|  |
| --- |
|  |

Especificación de requisitos de software

Proyecto:

Revisión

|  |  |  |
| --- | --- | --- |
|  |  |  |

**Instrucciones para el uso de este formato**

Este formato es una plantilla tipo para documentos de requisitos del software.

Está basado y es conforme con el estándar IEEE Std 830-1998.

Las secciones que no se consideren aplicables al sistema descrito podrán de forma justificada indicarse como no aplicables (NA).

Notas:

Los textos en color azul son indicaciones que deben eliminarse y, en su caso, sustituirse por los contenidos descritos en cada apartado.

Los textos entre corchetes del tipo “” permiten la inclusión directa de texto con el color y estilo adecuado a la sección, al pulsar sobre ellos con el puntero del ratón.

Los títulos y subtítulos de cada apartado están definidos como estilos de MS Word, de forma que su numeración consecutiva se genera automáticamente según se trate de estilos “Titulo1, Titulo2 y Titulo3”.

La sangría de los textos dentro de cada apartado se genera automáticamente al pulsar Intro al final de la línea de título. (Estilos Normal indentado1, Normal indentado 2 y Normal indentado 3).

El índice del documento es una tabla de contenido que MS Word actualiza tomando como criterio los títulos del documento.

Una vez terminada su redacción debe indicarse a Word que actualice todo su contenido para reflejar el contenido definitivo.

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

.

Ficha del documento

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Revisión** | **Autor** | **Verificado dep. calidad.** |
| 18/9/2025 | 1 | Felipe Leonardo | Ing. Carlos Pichardo - Director de Proyecto |

Documento validado por las partes en fecha:

|  |  |
| --- | --- |
| Por el cliente | Por la empresa suministradora |
|  |  |
| Carlos Pichardo (director del proyecto) | Equipo de Estudiantes (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

El presente documento constituye la Especificación de Requisitos de Software (SRS) del proyecto PLC4Units, un Controlador Lógico Programable (PLC) diseñado con fines educativos. Su objetivo principal es ofrecer una solución asequible y funcional que permita a estudiantes, docentes e instituciones educativas realizar prácticas de automatización y control sin incurrir en los elevados costos que suelen presentar los PLC comerciales.

Este SRS describe de manera estructurada los objetivos, el alcance, las definiciones, referencias y requisitos del sistema, proporcionando una base sólida para el diseño, implementación, validación y posterior mantenimiento del PLC4Units.

El propósito del documento es servir como guía tanto para el equipo de desarrollo como para los colaboradores y partes interesadas en el proyecto, definiendo con claridad las características funcionales del sistema, sus restricciones y dependencias.

**Objetivo:**

Creación de PLC educativo, (Controlador Lógico Programable) - **PLC4Units**

En este documento vamos a ir plasmando las ideas principales para la creación de un PLC con propósito educativo para atacar la falta de prácticas técnicas educativas en PLC haciendo uno asequible para las instituciones o estudiantes de forma que pueda satisfacer las necesidades básicas educativas.

## Propósito

El alcance del PLC4Units contempla un producto orientado a la enseñanza, capaz de replicar las funciones básicas de un PLC comercial, incluyendo manejo de entradas y salidas digitales/analógicas, programación en lenguajes estándar de PLC y simulación de procesos básicos, todo dentro de un entorno amigable y con costos reducidos.

La audiencia principal incluye:

* Estudiantes y docentes de áreas técnicas como electrónica, automatización y mecatrónica.
* Instituciones educativas que requieran equipos asequibles para prácticas con PLC.
* Desarrolladores, diseñadores y colaboradores involucrados en la construcción del PLC4Units.

## Alcance

El propósito de este documento es definir de manera clara y estructurada los requisitos del sistema para el desarrollo del **PLC4Units**, un Controlador Lógico Programable con fines educativos.   
Este documento servirá como guía para el diseño, implementación y validación del producto, estableciendo las características funcionales y no funcionales, así como las restricciones y dependencias.   
La audiencia principal de este documento son:

* Estudiantes y docentes en áreas técnicas (electrónica, automatización, mecatrónica).
* Instituciones educativas que requieran equipos asequibles para prácticas con PLC.
* Desarrolladores, diseñadores y colaboradores del proyecto.

## Personal involucrado

|  |  |
| --- | --- |
| Nombre | 1- Felipe M. Leonardo G,  2- Ing. Carlos Antonio Pichardo Viuque. |
| Rol | 1- Creador y desarrollador del proyecto,  2- CEO (Chief Executive Officer) ( Director de Proyecto) |
| Categoría profesional | 1- Ingeniero en Automatización: Experto en diseño, programación e implementación de sistemas de control.  2- Ejecutivo de Negocios y Estrategia: Responsable de la dirección y toma de decisiones clave del proyecto. |
| Responsabilidades | **1- Diseño y desarrollo técnico:** Encargado de la arquitectura de hardware y software del **PLC4Units**, la programación, las pruebas de funcionalidad, y la documentación técnica.  2- Responsable de la planificación estratégica, la gestión de recursos, la viabilidad financiera, y la supervisión general del proyecto. |
| Información de contacto | leonar2galan@gmail.com |
| Aprobación |  |

## Definiciones, acrónimos y abreviaturas

**PLC (Controlador Lógico Programable):** Dispositivo electrónico industrial diseñado para controlar procesos automáticos mediante programación.

**PLC4Units:** Nombre del proyecto que designa el Controlador Lógico Programable educativo objeto de esta especificación.

**IEC 61131-3:** Estándar internacional que define los lenguajes de programación para PLCs.

**Ladder Diagram (LD) / Diagrama de Escalera:** Lenguaje de programación gráfico basado en esquemas de circuitos eléctricos.

**Function Block Diagram (FBD):** Lenguaje de programación gráfico que utiliza bloques funcionales.

**Structured Text (ST):** Lenguaje de programación textual de alto nivel similar a Pascal.

**IDE (Entorno de Desarrollo Integrado):** Software para crear, editar y depurar programas.

**API (Interfaz de Programación de Aplicaciones):** Conjunto de funciones para interactuar con el sistema.

**ESP-IDF (Espressif IoT Development Framework):** Framework oficial para programar chips ESP32.

**GPIO (Entrada/Salida de Propósito General):** Pines programables del microcontrolador.

**I/O (Entradas/Salidas):** Terminales para conectar sensores y actuadores.

USB CDC (Communications Device Class): Protocolo para comunicación serial sobre USB.

**Modbus RTU:** Protocolo de comunicación industrial serie.

**Wi-Fi (Wireless Fidelity):** Tecnología de red inalámbrica.

**HTML (Lenguaje de Marcado de Hipertexto):**Lenguaje para crear páginas web.

**HMI (Interfaz Hombre-Máquina):** Dispositivo para interactuar con sistemas automatizados.

**SCADA (Sistema de Control y Adquisición de Datos):** Sistema de supervisión de procesos.

**OTA (Actualización Inalámbrica):** Capacidad de actualizar firmware remotamente.

**LED (Diodo Emisor de Luz):** Componente para indicaciones visuales.

**DC (Corriente Continua):** Tipo de alimentación eléctrica.

**VDC (Voltios de Corriente Continua):** Unidad de medida de voltaje.

**mA (miliamperios):** Unidad de medida de corriente.

## Referencias

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Referencia** | **Titulo** | **Ruta** | **Fecha** | **Autor** |
| 1 | Schneider Electric Modicon PLCs | https://www.se.com | 2025 | Schneider Electric |
| 2 | Mitsubishi MELSEC PLCs | https://www.mitsubishielectric.com/fa | 2025 | Mitsubishi Electric |
| 3 | Omron Sysmac PLCs | https://automation.omron.com | 2025 | Omron |
| 4 | Documentación oficial ESP32 | https://docs.espressif.com | 2025 | Espressif Systems |
| 5 |  |  |  |  |
| 6 |  |  |  |  |

Relación completa de todos los documentos relacionados en la especificación de requisitos de software, identificando de cada documento el titulo, referencia (si procede), fecha y organización que lo proporciona.

## Resumen

Este documento constituye la Especificación de Requisitos de Software (SRS) para el proyecto PLC4Units, un Controlador Lógico Programable educativo. El documento se organiza en cuatro secciones principales que establecen los fundamentos para el desarrollo del sistema.

La Introducción (Sección 1) define el propósito, alcance y objetivos del proyecto, identificando a los stakeholders principales y el contexto general de desarrollo. La Descripción General (Sección 2) presenta una visión del producto, su funcionalidad básica, características de los usuarios, restricciones y evolución previsible.

Los Requisitos Específicos (Sección 3) constituyen el núcleo del documento, detallando de manera exhaustiva las interfaces del sistema (usuario, hardware, software y comunicación), los requisitos funcionales esenciales y los requisitos no funcionales críticos como rendimiento, seguridad, fiabilidad y mantenibilidad. Finalmente, los Apéndices (Sección 4) contendrán información complementaria relevante para el proyecto.

Esta estructura sigue el estándar IEEE 830-1998 y garantiza que todos los aspectos técnicos y funcionales del PLC4Units queden especificados de manera clara y completa para guiar el proceso de desarrollo.

# Descripción general

## Perspectiva del producto

El PLC4Units es un producto independiente diseñado específicamente con fines educativos. No forma parte de un sistema mayor preexistente, aunque está concebido para integrarse de manera flexible con otros dispositivos y módulos complementarios (sensores, actuadores, software de simulación, etc.) utilizados en prácticas académicas.

Este controlador ofrece las funcionalidades básicas de un PLC comercial, pero adaptadas para un entorno de aprendizaje. Su arquitectura modular permite conectar entradas y salidas digitales y analógicas, así como comunicarse con computadoras u otros equipos para su programación y monitoreo.

En el contexto de un laboratorio educativo, el PLC4Units puede actuar como núcleo central de prácticas de automatización y control, al que se pueden añadir periféricos o maquetas de procesos para simular escenarios industriales reales.

## Funcionalidad del producto

El PLC4Units proporcionará las funciones básicas que necesita un Controlador Lógico Programable de propósito educativo, permitiendo a estudiantes y docentes realizar prácticas reales de automatización y control. De forma resumida, el producto deberá ser capaz de:

**Gestión de Entradas y Salidas**

* Lectura de entradas digitales y analógicas.
* Activación de salidas digitales y generación de salidas analógicas.

**Programación en Lenguajes Estándar de PLC**

* Compatibilidad con lenguajes definidos por la norma IEC 61131-3, como Ladder Diagram (LD), Function Block Diagram (FBD) y Structured Text (ST).
* Carga y ejecución de programas creados por el usuario.

**Simulación y Monitoreo**

* Ejecución de programas con simulación de procesos básicos sin necesidad de hardware externo.
* Visualización del estado de las entradas, salidas y variables internas en tiempo real.

**Comunicación y Conectividad**

* Interfaz para conexión con PC u otros dispositivos para su programación, monitoreo y actualización.
* Posibilidad de integración con sensores, actuadores o maquetas educativas a través de módulos complementarios.

**Entorno Amigable y Seguro**

* Interfaz sencilla para facilitar la curva de aprendizaje.

Estas funcionalidades permitirán a los usuarios desarrollar, probar y ejecutar programas de automatización, así como comprender de manera práctica los conceptos fundamentales de un PLC comercial en un entorno educativo y de bajo costo.

## Características de los usuarios

|  |  |
| --- | --- |
| Tipo de usuario | Estudiantes en tecnologías, docentes o Instructores. |
| Formación | Cursan carreras técnicas o universitarias en áreas como electrónica, automatización, mecatrónica o ingeniería industrial o áreas a fines.  Profesionales con experiencia en enseñanza de áreas técnicas y en el uso de PLC comerciales. |
| Habilidades | Conocimientos en electricidad y programación; interés en aprender o enseñar conceptos de control y automatización. |
| Actividades | Realizar prácticas de programación de PLC, simulación de procesos, montaje de sistemas básicos de control. |

## 

## Restricciones

El desarrollo del **PLC4Units** debe considerar las siguientes restricciones y limitaciones para garantizar que el producto cumpla con su objetivo educativo y de bajo costo:

* **Costos limitados:** El diseño debe mantener un presupuesto asequible para estudiantes e instituciones, priorizando componentes económicos sin comprometer la funcionalidad básica de un PLC.
* **Hardware compacto y modular:** El sistema debe construirse sobre una plataforma de hardware de bajo consumo energético y tamaño reducido, facilitando su uso en entornos educativos con espacio limitado.
* **Compatibilidad con lenguajes estándar:** El software del PLC4Units debe ser compatible con lenguajes de programación PLC definidos en la norma IEC 61131-3 (LD, FBD, ST), sin incluir lenguajes propietarios que encarezcan el producto.
* **Restricciones de seguridad eléctrica:** El equipo debe cumplir con normas básicas de seguridad eléctrica para evitar riesgos durante las prácticas educativas.
* **Limitación en cantidad de entradas y salidas:** Para mantener el costo bajo, el número de entradas y salidas digitales/analógicas será inferior al de un PLC comercial de gama media, pero suficiente para prácticas básicas.
* **Sistema operativo y herramientas de desarrollo:** El software y las herramientas utilizadas para programar el PLC4Units deben ser multiplataforma (Windows/Linux) y, en la medida de lo posible, de código abierto para evitar costos de licencias.
* **Cumplimiento normativo:** Se deben respetar las normas y estándares educativos y técnicos aplicables al uso de equipos electrónicos en laboratorios y aulas.

## Suposiciones y dependencias

Durante el desarrollo del **PLC4Units**, se establecen las siguientes **suposiciones** y **dependencias**:

**Suposiciones:**

1. Se asume que los usuarios contarán con conocimientos básicos de electrónica y programación de PLC para aprovechar plenamente las funcionalidades del PLC4Units.
2. Se supone que las instituciones educativas dispondrán de instalaciones y fuentes de alimentación compatibles con el hardware del PLC4Units.
3. Se considera que los estudiantes y docentes tienen acceso a computadoras que cumplan los requisitos mínimos del software de programación del PLC.
4. Se asume que el equipo se utilizará únicamente con fines educativos, no para aplicaciones industriales críticas.

**Dependencias:**

1. Hardware: El funcionamiento del PLC4Units depende de la disponibilidad de microcontroladores, módulos de entradas/salidas y componentes electrónicos específicos.
2. Software: El PLC4Units depende de herramientas de programación compatibles con IEC 61131-3 (Ladder, FBD, ST), así como del software de simulación y control desarrollado para la plataforma.
3. Normativas y estándares: El diseño depende de la correcta implementación de normas eléctricas y de seguridad aplicables a entornos educativos.
4. Recursos externos: El proyecto puede depender de bibliotecas, controladores o documentación externa para ciertos módulos de software o hardware.

## Evolución previsible del sistema

El **PLC4Units** está diseñado para ser flexible, permitiendo futuras mejoras que amplíen su funcionalidad y capacidad educativa. Entre las evoluciones previstas se incluyen:

 **Ampliación de entradas y salidas:** Incrementar el número de canales digitales y analógicos para prácticas más completas.

 **Módulos de expansión externos:** Incorporar módulos adicionales que permitan mayor flexibilidad y nuevas aplicaciones educativas.

 **Mejoras en programación:** Implementar un entorno de programación Ladder simplificado y compatible con otros lenguajes estándar IEC 61131-3 (FBD, ST) para facilitar el aprendizaje.

 **Materiales didácticos complementarios**: Crear manuales de prácticas, simuladores virtuales y guías educativas que faciliten el aprendizaje y la enseñanza.

# Requisitos específicos

Esta es la sección más extensa y más importante del documento.

Debe contener una lista detallada y completa de los requisitos que debe cumplir el sistema a desarrollar. El nivel de detalle de los requisitos debe ser el suficiente para que el equipo de desarrollo pueda diseñar un sistema que satisfaga los requisitos y los encargados de las pruebas puedan determinar si éstos se satisfacen.

Los requisitos se dispondrán en forma de listas numeradas para su identificación, seguimiento, trazabilidad y validación (ej. RF 10, RF 10.1, RF 10.2,...).

Para cada requisito debe completarse la siguiente tabla:

|  |  |  |  |
| --- | --- | --- | --- |
| Número de requisito |  | | |
| Nombre de requisito |  | | |
| Tipo | Requisito | Restricción | |
| Fuente del requisito |  | | |
| Prioridad del requisito | Alta/Esencial | Media/Deseado | Baja/ Opcional |

y realizar la descripción del requisito

La distribución de los párrafos que forman este punto puede diferir del propuesto en esta plantilla, si las características del sistema aconsejan otra distribución para ofrecer mayor claridad en la exposición.

## Requisitos comunes de los interfaces

El sistema **PLC4Units** deberá cumplir con las siguientes funcionalidades principales:

* **Programación en lenguajes estándar**: permitir la programación en lenguaje Ladder simplificado (orientado a estudiantes) y, de forma opcional, en Texto Estructurado u otros lenguajes IEC 61131-3.
* **Manejo de entradas y salidas**: soportar entradas y salidas digitales y analógicas básicas para replicar procesos reales en entornos educativos.
* **Simulación integrada**: incluir un entorno de simulación que muestre en tiempo real el estado de entradas, salidas y variables internas del programa.
* **Transferencia de programas**: permitir la carga y descarga de programas desde y hacia un PC mediante conexión USB o inalámbrica.
* **Ejecución segura**: garantizar la ejecución estable de los programas cargados, evitando bloqueos y asegurando la protección de hardware y usuarios.

### Interfaces de usuario

|  |  |  |  |
| --- | --- | --- | --- |
| Número de requisito | IF-10.1 | | |
| 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 |

El sistema PLC4Units contará con dos interfaces de usuario principales diseñadas para ser intuitivas y didácticas. La primera es una Interfaz de Programación (IDE) en PC, una aplicación de escritorio con una apariencia moderna y sobria. Su área central es un editor visual donde los usuarios pueden arrastrar y soltar componentes como contactos y bobinas para construir programas en Ladder, con resaltado de sintaxis para otros lenguajes. Incluye menús estándar, una barra de herramientas de acceso rápido, un explorador de proyectos para organizar los elementos y una tabla de variables para gestionar y monitorear el estado de las entradas, salidas y datos internos en tiempo real durante la simulación o ejecución.

La segunda es una Interfaz de Estado física integrada en el hardware del PLC.

Esta interfaz proporciona información operativa básica mediante una serie de LEDs indicadores: un LED de Alimentación (verde), otro de Ejecución (verde) que muestra si el programa se está ejecutando, y un LED de Error/Comunicación (rojo/ámbar) para alertas. Además, incluye LEDs individuales para mostrar el estado activo/inactivo de cada entrada y salida digital. Todos los conectores de alimentación, entradas, salidas y el puerto de comunicación USB estarán claramente etiquetados y numerados para facilitar el cableado y la identificación, asegurando una experiencia de usuario segura y sencilla en el entorno educativo.

### Interfaces de hardware

|  |  |  |  |
| --- | --- | --- | --- |
| Número de requisito | IF-10.2 | | |
| 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 contará con las siguientes especificaciones técnicas:
* 8 entradas digitales con opto aislamiento, compatibles con señales de 0 a 24 VDC, e incorporarán filtrado digital para supresión de interferencias y antirrebote (debounce).
* 8 salidas digitales, que podrán implementarse con tecnología de relé o MOSFET, capaces de manejar cargas de hasta 5A en corriente continua (DC).
* entradas analógicas configurables, que admitirán señales en los rangos estándar de 0-10 VDC o 4-20 mA, ofreciendo una resolución mínima de 12 bits para una digitalización precisa de la señal.
* Alimentación principal de 24 VDC, que integrará protecciones contra sobretensiones y polaridad inversa mediante fusibles y diodos, así como reguladores de voltaje internos para generar los niveles de 5V y 3.3V requeridos por la electrónica de control.
* Sistema de conexión mediante borneras o terminales de bloque desmontables, las cuales estarán claramente identificadas con serigrafía para un cableado intuitivo y libre de errores.

### 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 |

El núcleo del firmware del PLC4Units será desarrollado en los lenguajes de programación C y C++. Para aprovechar las capacidades del hardware y garantizar un rendimiento robusto, se utilizará el framework oficial ESP-IDF (IoT Development Framework) como opción principal, contemplándose también la posibilidad de emplear Arduino Core para agilizar el desarrollo de prototipos en fases iniciales.

El sistema software integrará un conjunto de librerías estándar especializadas para la gestión de bajo nivel de los periféricos del hardware. Estas librerías proporcionarán las funcionalidades esenciales para:

* El manejo de Entradas y Salidas (E/S) digitales.
* La gestión de comunicaciones, incluyendo serial (UART), el protocolo industrial RS-485 para potenciales expansiones, y la conectividad Wi-Fi.
* El control de temporizadores, interrupciones y otras funciones del sistema.

Se definirá una API (Interfaz de Programación de Aplicaciones) interna bien estructurada, que actuará como una capa de abstracción entre la lógica de aplicación principal y el hardware subyacente.

### 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 |

**Conexión Principal (USB):**

* Cómo: Cable USB.
* Para qué: Programar el PLC y ver lo que hace en tiempo real desde la computadora. Es la forma más fácil y directa de conectarse.

**Conexión Industrial Básica (RS-485/Modbus):**

* Cómo: Terminales de tornillo en el PLC.
* Para qué: Conectar varios PLCs entre sí o a pantallas (HMI) simples, imitando cómo se comunican los equipos en una fábrica real.

**Conexión Inalámbrica (Wi-Fi):**

* Cómo: Chip integrado en el PLC.
* Para qué:
* Conectarse al Wi-Fi del laboratorio.
* Crear su propia red para conectarse directamente sin internet.
* Monitorear el PLC desde un celular o tableta usando una página web simple.

## Requisitos funcionales

El sistema PLC4Units deberá realizar las siguientes funciones fundamentales para cumplir con su propósito educativo. Estas funciones definen las acciones específicas que el software debe ejecutar al procesar entradas, implementar la lógica de control y generar las salidas correspondientes.

### Requisito funcional 1

|  |  |  |  |
| --- | --- | --- | --- |
| Número de requisito | RNF-01 | | |
| Nombre de requisito | **Gestión del Ciclo de Escaneo** | | |
| Tipo | Requisito | Restricción | |
| Fuente del requisito | Director del proyecto / Equipo de desarrollo | | |
| Prioridad del requisito | Alta/Esencial | Media/Deseado | Baja/ Opcional |

Descripción:

El firmware del PLC debe ejecutar un ciclo de escaneo continuo y deterministico. Cada ciclo debe seguir estrictamente la secuencia:

1. Lectura de Entradas: Leer el estado físico de todas las entradas digitales y analógicas y almacenar sus valores en la memoria de imagen de entradas.
2. Ejecución del Programa: Ejecutar la lógica de control del programa de usuario (p. ej., diagrama ladder, texto estructurado) utilizando los valores de la memoria de imagen de entradas.
3. Escritura de Salidas: Actualizar el estado físico de todas las salidas digitales y analógicas según los resultados calculados en la memoria de imagen de salidas.  
   El sistema debe garantizar que el tiempo de este ciclo completo sea de 100 ms como máximo para aplicaciones educativas típicas.

### Requisito funcional 2

|  |  |  |  |
| --- | --- | --- | --- |
| Número de requisito | RNF-01 | | |
| Nombre de requisito | **Gestión del Ciclo de Escaneo** | | |
| 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 permitir la creación, edición y ejecución de programas de control utilizando el lenguaje Ladder Diagram (LD), conforme a la norma IEC 61131-3. Debe soportar los elementos básicos:

* Contactos Normalmente Abiertos (NO) y Normalmente Cerrados (NC).
* Bobinas (Coils) de salida, set (S) y reset (R).
* Temporizadores (Timers): TON (Timer On-Delay), con base de tiempo configurable.
* Contadores (Counters): CTU (Count Up).   
  El runtime del firmware debe ser capaz de interpretar el bytecode compilado del programa ladder y ejecutar la lógica en cada ciclo de escaneo, respetando la evaluación de rungs de izquierda a derecha y de arriba a abajo.

### Requisito funcional 3

|  |  |  |  |
| --- | --- | --- | --- |
| Número de requisito | RNF-01 | | |
| Nombre de requisito | **Gestión del Ciclo de Escaneo** | | |
| 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 permitir la transferencia bidireccional de programas entre el IDE en el PC y el hardware del PLC. A través del puerto USB (emulación de puerto serie CDC):

* Carga (Download): El firmware debe recibir el programa compilado (bytecode) desde el IDE, almacenarlo en memoria no volátil y prepararse para su ejecución.
* Descarga (Upload): El firmware debe ser capaz de enviar el programa almacenado de vuelta al IDE upon request.
* Control de Ejecución: El firmware debe responder a comandos remotos del IDE para iniciar (RUN), detener (STOP) y monitorear el estado del programa (STATUS).  
  La comunicación debe implementar un protocolo simple con checksum de verificación de integridad de datos para asegurar que las transferencias sean libres de errores.

## 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 sistema del PLC debe garantizar un tiempo de ciclo de operación completo, que abarque la lectura de todas las entradas, la ejecución de la lógica de control programada por el usuario y la actualización de las salidas, con una duración máxima de 100 milisegundos.

### 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 |

El sistema incorporará medidas básicas de seguridad centradas en la protección del acceso y la integridad del software. Todas las funciones críticas, como la programación y modificación de configuración, requerirán autenticación por contraseña configurable. Las comunicaciones con la interfaz web de gestión utilizarán conexiones HTTPS para prevenir interceptación de datos.

Para garantizar la integridad del sistema, toda transferencia de firmware o programas incluirá verificaciones criptográficas que impedirán la instalación de código malicioso o corrupto. El diseño contará con un watchdog timer que supervisará el funcionamiento y ejecutará reinicios controlados ante bloqueos. La arquitectura modular aislará funcionalidades críticas para limitar el impacto de posibles fallos. Estas medidas asegurarán un entorno estable y confiable para prácticas educativas.

### Fiabilidad

El sistema PLC4Units deberá garantizar un alto nivel de fiabilidad para permitir su uso continuo en entornos educativos. El hardware tendrá un tiempo medio entre fallos (MTBF) mínimo de 5.000 horas en condiciones normales de laboratorio, mientras que el software deberá mantener una disponibilidad del 99 % durante las prácticas y simulaciones, evitando bloqueos y pérdidas de datos.

Se considera aceptable un máximo de un incidente crítico por semestre académico en cada equipo, suponiendo un uso regular de entre 4 y 6 horas diarias. Las actualizaciones y parches del entorno de programación deberán planificarse para minimizar interrupciones, asegurando que las clases en curso no se vean afectadas.

### Disponibilidad

El sistema PLC4Units debe garantizar una alta disponibilidad operativa durante su uso en entornos educativos. Se exige una disponibilidad igual o superior al 99.5% en horario lectivo típico (8 horas/día, 5 días/semana), lo que permite un margen mínimo de interrupciones no planificadas para no afectar el desarrollo de las prácticas.

Esta métrica se aplica específicamente al funcionamiento del firmware y hardware central del PLC, asegurando su capacidad para ejecutar programas, responder a entradas/salidas y comunicarse con el software de monitorización. Quedan excluidas las interrupciones por errores en el programa del usuario, fallos en el software IDE del PC o el tiempo dedicado a mantenimiento planificado como actualizaciones de firmware.

Para lograr este objetivo, se implementarán estrategias como un watchdog timer que supervise y reinicie el sistema ante bloqueos.

### 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 |

El sistema debe ser fácil de mantener y actualizar. El código estará organizado en módulos independientes con documentación clara, lo que permitirá modificar o corregir una parte sin afectar el resto del sistema.

Se implementará un sistema de control de versiones para gestionar los cambios en el firmware y software. Las actualizaciones del firmware podrán realizarse por USB o de manera inalámbrica, con verificación automática para garantizar su correcta instalación.

El diseño incluirá funciones de diagnóstico que permitirán identificar problemas de manera rápida mediante mensajes de error claros y herramientas de monitoreo accesibles.

### Portabilidad

El sistema PLC4Units priorizará una portabilidad integral que garantice su adaptabilidad a diversos entornos educativos y futuras evoluciones tecnológicas. A nivel de software, el diseño se basará en frameworks multiplataforma y herramientas de código abierto para asegurar compatibilidad nativa con Windows, Linux y macOS sin costos de licencias. El núcleo del firmware se desarrollará en C/C++ estándar, evitando dependencias de compiladores específicos, mientras que una capa de abstracción de hardware (HAL) aislará la lógica de aplicación de los drivers específicos, facilitando migraciones entre plataformas.

En cuanto al hardware, el PLC se diseñará con un factor de forma compacto (máx. 100x80x60 mm) y compatible con rieles DIN estándar (EN 60715 TH35-7.5), permitiendo su integración en bancos de prueba existentes. La electrónica seguirá un enfoque modular, con un sistema de procesamiento (SoM) separable de las interfaces de E/S y conectores industrializados para facilitar su adaptación y actualización.

Todas las herramientas de desarrollo y dependencias software serán gratuitas y de código abierto, asegurando la accesibilidad para instituciones educativas. La estrategia incluirá integración continua multiplataforma y containerización para mantener la consistencia entre entornos, con el objetivo de que más del 85% del código sea platform-agnóstico y el porting a nuevas arquitecturas requiera menos de 2 meses-hombre de esfuerzo.

## 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 |

El sistema cumplirá con normas básicas de seguridad eléctrica y compatibilidad electromagnética para uso en laboratorios educativos.

# Apéndices

# Esquema de Numeración de Requisitos

El sistema de numeración utilizado en este documento sigue la siguiente estructura:

* **IF-XX**: Requisitos de Interfaces
* **RF-XX**: Requisitos Funcionales
* **RNF-XX**: Requisitos No Funcionales
* **OR-XX**: Otros Requisitos

Donde XX representa un número secuencial único para cada categoría.

**4.2 Estándares de Calidad Aplicables**

El desarrollo del PLC4Units deberá cumplir con los siguientes estándares de calidad:

* **IEEE 830-1998**: Para la especificación de requisitos de software
* **IEC 61131-3**: Para lenguajes de programación de PLCs
* **ISO 9001**: Principios de gestión de calidad (como referencia)
* **Normas de seguridad eléctrica**: IEC 61010-1 para equipos de laboratorio

**4.3 Supuestos Técnicos Adicionales**

1. **Compatibilidad de navegadores**: La interfaz web será compatible con las últimas versiones de Chrome, Firefox y Edge
2. **Sistemas operativos soportados**: Windows 10/11, Linux Ubuntu 20.04+, macOS 12+
3. **Resolución mínima**: 1024x768 píxeles para la interfaz IDE

**4.4 Glosario de Términos Técnicos**

**Ciclo de Escaneo**: Proceso repetitivo que ejecuta un PLC para leer entradas, procesar la lógica y actualizar salidas.

**Watchdog Timer**: Mecanismo de hardware/software que supervisa el correcto funcionamiento del sistema.

**Optoaislamiento**: Técnica de separación eléctrica entre circuitos usando componentes ópticos.

**Bytecode**: Código intermedio resultado de la compilación, interpretable por el runtime.

**4.5 Referencias a Documentos Externos**

* Manual de usuario del PLC4Units (por desarrollar)
* Especificaciones técnicas del hardware (por desarrollar)
* Guía de instalación y configuración (por desarrollar)
* Protocolo de comunicación PLC-IDE (por desarrollar)