

Unidad Número 4

Propiedades son del software.

Requisitos son del usuario.

Requerimiento (IEEE 729)

Es una condición o capacidad necesitada por un usuario para resolver un problema o lograr un objetivo.

Una condición o capacidad que debe ser alcanzada o poseída por un sistema para satisfacer un contrato, estándar, especificación, o algún otro documento formalmente impuesto.

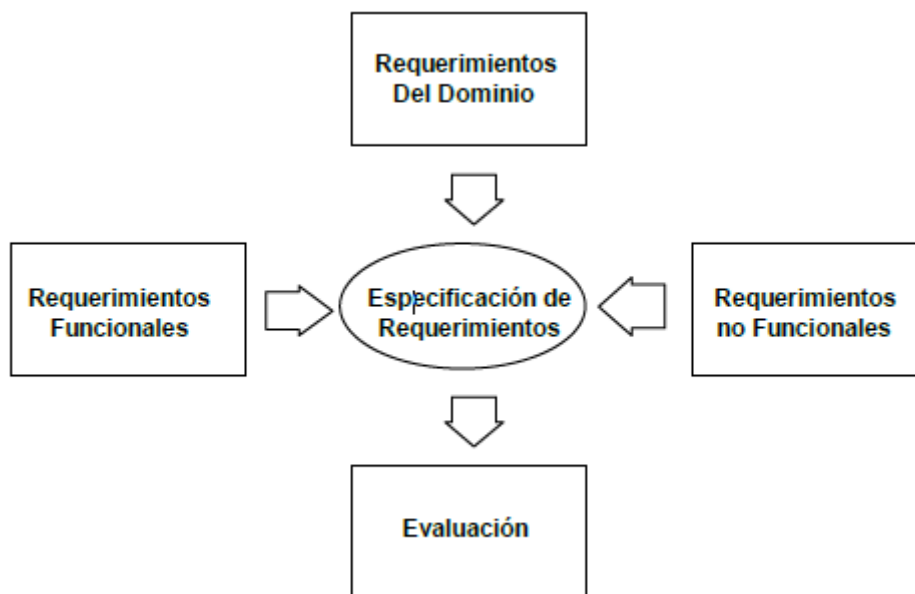
- Requerimientos del usuario: son declaraciones en lenguaje natural y en diagramas, de los servicios que se espera que el sistema provea y las restricciones bajo las cuales debe operar.
Su función es describir los requerimientos funcionales y no funcionales de forma tal que sean comprensibles por quienes no tienen un conocimiento técnico detallado.
Los requerimientos del usuario deberán enfocarse a los recursos principales a proveer por el sistema.

Ejemplo: El sistema debe proveer de un medio para graficar la evolución de las cuentas a cobrar.

- Los requerimientos del sistema: Son descripciones más detalladas de los requerimientos del usuario.
Son la base para definir el contrato entre el cliente y el desarrollador y por lo tanto debe ser una descripción consistente y completa del sistema.
Esta especificación es el punto de partida para el diseño del sistema, por parte de los Ingenieros de Software.

Ejemplo: Al usuario se le proveerá la posibilidad de elegir el período en el cual se quiere efectuar el análisis de la evolución de las cuentas a cobrar.

Esquema del proceso de Especificación de Requerimientos



Categorías de Requerimientos

- ✓ Funcionales: Son declaraciones de los servicios o funcionalidad que proveerá el sistema, de la manera en que éste reaccionará a entradas particulares (eventos externos) y de cómo se comportará en situaciones particulares. En algunos casos, los requerimientos funcionales de los sistemas también declaran explícitamente lo que el sistema no debe hacer.
 - *Ejemplo*: El usuario deberá tener la posibilidad de generar el listado de alumnos que se encuentran cursando determinada asignatura en un cuatrimestre.
 - *Ejemplo*: El sistema deberá controlar que el alumno tenga aprobadas las asignaturas correlativas para permitir su inscripción a examen final.
- ✓ No funcionales: Son restricciones de los servicios o funciones ofrecidas por el sistema. Incluyen restricciones de tiempo, sobre el proceso de desarrollo, estándares, etc.
 - *Interface*:
 - Help
 - Visualización de datos
 - Amigable
 - *Seguridad*:
 - Permiso de acceso
 - Niveles de seguridad
 - Distribución de datos
 - *Performance*:
 - Tiempo real

- Restricciones de tiempo
- Velocidad de procesamiento

Sommerville agrupa a los requerimientos no funcionales en tres categorías:

- **Requerimientos del producto:**
Son aquellos que especifican el comportamiento del producto.
- **Requerimientos organizacionales:**
Se derivan de las políticas y procedimientos existentes en la organización, tanto del cliente como del desarrollador de software.
- **Requerimientos externos:**
Se derivan de factores externos al sistema. Por ejemplo, la manera en que el sistema interactúa con otros sistemas de la organización; requerimientos legales o requerimientos éticos.

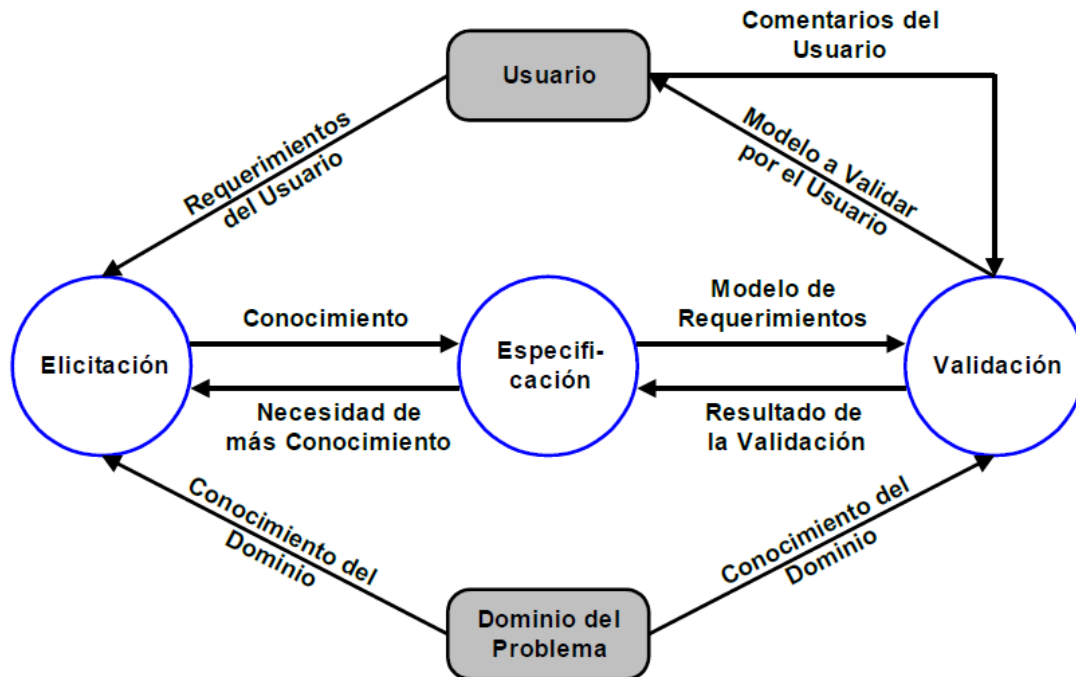
- ✓ **Dominio:** Son requerimientos que provienen del dominio de aplicación del sistema y que reflejan las características de ese dominio. Estos pueden ser a su vez, funcionales o no funcionales.

-Funcionales: contenido de la información.

-No funcionales: uso de cierta tecnología o formato.

- Sistema de calificación de un estudiante
- Normas de facturación
- Formato de los datos
- Condiciones para auditoría
- Comunicación con sistemas de jerarquía superior externos a la organización

Procesos de la Ingeniería de Requerimientos



Especificación de Requerimientos (Loucopoulos)

Tres subprocesos:

- Elicitación

Captura las necesidades de los clientes que provienen del dominio del problema y de los usuarios, que serán recuperados a través de las técnicas de recolección de datos. Este conocimiento capturado es la entrada al proceso de Especificación.

(Notas de relevamiento)

- Especificación

Se encarga de representar el conocimiento en un conjunto de notaciones, utilizando las técnicas de documentación. Estos modelos de requerimientos, serán la entrada al subproceso de validación.

(Fichas de caso de uso, diagrama caso de uso)

- Validación

Los modelos de requerimientos se presentarán al usuario para que este certifique que representa sus necesidades. También serán contrastados contra el dominio del problema. Si los modelos son correctos y están completos, el proceso de requerimientos ha concluido, de lo contrario es devuelto al subproceso de especificación con los problemas encontrados.

Actividades para determinación requerimientos (Elicitación)

- Reutilización de requerimientos: Si el analista ya ha trabajado antes analizando otros dominios de aplicación similares, sabrá los requerimientos propios del dominio, así como también donde habitualmente se producen los problemas y pudiendo, de esta manera anticipar los requerimientos e investigar directamente las áreas de conflicto.
 - Idea de base
 - Mejora global del proceso en tiempo
 - Similitud en sistemas en componentes
 - Calidad y evitar errores
 - Menor costo que elicitar desde cero
- Investigación del sistema: Estudio del sistema actual utilizando las herramientas para detectar hechos.

Técnicas de Elicitación

- Partiendo del usuario
- Escenarios
- Análisis de formularios
- Lenguaje natural

Partiendo del usuario

El más intuitivo y directo de los enfoques.

Dificultades

1. Poca claridad del usuario
2. Dificultad del usuario para transmitir su conocimiento
3. Diferencia entre usuarios y el analista
4. El usuario puede no querer el sistema

Técnicas:

- Entrevistas de comienzo y final abierto

El analista deja que el usuario hable respecto de cómo desarrolla sus tareas. Se deben realizar en un ambiente informal para estimular la confianza del usuario y disminuir la presión que puede provocarle al estar siendo entrevistado.

- Entrevistas estructuradas

Direcciona al usuario hacia aspectos específicos, obtengo información detallada, se utilizan preguntas cerradas, abiertas, de sondeo y de control.

➤ Brainstorming

Facilita la falta de consenso entre usuarios, ayuda a entender el dominio del problema, varios usuarios debaten entre ellos sobre cómo se le puede dar solución a una cuestión, ayuda al analista a entender el problema.

Escenarios / Casos de Uso

Es una historia que ilustra como un sistema puede satisfacer necesidades del usuario, descripción idealizada y detallada de una instancia específica de interacción hombre-maquina / hombre-sistema.

Ventajas:

1. Los usuarios tienen la facilidad de transmitir su experiencia a través de contar su historia.
2. Solución prometedora al problema de la comunicación.

Análisis de formularios

Es un modelo formal de datos con información sobre la organización y dan conocimientos del dominio.

Dificultades:

1. Contienen información que ya no se utiliza o dejó de ser relevante.
2. Incorporan al sistema información que ya no es necesaria para el mismo.

Lenguaje natural

Forma más habitual de representación del conocimiento, directo sobre el dominio del problema interactuando con un usuario o desde un texto escrito en lenguaje natural.

Ventajas:

1. Vocabulario preexistente
2. Informalidad
3. Sintaxis conocida

Desventajas:

1. Complejo por la lista de términos ilimitados
2. El significado de un término puede ser ambiguo

Administración de Requerimientos:

La administración de requerimientos es el proceso de comprender y controlar los cambios en los requerimientos del sistema.

- Cada requerimiento debe identificarse de manera única.
- Administración del cambio de los requerimientos.
- Políticas de rastreo de requerimientos (hacia adelante o hacia atrás).
- Herramientas CASE.

Rastreo de Requerimientos:

- Rastreo de la fuente: vincula los requerimientos con quienes los propusieron y la razón de éstos.
- Rastreo entre requerimientos: vincula los requerimientos dependientes entre sí en el documento de requerimientos.
- Rastreo del diseño: vincula los requerimientos a los módulos del diseño en el cual serán implementados.

Herramientas para administrar requerimientos:

- Almacenar requerimientos: los requerimientos deben mantenerse en un almacén de datos seguro y administrado.
- Administrar el cambio de los requerimientos: este proceso se simplifica si se dispone de una herramienta de ayuda.
- Administrar el rastreo de los requerimientos: para ayudar a descubrir relaciones entre los requerimientos.

Administrar el cambio de los requerimientos:

Esta administración se aplica a todos los cambios propuestos en los requerimientos.

- Análisis del problema y especificación del cambio
- Análisis del impacto del cambio y su costo
- Plan de implementación del cambio

Unidad Número 5

Especificación de Requerimientos de Software (SRS)

La SRS es una especificación para un producto de software en particular, programa, o conjunto de programas que realizan ciertas funciones en un ambiente específico. La SRS puede escribirse por uno o más representantes del proveedor, uno o más representantes del cliente, o por ambos.

Motivaciones para la SRS:

- Documentación de los requerimientos.
- Herramienta de comunicación de
 - La comprensión del dominio donde opera (Normativa de la AFIP para facturar)
 - Contexto de organización (Empresa o negocio)
 - El sistema empleado
- Es parte de un acuerdo contractual.

Pasos según ANSI/EEE830-1984

Introducción

- Propósito que se persigue con la SRS.
- Alcances que va a llegar ese producto.
- Definiciones, Acrónimos y Abreviaturas que utilizara.
- Referencias con el cumplimiento de un marco narrativo.

Descripción general

- Perspectiva del producto (Que se espera del mismo).
- Funciones del producto (Propiedades funcionales).
- Características del usuario (Para qué tipo de usuarios está pensado).
- Restricciones generales (Si existen restricciones para el uso).
- Presunciones y dependencias (Puesta en desarrollo y operación).

Especificación de requerimientos

Requerimientos funcionales

- Requerimiento funcional 1
 - Introducción (Explica brevemente en que consiste)
 - Entrada (Datos ingresados)
 - Procesamiento (Procesamiento del requerimiento)
 - Salida (Información finalizada)
- Requerimiento funcional 2
- Requerimiento funcional n

Requerimientos de las interfaces externas

- Interfaces de usuario (Interfaz gráfica o de texto).
- Interfaces de hardware (Hardware con el que va a interactuar el producto).
- Interfaces de software (Sobre que sistema operativo debe correr el producto).
- Interfaces de comunicación (Comunicación con otros productos u otros sistemas).

Requerimientos de performance

- Tiempo de respuestas que va a tener que cumplir el producto.

Restricciones de diseño

- Cumplimiento con estándares (Respetar 2 decimales en la moneda).
- Limitaciones de Hardware (Funcionar sobre equipos con poca memoria).

Atributos

- Disponibilidad (Tiene que activo los 365 días del año).
- Seguridad (Posibilidad de acceder a los datos del sistema o ingreso al mismo).
- Mantenibilidad (Mantenimiento del mismo por el equipo desarrollador y con qué disponibilidad).
- Transferibilidad/Conversiones (Es posible el reemplazo del mismo o adaptaciones).

Otros requerimientos

- Bases de datos (Ciertos motores de base de datos).
- Operaciones (Operar en ciertas condiciones).
- Adaptación al sitio (Adaptar el sistema a la organización).

Unidad Número 6

Herramientas Computer Aided Software Engineering (CASE).

La primera CASE fue Excelator la que se puso en producción en 1984 sobre pc's.

En la década de los '90 aparecen las CASE para mainframes impulsadas por IBM.

Objetivos de las CASE

- Asistir para mejorar la productividad en el desarrollo y mantenimiento del software.
- Aumentar la calidad de los productos de software.
- Reducir el tiempo y el costo del proceso de desarrollo y facilitar el mantenimiento de los productos de software.
- Mejorar la planificación y administración de los proyectos de desarrollo de software.
- Aumentar la biblioteca de conocimiento de una organización ayudando la búsqueda de soluciones para los requisitos.
- Automatizar el desarrollo del software, la documentación, la generación de código, las pruebas de errores y la gestión del proyecto.
- Ayudar a la reutilización del software, portabilidad y estandarización de la documentación.
- Posibilitar la gestión de todas las fases de desarrollo de software con una misma herramienta.
- Facilitar el uso de las distintas metodologías propias de la ingeniería del software.

Componentes de las CASE

- Diccionario de datos o repositorio
- Meta modelo
- Facilidad para la carga de datos
- Capacidad de comprobar errores de consistencia
- Interfaz de usuario amigable

Tipos de herramientas CASE

Toolkit: es una colección de herramientas integradas que permiten automatizar un conjunto de tareas de algunas de las fases del ciclo de vida: Planificación estratégica, Análisis, Diseño, Generación de código.

Ejemplo: Asistir solo a una fase (Administrar requisitos)

Workbench: Son conjuntos integrados de herramientas que dan soporte a la automatización del proceso completo de desarrollo del producto de software. Permiten cubrir el ciclo de vida completo. El producto final aportado por ella es el código ejecutable y su documentación.

Ejemplo: Asistir al ciclo completo

Fase dependiendo el ciclo de vida al que se aplica

- ✓ Upper Case: Planificación estratégica, requerimientos (Etapa temprana).
- ✓ Middle Case: Análisis y Diseño (Etapa media).
- ✓ Lower Case: Generación de código, test e implantación (Etapa final).