|  |
| --- |
|  |

Especificación de requisitos de software

Proyecto: PLC4uni

Revisión 1



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

**Instrucciones para el uso de este formato**

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

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

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

Notas:

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

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

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

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

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

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

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

.

Ficha del documento

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

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

|  |  |
| --- | --- |
| Por el cliente | Por la empresa suministradora |
| Instituto Tecnológico de Las Américas-ITLA | Equipo de Desarrollo |
| Prof. Carlos Pichardo | Est. Henry Abdias Bautista Portes |

Contenido

[Ficha del documento 3](#_Toc33411057)

[Contenido 4](#_Toc33411058)

[1 Introducción 6](#_Toc33411059)

[1.1 Propósito 6](#_Toc33411060)

[1.2 Alcance 6](#_Toc33411061)

[1.3 Personal involucrado 6](#_Toc33411062)

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

[1.5 Referencias 6](#_Toc33411064)

[1.6 Resumen 6](#_Toc33411065)

[2 Descripción general 7](#_Toc33411066)

[2.1 Perspectiva del producto 7](#_Toc33411067)

[2.2 Funcionalidad del producto 7](#_Toc33411068)

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

[2.4 Restricciones 7](#_Toc33411070)

[2.5 Suposiciones y dependencias 7](#_Toc33411071)

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

[3 Requisitos específicos 7](#_Toc33411073)

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

[3.1.1 Interfaces de usuario 8](#_Toc33411075)

[3.1.2 Interfaces de hardware 8](#_Toc33411076)

[3.1.3 Interfaces de software 8](#_Toc33411077)

[3.1.4 Interfaces de comunicación 8](#_Toc33411078)

[3.2 Requisitos funcionales 8](#_Toc33411079)

[3.2.1 Requisito funcional 1 9](#_Toc33411080)

[3.2.2 Requisito funcional 2 9](#_Toc33411081)

[3.2.3 Requisito funcional 3 9](#_Toc33411082)

[3.2.4 Requisito funcional n 9](#_Toc33411083)

[3.3 Requisitos no funcionales 9](#_Toc33411084)

[3.3.1 Requisitos de rendimiento 9](#_Toc33411085)

[3.3.2 Seguridad 9](#_Toc33411086)

[3.3.3 Fiabilidad 9](#_Toc33411087)

[3.3.4 Disponibilidad 9](#_Toc33411088)

[3.3.5 Mantenibilidad 10](#_Toc33411089)

[3.3.6 Portabilidad 10](#_Toc33411090)

[3.4 Otros requisitos 10](#_Toc33411091)

[4 Apéndices 10](#_Toc33411092)

# Introducción

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

## Propósito

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

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

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

## Alcance

## *El proyecto contempla el diseño, implementación y validación de una placa de control lógico programable de uso académico. El PLC contará con entradas y salidas digitales, entradas analógicas, comunicación estándar mediante Modbus RTU (RS-485) y soporte de red mediante Wi-Fi (Modbus TCP básico), además de una interfaz local simple (pantalla y botones).*

## *El alcance incluye la entrega de hardware funcional, firmware básico, documentación técnica (manual de usuario, lista de materiales, guías de prácticas), y pruebas de verificación.*

## Personal involucrado

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

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

## Definiciones, acrónimos y abreviaturas

## *PLC: Programmable Logic Controller (Controlador Lógico Programable).*

## *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)*

## Referencias

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Referencia** | **Titulo** | **Ruta** | **Fecha** | **Autor** |
| 1 | Repositorio | https://github.com/HABP0705/Henry\_Bautista\_PLC4uni.git | 18/9/2025 | Henry Abdias Bautista Portes |
|  |  |  |  |  |

## 

## Resumen

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

# Descripción general

## Perspectiva del producto

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

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

## Funcionalidad del producto

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

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

## 

## Características de los usuarios

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

## 

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

## Restricciones

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

## Suposiciones y dependencias

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

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

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

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

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

## *.*Evolución previsible del sistema

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

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

# Requisitos específicos

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

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

## Requisitos comunes de los interfaces

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

### 

### Interfaces de usuario

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

### 

### Interfaces de hardware

* **Entradas digitales (8):** Optoaisladas, compatibles con 0–24 VDC, con filtrado contra rebotes.
* **Salidas digitales (8):** Tipo relé o MOSFET, para manejar cargas de baja potencia (hasta 5A DC).
* **Entradas analógicas (4):** Rango 0–10 VDC o 4–20 mA, resolución mínima de 12 bits.
* **Alimentación:** Entrada de 24 VDC con protecciones (fusibles y diodos). Reguladores internos de 5V/3.3V.
* **Conexiones:** Borneras desmontables con identificación clara.

### 

### Interfaces de software

* Firmware base en C/C++ usando ESP-IDF o Arduino Core.
* Librerías estándar para manejo de E/S, comunicación serial, RS-485 y Wi-Fi.
* API interna para pruebas: funciones read\_input(), write\_output(), read\_analog().
* Interfaz de configuración mediante aplicación en PC (serial/USB) o interfaz web sencilla en Wi-Fi.

### Interfaces de comunicación

* **USB-UART:** Para programación y diagnóstico.
* **RS-485 (Modbus RTU):** Comunicación industrial estándar. Velocidades configurables (9600 a 115200 bps).
* **Wi-Fi (Modbus TCP o HTTP básico):** Permite conexión con PC o móvil para pruebas y monitoreo.
* **Protocolo interno:** Comunicación simple con comandos tipo texto para configuración vía consola serial.

## Requisitos funcionales

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

**3.2.1 Requisito funcional 1 – Arranque y autodiagnóstico (RF-1)**

* **Tipo:** Requisito
* **Fuente:** Profesor
* **Prioridad:** Alta
* **Descripción:** Al encenderse, el PLC debe ejecutar un autodiagnóstico que verifique memoria, estado de entradas, salidas y comunicación. El resultado debe mostrarse en LEDs y en la pantalla. La duración máxima del test será de 5 segundos.

**3.2.2 Requisito funcional 2 – Lectura de entradas digitales (RF-2)**

* **Tipo:** Requisito
* **Fuente:** Equipo de desarrollo
* **Prioridad:** Alta
* **Descripción:** El sistema debe leer las ocho entradas digitales y mostrar su estado tanto en la pantalla como a través de Modbus. La latencia máxima de actualización será de 50 milisegundos.

**3.2.3 Requisito funcional 3 – Control de salidas digitales (RF-3)**

* **Tipo:** Requisito
* **Fuente:** Profesor
* **Prioridad:** Alta
* Descripción: El PLC debe permitir activar o desactivar cada salida digital desde la interfaz local o mediante comandos Modbus. El tiempo máximo de reacción será de 50 milisegundos.

**3.2.4 Requisito funcional 4 – Lectura de entradas analógicas (RF-4)**

* **Tipo:** Requisito
* **Fuente:** Equipo de desarrollo
* **Prioridad:** Media
* **Descripción:** El PLC debe muestrear las cuatro entradas analógicas con una frecuencia mínima de 5 Hz y una resolución de 12 bits, entregando los valores en unidades de voltaje o corriente.

**3.2.5 Requisito funcional 5 – Comunicación RS-485 Modbus RTU (RF-5)**

* **Tipo:** Requisito
* **Fuente:** Profesor
* **Prioridad:** Alta
* **Descripción:** El PLC debe funcionar como esclavo Modbus RTU, respondiendo a comandos de lectura de entradas, escritura de salidas y consulta de registros.

**3.2.6 Requisito funcional 6 – Comunicación Wi-Fi (RF-6)**

* **Tipo:** Requisito
* **Fuente:** Equipo de desarrollo
* **Prioridad:** Media
* **Descripción:** El sistema debe habilitar la conexión Wi-Fi para monitoreo básico a través de Modbus TCP o interfaz web.

**3.2.7 Requisito funcional 7 – Señal de emergencia (RF-7)**

* **Tipo:** Restricción
* **Fuente:** Profesor
* **Prioridad:** Alta
* **Descripción:** El PLC debe disponer de una entrada de paro de emergencia (E-STOP). Al activarse, todas las salidas deben apagarse inmediatamente y permanecer así hasta que se reinicie el sistema de forma manual.

**3.2.8 Requisito funcional 8 – Actualización de firmware (RF-8)**

* **Tipo:** Requisito
* **Fuente:** Equipo de desarrollo
* **Prioridad:** Media
* **Descripción:** El PLC debe aceptar actualizaciones de firmware mediante USB o Wi-Fi, verificando la integridad del archivo mediante checksum antes de aplicarlo.

**3.2.9 Requisito funcional 9 – Registro de eventos críticos (RF-9)**

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

## 

## Requisitos no funcionales

### Requisitos de rendimiento

* Ciclo completo de escaneo (leer entradas, ejecutar lógica simple y actualizar salidas) ≤ 100 ms.
* Respuesta a petición Modbus RTU ≤ 100 ms en condiciones normales.

### 

### Seguridad

* Protección por contraseña para el acceso a configuración avanzada vía PC o Wi-Fi.
* Verificación de checksum antes de aceptar una actualización de firmware.
* Aislamiento galvánico entre señales de control y potencia.

### Fiabilidad

* MTTF mínimo: 10,000 horas en condiciones normales de laboratorio.
* Uso de watchdog interno del ESP32 para reinicio automático en caso de bloqueo.

### Disponibilidad

* El PLC debe estar disponible ≥ 99% del tiempo en horario de prácticas.
* El reinicio por fallo no debe superar los 5 segundos.

### Mantenibilidad

* El firmware debe estar documentado y comentado.
* Se debe disponer de manual de usuario y manual técnico.
* Puerto de depuración accesible (UART/JTAG) para recuperación.

### 

### Portabilidad

* El código debe estar escrito en C/C++ estándar para facilitar portarlo a otras placas ESP32 o STM32 en el futuro.
* Evitar dependencias fuertes de librerías propietarias.

## Otros requisitos

* **Requisitos legales:** Cumplir con normas básicas de seguridad eléctrica y compatibilidad electromagnética para uso en laboratorios educativos.
* **Requisitos académicos:** El diseño debe ser documentado de manera que sirva como material de aprendizaje (diagramas, explicaciones, guías).
* **Requisitos culturales:** El producto debe estar documentado en español, con posibilidad de versión en inglés si la universidad lo requiere.

# Apéndices

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

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

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

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

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

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

**Apéndice C — Cronograma resumido:**

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

**Apéndice D — Contacto del equipo:**

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

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