|  |
| --- |
|  |

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.** |
| 17/09/2025 | 17/09/2025 | Franjelis Luciano | Carlos Pichardo (Gerente de Proyecto) |

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

|  |  |
| --- | --- |
| Por el cliente | Por la empresa suministradora |
| Instituto Tecnologico de Las Americas (ITLA) | Equipo de Desarrollo |
| Profesor Carlos Antonio Pichardo | SR. Franjelis Luciano |

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

En el ámbito de la educación tecnológica contemporánea, la formación en sistemas de automatización y control constituye un eje estratégico para preparar a los futuros profesionales que habrán de desenvolverse en industrias cada vez más digitalizadas y competitivas. Sin embargo, el acceso a los Controladores Lógicos Programables (PLC), herramienta fundamental en el mundo de la automatización industrial, continúa siendo limitado en el espacio académico, ya sea por su elevado costo, su orientación eminentemente productiva o la falta de modelos específicamente diseñados para fines pedagógicos.

El proyecto **PLC4UNI** surge como respuesta a esta necesidad, con la intención de diseñar y desarrollar un PLC orientado a la educación superior. Su propósito no se restringe únicamente a suplir la ausencia de un dispositivo didáctico en el mercado, sino que se proyecta como una iniciativa transformadora que busca integrar la teoría con la práctica, fomentar la innovación en el aula y ofrecer un recurso que acerque al estudiante al entorno industrial real sin sacrificar la accesibilidad ni la simplicidad de uso.

En este sentido, **PLC4UNI** no pretende ser una mera réplica económica de un PLC comercial, sino un dispositivo concebido con criterios pedagógicos, capaz de adaptarse a diferentes niveles de enseñanza y de promover el aprendizaje significativo. El trasfondo del proyecto radica en comprender que la educación en automatización no debe limitarse al entrenamiento técnico, sino que debe estimular la capacidad crítica, la resolución de problemas y la creatividad de los futuros ingenieros y tecnólogos.

Así, la creación de un PLC para la educación universitaria se plantea como una apuesta estratégica para cerrar la brecha entre la academia y la industria, dotando a los estudiantes de un recurso práctico que no solo refuerce sus competencias técnicas, sino que también les permita experimentar con escenarios de innovación, investigación aplicada y esarrollo de proyectos interdisciplinarios.

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.

## Propósito

Este documento, denominado Especificación de Requisitos de Software (SRS), tiene como propósito fundamental definir de manera clara, exhaustiva y estructurada los requisitos funcionales y no funcionales del sistema **PLC4UNI**. El objetivo central de este proyecto es concebir y desarrollar un Controlador LógicoProgramable (PLC) de código abierto, accesible y pedagógicamente orientado, capaz de responder a las necesidades de enseñanza y aprendizaje en el ámbito universitario de la automatización y el control.

La intención de este documento no se limita únicamente a establecer lineamientos técnicos, sino que también busca sentar las bases para una innovación educativa que trascienda la simple replicación de dispositivos industriales. Se persigue un enfoque transformador, donde el PLC se convierta en un instrumento de integración entre la teoría y la práctica, permitiendo que los estudiantes experimenten con un recurso auténtico, adaptado a su nivel de formación, y que al mismo tiempo constituya un puente hacia el mundo profesional.

Este documento está dirigido a los stakeholders clave del proyecto, entre ellos:

* El equipo de desarrollo, encargado de materializar las especificaciones técnicas.
* Los docentes e investigadores, quienes encontrarán en el PLC4UNI un recurso versátil para el diseño de prácticas, proyectos y entornos de simulación aplicados.
* Las autoridades universitarias y gestores académicos, que podrán evaluar el impacto del proyecto en la calidad de la enseñanza y en la preparación de los futuros profesionales.

En este sentido, el SRS de **PLC4UNI** no solo servirá como base contractual y guía técnica para el diseño, implementación y validación del sistema, sino también como una declaración de intenciones estratégicas que refuerza el compromiso de la universidad con la innovación, la accesibilidad tecnológica y la formación integral de sus estudiantes.

* Propósito del documento
* Audiencia a la que va dirigido

## Alcance

El proyecto PLC4UNI tiene como alcance el diseño, desarrollo e implementación de un Controlador Lógico Programable (PLC) de código abierto y bajo costo, concebido específicamente para fines educativos en el nivel universitario. A diferencia de los PLC comerciales, cuyo diseño responde a criterios de eficiencia industrial y aplicaciones de producción en masa, este sistema busca priorizar la funcionalidad didáctica, la simplicidad operativa y la adaptabilidad curricular.

El sistema abarcará las siguientes dimensiones:

* Entorno de hardware: un dispositivo compacto, robusto y modular que permita la conexión de entradas y salidas digitales y analógicas, facilitando la experimentación con sensores, actuadores y pequeños sistemas de automatización.
* Entorno de software: una plataforma de programación accesible e intuitiva, con soporte para lenguajes estándar de automatización (por ejemplo, diagrama de escalera y texto estructurado), además de interfaces gráficas que permitan la comprensión gradual de los conceptos por parte del estudiante.
* Material didáctico complementario: guías de uso, manuales y prácticas integradas que orienten el aprendizaje autónomo y colaborativo, reforzando el vínculo entre la teoría y la práctica.

En términos de cobertura, el alcance del sistema se centra en el contexto educativo y no en la sustitución de los PLC industriales en ambientes productivos. No obstante, se busca garantizar que el estudiante adquiera competencias que puedan trasladarse sin fricciones al manejo de controladores comerciales, logrando un aprendizaje significativo y transferible.

De manera estratégica, el proyecto busca:

1. Reducir la brecha entre el acceso académico y la realidad industrial en materia de automatización.
2. Proveer a las universidades de una herramienta de formación práctica sostenible, que pueda actualizarse y escalarse en función de las necesidades curriculares.
3. Fomentar la investigación aplicada y el desarrollo de proyectos interdisciplinarios, aprovechando la apertura y flexibilidad del sistema.

En síntesis, el alcance de PLC4UNI se define no solo por la entrega de un dispositivo físico y su software asociado, sino también por la creación de un ecosistema educativo integral que promueva la innovación, la equidad en el acceso tecnológico y la formación de competencias críticas para el futuro profesional de los estudiantes.

* 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

## Personal involucrado

|  |  |
| --- | --- |
| Nombre | Carlos Antonio Pichardo Viuque |
| Rol | Gerente de Proyecto |
| Categoría profesional | Profesor de Diseño para mecatronico |
| Responsabilidades | Tomar decisiones |
| Información de contacto | cpichardo@itla.edu.do |
| Aprobación |  |

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.

## Definiciones, acrónimos y abreviaturas

Con el fin de garantizar la claridad conceptual en la interpretación de este documento, se definen a continuación los términos, acrónimos y abreviaturas más relevantes en el contexto del proyecto **PLC4UNI**:

* **PLC (Controlador Lógico Programable):** Dispositivo electrónico de control utilizado en la automatización de procesos industriales, diseñado para recibir señales de entrada, procesarlas mediante un programa lógico y generar señales de salida que actúan sobre dispositivos externos.
* **PLC4UNI:** Nombre asignado al proyecto de creación de un PLC didáctico, de código abierto y bajo costo, orientado a la educación universitaria.
* **SRS (Software Requirements Specification):** Documento de Especificación de Requisitos de Software, basado en la norma IEEE Std 830-1998, cuyo propósito es definir de manera detallada y estructurada los requisitos del sistema a desarrollar.
* **Stakeholder:** Cualquier persona, grupo u organización que tiene interés en el desarrollo, implementación o resultado del sistema. En el caso de este proyecto, incluye profesores, estudiantes, investigadores, desarrolladores y autoridades universitarias.
* **Lenguaje Ladder (LD):** Lenguaje de programación gráfico estándar utilizado en PLCs, basado en esquemas que emulan diagramas eléctricos de control por contactos.
* **Texto Estructurado (ST):** Lenguaje de programación textual utilizado en PLCs, similar a lenguajes de alto nivel, que facilita la implementación de algoritmos más complejos.
* **Open Source (Código abierto):** Modelo de desarrollo en el cual el software y, en algunos casos, el hardware, se liberan bajo licencias que permiten su estudio, modificación y redistribución.
* **Automatización educativa:** Enfoque pedagógico que integra tecnologías de automatización y control en procesos de enseñanza-aprendizaje, con el fin de fortalecer competencias prácticas y teóricas en el área de la ingeniería.
* **IEEE Std 830-1998:** Estándar internacional que establece las directrices para la elaboración de documentos de especificación de requisitos de software.
* **ESP32:** Microcontrolador con Wi-Fi y Bluetooth integrados, base del hardware del sistema.
* **I/O:** Input/Output (Entradas y salidas digitales o analógicas).
* **ADC:** Analog-to-Digital Converter (convertidor analógico-digital).
* **HMI:** Human Machine Interface (Interfaz hombre-máquina).
* **Modbus RTU/TCP:** Protocolos de comunicación industrial.
* **RS-485:** Bus serial diferencial para comunicaciones industriales.
* BOM: Bill of Materials (lista de materiales)

Definición de todos los términos, abreviaturas y acrónimos necesarios para interpretar apropiadamente este documento. En ella se pueden indicar referencias a uno o más apéndices, o a otros documentos.

## Referencias

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Referencia** | **Titulo** | **Ruta** | **Fecha** | **Autor** |
|  |  |  |  |  |
| IEEE Std 830-1998. | IEEE Recommended Practice for Software Requirements Specification | Institute of Electrical and Electronics Engineers | 1998 | IEEE |
| IEC 61131-3 | *Programmable Controllers – Part 3: Programming Languages* | International Electrotechnical Commission | 2013 | IEC |
| González & Sánchez (2019) | Automatización industrial y control programable | Editorial Alfaomega, México | 2019 | González, A. & Sánchez, R. |
| Siemens Documentation | PLC Programming Basics | Siemens AG Documentation Center | 2021 | Siemens AG |
| Rockwell Technical Paper | Introduction to Programmable Logic Controllers | Rockwell Automation Technical White Paper | 2020 | Rockwell Automation |
| OpenPLC Project | Open Source PLC | <http://www.openplcproject.com> | 2025 | OpenPLC Project |
| Qualitatis Template | Plantilla de especificación de requisitos de software conforme al estándar IEEE 830 | http://www.qualitatis.org | - | Qualitatis.org |

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

El presente documento constituye la **Especificación de Requisitos de Software (SRS)** para el proyecto **PLC4UNI**. En él se establecen los fundamentos conceptuales, los objetivos, el alcance y los actores involucrados en el desarrollo del sistema, además de detallar los requisitos funcionales y no funcionales que guiarán la implementación del mismo.

El documento se encuentra organizado de la siguiente manera:

* **Capítulo 1 – Introducción:** Presenta el propósito del documento, el alcance del sistema, el personal involucrado, las definiciones y referencias clave, así como un panorama general del proyecto.
* **Capítulo 2 – Descripción general:** Ofrece una visión global del producto, incluyendo su perspectiva, funcionalidades principales, características de los usuarios, restricciones, suposiciones y posibles evoluciones del sistema.
* **Capítulo 3 – Requisitos específicos:** Detalla los requisitos funcionales y no funcionales del sistema, organizados de manera estructurada y con trazabilidad, para orientar el desarrollo técnico y las pruebas de validación.
* **Capítulo 4 – Apéndices:** Contiene información complementaria que respalda el contenido principal, como referencias técnicas adicionales, diagramas, ejemplos de casos de uso y consideraciones académicas.

De esta forma, el documento asegura una estructura lógica y progresiva, que permite a los distintos stakeholders comprender, validar y dar seguimiento al desarrollo del proyecto **PLC4UNI**, garantizando tanto la claridad técnica como la coherencia pedagógica del mismo.

* Descripción del contenido del resto del documento
* Explicación de la organización del documento

# Descripción general

## Perspectiva del producto

El sistema **PLC4UNI** se concibe como un producto **independiente** diseñado específicamente para el ámbito educativo universitario, pero con una arquitectura que facilita su **integración** en ecosistemas de aprendizaje y laboratorios más amplios. Desde su naturaleza física —un controlador compacto y modular con entradas/salidas— hasta su componente lógico —firmware abierto y un entorno de programación accesible—, PLC4UNI actúa como el núcleo práctico donde convergen la experimentación, la enseñanza y la investigación aplicada en automatización y control.

Operativamente, PLC4UNI puede desplegarse en dos modos principales: (a) **modo autónomo**, donde el dispositivo funciona conectado directamente a sensores y actuadores en una bancada de prácticas y es programado desde un ordenador local; y (b) **modo integrado**, en el que el PLC forma parte de un conjunto mayor compuesto por estaciones de trabajo, un servidor de prácticas, sistemas de adquisición y plataformas de gestión del aprendizaje (LMS). Esta doble modalidad permite que el producto cubra tanto actividades presenciales de laboratorio como escenarios de enseñanza remota o semi-presencial, facilitando prácticas dirigidas, telelaboratorios y evaluación a distancia.

En cuanto a fronteras y puntos de unión, PLC4UNI expone interfaces claramente definidas que permiten su interoperabilidad con otros elementos del entorno educativo y con herramientas industriales de referencia. Entre las interfaces previstas se incluyen:

* **Entradas/Salidas físicas**: bloques digitales y analógicos con posibilidad de expansión mediante módulos, destinados a la conexión de sensores, pulsadores, relés y actuadores de laboratorio.
* **Interfaces de comunicación**: puerto USB para programación y registro de datos; interfaz Ethernet (y, opcionalmente, Wi-Fi) para comunicación local, administración remota y acceso a prácticas en red; puerto RS-485 para protocolo Modbus RTU cuando se desee interoperar con equipos industriales de referencia.
* **Interfaz de programación**: soporte para lenguajes educativos estandarizados (por ejemplo, diagrama de escalera y texto estructurado compatibles con IEC 61131-3), a través de un IDE amigable de escritorio y/o una alternativa web.
* **Integración con software**: capacidad para exportar e importar programas y registros en formatos comunes (p. ej. JSON/CSV para telemetría, repositorios Git para versiones de proyectos), y mecanismos para interoperar con simuladores y plataformas LMS.

Es importante delimitar lo que PLC4UNI **no** pretende ser: no se concibe como un sustituto directo de PLC industriales certificados para entornos productivos críticos ni como un dispositivo para control de procesos en instalaciones con requisitos de seguridad funcional avanzados. Su diseño prioriza la seguridad didáctica, el modularidad y la facilidad de uso, por encima de características industriales como certificaciones SIL/PL o envolventes para condiciones extremas.

Por último, desde la perspectiva del ciclo de vida del producto, PLC4UNI se plantea como una plataforma **abierta y extensible**: su firmware y herramientas de programación estarán disponibles bajo licencias abiertas que permitan su adaptación, mantenimiento y ampliación por parte de la comunidad académica. Esta apertura facilita, además, la integración con herramientas de simulación y marcos de trabajo ya existentes en el ámbito educativo (por ejemplo, proyectos OpenPLC y repositorios institucionales), respetando siempre la trazabilidad y la reproducibilidad de las prácticas docentes.

Indicar si es un producto independiente o parte de un sistema mayor. En el caso de tratarse de un producto que forma parte de un sistema mayor, un diagrama que sitúe el producto dentro del sistema e identifique sus conexiones facilitan la comprensión.

## Funcionalidad del producto

El sistema **PLC4UNI** debe proporcionar un conjunto de funcionalidades básicas y complementarias que aseguren su valor pedagógico en la enseñanza universitaria de la automatización y el control. Estas funcionalidades se agrupan en cuatro ejes principales:

### a) Adquisición y control de señales

* Procesamiento de **entradas digitales y analógicas** provenientes de sensores y dispositivos didácticos.
* Generación de **salidas digitales y analógicas** para el control de actuadores de baja potencia (relés, LEDs, motores, solenoides, etc.).
* Posibilidad de **ampliación modular** mediante tarjetas de expansión de E/S.

### b) Programación y ejecución

* Disponibilidad de un **entorno de programación accesible**, con soporte para lenguajes estándar IEC 61131-3 (diagrama de escalera y texto estructurado).
* Capacidad de **cargar, ejecutar y detener programas** de usuario en tiempo real.
* Herramientas de **monitoreo y depuración**, que permitan observar el estado de las variables y señales durante la ejecución.

### c) Comunicación e integración

* Conectividad básica mediante **USB** para programación y alimentación.
* Conectividad extendida mediante **Ethernet y Wi-Fi**, orientada a prácticas en red y acceso remoto.
* Soporte para **protocolos industriales (Modbus RTU/TCP)**, con el fin de facilitar la interoperabilidad con otros equipos de referencia.
* Opciones de integración con **plataformas educativas** (LMS y simuladores).

### d) Gestión y registro de datos

* **Almacenamiento y exportación** de registros en formatos estándar (CSV, JSON) para su análisis y elaboración de informes.
* Integración con **repositorios de control de versiones** para la gestión de proyectos de programación.

En términos generales, el PLC4UNI está concebido para **recibir señales de entrada, procesarlas mediante un programa lógico definido por el usuario y generar salidas físicas o virtuales en un entorno didáctico controlado**. Con ello, se asegura que el estudiante pueda experimentar con las mismas dinámicas de la automatización industrial, en un marco adaptado a la enseñanza universitaria.

Resumen de las funcionalidades principales que el producto debe realizar, sin entrar en información de detalle.

En ocasiones la información de esta sección puede tomarse de un documento de especificación del sistema de mayor nivel (ej. Requisitos del sistema).

Las funcionalidades deben estar organizadas de manera que el cliente o cualquier interlocutor pueda entenderlo perfectamente. Para ello se pueden utilizar métodos textuales o gráficos.

## Características de los usuarios

|  |  |
| --- | --- |
| Tipo de usuario | Estudiantes de pregrado, Estudiantes de posgrado, Docentes e instructores, Investigadores y desarrolladores |
| Formación | cursan programas de ingeniería, tecnología o carreras afines (electrónica, mecatrónica, energía, automatización, informática industrial)., cursan programas de especialización o maestría en automatización, control, sistemas embebidos o áreas afines. profesionales con experiencia en ingeniería, automatización y control, con capacidad de diseñar planes de estudio y prácticas formativas., profesionales o estudiantes avanzados en áreas de innovación tecnológica, electrónica y software educativo. |
| Habilidades | conocimientos básicos en electricidad, programación y sistemas digitales; capacidad para comprender instrucciones técnicas y realizar prácticas guiadas., dominio intermedio o avanzado en programación y modelado de sistemas; experiencia en investigación aplicada., elaboración de guías pedagógicas, supervisión de proyectos, manejo de entornos de programación y didáctica de la enseñanza técnica., experiencia en programación avanzada, integración de sistemas, desarrollo de hardware y software de código abierto. |
| Actividades | desarrollo de prácticas de laboratorio, elaboración de proyectos académicos, simulación de procesos básicos de automatización y validación de programas en hardware real., diseño de proyectos interdisciplinarios, integración del PLC con otras plataformas, validación experimental de modelos y exploración de nuevas metodologías de enseñanza., diseño de prácticas de laboratorio, seguimiento de proyectos estudiantiles, evaluación de competencias y gestión de entornos de aprendizaje presenciales o remotos., ampliación de las funcionalidades del PLC4UNI, creación de módulos complementarios, pruebas de compatibilidad con estándares industriales y publicaciones académicas derivadas del uso del sistema. |

En términos generales, los usuarios del PLC4UNI presentan un nivel educativo medio a avanzado, con competencias técnicas proporcionales a su etapa formativa. El sistema ha sido diseñado con un enfoque escalable y progresivo, de modo que sea accesible para principiantes, pero también suficientemente robusto para usuarios avanzados que deseen profundizar en entornos de investigación y desarrollo.

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

## Restricciones

El desarrollo y utilización del sistema **PLC4UNI** está sujeto a un conjunto de restricciones que delimitan su diseño, implementación y aplicación. Estas restricciones responden tanto a consideraciones técnicas como pedagógicas y normativas:

### a) Restricciones técnicas

**Recursos de hardware limitados**: el PLC está orientado a prácticas académicas, por lo que su capacidad de procesamiento, memoria y número de entradas/salidas será menor que la de un PLC industrial de gama alta.

**Compatibilidad con hardware didáctico**: el diseño se centra en la interacción con sensores y actuadores de baja potencia; no se contempla el control directo de cargas industriales de gran escala sin dispositivos intermedios.

**Lenguajes de programación restringidos**: se soportarán principalmente los lenguajes de la norma IEC 61131-3 más utilizados en el ámbito educativo (Ladder Diagram y Texto Estructurado), quedando fuera lenguajes avanzados menos comunes.

### b) Restricciones de software

**Sistema operativo base**: el firmware del PLC se diseñará para plataformas abiertas y ligeras (ej. RTOS embebido o Linux reducido), limitando la portabilidad directa a sistemas cerrados o propietarios.

**Dependencia de entornos abiertos**: el software de programación y simulación estará basado en herramientas de código abierto, lo que restringe la integración inmediata con IDEs propietarios de algunos fabricantes.

### c) Restricciones pedagógicas

**Orientación académica**: el producto no debe emplearse en entornos productivos industriales donde se requiera certificación de seguridad funcional (SIL/PL) o normativas estrictas de operación.

**Nivel de complejidad limitado**: el sistema está pensado para actividades de laboratorio y proyectos universitarios, evitando sobrecargar a los usuarios con configuraciones excesivamente técnicas o procesos de instalación complejos.

### d) Restricciones normativas y de seguridad

**Cumplimiento eléctrico básico**: el diseño se ajustará a normas de bajo voltaje y seguridad eléctrica aplicables a dispositivos educativos, evitando riesgos de electrocución o daños por sobrecarga.

**Licencias de software**: al basarse en un modelo de código abierto, los componentes deberán respetar licencias libres (GPL, MIT, etc.), lo que limita la inclusión de módulos con restricciones propietarias.

En síntesis, el PLC4UNI estará condicionado por la **simplicidad, la seguridad y la accesibilidad**, lo que implica dejar fuera funcionalidades avanzadas de entornos industriales. Estas limitaciones no representan una debilidad, sino una estrategia para asegurar que el producto cumpla su objetivo principal: ser un recurso didáctico, económico y adaptable al contexto universitario.

Descripción de aquellas limitaciones a tener en cuenta a la hora de diseñar y desarrollar el sistema, tales como el empleo de determinadas metodologías de desarrollo, lenguajes de programación, normas particulares, restricciones de hardware, de sistema operativo etc.

## Suposiciones y dependencias

El desarrollo y correcta operación del sistema **PLC4UNI** se sustenta en un conjunto de suposiciones y dependencias que deben considerarse como condiciones externas al propio producto, pero que influyen directamente en su diseño, implementación y utilización:

**a) Suposiciones**

**Disponibilidad de infraestructura educativa**: se asume que las universidades cuentan con laboratorios básicos de electricidad y electrónica donde el dispositivo pueda ser instalado y utilizado.

**Acceso a ordenadores personales**: se presupone que los estudiantes y docentes dispondrán de equipos de cómputo compatibles para instalar o ejecutar el entorno de programación asociado al PLC4UNI.

**Conectividad mínima**: se asume la existencia de redes locales o acceso a internet para habilitar funciones de integración, actualizaciones de software y prácticas remotas.

**Competencias previas de los usuarios**: se considera que los estudiantes poseen conocimientos básicos en programación estructurada y fundamentos de sistemas digitales, necesarios para comprender el funcionamiento del PLC.

**b) Dependencias**

**Dependencia de hardware externo**: el uso del sistema requiere sensores, actuadores y módulos didácticos complementarios para el desarrollo de prácticas de laboratorio.

**Dependencia de software educativo**: el PLC4UNI se apoyará en entornos de programación de código abierto, lo que implica que su funcionamiento depende de la estabilidad y continuidad de dichos proyectos comunitarios.

**Dependencia de normas técnicas**: la compatibilidad con lenguajes de programación está condicionada a la vigencia de la norma **IEC 61131-3**, utilizada como referencia para los entornos de automatización.

**Dependencia de soporte institucional**: el éxito del proyecto depende del compromiso de las universidades para incorporar el dispositivo en sus planes de estudio, capacitar a los docentes y mantener actualizado el software.

En conjunto, estas suposiciones y dependencias constituyen el **marco de viabilidad del sistema**. Si alguna de ellas se ve modificada (por ejemplo, la falta de conectividad o la discontinuidad de herramientas de software abierto), será necesario ajustar los requisitos del producto para garantizar su funcionamiento en el contexto real de uso.

Descripción de aquellos factores que, si cambian, pueden afectar a los requisitos. Por ejemplo una asunción puede ser que determinado sistema operativo está disponible para el hardware requerido. De hecho, si el sistema operativo no estuviera disponible, la SRS debería modificarse.

## Evolución previsible del sistema

El sistema **PLC4UNI** ha sido concebido como una plataforma **abierta, modular y escalable**, lo cual permite prever un conjunto de mejoras y evoluciones que podrán implementarse en fases posteriores, de acuerdo con las necesidades académicas y tecnológicas emergentes. Entre las principales líneas de evolución se destacan:

1. **Ampliación de módulos de hardware:** inclusión de nuevas tarjetas de expansión que integren más entradas/salidas, soporte para señales industriales de mayor potencia o módulos especializados (p. ej., comunicación inalámbrica avanzada, control de motores trifásicos o integración con sistemas de energía renovable).
2. **Extensión de lenguajes de programación soportados:** incorporación progresiva de otros lenguajes de la norma IEC 61131-3 (como Function Block Diagram – FBD o Sequential Function Chart – SFC), para ampliar las posibilidades pedagógicas y acercar aún más a los estudiantes a entornos industriales reales.
3. **Integración con tecnologías emergentes:** desarrollo de interfaces que permitan la conexión con entornos de simulación de gemelos digitales, plataformas de IoT educativo, inteligencia artificial aplicada al control de procesos y análisis de datos en la nube.
4. **Evolución pedagógica:** creación de repositorios abiertos de prácticas, librerías de proyectos compartidos y comunidades académicas que retroalimenten el uso del PLC4UNI, promoviendo un aprendizaje colaborativo y la generación de nuevo conocimiento.
5. **Certificación académica y estandarización:** en fases avanzadas, se contempla la posibilidad de desarrollar una certificación institucional de competencias adquiridas mediante el uso del PLC4UNI, estandarizando prácticas y asegurando la calidad formativa a nivel interuniversitario.

En suma, la evolución del sistema no se limita a la mejora técnica, sino que responde a una visión de sostenibilidad e innovación educativa, donde el **PLC4UNI** se consolide como una plataforma viva, capaz de adaptarse a los cambios de la industria y a las necesidades de la educación superior en el ámbito de la automatización y el control.

Identificación de futuras mejoras al sistema, que podrán analizarse e implementarse en un futuro.

# 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

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

El sistema **PLC4UNI**, basado en un microcontrolador ESP32 y diseñado para entornos académicos universitarios, debe integrar un conjunto de interfaces robustas que permitan la interacción entre el usuario, el hardware, el software y otros sistemas externos. Estos interfaces aseguran la usabilidad pedagógica, la escalabilidad del sistema y la compatibilidad con entornos industriales y educativos.

### Interfaces de usuario

Describir los requisitos del interfaz de usuario para el producto. Esto puede estar en la forma de descripciones del texto o pantallas del interfaz. Por ejemplo posiblemente el cliente ha especificado el estilo y los colores del producto. Describa exacto cómo el producto aparecerá a su usuario previsto.

El usuario interactuará con el PLC4UNI principalmente mediante un IDE multiplataforma (Windows y Linux), compatible con los lenguajes de programación Ladder Diagram (LD) y Texto Estructurado (ST), de acuerdo con el estándar IEC 61131-3.

El entorno de programación deberá incluir:

* Editor visual para creación de programas.
* Monitoreo en tiempo real de entradas, salidas y variables internas.
* Herramientas de depuración con mensajes claros en **español e inglés**.
* Documentación y ayuda contextual integrada.

Además, se implementará una **interfaz web ligera**, accesible mediante navegador, que permita programar y supervisar el dispositivo a través de conectividad Wi-Fi, ampliando las opciones de acceso remoto y prácticas de laboratorio virtual.

### Interfaces de hardware

Especificar las características lógicas para cada interfaz entre el producto y los componentes de hardware del sistema. Se incluirán características de configuración.

El PLC4UNI deberá incluir un conjunto de interfaces de hardware adaptadas al entorno académico y compatibles con prácticas industriales básicas. En cuanto a entradas, dispondrá de **16 canales de entrada en total**, ubicados en la parte superior del módulo. De estos, **2 pines estarán dedicados a la entrada de alimentación principal de 24 VDC y 0 V**, asegurando la correcta alimentación del sistema. Además, contará con **una entrada de 4–20 mA** para sensores industriales normalizados (2 pines), **una entrada de 0–10 VDC** para prácticas de control de procesos (1 pin), **una entrada de 0–10 VAC** con terminales de fase, neutro y tierra (3 pines), **una entrada para transformador de corriente (CT)** utilizada en medición de corriente alterna (2 pines), **cuatro entradas digitales tipo NPN de 24 VDC** (4 pines) y **dos entradas PNP con optoacoplador** (2 pines), garantizando aislamiento eléctrico y compatibilidad con distintos tipos de sensores discretos. De esta forma, se completa el total de **16 entradas**, que abarcan señales digitales, analógicas, alternas y de corriente de proceso.

Respecto a las salidas, el PLC4UNI dispondrá de **16 canales de salida**, diseñados para ofrecer flexibilidad tanto en aplicaciones educativas como industriales. Entre ellas se incluye **un puerto RS-485 bidireccional** (3 pines) para comunicación y prácticas con protocolos como Modbus RTU, **tres salidas tipo PNP de 24 VDC** (3 pines) destinadas al control de actuadores de baja potencia, **un relé con contactos común, normalmente abierto y normalmente cerrado** (3 pines) y **un relé con contactos común y normalmente abierto** (2 pines), que permiten la conmutación de cargas AC/DC. Además, incorpora **una salida analógica de 4–20 mA** (2 pines) y **una salida analógica de 0–10 VDC** (1 pin) para control de variadores, actuadores o simuladores de proceso, junto con **cuatro salidas tipo NPN (24 VDC)**, de las cuales **dos incluyen optoacopladores** para aislamiento de protección (4 pines en total). Esta configuración completa las **16 salidas del sistema**, proporcionando opciones de control digital, analógico y de potencia.

El dispositivo incluirá un **puerto USB 2.0** destinado exclusivamente a la **programación, transferencia de datos y alimentación auxiliar del microcontrolador** durante pruebas o mantenimiento. Finalmente, la **alimentación principal del PLC4UNI** se realizará mediante una fuente de **24 VDC**, con protección interna contra sobrecargas, polaridad inversa y cortocircuitos, garantizando la seguridad operativa en entornos académicos y de laboratorio.

El PLC4UNI deberá incluir un conjunto de interfaces de hardware adaptados al entorno académico y compatibles con prácticas industriales básicas. En cuanto a entradas, dispondrá de un mínimo de 16 entradas en total, de las cuales 12 serán digitales transistorizadas (24 VDC, tipo NPN/PNP, con aislamiento básico), distribuidas en 8 entradas estándar de 24 VDC, 2 entradas configuradas para señales de 0–5 V (compatibles con sensores y señales de laboratorio de bajo voltaje) y 2 entradas configuradas para 0–10 V (para prácticas con variadores y procesos industriales). Las 4 entradas restantes serán analógicas, con resolución mínima de 12 bits, configurables en 2 de 0–10 V (sensores de proceso como temperatura, presión o caudal), 1 de 0–5 V (señales de laboratorio) y 1 de 4–20 mA (para sensores industriales normalizados). Respecto a las salidas, el sistema contará con un total de 12 salidas, de las cuales 11 serán a relé (AC/DC, hasta 250 VAC / 30 VDC, 5A), aptas para cargas resistivas e inductivas como lámparas, contactores, bombas pequeñas o solenoides, y 1 salida a TRIAC (AC, hasta 10A), diseñada para cargas alternas de mayor potencia (ejemplo: calefactores y lámparas). Además, se incorporarán 2 salidas analógicas (0–10 V, resolución mínima de 8 bits) destinadas al control de variadores, actuadores y prácticas de control de procesos, y hasta 8 canales PWM configurables en frecuencia y ciclo útil, orientados al manejo de motores DC, servos y otros actuadores. El sistema también integrará puertos de expansión estándar I²C, SPI y UART, lo que permitirá la conexión de periféricos académicos como pantallas, memorias, módulos HMI y sensores adicionales. Finalmente, la alimentación eléctrica podrá realizarse mediante una entrada estándar de 24 VDC, con protección interna frente a sobrecargas, polaridad inversa y cortocircuitos, garantizando la seguridad en prácticas de laboratorio.

### Interfaces de software

Indicar si hay que integrar el producto con otros productos de software.

Para cada producto de software debe especificarse lo siguiente:

* Descripción del producto software utilizado
* Propósito del interfaz
* Definición del interfaz: contiendo y formato

El software asociado al PLC4UNI deberá contemplar las siguientes características:

* Compatibilidad con el estándar **IEC 61131-3** (LD y ST).
* Integración con **entornos de simulación** y repositorios abiertos (ej. OpenPLC, Git).
* Exportación e importación de proyectos en formatos estándar (ej. **CSV, JSON**) para análisis en Excel, Matlab o Python.
* Posibilidad de interacción con **plataformas LMS** (Moodle, Blackboard) mediante servicios web o intercambio de archivos.

### Interfaces de comunicación

Describir los requisitos del interfaces de comunicación si hay comunicaciones con otros sistemas y cuales son las protocolos de comunicación.

USB 2.0: programación, transferencia de datos y alimentación del dispositivo.

Ethernet (mediante módulo externo ENC28J60 o W5500): para conexión estable en prácticas distribuidas y administración remota.

Wi-Fi integrado (802.11 b/g/n): comunicación inalámbrica con el entorno de programación y acceso remoto vía navegador.

Bluetooth/BLE: conexión con dispositivos móviles para prácticas educativas y supervisión básica.

UART (RS-232/RS-485 con transceptor externo): soporte de protocolos industriales como Modbus RTU.

I²C / SPI: conexión con periféricos académicos y módulos de expansión.

## Requisitos funcionales (RF)

Esta sección contiene los requisitos funcionales del sistema PLC4UNI. Cada requisito se presenta en formato de tabla para garantizar trazabilidad, validación y claridad técnica.

Definición de acciones fundamentales que debe realizar el software al recibir información, procesarla y producir resultados.

En ellas se incluye:

* Comprobación de validez de las entradas
* Secuencia exacta de operaciones
* Respuesta a situaciones anormales (desbordamientos, comunicaciones, recuperación de errores)
* Parámetros
* Generación de salidas
* Relaciones entre entradas y salidas (secuencias de entradas y salidas, formulas para la conversión de información)
* Especificación de los requisitos lógicos para la información que será almacenada en base de datos (tipo de información, requerido)

Los requisitos funcionales pueden ser divididos en sub-secciones.

### Requisito funcional (1-5)

## RF 1

|  |  |
| --- | --- |
| Número de requisito | RF 1 |
| Nombre de requisito | Procesamiento de señales digitales |
| Tipo | Requisito |
| Fuente del requisito | Necesidad académica – prácticas básicas de automatización |
| Prioridad del requisito | Alta/Esencial |
| Descripción | El sistema deberá ser capaz de recibir y procesar entradas digitales (mínimo 8) y generar salidas digitales (mínimo 8) que permitan la interacción con sensores, pulsadores, LEDs y relés de baja potencia. |

## RF 2

|  |  |
| --- | --- |
| Número de requisito | RF 2 |
| Nombre de requisito | Procesamiento de señales analógicas |
| Tipo | Requisito |
| Fuente del requisito | Normas IEC 61131-3 y prácticas académicas en control de procesos |
| Prioridad del requisito | Alta/Esencial |
| Descripción | El PLC4UNI deberá soportar al menos 4 entradas analógicas (0–10 V o 4–20 mA) y 2 salidas analógicas, con el fin de realizar prácticas de control de variables continuas (temperatura, presión, velocidad, etc.). |

## RF 3

|  |  |
| --- | --- |
| Número de requisito | RF 3 |
| Nombre de requisito | Soporte para lenguajes de programación IEC 61131-3 |
| Tipo | Requisito |
| Fuente del requisito | Estándar IEC 61131-3 |
| Prioridad del requisito | Alta/Esencial |
| Descripción | El sistema deberá permitir la programación en al menos dos lenguajes estándar: Ladder Diagram (LD) y Texto Estructurado (ST), a través de un entorno de desarrollo integrado (IDE) accesible. |

## RF 4

|  |  |
| --- | --- |
| Número de requisito | RF 4 |
| Nombre de requisito | Conectividad USB y Ethernet |
| Tipo | Requisito |
| Fuente del requisito | Necesidades de interoperabilidad y prácticas en red |
| Prioridad del requisito | Alta/Esencial |
| Descripción | **Descripción: El PLC4UNI deberá incluir un total de 16 entradas y 16 salidas, además de un puerto USB destinado exclusivamente a la programación, alimentación y transferencia de datos. Este diseño simplificado busca mantener la robustez funcional necesaria para prácticas académicas e industriales básicas, reduciendo la complejidad de interconexión y priorizando la compatibilidad directa con dispositivos educativos.**  **Entradas (16 en total): La sección de entradas estará ubicada en la parte superior del dispositivo y contará con las siguientes conexiones y características:**   * **Entrada de alimentación principal: terminales de 24 VDC y 0 VDC, ocupando 2 pines de entrada.** * **Entrada de corriente 4–20 mA: para sensores industriales, ocupa 2 pines.** * **Entradas digitales tipo NPN (24 VDC): 4 canales dedicados, ocupando 4 pines.** * **Entrada analógica de 0–10 VDC: para señales de control de proceso, ocupa 1 pin.** * **Entrada de voltaje AC (0–10 VAC): con terminales de fase, neutro y GND, ocupando 3 pines.** * **Entrada de transformador de corriente (CT): dedicada a medición de corriente alterna, ocupa 2 pines.** * **Entradas tipo PNP con optoacoplador: 2 canales de aislamiento óptico, ocupando 2 pines.**   **Con esta configuración se alcanzan las 16 entradas totales, distribuidas entre señales digitales, analógicas, alternas y de corriente industrial, todas protegidas mediante aislamiento básico y acondicionamiento de señal.**  **Salidas (16 en total): El bloque de salidas estará diseñado para ofrecer compatibilidad tanto con cargas DC como AC, manteniendo la flexibilidad de uso didáctico e industrial. Su distribución será la siguiente:**   * **Puerto RS-485: interfaz bidireccional (entrada/salida) para comunicación con dispositivos externos o prácticas Modbus RTU, ocupando 3 pines.** * **Salidas tipo PNP (24 VDC): 3 canales independientes, cada uno ocupando 1 pin.** * **Relé con contactos C, NA y NC: salida de control trifilar, ocupa 3 pines.** * **Relé con contactos C y NA: salida bifilar estándar, ocupa 2 pines.** * **Salida analógica de 4–20 mA: para control de actuadores industriales, ocupa 2 pines.** * **Salida analógica de 0–10 VDC: para control de variadores o simulación de procesos analógicos, ocupa 1 pin.** * **Salidas tipo NPN (24 VDC): 4 canales, de los cuales 2 estarán optoacoplados para protección adicional, ocupando 4 pines en total.**   **Esta disposición completa las 16 salidas, asegurando compatibilidad con señales de control analógicas y digitales, además de incluir elementos de conmutación electromecánica e interfaces industriales.**  **Puerto de comunicación:**   * **USB 2.0: único puerto de comunicación del sistema, utilizado para la programación, transferencia de datos y alimentación del microcontrolador durante pruebas y mantenimiento.**   **Condiciones de seguridad: Todos los puertos deberán estar claramente identificados, documentados y protegidos contra sobrecargas, inversiones de polaridad y cortocircuitos. La alimentación principal será de 24 VDC, con fusibles de protección y optoaislamiento en los canales de entrada y salida más sensibles, garantizando la seguridad del usuario y del equipo durante las prácticas de laboratorio.** |

## RF 5

|  |  |
| --- | --- |
| Número de requisito | RF 5 |
| Nombre de requisito | Interfaz de usuario para depuración |
| Tipo | Requisito |
| Fuente del requisito | Necesidad académica – seguimiento de prácticas |
| Prioridad del requisito | Media/Deseado |
| Descripción | El entorno de programación deberá incluir herramientas de monitoreo en tiempo real, que permitan observar el estado de las variables y señales durante la ejecución del programa. |

### Requisito funcional (6-10)

**RF 6**

|  |  |
| --- | --- |
| Número de requisito | RF 6 |
| Nombre de requisito | Comunicación inalámbrica opcional |
| Tipo | Requisito |
| Fuente del requisito | Necesidad de prácticas remotas y enseñanza híbrida |
| Prioridad del requisito | Media/Deseado |
| Descripción | El sistema deberá permitir la incorporación de un módulo Wi-Fi opcional para habilitar la comunicación inalámbrica con el entorno de programación y plataformas educativas, facilitando el desarrollo de prácticas en red y tele laboratorios. |

### RF 7

|  |  |
| --- | --- |
| Número de requisito | RF 7 |
| Nombre de requisito | Soporte de protocolos industriales básicos |
| Tipo | Requisito |
| Fuente del requisito | Interoperabilidad académica con equipos de referencia |
| Prioridad del requisito | Alta/Esencial |
| Descripción | El PLC4UNI deberá soportar protocolos industriales de uso común, como Modbus RTU/TCP, con el fin de integrarse en prácticas donde se empleen dispositivos de referencia (sensores, variadores, HMI). |

## RF 8

|  |  |
| --- | --- |
| Número de requisito | RF 8 |
| Nombre de requisito | Registro y exportación de datos |
| Tipo | Requisito |
| Fuente del requisito | Necesidad académica – análisis de resultados |
| Prioridad del requisito | Alta/Esencial |
| Descripción | El sistema deberá ser capaz de registrar el estado de variables y eventos de ejecución, exportando los datos en formatos estándar (CSV, JSON) para análisis posterior en herramientas externas como Excel, Matlab o Python. |

## RF 9

|  |  |
| --- | --- |
| Número de requisito | RF 9 |
| Nombre de requisito | Integración con plataformas de gestión del aprendizaje (LMS) |
| Tipo | Requisito |
| Fuente del requisito | Estrategia pedagógica universitaria |
| Prioridad del requisito | Media/Deseado |
| Descripción | El entorno de programación deberá contar con mecanismos de integración con plataformas educativas (ej. Moodle, Blackboard), permitiendo la carga y validación de prácticas de los estudiantes de manera automática. |

## RF 10

|  |  |
| --- | --- |
| Número de requisito | RF 10 |
| Nombre de requisito | Ejecución y monitoreo en tiempo real |
| Tipo | Requisito |
| Fuente del requisito | Prácticas de automatización universitaria |
| Prioridad del requisito | Alta/Esencial |
| Descripción | El sistema deberá ejecutar los programas cargados en tiempo real, asegurando tiempos de ciclo compatibles con la simulación de procesos básicos de automatización (del orden de 1–10 ms), y permitiendo al usuario monitorear variables durante la ejecución. |

### Requisito funcional (11-15)

## RF 11

|  |  |
| --- | --- |
| Número de requisito | RF 11 |
| Nombre de requisito | Mecanismos de seguridad didáctica |
| Tipo | Requisito |
| Fuente del requisito | Normativas de seguridad educativa |
| Prioridad del requisito | Alta/Esencial |
| Descripción | El sistema deberá incorporar protecciones contra sobrecargas y errores comunes de conexión, garantizando que los estudiantes puedan operar el dispositivo sin riesgo eléctrico ni daño al hardware. |

## RF 12

|  |  |
| --- | --- |
| Número de requisito | RF 12 |
| Nombre de requisito | Modularidad y escalabilidad del hardware |
| Tipo | Requisito |
| Fuente del requisito | Necesidad de flexibilidad académica |
| Prioridad del requisito | Alta/Esencial |
| Descripción | El hardware del PLC4UNI deberá estar diseñado de manera modular, permitiendo la adición de nuevas tarjetas de expansión o periféricos, con el fin de adaptarse a distintos niveles de complejidad académica. |

## RF 13

|  |  |
| --- | --- |
| Número de requisito | RF 13 |
| Nombre de requisito | Repositorio de prácticas y ejemplos |
| Tipo | Requisito |
| Fuente del requisito | Enfoque pedagógico universitario |
| Prioridad del requisito | Media/Deseado |
| Descripción | El sistema deberá contar con un repositorio de prácticas preconfiguradas y ejemplos de programas que permitan a los estudiantes iniciarse rápidamente en el uso del PLC. |

## RF 14

|  |  |
| --- | --- |
| Número de requisito | RF 14 |
| Nombre de requisito | Compatibilidad con simuladores externos |
| Tipo | Requisito |
| Fuente del requisito | Integración académica |
| Prioridad del requisito | Alta/Esencial |
| Descripción | El PLC4UNI deberá permitir la integración con simuladores educativos y de procesos industriales, de forma que los estudiantes puedan validar programas tanto en entornos virtuales como en hardware real. |

## RF 15

|  |  |
| --- | --- |
| Número de requisito | RF 15 |
| Nombre de requisito | Soporte comunitario y actualización abierta |
| Tipo | Requisito |
| Fuente del requisito | Modelo open source |
| Prioridad del requisito | Media/Deseado |
| Descripción | El proyecto deberá contar con un sistema de distribución abierto (repositorio Git) para actualizaciones de firmware, software de programación y librerías de prácticas, asegurando su evolución a largo plazo con apoyo de la comunidad académica. |

## Requisitos no funcionales (RNF)

### Requisitos de rendimiento

Especificación de los requisitos relacionados con la carga que se espera tenga que soportar el sistema. Por ejemplo, el número de terminales, el número esperado de usuarios simultáneamente conectados, número de transacciones por segundo que deberá soportar el sistema, etc.

Todos estos requisitos deben ser mesurables. Por ejemplo, indicando “el 95% de las transacciones deben realizarse en menos de 1 segundo”, en lugar de “los operadores no deben esperar a que se complete la transacción”.

### RNF 1

|  |  |
| --- | --- |
| Número de requisito | RNF 1 |
| Nombre de requisito | Rendimiento |
| Tipo | Requisito |
| Fuente del requisito | Necesidad de prácticas en tiempo real |
| Prioridad del requisito | Alta/Esencial |
| Descripción | El sistema deberá procesar ciclos de ejecución en un tiempo máximo de 10 ms para garantizar la simulación de procesos en tiempo real. |

### Seguridad

Especificación de elementos que protegerán al software de accesos, usos y sabotajes maliciosos, así como de modificaciones o destrucciones maliciosas o accidentales. Los requisitos pueden especificar:

* Empleo de técnicas criptográficas.
* Registro de ficheros con “logs” de actividad.
* Asignación de determinadas funcionalidades a determinados módulos.
* Restricciones de comunicación entre determinados módulos.
* Comprobaciones de integridad de información crítica.

### RNF 2

|  |  |
| --- | --- |
| Número de requisito | RNF 2 |
| Nombre de requisito | Seguridad |
| Tipo | Requisito |
| Fuente del requisito | Normativas educativas y de seguridad eléctrica |
| Prioridad del requisito | Alta/Esencial |
| Descripción | El PLC deberá cumplir con normativas de bajo voltaje y seguridad eléctrica para dispositivos educativos, minimizando riesgos de electrocución y daños por sobrecarga. |

### Fiabilidad

Especificación de los factores de fiabilidad necesaria del sistema. Esto se expresa generalmente como el tiempo entre los incidentes permisibles, o el total de incidentes permisible.

### RNF 3

|  |  |
| --- | --- |
| Número de requisito | RNF 3 |
| Nombre de requisito | Fiabilidad |
| Tipo | Requisito |
| Fuente del requisito | Prácticas de laboratorio |
| Prioridad del requisito | Alta/Esencial |
| Descripción | El sistema deberá garantizar una disponibilidad de al menos 95% en sesiones de laboratorio, con mecanismos de recuperación frente a fallos menores de software. |

### Disponibilidad

Especificación de los factores de disponibilidad final exigidos al sistema. Normalmente expresados en % de tiempo en los que el software tiene que mostrar disponibilidad.

### RNF 4

|  |  |
| --- | --- |
| Número de requisito | RNF 4 |
| Nombre de requisito | Disponibilidad |
| Tipo | Requisito |
| Fuente del requisito | Entornos académicos |
| Prioridad del requisito | Media/Deseado |
| Descripción | El software asociado deberá estar disponible para múltiples sistemas operativos (Windows, Linux), facilitando el acceso a estudiantes y docentes. |

### Mantenibilidad

Identificación del tipo de mantenimiento necesario del sistema.

Especificación de quien debe realizar las tareas de mantenimiento, por ejemplo usuarios, o un desarrollador.

Especificación de cuando debe realizarse las tareas de mantenimiento. Por ejemplo, generación de estadísticas de acceso semanales y mensuales.

### RNF 5

|  |  |
| --- | --- |
| Número de requisito | RNF 5 |
| Nombre de requisito | Mantenibilidad |
| Tipo | Requisito |
| Fuente del requisito | Modelo open source |
| Prioridad del requisito | Media/Deseado |
| Descripción | El sistema deberá contar con documentación clara y repositorios abiertos que faciliten la corrección de errores, la actualización de funciones y la contribución comunitaria. |

### Portabilidad

Especificación de atributos que debe presentar el software para facilitar su traslado a otras plataformas u entornos. Pueden incluirse:

* Porcentaje de componentes dependientes del servidor.
* Porcentaje de código dependiente del servidor.
* Uso de un determinado lenguaje por su portabilidad.
* Uso de un determinado compilador o plataforma de desarrollo.
* Uso de un determinado sistema operativo.

### RNF 6

|  |  |
| --- | --- |
| Número de requisito | RNF 6 |
| Nombre de requisito | Portabilidad |
| Tipo | Requisito |
| Fuente del requisito | Entornos académicos diversos |
| Prioridad del requisito | Media/Deseado |
| Descripción | El software del PLC deberá ser portable a distintas plataformas, con un porcentaje mínimo de 80% de código independiente del sistema operativo, garantizando su uso en contextos diversos. |

## Otros requisitos

Cualquier otro requisito que no encaje en ninguna de las secciones anteriores.

Por ejemplo:

Requisitos culturales y políticos

Requisitos Legales

### RNF 7

|  |  |
| --- | --- |
| Número de requisito | RNF 7 |
| Nombre de requisito | Usabilidad del entorno de programación |
| Tipo | Requisito |
| Fuente del requisito | Necesidades pedagógicas universitarias |
| Prioridad del requisito | Alta/Esencial |
| Descripción | El IDE del PLC4UNI deberá permitir que un estudiante novato pueda crear su primer programa básico en menos de 15 minutos, con interfaz intuitiva, mensajes de error claros en español e inglés, y ayuda contextual integrada. |

### RNF 8

|  |  |
| --- | --- |
| Número de requisito | RNF 8 |
| Nombre de requisito | Escalabilidad del sistema |
| Tipo | Requisito |
| Fuente del requisito | Crecimiento institucional previsto |
| Prioridad del requisito | Media/Deseado |
| Descripción | El sistema deberá soportar la conexión simultánea de al menos 30 dispositivos PLC4UNI en una red local universitaria, manteniendo tiempos de respuesta inferiores a 500ms para operaciones básicas de programación y monitoreo. |

### RNF 9

|  |  |
| --- | --- |
| Número de requisito | RNF 9 |
| Nombre de requisito | Consumo energético eficiente |
| Tipo | Requisito |
| Fuente del requisito | Sostenibilidad educativa y costos operativos |
| Prioridad del requisito | Media/Deseado |
| Descripción | El PLC4UNI deberá operar con un consumo máximo de 5W en funcionamiento normal y entrar en modo de bajo consumo (menos de 1W) automáticamente tras 15 minutos de inactividad, con capacidad de despertar mediante comunicación USB o Ethernet. |

### RNF 10

|  |  |
| --- | --- |
| Número de requisito | RNF 10 |
| Nombre de requisito | Interoperabilidad con estándares industriales |
| Tipo | Requisito |
| Fuente del requisito | Transferencia de competencias al ámbito industrial |
| Prioridad del requisito | Alta/Esencial |
| Descripción | El sistema deberá ser compatible con al menos el 80% de las funciones básicas definidas en el estándar IEC 61131-3, permitiendo que los programas desarrollados puedan ser adaptados a PLCs comerciales con modificaciones mínimas (menos del 20% del código). |

### Requisito Legal 1

|  |  |
| --- | --- |
| Número de requisito | Requisito Legal 1 |
| Nombre de requisito | Licenciamiento de Código Abierto |
| Tipo | Requisito |
| Fuente del requisito | Modelo de distribución |
| Prioridad del requisito | Alta/Esencial |
| Descripción | El proyecto PLC4UNI deberá distribuirse bajo una licencia de código abierto compatible con el ecosistema educativo (preferiblemente GPL v3 o MIT), garantizando que las instituciones educativas puedan modificar, distribuir y contribuir al desarrollo del sistema sin restricciones legales. |

### Requisito Cultural 1

|  |  |
| --- | --- |
| Número de requisito | Requisito Cultural 1 |
| Nombre de requisito | Localización Educativa |
| Tipo | Requisito |
| Fuente del requisito | Entornos académicos internacionales |
| Prioridad del requisito | Alta/Esencial |
| Descripción | El sistema deberá soportar localización completa en español e inglés, incluyendo interfaz de usuario, mensajes de sistema, documentación técnica y materiales didácticos, facilitando su adopción en universidades hispanohablantes y su integración en programas de intercambio internacional. |

# Apéndices

Pueden contener todo tipo de información relevante para la SRS pero que, propiamente, no forme parte de la SRS.

**Apéndice A — Repositorio oficial del proyecto**

Con el objetivo de garantizar la transparencia, la trazabilidad y la colaboración académica, el proyecto PLC4UNI dispone de un repositorio oficial en línea. En este espacio se almacenan los diagramas electrónicos, el firmware, la documentación técnica y las versiones sucesivas del desarrollo. De este modo, cualquier miembro de la comunidad educativa puede auditar los avances, proponer mejoras y contribuir activamente.

* Enlace al repositorio: mira el repositorio: <https://github.com/FranjelisLuciano/Franjelis_Luciano_PLC4UNI>

**Apéndice B — Canales de difusión y redes sociales**

Para fortalecer la **visibilidad y el impacto académico del proyecto**, se han establecido distintos canales de difusión, orientados tanto a la comunidad universitaria como a espacios más amplios de intercambio:

* Página oficial de la asignatura en la plataforma de la universidad, donde se publicarán los avances formales y entregas.
* Grupo de WhatsApp del curso de Mecatrónica, con fines de **coordinación interna** y resolución rápida de incidencias.
* Publicaciones en redes sociales institucionales (Facebook, Instagram, LinkedIn académico), buscando **difundir logros y atraer colaboraciones externas**.

**Apéndice C — Cronograma resumido**

El desarrollo del PLC4UNI sigue un cronograma progresivo, que articula tanto la planificación técnica como la pedagógica. A continuación, se muestra un esquema de referencia:

* **Semana 1:** Presentación del proyecto y establecimiento del objetivo general.
* **Semana 2:** Identificación de requisitos y entrega del documento SRS inicial.
* **Semana 3:** Selección de los componentes base y diseño preliminar de la arquitectura.
* **Semana 4:** Elaboración de los primeros esquemáticos y retroalimentación con el docente.
* **Semana 5:** Revisión de los esquemáticos y preparación del diseño de la PCB.
* **Semana 6:** Finalización de la PCB y validación previa a fabricación.
* **Semana 7:** Adquisición de componentes y gestión de la PCB.
* **Semana 8:** Montaje inicial del prototipo en laboratorio.
* **Semana 9:** Pruebas de hardware: validación de entradas, salidas y fuentes de alimentación.
* **Semana 10:** Desarrollo de firmware básico (gestión de E/S digitales y analógicas).
* **Semana 11:** Integración de firmware avanzado (comunicación RS-485, Wi-Fi).
* **Semana 12:** Validación de requisitos funcionales y no funcionales.
* **Semana 13:** Corrección de fallos, refinamiento de hardware/firmware y creación de manuales.
* **Semana 14:** Presentación final, entrega de documentación y defensa académica del proyecto.

**Apéndice D — Contacto del equipo**

Para fines de **coordinación académica y soporte institucional**, los responsables del proyecto PLC4UNI son:

* **Director del proyecto:** Profesor Carlos Antonio Pichardo Viuque — correo institucional: cpichardo@itla.edu.do
* **Integrante del equipo de desarrollo:** Franjelis Luciano — correo: 20231509@itla.edu.do