

Metodologías Ágiles aplicadas a la Gestión del Conocimiento como herramienta colaborativa

Autor principal: Lorena Natalia Barale,

Colaboradores: Daniel Herrera, Alejandra Jewsbury, Silvia Lanza Castelli, Fanny Esther Montoya, Valeria Ortiz Quirós, Fernanda Rodríguez Aleua,

Departamento de Sistemas de Información
Facultad Regional Córdoba, Universidad Tecnológica Nacional
Maestro López esquina Cruz Roja,
lorebarale@gmail.com,

Resumen

La Gestión del conocimiento podemos entenderla como la instancia de gestión en la cual los involucrados obtienen una variedad de recursos para apoyar el desarrollo del conocimiento dentro del entorno de aprendizaje. (Davenport, T. H., & Prusak, L., 2001).

Nuestra investigación pretende demostrar la importancia del rol de conocimiento y su gestión para obtener valor agregado y componentes estratégicos para el aprendizaje.

Este modelo pedagógico de Gestión de conocimiento tiene un soporte teórico basado en tres ejes: un entorno virtual de aprendizaje colaborativo utilizando plataforma Moodle, la integración de los principios y prácticas de Scrum para la organización de equipos productivos y el soporte de herramientas de gestión de proyectos ágiles colaborativos. Se incorpora a esta investigación el seguimiento y mediciones del desempeño individual y grupal de los alumnos, obteniendo datos de la participación en las actividades de cada módulo que incluyen lecturas y análisis de casos, cuestionarios, entrega de trabajos y participación en distintos dispositivos del aula virtual. Además la aplicación de evaluación utilizando el 'juicio evaluativo' que propone mejoras a las prácticas existentes de educación superior por medio

de: autoevaluación, evaluación por pares, retroalimentación, rúbricas y uso de ejemplos para contribuir a que los alumnos puedan evaluar la calidad de sus trabajos.

Palabras clave: Gestión del Conocimiento. Metodologías ágiles. Scrum. Juicio valorativo

1. Introducción

La carrera de Ingeniería en Sistemas de Información, como todas las cosas tan relacionadas con la tecnología presentan cambios constantemente. Cambian los procesos, las metodologías, las herramientas, las técnicas utilizadas, los marcos de trabajo, el hardware, el software, las formas de liderazgo, la forma de trabajo de los equipos. El área educativa, no puede quedar ajena y es nuestro deber como educadores ayornarnos a estos avances tecnológicos para formar profesionales competentes al momento de salir al mercado laboral.

Abordar la gestión del conocimiento incorporando un marco de trabajo propuestos por las metodologías ágiles, nos pareció una manera de innovar en el aprendizaje y forma de interacción entre alumnos y docentes, a la vez que entendemos que esta propuesta no sólo incorpora nuevos conocimientos a los alumnos en técnicas y herramientas que

utilizarán en la vida profesional, sino que genera motivación hacia el aprendizaje.

2. Marco teórico

El desarrollo ágil es un marco conceptual que reconoce las distintas interacciones y cambios que ocurren en todo desarrollo de software. Si bien el Término "Ágil" fue definido en el "Manifiesto Ágil" en 2001, estas metodologías ágiles evolucionaron a partir de varios métodos:

-1930, Ciclo PDCA: Walter Shewhart propone el ciclo de "Planear", "Hacer", "Estudiar" y "Actuar", un concepto que luego fue difundido por Deming.

-1940: Taiichi Ohno inventa el método Kanban en Toyota. El Lean Manufacturing es una fuente de inspiración y precursor del movimiento ágil.

-1974: Un Paper de E.A. Edmonds presenta el concepto de "Proceso de Desarrollo de Software Adaptativo" en 1974. Asimismo, también durante los 70, Tom Gild publica conceptos sobre la Gestión de Proyectos Evolutiva (EVO).

-1992: Alistair Cockbur presenta los Métodos Crystal, el punto de inicio de la evolución de las metodologías de desarrollo de software que eventualmente resultaron en lo que hoy se conoce como el movimiento ágil.

-1993: Bill Opdyke presenta el concepto de "Refactorización" en su paper titulado "Creando Superclases Abstractas por medio de la Refactorización" (técnica para la reestructuración de piezas de código existente).

-1995: Programación en Pares (Pair Programming). Este concepto se convirtió en una parte integral de la Programación Extrema.

-1995: El método Scrum fue ideado por Ken Schwaber y Jeff Sutherland, quienes lo presentaron en la conferencia OOPSLA 95 (Object-Oriented Programming, Systems, Languages & Applications) en Austin Texas.

Mike Beedle fue uno de los pioneros en adoptar Scrum y colaboró con su adopción en muchas organizaciones.

Como se sabe, Scrum es prácticamente el estándar de facto para el desarrollo ágil.

-1997: Desarrollo guiado por funcionalidades Feature Driven Development (FDD). En él se definen mejores prácticas como son: Modelado de objetos de dominio, Desarrollo por funcionalidades, Propiedad individual de las clases (Código), Equipos de trabajo por funcionalidad, Inspecciones, Gestión de Configuración, Compilaciones regulares (periódicas) y visibilidad del avance y resultados.

- 1999: Desarrollo de Software Adaptativo. La metodología propone un ciclo de vida de 3 fases: Especulación, Colaboración y Aprendizaje.

- 1999: Programación Extrema / Extreme Programming (XP). Como parte de la Programación Extrema, también formuló los conceptos de Historias de Usuario y Planificación de Releases. La metodología especifica buenas prácticas para la planificación, gestión, diseño, codificación y pruebas.

-1999: Integración Continua: Kent Beck definió este concepto también, pero fue un paper de Martin Fowler el que lo popularizó.

El 17 de febrero de 2001 diecisiete críticos de los modelos de mejora del desarrollo de software basados en procesos, convocados por Kent Beck, quien había publicado un par de años antes Extreme Programming Explained, libro en el que exponía una nueva metodología denominada Extreme Programming, se reunieron en Snowbird, Utah para tratar sobre técnicas y procesos para desarrollar software. En la reunión se acuñó el término "Métodos Ágiles" para definir a los métodos que estaban surgiendo como alternativa a las metodologías formales (CMMI, SPICE) a las que consideraban excesivamente "pesadas" y rígidas por su carácter normativo y fuerte dependencia de planificaciones detalladas previas al desarrollo.

Los integrantes de la reunión resumieron los principios sobre los que se basan los métodos alternativos en cuatro postulados, lo que ha quedado denominado como Manifiesto Ágil.

Valores del manifiesto ágil

- Valorar más a los individuos y sus interacciones que a los procesos y las herramientas
- Valorar más el software funcionando que la documentación exhaustiva
- Valorar más la colaboración con el cliente que la negociación contractual
- Valorar más la respuesta ante el cambio que seguir un plan

Beneficios

En primer lugar, las metodologías ágiles mejoran la satisfacción del cliente dado que se involucrará y comprometerá a lo largo del proyecto. En cada etapa del desarrollo se informará al cliente sobre los progresos del mismo. De ese modo, el cliente puede sumar su experiencia para optimizar las características del producto final. Se pueden evitar así numerosos malentendidos dado que el cliente poseerá en todo momento una completa visión del estado del producto.

Asimismo, mejora la motivación e implicación del equipo de desarrollo. Pero esta mejora no es casual: las metodologías ágiles permiten a todos los miembros del equipo conocer el estado del proyecto en cualquier momento. Los compromisos son negociados y aceptados por todos los miembros del equipo y las ideas de cualquiera de sus integrantes son tenidas en cuenta.

Destacar que los procesos ágiles permiten ahorrar tanto tiempo como costes. El desarrollo ágil trabaja de un modo más eficiente y rápido que otras metodologías. Además, estos procesos ponen el foco en cumplir estrictamente el presupuesto y los plazos pactados a la hora de definir y planificar el proyecto.

Se trabaja con mayor velocidad y eficiencia. En las metodologías ágiles se trabaja

realizando entregas parciales pero funcionales del producto. De ese modo, es posible entregar en el menor intervalo de tiempo posible una versión funcional del producto.

Gracias a las entregas parciales (centradas en entregar en primer lugar aquellas funcionalidades que en verdad aportan valor) y a la implicación del cliente será posible eliminar aquellas características innecesarias del producto.

Las metodologías ágiles permiten mejorar la calidad del producto. La continua interacción entre los desarrolladores y los clientes tienen como objetivo asegurar que el producto final sea exactamente lo que el cliente quiere y necesita. Además, este enfoque permite abrazar la excelencia tecnológica, lo que permite obtener un producto tecnológicamente superior.

Por otro lado, esta metodología permite alertar rápidamente tanto de errores como de problemas. En la etapa de planificación, el equipo ha presentado una hoja de ruta anticipando y dando respuesta a los principales problemas técnicos y a la velocidad en la que se puede trabajar.

Y, finalmente, las metodologías ágiles permiten rentabilizar nuestras inversiones más rápidamente. Gracias a la realización de entregas tempranas el cliente tendrá rápido acceso a aquellas funcionalidades que en verdad aportan valor acelerando el retorno de la inversión.

Inconvenientes de las metodologías ágiles

Las metodologías ágiles son fáciles de entender pero difíciles de llevar bien a la práctica. Requiere un fuerte compromiso y una estrecha colaboración entre todas las partes.

Las metodologías ágiles pueden ser ineficientes en grandes organizaciones. En empresas grandes suele haber especialistas del diseño de software, las pruebas, etc.

En las primeras fases, no se conocerá cómo quedará el proyecto una vez finalizado, esto puede generar dudas en clientes acostumbrados a tenerlo todo bien planificado.

El uso de las metodologías ágiles en el mundo

En el desarrollo de Software durante el SXXI, el auge de las metodologías ágiles ha ido en incremento por los muchos beneficios citados previamente.

Es así que son muchas las organizaciones que han adoptado estos marcos de trabajo para sus proyectos. A continuación podemos citar grandes empresas que han adoptado metodologías ágiles:

- Apple
- Zara
- PayPal
- Facebook
- Spotify

En nuestro país, también han tenido gran auge y ha sido utilizado en muchas organizaciones tales como Globallogic, Sancor Seguros, Zurich, Softtek, EY, Accenture Argentina, Redbee Studios, Avaya, Baufest, ICBC Argentina.

En la ciudad de Córdoba, las metodologías ágiles han sido incorporadas en el desarrollo de software tanto por compañías multinacionales como por compañías locales pertenecientes al Córdoba Technology Cluster. Tanto es así que actualmente ya se está comenzando a utilizar estos marcos de trabajo en dependencias de organismos públicos como por ejemplo el área de Sistemas del Ministerio de Educación de Córdoba, que está organizando su personal de sistemas en grupos scrum para el desarrollo de los proyectos encargados al área.

3. Objetivos y Metodología

Existen muchas formas de aplicar las Metodologías Ágiles para adaptarlas a las necesidades de cada grupo de trabajo y proyecto. **Scrum** es la metodología ágil más popular hoy en día. Adopta la idea de que el proyecto tiene que desarrollarse de forma

iterativa e incremental. El proyecto avanza a través de una serie de iteraciones llamados "sprints", al final de cada "sprint" se obtiene un producto potencialmente listo para su uso.

Aunque esta forma de trabajo surgió en empresas de productos tecnológicos, es apropiada para cualquier tipo de proyecto con requisitos inestables y para los que requieren rapidez y flexibilidad, situaciones frecuentes en el desarrollo de determinados sistemas de software.

Roles Scrum

El **Product Owner** se asegura de que el equipo Scrum trabaje de forma adecuada desde la perspectiva del negocio. El Product Owner ayuda al usuario a escribir las historias de usuario, las prioriza, y las coloca en el Product Backlog.

El Scrum es facilitado por un **ScrumMaster o facilitador**, cuyo trabajo primario es eliminar los obstáculos que impiden que el equipo alcance el objetivo del sprint. El ScrumMaster no es el líder del equipo (porque ellos se auto-organizan), sino que actúa como una protección entre el equipo y cualquier influencia que le distraiga. El ScrumMaster se asegura de que el proceso Scrum se utiliza como es debido. El ScrumMaster es el que hace que las reglas se cumplan.

El **equipo** tiene la responsabilidad de entregar el producto. Es recomendable un pequeño equipo de 5 a 9 personas con las habilidades transversales necesarias para realizar el trabajo (análisis, diseño, desarrollo, pruebas, documentación, etc).

Ceremonias Scrum

Al comienzo de un sprint, el equipo de scrum tiene un evento de **planificación de sprint**. Uno de los objetivos de la reunión es identificar y comunicar cuánto del trabajo es probable que se realice durante el actual Sprint.

Cada día durante la iteración, tiene lugar una reunión o **scrum diario** de estado del

proyecto. La reunión tiene una duración fija de 15 minutos, de forma independiente del tamaño del equipo.

Al final de un sprint, el equipo realiza dos eventos: la revisión del sprint y la retrospectiva del sprint. En la **reunión de revisión de sprint** se presentan los trabajos completados durante el sprint.

Después de cada sprint, se lleva a cabo una **retrospectiva del sprint**, en la cual todos los miembros del equipo dejan sus impresiones sobre el sprint recién superado. El propósito de la retrospectiva es realizar una mejora continua del proceso.

Este marco de trabajo fue el elegido como uno de los tres pilares teóricos que dan soporte al modelo de Gestión del Conocimiento sobre el que se basa nuestra investigación, la que pretende demostrar la importancia del rol del conocimiento y su gestión para obtener un valor agregado al aprendizaje. Los otros dos pilares sobre los que se apoya esta investigación son: un entorno virtual de aprendizaje colaborativo utilizando la plataforma Moodle y la aplicación de evaluación utilizando el “juicio evaluativo”.

El uso del marco de trabajo Scrum permite mediante la organización de equipos productivos y el soporte de herramientas de gestión de proyectos ágiles colaborativos incorporar a esta investigación el seguimiento y mediciones del desempeño individual y grupal de los alumnos, obteniendo datos de la participación en las actividades de cada módulo que incluyen lecturas y análisis de casos, cuestionarios, entrega de trabajos y participación en distintos dispositivos del aula virtual.

La propuesta busca promover la creación, el agregado, utilización y reutilización de los conocimientos que van adquiriendo los alumnos en la asignatura Análisis de Sistemas. Todos los Trabajos Prácticos son instancias de aprendizajes y adquisición de conocimientos en sí mismos.

4. Resultados

Con el uso de Scrum se pretendió brindar un marco de trabajo para ordenar las prácticas de equipos y dinamizar la adquisición de conocimientos.

Una vez conformados los equipos de trabajo se comunicaron las consignas y los plazos de entrega para el desarrollo del trabajo práctico integrador, cuyo objetivo es integrar todos los conocimientos aprendidos en un documento de especificación de requerimientos. Para este proceso se les solicitó aplicar el framework de Scrum donde se establecieron una serie de sprints definidos por el docente, en función de las tareas que debían realizar. Luego se fue tomando por cada sprint, la definición de las actividades, asignación de tareas a un responsable del equipo identificado como Scrum master. Además para la planificación y seguimiento de los avances, los alumnos debían realizar un tablero en alguna herramienta de gestión de proyectos ágiles para ello los docentes proporcionaron tutoriales interactivos y capacitación en el uso general de la herramienta Trello y para hacer uso de repositorios de material en la nube donde la mayoría gestionaba sus contenidos con Drive y otras aplicaciones de Google.

Cabe destacar que todas las prácticas e interacciones o intervenciones que realizan los docentes son gestionadas a través de la plataforma Moodle. Esto ha permitido, incrementar el espacio de adquisición de nuevos conocimientos, reusarlos y compartirlos. Es tarea del product Owner (representado por docentes) validar los contenidos y gestionar los dispositivos que permiten a los alumnos gestionar sus conocimientos.

Al final del sprint se aplicó el dispositivo de evaluación utilizando el 'juicio evaluativo' que propone algo innovador al modelo que venimos aplicando y es mejorar a las prácticas existentes de educación superior por medio de: autoevaluación, evaluación por pares, retroalimentación, rúbricas y uso de ejemplos para contribuir a que los alumnos puedan colaborar y compartir sus trabajos e incrementar la

calidad de los mismos.

5. Conclusiones

Tomando en cuenta los tres pilares de nuestro trabajo en primer lugar utilizamos un marco de trabajo basado en Scrum, lo que nos permitió ordenar los diferentes equipos integrados por los grupos de alumnos de la Cátedra, en los que se buscaba que los mismos aprendan a desenvolverse en ambientes de trabajo colaborativos, desempeñando diferentes roles como parte de los equipos de trabajo, como Scrum Master liderando cada grupo e interactuando con los docentes quienes desempeñan el rol de Product Owner.

En segundo lugar los grupos utilizaron diferentes herramientas que les permitieron gestionar cada proyecto de manera colaborativa como Google Drive, Google Doc, Trello lo que les permitió gestionar sus trabajos grupales y mediante la Plataforma Moodle como un repositorio del conocimiento, en la que el equipo docente trabaja como moderador supervisando y gestionando de manera permanente las actividades de los alumnos.

Y en tercer lugar al final de cada trabajo se aplicó como dispositivo de evaluación el Juicio Evaluativo por medio de autoevaluación, evaluación entre pares en la que los diferentes grupos realizan críticas constructivas y aportes lo que permite enriquecer las propuestas de cada grupo.

6. Referencias

Martín Alaimo (2014) – Proyectos Ágiles con Scrum. Impresiones Dunken.

(2013). Una breve historia de las metodologías ágiles.
<http://www.pmoinformatica.com>

Eduardo Martinez, (2014). Las 8 grandes ventajas de las metodologías ágiles
<https://www.iebschool.com>

Pablo Domínguez, (2017). Las metodologías ágiles
<https://openclassrooms.com>

Sonia Mañe Vernia, (2016). Ejemplos de cómo las empresas usan la metodología ágil.
<https://www.iebschool.com/blog/metodologia-agil-agile-scrum/>

Diario EL CRONISTA. (2015) La agilidad le gana al software.
<https://www.cronista.com/itbusiness/La-agilidad-le-gana-al-software-20150303-0010.html>