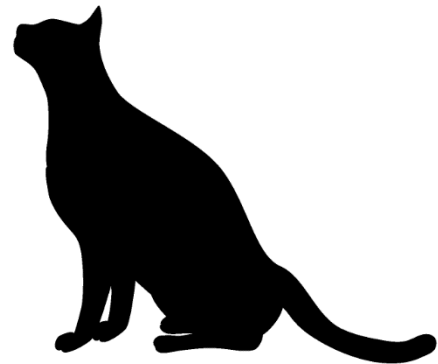


AgileByte

LEARN TO BECOME

**Ágil
como
un gato**



AgileByte

LEARN TO BECOME

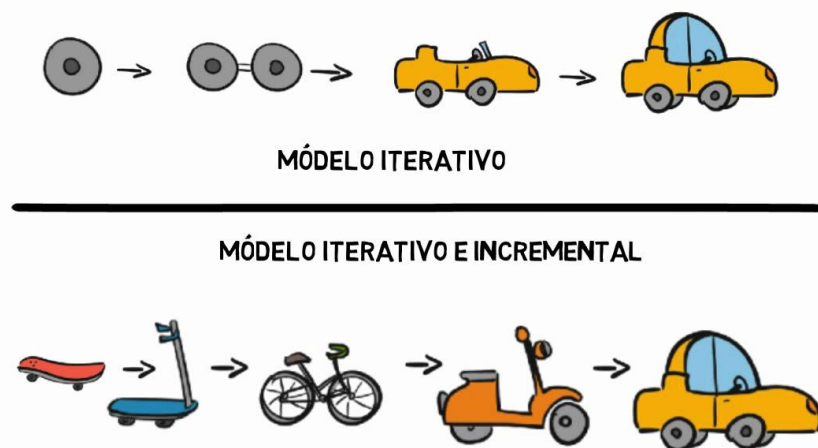
Módulo 1: Introducción a la gestión de proyectos de software ágil

Conceptos básicos de Agile. ¿De Agi-qué?

La palabra Agile engloba a todo un mundo de metodologías, principios, buenas prácticas, y términos que nos pueden confundir o abrumar.

Por ello, vamos a tratar de entender en qué consisten, comentaremos similitudes y características de las más importantes, pero sobre todo vamos a centrarnos en aterrizarlo para el desarrollo de software y qué es mejor en este mundillo.

Aunque conceptos hay muchos, vamos a ver los cuatro más importantes en la gestión de proyectos de software ágil, y para eso ten muy en mente esta imagen:



- **Entrega continua de valor:** se trata de proporcionar al cliente entregas de software iterativas e incrementales que generen valor de forma continua.
- **Adaptación a los cambios:** la metodología ágil se adapta a los cambios en el proyecto, ya sea a través de la modificación de los requisitos o la inclusión de nuevas funcionalidades.
- **Trabajo en equipo:** el equipo de desarrollo trabaja de forma colaborativa y multidisciplinaria para lograr los objetivos del proyecto.
- **Comunicación constante:** la comunicación entre el equipo de desarrollo y el cliente es constante para asegurar que el proyecto va en la dirección correcta.

Estos conceptos han sido traducidos a lemas fáciles de recordar que sintetizan perfectamente la esencia del cambio con respecto a lo que se hacía tradicionalmente



Por tanto, **Agile es mucho más que una metodología o un framework**, es un conjunto de valores y de principios a seguir para evitar que surjan típicos problemas del desarrollo de software, para eso se centra en seguir un ciclo de vida adaptativo frente a los cambios, donde el producto va evolucionando a lo largo de todo el proyecto.

Ni se diseña ni se planifica por adelantado, en su lugar se tiene una mayor flexibilidad ante la presencia de cambios por lo que el producto se va creando y presentando al cliente en intervalos cortos que se conocen bajo el nombre de 'sprints', donde idealmente cada sprint refleja una entrega de valor independiente y completa por sí misma.

Aunque es adelantarse, que te suene que todo esto aplicado al software, significa que cada sprint debe aportar una feature o funcionalidad completa, junto a sus tests y acoplable con el producto completo.

Tradicional o Ágil, ¿he ahí la cuestión!

Antes de sumergirnos en los diferentes marcos de trabajo ágiles, vamos a ver la diferencia entre lo ágil, y lo tradicional para desarrollar.

No rompemos el misterio si adelantamos que las metodologías ágiles son mucho más útiles para el mundo del software, al fin y al cabo se desarrollaron para esto...pero tampoco queremos generar la sensación de que lo tradicional es peor, sencillamente tienen aplicaciones diferentes y puntos fuertes que son aprovechados según el proyecto a desarrollar. Por ejemplo, solo con pensar en que me tocara llevar un proyecto de construcción de una presa con sprints de tres semanas a mi me da un parraque.

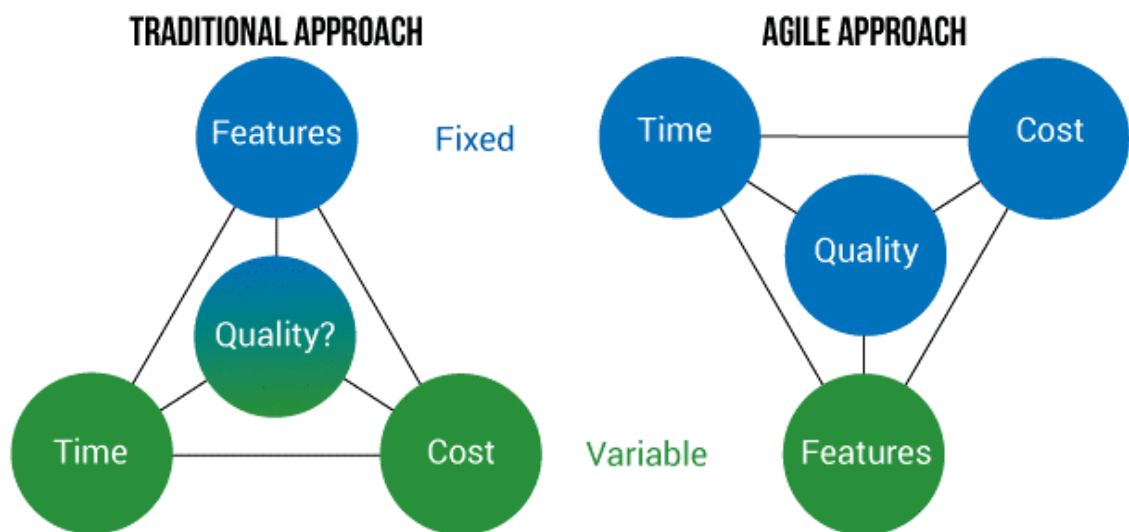


Entonces, vamos a atacar a la comparación por medio de los principios ágiles aplicados al desarrollo de software:

Aspectos de la Metodología	Tradicional	Agile
Priorización de las necesidades del cliente	Se enfoca en cumplir los requerimientos establecidos desde el principio y puede no tomar en cuenta las necesidades que surjan en el proceso	Se pone en primer lugar las necesidades del cliente, lo que significa que el software entregado generará valor para el usuario final
Entrega continua de software	El software se entrega al final del proyecto, lo que puede hacer que sea difícil ajustar el resultado final según las necesidades del cliente	El software se entrega de forma continua, lo que permite al cliente ver los avances y retroalimentar al equipo de desarrollo
Trabajo en equipo	Las decisiones son tomadas de manera jerárquica y los departamentos no colaboran de forma eficiente, lo que puede retrasar el proceso	El equipo de desarrollo trabaja de forma colaborativa y multidisciplinaria, lo que permite una mejor toma de decisiones y la resolución de problemas de forma más eficiente
Adaptación a los cambios	Los cambios pueden ser costosos y retrasar el proceso, lo que puede llevar a un producto final que no cumpla con las necesidades del cliente	Se adapta a los cambios en el proyecto para asegurar que el software entregado genere valor para el cliente
Comunicación constante	La comunicación no es constante, lo que puede llevar a una mala interpretación de los requisitos y a un producto final que no cumpla con las expectativas del cliente	La comunicación constante entre el equipo de desarrollo y el cliente permite la toma de decisiones informadas y la resolución de problemas de forma más eficiente



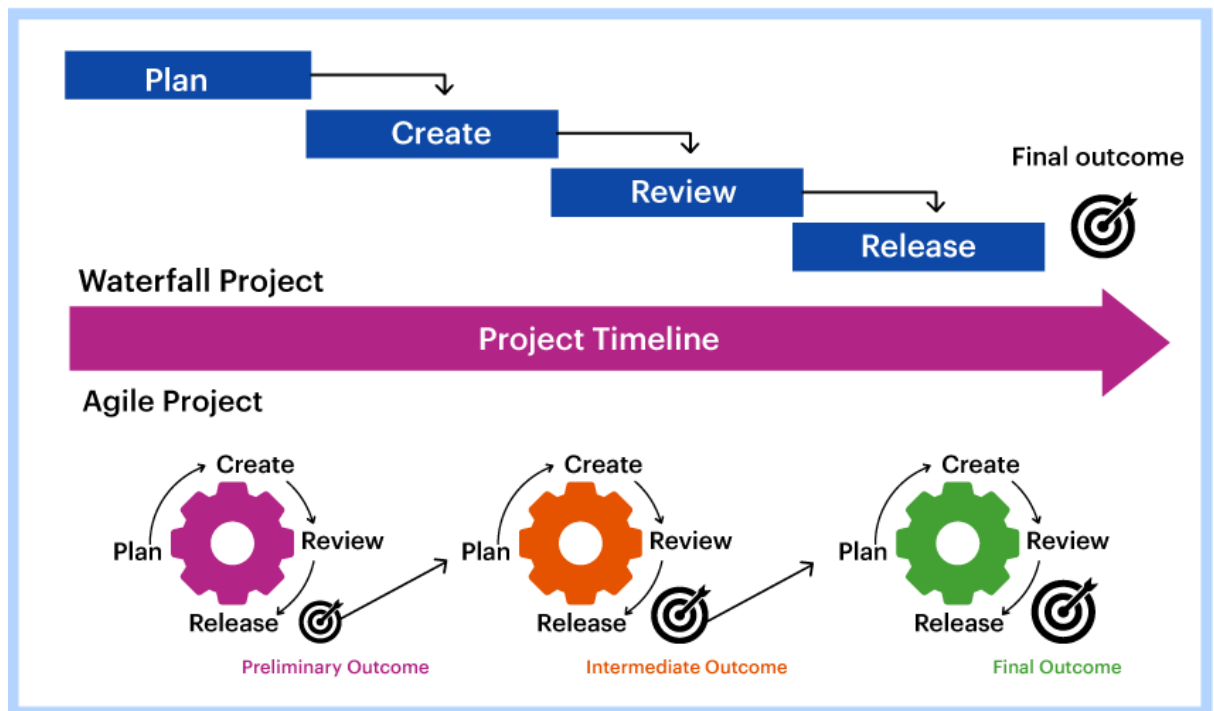
Mejora continua	Una vez finalizado el proyecto, no se enfoca en mejorar el proceso	Se enfoca en aprender de las experiencias y mejorar continuamente la metodología y el proceso de gestión de proyectos
------------------------	--	---



Resumiendo, la metodología ágil se enfoca en la entrega continua de valor al cliente, el trabajo en equipo, la adaptación a los cambios y la mejora continua. Mientras que la metodología tradicional se enfoca en la planificación detallada y la ejecución secuencial de las actividades.

Todo esto hace a la metodología ágil más flexible y eficiente por adaptarse mejor a los cambios y priorizar siempre las necesidades del cliente.

¡Cuidado!, no es oro todo lo que reluce!, más adelante veremos que esta flexibilidad genera un desgaste y unos riesgos que hay que saber gestionar y resolver, pero en su conjunto, podéis creernos cuando os decimos que merece la pena sin dudarlo.



No sólo Agile

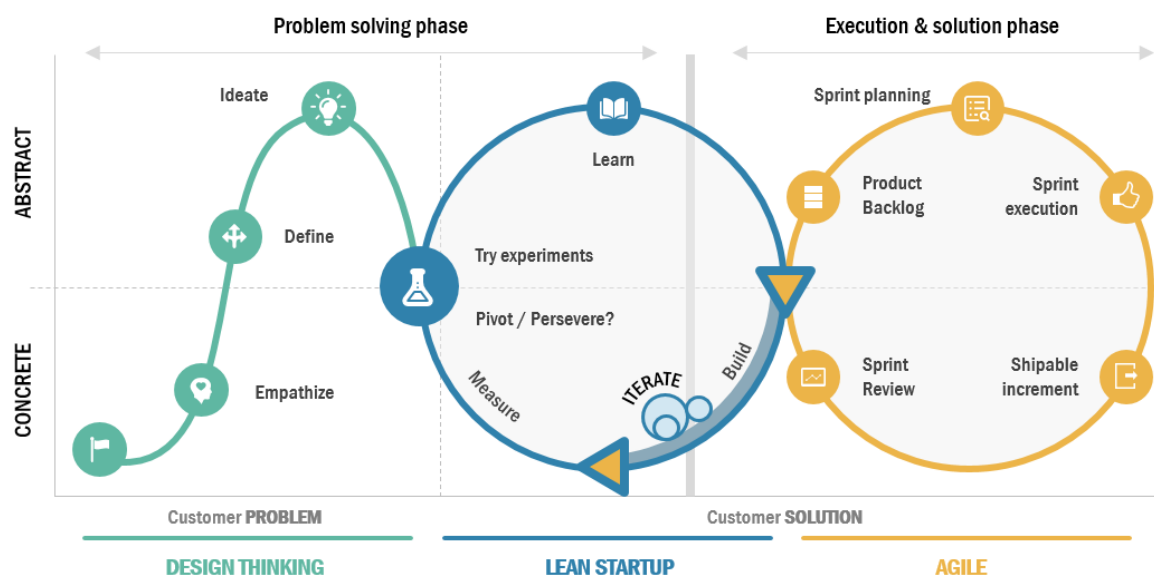
Estemos o no iniciados en la gestión de proyectos, es probable que hayas oído hablar de pensar fuera de la caja, pensamiento creativo, metodologías lean o agile y más metodologías de trabajo en una lista interminable.

Nosotros y para nuestro enfoque, vamos a hablar de las tres que nos han resultado más útiles en los proyectos de implantación de software y digitalización de procesos y modelos:

Design thinking, Lean Startup y Agile son tres enfoques diferentes, pero complementarios para el desarrollo de productos y servicios.



- **Design thinking** es un enfoque centrado en el usuario para la resolución de problemas que se utiliza para abordar problemas complejos y desafíos empresariales. Se basa en la comprensión profunda de las necesidades, deseos y comportamientos del usuario y en la creación de soluciones innovadoras que satisfagan esas necesidades.
- **Lean Startup** es un enfoque para el desarrollo de productos que se centra en la creación rápida de prototipos y en la validación continua del producto con los usuarios. Se basa en la creación de un MVP (Producto mínimo viable) y en la recopilación de datos y comentarios de los usuarios para iterar y mejorar el producto de manera continua.
- **Agile** es un marco de trabajo para la gestión de proyectos que se centra en la entrega de valor de manera rápida y frecuente. Se basa en equipos interdisciplinarios y autoorganizados que trabajan en sprints cortos y que se adaptan rápidamente a los cambios en los requisitos del proyecto.



En conjunto, estos enfoques se complementan para ayudar a los equipos a crear productos y servicios innovadores que satisfagan las necesidades de los usuarios de manera efectiva. Vamos a dejar claro lo importante y poner un ejemplo de una empresa que lo ha hecho bien aplicándolos:

- **Design thinking** proporciona un marco para la comprensión profunda de los usuarios, lo que quieren y cuales son sus problemas. Y luego fomenta aportar valor a través de soluciones innovadoras y creativas.

El mejor ejemplo es **Airbnb**. La empresa afrontó el proyecto pensando en cuales eran los problemas que tendrían sus usuarios al reservar alojamiento y resultó que

el mayor problema a dar solución era la falta de confianza de los usuarios al alquilar un espacio en línea a extraños.

Gracias a design thinking, se pusieron en el lugar del usuario y se centraron en crear

una serie de experiencias y servicios que aumentaron la confianza de los usuarios, como la posibilidad de ver las opiniones de otros usuarios, una garantía de protección y una herramienta para ver la autenticidad de las fotos de los alojamientos.



- **Lean Startup** proporciona un enfoque para la iteración rápida y la validación de la solución con los usuarios, desde el mínimo producto viable (mvp), va mejorando iterativamente.

La empresa **Dropbox** es un gran ejemplo. En lugar de construir una solución completa desde el principio, comenzaron con un simple video demostrando cómo funcionaría su producto y lo subieron a un sitio web para ver si había interés en él.

El video se hizo viral y generó una gran cantidad de interés, lo que les permitió validar su idea y obtener retroalimentación de los usuarios.

Después lanzaron su producto al mercado a través de una versión beta limitada y siguieron recopilando comentarios e iterando para mejorar su producto antes de lanzar su versión completa, lo que les permitió ahorrar tiempo y recursos.



- **Agile** proporciona un marco para la entrega rápida y frecuente de valor, ideal cuando se necesita ser muy flexible y adaptarse rápidamente a las circunstancias.

En **Spotify**, líder mundial de streaming de música, utilizaron equipos multifuncionales, procesos iterativos y colaboración constante para construir y lanzar el producto rápidamente.

Buscaban siempre agregar nuevas características en respuesta a las necesidades de los usuarios además de mejorar la eficiencia y la productividad de sus equipos de desarrollo.

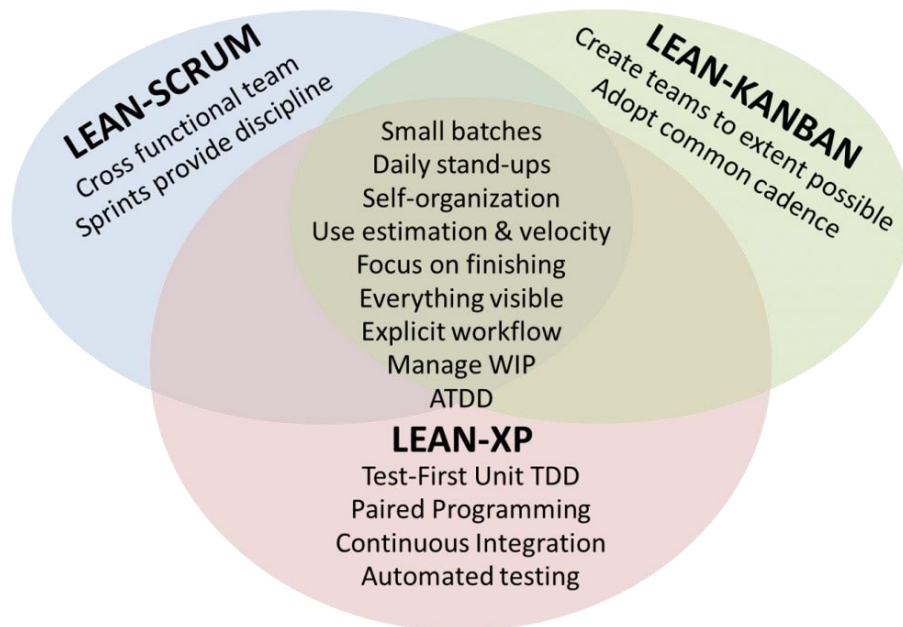


Trío de Ases: Scrum, Kanban y XP

Scrum, Extreme Programming (XP) y Kanban son los tres marcos de trabajo de desarrollo ágil que más se utilizan para gestionar proyectos de software.

Vamos a dar unas pinceladas para que nos suenen y en el siguiente módulo nos sumergiremos en profundidad.

- **Scrum** es un marco de trabajo centrado en la gestión de proyectos que se basa en sprints de tiempo fijo y entregas incrementales de producto. Se utiliza para gestionar proyectos complejos y para ayudar a los equipos a trabajar juntos de manera eficiente.
- **Extreme Programming (XP)** es un marco de trabajo que se enfoca en la excelencia técnica y en la colaboración estrecha entre los miembros del equipo. XP se centra en la entrega frecuente de software funcional de alta calidad a través de prácticas como la programación en parejas, las pruebas automatizadas y la integración continua.
- **Kanban** es un marco de trabajo que se enfoca en la gestión del flujo de trabajo y en la optimización del proceso de trabajo. Se basa en la visualización del flujo de trabajo y en la limitación del trabajo en progreso para maximizar la eficiencia y minimizar los tiempos de espera.



En conjunto, estos enfoques se complementan para ayudar a los equipos a gestionar proyectos de software de manera efectiva, igual que antes, vamos a dejar claro lo importante y poner un ejemplo de una empresa que lo ha hecho bien aplicándolos:

- **Scrum** proporciona un marco para la gestión de proyectos ágiles gracias a sus roles y ceremonias.

Google utiliza Scrum en gran parte de sus proyectos de desarrollo de software, incluyendo Google Search, Google Maps y Gmail. La empresa utiliza equipos Scrum autoorganizados para trabajar en diferentes características y proyectos, y utiliza reuniones diarias, reuniones de planificación de sprint y retrospectivas para mejorar continuamente el proceso de desarrollo.



- **XP** proporciona un enfoque para la excelencia técnica y la colaboración estrecha entre los miembros del equipo.

Microsoft ha utilizado XP en el desarrollo de sus sistemas operativos. No hay más que pensar que Microsoft lanza un sistema operativo en un plazo de tiempo muy corto y con una altísima calidad. Es el pilar de todos los programas y operaciones que se ejecutan en tu ordenador y si eso falla, el re

La empresa utiliza técnicas XP como programación en parejas, integración continua y pruebas automatizadas para mejorar la eficiencia del proceso de desarrollo.



- **Kanban** proporciona un marco para la gestión del flujo de trabajo y la optimización del proceso de trabajo.

Toyota utiliza Kanban en su proceso de desarrollo de automóviles para visualizar su flujo de trabajo, identificar cuellos de botella y mejorar la eficiencia del proceso. Además, lo usan para limitar el trabajo en progreso y mejorar la comunicación entre los miembros del equipo de desarrollo.



Los equipos pueden utilizar estos marcos de trabajo de manera combinada o en forma separada, según sus necesidades específicas y las características de su proyecto.

En el siguiente capítulo entraremos en profundidad en cada uno de estos marcos de trabajo.

