

Para entender estos conceptos, primero he tenido que descubrir qué es una metodología de desarrollo ágil y cuáles son sus diferencias con las tradicionales.

Las metodologías tradicionales tienen un enfoque muy sistemático, ejecutándose el trabajo a partir de fundamentos bien documentados y normalizados. Este método es interesante para problemas bien definidos y conocidos, con soluciones ya estudiadas.

Las metodologías de desarrollo ágil surgen como contrapartida a esta filosofía y la idea fundamental es la optimización de los recursos humanos, en vez de hacer tanto énfasis en los aspectos más técnicos. Los pilares están basados en la interacción entre individuos (en vez de énfasis en procedimientos), contacto con los clientes (en vez de énfasis en contratos), adaptación al desarrollo (en vez de énfasis en la planificación previa) y en la aportación de soluciones adecuadas (en vez del énfasis en exceso de documentación). Esta forma de trabajar es ideal para problemas poco definidos y proyectos complejos con requisitos cambiantes.

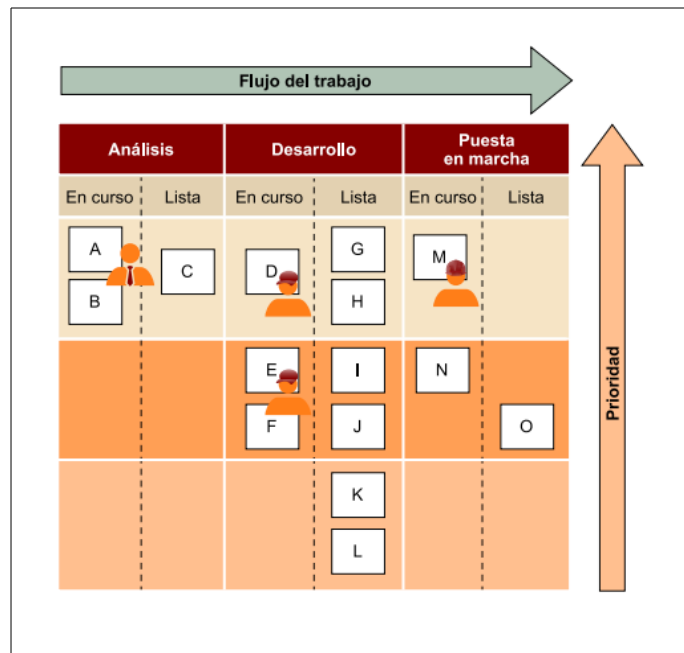
Una de estas metodologías es la **Scrum**. El fundamento de trabajo es definir bloques temporales cortos y fijos, denominados sprints. Estos sprints deben ser lo más pequeño posible (suelen variar de 1 a 4 semanas) y al final, deben resultar en un incremento real del software (es decir, que sea susceptible de ser entregado al cliente). Estos sprints se abordan iterativamente (Planificación previa → Sprint → Análisis).

La organización de estos trabajos en forma de sprint se fundamenta en la auto-organización de los equipos, mediante reuniones: al iniciar el proyecto, los desarrolladores necesitarán, por parte del cliente, una lista con los objetivos deseados y su prioridad. Por lo tanto, se establece el primer feedback cliente-equipo. Con estos objetivos, antes de cada sprint, se establece una reunión del equipo en la que se planificarán y asignarán las tareas de dicho sprint. Mientras se realiza el sprint, cada día se hacen reuniones cortas en las que los miembros del equipo exploran en común los progresos realizados y valoran el camino a seguir, con sus posibles dificultades. Esto tiene como fin inspeccionar lo que se está haciendo y permite su adaptación a los resultados que deben obtenerse si hay desviaciones. Finalmente, después de cada sprint, el equipo se pone en común para valorar el trabajo realizado, tanto a nivel de retrospectiva interna (analizando la forma de trabajar, dificultades, etc... para mejorar de forma continua), tanto como de demostración al cliente de los resultados obtenidos.

Para facilitar la labor organizativa, el personal se divide en 3 roles: el *product owner*, que representa al cliente y se encarga de adecuar los resultados a los deseados realmente a nivel profesional por este; el *Scrummaster*, que es el encargado de facilitar la auto organización del equipo, eliminando factores externos que puedan afectar a la productividad, y se asegura que se cumplen las reglas del proceso Scrum como es debido; el *equipo de desarrollo*, que son los encargados de realizar la parte técnica del proyecto (análisis, diseño, código...).

Otra metodología son los **paneles Kanban**, cuyo fundamento es la división del proceso en pequeñas partes perfectamente delimitadas, presentándolas de forma visual para mejorar el flujo de trabajo.

Estas representaciones agrupan las tareas según sus responsables, su prioridad y su estado. Esta representación y división permite conocer en todo momento el estado de cada parte del proyecto e identificar los puntos en los que se están produciendo más dificultades. Estos paneles no deben ser estáticos, sino que deben pensarse para cada equipo y proyecto y mejorarse de forma continua para ser siempre representativo de la realidad del proyecto y adecuarse a las necesidades específicas del cliente.



Con esta monitorización del proceso y sus necesidades, el método exige adecuar el esfuerzo destinado a las necesidades actuales de esa parte. Esto incluye tanto limitar el trabajo en las áreas que estén más avanzadas, como gestionar el flujo entre tareas.

Se insta a un trabajo en equipo, en los que las limitaciones se resuelvan trabajando de forma colaborativa y haciendo mejoras incrementales debido al autoanálisis. Es importante que las políticas sean claras y se cumplan de forma exigente, pero que el liderazgo no provenga de unos roles predeterminados, sino que sea un resultado de una auto-organización responsable de cada uno de los componentes del equipo.

## **Fuentes**

<https://proyectosagiles.org/que-es-scrum/>

<http://blogthinkbig.com/metodo-kanban/>

[https://www.exabyteinformatica.com/uoc/Audiovisual/Produccion\\_multimedia/Produccion\\_multimedia\\_\(Modulo\\_4\).pdf](https://www.exabyteinformatica.com/uoc/Audiovisual/Produccion_multimedia/Produccion_multimedia_(Modulo_4).pdf)

<http://agilemanifesto.org/>

<http://wikipedia.org>