

FASTBOOK 01

# Metodología Agile

Core Desarrollo



## 01. Metodología Agile

En este primer tema vamos a hablar de las metodologías ágiles de trabajo para los equipos de desarrollo, donde se explicarán desde las tradicionales hasta las que más se utilizan en la actualidad. Adicionalmente, se explicarán las aplicaciones que se utilizan para la gestión de los proyectos mediante la metodología Agile.

Tal y como se ha comentado en el párrafo anterior, existen dos tipos de metodologías que actualmente se están utilizando a la hora de comenzar con los desarrollos de un producto o servicio.

Tradicionalmente, se utilizaba la metodología waterfall (en los próximos puntos se explicará en detalle cómo funciona); sin embargo, en los últimos años han crecido las metodologías ágiles, las cuales, como su propio nombre indican, se adaptan mejor a los posibles cambios de requisitos, ya sean de negocio, funcionales o incluso de seguridad, que las metodologías del pasado.

*Autor: Jaime Morales*

 Metodología waterfall

 Metodología Agile

 Principales metodologías ágiles

 Metodologías tradicionales vs. metodología Agile

 Herramientas metodología Agile

 Conclusiones

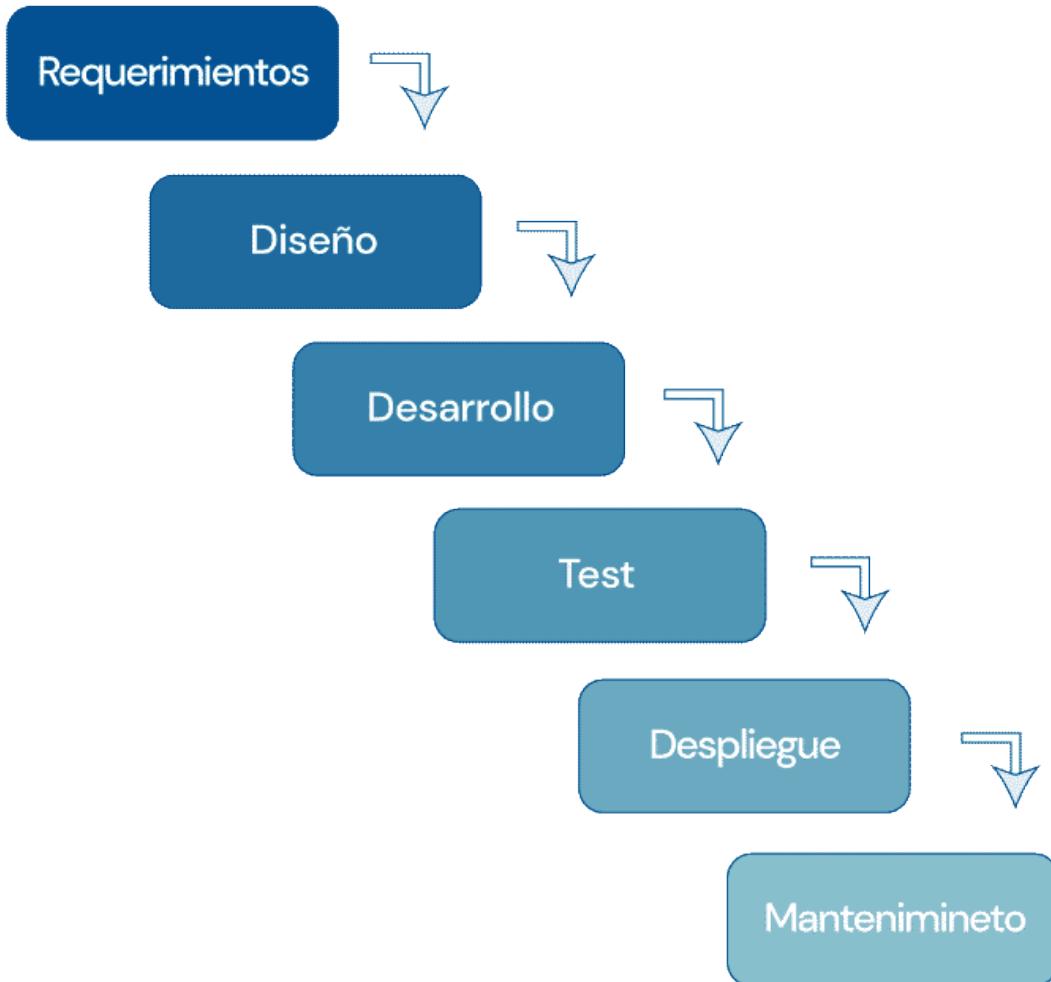
# Metodología waterfall



---

La metodología waterfall, también conocida como **desarrollo en cascada**, consiste en realizar un desarrollo de manera secuencial. Su punto de arranque, y el procedimiento más importante, es **redactar una serie de requisitos al comienzo del proyecto**. Estos deben estar cuanto más definidos mejor para que no se produzcan posibles cambios en el futuro; y, una vez definidos esos requisitos, podríamos realizar el diseño y, posteriormente, comenzar con los desarrollos.

Tal y como se muestra en la ilustración, solo cuando finaliza una tarea comienza la siguiente; por lo que podemos intuir algunas **ventajas e inconvenientes** de este tipo de metodología.



- La **mayor ventaja** es que se tiene un presupuesto cerrado, por lo que tanto el desarrollo como los plazos deberían ser los acordados y no se admiten nuevos desarrollos sin cambiar el presupuesto y/o el tiempo de desarrollo.
- Esta mayor ventaja se puede **convertir en un arma de doble filo**, ya que, si la estimación inicial de tiempo y costes es incorrecta, o no están bien definidos los requisitos, alguna de las partes impactadas en el desarrollo, ya sea un proyecto o aplicación, tendrán un **gasto adicional**.

# Metodología Agile



---

La metodología ágil ha ido cogiendo mucha fuerza en la mayoría de las compañías e incluso las mismas demuestran que su uso y cambiar la filosofía de esta ha provocado grandes mejoras a la hora de desarrollar sus propios productos. Compañías como Google, BBVA, BNP, etc., alardean ya de que la mayoría de sus proyectos se están desarrollando mediante esta metodología.

Existen dentro de la metodología Agile varios procesos diferentes, entre los que destacan principalmente **Scrum** y **Canvas**, ambos los definiremos a lo largo de este tema. Pero comencemos entendiendo qué es el método ágil.

---

La metodología Agile se define como algo más que una metodología flexible y rápida para el desarrollo de proyectos.

Es una filosofía distinta a la tradicional que permite **dividir el proyecto grande en proyectos pequeños** para permitir, hasta cierto punto, poder **modificar algunos de los requisitos iniciales**, de tal manera que el impacto sea muchísimo menor que en un proyecto con el flujo de escalada, ya que el resultado solo se podrá comprobar al final del todo.

---

**El origen de las metodologías Agile se debe a la industria del software.**

---

Esta industria comprobó que, mediante las metodologías tradicionales, se retrasaba mucho la entrega del producto final completo con todas sus partes y sin cambios adicionales.

El **objetivo** de estas metodologías, por tanto, es **desarrollar de una manera ágil algo constante y rápido** con plazos reducidos que focalicen los esfuerzos en una única parte del total del proyecto para no perder de vista el desarrollo concreto.

Bajo este precepto, veamos ahora las **principales ventajas de una metodología Agile**:

- Mejora la calidad.** Minimiza los errores en los entregables y mejora la experiencia y las funcionalidades para el cliente.
- Mayor compromiso.** Mejora la satisfacción del empleado y genera conciencia de equipo.
- Rapidez.** Acorta los ciclos de producción y minimiza los tiempos de reacción y toma de decisiones.
- Aumento de la productividad.** Al asignar mejor los recursos, y de forma más dinámica, mejora la producción según las prioridades que tenga la empresa.
- Reducción de costes.** La gestión ágil del proyecto elimina prácticamente la posibilidad de fracaso absoluto en el proyecto.

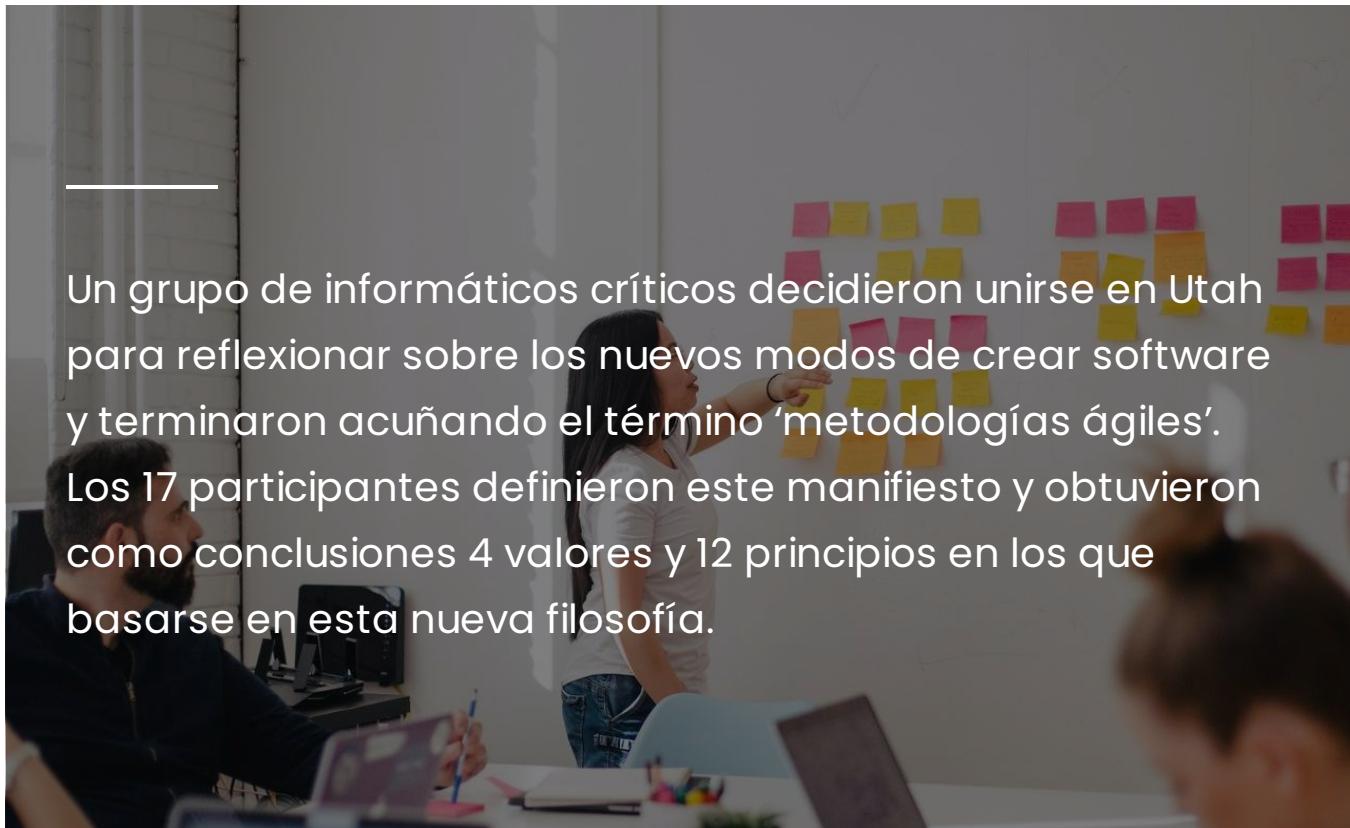
- Mayor control y capacidad de predicción.** La oportunidad de revisar y adaptar el producto a lo largo del proceso ágil permite a todos los miembros del proyecto ejercer un mayor control sobre su trabajo, cosa que permite mejorar la capacidad de predicción en tiempo y costes.
- Mejora de la calidad del producto.** Estas metodologías fomentan el enfoque proactivo de los miembros del equipo en la búsqueda de la excelencia del producto. Además, la integración, comprobación y mejora continua de las propiedades del producto también mejora considerablemente el resultado final.
- Mayor satisfacción del cliente.** El cliente está más satisfecho al verse involucrado y comprometido a lo largo de todo el proceso de desarrollo. Mediante varias demostraciones y entregas, el cliente vive a tiempo real las mejoras introducidas en el proceso.

## Manifiesto de metodología Agile

Existe un manifiesto de metodología Agile, **datado en 2001**, que define, para el desarrollo de software, una serie **valores y principios** sobre las nuevas formas de desarrollar mediante este método ágil.



Puedes consultar el manifiesto de metodología Agile en este [enlace](#).



Un grupo de informáticos críticos decidieron unirse en Utah para reflexionar sobre los nuevos modos de crear software y terminaron acuñando el término 'metodologías ágiles'. Los 17 participantes definieron este manifiesto y obtuvieron como conclusiones 4 valores y 12 principios en los que basarse en esta nueva filosofía.

Los **cuatro valores del manifiesto** estaban **centrados en las personas y la colaboración**, cuyo objetivo era ofrecer buenos productos a los clientes, donde se le daba más valor a las propias personas que formaban parte del proyecto.

Estos valores son los siguientes:

### **Los individuos e interacciones en los procesos y las herramientas**

En otras palabras: las personas son lo más importante por su capacidad para ser creativas e innovar en los procesos, estos deben adaptarse a la organización y no las personas. Es super importante que las personas se adapten de manera rápida a los cambios que van surgiendo, lo que provoca que esa adaptación se centre en el éxito de dichas personas y, por consecuencia, en el éxito de los proyectos de software.

## **Un software funcionando está en una documentación exhaustiva**

En este valor se quiere reflejar que es más importante que el desarrollo funcione correctamente que un documento donde se exprese con detalle los requisitos, por lo tanto, es más interesante el feedback que puedan dar los usuarios que cualquier otra documentación. No obstante, hay que dejar claro que el manifiesto no descarta la documentación, pero sí le da algo de menos valor frente a las metodologías más tradicionales.

## **La colaboración con el cliente prima sobre la negociación contractual**

Este valor defiende que el producto que se va a crear necesita retroalimentación continua durante su desarrollo. El cliente es un miembro más del equipo lo que provoca más valor que el cumplimiento estricto del contrato, aunque siempre hay que mantener también un equilibrio y no dejar que todo lo definido se pueda cambiar, sin que esto afecte al tiempo que se va a tardar en desarrollar y al coste.

## **La respuesta ante el cambio es más importante que seguir un plan**

El último de los valores de Agile define que no tiene sentido ser estrictos en escenarios volátiles, como es el desarrollo de software, por lo que destaca la capacidad de respuesta y adaptación a los cambios. Adicionalmente, promueve la anticipación frente al control y la planificación rígida, ya que estas últimas pueden provocar fracasos en los proyectos y, además, están más relacionadas con las metodologías tradicionales. Por último, no se renuncia a lo que los creadores de la metodología definen como componentes: los procesos, la documentación, los contratos y/o la planificación.

Una vez descritos los valores que aporta la metodología Agile, se van a describir **los doce principios** sobre los que se desarrolla la nueva filosofía Agile.

1

**Satisfacer al cliente mediante la entrega temprana y continua.** El primero de los principios pone el foco en la satisfacción del cliente, de tal manera que se realiza una entrega de software temprana y con frecuencia para que el cliente pueda validar los avances realizados. Gracias a este tipo de entregas, desde el primer momento, los clientes reciben el valor por el que están pagando y los desarrolladores, el feedback sobre su trabajo.

2

**Aprovechar el cambio como ventaja competitiva.** Se deben aprovechar los cambios en beneficio del cliente, aceptando modificaciones en los requisitos hasta en las últimas fases de un proyecto. En la gestión tradicional, un cambio al final del proyecto puede ser desastroso para el desarrollo por todo lo que puede conllevar en cambios. De este modo, el hecho de aprovecharlos aporta al cliente una ventaja competitiva, ya que atiende a las necesidades que van surgiendo a lo largo del proyecto.

3

**Entregar valor frecuente.** El tercer principio es una ampliación del primero, ya que nos habla de entregar actualizaciones del software más pequeñas cada menos tiempo, de manera que se reduzcan las posibilidades de error al tener mayor feedback del cliente. Este principio se aplicará cada vez con más intensidad al producir cambios semanalmente o incluso, en algunos casos, a diario.

4

**Cooperación negocio-desarrollares durante todo el proyecto.** Este manifiesto trata de poner fin a las barreras que existen entre negocio y desarrollo de software, con el objetivo de mejorar la compresión y la colaboración para conseguir mejores resultados. Cuanto mejor se entiendan ambos mundos, mejor será el producto que se entrega al final del desarrollo.

5

**Construir proyecto entorno a individuos motivados.** Ya se ha comentado que los miembros del equipo son clave para el proyecto o producto, por lo tanto, es necesario potenciar que se sientan motivados y que así puedan una mejor solución. Además, la confianza en el equipo para el desarrollo de software debe ser fundamental ya que un ambiente correcto y adecuado aumenta las posibilidades de obtener buen producto final.

6

**Utilizar comunicación cara a cara.** El sexto principio habla de la forma más adecuada para la ejecución de los proyectos de forma exitosa, es decir, la comunicación directa cara a cara. De hecho, garantiza que es la más efectiva ya que se reducen los tiempos de respuesta y los posibles malentendidos por posibles envíos de mensajes, correos, etc. Desde 2020 y con el auge del teletrabajo, se ha extendido bastante el uso de numerosas herramientas digitales para videoconferencias que facilitan este tipo de comunicaciones.

7

**Software funcionando como medida de progreso.** Es necesario ver el software en funcionamiento y no mostrar una diapositiva donde se puede enseñar cualquier cosa al cliente. El hecho de mostrar el software funcionando crea un vínculo de confianza con el cliente ya que es la evidencia de que se está trabajando en el mismo y ofrece más detalles del proceso de construcción.

8

**Promover y mantener un desarrollo sostenible.** El octavo principio persigue optimizar la forma de trabajar para evitar sobrecargas y poder entregar soluciones de manera frecuente con la diferencia adicional de que esta entrega aporte valor al cliente. Todas las partes involucradas en el proceso de desarrollo de software deben mantener un ritmo que pueda ser seguido por todos ellos. Lo habitual es que este principio vaya enfocado a equipos de software que tienen cargas de trabajo muy altas, una situación que suele conllevar a posibles errores o fallos, eso sin mencionar el aumento de tensiones en el equipo.

9

**La excelencia técnica mejora la agilidad.** Cuidar los aspectos técnicos a la hora de desarrollar un producto de software aporta agilidad, ya que será más sencillo realizar actualizaciones cuando se ha construido cuidadosamente y además cuenta con un buen diseño. Es importante destacar que normalmente la excelencia técnica tiene un valor intangible y los usuarios finales y clientes no le dan tanto valor, pero a la hora de realizar nuevas versiones se verán directamente afectados ambos lados, tanto el cliente como los desarrolladores de software, lo que impacta directamente en los costes del proyecto.

10

**La simplicidad es fundamental.** El décimo principio propone que actuemos de la forma más sencilla posible, de tal manera que el cliente no tenga que pagar por el esfuerzo realizado y sí por una solución que realmente atienda a sus necesidades. Dicho de otro modo: hay que intentar evitar en la medida de lo posible esfuerzos innecesarios y dedicar tiempo a lo que realmente aporta valor.

11

**Equipos autoorganizados para generar más valor.** El penúltimo postulado afirma que los equipos con mayor libertad de decisión cuentan con una mayor confianza y, por ende, son los que consiguen mejores resultados. Este principio está vinculado realmente a los anteriores de manera directa o indirectamente, ya que bajo el subyace el objetivo de alimentar una buena comunicación y colaboración entre negocio y los desarrolladores de software.

12

**Reflexión y ajustes frecuentes del trabajo de los equipos.** El último manifiesto hace referencia a la idea de mejora continua, es decir, los equipos deben revisar frecuentemente su trabajo y ajustarlo a las necesidades del proyecto, tarea que mejora del rendimiento.

# Principales metodologías ágiles



---

Existen diferentes **tipos de metodologías Agile**, aunque las más conocidas son **Scrum** y **Kanban**, las cuales albergan diferentes filosofías dentro de la metodología Agile con sus ventajas y desventajas cada una. A continuación, se van a describir los principales puntos de cada una de ellas para entender a qué tipos de proyectos se pueden asociar.

## Scrum

---

La metodología Scrum es un proceso cuyo objetivo principal es fomentar el trabajo en equipo.

Además, otro de los objetivos es mejorar la competitividad entre los equipos, tanto en los equipos de desarrollo de software como en los equipos de negocio.

De acuerdo con este método los equipos realizan **entregas** cada cierto tiempo, es decir, por sprints.

---

**Un sprint, por tanto, es un plazo de tiempo, normalmente de 2 a 3 semanas, donde se desarrollan las 'historias de usuario' (unidad de trabajo) acordadas en una reunión denominada 'Sprint Planning'.**

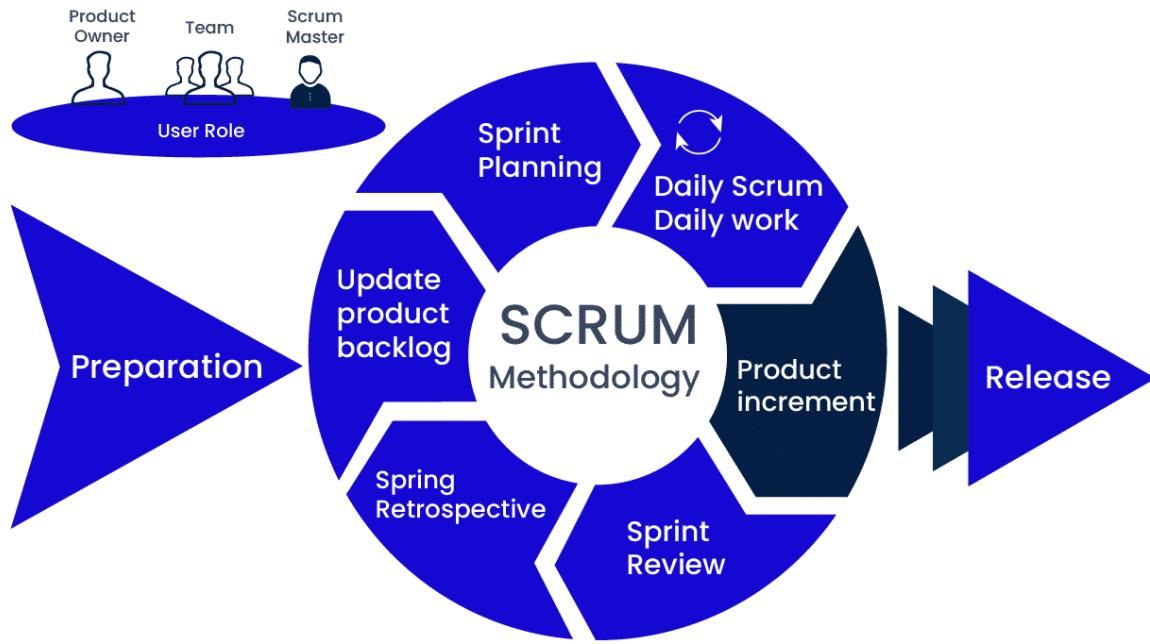
---

Una vez finalizado el sprint, se realiza una **evaluación de las tareas y procesos** en otra reunión llamada 'Sprint Review'.



Si quieras consultar historias de usuarios con ejemplos, haz clic en este [enlace](#).

La metodología Scrum se debe **aplicar en circunstancias donde, sobre todo, hay situaciones de incertidumbre**, por ejemplo, cuando los requisitos no están suficientemente definidos o no son tan claros como deberían. Esto logra que se resuelvan algunos de los problemas que surgen, ya que, gracias a las entregas, cada poco tiempo el cliente puede aportar ideas o decir las posibles mejoras del desarrollo que se está llevando a cabo.



También facilita la posibilidad de **reaccionar frente a los competidores directos**, más una ventaja añadida: el poder identificar y solucionar ineficiencias de forma sistemática.

Ahora vamos a definir cuáles son los **profiles** de la metodología:

1

### **Product owner**

Dueño del producto quien debe entender las necesidades de los stakeholders, es decir, de los clientes y los usuarios. Las **funciones principales de este rol** son las siguientes:

- Definir la visión del proyecto.
- Ayudar a finalizar la elección del Scrum Master para el proyecto.
- Identificar a los stakeholders.

- Priorizar los elementos de la lista priorizada de pendientes del producto (product backlog).
- Ayudar a crear historias de usuario.
- Mantener la lista priorizada de pendientes del producto.
- Aceptar y rechazar los entregables.
- Representar a los usuarios del producto o servicio con un profundo conocimiento de la comunidad de usuarios.
- Asegurar los recursos financieros del proyecto.

2

## Scrum Master

Quien modera y facilita las interacciones del equipo como facilitador y motivador.

---

**Este rol es el responsable de mantener y conservar un buen ambiente de trabajo con el objetivo de que el equipo sea más productivo.**

---

Estas son sus **tareas**:

- Ayudar a identificar a los stakeholders para el proyecto.

- Facilitar la selección del equipo Scrum.
- Garantizar que los recursos de respaldo estén disponibles para el funcionamiento del proyecto sin problemas.
- Ayudar al product owner en la creación de la lista priorizada de pendientes del producto y en la definición de criterios de aceptación.
- Determinar la duración del sprint.
- Apoyar al equipo Scrum en la estimación del esfuerzo necesario para completar las tareas acordadas para el sprint.
- Ayudar a actualizar el tablero Scrum y el registro de impedimentos.
- Facilitar las reuniones de revisión de la lista priorizada de pendientes del producto.
- Se asegura que los problemas que afectan al equipo Scrum se discutan y se resuelvan.
- Garantizar que exista un ambiente ideal para el equipo Scrum.

### 3

### Scrum Team

También conocido como el **equipo de desarrollo**, son las personas que trabajan en las historias de usuario y crean los entregables del proyecto. ¿Y cuáles son sus tareas principales?

- Asegurar una comprensión clara de las épicas y prototipos.

- Entender las historias de usuario en la lista priorizada de pendientes del producto.
- Estimar las historias de usuario aprobadas por el product owner.
- Asignar las historias de usuario que se hacen en un sprint.
- Desarrollar la lista de tareas en base a las historias de usuario ya convenidas y las dependencias.
- Crear entregables.
- Actualizar la tabla del trabajo pendiente y el tablero Scrum.
- Realizar las reuniones diarias de pie (daily).
- Identificar oportunidades de mejora en la reunión de retrospectiva del sprint.
- Participar en la reunión de retrospectiva del proyecto.



Puedes consultar estos perfiles de forma más detallada [aquí](#).

De acuerdo con el blog [comparasoftware.com](http://comparasoftware.com), la metodología Scrum tiene **5 fases principales**:

## **Inicio**

Identificar las necesidades básicas del sprint.

## **Planificación y estimación**

Crear, estimar y comprometer historias de usuario, identificar, estimar tareas y crear el sprint backlog o iteración de tareas (sprint 'Planning Inicial').

## **Implementación**

Discute el tiempo de desarrollo y se llevan las historias de usuario definidas para el sprint (sprint 'Planning/Daily').

## **Revisión y retrospectiva**

Revisión del proceso y/o evaluación autocrítica del equipo ('Spring Review').

## **Lanzamiento**

Desenlace del proyecto y entrega del producto.

Por último, repasemos los **conceptos destacados de su metodología** que debes manejar:

### ***Historias de usuario***

Representación de un requisito escrito en una o dos frases utilizando el lenguaje común del usuario.

### ***Sprint Planning***

Reunión cuyo objetivo es saber qué se puede obtener como resultado del sprint y cómo se logrará dicho trabajo.

### ***Daily***

Reunión corta diaria donde se cuenta los avances del día anterior y lo que se va a hacer en el día actual.

### **Sprint Review**

Reunión donde se verifica que se ha cumplido lo acordado en el 'Sprint Planning'.

### **Sprint Backlog**

Conjunto de tareas, identificadas por el Scrum Master, que tendrá que ser finalizadas en el total de sprints realizados por el equipo de desarrollo.

### **Sprint Retrospectiva**

Reunión que se realiza para ver qué detalles se pueden mejorar de cara al siguiente sprint.



Para saber más, consulta la guía de Scrum en este [enlace](#).

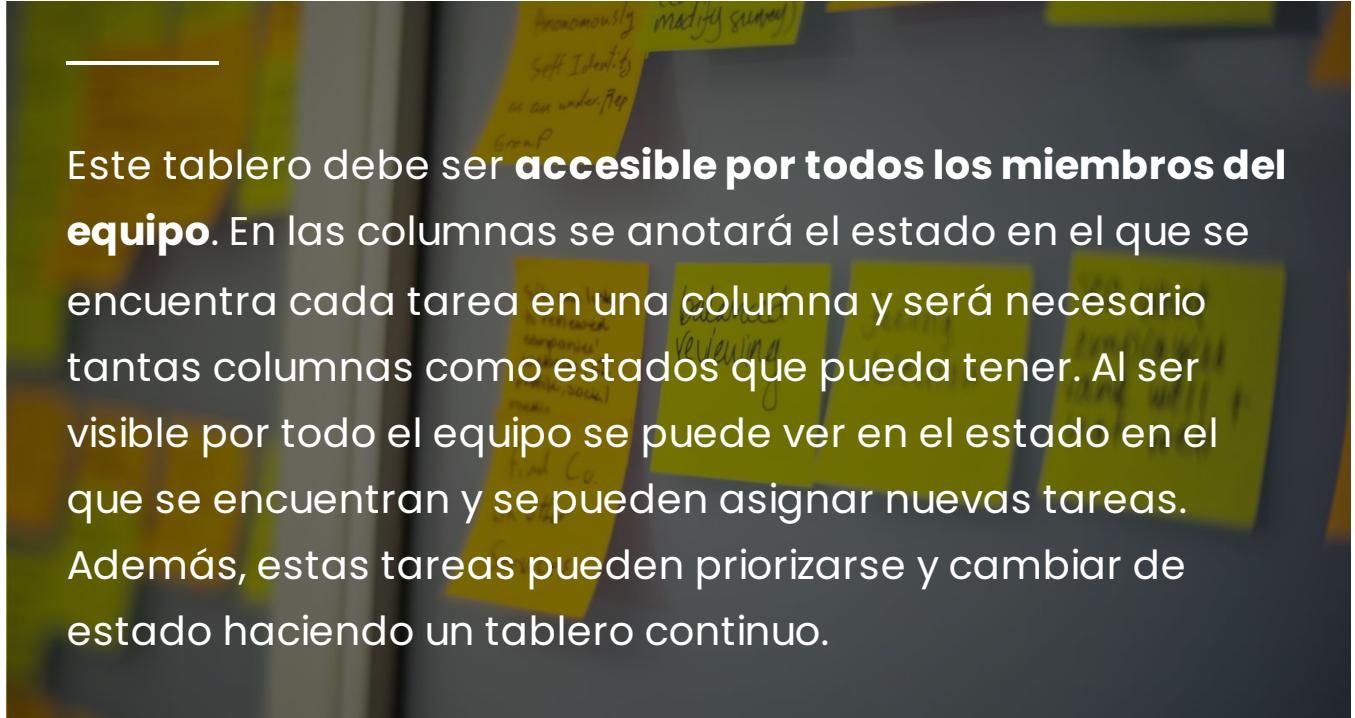
## Kanban

La [metodología Kanban](#) es un **sistema de producción tan eficiente como efectivo**. El objetivo es realizar las tareas hasta su finalización.

---

Kanban es una palabra japonesa formada por 'kan', que quiere decir visual y 'ban' que significa tarjeta, es decir, tarjetas visuales.

Es una **metodología muy sencilla** de adoptar por lo que los equipos podrían comenzar a utilizarlo sin problema, **lo mínimo que se necesitaría es un tablero** donde se definen tareas sencillas y marcar un ritmo asiduo para actualizarlo.



Este tablero debe ser **accesible por todos los miembros del equipo**. En las columnas se anotará el estado en el que se encuentra cada tarea en una columna y será necesario tantas columnas como estados que pueda tener. Al ser visible por todo el equipo se puede ver en el estado en el que se encuentran y se pueden asignar nuevas tareas. Además, estas tareas pueden priorizarse y cambiar de estado haciendo un tablero continuo.

Las **columnas** más habituales son las siguientes:

- To do:** tareas a realizar.
- In progress:** tareas que se están realizando.
- Blocked:** tarea bloqueada por cualquier motivo.
- To review:** tareas que se van a revisar.
- Done:** tareas realizadas.

Las **ventajas** que ofrece Kanban son las siguientes:

TRANSPARENCIA	EVITAR TAREAS INEFICIENTES	CONTROL DE TAREAS	FLEXIBILIDAD
Todos los miembros del equipo saben las tareas y el estado en el que se encuentra la tarea.			

TRANSPARENCIA	EVITAR TAREAS INEFICIENTES	CONTROL DE TAREAS	FLEXIBILIDAD
Evitar sobreproducción y limitación de los recursos.			

TRANSPARENCIA	EVITAR TAREAS INEFICIENTES	CONTROL DE TAREAS	FLEXIBILIDAD
Control del esfuerzo y se mejora la planificación.			

TRANSPARENCIA	EVITAR TAREAS INEFICIENTES	CONTROL DE TAREAS	FLEXIBILIDAD
Al surgir tareas imprevistas el equipo tiene una mejor capacidad de adaptación.			

## Metodología Lean

La metodología Lean es **una de las metodologías ágiles más famosas de los últimos tiempos** ya que pone el foco en las soluciones y sus usuarios centrándose en el valor de los recursos y la optimización de tiempo.

Esta metodología surge en Japón a finales del siglo XIX, a partir de la cultura que adoptaron las empresas japonesas en su búsqueda por la **optimización del trabajo** en las plantas de fabricación. Estas consiguieron increíbles resultados tanto en los puestos de trabajo como en las líneas de producción.

El primer pensamiento Lean Manufacturing vino de la mano de Sakichi Toyota, **fundador del Grupo Toyota**, que implementó esta metodología a finales de la década de los 40, cuando puso los cimientos de su fábrica. Este tenía un objetivo claro: reducir los procesos que no aportaban al producto final. Al hacerlo, lograron mejoras significativas en productividad, haciendo más eficientes los ciclos y costes de producción.

Rápidamente, su modelo se difundió entre las industrias de Japón y más tarde se extendería a Occidente bajo el término Lean, adaptándola y aplicándola a los distintos sectores productivos. Esta filosofía **transcende la metodología** constituyendo una manera de pensar que lleva a maximizar los recursos llegando a permear en muchos sectores, entre ellos, como es nuestro caso, Lean UX.

Los principios de esta filosofía se basan en la **optimización de recursos y tiempo para el desarrollo de productos de alta calidad**. La idea central es maximizar el valor de cara al cliente y minimizar las actividades o elementos que no repercutan en un beneficio directo (aquellas por las que el cliente no está dispuesto a pagar, por ejemplo, y en nuestro campo podríamos hablar de invertir tiempo y esfuerzo en funcionalidades que el usuario no va a utilizar), o lo que es lo mismo, crear más valor con menos recursos.

Como decimos, Toyota fue la pionera, pero podemos destacar otras marcas que han basado sus procesos en esta metodología, como Nike, Intel, Ford, etc. También es **la principal metodología de trabajo del modelo startup** que conocemos hoy en día.

El **principal objetivo de Lean es identificar el valor desde la perspectiva del cliente**. Hay que descubrir lo que es valioso para ellos y centrarse en eso. Es necesario identificar el flujo de valor de cada servicio o producto para poder eliminar todas las actividades y aspectos innecesarios que no añaden valor, mejorando así el flujo de actividades que podrá desarrollar sin aquellas que no aporten o incluso puedan llegar a entorpecer.

Además de eliminar las actividades innecesarias, se deben fortalecer aquellas que sí añaden valor, **eliminando aquellos obstáculos que interfieren en el proceso y reduciendo tiempos y costes**, y trabajando en mejora continua a través de procesos iterativos y de refinamiento.

Esta filosofía **busca la perfección y la calidad** a través de la mejora continua, busca la satisfacción del cliente y reduciendo tiempos y costes sin que ello signifique reducir calidad.

- i** Si te apetece profundizar más sobre cómo aplicar la metodología Lean a UX, puedes consultar este libro: *Lean UX: Cómo aplicar los principios Lean a la mejora de la experiencia de usuario* (UNIR Emprende). En él se abordan todas las perspectivas que intervienen en el proceso de diseño de gestión de clientes, stakeholders o proceso de trabajo entre equipo.

## OKR. El método de Google

OKR = *objectives and key results* (**objetivos y resultados clave**). Se trata de un método de trabajo creado por el exCEO de Intel, Andrew Grove, que Google incorporó con solo 1 año de vida, en 1999, cuando eran 40 trabajadores, y que hoy sigue usando (con más de 60 mil personas en todo el mundo). En la actualidad, lo usan también **LinkedIn y X y todas las compañías lo quieren incorporar**, puesto que tiene el potencial de **implementarlo tanto en pymes como en corporaciones multinacionales** con la misma eficiencia y transparencia.

Su objetivo es simplificar la forma de hacer frente a los objetivos estratégicos de una compañía o de un producto.

Impulsa **la alineación organizativa** a través de la consecución de objetivos comunes.

Gracias a este método se consigue uno de los **elementos básicos de la experiencia del equipo**: remar todos en la misma dirección.

### ¿Qué son y cómo funcionan los OKR's?

Este método es muy sencillo y se basa en dos pilares:

- **Objetivos** (*objectives*) ambiciosos e inspiradores.
- Los **resultados clave** (*key results*) son indicadores cuantificados y delimitados en el tiempo que permiten evaluar el cumplimiento de un objetivo determinado.

Vamos a conocer más cada **elemento** en un producto digital y cómo se complementan entre sí:

1

#### **Objectives**

Presentan una **dirección clara** de lo que el producto pretende conquistar. Cada objetivo puede ser formulado no solo para aclarar lo que se debe buscar, sino también para mantener al equipo comprometido y alineado hacia la *north star*.

Algunos **ejemplos** simples de objetivos podrían ser:

- Lanzar un MVP (mínimo producto viable).
  - Conseguir ganancias para finales de 2021.
  - Ampliar la gama de productos ofrecidos.
  - Mejorar la experiencia con nuestra aplicación.
- 

**Como ves, los objetivos son claros y específicos, no se cuantifican y suelen ser bastante genéricos para dejar claro cuál es el enfoque que se debe tener.**

2

### ***Key results***

Se encargará de dar forma al éxito del objetivo. Para ello, se trabaja en **especificar el objetivo y cuantificar un nivel de ambición para cada dimensión**. Conseguiremos así medir de forma objetiva y precisa la consecución del objetivo.

---

**Se trata de metas menores que ayudan directamente en la conquista de la meta principal.**

---

Vamos a usar los **mismos objetivos** citados antes, pero esta vez con la ayuda de los *key results*:

OBJETIVO #1	KR #1	KR #2	KR #3
Lanzar un MVP.			

OBJETIVO #1	KR #1	KR #2	KR #3
Recolectar feedback del primer 50% de los clientes.			

OBJETIVO #1	KR #1	KR #2	KR #3
Conseguir un 30% de compras del total visitas.			

OBJETIVO #1	KR #1	KR #2	KR #3
Crear 3 estudios de casos de uso según las visitas recibidas.			

---

**No debe multiplicarse el número de ítems que responde a cada objetivo. Con un rango de 2 a 5 resultados clave por objetivo es suficiente, pues permite enfocarse al máximo y no perderse en demasiados indicadores.**

Se deben diferenciar las dimensiones de las métricas cuantitativas.

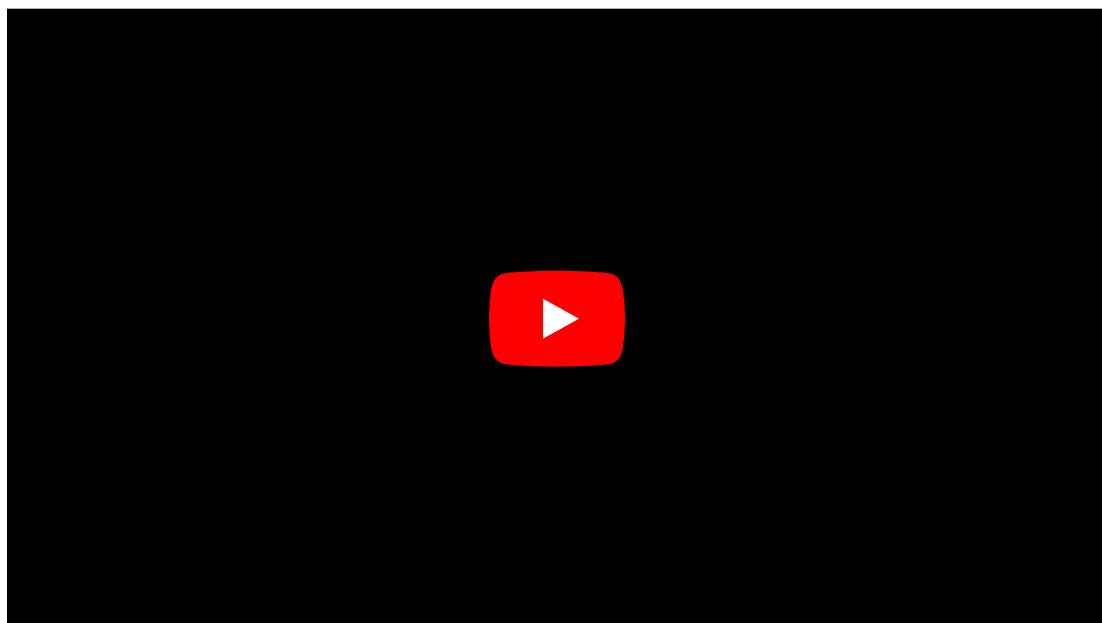
---

Las dimensiones son parámetros que podemos monitorizar, pero no son medibles.

Por ejemplo, 'desde qué dispositivos se conectan a nuestra plataforma' es un parámetro relevante ya que puede hacernos **marcar unos objetivos concretos** o dirigir nuestra estrategia. No podemos aportar un número, pero sí vislumbrar que nuestros usuarios se conectan desde tablet y desde smartphones, pero no desde desktop. Es un **valor cualitativo**. Así, si tienes tu producto desarrollado para desktop y **la versión responsive** (que se adapta a cualquier dispositivo) es mejorable, ya puedes ver hacia dónde dirigir tus próximas acciones.

Pero esto no es suficiente, es como si se quedara corto, ¿verdad? Ahí es donde entran las **métricas**. En este ejemplo sería: '¿cuántos usuarios acceden con tablet y cuántos con smartphone?'. Aquí aportamos un valor numérico, es una **métrica cuantitativa**.

En el siguiente vídeo puedes ver algunos ejemplos de métricas orientadas a un producto digital, de la mano del emprendedor e inversor Javier Megias, CEO y co-fundador de startupxplore.com.



---

**Recuerda: debemos poner metas ambiciosas. Si los OKRs se vuelven demasiado fáciles, será necesario apostar en ideas más grandes y que el producto consiga su máximo potencial.**

Conviene marcar una *north star*.

---

**La *north star* ha sido desde siempre la guía de los marineros. Es aquello que buscas cuando estás perdido.**

---

# Metodologías tradicionales vs. metodología Agile



Qualentum Lab

En este punto se va a realizar una comparación entre las metodologías Agile y las tradicionales.



Consulta [aquí](#) toda la tabla comparativa de metodologías Agile frente a las metodologías tradicionales.

Aspecto	Tradicional	Agile
Requisitos	Los requisitos no deben cambiar.	Los requisitos son cambiantes.
Requisitos (funcionalidades innecesarias)	Debido a la recolección inicial de requisitos es frecuente que se soliciten funcionalidades innecesarias.	El enfoque continuo no permite que se incluyan funciones innecesarias.

Aspecto	Tradicional	Agile
Tiempo	Existe un compromiso de tiempo de entrega de proyecto (no siempre se cumple).	Existe incertidumbre de la entrega de tiempo, pero como máximo 2 meses se entrega una parte (1 mes en Scrum).
Coste	Ídem.	Ídem.
Iteraciones	Pocas iteraciones.	Muchas iteraciones.
Riesgos	Los riesgos los asume el proveedor.	Depende del cliente al cambiar y compartir la responsabilidad de los riesgos.
Roles	Muchos roles.	Pocos roles.
Participación del equipo	Empodera al gerente de proyecto para el éxito de este, este decide si participa de este poder o no al equipo o no.	Empodera al equipo para trabajar de forma creativa e innovadora.
Arquitectura	Ejercicio que realiza al inicio.	Es un ejercicio constante durante el proyecto.

# Herramientas metodología Agile



Las dos herramientas que se van a explicar son tanto **Jira** como **Trello** para poder aplicar las metodologías Agile.

## Jira



Jira es una herramienta de software propietario que se utiliza para la gestión de proyectos, seguimiento de incidencias o errores.

Fue desarrollada por una empresa australiana cuyo nombre es Atlassian.

Jira Software **se lanzó en 2002 como una herramienta de seguimiento de incidencias y gestión de proyectos** para los equipos. Desde entonces, **más de 65.000 empresas** en todo el mundo han adoptado Jira por su flexibilidad para admitir cualquier tipo de proyecto y la posibilidad de trabajar con miles de aplicaciones e integraciones.

Una característica principal, la que interesa en este tema, es que se utiliza para equipos ágiles.

---

**En el caso de los equipos que usan metodologías ágiles, Jira Software proporciona tableros de Scrum y Kanban listos para usar.**

---

Los tableros son **centros de gestión de tareas**, donde estas se asignan a flujos de trabajo personalizables. Asimismo, los tableros ofrecen transparencia sobre el trabajo en equipo y visibilidad del estado de cada elemento de trabajo.

Las funciones de seguimiento del tiempo y los informes de rendimiento en tiempo real (diagrama de trabajo pendiente o de trabajo completado, informes de sprints, gráficos de velocidad) permiten a los equipos supervisar de cerca su productividad con el paso del tiempo. Con la vista de cronograma, puedes **planificar y hacer un seguimiento** de tu progreso en el panorama general.

Otras de las características que ofrece son las siguientes:

- Jira Software para equipos de gestión de proyectos.
- Jira Software para equipos de desarrollo de software.

- Jira Software para equipos de DevOps.
- Jira Software para equipos de gestión de productos.
- Jira Software para la gestión de tareas.
- Jira Software para supervisar errores.
- Jira Software para la gestión de requisitos y casos de prueba.



Puedes consultar la guía de Jira en este [enlace](#).

## Trello



---

Trello es una herramienta visual que permite a los equipos gestionar cualquier tipo de proyecto y flujo de trabajo o supervisar tareas. Así como añadir archivos, checklists o incluso automatizaciones.

Trello podría ser definido bajo estos **tres parámetros**:

- 1 Un conjunto de pizarras.
- 2 Cada pizarra dividida en columnas.
- 3 Las columnas contienen písit (tarjetas).

Estas son las **características principales**:

- Accesibilidad.
- Colaboración.
- Historial y seguimiento.
- Personalización.

Las **ventajas y desventajas** son las siguientes:

### Ventajas

- Simple y fácil de usar, pero con capacidad para proyectos pequeños y medianos.
- Se puede utilizar de varias formas, personal o profesional.

## Desventajas

- El plan gratuito puede quedarse corto.
- Menos opciones en algunos casos que competidores.



En este [enlace](#) puedes consultar todas las ventajas y desventajas de Trello.

# Conclusiones



---

Una vez entendidos los dos tipos de metodologías actuales a la hora de desarrollar un proyecto o producto, es necesario hacer referencia a que las grandes compañías están apostando por la metodología Agile debido a las **grandes ventajas que aporta**, sobre todo, en cuanto a la flexibilidad y la facilidad de adaptación a cambios pedidos por el cliente.

En este contexto es necesario entender bien las **posibilidades** que puede ofrecer la metodología Agile y tampoco olvidar que existen proyectos que podrían tener más sentido desarrollarlos como metodología tradicional, así como que hay compañías que aún no han realizado su total adaptación a las nuevas filosofías.

¡Enhорabuena! Fastbook superado



[Qualentum.com](http://Qualentum.com)