



Universidad
Tecmilenio®



Ingeniería de Software

Estimación del costo de
desarrollo de software

Semana 11



Te invito a realizar la siguiente actividad de bienestar-mindfulness antes de comenzar a revisar el tema:

https://youtu.be/_hSq87FzoIY



En los proyectos de desarrollo de software sin duda existen costos que van desde el personal necesario para recopilar y documentar los requisitos del cliente, construir, probar e implementar el software, costos administrativos, servicios de telecomunicaciones, renta de instalaciones, entre otras.



La fijación del precio de un proyecto de software es una decisión que no solo se debe basar en la estimación del software sino que debe ser una decisión del equipo de mercadotecnia. Además, se debe considerar el obtener la mayor ganancia posible y la estrategia de negocios.



El proceso de estimar el costo del esfuerzo comienza por determinar el costo por nivel de experiencia o puesto que requiere el proyecto de software. Es posible que los niveles salariales cambien dependiendo de las habilidades técnicas de una persona.



Algunos proyectos requieren del uso de alguna licencia especial de software para apoyar el desarrollo; el costo de adquisición y entrenamiento deben ser considerados como parte de los gastos.



Es necesario agregar a la estimación del costo, posibles gastos en relación a requerir equipo especial, para probar el desarrollo o equipo necesario para acceder a la red del cliente.



Desarrolla una tabla con los costos de un sistema de gestión escolar que permita el alta de alumnos y carga de calificaciones.



Todas las organizaciones de software, que participan en proyectos de construcción de sistemas a la medida, enfrentan el dilema de proponer un precio como pago por el esfuerzo de construir el software.



- Chemuturi, M. (2009). Software estimation best practices, tools & techniques. A complete guide for Software projects estimators. USA: J.Ross Publishing.
- Sommerville, I. (2011). Ingeniería de Software (9^a ed.). México: Pearson.



Ingeniería de Software

Six Sigma en proyecto
de software

Semana 11



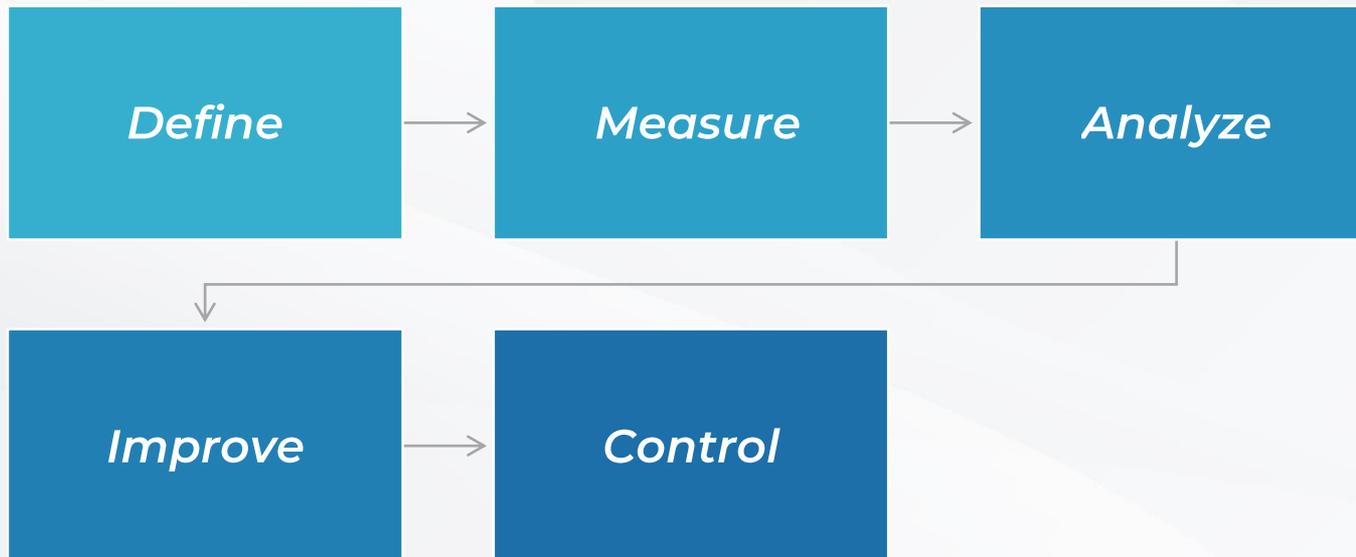
Las empresas que tienen actividades desarrollando software deben enfocar sus esfuerzos por fabricar productos con estándares de calidad muy altos, de lo contrario los clientes optarán por otras organizaciones con mejores productos.



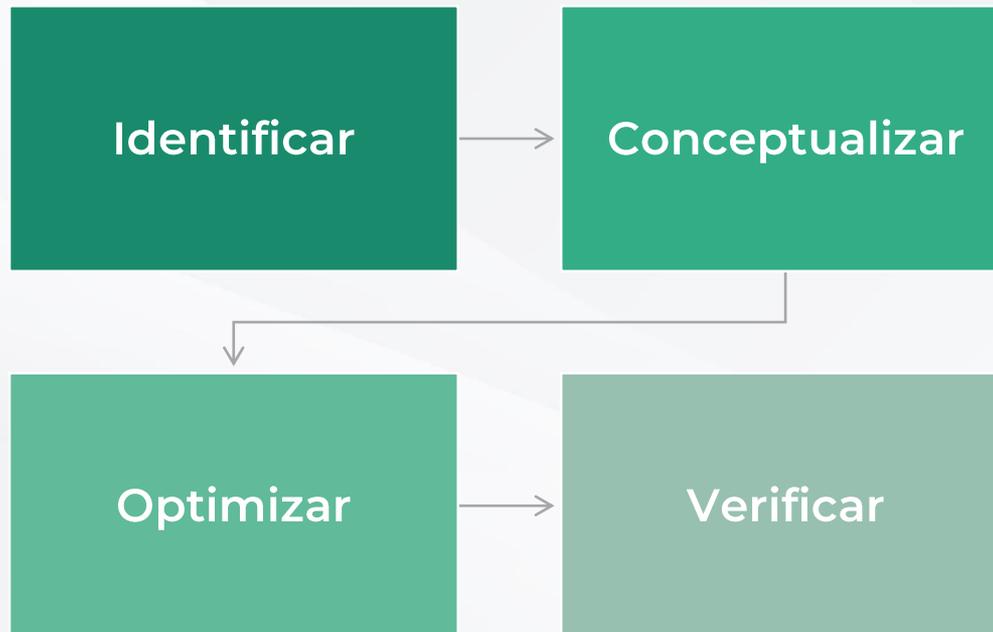
Six Sigma es una metodología que busca evitar la variación de un proceso. Su nombre proviene de la letra griega (σ) que es utilizada para expresar la desviación estándar en estadística, que es la distancia que existe al valor de la media.



La metodología de Six Sigma propone el siguiente proceso que permite mejorar los procesos hasta alcanzar cero defectos. Este proceso contiene cinco fases y se le conoce como DMAIC por sus siglas en inglés.



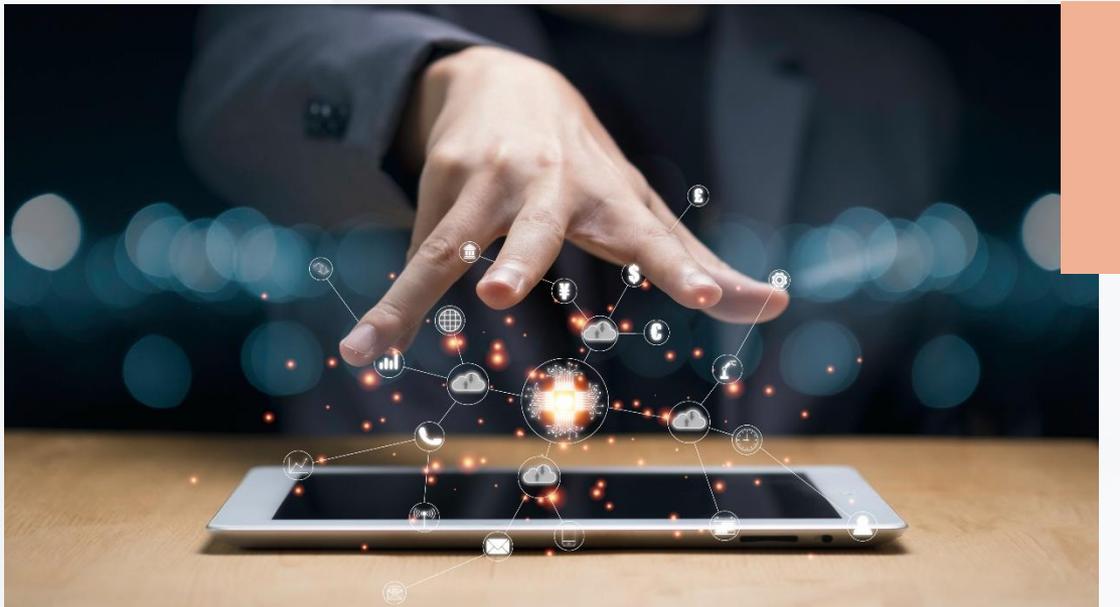
Las fases de DFSS (Design for Six Sigma) las puedes identificar por las siglas ICOV.



Describe la relación entre Six Sigma y PSP/TSP.



En la industria del software, la metodología Six Sigma se está utilizando para el proceso de desarrollo de software con el fin de eliminar los defectos y asegurar la calidad del producto.



- El-Haik, B. y Shaout, A. (2010). Software Design for Six Sigma: A Roadmap for Excellence. EEUU: John Wiley & Sons.



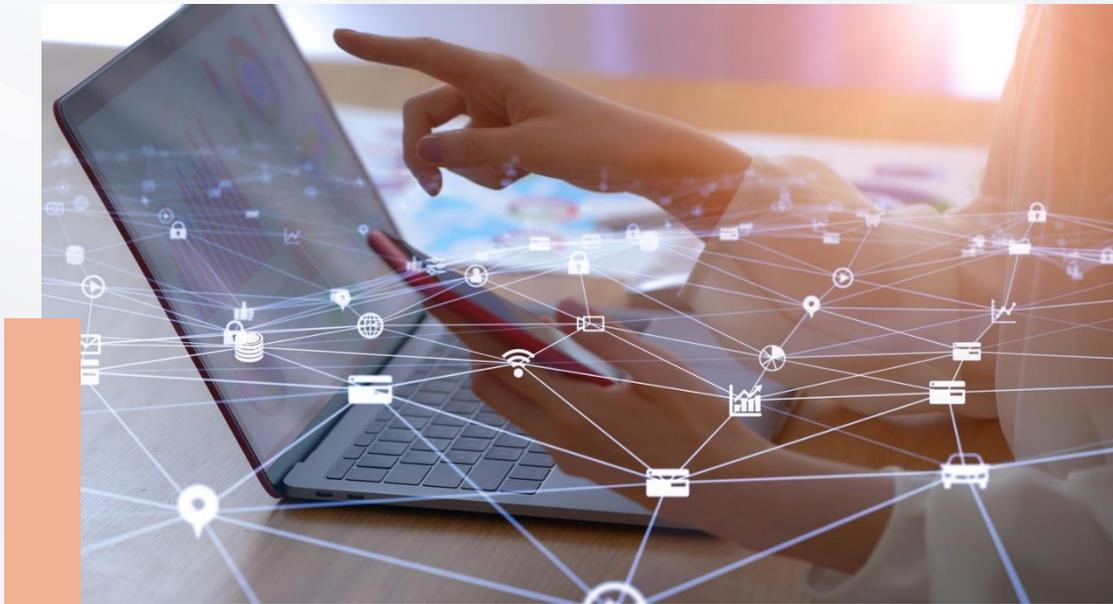
Ingeniería de Software

Fundamentos de la
calidad del software

Semana 11



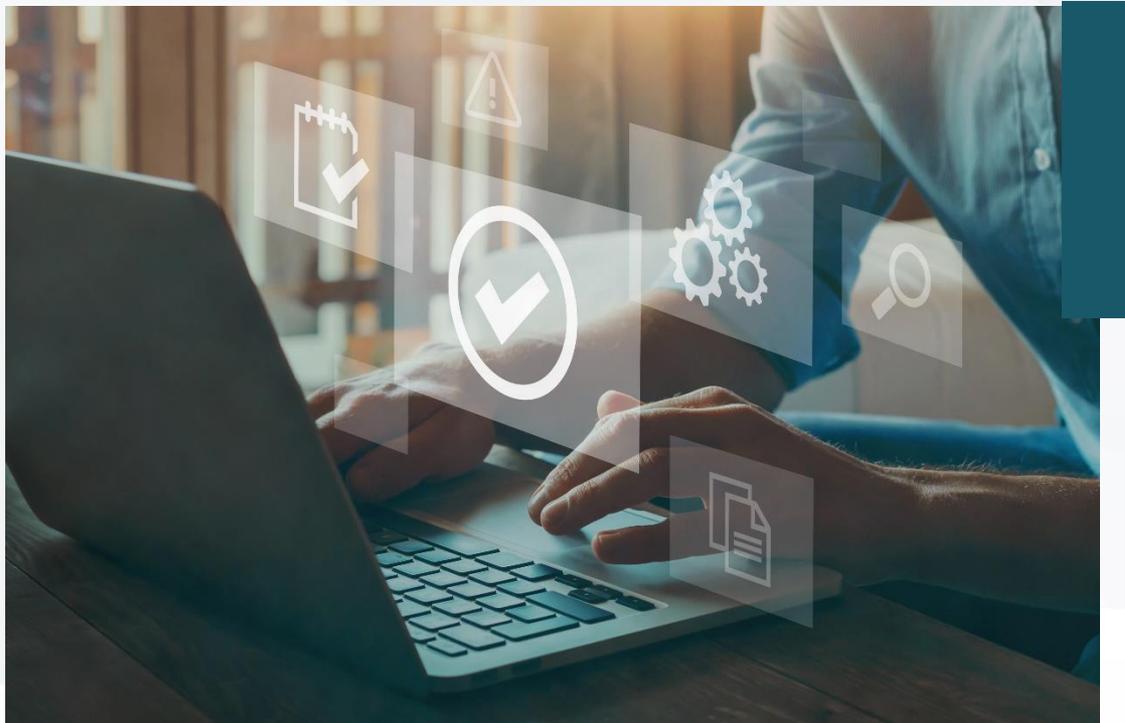
Las compañías dedicadas a la fabricación de software necesitan incluir aspectos de calidad en sus procesos si desean generar productos libres de defectos y que cumplan con las expectativas de sus clientes.



La calidad se define como las características de un producto o servicio que permiten satisfacer una necesidad declarada o implícita.



La calidad del software es “el proceso eficaz del software que se aplica de tal forma en que crea un producto útil que proporciona un valor medible a quienes lo producen y a quienes lo utilizan”.



Un estándar del software es un objeto o medida de comparación que define o representa la magnitud de una unidad.



Los métodos para gestionar la calidad son los siguientes:

Revisión del plan de calidad

Revisiones del método de diseño

Inspecciones

Perfiles de calidad y criterios de calidad



Enlista los atributos que constituyen la principal preocupación del usuario.



El camino hacia la generación de un producto de calidad no es sencillo, se requiere dedicar varias horas de análisis, establecimiento de métricas adecuadas, monitoreo del plan de calidad y el compromiso del equipo de desarrollo de enfocar todos sus esfuerzos por conseguir el mejor producto de software.



- Kendall E. y Kendall, J. (2011). *Systems analysis and design* (8ª ed.). New Jersey: Prentice Hall.
- Pohl, K. y Rupp, Chris. (2011). *Requirements Engineering Fundamentals*. EE. UU: Rocky Nook.
- Pressman, R. (2010). *Ingeniería de Software. Un enfoque práctico* (7ª ed.). México: McGraw-Hill.
- Sommerville, I. (2011). *Ingeniería de Software* (9ª ed.). México: Pearson.



Ingeniería de Software

Proceso del software
confiable

Semana 11



Cualquier empresa que desee mejorar la calidad del producto que entrega a su cliente, debe revisar el proceso que utiliza para generarlo, de otra manera solo estará esperando mejorar un resultado de forma fortuita.



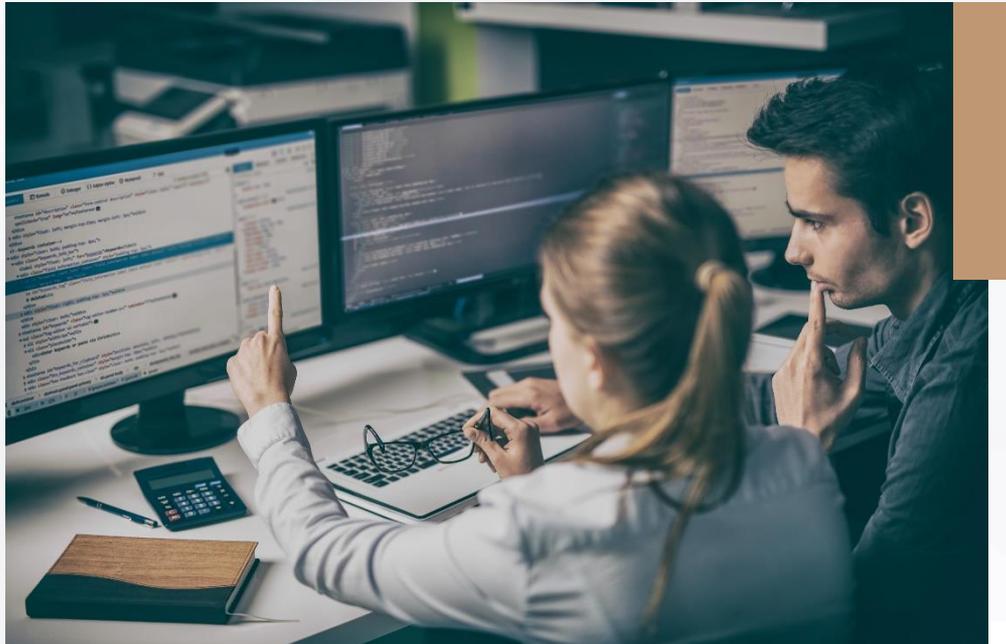
Un software confiable es aquel que realiza sus funciones de manera diligente durante su vida útil, la cual depende de la vida del hardware que lo contiene. Si el software mantiene un rendimiento sin defectos, podrás afirmar que es confiable.



Un software confiable es aquel que realiza sus funciones de manera diligente durante su vida útil, la cual depende de la vida del hardware que lo contiene. Si el software mantiene un rendimiento sin defectos, podrás afirmar que es confiable.



El diseño de la arquitectura del software es un proceso que también debe realizarse pensando en generar un producto confiable. Eso significa utilizar patrones arquitectónicos tolerantes a las fallas que incluyan mecanismos y componentes redundantes.



Describe detalladamente los procesos de ingeniería de software, aseguramiento de calidad, gestión y soporte.



El equipo técnico debe asegurarse que la recopilación y documentación de las necesidades o requerimientos del cliente estén completas y sean claras, seleccionar la arquitectura de software más adecuada, diseñar la interfaz de usuario que facilite las operaciones del sistema y evite que el usuario se confunda, elaborar los casos de prueba que verifiquen y validen el sistema, seleccionar y programar en el lenguaje que mejor se adapte a las necesidades del sistema, documentar y solucionar la mayor cantidad de defectos durante las pruebas, realizar las tareas de implementación que incluyan pruebas con el hardware, y llevar a cabo un proceso que establezca la operación del software.



- Chemuturi, M. (2010). Mastering Software Quality Assurance. EEUU: J. Ross Publishing.
- Sommerville, I. (2011). Ingeniería de Software (9^a ed.). México: Pearson.

