



Técnico en
< DESARROLLO DE SOFTWARE >

***Metodología de Desarrollo de
Software I***

(CC BY-NC-ND 4.0)
International

Attribution-NonCommercial-NoDerivatives 4.0



Atribución

Usted debe reconocer el crédito de una obra de manera adecuada, proporcionar un enlace a la licencia, e indicar si se han realizado cambios. Puede hacerlo en cualquier forma razonable, pero no de forma tal que sugiera que tiene el apoyo del licenciante o lo recibe por el uso que hace.



No Comercial

Usted no puede hacer uso del material con fines comerciales.



Sin obra derivada

Si usted mezcla, transforma o crea un nuevo material a partir de esta obra, no puede distribuir el material modificado.

No hay restricciones adicionales - Usted no puede aplicar términos legales ni medidas tecnológicas que restrinjan legalmente a otros hacer cualquier uso permitido por la licencia.

<http://creativecommons.org/licenses/by-nc-nd/4.0/>



Metodología de Desarrollo de Software I

Unidad IV

Metodologías ágiles de desarrollo de software

1. Metodología Ágil

Método que permite incorporar cambios con rapidez en el desarrollo de software. En muchas ocasiones, los modelos de gestión tradicionales no sirven para afrontar un reto que hoy en día resulta fundamental, incorporar cambios con rapidez y en cualquier fase del proyecto.

Historia

En febrero de 2001, tras una reunión celebrada en Utah-EEUU, nace el término “ágil” aplicado al desarrollo de software. En esta reunión participan un grupo de 17 expertos de la industria del software, incluyendo algunos de los creadores o impulsores de metodologías de software. Su objetivo fue esbozar los valores y principios que deberían permitir a los equipos desarrollar software rápidamente y respondiendo a los cambios que puedan surgir a lo largo del proyecto.

Se pretendía ofrecer una alternativa a los procesos de desarrollo de software tradicionales, caracterizados por ser rígidos y dirigidos por la documentación que se genera en cada una de las actividades desarrolladas.

Tras esta reunión se creó The Agile Alliance³, una organización, sin ánimo de lucro, dedicada a promover los conceptos relacionados con el desarrollo ágil de software y ayudar a las organizaciones para que adopten dichos conceptos. El punto de partida fue el Manifiesto Ágil, un documento que resume la filosofía “ágil”.

Manifiesto Ágil

Según el Manifiesto se valora:

- **Al individuo y las interacciones del equipo** de desarrollo sobre el proceso y las herramientas. La gente es el principal factor de éxito de un proyecto software. Es más importante construir un buen equipo que construir el entorno. Muchas veces se comete el error de construir primero el entorno y esperar que el equipo se adapte automáticamente. Es mejor crear el equipo y que éste configure su propio entorno de desarrollo en base a sus necesidades.
- Desarrollar **software que funciona** más que conseguir una buena documentación. La regla a seguir es “no producir documentos a menos que sean necesarios de forma inmediata para tomar una decisión importante”. Estos documentos deben ser cortos y centrarse en lo fundamental.
- La **colaboración con el cliente** más que la negociación de un contrato. Se propone que exista una interacción constante entre el cliente y el equipo de desarrollo. Esta colaboración entre ambos será la que marque la marcha del proyecto y asegure su éxito.

- **Responder a los cambios** más que seguir estrictamente un plan. La habilidad de responder a los cambios que puedan surgir a lo largo del proyecto (cambios en los requisitos, en la tecnología, en el equipo, etc.) determina también el éxito o fracaso del mismo. Por lo tanto, la planificación no debe ser estricta sino flexible y abierta.

Algunas Metodologías ágiles

Algunas metodologías ágiles de desarrollo de software:

- Adaptive Software Development (ASD).
- Agile Unified Process(AUP).
- Crystal Clear.
- Essential Unified Process (EssUP).
- Feature Driven Development (FDD).
- Lean Software Development (LSD).
- Kanban.
- Open Unified Process (OpenUP).
- Programación Extrema (XP).
- Método de desarrollo de sistemas dinámicos (DSDM).
- SCRUM.

Ejemplo de aplicación de metodologías ágiles

Se trata de evitar lo que tantas veces ha ocurrido:

Cuando el proyecto se encuentra bastante avanzado y no va por el buen camino o, simplemente, el cliente decide introducir cambios sustanciales, y esos cambios obligan

a desechar todo el trabajo realizado hasta entonces, lo que impide acabar en el plazo previsto. Dado que los cambios nunca van a dejar de existir, es necesario ser capaces de gestionar los proyectos de una forma más ágil.

Con ese objetivo, en los años 80 los japoneses Takeuchi y Nonaka estudiaron las prácticas de empresas con buenos resultados de rapidez y flexibilidad en la producción: Xerox, Canon, Honda, NEC, Epson, Brother, 3M y Hewlett-Packard. De ahí extrajeron la base de la metodología ágil SCRUM que, aunque nació en el ámbito tecnológico, ha ido creciendo hasta consolidarse en campos de actividad muy diferentes.

2. Metodologías Tradicionales

Teniendo en cuenta la filosofía de desarrollo de las metodologías, aquellas con mayor énfasis en la planificación y control del proyecto, en especificación precisa de requisitos y modelado, reciben el apelativo de Metodologías Tradicionales o Pesadas.

Estas metodologías tradicionales imponen una disciplina de trabajo sobre el proceso de desarrollo del software, con el fin de conseguir un software más eficiente. Para ello, se hace énfasis en la planificación total de todo el trabajo a realizar y una vez que está todo detallado, comienza el ciclo de desarrollo del producto software. Se centran especialmente en el control del proceso, mediante una rigurosa definición de roles, actividades, artefactos, herramientas y notaciones para el modelado y documentación detallada. Además, las metodologías tradicionales no se adaptan adecuadamente a los cambios, por lo que no son métodos adecuados cuando se trabaja en un entorno, donde los requisitos no pueden predecirse o bien pueden variar.

Entre las metodologías tradicionales o pesadas podemos citar:

- RUP (Rational Unified Procces)
- MSF (Microsoft Solution Framework)
- Win-Win Spiral Model
- Iconix

3. Metodologías de Desarrollo Ágiles Vs. Metodologías Tradicionales

Principales diferencias entre Metodologías de Desarrollo Ágiles Vs. Metodologías Tradicionales:

Para el desarrollo de software, se requiere de diversos elementos que agrupados hacen que el desarrollo de software sea o no exitoso.

Para esto existen las metodologías tradicionales que se modificaron para poder aplicarlas al desarrollo de software, aunque durante mucho tiempo fueron la única solución al desarrollo, hizo estas metodologías poco flexibles y muy cuadrículadas. Estas consistían en una serie fundamentos y conceptos aplicados al desarrollo de software, documentación, planificación y procesos. (Plantillas, técnicas de administración, revisiones, entre otras.)

Ante las dificultades de las metodologías tradicionales referentes al tiempo y flexibilidad, aparecen las metodologías ágiles como una respuesta metodológica, especialmente porque están orientadas a proyectos pequeños, constituyen una solución a la medida del entorno, simplificando las prácticas y asegurando la calidad del producto.

El término ágil, nace en febrero de 2001 en una reunión en Utah (EEUU), aplicado al desarrollo de software, su objetivo era idear los valores y principios que permitirían a los equipos de desarrollo crear software rápidamente, respondiendo a los cambios surgidos a lo largo del proyecto, ofreciendo una alternativa a los procesos tradicionales. En dicha reunión se creó The Agile Alliance, organización, sin ánimo de lucro, que promueve los conceptos relacionados con el desarrollo ágil de software, apoyando las organizaciones para que adopten dichos conceptos.

Según el Manifiesto ágil, se valora al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas, desarrollar software que funciona (más que conseguir una buena documentación), la colaboración con el cliente (más que la negociación mediante contrato), respuesta a los cambios (más que seguir rigurosamente algún plan).

Cuadro comparativo: Metodologías tradicionales y ágiles:

Metodologías Tradicionales	Metodologías Ágiles
<ul style="list-style-type: none">- Rigidez ante los cambios, de manera lenta o moderada.- Los clientes interactúan con el equipo de desarrollo mediante reuniones.- Grupos de gran tamaño y varias veces distribuidos en diferentes sitios.	<ul style="list-style-type: none">- Flexibilidad ante los cambios del proyecto de forma moderada a rápida.- Los clientes hacen parte del equipo de desarrollo.- Grupos pequeños (promedio 10 participantes in situ) en el mismo lugar.

<ul style="list-style-type: none"> - Dependencia de la arquitectura de software mediante modelos. - Poco Feedback lo que extiende el tiempo de entrega. - Mínimos roles. - Basadas en normas de estándares de desarrollo. - Procesos muy controlados por políticas y normas. - Seguimiento estricto del plan inicial de desarrollo. 	<ul style="list-style-type: none"> - Menor dependencia de la arquitectura de software - Continuo Feedback acortando el tiempo de entrega. - Diversidad de roles. - Basadas en heurísticas a partir de prácticas de producción de código. - Procesos menos controlados, pocas políticas y normas. - Capacidad de respuesta ante los cambios
---	--

Referencias

Sitio: leansoftwaredevelopment

- <https://leansoftwaredevelopment.wordpress.com/category/desarrollo-agil/>

Sitio: Ecured

- http://www.ecured.cu/index.php/Metodolog%C3%ADas_%C3%81giles
- http://www.ecured.cu/index.php/Metodolog%C3%ADa_%C3%A1gil
- http://www.ecured.cu/index.php/Metodolog%C3%ADas_Tradicionales

Sitio: Concierto

- <http://www.concierto.com/articulos/archivos/2007/02/desarrollo-de-software-adaptable/>

Sitio: rdsoporteymantenimientodepc

- <http://rdsoporteymantenimientodepc.blogspot.com/2014/03/metodologias-de-desarrollo-agiles-vs.html>

Sitio: eumed

- <http://www.eumed.net/libros-gratis/2009c/584/Metodologias%20tradicionales%20y%20metodologias%20agiles.htm>

Descargo de responsabilidad

La información contenida en este documento descargable en formato PDF o PPT es un reflejo del material virtual presentado en la versión online del curso. Por lo tanto, su contenido, gráficos, links de consulta, acotaciones y comentarios son responsabilidad exclusiva de su(s) respectivo(s) autor(es) por lo que su contenido no compromete al área de e-Learning del Departamento GES o al programa académico al que pertenece.

El área de e-Learning no asume ninguna responsabilidad por la actualidad, exactitud, obligaciones de derechos de autor, integridad o calidad de los contenidos proporcionados y se aclara que la utilización de este descargable se encuentra limitada de manera expresa para los propósitos educacionales del curso.

