional de equipos como motores o válvulas.

**Ciclo de Escaneo (Scan Cycle):** Proceso repetitivo donde el PLC lee entradas, ejecuta instrucciones y actualiza las salidas.

**Ladder (LD / Lenguaje Escalera):** Lenguaje gráfico de programación basado en esquemas similares a circuitos de relés eléctricos.

**FBD (Function Block Diagram / Diagrama de Bloques de Funciones):** Lenguaje visual donde las funciones se representan como bloques conectados entre sí.

**HMI (Interfaz Hombre-Máquina / Human Machine Interface):** Pantalla o entorno gráfico que facilita la interacción entre el operador y el PLC.

**USB (Universal Serial Bus):** Protocolo de comunicación estándar que posibilita la conexión y transferencia de datos entre un ordenador y el PLC.

## Referencias

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Referencia** | **Titulo** | **Ruta** | **Fecha** | **Autor** |
| 1 | Repositorio | https://github.com/HABP0705/Henry\_Bautista\_PLC4uni.git | 18/9/2025 | Henry Abdias Bautista Portes |
| 2 | IEC 61131-3: Programming Languages for PLCs | https://webstore.iec.ch/publication/4552 | 2013 | IEC |
| 3 | Programmable Logic Controllers: Principles and Applications | ------ | 2015 | J.W. Webb, R.A. Reis |
| 4 | Documentación oficial ESP32 | https://docs.espressif.com | 2025 | Espressif Systems |
| 5 | Modbus Protocol Specification V1.1b3 | https://modbus.org | 2012 | Modbus Organization |
| 6 | Siemens SIMATIC PLCs | https://support.industry.siemens.com | 2025 | Siemens |
| 7 | Allen-Bradley ControlLogix PLCs | https://literature.rockwellautomation.com | 2025 | Rockwell Automation |
| 8 | Schneider Electric Modicon PLCs | https://www.se.com | 2025 | Schneider Electric |
| 9 | Mitsubishi MELSEC PLCs | https://www.mitsubishielectric.com/fa | 2025 | Mitsubishi Electric |
| 10 | Omron Sysmac PLCs | https://automation.omron.com | 2025 | Omron |

## 

## Resumen

El presente documento detalla los requisitos necesarios para el diseño y desarrollo del proyecto **PLC4uni**. La introducción establece el contexto general, mientras que la sección 2 describe la visión global del sistema. La sección 3 especifica los requisitos funcionales y no funcionales, las interfaces y los criterios de aceptación. Posteriormente, los apéndices incluyen casos de prueba, plantilla de lista de materiales (BOM), matriz de trazabilidad y cronograma. Finalmente, se presenta un análisis de riesgos, las mitigaciones correspondientes y la validación del documento por las partes interesadas.

# Descripción general

## Perspectiva del producto

El proyecto PLC4uni es un producto independiente diseñado para el entorno académico. No depende de un sistema mayor para su funcionamiento, aunque puede integrarse con otros sistemas de enseñanza de automatización (p. ej. prácticas con SCADA o módulos de laboratorio). Su arquitectura permite conectarse a sensores y actuadores de baja potencia, así como a ordenadores mediante USB, RS-485 o Wi-Fi.

En la práctica, el PLC4uni se situará como un módulo de control central dentro de la bancada de prácticas de laboratorio, conectado a fuentes de alimentación de 24 VDC, a los dispositivos de campo (entradas/salidas), y a un ordenador para configuración o monitoreo.

## Funcionalidad del producto

**El PLC4uni ofrecerá las siguientes funcionalidades principales:**

* Lectura de entradas digitales opto-aisladas para detectar señales de sensores, pulsadores o interruptores.
* Activación de salidas digitales mediante relés o transistores para controlar actuadores de baja potencia (luces, motores pequeños, electroválvulas).
* Adquisición de señales analógicas (0–10 V o 4–20 mA) para procesar variables físicas como temperatura o presión.
* Comunicación estándar a través de RS-485 (Modbus RTU) y Wi-Fi (Modbus TCP), facilitando la conexión con PCs, HMIs o incluso con otros controladores.
* Interfaz local básica compuesta por una pantalla y botones, que permita visualizar estados y realizar configuraciones simples.
* Funciones de autodiagnóstico al arrancar y mecanismos de seguridad como paro de emergencia.
* Actualización de firmware por USB o Wi-Fi para mantenimiento y mejora continua.a

## 

## Características de los usuarios

|  |  |
| --- | --- |
| Tipo de usuario | Profesor universitario |
| Formación | profesional en ingeniería, automatización o áreas afines |
| Habilidades | Capacidad de diseñar prácticas, guiar el uso del PLC, evaluar resultados y resolver dudas técnicas |
| Actividades | Preparar y explicar ejercicios de programación y control; supervisar el trabajo de los estudiantes; usar el PLC4uni como herramienta de enseñanza. |

## 

|  |  |
| --- | --- |
| Tipo de usuario | Estudiante |
| Formación | nivel universitario |
| Habilidades | Experiencia académica básica en programación y electrónica, lectura de diagramas, uso de instrumentos de laboratorio. |
| Actividades | Realizar prácticas de conexión de sensores y actuadores; programar y probar el PLC; aprender conceptos de automatización mediante el uso del PLC4uni. |

## Restricciones

* **Hardware:** Uso de ESP32 como microcontrolador base. Se limita a un máximo de 8 entradas digitales, 8 salidas digitales y 4 entradas analógicas en la versión inicial.
* **Software:** Programación en C/C++ con entornos compatibles con ESP32 (Arduino IDE o ESP-IDF).
* **Normativas:** El diseño debe cumplir con normas básicas de seguridad eléctrica en laboratorio (protección de 24 VDC, aislamiento entre lógica y potencia).
* **Presupuesto:** Uso de componentes disponibles localmente y de bajo costo, limitando la inclusión de módulos avanzados en esta primera versión.
* **Tiempo:** El proyecto debe completarse dentro del calendario académico establecido (un cuatrimestre).

## Suposiciones y dependencias

## Se asume que los laboratorios disponen de fuentes de alimentación de 24 VDC, ordenadores con puertos USB y conexión a internet.

## Se supone la disponibilidad de módulos y componentes ESP32 en el mercado local o en tiempos de importación razonables.

## Se depende de la orientación y aprobación del profesor para cada etapa del proyecto.

## En caso de cambios en la disponibilidad de hardware (por ejemplo falta de relés o módulos RS-485), los requisitos deberán ajustarse.

## El proyecto depende del soporte básico de software (ESP-IDF o Arduino Core) para el ESP32, que debe estar disponible y estable.

## *.*Evolución previsible del sistema

A futuro, el **PLC4uni** puede evolucionar en varias direcciones:

* Ampliación del número de entradas y salidas digitales y analógicas.
* Inclusión de **módulos de expansión** externos para mayor flexibilidad.
* Implementación de un **entorno de programación en lenguaje Ladder** simplificado para estudiantes.
* Incorporación de **comunicación inalámbrica avanzada** (Bluetooth, MQTT, integración con plataformas IoT).
* Integración con **sistemas SCADA** de mayor nivel para prácticas avanzadas.
* Desarrollo de materiales didácticos adicionales (manuales de prácticas, simuladores virtuales complementarios).

# Requisitos específicos

Esta sección describe de manera detallada todos los requisitos que debe cumplir el sistema PLC4uni. La definición precisa de estos requisitos permitirá al equipo de desarrollo diseñar un sistema completo y verificable, y a los encargados de pruebas comprobar que cada funcionalidad se satisface.

Los requisitos se identifican con un número único (ej. RF-1, RF-2, RNF-1) para asegurar trazabilidad y seguimiento. Para cada requisito se indica su nombre, tipo (requisito o restricción), fuente (quién lo propuso o definió), prioridad (alta, media, baja) y una descripción.

## Requisitos comunes de los interfaces

El sistema PLC4uni contará con un conjunto de interfaces que permitirán la interacción entre el usuario, el hardware y el software del controlador. Estos requisitos abarcan tanto la manera en que los usuarios perciben y utilizan el sistema (interfaz de usuario), como la conexión del hardware con sensores y actuadores, la integración del software de control, y los mecanismos de comunicación con dispositivos externos.

### 

### Interfaces de usuario

|  |  |  |  |
| --- | --- | --- | --- |
| Número de requisito | IF-01 | | |
| Nombre de requisito | Interfaces de usuario | | |
| Tipo | Requisito | Restricción | |
| Fuente del requisito | Director / Equipo de desarrollo | | |
| Prioridad del requisito | Alta/Esencial | Media/Deseado | Baja/ Opcional |

**Descripción:**

El PLC contará con una pantalla LCD 16x2 (o equivalente OLED 128x64) para mostrar estados de entradas, salidas y mensajes de error. Tendrá 4 botones de usuario (navegación, selección, aceptar, cancelar) y un botón de reset. Incluirá LEDs indicadores para: alimentación (Power), ejecución (Run) y error (Error). El estilo del interfaz será simple, con textos claros y símbolos básicos (ON/OFF, ERR, etc.), pensado para estudiantes principiantes.

### 

### Interfaces de hardware

|  |  |  |  |
| --- | --- | --- | --- |
| Número de requisito | IF-02 | | |
| Nombre de requisito | Interfaces de hardware | | |
| Tipo | Requisito | Restricción | |
| Fuente del requisito | Equipo de desarrollo | | |
| Prioridad del requisito | Alta/Esencial | Media/Deseado | Baja/ Opcional |

**Descripción:**

El PLC dispondrá de:

* 8 entradas digitales optoaisladas, compatibles con 0–24 VDC y con filtrado contra rebotes.
* 8 salidas digitales tipo relé o MOSFET para cargas de hasta 5A DC.

# 4 entradas analógicas con rango 0–10 VDC o 4–20 mA y resolución mínima de 12 bits.

* Alimentación de 24 VDC con protecciones (fusibles y diodos), reguladores internos de 5V/3.3V.
* Conexiones mediante borneras desmontables con identificación clara.

### 

### Interfaces de software

|  |  |  |  |
| --- | --- | --- | --- |
| Número de requisito | IF-03 | | |
| Nombre de requisito | Interfaces de software | | |
| Tipo | Requisito | Restricción | |
| Fuente del requisito | Equipo de desarrollo | | |
| Prioridad del requisito | Alta/Esencial | Media/Deseado | Baja/ Opcional |

**Descripción:**

El firmware estará basado en C/C++ empleando ESP-IDF o Arduino Core. Incluirá librerías estándar para manejo de E/S, comunicación serial, RS-485 y Wi-Fi. Se dispondrá de una API interna con funciones como read\_input(), write\_output(), read\_analog(). Además, contará con interfaz de configuración mediante aplicación en PC (serial/USB) o interfaz web sencilla vía Wi-Fi.

### 

### Interfaces de comunicación

|  |  |  |  |
| --- | --- | --- | --- |
| Número de requisito | IF-04 | | |
| Nombre de requisito | Interfaces de comunicación | | |
| Tipo | Requisito | Restricción | |
| Fuente del requisito | Director del proyecto / Equipo de desarrollo | | |
| Prioridad del requisito | Alta/Esencial | Media/Deseado | Baja/ Opcional |

**Descripción:**

El PLC deberá incluir:

* USB-UART para programación y diagnóstico.
* RS-485 con protocolo Modbus RTU, velocidades configurables (9600 a 115200 bps).
* Conectividad Wi-Fi para Modbus TCP o HTTP básico, con acceso desde PC o móvil.
* Protocolo interno simple, basado en comandos tipo texto, para configuración vía consola serial.

## 

## Requisitos funcionales

Los requisitos funcionales definen las acciones principales que el PLC4uni debe realizar, considerando validación de entradas, secuencia de operaciones, manejo de errores y generación de salidas.

|  |  |  |  |
| --- | --- | --- | --- |
| Número de requisito | RF-1 | | |
| Nombre de requisito | Arranque y autodiagnóstico | | |
| Tipo | Requisito | Restricción | |
| Fuente del requisito | Director del proyecto | | |
| Prioridad del requisito | Alta/Esencial | Media/Deseado | Baja/ Opcional |

**Descripción:**

Al encenderse, el PLC debe ejecutar un autodiagnóstico que verifique memoria, estado de entradas, salidas y comunicación. El resultado debe mostrarse en LEDs y en la pantalla. La duración máxima del test será de 5 segundos.

|  |  |  |  |
| --- | --- | --- | --- |
| Número de requisito | RF-2 | | |
| Nombre de requisito | Lectura de entradas digitales | | |
| Tipo | Requisito | Restricción | |
| Fuente del requisito | Equipo de desarrollo | | |
| Prioridad del requisito | Alta/Esencial | Media/Deseado | Baja/ Opcional |

**Descripción:**

El sistema debe leer las ocho entradas digitales y mostrar su estado tanto en la pantalla como a través de Modbus. La latencia máxima de actualización será de 50 milisegundos.

|  |  |  |  |
| --- | --- | --- | --- |
| Número de requisito | RF-3 | | |
| Nombre de requisito | Control de salidas digitales | | |
| Tipo | Requisito | Restricción | |
| Fuente del requisito | Director del proyecto | | |
| Prioridad del requisito | Alta/Esencial | Media/Deseado | Baja/ Opcional |

**Descripción:**

El PLC debe permitir activar o desactivar cada salida digital desde la interfaz local o mediante comandos Modbus. El tiempo máximo de reacción será de 50 milisegundos.

|  |  |  |  |
| --- | --- | --- | --- |
| Número de requisito | RF-4 | | |
| Nombre de requisito | Lectura de entradas analógicas | | |
| Tipo | Requisito | Restricción | |
| Fuente del requisito | Equipo de desarrollo | | |
| Prioridad del requisito | Alta/Esencial | Media/Deseado | Baja/ Opcional |

**Descripción:**

El PLC debe muestrear las cuatro entradas analógicas con una frecuencia mínima de 5 Hz y una resolución de 12 bits, entregando los valores en unidades de voltaje o corriente.

|  |  |  |  |
| --- | --- | --- | --- |
| Número de requisito | RF-5 | | |
| Nombre de requisito | Comunicación RS-485 Modbus RTU | | |
| Tipo | Requisito | Restricción | |
| Fuente del requisito | Director del proyecto | | |
| Prioridad del requisito | Alta/Esencial | Media/Deseado | Baja/ Opcional |

**Descripción:**

El PLC debe funcionar como esclavo Modbus RTU, respondiendo a comandos de lectura de entradas, escritura de salidas y consulta de registros.

|  |  |  |  |
| --- | --- | --- | --- |
| Número de requisito | RF-6 | | |
| Nombre de requisito | Comunicación Wi-Fi | | |
| Tipo | Requisito | Restricción | |
| Fuente del requisito | Equipo de desarrollo | | |
| Prioridad del requisito | Alta/Esencial | Media/Deseado | Baja/ Opcional |

**Descripción:**

El sistema debe habilitar la conexión Wi-Fi para monitoreo básico a través de Modbus TCP o interfaz web.

|  |  |  |  |
| --- | --- | --- | --- |
| Número de requisito | RF-7 | | |
| Nombre de requisito | Señal de emergencia | | |
| Tipo | Requisito | Restricción | |
| Fuente del requisito | Director del proyecto | | |
| Prioridad del requisito | Alta/Esencial | Media/Deseado | Baja/ Opcional |

**Descripción:**

El PLC debe disponer de una entrada de paro de emergencia (E-STOP). Al activarse, todas las salidas deben apagarse inmediatamente y permanecer así hasta que se reinicie el sistema de forma manual.

|  |  |  |  |
| --- | --- | --- | --- |
| Número de requisito | RF-8 | | |
| Nombre de requisito | Actualización de firmware | | |
| Tipo | Requisito | Restricción | |
| Fuente del requisito | Equipo de desarrollo | | |
| Prioridad del requisito | Alta/Esencial | Media/Deseado | Baja/ Opcional |

**Descripción:**

El PLC debe aceptar actualizaciones de firmware mediante USB o Wi-Fi, verificando la integridad del archivo mediante checksum antes de aplicarlo.

|  |  |  |  |
| --- | --- | --- | --- |
| Número de requisito | RF-9 | | |
| Nombre de requisito | Registro de eventos críticos | | |
| Tipo | Requisito | Restricción | |
| Fuente del requisito | Director de proyecto | | |
| Prioridad del requisito | Alta/Esencial | Media/Deseado | Baja/ Opcional |

**Descripción:**  
El PLC debe almacenar un registro circular con al menos 100 eventos críticos (fallos de comunicación, activación del E-STOP, errores de hardware). El registro debe poder descargarse vía USB.

## 

## Requisitos no funcionales

### Requisitos de rendimiento

|  |  |  |  |
| --- | --- | --- | --- |
| Número de requisito | RNF-01 | | |
| Nombre de requisito | Ciclo de escaneo y respuesta Modbus | | |
| Tipo | Requisito | Restricción | |
| Fuente del requisito | Director del proyecto / Equipo de desarrollo | | |
| Prioridad del requisito | Alta/Esencial | Media/Deseado | Baja/ Opcional |

### Descripción:

### El PLC debe ejecutar un ciclo completo de escaneo (lectura de entradas, ejecución de lógica simple y actualización de salidas) en ≤ 100 ms. Además, la respuesta a una petición Modbus RTU debe ser ≤ 100 ms en condiciones normales.

### Seguridad

|  |  |  |  |
| --- | --- | --- | --- |
| Número de requisito | RNF-02 | | |
| Nombre de requisito | Seguridad y protección del sistema | | |
| Tipo | Requisito | Restricción | |
| Fuente del requisito | Director del proyecto | | |
| Prioridad del requisito | Alta/Esencial | Media/Deseado | Baja/ Opcional |

**Descripción:**

El sistema debe implementar protección por contraseña para el acceso a configuración avanzada vía PC o Wi-Fi, verificación de checksum antes de aceptar actualizaciones de firmware, y aislamiento galvánico entre señales de control y potencia.

### Fiabilidad

|  |  |  |  |
| --- | --- | --- | --- |
| Número de requisito | RNF-03 | | |
| Nombre de requisito | Fiabilidad del sistema | | |
| Tipo | Requisito | Restricción | |
| Fuente del requisito | Equipo de desarrollo | | |
| Prioridad del requisito | Alta/Esencial | Media/Deseado | Baja/ Opcional |

**Descripción:**

El PLC debe garantizar un MTTF mínimo de 10,000 horas en condiciones normales de laboratorio. Debe emplear el watchdog interno del ESP32 para reinicio automático en caso de bloqueo del sistema.

### Disponibilidad

|  |  |  |  |
| --- | --- | --- | --- |
| Número de requisito | RNF-04 | | |
| Nombre de requisito | Disponibilidad operativa | | |
| Tipo | Requisito | Restricción | |
| Fuente del requisito | Director del proyecto | | |
| Prioridad del requisito | Alta/Esencial | Media/Deseado | Baja/ Opcional |

**Descripción:**

El PLC debe estar disponible ≥ 99% del tiempo en horario de prácticas. En caso de fallo, el reinicio del sistema no debe superar los 5 segundos.

### Mantenibilidad

|  |  |  |  |
| --- | --- | --- | --- |
| Número de requisito | RNF-05 | | |
| Nombre de requisito | Mantenibilidad del sistema | | |
| Tipo | Requisito | Restricción | |
| Fuente del requisito | Equipo de desarrollo | | |
| Prioridad del requisito | Alta/Esencial | Media/Deseado | Baja/ Opcional |

**Descripción:**

El firmware debe estar documentado y comentado. Se debe disponer de un manual de usuario y un manual técnico. Además, el sistema debe contar con un puerto de depuración accesible (UART/JTAG) para recuperación en caso de fallo.

### 

### Portabilidad

|  |  |  |  |
| --- | --- | --- | --- |
| Número de requisito | RNF-06 | | |
| Nombre de requisito | Portabilidad del código | | |
| Tipo | Requisito | Restricción | |
| Fuente del requisito | Director del proyecto / Equipo de desarrollo | | |
| Prioridad del requisito | Alta/Esencial | Media/Deseado | Baja/ Opcional |

**Descripción:**

El código debe estar escrito en C/C++ estándar para facilitar su portabilidad a otras placas ESP32 o STM32 en el futuro, evitando dependencias fuertes de librerías propietarias.

## Otros requisitos

|  |  |  |  |
| --- | --- | --- | --- |
| Número de requisito | OR-01 | | |
| Nombre de requisito | Requisitos legales, académicos y culturales | | |
| Tipo | Requisito | Restricción | |
| Fuente del requisito | Director del proyecto / Equipo de desarrollo | | |
| Prioridad del requisito | Alta/Esencial | Media/Deseado | Baja/ Opcional |

**Descripción:**

El sistema debe cumplir con normas básicas de seguridad eléctrica y compatibilidad electromagnética para uso en laboratorios educativos. Además, el diseño debe estar documentado como material de aprendizaje (diagramas, explicaciones, guías). Toda la documentación debe elaborarse en español, con posibilidad de versión en inglés si la universidad lo requiere.

# Apéndices

**Apéndice A — Repositorio oficial del proyecto:**

Repositorio en GitHub donde se almacenan los diagramas electrónicos, el firmware, la documentación y los avances del proyecto:

* **Enlace al respositorio:** https://github.com/HABP0705/Henry\_Bautista\_PLC4uni.git

**Apéndice B — Redes sociales del proyecto:**

Para difusión y seguimiento académico se podrán utilizar los siguientes canales:

* Página oficial de la asignatura en la plataforma de la universidad.
* Grupo de WhatsApp del curso de Mecatrónica para coordinación interna.
* Publicaciones en redes sociales académicas de la universidad (Facebook, Instagram).

**Apéndice C — Cronograma resumido:**

* **Semana 1:** Presentación del proyecto, explicación del objetivo y conformación del equipo.
* **Semana 2:** Definición detallada de requisitos y entrega del documento SRS inicial.
* **Semana 3:** Selección de componentes principales y diseño conceptual de la arquitectura.
* **Semana 4:** Elaboración de los primeros esquemáticos y discusión con el profesor.
* **Semana 5:** Revisión y ajustes de esquemáticos; preparación del diseño de la PCB.
* **Semana 6:** Finalización del diseño de la PCB y validación antes de fabricación.
* **Semana 7:** Solicitud o fabricación de PCB y adquisición de componentes.
* **Semana 8:** Montaje inicial del prototipo en laboratorio.
* **Semana 9:** Pruebas de hardware (entradas, salidas, alimentación).
* **Semana 10:** Desarrollo del firmware básico (lectura de entradas/salidas).
* **Semana 11:** Integración de firmware avanzado (comunicación RS-485, Wi-Fi).
* **Semana 12:** Pruebas de validación de requisitos funcionales y no funcionales.
* **Semana 13:** Corrección de fallos, ajustes de hardware/firmware y redacción de manuales.
* **Semana 14:** Presentación final del proyecto, entrega de documentación y defensa del trabajo.

**Apéndice D — Contacto del equipo:**

**Director del proyecto:** Ing. Carlos Pichardo — correo institucional: cpichardo@itla.edu.do

**Integrante del equipo de desarrollo:** Henry Abdias Bautista Portes — correo: 20231696@itla.edu.do