



1. Análisis

1.1 Revisión de especificación de requisitos.

1.1.1 Norma IEEE830.

1.1.2 Trazabilidad de requisitos.

1.2 Descripción de procesos actuales.

1.3 Diagramas UML.

1.4 Estudio de Factibilidad.

1.5 Análisis Costo-Beneficio.

Revisión de especificación de requisitos.

- Ayuda a garantizar que todas las partes interesadas tengan un entendimiento común del sistema que se va a desarrollar. Esto evita cualquier malentendido durante las etapas posteriores del desarrollo.
- Sirve como punto de referencia para todas las partes interesadas durante el proceso de desarrollo.
- Ayuda a identificar cualquier brecha en los requisitos en una etapa temprana.
- Reduce el costo general y el tiempo de desarrollo, ya que evita la repetición del trabajo debido a cambios en los requisitos.



Tipos de Requisitos Especificaciones

Especificaciones de requisitos funcionales: (FRS) es un documento que captura las funciones que debe realizar un sistema. Incluye todas las funcionalidades, instalaciones, medidas de seguridad y otra información relevante.

Tipos de Requisitos Especificaciones

Especificaciones de requisitos de rendimiento: (PRS) es un documento que captura todos los aspectos relacionados con el rendimiento de un sistema. Esto incluye el tiempo de respuesta, el rendimiento de datos, la eficiencia, la escalabilidad, entre otras variables.



Tipos de Requisitos Especificaciones

Especificación de requisitos de configuraciones: (CRS) es un documento que captura toda la información relacionada con la configuración de un sistema. Esto incluye detalles como plataformas compatibles, dependencias de software/hardware/firmware, requisitos básicos para la funcionalidad del sistema.

Tipos de Requisitos Especificaciones

Especificaciones de requisitos comerciales: (BRS) es un documento que captura todos los aspectos relacionados con el negocio de un sistema. Esto incluye características tales como administración de usuarios, seguridad, integridad, calidad, gobernanza de datos.



Tipos de Requisitos Especificaciones

Especificaciones de requisitos de confiabilidad: (RRF) es un documento que captura toda la información relacionada con la confiabilidad de un sistema. Esto incluye aspectos como el tiempo de actividad, el tiempo de recuperación, los umbrales cuantitativos y cualitativos con base en niveles de servicio (SLA's).

Tipos de Requisitos Especificaciones

Especificaciones de requisitos de compatibilidad: (CRF) es un documento que captura toda la información relacionada con la compatibilidad de un sistema. Esto incluye aspectos como plataformas compatibles, dependencias de software/hardware/firmware, de los componentes involucrados en el sistema.



Tipos de Requisitos Especificaciones

Especificaciones de requisitos del sistema: SRS es un documento que captura todos los aspectos relacionados con el sistema. Esto incluye aspectos como la funcionalidad, el rendimiento, la escalabilidad, entre otras variables.

Características de un documento de especificación

- **Preciso** – El objetivo de un documento es que sea fácil de comprender. Nada debe quedar poco claro, por lo que no hay diferencia de opiniones entre las partes interesadas. Los requisitos especificados en el documento deben ser muy precisos para evitar cualquier tipo de confusión. No deben tener lagunas, ambigüedades, subjetividad, superlativos o comparaciones.
- **Mensurable** – Los requisitos en el documento deben ser medibles, por lo tanto, el artefacto terminado puede validarse y certificarse según los requisitos. El documento debe incluir suficiente información para que su equipo de desarrollo y evaluadores completen el artefacto y garanticen que cumple con los requisitos del usuario sin eventualidades.
- **Factible** – Los requisitos deben reflejar el estado real de los roles, procesos, tecnología en el cronograma de trabajo establecido por ambas partes.
- **Verificable** – Todos los involucrados en el equipo de desarrollo deben tener acceso al documento para que puedan consultarlo tantas veces como sea necesario.
- **Consistente** – Los requisitos deben ser compatibles. No debe haber ninguna contradicción entre las partes del documento dentro del entorno de la especificación de requisitos. El documento debe estar estructurado de manera consistente y la terminología debe usarse de la misma manera en todo el documento.

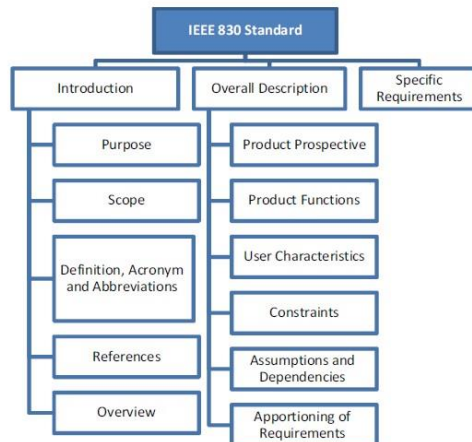


Componentes esenciales

- **Impulsores del negocio.**
- **Modelo de negocio.**
- **Requisitos funcionales y del ecosistema.**
- **Casos de uso del ecosistema** – diagramas basados en el lenguaje de modelado unificado llamado (UML), que explica todas las entidades externas clave que interactuarán con los componentes involucrados en el ecosistema.
- **Requerimientos Técnicos** – esta sección analiza todos los requisitos no funcionales que conforman el entorno técnico y las limitaciones técnicas involucradas en el ecosistema.
- **Cualidades** – definir confiabilidad, facilidad, seguridad, escalabilidad, disponibilidad y la mantenibilidad, de los componentes involucrados en el ecosistema.
- **Limitaciones y supuestos** – esta sección describe todas las limitaciones impuestas al diseño de los componentes involucrados en el ecosistema considerando el equipo de trabajo colaborativo.
- **Criterios de aceptación** – En esta sección son analizados los cumplimientos normativos establecidos considerando los componentes que interactúan en el ecosistema.

Estructura propuesta para un documento

1. Introducción.
 - 1.1. Propósito.
 - 1.2. Público objetivo.
 - 1.3. Uso previsto.
 - 1.4. Alcance.
 - 1.5 Definiciones y siglas.
2. Descripción general.
 - 2.1 Necesidades.
 - 2.2 Supuestos y Dependencias.
3. Características y requisitos del sistema.
 - 3.1 Requisitos funcionales.
 - 3.2 Requisitos de la interfaz externa.
 - 3.3 Requisitos del sistema.
 - 3.4 Requisitos no funcionales.



Especificación de Requisitos según el estándar de IEEE 830

IEEE Std. 830-1998

22 de Octubre de 2008

Resumen

Este documento presenta, en castellano, el formato de Especificación de Requisitos Software (ERS) según la última versión del estándar IEEE 830. Según IEEE, un buen Documento de Requisitos, pese a no ser obligatorio que siga estrictamente la organización y el formato dados en el estándar 830, si deberá incluir, de una forma o de otra, toda la información presentada en dicho estándar. El estándar de IEEE 830 no está libre de defectos ni de prejuicios, y por ello ha sido justamente criticado por múltiples autores y desde múltiples puntos de vista, llegándose a cuestionar incluso si es realmente un estándar en el sentido habitual que tiene el término en otras ingenierías. El presente documento no pretende pronunciarse ni a favor ni en contra de unos u otros: tan sólo reproduce, con propósitos fundamentalmente docentes, cómo se organizaría un Documento de Requisitos según el estándar IEEE 830.

<https://moodle.ucags.edu.mx/course/view.php?id=2082§ion=2>

Norma IEEE 830 - 2009

1. Introducción

- Propósito
- Alcance
- Definición, Acrónimos y Abreviaturas
- Referencias
- Descripción general

2. Descripción general

- Perspectiva del producto
- Funciones del producto
- Características del usuario
- Restricciones
- Suposiciones y dependencias
- Prorrato de requisitos

3. Requisitos específicos

- Interfaces externas
- Funciones
- Requisitos de rendimiento
- Requisitos de la base de datos lógica
- Restricciones de diseño
- Atributos de calidad del sistema
- Organizar los requisitos específicos

4. Información de apoyo

- Tabla de contenido
- Índice
- Apéndices

Shoub, Khaled & Carvalho, Marco & Bond, Walter. (2017). Recommended Practices for the Specification of Multi-Agent Systems Requirements. 10.1109/UEMCON.2017.8249021.



Trazabilidad de requisitos

La trazabilidad permite que los participantes logren propósitos claros dentro de la gestión del proyecto. Además, proporciona elementos que ayudan a la comunicación entre los equipos de trabajo (Team), ya que brinda mayor información para la comprensión del requerimiento y apoya el control de las actividades, gestión de cambios durante todo el ciclo de vida.

La trazabilidad como soporte a los flujos de trabajo

- Los modelos de trazabilidad reconocen tres elementos básicos: los participantes (stakeholders), las fuentes (documentos y modelos) y los objetos o artefactos para ser trazados.
- Estos elementos y su evolución se deben identificar explícitamente en cada flujo de trabajo (Work Flow), para así controlar y soportar el trazado en las fases del proceso.
- Por lo tanto, es necesario que un flujo de control de la trazabilidad apoye los flujos de trabajo en cada iteración. Los modelos de trazabilidad se deben generar por iteración para que los grupos de trabajo tomen decisiones acerca del alcance del desarrollo y del impacto del cambio. Así, se realizarán negociaciones oportunas con los participantes del proyecto. Además, se proveerán elementos para verificar la consistencia e integridad de los modelos de la solución.



- **Establecer criterios para el modelo de trazabilidad.**

Determinar qué participantes, documentos, elementos de modelo participarán en el trazado. Además, se establecen criterios de control del impacto del cambio, tales como operaciones de trazado y método de análisis “**costo-beneficio**”. Estos criterios establecen la forma como los participantes elaborarán e interpretarán los modelos de trazabilidad, logrando un **estándar para todos los proyectos considerando su arquitectura**.

- **Seleccionar elementos de modelo para el trazado.**

Clasificación de los elementos de modelo proporcionados por el flujo de trabajo (Work Flow), en una iteración determinada. Aunque los casos de uso son el centro del desarrollo y de la toma de decisiones.

- **Crear/Actualizar elementos predecesores, sucesores y vínculos de trazado.**

Creación o actualización de los modelos de trazabilidad estableciendo el orden de los elementos (**predecesores-sucesores**), de los componentes involucrados.



- **Verificar completitud y consistencia.**

Reconocer inconsistencias en los “caso de uso” dentro del requerimiento para el desarrollo del prototipo, considerando los componentes, asuntos relacionados con eventualidades (issues), control de cambios (versiones), dentro del ciclo de vida del proyecto.

- **Evaluar el escenario de cambio.**

Evaluar el impacto (tiempo/costo, riesgos, esfuerzo-beneficio), de los cambios solicitados por los participantes, considerando el ciclo de vida del proyecto.

- **Identificar operaciones de cambio y elementos de modelo afectados.**

Reconocimiento de las operaciones de cambio que se deben aplicar a los elementos (crear, modificar, elimina), evitando la "reacción en cadena" durante todo el ciclo de vida del proyecto.



- **Analizar costo-beneficio.**

Realizar la estimación del costo más esfuerzo que requieren los cambios solicitados por los participantes, con base en el modelo de trazabilidad, calculándolos de forma cuantitativa y cualitativa.

Tabla propuesta

Casos de uso \ Requisitos	CU 1	CU 2	CU 3	CU4
Requisito 1	*			
Requisito 2		*		
...			*	
Requisito (N)				*



¿Qué incluye una **matriz de trazabilidad de requisitos**?

- **Nombre y descripción de los requisitos:** nombre concreto de un requisito específico que se necesita para el proyecto, además de una breve descripción considerando el contexto.
- **El ID del requisito:** diferenciar requisitos funcionales similares pero diferentes. cada requisito debería tener un número de “ID único”, para que el equipo pueda identificarlo.
- **La historia del usuario:** es una breve explicación de una función del sistema escrita desde el punto de vista del usuario final, considerando el formato de “Como [perfil], quiero [objetivo del sistema], para lograr [resultado]”.
- **El entregable:** es el producto final y que se alinea con los detalles expuestos en la matriz de trazabilidad de requisitos.
- **El responsable:** es la persona responsable de cumplir con un requisito específico.

Descripción de procesos actuales

Un proceso para generar un sistema es un conjunto de personas, estructuras de organización, reglas, políticas (cumplimientos), actividades, procedimientos, componentes, metodologías, herramientas utilizadas o creadas específicamente para definir, desarrollar, ofrecer un servicio y/o microservicio, para innovar y extender una automatización dentro de un ambiente controlado relacionado con tecnologías de información.



Descripción de procesos actuales

- Permite estandarizar esfuerzos, promover reúso, repetición y consistencia entre proyectos.
- Provee la oportunidad de introducir mejores prácticas internacionales.
- Permite entender que las herramientas deben ser utilizadas para soportar un proceso.
- Establece la base para una mayor consistencia y mejoras futuras.
- Define cómo manejar los cambios y liberaciones por cada iteración.
- Define cómo lograr la transición del sistema a la operación y soporte.

Elementos

- Gestión de actividades.
- Flujo de trabajo (Work Flow)
- Roles (Stakeholders)
- Artefactos (productos)
- Cumplimientos.

Modelos genéricos

- CMM - modelo de madurez de capacidades - estándar de facto
- **CMMI** - modelo integrado
- ISO 9001 - 2000 - sistema para administración de la calidad - estándar
- ISO/IEC 15504 -marco para evaluación de proceso.
- MoProSoft - modelo de procesos para la industria (México)

Modelos específicos

- UP - proceso de desarrollo
- **RUP - proceso unificado de desarrollo**
- **PSP - enfocado en individuos**
- **TSP - enfocado en equipos (incluye PSP)**



Modelo SAMM

El grupo de trabajo internacional llamado OWASP- SAMM [Open Web Application Security Project - Software Assurance Maturity Model]

Tiene un modelo de madurez ágil para la “mejora continua”, relacionada con la seguridad en el ciclo de vida para un sistema, considerando los siguientes aspectos.

Gobernanza	Diseño	Implementación	Verificación	Operaciones
<ul style="list-style-type: none">• Estrategia y Métricas• Política y Cumplimiento• Educación y Orientación	<ul style="list-style-type: none">• Evaluación de amenazas• Requerimientos de seguridad• Arquitectura de seguridad	<ul style="list-style-type: none">• Construcción segura• Implementación segura• Gestión de eventualidades	<ul style="list-style-type: none">• Evaluación de arquitectura• Pruebas basadas en requisitos• Pruebas de seguridad	<ul style="list-style-type: none">• Administración de incidentes• Gestión del Medio Ambiente• Gestión operativa

<https://owasp-samm.org/model/>

OWASP- New Zealand | Friday, February 21, 2020

El ciclo de vida del desarrollo de sistemas - (SDLC) Systems Development Life Cycle

El objetivo del SDLC es reducir los riesgos del proyecto por medio de una planificación anticipada que permita que el sistema cumpla las expectativas durante la fase de producción y posteriormente.

Esta metodología establece una serie de pasos que dividen el proceso de desarrollo del sistema en tareas que se pueden asignar, completar y medir.

Funcionalidad	Modelos
<ul style="list-style-type: none">• Planificación• Diseño• Implementación• Pruebas• Despliegue• Mantenimiento	<ul style="list-style-type: none">• Cascada• Iterativo• Espiral• Ágil



Tipos de modelos

- **Cascada:** dispone todas las etapas secuencialmente de modo que las nuevas etapas dependan del resultado de la etapa anterior.
- **Iterativo:** los equipos comienzan el desarrollo del sistema con un pequeño subconjunto de requisitos. Posteriormente, se mejoran las versiones de manera iterativa a lo largo del tiempo hasta que esté listo para pasar a producción. El equipo produce una nueva versión del sistema al final de cada iteración.
- **Espiral:** combina los pequeños ciclos repetidos del modelo iterativo con el flujo secuencial y lineal del modelo de cascada para dar prioridad al análisis de riesgos, considera las mejoras graduales del sistema mediante la creación de prototipos en cada fase.
- **Ágil:** El modelo ágil dispone las fases del SDLC en varios ciclos de desarrollo. El equipo itera a través de las fases rápidamente y solo se hacen pequeños cambios progresivos del sistema en cada ciclo. Los requisitos, planes y resultados son evaluados continuamente para responder con rapidez a los cambios. El modelo ágil es iterativo y progresivo, por lo que es más eficiente que otros modelos de procesos.