

TESTING DE SOFTWARE



- QUÉ ES
 - El proceso consistente en demostrar que el sistema no presenta errores.
- QUÉ NOS APORTA:
 - Calidad durante todo el proceso.
 - Disminución de costos.
 - Reducción de riesgos.
 - Optimización de recursos.
 - El seguimiento de estándares.
- COSTOS:
 - Los recursos que se destinarán en un proyecto para asegurar que el servicio o producto final cumpla satisfactoriamente con los requisitos preestablecidos por el cliente.
- ETAPAS:
 - Requerimientos.
 - Diseño
 - Codificación
 - Pruebas de desarrollo
 - Pruebas aceptación
 - Producción
- SIN TESTING:
 - Baja calidad
 - Pérdida de imagen
 - Pérdidas económicas
 - Fallas en sistemas críticos
- TIPOS:
 - De caja negra:
 - No esta basada en el conocimiento del código o diseño interno, determina la funcionalidad del sistema.
 - De caja blanca:
 - Esta basada en la lógica interna de la aplicación y el código. Hace una cobertura de declaraciones del código, ramas, caminos y condiciones.
 - Unitaria:
 - Basada en la funcionalidad de los módulos del programa, como funciones, procedimientos, módulos de clase.
 - Integración incremental:
 - Cuando nuevas funciones son ingresadas al sistema se hace la prueba basándose en la funcionalidad, la dependencia con otros módulos y la integración con el programa completo.
 - De integración:
 - Se basa en las pruebas de conexiones y comunicaciones entre diferentes módulos.

- Funcional:
 - La caja negra hace la prueba funcional de los requerimientos de la aplicación y generalmente es realizada por el programador, en cambio, la prueba funcional es realizada por los testers.
- De sistema:
 - Es una prueba de caja negra incluyendo todos los componentes del sistema desde el hardware a la documentación.
- De fin a fin:
 - Es similar a la prueba de sistema pero esta involucra la interacción con otro hardware, bases de datos y redes.
- De sanidad:
 - Determina si la nueva versión de un software esta bien realizada y si necesita un nuevo esfuerzo en la prueba de software. Por ejemplo la nueva versión de un programa cumple con casi todos los requisitos pero destruye la base de datos al leerla, por lo tanto se dice que este software no esta en una condición sana.
- De aceptación:
 - Es la prueba final basada en las especificaciones del usuario o basada en el uso del programa por el usuario final luego de un periodo de tiempo.
- De carga:
 - Esta basada en las aplicaciones bajo cargas pesadas, generalmente usadas en sitios web y en servidores con gran cantidad de datos donde se determina en cuales puntos existen degradaciones del sistema.
- De rendimiento:
 - Es una de las pruebas finales y sirve para definir los requerimientos y la calidad del software, en base a las pruebas de carga y estrés. Incluye entrevistas con el usuario y programador.
- De estrés:
 - Es una prueba de carga y rendimiento o respuesta basada en la funcionalidad del sistema bajo cargas pesadas, un gran numero de repeticiones, manejo de grandes datos y demasiadas preguntas a bases de datos grandes.
- De instalación y desinstalación:
 - Determina la eficiencia de los procesos que instalan y desinstalan las aplicaciones del programa.
- De recuperación:
 - Es la prueba que evalúa que tan bien se recupera el sistema luego de bloqueos, fallas del hardware u otros problemas catastróficos.
- De compatibilidad:
 - Evalúa el desempeño del software en diferentes hardware, sistemas operativos, redes, etc.

- De exploración:
 - Es una prueba informal del software que no esta basada en ningún plan o caja de prueba y a menudo los testers aprenden del programa al explorar todas las aplicaciones posibles.
- De anuncio:
 - Es similar a la prueba de exploración pero los testers deben tener suficiente noción sobre el funcionamiento del programa antes de comenzar esta prueba. Incluye reunión con analistas y programadores.
- De usuario:
 - Determina si el usuario se desenvuelve satisfactoriamente con el programa.
- De comparación:
 - En esta prueba se comparan los pro y los contra del programa con los programas creados con la competencia.
- Alfa (Alpha):
 - Es la prueba cuando la aplicación esta cerca de la entrega al usuario. Se hacen pequeños cambios generalmente en el diseño de interfaces. Esta prueba es hecha por usuarios.
- Beta:
 - Es la búsqueda de defectos en el programa completo. Generalmente es hecha por usuarios.
- De mutación:
 - Esta prueba esta basada en la introducción deliberada de diferentes códigos externos al programa para reexaminar si estos defectos pueden ser detectados. Requiere gran disponibilidad de recursos de computación.
- CUESTIONES CLAVES DEL TESTING:
 - Test Plan
 - El tester no debe ser el desarrollador
 - La empresa desarrolladora no debe hacer el testeo final
 - Revisión de resultados
 - Registro de resultados
 - Equipo de testers
 - Recursos del Test Plan