

---

# Especificación de requisitos de software

Proyecto: PLC 4 UNI  
Revisión [99.99]



09/2025  
[Mes de año]

# **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 “[Inserte aquí el texto]” 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.*

# Ficha del documento

| Fecha      | Revisión | Autor                | Verificado dep. calidad. |
|------------|----------|----------------------|--------------------------|
| 16/09/2025 | 16/09    | Cristopher Perez Roa | [Firma o sello]          |

Documento validado por las partes en fecha: [Fecha]

| Por el cliente        | Por la empresa suministradora |
|-----------------------|-------------------------------|
|                       |                               |
| Fdo. D./ Dña [Nombre] | Fdo. D./Dña [Nombre]          |

## Contenido

|   |           |
|---|-----------|
| <b>FICHA DEL DOCUMENTO</b>                      | <b>3</b>  |
| <b>CONTENIDO</b>                                | <b>4</b>  |
| <b>1 INTRODUCCIÓN</b>                           | <b>6</b>  |
| 1.1 Propósito                                   | 6         |
| 1.2 Alcance                                     | 6         |
| 1.3 Personal involucrado                        | 7         |
| 1.4 Definiciones, acrónimos y abreviaturas      | 7         |
| 1.5 Referencias                                 | 7         |
| 1.6 Resumen                                     | 8         |
| <b>2 DESCRIPCIÓN GENERAL</b>                    | <b>8</b>  |
| 2.1 Perspectiva del producto                    | 8         |
| 2.2 Funcionalidad del producto                  | 8         |
| 2.3 Características de los usuarios             | 9         |
| 2.4 Restricciones                               | 9         |
| 2.5 Suposiciones y dependencias                 | 9         |
| 2.6 Evolución previsible del sistema            | 9         |
| <b>3 REQUISITOS ESPECÍFICOS</b>                 | <b>10</b> |
| <b>3.1 Requisitos comunes de los interfaces</b> | <b>11</b> |
| 3.1.1 Interfaces de usuario                     | 11        |
| 3.1.2 Interfaces de hardware                    | 11        |
| 3.1.3 Interfaces de software                    | 11        |
| 3.1.4 Interfaces de comunicación                | 11        |
| <b>3.2 Requisitos funcionales</b>               | <b>11</b> |
| 3.2.1 Requisito funcional 1                     | 12        |
| 3.2.2 Requisito funcional 2                     | 12        |
| 3.2.3 Requisito funcional 3                     | 12        |
| 3.2.4 Requisito funcional n                     | 12        |
| <b>3.3 Requisitos no funcionales</b>            | <b>13</b> |
| 3.3.1 Requisitos de rendimiento                 | 13        |
| 3.3.2 Seguridad                                 | 13        |
| 3.3.3 Fiabilidad                                | 13        |
| 3.3.4 Disponibilidad                            | 13        |



---

|            |                         |           |
|------------|-------------------------|-----------|
| 3.3.5      | Mantenibilidad          | 13        |
| 3.3.6      | Portabilidad            | 13        |
| <b>3.4</b> | <b>Otros requisitos</b> | <b>14</b> |
| <b>4</b>   | <b>APÉNDICES</b>        | <b>14</b> |

# 1 Introducción

[Inserte aquí el texto]

La introducción de la Especificación de requisitos de software (SRS) debe proporcionar una vista general de la SRS. Debe incluir el objetivo, el alcance, las definiciones y acrónimos, las referencias, y la vista general del SRS.

Este documento reúne los requisitos para el desarrollo de un PLC educativo, pensado especialmente para estudiantes que están aprendiendo automatización y control. La idea surge de la necesidad de contar con una herramienta que sea accesible, práctica y fácil de usar, pero que al mismo tiempo ofrezca la experiencia real de trabajar con un controlador lógico programable.

El objetivo principal es que este PLC se convierta en un apoyo en el proceso de enseñanza, permitiendo a los estudiantes experimentar, equivocarse, corregir y aprender de una manera más dinámica. A diferencia de los equipos industriales, que suelen ser muy costosos y complicados, este proyecto busca acercar la tecnología al aula, adaptándola a un entorno académico.

## 1.1 Propósito

[Inserte aquí el texto]

- Este documento tiene como propósito definir los requisitos de un PLC educativo, creado para que los estudiantes aprendan y practiquen automatización de forma sencilla y accesible. Está dirigido a estudiantes, docentes y desarrolladores que buscan una herramienta práctica y asequible para el ámbito académico.

Esta dirigido:

- Estudiantes de ingeniería, electrónica, mecatrónica y carreras afines, que utilizarán el PLC como herramienta de aprendizaje.
- Docentes y tutores, que lo emplearán como recurso pedagógico dentro de clases y laboratorios.
- Desarrolladores e investigadores, interesados en mejorar o adaptar el diseño para futuras versiones.

## 1.2 Alcance

[Inserte aquí el texto]

- *Identificación del producto(s) a desarrollar mediante un nombre*
- *Consistencia con definiciones similares de documentos de mayor nivel (ej. Descripción del sistema) que puedan existir*

### 1.3 Personal involucrado

|                         |   |
|-------------------------|---|
| Nombre                  | Toda la clase de electiva                               |
| Rol                     | Desarrollo del diseño y la interfaz del plc             |
| Categoría profesional   | mecatrónica   |
| Responsabilidades       | Diseñar la interfaz y supervisar el desarrollo del PLC. |
| Información de contacto | 20220027tla.edu.do                                      |
| Aprobación              | Prf. Carlos Pichardo Viuque                             |

*Relación de personas involucradas en el desarrollo del sistema, con información de contacto.*

*Esta información es útil para que el gestor del proyecto pueda localizar a todos los participantes y recabar la información necesaria para la obtención de requisitos, validaciones de seguimiento, etc.*

### 1.4 Definiciones, acrónimos y abreviaturas

- PLC (Controlador Lógico Programable): Dispositivo electrónico que permite automatizar y controlar procesos mediante programación, utilizado aquí con fines educativos.
- Sensor: Componente que detecta cambios en el entorno (como luz, temperatura o posición) y envía señales al PLC.
- Actuador: Dispositivo que recibe señales del PLC para realizar acciones físicas, como encender luces o mover un motor.
- Interfaz de usuario: Medio por el cual los estudiantes interactúan con el PLC, visualizando información y controlando funciones.
- Señales digitales: Señales que solo pueden estar en dos estados (0 o 1), utilizadas para la lógica de control.
- GUI: Graphical User Interface (Interfaz gráfica de usuario).
- IO: Input/Output (Entrada/Salida de datos).
- EDU: Indica que el producto es de uso educativo.

### 1.5 Referencias

| Referencia | Título                      | Ruta  | Fecha      | Autor                |
|------------|-----------------------------|---|------------|----------------------|
| 1          | Manual del PLC educativo    | /Documentos/PLC_Educativo/Manual.pdf            | 01/09/2025 | Equipo de desarrollo |
| 2          | Guía de programación de PLC | /Documentos/PLC_Educativo/Guía_Programación.pdf | 20/07/2025 | Equipo de desarrollo |

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

## 1.6 Resumen

[[Inserte aquí el texto](#)]

- Descripción del contenido: El documento incluye el propósito y alcance del proyecto, los términos y acrónimos necesarios, referencias de soporte, personal involucrado y sus responsabilidades, así como los requisitos funcionales y no funcionales del PLC. Además, se detallan las características de la interfaz, colores y funcionalidades diseñadas especialmente para estudiantes.
- Organización del documento: La información se organiza de manera que el lector pueda comprender primero el objetivo y alcance, luego los términos clave y referencias, seguido por el personal responsable y finalmente los requisitos y funcionalidades del sistema, proporcionando una visión completa y coherente del PLC educativo y sus usos en un entorno académico.

## 2 Descripción general

### 2.1 Perspectiva del producto

[[Inserte aquí el texto](#)]

- El PLC educativo es un producto independiente, diseñado específicamente para fines académicos y de formación práctica en el área de la automatización y el control. Su desarrollo no depende de un sistema mayor, aunque puede complementarse con equipos de laboratorio como sensores, actuadores, simuladores y software de programación, lo que amplía sus posibilidades de uso en actividades educativas.
- Este producto busca ofrecer a los estudiantes un entorno seguro y sencillo para experimentar con lógica de control, programación y manejo de señales digitales y analógicas. Gracias a su interfaz intuitiva y su diseño visual con colores cálidos y suaves (como azul claro y crema), se integra de manera amigable en entornos de aprendizaje, fomentando la participación y la comprensión de los conceptos.

### 2.2 Funcionalidad del producto

[[Inserte aquí el texto](#)]

El **PLC educativo** está diseñado para ofrecer a los estudiantes una herramienta práctica y sencilla con la cual puedan comprender los fundamentos de la programación y el control de procesos automatizados. Sus funcionalidades principales se resumen en lo siguiente:

- Programación básica y avanzada: Permite a los estudiantes crear, modificar y ejecutar programas de control mediante un entorno accesible y pensado para la enseñanza.
- Manejo de entradas y salidas: Reconoce señales digitales y analógicas provenientes de sensores, y envía señales de control hacia actuadores como luces, motores o alarmas.



- Interfaz amigable: Cuenta con un sistema de uso intuitivo, con indicadores visuales claros y un diseño en colores cálidos y suaves (azul claro y crema), que favorecen la concentración y el aprendizaje.
- Simulación educativa: Posibilita la recreación de escenarios sencillos de automatización, permitiendo que los estudiantes prueben y entiendan la lógica de control sin riesgos.

## 2.3 Características de los usuarios

|                 |   |
|-----------------|---|
| Tipo de usuario | Estudiantes de nivel técnico y universitario  |
| Formación       | Conocimientos básicos en electricidad y electrónica   |
| Habilidades     | Capacidad para seguir instrucciones de prácticas de laboratorio, interpretar esquemas sencillos, realizar conexiones básicas de sensores y actuadores, y manejar entornos gráficos de programación.   |
| Actividades     | Los usuarios emplearán el PLC para desarrollar ejercicios prácticos de automatización, aprender lógica de control, probar programas en escenarios simulados y ejecutar prácticas que integren sensores y actuadores, reforzando así sus conocimientos teóricos. |

*Descripción de los usuarios del producto, incluyendo nivel educacional, experiencia y experiencia técnica.*

## 2.4 Restricciones

El diseño y desarrollo del PLC educativo presenta una serie de limitaciones que deben ser tomadas en cuenta para garantizar su correcto funcionamiento y accesibilidad en el entorno académico:

- Hardware limitado: El dispositivo contará con una cantidad reducida de entradas y salidas digitales y analógicas, suficientes para fines educativos, pero no para aplicaciones industriales de gran escala.
- Lenguaje de programación: Se utilizará un lenguaje sencillo y estandarizado para facilitar el aprendizaje; no se incluirán lenguajes avanzados que puedan dificultar el uso a estudiantes principiantes.
- Compatibilidad de software: El sistema estará diseñado para funcionar en sistemas operativos comunes de uso educativo (Windows y Linux), evitando dependencias complejas.
- Normas de seguridad: El producto deberá cumplir con estándares básicos de seguridad eléctrica y de laboratorio, evitando riesgos en el uso por estudiantes.

## 2.5 Suposiciones y dependencias

El desarrollo del PLC educativo se apoya en ciertos factores que se consideran estables, pero que, en caso de cambiar, podrían afectar los requisitos definidos:

- Disponibilidad de sistemas operativos comunes: Se asume que los entornos Windows y Linux estarán disponibles en los equipos donde se use el PLC. Si alguno de ellos dejara de ser compatible, habría que modificar la especificación.
- Acceso a hardware educativo básico: Se considera que los estudiantes tendrán acceso a sensores y actuadores simples (luces, motores pequeños, pulsadores) para realizar prácticas. La falta de estos componentes limitaría la experiencia de aprendizaje.
- Recursos de laboratorio: Se asume que el producto será utilizado en entornos con energía eléctrica estable y condiciones seguras de laboratorio. Si estas condiciones no están garantizadas, el uso del PLC podría verse restringido.

## 2.6 Evolución previsible del sistema

El PLC educativo está pensado como una primera versión funcional y accesible, pero se prevé que en el futuro pueda ampliarse y mejorar sus capacidades para adaptarse a nuevas necesidades de enseñanza. Algunas de las mejoras posibles son:

- Ampliación de entradas y salidas: Incorporar mayor número de puertos digitales y analógicos para prácticas más avanzadas.
- Conectividad inalámbrica: Añadir soporte para Wi-Fi o Bluetooth, facilitando la conexión con dispositivos móviles o entornos de simulación en la nube.
- Entorno de programación más completo: Incluir lenguajes visuales de bloques o integración con plataformas como Arduino o Python, para un aprendizaje más versátil.
- Compatibilidad multiplataforma ampliada: Extender el uso del software a sistemas operativos adicionales, incluyendo macOS y aplicaciones móviles.

## 3 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     | RF-027  |  |   |
| Nombre de requisito     | Entradas digitales configurables                  |  |   |
| Tipo                    | <input checked="" type="checkbox"/> Requisito     | <input type="checkbox"/> Restricción   |   |
| Fuente del requisito    | Necesidad educativa / prácticas de control básico |  |   |
| Prioridad del requisito | <input checked="" type="checkbox"/> Alta/Esencial | <input type="checkbox"/> Media/Deseado | <input type="checkbox"/> 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.*

### 3.1 Requisitos comunes de los interfaces

[Inserte aquí el texto]

*Descripción detallada de todas las entradas y salidas del sistema de software.*

#### 3.1.1 Interfaces de usuario

El sistema contará con una interfaz gráfica amigable e intuitiva para estudiantes y docentes, con las siguientes características:

- Pantalla principal con menú de opciones (nueva práctica, cargar proyecto, configuración).
- Editor de programas con bloques gráficos y código textual.
- Colores diferenciados para entradas (verde), salidas (naranja) y operaciones lógicas (azul).
- Panel de simulación en tiempo real que muestre el estado de entradas y salidas.
- Mensajes de error y alertas claras en caso de configuraciones inválidas o compilación fallida.

#### 3.1.2 Interfaces de hardware

El sistema deberá contar con conexiones físicas para interactuar con el entorno:

- Entradas digitales: mínimo 8, compatibles con sensores de 24VDC.
- Entradas analógicas: mínimo 2, rango 0–10V para potenciómetros o transductores.
- Salidas digitales: mínimo 8, tipo relé o transistor para control de actuadores (luces, motores).
- Puerto USB: para conexión con la PC y carga de programas.
- Pantalla LCD integrada (opcional): para mostrar estados básicos sin PC.

#### 3.1.3 Interfaces de software

El sistema deberá integrarse con software complementario para su programación y simulación:

- IDE propio (software educativo del PLC):
  - Propósito: permitir programación, simulación y carga al hardware.
  - Formato de interfaz: archivos de proyecto (\*.plcproj).
- Exportación de datos: compatibilidad para guardar programas en formato texto (\*.txt, \*.csv).
- Integración opcional: con software SCADA educativo mediante protocolo OPC-UA.

#### 3.1.4 Interfaces de comunicación

El sistema podrá comunicarse con otros dispositivos y sistemas mediante:

- USB 2.0 / 3.0: principal vía de programación.

- Protocolo Modbus RTU/TCP: para comunicación con otros PLCs o sistemas SCADA.
- Bluetooth (opcional): para conexión inalámbrica con dispositivos móviles en entornos de práctica.

### **3.2 Requisitos funcionales**

El software deberá realizar las siguientes acciones principales:

- Validación de entradas antes de su procesamiento (valores dentro del rango permitido).
- Procesamiento de la lógica definida por el usuario (ladder o bloques).
- Generación de salidas en función de entradas y condiciones programadas.
- Manejo de errores y recuperación automática en caso de fallos de comunicación o ejecución.
- Registro de eventos y resultados de simulación para retroalimentación al usuario.

#### **3.2.1 Requisito funcional 1**

- Nombre: Validación de entradas
- Descripción: El sistema verificará que las señales de entrada estén dentro del rango permitido antes de procesarlas.

#### **3.2.2 Requisito funcional 2**

- Nombre: Ejecución de lógica de control
- Descripción: El sistema deberá ejecutar la lógica programada por el usuario de acuerdo con las instrucciones definidas (contactos, bobinas, temporizadores, contadores).

#### **3.2.3 Requisito funcional 3**

- Nombre: Simulación en tiempo real
- Descripción: El software permitirá al usuario simular el comportamiento del programa sin necesidad de cargarlo en el hardware.

#### **3.2.4 Requisito funcional n**

- Nombre: Comunicación con hardware PLC
- Descripción: El sistema deberá enviar y recibir datos correctamente entre el software de programación y el hardware del PLC a través de USB.

### **3.3 Requisitos no funcionales**

#### **3.3.1 Requisitos de rendimiento**

- El sistema debe permitir la ejecución de hasta 20 usuarios simultáneos en la plataforma de simulación sin degradar el rendimiento.
- El 95% de las operaciones (compilación, carga de programa y simulación) deberán completarse en menos de 2 segundos.
- El hardware del PLC debe poder procesar programas de hasta 200 instrucciones sin pérdida de velocidad.
- El software debe poder gestionar un promedio de 50 interacciones por minuto durante las prácticas sin fallos.

#### **3.3.2 Seguridad**

- El acceso al software estará protegido por usuarios y contraseñas diferenciados para estudiantes y docentes.
- Se implementarán logs de actividad que registren acciones críticas (carga de programas, simulaciones, errores).
- Toda comunicación entre PC y PLC deberá usar un protocolo seguro (ej. cifrado básico en USB o Modbus TCP).
- El sistema verificará la integridad de archivos de programas antes de cargarlos en el hardware para evitar fallos maliciosos o accidentales.

#### **3.3.3 Fiabilidad**

- El sistema deberá tener una tasa de error menor al 2% en operaciones de compilación y carga.
- El hardware del PLC deberá mantener un tiempo medio entre fallos (MTBF) mínimo de 5,000 horas.
- El software deberá recuperarse automáticamente en caso de pérdida de conexión con el PLC en un máximo de 5 segundos.

#### **3.3.4 Disponibilidad**

- El sistema deberá estar disponible al menos el 99% del tiempo de uso en horario académico (8:00 am – 8:00 pm).
- Las actualizaciones del software no deberán dejarlo inactivo por más de 2 horas seguidas.
- El hardware deberá encender y estar listo para su uso en un máximo de 15 segundos

#### **3.3.5 Mantenibilidad**

- El software deberá permitir actualizaciones periódicas sin necesidad de reinstalar completamente el sistema.
- El equipo de desarrollo será responsable del mantenimiento correctivo (errores y fallos).

- Los docentes podrán realizar mantenimiento preventivo, como la generación de reportes semanales de uso y estadísticas de acceso.
- Las tareas de mantenimiento no deberán superar el 5% del tiempo de operación mensual.

### **3.3.6 Portabilidad**

- El software deberá funcionar en Windows 10 o superior, y ser adaptable a Linux en futuras versiones.
- Al menos el 80% del código fuente será independiente del sistema operativo, garantizando su migración a otros entornos.
- Los programas desarrollados deberán exportarse en formatos estándar (\*.txt, \*.csv) para reutilización en otras plataformas.
- La aplicación deberá poder ejecutarse en equipos con mínimo 4 GB de RAM y procesador i3 o equivalente.

### **3.4 Otros requisitos**

- Requisitos culturales y educativos: El sistema debe estar disponible en idioma español, con opción futura a inglés para facilitar su uso internacional.
- Requisitos legales: Cumplir con normativas de seguridad eléctrica (24VDC seguros para estudiantes) y licencias de software libre o académico.
- Requisitos políticos: El producto debe alinearse con programas nacionales de educación técnica y profesional.

## **4 Apéndices**

En esta sección se podrán incluir:

- Diagramas eléctricos y de conexión del PLC.
- Ejemplos de prácticas preconfiguradas.
- Glosario de símbolos usados en diagramas ladder.
- Manuales de usuario y guías rápidas para estudiantes y docentes.