



UTT

UNIVERSIDAD TECNOLÓGICA DE TIJUANA

GOBIERNO DE BAJA CALIFORNIA

PRESENTADO POR

Gael Breton Rendon

GRUPO

10° B

MATERIA

Desarrollo móvil integral

PROFESOR

Ray Parra

Tijuana, Baja California, 07 de enero del 2025

La **metodología ágil** es un enfoque para la gestión de proyectos de software que promueve la colaboración, la flexibilidad, la entrega continua de valor y la capacidad de adaptarse a los cambios. A lo largo de los años, se ha convertido en la metodología preferida por muchos equipos de desarrollo de software debido a su énfasis en la iteración, la mejora continua y el enfoque centrado en el cliente.

Cuando se selecciona una metodología ágil para un proyecto, es importante considerar varios factores, ya que no todas las metodologías ágiles son iguales y no todas se adaptan de la misma manera a todos los contextos de proyecto. La **selección de la metodología ágil** depende de las características específicas del proyecto, el equipo y los objetivos del cliente.

Conceptos Clave de las Metodologías Ágiles

Las metodologías ágiles comparten ciertos principios clave, que se derivan del **Manifiesto Ágil** de 2001, que establece un enfoque flexible y colaborativo para el desarrollo de software. Estos principios incluyen:

1. **Interacción y colaboración con el cliente:** Priorizar la satisfacción del cliente mediante la entrega continua de software funcional.
2. **Entrega continua de valor:** Trabajar en ciclos cortos (iteraciones) para producir versiones incrementales del producto.
3. **Adaptación al cambio:** Aceptar y adaptarse a los cambios incluso en fases avanzadas del proyecto.
4. **Desarrollo en equipo:** El trabajo conjunto y la comunicación constante entre los miembros del equipo son esenciales.
5. **Reflexión continua:** Realizar retrospectivas regularmente para identificar mejoras y optimizar los procesos.

Principales Metodologías Ágiles

1. **Scrum:**
 - **Descripción:** Scrum es una de las metodologías ágiles más populares y se basa en la estructura de trabajo en iteraciones conocidas como **sprints**. Cada sprint tiene una duración fija (usualmente de 2 a 4 semanas) y al final de cada sprint se entrega un incremento del producto.
 - **Características:** Roles específicos (Scrum Master, Product Owner, Equipo de Desarrollo), ceremonias (planificación del sprint, reuniones diarias, revisión del sprint, retrospectiva) y artefactos (product backlog, sprint backlog, incremento).
 - **Cuándo usarlo:** Ideal para proyectos con requisitos cambiantes o cuando se necesita una alta transparencia, colaboración constante y gestión del progreso en ciclos cortos.

2. Kanban:

- **Descripción:** Kanban es una metodología que se centra en la visualización del flujo de trabajo, utilizando un **tablero Kanban** para representar las tareas y su estado (por ejemplo, "Pendiente", "En Progreso", "Terminado").
- **Características:** No hay sprints definidos, y el enfoque está en la mejora continua del flujo de trabajo. Kanban utiliza el concepto de **limitar el trabajo en curso (WIP)** para evitar cuellos de botella.
- **Cuándo usarlo:** Ideal para equipos con procesos de trabajo continuos o que no necesitan la estructura rígida de Scrum, y para entornos donde las prioridades cambian constantemente.

3. Extreme Programming (XP):

- **Descripción:** XP es una metodología ágil que pone énfasis en la calidad técnica y en las mejores prácticas de desarrollo, como la **programación en pareja** (pair programming), la **integración continua**, las **pruebas automatizadas** y la **refactorización continua**.
- **Características:** Se enfoca en una colaboración más estrecha entre desarrolladores y clientes, así como en un ciclo de retroalimentación continuo. XP promueve ciclos de desarrollo más rápidos con un enfoque fuerte en las pruebas de calidad.
- **Cuándo usarlo:** Ideal para proyectos que requieren un alto nivel de calidad técnica, donde el código es complejo y es esencial la retroalimentación constante.

4. Lean Software Development:

- **Descripción:** Basada en los principios del Lean Manufacturing, esta metodología busca eliminar el **desperdicio**, mejorar la eficiencia y maximizar el valor para el cliente con la menor cantidad de trabajo posible.
- **Características:** Lean se enfoca en la entrega continua, la eliminación de ineficiencias, la optimización de recursos y la toma de decisiones rápidas.
- **Cuándo usarlo:** Ideal para proyectos con restricciones de recursos o que buscan mejorar el rendimiento en un entorno competitivo.

5. Feature-Driven Development (FDD):

- **Descripción:** FDD es una metodología ágil centrada en la entrega de características funcionales de manera iterativa y predecible.
- **Características:** Se organiza alrededor de la creación de una **lista de características** que se desarrollan en iteraciones. Cada iteración tiene un conjunto de características definidas para ser completadas.

- **Cuándo usarlo:** Ideal para proyectos más grandes que necesitan un enfoque estructurado, especialmente cuando se trata de una gran cantidad de características con un enfoque en la funcionalidad del producto.

6. Crystal:

- **Descripción:** Crystal es una familia de metodologías ágiles que abogan por un enfoque más flexible dependiendo de las características del equipo y el proyecto. Algunas de las variantes de Crystal incluyen Crystal Clear, Crystal Yellow, y Crystal Orange.
- **Características:** Se centra en la importancia de la comunicación dentro del equipo, y la metodología varía en función del tamaño del equipo, la complejidad del proyecto y los riesgos.
- **Cuándo usarlo:** Ideal para equipos pequeños o proyectos con menos complejidad técnica.