|  |
| --- |
|  |

Especificación de requisitos de software

Proyecto: PLC 4 uni

Revisión 1.0

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

Ficha del documento

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Revisión** | **Autor** | **Verificado dep. calidad.** |
|  |  |  |  |

Documento validado por las partes en fecha:

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

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 proyecto **PLC 4 Uni** busca desarrollar un controlador lógico programable (PLC) para uso educativo. Su objetivo es que estudiantes y profesores de la institución y de cualquier otro centro educativo tengan una herramienta **práctica y accesible** para la enseñanza de **automatización, control y programación** de PLCs.

Este documento establece los **requisitos de hardware y software** para el desarrollo, sirviendo como guía formal para las etapas de diseño, desarrollo y validación. Así, se garantiza que el producto final cumpla con los **estándares de calidad** y las expectativas de los usuarios, modernizando la enseñanza de la automatización industrial al complementar la teoría con una herramienta tangible.

## Propósito

## El objetivo principal es ofrecer a los estudiantes un dispositivo accesible, práctico y adaptado a las necesidades académicas, que permita realizar prácticas de programación, simulación y control de procesos en un entorno real, fomentando así el desarrollo de competencias técnicas y fortaleciendo la formación en mecatrónica y áreas afines.

## Alcance

El proyecto contempla el diseño de un PLC educativo con entradas y salidas digitales, capacidad de expansión, conexión con PC mediante USB, Bluetton y wi-fi y un entorno de programación amigable. El PLC se utilizará exclusivamente con fines didácticos en laboratorios de mecatrónica, automatización y áreas a fines.

## Personal involucrado

|  |  |
| --- | --- |
| Nombre | Pichardo |
| Rol | Dirigente de proyecto |
| Categoría profesional | Maestro |
| Responsabilidades | -Coordinar y supervisar el desarrollo del proyecto.  -Gestionar tiempos, recursos y actividades del equipo.  -Validar los avances y cumplimiento de requisitos.  -Ser el enlace entre el equipo y la institución. |
| Información de contacto | cpichardo@itla.edu.do |
| Aprobación | Aprobado |

|  |  |
| --- | --- |
| Nombre | Lowenkis García |
| Rol | Desarrollador |
| Categoría profesional | Estudiante |
| Responsabilidades | -Diseñar y programar el funcionamiento del PLC educativo.  -Implementar las entradas/salidas digitales y posibles expansiones.  -Integrar el hardware con el software de programación.  -Documentar el proceso de desarrollo y realizar pruebas de funcionamiento. |
| Información de contacto | 20230845@itla.edu.do |
| Aprobación | Aprobado |

|  |  |
| --- | --- |
| Nombre | Santo Mateo |
| Rol | Usuario |
| Categoría profesional | Docente |
| Responsabilidades | -Utilizar el PLC en las prácticas educativas.  -Ejecutar programas de prueba y simulaciones.  -Reportar fallos, limitaciones o mejoras observadas.  -Seguir las instrucciones y protocolos de seguridad establecidos. |
| Información de contacto | smateo@itla.edu.do |
| Aprobación | Aprobado |

## Definiciones, acrónimos y abreviaturas

 PLC**:** Controlador Lógico Programable.

 **SRS:** Especificación de Requisitos de Software.

 **GPIO:** Pines de Entrada/Salida de Propósito General.

 **ESP32:** Microcontrolador de sistema en chip (SoC) de bajo consumo con conectividad Wi-Fi y Bluetooth.

## Referencias

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Referencia** | **Titulo** | **Ruta** | **Fecha** | **Autor** |
|  |  |  |  |  |
|  |  |  |  |  |

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 describe los requisitos funcionales y no funcionales del proyecto PLC 4 Uni, un controlador lógico programable con fines educativos. A lo largo del documento se detallan el propósito del proyecto, su alcance, las personas involucradas, así como las características técnicas y pedagógicas que lo convierten en una herramienta accesible y práctica para estudiantes y docentes. La organización del documento sigue la estructura recomendada por la norma IEEE 830, comenzando con una introducción general del sistema, continuando con una descripción detallada de sus funcionalidades principales, las restricciones y dependencias, y finalizando con los requisitos específicos que guiarán su diseño y desarrollo.

# Descripción general

## Perspectiva del producto

* El "PLC 4 uni" se concibe como un producto independiente y autónomo, diseñado para operar en un entorno educativo. La arquitectura del sistema, que integra hardware de bajo costo y software desarrollado a medida, lo distingue de las soluciones comerciales, ya que su enfoque principal es la transparencia y la facilidad de aprendizaje. El sistema se integrará en el entorno de laboratorio del instituto, permitiendo la interacción con los equipos de los estudiantes a través de interfaces de comunicación estándar.

## Funcionalidad del producto

Las principales funcionalidades del sistema incluyen:

* Capacidad de Programación: Permitirá la programación de lógica de control en lenguaje C, proporcionando a los usuarios un control granular sobre las operaciones del PLC.
* Gestión de Entradas/Salidas: Facilitará la adquisición de datos de los sensores de entrada y la activación de los actuadores de salida.
* Monitoreo del Sistema: Proporcionará una visualización del estado del PLC y de la ejecución del programa para facilitar el diagnóstico y la depuración.
* Expansión: Contara con módulo de expansión para poder conectar mas entradas y/o salidas según sean requeridas

## Características de los usuarios

|  |  |
| --- | --- |
| Tipo de usuario | Docentes / estudiantes |
| Formación | Maestros: docentes con conocimiento previos en áreas afines para una óptima enseñanza.  Estudiantes: que estén cursado carreras técnicas o carreras de ingeniería que tengas relación al producto. |
| Habilidades | Conocimientos básicos de electricidad y circuitos. Capacidad para conectar cables a terminales de tornillo de forma segura. Habilidad para usar software en un PC para el entorno de programación. |
| Actividades | **Estudiante:** Su rol principal es conectar los componentes externos (sensores y actuadores) al PLC, cargar los programas desarrollados, realizar prácticas de laboratorio y verificar la funcionalidad de la lógica de control. **Profesor:** Sus tareas incluyen diseñar y proponer ejercicios prácticos, demostrar el funcionamiento del PLC en tiempo real y evaluar el desempeño de los estudiantes en sus proyectos. |

## Restricciones

* Compatibilidad con sistemas operativos comunes (Windows/Linux).
* **Componentes de Hardware:** El diseño del sistema se basará en componentes accesibles y de bajo costo como el **ESP32** y los relés.
* **Limitaciones Presupuestarias:** El desarrollo está condicionado a un presupuesto limitado, lo que exige una optimización de los recursos y la selección de tecnologías costo-efectivas.
* Alimentacio: se limita a un voltaje de solo 24 voltios

## Suposiciones y dependencias

El desarrollo del proyecto "PLC 4 uni" se basa en las siguientes suposiciones clave, cuya validez es fundamental para el éxito del proyecto:

* Disponibilidad del Hardware: Se asume que los componentes de hardware seleccionados, como el ESP32 y los relés, estarán disponibles y accesibles a lo largo del ciclo de vida del proyecto.
* Conocimientos Previos del Usuario: Se presupone que los usuarios finales (estudiantes y maestros) poseen los conocimientos básicos de electricidad y programación en lenguaje C necesarios para operar y programar el sistema. Si esta suposición no se cumple, el tiempo y los recursos dedicados a la capacitación podrían aumentar significativamente.
* Estabilidad del Software de Apoyo: Se depende de que el entorno de desarrollo y las herramientas de compilación para el ESP32 mantengan su estabilidad y compatibilidad con el sistema operativo de las computadoras de los usuarios.

## Evolución previsible del sistema

El "PLC 4 uni" ha sido concebido como una plataforma modular y escalable, con un potencial de crecimiento significativo. Las futuras mejoras y evoluciones previstas para el sistema incluyen:

* Ampliación de Módulos de Hardware: Se prevé la integración de módulos adicionales como entradas y salidas analógicas de alta precisión, así como interfaces de comunicación industrial (por ejemplo, Modbus o CAN).
* Interfaz de Usuario Gráfica (GUI): A futuro, se podría desarrollar una GUI de escritorio o web para simplificar la programación y el monitoreo para usuarios menos experimentados.
* Integración con Sensores Especializados: La plataforma podría evolucionar para interactuar con sensores más complejos, como los de temperatura o presión, expandiendo su aplicabilidad en prácticas de laboratorio más avanzadas.

# Requisitos específicos

## Requisitos comunes de los interfaces

El sistema de software del "PLC 4 uni" debe gestionar de manera integral la interacción con el entorno externo, ya sea a través de interfaces de usuario, hardware, otros programas o redes de comunicación. A continuación se detallan los requisitos específicos para cada tipo de interfaz.

### Interfaces de usuario

Aunque el sistema no contará con una pantalla integrada, el software debe ser diseñado para interactuar con el usuario a través de un entorno de desarrollo en una PC. Los requisitos para la interfaz de usuario son los siguientes:

* El software deberá proporcionar un medio claro y conciso para que los usuarios puedan cargar sus programas de control en el microcontrolador.
* Se debe incluir una retroalimentación visual para indicar el progreso de la carga y el estado de la conexión (ej. "Conectado", "Cargando", "Error de conexión")..

### Interfaces de hardware

El software debe ser capaz de gestionar las interacciones lógicas con los componentes de hardware que se conecten al ESP32. Esto incluye:

* Configuración de los pines GPIO para que funcionen como entradas o salidas digitales.
* Lectura del estado lógico (alto/bajo) de los pines de entrada.
* Envío de señales lógicas (alto/bajo) a los pines de salida para controlar los relés.
* Gestión de los pines para entradas analógicas (en caso de que se implementen en futuras fases), incluyendo la lectura de valores de voltaje..

### Interfaces de software

El software debe ser capaz de gestionar las interacciones lógicas con los componentes de hardware que se conecten al ESP32. Esto incluye:

* Configuración de los pines GPIO para que funcionen como entradas o salidas digitales.
* Lectura del estado lógico (alto/bajo) de los pines de entrada.
* Envío de señales lógicas (alto/bajo) a los pines de salida para controlar los relés.
* Gestión de los pines para entradas analógicas (en caso de que se implementen en futuras fases), incluyendo la lectura de valores de voltaje.

### Interfaces de comunicación

El software debe gestionar las interfaces de comunicación que permiten la conexión del PLC con el exterior:

* + Interfaz USB: El sistema debe establecer una conexión fiable a través del puerto USB para permitir la carga del firmware y la comunicación serie con la PC del usuario para fines de depuración y monitoreo.
  + Interfaz Wi-Fi: El software debe ser capaz de configurar el módulo Wi-Fi del ESP32 como un punto de acceso o un cliente de red, permitiendo la transmisión de datos de telemetría (estado de entradas/salidas) a un cliente externo para monitoreo remoto.

## Requisitos funcionales

Los requisitos funcionales detallan las acciones fundamentales que el software debe realizar para procesar información y controlar el hardware, reflejando su rol como cerebro del sistema.

**Requisito Funcional 1 (RF-1): Carga de Programa**

* **Descripción:** El sistema debe permitir al usuario cargar un programa de control compilado desde un PC a través de la interfaz USB. Una vez cargado, el firmware debe ser capaz de almacenar este programa en su memoria no volátil, asegurando que no se pierda si se interrumpe la alimentación.

**equisito Funcional 2 (RF-2): Lectura de Entradas**

* **Descripción:** El software debe escanear y detectar de forma continua el estado de todas las entradas físicas (digitales y analógicas) conectadas al microcontrolador. Esta lectura debe ser precisa y en tiempo real para que la lógica de control pueda reaccionar a los cambios externos de manera inmediata.

**Requisito Funcional 3 (RF-3): Control de Salidas**

* **Descripción:** El sistema debe ejecutar la lógica de control del programa del usuario y, basándose en el estado de las entradas, activar o desactivar las salidas físicas (relés). Esta acción debe ser directa y confiable, sin demoras perceptibles.

**equisito Funcional 4 (RF-4): Ejecución del Firmware**

* **Descripción:** El **firmware** del PLC es el software base que proporciona las instrucciones de bajo nivel para que el hardware funcione y ejecute el programa del usuario. Debe gestionar un ciclo de escaneo continuo que incluye la lectura de entradas, la ejecución del programa y la actualización de las salidas. Este ciclo debe ser predecible y repetitivo para garantizar un control fiable.

**Requisito Funcional 5 (RF-5): Regulación de Voltaje**

* **Descripción:** El sistema debe incluir la funcionalidad de regulación de voltaje para garantizar que los componentes internos y externos reciban una alimentación estable. Esto es crucial para prevenir fallos o daños por fluctuaciones eléctricas.

**Requisito Funcional 6 (RF-6): Diagnóstico y Depuración**

* **Descripción:** El software debe proporcionar una interfaz de depuración que permita al usuario monitorear en tiempo real el estado de las variables internas, las entradas y las salidas a través de la conexión USB o Wi-Fi. Esto facilita la resolución de problemas en el programa del usuario.

## Requisitos no funcionales

### Requisitos de rendimiento

**Rendimiento:** La latencia en la ejecución de la lógica de control no deberá exceder los límites perceptibles para el usuario, garantizando una operación fluida. El software deberá ser eficiente en el uso de los recursos del microcontrolador.

### Seguridad

Seguridad: El sistema debe ser robusto y resiliente a fallas de software que puedan comprometer la integridad del hardware. No se requieren protocolos de seguridad de nivel industrial, dado el contexto educativo del proyecto..

### Fiabilidad

**Fiabilidad:** El "PLC 4 uni" deberá operar de manera estable y continua sin fallas inesperadas. En caso de error, el sistema deberá recuperarse sin causar daños al hardware o a los datos del usuario.

### Disponibilidad

**Disposición:** El software debe ser altamente disponible, permitiendo su uso continuo por parte de los estudiantes y maestros. El tiempo de inactividad debe ser mínimo, limitado solo a las tareas de mantenimiento o a las actualizaciones necesarias del firmware.

### Mantenibilidad

**Mantenibilidad:** El código fuente del software debe ser legible, bien documentado y modular, facilitando futuras modificaciones, depuraciones y mejoras por parte de los desarrolladores.

### Portabilidad

**Portabilidad:** El software complementario (drivers y utilidades) deberá ser compatible con los sistemas operativos de escritorio más comunes para maximizar la accesibilidad para los estudiantes.

## Otros requisitos

Esta sección abarca cualquier requisito que no encaje en las categorías anteriores (funcionales y no funcionales), pero que es crítico para el éxito del proyecto.

* **Requisitos de documentación:** Se debe generar una documentación técnica completa que incluya un manual del usuario, una guía de instalación del software complementario y una descripción detallada del código fuente del firmware.
* **Requisitos legales y normativos:** Aunque el proyecto es de naturaleza educativa, debe cumplir con las normativas locales de seguridad eléctrica para dispositivos electrónicos de bajo voltaje.
* **Requisitos de soporte:** Se debe definir un plan de soporte para los usuarios (estudiantes y maestros), que podría incluir la creación de un repositorio en línea para la comunidad del proyecto (por ejemplo, un foro o un espacio de preguntas frecuentes) donde se puedan resolver dudas técnicas.
* **Requisitos culturales y de idioma:** La interfaz y la documentación deben estar disponibles en español para asegurar la máxima accesibilidad y comprensión por parte de los usuarios locales.

# Apéndices

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