



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

UNIVERSIDAD DE LAS FUERZAS ARMADAS ESPE

GRUPO #1

FECHA: 26/05/2024

NRC: 14870

TEMA: Tipos de Requisitos No Funcionales según Sommerville

Instrucciones: Investigar los Requisitos No Funcionales, definición y un ejemplo por tipo y subtipo.

De producto

1. Usabilidad

Es la facilidad con la que los usuarios pueden utilizar e interactuar con el sistema y cumplir sus objetivos.

Ejemplo:

En un sistema de gestión de contenido: *El sistema debe mostrar una barra de progreso visual y detallada durante la carga de contenido multimedia, indicando claramente el porcentaje completado y proporcionando estimaciones precisas del tiempo restante, con una precisión de al menos el 98%, para archivos de hasta 200 MB de tamaño.*

2. Eficiencia

La eficiencia es el rendimiento y el uso eficiente de los recursos del sistema.

Ejemplo:

El sistema debe ser capaz de admitir al menos 700 usuarios concurrentes realizando operaciones típicas, como la visualización de contenido, la carga de archivos y la interacción con funciones de la plataforma, sin experimentar una disminución del tiempo de respuesta promedio de más del 60%.

- **Performance:**

El performance es el tiempo de respuesta y la velocidad del sistema.

Ejemplo:

El tiempo total de carga de la página principal, desde que el usuario inicia la solicitud hasta que todos los recursos necesarios se han cargado completamente, no debe exceder los 4 segundos, Incluso con condiciones de alta demanda, como picos de tráfico de solicitudes y cargas pesadas de usuarios concurrentes

- **Espacio**

Este tipo de requerimiento es el uso eficiente del almacenamiento y de la memoria

Ejemplo:

El sistema debe asegurarse de que la cantidad total de espacio utilizado en disco para almacenar los datos del usuario, incluidos archivos, documentos y otros recursos relacionados, no supere los 100 megabytes (MB) en cualquier momento.

3. Confiabilidad

Se centran en la disponibilidad, integridad y capacidad de recuperación del sistema.

Ejemplo:

El sistema debe tener una disponibilidad del 90%, lo que significa que puede estar fuera de servicio por no más de 25 horas al año debido a mantenimiento o problemas técnicos.

4. Portabilidad

Se refieren a la capacidad del sistema para ser instalado y ejecutado en diferentes entornos.

Ejemplo:

El sistema debe ser compatible con los principales navegadores web , Chrome, Firefox, Safari y ser responsive, adaptándose a diferentes tamaños de pantalla en dispositivos móviles y de escritorio.

Organizacionales

Estos tipos se derivan de las políticas y procedimientos de una organización.

1. De entorno

Describen el ambiente operativo en el que se debe desenvolver el sistema.

Ejemplo:

Dentro de un sistema bancario: *El sistema debe ser capaz de funcionar de manera confiable en un entorno con una disponibilidad del servicio del 99.99% durante el horario de operación establecido, lo que permite un tiempo máximo de inactividad de no más de 5 minutos por mes para mantenimiento planificado*

2. Operacionales

Procedimientos operativos que describen cómo será usado el sistema dentro del contexto de la organización.

Ejemplo:

Dentro de un sistema bancario: *El sistema debe ser capaz de manejar simultáneamente al menos 1000 transacciones de usuarios únicos por hora durante el horario pico de operación, con un tiempo de respuesta promedio no superior a 2 segundos para cualquier solicitud.*

3. De desarrollo

El lenguaje de programación a usar, estándares de codificación, patrones de diseño y programación.

Ejemplo:

Dentro de un sistema bancario: *El sistema debe ser desarrollado utilizando el lenguaje de programación Java versión 11, siguiendo los estándares de codificación establecidos por la organización, que incluyen la adhesión estricta a las convenciones de codificación Java Code Style. Además, se debe mantener una cobertura de prueba del código del 90% o superior, medida mediante pruebas unitarias automatizadas.*

Externos

Los requisitos no funcionales externos son aquellos que se refieren a condiciones externas impuestas al sistema, a menudo derivadas de normas, regulaciones, o expectativas de la sociedad y del mercado.

1. Interoperabilidad

La interoperabilidad es la capacidad del sistema de trabajar y comunicarse con otros sistemas. Esto es vital en entornos donde diferentes sistemas deben interactuar para lograr un objetivo común.

Ejemplo:

- Sistema de Información de Hospital: Debe poder intercambiar datos con sistemas de laboratorio, sistemas de gestión de farmacia y registros nacionales de salud, utilizando estándares como HL7 para asegurar una comunicación eficiente y precisa.

2. Éticos

Los requisitos éticos aseguran que el sistema se desarrolle y opere de manera que cumpla con los principios éticos y morales aceptados. Estos pueden incluir la responsabilidad, la equidad y la protección de los derechos de los usuarios.

Ejemplo:

- Sistema de Admisión Universitaria: Debe asegurar la igualdad de oportunidades y no permitir sesgos en la admisión de estudiantes. Por ejemplo, el sistema debe procesar las solicitudes de manera anónima para evitar cualquier forma de discriminación.

3. Legales

Los requisitos legales garantizan que el sistema cumple con todas las leyes y regulaciones aplicables. Estos pueden abarcar diferentes áreas, tales como privacidad de datos, derechos de autor y regulaciones específicas del sector.

3.1 Privacidad

Los requisitos de privacidad aseguran que el sistema maneje los datos personales de acuerdo con las leyes y regulaciones de protección de datos, como el GDPR en la UE o la CCPA en California.

Ejemplo:

- Servicio de Almacenamiento en la Nube: Debe cumplir con las normativas de privacidad de datos, permitiendo a los usuarios controlar el acceso a su información, solicitar la eliminación de datos y asegurar que la transferencia de datos esté encriptada.

3.2 Seguridad

Los requisitos de seguridad aseguran que el sistema proteja la información y los recursos contra accesos no autorizados, ataques y otros riesgos de seguridad.

Ejemplo:

- Sistema de Pago Electrónico: Debe cumplir con los estándares PCI DSS, asegurando que la información de las tarjetas de crédito está encriptada, utilizando autenticación multifactor y realizando auditorías regulares para detectar y mitigar vulnerabilidades.

BIBLIOGRAFÍA

- Sommerville, I. (s. f.). *Ingeniería de Software*.
- PMOInformatica. (2015, abril 7). Requerimientos no funcionales: Una introducción. PMOInformática.
<https://www.pmoinformatica.com/2015/04/requerimientos-no-funcionales-una.html>
- Molina Hernández, Y., Granda Dihigo, A., & Velázquez Cintra, A. (2019). Los requisitos no funcionales del software. Una estrategia para su desarrollo en el Centro de Informática