**SIS 202**

# Diseño funcional

**15 minutos**



## Generalidades

En el curso anterior usted aprendió acerca de los aspectos físicos (hardware) de un sistema instrumentado de seguridad. Ahora concentraremos nuestra atención a los aspectos funcionales (software).

Usted puede pensar que el software es algo fácil de arreglar si hay un problema. Pero en sistemas de seguridad, es esencial hacer lo correcto. Aproximadamente la mitad de los accidentes ocasionados por fallas de control y del sistema de seguridad se originan debido a errores o decisiones deficientes tomadas antes de que los sistemas salgan de la mesa de dibujo. Así que una buena ingeniería en este punto hace una gran diferencia en qué tan bien – o incluso si – su sistema instrumentado de seguridad hará su trabajo.


Fuente: Oficina Ejecutiva de Salud y Seguridad

Este curso describe aspectos clave para garantizar que su software permita que el sistema de seguridad ejecute sus funciones.

Al final del curso, hay un breve examen que puede usar para confirmar lo que ha aprendido – y ganar valiosos Puntos de Recompensa.

**Sugerencia**

Mientras estudia los temas de este curso, busque las respuestas a estas preguntas:

- ¿Cuáles son los tres tipos de software de un sistema de seguridad?
- ¿Cuáles son los dos lados del “modelo V” de desarrollo de software?
- ¿Cuáles son los beneficios de usar módulos de software certificados para sistemas instrumentados de seguridad?

¿Listo(a) para comenzar? Sólo haga clic en el icono “>” a continuación.

## Tipos de software

La norma IEC 61511 identifica tres tipos de software:

1. **Aplicación:** El software que usted desarrolla específicamente para su solución SIS – en otras palabras, la configuración del sistema.
2. **Utilidad:** Las herramientas de software que usted usa para desarrollar, verificar y dar mantenimiento al software de aplicación.
3. **Integrado:** El software (también llamado firmware) que está "integrado" a los productos SIS.

El software de **utilidad** e **integrado** es generalmente proporcionado por los fabricantes de instrumentos y de sistemas de control como parte de sus productos. Cuando estos productos están certificados para aplicación SIS, los proveedores generalmente tienen la responsabilidad primaria de garantizar que este software cumple con las normas IEC 61508.

Por otro lado, toda la responsabilidad de garantizar que el software de **aplicación** cumpla con los requisitos, es de usted – aunque los consultores e integradores pueden ayudar.

## Ciclo de vida de desarrollo

El gigante de las computadoras IBM usa la regla "1-10-100" para explicar la importancia del diseño de buen software: Por cada error de software que se pueda quitar por $1 durante la fase de diseño, si se espera para quitarlo durante las pruebas costará $10, y si se espera hasta que se ha entregado el software costará $100.

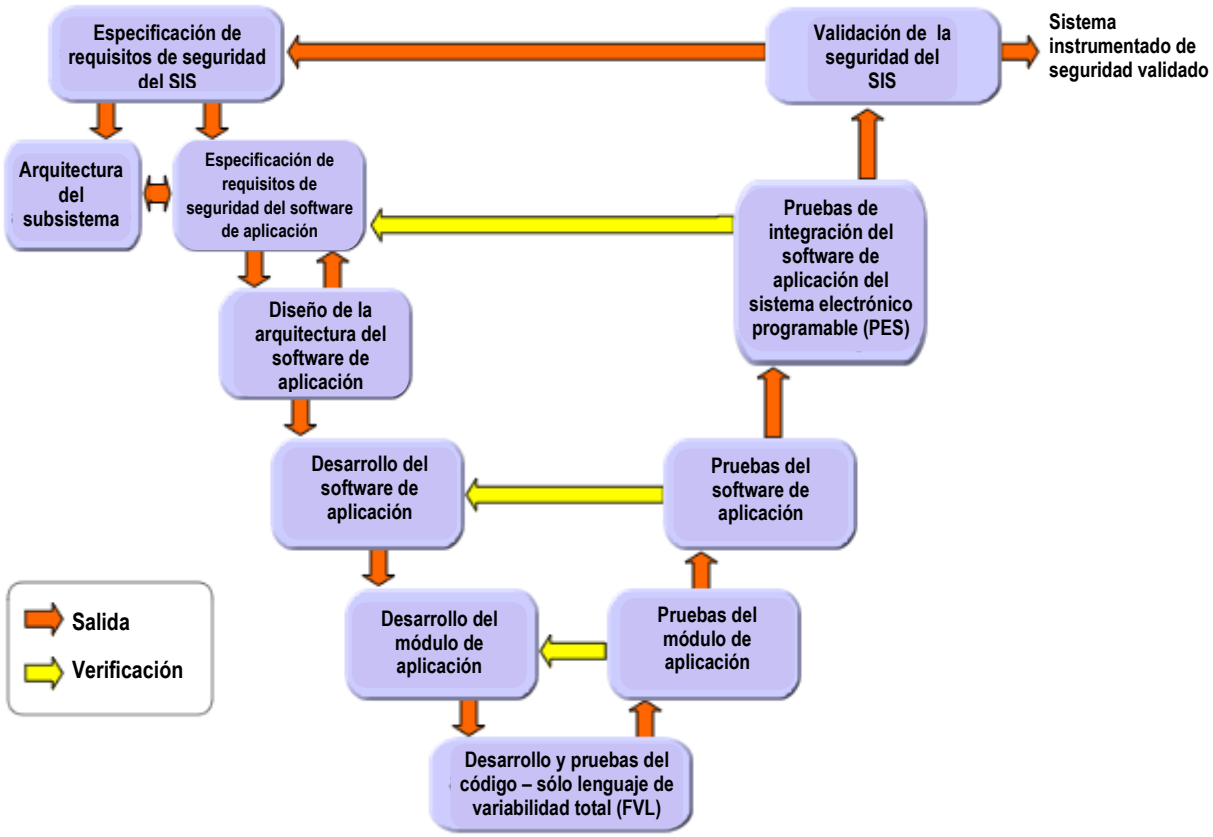
En el mundo de la seguridad, los costos pueden ser mayores – y medidos en términos de vida y miembros. Por eso es importante garantizar la calidad del software desde el principio, y continuar verificándola a lo largo del proceso de desarrollo.

IEC 61511 permite alguna flexibilidad en la manera en que usted desarrolle el software de aplicación para su sistema instrumentado de seguridad. Sin embargo, requiere que el proceso de desarrollo sea estructurado con cuidado para evitar errores de ingeniería que ocasionen fallas peligrosas durante la operación. También requiere que se verifique y se valide que la solución de aplicación del software funcione como se define en la documentación de diseño.

La norma incluye el "modelo V" de desarrollo de software popular para ilustrar las actividades necesarias para garantizar que esto suceda. El lado izquierdo de la V muestra las actividades de desarrollo de software, y el lado derecho muestra las actividades correspondientes de verificación y validación.

**Haga clic en la imagen para ampliarla.**

Ciclo de vida de desarrollo de software (modelo V) de IEC 61511-1




El proceso comienza con la especificación de los requisitos de seguridad (SRS) y progresa a través del diseño cada vez más detallado y etapas de desarrollo. Luego, una serie de pruebas cada vez a mayor escala verifica que el trabajo hecho en cada etapa haya cumplido con los requisitos de seguridad. Al final del proceso, la exitosa prueba de integración conduce al software validado.

Este proceso – incluyendo la planificación y documentación de las pruebas – se describe con más detalle en el siguiente curso, **SIS 203 – Verificación y validación del sistema instrumentado de seguridad.**

## Módulos de software certificados

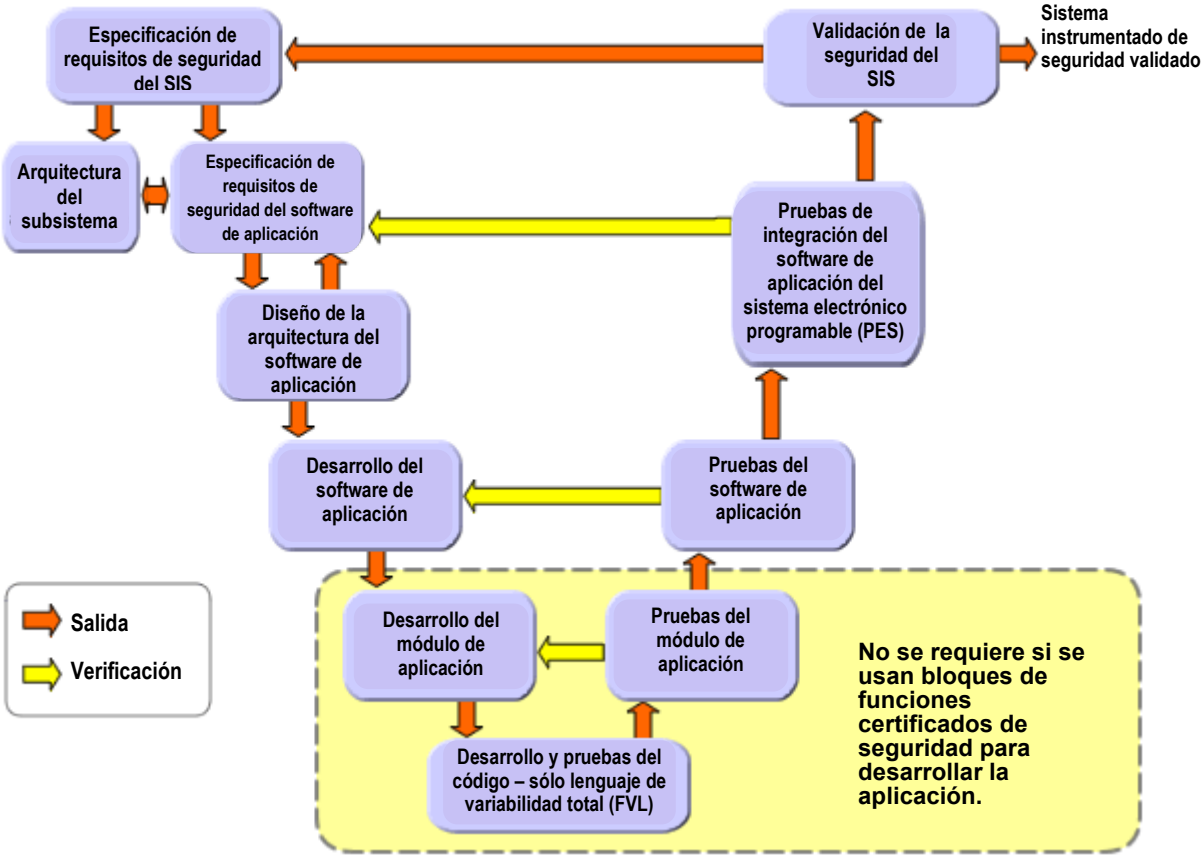
El diseño de software moderno generalmente usa código modular que se desarrolla una vez pero se usa repetidamente – evitando el tiempo, el costo y los errores que pueden resultar de “reinventar la rueda”.

La norma IEC le proporciona dos opciones: crear y validar su propia librería de módulos de software de aplicación, o usar módulos predesarrollados, probados y certificados por un tercero.

Cuando usted usa módulos certificados proporcionados por fabricantes de productos SIS, no sólo evita el tiempo y el costo asociados con el desarrollo de ese código. También evita el requisito de probar cada módulo. El proveedor y la organización certificadora ya han hecho el trabajo por usted.

**Haga clic en la imagen para ampliarla.**

## Ciclo de vida de desarrollo de software (modelo V) de IEC 61511-1




Si usted decide crear sus propios módulos de desarrollo de aplicación, debe comprender que la norma IEC establece requisitos muy estrictos para diseñar, desarrollar y probar estos módulos de software reusables.

El resultado final es que los módulos de software de aplicación deben ser “a prueba de balas”, y quien los desarrolle asume la responsabilidad y el riesgo de garantizar que eso sea así.

**La ventaja PlantWeb**

El sistema DeltaV SIS que es parte del sistema instrumentado de seguridad inteligente de Emerson incluye una completa paleta de bloques de funciones certificados por TÜV incluyendo Voter (votante), Cause and Effect Matrix (matriz de causa y efecto), Step Sequencer (secuenciador por pasos) y State Transition Table (tabla de transición de estado).

Poderosos bloques inteligentes de funciones, tales como bloques de votante MooN (M de N) con funcionalidad de desviación integrada reducen lo que una vez requirió el desarrollo de páginas de lógica de escalera a una simple actividad de configuración arrastrando y soltando (drag-and-drop).

Otras capacidades del software DeltaV SIS, tales como una máquina de estado de alarmas integrada a la norma EEMUA 191, simulación fuera de línea, secuencia de registrador de eventos, manipulación de desviación y anulación, hacen que el mantenimiento de sistemas instrumentados de seguridad sea fácil y menos complejo.

Todas estas capacidades integradas ayudan a automatizar el cumplimiento con IEC 61511, simplificando así los requisitos de documentación y reduciendo sus costos del ciclo de vida y los riesgos.

## Herramientas de utilidad de software

Como cualquier otro oficio, el desarrollo de software tiene herramientas para acelerar y simplificar el trabajo.

IEC 61511 agrupa cosas tales como los lenguajes de programación de aplicación, herramientas de gestión de configuración, simulaciones, arneses de prueba y medición de cobertura de prueba automática bajo el encabezado **software utility tools** (herramientas de utilidad de software). La norma le permite una considerable flexibilidad en la selección de estas herramientas – incluyendo el uso o desarrollo de sus propias herramientas.

Sin embargo, antes de poder usar una herramienta (comprada o desarrollada) para ayudar a lograr el cumplimiento con IEC, el manual de usuario de la herramienta debe documentar completamente lo siguiente:

- Cómo usar la herramienta
- Restricciones de uso
- Puntos débiles conocidos
- Limitaciones de la versión

…y temas similares.

Debido a la importancia de las herramientas de desarrollo de aplicación, la norma requiere que usted use un **intérprete/compilador del lenguaje comprobado** que pueda detectar errores de programación y de sintaxis – y que no introduzca errores él mismo.

**La ventaja PlantWeb**

Una herramienta de utilidad de software clave en el sistema instrumentado de seguridad inteligente de Emerson es el software AMS™ Suite: Intelligent Device Manager, que le permite verificar que la configuración e instalación de los instrumentos de campo sea adecuada. Su capacidad QuickCheck (revisión rápida) también simplifica la validación de interlock al permitirle poner varios instrumentos en modo de revisión fijo al mismo tiempo. Además, durante la modificación del sistema instrumentado de seguridad se puede usar para comparar las nuevas configuraciones de los dispositivos con otras anteriores.

Para ver un estudio de un tercero sobre éstas y otras capacidades, consulte “AMS Safety Analysis: Using AMS Suite in Safety Instrumented System Applications” (Análisis de seguridad de AMS: Uso de AMS Suite en aplicaciones de sistemas instrumentados de seguridad).
<www.emersonprocess.com/sis/resources/ams_safety_analysis.pdf>

## Sumario

En este curso usted ha aprendido que:

- La buena ingeniería de software es esencial para evitar problemas de seguridad que se “diseñan” en el sistema instrumentado de seguridad.
- La norma IEC 61511 identifica tres tipos de software: software de aplicación, software de utilidad y software integrado (también llamado firmware).
- Aunque los proveedores tienen generalmente la responsabilidad primaria por el software de utilidad y el integrado, depende de usted garantizar que el software específico a la aplicación cumpla con la norma.
- Para cada fase del diseño y desarrollo de software, hay una fase de pruebas correspondiente para verificar que el software cumpla con los requisitos.
- El uso de módulos de software precertificados puede reducir el tiempo y costo de desarrollo y pruebas.