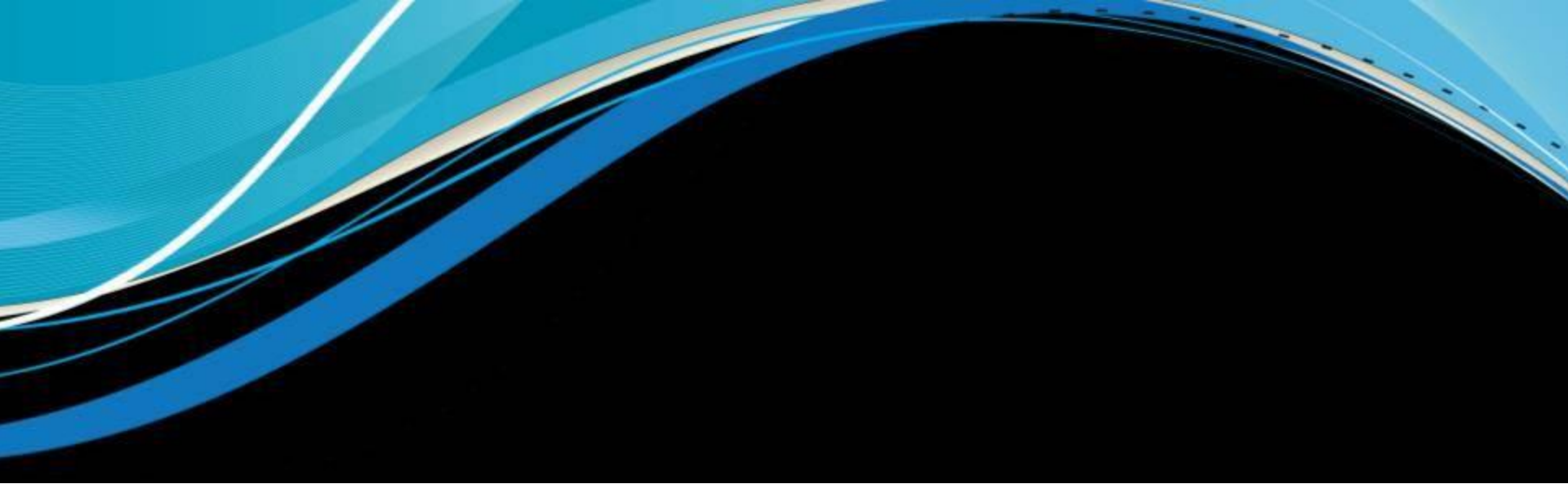


4.2 Introducción a las metodologías ágiles de desarrollo de SW



Indice

1. Introducción
2. Manifiesto ágil
3. Principios del manifiesto ágil
4. Comparativa
5. Modelo ágil o prescriptivo
6. Ciclo de vida



Introducción

Introducción

- **Desarrollo de SW** → Metodologías
- **Metodologías tradicionales** o pesadas
 - Orientados a la documentación
 - Muy prescriptivas
- **Metodologías ágiles** o ligeras
 - Enfoque iterativo e incremental
 - Adaptativas

Introducción

- No existe una metodología “universal” o “ideal”
 - Los métodos OO son adecuados para sistemas interactivos, pero no para sistemas en tiempo real con requisitos severos. (Sommerville 2002)
- La metodología depende de:
 - El tamaño y estructura de la organización
 - El tipo de aplicación

Introducción

- En los 80 y principios de los 90:
 - Cuidadosa planificación del proyecto
 - Formalizar y asegurar la calidad
 - Métodos de análisis y diseño soportados por herramientas CASE
 - Proceso de desarrollo controlado y riguroso
- Desarrollo de sistemas SW grandes y de larga vida
 - Equipo de desarrollo disperso
 - Empresas distintas
 - Sobrecarga en planificación, análisis, diseño y documentación del sistema

Introducción

- Problemas en aplicaciones de tamaño medio o pequeño usando metodologías pesadas:
 - La sobrecarga de trabajo (requisitos, diseño, documentación) domina el proceso de desarrollo
 - Se invierte más tiempo a cómo se va a desarrollar el sistema que a la programación y las pruebas
 - Cada vez que cambia un requisito hay que hacer mucho trabajo de rediseño y redocumentación
- Resultado:
 - Cliente insatisfecho y equipo de desarrollo frustrado

Introducción

- A finales de los 90 aparecen las metodologías ágiles:
 - Atención al SW más que al diseño y la documentación
 - Iterativos e incrementales
 - Menos restrictivos
- Utilidad:
 - Aplicaciones de tamaño pequeño o medio con requisitos que cambian rápidamente durante el proceso de desarrollo

Manifiesto ágil

**Individuos e
interacciones**

sobre procesos y
herramientas

**Software
funcionando**

sobre documentación
extensiva

**Colaboración con el
cliente**

sobre negociación
contractual

**Respuesta ante el
cambio**

sobre seguir un plan

Principios del manifiesto ágil

- Nuestra mayor prioridad es **satisfacer al cliente** mediante la **entrega** temprana y continua de **software con valor**
- Aceptamos que los **requisitos cambien**, incluso en etapas tardías del desarrollo. Los procesos Ágiles aprovechan el cambio para proporcionar ventaja competitiva al cliente
- Entregamos **software funcional frecuentemente**, entre dos semanas y dos meses, con preferencia al periodo de tiempo más corto posible

Principios del manifiesto ágil

- Los responsables de negocio y los desarrolladores **trabajamos juntos** de forma cotidiana durante todo el proyecto
- Los proyectos se desarrollan en torno a **individuos motivados**. Hay que darles el entorno y el apoyo que necesitan, y confiarles la ejecución del trabajo
- El método más eficiente y efectivo de comunicar información al equipo de desarrollo y entre sus miembros es la **conversación cara a cara**

Principios del manifiesto ágil

- El **software funcionando** es la medida principal de progreso
- Los procesos ágiles promueven el desarrollo sostenible. Los promotores, desarrolladores y usuarios debemos ser capaces de mantener un **ritmo constante** de forma indefinida
- La atención continua a la **excelencia técnica** y al **buen diseño** mejora la agilidad
- La **simplicidad**, o el arte de maximizar la cantidad de trabajo no realizado, es esencial

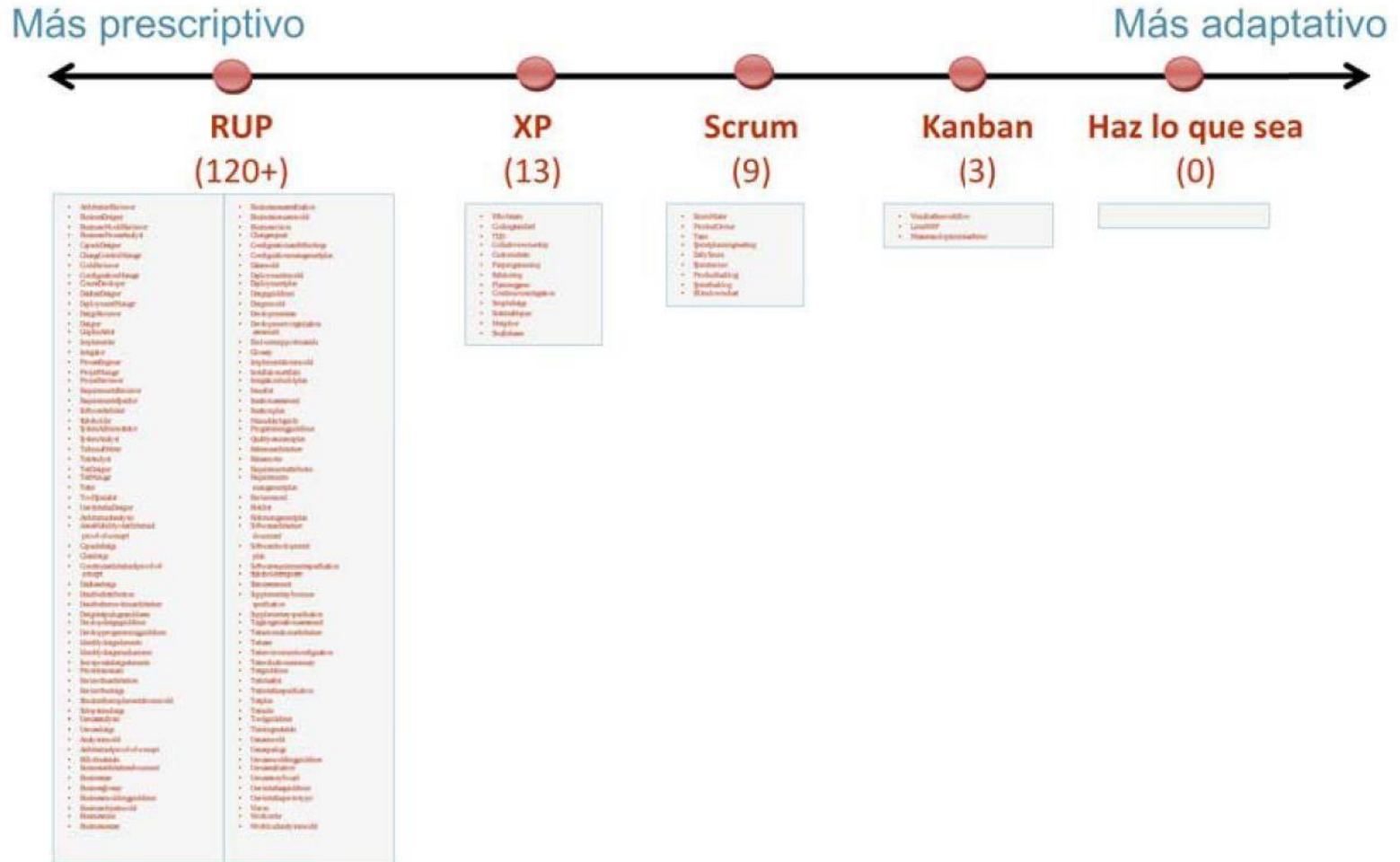
Principios del manifiesto ágil

- Las mejores arquitecturas, requisitos y diseños emergen de **equipos auto-organizados**
- A intervalos regulares el equipo **reflexiona** sobre **cómo ser más efectivo** para a continuación ajustar y perfeccionar su comportamiento en consecuencia

Firmantes de los principios y el manifiesto ágil:

- Kent Beck, Mike Beedle, Arie van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, James Grenning, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C. Martin, Steve Mellor, Ken Schwaber, Jeff Sutherland y Dave Thomas

Comparativa



Modelo ágil o prescriptivo

- Según las características de un proyecto:

	Ágil	Prescriptivo
Prioridad del negocio	Valor	Cumplimiento
Estabilidad de requisitos	Entorno inestable	Entorno estable
Rigidez del producto	Modificable	Difícil de modificar
Coste del prototipado	Bajo	Alto
Criticidad del sistema	Baja	Alta
Tamaño del equipo	Pequeño	Grande

- Según las condiciones de la organización:

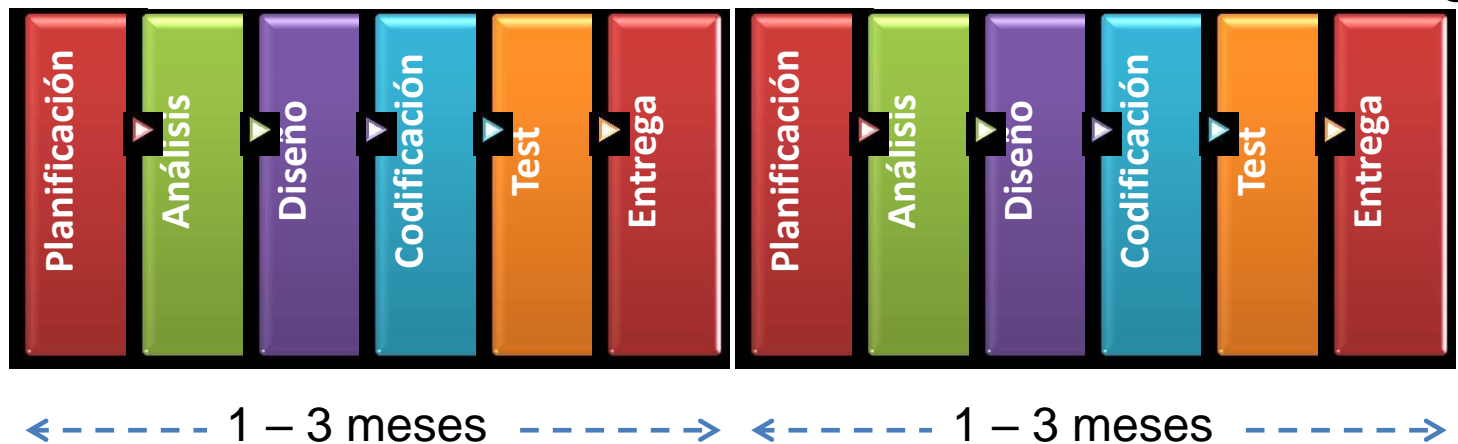
	Ágil	Prescriptivo
Nivel profesional	Senior	Junior
Cultura organizativa	Horizontal, flexible	Vertical, rígida
Modelo de desarrollo	Personas	Procesos

Ciclo de vida

Ciclo de vida en cascada



Ciclo de vida iterativo



€: Produce valor