

Master en Dirección de Proyectos Informáticos



Trabajo Fin de Master

Estudio sobre las metodologías ágiles y metodologías tradicionales para gestión de proyectos de software

ESCUELA POLITECNICA
SUPERIOR

Autor: David Jordán Marcos

Tutor: José Carlos Ciria Cosculluela

UNIVERSIDAD DE ALCALÁ



Escuela Politécnica Superior

Master Universitario en Dirección de Proyectos Informáticos

Trabajo Fin de Master

**Estudio sobre las metodologías ágiles y
metodologías tradicionales para la gestión de
proyectos de software**

Autor: David Jordán Marcos

Septiembre 2020

Universidad de Alcalá
Escuela Politécnica Superior

Master Universitario en Dirección de Proyectos Informáticos

Trabajo Fin de Master

**Estudio sobre las metodologías ágiles y
metodologías tradicionales para la gestión de
proyectos de software**

Autor : **D. David Jordán Marcos**

Director Master : **Dr. D. Roberto Barchino Plata**

Tribunal evaluador :

Presidente del Tribunal : **Jose Amelio Medina**

Vocal 1º: **Javier Albert**

Vocal 2º: **José Carlos Ciria Cosculluela**

Calificación : _____

Alcalá de Henares a 11 de Septiembre de 2020

ÍNDICE GENERAL

1.	RESUMEN DEL PROYECTO.....	9
2.	ABSTRACT	10
3.	INTRODUCCIÓN	11
3.1.	Presentación	11
3.2.	Objetivos	12
4.	MARCO TEÓRICO.....	13
4.1.	Metodologías tradicionales	13
4.2.	Metodologías ágiles	14
4.2.1.	Manifiesto ágil.....	14
4.2.2.	Scrum	16
4.2.2.1.	Scrum Team.....	16
4.2.2.2.	Historias de usuario.....	17
4.2.2.3.	Sprint	18
4.2.2.4.	Reuniones	19
4.2.3.	Kanban	20
4.2.4.	Otras metodologías	21
4.2.4.1.	Lean Startup.....	21
4.2.4.2.	Xtremme Programming (XP).....	22
5.	DIFERENCIAS ENTRE METODOLOGÍA ÁGIL Y TRADICIONAL	23
6.	ELECCIÓN DE METODOLOGÍA	26
6.1.	Metodología tradicional.....	26
6.1.1.	Ventajas.....	26
6.1.2.	Inconvenientes	26
6.2.	Frameworks ágiles.....	27
6.2.1.	Ventajas.....	27
6.2.2.	Inconvenientes	27
6.3.	¿Qué metodología es mejor?.....	28
6.3.1.	Cuándo elegir una metodología tradicional.....	28
6.3.2.	Cuándo elegir un framework ágil	29
6.3.3.	¿Existe un punto intermedio?	29
7.	CAMBIO HACIA METODOLOGÍA ÁGIL.....	30
7.1.	La llamada a la acción	30
7.2.	¿Está la organización preparada?	31
7.3.	La importancia de los mentores.....	31

7.4.	Formar un equipo que lidere la transformación.....	32
7.5.	Entorno y hábitos.....	33
7.6.	Demostración de la agilidad: Pruebas.....	34
7.7.	Evolucionar.....	34
7.8.	Cuidado con la falsa agilidad.....	35
7.9.	SHU-HA-RI.....	35
8.	OBSTÁCULOS EN LA TRANSFORMACIÓN ÁGIL.....	36
8.1.	No hay necesidad de cambiar.....	36
8.2.	Efecto novedad.....	37
8.3.	Este nuevo método de trabajo no funcionará.....	37
8.4.	Falta de motivación ante el cambio.....	38
8.5.	Resistencia al cambio.....	39
8.6.	Miedo al cambio.....	40
8.7.	Las reuniones son una pérdida de tiempo.....	41
8.7.1.	Daily.....	41
8.7.2.	Retrospectiva.....	42
8.7.2.1.	Técnica de la Estrella de Mar.....	43
8.7.2.2.	Técnica de las 4L.....	44
8.8.	Si algo falla, aparece la negatividad.....	45
8.9.	Realizar un cambio para no cambiar nada.....	47
8.10.	Mis empleados no quieren cambiar.....	48
8.11.	Contratar a gente experta, para nada.....	49
8.12.	Ya no es necesario documentar: Error.....	49
8.13.	Agile significa que no necesitamos procesos.....	50
8.14.	Yo soy un directivo, eso es de técnicos.....	51
8.15.	Cuéntame lo que quiero escuchar, no la verdad.....	52
9.	ESTADÍSTICAS DE TRANSFORMACIÓN ÁGIL.....	53
9.1.	Tasa de éxito según metodología.....	53
9.2.	Tasa de éxito según metodología y tamaño.....	54
9.3.	¿Por qué cambiar a Agile?.....	55
9.3.1.	Técnicas Agile más utilizadas.....	55
9.3.2.	Metodologías Agile más utilizadas.....	56
10.	CONCLUSIONES.....	57
11.	LÍNEAS FUTURAS.....	58
12.	BIBLIOGRAFÍA.....	59

1. RESUMEN DEL PROYECTO

En este proyecto se pretende poner en práctica los conocimientos sobre metodologías ágiles y gestión de proyectos aprendidos a lo largo de la realización del Master.

Se realizará una introducción a las metodologías ágiles, cuáles son sus principales características, qué metodologías existen y qué diferencias principales tienen con respecto a los métodos de trabajo tradicionales.

A continuación, se llevará a cabo un estudio para decidir en qué tipo de proyectos es recomendable realizar un cambio en la metodología de trabajo y en cuáles es conveniente mantener una estructura tradicional.

Es importante entender que no es tarea fácil para una organización llevar a cabo un cambio en su filosofía de trabajo. Por tanto, se indicarán algunas pautas a la hora de realizar el cambio ágil en una organización.

Para finalizar, se van a estudiar algunos de los problemas más comunes cuando una organización inicia el viaje hacia la transformación ágil y cómo podrían solventarse.

2. ABSTRACT

In this project, the knowledge about agile methodologies and project management learned during the completion of the Master will be put into practice.

An introduction to agile methodologies, their main characteristics, existing methodologies and the main differences with respect to traditional work methods will be made.

Next, a study will be carried out to decide in which type of projects it is advisable to make a change in the work methodology and in which it is advisable to maintain a traditional structure.

It is important to understand that it is not an easy task for an organization to carry out a change in its work philosophy. Therefore, some guidelines will be indicated when making agile change in an organization.

Finally, some of the most common problems when an organization begins the journey towards agile transformation and how they could be solved will be studied.

3. INTRODUCCIÓN

3.1. Presentación

El desarrollo de software es sin duda uno de los sectores en alza hoy en día, sin embargo, en algunas ocasiones, los métodos mediante el cual se gestionan esos proyectos están basados en unas metodologías de trabajo tradicionales que podrían (o no) actualizarse.

Cada vez con más frecuencia se escuchan los términos “Agile”, “Scrum” o “Kanban”, entre otros. En ocasiones sin comprender a qué hacen referencia, si estos nuevos métodos de trabajo pueden utilizarse en determinado entorno o los cambios y sacrificios que puede suponer realizar un cambio en el modo de trabajar de toda una organización.

Actualmente, muchas empresas están tomando la decisión de actualizar sus métodos de trabajo a estas filosofías de desarrollo de software ágil, pero no siempre es tan sencillo.

Primero porque no todos los proyectos son aptos, según sus características, para utilizar estas metodologías de trabajo. Y segundo, es importante comprender lo que implica para una empresa modificar el modo de trabajar tanto de los empleados como de los directivos. Por norma general, los cambios no son bien recibidos, y es común que haya cierta resistencia al cambio.

En este proyecto, se realizará un estudio de las principales diferencias entre las metodologías de trabajo ágiles y tradicionales. También en qué tipo de proyectos es interesante utilizar este enfoque.

Aunque el cambio no es igual para todas las empresas, se van a indicar algunos puntos a tener en cuenta para lograr el éxito al adoptar una nueva metodología. También se van a exponer algunos de los problemas más comunes que sufre una organización cuando inicia el cambio ágil.

3.2. Objetivos

El principal objetivo de este proyecto es comprender las dificultades que supone para una organización afrontar un cambio en su metodología de trabajo, desde los empleados hasta la dirección.

Para ello, se deberán cumplir los siguientes objetivos:

- Comprender las principales diferencias entre las metodologías de trabajo tradicionales y ágiles
- Entender en qué tipo de proyectos o situaciones es conveniente el uso de cada uno de esos métodos de gestión de proyectos
- Conocer los principales problemas que pueden suceder durante una transformación ágil y cómo solucionarlos.

4. MARCO TEÓRICO

4.1. Metodologías tradicionales

Históricamente, han sido las más utilizadas a la hora de realizar desarrollo software. Sin embargo, poco a poco están siendo desbancadas por un estilo de desarrollo ágil debido a los problemas que presenta este tipo de método de trabajo.

Se denominan desarrollo “waterfall” o “cascada” debido a que el inicio de una etapa está condicionado por el final de la anterior. Las etapas de un desarrollo en cascada son las indicadas e la figura 1.



Figura 1. Fases desarrollo cascada

Centran su esfuerzo en una planificación completa de todo el proceso y, una vez que todo está detallado, comienza el desarrollo software. Son de carácter predictivo, tratan de adivinar el proyecto final, documentando todos los requisitos y teniendo en cuenta todos los problemas que puedan surgir.

En este modo de trabajar, se realiza un férreo control de proceso, definiendo previamente los roles de cada miembro del equipo, así como las actividades que realizará cada uno de ellos.

Otro punto importante en estas metodologías es la documentación detallada. Se cuenta con una documentación muy extensa sobre el funcionamiento del software, y esto es precisamente un punto en contra, debido a que puede no ser de ayuda cuando alguien desee consultarla y es muy fácil que quede desactualizada.

4.2. Metodologías ágiles

Más que una metodología en sí, cuando se habla de metodologías ágiles, es más correcto hablar de **frameworks ágiles**.

Esta denominación se debe a que, en términos de agilidad, se parte de la idea de que es algo flexible, que debe ser capaz de adaptarse a cualquier entorno de trabajo o cambio, por lo que no hay una sola manera de llevar a la práctica una metodología ágil.

Es común pensar que algunos de los frameworks actuales más conocidos como Scrum o XP, surgieron a raíz de la firma en 2001 del Manifiesto ágil, pero no es correcto.

Algunas de estas nuevas metodologías o frameworks ágiles, son métodos de trabajo que ya existían anteriormente (Scrum, eXtreme Programming, FDD, DSDM, ...), pero es en 2001 cuando, mediante la firma del Manifiesto ágil, se engloban todas bajo el mismo nombre. En la figura 2 se puede observar en una línea temporal qué métodos de trabajo ágiles existían antes de la firma del manifiesto ágil.

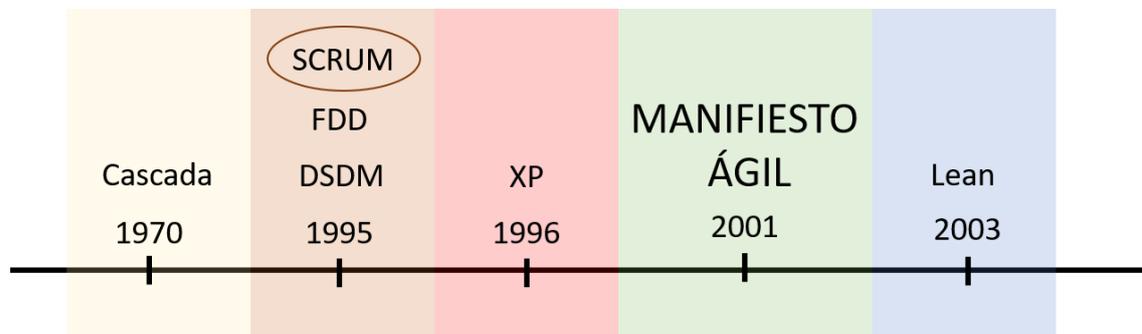


Figura 2. Línea temporal frameworks ágiles

La más popular es Scrum. De todos los frameworks ágiles, es el que está más tutorizado. Cuando en una organización se comienza a introducir la agilidad, es conveniente empezar por ella.

En los siguientes puntos, se va a desarrollar brevemente algunas de las principales características de los frameworks ágiles más utilizados.

4.2.1. Manifiesto ágil

La agilidad no puede entenderse sin conocer los valores clave y los 12 principios de este manifiesto. En 2001, un grupo de críticos de los modelos de desarrollo de software basado en procesos se reunió para exponer nuevas técnicas de desarrollo de software. Es en esta reunión donde se acuña por primera vez el término “metodología ágil” para hacer referencia a estos nuevos métodos de desarrollo surgidos como alternativa a los clásicos. Los cuatro valores del manifiesto ágil son los que se observan en la figura 3.



Figura 3. Valores ágiles. Imagen propia, extraída de (agilemanifesto.org)

- **Individuos e interacciones** por encima de procesos y herramientas: El equipo de trabajo debe trabajar unido para alcanzar los objetivos.
- **Software que funciona** por encima de excesiva documentación: Tradicionalmente, se genera una gran cantidad de documentación. Es más importante gastar tiempo y energías en un software que funcione y que poder entregar al cliente.
- **Colaboración con el cliente** por encima de las negociaciones: La relación entre el cliente y el equipo de desarrollo debe ser fluida, es importante generar un ambiente de confianza.
- **Respuesta al cambio** por encima del seguimiento de un plan establecido: Es necesario que el equipo esté preparado para los cambios que puedan surgir durante el proceso de desarrollo de software.

Por otro lado, en el manifiesto ágil, se redactan también los 12 principios que surgen de los valores anteriores:

1. Satisfacer al cliente mediante tempranas y continuas entregas de software
2. Dar la bienvenida a los cambios. Se capturan cambios para que el cliente tenga una ventaja competitiva
3. Entrega frecuente de software que funcione con el menor intervalo de tiempo posible.
4. Negocio y desarrollo deben trabajar unidos a lo largo del proyecto
5. Individuos motivados. Proporcionar entorno y apoyo necesario y confiar en ellos.
6. El diálogo cara a cara como método para comunicar información dentro de un equipo de desarrollo
7. El software que funciona es la medida principal de progreso
8. Los procesos ágiles promueven un desarrollo sostenible. Promotores, desarrolladores y usuarios deben ser capaces de mantener una paz constante
9. La atención continua a la calidad técnica y al buen diseño mejora la agilidad.
10. La simplicidad es esencial
11. Las mejores arquitecturas, requisitos y diseños surgen de los equipos auto organizados
12. Regularmente, el equipo reflexiona sobre cómo llegar a ser más efectivo.

4.2.2. Scrum

Es probablemente el framework ágil más conocido y utilizado. Es un método de trabajo especialmente útil cuando se utiliza para gestionar la imprevisibilidad de los proyectos y controlar los riesgos.

Aunque nació centrado en el desarrollo de software, se ha demostrado que el uso de Scrum puede trasladarse casi a cualquier contexto. La idea de Scrum es realizar entregas parciales del producto final, ejecutando el proyecto en ciclos cortos y de una determinada duración. Cada iteración, puede ser entregada para su revisión.

Es un marco de trabajo que proporciona unas pautas, pero no obliga a seguir al pie de la letra todas ellas. La ventaja que tiene Scrum es que permite adoptar únicamente aquellas partes que resulten más interesantes para el proyecto.

4.2.2.1. Scrum Team

En Scrum, todo el equipo está compuesto por una serie de roles. Es importante entender la función de cada uno de ellos para poder trabajar mejor, cometer menos errores y alcanzar los objetivos.

- **Product Owner:** Es el contacto entre el equipo de desarrollo y el cliente. Recibirá las propuestas del cliente y debe tener la capacidad de realizar una priorización de tareas, incluso negarse a realizar aquellas que por algún motivo no puedan ser llevadas a cabo.

Es deseable que entienda y conozca Scrum, pero no es absolutamente necesario, ya que su función es la de realizar de puente de información entre equipos y de priorizar y definir las tareas.

- **Scrum Master:** Es el experto en Scrum y su función es la de asegurar que el equipo de desarrollo trabaje según las guías de Scrum. No es un jefe o un gestor de tareas o trabajo, sino una persona que indica las pautas de trabajo, dirige las reuniones de seguimiento y sirve de facilitador en caso de que el equipo de desarrollo tenga algún bloqueo.
- **Equipo de desarrollo:** Es el equipo encargado de crear el producto. Está formado por programadores, testers y cualquier persona encargada de desarrollar el producto. Debe ser un equipo auto-organizado, multidisciplinar, comunicativo, colaborativo, etc.
- **Stakeholders:** Aunque no forman parte del equipo de Scrum como tal, son importantes debido a que forman parte de la toma de decisiones. Por ejemplo, el equipo de contabilidad, marketing, ventas, etc.

4.2.2.2. Historias de usuario

El backlog completo contiene todos los requisitos en forma de historias de usuario. En metodologías ágiles son muy utilizadas, especialmente en Scrum. Son descripciones de las funcionalidades que tendrá el software y se elaboran entre el cliente y el Product Owner. Se componen de las 3C

- **Card:** Es una tarjeta con la descripción de la funcionalidad
- **Conversation:** Es la conversación entre cliente, desarrollo y Product Owner sobre la funcionalidad para confirmar que se desarrollará lo deseado
- **Confirmation:** Serán los criterios de aceptación. Una checklist que validará la historia de usuario antes de ser entregada.

En la siguiente figura se muestra un ejemplo de historia de usuario, con los elementos comunes que suelen introducirse:

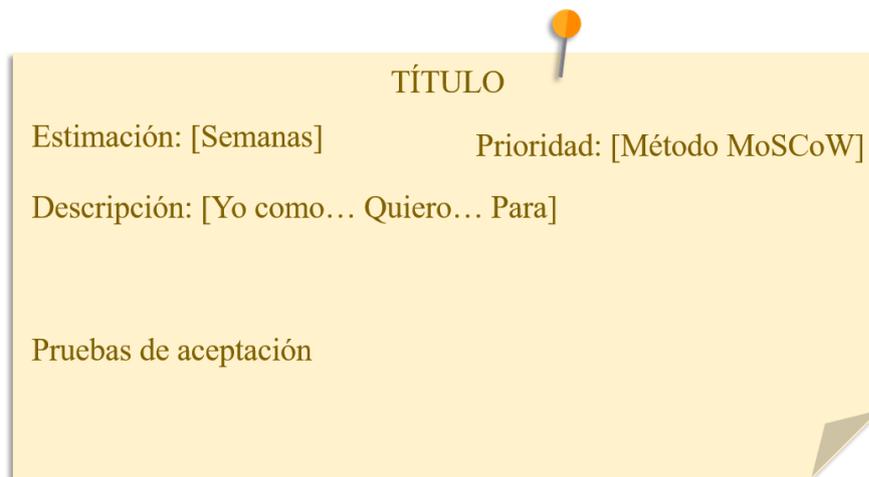


Figura 4. Representación de Historia de Usuario.

La estimación suele representarse en semanas y la prioridad puede determinarse según el método MoSCoW:

- **Must:** Esta historia es importante, por lo que debe completarse para dar por finalizado el proyecto
- **Should:** Aunque el éxito del proyecto no depende de esta funcionalidad, se debe intentar cubrir

- **Could:** Son requisitos o historias de usuario que, si el tiempo lo permite y no afecta a otras funcionalidades, se deben intentar completar.
- **Would:** Se completará únicamente si al finalizar el desarrollo hay tiempo de sobra.

La descripción de la historia de usuario se realiza mediante el siguiente método:

- **Yo como:** Siendo la persona que utilizará la funcionalidad
- **Quiero:** En este punto se especifica la funcionalidad
- **Para:** La finalidad de realizar la funcionalidad. Debe ser específico y aportar valor real

4.2.2.3. Sprint

Se denomina así a los ciclos o iteraciones dentro del proyecto. La duración de cada sprint es de unas 2 a 4 semanas como máximo.

Los proyectos Scrum cuentan con un **Sprint Backlog** (definido por el Product Owner) en el cual estarán planteados los requisitos y las funcionalidades que se desarrollarán. Cada funcionalidad está formada por historias de usuario, cada una con sus correspondientes tareas.

Debido a que el número de tareas o funcionalidades definidas en el Backlog es muy elevado, se divide en sprints. El sprint y el alcance del mismo lo define el Scrum Master.

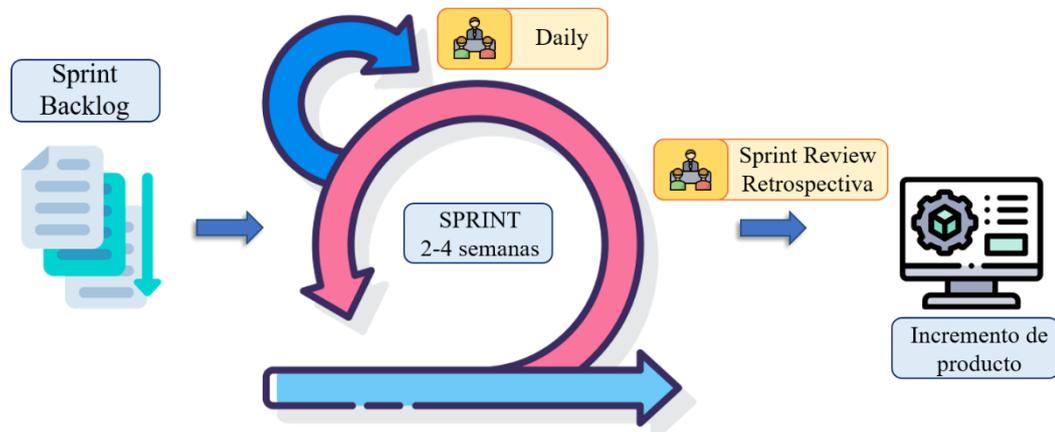


Figura 5. Sprint

Al finalizar cada sprint, se logrará un incremento funcional del producto que podrá ser entregado al cliente para su revisión.

4.2.2.4. Reuniones

En proyectos Scrum, la comunicación y la constante mejora es muy importante, es por eso que existen varias reuniones.

- **Reunión de planificación de Sprint:** En esta reunión se seleccionan los ítems o las historias de usuario que se desarrollarán en el próximo sprint, formando así el Sprint Backlog. En esta reunión participa de manera colaborativa todo el equipo de Scrum.
- **Daily:** Como su propio nombre indica es una reunión que se debe celebrar diariamente y en la que participarán todos los integrantes del equipo Scrum, así como el Scrum Master, que será la persona encargada de guiar la reunión.

No debe durar más de 15 minutos y debe ser una reunión ágil, en la que cada integrante cuente qué hizo el día anterior, que hará durante el día actual, y si hay algún impedimento para realizar el trabajo. El Scrum Master se encargará de solucionar cualquier problema o bloqueo que tenga el equipo.

- **Sprint Review:** Esta reunión se celebra al finalizar el sprint y antes de comenzar el siguiente. Asistirán el Product Owner, el Scrum Master y si es necesario el equipo de desarrollo. Es una reunión formal en la que se realizará una revisión del sprint, indicando si se ha llegado o no a los objetivos marcados y se tomarán decisiones en base a los resultados.

Es común revisar un diagrama llamado **Burndown Chart**, en el cual se revisa el trabajo restante para completar el producto, teniendo de ese modo una visión del proyecto y de la velocidad del equipo.

- **Retrospectiva:** Es una reunión en la que asisten el Scrum Master y los miembros del equipo de desarrollo. La idea es que sea una reunión ágil e incluso divertida, en la que participe todo el equipo y en la que cada miembro del equipo puede proponer mejoras o puntos a mantener. Es una reunión importante y que debe realizarse siempre porque puede otorgar oportunidades de mejora.

4.2.3. Kanban

Es una metodología que está cada vez más de moda y viene de la metodología Lean. La idea de Lean es realizar el trabajo por partes, creando pequeños avances que aporten valor, es decir, partir de una base e ir mejorando y añadiendo funcionalidades poco a poco.

Para crear cada uno de esos pequeños avances, uno de los métodos que se utiliza es el Kanban. El objetivo principal que tiene esta metodología es la visualización rápida de todo el trabajo de un simple vistazo (las tareas sin empezar, las tareas en curso y las tareas finalizadas) así como las personas encargadas de cada una de esas tareas.

De este modo, se obtiene una visión general del estado del proyecto, y se logra no sobrecargar de trabajo a ningún integrante del equipo. En la siguiente imagen se muestra un tablero Kanban básico.

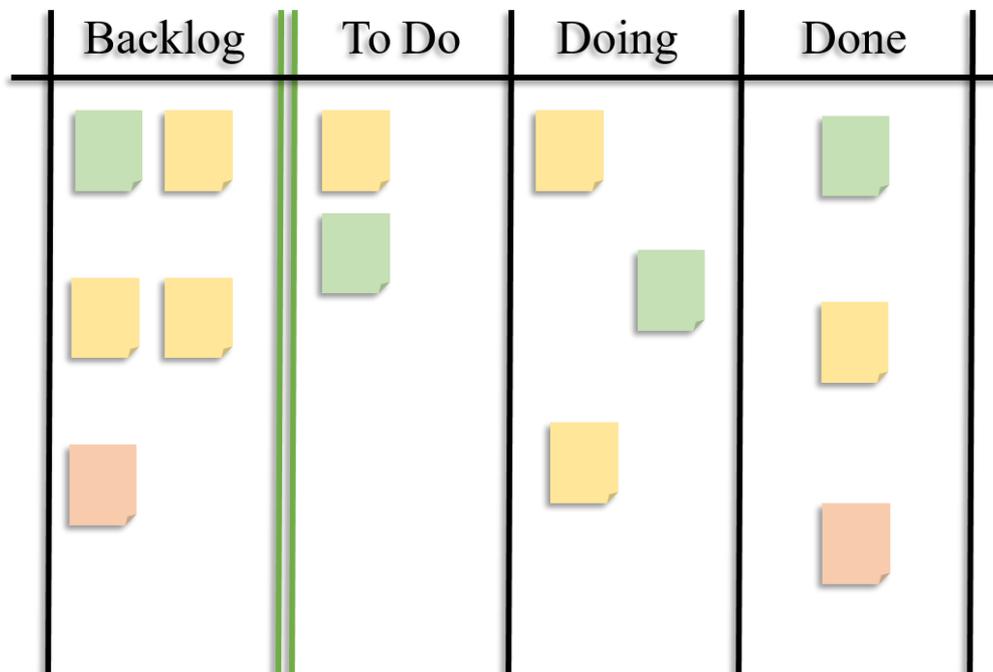


Figura 6. Imagen propia. Tablero KANBAN básico

En el tablero se observan cuatro columnas. En la columna **Backlog** se incluyen todas las tareas en espera ordenadas por prioridad. Estas tareas pasarán a **To Do** en el momento de comenzar con su ejecución, asignándose a un miembro del equipo.

En la columna **Doing** se muestran las tareas que están siendo realizadas en ese momento, indicando la persona que la está llevando a cabo. Una vez que estas tareas se finalizan, se mueven a la columna **Done**.

4.2.4. Otras metodologías

Scrum es la metodología o framework más utilizada actualmente y Kanban se está convirtiendo en una manera cada vez más común de obtener una visión general del proyecto. Sin embargo, cabe destacar que existen otras metodologías ágiles de trabajo. Algunas de ellas son:

4.2.4.1. Lean Startup

Aunque el nombre y el origen de esta filosofía de trabajo se relaciona con empresas de reciente creación, puede aplicarse a cualquier empresa de cualquier sector. Se consigue aumentar el crecimiento de la empresa acelerando la creación de nuevos productos o servicios para los clientes.

Esta metodología viene de la empresa japonesa Toyota, ya que la aplicaba bajo el nombre de “Lean Manufacturing” en sus líneas de montaje. Si cualquier recurso gastado (ya sea tiempo, componente, nueva maquinaria) si no aumenta el valor del producto ofrecido al cliente, se elimina.

Eric Ries, con el libro “The Lean Startup” adapta esta filosofía, cambiando su nombre, estableciendo la cultura de esta filosofía y aplicándola a las nuevas empresas.

Cualquier cosa que se realice debe ser pequeña y proporcionar un valor directo a los clientes, desperdiciando el menor número de recursos posibles. Esto a través de tres pilares:

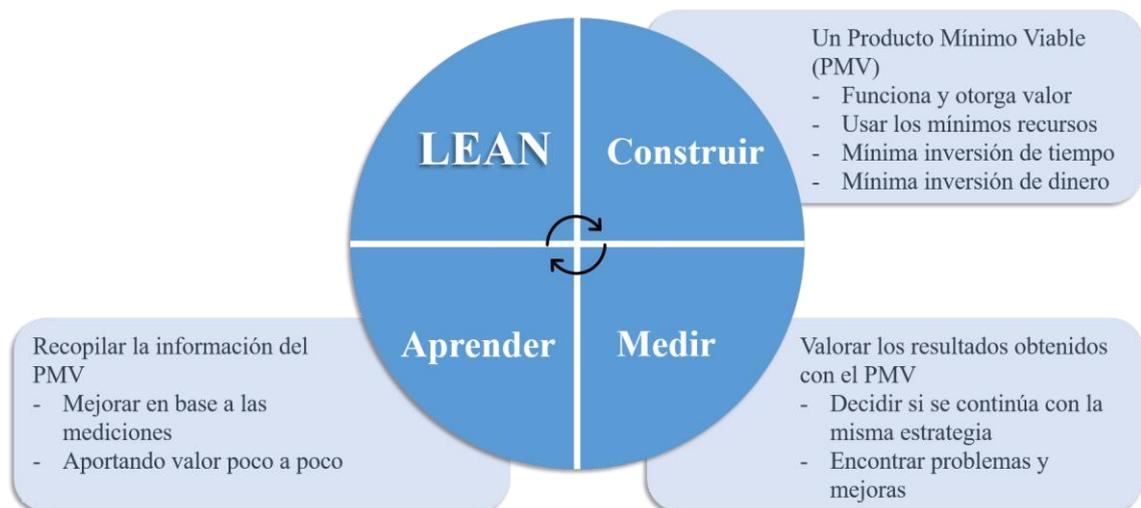


Figura 7. Pilares Lean

4.2.4.2. Xtremme Programming (XP)

Esta metodología se utiliza en proyectos con un corto plazo de entrega. La idea es realizar una programación rápida contando en el equipo con el usuario final.



Figura 8. Principales características Xtremme Programming (XP)

Se llega a esta meta mediante un desarrollo iterativo e incremental, realizando pequeños avances. La idea es automatizar el mayor número de pruebas, tanto unitarias como de regresión.

La programación por parejas es importante, debido a que se obtienen varios puntos de vista del mismo código, sacrificando productividad para lograr un código más rápido y efectivo.

Se intenta buscar el error 0, corrigiendo todos los fallos antes de comenzar el desarrollo de una nueva funcionalidad. Se desarrolla un software simple y funcional, conocido por todo el equipo y de responsabilidad común, evitando los responsables de ciertas partes de código.

5. DIFERENCIAS ENTRE METODOLOGÍA ÁGIL Y TRADICIONAL

Una vez se han expuesto las principales características de los métodos de trabajo tradicionales como los ágiles y se cuenta con una base teórica suficiente para comprenderlos, se puede realizar una comparación entre ambos.

Una de las primeras diferencias se encuentra en el propio nombre. Mientras que una metodología tradicional se toma como algo fijo, no variable, las metodologías ágiles se denominan “frameworks” con la finalidad de remarcar la idea de que en la agilidad hay muchas maneras diferentes de llevar a cabo una determinada “metodología” en función de las necesidades del proyecto.

En la figura 9 se muestra una gráfica en la cual se observa qué tipo de metodología es en función de su orientación. Si es una metodología orientada a proceso y predictiva, es decir, trata de adivinar el futuro con unos requisitos y un diseño cerrado, se trataría de una metodología tradicional. En cambio, si está orientada a personas y es capaz de adaptarse a los cambios, se trataría de un framework ágil.

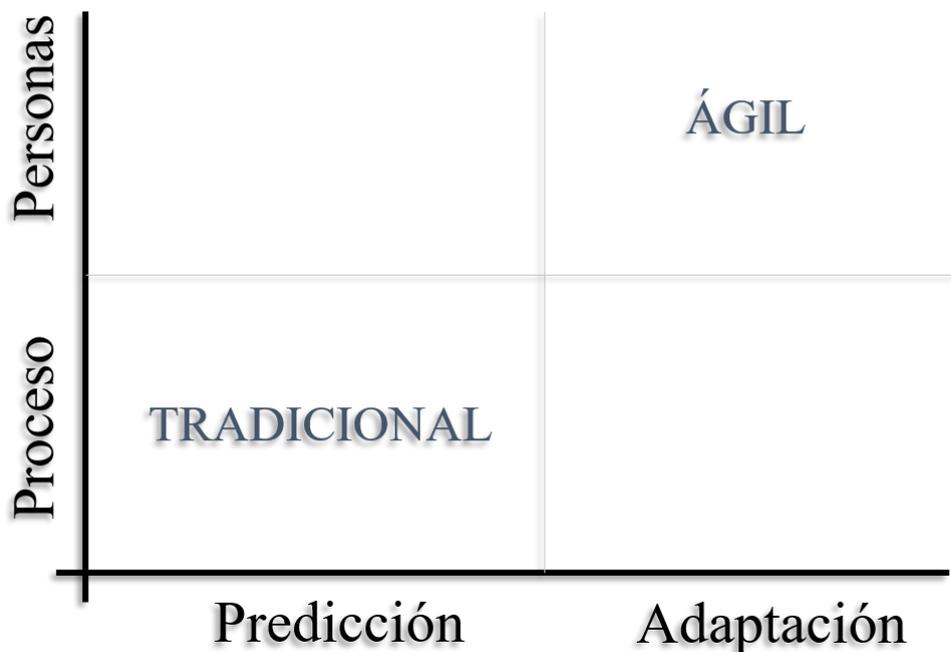


Figura 9. Adaptación del esquema metodológico según orientación de Javier Garzás

Modelo de desarrollo: Los modelos tradicionales siguen un enfoque lineal. Una vez que comienza el desarrollo, no se entrega nada al cliente hasta que éste ha finalizado. Esto conlleva muchos inconvenientes, debido a que, si al cliente no le satisface o desea cambios, no podrá llevarlos a cabo hasta el final del desarrollo.

Por otro lado, en proyectos ágiles, el enfoque es iterativo, con cada pequeño desarrollo se entrega al cliente un producto que puede ver, probar y sobre el que puede proponer cambios.

Escala: Si bien tanto las metodologías tradicionales como ágiles pueden adaptarse a cualquier tipo de proyecto, los frameworks ágiles suelen utilizarse en proyectos de escala media o pequeña y las metodologías tradicionales en grandes proyectos.

Requisitos: En un modelo tradicional, los requisitos se definen antes de comenzar el desarrollo de software. Sin embargo, en un modelo ágil, los requisitos son dinámicos y pueden modificarse antes de cada sprint, lo que aumenta la flexibilidad ante cualquier cambio.

Equipos de trabajo: En proyectos ágiles, los equipos de trabajo han de ser auto-organizados, pequeños, coordinados y sincronizados. Son equipos multidisciplinares. En cambio, en proyectos tradicionales, los equipos se coordinan en el momento en el cual es necesario realizar una integración. Suelen ser equipos especializados

Implicación y participación del cliente: En proyectos con enfoque ágil la implicación del cliente es muy alta. Participan desde el principio del proyecto y durante su desarrollo, proponiendo cambios cuando es necesario. En un entorno tradicional, la participación del cliente es mucho más baja. Se implica al inicio, pero no una vez que comienza el desarrollo.

Planificación: La planificación en modelos tradicionales es exhaustiva, se trata de anticipar todos los sucesos. Sin embargo, en modelos ágiles, la planificación se hace por sprints. Si bien al inicio del proyecto se realiza una planificación a alto nivel, los sprints son realmente lo que marca los tiempos y paquetes de desarrollo del proyecto.

Tareas: En entornos ágiles, el facilitador (o Scrum master) es la persona que realiza las estimaciones y lleva a cabo el reparto de tareas. En cambio, en modelos tradicionales esta figura no existe, por lo que el gestor del proyecto es el encargado de realizar las estimaciones, que deben ser aprobadas por el cliente.

Documentación: En los modelos tradicionales existe la convicción de que es necesario documentar exhaustivamente todos los procesos y el software desarrollado. Normalmente se genera una gran cantidad de documentación que luego no será utilizada.

En los modelos ágiles, en contra de lo que comúnmente pueda pensarse, si se genera documentación, pero únicamente la necesaria para poder comprender el código y que pueda ser de utilidad real.

Procesos: En los proyectos tradicionales suele existir un gran número de procesos y una gestión férrea de ellos. En proyectos ágiles deben mantenerse únicamente aquellos que aporten valor, realizando una mejora continua de ellos. La eficiencia de los procesos es clave en proyectos ágiles.

Presupuesto: En proyectos ágiles es conveniente que el presupuesto no sea cerrado, dado que es muy normal que se produzcan cambios. Los proyectos tradicionales funcionan con un presupuesto cerrado o muy poco flexible.

Jerarquía: En modelos tradicionales, la jerarquía de mando está muy definida (jefe de proyecto, analistas, diseñadores, programadores, testers, etc.) Y cada empleado rinde cuentas de manera individual a su superior.

En proyectos ágiles, existe el Product Owner, Scrum master o facilitador y el equipo de desarrollo. Independientemente de su rol interno, el equipo de desarrollo funcionará como uno solo, auto gestionándose y rindiendo cuentas como un equipo y no de manera individual.

En la siguiente tabla se muestra un resumen de todos los aspectos que diferencian a un proyecto que utiliza metodología tradicional con respecto a un framework ágil.

	Ágil	Tradicional
Escala	Pequeños y medianos	Grandes
Orientación	Personas	Proceso
Preferencias	Adaptación	Predicción
Modelo de desarrollo	Iterativo, entrega evolutiva	Lineal
Requisitos	Dinámicos	Definidos al inicio
Equipo de trabajo	Auto-organizado, multidisciplinar	Especializados
Implicación de cliente	Alta	Baja
Planificación	Sprints	Exhaustiva
Tareas	Facilitador o Scrum Master	Gestor del proyecto
Documentación	Sólo la necesaria	Exhaustiva
Procesos	Aportan valor. Mejora continua	Control férreo.
Presupuesto	Abierto a cambios	Cerrado
Jerarquía	Equipo	Individual

Tabla 1. Resumen diferencias entre metodología ágil y tradicional

6. ELECCIÓN DE METODOLOGÍA

Al comenzar un nuevo proyecto, una de las primeras decisiones que se deben tomar es la elección de la metodología más adecuada en función de las características del proyecto, de las personas, de la dirección, del cliente, etc.

Por ello, primero es importante conocer las principales ventajas y desventajas que tienen las metodologías de trabajo tradicionales y las ágiles.

6.1. Metodología tradicional

6.1.1. Ventajas

- Ideal en proyectos maduros que no requieren de muchos cambios.
- Planificación sencilla debido a que ambas partes llegan a un acuerdo cerrado en etapas tempranas.
- Necesitan poco presupuesto y pocas herramientas para ser llevados a cabo.
- Es un modelo muy conocido y que ha sido utilizado (y se sigue utilizando) mucho.
- Sencillo de implementar y de comprender. La metodología que utiliza es muy efectiva. Se basa en definir antes que diseñar y diseñar antes de desarrollar software.
- La presencia del cliente no es necesaria salvo en revisiones o procesos de aprobación.

6.1.2. Inconvenientes

- No sirve en proyectos en los que se requiera realizar cambios sobre la marcha. En el trabajo diario, siempre surgen inconvenientes, y es complicado gestionar un proyecto mediante métodos tradicionales si no se sigue una secuencia lineal.
- El cliente no tiene una visión de cómo está el producto, debido a que es necesario mucho tiempo para poder ver resultados.
- No puede comenzarse una etapa hasta que la anterior haya finalizado.
- Si se detectan errores, es necesario un rediseño, lo que produce un aumento de los costes del proyecto.

6.2. Frameworks ágiles

6.2.1. Ventajas

- Realización de entregas periódicas de software que funciona y puede ser probado al cliente. Normalmente al finalizar cada SPRINT. Esto conlleva una mayor satisfacción para el cliente.
- Gran flexibilidad en el caso de ser necesario aplicar cambios a los requisitos o al producto. Tras cada SPRINT, el cliente recibe un paquete del producto final que puede ser testado, de modo que, si el cliente requiere cambios, estos pueden implementarse en futuros sprints.
- Velocidad con la que es posible implementar los cambios requeridos por el cliente.
- Se eliminan tareas innecesarias que no aportan valor al producto final, es decir, se elimina el trabajo innecesario.
- Se disminuye (no elimina) la excesiva documentación
- La motivación del equipo con respecto al desarrollo del proyecto aumenta, debido a que se les hace partícipes de este.
- Mejor calidad final del producto debido a la continua interacción entre el cliente y el equipo de desarrollo.
- A largo plazo, implica una disminución del tiempo de desarrollo y por tanto una disminución de los costes.
- Mejora continua de los procesos de trabajo.

6.2.2. Inconvenientes

- Es fácil caer en el error que un desarrollo ágil no necesita documentación.
- La falta de documentación puede hacer caer en malentendidos a cliente y desarrollo. Algo escrito “no se puede borrar” sin embargo, algo dicho de palabra puede crear problemas.
- En ocasiones, si no se cuenta con la documentación adecuada, puede producirse una disminución de la reusabilidad del código.
- Deuda técnica: Estos métodos de trabajo se centran en las personas y no en la documentación, por tanto, si un proyecto fracasa, la comprensión del sistema queda en su mayor parte en manos del equipo de desarrollo.
- Miedo al fracaso: si un proyecto ágil fracasa, es común volver a los métodos de trabajo tradicionales.
- Suele existir una fuerte dependencia de los responsables. La persona encargada de liderar el proyecto centraliza las decisiones.

6.3. ¿Qué metodología es mejor?

Como se ha estudiado en los apartados anteriores, no existe un método de trabajo perfecto debido a que cada uno tiene sus ventajas e inconvenientes, por lo que hablar de qué método es mejor no es tan sencillo. La decisión de qué camino seguir, dependerá de varios factores y la elección final será del gerente del proyecto, previo consenso con la dirección y el cliente.

6.3.1. Cuándo elegir una metodología tradicional

Los modelos de trabajo tradicionales o en cascada son idóneos en proyectos que se encuentran muy bien definidos y en los cuales se conoce que existen pocos riesgos y se prevén pocos cambios, es decir, el alcance del proyecto está bien definido.

- Un enfoque tradicional es una buena idea cuando los requisitos del proyecto están muy claros, es decir, bien definidos desde el inicio del proyecto.
- La tecnología utilizada debe ser una tecnología conocida. Este tipo de métodos de trabajo no es apropiado si se intentan introducir nuevas tecnologías o herramientas.
- Es muy común el uso de estas metodologías en organizaciones muy grandes que cuentan con muchos recursos y mucho personal.
- Si el proyecto no tiene incertidumbre, es decir, no cuenta con una serie de riesgos potenciales que puedan hacer que el proyecto necesite una reestructuración.
- En caso de que el cliente no desee formar parte del proceso total, sino que desee únicamente ver el resultado final.

En conclusión, es apropiado utilizar un método de trabajo en cascada cuando las etapas necesitan de la finalización de la etapa anterior para poder comenzar. Por ejemplo, extrapolando estas ideas fuera del desarrollo de software, para poder poner el tejado de una casa, primero es necesario terminar las paredes, y antes que el tejado, los cimientos.

El inconveniente que puede ocurrir es que surjan retrasos. Esto se debe a que, si el equipo encargado de los cimientos no finaliza su tarea, no se puede dar comienzo a la siguiente.

6.3.2. Cuándo elegir un framework ágil

Al contrario que los métodos de trabajo tradicionales, son recomendables si se cumplen las siguientes características:

- Los requisitos del proyecto no están claros. Hay muchas posibilidades de que se produzcan cambios o de que se añadan nuevas funcionalidades sobre la marcha
- Si la tecnología que va a utilizarse para desarrollar el proyecto es nueva o los integrantes del equipo de desarrollo no están familiarizados con ella.
- En caso de existir riesgos que puedan suponer un cambio en los tiempos o en la planificación del proyecto.
- Si no se puede disponer de un gran número de recursos. Por norma general, los equipos ágiles están formados por un pequeño número de integrantes.
- La relación con el cliente es fluida. La comunicación entre el equipo de desarrollo y el cliente debe poder ser constante.

Por tanto, se utilizará un enfoque ágil siempre y cuando el modelo de trabajo deba ser adaptativo, de modo que se comenzará el trabajo sobre unas funcionalidades básicas y poco a poco se irá añadiendo más contenido y nuevas funcionalidades.

6.3.3. ¿Existe un punto intermedio?

Las posturas ágiles recomiendan que se implante en las organizaciones una forma de trabajar 100% ágil, pero lo cierto es que a la hora de plantear un método de trabajo es común realizar un “híbrido” entre ambas.

Esto es debido a que, cuando se realiza un estudio sobre qué framework de trabajo debería utilizarse, en función de los factores estudiados en puntos anteriores (cliente, proyecto, tecnologías, etc.) se decide tomar un camino intermedio.

Al fin y al cabo, las características principales de los proyectos ágiles es que son apropiados para organizaciones flexibles cuyos proyectos es muy probable que sufran cambios a lo largo del desarrollo. Y los tradicionales suelen ser organizaciones más rígidas con proyectos que por norma general no sufren o sufren pocos cambios, entonces ¿Qué pasa si una organización es flexible, pero sus proyectos no sufren cambios? o bien ¿Qué sucede si la organización no es flexible y el proyecto puede sufrir cambios?

Es por ello que en muchas empresas se opta por adaptar la agilidad a los procesos que convienen con el fin de obtener los mejores resultados.

7. CAMBIO HACIA METODOLOGÍA ÁGIL

Primero se debe responder a la pregunta: ¿Qué es la transformación ágil? Es una transformación de una organización hacia los principios y valores del manifiesto ágil, cambiando la forma de pensar de la empresa, llevándola a un ámbito más colaborativo, flexible, creativo, auto organizado, eficiente y con facilidad para el cambio.

“La agilidad es la habilidad de una organización para renovarse, adaptarse, cambiar rápidamente y tener éxito en un entorno cambiante, ambiguo y turbulento”. Aaron De Smet.

En el entorno actual, en el cual la competitividad entre las organizaciones del mismo sector es muy grande, muchas empresas toman la decisión de enfrentar una transformación hacia la agilidad. Por norma general, este reto implica un cambio profundo y constante durante mucho tiempo en las organizaciones hasta lograr el objetivo final.

Si la organización lleva mucho tiempo gestionada mediante métodos tradicionales, este cambio implicará una renovación en la mentalidad, en los valores, en las actitudes, etc. Es decir, un cambio en la cultura completa de la organización.

Es habitual pensar que existen unos pasos definidos que puedan guiar a una organización hacia una transformación exitosa, sin embargo, esto no ocurre así. No se puede utilizar el modelo de cambio de una empresa y usarlo exactamente en otra, debido a que cada organización tiene sus características y el plan final dependerá de factores como la situación de la empresa, el número de empleados, la implicación de la dirección, etc.

Hoy en día, la mayor parte de las empresas no son ágiles, ya que en un estudio realizado por la empresa Deloitte (Rewriting the rules for the digital age), se indica que el 19% de las empresas encuestadas definen la cultura de su empresa como no ágil. Sin embargo, el 94% de ellas creen en adoptar la agilidad como un paso fundamental para el éxito.

A continuación, se expondrán una serie de puntos a tener en cuenta para lograr una transformación ágil exitosa.

7.1. La llamada a la acción

Uno de los pilares del éxito de una transformación ágil es la implicación de la dirección, por ello es importante que se planteen si ellos mismos son conscientes de los desafíos que deberán asumir, empezando por empaparse de la numerosa bibliografía que existe sobre la agilidad.

Debido al alcance tan global que tiene la agilidad, en muchas ocasiones sucede que en organizaciones con mentalidad tradicional se llevan a cabo ciertas prácticas ágiles. Es importante que la dirección localice esas áreas donde existen esos pequeños avances para comprobar los progresos que han tenido.

Las personas encargadas de llevar a cabo estos pequeños movimientos son valiosos aliados en torno a los cuales sentar las bases para comenzar el camino Ágil. Por otro lado, también deben conocer los esfuerzos que fracasaron, así como las razones que les llevaron al fracaso

7.2. ¿Está la organización preparada?

Se debe establecer un balance entre los riesgos y beneficios que puede tener comenzar con la transformación ágil. La dirección de la empresa debe ser totalmente consciente de que algunas de las prácticas tradicionales (presupuesto, documentación, recursos humanos, etc.) cambiarán radicalmente.

Por supuesto, todos estos cambios no se realizarán el primer día, pero debe existir una concienciación con respecto a que deberán realizarse.

Es importante intentar que el cambio crezca organizativamente desde dentro, para que se sienta como un cambio propio de la organización. Si se trata de imponer cierto método de cambio, porque en otra empresa funcionó (en cambios tan profundos como Agile, “copiar y pegar” no funciona), puede provocar resistencia tanto en los empleados como en la parte institucional de la organización.

La dirección podría finalmente decidir no seguir adelante (no tiene por qué ser algo malo) si no está convencida de lograr los objetivos. Es mejor tomar la decisión de mantener las cosas tal y como están que implantar un Agile superficial que quizá suponga un impedimento para emprender el viaje en un futuro.

7.3. La importancia de los mentores

Como ya se ha estudiado, no se puede tomar el modelo de una empresa e implantarlo directamente en otra, pero si es muy interesante conocer los pasos que han seguido, los problemas que han surgido, etc. Para realizar esto, se pueden realizar visitas a estas empresas, o bien acudir a conferencias en las que se traten este tipo de cuestiones (si bien es cierto que en las conferencias se suele maquillar la realidad).

La clave para tener éxito en la transformación ágil es adoptar una mentalidad ágil, no un proceso o metodología determinado.

Aunque es recomendable consultar a expertos y conocer casos de éxito, no se deben seguir sus pasos al pie de la letra. Algunas empresas que implementan la agilidad siguiendo los procesos definidos en cierta metodología no consiguen adaptarse y lograr beneficios. Sin embargo, otras empresas que demostraban agilidad, no implementaban todas las prácticas ágiles de manual.

7.4. Formar un equipo que lidere la transformación

Aquellos equipos de la organización que hayan logrado implantar algunos comportamientos ágiles podrán formar parte del equipo que se encargará de liderar el cambio. Debe incluir miembros tanto de la dirección como de fuera de ella debido a que la transformación ágil es un viaje interminable y afectará a numerosas áreas de la organización.

- El equipo no debe estar formado únicamente por miembros de la dirección → Podría entenderse como una orden y no calar en los empleados
- El equipo no debe estar formado únicamente por miembros ajenos a la dirección → Podría no contar con la autoridad suficiente para llevar a cabo el cambio.

Por tanto, la mejor solución suele ser formar un equipo mixto, formado por miembros de la dirección y líderes de equipos con conocimientos e iniciativa ágil para llevar a cabo la transformación. La transformación será una tarea muy dura, por lo que, independientemente de su procedencia, deben reunir las siguientes características:

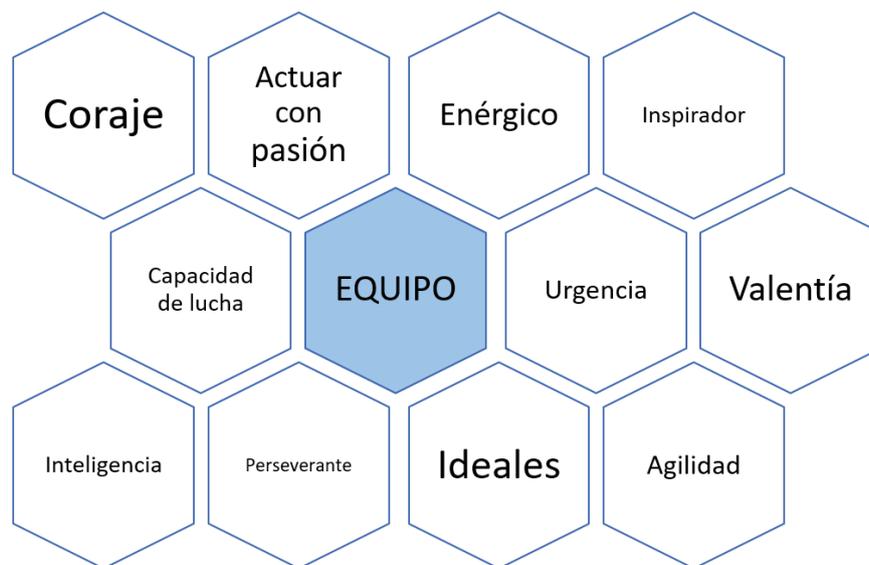


Figura 10. Características del equipo

Ante todo, debe ser un equipo en el cual el sentimiento de confianza entre los miembros sea muy grande, que cuente con el coraje y la valentía de hablar claro a sus superiores, pero la inteligencia de hacerlo cuando y como sea correcto. Debe tener capacidad de luchar contra las adversidades que surjan y actuar con pasión, perseverancia y energía para lograr inspirar y motivar a la organización hacia el cambio de mentalidad.

En organizaciones grandes, el cambio puede llevar décadas. Por ejemplo, Microsoft comenzó su cambio hacia la agilidad en 2008 y hoy todavía lo continúa.

7.5. Entorno y hábitos

Iniciar el cambio de mentalidad hacia la agilidad implica cambiar muchos hábitos. Y muchos de esos hábitos están asociados al entorno de trabajo. Por ejemplo, hay quien toma un café automáticamente al llegar a la oficina (dependiendo de si el entorno lo permite), o quienes deciden reunirse en el mismo lugar como costumbre.

Por ello, realizar pequeñas modificaciones en el entorno de trabajo puede llevar a modificar algunos hábitos. Hay muchos ejemplos de empresas que han llevado el cambio ágil hasta un cambio en la decoración de las oficinas. El caso más claro de esto son las oficinas de Google. Cuenta con numerosas salas de reunión, relajación, trabajo en equipo, zonas para pintar en las paredes, etc.

No todas las empresas cuentan con los recursos necesarios para llevar a cabo una transformación tan radical. Sin embargo, pequeños cambios pueden suponer grandes avances, por ejemplo:

- Juntar a todos los miembros de los equipos de trabajo multifuncionales.
- Cambiar los despachos de los jefes por zonas para equipos donde puedan trabajar juntos.
- Realizar las reuniones de equipo en lugares como cafetería. De este modo se cambia la sala de reuniones por un ambiente más distendido.
- Decorar paredes con mensajes motivadores recordando los principios de la agilidad.
- Si los miembros del equipo realizan la daily sentados o con una actitud apática, se puede plantear una “Zona de dailys” dibujando un círculo en el suelo.

No en todas las empresas existe la misma flexibilidad para cambiar el entorno, pero en la medida de lo posible, realizar pequeños cambios ayudará a los equipos a habituarse a la nueva filosofía de trabajo.

7.6. Demostración de la agilidad: Pruebas

Es muy difícil implantar la agilidad en toda la organización al mismo tiempo (especialmente en empresas muy grandes), por tanto, se puede comenzar demostrando que la agilidad funciona implantándola en pequeños equipos y utilizarlos para difundir la manera de trabajar a otros departamentos de la organización.

Los equipos exitosos hablarán del cambio e inspirarán a otros y a la dirección a llevarlo a cabo en más partes de la organización. Un buen ejemplo es Microsoft:

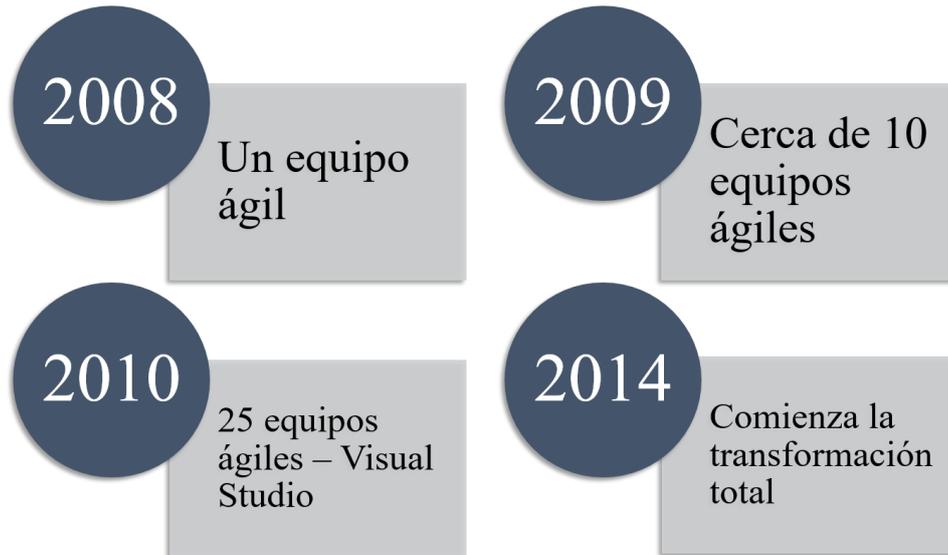


Figura 11. Ejemplo de implantación ágil de Microsoft

Es importante no ceder ante el fracaso. Todas las organizaciones encuentran contratiempos en su camino hacia la agilidad. Es imposible sin una etapa inicial en la que el fracaso se dé diariamente. Lo importante es no caer en las viejas costumbres y continuar el camino ágil.

7.7. Evolucionar

Comenzar con la implantación de la agilidad en equipos es el primer paso, pero no debe terminar en ese punto. Una vez que se ha conseguido que los equipos trabajen como tal, el siguiente paso es lograr que todos los equipos colaboren como uno solo.

Para ello se pueden aumentar las reuniones interdepartamentales, tener en cuenta a los niveles inferiores a la hora de tomar algunas decisiones y provocar un cambio de mentalidad en la organización, de este modo, la idea de agilidad continuará su continua evolución, adaptando las ideas a los cambios y haciendo que todos los equipos formen parte de esa idea.

7.8. Cuidado con la falsa agilidad

Debido al elevado número de directivos que conocen la agilidad, en comparación con 2001 cuando se crea el Manifiesto Ágil, está aumentando considerablemente el riesgo de utilizar la palabra Ágil en organizaciones que realmente no lo son.

Por ejemplo, una práctica ágil como el sprint review puede convertirse erróneamente en un test de evaluación de desempeño para los miembros del equipo, y esta no es su finalidad.

7.9. SHU-HA-RI

Este concepto de las artes marciales japonesas describe las fases hasta la maestría. Aunque se trata de un concepto muy antiguo, se está aplicando cada vez más en la gestión de proyectos.

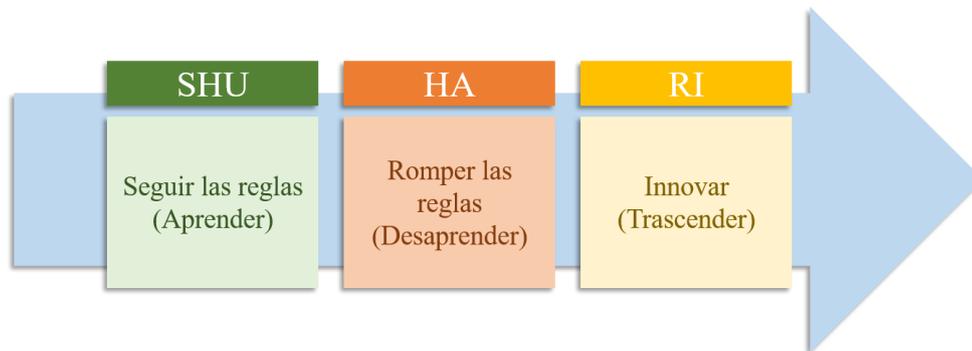


Figura 12. Pasos Shu-Ha-Ri

- **SHU**: En la primera etapa, se deben seguir con precisión las enseñanzas de un maestro. En este caso, los libros, documentación y base teórica para realizar un cambio de mentalidad en la organización.
- **HA**: Cuando las bases están implantadas, se estudia y comprende a fondo la teoría y se es capaz de explorar nuevas alternativas.
- **RI**: En esta etapa, se ponen en práctica sus propias teorías, adaptando la teoría aprendida en las anteriores etapas.

Al finalizar esta etapa, los principios y prácticas ágiles están muy arraigadas en toda la organización. El personal usará los procesos ágiles de una manera fluida.

8. OBSTÁCULOS EN LA TRANSFORMACIÓN ÁGIL

El fracaso a la hora de implementar la agilidad en una organización es muy común. En una investigación realizada por Michael Sahota mediante la realización de encuestas en congresos ágiles y expuesta en su libro “An Agile Adoption and Transformation Survival Guide, 2012”, examina del 0 al 5 (Siendo 0 un fracaso y 5 un éxito) el nivel de éxito de la adopción ágil en las empresas, obteniendo los siguientes resultados:

Congreso	Fecha	0	1	2	3	4	5	Nº	Media
Play4Agile	02/2010		1	6	5	1		13	2.5
XP Toronto	05/2011	1	3	7	10	5		26	2.6
Agile Tour Toronto	11/2011		5	12	23	4		44	2.6
Agile New England	12/2011		1	8	30	10		49	3.0

Tabla 2. Fracaso de la agilidad. Adaptación del estudio de Michael Sahota

Como se puede observar, los resultados no buenos. Cuando una organización comienza el viaje hacia una transformación ágil encuentra numerosos problemas que pueden hacer que este viaje fracase si no se tratan de la manera adecuada.

A continuación, se expondrán una serie de problemas que suelen surgir en las empresas que inician el cambio ágil, y algunas posibles soluciones.

8.1. No hay necesidad de cambiar

En muchas ocasiones, los empleados, o la directiva no creen que sea necesario un cambio. Si las cosas funcionan bien ¿Por qué cambiarlas?

La respuesta es: “porque pueden funcionar mejor”. Una de las bases de la agilidad es una mejora continua en los procesos, eliminando aquellos aspectos que no funcionan o no son necesarios.

Es muy importante que toda la organización entienda la razón por la cual es necesario implementar un cambio en la metodología para abrazar la agilidad. Para ello, deben conocer los aspectos en los cuales los procesos de la organización fallan o son mejorables, y como estos nuevos frameworks de trabajo pueden mejorarlos.

En el siguiente apartado “**estadísticas de transformación ágil**”, se muestra un estudio con una comparativa entre las probabilidades de éxito de proyectos dirigidos bajo metodología tradicional, con respecto a proyectos dirigidos utilizando un enfoque ágil. En este estudio queda claro que los porcentajes de éxito de proyectos ágiles es siempre superior, independientemente del tamaño de la organización.

8.2. Efecto novedad

Este efecto es aplicable en todos los sentidos de la vida y, por supuesto, se puede aplicar también al desarrollo de software. Pero hay una diferencia fundamental entre la novedad y la constancia.

La situación es similar a la de alguien que se apunta a un gimnasio. Al principio, la novedad hace que compre ropa nueva, acuda a diario, tenga una dieta saludable, etc. Sin embargo, es la constancia la que hará que esta persona obtenga resultados a largo plazo

En desarrollo de software y especialmente a la hora de afrontar un cambio hacia metodologías de trabajo ágiles, la constancia es muy importante.

Es muy común que, pasados los primeros meses de trabajo utilizando estos nuevos frameworks de trabajo, la rutina se apodere de los miembros del equipo y se vuelva a retroceder, adoptando de nuevo prácticas que no deberían utilizarse.

El único modo de lograr objetivos y lograr adoptar a largo plazo una cultura de trabajo ágil es la constancia, superar la fase inicial de novedad, continuar con trabajo duro, aprender de los errores y mejorar diariamente.

8.3. Este nuevo método de trabajo no funcionará

Un problema muy usual es la falta de confianza en los nuevos métodos de trabajo. Las técnicas o frameworks ágiles utilizados no son nuevos. Algunas metodologías son muy novedosas, pero otras como por ejemplo Scrum, tienen más de 25 años y su funcionamiento está más que comprobado.

**“Si buscas resultados distintos, no hagas siempre lo mismo”
Albert Einstein.**

Es importante que tanto la dirección como los diferentes equipos estén abiertos a probar y aprender a utilizar estas nuevas herramientas. Deben entender que la adopción de la agilidad puede proporcionar un nuevo enfoque y un salto de calidad a la organización.

8.4. Falta de motivación ante el cambio

En muchas ocasiones los miembros de la organización no se sentirán motivados a cambiar. Independientemente de que crean que el cambio es necesario o no.

La motivación es el impulso que mueve a un individuo para lograr unas metas y persistir en ello hasta lograrlo. Es importante que, de cara a comenzar el viaje hacia la agilidad, todos los miembros del equipo se encuentren motivados por ese cambio.

Nueva identidad: Si los miembros de la organización se sienten parte del cambio, parte de esta nueva identidad de la compañía, es más probable que tengan motivación para llevarlo a cabo. Para ello se pueden aplicar nuevas prácticas con ideas innovadoras que logren que los empleados aumenten el sentimiento de pertenencia con la compañía.

El ejemplo de esto es el fabricante de automóviles japoneses Toyota. En el libro “Las claves del éxito de Toyota”, de Jeffrey K. Liker, cuenta que, en los años 80, para lograr que los empleados se sintieran motivados con su trabajo, les dieron la responsabilidad de parar las líneas de producción si detectaban cualquier error (esto ahora es común, pero en los años 80 era una práctica difícil de ver), así como proponer cambios y mejoras que serían implantados si son interesantes. Estas medidas continúan en la actualidad.

Promueve también el Club de Asistencia, que se basa en recompensar el absentismo cero, mediante una invitación a un banquete y el sorteo de coches de la compañía entre los miembros del club.

Con esto consigue que entre el 60% y 70% de los empleados formen parte del Club de Asistencia, que en las cadenas de montaje se promuevan unas 80.000 ideas de mejora de las cuales se implementa el 90%.

Cubrir las **necesidades** de los empleados. Unos empleados satisfechos estarán mucho más motivados para el trabajo que implicará el cambio hacia una mentalidad y método de trabajo ágil.

Jeffrey indica también que Toyota aplica hasta 5 teorías de la motivación (Jerarquía de las necesidades de Maslow, Enriquecimiento de trabajo de Herzberg, Dirección científica de Taylor, Modificación del comportamiento y Fijación de metas). Permitiendo y valorando la mejora continua, trabajo orientado a objetivos, retroalimentación, rotación de puestos de trabajo y autorrealización entre otros.

8.5. Resistencia al cambio

Siempre que se desea realizar un cambio de mentalidad, o implementar una nueva idea, es común que haya miembros que no deseen realizar ese cambio y se nieguen a él. Pero esta negación puede venir por diversos factores, siendo algunos más preocupantes que otros.

En el libro “Beyond the Wall of Resistance: Why 70% of all changes still fail” de Rick Mauer se hace referencia a tres niveles de negación al cambio. Hay personas a las que el miedo al fracaso, o a perder su empleo les obliga a decir que no.

- **No, no lo entiendo:** Es probablemente el no más fácil de solucionar. Es importante que los miembros del equipo comprendan que el cambio no conlleva un examen constante. Aprender significa equivocarse, y se alcanzará el objetivo a base de aciertos y errores de los que aprender. Se deberá ofrecer el nivel de formación necesario e incluso contar con expertos que sirvan de guía.
- **No, no me gusta:** Este significa instinto de supervivencia. Entra el juego el miedo a perder el control, puesto de trabajo o estatus en la empresa. Puede no llegar a comprenderse si se trata realmente de algo que no gusta o si es temor a lo desconocido. Intentar justificar la necesidad de cambio en base a datos no tendrá resultado, debido a que, aunque el cambio esté justificado, puede no estarlo para esa persona o ese momento en concreto. Es más interesante abordar este cambio como un aprendizaje, como la adquisición de nuevas habilidades que puedan servir de mejora profesional, y no de tumba.
- **No, no me gusta el líder:** En este caso es una negación debido a una falta de confianza en la persona que lidera el cambio. Es la negación más problemática porque, aunque el cambio esté justificado, no querrán que el líder actual lo lleve a cabo. También tiene solución, pero no es tan sencillo y normalmente pasa porque una de las dos personas cambie de empleo o de departamento.

8.6. Miedo al cambio

En el artículo “Es mayor el miedo al cambio, que continuar sufriendo la situación actual” publicado por Javier Garzás, experto en metodologías ágiles y gestión del cambio, expone dos fases en las cuales se dialoga con los integrantes del equipo para conocer la situación actual y su aceptación del cambio.

- **Fase 1:** Cuando en una empresa se propone un cambio hacia metodologías de trabajo ágiles, es interesante reunirse con los integrantes del equipo para conocer cuáles son sus impresiones. Los problemas siempre son similares, relacionados con la mentalidad tradicional: requisitos cerrados, proyectos enormes, temporización inflexible, fechas que no son reales, testing ineficiente, etc.
- **Fase 2:** Por lo tanto, es interesante proponer un cambio para intentar solventar todos esos problemas, pero en este punto, la mentalidad del equipo cambia y comienzan los impedimentos hacia el cambio:
Posibles problemas con los clientes, retrasos en entregas, discrepancias entre directiva y desarrollo, escepticismo con respecto al buen funcionamiento de esos métodos en este contexto, etc.

Lo curioso es que, todos estos impedimentos son simplemente “posibilidades”. No se tiene la certeza de si esos problemas surgirán (o no), pero lo que sí está claro, es que los puntos en común alcanzados en la fase 1 sí están sucediendo. En este punto, parece que la situación anterior no era tan mala, pero no es así.

Es por eso que el título del artículo cobra mucho sentido debido a que sucede exactamente eso, la gente tiene mayor miedo a abandonar su zona de confort, que lo que les supone continuar sufriendo unas condiciones de trabajo que no son de su agrado.

8.7. Las reuniones son una pérdida de tiempo

Algunos de los puntos fuertes de la agilidad son la retroalimentación y la comunicación entre miembros del equipo y entre equipos. Para ello, es muy común el uso de diversas reuniones de seguimiento. En este sentido, las dos que generan mayor animadversión son la daily y la retrospectiva.

Es muy común que estas reuniones sean vistas por los miembros del equipo (incluso por parte de la directiva) como una pérdida de tiempo, algo que no aporta valor y que debería ser eliminado para aumentar la productividad. Esto suele suceder porque las reuniones no se están llevando a cabo de la manera correcta.

8.7.1. Daily

Como se ha contado en el apartado de reuniones de Scrum, una daily es una reunión diaria, de poca duración, dirigida por el Scrum Master, en la que cada uno de los miembros del equipo expone el trabajo que realizó el día anterior, las tareas que realizará y si tiene algún obstáculo para llevarlas a cabo.

La teoría es sencilla, pero a la hora de ser llevada a la práctica se cometen muchos errores que terminan por hacer que la reunión sea ineficiente, innecesaria, que se evite o que se convierta en una rutina evitable. Esto es un problema, debido a que, en entornos ágiles, estas reuniones son vitales.

Es importante entender que las daily no son reuniones en las cuales se deba informar a los jefes de los avances en los proyectos, son reuniones en las que se informa a los demás integrantes del equipo acerca del trabajo que se está realizando.

La puntualidad es importante, es común realizar las reuniones en el puesto de trabajo, siempre a la misma hora (idealmente a primera hora), de tal forma que todo el equipo asista.

Las reuniones deben prepararse, es importante que cada miembro recuerde las tareas que realizó el día anterior, así como que conozca las tareas que debe realizar a lo largo del día. Debe realizar un breve resumen del trabajo realizado, a grandes rasgos. No entrar en resúmenes detallados o excesivamente técnicos.

En estas reuniones no se trata de resolver los problemas que surjan (no quiere decir que no se proporcionen algunos consejos si es un problema rápido). Los problemas se solucionarán posteriormente con la ayuda del Scrum Master. En la reunión lo importante es plantear estos problemas para que el equipo sea consciente de ellos.

El turno de palabra es importante, es común proponer un orden de intervención o utilizar algún elemento que otorgue el turno de palabra. No se debe superar el tiempo máximo de intervención de cada participante. En este punto, el Scrum Master será el encargado de sintetizar la información y animar a los participantes a finalizar su turno si se está excediendo el tiempo.

Debe participar todo el equipo. En ocasiones, debido a la falta de confianza hay miembros del equipo que no participan en la reunión. Con el fin de mejorar este ámbito, se pueden realizar tomando un café o en un ambiente distendido.

En la siguiente figura se muestra un resumen rápido de las buenas y malas prácticas explicadas anteriormente a la hora de realizar la Daily.

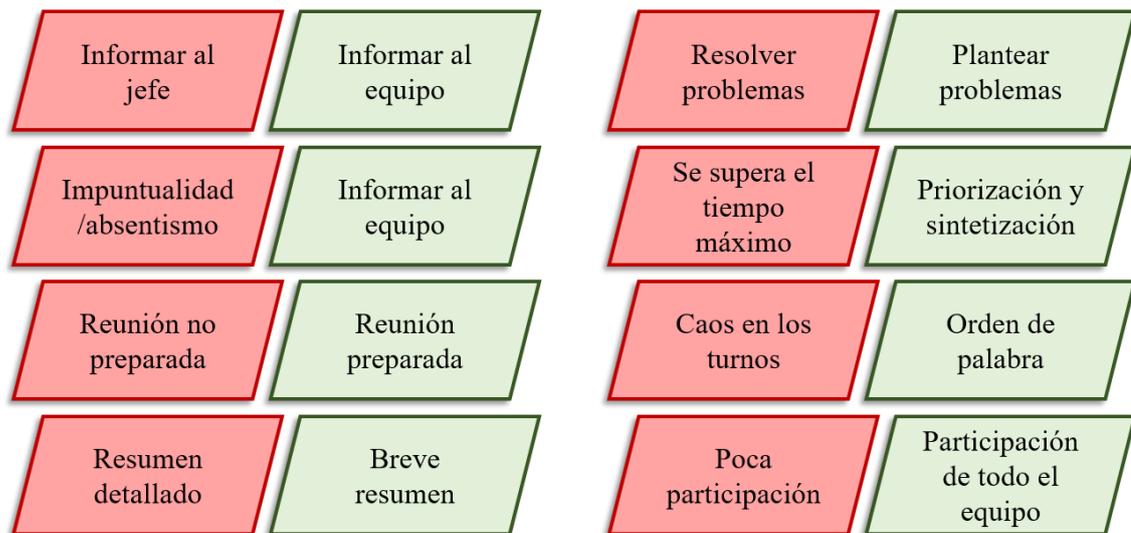


Figura 13. Buenas y malas prácticas en Daily Meeting

También es muy común el aburrimiento durante la realización de la reunión. Esto sucede principalmente porque a la persona no le interesa lo que los demás tienen que contar y no presta atención al resto de participantes para “reconectar” cuando le toca el turno.

Esto denota una falta de trabajo en equipo, pero también puede producirse porque en la reunión se utiliza un lenguaje excesivamente especializado. En estas reuniones participan desarrolladores, testers, diseñadores, etc. Cada uno de ellos con sus tecnicismos, por lo tanto, es importante utilizar un lenguaje coloquial evitando los tecnicismos.

También puede deberse a que el equipo en conjunto no persigue el mismo objetivo o están centrados en tareas muy diferentes no relacionadas entre sí.

8.7.2. Retrospectiva

Una retrospectiva es una reunión que se realiza al finalizar un sprint en la que el equipo debe autoevaluarse, definiendo acciones de mejora que se podrán incorporar en futuros sprints.

En muchos equipos, estas reuniones son vistas como una pérdida de tiempo y que no consiguen aportar un valor añadido. Esto es debido principalmente a los siguientes puntos:

- **No se sacan conclusiones:** La reunión finaliza, pero no se ha logrado un consenso sobre las posibles mejoras a implantar o bien, esas mejoras son difíciles de implantar a corto plazo.
- **Reunión muy larga:** Es importante definir un tiempo de reunión, no muy elevado, y que se respete en la medida de lo posible.
- **Reuniones repetitivas:** La retrospectiva debe ser una reunión amena, en la que todos participen. Es por ello que hay diversas técnicas para llevarla a cabo y no

caer en la monotonía (dinámica de la estrella de mar, dinámica de las 4L, entre otras). Esto puede solventar también el problema de la falta de confianza, que en muchas ocasiones provoca que los miembros del equipo no propongan ideas, creando un ambiente más ameno.

- **¿El Product Owner debe participar?:** La respuesta es sí. Es un miembro más del equipo y en muchas ocasiones se comete el error de mantenerle al margen de este tipo de reuniones debido a que se entiende que es un tema que no le aplica.
- **Falta de preparación:** La retrospectiva es una reunión que se debe planificar con tiempo. Por tanto, todos los miembros del equipo deben preparar la reunión a lo largo del sprint, anotando las ideas de mejora, los puntos débiles y los comportamientos a eliminar.

Algunas de las técnicas de retrospectiva utilizadas son las siguientes

8.7.2.1. Técnica de la Estrella de Mar

Mediante esta dinámica se intenta obtener una gran cantidad de opiniones de todos los miembros del equipo. Para ello se definen 5 categorías, y cada integrante del equipo debe proponer (al menos) una idea o una modificación para cada una de ellas, escribiendo en un post-it y pegándola en el diagrama.

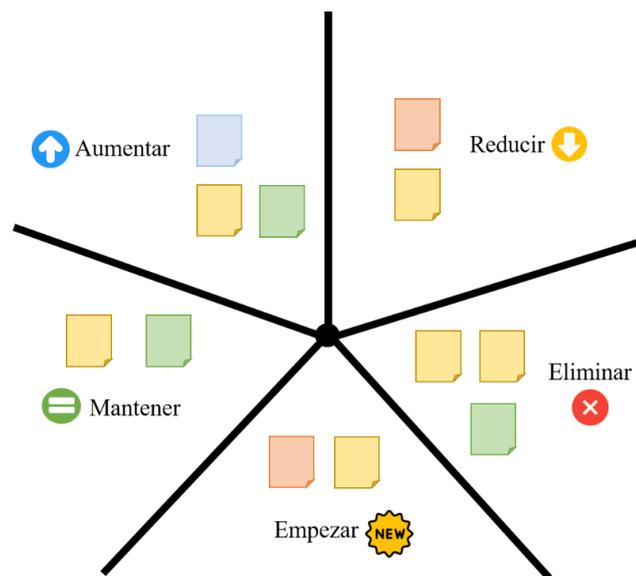


Figura 14: Dinámica estrella para retrospectivas

- **Empezar:** Son ideas que sería interesante comenzar a llevar a cabo a partir del próximo sprint
- **Eliminar:** Son comportamientos o procesos que sería interesante eliminar.
- **Aumentar:** Actividades que pueden haber influido en un buen resultado y que es interesante incidir en ellas de cara a futuros sprints
- **Reducir:** Al contrario que el anterior, son actividades que pueden no haber aportado el valor deseado y es conveniente reducir su uso
- **Mantener:** Actividades que el equipo desea mantener

Este cuadro se mantendrá intacto hasta la siguiente retrospectiva, en la cual el equipo puede votar para realizar cambios en los cuadros y mover ideas de uno a otro.

8.7.2.2. Técnica de las 4L

Es quizá la técnica más utilizada en la retrospectiva. La idea es similar a la técnica de la estrella de mar, pero está centrada en cosas que cada individuo ha aprendido, o que cree que el equipo debe hacer mejor. Se basa en cuatro palabras en inglés (liked, learned, lacked, longed for)

- **Liked** (me ha gustado): Comportamientos o técnicas individuales o colectivos que han parecido interesantes
- **Learned** (he aprendido): Cosas que cada miembro del equipo ha aprendido durante el último sprint
- **Lacked** (se debe mejorar): Actividades que el equipo ha realizado, pero en las que hay un margen de mejora
- **Longed for** (se deberían hacer): Actividades que se habría sido interesante realizad.

La idea es que cada participante anote en un post-it al menos una idea por cada categoría. Estas ideas serán votadas y debatidas para después compararse con el resultado de la retrospectiva anterior.

8.8. Si algo falla, aparece la negatividad

En el camino hacia una transformación ágil, no todo irá sobre ruedas y en muchas ocasiones habrá que superar momentos difíciles. El camino hacia el éxito es una carrera de fondo, y no una carrera de velocidad (como muchos esperan).

En algún momento del camino, alguien propondrá la pregunta: “Merece la pena continuar con la transformación ágil?”. John Fisher, investigó las reacciones de las personas ante un cambio organizacional, creando de este modo la Curva de Transición Personal.



Figura 15. Imagen propia. Adaptación curva de Fisher

La curva comienza con una fase de ansiedad en la cual los equipos no están seguros de poder llevar a cabo el cambio, aunque existe la felicidad por realizar cambios. A partir de este punto, comienzan las dudas.

En esta curva, Fisher indica dos puntos clave en los cuales se debe evitar caer, y que una vez superados, llevarán a aceptar los problemas y avanzar en el camino al objetivo final.

- El **primero** de ellos es no superar el sentimiento de amenaza, lo que resultaría en miedo y ansiedad, y terminaría por el abandono debido a que no se consigue llevar a cabo el cambio.
- El **segundo** punto crítico es la depresión, cuando la moral personal o del equipo está más baja. En este punto, se puede llegar a reaccionar con hostilidad, tratando de volver a las viejas costumbres de trabajo y abandonando el camino hacia la agilidad.

Superado el segundo punto crítico, lo siguiente es aceptar los problemas, superarlos y continuar el camino hacia el éxito.

Otro modelo interesante es el llamado “The Emotional Cycle of Change” de los psicólogos Don Kelley y Daryl Connor, que puede observarse en la figura 16.

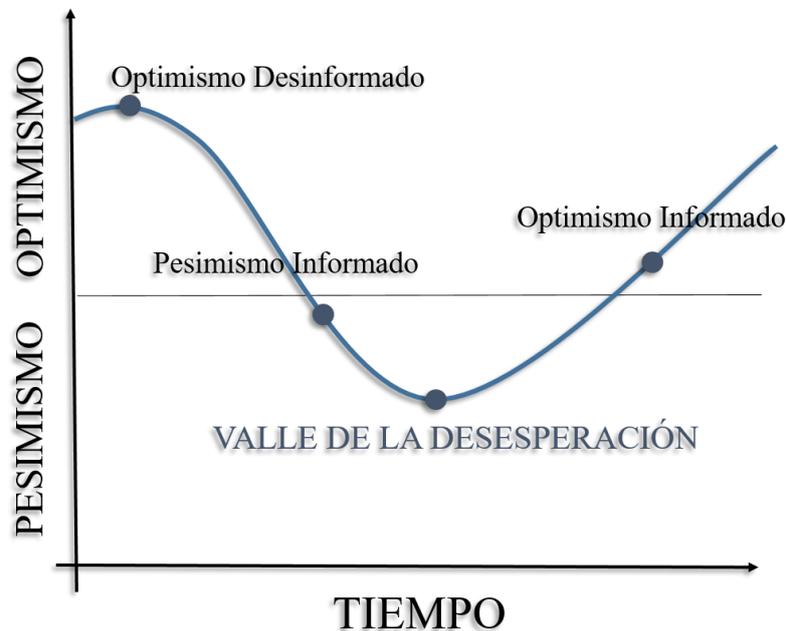


Figura 16. Imagen propia. Adaptación modelo Don Kelley y Daryl Connor

En este modelo se comienza con una fase inicial (**optimismo desinformado**) donde se imaginan los beneficios que se obtendrán con el cambio, pero no existe el conocimiento de los esfuerzos que serán necesarios para llevarlo a cabo. A continuación, una fase (**pesimismo informado**) en la que se entra en un estado de negatividad debido al conocimiento del esfuerzo que será necesario para obtener éxito, entrando de este modo en el “**valle de la desesperación**”.

En este punto, la tentación de volver a las viejas prácticas es grande debido a la incertidumbre que provoca el cambio. Si se supera esta fase, se llega al **optimismo informado**. En este punto, comienzan a verse resultados, se es plenamente consciente del esfuerzo que se ha realizado y que será necesario realizar y se comienza el viaje hacia el éxito.

Por tanto, es importante no ceder ante la negatividad, mantener una **actitud positiva**, si bien no es una certeza para lograr el éxito, si es una condición indispensable. No es común ver equipos pesimistas obtener grandes éxitos.

8.9. Realizar un cambio para no cambiar nada

Cuando las organizaciones comienzan el viaje hacia la transformación ágil, deben tener claras las metas que quieren conseguir para asegurarse que, al final del camino, se han logrado y no se ha vuelto a las formas de trabajar tradicionales.

Muchas empresas realizan una profunda transformación, pero es muy sencillo confundir los comportamientos tradicionales con nombres ágiles. Este punto ya se ha tratado anteriormente en el apartado “Cambio hacia metodología ágil”, pero es interesante ampliarlo con tecnicismos que pueden convertir lo que parece una práctica ágil, en una manía tradicional “disfrazada”. En la siguiente figura se muestran las prácticas tradicionales (columna de la derecha) que se pueden esconder bajo nombres ágiles.

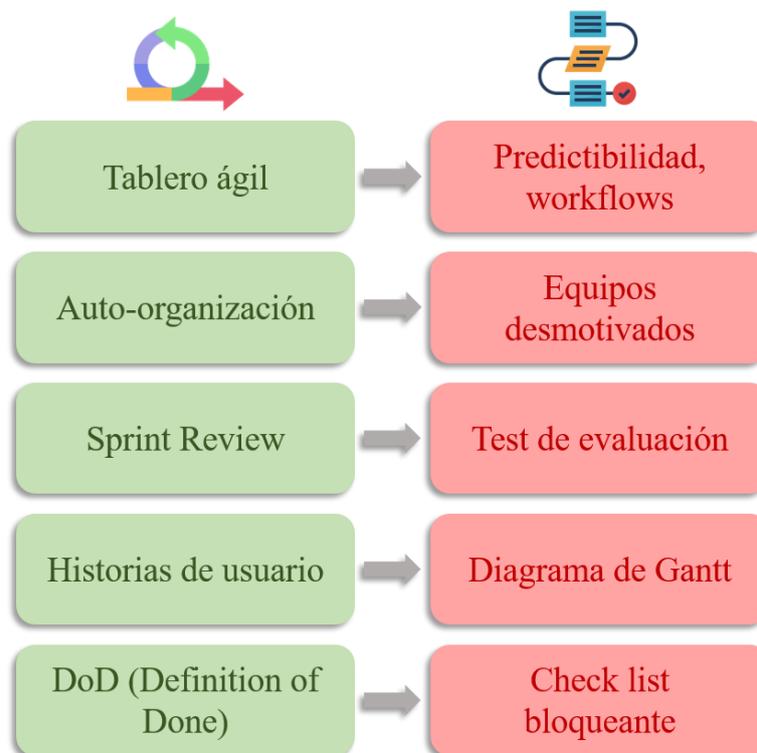


Figura 17. Metodologías tradicionales ocultas bajo nombres ágiles

Es importante mantener siempre la perspectiva de lo que se desea conseguir y evitar caer en las prácticas comunes que están más arraigadas en la organización. Para ello se pueden definir evaluaciones que permitan determinar la madurez de la implantación, comparando el punto de partida con el punto actual.

8.10. Mis empleados no quieren cambiar

A la hora de plantear una transformación ágil es importante conocer las circunstancias de toda la organización, y uno de los factores más importantes son los empleados y su predisposición al cambio.

Implantar agilidad en una organización es una manera de pensar que debe arraigar en toda la empresa, de modo que la dirección, no debe presuponer que los empleados no quieren cambiar, no quieren evolucionar y no quieren responsabilidades, sino que deben pensar que los empleados desean trabajar, crecer profesionalmente y afrontar retos y responsabilidades (Teoría motivacional X e Y de Douglas McGregor). En la siguiente imagen se explican las características que tienen los empleados según esta teoría



Figura 18. Teoría X e Y de la Motivación de McGregor

Hoy en día, y especialmente en agile, la teoría que predomina y en la que más se cree es en la teoría Y de McGregor.

Si se decide no realizar una transformación ágil porque los empleados “no quieren cambiar” se está cometiendo un error de atribución al presuponer que los empleados no desean cambiar y achacar el fracaso o la negativa al cambio a ese factor.

Los empleados realizan cambios en su vida diaria (bodas, hijos, casas, cambios de empleo, etc.), por lo tanto, no se debe presuponer que se opondrán totalmente a un cambio en su vida profesional.

8.11. Contratar a gente experta, para nada

Una práctica muy común en muchas empresas a la hora de abordar una transformación ágil es contratar a expertos en la materia para que les guíen en el camino. El problema es cuando desde la organización no se permite a estos expertos realizar su trabajo o se les ubica bajo la supervisión de alguien con menos conocimientos de agilidad. La siguiente frase de Steve Jobs refleja a la perfección este punto:

“No tiene sentido contratar a personas inteligentes para decirles lo que tienen que hacer; contratamos a personas inteligentes para que nos digan lo que tenemos que hacer”
Steve Jobs.

Si una organización contrata un experto para que ayude en el camino hacia la transformación ágil, es importante dejar a estas personas hacer su trabajo.

8.12. Ya no es necesario documentar: Error

Antes de que comenzara la “moda” de la agilidad, lo normal era realizar documentación de absolutamente todo el software, de modo que los documentos que se realizaban eran muy extensos y difíciles de mantener actualizados. Esto provocaba que cuando era necesario consultar esa documentación, no servía.

Es muy común que cuando en una empresa realiza una transformación hacia la agilidad, piense que implica no realizar documentación. Esto no es así, el manifiesto ágil indica que es más importante un Software que funciona que una documentación excesiva, no que la documentación no deba existir.

Porque, si no existe, ¿Qué sucede si un desarrollador abandona la compañía?, ¿Qué sucede si un nuevo empleado debe comprender el código existente?, ¿Qué sucede si el sistema crece y se pierde la perspectiva del mismo?

Esta tarea es necesaria y debe realizarse, pero evitando los voluminosos y detallados documentos que explican hasta el último detalle del software. Es importante generar documentos sin detalles innecesarios y relacionados con los requisitos.

Está claro que se debe documentar, pero no en exceso, pero ¿Cómo saber cuándo es suficiente la documentación existente? Una posible solución es limitar el número de páginas con las que contará la documentación, de modo que el tiempo que se empleará en ella no será excesivo y siempre existirá una base a la que consultar en caso de ser necesario.

La conclusión es: **Se debe documentar, pero de manera ágil**

8.13. Agile significa que no necesitamos procesos

Uno de los puntos del Manifiesto Ágil indica que se priorizarán los individuos e interacciones frente a los procesos y las herramientas. Se tiende a pensar en que los procesos son algo propio de desarrollo en cascada, y no de equipos ágiles.

Este punto puede ser engañoso, debido a que no se trata de eliminar los procesos, sino de eliminar aquellos que no son útiles y mejorar los procesos que son realmente necesarios.

En la “Guía Definitiva de Scrum (2013)”, de Jeff Sutherland y Ken Schwaber, se indican las bases para el control de procesos:

Transparencia

- El proceso debe ser visible para los responsables del resultado final
- Se debe compartir una definición común de “terminado”
- Utilizar un lenguaje común

Inspección

- Revisión continua de procesos
- No debe interferir en la realización del trabajo

Adaptación

- Reajuste de los procesos
- Debe realizarse cuanto antes

Tabla 3. Bases de control de procesos. “Guía definitiva de Scrum, Jeff Sutherland y Ken Schwaber”

La inspección y adaptación suele realizarse durante las reuniones (planificación de sprint, Daily, revisión y retrospectiva). Para Jeff Sutherland, la eficiencia del proceso es clave en equipos de alto rendimiento y se debe tener en cuenta que:

- Es responsabilidad del Scrum Master maximizar la eficiencia de los procesos
- El tiempo debe medirse en horas dedicadas y no en tiempo.
- Un equipo debe tener estrategias
- No crear historias de usuario que no aporten valor.

8.14. Yo soy un directivo, eso es de técnicos

Para que el cambio ágil llegue a producirse de una manera exitosa, es imprescindible que este se lleve a cabo en toda la organización, desde la directiva hasta el último empleado de la organización.

Es un clásico que la directiva no entienda lo que implica el cambio hacia una mentalidad ágil o ni siquiera comprenda la razón por la cual se está implantando agile en su organización.

Cuando ante preguntas del estilo: ¿Por qué estáis implantando esta agile? ¿Por qué habéis seleccionado una metodología determinada con respecto a otra? ¿Cómo se está realizando el cambio de mentalidad? La respuesta es “Eso es de técnicos”, implica que algo no está funcionando y es más que probable que el viaje ágil se quede a medio camino.

Esto lleva además a otro problema y es que esos denominados “técnicos” saben que desde la directiva no saben de qué trata el trabajo que se está realizando, lo que disminuye la confianza en la directiva y en el proyecto.

La parte directiva deber ser la principal interesada y la principal precursora de implantar frameworks de trabajo ágiles en su empresa. Y es que una de las bases de la agilidad es que **negocio y departamento de IT estén alineados**, y para ello es imprescindible que desde la parte directiva comprendan el trabajo que se está realizando.

Por ejemplo, en Scrum, el Product Owner es parte del equipo y al mismo tiempo pertenece al área de negocio.

En el estudio “13th anual State of Agile Report”, realizado con más de 1300 empresas se indica que uno de los obstáculos más grandes a la hora de finalizar con éxito una transformación ágil en una empresa es el apoyo insuficiente y la falta de conocimientos por parte de la dirección (40%).

8.15. Cuéntame lo que quiero escuchar, no la verdad

La transformación de una organización hacia la agilidad no es una tarea sencilla, ni que vaya a finalizarse con éxito en poco tiempo (ni siquiera se tiene la certeza de lograr una transformación exitosa). Es un cambio de mentalidad y de enfoque que debe ver sus resultados a largo plazo.

Por norma general, las empresas medianas y grandes no inician una transformación sin ayuda, sino que suelen contar con el consejo o participación de un **Agile Coach**. Este, al igual que el Scrum Master, debe ser una persona experta en agilidad, pero su principal labor, al contrario que la del Scrum Master, es la de lograr que los diferentes equipos ágiles encajen entre sí y en la organización.

También es la persona que se encargará de poner en valor el trabajo de estos equipos frente a la directiva. Es una persona que será capaz de gestionar las emociones tanto personales como grupales, así como mantener la motivación del equipo en caso de ser necesario.

Las empresas buscan obtener unos resultados de manera rápida y sencilla. Siempre habrá un Agile Coach que proporcionará una visión errónea a la directiva sobre la “facilidad” de adoptar el cambio o sobre los progresos que se están haciendo en ese camino, simplemente para que la dirección escuche lo que quiere.

9. ESTADÍSTICAS DE TRANSFORMACIÓN ÁGIL

Debido a todos estos posibles problemas muchas organizaciones deciden abandonar el viaje ágil o volver a algunas de las antiguas prácticas. En este apartado se van a exponer algunas de las estadísticas obtenidas en el estudio: “Standish Group CHAOS Report Series, Decision Latency Theory, James Johnson 2015”, en estos estudios se muestran las tasas de fracaso y éxito de proyectos tradicionales y ágiles

En este estudio se indican tres posibles estados para definir el estado de un proyecto: exitoso, dudoso y fallado en función de tres variables: entrega a tiempo, entrega en presupuesto y alcance deseado.

- **Un proyecto exitoso** es aquel que cumple las tres constantes
- **Un proyecto en duda** es aquel que ha cumplido dos de las tres constantes, es decir, existen dudas acerca de si ha sido un éxito o no.
- **Un proyecto fallado** es aquel que ha sido cancelado, ha finalizado, pero sin cumplir las variables o que se completa, pero no se utiliza.

9.1. Tasa de éxito según metodología

En el informe de Standish Group CHAOS Report de 2015, en un estudio realizado sobre más de 10.000 organizaciones se indican las tasas de éxito de los proyectos en función de la metodología de trabajo utilizada (tradicional o ágil):

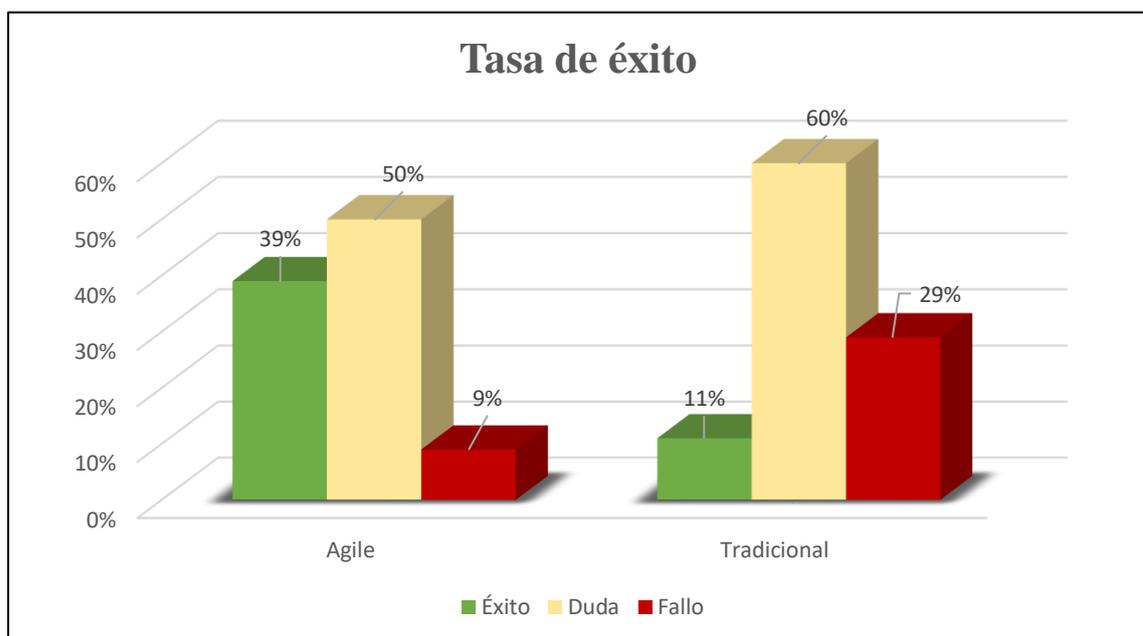


Gráfico 1. Gráfico propio. Tasa de éxito. Estadísticas extraídas de: Standish Group CHAOS Report (2015)

En la gráfica se observa que la tasa de éxito de proyectos realizados bajo metodología ágil es superior a la de proyectos realizados bajo métodos de trabajo tradicionales, así como que la tasa de fallo es superior en proyectos tradicionales.

9.2. Tasa de éxito según metodología y tamaño

Uno de los factores más importantes en el éxito de una organización es el tamaño del proyecto. En la siguiente gráfica se muestra una comparativa del éxito de un proyecto en función de la metodología utilizada.

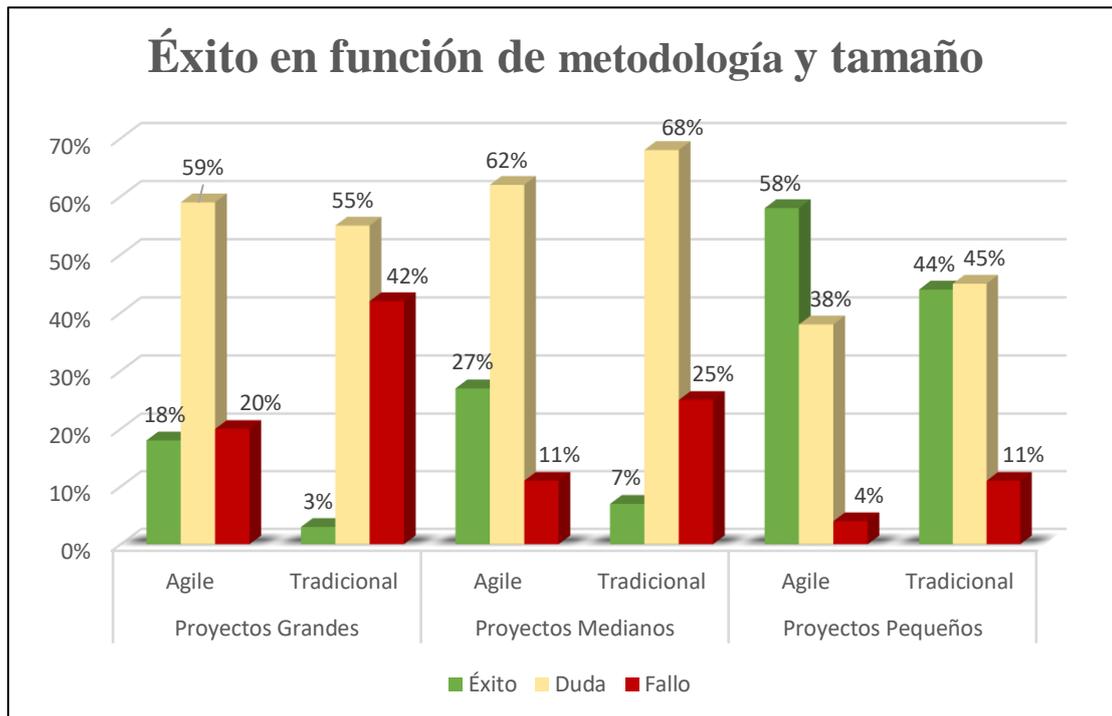


Gráfico 2. Gráfico propio. Éxito en función de metodología y tamaño. Estadísticas extraídas de: Standish Group CHAOS Report (2015)

Como se observa en la gráfica, los proyectos ágiles funcionan especialmente bien en proyectos pequeños, donde obtienen su tasa de éxito más elevada y su tasa de fracaso más baja.

Sucede lo mismo con los proyectos tradicionales, pero visualizando el gráfico en su conjunto, queda patente que los proyectos realizados bajo frameworks ágiles obtienen una tasa de éxito mayor y una tasa de fracaso menor.

9.3. ¿Por qué cambiar a Agile?

En el estudio “13th anual state of agile report” de Collabnet VersionOne, realizado en 2019 sobre más de 1300 empresas, cuando se pregunta sobre cómo las organizaciones miden el éxito de las transformaciones ágiles, los encuestados indicaron que la satisfacción del cliente o del usuario, el aumento del valor comercial y el tiempo de entrega fueron las medidas de éxito más valoradas.

Seguidas de la calidad, productividad, previsibilidad, mejora de procesos y visibilidad del proyecto.



Figura 19. Imagen propia. Estadísticas de 13th Annual State of Agile Report 2019. Medidas de éxito.

9.3.1. Técnicas Agile más utilizadas

Otro dato importante a tener en cuenta son las técnicas Agile más utilizadas, en casi todas las empresas se utiliza mucho un pilar fundamental en entornos ágiles que son las reuniones. En un 80% de las empresas se realizan todas las reuniones ágiles, siendo la Daily la reunión estrella, realizándose en un 86% de ellas.



Figura 20. Imagen propia. Estadísticas de 13th Annual State of Agile Report 2019. Técnicas ágiles más utilizadas.

9.3.2. Metodologías Agile más utilizadas

En el estudio “13th anual state of agile report” se indica que las metodologías ágiles más utilizadas en 2019, de entre las empresas encuestadas, fueron las siguientes:

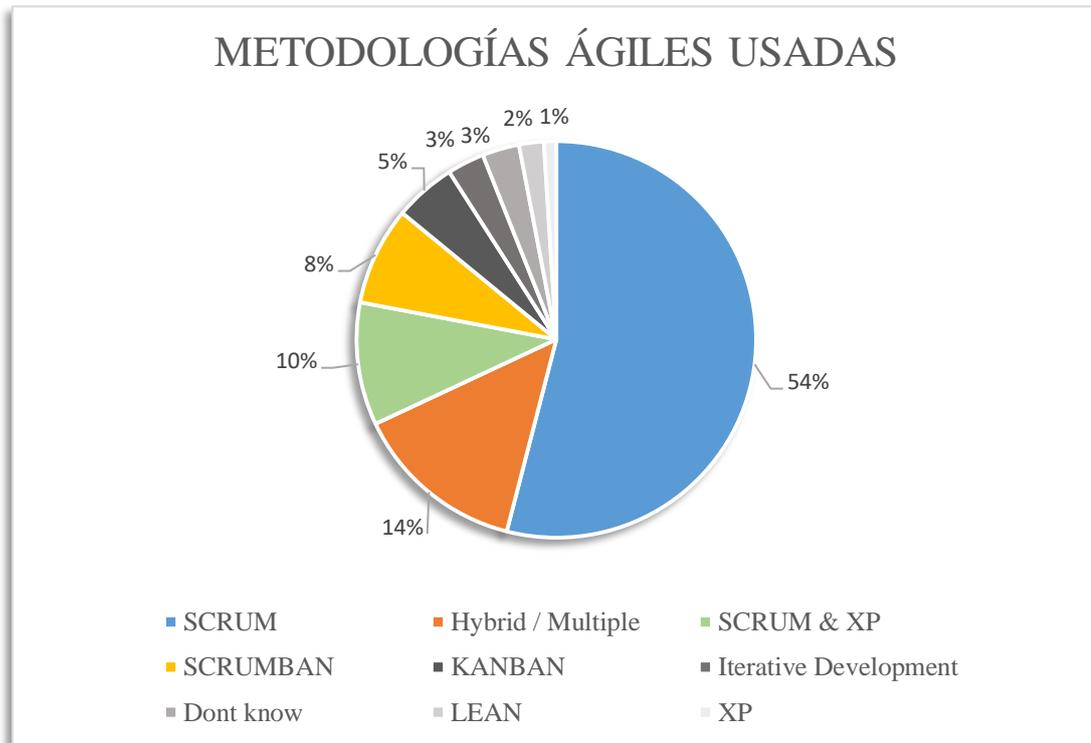


Gráfico 3: Gráfico propio. Metodologías ágiles más utilizadas. Estadísticas extraídas de 13th Annual State of Agile Report 2019

Se observa que SCRUM es utilizado en un elevado porcentaje de empresas, resultando SCRUM, SCRUM & XP y SCRUMBAN en conjunto un 72% del total.

10. CONCLUSIONES

Mediante este trabajo de fin de master se ha conseguido realizar una introducción tanto a las metodologías tradicionales como a los frameworks ágiles. Aunque no se ha profundizado en cada una de las diferentes metodologías existentes de cada tipo, si se han contado las bases de las más importantes. También se han expuesto diversos puntos que se deben tener en cuenta a la hora de comenzar una transformación ágil en una organización, así como algunos de los problemas más comunes que surgen durante esa transformación.

A continuación, se exponen las conclusiones más relevantes extraídas tras la consecución del trabajo:

- En cuanto a qué metodología es mejor, es una tarea complicada decidir si es conveniente utilizar metodologías tradicionales o una filosofía ágil, debido a que dependerá de diversos factores. Adoptar una mentalidad ágil no asegura el éxito, sino que depende mucho del entorno del proyecto.
- Una manera de lograr implantar con éxito una metodología es seleccionar la que más convenga en función del entorno y adaptando aquellos puntos que cumplan con las necesidades del proyecto.
- Cuando se inicia una transformación ágil, pueden surgir problemas debidos al miedo que sienten los miembros de la organización frente al cambio. Muchos de estos miedos son debidos a la inseguridad, falta de conocimientos o el temor a perder el puesto de trabajo. Lo importante en este punto es conseguir superar esa etapa de duda y no volver a las prácticas anteriores.
- Otros problemas son responsabilidad de la directiva de la organización, debido principalmente a una falta de implicación con el cambio, desconociendo lo que supone, tratando de seguir al pie de la letra los mismos pasos que llevaron al éxito a otras empresas, o no contando con la opinión de los expertos.
- La tasa de éxito de proyectos ágiles con respecto a proyectos tradicionales es superior, independientemente del tamaño de la organización. Sin embargo, esas mismas estadísticas muestran que muchos proyectos ágiles fallan. Y es que modificar la filosofía de una empresa es una tarea muy complicada, a largo plazo, que supone la implicación de toda la organización y que no siempre asegura el éxito.

11. LÍNEAS FUTURAS

Hoy en día, el software está cada vez más presente en la vida diaria, lo que supone que el desarrollo de software y por tanto las metodologías de desarrollo avanzan rápidamente, adaptándose a las necesidades de la organización y del cliente y aumentando la competencia entre las empresas.

Conforme evolucione el desarrollo de software, las organizaciones deberán adaptarse y se crearán nuevas metodologías de trabajo que quizá aparten a un lado a los frameworks ágiles.

Como líneas futuras relacionadas con desarrollo ágil de software se podrían tratar los siguientes temas:

- Realizar una comparación de metodologías de trabajo tradicionales y ágiles en un mismo proyecto, evaluando la conveniencia de utilizar una u otra en función de su entorno y características.
- Profundizar en el Management 3.0, un nuevo estilo de liderazgo que implica trabajar juntos para lograr los objetivos de la organización de la manera más eficiente posible.
- Realizar encuestas a profesionales que se encuentren trabajando bajo estas metodologías, con el fin de comparar las impresiones y satisfacción laboral en cada uno de los casos.
- Analizar el futuro del Agile, si tiene fecha de caducidad y posibles nuevas corrientes que estén surgiendo.

12. BIBLIOGRAFÍA

LIBROS

K. S.-B. Object design and implementation and undefined 1997, “Scrum development process,” *Springer*

“An Agile Adoption and Transformation Survival Guide, 2012”, Michael Sahota

“Las claves del éxito de Toyota”, de Jeffrey K. Liker

“Decision Latency Theory, Standish Group CHAOS Report Series” Jim Johnson

“23 Historias de equipos Ágiles”, Javier Garzás, 2020

“SWITCH. How to change things when change is hard”, Chip Heath & Dan Heath

“La guía definitiva de Scrum: Las Reglas del juego” 2013. Ken Schwaber y Jeff Sutherland

“Beyond the Wall of Resistance: Why 70% of all changes still fail” 2010. Rick Mauer

WEBS

Escuela de negocios Fedá. (Consultado:21/07/2020). Gestión Ágil vs Gestión Tradicional de Proyectos. Recuperado de:

<https://www.escueladenegociosfedá.com/blog/50-la-huella-de-nuestros-docentes/471-gestion-agil-vs-gestion-tradicional-de-proyectos-como-elegir>

SCIO (Consultado: 21/07/2020). Metodología Tradicional o ágil ¿Cuál es la mejor opción? Recuperado de: <https://www.scio.com.mx/blog/metodologia-tradicional-o-agil-software/>

Ariad (Consultado 23/07/2020). Metodología Agile vs Waterfall. Recuperado de: <https://www.ariadgroup.com/es/blog/todo-sobre-Scrum-agile-metodologia/metodologia-agile-vs-waterfall>

Deloitte (Consultado 23/07/2020). ¿Cuál metodología es más adecuada para tu proyecto? Recuperado de: <https://www2.deloitte.com/es/es/pages/technology/articles/waterfall-vs-agile.html>

Learning Consortium (Consultado 24/07/2020). Recuperado de: <http://www.sdlearningconsortium.org/>

Javier Garzás (Consultado 25/07/2020). Desarrollo ágil o tradicional. ¿Existe el punto intermedio? Recuperado de: <https://www.javiergarzas.com/2010/08/agiles-formales-e-hibridos.html>

Javier Garzás (Consultado 27/07/2020). Aplícate el Shuhari. Recuperado de: <https://www.javiergarzas.com/2017/03/aplicate-el-shuhari.html>

Javier Garzás (Consultado 27/07/2020). Yo no sé cuál es la mejor manera ágil de trabajar. Recuperado de: <https://www.javiergarzas.com/2019/03/yo-no-se-cual-es-la-mejor-manera-agil-de-trabajar.html>

AgileScrum (Consultado 27/07/2020). ¿Qué es Shuhari? Recuperado de:

<https://www.agileScrum.cl/post/qu%C3%A9-es-shu-ha-ri>

Forbes (Consultado 29/07/2020). The 12 stages of the agile transformation journey.

Recuperado de: <https://www.forbes.com/sites/stevedenning/2018/11/04/the-twelve-stages-of-the-agile-transformation-journey/amp/>

Vitality Chicago (Consultado 29/07/2020). Agile Project Success Rates are 2X Higher

than Traditional Projects. Recuperado de: <http://vitalitychicago.com/blog/agile-projects-are-more-successful-traditional-projects>

Apiumhub (Consultado 29/07/2020). Transformación ágil: Pasos, estadísticas y un

ejemplo práctico. Recuperado de: <https://apiumhub.com/es/tech-blog-barcelona/transformacion-agil-pasos-estadisticas/>

McKinsey (Consultado 30/07/2020). El camino hacia una organización ágil.

Recuperado de: <https://www.mckinsey.com/business-functions/organization/our-insights/the-journey-to-an-agile-organization/es-cl#>

ESTUDIOS

Deloitte University Press. Rewriting the rules for the digital age (2017). Recuperado

de: <https://www2.deloitte.com/content/dam/Deloitte/lu/Documents/human-capital/lu-hc-2017-global-human-capital-trends-gx.pdf>

Hewlett Packard Enterprise (Consultado 03/08/2020). Agile is the new normal.

Recuperado de: <https://softwaretestinggenius.com/docs/4aa5-7619.pdf>

Collabnet VersionOne (2019). 13th annual state of agile report. Recuperado de:

https://stateofagile.com/?_ga=2.102143105.1272801324.1598357220-1659206522.1598357220

Standish Group. Chaos Report 2015 (2015). Recuperado de:

https://www.standishgroup.com/sample_research_files/CHAOSReport2015-Final.pdf

Miguel Ángel Almazán Gómez (2014). Estudio del efecto Fisher. Recuperado de:

<https://zagan.unizar.es/record/17139/files/TAZ-TFG-2014-2318.pdf>

Guía de Scrum: Reglas del juego (2013). Jeff Sutherland y Ken Schwaber. Recuperado

de: <https://www.Scrumguides.org/docs/Scrumguide/v1/Scrum-guide-es.pdf>

