

SCRUM: A TRAVÉS DE UNA APLICACIÓN MÓVIL

Scrum: Through a Mobile Application

Julio Alberto Higuera

Ingeniero de Sistemas

Especialización

Ingeniería del Software, UDFJC

Correo electrónico: cbl886@gmail.com

Carlos Mario Durán Camelo

Ingeniero de Sistemas

Especialización

Ingeniería del Software, UDFJC

Correo electrónico:

c_duran_camelo@hotmail.com

Orlando Torres Cediel

Ingeniero de Sistemas

Especialización

Ingeniería del Software, UDFJC

Correo electrónico:

orlandot2401@gmail.com

Artículo investigación

Fecha de Recepción: 2014-06-01

Fecha de Aceptación: 2014-08-28

Resumen

Las metodologías ágiles fundamentadas en el manifiesto ágil (2001) centran el desarrollo del software en valores y principios en los que prima el correcto desarrollo de las personas y los equipos, la interacción con el cliente y sus necesidades, la continua entrega de software de valor y uso y una constante libertad y bienvenida a los cambios de requerimientos. SCRUM se ha destacado como la metodología ágil más utilizada en el mundo con una gran aceptación, en la que se definen claramente unos roles, artefactos y reuniones específicas que miden de manera constante el desarrollo del proyecto y beneficia la comunicación del equipo, sin embargo la ubicación geográfica de los miembros del equipo, así como las dificultades que pueden surgir para poder tener el equipo centralizado puede arrojar unos retos que se hace imprescindible abordar en un mundo globalizado. Una solución virtual a través de dispositivos móviles que se fundamente en las principales características de Scrum puede resolver el problema siempre que se cuente con un equipo comprometido y un seguimiento constante al proceso y esta aplicación no se aleje de los principios fundamentales de SCRUM.

Palabras clave: Ágil, dificultades, metodologías, SCRUM, Solución móvil.

Abstract

The agile methodologies grounded in the agile manifesto (2001) focus on software development values and principles on which the correct raw development of individuals and teams, the interaction with customers and their needs, continuous delivery of valuable software and constant use and freedom and welcome changing requirements. SCRUM has emerged as agile methodology most commonly used in the world wide acceptance, in which the roles, artifacts and specific meetings that constantly measure the project and team communication benefits are clearly defined, but the geographical location of team members, and the difficulties that may arise in order to have a centralized team can shed some challenges and it is essential to address in a globalized world. A virtual solution through mobile devices that is based on the main features of Scrum can solve the problem provided you have a committed team and a constant monitoring process and this application does not depart from the fundamental principles of Scrum Web environment and in real time.

Keywords: Agile, difficult, methodology, mobile solution, SCRUM.

INTRODUCCIÓN

En la actualidad, las metodologías de desarrollo ágil se han puesto a la vanguardia en la realización de proyectos debido a las ventajas que ofrecen frente a metodologías tradicionales, tales como ofrecer una rápida respuesta a los cambios en los requerimientos a lo largo del desarrollo del proyecto, realizar seguimiento de los proyectos gracias a su proceso iterativo, la evaluación del proyecto por parte de los clientes permitiendo realizar ajustes a tiempo, su simplicidad, etcétera.

La metodología SCRUM es, quizás, la más popular de estas metodologías. Ahora bien, su desarrollo implica la cercanía entre el equipo de trabajo y la presencia de todos, para desarrollar adecuadamente características básicas tales como el daily meeting o el seguimiento al tablero de tareas.

Esto no siempre parece ser posible, equipos de trabajo en los que los integrantes están ubicados en diferentes lugares o las dificultades de movilidad en las grandes ciudades para cumplir con este requerimiento pueden impedir que esta condición netamente imprescindible se cumpla a cabalidad.

Si se consigue una herramienta que permita tener todas las características de la metodología de manera virtual tales como el tablero de tareas, en donde plasmar las dificultades existentes para el desarrollo de alguna de esas tareas, el progreso en ese desarrollo, etc. Que además no se aparte de los principios fundamentales de desarrollo de la misma y, por el contrario, “obligue” a ser cumplidos, que se alimente de manera individual pero sea de visión global para el equipo y generé las estadísticas de seguimiento al proyecto se podría superar la necesidad de esa cercanía grupal sin perder la objetividad del trabajo.

El desarrollo de una aplicación móvil que contemple estas variables y que sea de fácil acceso y descarga es una solución a esta problemática, y una planeación responsable en el desarrollo del mismo es la base de una solución óptima.

Metodologías Ágiles

Las metodologías tradicionales para el desarrollo de software fueron creciendo y enriqueciéndose con los cada vez mejor estructurados sistemas de modelado tales como el UML y la forma como se pudo

describir una arquitectura de manera más clara, sin embargo con el correr de los años se fueron quedando inmersas dentro de un cuello de botella cada vez más grande:

Los mercados son muy cambiantes y las estrategias comerciales y de crecimiento se adaptan a esto rápidamente o sucumben, por ello si se requiere de un software su implementación la cual se da mucho tiempo después del momento de planeación por lo cual se corre un alto riesgo de obsolescencia y de tener que “rediseñarse” para nuevamente caer en el mismo ciclo de pérdida. Esto en realidad no es nuevo, ni siquiera de este siglo, en la década de los 80 ya era notoria esta necesidad, las empresas comenzaban a notar que el viejo enfoque secuencial para el desarrollo de nuevos productos no era funcional [1].

Nacieron entonces una serie de nuevas metodologías para el desarrollo de software que buscaron “rediseñar” la manera como se planteaba este desarrollo; fueron varias las escuelas que empezaron a tomar ciertos patrones como sus valores póstumos, y que, en esencia buscaban dar soluciones de software más rápidas y eficientes que se adaptarán mejor a los cambios y que pudieran ser puestas en práctica con mayor rapidez.

Sin ser la antítesis a la Ingeniería del Software tradicional, esta nueva forma de desarrollar software es la combinación de una filosofía (entregas tempranas, necesidades del cliente satisfechas, equipos pequeños, métodos informales, simplicidad) y unas directrices de desarrollo (entregas por encima de análisis y diseño, comunicación activa desarrollador-cliente). Para el año 2001, se consigue el primer acuerdo bajo el cual se registrarán estas metodologías: el manifiesto ágil.

Manifiesto Ágil

Liderados por Kent Beck, un grupo de 17 profesionales se reúnen con el ánimo de unificar conceptos y definir parámetros sobre los cuales las metodologías que lideraban basarían su desarrollo y mantendrían fidelidad; en esta reunión se acordaron cuatro valores fundamentales:

- 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.

Y no significa que solo se valora lo primero y no lo segundo, simplemente se valora más lo primero.

A su vez se definieron los 12 principios en los que se basan las metodologías ágiles, estos son:

1. La mayor prioridad es satisfacer al cliente mediante la entrega temprana y continua de software con valor.
2. Aceptación al hecho de que los requisitos cambien, incluso en etapas tardías del desarrollo. Los procesos Ágiles aprovechan el cambio para proporcionar ventaja competitiva al cliente.
3. Se entrega software funcional frecuentemente, entre dos semanas y dos meses, con preferencia al periodo de tiempo más corto posible.
4. Los responsables de negocio y los desarrolladores trabajan juntos de forma cotidiana durante todo el proyecto.
5. 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.

6. El método más eficiente y efectivo de comunicar información al equipo de desarrollo y entre sus miembros es la conversación cara a cara.
7. El software funcionando es la medida principal de progreso.
8. Los procesos Ágiles promueven el desarrollo sostenible. Los promotores, desarrolladores y usuarios deben ser capaces de mantener un ritmo constante de forma indefinida.
9. La atención continua a la excelencia técnica y al buen diseño mejora la Agilidad.
10. La simplicidad, o el arte de maximizar la cantidad de trabajo no realizado, es esencial.
11. Las mejores arquitecturas, requisitos y diseños emergen de equipos auto-organizados.
12. A intervalos regulares el equipo reflexiona sobre cómo ser más efectivo para a continuación ajustar y perfeccionar su comportamiento consecuentemente.

En el manifiesto además se consolidó de manera definitiva el término “ágil” como el más apropiado para lo que antes era conocido como “ligero” y que generaba la controversia al parecer demasiado simple y sin fundamento, lo que además se aclaró al formalizar de esta manera estas metodologías [2].

Contextualización de la Metodología Scrum

Es un término con raíces francesas que significa “aglomeración personas”, y que se introdujo al rugby para la descripción de una jugada en la que, por decisión del árbitro, los dos equipos arman una fila de jugadores, una en frente de la otra y agachados y agarrados forcejean contra el otro equipo, teniendo en la mitad de ellos el ba-

lón, que sin ser tocado, deberá quedar detrás del equipo que gana quedando en propiedad del balón y de la jugada con un jugador que atrás espera para tomarlo.

Cuando a mediados de los años 80 se empezaron a estudiar los nuevos procesos de desarrollo de productos exitosos de empresas de Estados Unidos y Japón (cámaras de fotos de Canon, fotocopiadoras de Xerox, automóviles de Honda, etc.), se encontró que los equipos que desarrollaron estos productos partían de requisitos muy generales, así como novedosos, y debían salir al mercado en mucho menos del tiempo del que se tardó en lanzar productos anteriores.

Estos equipos seguían patrones de ejecución de proyecto muy similares. En este estudio se comparaba la forma de trabajo de estos equipos altamente productivos y multidisciplinarios con la colaboración entre los jugadores de Rugby y su formación de Scrum (melé en español). [3]

Así pues, Jeff Sutherland, John Scumniotales y Jeff McKenna concibieron, ejecutaron y documentaron el primer Scrum para desarrollo ágil de software en 1993 del que se tenga conocimiento, utilizando el estudio de gestión de equipos de Takeuchi y Nonaka como base, y en 1995 Ken Schwaber formalizó el proceso para la industria de desarrollo de software; a partir de este momento miles de proyectos en todo el mundo han utilizado Scrum para el desarrollo de productos, tanto en empresas pequeñas, “startups” con tan sólo 3 personas desarrollando un producto, como en multinacionales de la talla de Google, Microsoft, Amazon, Phillips Medical, IBM, Intel, entre otros.

SCRUM entonces es una metodología de desarrollo ágil, y como tal, responde a los valores fundamentales expresados en el manifiesto ágil.

Desarrollo general de la metodología Scrum

La ejecución de la metodología SCRUM tiene como actores claramente definidos un Scrum Master, encargado de apoyar al grupo de desarrollo para que todos los objetivos sean cumplidos en su totalidad, el product Owner o Dueño de Producto que es el interesado en el producto a desarrollar y generalmente quien proporciona los recursos para la ejecución del proyecto, los Stakeholder (Expertos del negocio), que generalmente son los que conocen de forma específica los detalles del negocio, fundamentales a la hora de la definición de requisitos desarrolladores que cumplen con las tareas asignadas por el Scrum Master.

La definición de tareas comienza con una lista de requerimientos, priorizada de acuerdo a su importancia, definiendo requisitos, estimación de inicio, un test, nota y solicitante, con lo cual se denominan las funcionalidades, que se registran en un documento de nombre Product Backlog de uso general de todo el equipo del SCRUM, con esta información socializada se detalla cómo se va a desarrollar el proyecto y se desglosa en pequeñas tareas que no pasan de 2 días y que son asignadas a un desarrollador así el Sprint (ejecución de la tarea global), que usualmente se recomienda tenga un lapso de desarrollo de entre 2 y 4 semanas.

En una reunión diaria llamada “Daily Meeting” se definen todas las actividades que se van a desarrollar al día siguiente y se analizan los inconvenientes en el cumplimiento de las actividades del día precedente.

Por lo general este proceso es desarrollado en un tablero donde las tareas y tiempos estimados son escritos en pequeños papeles, así como las observaciones pertinentes y el cual se va trasladando de una columna

a otra de acuerdo al desenvolvimiento que se tenga al momento del desarrollo de la actividad realizada por el El Scrum Team. [4]

¿Por qué elegir SCRUM?

En este punto centramos como datos estadísticos los datos tomados de Versión One, que realiza anualmente una encuesta sobre el estado de las prácticas ágiles y que en su último reporte (2013) cuenta con la nada despreciable cifra de 3501 encuestados entre agosto 4 y octubre 16 de 2013, y a cuyo reporte completo se puede acceder a través de: <http://www.versionone.com/pdf/> [5]

Podemos apreciar que el uso de las metodologías ágiles se ha generalizado en el mundo para el desarrollo del software. En la figura 1 se puede observar que la mayoría de empresas constructoras de Software hoy día están desarrollando proyectos con metodologías ágiles y sigue en constante crecimiento, toda vez que esta misma estadística el año anterior arrojaba 4 puntos menos para el uso de estas metodologías.

Porcentaje de compañías constructoras de software que implementan metodologías ágiles

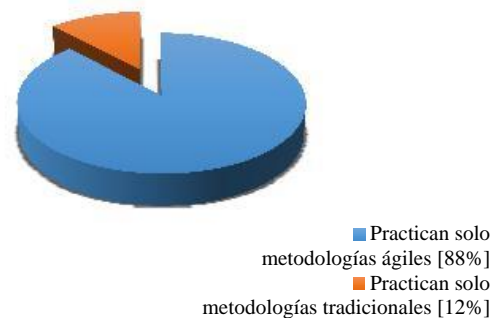


Figura 1. Porcentaje de compañías constructoras de software que implementan metodologías ágiles.

De la Figura 2, se puede concluir que Scrum es la metodología ágil más utilizada, por lo menos el 73% de los equipos de desarrollo ágil la utilizan, si se suman

aqueellos que la utilizan de manera pura (55%) y aquellos que la combinan con la metodología Xp o Kanban (18%).

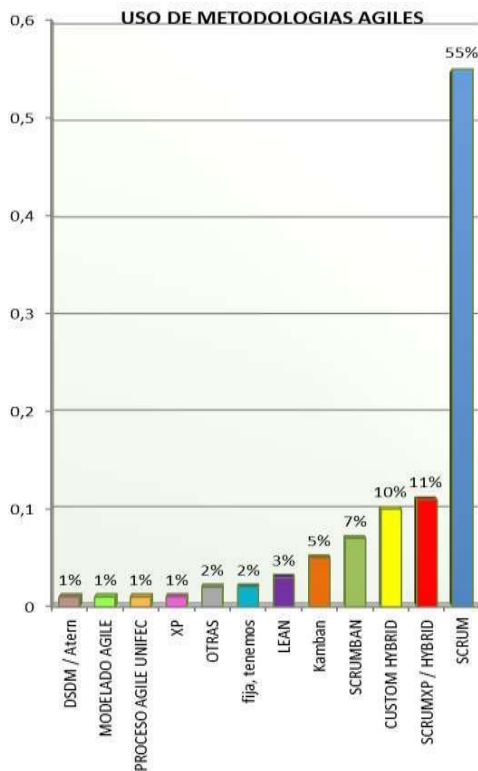


Figura 2. Porcentaje de Uso de las Metodologías ágiles.

Especificación de las características más importantes utilizadas en SCRUM

Con el ánimo de profundizar mucho más en las técnicas de Scrum en este capítulo se especificará de forma más detallada las prácticas más utilizadas en Scrum, definiendo su importancia.

Reuniones del Sprint

Se puede definir el sprint como el pulso del proyecto que a culminación de cada uno de estos permiten el crecimiento del mismo. Cada equipo define el sprint por un lapso de entre 2 y 4 semanas y, para el desarrollo del mismo hace una serie de reuniones de vital importancia, estas son:

- Primera Reunión Parte 1:

En ella se definen claramente los requisitos a satisfacer durante este sprint, generalmente el Product Owner (Dueño de producto-Cliente) debe estar presente para poder guiar al equipo en la dirección correcta y responder todas las preguntas que a estas alturas suelen ser bastantes. El Scrum Master debe asegurarse que cualquier stakeholder que maneje información de importancia para el sprint esté presente o por lo menos tengamos contacto directo telefónico, o por conferencia.

Cualquier ítem del backlog que se desea sea desarrollado durante el sprint que no haya sido estimado con anterioridad será estimado de manera inmediata durante la reunión, lo que no es una excusa para evitar la preparación del Backlog Grooming (Revisión de las historias de usuario a introducir en el Sprint).

A la culminación de esta, el equipo se compromete con el Product Owner a cumplir con los requisitos definidos en una estimación de tiempo que se basa en la experiencia del equipo. [6]

- Primera Reunión Parte 2:

Una vez se han sido claramente definidos los requisitos que se deben cumplir en el sprint la segunda parte de la reunión busca entonces concretar la forma en que se desarrollará este sprint que permite desarrollar un WorkShop de diseño en el que se especifican las tareas hasta el más alto nivel de definición lo que termina por generar un backlog de estas plasmadas normalmente en un tablero de tareas (Taskboard), si el Scrum Master considera necesario la presencia del productOwner o de alguno o algunos de los Stakeholder por el detalle a conseguir se asegurará de que estén presentes. [6]

- Reunión diaria (Daily Meeting)

Reunión con una duración estimada de 15 minutos generalmente hecha con los integrantes del equipo de pie con el ánimo de centrar la atención en la reunión, en la que se pretende hacer la revisión de las tareas realizadas por cada miembro durante las 24 horas anteriores, las dificultades encontradas para la culminación exitosa de estas y la proposición de las tareas a realizar durante las 24 horas siguientes. [7]

Es tarea del Scrum Master prestar atención a los detalles de la ejecución de la misma, la hora de inicio, la seriedad del ejercicio, la duración de las intervenciones, la forma centrada en que cada uno expone su parte sin demorarse en detalles técnicos que no vienen a lugar ni en historias que no tengan que ver, así mismo asegurar que si se presenta el Product Owner lo hace solo como observador. Es importante tener claro que en estas reuniones no se pretende brindar un informe a los “clientes” sino más bien mantener la comunicación del equipo y el conocimiento de todos acerca de la evolución del sprint. [8]

Artefactos:

Se definen cuatro artefactos en Scrum a saber:

Product Backlog

El product Backlog, es conocido como la lista de tareas a desarrollar definidas y especificadas en la primera reunión.

Dado que se consideran los requerimientos emergentes, lo que significa que no se conoce ni se puede conocer de antemano todas y cada una de las características que se desea que contenga el producto, se considera el Product Backlog como un documento viviente, que requiere una constante preparación (grooming) para mantenerlo actualizado y útil con el paso del tiempo.

Se agregarán nuevos ítems; algunos ítems existentes serán desagregados en varios ítems más pequeños; otros ítems serán removidos al darnos cuenta que ya no son necesarios. Es más, los ítems deben ser estimados para conocer la relación costo beneficio de los mismos, lo que influirá de manera directa en la ubicación que el mismo tendrá en el backlog. [6]

Para determinar que una tarea se ha definido adecuadamente, esta debe cumplir la regla INVEST:

Independiente: No depender de otras.

Negociable: Mientras no esté incluida dentro de un sprint, puede ser renegociada, cambiada, ampliada veces que sean necesarias.

Valorable: Aportar valor al usuario y al negocio.

Estimable: Debe ser comprensible y estimable por los miembros del equipo.

Pequeña (S por Small): Disponer de un tamaño adecuado para poder priorizarla e incluirla dentro de un sprint.

Testeable: Debe contar con una serie de pruebas o criterios que permitan asegurar que cumplen objetivo[9].

Sprint Backlog

Es el tablero de tareas, en el que se plasman las tareas del product Backlog, para verificar el estado en que se encuentran, este tablero, es una emulación semejante a la del tablero Kanban, en el que se identifican las columnas de Tareas por hacer, las que están en ejecución, en prueba y las terminadas, y, en el que además, como es la filosofía de Kanban se puede identificar los cuellos de botella o represamiento de tareas [6].

Burndown del Sprint

El gráfico de burndown de tareas del sprint tiene por objeto ayudar al equipo en la monitorización de su progreso y ser el indicador principal que informa las posibilidades de alcanzar su compromiso al finalizar el sprint.

El formato clásico requiere que el equipo estime la duración de cada tarea en horas de forma diaria. El burndown deberá completarse de forma tal que grafica cuántas horas de trabajo restan para concluir el sprint. [6]

Backlog de Impedimentos

El backlog de impedimentos consiste en la lista de situaciones que están impidiendo que el equipo progrese. Estas son situaciones que el Scrum Master debe eliminar con el fin de que el equipo trabaje de la mejor manera posible. Se debe entender que los impedimentos son de cualquier tipo, sea cual sea la magnitud de lo que representan. Es evidente que de una rápida solución dependerá notoriamente el rendimiento del equipo, de allí que un buen Scrum Master suele tener la meta de resolver estas situaciones en un tiempo máximo de 24 horas.

Dificultades en la aplicación de la metodología Scrum

Scrum es una metodología claramente definida, en la que los equipos son prácticamente auto dirigidos, no obstante para conseguir esto se requiere del profesionalismo del equipo de trabajo y el compromiso real para el sano desarrollo; indudablemente esta es la primera dificultad, por otra parte, como se puede notar, esta metodología está planteada para equipos de trabajo que comparten su espacio de trabajo, sin embargo no siempre se cuenta con esta disposición de equipos; hay equipos de trabajo distribuidos, haciendoreferencia al hecho

de no compartir la misma ubicación física y/o temporal.

Usar Scrum en equipos distribuidos supone una serie de retos a superar y que provienen de la distancia física, las diferencias culturales e idiomáticas, el salto empresarial y los horarios particulares de cada equipo y persona del equipo. Estas diferencias básicamente afectan e impactan, sobre todo, en la comunicación entre los miembros del equipo y, por ende, en la generación de un flujo de trabajo ágil, uno de los fundamentos de Scrum, pudiéndose dar situaciones con falta de sincronismo, pérdidas de afecciones personales, desmotivación o incomprensión.

Esta situación puede tener matices más complejos si nos encontramos con equipos de trabajo en los que por ejemplo haya diferencia idiomática, distintas nacionalidades, diferencias culturales o un total desajuste de horario (desfases de 6 o más horas debido a la ubicación geográfica de los integrantes) o de calendario. Scrum tiene como uno de los principales objetivos el facilitar la comunicación entre todos los integrantes del proyecto para minimizar el riesgo de que aparezcan trabas, debido a suposiciones y malos entendidos durante el desarrollo.

En un proyecto distribuido resultaría fundamental, desde un inicio, definir las fechas y horas en las que se está o no se está disponible, definiendo cuáles serán las fiestas locales, nacionales, horarios de comida, etc.

Respecto a las diferencias culturales, algunos puntos clave como la educación, formas de trabajo, tradiciones, etc. Debe existir la consciencia de los otros puntos de vista y tomar el tiempo necesario para conocer a fondo a todos los interlocutores.

Finalmente existen otras adicionales a tener en cuenta, como es el incremento en

tiempos que supone cuando se debe realizar validaciones. Cuando se trabaja en sedes muy distantes, un error en el envío, embalaje, fallo en el prototipo, puede suponer penalizaciones adicionales por temas de transporte. Aspectos logísticos a considerar específicamente cuando se generan entregables físicos como resultado de los diferentes sprints[10].

Evidentemente el reto a superar en un equipo distribuido tiene que ver con la manera de implementar uno de los principios básicos en que se sustenta Scrum: la continua y fluida comunicación entre los integrantes del equipo. El día a día de las personas es muy importante para lograr un equipo comprometido.

Posibles soluciones a las dificultades

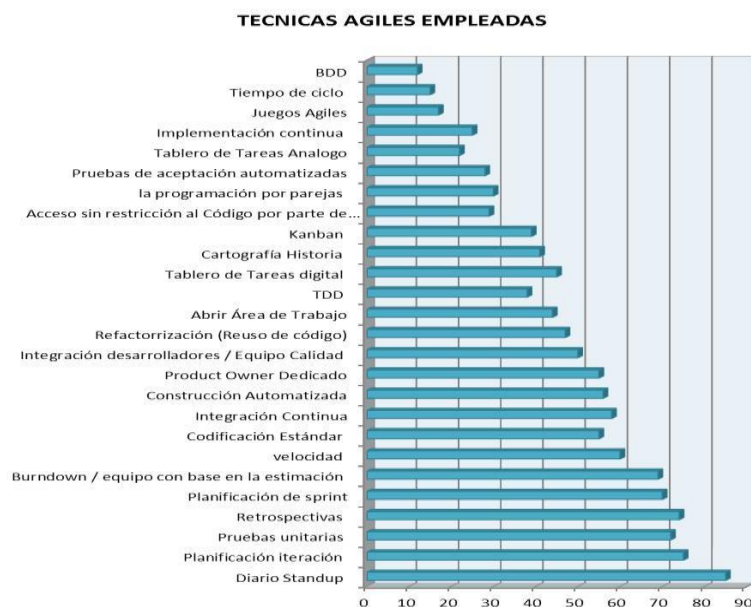
En este punto es entonces donde se plantea la utilización adecuada de la tecnología que permita utilizar elementos tan simples y sencillos como un dispositivo móvil que permita consignar la información del avance del proyecto, identificar y visualizar el tablero de tareas, las dificultades de

otros en las que probablemente se tiene alguna incidencia, una visualización del Burndown acompañado de ser posible de otras técnicas tales como video conferencias, que permita a la vez interactuar.

Es de notar que si bien es cierto que un equipo distribuido expone de la manera más evidente y cruda estas dificultades, estas también pueden aparecer sin necesidad de situaciones tan extremas; los problemas de movilidad en las grandes ciudades echan a perder grandes cantidades de tiempo productivo y podemos entonces encontrar que una solución virtual móvil puede redundar además en una solución de más amplio mercado.

La responsabilidad en el planteamiento de esta solución virtual y móvil debe estar acompañada de un estudio de los factores más críticos de Scrum descritos anteriormente para no perder ninguna de sus bondades, además de acuerdo al referenciado anteriormente de Versión One, se puede notar en la gráfica 3, las técnicas ágiles más utilizadas y que formarían parte necesaria de esta aplicación.

Figura 3. Técnicas ágiles más empleadas



Como se puede notar el manejo de las reuniones diarias, cronometradas con tiempo para no hacerlas más largas, con el debido registro de lo hecho en las últimas 24 horas, lo que se propone hacer en las próximas 24 y el planteamiento de las dificultades encontradas es de extenso uso, el detalle de la planificación de la iteración, la planificación del sprint, el taskboard (tablero de tareas virtual) al que se accedería con total tranquilidad, el acceso al Burn-down del sprint que permita visualizar el desenvolvimiento del equipo, son también características de necesario desarrollo en la aplicación[5].

CONCLUSIONES

A partir de este artículo se concluye la importancia que las metodologías ágiles han tomado en el desarrollo de software siendo hoy día de uso normal en las empresas desarrolladoras de software.

La metodología SCRUM es, sin lugar a dudas, por sus principios y su formalismo sencillo pero de estricto control y seguimiento la más aceptada e implementada dentro de estas empresas.

Pese a las grandes ventajas y aciertos que esta metodología conlleva también es posible visualizar algunas dificultades, que tienen que ver con el compromiso de los miembros del equipo y la ubicación geográfica o diversidad cultural en el equipo.

Estas dificultades no solo se presentan en situaciones extremas ya que finalmente el hecho de suponer un grupo centralizado exige de parte del grupo situaciones que no necesariamente todos pueden cumplir y por ende, impedir su correcto desarrollo.

El desarrollo de una aplicación que virtualmente permita hacer esta conexión de los miembros del equipo de trabajo, registrar

los progresos sin perder de vista las prácticas y elementos fundamentales de esta metodología se convierte en una solución a esta dificultad con lo que su implementación amplía sus posibilidades pudiendo aprovechar sus bondades sin marginarse de su implementación cuando aparentemente se tiene un contexto adverso.

Desarrollar esta solución generaría la posibilidad de combinar talentos geográficamente distribuidos y localmente difíciles de concentrar en un mismo lugar, o que por motivos de costes se harían prohibitivos.

Las posibilidades de reducción de costos de traslados, tanto económicos como personales de los miembros del equipo, se convierten en un atractivo adicional para el desarrollo de esta aplicación permitiendo la coordinación del trabajo remoto, sea geográfica o temporalmente distribuido, y al mismo tiempo permitiendo una visibilidad y producción de valor en forma continua.

El desarrollo de esta aplicación además ampliaría la posibilidad de disponer de mayores registros de las actividades realizadas a través de la plataforma de interacción, induciendo a una manifestación más explícita de las cuestiones tratadas durante el desarrollo de las actividades, facilitando tanto la obtención de métricas como análisis posteriores para facilitar la gestión de los procesos.

REFERENCIAS

- [1] Takeuchi, Hirotaka. The New Product Development Game. New York: Harvard Business Review, 1986.
- [2] Beck, Kent. Manifiesto por el Desarrollo Ágil del Software. s. l.: 2001. <http://agilemanifesto.org/iso/es>
- [3] www.proyectosagiles.org

<http://www.proyectosagiles.org/historia-de-scrum>

[4] Martínez, Eduardo.

www.comunidad.iebschool.com

<http://comunidad.iebschool.com/iebs/geral/metodologia-scrum/>

[5] VersionOne. 8th Annual State of Agile. s.l., 2013.

http://www.versionone.com/pdf/2013-state-of-agile-survey.pdf?utm_campaign=2014%20State%20of%20Agile%20Auto%20Responder&utm_medium=email&utm_source=Eloqua

[6] Hundermark, Peter. Mejor Scrum. 2009. Traducción por Alan Cymment, 2011.

[7] Yip, Jason. It's Not Just Standing Up: Patterns of Daily Stand-up Meetings. MartinFowler.com. 2006.

<http://martinfowler.com/articles/itsNotJustStandingUp.html>

[8] Vázquez, Pepe.

www.gestiondeproyectosit.es

Antipatrones en el Daily

<http://www.gestiondeproyectosit.es/blog/2013/06/antipatrones-en-el-daily/>

[9] TFC: Gestion de Proyectos Ágiles.mPeña, Rodrigo Gracia. 2013.

<http://openaccess.uoc.edu/webapps/o2/bitstream/10609/23087/1/rgraciapenTFC0613.pdf>

[10] Herranz, Raul y Mamoghli, Noel y otros.

http://www.scrummanager.net/files/scrum_distribuido.pdf