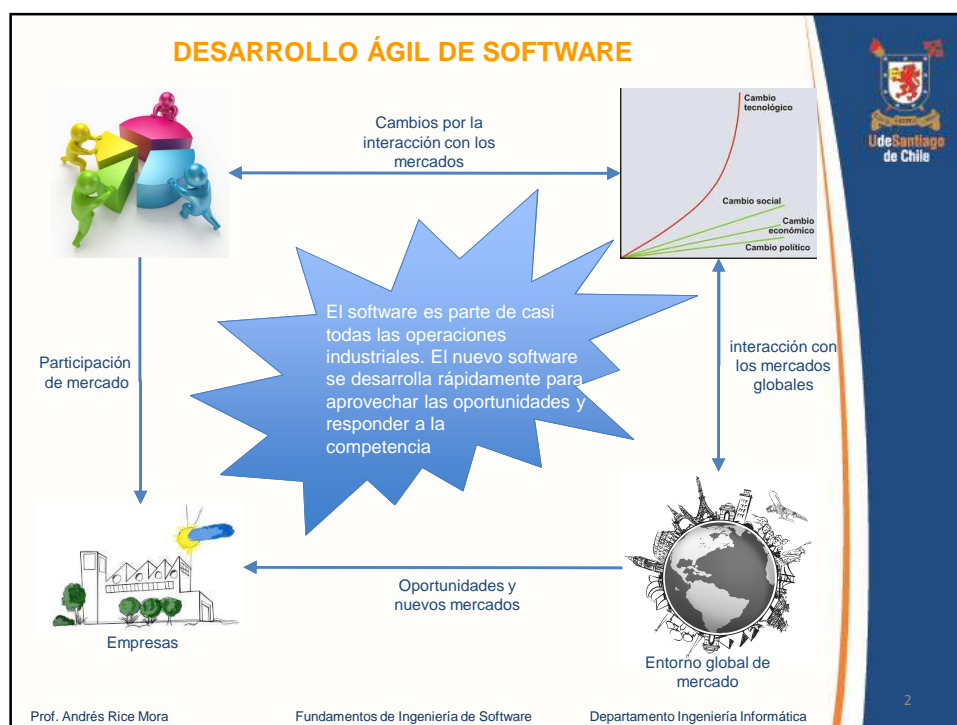


2.3 DESARROLLO ÁGIL DE SOFTWARE

Prof. Andrés Rice Mora Fundamentos de Ingeniería de Software Departamento Ingeniería Informática



DESARROLLO ÁGIL DE SOFTWARE

- En la actualidad, **la entrega rápida del SW es el requerimiento fundamental**
- Las empresas están dispuestas a **negociar la calidad del software y el compromiso con los requerimientos**, siempre que se logre mayor celeridad en la implementación del software
- Debido a que las industrias funcionan en un **entorno cambiante**, es **prácticamente imposible derivar un conjunto completo de requerimientos** estable de software
- **Es probable que debido a factores externos, los requerimientos cambien rápida e impredeciblemente.**
- Los procesos de desarrollo de software que buscan especificar **por completo los requerimientos** y, luego, diseñar, construir y probar el sistema, **no están orientados al desarrollo rápido de software**



DESARROLLO ÁGIL DE SOFTWARE

- Durante algún tiempo, se reconoció la necesidad de desarrollo y de procesos de sistema rápidos que administraran los requerimientos cambiantes
- En la década de 1980 (Mills *et al.*, 1980) **se introdujo el desarrollo incremental; lenguajes de cuarta generación** y la idea del **software de desarrollo y entregas rápidas** (Martin, 1981)
- La idea prosperó más a finales de la década de 1990, **con el desarrollo de enfoques ágiles** como el **DSDM** (Stapleton, 1997), **Scrum** (Schwaber y Beedle, 2001) y **la programación extrema** (Beck, 1999; Beck, 2000)
- Los procesos de desarrollo del software rápido se diseñan para **producir rápidamente un software útil**
- El software **no se desarrolla como una sola unidad, sino como una serie de incrementos**, y cada uno de ellos **incluye una nueva funcionalidad** del sistema



DESARROLLO ÁGIL DE SOFTWARE

Los desarrollo de software rápido comparten algunas características fundamentales:

1. **Los procesos de especificación, diseño e implementación están entrelazados.** No existe una especificación detallada del sistema, y la documentación del diseño se minimiza o es generada automáticamente por el entorno de programación que se usa para implementar el sistema. **El documento de requerimientos del usuario define sólo las características más importantes del sistema**
2. **El sistema se desarrolla en diferentes versiones.** Los usuarios finales y otros colaboradores del sistema intervienen en la especificación y evaluación de cada versión. Ellos podrían proponer cambios al software y nuevos requerimientos que se implementen en una versión posterior del sistema.



DESARROLLO ÁGIL DE SOFTWARE

3. **Las interfaces de usuario se desarrolla usando un sistema de elaboración interactivo**, que permita que el diseño de la interfaz se cree rápidamente en cuanto se dibujan y colocan iconos en la interfaz. En tal situación, el sistema puede generar una interfaz basada en la Web para un navegador o una interfaz para una plataforma específica.
4. **Los métodos ágiles son métodos de desarrollo incremental** donde los incrementos demoran dos a tres semanas en liberarse para los clientes.
5. **Involucran a los clientes en el proceso** de desarrollo para conseguir una rápida retroalimentación sobre los requerimientos cambiantes
6. **Minimizan la cantidad de documentación con el uso de comunicaciones informales**, en vez de reuniones formales con documentos escritos



DESARROLLO ÁGIL DE SOFTWARE

La filosofía detrás de los métodos ágiles se refleja en el manifiesto ágil, que acordaron muchos de los desarrolladores líderes de estos métodos.

Este manifiesto afirma:

Estamos descubriendo mejores formas para desarrollar software, al hacerlo y al ayudar a otros a hacerlo. Gracias a este trabajo llegamos a valorar:

- *A los individuos y las interacciones sobre los procesos y las herramientas*
- *Al software operativo sobre la documentación exhaustiva*
- *La colaboración con el cliente sobre la negociación del contrato*
- *La respuesta al cambio sobre el seguimiento de un plan*

Esto es, aunque exista valor en los objetos a la derecha, valoraremos más los de la izquierda.

DESARROLLO ÁGIL DE SOFTWARE

Algunos métodos ágiles conocidos:

- Programación extrema (Beck, 1999; Beck, 2000)
- Scrum (Cohn, 2009; Schwaber, 2004; Schwaber y Beedle, 2001)
- Crystal (Cockburn, 2001; Cockburn, 2004)
- Desarrollo de software adaptativo (Highsmith, 2000)
- Dynamic systems development method (DSDM) (Stapleton, 1997; Stapleton, 2003)
- Desarrollo dirigido por características (Palmer y Felsing, 2002)
- Al integrar estos métodos ágiles con métodos tradicionales resulta los modelos ágiles (Ambler y Jeffries, 2002) y ejemplificaciones ágiles del Proceso Racional Unificado (Larman, 2002)

DESARROLLO ÁGIL DE SOFTWARE

Principios compartido por los métodos ágiles:

Principio	Descripción
Participación del cliente	Los clientes deben intervenir estrechamente durante el proceso de desarrollo. Su función consiste en ofrecer y priorizar nuevos requerimientos del sistema y evaluar las iteraciones del mismo.
Entrega incremental	El software se desarrolla en incrementos y el cliente especifica los requerimientos que se van a incluir en cada incremento.
Personas, no procesos	Tienen que reconocerse y aprovecharse las habilidades del equipo de desarrollo. Debe permitirse a los miembros del equipo desarrollar sus propias formas de trabajar sin procesos establecidos.
Adoptar el cambio	Esperar a que cambien los requerimientos del sistema y, de este modo, diseñar el sistema para adaptar dichos cambios.
Mantener simplicidad	Enfocarse en la simplicidad tanto en el software a desarrollar como en el proceso de desarrollo. Siempre que sea posible, trabajar de manera activa para eliminar la complejidad del sistema.

DESARROLLO ÁGIL DE SOFTWARE

Para cumplir con los principios se requiere :

- 1. Clientes que deseen y puedan pasar tiempo con el equipo de desarrollo,** y represente a todos los participantes del sistema
- 2. Miembros del equipo con personalidad adecuada** para interactuar con otros integrantes del equipo
- 3. Capacidad para priorizar cambios,** sobre todo cuando cada participante ofrece diversas prioridades a los cambios
- 4. Mantener la simplicidad requerida, bajo la presión de fechas de entrega**
- En muchas organizaciones, resulta **difícil moverse hacia un modelo de trabajo donde los procesos sean informales** y estén definidos por equipos de desarrollo

DESARROLLO ÁGIL vs PLAN

Es posible, para un proyecto de software, incluir prácticas de enfoques ágil y basado en un plan.

Para decidir sobre el equilibrio entre un enfoque basado en un plan y uno ágil, se deben responder algunas preguntas técnicas, humanas y organizacionales:

1. ¿Es importante tener una especificación y un diseño muy detallados antes de dirigirse a la implementación?

Si su respuesta es sí, probablemente usted tenga que usar un enfoque basado en un plan.

2. ¿Es práctica una estrategia de entrega incremental, donde se dé el software a los clientes y se obtenga así una rápida retroalimentación de ellos?

De ser el caso, considere el uso de métodos ágiles o un modelo incremental por plan

DESARROLLO ÁGIL vs PLAN

3. ¿Qué tan grande es el sistema que se desarrollará?

Cuando el sistema logra diseñarse con un pequeño equipo asignado que se comunique de manera informal. Esto sería imposible para los grandes sistemas que precisan equipos de desarrollo más amplios, de manera que tal vez se utilice un enfoque basado en un plan.

4. ¿Qué tipo de sistema se desarrollará?

Los sistemas que demandan mucho análisis antes de la implementación (por ejemplo, sistema en tiempo real con requerimientos de temporización compleja), por lo general, necesitan un diseño bastante detallado para realizar este análisis. En tales circunstancias, quizá sea mejor un enfoque basado en un plan.

DESARROLLO ÁGIL vs PLAN**5. ¿Cuál es el tiempo de vida que se espera del sistema?**

Los sistemas con lapsos de vida prolongados podrían requerir más documentación de diseño, para comunicar al equipo de apoyo los propósitos originales de los desarrolladores del sistema. Sin embargo, los defensores de los métodos ágiles argumentan acertadamente que con frecuencia la documentación no se conserva actualizada, ni se usa mucho para el mantenimiento del sistema a largo plazo.

6. ¿Qué tecnologías se hallan disponibles para apoyar el desarrollo del sistema?

Los métodos ágiles se auxilian a menudo de buenas herramientas para seguir la pista de un diseño en evolución. Si se desarrolla un sistema con un IDE sin contar con buenas herramientas para visualización y análisis de programas, entonces posiblemente se requiera más documentación de diseño.

DESARROLLO ÁGIL vs PLAN**7. ¿Cómo está organizado el equipo de desarrollo?**

Si el equipo de desarrollo está distribuido, o si parte del desarrollo se subcontrata, entonces tal vez se requiera elaborar documentos de diseño para comunicarse a través de los equipos de desarrollo. Quizá se necesite planear por adelantado cuáles son.

8. ¿Existen problemas culturales que afecten el desarrollo del sistema?

Las organizaciones de ingeniería tradicionales presentan una cultura de desarrollo basada en un plan, pues es una norma en ingeniería. Esto requiere comúnmente una amplia documentación de diseño, en vez del conocimiento informal que se utiliza en los procesos ágiles.

DESARROLLO ÁGIL vs PLAN

9. ¿Qué tan buenos son los diseñadores y programadores en el equipo de desarrollo?

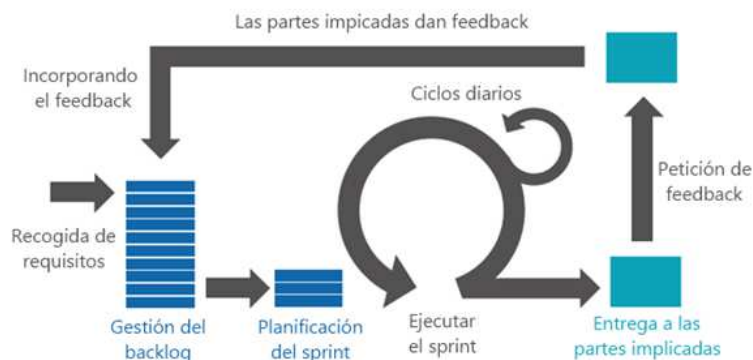
Se argumenta en ocasiones que los métodos ágiles requieren niveles de habilidad superiores a los enfoques basados en un plan, en que los programadores simplemente traducen un diseño detallado en un código. Si usted tiene un equipo con niveles de habilidad relativamente bajos, es probable que necesite del mejor personal para desarrollar el diseño, siendo otros los responsables de la programación.

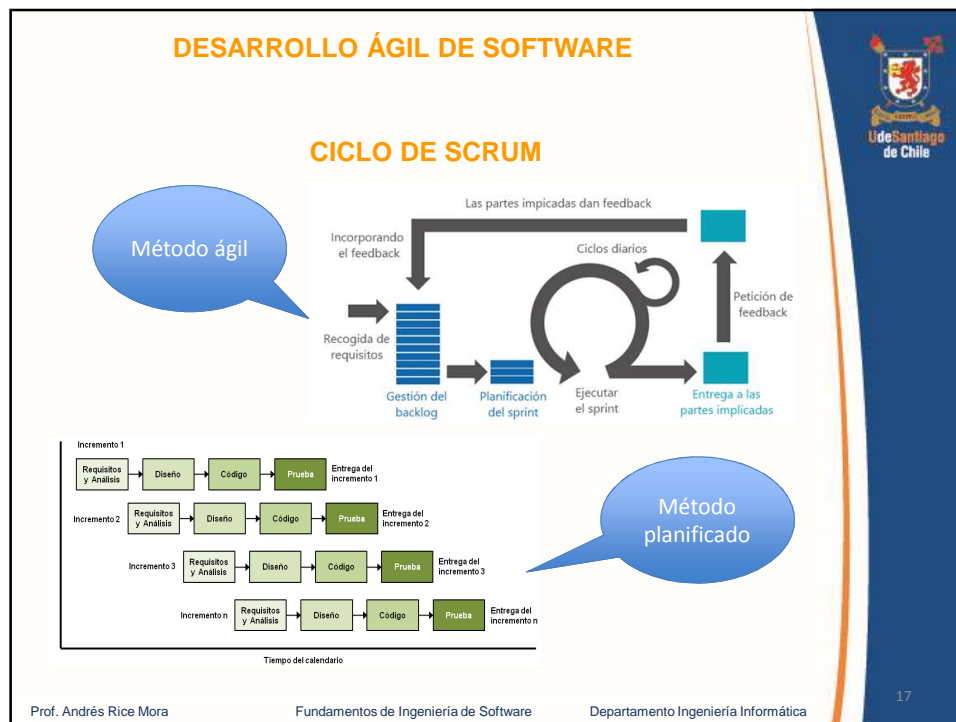
10. ¿El sistema está sujeto a regulación externa?

Si un regulador externo tiene que aprobar el sistema (por ejemplo, la Agencia de Aviación Federal [FAA] aprueba el software que es crítico para la operación de una aeronave), entonces, tal vez se le requerirá documentación detallada como parte del sistema de seguridad.

DESARROLLO ÁGIL DE SOFTWARE

CICLO DE SCRUM





DESARROLLO ÁGIL DE SOFTWARE

PROGRAMACION EXTREMA (XP)

ACTIVIDAD DE AUTOESTUDIO QUE DEBE DESARROLLAR EL ESTUDIANTE (PÁG. 64 Y SIGUIENTES DEL LIBRO ESTUDIO)

UdeSantiago de Chile


18

Prof. Andrés Rice Mora Fundamentos de Ingeniería de Software Departamento Ingeniería Informática

DESARROLLO ÁGIL DE SOFTWARE

METODOLOGÍAS ÁGILES

- Agile Database Techniques (AD)
- Agile Modeling (AM)
- Adaptive Software Development (ASD)
- Agile Unified Process (UAP)
- Crystal Clear
- Feature Driven Development (FDD)
- Dynamic Systems Development Method (DSDM)
- Lean Development
- Scrum
- Test-Driven Design (TDD)
- Xbreed
- Extreme Programming (XP)
- Rapid Development (RAD)
- otras



19

Prof. Andrés Rice Mora

Fundamentos de Ingeniería de Software

Departamento Ingeniería Informática