

Calidad y procesos en la industria del software

Tabla de contenido

Calidad.....	2
Definición: ISO 8402	2
Calidad y procesos	2
Gestión cuantitativa	2
Definiciones: Verificación Vs. Validación	3
Un poco de historia.....	3
Calidad y procesos	4
PDCA.....	4
Enfoques vigentes hoy	5
Mejora continua	5
Calidad del software	5
Calidad y procesos en la industria del software	6
Perspectiva histórica	6

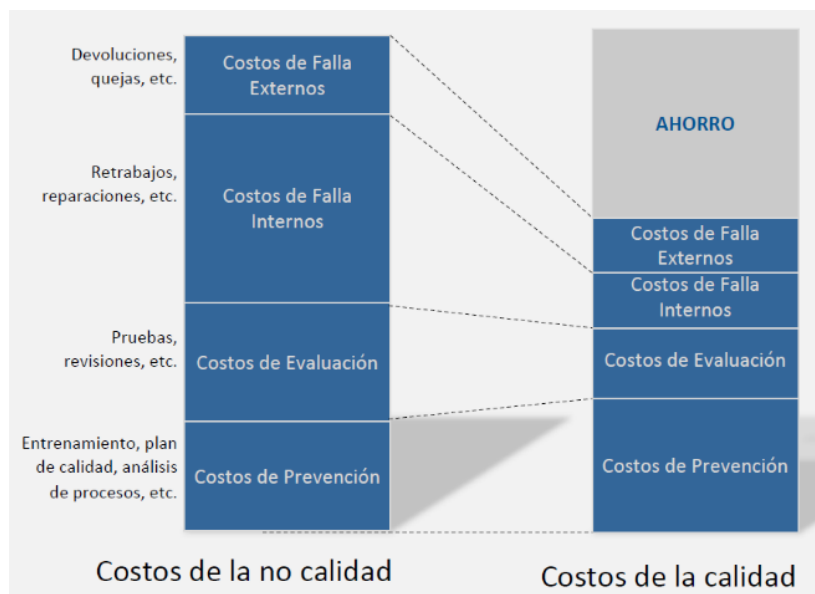
Calidad

Definición: ISO 8402

Antes que nada deberíamos definir qué es calidad. Hay múltiples definiciones posibles. Prácticamente todos los autores que han escrito acerca de calidad han dado su propia definición, nosotros nos quedamos con esta que es la estándar de la ISO y es que calidad es:

*Totalidad de prestaciones y características de un producto o servicio que generan su capacidad de **satisfacer necesidades** explícitas o implícitas*

Claramente la calidad no es gratis. Cuando no hay calidad, incurrimos en los llamados **costos de falla externos**, que son aquellos que tienen que ver con todas las acciones necesarias para atender los reclamos de los clientes. Pensemos en lo que nos pasa en nuestra vida personal cuando tenemos que llamar a algún fabricante por algún electrodoméstico que acabamos de comprar y que no funciona de acuerdo a lo previsto o que tiene algún tipo de falla. Ese fabricante tiene que tener dispuesta una mesa de ayuda, tiene que tener una línea de teléfono dedicada, tiene que tener personal capacitado para atender todos esos reclamos. También hay una serie de costos llamados **costos de falla internos**, que son todos los relacionados con las actividades necesarias para reparar esos productos defectuosos que nos han llegado. La situación habría sido muy diferente si hubiéramos invertido más en prevención y en actividades de evaluación.



Como muestra el gráfico, invertir en estos 2 rubros claramente produce ahorros, ya que un producto de mayor calidad, una vez entregado para su uso, necesitará menos retrabajo y producirá menos reclamos. Invertir en entrenar a la gente, en definir procesos estándar, en tener buenas herramientas, claramente es económicamente conveniente.

Calidad y procesos

Gestión cuantitativa

La preocupación por la calidad y los procesos no es algo nuevo, lleva siglos. La industria tiene una larga historia acerca de su interés por tener procesos estables, bajo control estadístico. Hoy hablamos de cuatro grandes elementos en el mundo de la calidad:

- **Control:** El control de la calidad es equiparable a lo que en nuestra profesión conocemos como testing. Son todas las actividades y técnicas que utilizamos para evaluar si un producto o servicio cumple con determinados requisitos de calidad.
- **Aseguramiento:** El aseguramiento de la calidad, busca proveer cierto nivel de confianza de que el producto o servicio satisface determinados requisitos de calidad. Parece un juego de palabras, un trabalenguas, pero en realidad, el aseguramiento de la calidad lo que busca es que estemos seguros de que hicimos las cosas de

acuerdo a lo que dicen nuestros estándares, mientras que control de calidad lo que apunta es ver si realmente nuestro producto o servicio cumple con los requisitos.

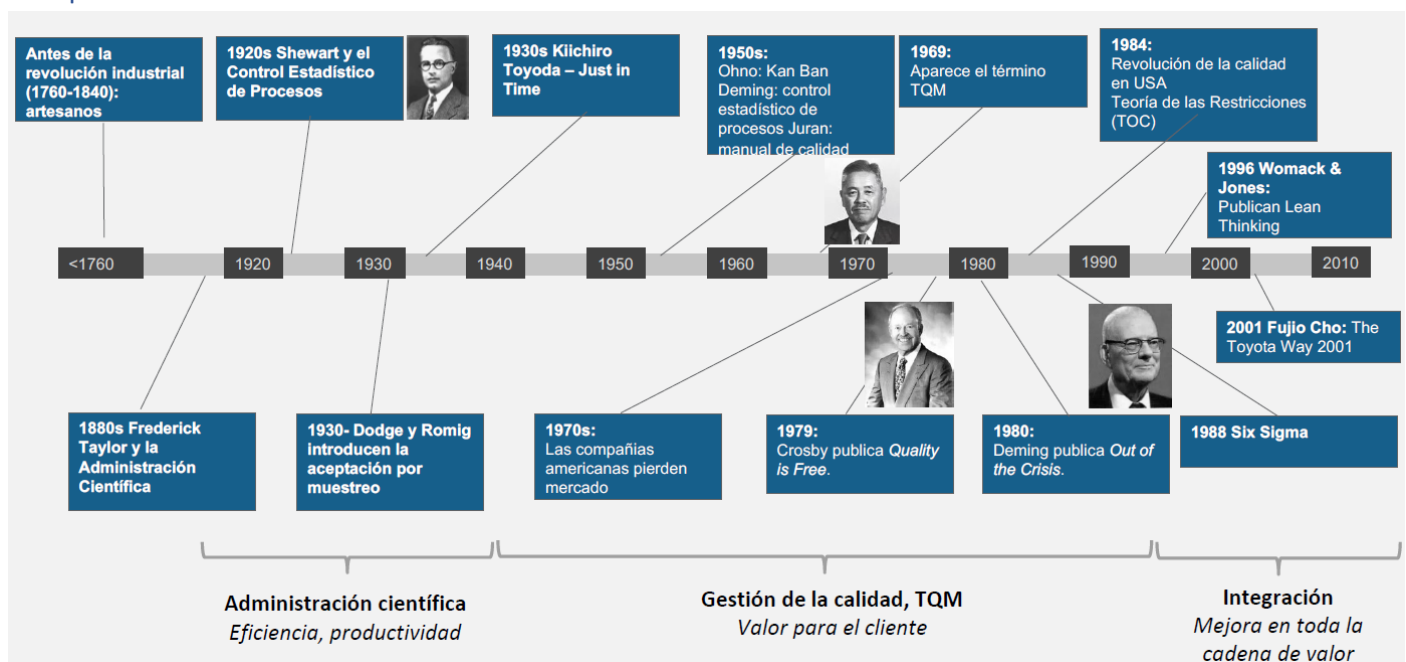
- **Mejora:** Mejora de la calidad lo que busca es la mejora continua de los procesos productivos.
- **Gestión de la calidad:** Es el paraguas que le da el entorno lo más eficaz posible para que esto funcione adecuadamente, provee los aspectos de planificación, organización, etc.

[V / F] Mayor inversión en prevención y actividades de evaluación conlleva a una mayor calidad y un ahorro de costos futuros.

Definiciones: Verificación Vs. Validación

Hay 2 términos que son parecidos pero que no son lo mismo: Verificación y validación. **Verificación** tiene que ver con la pregunta “¿Lo hicimos bien?”, mientras que **validación** con la pregunta “Hace lo que tiene que hacer?”. Para hacer el paralelo con el desarrollo de software, verificación, es equiparable al testing, mientras que validación se parece, o tiene que ver, con el test de aceptación. Esto lo que quiere decir es: el producto que desarrollamos satisface los requisitos, ¿sí o no? La validación tiene que ver con el producto que desarrollamos cumple con las necesidades y expectativas del cliente, esa es la diferencia, a veces esos términos se utilizan de manera intercambiable, con lo cual hay que tener cuidado con la bibliografía a la que uno accede porque a veces uno habla de verificación y en realidad se está refiriendo a lo que llamamos nosotros en este contexto validación.

Un poco de historia



Desde la Revolución industrial hasta nuestros días hemos ido madurando todos estos conceptos que acabamos de presentar. La primera etapa en esta historia ha sido la administración científica que estuvo muy, muy enfocada en el trabajo mecánico de los operarios y en la producción en masas. Eso a veces incluye a lo que llamamos el Fordismo por Henry Ford. Los años 20 del siglo pasado, Walter Shewart trabajó en el control estadístico de procesos. La idea era que la estabilidad de los procesos era muy importante para poder tener resultados predecibles. Imaginémosnos que si uno fabrica un millón de unidades, no puede hacer una prueba sobre ese millón de unidades, tienen que poder tomar una muestra estadísticamente significativa de ese lote y hacer pruebas sobre esa muestra. Y hacía falta entonces tener algún tipo de aseguramiento, de garantía, de que esa muestra era estadísticamente significativa.

En Japón, mientras tanto, en esa época comienza un movimiento que llega hasta nuestros días y que hoy se conoce como “lean manufacturing”. Todo esto arranca de la mano de los hermanos Toyoda fundadores y directores de la empresa Toyota. Después de la Segunda Guerra Mundial, Japón queda completamente devastado. Se decide reconstruir Japón. Los aliados ayudan en esa iniciativa. Arranca entonces para esa época el movimiento de calidad de muchos de los gurúes de Estados Unidos que viajan a Japón a ayudar a reconstruir ese país. Son enviados allá como parte del Plan Marshall para recuperar la economía japonesa. Esto, sumado a lo que ya venía pasando, hace que la industria japonesa explote, y justamente desafíe a la industria automovilística norteamericana. Para los años 80, la industria americana está en peligro porque los japoneses habían aprendido a fabricar automóviles de una manera impresionante, con mucha calidad, a costos mucho más interesantes, lo cual hace trastabillar un poco a la industria en general. Por ese entonces aparece un documental en la televisión americana que se llamó “Porque Japón pudo y nosotros no”, en donde, justamente, se dan a conocer las ideas de estos gurúes americanos que a lo mejor no eran escuchados en su país natal y que sí tienen éxito apuntalando lo que se venía haciendo en Japón de la mano de Toyota y tantas otras.

Esto inicia el proceso de calidad, el movimiento de calidad en Estados Unidos que llega hasta nuestros días. Este movimiento de calidad llega hasta nuestros días con desarrollos como Six Sigma y lean, que son los que están dominando el pensamiento acerca de la calidad.

Calidad y procesos

PDCA

Mencionamos recién a Walter Shewart con sus ideas del control estadístico de procesos. Shewart también es el autor de un ciclo de mejora continua, hablar de calidad y no hablar de mejora continua es prácticamente ridículo. La calidad se debe mejorar continuamente. Encontrar mejores formas de trabajar es responsabilidad de todos, desde el primer empleado hasta el director de la compañía. Walters Shewart en los años 20 crea este ciclo de mejora llamado

PDCA o “**Plan – Do – Check – Act**”, que fue luego muy popularizado por Deming, uno de los gurúes que va a Japón como parte del Plan Marshall que mencionábamos antes. Este ciclo de mejora continua es lo que ha inspirado muchos de los procesos de desarrollo iterativos e incrementales de desarrollo de software que hoy vemos en el mercado. Lo que propone este proceso son cuatro fases:



- 1°) **Plan:** Se establecen los objetivos y el plan para alcanzar.
- 2°) **Do:** Se ejecuta el plan, que básicamente es desarrollar y probar pequeños cambios. Imaginemos que esto lo usábamos en un proceso productivo. Durante la planificación lo que tratamos de entender es cuáles son los problemas que tiene ese proceso productivo; hacemos un pequeño cambio; lo probamos; se reúnen los datos.
- 3°) **Check:** Esa mejora que hicimos, ¿sirve para resolver el problema originalmente identificado? Eso se evalúa en la etapa de check. ¿Se mejoró?, ¿se puede mejorar más?
- 4°) **Act:** Se implementan esos cambios en la fase de act, y en función de los resultados obtenidos, se inicia un nuevo ciclo para volver a identificar nuevas oportunidades de mejora.

Es un ciclo que no termina nunca, y eso ha permeado, y está en todas las escuelas de pensamiento de calidad que vemos hoy en día.

¿Lo hicimos bien?

- ☒ **Verificación**
- ☐ Validación

Enfoques vigentes hoy

Hace un momento mencionábamos a Six Sigma y a Lean como 2 de los movimientos sobre las escuelas de pensamiento de calidad que están vigentes hoy en día. Podríamos agregar una tercera: “Teoría de las restricciones” (TOC). Las 3 están emparentadas: **Six Sigma** empieza en General Electric como una metodología para resolver problemas en los procesos. Usa herramientas de distintos tipos, algunas estadísticas están muy orientadas a reducir la variación y el énfasis está puesto justamente en resolver problemas. **Lean thinking**, por otro lado, usa muchas de las mismas herramientas, pero tiene un foco muy orientado a remover el desperdicio y a facilitar el flujo de trabajo. Y la tercera, la **teoría de restricciones** está, también muy relacionado con las otras 2, y lo que busca es identificar y resolver las restricciones que tenemos en cualquier sistema productivo, en cualquier sistema en general.

Esto viene de la mano de un señor que se llamó Eliyahu Goldratt, que escribió una serie de libros muy interesantes, entre ellos “La meta”, en donde plantea, algo que, por ahí puede parecer obvio, pero a veces no tanto, y es que en cualquier sistema hay una única restricción que condiciona el funcionamiento del todo. Es como decir que en una cadena el eslabón más débil es el que termina definiendo la fortaleza de la cadena. O lo mismo que, cuando salimos a correr en grupo, y queremos todos correr al mismo ritmo, el grupo va a terminar corriendo al ritmo del que corre más lento. Lo interesante de la teoría de restricciones es que la mejora tiene que estar orientada a resolver esa restricción, a resolver esa máquina, a resolver esos problemas que tenemos con esa máquina que no está funcionando bien. Todas las mejoras que hagamos alrededor, son simplemente cosméticas. Si no nos focalizamos en resolver el verdadero problema que es esa restricción única que es la que condiciona el funcionamiento de todo el sistema.

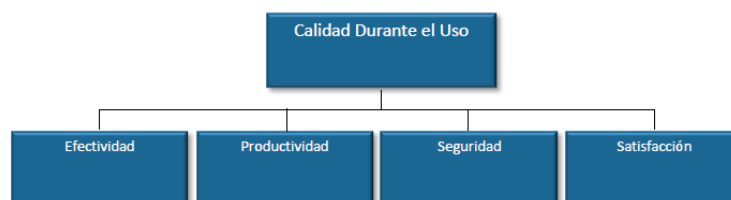
Mejora continua

Lean, por otro lado, está muy enfocado en la mejora continua, en poder eliminar esos tiempos de espera. En cualquier proceso nos vamos a encontrar que la mayoría de las actividades se quedan esperando muchas veces a que haya un OK, a que esté disponible algún tipo de recurso. En general, todas las metodologías que hemos mencionado apuntan a mejorar la calidad de forma integral, en particular lean, lo que busca es remover el desperdicio, remover esos tramos grises que están ahí para poder mejorar la calidad, acortar el tiempo de ciclo, remover las restricciones, aumentar la colaboración. Y esto que se dice fácil en la práctica es bastante complicado de implementar y la mayoría de las filosofías que hoy tenemos disponibles en el mundo del software están sumamente inspiradas, sobre todo en estas últimas 3 que hemos mencionado y en particular en lean.

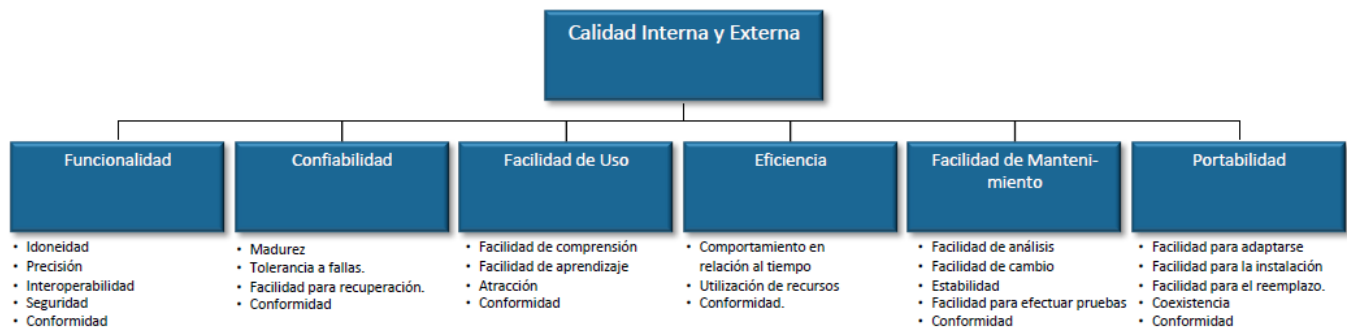
Calidad del software

En el ámbito del software, la calidad ha sido, por supuesto, una preocupación permanente. Sabemos que el software tiene un impacto altísimo en el funcionamiento de la economía moderna. Sabemos el impacto que tiene el software cuando no funciona bien. Hay 2 grandes grupos de factores de calidad de software:

- ◆ Un primer grupo, que tiene que ver con factores que son **observables durante su uso**:



- ◆ Otro grupo que son más bien **internos y externos**:



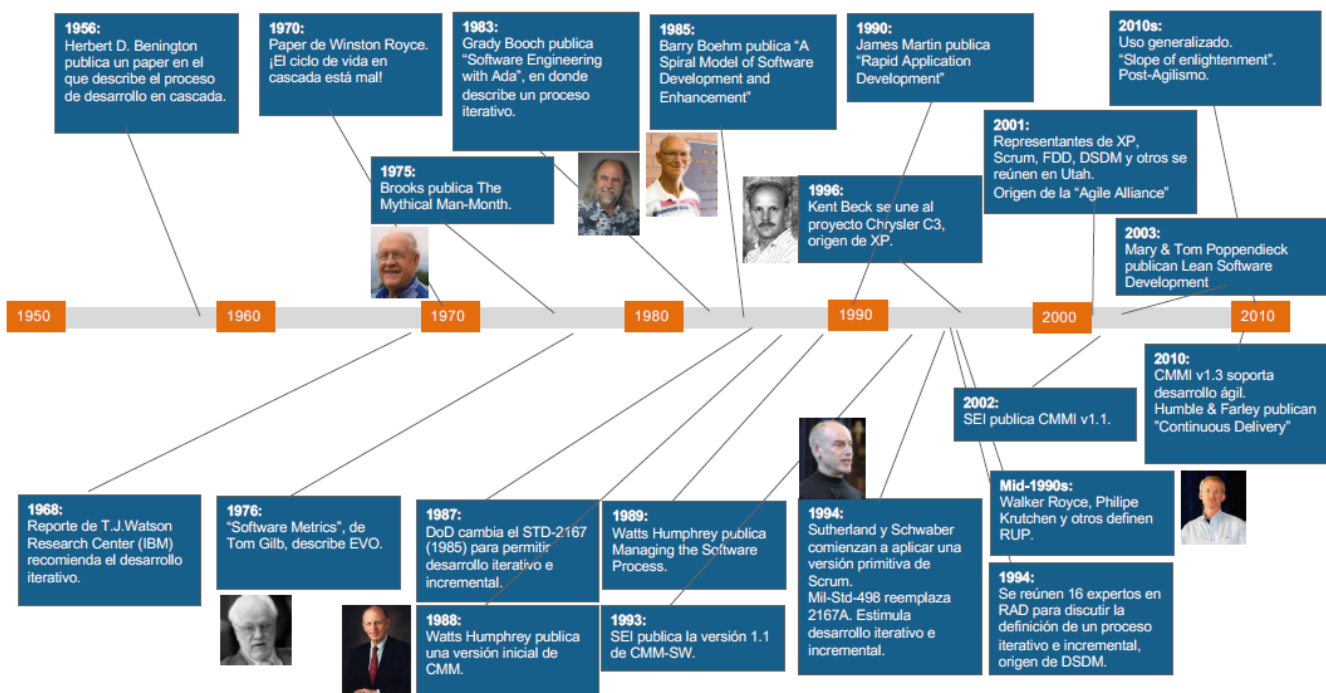
A los usuarios por supuesto, le va a interesar que el software haga correctamente lo que se supone que debe hacer, es decir, que cumpla bien los requisitos. Mientras que a los desarrolladores les va a interesar que sea fácil de mantener, que sea fácil de entender. A la gente del data center le va a interesar que hagan un buen uso de los recursos.

El *testing* está asociado con...

- ☐ Validación
- ☒ Verificación

Calidad y procesos en la industria del software

Perspectiva histórica



Prácticamente desde el inicio del software, la preocupación por la calidad ha sido algo sistemático. Al igual que pasó en otras industrias, se buscó permanentemente nuevos métodos, nuevos enfoques. Se llegó a la conclusión, obviamente que, hacía falta tener gente capacitada, buenas herramientas, pero también, buenos procesos, procesos eficientes, procesos que permitieran producir software de calidad. En el año 56 había aparecido un paper inicial, en donde se proponía desarrollar software de la misma manera que se desarrollaba hardware. Rápidamente se llegó a la conclusión de que el software era un animal distinto, que había que enfocar el desarrollo de otra manera. Ya en el año 70, Winston Royce decía que el ciclo de vida en cascada, heredado de la ingeniería de hardware, era incorrecto. En ese ciclo de vida lo que se planteaba era una idea de desarrollo secuencial, en donde primero había que entender cuál era el problema, después desarrollar la solución y después probarla, y hoy sabemos que las actividades de desarrollo de software se van ejecutando de manera no necesariamente secuencial. Sabemos que entendemos un

poco el problema, formulamos una solución, ese entendimiento de la solución me permite reformular y entender un poco mejor el problema y eso ha dado origen a infinidad de propuestas que apuntan a que el desarrollo de software es una actividad iterativa e incremental inspirados en modelos como PDCA propuesto por Shewart hace casi 100 años. Hoy contamos con infinidad de modelos de proceso o modelos de referencia que lo que proponen justamente son ciclos, ciclos continuos de desarrollo en donde vamos refinando paulatinamente nuestro entendimiento del problema y nuestra formulación de la solución.

A lo largo de estos 50 años hemos entendido claramente que el software es una entidad diferente, maleable, intangible, que requiere un proceso distinto. Ya a mediados de los 70, Tom Gilb presentaba, probablemente uno de los primeros ejemplos de propuesta de ciclo de vida iterativo e incremental. Ya antes, en el año 68, un reporte de IBM decía, advertía y sugería que el desarrollo de software no podía ser secuencial, no podía inspirarse demasiado en la ingeniería tradicional, sino que hacía falta un enfoque distinto. Hubo mucha gente que estudió la mejor forma de producir software, entre ellos Watts Humphrey, que fue uno de los pioneros del estudio de la calidad en nuestra industria. Humphrey escribió un libro famoso, publicado a fines de los 80, llamado “Managing the Software Process”, que planteaba un esquema a seguir para mejorar progresivamente la calidad del proceso de desarrollo de software a través de los llamados niveles de madurez. Hoy en día esos 5 niveles de madurez son ampliamente conocidos y se utilizan en otros en otros ámbitos. También en los 80 Barry Boehm hablaba del ciclo de vida en espiral. A mediados de los 90 ya comienzan los primeros intentos con Extreme Programming y Scrum que son, digamos, enfoques claramente inspirados en el ciclo PDCA, y llegamos a nuestros días en donde ya los métodos, los llamados métodos ágiles que están muy inspirados en PDCA, en la filosofía Lean y en tantos otros movimientos de calidad están ampliamente difundidos. Hoy la preocupación más grande pasa por ver de qué manera conseguimos llevar la agilidad más allá del ámbito de desarrollo de software, hablamos de la integración entre desarrollo y operaciones (DevOps), hablamos de integración entre desarrollo de operaciones y seguridad informática (DevSecOps), hablamos de transformación digital, hablamos de integrar no solamente desarrollo de operaciones y seguridad, sino también las áreas de negocio con los grupos de desarrollo, operaciones y seguridad.

Un largo camino se ha recorrido. La preocupación por la calidad del software sigue vigente. Hoy sabemos mucho más, entendemos mucho más. Todavía nos queda un montón por aprender y por entender. Está claro que la tarea de software no depende únicamente de tener la última herramienta de moda, de saber el último lenguaje de programación, de saber utilizar el último framework, sino que necesitamos gente capacitada, gente motivada y un muy buen proceso; un proceso que nos oriente, nos organice, no necesariamente un proceso que nos agobie, sino un proceso que nos permita ser creativos y usar esa creatividad en producir software de calidad en forma económica.

¿Cuál o cuáles de los siguientes son enfoques de calidad vigentes?

- ☐ JIT
- ☒ **Lean**
- ☒ **ToC**
- ☐ PDCA