
Especificación de requisitos de software

Proyecto: PrecoLogic
Revisión 1.0

Logo

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

Ficha del documento

Fecha	Revisión	Autor	Verificado dep. calidad.
NA	1.0	Wilfer Javier Ventura	Carlos Antonio Pichardo Viuque

Documento validado por las partes en fecha: N/A

Por el cliente	Por la empresa suministradora
Fdo. D./ Dña NA	Fdo. D./Dña Carlos Antonio Pichardo Viuque

Contenido

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

3.3.6 Portabilidad

¡Error! Marcador no definido.

3.4 Otros requisitos

¡Error! Marcador no definido.

4 APÉNDICES

10

1 Introducción

1.1 Propósito

Propósito del documento:

Este documento tiene como propósito definir de manera estructurada y detallada los requisitos funcionales y no funcionales del sistema denominado **PLC/Relé Programable basado en ESP32**. La SRS servirá como guía técnica para el diseño, desarrollo, validación y mantenimiento del sistema, asegurando que todas las funcionalidades requeridas sean correctamente implementadas.

Audiencia a la que va dirigido:

Este documento está dirigido a desarrolladores de software y hardware embebido, ingenieros electrónicos, técnicos de automatización, docentes y evaluadores del ITLA, así como a cualquier persona involucrada en el desarrollo, prueba e implementación del prototipo.

1.2 Alcance

El sistema a desarrollar consiste en un **PLC (Controlador Lógico Programable) o relé programable de bajo costo**, basado en el microcontrolador **ESP32**, capaz de ejecutar rutinas lógicas configurables para el control de dispositivos eléctricos e industriales.

El producto incluirá:

- Entradas digitales y analógicas.
- Salidas digitales (relevadores o transistores).
- Interfaz de programación mediante conexión Wi-Fi o USB.
- Interfaz web o aplicación básica para cargar programas tipo ladder o lógica secuencial.

El objetivo es proporcionar una alternativa económica y educativa a los PLC comerciales, compatible con entornos de enseñanza y automatización básica.

Este proyecto se mantiene consistente con los principios definidos en la **Descripción del Sistema General del Proyecto ITLA**, enfocado en el desarrollo de soluciones IoT e industriales con microcontroladores programables.

1.3 Personal involucrado

Nombre	Rol	Categoría profesional	Responsabilidades	Información de contacto	Aprobación
Wilfer Javier Ventura	Desarrollador principal /	Estudiante de Ingeniería	Diseño del hardware, desarrollo del firmware	Wilferjavier@gmail.com	✓
Supervisor académico	Carlos Antonio Pichardo Viuque	Docente ITLA	Supervisar el desarrollo técnico y documental del proyecto.	N/A	<input type="checkbox"/>
Colaboradores	Asistentes de laboratorio / validación	Técnicos del laboratorio de electrónica	Pruebas funcionales y validación del prototipo.	N/A	<input type="checkbox"/>

1.4 Definiciones, acrónimos y abreviaturas

Término	Definición
ESP32	Microcontrolador de doble núcleo con Wi-Fi y Bluetooth integrado, fabricado por Espressif.
PLC	Controlador Lógico Programable, usado para automatizar procesos industriales.
GPIO	General Purpose Input/Output: pines configurables de entrada o salida digital.
IDE	Entorno de desarrollo integrado.
Wi-Fi AP	Modo de punto de acceso inalámbrico del ESP32.
RTOS	Sistema operativo en tiempo real (Real-Time Operating System).
Firmware	Software que controla el hardware embebido.

1.5 Referencias

Referencia	Título	Ruta	Fecha	Autor
N/A	N/A	N/A	N/A	N/A

1.6 Resumen

El documento se organiza de la siguiente manera:

- **Capítulo 1:** Introducción y contexto general del sistema.
- **Capítulo 2:** Descripción general, usuarios y limitaciones.
- **Capítulo 3:** Requisitos funcionales y no funcionales detallados.
- **Capítulo 4:** Apéndices con diagramas, tablas de pines y referencias técnicas.

2 Descripción general

2.1 Perspectiva del producto

El sistema **PLC basado en ESP32** es un producto autónomo, pero puede integrarse como parte de un sistema mayor de automatización o IoT.

2.2 Funcionalidad del producto

El producto permitirá:

- Configurar lógicas de control tipo *ladder* o *if-then*.
- Leer entradas digitales y analógicas.
- Activar salidas según la lógica cargada.
- Monitorear estados en tiempo real vía Wi-Fi.
- Guardar rutinas programadas en memoria flash.
- Actualizar firmware mediante conexión USB o red.

2.3 Características de los usuarios

Tipo de usuario	Formación	Habilidades	Actividades
Estudiantes técnicos	Nivel técnico o universitario	Conocimientos básicos de electrónica y programación	Programar, cargar y probar lógicas.
Ingenieros de automatización	Nivel profesional	Dominio de lógica industrial y redes	Integrar el PLC a sistemas mayores.
Docentes	Formación técnica	Evaluación de prácticas y enseñanza	Supervisar y validar el uso educativo.

2.4 Restricciones

Microcontrolador limitado a 520 KB de RAM.

Número máximo de 16 entradas y 8 salidas.

Compatible únicamente con entorno Arduino o PlatformIO.

Alimentación de 5 V o 12 V (según configuración).

Interfaz gráfica limitada a red local Wi-Fi.

2.5 Suposiciones y dependencias

- Se asume que el usuario dispone de conexión Wi-Fi local para configuración.
- El sistema operativo utilizado será **FreeRTOS (nativo del ESP32)**.
- Dependencia del compilador **Arduino-ESP32** y librerías estándar.
- Las actualizaciones se harán manualmente vía USB o OTA (Over The Air).

2.6 Evolución previsible del sistema

- Versión futura con **modbus TCP/RTU**.
- Expansión mediante módulos I/O externos (I2C/SPI).
- Aplicación móvil para programación remota.

3 Requisitos específicos

3.1 Requisitos comunes de los interfaces

3.1.1 3.1.1 Interfaces de usuario

- Interfaz web accesible desde navegador (por IP del dispositivo).
- Menús: Configuración de entradas/salidas, edición de lógica, monitoreo.
- Diseño minimalista y adaptable a móviles.

3.1.2 3.1.2 Interfaces de hardware

- 16 GPIO configurables.
- 8 salidas por relé/transistor.
- 4 entradas analógicas (ADC).
- Comunicación serial UART y Wi-Fi.

3.1.3 3.1.3 Interfaces de software

- Comunicación con entorno de desarrollo por USB (serial).
- Posible integración con sistemas SCADA o MQTT.

3.1.4 3.1.4 Interfaces de comunicación

- Protocolo TCP/IP sobre Wi-Fi.
- UART para consola local.
- En versiones futuras: Modbus TCP/RTU.

3.2 Requisitos funcionales

Nº	Nombre	Tipo	Fuente	Prioridad
RF1	Lectura de entradas digitales	Requisito	Usuario	Alta
RF2	Activación de salidas según lógica	Requisito	Usuario	Alta
RF3	Interfaz web de configuración	Requisito	Supervisor	Media
RF4	Almacenamiento de programa	Requisito	Técnico	Alta
RF5	Visualización en tiempo real	Requisito	Usuario	Media

3.3 Requisitos no funcionales

3.3.1 Rendimiento

- El sistema debe procesar las entradas/salidas en menos de 10 ms por ciclo.
- Debe soportar al menos 10 usuarios conectados simultáneamente al panel web.

3.3.2 Seguridad

- Protección con contraseña en acceso web.
- Uso de protocolo HTTP seguro (WPA2).
- Logs básicos de acceso.

3.3.3 Fiabilidad

- Tiempo medio entre fallos (MTBF) ≥ 1000 horas.
- Recuperación automática tras cortes de energía.

3.3.4 Disponibilidad

- Debe estar disponible el 95 % del tiempo en condiciones normales.

3.3.5 Mantenibilidad

- Firmware actualizable por OTA.
- Código modular y documentado.

3.3.6 Portabilidad

- Compatible con ESP32-WROOM-32 y ESP32-S3.
- Compilable en Windows, Linux o macOS.

3.4 Otros requisitos

- Cumplimiento de normativas eléctricas básicas (< 24 V DC).
- Documentación técnica en español e inglés.
- En caso de uso educativo, incluir guía didáctica.

4 Apéndices

- Diagrama de bloques del sistema.
- Tabla de pines ESP32 utilizados.
- Código fuente del firmware base.
- Manual de usuario.