

# Análisis Métricas Externas 9126-2 (ISO-IEC TR 9126-2 Software engineering - Product quality Part 2: External Metrics)

Identificación del documento: Guía de Calidad 3: GUIA-CALIDAD-00003

Autor: Integrantes de Comisión de Calidad CESSI (Grupo Producto).

Fecha: 08-10-2007

**Resumen:** El presente documento constituye un aporte de la Comisión de Calidad CESSI 2007. Su objetivo es analizar las métricas en lo referente a calidad de producto tomando como base la norma ISO/IEC 9126 Parte 2: External Metrics.

Las dos normas actualmente en proceso de estudio / traducción en IRAM:

ISO/ IEC 14598: Tecnología de la Información- Ingeniería de Software- Evaluación del producto de Software.

ISO / IEC 9126: Ingeniería de Software – Calidad del Producto

Es importante tener en cuenta que la ISO/IEC 9126 es mencionada en el contexto de la Ley de Promoción de la Industria del Software como norma reconocida en Calidad de Producto. . Asimismo mencionar que la 9126 no es certificable en sí misma, sino a través de otra norma. En este caso la norma certificable es la ISO/IEC 14598 (no mencionada en ley porque anteriormente ambas estaban unificadas en 9126) que utiliza como guía la norma ISO/IEC 9126 que define un modelo de Calidad de Software (u otro Modelo de Calidad)

**Palabras claves:** Plan; Métricas, Análisis; Calidad de Producto; Normas; IRAM; CESSI

## Contenido

1. REFERENCIAS / BIBLIOGRAFÍA.....	5
2. OBJETIVO .....	5
3. ESTRUCTURA DEL ANÁLISIS EFECTUADO SOBRE CADA MÉTRICA.....	5
8. ANÁLISIS MÉTRICAS EXTERNAS .....	6
8.1. FUNCIONALIDAD (FUNCTIONALITY).....	6
8.1.1. Adaptabilidad (Suitability). ....	6
8.1.1.1 Functional Adequacy. ....	6
8.1.1.2 Functional Implementation Completeness.....	6
8.1.1.3 Functional Implementation Coverage.....	7
8.1.1.4 Functional Specification Stability (Volatility). ....	7
8.1.2 Precisión (Accuracy). ....	8
8.1.2.1 Accuracy to expectation.....	8
8.1.2.2 Computational Accuracy. ....	8
8.1.2.3 Precision. ....	9
8.1.3 Inter-operación – Capacidad para interactuar (Interoperatbility). ....	9

8.1.3.1	Data Exchangeability (Data format based).....	9
8.1.3.2	Data Exchangeability (User's success attempt based).....	10
8.1.4	<i>Seguridad (Security)</i> .....	10
8.1.4.1	Access Auditability.....	10
8.1.4.2	Access Controllability.....	11
8.1.4.3	Data Corruption Prevention.....	11
8.1.5	<i>Compatibilidad en Funcionalidad (Functionality Compliance)</i> .....	12
8.1.5.1	Functional Compliance.....	12
8.1.5.2	Interface Standard Compliance.....	12
8.2	<b>CONFIABILIDAD (RELIABILITY)</b> .....	13
8.2.1	<i>Madurez (Maturity)</i> .....	13
8.2.1.1	Estimated latent fault density.....	13
8.2.1.2	Failure density against test cases.....	14
8.2.1.3	Failure Resolution.....	15
8.2.1.4	Fault Density.....	16
8.2.1.5	Fault Renoval.....	18
8.2.1.6	Mean time between failures.....	19
8.2.1.7	Test Coverage (Specified operation scenario testing coverage).....	20
8.2.1.8	Test Maturity.....	21
8.2.2	<i>Tolerancia a Fallas (Fault tolerance)</i> .....	21
8.2.2.1	Breakdown Avoidance.....	21
8.2.2.2	Failure Avoidance.....	22
8.2.2.3	Incorrect Operation Avoidance.....	22
8.2.3	<i>Recuperación – Capacidad de recuperación (Recoverability)</i> .....	23
8.2.3.1	Availability.....	23
8.2.3.2	Mean down time.....	23
8.2.3.3	Mean recovery time.....	24
8.2.3.4	Restartability.....	25
8.2.3.5	Restorability.....	26
8.2.3.6	Restore effectiveness.....	27
8.2.4	<i>Compatibilidad en Confiabilidad (Reliability Compliance)</i> .....	27
8.2.4.1	Reliability Compliance.....	27
8.3	<b>USABILIDAD (USABILITY)</b> .....	28
8.3.1	<i>Entendible (Understandability)</i> .....	28
8.3.1.1	Completeness of description.....	28
8.3.1.2	Demonstration accessibility.....	29
8.3.1.3	Demonstration accessibility in use.....	29
8.3.1.4	Demonstration effectiveness.....	30
8.3.1.5	Evident functions.....	30
8.3.1.6	Function understandability.....	31
8.3.1.7	Understandable input and output.....	31
8.3.2	<i>Aprendizaje – Capacidad para ser aprendido (Learnability)</i> .....	32
8.3.2.1	Ease of function learning.....	32
8.3.2.2	Ease of learning to perform a task in use.....	32
8.3.2.3	Effectiveness of the user documentation and/or help system.....	33
8.3.2.4	Effectiveness of the user documentation and/or help systems in use.....	33
8.3.2.5	Help accessibility.....	34
8.3.2.6	Help frequency.....	34
8.3.3	<i>Operación - Capacidad de Operación (Operability)</i> .....	35
8.3.3.1	<i>(A) Conforms with operational user expectations</i> .....	35
8.3.3.1.1	Operational consistency in use.....	35
8.3.3.2	<i>(B) Controlable</i> .....	35
8.3.3.2.1	Error correction.....	35

8.3.3.2.2	Error correction in use. ....	36
8.3.3.3	(C) <i>Suitable for the task operation</i> . ....	36
8.3.3.3.1	Default value availability in use. ....	36
8.3.3.4	(D) <i>Self descriptive (Guiding)</i> . ....	37
8.3.3.4.1	Message understandability in use. ....	37
8.3.3.4.2	Self – explanatory error messages. ....	37
8.3.3.5	(E) <i>Operational error tolerant (Human error free)</i> . ....	38
8.3.3.5.1	Operational error recoverability in use. ....	38
8.3.3.5.2	Time between human error operations in use. ....	38
8.3.3.5.3	Undoability (User error correction). ....	39
8.3.3.6	(F) <i>(Suitable for individualisation)</i> . ....	39
8.3.3.6.1	Customisability. ....	39
8.3.3.6.2	Operation procedure reduction. ....	40
8.3.3.6.3	Physical accessibility. ....	40
8.3.4	<i>Atractivo – Capacidad de ser atractivo (Attractiveness)</i> . ....	41
8.3.4.1	Attractive Interaction. ....	41
8.3.4.2	Interface appearance customisability. ....	41
8.3.5	<i>Compatibilidad en Usabilidad (Usability Compliance)</i> . ....	41
8.3.5.1	Usability Compliance. ....	41
8.4	EFICIENCIA (EFFICIENCY). ....	42
8.4.1	<i>Performance - Comportamiento en Tiempo (Time Behaviour)</i> . ....	42
8.4.1.1	(A) <i>Response Time</i> . ....	42
8.4.1.1.1	Response Time. ....	42
8.4.1.1.2	Response Time (Mean Time to response). ....	42
8.4.1.1.3	Response Time (Worst case response time ratio). ....	43
8.4.1.2	(B) <i>Throughput</i> . ....	43
8.4.1.2.1	Throughput. ....	43
8.4.1.2.2	Throughput (Mean amount of throughput). ....	43
8.4.1.2.3	Throughput (Worst case throughput ratio). ....	44
8.4.1.3	(C) <i>Turnaround Time</i> . ....	44
8.4.1.3.1	Turnaround time. ....	44
8.4.1.3.2	Turnaround time (Mean time for turnaround). ....	44
8.4.1.3.3	Turnaround time (Worst case turnaround time ratio). ....	44
8.4.1.3.4	Waiting time. ....	44
8.4.2	<i>Utilización de Recursos (Resource Utilisation)</i> . ....	45
8.4.2.1	(A) <i>I/O Devices resource utilization</i> . ....	45
8.4.2.1.1	I/O devices utilisation. ....	45
8.4.2.1.2	I/O loading limits. ....	45
8.4.2.1.3	I/O related errors. ....	46
8.4.2.1.4	Mean I/O fulfilment ratio. ....	46
8.4.2.1.5	User waiting time of I/O devices utilisation. ....	47
8.4.2.2	(B) <i>Memory resource utilization</i> . ....	47
8.4.2.2.1	Maximum memory utilisation. ....	47
8.4.2.2.2	Mean occurrence of memory error. ....	48
8.4.2.2.3	Ratio of memory error/time. ....	48
8.4.2.3	(C) <i>Transmission resource utilization</i> . ....	49
8.4.2.3.1	Maximum transmission utilisation. ....	49
8.4.2.3.2	Media device utilisation balancing. ....	49
8.4.2.3.3	Mean occurrence of transmission error. ....	50
8.4.2.3.4	Mean of transmission error per time. ....	51
8.4.2.3.5	Transmission capacity utilisation. ....	51
8.4.3	<i>Compatibilidad en Eficiencia (Efficiency Compliance)</i> . ....	52
8.4.3.1	Efficiency Compliance. ....	52

8.5	MANTENIMIENTO (MAINTAINABILITY).....	52
8.5.1	<i>Diagnóstico - Capacidad para ser analizado / diagnosticado (Analysability)</i> .....	52
8.5.1.1.	Audit trail capability.....	55
8.5.1.2	Diagnostic function support.....	55
8.5.1.3	Failure analysis capability.....	56
8.5.1.4	Failure analysis efficiency.....	57
8.5.1.5	Status monitoring capability.....	58
8.5.2	<i>Cambios - Capacidad de ser cambiado / modificado (Changeability)</i> .....	58
8.5.2.1	Change cycle efficiency.....	61
8.5.2.2	Change implementation elapse time.....	62
8.5.2.3	Modification complexity.....	62
8.5.2.4	Parametrised modifiability.....	63
8.5.2.5	Software change control capability.....	64
8.5.3	<i>Estabilidad (Stability)</i> .....	65
8.5.3.1	Change success ratio.....	68
8.5.3.2	Modification impact localisation (Emerging failure after change).....	68
8.5.4	<i>Testing – Capacidad de ser testeado (Testability)</i> .....	70
8.5.4.1	Availability of built-in test function.....	72
8.5.4.2	Re-test efficiency.....	72
8.5.4.3	Test restartability.....	72
8.5.5	<i>Compatibilidad en Mantenimiento (Maintainability Compliance)</i> .....	74
8.5.5.1	Maintainability compliance.....	76
8.6	PORTABILIDAD (PORTABILITY).....	76
8.6.1	<i>Adaptabilidad (Adaptability)</i> .....	76
8.6.1.1	Adaptability of data structures.....	76
8.6.1.2	Hardware environmental adaptability.....	77
8.6.1.3	Organisational environment adaptability (Organisation adaptability to infrastructure of organisation).....	79
8.6.1.4	Porting user friendliness.....	80
8.6.1.5	System software environmental adaptability.....	81
8.6.2	<i>Instalación – Capacidad para ser instalado (Installability)</i> .....	82
8.6.2.1	Ease of installation.....	82
8.6.2.2	Ease of Setup retry.....	83
8.6.3	<i>Coexistencia – Capacidad para coexistir (Co-existence)</i> .....	84
8.6.3.1	Available co-existence.....	84
8.6.4	<i>Upgrades / Reemplazo – Capacidad para reemplazar (Replaceability)</i> .....	85
8.6.4.1	Continued use of data.....	85
8.6.4.2	Function inclusiveness.....	85
8.6.4.3	User support functional consistency.....	86
8.6.5	<i>Compatibilidad en Portabilidad (Portability Compliance)</i> .....	87
8.6.5.1	Portability compliance.....	87

Histórico del documento		
Versión	Estado / cambios	Fecha / Departamento/ Autor
1.0	Versión inicial.	8/10/2007/ Miembros CESSI Comisión de Calidad (Grupo Producto).
2.0	Consolidación e Integración del análisis realizado por los diferentes grupos.	27/11/2007/ Miembros CESSI Comisión de Calidad (Grupo Producto).
2.1	Actualización.	05/12/2007/ Miembros CESSI Comisión de Calidad (Grupo Producto).

## 1. Referencias / Bibliografía

Referencia	Descripción / Comentario
ISO/ IEC 14598	<p>“Tecnología de la Información- Ingeniería de Software- Evaluación del producto de Software”</p> <p>Consta de las siguientes partes:</p> <ul style="list-style-type: none"> <li>14598-1: Descripción General</li> <li>14598-2: Planificación y Gestión</li> <li>14598-3: Proceso para Desarrolladores (mejora)</li> <li>14598-4: Proceso para Compradores (adquisición)</li> <li>14598-5: Proceso para evaluadores (evaluación independiente, de terceras partes)</li> <li>14598-6: Documentación de los módulos de evaluación</li> </ul>
ISO/IEC 9126	<p>ISO 9126: Ingeniería de Software – Calidad de Producto</p> <p>Consta de las siguientes partes:</p> <ul style="list-style-type: none"> <li>ISO 9126-1: Modelo de Calidad. Características y subcaracterísticas asociadas de calidad.</li> <li><b>ISO 9126-2: Métricas externas</b></li> <li>ISO 9126-3: Métricas internas</li> <li>ISO 9126-4: Métricas de calidad en uso</li> </ul>

## 2. Objetivo

El presente documento constituye un aporte de la Comisión de Calidad CESSI 2007. Su objetivo es analizar las métricas en lo referente a calidad de producto tomando como base la norma ISO/IEC 9126 Parte 2: External Metrics.

ISO/ IEC 14598: Tecnología de la Información- Ingeniería de Software- Evaluación del producto de Software  
 ISO / IEC 9126: Ingeniería de Software – Calidad del Producto  
 ISO / IEC TR 9126-2: Ingeniería de Software – Calidad del Producto – Parte 2, Métricas Externas.

## 3. Estructura del Análisis efectuado sobre cada métrica.

En caso de que la misma no aplique mencionar **N/A**, con su correspondiente justificación.

### **IMPORTANCIA GENERAL / ESCENARIOS.**

<Se deberá clasificar la misma (salvo excepciones, casos particulares) en Alto, Medio, Bajo; es decir priorizar la métrica.

Luego los escenarios corresponden a:

**C:** Compra o adquisición de un producto.

**E:** COTS o Enlatado.

**D:** Desarrollo a medida.>

### **VALIDACIONES.**

<Se deberá validar la formula de medición, si la misma es aplicable o resulta obsoleta. Bajo este ultimo caso proponer formulas alternativas, etc>

### **PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

<Se deberán describir la precondiciones necesarias, por ejemplo contar con la documentación actualizada, con un histórico de información; sugerencias por ejemplo que la empresa provea el lote de test cases ejecutados y que el evaluador valide e inspeccione una muestra (si la métrica es muy extensa).>

**EJEMPLOS.**

<Se deberá detallar ejemplos de aplicación que entendemos se estaría reflejando la métrica>

**GLOSARIO.**

<Especificar un glosario para dicha métrica, definición de términos en caso de temas no claros.>

## 8. Análisis Métricas Externas

### 8.1. Funcionalidad (Functionality)

#### 8.1.1. Adaptabilidad (Suitability).

##### 8.1.1.1 Functional Adequacy.

Adecuación Funcional.

###### 8.1.1.1.1 IMPORTANCIA GENERAL / ESCENARIOS.

Alta (CED).

###### 8.1.1.1.2 VALIDACIONES.

Correcta.

**Observaciones:**

¿Cuán adecuadas son las funciones evaluadas?

Valor esperado = 1

Se sugiere utilizarlo según estándar.

###### 8.1.1.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.

- ☐ Manual del usuario
- ☐ Especificación de requerimientos
- ☐ Software
- ☐ Lista de defectos conocidos

###### 8.1.1.1.4 EJEMPLOS.

La función de Alta de Empleado tiene 3 subfunciones, 2 están correctas y una no  $X=1-A/B$   
 $X=1-(1/3)$   $X=66\%$ .

###### 8.1.1.1.5 GLOSARIO.

#### 8.1.1.2 Functional Implementation Completeness.

Compleitud de la implementación funcional.

###### 8.1.1.2.1 IMPORTANCIA GENERAL / ESCENARIOS.

Alta (CED).

###### 8.1.1.2.2 VALIDACIONES.

Correcta.

**Observaciones:**

¿Cuán completa es la implementación de acuerdo a la especificación de requerimientos?

Se sugiere utilizarlo según estándar.

**8.1.1.2.3 PRECONDICIONES / DOCUMENTACION DE RESPALDO.**

- ☐ Manual del usuario
- ☐ Especificación de requerimientos
- ☐ Software
- ☐ Lista de defectos conocidos

**8.1.1.2.4 EJEMPLOS.**

X=1-A/B

**8.1.1.2.5 GLOSARIO.**

**8.1.1.3 Functional Implementation Coverage.**

Cobertura de la implementación funcional.

**8.1.1.3.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Baja (CED).

**8.1.1.3.2 VALIDACIONES.**

Correcta.

**Observaciones:**

Haga el test funcional (Test de caja negra) de acuerdo a la especificación de requerimientos. Cuente el número de funciones incorrectamente implementadas o funciones que faltan implementar versus el total de funciones descriptas en la especificación de requerimientos. Cuente el número de funciones que están completas contra las que no lo están.  
Se sugiere utilizarlo según estándar.

**8.1.1.3.3 PRECONDICIONES / DOCUMENTACION DE RESPALDO.**

Especificación de requerimientos de Software.

**8.1.1.3.4 EJEMPLOS.**

Suma de las validaciones anteriores

**8.1.1.3.5 GLOSARIO.**

**8.1.1.4 Functional Specification Stability (Volatility).**

Estabilidad de la implementación funcional.

**8.1.1.4.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Media (CED).

**8.1.1.4.2 VALIDACIONES.**

Correcta.

**Observaciones:**

Cuente el número de funciones descripta en la especificación funcional que han sido cambiadas luego de ser pasado el producto a Producción – Está en operación. Compare éste número con el total de funciones descriptas en la especificación de requerimientos.

Encontramos: poco práctica su obtención

Poco comprobable su veracidad.

Consideramos que hay que tener en cuenta la tipificación del cambio al cual corresponda:

**8.1.1.4.3 PRECONDICIONES / DOCUMENTACION DE RESPALDO.**

- ☐ Log de cambios al producto en producción por función
- ☐ Manual del usuario
- ☐ Especificación de requerimientos
- ☐ Software
- ☐ Lista de defectos conocidos

**8.1.1.4.4 EJEMPLOS.**

Ejemplos:

- 1) Cambios ocasionado por un Análisis incompleto – Inestabilidad funcional.
- 2) Cambios ocasionados por una mala implementación – Inestabilidad de la implementación.
- 3) Cambio ocasionado por cambios en el Negocio o cambio de la necesidad de Usuario.  
No aplica para éste indicador.

**8.1.1.4.5 GLOSARIO.**

**8.1.2 Precisión (Accuracy).**

**8.1.2.1 Accuracy to expectation.**

Precisión / Cercanía a las expectativas.

**8.1.2.1.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta (CED).

**8.1.2.1.1 VALIDACIONES.**

Correcta.

**Observaciones:**

Cantidad de desconformidades respecto a los casos de prueba en el tiempo.

Consideramos:

Expectativa de la persona que pidió la prueba, informando que porcentajes de casos deberían ser exitosos.

**8.1.2.1.2 PRECONDICIONES / DOCUMENTACION DE RESPALDO.**

- ☐ Casos de prueba
- ☐ Resultado de la prueba
- ☐ Expectativa del requirente de las pruebas.

**8.1.2.1.3 EJEMPLOS.**

Cantidad total de desconformidades según las expectativas.

**8.1.2.1.4 GLOSARIO.**

**8.1.2.2 Computational Accuracy.**

Precisión computacional.

**8.1.2.2.1 IMPORTANCIA GENERAL / ESCENARIOS.**



Alta (CED).

**8.1.2.2.2 VALIDACIONES.**

Correcta.

**Observaciones:**

Cantidad de disconformidades computacionales respecto a la especificación en el tiempo  
Se sugiere utilizar según estándar.

**8.1.2.2.3 PRECONDICIONES / DOCUMENTACION DE RESPALDO.**

- ☐ Especificación
- ☐ Casos de prueba
- ☐ Resultado de la prueba

**8.1.2.2.4 EJEMPLOS.**

Casos de pruebas fallidos según los esperados.

**8.1.2.2.5 GLOSARIO.**

**8.1.2.3 Precision.**

Precisión.

**8.1.2.3.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta (CED).

**8.1.2.3.2 VALIDACIONES.**

Correcta.

**Observaciones:**

¿Cuán frecuente el Usuario final encuentra resultados con precisión inadecuada?  
Se sugiere utilizar según estándar.

**8.1.2.3.3 PRECONDICIONES / DOCUMENTACION DE RESPALDO.**

- ☐ Casos de prueba
- ☐ Resultado de la prueba

**8.1.2.3.4 EJEMPLOS.**

De los casos de prueba que fallaron, cuántos se deben a errores en la precisión de datos.

**8.1.2.3.5 GLOSARIO.**

**8.1.3 Inter-operación – Capacidad para interactuar (Interoperability).**

**8.1.3.1 Data Exchangeability (Data format based).**

Interoperabilidad / Intercambio de Datos.

**8.1.3.1.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta (CED).

**8.1.3.1.2 VALIDACIONES.**

Correcta.

**Observaciones:**

¿Cuán correctamente la función de intercambio de datos ha sido implementada, según la especificación de intercambio de datos?

Ejecución de los casos de prueba y medir el resultado.

Se sugiere utilizar según estándar.

**8.1.3.1.3 PRECONDICIONES / DOCUMENTACION DE RESPALDO.**

- ☐ Especificación
- ☐ Casos de prueba
- ☐ Resultado de la prueba

**8.1.3.1.4 EJEMPLOS.**

**8.1.3.1.5 GLOSARIO.**

**8.1.3.2 Data Exchangeability (User's success attempt based).**

Interoperabilidad / Intercambio de Datos exitosos del usuario

**8.1.3.2.1 IMPORTANCIA GENERAL / ESCENARIOS..**

Alta (CD).

**8.1.3.2.2 VALIDACIONES.**

Correcta.

**Observaciones:**

¿Cuán frecuente el Usuario final encuentra fallas en el intercambio de datos entre software origen y el otro software?

Consideramos que esta información no es sencilla de obtener. Pero sugerimos utilizarla según estándares.

**8.1.3.2.3 PRECONDICIONES / DOCUMENTACION DE RESPALDO.**

- ☐ Especificación
- ☐ Casos de prueba
- ☐ Resultado de la prueba
- ☐ Lista de defectos reportados por Usuario.

**8.1.3.2.4 EJEMPLOS.**

**8.1.3.2.5 GLOSARIO.**

**8.1.4 Seguridad (Security).**

**8.1.4.1 Access Auditability.**

Auditoría de accesos.

**8.1.4.1.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta (CED).

**8.1.4.1.2 VALIDACIONES.**

Correcta.

**Observaciones:**

¿Cuán completo es el rastro de auditoría concerniente al acceso de Usuarios al sistema y a los datos?

Debe evaluarse el tipo de sistema para determinar que auditoria es representativa para el producto. Se sugiere utilizar según estándar.

**8.1.4.1.3 PRECONDICIONES / DOCUMENTACION DE RESPALDO.**

- ☐ Especificación de seguridad
- ☐ Casos de prueba
- ☐ Resultado de la prueba
- ☐ Log de accesos al sistema.

**8.1.4.1.4 EJEMPLOS.**

**8.1.4.1.5 GLOSARIO.**

**8.1.4.2 Access Controllability.**

Control de permisos de acceso.

**8.1.4.2.1 IMPORTANCIA GENERAL/ ESCENARIOS.**

Alta (CED).

**8.1.4.2.2 VALIDACIONES.**

Correcta.

**Observaciones:**

¿Cuán controlable es el acceso al Sistema?

Control de perfiles de usuario y accesos .

Se sugiere utilizar según estándar.

**8.1.4.2.3 PRECONDICIONES / DOCUMENTACION DE RESPALDO.**

- ☐ Especificación de seguridad
- ☐ Casos de prueba
- ☐ Resultado de la prueba
- ☐ Log de accesos al sistema

**8.1.4.2.4 EJEMPLOS.**

**8.1.4.2.5 GLOSARIO.**

**8.1.4.3 Data Corruption Prevention.**

Seguridad / Prevención de corrupción de datos.

**8.1.4.3.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta (CED).

**8.1.4.3.2 VALIDACIONES.**

Correcta.

**Observaciones:**

¿Cuál es la frecuencia de los eventos de corrupción de datos?

Contempla distintos tipos de corrupción de datos, mayores y menores, relacionados con la recuperación de la misma, segundos afectados y la importancia de los datos.

Mayor:

-Recuperación y reproducción son imposible.

-Los datos son importantes.

-Afecta la distribución de segundos.

Menor:

-Recuperación y reproducción son posibles.

-No afecta la distribución.

-Poco importantes los datos.

Se sugiere utilizar según estándar.

**8.1.4.3.3 PRECONDICIONES / DOCUMENTACION DE RESPALDO.**

- ☐ Especificación
- ☐ Casos de prueba
- ☐ Resultado de la prueba
- ☐ Pedido de cambio de datos en Producción.
- ☐ Pasos Para el recupero de información

**8.1.4.3.4 EJEMPLOS.**

**8.1.4.3.5 GLOSARIO.**

**8.1.5 Compatibilidad en Funcionalidad (Functionality Compliance).**

**8.1.5.1 Functional Compliance.**

Adhesión a estándares / Adhesión funcional.

**8.1.5.1.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta (CED).

**8.1.5.1.2 VALIDACIONES.**

Correcta.

**Observaciones:**

¿Cuán de acuerdo está la funcionalidad del producto con las regulaciones, estándares o convenciones?

Se Sugiere utilizar según estándar.

**8.1.5.1.3 PRECONDICIONES / DOCUMENTACION DE RESPALDO.**

- ☐ Especificación de los estándares a adherir
- ☐ Casos de prueba
- ☐ Resultado de la prueba

**8.1.5.1.4 EJEMPLOS.**

Cálculo de ingresos brutos (IIBB)

**8.1.5.1.5 GLOSARIO.**

**8.1.5.2 Interface Standard Compliance.**

Adhesión a estándares / Adhesión estándares de interface.

**8.1.5.2.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta (CED).

**8.1.5.2.2 VALIDACIONES.**

Correcta.

**Observaciones:**

¿Cuán de acuerdo están construidas las interfaces respecto a las regulaciones, estándares o convenciones?

Se Sugiere utilizar según estándar.

**8.1.5.2.3 PRECONDICIONES / DOCUMENTACION DE RESPALDO.**

- ☐ Especificación de los estándares a adherir
- ☐ Casos de prueba
- ☐ Resultado de la prueba

**8.1.5.2.4 EJEMPLOS.**

Cálculo de dígito verificador (DV) de CBU

**8.1.5.2.5 GLOSARIO.**

## **8.2 Confiabilidad (Reliability).**

### **8.2.1 Madurez (Maturity).**

#### **8.2.1.1 Estimated latent fault density.**

Especifica problemas existentes que pueden aparecer como fallas futuras

##### **8.2.1.1.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alto (CED)

##### **8.2.1.1.2 VALIDACIONES.**

La fórmula de esta métrica es válida y aplicable.

**Método de Aplicación:** Contar el número de fallas detectadas durante un periodo de prueba definido y predecir el potencial numero de futuras fallas usando un modelo de estimación de crecimiento.

**Formula:**  $X = \{ABS(A1 - A2)\} / B$

Donde

X = densidad residual estimada de fallas latentes

ABS()= Valor absoluto

A1 = Número total de fallas latentes predichas en un producto de software

A2 = Número total de fallas actualmente detectadas

B= tamaño del producto

**Interpretación de la métrica:**  $X \geq 0$

Depende de la etapa de la prueba. En las etapas finales, se tiende a que el resultado sea menor.

**Escala de Métrica:** Absoluta

**Tipo Medición:**

A1 = contar

A2 = contar  
B = tamaño  
X = contar/tamaño

**Notas:**

1. Cuando en número de fallas actualmente detectadas se torna mayor que el número total de fallas latentes predichas, se recomienda realizar una nueva predicción y estimar un número mayor. Los numero estimados mayores son para intentar predecir fallas latentes razonables, pero no para hacer que el producto se vea mejor.
2. Se recomienda usar varios modelos de estimación y elegir el mas apropiado y repetir la predicción con monitoreo de fallas detectadas.
3. Puede ser de ayuda predecir un valor mínimo y máximo de fallas latentes.

**8.2.1.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Contar con la documentación necesaria y actualizada.

Contar con reporte de errores y fallas.

Suministrar documentación de respaldo para medir el tamaño del software (casos de uso, complejidad de casos de uso, puntos de función)

Suministrar documentación de respaldo para medir la futura ocurrencia de errores (análisis de tendencia de errores, problemas recurrentes, errores ocurridos)

**8.2.1.1.4 EJEMPLOS.**

Definir un período de tiempo para tomar la métrica.

En dicho tiempo, recolectar las fallas del sistema (A2).

La complejidad del software (tamaño del Software (B)), se puede utilizar puntos de función o complejidad por caso de uso.

Para predecir el potencial número de futuras fallas (A1) se puede utilizar un método que utilice un modelo de estimación de crecimiento.

**8.2.1.1.5 GLOSARIO.**

Falla = Incidencia = Error

**8.2.1.2 Failure density against test cases.**

Especifica cuántas fallas fueron detectadas en un periodo de tiempo de prueba

**8.2.1.2.1 IMPORTANCIA GENERAL / ESCENARIOS.**

(CED) Alto

**8.2.1.2.2 VALIDACIONES.**

La fórmula de esta métrica es válida y aplicable.

**Método de Aplicación:** Contar el numero de fallas detectadas y los casos de pruebas realizados

**Formula:**  $X = A1 / A2$

Donde

A1 = Numero de fallas detectadas

A2 = Numero de casos de pruebas realizados

**Interpretación de la métrica:**  $X \geq 0$

Depende de la etapa de la prueba. En las etapas finales, cuanto menor es, mejor.

**Escala de Métrica: Absoluta**

**Tipo Medición:**

A1 = contar

A2 = contar

X = contar/contar

**Notas:**

1. Un resultado mayor es mejor en las primeras etapas de las pruebas. Por el contrario, un resultado menor es lo mejor en las ultimas etapas de pruebas u operaciones. Se recomienda monitorear la tendencia de esta medida a lo largo del tiempo.
2. Esta métrica depende mucho de casos de pruebas adecuados, que deben ser diseñados para incluir el caso apropiado. Ejemplo: casos normal, excepcional y anormales.

**8.2.1.2.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Contar con la documentación necesaria y actualizada.

Suministrar el lote adecuado de casos de prueba según los requerimientos.

Reporte de Errores detectados en las pruebas.

Documentación de respaldo (especificación de requerimientos, especificación de la prueba e informe de la prueba)

**8.2.1.2.4 EJEMPLOS.**

Definir un período de tiempo para tomar la métrica.

Armar el lote de casos de prueba según los requerimientos, utilizando la documentación con la que se cuente en el proyecto. Contar la cantidad de casos de prueba en el lote a ejecutar (A2).

Ejecutar el lote de prueba y reportar los errores encontradas durante la ejecución (A1).

**8.2.1.2.5 GLOSARIO.**

Falla = Incidencia = Error

**8.2.1.3 Failure Resolution**

Solución de Fallas.

**8.2.1.3.1 IMPORTANCIA GENERAL / ESCENARIOS.**

(CED) Alto

**8.2.1.3.2 VALIDACIONES.**

La fórmula de esta métrica es válida y aplicable.

**Método de Aplicación:** Contar el numero de fallas que no volvieron a ocurrir durante un periodo de prueba bajo condiciones similares.

Mantener un reporte de resolución de fallas describiendo el estado de todas ellas.

**Formula:**  $X = A1 / A2$

Donde

A1 = Numero de fallas resueltas

A2 = Numero total de fallas detectadas

**Interpretación de la métrica:**  $0 \leq X \leq 1$

Cuanto más cerca de 1.0 mejor, dado que mas fallas fueron resueltas.

**Escala de Métrica:** Absoluta

**Tipo Medición:**

A1 = contar

A2 = contar

X = contar/contar

**Notas:**

1. Se recomienda monitorear la tendencia cuando se usa esta métrica.
2. El número total de fallas latentes predichas puede ser estimado usando un modelo de crecimiento ajustado con el historial actual y con un producto de software similar. En ese caso, el número de fallas actuales y predichas puede ser comparado y el número de fallas residuales no resueltas puede ser medido.
3. Cuando se cuentan fallas, prestar atención en:
  - a. La no duplicación de errores, porque múltiples reportes pueden contener la misma información de fallas que otros.
  - b. Posibilidad de otros errores que no sean fallas, dado que los usuarios o testers no darse cuenta de si son problemas de operación, ambiente o error de software.

#### **8.2.1.3.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Contar con la documentación necesaria y actualizada.

Contar con un Reporte de fallas detectadas, describiendo el estado actualizado de las mismas.

Contar con un Reporte de fallas resueltas.

Documentación de respaldo (informe de la prueba, informe de resolución de incidencias).

#### **8.2.1.3.4 EJEMPLOS.**

Definir un tiempo para la recolección de la métrica.

Contar con un registro de las fallas detectadas y el historial de resolución de las mismas, pudiendo medir el número de fallas resueltas (A1) y el número de fallas detectadas (A2).

Por ejemplo, fallas en proceso de resolución, fallas pendientes, fallas resueltas. En este caso se tomarán las fallas que se encuentren en estado resuelto. Este punto dependerá de los diferentes estados que se definan previamente, pero se debería tener en cuenta que el estado Resuelto cuenta con la aceptación del Usuario que generó la falla.

Se podría medir también la capacidad estimada de resolución, para tal medida se puede tomar el número de fallas actuales y reemplazarla por el número de fallas predichas para poder medir la capacidad de resolución según fallas predichas.

#### **8.2.1.3.5 GLOSARIO.**

Falla = Incidencia = Error

#### **8.2.1.4 Fault Density.**

Cantidad de fallas detectadas en un periodo de pruebas definido

##### **8.2.1.4.1 IMPORTANCIA GENERAL / ESCENARIOS.**



Alto (CED)

**8.2.1.4.2 VALIDACIONES.**

La fórmula de esta métrica es válida y aplicable.

**Método de Aplicación:** Contar el numero de fallas detectadas y calcular la densidad.

**Formula:**  $X = A / B$

Donde

A = Número de fallas detectadas

B = Tamaño de producto

**Interpretación de la métrica:**  $X \geq 0$

Depende de la etapa de la prueba. En las etapas finales, se espera que el resultado tienda a ser cada vez menor.

**Escala de Métrica:** Absoluta

**Tipo Medición:**

A = contar

B = contar

X = contar/tamaño

**Notas:**

1. Un resultado mayor es mejor, en las primeras etapas de las pruebas. Por el contrario, un resultado menor es mejor en las últimas etapas de pruebas u operaciones. Se recomienda monitorear la tendencia de esta medida a lo largo del tiempo.
2. El número de fallas detectadas dividido el número de casos de pruebas indica la efectividad de los casos de pruebas.
3. Cuando se cuentan fallas, prestar atención en:
  - a. La no duplicación, porque múltiples reportes pueden contener la misma información de fallas que otros.
  - b. Posibilidad de otros errores que no sean fallas, dado que los usuarios o testers no darse cuenta de si son problemas de operación, ambiente o error de software.

**8.2.1.4.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Contar con la documentación necesaria y actualizada.

Suministrar el lote de casos de prueba adecuado.

Contar con reporte de errores y fallas.

Suministrar documentación de respaldo para medir el tamaño del software (casos de uso, complejidad de casos de uso, puntos de función)

Suministrar documentación de respaldo para generar el lote de pruebas (especificación de requerimientos, casos de uso, documentación funcional)

**8.2.1.4.4 EJEMPLOS**

Definir un tiempo para la recolección de la métrica.

Armar el lote de casos de prueba según los requerimientos, utilizando la documentación con la que se cuente en el proyecto.

Ejecutar el lote de prueba y reportar los errores encontradas durante la ejecución (A1).

Utilizar el tamaño del producto (B) para medir la densidad, ya que la cantidad de errores detectados puede depender de la medida del Software. Para medir el tamaño del software se puede utilizar puntos de función o complejidad por casos de uso.

#### 8.2.1.4.5 GLOSARIO

Falla = Incidencia = Error

#### 8.2.1.5 Fault Renoval

Cuantas fallas fueron corregidas.

##### 8.2.1.5.1 IMPORTANCIA GENERAL / ESCENARIOS.

Medio (CED)

##### 8.2.1.5.2 VALIDACIONES.

La fórmula de esta métrica es válida y aplicable.

**Método de Aplicación:** Contar el numero de fallas removidas durante las pruebas y comparar con el total de fallas detectados y el total de fallas predichas

**Formula:**  $X = A1 / A2$   
 $Y = A1 / A3$

Donde

A1 = Número de fallas resueltas

A2 = Número total de fallas detectadas

A3 = Número total de fallas latentes predichas en el producto de software.

**Interpretación de la métrica:**  $0 \leq X \leq 1$

$Y \geq 0$  Cuanto más cerca de 1.0 mejor, dado que mas fallas fueron resueltas.

**Escala de Métrica:** Absoluta

**Tipo Medición:**

A1 = contar

A2 = contar

A3 = contar

X = contar/contar

Y = contar/contar

**Entrada a la medición:** Reporte prueba, base de datos de la organización.

**Notas:**

1. Se recomienda monitorear la tendencia durante un periodo de tiempo definido.
2. El número total de fallas predichas latentes puede ser estimado usando modelos de crecimiento ajustados al actual histórico relacionado con producto similar de software.
3. Se recomienda monitorear el ratio de resolución de fallas estimadas Y, para que si  $Y > 1$ , se investigue la razón ya sea porque mas fallas fueron detectadas antes o porque el producto de software contiene un numero inusual de fallas. Caso contrario,  $Y < 1$ , investigar si es porque existen menos que el numero usual de defectos en los productos de software o porque el testing no fue adecuado para detectar todas las posibles fallas.
4. Cuando se cuentan fallas preste atención a la posibilidad de duplicación, dado que múltiples reportes pueden contener las mismas fallas que otro reporte.

##### 8.2.1.5.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.

Contar con la documentación necesaria y actualizada.

Contar con un Reporte de fallas detectadas, describiendo el estado actualizado de las mismas.

Contar con un Reporte de fallas resueltas.

Suministrar documentación de respaldo para medir la futura ocurrencia de errores (análisis de tendencia de errores, problemas recurrentes, errores ocurridos)

Documentación de respaldo (informe de la prueba, informe de resolución de incidencias).

#### **8.2.1.5.4 EJEMPLOS**

Definir un período de tiempo para tomar la métrica.

Contar con un registro de las fallas detectadas (A2) y el historial de resolución de las mismas, pudiendo medir el número de fallas resueltas y el número de fallas detectadas.

Por ejemplo, fallas en proceso de resolución, fallas pendientes, fallas resueltas. En este caso se tomarán las fallas que se encuentren en estado resuelto (A1). Este punto dependerá de los diferentes estados que se definan previamente, pero se debería tener en cuenta que el estado Resuelto cuenta con la aceptación del Usuario que generó la falla.

Para predecir el potencial número de futuras fallas se puede utilizar un método que utilice un modelo de estimación de crecimiento (A3).

#### **3.2.1.5.1 GLOSARIO**

Falla = Incidencia = Error

#### **8.2.1.6 Mean time between failures**

Cuán seguido falla el software en operación

##### **8.2.1.6.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alto (CED)

##### **8.2.1.6.2 VALIDACIONES.**

La fórmula de esta métrica es válida y aplicable.

**Método de Aplicación:** Contar el número de fallas ocurridas durante un periodo definido de operación y computar el intervalo promedio entre fallas

##### **Formula:**

(a)  $X = T1 / A$

(b)  $Y = T2 / A$

Donde

T1= tiempo de operación

T2 = suma de intervalos entre ocurrencias de fallas consecutivas

A = numero total de fallas detectadas (fallas ocurridas durante el periodo de tiempo)

##### **Interpretación de la métrica:**

X, Y >0

Cuanto más largo mejor, a medida que el tiempo es mayor entre fallas.

##### **Escala de Métrica:**

(a) Proporción

(b) Proporción

**Tipo Medición:**

A = contar

T1 = tiempo

T2 = tiempo

X = tiempo/contar

Y = tiempo/contar

**Entrada a la medición:** Reporte prueba, reporte de operación

**Notas:**

1. La siguiente investigación puede ser de ayuda:
  - a. Distribución del intervalo de tiempo entre ocurrencia
  - b. Cambios de duración de tiempos con los intervalos de operación de tiempo
  - c. Distribución indicando que función tiene fallas frecuentes y operación debido a funcionalidad y dependencia de uso.
2. Tasa de falla puede ser una alternativa a utilizar.

**8.2.1.6.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Contar con la documentación necesaria y actualizada.

Contar con un Reporte de fallas detectadas, describiendo tiempos de ocurrencia y tiempos de resolución de las mismas.

Documentación de respaldo (informe de la prueba, informe de resolución de incidencias).

**8.2.1.6.4 EJEMPLOS**

Definir un período de tiempo para tomar la métrica (T1).

Contar con un registro de las fallas detectadas (A2) y el historial de resolución de las mismas teniendo en cuenta indicar los tiempos de ocurrencia de las fallas (T2) y el tiempo de resolución de las mismas.

El estado de las fallas dependerá de los diferentes estados que se definan previamente para la organización, pero se debería tener en cuenta que el estado Resuelto cuenta con la aceptación del Usuario que generó la falla.

**8.2.1.6.5 GLOSARIO**

Falla = Incidencia = Error

**8.2.1.7 Test Coverage (Specified operation scenario testing coverage)**

**8.2.1.7.1 IMPORTANCIA GENERAL/ ESCENARIOS.**

Alto (CED)

**8.2.1.7.2 VALIDACIONES.**

La fórmula de esta métrica es válida y aplicable.

Los datos que se validarán son:

- A y B deben ser números enteros.
- A debe ser  $\leq$  a B
- X debe ser un resultado entre 0 y 1

**8.2.1.7.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Contar con la documentación necesaria y actualizada.

Suministrar el lote de casos de prueba según los requerimientos

Documentación de respaldo (especificación de requerimientos, especificación de la prueba, informe de la prueba y manual del usuario)

#### **8.2.1.7.4 EJEMPLOS**

A= 3, B= 3, X= 1

De este ejemplo se deduce que se realizaron casos de prueba que consideran todos los requerimientos solicitados.

A= 2, B= 4, X= 0.5

De este ejemplo se deduce que se realizaron casos de prueba que no consideran todos los requerimientos solicitados.

#### **8.2.1.7.5 GLOSARIO**

### **8.2.1.8 Test Maturity.**

#### **8.2.1.8.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alto (CED)

#### **8.2.1.8.2 VALIDACIONES.**

La fórmula de esta métrica es válida y aplicable.

Los datos que se validarán son:

- A y B deben ser números enteros.
- A debe ser  $\leq$  a B
- X debe ser un resultado entre 0 y 1

#### **8.2.1.8.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Contar con la documentación necesaria y actualizada.

Suministrar el lote de casos de prueba según los requerimientos

Documentación de respaldo (especificación de requerimientos, especificación de la prueba e informe de la prueba)

#### **8.2.1.8.4 EJEMPLOS**

A= 2, B= 3, X= 0.67

De este ejemplo se deduce que no se realizaron los casos de prueba suficientes para considerar los requerimientos.

#### **8.2.1.8.5 GLOSARIO**

### **8.2.2 Tolerancia a Fallas (Fault tolerance).**

#### **8.2.2.1 Breakdown Avoidance.**

#### **8.2.2.1.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alto (ED)

#### **8.2.2.1.2 VALIDACIONES.**

La fórmula de esta métrica es válida y aplicable.

Los datos que se validarán son:

- A y B deben ser números enteros.
- A debe ser  $\leq$  a B
- X debe ser un resultado entre 0 y 1

#### **8.2.2.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Contar con la documentación necesaria y actualizada.

Documentación de respaldo (informe de la prueba e informe de la operación)

#### **8.2.2.1.4 EJEMPLOS**

A= 2, B= 3, X= 0.33

De este ejemplo se deduce que existe un 33% de probabilidad que se produzca un breakdown en el ambiente de producción respecto de un producto de software determinado.

#### **8.2.2.1.5 GLOSARIO**

Breakdown significa que la ejecución de algunas tareas del usuario son suspendidas.

Defecto es la terminación de la habilidad de un producto de realizar la función requerida o no habilidad de realizar algo dentro de los límites especificados de manera previa.

### **8.2.2.2 Failure Avoidance.**

#### **8.2.2.2.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alto (ED)

#### **8.2.2.2.2 VALIDACIONES.**

La fórmula de esta métrica es válida y aplicable.

Los datos que se validarán son:

- A y B deben ser números enteros.
- A debe ser  $\leq$  a B
- X debe ser un resultado entre 0 y 1

#### **8.2.2.2.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Contar con la documentación necesaria y actualizada.

Suministrar el lote de casos de prueba según el modelo de falla

Documentación de respaldo (informe de la prueba e informe de la operación)

#### **8.2.2.2.4 EJEMPLOS**

A= 3, B= 3, X= 1

De este ejemplo se deduce que se ejecutaron todos los casos de prueba posibles considerando el modelo de falla.

#### **8.2.2.2.5 GLOSARIO**

Falla: es un paso, proceso o definición de datos incorrectos de un programa de computadora.

### **8.2.2.3 Incorrect Operation Avoidance.**

#### **8.2.2.3.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alto (CED)

#### **8.2.2.3.2 VALIDACIONES.**

La fórmula de esta métrica es válida y aplicable.

Los datos que se validarán son:

- A y B deben ser números enteros.
- A debe ser  $\leq$  a B
- X debe ser un resultado entre 0 y 1

#### **8.2.2.3.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Contar con la documentación necesaria y actualizada.

Documentación de respaldo (informe de prueba e informe de operación)

#### **8.2.2.3.4 EJEMPLOS**

A= 2, B= 3, X= 0.67

De este ejemplo se deduce que gran parte de la funciones fueron implementadas con capacidad de evitar operaciones incorrectas.

#### **8.2.2.3.5 GLOSARIO**

### **8.2.3 Recuperación – Capacidad de recuperación (Recoverability).**

#### **8.2.3.1 Availability.**

##### **8.2.3.1.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alto (ED)

##### **8.2.3.1.2 VALIDACIONES.**

La fórmula de esta métrica es válida y aplicable.

Los datos que se validarán son:

- A<sub>1</sub> y A<sub>2</sub> deben ser números enteros.
- A<sub>1</sub> debe ser <= a A<sub>2</sub>
- X e Y deben ser un resultado entre 0 y 1
- T<sub>0</sub> y T<sub>r</sub> > 0

##### **8.2.3.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Contar con la documentación necesaria y actualizada.

Suministrar el lote de casos de uso

Documentación de respaldo (informe de la prueba e informe de la operación)

##### **8.2.3.1.4 EJEMPLOS**

A<sub>1</sub>=2 , A<sub>2</sub> = 2, X= 1

De este ejemplo, se deduce que el sistema está 100% disponible para ser usado en un período de tiempo especificado.

##### **8.2.3.1.5 GLOSARIO**

#### **8.2.3.2 Mean down time.**

Tiempo de Inactividad.

##### **8.2.3.2.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta (CED)

##### **8.2.3.2.2 VALIDACIONES.**

La fórmula de esta métrica es válida y aplicable ya que la misma considera estimar los tiempos medios en el cual el sistema permanece inaccesible cuando se produce una falla, antes de su reinicio.

A las vez, podemos indicar, que es relevante pues impacta en los costos relacionados con la inoperancia del sistema.

**Método de Aplicación:** Estimar el tiempo de inactividad durante un período de prueba determinado y contabilizar los tiempos inactivos provocados por fallas en el sistema.

**Fórmula:**

$$X = T / N$$

Dónde:

T = Sumatoria total de tiempos inactivos

N = Cantidad de interrupciones

El peor caso o distribución de tiempos inactivos debe ser medido.

**Interpretación de la métrica:**

$0 < X$

Cuanto más pequeños son los tiempos mejor es el resultado pues el sistema estará inactivo menor tiempo.

**Escala de Métrica:**

Proporción

**Tipo Medición:**

T = Tiempo (de inoperancia)

N = Contar

X = Tiempo/Contar

**Notas:**

3. Se recomienda que esta métrica de recuperabilidad incluya solamente la recuperación automática proporcionada por el software y excluya el trabajo de mantenimiento humano.
4. Es necesario que el valor de (X) se encuentre entre el intervalo  $<0,1>$  si se realiza una sumatoria de características.

#### **8.2.3.2.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

- ☐ Documentación necesaria y actualizada sobre el manual operacional del software y del usuario.
- ☐ Reportes Operacionales donde se detallen las fallas y los tiempos de inactividad.
- ☐ Documentación de respaldo de testeos efectuados (informe de la prueba, informe de tiempos de inactividad ante cada fallo, categorías de fallos).
- ☐ Especificaciones de requerimientos del cliente/ del negocio u otras.

#### **8.2.3.2.4 EJEMPLOS.**

La misma es aplicable p.ej.: en un call center donde encontramos:

- ☐ tiempos de respuesta de alta criticidad
- ☐ elevado volumen de transacciones monetarias

De ello deducimos que la falta de respuesta en tiempo y forma durante prolongados lapsos de inactividad del sistema -producidos por fallos- incurriría en elevados costos directos.

Definir un período de tiempo para tomar la métrica.

Producir distintos fallos durante el período de tiempo definido.

Tomar y registrar el tiempo de inactividad ante cada fallo.

Aplicar la fórmula de la métrica indicada.

#### **8.2.3.2.5 GLOSARIO.**

#### **8.2.3.3 Mean recovery time.**

Tiempo de recupero.

#### **8.2.3.3.1 IMPORTANCIA GENERAL / ESCENARIOS.**



Alta (CED)

#### **8.2.3.3.2 VALIDACIONES.**

La fórmula de esta métrica es válida y aplicable ya que la misma considera estimar los tiempos medios que le lleva al sistema el recupero total desde el inicio del mismo.

**Método de Aplicación:** Estimar el tiempo de recupero completo durante un período de prueba determinado y contabilizar los tiempos.

#### **Fórmula:**

$X = \text{Sumatoria (T) / B}$

#### **Dónde:**

T = Tiempo que tarda en restablecerse el sistema luego de cada caída.

N = Cantidad de casos en que se producen recuperos

El peor caso o distribución de tiempos inactivos debe ser medido.

#### **Interpretación de la métrica:**

$0 < X$

Cuanto más pequeños son los tiempos mejor es el resultado.

#### **Escala de Métrica:**

Proporción

#### **Tipo Medición:**

T = Tiempo (de inoperancia)

N = Contar

X = Tiempo/Contar

#### **8.2.3.3.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

- ☐ Documentación necesaria y actualizada sobre el manual operacional del software y del usuario.
- ☐ Reportes Operacionales donde se detallen las fallas y los tiempos de recupero desde el inicio.
- ☐ Documentación de respaldo de testeos efectuados (informe de la prueba, informe de tiempos de recupero desde el inicio ante cada fallo, categorías de fallos).
- ☐ Especificaciones de requerimiento del cliente/del negocio u otras.

#### **8.2.3.3.4 EJEMPLOS**

La misma es aplicable p.ej.: en un call center donde encontramos:

- ☐ tiempos de respuesta de alta criticidad
- ☐ elevado volumen de transacciones monetarias

De ello deducimos que la falta de respuesta en tiempo y forma durante prolongados lapsos de inactividad del sistema -producidos por fallos- incurriría en elevados costos directos.

#### **8.2.3.3.5 GLOSARIO**

#### **8.2.3.4 Restartability.**

Capacidad de restart-up brindando servicio a usuarios.

#### **8.2.3.4.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta (CED)

#### **8.2.3.4.2 VALIDACIONES.**

En relación al propósito de la métrica entendemos que su significado no se ajusta en sentido estricto con el método de aplicación, y lo re-definimos como la capacidad/efectividad de reinicio del sistema (brindando servicio) en un lapso de tiempo determinado ante un evento dado.

##### **Formula:**

$$X=A/B$$

A= Cantidad de reinicios que alcanzan los tiempos requeridos durante el testing u operación.

B= Cantidad total de reinicios durante durante el testing u operación.

Cabe aclarar que se deberá catalogar dichos tiempos de reinicio considerando un categoría de eventos (nivel de severidad de inoperabilidad como ser destrucción de la base de datos, perdida de múltiple transacciones, destrucción temporal de datos, perdida de una sola transacción entre otros)

#### **8.2.3.4.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

- ☐ Manual del usuario/operación.
- ☐ Niveles de servicio especificados o requeridos por el cliente.
- ☐ Casos de prueba (preparación/ejecución).
- ☐ Categorización de eventos con los tiempos de reinicio del sistema según su severidad.

#### **8.2.3.4.4 EJEMPLOS**

Se puede aplicar por ejemplo ante la caída de un Switch transaccional (time-out de transacciones) por carga masiva de operaciones.

#### **8.2.3.4.5 GLOSARIO**

### **8.2.3.5 Restorability.**

Restauración del sistema.

#### **8.2.3.5.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta (CED)

#### **8.2.3.5.2 VALIDACIONES.**

El propósito es medir la capacidad del producto para restaurarse luego de un evento anormal o de un pedido por el usuario. Para ello se debe contar el número de restauraciones satisfactorias y compararlas con el número de testeos de restauración requeridos en la especificación.

$$X=A/B$$

A= Número de pruebas de restauración satisfactorias.

B= Número de casos de prueba de restauración según requerimiento.

Fórmula válida.

#### **8.2.3.5.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

- ☐ Manual del usuario/operación.
- ☐ Especificación de Requerimiento.
- ☐ Niveles de servicio especificados o requeridos por el cliente.
- ☐ Casos de prueba (preparación/ejecución).

#### **8.2.3.5.4 EJEMPLOS**

La presente métrica puede ser aplicable en sistemas donde el usuario puede necesitar restaurar el sistema a un punto anterior, si el software presentara inconvenientes de performance, o fallos que llevaran a la pérdida de datos.

#### **8.2.3.5.5 GLOSARIO**

#### **8.2.3.6 Restore effectiveness.**

Eficiencia del sistema en restaurarse.

##### **8.2.3.6.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta (CED)

##### **8.2.3.6.2 VALIDACIONES.**

El propósito de la métrica es medir la efectividad del sistema para restaurarse (como primera medida el sistema debe poseer dicha capacidad). Para ello se debe contar el número de pruebas de restauración satisfactorias según los tiempos establecidos, y compararlo con los tiempos de restauración especificados.

$X=A/B$

A= Número de pruebas de restauración satisfactorias en función del tiempo establecido.

B= Número de casos de prueba de restauración preparados.

Fórmula válida.

##### **8.2.3.6.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

- ☐ Manual del usuario/operación.
- ☐ Especificación de Requerimiento.
- ☐ Niveles de servicio especificados o requeridos por el cliente.
- ☐ Casos de prueba (preparación/ejecución).

#### **8.2.3.6.4 EJEMPLOS**

La presente métrica puede ser aplicable en sistemas donde el usuario puede necesitar restaurar el sistema a un punto anterior, si el software presentara inconvenientes de performance, o fallos que llevaran a la pérdida de datos.

#### **8.2.3.6.5 GLOSARIO**

### **8.2.4 Compatibilidad en Confiabilidad (Reliability Compliance).**

#### **8.2.4.1 Reliability Compliance.**

Compatibilidad en la confiabilidad.

##### **8.2.4.1.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta (CED)

#### 8.2.4.1.2 VALIDACIONES.

El propósito es medir la confiabilidad del producto según regulaciones, estándares, convenciones y normas. Se debe contar el número de ítems que cumplen con los estándares y compararlo con el número de ítems que se solicitan en las especificaciones.

$X=1 - A/B$

A= Número de ítems de confiabilidad especificados que no han sido implementados.

B= Total de ítems de confiabilidad especificados.

Fórmula válida.

Cabe aclarar que para mayor entendimiento respecto del propósito establecido, la siguiente fórmula también es válida y genera el mismo resultado:

**$X=A/B$**

donde **A** es el número de ítems que fueron implementados y **B** el número total de ítems especificados.

#### 8.2.4.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.

Se deberá contar con la siguiente documentación actualizada:

- ☐ Manual de Usuario
- ☐ Guías de Estándares y Convenciones
- ☐ Checklist de verificación
- ☐ Test Cases
- ☐ Certificaciones
- ☐ Normas y Convenciones a las que da cumplimiento el software

#### 8.2.4.1.4 EJEMPLOS

Testeo de aplicativo de acuerdo a los estándares y regulaciones que se indican en la especificación que debe cumplir el producto.

#### 8.2.4.1.5 GLOSARIO

### 8.3 Usabilidad (Usability).

#### 8.3.1 Entendible (Understandability).

##### 8.3.1.1 Completeness of description.

Exhaustividad de la descripción.

Responde a: ¿Qué proporción de las funciones (o tipos de funciones) es comprendida después de leer la descripción del producto?

##### 8.3.1.1.1 IMPORTANCIA GENERAL / ESCENARIOS.

Media. (CED).

Depende del tipo de producto y el estado del desarrollo. En un producto estándar indica si los potenciales compradores pueden comprender el alcance del producto a partir de su descripción. En un desarrollo a medida puede ser importante verificar que la descripción del producto establece claramente su alcance.

##### 8.3.1.1.2 VALIDACIONES.

$$X = A / B$$

A = Número de funciones (o tipo de funciones) comprendidas

B = Número total de funciones

Fórmula correcta.

**8.3.1.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Se debe disponer de un documento de alcance del producto. No es aplicable a documentación extensa, como un manual de usuario o un manual de requerimientos.

**8.3.1.1.4 EJEMPLOS.**

Permite revisar la calidad de documentación genérica del producto, por ejemplo un documento de funcionalidad general del sistema.

**8.3.1.1.5 GLOSARIO.**

**8.3.1.2 Demonstration accessibility.**

Accesibilidad de las demostraciones

Responde a: ¿Qué proporción de las demostraciones o tutoriales puede acceder el usuario?  
Indica si los usuarios pueden encontrar las demostraciones o tutoriales en un ambiente de test conducido.

**8.3.1.2.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta. (CED).

Depende de si resulta importante que el usuario sea auto suficiente en el uso del sistema.

**8.3.1.2.2 VALIDACIONES.**

$$X = A / B$$

A = Número de demostraciones o tutoriales que el usuario accedió exitosamente

B = Número total de demostraciones / tutoriales disponibles

Fórmula correcta.

**8.3.1.2.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo. Manual o ayudas completos. El evaluador sugiere los temas que el operador debe encontrar.

**8.3.1.2.4 EJEMPLOS.**

**8.3.1.2.5 GLOSARIO.**

**8.3.1.3 Demonstration accessibility in use.**

Accesibilidad de las demostraciones durante el uso

Responde a: ¿Qué proporción de las demostraciones o tutoriales puede acceder el usuario cuando realmente necesita hacerlo durante la operación? Indica si los usuarios pueden encontrar las demostraciones o tutoriales mientras usan el producto.

**8.3.1.3.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta. (CED).

Depende de si resulta importante que el usuario sea auto suficiente en el uso del sistema.

**8.3.1.3.2 VALIDACIONES.**

$$X = A / B$$

A = Número de casos en los cuales el usuario ve la demostración cuando necesitaba verla

B = Número de casos en los que el usuario intentó ver una demostración durante el período de observación

Fórmula correcta. Muy difícil de aplicar en la realidad, pues requiere medir exactamente cuántas veces el usuario necesitó una demostración o tutorial y pudo encontrarla.

**8.3.1.3.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo. Manual o ayudas completos.

**8.3.1.3.4 EJEMPLOS.**

**8.3.1.3.5 GLOSARIO.**

**8.3.1.4 Demonstration effectiveness.**

Efectividad de las demostraciones

Responde a: ¿Qué proporción de las funciones puede el usuario operar satisfactoriamente después de ver una demostración o tutorial? Indica si los usuarios pueden realizar una función satisfactoriamente gracias a la demostración o tutorial en un ambiente de test.

**8.3.1.4.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta. (CED).

Depende de si resulta importante que el usuario sea auto suficiente en el uso del sistema.

**8.3.1.4.2 VALIDACIONES.**

$X = A / B$

A = Número de funciones operadas exitosamente

B = Número de demostraciones / tutoriales accedidos

Fórmula correcta.

**8.3.1.4.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo. Manual o ayudas completos.

**8.3.1.4.4 EJEMPLOS.**

**8.3.1.4.5 GLOSARIO.**

**8.3.1.5 Evident functions.**

Funcionalidad evidente

Responde a: ¿Qué proporción de las funciones pueden ser identificadas por el usuario basándose en las condiciones iniciales? Indica si los usuarios pueden ser capaces de encontrar una función explorando la interfaz (menús, botones).

**8.3.1.5.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta. (CED).

Depende de si resulta importante que el usuario sea auto suficiente en el uso del sistema.

**8.3.1.5.2 VALIDACIONES.**

$X = A / B$

A = Número de funciones (o tipo de funciones) identificadas por el usuario

B = Número total de funciones reales (o tipos de funciones)

Fórmula correcta.

**8.3.1.5.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo. El evaluador debe sugerir la funcionalidad a detectar por el usuario que participa del test.

**8.3.1.5.4 EJEMPLOS.**

**8.3.1.5.5 GLOSARIO.**

**8.3.1.6 Function understandability.**

Comprensión de la funcionalidad

Responde a: ¿Qué proporción de las funciones pueden ser comprendidas correctamente por el usuario? Indica si los usuarios pueden ser capaces de comprender correctamente una función explorando la interfaz (menús, botones).

**8.3.1.6.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta. (CED).

Depende de si resulta importante que el usuario sea auto suficiente en el uso del sistema.

**8.3.1.6.2 VALIDACIONES.**

$X = A / B$

A = Número de funciones cuyo propósito es correctamente descripto por el usuario

B = Número total de funciones accesibles desde la interfaz

Fórmula correcta.

**8.3.1.6.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo. El evaluador debe sugerir la funcionalidad a comprender por el usuario que participa del test.

**8.3.1.6.4 EJEMPLOS.**

**8.3.1.6.5 GLOSARIO.**

**8.3.1.7 Understandable input and output.**

Comprensión de los datos de entrada y salida

Responde a: ¿Puede el usuario comprender lo que se requiere como datos de entrada y lo que es provisto como datos de salida por el sistema? Indica si los usuarios pueden entender el formato en que los datos deben ser ingresados e identificar correctamente el significado de los datos de salida.

**8.3.1.7.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta. (CED)

**8.3.1.7.2 VALIDACIONES.**

$X = A / B$

A = Número de datos de entrada y salida que el usuario comprendió correctamente

B = Número de datos de entrada y salida accesibles de la interfaz

Fórmula correcta.

**8.3.1.7.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo. El evaluador debe conocer cuáles son los casos que merecen ser puestos a prueba, para no usar ejemplos obvios.

**8.3.1.7.4 EJEMPLOS.**

**8.3.1.7.5 GLOSARIO.**

**8.3.2 Aprendizaje – Capacidad para ser aprendido (Learnability).**

**8.3.2.1 Ease of function learning.**

Facilidad de aprendizaje de la funcionalidad

Responde a: ¿Cuánto tiempo le toma al usuario aprender a usar una función?

**8.3.2.1.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Media. (CED)

**8.3.2.1.2 VALIDACIONES.**

T = tiempo medio que toma aprender a usar una función correctamente.

Fórmula correcta. Difícil de medir en la realidad: no todos los usuarios son iguales, no es fácil estar seguro de que se comprendió correctamente la funcionalidad, no es fácil comparar dificultad de funcionalidades y sus correspondientes expectativas de tiempo de aprendizaje.

**8.3.2.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo. El evaluador debe conocer cuáles son los casos que merecen ser puestos a prueba, para no usar ejemplos obvios.

**8.3.2.1.4 EJEMPLOS.**

Puede ser usado sobre temas claves en el uso de un sistema. Por ejemplo, que requieran ser aprendidos rápidamente por la cantidad de personal que rota en el puesto que usa esa función.

**8.3.2.1.5 GLOSARIO.**

**8.3.2.2 Ease of learning to perform a task in use.**

Facilidad para aprender a realizar una tarea durante el uso

Responde a: ¿Cuánto tiempo le toma al usuario aprender a realizar eficientemente una tarea? Indica el tiempo entre que el usuario comprende lo que debe hacer y lo realiza eficientemente

**8.3.2.2.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Media. (CED)

**8.3.2.2.2 VALIDACIONES.**

T = suma del tiempo de operación del usuario hasta que el mismo alcanza a ejecutar la tarea específica en un tiempo corto.

Fórmula correcta. Difícil de medir en la realidad: no todos los usuarios son iguales, no es fácil definir lo que significa operar eficientemente.

**8.3.2.2.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo. El evaluador debe conocer cuáles son los casos que merecen ser puestos a prueba, para no usar ejemplos obvios.

**8.3.2.2.4 EJEMPLOS.**



Puede ser usado sobre temas claves en el uso de un sistema. Por ejemplo, que requieran ser aprendidos rápidamente por la cantidad de personal que rota en el puesto que usa esa función.

### **8.3.2.2.5 GLOSARIO.**

#### **8.3.2.3 Effectiveness of the user documentation and/or help system.**

Efectividad de la documentación de usuario y/o ayudas del sistema

Responde a: ¿Qué proporción de las tareas pueden ser completadas correctamente después de consultar la documentación de usuario y las ayudas del sistema?

##### **8.3.2.3.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta. (CED)

Depende de si resulta importante que el usuario sea auto suficiente en el uso del sistema.

##### **8.3.2.3.2 VALIDACIONES.**

$X = A / B$

A = Número de tareas completadas exitosamente luego de acceder la ayuda en línea o la documentación

B = Número total de tareas testeadas

Fórmula correcta.

##### **8.3.2.3.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo. Manual o ayudas completos. El evaluador debe conocer cuáles son los casos que merecen ser puestos a prueba, para no usar ejemplos obvios.

##### **8.3.2.3.4 EJEMPLOS.**

##### **8.3.2.3.5 GLOSARIO.**

#### **8.3.2.4 Effectiveness of the user documentation and/or help systems in use.**

Efectividad de la documentación de usuario y/o ayudas del sistema durante el uso

Responde a: ¿Qué proporción de las funciones pueden ser usadas correctamente después de consultar la documentación de usuario y las ayudas del sistema?

##### **8.3.2.4.1 IMPORTANCIA GENERAL / ESCENARIOS.**

(CED) Alta.

Depende de si resulta importante que el usuario sea auto suficiente en el uso del sistema.

##### **8.3.2.4.2 VALIDACIONES.**

$X = A / B$

A = Número de funciones que pueden ser usadas

B = Número total de funciones provistas

Fórmula correcta. Parece difícil la medición pues se basa en observar el comportamiento del usuario durante el uso normal del sistema, y comparar la cantidad de funciones que usa correctamente con la cantidad de funciones totales. Ambos términos son difíciles de establecer. Además, si opera mal una función habría que estar seguro que consultó correctamente la documentación.

##### **8.3.2.4.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo. Manual o ayudas completos. El evaluador debe conocer cuáles son los casos que debe considerar y cómo se los califica en “uso correcto o incorrecto”.

**8.3.2.4.4 EJEMPLOS.**

**8.3.2.4.5 GLOSARIO.**

**8.3.2.5 Help accessibility.**

Accesibilidad de la ayuda.

Responde a: *¿Qué proporción de tópicos de ayuda en línea puede el usuario encontrar?*

**8.3.2.5.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta. (CED)

Depende de si resulta importante que el usuario sea auto suficiente en el uso del sistema.

**8.3.2.5.2 VALIDACIONES.**

$$X = A / B$$

A = Número de tareas para las cuales se encontró la ayuda en línea correcta

B = Número total de tareas testeadas

Fórmula correcta.

**8.3.2.5.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo. Ayudas completas.

**8.3.2.5.4 EJEMPLOS.**

**8.3.2.5.5 GLOSARIO.**

**8.3.2.6 Help frequency.**

Frecuencia de la ayuda.

Responde a: *¿Cuán frecuentemente el usuario debe acceder a la ayuda para comprender la operatoria hasta poder completar la tarea?*

**8.3.2.6.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta. (CED)

Depende de si resulta importante que el usuario sea auto suficiente en el uso del sistema.

**8.3.2.6.2 VALIDACIONES.**

$$X = A$$

A = número de accesos a la ayuda hasta que el usuario completa su tarea

Fórmula correcta.

**8.3.2.6.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo. Ayudas completas.

**8.3.2.6.4 EJEMPLOS.**

**8.3.2.6.5 GLOSARIO.**

**8.3.3 Operación - Capacidad de Operación (Operability).****8.3.3.1 (A) Conforms with operational user expectations.**

*Conformidad con las expectativas operacionales del usuario*

**8.3.3.1.1 Operational consistency in use.**

Consistencia de la operación durante el uso

Responde a: ¿Son consistentes los componentes de la interfaz de usuario? Indica el grado de conformidad de la interfaz con la expectativa del usuario.

**8.3.3.1.1.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Media. (CED)

**8.3.3.1.1.2 VALIDACIONES.**

a)  $X = 1 - A / B$

A = Número de mensajes o funciones que el usuario encontró inconsistentes con su expectativa

B = Número de mensajes o funciones

b)  $Y = N / UOT$

N = Número de operaciones que el usuario encontró inconsistentes con su expectativa

UOT = tiempo de operación del usuario (durante el período de observación)

Fórmulas correctas.

**8.3.3.1.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo. Ayudas completas.

**8.3.3.1.1.4 EJEMPLOS.****8.3.3.1.1.5 GLOSARIO.****8.3.3.2 (B) Controlable.****8.3.3.2.1 Error correction.**

Corrección de errores

Responde a: ¿Puede el usuario corregir sencillamente errores en las tareas?

**8.3.3.2.1.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Media. (CED)

**8.3.3.2.1.2 VALIDACIONES.**

$T = T_c - T_s$

T<sub>c</sub> = hora en que se completó la corrección de un error determinado

T<sub>s</sub> = hora en que se inició la corrección de un error determinado

Fórmula correcta.

**8.3.3.2.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo. Ayudas completas. Se debe disponer de la serie de errores a ser evaluados.

**8.3.3.2.1.4 EJEMPLOS.**

El set de errores puede contemplar errores de ingreso de datos incorrectos, de problemas de grabación u operacionales.

**8.3.3.2.1.5 GLOSARIO.**

**8.3.3.2.2 Error correction in use.**

Corrección de errores durante el uso

Responde a: ¿Puede el usuario recobrase sencillamente de sus errores o reintentar tareas? ¿Puede el usuario recuperar sencillamente sus datos de input?

**8.3.3.2.2.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta. (CED)

**8.3.3.2.2.2 VALIDACIONES.**

a)  $X = A / UOT$

A = Número de veces en que el usuario consigue cancelar su error de operación

UOT = tiempo de operación del usuario (durante el período de observación)

b)  $X = A / B$

A = Número de ventanas o formularios donde los datos de entrada pudieron ser exitosamente modificados

B = Número de ventanas o formularios donde el usuario intentó modificar los datos de entrada durante el período de observación

Fórmulas correctas. Difíciles de medir dado que deben ocurrir durante la operación de manera no forzada.

**8.3.3.2.2.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo. Ayudas completas.

**8.3.3.2.2.4 EJEMPLOS.**

**8.3.3.2.2.5 GLOSARIO.**

**8.3.3.3 (C) Suitable for the task operation**

*Adaptable para la operación de tareas*

**8.3.3.3.1 Default value availability in use.**

Disponibilidad de valores por defecto durante el uso

Responde a: ¿Puede el usuario seleccionar valores a los parámetros para su conveniente operación de manera sencilla?

**8.3.3.3.1.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Media. (CED)

**8.3.3.3.1.2 VALIDACIONES.**

$X = 1 - A / B$

A = Número de veces que el usuario falló en establecer valores de parámetros (porque no pudo utilizar valores default provistos por el software)

B = Número total de veces en que el usuario intentó establecer o seleccionar valores de parámetros

Fórmula correcta.

**8.3.3.3.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo.

**8.3.3.3.1.4 EJEMPLOS.**

Puede usarse para determinar la existencia de mecanismo de personalización de la interfaz por usuario, y la sencillez y eficacia del método utilizado para establecer los parámetros que el usuario necesita.

**8.3.3.3.1.5 GLOSARIO.****8.3.3.4 (D) Self descriptive (Guiding).**

*Auto descriptivo*

**8.3.3.4.1 Message understandability in use.**

Comprensión de los mensajes durante el uso

Responde a: ¿Puede el usuario entender sencillamente los mensajes de error del sistema? ¿Hay algún mensaje que le provoque demoras al usuario para entenderlo, antes de comenzar la siguiente acción? ¿Puede el usuario conservar sencillamente un mensaje importante?

**8.3.3.4.1.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta. (CED)

**8.3.3.4.1.2 VALIDACIONES.**

$X = A / UOT$

A = Número de veces en que el usuario se detiene por un período largo o repetidamente falla en la misma operación, debido a la poca comprensión de un mensaje.

UOT = tiempo de operación del usuario (durante el período de observación)

Fórmula correcta. Difícil de medir dado que deben ocurrir durante la operación de manera no forzada, donde es azarosa la presencia de mensajes de error. Podría realizarse con un test donde se indique al operador los pasos a seguir llevándolo a caminos erróneos.

**8.3.3.4.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo. Ayudas completas.

**8.3.3.4.1.4 EJEMPLOS.****8.3.3.4.1.5 GLOSARIO.****8.3.3.4.2 Self – explanatory error messages.**

Mensajes de error auto explicativos

Responde a: ¿En qué proporción de las condiciones de error el usuario propone la acción de recuperación correcta?

**8.3.3.4.2.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta. (CED)

**8.3.3.4.2.2 VALIDACIONES.**

$X = A / B$

A = Número de condiciones de error para las cuales el usuario propone de recuperación correcta.

B = Número de condiciones de error testeadas

Fórmula correcta.

**8.3.3.4.2.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo. Ayudas completas.

**8.3.3.4.2.4 EJEMPLOS.**

**8.3.3.4.2.5 GLOSARIO.**

**8.3.3.5 (E) Operational error tolerant (Human error free).**

*Tolerancia a errores de operación (libre de errores humanos)*

**8.3.3.5.1 Operational error recoverability in use.**

Recuperación ante errores de operación durante el uso

Responde a: ¿Puede el usuario recuperarse sencillamente en las peores situaciones?  
Toma en cuenta el peor de los casos: donde el sistema no advierte al usuario el riesgo que corre debido al error.

**8.3.3.5.1.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Media. (CED)

**8.3.3.5.1.2 VALIDACIONES.**

$X = 1 - A / B$

A = Número de situaciones en que se fracasa el recuperarse (luego de un error o cambio) en las cuales el usuario no fue informado del riesgo por el sistema

B = Número de errores de usuario o cambios

Fórmula correcta. Difícil de medir dado que deben ocurrir durante la operación de manera no forzada, donde es azarosa la presencia de errores.

**8.3.3.5.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo.

**8.3.3.5.1.4 EJEMPLOS.**

**8.3.3.5.1.5 GLOSARIO.**

**8.3.3.5.2 Time between human error operations in use.**

Tiempo entre errores humanos de operación durante el uso

Responde a: ¿Puede el usuario operar el sistema durante bastante tiempo sin errores humanos? Contempla errores sencillos de entrada de datos, errores reiterativos por incomprensión de la tarea y pausas durante un tiempo significativo debido a dudas respecto a los pasos a seguir.

**8.3.3.5.2.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Media. (CED)

**8.3.3.5.2.2 VALIDACIONES.**

$X = T / N$

T = tiempo de operación durante la observación

N = número de ocurrencia de errores de operación humanos

Fórmula correcta. Difícil de medir dado que deben ocurrir durante la operación de manera no forzada, donde es azarosa la presencia de errores.

**8.3.3.5.2.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo..

**8.3.3.5.2.4 EJEMPLOS.**

**8.3.3.5.2.5 GLOSARIO.**

**8.3.3.5.3 Undoability (User error correction).**

Capacidad de “deshacer” (corrección de errores del usuario)

Responde a: ¿Qué tan frecuentemente corrige el usuario satisfactoriamente errores de ingreso de datos? ¿Qué tan frecuentemente el usuario deshace errores correctamente?

**8.3.3.5.3.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Media. (CED)

**8.3.3.5.3.2 VALIDACIONES.**

a)  $X = A / B$

A = Número de errores de ingreso de datos que el usuario corrige exitosamente.

B = Número de intentos de corregir errores de ingreso de datos

b)  $Y = A / B$

A = Número de condiciones de error que el usuario corrige exitosamente.

B = Número de condiciones de error testeadas

Fórmulas correctas.

**8.3.3.5.3.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo. El evaluador debe conocer el test de situaciones de error a provocar.

**8.3.3.5.3.4 EJEMPLOS.**

**8.3.3.5.3.5 GLOSARIO.**

**8.3.3.6 (F)(Suitable for individualisation.**

*Preparado para personalizaciones*

**8.3.3.6.1 Customisability.**

Personalizable

Responde a: ¿Puede el usuario personalizar con facilidad la operación de los procesos según su conveniencia? ¿Puede un usuario, que capacita usuarios finales, setear templates de operación de procesos para prevenir sus errores? ¿Qué proporción de las funciones pueden ser personalizadas?

**8.3.3.6.1.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Media. (CED)

**8.3.3.6.1.2 VALIDACIONES.**

$X = A / B$

A = Número de funciones satisfactoriamente personalizadas.

B = Número de intentos de personalizar  
Fórmula correcta.

### **8.3.3.6.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo.

### **8.3.3.6.1.4 EJEMPLOS.**

### **8.3.3.6.1.5 GLOSARIO.**

## **8.3.3.6.2 Operation procedure reduction.**

Simplificación de procedimientos operativos

Responde a: ¿Puede el usuario reducir con facilidad un procedimiento operativo según su conveniencia? La reducción debe medirse como cantidad de teclas pulsadas, datos ingresados, clicks del mouse, etc.

### **8.3.3.6.2.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Media. (CED)

### **8.3.3.6.2.2 VALIDACIONES.**

$$X = 1 - A / B$$

A = Número de pasos de los procedimientos reducidos luego de su personalización.

B = Número de pasos de los procedimientos antes de la personalización

Fórmula correcta.

### **8.3.3.6.2.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo.

### **8.3.3.6.2.4 EJEMPLOS.**

### **8.3.3.6.2.5 GLOSARIO.**

## **8.3.3.6.3 Physical accessibility.**

Accesibilidad

Responde a: ¿Qué proporción de las funciones pueden ser accedidos por usuarios con alguna incapacidad física? Debe considerar la ceguera o disminución visual, la incapacidad de operar un mouse, etc.

### **8.3.3.6.3.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Media. (CED)

### **8.3.3.6.3.2 VALIDACIONES.**

$$X = A / B$$

A = Número de funciones exitosamente accedidas.

B = Número de funciones

Fórmula correcta.

### **8.3.3.6.3.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo.

### **8.3.3.6.3.4 EJEMPLOS.**

### **8.3.3.6.3.5 GLOSARIO.**



### **8.3.4 Atractivo – Capacidad de ser atractivo (Attractiveness).**

#### **8.3.4.1 Attractive Interaction.**

Interacción atractiva

Responde a: ¿Qué tan atractiva es la interfaz de usuario?

##### **8.3.4.1.1 IMPORTACION GENERAL / ESCENARIOS.**

Media. (CED)

##### **8.3.4.1.2 VALIDACIONES.**

Depende del cuestionario que se realice.

##### **8.3.4.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo.

##### **8.3.4.1.4 EJEMPLOS.**

##### **8.3.4.1.5 GLOSARIO.**

#### **8.3.4.2 Interface appearance customisability.**

Personalización de la apariencia de la interfaz

Responde a: ¿Qué proporción de los elementos de la interfaz pueden ser personalizados en su apariencia para la satisfacción del usuario?

##### **8.3.4.2.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Media. (CED)

##### **8.3.4.2.2 VALIDACIONES.**

$X = A / B$

A = Número de elementos de la interfaz personalizados en su apariencia a satisfacción del usuario.

B = Número de elementos de la interfaz que el usuario desea personalizar

Fórmula correcta.

##### **8.3.4.2.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo.

##### **8.3.4.2.4 EJEMPLOS.**

##### **8.3.4.2.5 GLOSARIO.**

### **8.3.5 Compatibilidad en Usabilidad (Usability Compliance).**

#### **8.3.5.1 Usability Compliance.**

Compatibilidad en usabilidad

Responde a: ¿Qué tanto adhiere el sistema a estándares, convenciones, guías de estilo o regulaciones relativas a usabilidad?

##### **8.3.5.1.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta. (CED)

**8.3.5.1.2 VALIDACIONES.**

$$X = 1 - A / B$$

A = Número de ítems de compatibilidad que no se vieron implementados.

B = Número de ítems de compatibilidad especificados

Fórmula correcta.

**8.3.5.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sistema completo. Set de estándares que se determine que se deben cumplir.

**8.3.5.1.4 EJEMPLOS**

**8.3.5.1.5 GLOSARIO.**

## 8.4 Eficiencia (Efficiency).

### 8.4.1 Performance - Comportamiento en Tiempo (Time Behaviour).

#### 8.4.1.1 (A) Response Time.

##### 8.4.1.1.1 Response Time.

###### 8.4.1.1.1.1 IMPORTANCIA GENERAL / ESCENARIOS.

Alto. (CED)

###### 8.4.1.1.1.2 VALIDACIONES.

Fórmula válida. Mide el tiempo que lleva en completar la ejecución de una determinada tarea.

###### 8.4.1.1.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.

Para esta métrica sólo es necesario tomar el tiempo de ejecución de la tarea.

###### 8.4.1.1.1.4 EJEMPLOS.

Tiempo total de ejecución de un alta de un cliente en una base.

Tiempo total de ejecución de la generación de un reporte.

###### 8.4.1.1.1.5 GLOSARIO.

N/A

##### 8.4.1.1.2 Response Time (Mean Time to response).

Tiempo medio de respuesta

###### 8.4.1.1.2.1 IMPORTANCIA GENERAL / ESCENARIOS.

Baja (CE), Medio (D)

###### 8.4.1.1.2.2 VALIDACIONES.

Fórmula válida. Mide el tiempo promedio que lleva en completar la ejecución de tareas concurrentes. Desde que comienzan hasta que se obtiene la respuesta.

###### 8.4.1.1.2.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.

Para esta métrica sólo es necesario tomar el tiempo de ejecución de las tareas. Sería interesante conocer tiempos de ejecuciones históricas de las tareas que se están midiendo.

**8.4.1.1.2.4 EJEMPLOS.**

Promedio de tiempo de ejecución de las tareas 1), 2) y 3) a la vez:

- 1) Alta de un cliente en una base.
- 2) Generación de un reporte.
- 3) Envío de información mediante Interfaz.

**8.4.1.1.2.5 GLOSARIO.**

N/A

**8.4.1.1.3 Response Time (Worst case response time ratio).**

Tiempo de respuesta (caso límite o peor caso)

**8.4.1.1.3.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Baja (CE), Medio (D)

**8.4.1.1.3.2 VALIDACIONES.**

Fórmula válida. Mide el tiempo máximo que puede tardar una (o varias) función en la peor condición de procesamiento del sistema.

**8.4.1.1.3.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Sería necesario conocer cual es el tiempo adecuado de ejecución de la función para poder comparar con el tiempo real.

**8.4.1.1.3.4 EJEMPLOS.**

Ejecutar una cantidad determinada de funciones concurrentes de interfaz, esperando respuestas de la misma.

**8.4.1.1.3.5 GLOSARIO.**

N/A

**8.4.1.2 (B) Throughput.**

**8.4.1.2.1 Throughput.**

Cantidad de tareas ejecutadas exitosamente en un determinado tiempo.

**8.4.1.2.1.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alto (CED)

**8.4.1.2.1.2 VALIDACIONES.**

Fórmula válida. Mide la cantidad de tareas ejecutadas exitosamente en un determinado tiempo.

**8.4.1.2.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

N/A

**8.4.1.2.1.4 EJEMPLOS.**

Ejecutar una cantidad determinada de tareas de acceso a bases de datos en un periodo de tiempo y calcular cuantas de ellas finalizaron exitosamente.

**8.4.1.2.1.5 GLOSARIO.**

N/A

**8.4.1.2.2 Throughput (Mean amount of throughput).**

Cantidad promedio de tareas ejecutadas exitosamente en un determinado tiempo.

**8.4.1.2.2.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Baja (CE), Medio (D)

**8.4.1.2.2.2 VALIDACIONES.**

Fórmula válida. Mide el promedio de cantidad de tareas ejecutadas exitosamente en un determinado tiempo.

**8.4.1.2.2.3 PRECONDICIONES / DOCUMENTACION DE RESPALDO.**

N/A

**8.4.1.2.2.4 EJEMPLOS.**

Ejecutar una cantidad determinada de tareas de acceso a bases de datos en un periodo de tiempo y calcular el promedio de aquellas que finalizaron exitosamente.

**8.4.1.2.2.5 GLOSARIO.**

N/A

**8.4.1.2.3 Throughput (Worst case throughput ratio).**

Cantidad de tareas ejecutadas concurrentemente en el peor escenario (caso límite o peor caso)

**8.4.1.2.3.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Baja (CE), Medio (D)

**8.4.1.2.3.2 VALIDACIONES.**

Fórmula válida. Mide la cantidad de tareas ejecutadas concurrentemente en el peor escenario de procesamiento en un periodo de tiempo

**8.4.1.2.3.3 PRECONDICIONES / DOCUMENTACION DE RESPALDO.**

N/A

**8.4.1.2.3.4 EJEMPLOS.**

Ejecutar una cantidad determinada de tareas de acceso a bases de datos concurrentemente en el peor escenario de procesamiento durante un periodo de tiempo.

**8.4.1.2.3.5 GLOSARIO.**

N/A

**8.4.1.3 (C) Turnaround Time.**

**8.4.1.3.1 Turnaround time.**

N/A debido a que se podría unificar esta métrica con la Response Time ya que ambas están referidas a tiempos de ejecuciones de tareas.

**8.4.1.3.2 Turnaround time (Mean time for turnaround).**

N/A debido a que se podría unificar esta métrica con la Response Time ya que ambas están referidas a tiempos de ejecuciones de tareas.

**8.4.1.3.3 Turnaround time (Worst case turnaround time ratio).**

N/A debido a que se podría unificar esta métrica con la Response Time ya que ambas están referidas a tiempos de ejecuciones de tareas.

**8.4.1.3.4 Waiting time.**

Tiempo que los usuarios gastan esperando la respuesta del sistema.

**8.4.1.3.4.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alto (CED)

**8.4.1.3.4.2 VALIDACIONES.**

Fórmula válida. Mide la proporción del tiempo que los usuarios “pierden” esperando la respuesta del sistema a la ejecución de una tarea, con respecto al tiempo total de ejecución de la misma

**8.4.1.3.4.3 PRECONDICIONES / DOCUMENTACION DE RESPALDO.**

N/A

**8.4.1.3.4.4 EJEMPLOS.**

Ejecutar una tarea de consulta a una base de datos para medir la proporción de tiempo de espera de la respuesta.

**8.4.1.3.4.5 GLOSARIO.**

N/A

**8.4.2 Utilización de Recursos (Resource Utilisation).**

**8.4.2.1 (A) I/O Devices resource utilization.**

**8.4.2.1.1 I/O devices utilisation.**

Ineficiencia a causa de la alta utilización de los dispositivos de E/S.

**8.4.2.1.1.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alto (CD), Medio (E)

**8.4.2.1.1.2 VALIDACIONES.**

Fórmula válida. Mide el tiempo ocupado por los dispositivos de E/S con respecto al tiempo especificado en el diseño para ser utilizados por los dispositivos de E/S.

**8.4.2.1.1.3 PRECONDICIONES / DOCUMENTACION DE RESPALDO.**

- ☐ Especificación de tiempos a utilizar por los dispositivos de E/S.
- ☐ Histórico de tiempos utilizados por los distintos dispositivos de E/S.
- ☐ Resultados de test de performance.

**8.4.2.1.1.4 EJEMPLOS.**

1. Ejecutar concurrentemente un gran número de tareas de entradas/salidas, registrar la utilización de dispositivos de E/S y comparar con los objetivos de diseño.
2. Asignación de memoria menor a la requerida por un programa, aumenta las E/S para traer y llevar las instrucciones que se ejecutan.

**8.4.2.1.1.5 GLOSARIO.**

N/A.

**8.4.2.1.2 I/O loading limits.**

Límite de utilización de E/S en la realización de una función.

**8.4.2.1.2.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alto (CD), Medio (E)

**8.4.2.1.2.2 VALIDACIONES.**

Fórmula válida. Realiza N evaluaciones, en cada una registra la cantidad de mensajes de E/S. Selecciona la evaluación con mayor cantidad de mensajes. Mide la cantidad de mensajes de E/S de la evaluación seleccionada respecto a la cantidad de mensajes de E/S requeridos.

**8.4.2.1.2.3 PRECONDICIONES / DOCUMENTACION DE RESPALDO.**

- ☐ Especificación cantidad de mensajes de E/S requeridos.
- ☐ Resultados de cada una de las evaluaciones.
- ☐ Resultados test de stress.

**8.4.2.1.2.4 EJEMPLOS.**

Simular una condición donde un sistema de Chat alcance una situación de carga máxima, por ejemplo de usuarios conectados todos interactuando con el sistema concurrentemente.

**8.4.2.1.2.5 GLOSARIO.**

Es aplicable a programas que trabajan bajo cargas variables, interactivos, de tiempo real y de control de proceso.

**8.4.2.1.3 I/O related errors.**

Frecuencia con la que el usuario encuentra problemas en operaciones relacionadas a los dispositivos de E/S.

**8.4.2.1.3.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alto (CD), Medio (E).

**8.4.2.1.3.2 VALIDACIONES.**

Fórmula válida. Mide el número de mensajes de warnings o de fallas de sistema recibidos durante un tiempo de observación donde el usuario opera el sistema.

**8.4.2.1.3.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

- ☐ Histórico de mediciones en igualdad de condiciones.

**8.4.2.1.3.4 EJEMPLOS.**

Ejecutar concurrentemente un gran número de tareas de entradas/salidas tanto a dispositivos en buen funcionamiento como con mal funcionamiento (mecánico o eléctrico, por ejemplo papel atascado, sector defectuoso en el disco, etc.) durante un tiempo x de observación.

**8.4.2.1.3.5 GLOSARIO.**

N/A.

**8.4.2.1.4 Mean I/O fulfilment ratio.**

Número promedio de mensajes de error y fallas, relacionados a E/S, durante un tiempo especificado y utilización especificada.

**8.4.2.1.4.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Medio (ED), Bajo (C).

**8.4.2.1.4.2 VALIDACIONES.**

Fórmula válida. Realiza N evaluaciones, en cada una registra la cantidad de mensajes de error de E/S. Realiza el promedio de todas las mediciones. Mide la cantidad de mensajes de E/S promedio respecto a la cantidad de mensajes de E/S requeridos.

**8.4.2.1.4.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

- ☐ Especificación cantidad de mensajes de E/S requeridos por unidad de tiempo.
- ☐ Especificación de uso para esa unidad de tiempo.
- ☐ Resultados de cada una de las evaluaciones.
- ☐ Promedio del total de evaluaciones.

**8.4.2.1.4.4 EJEMPLOS.**

Durante x periodo de tiempo disparar la carga de solicitudes de escritura en disco para provocar un cuello de botella.

**8.4.2.1.4.5 GLOSARIO.**

N/A.

**8.4.2.1.5 User waiting time of I/O devices utilisation.**

Impacto de la utilización de E/S en el tiempo de espera de un usuario.

**8.4.2.1.5.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alto (CED)

**8.4.2.1.5.2 VALIDACIONES.**

Fórmula válida. Mide el tiempo que el usuario consume esperando la finalización de la operación de E/S.

**8.4.2.1.5.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Mediciones de tiempo de espera para la misma carga de solicitudes para distintos modelos, del mismo dispositivo de E/S, con distintos algoritmos de atención de solicitudes.

**8.4.2.1.5.4 EJEMPLOS.**

Ejecutar concurrentemente un gran número de solicitudes de impresión y medir el tiempo de espera del usuario hasta que recibe su impresión.

**8.4.2.1.5.5 GLOSARIO.**

N/A.

**8.4.2.2 (B) Memory resource utilization.**

**8.4.2.2.1 Maximum memory utilisation.**

Utilización máxima de memoria para la realización de una función.

**8.4.2.2.1.1 IIMPORTANCIA GENERAL / ESCENARIOS.**

Alto (ED), Bajo (C)

**8.4.2.2.1.2 VALIDACIONES.**

Fórmula válida. Realiza N evaluaciones, en cada una registra la cantidad de mensajes de error relacionados con la memoria. Selecciona la evaluación con mayor cantidad de mensajes de error. Mide la cantidad de mensajes de error, relacionados con la memoria, de la evaluación seleccionada respecto a la cantidad de mensajes de error, relacionados con la memoria, requeridos.

### 8.4.2.2.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.

- ☐ Especificación cantidad de mensajes de error, relacionados con la memoria, requeridos.
- ☐ Resultados de cada una de las evaluaciones.

### 8.4.2.2.1.4 EJEMPLOS.

Diseño pobre, muchas asignaciones de memoria sin liberar.

### 8.4.2.2.1.5 GLOSARIO.

N/A.

### 8.4.2.2.2 Mean occurrence of memory error.

Número promedio de mensajes de error y fallas, relacionados con la memoria, durante un tiempo especificado y una carga especificada del sistema.

#### 8.4.2.2.2.1 IMPORTANCIA GENERAL / ESCENARIOS.

Alto (D), Bajo (CE)

#### 8.4.2.2.2.2 VALIDACIONES.

Fórmula válida. Realiza N evaluaciones, en cada una registra la cantidad de mensajes de error relacionados con la memoria. Realiza el promedio de todas las mediciones. Mide la cantidad de mensajes de error, relacionados con la memoria, promedio respecto a la cantidad de mensajes de error, relacionados con la memoria, requeridos.

### 8.4.2.2.2.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.

- ☐ Especificación cantidad de mensajes de error, relacionados con la memoria, requeridos por unidad de tiempo.
- ☐ Especificación de uso de memoria para esa unidad de tiempo.
- ☐ Resultados de cada una de las evaluaciones.
- ☐ Promedio del total de evaluaciones.
- ☐ Resultados de test de stress.

### 8.4.2.2.2.4 EJEMPLOS.

Durante x periodo de tiempo disparar la carga de solicitudes de tratamiento de error para provocar un cuello de botella en el uso de la memoria.

### 8.4.2.2.2.5 GLOSARIO.

Test de stress para la carga dinámica de la memoria, donde una rutina no se carga hasta que se la invoque.

Util cuando se requieren grandes cantidades de código para manejar situaciones que ocurren con poca frecuencia como por ejemplo las rutinas de error.

### 8.4.2.2.3 Ratio of memory error/time.

Cantidad de errores de memoria que experimenta el usuario durante un tiempo especificado y utilización de recursos especificada.

#### 8.4.2.2.3.1 IMPORTANCIA GENERAL / ESCENARIOS.



Alto (DE), Bajo (C).

**8.4.2.2.3.2 VALIDACIONES.**

Fórmula válida. Mide el número de warnings o mensajes de fallas del sistema durante el tiempo de observación donde el usuario opera el sistema.

**8.4.2.2.3.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Histórico de mediciones en igualdad de condiciones.

**8.4.2.2.3.4 EJEMPLOS.**

Ejecutar concurrentemente un gran número de programas que no estén cargados en memoria durante un tiempo x de observación.

**8.4.2.2.3.5 GLOSARIO.**

N/A.

**8.4.2.3 (C) Transmission resource utilization.**

**8.4.2.3.1 Maximum transmission utilisation.**

Utilización máxima de transmisión para la realización de una función.

**8.4.2.3.1.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alto (ED), Bajo (C).

**8.4.2.3.1.2 VALIDACIONES.**

Fórmula Incompleta. Realiza N evaluaciones, en cada una registra la cantidad de mensajes de error relacionados con la transmisión. Selecciona la evaluación con mayor cantidad de mensajes de error. Mide la cantidad de mensajes de error, relacionados con la transmisión, de la evaluación seleccionada respecto a la cantidad de mensajes de error, relacionados con la transmisión, requeridos.

**8.4.2.3.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

- ☐ Especificación cantidad de mensajes de error, relacionados con la transmisión, requeridos.
- ☐ Resultados de cada una de las evaluaciones.
- ☐ Especificación detallada del ambiente.

**8.4.2.3.1.4 EJEMPLOS**

1. Diseño pobre del esquema de red que hace que se sature los canales ante las solicitudes de conexión de operadores.
2. Usuario conectado via VPN que ocupa todo el canal.
3. Varios operadores haciendo copias de archivos de un Server a otro, ocupando cada uno de ellos un porcentaje de la carga del servidor.

**8.4.2.3.1.5 GLOSARIO.**

Es complicado reproducir el ambiente y la carga de los canales de transmisión para realizar mediciones.

**8.4.2.3.2 Media device utilisation balancing.**

Grado de sincronización, entre diferentes medios, en un determinado periodo de tiempo.

**8.4.2.3.2.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Medio (D)

### 8.4.2.3.2.2 VALIDACIONES.

Fórmula Incompleta. Mide el tiempo dedicado de continuo a un recurso respecto al tiempo requerido durante el cual se esperan que distintos medios de comunicación finalicen sus tareas con sincronización.

### 8.4.2.3.2.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.

- ☐ Cantidad de carga a ser distribuida.
- ☐ Reportes sobre alternativas vía hardware.
- ☐ Reportes sobre alternativas vía software.

### 8.4.2.3.2.4 EJEMPLOS.

1. Varias conexiones concurrentes a un webservice.
2. Gran número de solicitudes de conexión que para que sean atendidas en idéntica cantidad de tiempo (hay que configurar la red para que se distribuyan, de manera equitativa, la cantidad de solicitudes de conexión y que se atiendan en paralelo por distintos canales).

### 8.4.2.3.2.5 GLOSARIO.

Se puede resolver por hardware.

Si se resuelve por software, es necesario asistirse con programas que miden la carga de transmisión, lo cuál también es riesgoso porque dos programas distintos deberían poder obtener la misma medición en idénticos ambientes, cosa improbable porque cada programa está programado de una determinada manera haciendo utilización de los recursos de manera diferente.

### 8.4.2.3.3 Mean occurrence of transmission error.

Número promedio de mensajes de error y fallas, relacionados con la transmisión, durante un tiempo especificado y una carga especificada del sistema.

### 8.4.2.3.3.1 IMPORTANCIA GENERAL / ESCENARIOS.

Alto (D), Bajo (E).

### 8.4.2.3.3.2 VALIDACIONES.

Fórmula Incompleta. Realiza N evaluaciones, en cada una registra la cantidad de mensajes de error relacionados con la transmisión. Realiza el promedio de todas las mediciones. Mide la cantidad de mensajes de error, relacionados con la transmisión, promedio respecto a la cantidad de mensajes de error, relacionados con la transmisión, requeridos.

### 8.4.2.3.3.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.

- ☐ Especificación cantidad de mensajes de error, relacionados con la transmisión, requeridos por unidad de tiempo.
- ☐ Especificación de uso de transmisión para esa unidad de tiempo.
- ☐ Resultados de cada una de las evaluaciones.
- ☐ Promedio del total de evaluaciones.
- ☐ Resultados de test de stress.

### 8.4.2.3.3.4 EJEMPLOS.

1. Copias de archivos de red a red interrumpida por problemas de sobrecarga de la red.

2. Envío de mensajes que son rebotados por estar saturada la red.

**8.4.2.3.3.5 GLOSARIO.**

Depende mucho de las características físicas de la red.

**8.4.2.3.4 Mean of transmission error per time.**

Cantidad de mensajes de error relacionados a la transmisión son experimentados durante un tiempo especificado y utilización especificada de un recurso.

**8.4.2.3.4.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alto (D), Bajo (E).

**8.4.2.3.4.2 VALIDACIONES.**

Fórmula Incompleta. Mide el número de warnings o mensajes de fallas del sistema durante el tiempo de observación donde el usuario opera el sistema.

**8.4.2.3.4.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

- ☐ Histórico de mediciones en igualdad de condiciones.

**8.4.2.3.4.4 EJEMPLOS.**

1. Copias de archivos de red a red interrumpida por problemas de sobrecarga de la red durante una x cantidad de tiempo.
2. Envío de mensajes que son rebotados por estar saturada la red durante una x cantidad de tiempo.

**8.4.2.3.4.5 GLOSARIO.**

Depende mucho de las características físicas de la red.

**8.4.2.3.5 Transmission capacity utilisation.**

N/A.

Mide la capacidad del sistema de software de tareas performantes dentro de la capacidad de transmisión esperada.

La performance depende del hardware utilizado.

**8.4.2.3.5.1 IMPORTANCIA GENERAL / ESCENARIOS.**

N/A.

**8.4.2.3.5.2 VALIDACIONES.**

Fórmula válida. Mide la capacidad de transmisión respecto de la especificada en el diseño para ser usada por el sistema durante la ejecución.

**8.4.2.3.5.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

- ☐ Especificaciones de diseño de transmisión esperada.
- ☐ Condiciones de ambiente especificada.

**8.4.2.3.5.4 EJEMPLOS.**

Tiempo que tarda en copiarse un archivo, de un servidor a otro, de x tamaño que debería ser copiado en t cantidad de tiempo.

**8.4.2.3.5.5 GLOSARIO.**

N/A.

### **8.4.3 Compatibilidad en Eficiencia (Efficiency Compliance).**

#### **8.4.3.1 Efficiency Compliance.**

N/A.

Los estándares y/o certificaciones no están difundidos masivamente contra los cuales realizar una comparación.

Son atributos dependientes de muchos factores como para que se pueda determinar con objetividad el grado de cumplimiento con algún estándar o certificación.

##### **8.4.3.1.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Bajo. N/A

##### **8.4.3.1.2 VALIDACIONES.**

Fórmula obsoleta. Se puede utilizar una métrica presencial para indicar si cumple algún estándar y de ser cierto, qué estándares cumple el componente. Ídem para indicar si tiene algún tipo de certificación.

##### **8.4.3.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Estándares.

##### **8.4.3.1.4 EJEMPLOS.**

N/A.

##### **8.4.3.1.5 GLOSARIO.**

Métrica presencial: indica si el atributo está presente en componente o no.

## **8.5 Mantenimiento (Maintainability).**

### **8.5.1 Diagnóstico - Capacidad para ser analizado / diagnosticado (Analysability)**

## GUIA-CALIDAD-00003

	Nombre de la métrica	Propósito de la métrica	Método de aplicación	Formula	Interpretación del valor medido	Tipo de escala métrica	Tipo de medida	Input para la medición	ISO/IEC 12207 Referencias	Audiencia
<b>8.5.1.1</b>	<b><u>Audit trail capability</u></b>  <b><u>Capacidad de traza para auditoría</u></b>	<u>¿Puede el usuario identificar la operación específica que causó la falla?</u>  <u>¿Puede el soporte encontrar fácilmente la operación específica que generó la falla?</u>	<u>Observación del comportamiento del usuario / mantenedor, cuando intenta resolver la falla</u>	<u><math>X = A/B</math></u>  <u>A = Numero de data actualmente grabada durante la operación</u> <u>B = Numero de data planificada a ser grabada lo suficiente para monitorear el status del software durante la operación</u>	<u><math>0 \leq X</math></u>  <u>Cuanto más cercano a 1 es mejor</u>	<u>Absoluta</u>	<u>A , B, X</u>  <u>Contable</u>	<u>Reporte de la resolución del problema</u>  <u>Reporte de Operaciones</u>	<u>5.3 Qualification testing</u>  <u>5.4 Operation</u>  <u>5.5 Maintenance</u>	<u>Desarrollador</u>  <u>Soporte</u>  <u>Operador</u>
<b>8.5.1.2</b>	<b><u>Diagnostic function support</u></b>  <b><u>Soporte funciones diagnostico</u></b>	<u>¿Qué tan capaces son las funciones de diagnostico en soportal el análisis casual?</u>  <u>¿Puede el usuario identificar la operación específica que ha causado la falla?</u>	<u>Observar el comportamiento de los usuarios o mantenedores quienes están tratando de resolver las fallas usando las funciones diagnostico</u>	<u><math>X = A/B</math></u>  <u>A=Numero de fallas las cuales el mantenedor puede diagnosticar (utilizando las funciones de diagnostico) para entender la relación causa-efecto</u>  <u>B= Total de fallas registradas</u>	<u><math>0 \leq X \leq 1</math></u>  <u>Cuanto más cercano a 1 es mejor</u>	<u>Absoluta</u>	<u>A , B, X</u>  <u>Contable</u>	<u>Reporte de la resolución del problema</u>  <u>Reporte de Operaciones</u>	<u>5.3 Qualification testing</u>  <u>5.4 Operation</u>  <u>5.5 Maintenance</u>	<u>Desarrollador</u>  <u>Soporte</u>  <u>Operador</u>
<b>8.5.1.3</b>	<b><u>Failure analysis capability</u></b>	<u>¿Puede el usuario identificar una operación específica</u>	<u>Observar el comportamiento del usuario /</u>	<u><math>X = 1 - A / B</math></u>  <u>A= Numero de</u>	<u><math>0 \leq X \leq 1</math></u>	<u>Absoluta</u>	<u>A , B, X</u>  <u>Contable</u>	<u>Reporte de la resolución del problema</u>	<u>5.3 Qualification</u>	<u>Usuario</u>  <u>Desarrollador</u>

## GUIA-CALIDAD-00003

	<b>Capacidad análisis de las fallas</b>	<u>que causa la falla?</u> <u>¿Puede el usuario fácilmente encontrar la causa de la falla?</u>	<u>mantenedor quien está tratando de resolver las fallas</u>	<u>fallas, las cuales no se encontró la causa</u> <u>B=Total de numero de fallas registradas</u>	<u>Cuanto más cercano a 1 es mejor</u>			<u>Reporte de Operaciones</u>	<u>testing</u> <u>5.4 Operation</u> <u>5.5 Maintenance</u>	<u>or</u> <u>Soporte</u> <u>Operador</u>
<b>8.5.1.4</b>	<b>Failure analysis efficiency</b>  <b>Eficiencia en el análisis de las fallas</b>	<u>¿Puede el usuario eficientemente analizar la causa de la falla?</u>  <u>¿Puede el mantenedor encontrar fácilmente la causa de la falla?</u>	<u>Observación del comportamiento del usuario / mantenedor, cuando intenta resolver la falla</u>	<u><math>X = \text{Sum}(T) / N</math></u>	<u><math>0 \leq X</math></u>  <u>Cuanto menor sea es mejor</u>	<u>Ratio</u>	<u>T Time</u> <u>N Count</u>	<u>Reporte de la resolución del problema</u>  <u>Reporte de Operaciones</u>	<u>5.3 Qualification testing</u> <u>5.4 Operation</u> <u>5.5 Maintenance</u>	<u>Desarrollador</u> <u>or</u> <u>Soporte</u> <u>Operador</u>
<b>8.5.1.5</b>	<b>Status monitoring capability</b>  <b>Capacidad de monitorear el status</b>	<u>¿Puede el usuario identificar la operación específica que causa la falla mediante los datos generados por el monitoreo durante la operación?</u>  <u>¿Puede el mantenedor encontrar fácilmente la causa de la falla mediante la data generada por el monitoreo durante la operación?</u>	<u>Observar el comportamiento de los usuarios o mantenedores quienes están tratando de obtener la información grabada de status durante la operación del software</u>	<u><math>X = 1 - A / B</math></u>  <u>A=Número de casos en que el mantenedor u usuario fallaron al obtener datos del monitoreo.</u>  <u>B= Número de casos en los cuales el mantenedor intenta tomar monitor data status grabada durante la operación del software</u>	<u><math>0 \leq X \leq 1</math></u>  <u>Cuanto más cercano a 1 es mejor</u>	<u>Absoluta</u>	<u>A , B, X</u>  <u>Contable</u>	<u>Reporte de la resolución del problema</u>  <u>Reporte de Operaciones</u>	<u>5.3 Qualification testing</u> <u>5.4 Operation</u> <u>5.5 Maintenance</u>	<u>Usuario</u> <u>Desarrollador</u> <u>or</u> <u>Soporte</u> <u>Operador</u>

#### 8.5.1.1. Audit trail capability.

Capacidad de traza para auditoría.

##### 8.5.1.1.1 IMPORTANCIA GENERAL / ESCENARIOS.

Baja (CED)

Si bien es simple contar desde un log de auditoría la cantidad de datos registrados por la traza, es difícil medir la cantidad de datos que deberían haberse registrado en una ejecución determinada.

##### 8.5.1.1.2 VALIDACIONES

La fórmula es aplicable pero solamente en un contexto donde el software haya sido concebido de manera que contemple el registro de las ejecuciones/datos en logs de auditorías.

La fórmula refleja una relación entre la data almacenada y la data necesaria a almacenar para verificar la auditoría, este número no permite demostrar que la información posee el suficiente detalle para identificar la causa de la falla.

##### 8.5.1.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.

Reportes de Auditoría

Archivos de seguimientos de ejecución

Traza de ejecución

Implica que el software haya sido concebido incluyendo conceptos de traza para auditoría.

##### 8.5.1.1.4 EJEMPLOS.

Un reporte de auditoría, puede almacenar las transacciones principales de un sistema, y la falla estar en una consulta embebida siendo imposible su identificación partiendo de un archivo de seguimiento de ejecución sin el suficiente detalle e información, así como conocimiento del sistema.

De esta manera no siempre estaríamos en condiciones de satisfacer el objetivo de la métrica.

##### 8.5.1.1.5 GLOSARIO.

#### 8.5.1.2 Diagnostic function support

##### 8.5.1.2.1 IMPORTACION GENERAL / ESCENARIOS.

Baja (CED)

##### 8.5.1.2.2 VALIDACIONES.

###### **Redefinición del propósito de la métrica**

¿Qué tan útiles son las funciones de diagnósticos definidas para detectar las causas de las fallas?

Esta métrica evalúa la cantidad de fallas que pudieron ser diagnosticadas (por las diagnostic functions) sobre el total de fallas registradas.

##### 8.5.1.2.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.

Reportes de Auditoría

- ❑ Archivos de seguimientos de ejecución (logs con resultados de las diagnostic functions)

❑ Traza de ejecución

Reporte de Operación: el mismo debe contar con el detalle suficiente de las operaciones realizadas para reproducir la falla.

**8.5.1.2.1 EJEMPLOS.**

NA

**8.5.1.2.2 GLOSARIO.**

Diagnostic Functions son funciones específicas del sistema definidas para poder evaluar el comportamiento del sistema en operación determinadas, a través de ellas podríamos rescatar información para identificar los puntos donde ocurren las fallas del sistema.

**8.5.1.3 Failure analysis capability**

Capacidad de análisis de fallas

**8.5.1.3.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Baja (CED)

Esta métrica original relaciona el número de fallas reportadas para las que aún no se han encontrado sus causas, con el número total de fallas registradas.

**Redefinición de la métrica**

Medio

La métrica que proponemos relaciona el número de fallas reportadas para las que se han encontrado sus causas, con el número total de fallas registradas.

**8.5.1.3.2 VALIDACIONES.**

**Redefinición de la métrica**

$$X = A / B$$

A = número de fallas para las que se encontraron sus causas

B = número total de fallas registradas

Cuanto más cercano al número uno, implica que a la mayoría de las fallas se le identificó la operación que las originó, por ende es mejor.

**8.5.1.3.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Reporte de las fallas: a fin de cuantificar la cantidad de fallas.

Reporte de Resolución: a fin de cuantificar la cantidad de fallas que se han resuelto identificando sus causas de manera de obtener la diferencia entre cantidad total de fallas y las resueltas con causas identificadas.

**8.5.1.3.4 EJEMPLOS.**

Mantener un registro en un tracker de las fallas reportadas, estado de resolución, causa identificada. El input del tracker sería el Report Operation y el Problem Resolution Report. Luego, se podrá realizar consultas sobre la base de tracker a fin de obtener la información cuantificada para la métrica.



**8.5.1.3.5 GLOSARIO.**

Operation Report= reporte de falla. Puede contener el detalle de la operación que se estaba realizando.

Problem Resolution report = Reporte de resolución del problema. Indica la resolución de una falla.

**8.5.1.4 Failure analysis efficiency.**

Eficiencia en el análisis de fallas

**8.5.1.4.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Bajo (CED)

**8.5.1.4.2 VALIDACIONES.**

Difícilmente aplicable debido a que en raras oportunidades se registra discriminadamente los esfuerzos insumidos en las tareas de análisis y resolución. En general, el tiempo insumido en ambas tareas se registra como un todo.

La métrica refleja el tiempo promedio de análisis de las causas por falla resuelta, cuanto menor sea dicho tiempo, más eficiente es el análisis de la falla.

La especificación de la métrica, no indica que representa la variable N por lo cual asumimos N como el número de fallas resueltas y SUM(T) como el total del tiempo transcurrido o esfuerzo en horas-hombre insumido en analizar la falla para aquellas que fueron resueltas.

**8.5.1.4.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Se requiere reporte de horas insumidas en análisis de cada falla.

Operation Report= reporte de falla. Puede contener el detalle de la operación que se estaba realizando.

Problem Resolution report = Reporte de resolución del problema. Indica la resolución de una falla.

**8.5.1.4.4 EJEMPLOS.**

NA

**8.5.1.4.5 GLOSARIO.**

Operation Report= reporte de falla. Puede contener el detalle de la operación que se estaba realizando.

Problem Resolution report = Reporte de resolución del problema. Indica la resolución de una falla.

#### **8.5.1.5 Status monitoring capability.**

Capacidad de monitoreo de estados.

##### **8.5.1.5.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alto (CED)

##### **8.5.1.5.2 VALIDACIONES.**

El propósito de la métrica establece si el usuario puede identificar operaciones específicas de fallo en el monitoreo de datos durante la operación.

Primero se deberá determinar si el producto de sw tiene funcionalidad de monitoreo y seguimiento incorporada de datos de cada función según se requiera. Y en caso de que posea, al ejecutarla si hay fallos o errores considerando situaciones normales y casos limites (pruebas de volumen y stress)

##### **8.5.1.5.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

- ☐ Manual de Usuario.
- ☐ Manual de Operación.
- ☐ Manual de Mantenimiento.

##### **8.5.1.5.4 EJEMPLOS.**

Una transacción que efectúa un vuelco de información masiva al host (por operar fuera de línea), por lo tanto la misma deberá considerar el monitoreo del proceso, informando status de cada transacción, performance adecuada.

##### **8.5.1.5.5 GLOSARIO.**

### **8.5.2 Cambios - Capacidad de ser cambiado / modificado (Changeability)**

## GUIA-CALIDAD-00003

	<u>Nombre de la métrica</u>	<u>Propósito de la métrica</u>	<u>Método de aplicación</u>	<u>Formula</u>	<u>Interpretación del valor medido</u>	<u>Tipo de escala métrica</u>	<u>Tipo de medida</u>	<u>Input para la medición</u>	<u>ISO/IEC 12207 Ref.</u>	<u>Audiencia</u>
<b>8.5.2.1</b>	<b><u>Change Cycle efficiency</u></b> <b><u>Eficiencia en el ciclo de cambio</u></b>	<u>¿Pueden los problemas del usuario ser resueltos según sus expectativas/ satisfacción en una escala de tiempo aceptables?</u>	<u>Controlar la interacción entre el usuario y el proveedor.</u>  <u>Grabar el tiempo que demanda desde el requerimiento inicial del usuario hasta la resolución del problema.</u>	<u>Promedio del tiempo</u> $Tav = \text{Sum}(Tu) / N$  $Tu = Trc - Tsn$  <u>Tsn = Tiempo en el cual el usuario envía el requerimiento al proveedor para mantenimiento con el problema</u>  <u>Trc = Tiempo en el cual el usuario recibe la versión revisada o el reporte de status</u>  <u>N = nro. de versiones</u>	<u><math>0 &lt; Tav</math></u>  <u>El tiempo más pequeño es el mejor, excepto que el numero de versiones revisadas sea alto</u>	<u>Ratio</u>	<u>Tav, Tu</u> <u>Tiempo</u>  <u>N</u> <u>Contable</u>	<u>Reporte de la resolución del problema</u>  <u>Reporte de mantenimiento</u>  <u>Reporte de Operaciones</u>	<u>5.3 Qualification testing</u>  <u>5.4 Operation</u>  <u>5.5 Maintenance</u>	<u>Desarrollador</u>  <u>Soporte</u>  <u>Operador</u>
<b>8.5.2.2</b>	<b><u>Change implementation elapse time</u></b> <b><u>Tiempo transcurrido en la implementación del cambio</u></b>	<u>¿Puede el usuario fácilmente introducir un cambio en el software para resolver la falla?</u>	<u>Observar el comportamiento de los usuarios o mantenedores mientras tratan de generar el cambio. Investigar reportes de resolución de problemas.</u>	<u>Promedio del tiempo</u> $Tav = \text{Sum}(Tm) / N$  $Tm = Tout - Tin$  <u>Tout = Tiempo en el que las fallas son removidas con los cambios del software</u>  <u>Tin= Time en el cual las causas de las fallas se hallaron</u>  <u>N = nro. de fallas</u>	<u><math>0 &lt; Tav</math></u>  <u>El tiempo más pequeño es el mejor, excepto que el numero de versiones revisadas sea alto</u>	<u>Ratio</u>	<u>Tav, Tu</u> <u>Tiempo</u>  <u>N</u> <u>Contable</u>	<u>Reporte de la resolución del problema</u>  <u>Reporte de mantenimiento</u>  <u>Reporte de Operaciones</u>	<u>5.3 Qualification testing</u>  <u>5.4 Operation</u>  <u>5.5 Maintenance</u>	<u>Desarrollador</u>  <u>Soporte</u>  <u>Operador</u>

## GUIA-CALIDAD-00003

<b>8.5.2.3</b>	<b><u>Modification complexity</u></b>  <b><u>Complejidad del cambio</u></b>	<u>¿Puede el mantenedor fácilmente realizar el cambio en el software para resolver el problema?</u>	<u>Observar el comportamiento del mantenedor, quien es el que está tratando de hacer un cambio en el software.</u>  <u>Investigar los reportes de resolución de problemas o reportes de mantenimiento</u>	<u><math>T = \text{Sum} (A/B) / N</math></u>  <u>A = Trabajo insumido en el cambio</u>  <u>B= Tamaño del cambio</u>  <u>N = Nro de cambios</u>	<u><math>0 &lt; T_{av}</math></u>  <u>El tiempo más pequeño es el mejor.</u>	<u>Ratio</u>	<u>A y T</u> <u>Tiempo</u>  <u>B Tamaño</u>  <u>N</u> <u>Contable</u>	<u>Reporte de la resolución del problema</u>  <u>Reporte de mantenimiento</u>  <u>Reporte de Operaciones</u>	<u>5.3</u> <u>Qualificati</u> <u>on testing</u>  <u>5.4</u> <u>Operation</u>  <u>5.5</u> <u>Maintenan</u> <u>ce</u>	<u>Desarrolla</u> <u>dor</u>  <u>Soporte</u>  <u>Operador</u>
<b>8.5.2.4</b>	<b><u>Parameterised modifiability</u></b>  <b><u>Modificación a través de la parametrización</u></b>	<u>¿Puede el usuario del mantenedor modificar fácilmente los parámetros para genera un cambio en el software y resolver las fallas?</u>	<u>Observar el comportamiento de los usuarios o mantenedores mientras trata de generar el cambio. Investigar reportes de resolución de problemas.</u>	<u><math>X = 1 - A / B</math></u>  <u>A= Nro de casos en los cuales el mantenedor falla en el cambio utilizando parámetros</u>  <u>B= Nro de casos en los cuales el mantenedor intenta hacer un cambio utilizando parámetros</u>	<u><math>0 \leq X \leq 1</math></u>  <u>El más cercano a 1 es el mejor</u>	<u>Absoluta</u>	<u>A, B, X</u>  <u>Contable</u>	<u>Reporte de la resolución del problema</u>  <u>Reporte de mantenimiento</u>  <u>Reporte de Operaciones</u>	<u>5.3</u> <u>Qualificati</u> <u>on testing</u>  <u>5.4</u> <u>Operation</u>  <u>5.5</u> <u>Maintenan</u> <u>ce</u>	<u>Desarrolla</u> <u>dor</u>  <u>Soporte</u>  <u>Operador</u>
<b>8.5.2.5</b>	<b><u>Software Change Control Capability</u></b>  <b><u>Capacidad de Control del Cambio en el Software</u></b>	<u>¿Puede el usuario identificar fácilmente las versiones?</u>  <u>¿Puede el mantenedor fácilmente cambiar el software para resolver las fallas?</u>	<u>Observar el comportamiento de los usuarios o mantenedores mientras trata de generar el cambio. Investigar reportes de resolución de problemas.</u>	<u><math>X = A / B</math></u>  <u>A=Nro. De datos en el log de cambios que se almaceno.</u>  <u>B=Nro de datos que estaba planificado almacenar necesario para generar una trza de cambios.</u>	<u><math>0 \leq X \leq 1</math></u>  <u>El más cercano a 1 es el mejor</u>	<u>Absoluta</u>	<u>A, B, X</u>  <u>Contable</u>	<u>Manual del usuario</u>  <u>Reporte de la resolución del problema</u>  <u>Reporte de mantenimiento</u>  <u>Reporte de Operaciones</u>	<u>5.3</u> <u>Qualificati</u> <u>on testing</u>  <u>5.4</u> <u>Operation</u>  <u>5.5</u> <u>Maintenan</u> <u>ce</u>	<u>Desarrolla</u> <u>dor</u>  <u>Soporte</u>  <u>Operador</u>

#### 8.5.2.1 Chance cycle efficiency

Eficiencia en el ciclo de cambio.

##### 8.5.2.1.1 **IMPORTANCIA GENERAL / ESCENARIOS.**

Bajo (CED)

**Redefinición de métrica**

Alto (CED)

##### 8.5.2.1.2 **VALIDACIONES.**

###### **Métrica Original**

La métrica refleja el tiempo promedio de resolución en relación a las **versiones revisadas**.

Tiempo promedio de resolución, desde que el cliente reporta la falla hasta que se le resuelve la misma.

Este tiempo cuanto menor sea, mejora la escala de tiempo que se insume para implementar un cambio.

###### **Redefinición de métrica**

Más que reflejar la eficiencia en el ciclo de cambio, calcularía el tiempo promedio de resolución de fallas considerando tiempo promedio transcurrido desde que se reporta la falla hasta que se obtiene como respuesta el release correspondiente a la resolución de la falla. En este caso, modificaríamos la fórmula de manera de reflejarla como:

Average Time :  $T_{av} = \text{Sum}(T_u) / N$

Dónde:

$T_u = T_{rc} - T_{sn}$

$T_{rc}$ = Momento de recepción del release con la resolución (día y hora)

$T_{sn}$ = Momento de envío del reporte de falla (día y hora)

Consideramos que es más aplicable contabilizar el tiempo que transcurre desde que se reporta la falla hasta que se entrega la versión corregida, y promediarlo por el total de fallas reportadas. Si consideramos este punto podríamos tratarla con Importancia ALTA.

##### 8.5.2.1.3 **PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Reporte de Operación: el mismo debe contar con el detalle de las operaciones de forma tal de identificar la falla y por ende la operación.

Registro de resoluciones recibidas.

Ambos reportes deben contener el detalle de fecha y hora para poder calcular el tiempo transcurrido, idealmente esta información debería estar almacenada en un tracker (podríamos considerarlo como el reporte de mantenimiento)

##### 8.5.2.1.4 **EJEMPLOS.**

NA

##### 8.5.2.1.5 **GLOSARIO.**

Operation Report= reporte de falla. Puede contener el detalle de la operación que se estaba realizando.

Problem Resolution report = Reporte de resolución del problema. Indica la resolución de una falla.

Maintenance Report = Reporte de mantenimiento, que incluye reporte de la falla y el registro de solución.

#### **8.5.2.2 Change implementation elapse time**

Tiempo transcurrido en la implementación del cambio

##### **8.5.2.2.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Bajo. Consideramos que con la métrica anterior podríamos estar cubriendo estos aspectos.

##### **8.5.2.2.2 VALIDACIONES.**

No es fácilmente aplicable dado que es difícil delimitar el tiempo transcurrido desde que se conoce la causa de la falla hasta su resolución, ya que en general se trata el análisis de la causa y su resolución como un todo.

La métrica refleja cuanto tiempo demanda implementar un cambio exitosamente habiendo detectado ya la causa de la falla, en relación a las fallas registradas y solucionadas. El tiempo evaluado es desde que se identifica la solución del problema hasta la resolución del mismo. Cuanto menor sea el tiempo, se asegura facilidad de cambio.

##### **8.5.2.2.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

- ☐ Registro de horas
- ☐ Reporte de Error
- ☐ Doc / Reporte de cambios

##### **8.5.2.2.4 EJEMPLOS.**

##### **8.5.2.2.5 GLOSARIO.**

Operation Report= reporte de falla. Puede contener el detalle de la operación que se estaba realizando.

Problem Resolution report = Reporte de resolución del problema. Indica la resolución de una falla.

#### **8.5.2.3 Modification complexity**

##### **8.5.2.3.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alto (CED)

##### **8.5.2.3.2 VALIDACIONES.**

Es aplicable.

La métrica refleja una la relación entre insumidos, tamaño del software y el número de cambios realizados.

Para aplicarla se deberá definir una unidad de medida de "tamaño del software" (línea de código, requerimientos, puntos de función, transacciones de casos de uso, etc.)

Asumimos que el tiempo empleado en la fórmula corresponde al que transcurre una vez identificado el problema y hasta que se realiza el cambio que lo resuelve.

**8.5.2.3.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

- ☐ Reporte de falla
- ☐ Doc / Reporte de cambios
- ☐ Registro de horas insumidas en las tareas.
- ☐ Registro de cambios realizados para resolver problemas.
- ☐ Tamaño del software identificado.

**8.5.2.3.4 EJEMPLOS.**

NA

**8.5.2.3.5 GLOSARIO.**

Operation Report= reporte de falla. Puede contener el detalle de la operación que se estaba realizando.

Problem Resolution report = Reporte de resolución del problema. Indica la resolución de una falla.

Maintenance Report = Reporte de Mantenimiento que contenga los cambios realizados a fin de poder contabilizarlos.

**8.5.2.4 Parametrised modifiability**

**8.5.2.4.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Bajo. / NA

La métrica refleja la capacidad de resolución de fallas a través de modificación de parámetros.

**8.5.2.4.2 VALIDACIONES.**

Interpretamos que la resolución de una falla no debería ser mediante la modificación un parámetro, la utilización del mismo es justamente para hacer que una aplicación acepte un cambio fácilmente pero no con el fin de la corrección de una falla, sino con el fin de cambiar su comportamiento funcional.

Además, la métrica requiere contar la cantidad de intentos fallidos para resolver los problemas a través de parámetros actividad que no resulta habitual. En general se llevan a cabo todas las actividades necesarias para resolver una falla y se registran a lo sumo los insumidos en toda la operación y el resultado de la resolución, pero no es una práctica que resulte habitual registrar los intentos fallidos, a menos que la verificación de la corrección pase por un ciclo de testing.

**8.5.2.4.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Manual de Usuario o donde se reflejen los parámetros del sistema

**8.5.2.4.4 EJEMPLOS.**

NA

**8.5.2.4.5 GLOSARIO.**

NA

**8.5.2.5 Software change control capability.**

Capacidad de control de cambios del software.

NA

Se entiende que los componentes, fuentes, paquetes, etc deberían ser llevados bajo un repositorio común con control de cambios y tags de las versiones liberadas, branches de versiones internas en desarrollo, etc. Con lo cual es llevado con un herramienta externa, ej. Subversión, CVS contemplando una gestión de Configuración. También documentación correspondiente entregada al cliente, procedimientos de actualización de versiones, releases.



### **8.5.3 Estabilidad (Stability)**

## GUIA-CALIDAD-00003

	<u>Nombre de la métrica</u>	<u>Propósito de la métrica</u>	<u>Método de aplicación</u>	<u>Formula</u>	<u>Interpretación del valor medido</u>	<u>Tipo de escala métrica</u>	<u>Tipo de medida</u>	<u>Input para la medición</u>	<u>ISO/IEC 12207 Ref.</u>	<u>Audiencia</u>
8.5.3.1	<b><u>Change Success Ratio</u></b>  <b><u>Tasa de éxito del cambio</u></b>	¿Puede el operador utilizar el sistema sin fallas después de un cambio?  ¿Puede el mantenedor fácilmente mitigar las fallas causadas por un mantenimiento?	Observación del comportamiento del usuario o mantenedor quien está operando el software después del mantenimiento  Contabilizar las fallas, las cuales el usuario o mantenedor encuentran antes y después del cambio.  Investigar los reportes de resolución de problemas, reportes de mantenimiento u operación	$X = Na / Ta$  $Y = X / (Nb/Tb)$  <u>Na = nro. De casos en el que el usuario encontró fallas durante la operación después de hacer el cambio</u>  <u>Nb = nro de casos en los cuales el usuario encuentre fallas durante la operación antes de hacer el cambio</u>  <u>Ta Tiempo de operación estimado como de observación después del cambio</u>  <u>Tb Tiempo de operación de estimado como de observación antes del cambio</u>	$0 \leq X, Y$  Cuanto más pequeño y cercano a cero es mejor	Ratio	<u>Na,Nb Contable</u>  <u>Ta,Tb Tiempo</u>  $X = \text{Contable} / \text{tiempo}$  $Y = \frac{\text{Cont/Tiempo}}{\text{Cont} / \text{tiempo}}$	<u>Reporte de resolución de problemas</u>  <u>Reporte de mantenimiento o</u>  <u>Reporte de Operaciones</u>	<u>5.3 Qualification testing</u>  <u>5.4 Operation</u>  <u>5.5 Maintenance</u>	<u>Desarrollador</u>  <u>Soporte</u>  <u>Operador</u>
8.5.3.2	<b><u>Modification impact localisation</u></b>	¿Puede el usuario operar el sistema sin fallas después del mantenimiento?  ¿Puede el mantenedor mitigar fácilmente las fallas causadas por mantenimiento como defecto secundario?	Contar las fallas ocurridas después del cambio, las cuales son encadenadas y afectadas por el cambio	$X = A / N$  <u>A = nro. Fallas que surgieron después de resolver un cambio durante un tiempo especific.</u>  <u>N = nro. de fallas resueltas</u>	$0 \leq x$  Cuando mas pequeño y cercano a 0 es mejor	Absoluta	<u>A,N,X Contable</u>	<u>Reporte de resolución de problemas</u>  <u>Reporte de Operaciones</u>	<u>5.3 Qualification testing</u>  <u>5.4 Operation</u>  <u>5.5 Maintenance</u>	<u>Desarrollador</u>  <u>Soporte</u>  <u>Operador</u>

## GUIA-CALIDAD-00003

---

--	--	--	--	--	--	--	--	--	--	--

### 8.5.3.1 Change success ratio.

Tasa de éxito del cambio.

#### 8.5.3.1.1 IMPORTANCIA GENERAL / ESCENARIOS.

Alto (CED)

#### 8.5.3.1.2 VALIDACIONES.

Es aplicable.

La métrica refleja la "Tasa de éxito de cambio".

La frase "Number of cases which user encounters failures during operation..." la redefinimos como "Número de fallas identificadas...".

Na = Número de fallas identificadas **después** que ocurrió el cambio.

Nb = Número de fallas identificadas **antes** que ocurra el cambio.

Ta = Tiempo de operación durante un período de observación determinado, luego del cambio.

Tb = Tiempo de operación durante un período de observación determinado, antes del cambio.

Por lo tanto:

$$X = Na / Ta$$

$$Y = X / (Nb/Tb)$$

Es decir, la métrica estaría considerando la relación entre las fallas que ocurren antes y después del cambio (antes / después), de manera de identificar si las fallas van disminuyendo luego de los cambios.

Cuanto más se acerca a cero X (antes) e Y (la relación entre antes/después) es mejor porque implica que hubo pocas fallas después del cambio.

#### 8.5.3.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.

- ☐ Registro de horas insumidas para las observaciones pre y post cambios.
- ☐ Reporte de fallas por cada versión.

#### 8.5.3.1.4 EJEMPLOS.

NA

#### 8.5.3.1.5 GLOSARIO.

NA

### 8.5.3.2 Modification impact localisation (Emerging failure after change).

#### 8.5.3.2.1 IMPORTANCIA GENERAL / ESCENARIOS.

Alto (CED)

#### 8.5.3.2.2 VALIDACIONES.

Es aplicable.

La métrica refleja las fallas que se generan a raíz de haber resuelto otras fallas en un determinado periodo.

**8.5.3.2.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

- ☐ Herramienta de seguimiento de fallas
- ☐ Reportes de fallas
- ☐ Reportes de resolución de fallas.

**8.5.3.2.4 EJEMPLOS.**

NA

**8.5.3.2.5 GLOSARIO.**

Operation Report= reporte de falla. Puede contener el detalle de la operación que se estaba realizando.

Problem Resolution report = Reporte de resolución del problema. Indica la resolución de una falla.

#### **8.5.4 Testing – Capacidad de ser testeado (Testability).**

## GUIA-CALIDAD-00003

	<u>Nombre de la métrica</u>	<u>Propósito de la métrica</u>	<u>Método de aplicación</u>	<u>Formula</u>	<u>Interpretación del valor medido</u>	<u>Tipo de escala métrica</u>	<u>Tipo de medida</u>	<u>Input para la medición</u>	<u>ISO/IEC 12207 Ref.</u>	<u>Audiencia</u>
8.5.4.1	<b><u>Availability of built-in test function</u></b>  <b><u>Capacidad para testear una function</u></b>	<u>¿Puede un usuario y mantenedor realizar un testing operar fácilmente sin una preparación para el test ?</u>	<u>Observación del comportamiento del usuario o mantenedor quien está testeando el software después del mantenimiento</u>	$X = A / B$  <u>A = nro. De casos en los cuales el mantenedor puede usar suitably test function</u>  <u>B= nro. de casos de oportunidades de test</u>	$0 \leq X \leq 1$  <u>Cuanto mas cercano a uno es mejor</u>	<u>Absolute</u>	<u>A , B , X Contable</u>	<u>Reporte de resolución de problemas</u>  <u>Reporte de Operaciones</u>	<u>5.3 Qualification testing</u>  <u>5.4 Operation</u>  <u>5.5 Maintenance</u>	<u>Desarrollador</u>  <u>Soporte</u>  <u>Operador</u>
8.5.4.2	<b><u>Re Test effincey</u></b>  <b><u>Eficiencia en el re-test</u></b>	<u>Puede el usuario y mantenedor fácilmente realizar un test operacional y determinar cuando el software esta listo para operar o no?</u>	<u>Observación del comportamiento del usuario o mantenedor quien está testeando el software después del mantenimiento</u>	$X = \text{Sum}(T) / N$  <u>T = tiempo que se invierte en testear para asegurarse que una falla reportada fue solucionada</u>  <u>N= nro. de fallas resueltas</u>	$0 < X$  <u>Cuanto mas pequeño es mejor</u>	<u>Ratio</u>	<u>T tiempo N Contable</u>  <u>X= Tiempo / contable</u>	<u>Reporte de resolución de problemas</u>  <u>Reporte de Operaciones</u>	<u>5.3 Qualification testing</u>  <u>5.4 Operation</u>  <u>5.5 Maintenance</u>	<u>Desarrollador</u>  <u>Soporte</u>  <u>Operador</u>
8.5.4.3	<b><u>Test Restartability</u></b>  <b><u>Test que se puede retomar</u></b>	<u>Puede el usuario u mantenedor fácilmente realizar un test operacional con puntos de chequeo luego del mantenimiento</u>	<u>Observación del comportamiento del usuario o mantenedor quien está testeando el software después del mantenimiento</u>	$X = A / B$  <u>A = nro. De casos en los cuales el mantenedor puede comenzar y parar un test en los puntos deseados para verificar passo por paso</u>  <u>B= nro. de casos que pausan en la ejecución de un test</u>	$0 \leq X \leq 1$  <u>Cuanto mas cercano a uno es mejor</u>	<u>Absolute</u>	<u>A , B , X Contable</u>	<u>Reporte de resolución de problemas</u>  <u>Reporte de Operaciones</u>	<u>5.3 Qualification testing</u>  <u>5.4 Operation</u>  <u>5.5 Maintenance</u>	<u>Desarrollador</u>  <u>Soporte</u>  <u>Operador</u>

#### 8.5.4.1 Availability of built-in test function

##### 8.5.4.1.1 IMPORTANCIA GENERAL / ESCENARIOS.

Baja. (ED)

##### 8.5.4.1.2 VALIDACIONES.

Vemos dificultoso en la práctica llevar registro de las built-in test function empleadas en cada test opportunity.

##### 8.5.4.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.

- ☐ Registro de las built-in tests empleadas por cada test opportunity.
- ☐ Registro de Test Opportunities presentadas.

##### 8.5.4.1.4 EJEMPLOS.

NA

##### 8.5.4.1.5 GLOSARIO.

Test Opportunities: Momentos en el tiempo, en que se necesitan ejecutar tests.

Built-in Test function: Tares previas requeridas para poder cumplir con las Test Opportunities.

#### 8.5.4.2 Re-test efficiency.

##### 8.5.4.2.1 IMPORTANCIA GENERAL / ESCENARIOS.

Alta (CED)

##### 8.5.4.2.2 VALIDACIONES.

La métrica refleja el tiempo consumido en el testing en relación a las fallas resueltas. (tiempo promedio de testing por falla resuelta)

Tener presente que esta medición es únicamente sobre los N de fallas **resueltas exitosamente**

##### 8.5.4.2.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.

- ☐ Registro de horas.
- ☐ Registro de fallas y sus resultados.

##### 8.5.4.2.4 EJEMPLOS.

NA

##### 8.5.4.2.5 GLOSARIO.

NA

#### 8.5.4.3 Test restartability

##### 8.5.4.3.1 IMPORTANCIA GENERAL / ESCENARIOS.

Baja (CED)



Interpretamos que el propósito de la métrica es evaluar la capacidad del software para la ejecución en un determinado punto y poder reiniciarla a partir del mismo, sin necesidad de recomenzar toda la operación.

#### **8.5.4.3.2VALIDACIONES.**

Como resultado de una operación de test, no resulta habitual registrar las veces que se reinicia la operación desde su comienzo o se retoma desde un check point.

#### **8.5.4.3.3PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Reporte de ejecución de la operación.

#### **8.5.4.3.4EJEMPLOS.**

NA

#### **8.5.4.3.5GLOSARIO.**

Operational Testing: Testing funcional (de caja negra).

**8.5.5 Compatibilidad en Mantenimiento (Maintainability Compliance).**

## GUIA-CALIDAD-00003

	<u>Nombre de la métrica</u>	<u>Propósito de la métrica</u>	<u>Método de aplicación</u>	<u>Formula</u>	<u>Interpretación del valor medido</u>	<u>Tipo de escala métrica</u>	<u>Tipo de medida</u>	<u>Input para la medición</u>	<u>ISO/IEC 12207 Ref.</u>	<u>Audiencia</u>
8.5.5.1	<u>Maintainability compliance</u>	<u>Como puede cumplir con determinadas reglas un producto al que se le aplica standares y convenciones?</u>	<u>Contar el numero de ítems que requieren cumplir con las normas que se han encontrado y comparar con el numero de ítems que requieren cumplir con las especificación</u>	$X = 1 - A / B$ <p>A= Numero de ítems que cumplen con las normas de mantenibilidad que no se han encontrado durante el testeo</p> <p>B= Total de numero de ítems que cumplen con las normas de mantenibilidad especificadas</p>	$0 \leq X \leq 1$ <p>Cuanto mas cercano a 1.0 es mejor</p>	<u>Abosluta</u>	<u>A,B,X Contable</u>	<u>Descripción del producto</u> <u>Manual del usuario</u> <u>Especificaciones técnicas</u> <u>Standares aplicados</u> <u>Especificacion del testing</u>	<u>5.3 Qualification testing</u> <u>6.5 Validation</u>	<u>Proveedor</u> <u>Usuario</u>

#### 8.5.5.1 Maintainability compliance.

##### 8.5.5.1.1 IMPORTANCIA GENERAL / ESCENARIOS.

Alto (CED)

Refleja la relación entre los ítems que adhieren a los estándares/convenciones y el total de ítems que debieran adherir.

##### 8.5.5.1.2 VALIDACIONES.

La fórmula aplica tal como está definida.

##### 8.5.5.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.

Redefinimos el input de la métrica como:

Especificaciones de Calidad a Cumplir (Item – Convencion a cumplir por el ítem)

Registro del Control (checklist) con los resultados de la adherencia.

##### 8.5.5.1.4 EJEMPLOS.

NA

##### 8.5.5.1.5 GLOSARIO.

NA

### 8.6 Portabilidad (Portability).

#### 8.6.1 Adaptabilidad (Adaptability).

##### 8.6.1.1 Adaptability of data structures.

Adaptabilidad de estructuras de datos.

##### 8.6.1.1.1 IMPORTANCIA GENERAL / ESCENARIOS.

Alta (CED)

##### 8.6.1.1.2 VALIDACIONES.

Métrica

Pone foco en las estructuras de datos luego de la adaptación.

Calcula la relación entre número de datos que se encuentran operables luego de la adaptación del software a un ambiente operacional sobre el total de datos que debieran estar disponibles.

Se entiende por datos: archivos de datos, bases de datos, que deben ser adaptados a diferentes volúmenes de datos.

$X = A/B$

A= Numero de datos que se encuentran operables

B= Numero de Datos que se esperan estén operables

**Métrica adicional**

Según nuestro análisis, en algunos casos donde las bases de datos son de gran volumen o las estructuras de datos resultan muy complejas es probable que sea difícil implementar esta métrica.

Como alternativa, se propone la siguiente:

**Metodo de Aplicación:** Realizar un pasaje de ambiente y un Smoke Test para evaluar estabilidad de la aplicación (se provee uno de ejemplo)

**Formula:**  $X = 1 - (A / B)$

Donde

A = Numero de casos fallidos del Smoke Test atribuibles a problemas de estructura de datos

B = Numero total de casos del Smoke Test

**Interpretacion de la métrica:**  $0 \leq X \leq 1$  Cuanto mas cercano a 1 es mejor

**Escala de Métrica:** Absoluta

Tipo Medicion:

A = Contar

B = Contar

$X = 1 - (\text{Contar}/\text{Contar})$

**Entrada a la medicion:** Reporte de Smoke test

#### **8.6.1.1.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Se deberá contar con la siguiente documentación actualizada:

- ☐ Documentacion de Diseño de Base de Datos
- ☐ Diagrama de Entidad Relación
- ☐ Layout de las tablas
- ☐ Log del despliegue
- ☐ Documentación funcional
- ☐ Manual de usuario
- ☐ Manual de instalación
- ☐ Template de Smoke Test usado por la organizacion

#### **8.6.1.1.1.4 EJEMPLOS.**

Despliegue de version de un ambiente a otro (diferentes ambientes de testing)

#### **8.6.1.1.1.5 GLOSARIO.**

Smoke Test: Permite determinar si la versión está en condiciones para que sean iniciados los esfuerzos de testing

### **8.6.1.2 Hardware environmental adaptability.**

Adaptabilidad en Hardware.

#### **8.6.1.2.1 IMPORTANCIA GENERAL / ESCENARIOS**

Alta (CED)

#### **8.6.1.2.2 VALIDACIONES.**

Métrica

Pone foco en el resultado de las pruebas luego del cambio de hardware

$$X = 1 - A/B$$

A = Número de funciones operacionales de las cuales algunas tareas no se completaron o no cumplieron nivel adecuado durante el testing debido al ambiente de hardware

B = Número total de funciones testeadas

Observación: Las alternativas de Hardware son numerosas y variables de manera que el equipo de Laboratorio deberá determinar, de acuerdo al producto, tecnología, necesidades de hardware, etc. distintas opciones de prueba de hardware debidamente justificadas. Esto debería ser compatible con la disponibilidad de hardware en el laboratorio, o en su defecto ser provista por la organización evaluada.

Dado que algunos productos pueden ser probados en un único hardware por restricciones (ejemplo mainframe, etc) mientras que otros tienen distintas variantes pudiéndose asumir que x alternativas son representativas, etc, estas decisiones deberán ser debidamente justificadas.

Como alternativa a esta métrica que contempla la prueba de todas las funciones operacionales, se proponen dos alternativas. La primera es una simplificación de la existente (OPCION I) y la segunda (OPCION II) nuevamente está asociada a un Smoke Test dado que este tipo de prueba contempla chequear distintos aspectos del software para evaluar estabilidad.

### OPCION I

Idéntica a la métrica propuesta en el documento con la siguiente observación sobre la prueba.

Seleccionar del set de pruebas funcionales (provisto por la organización o del set armado en el laboratorio) distintos casos que sean representativos de las distintas funcionalidades de manera de tener un set más reducido, pero representativo.

De este modo:

A = Idem, pero sobre el set seleccionado

B = Total de funciones testeadas (tomar set seleccionado)

### OPCION II

**Método de Aplicación:** Observar el resultado del Smoke Test luego de adaptar al producto a distintos hardware

**Formula:**  $X = 1 - (A / B)$

Donde

A = Número de casos fallidos del Smoke Test

B = Número total de casos del Smoke Test

**Interpretación de la métrica:**  $0 \leq X \leq 1$  Cuanto mas cercano a 1 es mejor

**Escala de Métrica:** Absoluta

Tipo Medición:

A = Contar

B = Contar  
X = 1 – (Contar/Contar)

**Entrada a la medicion:** Reporte de Smoke test

**8.6.1.2.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

- ☐ Documento de arquitectura
- ☐ Requisitos mínimos de hardware
- ☐ Evidencias de pruebas con distintos Hardware
- ☐ Documentación funcional
- ☐ Manuales de usuario
- ☐ Manual de instalación
- ☐ Template de Smoke Test usado por la organización
- ☐ Ejemplo de pruebas funcionales usadas por la organización
- ☐ Equipos (hardware) de distintas características y masters asociados

**8.6.1.2.4 EJEMPLOS.**

Despliegue de version en un ambiente nuevo con otra configuración de Hardware

**8.6.1.2.5 GLOSARIO.**

Smoke Test: Permite determinar si la versión está en condiciones para que sean iniciados los esfuerzos de testing

**8.6.1.3 Organisational environment adaptability (Organisation adaptability to infrastructure of organisation).**

Adaptabilidad en ambientes organizacionales.

**8.6.1.3.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta (CED)

**8.6.1.3.2 VALIDACIONES.**

Métrica

Pone foco en el resultado de la adaptación a otro “ambiente organizacional”. Este término se refiere al ambiente de la operación del negocio.

$X = 1 - A/B$

A = Número de funciones operacionales de las cuales algunas tareas no se completaron o no cumplieron nivel adecuado durante el testing debido al ambiente de negocio organizacional

B=Número total de funciones testeadas

OK. Entendemos que en este caso no sería útil ofrecer como alternativa un Smoke Test dado que los aspectos del negocio deben ser chequeados en detalle.

**8.6.1.3.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO**

- ☐ Documentación del producto que indique su propósito y negocio al que va dirigido
- ☐ Documentación funcional Ej: Documentación de Casos de Uso
- ☐ Manuales del usuario
- ☐ Manual de instalación
- ☐ Manual de customización (para lograr adaptarlo a distintos negocios)

- ❑ Ejemplo de pruebas funcionales usadas por la organización

### 8.6.1.3.4 EJEMPLOS

Seleccionar ejemplos representativos de ambientes organizacionales (ej: dos empresas que se ocupan del mismo negocio) y evaluar el resultado de la customización.

### 8.6.1.3.5 GLOSARIO.

Customizar: Adaptar un producto a una necesidad

### 8.6.1.4 Porting user friendliness.

Adaptabilidad/Facilidad de uso del usuario.

#### 8.6.1.4.1 IMPORTANCIA GENERAL / ESCENARIOS.

Media (CED)

#### 8.6.1.4.2 VALIDACIONES.

Métrica

Pone foco en el tiempo que demora un usuario en completar la adaptación del software al ambiente del usuario, cuando se intenta instalar o cambiar instalación

T = Tiempo total

Si bien el tiempo que demora no es un dato menor, tener en cuenta que puede depender de la experiencia, conocimiento del usuario que realiza la configuración, cantidad de operaciones requeridas, etc. Es por esto que se propone una métrica adicional que indica la cantidad de operaciones realizadas.

Asimismo entendemos que ambas métricas deben ser analizadas función del tamaño del producto a evaluar. Es evidente que un producto pequeño y sencillo será mucho más rápido de adaptar que uno con muchas funcionalidades y complejidad (ejemplo un ERP).

Como dato de tamaño se puede considerar, cantidad de casos de uso, cantidad de puntos funcionales o alguna otra medida de tamaño de producto. Se sugiere tener categorías según la métrica de tamaño que se esté usando y realizar comparaciones con los valores obtenidos.

Métrica adicional

### Measurement Formula

T = Número de acciones requeridas por el usuario que realiza la configuración para lograr la adaptación del software

### Interpretación

Cuanto menor sea es mejor.

**Como se mencionó previamente, se debe analizar el tiempo total (métrica del documento), la cantidad de operaciones (métrica nueva), el tiempo promedio por operación y luego evaluarlo según categoría de producto (tamaño)**

### 8.6.1.4.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO

- ❑ Documento de arquitectura



- ☐ Documentación del producto que indique su propósito y negocio al que va dirigido
- ☐ Documentación funcional Ej: Documentación de Casos de Uso
- ☐ Manual de customización (para lograr adaptarlo a distintos negocios)
- ☐ Manual usuario
- ☐ Manual de instalación

#### **8.6.1.4.4 EJEMPLOS**

Seleccionar ejemplos representativos de ambientes organizacionales (ej: dos empresas que se ocupan del mismo negocio) y evaluar el resultado de la customización.

#### **8.6.1.4.5 GLOSARIO.**

Customizar: Adaptar un producto a una necesidad

### **8.6.1.5 System software environmental adaptability**

Adaptabilidad en ambientes de software.

#### **8.6.1.5.1 IMPORTANCIA GENERAL / ESCENARIOS**

Alta (CED)

#### **8.6.1.5.2 VALIDACIONES.**

Métrica

Pone foco en ambiente de software, ej: adaptabilidad al Sistema Operativo, Software de Red, etc. Nótese diferencia con “ambiente organizacional”

$X = 1 - A/B$

A = Número de funciones operacionales de las cuales algunas tareas no se completaron o no cumplieron nivel adecuado durante el testing combinado con el sistema operativo, o software de aplicación concurrente

B = Total de funciones testeadas

Como alternativa, se propone la siguiente métrica que no requiere una prueba total:

**Método de Aplicación:** Luego de instalar en otra configuración de Software realizar un Smoke Test para evaluar estabilidad de la aplicación (se provee uno de ejemplo)

**Formula:**  $X = 1 - (A / B)$

Donde

A = Numero de casos fallidos del Smoke Test

B = Numero total de casos del Smoke Test

**Interpretación de la métrica:**  $0 \leq X \leq 1$  Cuanto mas cercano a 1 es mejor

**Escala de Métrica:** Absoluta

Tipo Medición:

A = Contar

B = Contar

$X = 1 - (\text{Contar}/\text{Contar})$

**Entrada a la medición:** Reporte de Smoke test

## 8.6.1.5.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO

- ☐ Documentación de arquitectura
- ☐ Documentación del producto que indique compatibilidad con SO, software base, software de red, etc
- ☐ Documentación funcional Ej: Documentación de Casos de Uso
- ☐ Manual de customización
- ☐ Manual de usuario
- ☐ Manual de instalación
- ☐ Template de Smoke Test usado en la organización
- ☐ Evidencias de pruebas sobre distintas plataformas
- ☐ Proveer masters de las distintas configuraciones/plataformas (servidor y terminales), equipos previamente instalados.
- ☐ Sería deseable contar con soporte de personal calificado de la organización evaluada para dar soporte en estas actividades.
- ☐ Como adicional debido a que resulta complejo contar con las distintas configuraciones en distintas terminales se puede considerar virtualizar los distintos ambientes.

## 8.6.1.5.4 EJEMPLOS

Analizar el documento de compatibilidad con plataformas y seleccionar las que se considere apropiado evaluar. Tener en cuenta que puede resultar complejo contar con distintos ambientes bajo distintas plataformas, así como la instalación del SO, etc. En este caso se requiere a la organización evaluada que provea de masters, equipo y personal calificado para realizar las distintas instalaciones. Asimismo se puede considerar la virtualización de las distintas configuraciones.

## 8.6.1.5.5 GLOSARIO.

Smoke Test: Permite determinar si la versión está en condiciones para que sean iniciados los esfuerzos de testing

## 8.6.2 Instalación – Capacidad para ser instalado (Installability).

### 8.6.2.1 Ease of installation

Facilidad/Capacidad en instalación.

#### 8.6.2.1.1 IMPORTANCIA GENERAL / ESCENARIOS.

Alta (CED)

#### 8.6.2.1.2 VALIDACIONES.

Métrica

Pone foco en la facilidad de instalación desde el punto de vista del número de intentos éxitos vs total de intentos

$X = A/B$

A= Número de casos en los cuales el usuario tuvo éxito en cambiar la instalación

B= Número de intentos

Según lo que analizamos, esta métrica estaría midiendo el éxito o no de las instalaciones ignorando la dificultad y complejidad de los pasos a seguir. Se sugiere mirar esta métrica en conjunto con la métrica (3.6.1.4 Porting user friendliness) ya que la misma considera el tiempo y número de operaciones requeridas para la instalación, así como el tamaño del producto a medir.

### 8.6.2.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO

- ☐ Documento de arquitectura
- ☐ Documentación del producto que indique su propósito y negocio al que va dirigido
- ☐ Documentación funcional Ej: Documentación de Casos de Uso
- ☐ Manual de customización (para lograr adaptarlo a distintos negocios)
- ☐ Manual usuario
- ☐ Manual de instalación

### 8.6.2.1.4 EJEMPLOS

Seleccionar ejemplos representativos de ambientes organizacionales (ej: dos empresas que se ocupan del mismo negocio) y evaluar el resultado de la customización.

### 8.6.2.1.5 GLOSARIO.

#### 8.6.2.2 Ease of Setup retry.

Facilidad/Capacidad en vuelta atrás.

##### 8.6.2.2.1 IMPORTANCIA GENERAL / ESCENARIOS.

Alta. (CED)

##### 8.6.2.2.2 VALIDACIONES.

#### Métrica:

Según nuestro análisis, es una métrica importante de considerar ya que la misma contempla la facilidad de vuelta atrás (retry) en caso de ser requerida. La métrica consiste en considerar el número de casos en los cuales falla la vuelta atrás (retry) sobre el número total de casos en intentos de vuelta atrás (retry) requeridos.

Los fallos entendemos que pueden darse por:

- ☐ Falta de consideraciones/entendimiento en los manuales.
- ☐ Falta de capacitación.
- ☐ Ejecución inapropiada sin considerar backups e impactos con otras aplicaciones.
- ☐ La vuelta atrás es compleja, con varias dependencias y no esta automatizado.

**Ahora bien para el método se propone como evidencia contemplar los logs de las diferentes ejecuciones y no el comportamiento del operador de mantenimiento.**

#### Métrica Adicional:

**Complex of Set-up Retry:** Medir el grado de complejidad en la vuelta atrás (retry) de un producto de software con sus dependencias. Determinar de cuantos componentes su vuelta atrás (retry) esta automatizada vía batch o silence instalation. (Ejemplo con herramienta: SMS)

#### Formula:

$$X = A / B$$

A= Componentes automatizados.

B= Componentes totales a considerar para la vuelta atrás (retry).

**Close to 1 is the best.**

**Absoluto.**

**Glosario:**

**SMS:** System Management Server Installer (Microsoft), programa que genera un exe considerando una serie de componentes en modo batch.

**8.6.2.2.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Se deberá contar con la siguiente documentación actualizada del producto de software y de todas sus dependencias (según corresponda):

- ☐ Manual de Instalación y Mantenimiento (identificando contingencia de vuelta atrás)
- ☐ Políticas y procedimientos de backups.
- ☐ Manual de Operación.
- ☐ Reporte de Resolución de problemas. (identificando los problemas más frecuentes, detallando el caso y paso a paso su resolución)
- ☐ Niveles de SLA.

**8.6.2.2.4 EJEMPLOS.**

Despliegue de nueva versión en un Call Center (Crítico) y necesidad de vuelta atrás.

**8.6.2.2.5 GLOSARIO.**

SLA: *Service Level Agreement*, corresponden a los niveles de servicio requerido de solución.

**8.6.3 Coexistencia – Capacidad para coexistir (Co-existence).**

**8.6.3.1 Available co-existence.**

Capacidad para coexistir disponible.

**8.6.3.1.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alta (CED)

**8.6.3.1.2 VALIDACIONES.**

**Métrica:**

Según nuestro análisis, es una métrica sumamente importante ya que el aplicativo debe funcionar correctamente en forma integral coexistiendo con otros software.

Por ejemplo que el aplicativo funcione correctamente con otras aplicaciones ejecutándose al mismo tiempo (aplicaciones contables, Antivirus o determinadas soluciones corporativas)

Los tiempos de respuesta deberán ser los apropiados.

La métrica considera número de limitaciones o fallas que el usuario contempla en operación con concurrencia con otro software sobre el tiempo de duración de la concurrencia.

**8.6.3.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Se deberá contar con la siguiente documentación actualizada:

- ☐ Manual de Instalación y Mantenimiento.
- ☐ Manual de Operación.
- ☐ Reporte de Resolución de problemas. (identificando los problemas mas frecuentes, detallando el caso y paso a paso su resolución)
- ☐ Documento de Masters de terminales y/o servidores del ambiente productivo.

**8.6.3.1.4 EJEMPLOS.**

Aplicativo en forma transaccional de impacto centralizado en forma conjunta con antivirus, aplicación contable.

#### **8.6.3.1.5 GLOSARIO**

**Co-existencia:** El software deberá ser capaz de ejecutarse en paralelo con otros productos de software sin sufrir anomalías.

### **8.6.4 Upgrades / Reemplazo – Capacidad para reemplazar (Replaceability).**

#### **8.6.4.1 Continued use of data.**

Continuidad en el uso de datos.

##### **8.6.4.1.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Media. (CED)

##### **8.6.4.1.2 VALIDACIONES.**

#### **Métrica:**

Según nuestro análisis dicha métrica mide el grado de disponibilidad y compatibilidad de los conjunto o set de datos, al efectuar una migración o al reemplazar o efectuar un up-grade de la versión incluso una vuelta a una versión anterior.

La métrica contempla la cantidad de datos reemplazados y confirmados que van a ser utilizados continuamente sobre la cantidad de datos de otra aplicación u otra versión planificados a ser reutilizados.

**No medir según comportamiento, sino bajo un esquema de registro o Issue Tracking de los problemas o fallos.**

##### **8.6.4.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Se deberá contar con la siguiente documentación actualizada:

- ☐ Manual de Usuario.
- ☐ Modelo de Datos.
- ☐ Documento de Especificación de ambientes.

##### **8.6.4.1.4 EJEMPLOS.**

Migración por actualización tecnológica.

##### **8.6.4.1.5 GLOSARIO.**

No es requerido.

#### **8.6.4.2 Function inclusiveness.**

Inclusión de funcionalidad.

##### **8.6.4.2.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Media (CED)

##### **8.6.4.2.2 VALIDACIONES.**

#### **Métrica:**

Según nuestro análisis dicha métrica mide el grado de similitud de nuevas funciones incorporadas al efectuar una migración o al reemplazar o efectuar un up-grade de la versión.

La métrica contempla la cantidad de funciones que producen resultados similares a los que producía anteriormente sobre el número de funciones testeadas las cuales son similares a funciones provistas en otro software a ser reemplazado.

**No medir según comportamiento, sino bajo un esquema de registro o Issue Tracking de los problemas o fallos.**

#### **8.6.4.2.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Se deberá contar con la siguiente documentación actualizada:

- ☐ Manual de Usuario.
- ☐ Modelo de Datos.
- ☐ Documento de Especificación de ambientes.
- ☐ Requerimientos
- ☐ Entendimiento.

#### **8.6.4.2.4 EJEMPLOS.**

Migración por actualización tecnológica, mejora en funcionalidad, tiempos de respuesta.

#### **8.6.4.2.5 GLOSARIO.**

No es requerido.

### **8.6.4.3 User support functional consistency.**

Consistencia Funcional en relación al usuario.

#### **8.6.4.3.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Alto (CED)

#### **8.6.4.3.2 VALIDACIONES.**

##### **Métrica:**

Según nuestro análisis, es una métrica de importancia ya que el aplicativo debe funcionar correctamente y consistentemente en base a la solicitud y especificación del cliente/usuario. La métrica contempla la cantidad de nuevas funciones inaceptables sobre la cantidad total de nuevas funciones.

**No medir según comportamiento, sino bajo un esquema de registro o Issue Tracking.**

#### **8.6.4.3.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Se deberá contar con la siguiente documentación actualizada:

- ☐ Manual de Usuario.
- ☐ Test Cases.
- ☐ Requerimientos.
- ☐ Entendimiento.

#### **8.6.4.3.4 EJEMPLOS.**

Nueva funcionalidad transaccional a aplicativo para incluir nuevos servicios que están especificados en la solicitud de cambio, firmada por el usuario.

#### **8.6.4.3.5 GLOSARIO.**

No es requerido.

## **8.6.5 Compatibilidad en Portabilidad (Portability Compliance).**

### **8.6.5.1 Portability compliance.**

Compatibilidad en la portabilidad.

#### **8.6.5.1.1 IMPORTANCIA GENERAL / ESCENARIOS.**

Media / Alta (CED)

#### **8.6.5.1.2 VALIDACIONES.**

#### **Métrica:**

Según nuestro análisis, se deberán cumplir los estándares y políticas de la organización aplicado al producto en todas las etapas según corresponda.

La métrica considera número de ítems de conformidad especificados que no han sido implementados durante el testing sobre el número total de ítems de conformidad especificados.

#### **8.6.5.1.3 PRECONDICIONES/DOCUMENTACION DE RESPALDO.**

Se deberá contar con la siguiente documentación actualizada:

- ☐ Manual de Usuario
- ☐ Guías de Standares y Convenciones
- ☐ Checklist de Verificación
- ☐ Test Cases
- ☐ Certificaciones
- ☐ Normas y Convenciones a las que da cumplimiento el software

#### **8.6.5.1.4 EJEMPLOS.**

Testeo de aplicativo de acuerdo a los standares y regulaciones que se indican en la especificación del producto

#### **8.6.5.1.5 GLOSARIO.**

No es requerido.