**METODOLOGÍAS ÁGILES**

Principios y valores:

* Realizar entregas rápidas y continuas (si hay partes grandes, se trocean para permitir esto)
* Equipos multidisciplinares (“scrums”) que trabajen juntos. Estos “scrums” se informan entre sí sobre sus avances en el proyecto
* Ver *Manifiesto*
* Herramientas o metodologías englobadas en el concepto *Agile:* **Agile Modeling, Agile Unified Process, Crystal Clear, Extreme Programming, Scrum.**..

**Manifiesto:**

SCRUM – *Gestión ágil* del desarrollo de un **producto** de software

Si va unido a *desarrollo ágil,* mucho mejor.

**PASOS:**

1 – Describe utilizando tarjetas las *personas (roles)* a las que va dirigido el producto. Ponles nombre, apellidos, cara, a qué se dedican, qué aficiones tienen, si tienen familia... Las tarjetas han de estar visibles para el equipo de desarrollo.

2 – De quién, para qué y cómo. Escribe “historias de usuario”. Luego realiza pruebas de aceptación (validación con potenciales usuarios reales, supongo).

3 – Reunir un equipo de desarrolladores técnicamente cualificado, entre 3 y 10, que sepan de todo, utilicen el *desarrollo ágil* y se lleven bien. (seguir)

4 – MVP (Mínimo Producto Viable). Hay que ir a lo sencillo. La idea de “terminado” es presentar al usuario un producto que ya pueda utilizar.

5 – Sprints

6 – Daily meetings. Revisar cada día el panel de Scrum, con todos los desarrolladores, todas las tarjetas asignadas. No más de diez minutos.

7 – Validación del trabajo.

8 – Mejora contínua

GLOSARIO DE TÉRMINOS Y CONCEPTOS

gestión ágil:

desarrollo ágil:

elevator pitch: un resumen que explique tu proyecto a un posible inversor durante un trayecto en ascensor ;P

MVP: PMV – Producto Mínimo Viable

sprint:

planning poker:

**INTEGRACIÓN CONTINUA**

Se suele relacionar con la XP (eXtreme Programming) y el desarrollo ágil.

Puntos importantes:

Se trata de evitar que, en el momento de integrar y desplegar un proyecto, surjan errores de última hora; esto se logra, como el propio nombre indica, realizando la integración continua, es decir, un proceso automatizado que, después de que cada desarrollador suba el código al repositorio, obtenga la última versión del proyecto, lo compile, ejecute el conjunto de pruebas unitarias, y si todo ha ido bien, deje los binarios en una ubicación conocida (despliegue). Este proceso debe realizarse lo más frecuentemente posible; así, con cada cambio que se suba al repositorio se detectarán los problemas de integración y se harán las pruebas de todo el proyecto.

Es importante también la “entrega de valor”: en cualquier momento se tendrá disponible un binario compilado y con las pruebas pasadas, es decir, funcionando, aunque sea parcialmente.

COSAS SOBRE LAS QUE HAY QUE INFORMARSE

* Desarrollo ágil
* Integración continua
* Agile
* Kanboard
* Redmine vs. Jira (administración de tareas, seguimiento de errores e incidencias, gestión operativa de proyectos)