



# Calidad del Software

# Calidad del software:

- Es el grado en el que un conjunto de características inherentes cumple con los requisitos. (ISO, 2000a)
- Conjunto de propiedades o características de un producto o servicio que le confieren aptitud para satisfacer unas necesidades expresadas o implícitas.

# Calidad del software:

- Concordancia con los requisitos funcionales y de rendimiento explícitamente establecidos, con los estándares de desarrollo explícitamente documentados, y con las características implícitas que se espera de todo software de desarrollo profesionalmente.


# Componentes de la calidad

La calidad de un sistema informático puede descomponerse factores que contribuyen a la misma.

Wilkin y Castleman (2003) describen un instrumento multidimensional (denominado QUALIT) capaz de medir la calidad de los sistemas de información entregados y diferencia los siguientes componentes:

- **Calidad del sistema.**

Juicio global sobre el grado en que los componentes técnicos del mismo proporcionan la calidad de la información y servicio requerido por los *stakeholders*.

- 
- **Calidad de la información proporcionada a los *stakeholders***(usuarios del sistema)
  - **Calidad del servicio.**
    - Proporcionado por el departamento de SI y el personal de soporte

# Factores de calidad ISO 9126

- Funcionalidad
- Confiabilidad
- Usabilidad
- Eficiencia
- Facilidad de recibir Mantenimiento
- Portabilidad

## *Funcionalidad.*

- Capacidad del producto de SW para proporcionar funciones que satisfacen necesidades declaradas e implícitas cuando se usa bajo condiciones especificadas.

# Funcionalidad

- Esta característica se subdivide en:
- **Adecuación**  
Proporcionar un conjunto apropiado de funciones para tareas y objetivos de usuario específicos.
- **Exactitud**  
Proporcionar los resultados y efectos correctos o acordados, con el grado necesario de precisión.



# Funcionalidad

- **Interoperabilidad**

Interactuar con uno o más sistemas especificados.

- **Seguridad de Acceso**

Proteger información y datos de manera que las personas o sistemas no autorizados no puedan leerlos o modificarlos, al tiempo que no se deniega el acceso a las personas o sistemas autorizados.

- **Cumplimiento funcional**

Adherirse a normas, convenciones o regulaciones en leyes y prescripciones similares relacionadas con la funcionalidad

# ***Confiabilidad.***

- Cantidad del tiempo que el software se encuentra disponible para su uso según lo indican los siguientes atributos:

# Confiabilidad

- **Madurez**
- Evitar fallas como resultado de fallos en el SW.
- **Tolerancia a fallos**
- Mantener un nivel de prestaciones en caso de fallos de SW o de infringir sus interfaces especificados.
- **Capacidad de recuperación**
- Reestablecer un nivel de prestaciones especificado y de recuperar los datos directamente afectados en caso de fallo.
- **Cumplimiento de la fiabilidad.**
- Adherirse a normas, convenciones o regulaciones relacionadas con la fiabilidad.

# Usabilidad

- Capacidad del producto del SW para ser entendido, aprendido, usado y ser atractivo para el usuario, cuando se usa bajo condiciones especificadas.

# Usabilidad

- Esta característica se subdivide en:

- **Capacidad para ser entendido**

Permite al usuario entender si el SW es adecuado y cómo puede ser usado para unas tareas o condiciones de uso particulares.

- **Capacidad para ser aprendido**

Permite al usuario aprender sobre su aplicación.

- **Capacidad para ser operado**

Permite al usuario operarlo y controlarlo.

- **Capacidad de atracción**

- Capacidad del producto de software para ser atractivo al usuario.

- **Cumplimiento de la usabilidad**

Adherirse a normas, convenciones, guías de estilo o regulaciones relacionadas con la usabilidad.

# Eficiencia.

- Capacidad del producto de SW para proporcionar prestaciones apropiadas, relativas a la cantidad de recursos usados, bajo condiciones determinadas.

# Eficiencia.

- Esta característica se subdivide en:
- **Comportamiento temporal**
- Proporcionar tiempos de respuesta, tiempos de proceso y potencia apropiados, bajo condiciones determinadas.
- **Utilización de recursos**
- Usar las cantidades y tipos de recursos adecuados cuando el SW lleva a cabo su función bajo condiciones determinadas.
- **Cumplimiento de la eficiencia**
- Adherirse a normas o convenciones relacionadas con la eficiencia.

# *Mantenibilidad*

- Capacidad del producto de SW para ser modificado. Las modificaciones podrían incluir correcciones, mejoras o adaptación del SW a cambios en el entorno, y requisitos y especificaciones funcionales.



# *Mantenibilidad*

- Esta característica se subdivide en:
- **Capacidad para ser analizado**
- Capacidad para ser diagnosticadas deficiencias o causas de los fallos en el SW, o para identificar las partes que han de ser modificadas.
- **Capacidad para ser cambiado**
- Permite que una determinada sea implementada.
- **Estabilidad**
- Evitar efectos inesperados debidos a modificaciones de software.
- **Capacidad para ser probado**
- Permite que el SW modificado sea válido.
- **Cumplimiento de la mantenibilidad**
- Adherirse a normas o convenciones relacionadas con la mantenibilidad

# *Portabilidad.*

- Capacidad del producto de SW para ser transferido de un entorno a otro.

# *Portabilidad.*

- Esta característica se subdivide en:
- **Adaptabilidad**
- Adaptarse a diferentes entornos especificados, sin aplicar restricciones o mecanismos distintos de aquellos proporcionados para éste propósito por el propio SW considerado.
- **Instalabilidad**
- Capacidad para ser instalado en un entorno especificado.
- **Coexistencia**
- Para coexistir con otro SW independiente, en un entorno común, compartiendo recursos comunes.
- **Capacidad para ser reemplazado**
- Ser usado en lugar de otro producto de software, para el mismo propósito, en el mismo entorno.
- **Cumplimiento de la portabilidad**
- Adherirse a normas o convenciones relacionadas con la portabilidad.
-