**Interactividad: Características que se deben tener en cuenta al elegir un *software* de mantenimiento.**

Por favor, diseñar un esquema interactivo como el que se muestra a continuación, sobre las características que se deben tener en cuenta al elegir un *software* de mantenimiento.



Ref: [file:///Volumes/Multimedia\_1/Interactividades/Edge/6%20items/computador/Computador.html](file:///C:\Volumes\Multimedia_1\Interactividades\Edge\6%20items\computador\Computador.html)

Título: Características que se deben tener en cuenta al elegir un *software* de mantenimiento.

Instrucción al estudiante: Para conocer las características que se deben tener en cuenta al elegir un *software* de mantenimiento, puedes hacer clic en cada ítem.

Ítems: Funcionalidad

Confiabilidad

Usabilidad

Eficiencia

Mantenibilidad

Portabilidad

Información que se despliega:

1. **Funcionalidad**. Se describe como una serie de atributos que se apoyan en la existencia de un grupo de funciones y sus propiedades específicas. Las funciones son aquellas que satisfacen o afectan a un grupo de usuarios. Es la capacidad del *software* para cumplir y proveer las funciones para satisfacer las necesidades explícitas e implícitas, cuando es utilizado en funciones específicas.

Esta funcionalidad se subdivide en cinco criterios:

* Adecuación. Se refiere a la capacidad del *software* de cumplir con las tareas y objetivos especificados por el usuario.
* Exactitud. Es la capacidad del *software* para hacer los procesos y entregar los resultados esperados por el usuario.
* Interoperabilidad. Su capacidad para interactuar con otros sistemas específicos.
* Seguridad. La protección de la información y los datos, estableciendo los usuarios autorizados y los no autorizados.
* Conformidad de la funcionalidad. Capacidad del *software* para cumplir con los estándares referentes a la funcionalidad.

1. **Confiabilidad**. Se entiende como un conjunto de atributos que modifican el *software* para mantener su nivel de capacidad bajo condiciones específicas o por un periodo de tiempo establecido. Se refiere a la capacidad del *software* para asegurar un nivel de funcionamiento cuando se utiliza en condiciones concretas. Esta confiabilidad se divide en cuatro criterios:

* Madurez. Capacidad del *software* para evitar fallas cuando encuentra errores.
* Tolerancia. Su capacidad para mantenerse funcionando cuando encuentra los errores.
* Recuperabilidad. Capacidad para restablecer su funcionamiento adecuado y recuperar los datos afectados en caso de falla.
* Conformidad de la fiabilidad. Es la capacidad del *software* de cumplir con los estándares o normas relacionados a la fiabilidad.

1. **Usabilidad**. Se puede entender como el conjunto de cualidades que producen el esfuerzo necesario para el uso y esfuerzo individual para tal uso, para un conjunto de usuarios. Es la capacidad del *software* de ser entendido, aprendido y usado en forma fácil y atractiva. Comprende:

* Entendimiento. Permite al usuario entender si es adecuado y cómo utilizar las tareas y condiciones de aplicación. Debe considerarse la documentación y ayudas que suministra el *software*.
* Aprendizaje. Cómo permite al usuario aprender su uso.
* Operatividad. Permitir al usuario operarlo y controlarlo.
* Atracción. La presentación del *software* debe ser atractiva al usuario.
* Conformidad de uso. Es la capacidad del *software* de cumplir los estándares o normas relacionados a su usabilidad.

1. **Eficiencia**. Se define como el conjunto de cualidades que se apoyan en la relación entre el nivel de comportamiento del *software* y la cantidad de recursos usados, bajo las condiciones establecidas. Es la forma del desempeño adecuado, de acuerdo al número de recursos utilizados según las condiciones planteadas. Debe tenerse en cuenta la configuración del *hardware* y el sistema operativo. Comprende:

* Comportamiento de tiempos. Se refiere a los tiempos adecuados de respuesta y procesamiento.
* Utilización de recursos. La capacidad de utilizar cantidades y tipos adecuados de recursos tales como el recurso humano, el *hardware*, los dispositivos externos, etc.
* Conformidad de eficiencia. Es la capacidad que tiene el *software* de cumplir con los estándares relacionados con la eficiencia.

1. **Mantenibilidad.** Corresponde al conjunto de cualidades que se mantienen para hacer las modificaciones necesarias, incluyendo reformas o mejoras del *software*, cambios en el entorno y especificaciones de requerimientos funcionales.

Sus características son:

* Capacidad de ser analizado. Que permita hacer diagnóstico de deficiencias o causas de fallas, o la identificación de partes modificadas.
* Cambiabilidad. Poder realizar la implementación de una modificación.
* Estabilidad. Evita efectos inesperados para modificaciones del *software.*
* Facilidad de prueba. Permite realizar pruebas a las modificaciones sin poner en riesgo los datos.
* Conformidad de facilidad de mantenimiento. La capacidad que tiene el *software* para cumplir con los estándares de facilidad de mantenimiento.

1. **Portabilidad**. Conjunto de cualidades que soportan la capacidad del *software* para transferir el comportamiento de uno a otro. Hace referencia a la capacidad que tiene el *software* para ser trasladado de un entorno al otro. Se caracteriza por:

* Adaptabilidad. Que se pueda adaptar a diferentes entornos específicos sin implicar reacciones negativas ante el cambio, como en el caso del *hardware* o el sistema operativo.
* Facilidad de instalación. Facilidad para instalar en un entorno específico o por el usuario final.
* Coexistencia. Poder coexistir con varios *software* y compartir recursos comunes con otro *software* o dispositivo.
* Reemplazabilidad. Que pueda ser reemplazado por otro *software* del mismo tipo y para el mismo objetivo, como por ejemplo una nueva versión.