

INGENIERÍA DE SOFTWARE Y DE REQUERIMIENTOS

ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE

ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS

M.Sc. Ing. Raúl Córdova
2022

6.1 INTRODUCCIÓN

Como muestra la Figura 1, el proceso de requisitos tiene la forma de una espiral.

Luego de negociar requisitos es necesario documentarlos.

Se obtiene un borrador que luego se debe validar.

La validación permite encontrar errores o detectar lagunas de información.

Por tanto, el proceso es iterativo.

Al final se tendrá el documento de Especificación de Requisitos de Software (ERS – SRS) validado y listo para ser usado en las siguientes etapas del proceso de Ingeniería de Software.

6.1 INTRODUCCIÓN

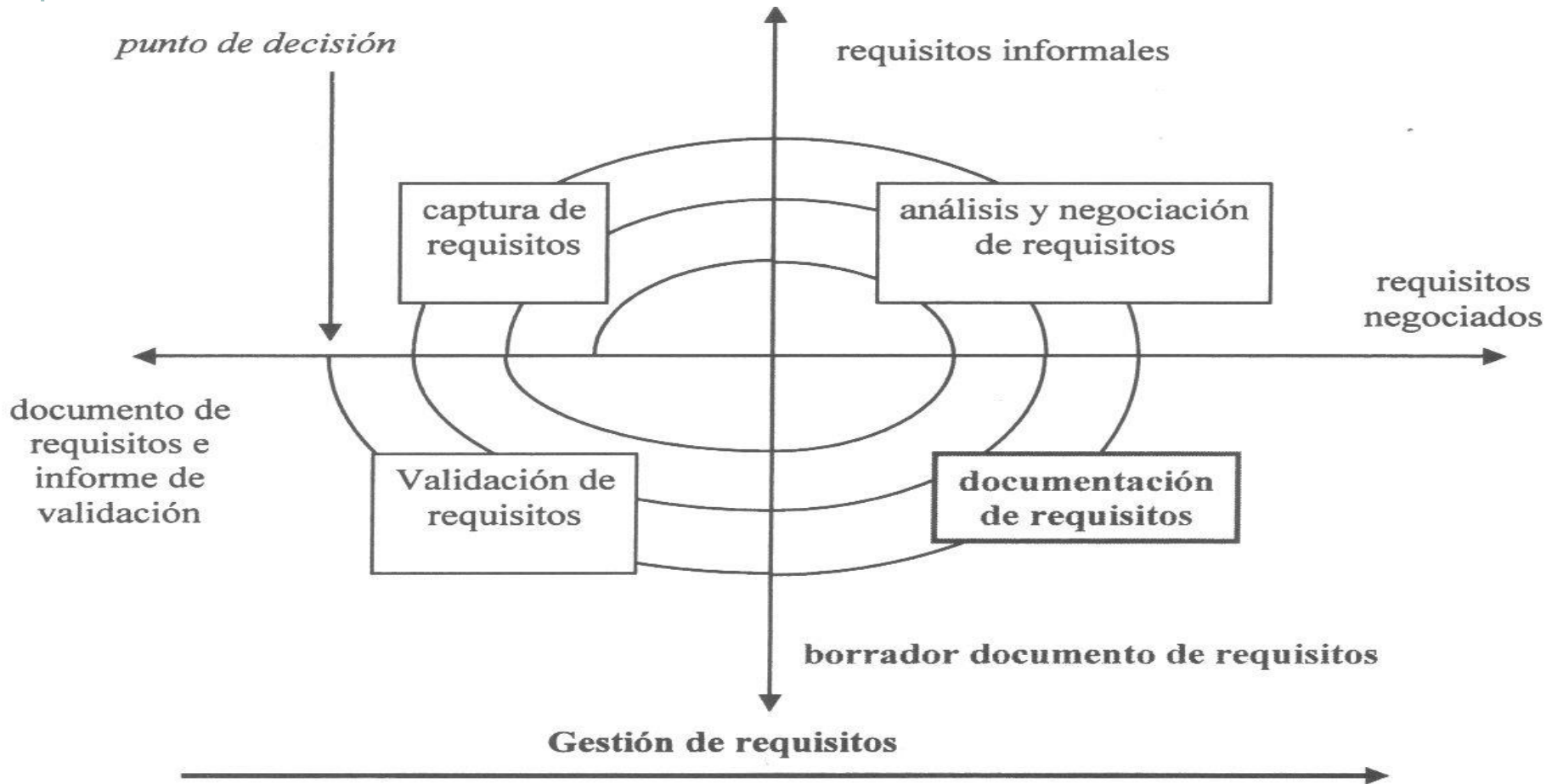


Figura 1. Proceso de la ingeniería de requisitos.

6.1 INTRODUCCIÓN

Objetivos del usuario o cliente:

- Definir la necesidad, es decir, lo que se quiere y lo que no se quiere que haga el sistema.

Objetivos del desarrollador:

- Facilitar la comunicación entre cliente, usuarios y desarrolladores
- Permitir iniciar la actividad de diseño
- Soportar las actividades de pruebas del sistema
- Controlar la evolución del sistema

6.2 DOCUMENTACIÓN DE REQUERIMIENTOS

Un documento de requisitos debe contener:

- Información acerca del problema
- Interfaz externa del sistema con su entorno (software, hardware, usuarios, puertos de comunicación)
- Propiedades y comportamiento del sistema
- Restricciones de diseño y fabricación del producto
- Descripciones acerca de cómo el sistema ayudará a sus usuarios a realizar mejor sus tareas
- Restricciones acerca de la tecnología que será utilizada en la construcción del sistema
- Restricciones acerca de las propiedades emergentes del sistema (requisitos no funcionales)

6.3 ESCRITURA DE REQUERIMIENTOS

Un conjunto de requerimientos se puede agrupar según:

1. El mismo estímulo externo. Ejemplo: en un sistema automático de aterrizaje se pueden agrupar los requisitos relativos al sensor de altitud, al nivel de combustible, etc.
2. La misma característica del sistema. Ejemplo: en un sistema de telefonía se pueden agrupar los requisitos relativos a llamadas locales, internacionales, etc.

6.3 ESCRITURA DE REQUERIMIENTOS

3. La misma respuesta del sistema. Ejemplo: en un sistema de nóminas se pueden agrupar los requisitos relativos a generar pagos, generar listas de empleados actuales, generar listas de salario medio por cargo, etc.
4. El mismo objeto del mundo real. Ejemplo: en un sistema de biblioteca se pueden agrupar los requisitos relativos a libros, a usuarios, etc.

6.3 ESCRITURA DE REQUERIMIENTOS



5. La misma clase de usuarios. Ejemplo: en el sistema de la biblioteca se pueden separar los requisitos del bibliotecario, del usuario, etc.
6. La misma clase de función. Ejemplo: datos de entrada, interfaz de usuario, generación de informes, mantenimiento de la base de datos, etc.

También se puede establecer una jerarquía multinivel. Primero se clasifican los requisitos por usuario y luego por características del sistema, por ejemplo.

6.4 EL ESTÁNDAR IEEE 830

Tabla de Contenidos:

1. Introducción

1.1 Propósito

1.2 Alcance

1.3 Definiciones, acrónimos y abreviaturas

1.4 Referencias

1.5 Visión general del documento

6.4 EL ESTÁNDAR IEEE 830

2. Descripción general

2.1 Perspectiva del producto

2.2 Funciones del producto

2.3 Características de los usuarios

2.4 Restricciones

2.5 Suposiciones y dependencias

6.4 EL ESTÁNDAR IEEE 830

3. Requisitos específicos

3.1 Interfaces externas

3.2 Funciones

3.3 Requisitos de rendimiento

3.4 Restricciones de diseño

3.5 Atributos de calidad

4. Apéndices

6.4.1. INTRODUCCIÓN

- Aquí se proporciona una introducción a todo el documento.
- Consta de las siguientes secciones:
 - Propósito
 - Alcance
 - Definiciones, acrónimos y abreviaturas
 - Referencias
 - Visión general del documento

6.4.1. INTRODUCCIÓN

1.1 Propósito: define el propósito del documento y se especifica a quién va dirigido.

1.2 Alcance:

- Se define el nombre del sistema
- Se explica lo que el sistema hará y no hará
- Se describen los beneficios, objetivos y metas que se espera alcanzar con el futuro sistema
- Se referencian todos los documentos de nivel superior (p.e. SyRS) con los cuales este documento debe tener consistencia, si existen.

6.4.1. INTRODUCCIÓN

1.3 Definiciones, acrónimos y abreviaturas:

- Aquí se definirán todos los términos, acrónimos y abreviaturas utilizadas en la ERS.

1.4 Referencias:

- Aquí se mostrará la lista completa de todos los documentos referenciados en la ERS.

1.5 Visión general del documento: describe brevemente los contenidos y la organización del resto de la ERS.

6.4.2. DESCRIPCIÓN GENERAL

Aquí se describen todos aquellos factores que afectan al producto y a sus requisitos.

No se describen los requisitos, sino su contexto, permitiendo definir con detalle los requisitos en la sección 3, haciendo que sean más fáciles de entender.

Consta de las siguientes secciones:

- Perspectiva del producto
- Funciones del producto
- Características de los usuarios
- Restricciones
- Suposiciones y dependencias

6.4.2. DESCRIPCIÓN GENERAL

2.1 Perspectiva del producto:

- Aquí se debe relacionar el futuro sistema (producto software) con otros productos.
- Si el producto es totalmente independiente de otros productos, también se debe especificar aquí.
- Si la ERS define un producto que es parte de un sistema mayor, se deben relacionar los requisitos del sistema mayor con la funcionalidad del producto software y se deben identificar las interfaces entre el producto mayor y el producto aquí descrito.
- Se recomienda utilizar diagramas de bloques.

6.4.2. DESCRIPCIÓN GENERAL

2.2 Funciones del producto:

- En esta subsección se mostrará un resumen, a grandes rasgos, de las funciones del futuro sistema.
- Las funciones deberán mostrarse de forma organizada, por ejemplo, dividiendo al sistema en módulos.
- También pueden utilizarse gráficos, siempre y cuando dichos gráficos reflejen las relaciones entre funciones y no el diseño del sistema.

6.4.2. DESCRIPCIÓN GENERAL

2.3 Características de los usuarios:

- En esta subsección se describirán las características generales de los usuarios del producto, tales como nivel educacional, experiencia general y experiencia técnica.

6.4.2. DESCRIPCIÓN GENERAL

2.4 Restricciones: aquí se describirán aquellas limitaciones que se imponen sobre los desarrolladores del producto, como:

- Políticas de la empresa
- Limitaciones del hardware
- Interfaces con otras aplicaciones
- Operaciones paralelas
- Funciones de auditoría
- Funciones de control
- Lenguaje(s) de programación
- Protocolos de comunicación
- Requisitos de habilidad
- Criticidad de la aplicación
- Consideraciones acerca de la seguridad

6.4.2. DESCRIPCIÓN GENERAL

2.5 Suposiciones y dependencias:

Se describen aquellos factores que, si cambian, pueden afectar a los requisitos.

Por ejemplo, los requisitos pueden presuponer una cierta organización de ciertas unidades de la empresa, o pueden presuponer que el sistema correrá sobre cierto sistema operativo.

Si cambian dichos detalles organizativos en la empresa, o si cambian ciertos detalles técnicos, como el sistema operativo, puede ser necesario revisar y cambiar los requisitos.

6.4.3. REQUISITOS ESPECÍFICOS

Esta sección contiene los requisitos a un nivel de detalle suficiente como para permitir a los diseñadores diseñar un sistema que satisfaga estos requisitos, y que permita al equipo de pruebas planificar y realizar las pruebas que demuestren si el sistema satisface, o no, los requisitos.

Todo requisito describirá comportamientos externos del sistema, perceptibles por los usuarios, operadores y otros sistemas.

6.4.3. REQUISITOS ESPECÍFICOS

Los requerimientos deberían incluir como mínimo:

- Una descripción de cada entrada (estímulo) al sistema
- Una descripción de cada salida (respuesta) del sistema
- Toda la funcionalidad realizada por el sistema en respuesta a una entrada o como apoyo a una salida

Es la sección más larga e importante de la ERS.

6.4.3. REQUISITOS ESPECÍFICOS

Deberán aplicarse los siguientes principios:

- El documento debería ser perfectamente legible por personas de muy distintas formaciones e intereses.
- Deberán referenciarse aquellos documentos relevantes que poseen alguna influencia sobre los requisitos.
- Todo requisito deberá ser unívocamente identificable mediante algún código o sistema de numeración adecuado.
- Los requisitos deben cumplir con las características de calidad ya conocidas como: no ambigüedad, concisión, consistencia interna y externa, realizabilidad, verificabilidad, independencia del diseño, no redundancia, etc.

6.4.3. REQUISITOS ESPECÍFICOS

3.1 Interfaces externas:

Se describirán los requisitos de interfaz de usuario, interfaces de hardware, interfaces de software, interfaces de comunicación e interfaces de sistema, estas últimas solo si el software es parte de un sistema mayor.

También se describirán las restricciones de memoria, operaciones y requerimientos de adaptación al sitio.

6.4.3. REQUISITOS ESPECÍFICOS

3.1.1 Interfaces de usuario. Se debe especificar lo siguiente:

- a) Las características lógicas de cada interface entre el producto software y sus usuarios, lo que incluye las características de configuración necesarias para cumplir con los requerimientos de software, tales como:
- formatos de pantalla,
 - diseño de páginas o ventanas,
 - contenido de cualquier reporte o menú,
 - disponibilidad de las teclas de función programables

6.4.3. REQUISITOS ESPECÍFICOS

- b) Todos los aspectos para la optimización de la interface con la persona que debe usar el sistema, lo que implica listar lo que hará y no hará el sistema cuando interaccione con el usuario.
 - Ejemplo: un requerimiento para la opción de escoger entre mensajes largos o cortos de error, que también deben ser verificables, tal como: “un digitador puede realizar una función X en Z minutos después de 1 hora de entrenamiento”.

6.4.3. REQUISITOS ESPECÍFICOS

3.1.2 Interfaces de hardware

Aquí se deben especificar las características lógicas de cada interface entre el producto software y los componentes de hardware del sistema, que incluye características de configuración como:

- Número de puertos
- Conjuntos de instrucciones

También se deben describir los dispositivos a ser utilizados, junto con sus características.

6.4.3. REQUISITOS ESPECÍFICOS

3.1.3 Interfaces de software

Aquí se deben especificar otros productos de software requeridos, como:

- Sistemas de gestión de datos
- Sistema operativo
- Paquetes matemáticos

También se especifican las interfaces con otros sistemas software, como el de Facturación Electrónica del SRI.

6.4.3. REQUISITOS ESPECÍFICOS

3.1.3 Interfaces de software

Para cada producto requerido, se debería indicar:

- El nombre
- El Mnemónico (nombre simbólico)
- El número de especificación
- El número de versión
- La fuente

6.4.3. REQUISITOS ESPECÍFICOS

3.1.3 Interfaces de software

Por cada interface, se debería indicar:

- El propósito de su relación con el producto software en desarrollo.
- Definición de las interfaces en términos del contenido y formato de los mensajes.
- Una referencia al documento que define la interface.

6.4.3. REQUISITOS ESPECÍFICOS

3.1.4 Interfaces de comunicación

Aquí se deberían especificar las diferentes interfaces para comunicación, tales como protocolos de red.

También otras características como pórtilos usados, velocidades de transmisión, etc.

6.4.3. REQUISITOS ESPECÍFICOS

3.1.5 Restricciones de memoria

Aquí se deberían especificar las características y los límites mínimos y máximos de las memorias principal y secundaria.

6.4.3. REQUISITOS ESPECÍFICOS

3.1.6 Operaciones: aquí se deberían especificar las operaciones normales y especiales requeridas por los usuarios, tales como:

- a) Los diferentes modos de operaciones en la organización del usuario (como operaciones iniciadas por el usuario)
- b) Periodos de operaciones interactivas y periodos de operaciones no atendidas
- c) Funciones de soporte al procesamiento de datos
- d) Operaciones de respaldo y recuperación

6.4.3. REQUISITOS ESPECÍFICOS

3.1.7 Requerimientos de adaptación al sitio. Se deberían:

- a) Definir los requerimientos para cualquier dato o secuencias de inicialización específicas para un sitio, misión o modo de operación determinados, como valores de grilla, límites de seguridad, etc.
- b) Especificar las características del sitio o relacionadas a la misión, que deberían ser modificadas para adaptar el software a una instalación en particular.
- c) Determinar las condiciones de las instalaciones eléctricas, de seguridad, ambientales, de humedad, etc., necesarias para el adecuado funcionamiento del producto software.

6.4.3. REQUISITOS ESPECÍFICOS

3.2 Funciones.

En esta subsección se deberá especificar todas aquellas acciones (funciones) que deberá llevar a cabo el software.

Si se considera necesario, podrán utilizarse notaciones gráficas y tablas, pero siempre supeditadas al lenguaje natural, y no al revés.

Normalmente son aquellas expresables como “El sistema deberá....”.

Estas expresiones deben incluir:

6.4.3. REQUISITOS ESPECÍFICOS

3.2 Funciones.

- a) Chequeos de validación para las entradas
- b) Secuencia exacta de las operaciones
- c) Respuestas ante situaciones anormales, como
 - 1) Overflows (desbordamientos)
 - 2) Problemas de comunicación
 - 3) Manejo y recuperación de errores
- d) Efecto de los parámetros
- e) Relaciones entre las salidas y entradas, incluyendo:
 - a) Secuencias de entrada/salida
 - b) Fórmulas para la conversión de las entradas en salidas

6.4.3. REQUISITOS ESPECÍFICOS

3.2 Funciones.

También puede ser apropiado dividir los requerimientos funcionales en subfunciones o subprocesos.

Esto no implica que el diseño del software también será dividido de la misma manera.

6.4.3. REQUISITOS ESPECÍFICOS

3.2 Funciones.

Esta subsección se puede organizar de múltiples formas, tal como se indicó en la sección 6.1.3 de este documento.

Para nuestro caso se escogerá la opción de organizar los requerimientos funcionales por Módulos, que permiten definir la funcionalidad de cada parte del sistema de manera altamente cohesa y bajamente acoplada.

La organización será la siguiente:

6.4.3. REQUISITOS ESPECÍFICOS

3.2 Funciones.

3.2.1 Módulo 1

3.2.1.1 Propósito del módulo

3.2.1.2 Requerimientos del módulo

3.2.1.2.1 Requerimiento funcional M11

3.2.1.2.1.1 Descripción

3.2.1.2.1.2 Entradas

3.2.1.2.1.3 Proceso

3.2.1.2.1.4 Salidas

3.2.1.2.2 Requerimiento funcional M12

3.2.1.2.2.1 Descripción

3.2.1.2.2.2 Entradas

3.2.1.2.2.3 Proceso

3.2.1.2.2.4 Salidas

6.4.3. REQUISITOS ESPECÍFICOS

3.2 Funciones.

3.2.2 Módulo 2

3.2.2.1 Propósito del módulo

3.2.2.2 Requerimientos del módulo

3.2.2.2.1 Requerimiento funcional M21

3.2.2.2.1.1 Descripción

3.2.2.2.1.2 Entradas

3.2.2.2.1.3 Proceso

3.2.2.2.1.4 Salidas

3.2.2.2.2 Requerimiento funcional M22

3.2.2.2.2.1 Descripción

3.2.2.2.2.2 Entradas

3.2.2.2.2.3 Proceso

3.2.2.2.2.4 Salidas

6.4.3. REQUISITOS ESPECÍFICOS

3.2 Funciones.

3.2.m Módulo m

3.2.m.1 Propósito del módulo

3.2.m.2 Requerimientos del módulo

3.2.m.2.1 Requerimiento funcional Mm1

3.2.m.2.1.1 Descripción

3.2.m.2.1.2 Entradas

3.2.m.2.1.3 Proceso

3.2.m.2.1.4 Salidas

3.2.m.2.2 Requerimiento funcional Mm2

3.2.m.2.2.1 Descripción

3.2.m.2.2.2 Entradas

3.2.m.2.1.3 Proceso

3.2.m.2.1.4 Salidas

6.4.3. REQUISITOS ESPECÍFICOS

3.3 Requisitos de rendimiento

Aquí se deben especificar los requerimientos numéricos tanto estáticos como dinámicos del rendimiento del software y de la interacción de los usuarios con el software.

Los requerimientos numéricos estáticos pueden incluir:

- a) El número de terminales a ser soportadas
- b) El número de usuarios simultáneos a ser soportados
- c) La cantidad y el tipo de información a ser manejada

6.4.3. REQUISITOS ESPECÍFICOS

3.3 Requisitos de rendimiento

Los requerimientos numéricos dinámicos (aquellos dependientes del tiempo) pueden incluir el número de transacciones, número de tareas y cantidad de datos a ser procesados dentro de ciertos periodos de tiempo, tanto en condiciones normales como en condiciones de pico de trabajo.

Todos estos requerimientos deberían establecerse en términos medibles, por ejemplo:

- Throughput: como número de transacciones por segundo
- Modo degradado: modo aceptable de operación cuando el sistema ha sido degradado en cierta manera.

6.4.3. REQUISITOS ESPECÍFICOS

3.4 Restricciones de diseño

3.4.1 Aquí se deben especificar las restricciones de diseño impuestas por otros estándares, limitaciones de hardware, etc., así como los requerimientos de bases de datos.

3.4.2 También se deben definir los requerimientos derivados de los estándares o regulaciones, incluyendo:

- a) Formato de reportes
- b) Nombres de datos
- c) Procedimientos de control
- d) Pistas de auditoría

6.4.3. REQUISITOS ESPECÍFICOS

Por ejemplo, se podría especificar el requerimiento de rastrear las actividades de procesamiento.

Estos rastreos son necesarios para algunas aplicaciones que tienen que cumplir con regulaciones mínimas o estándares financieros.

Un requisito de pistas de auditoría puede, por ejemplo, establecer que todos los cambios a la base de datos de rol de pagos sean registrados en un archivo de pistas de auditoría, almacenando los valores anteriores y posteriores al cambio, así como el usuario que hizo el cambio.

6.4.3. REQUISITOS ESPECÍFICOS

3.4.3 Requerimientos de bases de datos

Aquí se deben especificar los requerimientos lógicos para cualquier información a ser almacenada en la base de datos, lo cual puede incluir:

- a) Tipos de información usados por varias funciones
- b) Frecuencia de uso
- c) Capacidades de acceso
- d) Entidades de datos y sus relaciones
- e) Restricciones de integridad
- f) Requerimientos de conservación de datos

6.4.3. REQUISITOS ESPECÍFICOS

3.5 Atributos de calidad

Aquí se definen los requerimientos no funcionales.

Se deben especificar cuantitativamente, con el fin de que puedan ser verificados de manera objetiva.

Se pueden especificar los siguientes atributos:

- Confiabilidad
- Mantenibilidad
- Disponibilidad
- Seguridad
- Usabilidad
- Portabilidad

6.4.3.1 REQUISITOS DE CONFIABILIDAD

Tiempo Medio Entre Fallas (MTBF): usualmente especificado en horas, pero podría también ser especificado en días, meses o años.

Tiempo Medio de Reparación (MTTR): tiempo permitido para que el sistema esté fuera de operación después de una falla.

Precisión: número de dígitos decimales permitidos

Exactitud: porcentaje máximo de variación permitido entre el valor obtenido y el valor correcto.

6.4.3.1 REQUISITOS DE CONFIABILIDAD

Número máximo de errores o tasa de defecto: expresada en errores/KLOC(kilo líneas de código) o errores/punto de función.

Errores o tasa de defecto: categorizada en términos de fallas menores, significativas y críticas. El o los requerimientos deben definir qué significa cada una. Por ejemplo, una pérdida completa de datos o una incapacidad de usar ciertas partes de la funcionalidad de un sistema, puede ser definida como una “falla crítica”.

6.4.3.2 REQUISITOS DE MANTENIBILIDAD

Son aquellos que facilitan el mantenimiento del producto software, como:

- Estándares de codificación
- Convenciones de nombres
- Bibliotecas de clases
- Acceso al mantenimiento
- Herramientas de mantenimiento

6.4.3.3 REQUISITOS DE DISPONIBILIDAD

La disponibilidad se puede definir en porcentaje o en periodos de tiempo:

- El sistema estará disponible 99,98% en el año
- El sistema estará disponible 24/7

También se pueden definir aquellos factores requeridos para garantizar un nivel definido de disponibilidad del sistema software, como:

- Chequeo
- Recuperación
- Reinicio

6.4.3.4 REQUISITOS DE SEGURIDAD

Deberían especificar los factores que protegen al software del acceso, uso, modificación, destrucción o difusión accidental o maliciosa de la información.

Los requerimientos específicos en esta área podrían incluir la necesidad de:

- a) Utilizar ciertas técnicas de criptografía
- b) Mantener una bitácora o conjuntos de datos históricos
- c) Asignar ciertas funciones a diferentes módulos
- d) Restringir las comunicaciones entre ciertas áreas del programa
- e) Chequear la integridad de datos para variables críticas
- f) Determinar las seguridades físicas

6.4.3.5 REQUISITOS DE USABILIDAD

Aquí deben establecerse todos los requerimientos que afectan a la usabilidad, como:

- Especificar el tiempo de entrenamiento requerido para que usuarios normales y usuarios avanzados se vuelvan productivos en el uso de ciertas operaciones del sistema. Ejemplo: se requiere un día para entrenar a un usuario avanzado en el uso del módulo de Administración del Sistema.
- Especificar tiempos de tareas medibles para tareas típicas. Por ejemplo: un usuario normal registrará los datos de un cliente en máximo 10 minutos.
- Especificar requerimientos para cumplir con estándares de usabilidad comunes, utilizados para las interfaces de usuario, como los estándares IBM CUA o Microsoft GUI.

6.4.3.6 REQUISITOS DE PORTABILIDAD

Permiten especificar los atributos del software relativos a facilitar su portabilidad a otras máquinas y/o sistemas operativos, lo cual puede incluir lo siguiente:

- a) Porcentaje de componentes con código dependiente de la máquina
- b) Porcentaje de código que es independiente de la máquina
- c) Uso de un lenguaje portable probado
- d) Uso de un compilador particular o subconjunto de un lenguaje
- e) Uso de un sistema operativo particular

6.4.4 APÉNDICES

4. Apéndices

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

Ejemplo:

Formatos de entrada/salida de datos, por pantalla o impresos

Estándares de programación

Formularios, planillas, facturas, etc.