

# Metodología ágil scrumban en el proceso de desarrollo y mantenimiento de software de la norma moprosoft

David Ibarra Guzmán, Ulises Castañeda Islas, Carlos Pérez Corona, Blanca Estela Pedroza Méndez

Instituto Tecnológico de Apizaco, División de Estudios de Posgrado e Investigación, Tlaxcala, México

ibarra\_david50@hotmail.com,sesilucasta@gmail.com,  
cperez\_corona@hotmail.com,thelismolina@hotmail.com

**Resumen.** En este artículo se propone el uso de la metodología ágil Scrumban en las actividades del proceso de Desarrollo y Mantenimiento de Software (DMS) de la norma mexicana MoProSoft para la industria de desarrollo de software, con la finalidad de utilizar las mejores prácticas de ambas metodologías (Scrum y Kanban) durante las diferentes fases del proceso de desarrollo de software que describe la categoría Operación (OPE) de la norma MoProSoft. La propuesta es utilizar un tablero de control o sistema de tarjetas que permita visualizar de manera fácil y práctica el avance de las actividades de un proyecto, las personas que están trabajando en cada una de las tareas, así como en qué fase y estado se encuentran cada una de éstas con el apoyo de tarjetas visuales.

**Palabras clave:** MoProSoft, Scrum, Kanban, Scrumban, Metodologías ágiles.

## 1 Introducción

Hoy en día las empresas que desarrollan software se enfrentan a grandes retos tanto de demanda como de calidad en sus productos, el uso de las tecnologías de la información en la industria de cualquier tipo se ha vuelto una herramienta de primera necesidad y entre la sociedad un estilo de vida, la mayoría dependemos de alguna forma de tecnología, sin embargo, a pesar de que existen muchas empresas alrededor del mundo que se dedican a satisfacer la necesidad de contar con sistemas o aplicaciones de software que procesen la información, no todas lo hacen de forma correcta, ordenada y con calidad. Actualmente existe un alto índice de fracaso en los proyectos de desarrollo de software, las empresas no alcanzan a terminar en tiempo y dentro de los costos planificados por falta de implementación de un modelo de procesos de calidad. Según cifras de la organización “The Standish Group” en su estudio de proyectos que realizó en 2013, determinó que el 18% de

los proyectos examinados fallaron, el 43% no cumplió los requisitos y solamente el 39% tuvo éxito [7].

En México, son pocas las empresas de desarrollo de software que cumplen con sus compromisos de fechas de entrega y satisfacen las necesidades del cliente una vez que terminan el producto de software, la mayoría de éstas no cumplen con el calendario, los costos de producción de una aplicación de software se elevan, la funcionalidad que requiere el cliente no es totalmente satisfactoria y los problemas en la calidad del software terminan por generar demasiados problemas durante las diferentes etapas del desarrollo de software. Por esta razón se propone el uso de metodologías ágiles, como Scrum y Kanban durante las actividades que establece cada una de las fases del proceso de Desarrollo y Mantenimiento de Software (DMS) de la norma MoProSoft.

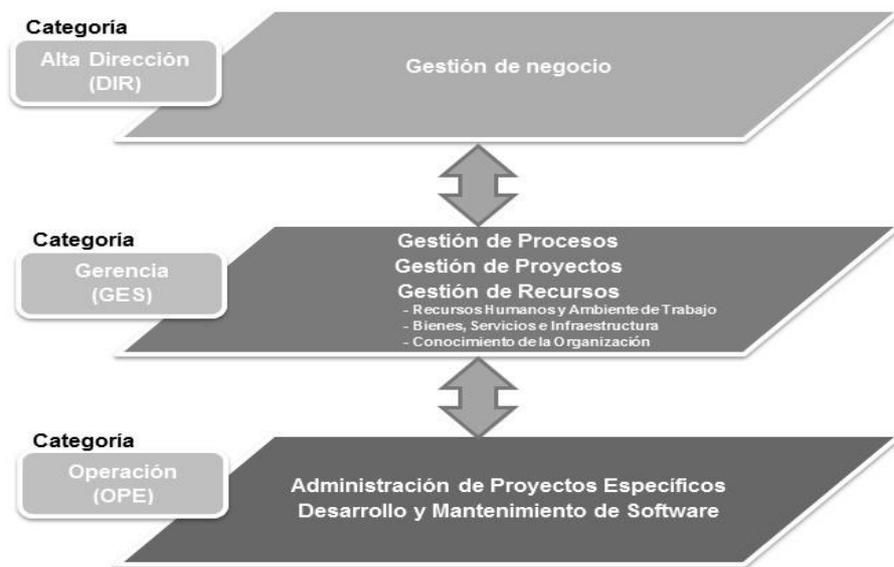
## **2 Moprosoft**

En 2002 se crea el modelo MoProSoft para las pequeñas y medianas empresas (PyMEs) con el objetivo de guiar y mejorar los procesos que se implementan para el desarrollo de software en México [1]. Como se muestra en la figura 1, este modelo consta de tres categorías y considera nueve procesos:

1. Categoría de alta dirección (DIR). Esta categoría aborda las prácticas relacionadas con la gestión del negocio. Proporciona los lineamientos a los procesos de la Categoría de Gerencia y se retroalimenta con la información generada por ellos [2].
2. Categoría de gerencia (GER). Esta categoría aborda las prácticas de gestión de procesos, proyectos y recursos en función de los lineamientos establecidos en la Categoría de Alta Dirección. Proporciona los elementos para el funcionamiento de los procesos de la Categoría de Operación, recibe y evalúa la información por éstos y comunica los resultados a la Categoría de Alta Dirección [2].
3. Categoría de operación (OPE). Categoría de procesos que aborda las prácticas de los proyectos de desarrollo y mantenimiento de software (DMS). Esta categoría realiza las actividades de acuerdo a los elementos proporcionados por la Categoría de Gerencia y entrega a ésta la información y productos generados [2].

En este trabajo se hace referencia específicamente al proceso de Desarrollo y Mantenimiento de Software (DMS) que se encuentra en la categoría de Operación (OPE). El propósito del DMS es la realización de las actividades de análisis, diseño, construcción, integración y pruebas de productos de software nuevos o modificados cumpliendo con los requerimientos especificados.

El proceso de Desarrollo y Mantenimiento de Software se compone de uno o más ciclos de desarrollo, en donde cada ciclo está compuesto por diferentes fases como se muestra en la tabla 1.



**Fig. 1.** Estructura de procesos del MoProSoft.

**Tabla 1.** Fases del DMS.

Fase	Descripción
F1:	Realización de la fase de inicio
F2:	Realización de la fase de requerimientos
F3:	Realización de la fase de análisis y diseño
F4:	Realización de la fase de construcción
F5:	Realización de la fase de integración y pruebas
F6:	Realización de la fase de cierre

### 3 Scrum

Scrum es un modelo que organiza a las personas en equipos pequeños, interdisciplinarios y auto-organizados [3], divide el trabajo en una lista de entregables pequeños y concretos [3], llamados sprint e incrementos, asigna a cada actividad de la lista un orden de prioridad que se determina en colaboración con el cliente, en base a la revisión realizada de un entregable después de cada iteración y además estima el esfuerzo relativo que tiene cada actividad. Los roles más importantes de Scrum son: Product Owner (Propietario del producto), Scrum Master (facilitador) y Team (equipo de desarrollo) como se muestra en la figura 2. En la tabla 2 se muestra la descripción de roles que Scrum utiliza para el desarrollo de un proyecto de software.

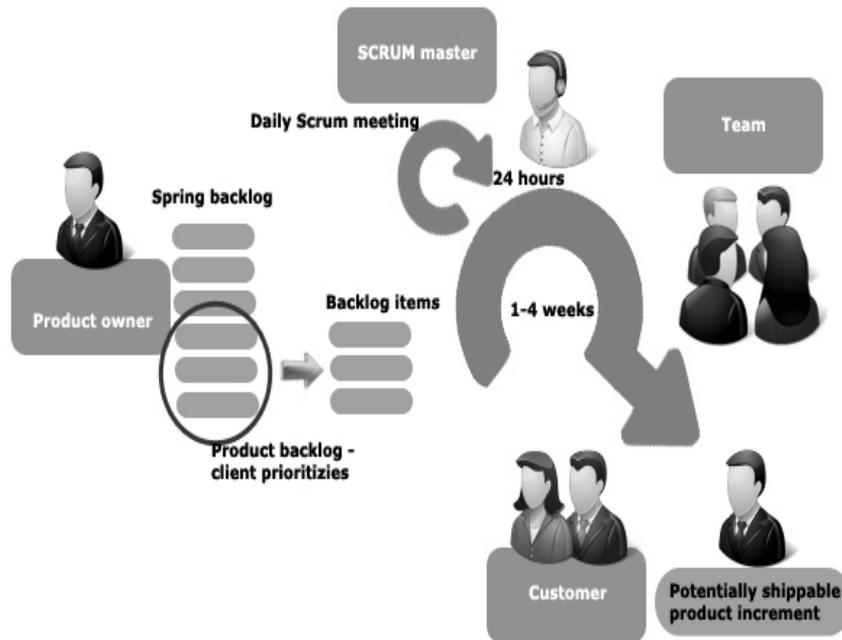


Fig. 2. Metodología ágil Scrum.

Tabla 2. Roles de Scrum.

Nombre	Descripción
Product Owner	Es la persona que se encarga del backlog

	(lista de requerimientos de un proyecto), es el cliente o un representante.
Scrum Master	Es la persona que se encarga de que las reglas se hagan efectivas, no es el líder del equipo.
Team	Es el equipo de desarrollo.

Scrum divide el tiempo en iteraciones cortas de longitud fija (generalmente de 1 a 4 semanas), con código potencialmente entregable y demostrado después de cada iteración y optimiza el proceso mediante una retrospectiva después de cada iteración [3].

En la Tabla 3 se muestran los productos que utiliza Scrum: Product Backlog, Sprint Backlog, Increment of potentially shippable product functionality y Burndown Chart.

**Tabla 3.** Productos de Scrum.

Nombre	Descripción
Product Backlog	Es una lista de los requerimientos del proyecto para realizar un producto de software completo y funcional, la lista tiende a cambiar con el tiempo de acuerdo a las condiciones.
Sprint Backlog	Es una lista de tareas que el equipo de desarrollo define para las iteraciones o sprint.
Increment of potentially shippable product functionality	Es un producto de software que cumple con todos los requerimientos funcionales para el sprint.
Burndown Chart	Es una gráfica donde se muestra el trabajo a lo largo del sprint.

## 4 Kanban

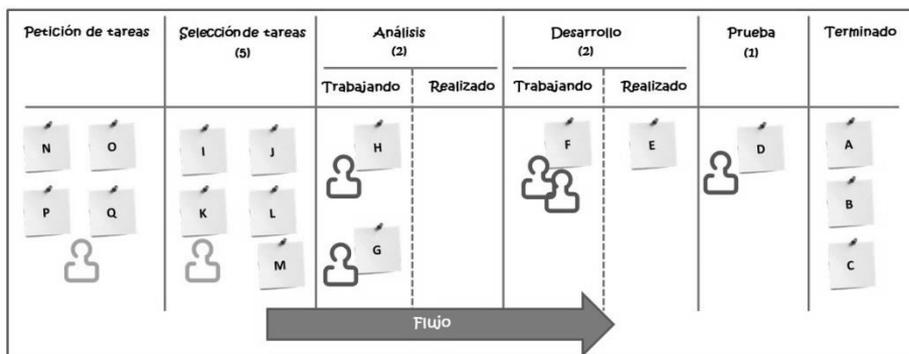
El método Kanban en el desarrollo de software fue originado en 2004, cuando David J. Anderson [6] estaba ayudando a un pequeño equipo de TI de Microsoft que estaba funcionando inadecuadamente.

El método Kanban en el desarrollo de software maneja equipos de proyectos para visualizar el flujo de trabajo, limita el trabajo en progreso (WIP) en cada etapa del flujo de trabajo y mide el tiempo de ciclo (lead time) como se muestra en la tabla 4.

**Tabla 4.** Principales características de Kanban.

Característica	Descripción
Visualiza el flujo de trabajo	Divide el trabajo en bloques, escribe cada elemento en una tarjeta y se coloca en el tablero. Utiliza columnas con nombre para ilustrar dónde está cada elemento en el flujo de trabajo.
Limita el trabajo en curso (WIP)	Asigna límites concretos al número de elementos que pueden estar en progreso en cada estado del flujo de trabajo.
Mide el tiempo de ciclo (lead time) medio para completar un elemento	Optimiza el proceso para que el lead time sea tan pequeño y predecible como sea posible.

El tablero Kanban proporciona visibilidad del proceso del software, en el muestra el trabajo asignado para cada desarrollador, comunica claramente las prioridades y resalta los cuellos de botella, así el equipo se concentra en resolver los problemas que bloquean el proceso y restauran el flujo productivo. En la figura 3 se muestra un ejemplo de un tablero básico Kanban.



**Fig. 3.** Tablero Kanban.

Kanban limita el trabajo en curso de acuerdo a la capacidad del equipo, que equilibra la demanda contra el rendimiento del trabajo liberado por el equipo. Esto ayuda a visualizar los problemas del proceso, minimiza los defectos y mantiene un flujo estable. Al limitar el trabajo en curso se consigue un ritmo de desarrollo sostenible, elevando la calidad de los productos y un mayor rendimiento de los integrantes del equipo. El flujo estable y la calidad en el producto ayuda a reducir el tiempo de ciclo (lead time), generando la libe-

ración de entregables de forma más regular que incrementa la confianza del cliente hacia la empresa desarrolladora de software [5].

## 5 Moprosoft-Scrumban

La propuesta que se pretende implementar es la combinación de las mejores prácticas de desarrollo de software de las metodologías ágiles Scrum y Kanban, con la finalidad de apoyar al proceso de Desarrollo y Mantenimiento de Software (DMS) dentro de la empresa Miracle Business Network S.A. de C.V. que actualmente se encuentra certificada en el nivel 2 de la norma Moprosoft.

En base a las fases del proceso DMS de MoProSoft descritas en la parte superior, se propone un tablero de control Kanban como se muestra en la figura 4. Debe tomarse en cuenta los recursos humanos programados o planeados por el Responsable de Administración del Proyecto Específico (RAPE) y el rol de cada integrante del equipo de desarrollo que especifica la norma MoProSoft para el proceso DMS (ver figura 5), se establece el límite de trabajo en curso (WIP) en cada etapa del flujo de trabajo de un sprint o iteración para un entregable.

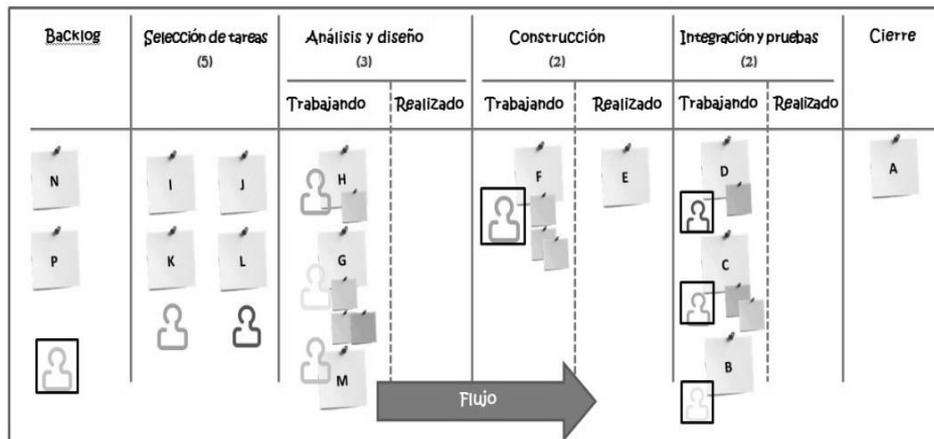


Fig. 4. Tablero MPS-Scrumban.

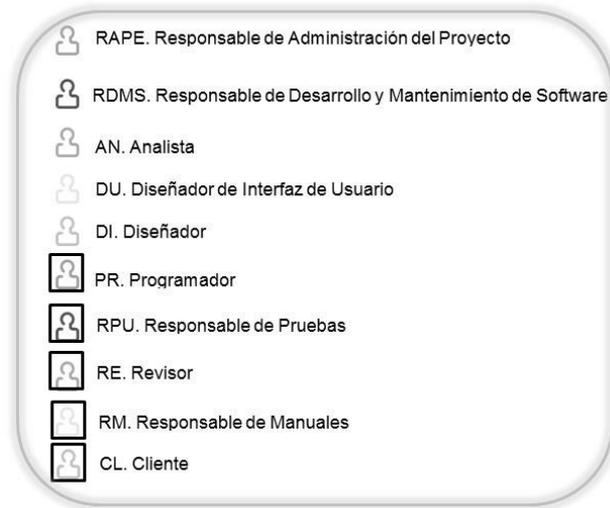


Fig. 5. Roles del proceso Desarrollo y Mantenimiento de Software MoProSoft.

Cada tarjeta visual debe contener la siguiente información: descripción de la actividad, fecha en que el elemento se agrega al tablero, fecha límite de entrega (si aplica), el nivel de prioridad y el nombre de la persona que trabaja en la actividad en el momento actual [4] como se muestra en la figura 6. Los integrantes del equipo, al momento de comenzar a realizar una actividad que se encuentre en la columna de selección de tareas deben tomar en cuenta la información sobre el dato de prioridad que se muestre en la tarjeta, se debe considerar en primer lugar las actividades marcadas como “Urgente”, posteriormente las de prioridad “Alta”, sino existe ninguna tarjeta con ese tipo de prioridades se debe considerar aquellas en donde la fecha límite de entrega está en riesgo y finalmente las actividades más antiguas, como se detalla en la tabla 5.



Fig. 6. Tarjeta visual para tablero Kanban.

Una historia o característica de lo que el cliente o propietario del producto necesita, consta de un conjunto de tareas que se tienen que hacer para poder implementar dicha característica en el producto final que se entrega al cliente. Las características (historias) son productos entregables. Estas fluyen a través del tablero de izquierda a derecha y su estado en el flujo de trabajo está indicado por la columna en donde esta se encuentra. El límite de trabajo en curso (WIP) en cada columna aplica solo para las características, no para las tareas [4].

**Tabla 5.** Prioridad de actividades.

Prioridad	Descripción
Urgente	Debe ser inmediatamente atendida y mantenida en constante movimiento por cada etapa del tablero. Se puede interrumpir el trabajo que se esté realizando en ese momento e incluso cambiar el límite del trabajo en curso (WIP) si es necesario.
Alta	Debe ser atendida una vez que se termine la actividad que se esté realizando en ese momento, este tipo de prioridad no puede cambiar el límite del trabajo en curso (WIP).
Fecha de límite	Se debe atender solamente si la fecha de límite de entrega está en riesgo.
Fecha en que se agrega al tablero	Se debe atender la que tenga fecha de puesta en el tablero más antigua.

Las tareas forman parte de una característica (historia), es decir, una característica puede tener una o más tareas que tiene que realizar un integrante del equipo para poder ser implementada. Las tareas no fluyen a través del tablero, pero su estado es indicado por tarjetas de colores [4], como se muestra en la figura 7, tarjeta en color azul indica que se trata de una tarea que se lleva a cabo para una determinada característica, verde indica tarea completada y roja tarea bloqueada, todas estas durante la etapa de **Construcción**; naranja indica que se trata de un defecto encontrado en la etapa de **Integración y Pruebas**.

Cuando una característica llega a la etapa de **Construcción - Realizado** entonces todas las tareas son apartadas. Si existen defectos encontrados durante la etapa de **Integración y Pruebas** entonces tarjetas de defectos en color naranja son agregadas dentro de columna **Trabajando** de dicha etapa.

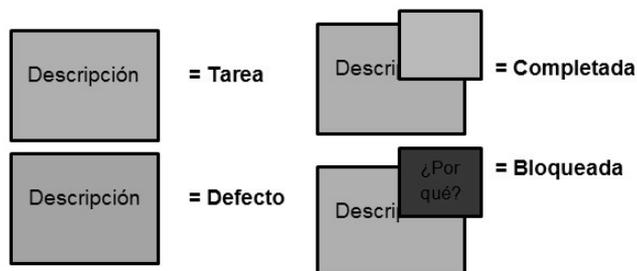


Fig. 7. Tarjetas de colores que indican el estado de las tareas.

## 6 Conclusiones

Las metodologías ágiles para desarrollo de software son herramientas que ayudan a organizar, administrar y controlar mejor las actividades que se llevan a cabo para la implementación de un sistema o aplicación de software. Con la proliferación de empresas de consultoría y desarrollo de software en México, es primordial que una empresa tenga ventaja competitiva en el mercado de las tecnologías de la información y comunicación con respecto a las demás, para irse posicionando en el medio y posteriormente convertirse y/o mantenerse como líder en soluciones que se ofrecen a los clientes, realizando el trabajo dentro del tiempo y costo planeado, cumpliendo con las necesidades y expectativas de los usuarios finales y entregando productos de calidad. Como dice Henrik Kniberg, “lo importante no es la herramienta con que se empieza, sino la forma en que se mejora el uso de esta y además se expande el conjunto de herramientas con el tiempo”, por tal motivo en este trabajo se ha propuesto el uso de las mejores prácticas de dos metodologías de desarrollo ágil, Scrum y Kanban bajo la norma MoProsoft para empresas mexicanas y de esta manera competir en proyectos de negocios globalizados con cualquier empresa del mundo.

## Referencias

1. H. Oktaba, et al.: MoProSoft: Process Model to Industry Software, Ministry of Economy, Mexico (2005).
2. M. Dávila Muñoz, H. Oktaba: Especialización de MoProSoft basada en el Método Ágil Scrum (MPS-Scrum),
3. H. Kniberg , M. Skarin: Kanban and Scrum - making the most of both, USA (2009)

4. H. Kniberg: Kanban kick-start example, Noviembre 2009. [En línea]. Disponible: <http://www.crisp.se/gratis-material-och-guider/kanban>. [Último acceso: 25 octubre 2013].
5. M. Ovais Ahmad, J. Markkula, M. Ovio: Kanban in software development: A systematic literature review, 2013 39th Euromicro Conference Series on Software Engineering and Advanced Applications, (2013).
6. D. J. Anderson: Kanban: Successful Evolutionary Change for Your Technology Business. Sequim, WA: Blue Hole Press, (2010).
7. The Standish Group International: CHAOS Manifesto 2013, 2013. [En línea]. Disponible: <http://www.standishgroup.com>, (2013).