|  |
| --- |
|  |

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 semestre).*

## 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 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.

### 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.

### 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.

### 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

### Interfaces de comunicación

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

## Requisitos funcionales

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)

Las requisitos funcionales pueden ser divididos en sub-secciones.

### Requisito funcional 1

### Requisito funcional 2

### Requisito funcional 3

### Requisito funcional n

## Requisitos no funcionales

### 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”.

### 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.

### 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.

### 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.

### 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.

### 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.

## Otros requisitos

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

Por ejemplo:

Requisitos culturales y políticos

Requisitos Legales

# Apéndices

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