Imagen que contiene Código QR

El contenido generado por IA puede ser incorrecto.

**Ingeniería de Software I**

**Metodología Ágil – XP y Scrum**

**Grupo: 3S2-SIS-S**

**Docente:**

Leonel Antonio Martínez Zúñiga

**Integrantes:**

Alberto Gabriel Canda Hernández

Yesbelin Valesca Alvarado

Richard Scott Bonilla Rodríguez

21 de Agosto de 2025

**Metodología XP**

La programación extrema es una metodología ágil de gestión de proyectos que se centra en la velocidad y la simplicidad con ciclos de desarrollo cortos y con menos documentación.

Al igual que otras metodologías ágiles, la programación extrema es un método de desarrollo de software dividido en sprints de trabajo. Los marcos ágiles siguen un proceso iterativo, en el que se completa y revisa el marco al final de cada sprint, refinándolo para adaptarlo a los requisitos cambiantes y alcanzar la eficiencia máxima. Al igual que otros métodos ágiles, el diseño de la programación extrema permite a los desarrolladores responder a las solicitudes de los clientes, adaptarse y realizar cambios en tiempo real. Sin embargo, la programación extrema es mucho más disciplinada; realiza revisiones de código frecuentes y pruebas unitarias para realizar cambios rápidamente. Además, es muy creativa y colaborativa, ya que promueve el trabajo en equipo durante todas las etapas de desarrollo.



**¿Quién desarrolló la programación extrema?**

Los orígenes de XP se remontan a fines de la década de 1990, cuando Kent Beck la creó para gestionar el desarrollo de un sistema de software de nómina para Chrysler llamado Proyecto C3. El objetivo al implementar la programación extrema era (y sigue siendo) eliminar la resistencia a cambiar el código en un proyecto de desarrollo. En los métodos de desarrollo de software más tradicionales, es muy común que el código no se cambie una vez que está escrito (excepto para la depuración). Con la programación extrema, en cambio, el código se examina con tanto detalle que los desarrolladores pueden decidir modificarlo por completo luego de una sola iteración.

**¿Cuándo deberías implementar la programación extrema?**

Como la programación extrema se centra en el desarrollo de software, suele ser implementada solamente por los equipos de ingeniería. Incluso los equipos de software, suelen usarla únicamente para determinadas configuraciones. Para obtener el máximo beneficio de la programación extrema, recomendamos usarla en los siguientes casos:

* Para gestionar un equipo más pequeño. Debido a su naturaleza altamente colaborativa, la programación extrema funciona mejor en equipos pequeños de menos de diez personas.
* Si estás constantemente en contacto con tus clientes. La programación extrema incorpora los requisitos de los clientes a lo largo del proceso de desarrollo y también se basa en ellos para las pruebas y aprobaciones.
* Si trabajas con un equipo flexible que pueda aceptar el cambio (sin resentimientos). Dada su propia naturaleza, la programación extrema a menudo requerirá que todo el equipo deseche todo su arduo trabajo. Algunas reglas también permiten que algunos miembros del equipo realicen cambios en cualquier momento, lo que supondría un problema si los demás compañeros del equipo se lo toman como algo personal.
* Si dominas los aspectos técnicos de la codificación. La programación extrema no es para principiantes ya que necesitas poder trabajar e implementar cambios rápidamente.

**Ciclo de vida de la programación extrema (XP)**

El ciclo de vida de XP fomenta la integración continua, ya que requiere que los miembros del equipo trabajen casi constantemente, cada hora o todos los días. Sin embargo, el ciclo de vida completo se estructura de la siguiente manera:

* Extraer trabajos sin finalizar de las historias de usuarios
* Priorizar los elementos más importantes
* Comenzar con la planificación iterativa
* Incorporar un plan realista
* Mantener una comunicación constante con todas las partes interesadas y empoderar al equipo
* Presentar el trabajo
* Recibir comentarios
* Regresar a la etapa de planificación iterativa y repetir si es necesario

**Metodología Scrum**

El marco “Scrum”, tal como lo conocemos hoy, se introdujo por primera vez en un artículo de la Harvard Business Review en 1986 El nuevo juego de desarrollo de nuevos productos (The New New Product Development Game), escrito por Hirotaka Takeuchi e Ikujiro Nonaka. Takeuchi y Nonaka tomaron el término “Scrum” del rugby, explicando que “como en el rugby, los miembros del equipo se pasan la pelota entre sí, a medida que avanzan como una unidad por el campo de juego”.

Más tarde Ken Schwaber y Jeff Sutherland en 1993 crearon el proceso de desarrollo Scrum y en 1995 Ken Schwaber publica por primera vez un informe hablando de la metodología y Proceso de Desarrollo SCRUM y se publica también el Manifiesto ágil.

La metodología Scrum de Schwaber y Sutherland fue en parte un rechazo al modelo de cascada de desarrollo de software. En el modelo de cascada, los proyectos se dividen en fases secuenciales, donde los entregables de cada fase desbloquean la siguiente fase de trabajo. Schwaber y Sutherland creían que los programadores podrían beneficiarse de un enfoque más flexible e iterativo que les permitiera responder y adaptarse continuamente a su entorno para construir el mejor producto final para sus clientes.

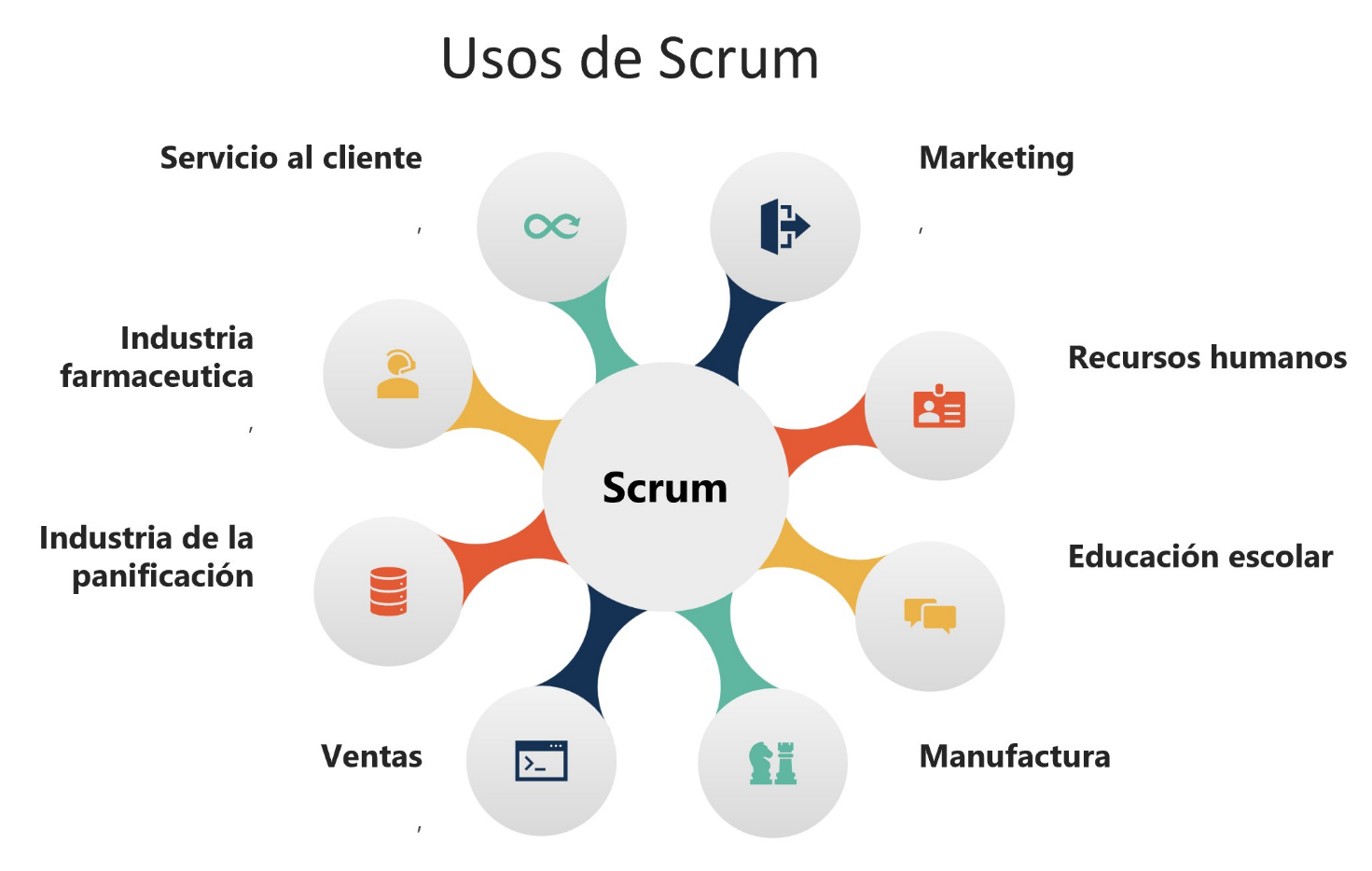
Desde su publicación inicial, Schwaber y Sutherland han publicado la Guía de Scrum, un documento dinámico que actualizan de forma regular en el que se explica detalladamente qué es Scrum y cómo ponerlo en práctica. De acuerdo con la Guía de Scrum, Scrum alienta a “los equipos a observar cuán efectivas son sus técnicas de trabajo y los desafía a evolucionar y mejorarlas continuamente”.

**¿Qué es Scrum?**

Scrum es un marco de trabajo ágil a través del cual las personas pueden abordar problemas complejos adaptativos a la vez que se entregan productos de forma eficiente y creativa con el máximo valor. Así, Scrum es una metodología que ayuda a los equipos a colaborar y realizar un trabajo de alto impacto. La metodología Scrum proporciona un plan de valores, roles y pautas para ayudar a tu equipo a concentrarse en la iteración y la mejora continua en proyectos complejos. Por otra parte, deberás saber que en Scrum se trabaja con equipos pequeños multidisciplinares en ciclos iterativos centrados en el cliente y se crea un producto de forma incremental.

**Usos de Scrum**

Ya sabes lo que es Scrum, pero tal vez te estés preguntando si es útil para tu empresa o no. Inicialmente, la metodología Scrum fue concebida para el desarrollo de software en entornos complejos. Al funcionar tan bien en este tipo de entornos su uso se extendió a otros tipos de entornos e industrias. Así, hoy Scrum es una metodología utilizada en escuelas, gobiernos, equipos de Marketing porque ha demostrado ser una gran herramienta para gestionar la complejidad.

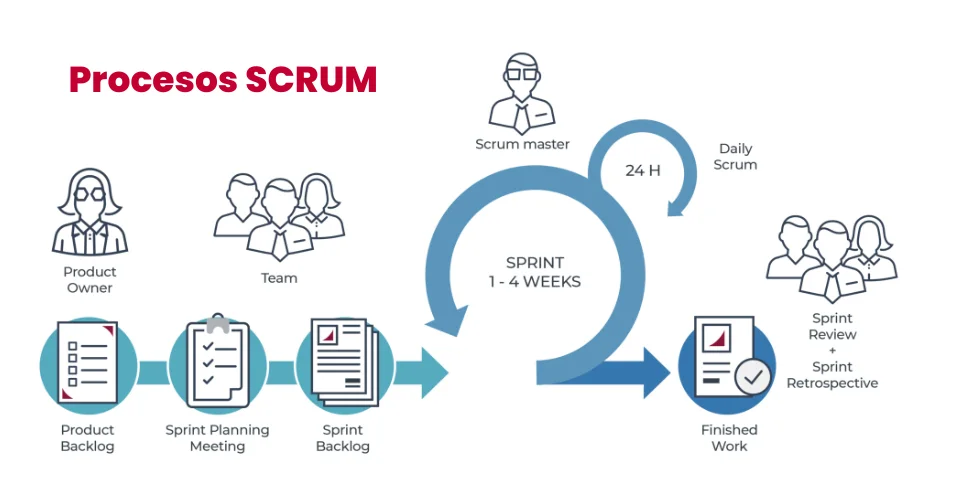


**Cómo funciona Scrum**

Lo más importante que debes saber para entender qué es Scrum y poder ejecutar un proceso basado en este marco de trabajo es tener en cuenta que se basa en un sistema de mejora continua.

La metodología Scrum se basa, como veíamos anteriormente, en tres pilares: los eventos, los roles y los artefactos. Además, en la metodología Scrum se trabaja en sprints de una duración determinada que suele fijarse entre una semana y un mes. ¿Y por qué los sprints no pueden durar más de un mes? El motivo es porque si aumenta este período también aumenta el riesgo de que tanto los requerimientos de lo que estamos desarrollando como el contexto cambien. Al final de cada sprint se genera un incremento que se suma al resto de producto desarrollado

Cuando entiendes qué es Scrum, reconoces que es posible que no sepas nada al comienzo de un sprint, y puedes ajustar tus procesos y necesidades según sea necesario en función de la información que obtengas durante el proceso de sprint.



Principios de la metodología Scrum

Existen seis principios de Scrum que ayudan a definir qué es Scrum y a aplicar la metodología Scrum para aprovechar sus beneficios. Son los siguientes:

* Control sobre el proceso empírico. Los equipos Scrum creen en la transparencia, la inspección y la adaptación.
* Autoorganización. Aunque tu equipo Scrum tendrá roles y reglas, cada miembro Scrum está facultado para asumir la responsabilidad de sus tareas y su trabajo. En Scrum se cree que la responsabilidad compartida produce equipos más creativos y dinámicos.
* Colaboración. Tu equipo obtendrá los mejores resultados si trabaja en conjunto durante y después del sprint de Scrum.
* Priorización basada en valores. El objetivo del sprint de Scrum es ofrecer el mayor valor comercial. Para hacer eso, debes dar prioridad a tu trabajo desde el inicio del proceso Scrum.
* Duración limitada (timeboxing). El proceso Scrum tiene varias actividades basadas en el tiempo, como el sprint en sí, las reuniones diarias de actualización y el análisis retrospectivo. Debido a que Scrum se basa en la creencia de la mejora continua, es importante establecer una duración limitada para el trabajo para pasar a la siguiente tarea y mejorar el trabajo futuro.
* Desarrollo iterativo. En Scrum, tu primer producto no será perfecto. Pero al construir de manera iterativa, tu equipo podrá adaptarse mejor a las necesidades del cliente y modificar el producto y sus resultados en función de la priorización basada en el valor.