

## Introducción a la calidad y pruebas de software

### 1. Explica los principios de “Statical Quality Control”

El control de calidad estadístico realmente se hizo realidad durante la Segunda Guerra Mundial. La necesidad de artículos relacionados con la guerra producidos en masa, como visores de bombas, radares precisos y otros equipos electrónicos, al menor costo posible aceleró el uso de estadísticas gráficas de muestreo y control de calidad. Desde la Segunda Guerra Mundial estas técnicas estadísticas han sido refinados y afilados. El uso de las computadoras en la última década también ha amplió el uso de estas técnicas.

La Segunda Guerra Mundial prácticamente destruyó la capacidad de producción japonesa. En lugar de reorganizar sus viejos métodos de producción, los japoneses solicitaron la ayuda del difunto Dr. W. Edwards Deming, del Departamento de Agricultura de los Estados Unidos, para ayudarlos a desarrollar un plan general. En una serie de seminarios con planificadores japoneses, enfatizó una filosofía que se conoce hoy como los 14 puntos de Deming. Estos 14 puntos se enumeran en la siguiente página. Enfatizó que la calidad se origina en la mejora del proceso, no en la inspección, y que la calidad la determinan los clientes.

El fabricante debe ser capaz, a través de estudios de mercado, para anticiparnos a las necesidades de los clientes. La alta dirección tiene la responsabilidad de la mejora a largo plazo. Otro de sus puntos, y uno que el japonés apoya firmemente, es que cada miembro de la empresa debe contribuir a la mejora a largo plazo. Para lograr esta mejora, la educación permanente y la formación es necesaria.

## 2. Explica el concepto de “Lean Principles”



Figure 1. The Five Lean Principles

### 1. Define Value

Para comprender mejor el primer principio de la definición del valor para el cliente, es importante comprender qué es el valor. El valor es lo que el cliente está dispuesto a pagar. Es primordial descubrir las necesidades reales o latentes del cliente. A veces, los clientes pueden no saber lo que quieren o no pueden articularlo. Esto es especialmente común cuando se trata de nuevos productos o tecnologías. Existen muchas técnicas, como entrevistas, encuestas, información demográfica y análisis web, que pueden ayudarlo a descifrar y descubrir lo que los clientes consideran valioso. Mediante el uso de estas técnicas cualitativas y cuantitativas, puede descubrir qué quieren los clientes, cómo quieren que se entregue el producto o servicio y el precio que pagan.

## 2. Map the Value Stream\

El segundo principio Lean es identificar y mapear el flujo de valor. En este paso, el objetivo es utilizar el valor del cliente como punto de referencia e identificar todas las actividades que contribuyen a estos valores. Las actividades que no agregan valor al cliente final se consideran desperdicio. Los residuos se pueden dividir en dos categorías: sin valor agregado pero necesarios y sin valor e innecesarios. Lo último es puro desperdicio y debe eliminarse, mientras que lo primero debe reducirse tanto como sea posible. Al reducir y eliminar procesos o pasos innecesarios, puede asegurarse de que los clientes obtengan exactamente lo que desean y, al mismo tiempo, reducir el costo de producción de ese producto o servicio.

## 3. Create Flow

Después de eliminar los desechos del flujo de valor, la siguiente acción es garantizar que el flujo de los pasos restantes se desarrolle sin interrupciones ni demoras. Algunas estrategias para garantizar que las actividades de valor agregado fluyan sin problemas incluyen: desglosar los pasos, reconfigurar los pasos de producción, nivelar la carga de trabajo, crear departamentos multifuncionales y capacitar a los empleados para que sean polivalentes y adaptables.

## 4. Establish Pull

El inventario se considera uno de los mayores desperdicios en cualquier sistema de producción. El objetivo de un sistema basado en pull es limitar el inventario y los elementos de trabajo en proceso (WIP) y, al mismo tiempo, garantizar que los materiales y la información necesarios estén disponibles para un flujo de trabajo fluido. En otras palabras, un sistema basado en extracción permite la entrega y fabricación justo a tiempo donde los productos se crean en el momento en que se necesitan y en las cantidades justas necesarias. Los sistemas basados en extracción siempre se crean a partir de las necesidades de los clientes finales. Al seguir el flujo de valor y trabajar hacia atrás a través del sistema de producción, puede asegurarse de que los productos producidos puedan satisfacer las necesidades de los clientes.

## 5. Pursue Perfection

Los desperdicios se evitan mediante el logro de los primeros cuatro pasos:

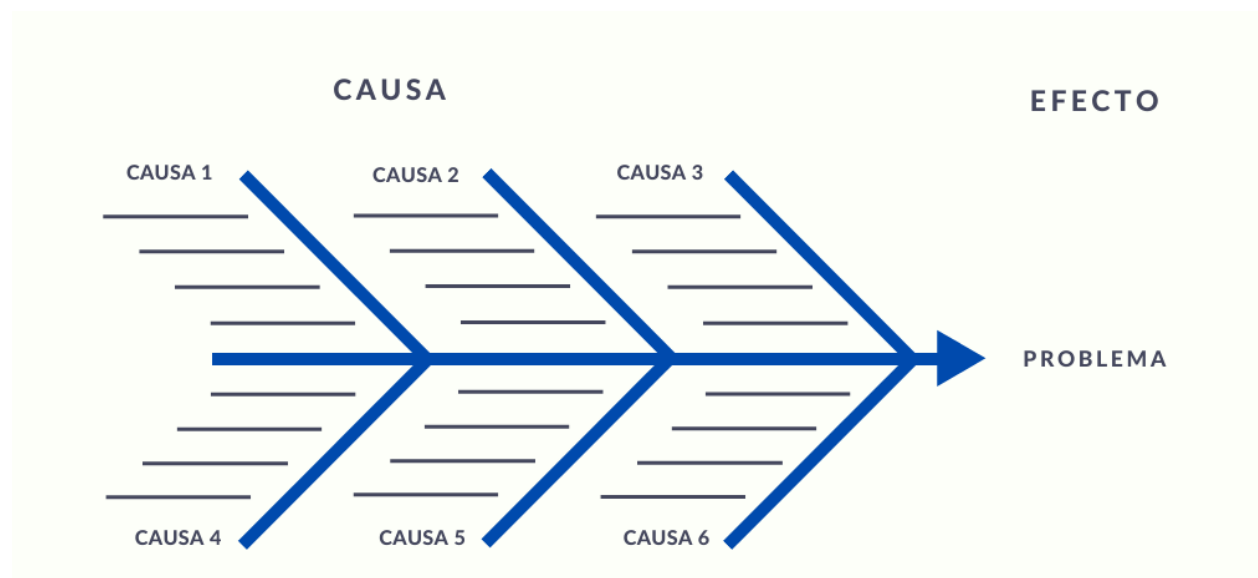
- 1) identificación del valor,
- 2) mapeo del flujo de valor,
- 3) creación del flujo y
- 4) adopción de un sistema pull.

Sin embargo, el quinto paso de perseguir la perfección es el más importante de todos. Hace que el pensamiento Lean y la mejora continua de procesos sean parte de la cultura organizacional. Cada empleado debe esforzarse por alcanzar la perfección mientras entrega productos basados en las necesidades del cliente. La empresa debe ser una organización de aprendizaje y siempre encontrar formas de mejorar un poco cada día.

### 3. ¿Qué es el diagrama de “Ishikawa”? Menciona un ejemplo de cómo construirías ese diagrama

El diagrama de Ishikawa —también conocido como de espina de pescado— es una herramienta visual que tiene un formato de gráfico. Además, su principal función es ayudar en los análisis de organización. La mayoría de las veces se lo emplea para encontrar la causa de un problema en su raíz.

De esa forma, el diagrama tiene como objetivo ayudar al equipo a llegar a las causas reales de cuellos de botella que acometen a los procesos operativos y organizacionales de la empresa. En otras palabras, podemos decir que su propósito es desenmascarar situaciones no deseadas exponiendo su verdadero motivo.



#### **4. Explica las diferencias entre validación y verificación**

La verificación es un ejercicio teórico diseñado para estar seguro de que ningún requerimiento deja de cumplirse en el diseño, mientras que la validación es un ejercicio práctico que garantiza que el producto funcionará para cumplir con los requerimientos. Juntos, garantizan que el producto diseñado conseguirá satisfacer las necesidades del cliente.

#### **5. Explica y menciona un ejemplo relacionado al software de failure, error and fault**

- Software de failure: Visualización de un mensaje de alerta que no fue definido por el desarrollador.
- Software de error: Un error de omisión de un campo en las últimas fases del desarrollo (la descripción de un producto, la clave de un empleado o las placas de un vehículo no incluida en la base de datos).
- Software de fault: Un paso, proceso o definición de dato incorrecto de un programa de software.

#### **6. ¿Cuál es la diferencia entre “fault injection” y “fault simulation”?**

Fault injection (o “inyección de fallas”) es cuando los desarrolladores agregan errores deliberadamente en su software para verificar errores más adelante y fault simulation (o “simulación de errores”) es básicamente una proyección a futuro de los errores que podría llegar a presentar el sistema.

7. Menciona un ejemplo de pruebas de caja blanca y pruebas de caja negra. Explica el concepto de pruebas de caja gris.

#### **Caja Negra**

En las pruebas de caja negra desconocemos la estructura interna del sistema. No sabemos cómo está construido, con qué tecnología, con qué arquitectura, etc. Sin embargo, podemos diseñar las pruebas a partir de la observación de las entradas y salidas de este que suelen estar descriptas en los modelos funcionales.

## Caja Blanca

Cuando armamos los casos de prueba a partir de la estructura del programa, estamos haciendo pruebas de caja blanca. Es decir que los casos de prueba los armamos a partir del código, el flujo de datos o de control, la estructura de la base de datos, etc. ¡Ojo! Esto no significa que vamos a estar buscando defectos en el código... Eso es otra cosa y se llama análisis estático.

## Caja Gris

Muchas veces el límite no está claro, o tal vez estamos siguiendo un enfoque de caja negra, pero como sabemos algo de lo que sucede dentro entonces aprovechamos esa información. Y esto es lo que se suele llamar “caja gris”.

## **8. Explica los conceptos de prueba unitaria, integración, sistema, aceptación y regresión**

### Prueba Unitaria

Una prueba unitaria se utiliza para comprobar que un método concreto del código de producción funciona correctamente, probar las regresiones o realizar pruebas relacionadas (buddy) o de humo. Una prueba por orden se utiliza para ejecutar otras pruebas en un orden especificado.

Consisten en probar métodos y funciones individuales de las clases, componentes o módulos que usa tu software. En general, las pruebas unitarias son bastante baratas de automatizar y se pueden ejecutar rápidamente mediante un servidor de integración continua.

### Integración

El término integración de sistemas está relacionado con la manera de gestión y organización que tienen las distintas empresas públicas y privadas. Debido al constante cambio y evolución que sufren sus departamentos y subsistemas de información, trabajo o transmisión de datos, cada vez es más necesario que todos trabajen bajo un mismo procedimiento que los integre todos, que ayude a aumentar la eficiencia y productividad,

así como a conseguir de una manera más rápida y eficaz sus objetivos estratégicos y empresariales.

### Sistema

Se entiende por un sistema a un conjunto ordenado de componentes relacionados entre sí, ya se trate de elementos materiales o conceptuales, dotado de una estructura, una composición y un entorno particulares. Se trata de un término que aplica a diversas áreas del saber, como la física, la biología y la informática o computación.

### Aceptación

Históricamente, como usuario, sólo se veía por primera vez un nuevo producto de software mucho después de que la mayoría de los desarrolladores y probadores se hubieran ido.

Debido a ese largo intervalo, las pruebas de aceptación eran cruciales para determinar si el software cumplía las expectativas y era viable para sus usuarios, es decir, si era aceptable y apto para salir al mercado. Sólo si lo era, se añadía a un entorno de producción y se utilizaba en el curso normal de la actividad empresarial.

En el desarrollo ágil, las pruebas de aceptación forman parte del proceso y no son una ocurrencia tardía. Sin embargo, la intención sigue siendo la misma: verificar que el software cumple las expectativas desde el punto de vista del cliente y de los usuarios finales.

### Regresión

Una regresión se produce cuando, tras la implementación de un cambio, un software falla en alguna de las partes que han permanecido inalterables.

- 9. Define un proyecto de software, ya sea personal u open source, después identifica fuentes para derivar casos de prueba.**