

UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO

ESCUELA PROFESIONAL DE INGENIERÍA INFORMÁTICA Y DE
SISTEMAS



“Métricas de software de Requerimiento no funcionales”

DOCENTE: Ing ROXANA LISETTE QUINTANILLA PORTUGAL

CURSO: INGENIERIA DE SOFTWARE

ESTUDIANTE:

CODIGO:

INCA CRUZ CARLOS EDUARDO

171943 P

BUSTAMANTE FLORES ERICK ANDREW

174909 P

CHOQUE SARMIENTO LEYDI DIANA

174911 P

HUANCARA CCOLQQUE ALEX HELDER

174912 P

PEREIRA CHINCHERO RICHARD MIKHAEL

171916 P

QUISPE CHAMBILLA CARLOS ENRIQUE

174447 P

SARCO JACINTO DANIEL EDUARDO

174452 P

VEGACENTENO OLIVERA RONALDINHO

140934 P

P= Participo

NP = No Participo

Cusco – 2021

Métricas de software de Requerimiento no Funcioanles.

Análisis no Funcional (Técnica de reconocimiento en ISO 25010).

Pasamos a extraer la información de “ISO 25000 calidad de software y datos) en el cual encontramos modelos de calidad.

1. Definición.

Para poder hallar un modelo de calidad se estable evaluar el sistema con respecto a unos puntos ya definidos los cuales nos permiten tener una evaluación correcta, muchas veces creemos que la creación de un software es sencillo pero existen etapas y modelos que se debe seguir para el correcto funcionamiento y calidad.

El modelo de calidad del producto definido por la ISO/IEC 25010 se encuentra compuesto por las ocho características de calidad que se muestran en la siguiente figura:



Adecuación Funcional.

Representa la capacidad del producto software para proporcionar funciones que satisfacen las necesidades declaradas e implícitas, cuando el producto se usa en las condiciones especificadas. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

- **Completitud funcional.** Grado en el cual el conjunto de funcionalidades cubre todas las tareas y los objetivos del usuario especificados.
- **Corrección funcional.** Capacidad del producto o sistema para proveer resultados correctos con el nivel de precisión requerido.
- **Pertinencia funcional.** Capacidad del producto software para proporcionar un conjunto apropiado de funciones para tareas y objetivos de usuario especificados.

Eficiencia de desempeño.

Esta característica representa el desempeño relativo a la cantidad de recursos utilizados bajo determinadas condiciones. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

- **Comportamiento temporal.** Los tiempos de respuesta y procesamiento y los ratios de throughput de un sistema cuando lleva a cabo sus funciones

bajo condiciones determinadas en relación con un banco de pruebas (benchmark) establecido.

- **Utilización de recursos.** Las cantidades y tipos de recursos utilizados cuando el software lleva a cabo su función bajo condiciones determinadas.
- **Capacidad.** Grado en que los límites máximos de un parámetro de un producto o sistema software cumplen con los requisitos.

Compatibilidad.

Capacidad de dos o más sistemas o componentes para intercambiar información y/o llevar a cabo sus funciones requeridas cuando comparten el mismo entorno hardware o software. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

- **Coexistencia.** Capacidad del producto para coexistir con otro software independiente, en un entorno común, compartiendo recursos comunes sin detrimento.
- **Interoperabilidad.** Capacidad de dos o más sistemas o componentes para intercambiar información y utilizar la información intercambiada.

Usabilidad.

Capacidad del producto software para ser entendido, aprendido, usado y resultar atractivo para el usuario, cuando se usa bajo determinadas condiciones. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

- **Capacidad para reconocer su adecuación.** Capacidad del producto que permite al usuario entender si el software es adecuado para sus necesidades.
- **Capacidad de aprendizaje.** Capacidad del producto que permite al usuario aprender su aplicación.
- **Capacidad para ser usado.** Capacidad del producto que permite al usuario operarlo y controlarlo con facilidad.
- **Protección contra errores de usuario.** Capacidad del sistema para proteger a los usuarios de hacer errores.
- **Estética de la interfaz de usuario.** Capacidad de la interfaz de usuario de agradar y satisfacer la interacción con el usuario.
- **Accesibilidad.** Capacidad del producto que permite que sea utilizado por usuarios con determinadas características y discapacidades.

Fiabilidad.

Capacidad de un sistema o componente para desempeñar las funciones especificadas, cuando se usa bajo unas condiciones y periodo de tiempo determinados. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

- **Madurez.** Capacidad del sistema para satisfacer las necesidades de fiabilidad en condiciones normales.
- **Disponibilidad.** Capacidad del sistema o componente de estar operativo y accesible para su uso cuando se requiere.
- **Tolerancia a fallos.** Capacidad del sistema o componente para operar según lo previsto en presencia de fallos hardware o software.
- **Capacidad de recuperación.** Capacidad del producto software para recuperar los datos directamente afectados y reestablecer el estado deseado del sistema en caso de interrupción o fallo.

Seguridad.

Capacidad de protección de la información y los datos de manera que personas o sistemas no autorizados no puedan leerlos o modificarlos. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

- **Confidencialidad.** Capacidad de protección contra el acceso de datos e información no autorizados, ya sea accidental o deliberadamente.
- **Integridad.** Capacidad del sistema o componente para prevenir accesos o modificaciones no autorizados a datos o programas de ordenador.
- **No repudio.** Capacidad de demostrar las acciones o eventos que han tenido lugar, de manera que dichas acciones o eventos no puedan ser repudiados posteriormente.
- **Responsabilidad.** Capacidad de rastrear de forma inequívoca las acciones de una entidad.
- **Autenticidad.** Capacidad de demostrar la identidad de un sujeto o un recurso.

Mantenibilidad

Esta característica representa la capacidad del producto software para ser modificado efectiva y eficientemente, debido a necesidades evolutivas, correctivas o perfectivas. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

- **Modularidad.** Capacidad de un sistema o programa de ordenador (compuesto de componentes discretos) que permite que un cambio en un componente tenga un impacto mínimo en los demás.
- **Reusabilidad.** Capacidad de un activo que permite que sea utilizado en más de un sistema software o en la construcción de otros activos.
- **Analizabilidad.** Facilidad con la que se puede evaluar el impacto de un determinado cambio sobre el resto del software, diagnosticar las deficiencias o causas de fallos en el software, o identificar las partes a modificar.

- **Capacidad para ser modificado.** Capacidad del producto que permite que sea modificado de forma efectiva y eficiente sin introducir defectos o degradar el desempeño.
- **Capacidad para ser probado.** Facilidad con la que se pueden establecer criterios de prueba para un sistema o componente y con la que se pueden llevar a cabo las pruebas para determinar si se cumplen dichos criterios.

Portabilidad

Capacidad del producto o componente de ser transferido de forma efectiva y eficiente de un entorno hardware, software, operacional o de utilización a otro. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

- **Adaptabilidad.** Capacidad del producto que le permite ser adaptado de forma efectiva y eficiente a diferentes entornos determinados de hardware, software, operacionales o de uso.
- **Capacidad para ser instalado.** Facilidad con la que el producto se puede instalar y/o desinstalar de forma exitosa en un determinado entorno.
- **Capacidad para ser reemplazado.** Capacidad del producto para ser utilizado en lugar de otro producto software determinado con el mismo propósito y en el mismo entorno.

Adecuación funcional	
Compleitud funcional	En un 60% – 70%
Corrección funcional	En un 100% en base a lo desarrollado
Pertinencia funcional	En un 60-70%

Eficiencia y desempeño	
Comportamiento temporal	No implementado
Utilización de recursos	-
Capacidad	-

Compatibilidad	
Coexistencia	No tiene coexistencia con ningún otro sistema
Interoperabilidad	El aplicativo se encuentre inter conectado en la nube ya que este mismo es de entorno web, por lo cual la relación de la base de datos (backend) con la interfaz (frontend) se encuentra presente.

Fiabilidad

Madurez	El sistema cumple con las necesidades del usuario.
Disponibilidad	Al no haber sido subido a un servidor aun no se cuenta con la disponibilidad de este
Tolerancia a fallos	Presenta una tolerancia mínima a los fallos.
Capacidad de recuperación	No presenta.

Seguridad	
Confidencialidad	El aplicativo no cuenta con la protección de datos y la base de datos al ser subido a una FireBase nos permite implementar la confidencialidad
Integridad	No cuenta con integridad
No repudio	No almacena la información de las acciones o eventos realizados
Responsabilidad	No rastrea las acciones de una entidad.
Autenticidad	No permite la identificación de los usuarios.

Mantenibilidad	
Modularidad	Cuenta con el modularidad dentro de sus componentes.
Reusabilidad	El sistema es reusable ya que este se encuentra subido de manera completa a un repositorio en GitHub.
Analizabilidad	Si es analizable, debido a la relación entre sus módulos
Capacidad para ser modificado	El código puede ser modificado al ser descargado del repositorio, esto entendiendo la lógica del producto.
Capacidad para ser probado	No cuenta con la capacidad de pruebas.

Portabilidad	
Adaptabilidad	Este se limita al entorno local ya que este no se encuentra subido a algún hosting.
Capacidad de ser instalado	No requiere instalación
Capacidad para ser reemplazado	No tiene esta funcionabilidad