nacap

Glosario

Gestión de defectos, combinación de valores de prueba y exploración



A

ANÁLISIS DE CAUSA RAÍZ (RCA)

Proceso utilizado para identificar la causa principal de un defecto o problema en el software, permitiendo implementar soluciones para prevenir su recurrencia.

B

BUG

Término comúnmente utilizado para referirse a un error, defecto o fallo en el software que produce un resultado incorrecto o no deseado.

C

CASOS DE PRUEBA (TEST CASES)

Conjunto de condiciones o variables bajo las cuales un probador determinará si un sistema o una de sus características funciona según lo especificado.

COMBINACIÓN DE VALORES DE PRUEBA

Técnica de prueba que implica la creación de casos de prueba que combinan diferentes valores de entrada para asegurar que el sistema funcione correctamente bajo varias condiciones.

D

DEFECTO (DEFECT)

Imperfección en un componente o sistema que puede causar que el producto falle al realizar la función requerida. Los defectos pueden surgir de errores en el código, diseño o requisitos.

E

EXPLORACIÓN (EXPLORATORY TESTING)

Técnica de prueba en la que los casos de prueba no están predefinidos, sino que se diseñan, ejecutan y modifican sobre la marcha, con el objetivo de descubrir defectos de manera intuitiva y exploratoria.

G

GESTIÓN DE DEFECTOS

Proceso sistemático para identificar, registrar, rastrear y resolver defectos en el software, asegurando que se aborden de manera eficiente y efectiva.

M

MATRIZ DE TRAZABILIDAD DE REQUISITOS (RTM)

Documento que vincula los requisitos del proyecto con sus respectivas pruebas, asegurando que todos los requisitos se prueben y se verifiquen adecuadamente.

P

PLAN DE PRUEBAS (TEST PLAN)

Documento que describe el alcance, enfoque, recursos y calendario de las actividades de prueba previstas. Identifica los elementos a probar, las tareas de prueba, quién las llevará a cabo y cualquier riesgo de prueba.

PRIORIDAD (PRIORITY)

Clasificación de un defecto basada en la importancia y urgencia con la que debe ser solucionado. La prioridad determina el orden en que se deben abordar los defectos.

R

REVISIÓN DE CÓDIGO (CODE REVIEW)

Inspección del código fuente por otros desarrolladores o probadores para identificar defectos, mejorar la calidad del código y asegurar el cumplimiento de las normas de codificación.

S

SEVERIDAD (SEVERITY)

Clasificación de un defecto basada en el impacto que tiene sobre el funcionamiento del sistema. Los niveles de severidad ayudan a determinar la criticidad del defecto y su efecto en el software.

SISTEMA DE GESTIÓN DE DEFECTOS (DMS)

Herramienta o software utilizado para registrar, rastrear y gestionar defectos durante el ciclo de vida del desarrollo de software.

T

TEST SCRIPT

Secuencia de acciones específicas, pasos y datos utilizados para realizar una prueba en el software, generalmente automatizada.

V

VERIFICACIÓN Y VALIDACIÓN (V&V)

Procesos utilizados para asegurar que un sistema cumple con los requisitos y que es adecuado para su uso previsto. La verificación evalúa si el producto se está desarrollando correctamente, mientras que la validación asegura que el producto cumple con las necesidades del usuario.

Referencias Bibliográfica

IEEE Standard for Software and System Test Documentation, IEEE Std 829-2008. Institute of Electrical and Electronics Engineers, 2008.

Kaner, C., Falk, J., & Nguyen, H. Q. (1999). Testing Computer Software (2nd ed.). Wiley.

Myers, G. J., Sandler, C., & Badgett, T. (2011). The Art of Software Testing (3rd ed.). Wiley.

Sommerville, I. (2011). Software Engineering (9th ed.). Pearson.

ISO/IEC/IEEE International Standard - Systems and software engineering -- Vocabulary, ISO/IEC/IEEE 24765:2010(E), 2010.

Black, R. (2009). Advanced Software Testing - Vol. 2, 2nd Edition. Rocky Nook.