

Sistema de Gestión de Evento Holimori
Documento de Aseguramiento de la Calidad para la
Fase de Codificación

Versión 1.0

22/01/2021

1. Codificación

1.1. Objetivo

El objetivo es documentar el proceso de desarrollo del producto software, conocer el funcionamiento de este a detalle y verificar las pruebas de software realizadas y su funcionamiento integral.

1.2. Entradas

- Listado de código
- Programa ejecutable
- Plan de Proyecto

1.3. Procesos

1.3.1. Análisis de factores de codificación

- Implementación de reglas de autorización

Las reglas de autorización se definen para especificar las condiciones que deben cumplirse para que se permita el acceso a un objeto protegido.

Las reglas también pueden utilizarse para implementar una política de autorización ampliable y basada en atributos utilizando los atributos del entorno empresarial o los atributos de fuentes externas de confianza.

- Implementación de control de integridad de archivos.

El monitoreo de integridad de archivos implica contar con un control de versiones para los archivos y así tener la información suficiente para recuperarse rápidamente ante un incidente o cambio no autorizado

- Implementación de procedimientos de seguridad.

Para la implementación de un sistema de seguridad se debe apoyar en un Sistema de Gestión de Seguridad de la Información (SGSI), mediante el cual se logra proteger los datos imprescindibles. Este sistema está basado en tres dimensiones fundamentales, las cuales son:

Confidencialidad: Consiste en no dar acceso a la información a individuos, entidades o procesos no autorizados.

Integridad: La conservación de la completitud y exactitud de la información y sus métodos de proceso.

Disponibilidad: Individuos, entidades o grupos autorizados tienen acceso a la información y a sus sistemas de tratamiento cuando lo requieran.

- Cumplimiento del programa con la metodología a adoptar.

Se debe utilizar una metodología de desarrollo de acuerdo a las necesidades del proyecto, se puede utilizar metodologías

iterativas o secuenciales.

- **Seguimiento a diseño.**

Todo software de calidad debe realizar codificación de software siguiendo los lineamientos establecidos en el documento de diseño.

- **Mantenimiento del programa.**

Después de que el producto software se encuentre en producción debe recibir mantenimiento para adaptarse al entorno de constante cambio, migraciones o implementar nuevas funcionalidades requeridas.

1.4. Salidas

1.4.1. Revisión de la base de datos

1.4.2. Información de la documentación del lenguaje de programación

1.

1.5. Lista de Verificación (Checklist)

Item	Cumple	Descripción
¿Es considerada una responsabilidad del programador la verificación y la validación de los programas?	NO	El programador es el encargado de llevar a la realidad las especificaciones y requerimientos antes detallados por el analista en conjunto con el cliente, siendo este el caso, no es responsabilidad del programador verificar y validar el desarrollo a corto o largo plazo del software.
¿El programador entiende los conceptos implicados, y los tendrá presentes en la verificación de la programación?	SI	El programador entiende los conceptos implicados del sistema y los enfocará en el óptimo desarrollo del software, aún así, según el detalle de obligaciones de cada integrante del proyecto, el programador no es el encargado de hacer la verificación general del programa.
¿El programa será sometido a un test completo antes de ingresar a un nivel superior de test?	SI	El programador trabaja en conjunto con el área de testing, en la documentación se especifica 3 niveles de testeo, en cada uno el software completo se somete a una evaluación,

		si se encuentran errores, se impedirá el paso al siguiente nivel de testing
¿Todos los defectos no cubiertos están registrados en detalle?	SI	En la documentación se puede apreciar un apartado que detalla los defectos encontrados en cada nivel de testeo
¿Todos los defectos no cubiertos fueron corregidos antes de ingresar al siguiente nivel de verificación?	SI	Todos los defectos fueron corregidos en cada nivel de testeo, debido a que se imposibilita el acceso al nivel superior de testing si se encuentra al menos un error en el programa. La documentación indica que los defectos se solucionaron en base a lo que se previó como posibles errores antes de entrar a la fase de testing

Breve resumen

Tomando en cuenta los checklist se concluye que en la fase de instalación se comprueba que se revisa los cambios periódicamente y en el caso que se detecte algún error estos problemas son documentados y comunicados. Se verificó el funcionamiento del sistema de acuerdo a los lineamientos establecidos.

1.6. Métricas

MÉTRICAS PARA CÓDIGO FUENTE

Usando un conjunto de medidas primitivas que pueden derivarse

- n1 número de operadores distintos que aparecen en un programa
- n2 número de operandos distintos que aparecen en un programa
- N1 número total de ocurrencias de operador
- N2 número total de ocurrencias de operando

Usa estas medidas primitivas para desarrollar: expresiones para la longitud del programa Global. volumen mínimo potencial para un algoritmo. volumen real (número de bits requeridos para especificar un programa).

$$V = N \log_2 (n_1 + n_2)$$

Nivel del programa (una medida de complejidad del software). Nivel del lenguaje (una constante para un lenguaje determinado). Esfuerzo de desarrollo, tiempo de desarrollo e incluso . El número proyectado de fallas en el software. Longitud total $N = n_1 \log_2 n_1 + n_2 \log_2 n_2$