

# Q-Scrum: una fusión de Scrum y el estándar ISO/IEC 29110

Ariel Pasini<sup>1</sup>, Silvia Esponda<sup>1</sup>, Marcos Boracchia<sup>1</sup>, Patricia Pesado<sup>1,2</sup>

<sup>1</sup>Instituto de Investigación en Informática LIDI (III-LIDI),

Facultad de Informática, UNLP, 50 y 120, La Plata, Buenos Aires, Argentina

<sup>2</sup> CIC (Comisión de Investigaciones Científicas de la Provincia de Bs. As.), Argentina

{apasini, sesponda,marcosb,ppesado}@lidi.info.unlp.edu.ar

**Abstract.** Se realiza una comparación de la metodología de desarrollo ágil Scrum con los requerimientos del estándar ISO/IEC 29110. Se analizan las competencias de los roles ISO/IEC 29110 versus los roles de la metodología Scrum, los documentos que exige el estándar ISO/IEC 29110 contra los documentos definidos por Scrum y las actividades definidas en el estándar ISO/IEC 29110 respecto las de Scrum. Se presenta Q-Scrum, el modelo propuesto que permite una aproximación de los desarrollos en Scrum a los requerimientos del estándar ISO/IEC 29110, fusionando los roles, documentos y actividades de ambos modelos.

**Keywords:** Ingeniería de Software -Calidad – PyMEs - Metodologías ágiles - Scrum – ISO/IEC 29110

## 1 Introducción

Las metodologías ágiles representan una alternativa para el desarrollo de sistemas de software, centrada en el factor humano y el producto software, valorizando la relación con el cliente y el desarrollo incremental del software. Estas metodologías ofrecen entregas frecuentes de software funcional, permitir cambios de requerimientos y participación directa del cliente en el desarrollo. Una de estas metodologías es Scrum [1], que se define como un proceso iterativo incremental y empírico para administrar y controlar el trabajo de desarrollo. Actualmente Scrum es la metodología más utilizada en PyMEs desarrolladoras de software. La decisión de implantar metodologías de desarrollo, indica que la organización ha adquirido experiencia y se encuentra en un proceso de madurez que es de esperar, se afiance con el tiempo [2][3][4].

El estándar ISO/IEC 29110 Perfil Básico es un conjunto de buenas prácticas en el desarrollo del software para asistir y evaluar a las PyMEs desarrolladoras de software en el proceso de mejora. Está compuesto del Proceso de Administración de Proyecto (AP) y del Proceso de Implementación Software (IS), cada uno de ellos posee un

conjunto de roles, actividades y documentos externos, que se deben satisfacer al momento de evaluar el estado de los procesos [5].

En el camino de obtener mejor calidad en las empresas desarrolladoras de software de pequeño y mediano porte, surge la necesidad de compatibilizar la utilización de metodologías ágiles tipo Scrum y estándares de buenas prácticas como ISO/IEC 29110. Sin embargo la estructura y documentación definida por Scrum para sus desarrollos es insuficiente para satisfacer los requisitos del estándar ISO/IEC 29110, por lo cual es necesario desarrollar un nuevo modelo.

Q-Scrum es una propuesta de modelo orientada a PyMES, que proporciona una estructura de roles, documentos y actividades capaces de satisfacer el estándar, con la idea que las empresas la puedan usar como punto de partida en la mejora de sus procesos de desarrollo.

En la sección 2 y 3 se presenta una breve descripción de Scrum y el estándar ISO/IEC 29110, respectivamente, haciendo hincapié en los roles, documentos y actividades, que luego serán relacionados y comparados en la sección 4. En la 5 se presenta Q-Scrum, como una adaptación de Scrum para satisfacer los requisitos del estándar ISO/IEC 29110. Por último se presentan las conclusiones obtenidas del presente trabajo en la sección 6.

## 2 Scrum

Scrum es un marco de trabajo ágil para desarrollo de software. El trabajo se organiza en ciclos llamados sprints que son iteraciones de corta duración, típicamente de 2 a 4 semanas. Durante cada sprint, el equipo selecciona un conjunto de requerimientos de una lista priorizada, de manera que las funciones desarrolladas al principio del proyecto son las de más alto valor. Al final de cada sprint se entrega un producto de software ejecutable en el ambiente requerido por el cliente. No es un



Fig. 1. Scrum

proceso prescriptivo, no describe qué hacer en cada circunstancia, sólo ofrece un marco de trabajo y un conjunto de prácticas que mantienen todo visible y guían los esfuerzos para obtener el resultado más valioso posible.

Scrum coloca todas sus prácticas en un proceso con estructura iterativa e incremental. Esto se muestra en la figura 1, donde el lazo mayor representa una

iteración, que se repite en el tiempo, y que abarca las actividades de desarrollo. La salida de la iteración es un incremento del producto. El lazo más pequeño representa la inspección diaria que tiene lugar durante la iteración, en la cual los miembros del equipo se reúnen para inspeccionar las actividades de todos los miembros y hacer las adaptaciones apropiadas. La iteración es dirigida por la lista de requerimientos (Product backlog). Este ciclo se repite hasta que finalice el proyecto.

Al comienzo de cada iteración, el equipo revisa lo que debe hacer y selecciona lo que cree que se puede convertir en un incremento de la funcionalidad potencialmente entregable. Luego el equipo trabaja haciendo su mejor esfuerzo en el resto de la iteración. Al final de la iteración, el equipo presenta el incremento de la funcionalidad, el cual es construido de manera tal que los involucrados puedan inspeccionar la funcionalidad y oportunamente hacer adaptaciones al proyecto [6] [7],[8].

Las tablas 1 y 2 presentan los roles y documentos que se utilizan en Scrum

<b>Rol</b>	<b>Competencia</b>
Product Owner	El PO representa a quien tiene un interés en el proyecto y el producto resultante. Sus principales responsabilidades son: definir los requerimientos del producto a desarrollar durante el proyecto, ajustar los requerimientos y prioridades a lo largo de todo el proyecto, aceptar o rechazar el producto de software.
Scrum Master	El SM es el líder que facilita el trabajo. Es responsable del proceso de Scrum, de ser necesario enseñándolo a cada uno de los involucrados en el proyecto. Se asegura de que cada uno sigue las reglas y prácticas de Scrum. Sus principales responsabilidades son conducir la reunión Daily Scrum (DS), conocer el estado de las tareas, identificar barrera y dependencias que impidan el flujo de Scrum y observar y resolver conflictos personales.
Scrum Team	El Equipo es interdisciplinario y con 7±2 integrantes que son los encargados de conocer cómo convertir los requerimientos en un incremento de la funcionalidad y de desarrollar dicho incremento.

**Table 1.** Roles Scrum

<b>Documentos</b>	<b>Descripción</b>
Product Backlog.	Es un documento de alto nivel para todo el proyecto. Contiene descripciones genéricas de todos los requerimientos, funcionales y no funcionales, contiene estimaciones realizadas a grandes rasgos, tanto del valor para el negocio, como del esfuerzo de desarrollo requerido, la prioridad de las diferentes tareas, etc. Es dinámico, nunca está completo, evoluciona junto con el producto

Sprint Backlog.	Documento detallado que contiene las tareas que el Team va a implementar durante el presente sprint.
-----------------	--

**Table 2.** Documentos Scrum

### 3 ISO/IEC 29110

La industria de software en PyMEs creció exponencialmente en los últimos años, pero carecía de estándares o modelos de mejora que tuviesen en cuenta la estructura y capacidad interna de las mismas. Con la intención de ayudar a este sector, ISO a través del SC7-WG24, inició su trabajo para lograr que sus estándares de procesos de software (o adaptaciones de éstos) se pudieran aplicar a pequeñas y medianas empresas desarrolladoras de software.

Este grupo estableció un marco común para describir perfiles evaluables del ciclo de vida de software para uso en PyMEs.

La norma define tres perfiles: Perfil Básico, Perfil Intermedio y Perfil Avanzado. El primero de los perfiles se ha publicado en el año 2010 bajo el nombre de ISO/IEC 29110 Perfil Básico, los otros aún permanecen en desarrollo.

El Perfil Básico está compuesto del **Proceso de Administración de Proyecto (AP)**, con el objetivo de establecer y llevar a cabo de manera sistemática las tareas de los proyectos de implementación de software, cumplir con los objetivos del proyecto en calidad, tiempo y costo esperados, y del **Proceso de Implementación Software (IS)**, con el propósito de asegurar la realización sistemática de las actividades de análisis, diseño, construcción, integración y pruebas de productos de software, nuevos o modificados de acuerdo a los requisitos especificados [5].

Cada uno de estos procesos, está compuesto por un conjunto de actividades, roles y documentos que deben ser contemplados para la ejecución del mismo. La descripción de los roles y documentos se presentan en las tablas 3 y 4 respectivamente, que serán utilizados para el análisis realizado en este trabajo.

<b>Rol</b>	<b>Competencia</b>
Cliente	Conocimiento de los procesos del cliente y capacidad de explicar los requisitos del cliente. Tiene la facultad de aprobar los requisitos y sus cambios. Conocimiento y experiencia en el dominio de aplicación
Líder de Proyecto	Capacidad de liderazgo. Experiencia en planificación, gestión de personal, delegación y supervisión, finanzas y desarrollo de software.
Equipo	Conocimiento y experiencia de acuerdo a su función en el proyecto. Conocimiento de las normas utilizadas por el cliente y/o por la PyMEs.
Analista	Conocimiento y experiencia de elicitar, especificar y analizar los requisitos. Conocimiento en diseño de interfaces de usuario y criterios ergonómicos. Conocimiento de las técnicas de revisión. Experiencia en el desarrollo y mantenimiento de software.
Desarrollador	Conocimiento y experiencia en los componentes de software y diseño de la arquitectura. Conocimiento de las técnicas de revisión. Conocimiento y experiencia en la planificación y realización de pruebas de integración.

	Experiencia en el desarrollo y mantenimiento de software.
Programador	Conocimiento y / o experiencia en la programación, integración y pruebas unitarias Conocimiento de las técnicas de revisión. Experiencia en el desarrollo y mantenimiento de software

**Table 3.** Roles ISO 29110

Documentos	Descripción
Declaración de trabajo	<b>Descripción del producto</b> , contiene: Propósito. Requerimientos generales. Alcance. Objetivos. Entregables.
Configuración del Software	<b>Identificación de conjunto de productos de software que se deben mantener actualizados</b> , contiene: Especificación de requerimientos, Diseño de software, Registro de trazabilidad, Software, Componentes, Casos de prueba, Reportes de pruebas, Manual de usuario, Documentación de mantenimiento.
Solicitud de cambio	<b>Documentación que identifica las solicitudes de cambios</b> , contiene: Propósito, estado de la solicitud, solicitante, impacto.
Plan de Proyecto	<b>Descripción de cómo el proyecto y sus actividades serán ejecutadas</b> , contiene: Descripción del producto, Propósito, Requerimientos generales, Alcance, Objetivos, Entregables, Tareas, Estimación de tiempo/costo/duración, Composición del equipo de trabajo, Riesgos.
Registro de aceptación	<b>Documento que establece la aceptación de los entregables por el cliente</b> , contiene: Registro de recepción de entregable, Fecha de recepción, Criterios de aceptación.
Minutas de reunión	<b>Registro de acuerdos establecidos con el cliente y/o equipo de trabajo</b> , contiene: Propósito de la reunión, asistentes, fecha, logros, cuestiones planteadas.

**Table 4.** Documentos ISO 29110

## 4 Roles, documentos y actividades de Scrum e ISO/IEC 29110

### 4.1 Roles

Analizando las competencias de los roles entre ISO/IEC 29110 y Scrum, se observa una relación directa para *Cliente*, *Líder de Proyecto* y *Equipo*, pero el resto de los roles del estándar están contemplados en las competencias del Scrum Team. En Scrum no se diferencian ni especifican las funciones de los integrantes del equipo, por lo cual los roles de *Analista*, *Desarrollador* y *Programador* no se pueden relacionar de forma directa. Ver tabla 5.

ISO 29110\ Scrum	Product Owner	Scrum Master	Scrum Team
Cliente	X		
Lider del Proyecto		X	

<b>Equipo</b>	X
<b>Analista</b>	*
<b>Desarrollador</b>	*
<b>Programador</b>	*

**Table 5.** - Comparación de roles

## 4.2 Documentos

Análogamente al análisis de roles, se estableció una comparación entre los documentos que solicita el estándar y los que utiliza la metodología. El estándar es muy riguroso en la definición de los documentos mientras que Scrum los maneja informalmente, por lo que, en algunos casos se podría establecer una mayor relación, dependiendo de la manera que sea construido.

En la tabla 6 se presentan las diferencias entre los documentos recomendados por el estándar y los que determina Scrum

<b>ISO/IEC 29110</b>	<b>Scrum</b>	<b>Observaciones</b>
Declaración de trabajo	Product Backlog	No posee una estructura definida, por lo tanto se puede acercar tanto como se desee al producto en cuestión.
Configuración del Software		Cada uno de los elementos de la configuración del software representan un producto en la norma, que no se corresponde con ningún artefacto en la metodología Scrum.
Solicitud de cambio	Sprint Backlog	Dado que en cada sprint puede incorporar modificaciones/mejoras en los requerimientos, es posible considerarlo una solicitud de cambio.
Plan de Proyecto	Product Backlog	No posee una estructura definida, por lo tanto se puede acercar tanto como se desee al producto en cuestión.
Registro de aceptación		Scrum no presenta un documento formal para registrar la aceptación de productos, pero en la práctica se deja constancia informal.
Minutas de reunión		Scrum no presenta un documento formal para registrar las minutas, pero se deja constancia de las reuniones Daily Scrum.

**Table 6.** - Comparación de documentos

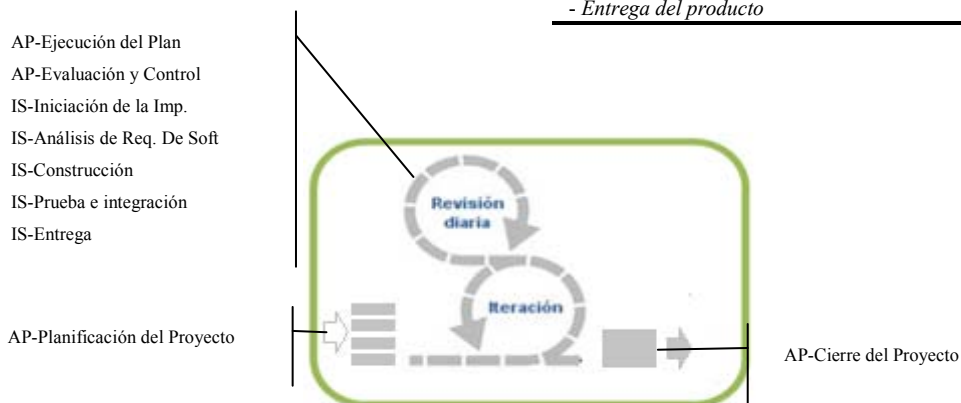
## 4.3 Actividades

Como se mencionó anteriormente, ISO/IEC 29110 presenta dos grandes procesos (AP e IS) que abarcan todas las actividades a realizar durante el ciclo de vida de un proyecto.

En el proceso AP, las actividades de la etapa **Planificación de Proyecto**, equivalen a la recepción del **Product Owner** con la lista de requerimientos que se utiliza para

crear el **Product Backlog**. Las actividades de las etapas **Ejecución del plan de proyecto** y **Evaluación y Control del Proyecto** se relacionan con la ejecución del Sprint y las de la etapa **Cierre del Proyecto**, son equivalentes a la entrega final del proyecto. El Proceso de IS define actividades que están directamente ligadas al Sprint.

Actividades AP	Actividad IS
<b>Planificación del Proyecto</b>	<b>Iniciación de la Implementación</b>
<ul style="list-style-type: none"> <li>- Revisar la Declaración de trabajo</li> <li>- Establecer tareas a realizar con dependencia y duración</li> <li>- Establecer puntos de V&amp;V</li> <li>- Definir equipo de trabajo con roles y responsabilidades</li> <li>- Definir capacitaciones</li> <li>- Estimar esfuerzo, costo y calendario</li> <li>- Identificar Riesgos</li> </ul>	<ul style="list-style-type: none"> <li>- Revisar el Plan de Proyecto con el equipo de trabajo y establecer tareas a realizar</li> <li>- Establecer el compromiso del equipo y el Lider</li> <li>- Establecer el ambiente de Implementación</li> </ul>
<b>Ejecución del Plan de Proyecto</b>	<b>Análisis de Requerimientos de Soft</b>
<ul style="list-style-type: none"> <li>- Registrar el progreso del proyecto</li> <li>- Analizar y evaluar los cambios y su impacto.</li> <li>- Aprobar los cambios en el Plan.</li> <li>- Mantener reuniones con el equipo de trabajo y el cliente.</li> <li>- Actualizar el Repositorio</li> </ul>	<ul style="list-style-type: none"> <li>- Revisar tareas asignadas</li> <li>- Elicitar, analizar y especificar requerimientos</li> <li>- V&amp;V los requerimientos</li> <li>- Control de versiones</li> </ul>
<b>Evaluación y Control del Proyecto</b>	<b>Arquitectura y Diseño Detallado del software</b>
<ul style="list-style-type: none"> <li>- Evaluar el progreso del Plan</li> <li>- Identificar y evaluar desviaciones y problemas de costo, calendario, técnicos.</li> <li>- Documentar cambios y acciones correctivas.</li> <li>- Actualizar el Repositorio</li> </ul>	<ul style="list-style-type: none"> <li>- Diseñar arquitectura. Componentes</li> <li>- Rever especificación de requerimientos</li> <li>- Verificar Diseño y casos de prueba</li> <li>- Control de versiones</li> </ul>
<b>Cierre del Proyecto</b>	<b>Construcción</b>
<ul style="list-style-type: none"> <li>- Realizar la entrega del producto según lo acordado.</li> <li>- Realizar soporte al cliente</li> <li>- Finalizar el proyecto y firmar aceptación.</li> </ul>	<ul style="list-style-type: none"> <li>- Rever el diseño para determinar secuencia de construcción.</li> <li>- Codificar.</li> <li>- Trazabilidad.</li> </ul>
	<b>Prueba e integración</b>
	<ul style="list-style-type: none"> <li>- Integrar componentes</li> <li>- Realizar pruebas y documentar</li> <li>- Verificar líneas base</li> </ul>
	<b>Entrega</b>
	<ul style="list-style-type: none"> <li>- Controlar Documentación</li> <li>- Entrega del producto</li> </ul>



**Fig. 2.** Relación entre las actividades de Scrum y el estándar

En la figura 2 se observan las actividades de las etapas de cada proceso del estándar, y en cursiva se representan las actividades que son realizadas en Scrum.

## 5 Modelo Q-Scrum

Se presenta Q-Scrum, un modelo de procesos basado en Scrum, que integra un conjunto de roles, documentos, y actividades, capaz de generar los documentos necesarios para satisfacer los requisitos del estándar ISO/IEC 29110.

### 5.1 Roles

En la tabla 5 de la sección 4.1 se aprecia que con los roles de Scrum no es posible satisfacer los requisitos del estándar. Por lo que será necesario redefinir la estructura de roles. Q-Scrum propone crear un nuevo rol, Q-Scrum Analyst, que contemple las competencias de análisis y documentación que claramente son realizadas específica y separadamente de las otras competencias, manteniendo en el Q-Scrum Team, las competencias de implementación (tanto de desarrolladores como programadores).

	Rol	Competencias
QPO	Q-Product Owner	Product Owner / Cliente
QSM	Q-Scrum Master	Scrum Master / Líder de Proyecto
QST	Q-Scrum Team	Scrum Team / Equipo- Programadores – Desarrolladores
QSA	Q-Scrum Analyst	Analista

**Table 7.** Roles Q-Scrum

### 5.2 Documentos

En 4.2 se presentaron las diferencias entre los documentos de Scrum y el estándar, donde es evidente que la documentación generada por Scrum es insuficiente para satisfacer el estándar. Q-Scrum propone estructurar los documentos Product Backlog y Sprint Backlog, sin perder la flexibilidad de Scrum, formalizar los documentos de Registro de Aceptación y Minutas de reunión, que son utilizados habitualmente en desarrollos Scrum, e incorporar el documento de Configuración del software.

	Q-Scrum	Observaciones
dQPB	Q-Product Backlog	Plantilla básica, con la información requerida por la descripción de trabajo y el plan de proyecto del estándar, que inicialmente se completó con la información básica para iniciar el proyecto y se fue actualizando a lo largo del desarrollo.
dQSC	Q-Software Configuration	Documento donde se irán incorporando todos los registros de las actividades realizadas.

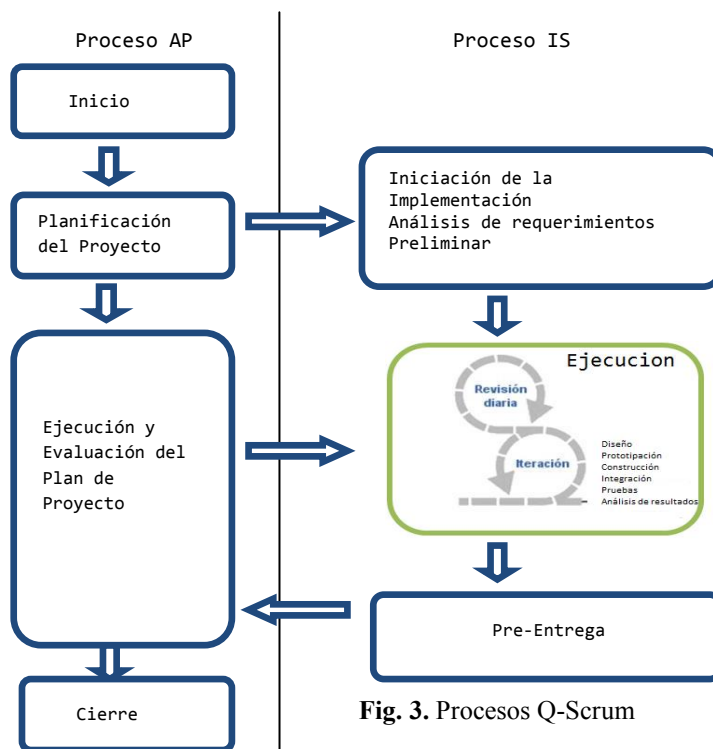
dQSB	Q-Sprint Backlog	Plantilla básica, que incluye las solicitudes de cambio, donde se incorporarán los requerimientos de cada sprint.
dQAR	Q-Accepted Record	Plantilla básica donde se registrará la aceptación de los productos
dQMR	Q-Meeting Record	Plantilla básica donde se registrarán las decisiones de las reuniones como por ejemplo las reuniones Daily Scrum.

**Table 8.** Documentos Q-Scrum

### 5.3 Actividades

En base a las relaciones entre las actividades descritas en 4.3, Q-Scrum propone modificar los procesos de AP y IS para soportar la nueva estructura de roles y generar/mantener los documentos de Q-Scrum.

El proceso de AP quedaría compuesto por las etapas **Inicio**, **Planificación de Proyecto**, **Ejecución y Evaluación de proyecto** y **Cierre**. El Proceso de IS con las etapas **Iniciación de la Implementación y Análisis de requerimientos preliminar**, **Ejecución** y **Pre-Entrega**. La Figura 3 presenta la estructura y relación de los procesos del modelo propuesto.



**Fig. 3.** Procesos Q-Scrum

## Procesos AP.

### *Inicio.*

Se recibe una petición del QPO, el QSM da inicio al proyecto generando el dQPB, se asigna un QSA.

### *Planificación del Proyecto.*

El QSM y el QSA tomarán el dQPM preliminar y analizarán la factibilidad, riesgos, tareas a realizar, dependencias, duración, puntos de VyV, estimarán esfuerzos, costos y calendario. En base a eso asignarán un QST. El QSM actualizará el sQPM y el QSA generará el dQCS.

El QST ya iniciará el proceso de IS.

### *Ejecución y Evaluación del Plan de Proyecto.*

El QSM y QSA registrarán y evaluarán el progreso del proyecto, analizarán posibles cambios, correcciones de costos y calendarios, ante cada spring.

El QSA registrará los cambios en el dQCS.

### *Cierre.*

El QSM registrará la entrega final con un dQAR y finalizará el proyecto.

## Proceso de IS.

### *Iniciación de la Implementación y Análisis de requerimientos preliminar.*

El QSM y el QST realizarán la primera reunión generando el dQSB del primer sprint. El QSA formalizará la reunión en un dQMR.

### *Ejecución.*

Cada iteración del Sprint realizará las siguientes sub-etapas:

<b>Actividad</b>	<b>Rol</b>	<b>Registro</b>
<b>Análisis</b>		
- Revisar tareas asignadas	QSM	dQPB y dQMR
- Elicitar, analizar y especificar requerimientos	QSA	dQSB
- V&V de los requerimientos	QSM	dQPB y dQCS
<b>Diseño y Prototipado</b>		
- Diseñar arquitectura. Componentes	QST	dQSB y dQPB
- Rever especificación de req	QSM	dQSB y dQPB
- Verificar Diseño y casos de prueba	QSM	dQPB y dQCS
<b>Construcción</b>		
- Codificar	QST	
- Trazabilidad	QSA	dQCS
<b>Prueba e integración</b>		

-Integrar componentes	QST	dQSB
- Realizar pruebas y documentar	QST	dQSB
- Verificar líneas base	QSA	dQCS
<b>Pre – Entrega</b>		
-Controlar Documentación	QST	dQCS
-Pre - Entrega del producto	QST	dQCS

**Entrega.**

Cerrada la iteración, el QSM registrará la entrega en un dQAR y el QSA actualizará el dQCS.

## 6 Conclusiones

Se presentó una descripción de Scrum y del estándar ISO/IEC 29110 y una relación entre ellos en función de los roles, documentos y actividades, concluyendo que la metodología Scrum aplicada de forma directa no es capaz de satisfacer los requisitos del estándar.

Se propone Q-Scrum, agregando a Scrum el rol analista (QSA), incorporando un documento para la gestión de la configuración del software (dQSC) y formalizando las minutas (dQMR) y los documentos de aceptación (dQAR). Se fusionaron las actividades de Scrum con los procesos de AP e IS, generando un modelo de procesos capaz de satisfacer los requisitos del estándar.

Se ha iniciado la aplicación del modelo Q-Scrum en PyMEs de la región, que desarrollan bajo la metodología Scrum y poseen intenciones de lograr una mejora de proceso. La retroalimentación recibida de estas experiencias permitirá ajustar el modelo propuesto.

## 7 Referencias

- [1] Comunidad Latinoamericana de Metodologías Ágiles <http://www.agiles.org> Julio 2013
- [2] Muñoz, Oktaba, “Especialización de MoProSoft basada en el método ágil Scrum”, Editorial académica Española Año 2011
- [3] A. Pasini, S. Esponda, P. Pesado and R. Bertone., Aseguramiento de calidad en PYMES que desarrollan software. una experiencia desde el proyecto COMPETISOFT. 2008. pp. 957-966.
- [4] Piattini, Oktaba, Orozco, “COMPETISOFT. Mejora de procesos software para pequeñas y medianas empresas”, Editorial Ra-Ma, Año 2008
- [5] ISO/IEC 29110:2011, “Software engineering -- Lifecycle profiles for Very Small Entities (VSEs)” 2011, ISO

[6] Henrik Kniberg “Scrum y xp desde las trincheras” (2007) Libro Online InfoQ <http://www.infoq.com/minibooks/scrum-xp-from-the-trenches>, julio 2013

[7] Schwaber, Ken Agile Project Management with scrum, Redmon Wshington: Microsoft Press 2004

[8]Pablo Lledó , Gestión Ágil de Proyectos – Pablo Lledó. ISBN: 978-1-4669-2119-1, Trafford Published, 2012.