

MODELOS DE PROCESOS, SELECCIÓN DEL ESTANDAR ADECUADO PARA NUESTRA ORGANIZACIÓN.

MÁSTER UNIVERSITARIO EN
INVESTIGACIÓN EN INGENIERÍA DE
SOFTWARE Y SISTEMAS
INFORMÁTIVOS. INGENIERÍA DE
SOFTWARE, 31105128

Shahin Alawi Barragán - María Magdalena Arcilla Cobián

2014/2015 Septiembre

Modelos de Procesos, Selección del Estándar Adecuado para Nuestra Organización.

Máster Universitario en Investigación en Ingeniería de Software y Sistemas Informáticos. Ingeniería de Software. Código 31105128

Shahin Alawi Barragán - María Magdalena Arcilla Cobián

Tipo A

AUTORIZACIÓN

Autorizo/amos a la Universidad Nacional de Educación a Distancia a difundir y utilizar, con fines académicos, no comerciales y mencionando expresamente a sus autores, tanto la memoria de este Trabajo Fin de Máster, como el código, la documentación y/o el prototipo desarrollado.

Firma del/los Autor/es

RESUMEN

Los modelos de procesos en empresas relacionadas con el sector TIC son fundamentales para ofrecer a sus clientes productos competitivos y de calidad. Hoy en día las organizaciones necesitan adaptarse a las nuevas tendencias tecnológicas, por lo que toman especial relevancia las prácticas llevadas a cabo en cuanto a la gestión y mejora de procesos software para lograr mantener unos estándares de calidad competitivos. Veremos los dos grupos de metodologías que hoy en día se usan, tradicionales y ágiles. Concretamente hablaremos de CMMI-DEV, ITIL y SPICE como metodologías tradicionales y hablaremos de KANBAN, SCRUM y XP como metodologías ágiles. Mencionaremos las características generales de estos modelos y las principales características de los dos tipos de metodologías, ayudándonos en la selección del modelo más adecuado para nuestra organización. Además, tendremos que tener en cuenta una serie de aspectos importantes para apoyar nuestra selección de modelo de procesos, tales como la definición del contexto de la organización o conocer la amplia variedad de modelos de procesos existentes. Introduciremos el concepto de la combinación de modelos, practica cada vez más extendida para lograr gestionar todo tipo de proyectos en una misma organización, ya que ningún modelo de proceso puede gestionar por si solo todo tipo de proyectos obteniendo las mismas ventajas que logramos al combinar distintos modelos de procesos.

PALABRAS CLAVE

CMMI-DEV, gestión y mejora, ITIL, Kanban, metodologías ágiles, modelos de procesos, procesos software, Scrum, selección de modelos, SPICE, XP.

EXECUTIVE SUMMARY

The process models in companies related to the IT sector are essential to provide customers competitive and quality products. Today organizations need to adapt to new technological trends, so take special relevance practices carried out in terms of management and process improvement software in order to maintain a competitive quality standards. We will see both sets of methodologies that are used today, traditional and agile. Specifically we talk about CMMI-DEV, SPICE and ITIL as traditional methodologies and discuss Kanban, Scrum and XP as agile methodologies. We mention the general characteristics of these models and the main characteristics of the two kinds of methodologies, helping in selecting the most appropriate model for our organization. In addition, we must take into account a number of important aspects to support our model selection process, such as defining the context of the organization meet the wide variety of models existing processes. We introduce the concept of combining models increasingly widespread practice to achieve manage all types of projects in the same organization, since no process model alone can manage all types of projects getting the same benefits we achieved by combining different process models.

KEY WORDS

CMMI-DEV, management and improvement, ITIL, Kanban, agile methodologies, process models, software processes, Scrum, model selection, SPICE, XP.

INDICE

RESUMEN	4
EXECUTIVE SUMMARY	4
ÍNDICE	5
LISTA DE FIGURAS Y TABLAS	8
1 INTRODUCCIÓN	9
1.1 Estructura	10
2 PLANTEAMIENTO DEL PROBLEMA	12
2.1 ¿Por qué usar modelos de procesos?	13
2.2 Selección del modelo de procesos adecuado	14
2.2.1 Modelo de procesos inadecuado	14
2.2.2 Desconocemos la variedad de modelos	15
2.2.3 Desconocemos la estimación de beneficios	17
3 ESTADO DE LA CUESTIÓN	19
3.1 Conceptos generales	20
3.1.1 Procesos software	20
3.1.2 Áreas de proceso	20
3.1.3 Modelos de procesos	21
3.1.4 Ingeniería software	21
3.2 El método en las organizaciones	22
3.2.1 Proyectos gestionados sin modelos de procesos	22
3.2.2 Proyectos gestionados con modelos de procesos	23
3.3 Metodologías tradicionales	24

3.3.1	CMMI	25
3.3.2	ITIL	32
3.3.3	SPICE, ISO 15504	39
3.4	Metodologías Ágiles	42
3.4.1	SCRUM	43
3.4.2	KANBAN	46
3.4.3	XP	48
3.5	Modelos Tradicionales vs Ágiles	52
4	RESOLUCIÓN	55
4.1	Siete aspectos importantes a tener en cuenta	56
4.1.1	Analizar la gestión actual en la organización ...	56
4.1.2	Definir los objetivos de la organización	57
4.1.3	Modelos de Proceso disponibles	58
4.1.4	Seleccionar el Modelo según los objetivos fijados	58
4.1.5	Formarse en el Modelo seleccionado	59
4.1.6	Compromiso con el método	60
4.1.7	La implantación del Modelo seleccionado	60
4.2	La selección del Modelo adecuado	61
4.3	La convivencia de Modelos	62
4.3.1	CMMI-DEV e ITIL	63
4.3.2	CMMI-DEV y SPICE	64
4.3.3	CMMI-DEV y SCRUM	64
4.3.4	ITIL y XP	65
4.3.5	SCRUM y KANBAN	65
4.3.6	SCRUM y XP	65
5	CONCLUSIONES Y LÍNEAS FUTURAS	67

BIBLIOGRAFÍA Y REFERENCIAS	69
ACRONIMOS	71

LISTA DE FIGURAS Y TABLAS

Tabla 1. Ámbito de estudio (pag. 15).

Tabla 2. Niveles CMM (pag. 16).

Figura 1. Conocimiento de estándares, normas oficiales y metodologías, según el tamaño de la organización, tanto en oferta como en demanda (%) (pag. 26).

Figura 2. Línea temporal sobre la historia de CMM (pag. 27).

Figura 3. Diagrama conceptual de la representación continua en la parte superior y por etapas en la parte inferior de la figura (pag. 29).

Figura 4. Diagrama conceptual de las áreas de proceso y la representación continua en la parte superior y por etapas en la parte inferior de la figura (pag. 30).

Figura 5. Línea temporal sobre la historia de ITIL (pag. 33).

Figura 6. Las 5 fases ITIL relacionadas entre sí (pag. 34).

Figura 7. Línea temporal sobre la historia de SPICE (pag. 40).

Figura 8. Esquema conceptual del proceso SCRUM (pag.46).

Figura 9. Muro de tareas KANBAN (pag.48).

Tabla 3. Características generales de 3 modelos tradicionales; CMMI-DEV, ITIL y SPICE (pag. 52).

Tabla 4. Características generales de 3 metodologías ágiles; SCRUM KANBAN y XP (pag. 53).

Tabla 5. Comparativa de las características generales de todos los modelos tradicionales y ágiles (pag. 54).

Figura 10. Ejemplo del cuestionario para ayudar en la selección del tipo de metodología (pag. 62).

Capítulo 1

Introducción

La selección del modelo de proceso más adecuado para nuestra organización, a priori, puede resultar difícil. En la actualidad existen un gran número de ellos y podría parecernos que no nos ofrecen suficiente información como para decantarnos por un modelo frente a otro, a pesar de que los modelos cuenten con una documentación detallada. También podríamos encontrarnos en la situación en que al seleccionar un modelo por los buenos resultados obtenidos al implantarlo en una empresa determinada, no obtengamos los resultados deseados ya que el contexto no es similar. Otro escenario que podría darse, es que al desconocer la existencia de un modelo que se ajusta a nuestras necesidades, eligiéramos uno que no fuese el más adecuado para conseguir nuestros objetivos.

El hecho de que exista una gran variedad de modelos, desde nuestro punto de vista, es positivo, aunque el verdadero problema lo encontramos a la hora de seleccionar el modelo o modelos adecuados para nuestra organización y la correcta ejecución. Para aplicar el modelo más adecuado en nuestra organización, debemos entender el contexto en el cual se aplicará y los objetivos que pretendemos conseguir con su aplicación. Podremos comprobar que no siempre necesitaremos un modelo de proceso al completo, sino que únicamente nos convendría aplicar partes de un modelo, y en ocasiones, en combinación con otras partes de otros modelos, obteniendo así los objetivos deseados.

Mediante este trabajo podremos ver que la convivencia entre distintos modelos no solo es posible, sino que en determinados escenarios es la mejor opción. Por ello, la convivencia de modelos nos permitirá aprovechar sus características cubriendo nuestras necesidades en cuanto a la gestión y mejora de procesos, sin tener que aplicar necesariamente cada una de las partes especificadas en dichos modelos. Además, tendremos la oportunidad de reunir varios modelos de procesos importantes con sus principales características, facilitando así una preselección de los mejores candidatos a implantar en una organización según el objetivo a alcanzar.

La correcta selección y ejecución de cualquier modelo, nos permitirá gestionar y mejorar todos los procesos involucrados en cualquier proyecto relacionado con las empresas TIC, con la seguridad de obtener los mejores resultados posibles frente a los proyectos gestionados sin una metodología bien definida y clara para las personas que colaboran en ellos. Con una gestión controlada minimizaremos el nivel de incertidumbre que nos encontramos a la hora de afrontar un nuevo proyecto, los principales aspectos se gestionarían de la mejor forma; organización, planificación, recursos humanos, proveedores, calidad, presupuestos, control, mejora continuada, etc.

1.1. Estructura

En los sucesivos capítulos desarrollaremos el planteamiento del problema, el estado de la cuestión, la resolución, la conclusión y la línea de trabajos futuros. El capítulo 2 contiene el planteamiento del problema, nos pone en situación en cuanto a la problemática existente al seleccionar un modelo de procesos para nuestra organización, explicando los motivos por los cuales es necesaria la implantación de un modelo de procesos y algunos motivos por los cuales la selección del modelo de procesos no se realiza adecuadamente, es decir, en base a unos criterios justificados.

El estado de la cuestión, que exponemos en el capítulo 3, describe la importancia del uso de metodologías y modelos en las organizaciones para gestionar proyectos, además de ver los dos tipos de métodos; tradicionales y ágiles. Veremos tres modelos de cada tipo de metodología como muestra de la variedad existente de modelos de procesos y metodologías. El capítulo 3 finaliza con la comparativa de características generales entre los tres métodos de cada tipo de metodología y una comparativa de características generales entre los dos tipos de metodologías.

El siguiente capítulo, el número 4, es la resolución al problema que planteamos en este trabajo y es donde describimos una serie de aspectos a tener en cuenta a la hora de seleccionar un modelo de procesos para nuestra organización de forma adecuada. También describimos algunas características que deberemos identificar en nuestra organización para seleccionar un tipo de modelo frente a otro (tradicionales o ágiles) e introduciremos el tema “el puzzle de modelos y metodologías”, que consiste en la combinación de métodos dentro de las organizaciones para gestionar y mejorar cualquier tipo de proceso y proyecto.

Por último, el capítulo 5 expone la conclusión obtenida al realizar este trabajo y la línea de trabajos futuros en la que enfocar próximas investigaciones.

Capítulo 2

Planteamiento del Problema

Cualquier organización relacionada con la industria del software que se precie, necesitará gestionar sus propios procesos de negocio **[1]** (producción, comercial, financiero o administrativo) de la mejor forma posible. Resulta contradictorio que un elemento tan importante en las empresas actuales no sea desarrollado en una gran parte de las compañías, de acuerdo a un proceso previsible, controlado, definido, repetible y gestionado, del mismo modo que ocurre en infinidad de otros sectores de negocio como la fabricación, la aeronáutica o la distribución. Es por ello que, los modelos de proceso juegan un papel tan importante en la gestión y mejora de los procesos software.

Este trabajo pretende ser una guía para ayudarnos a entender los principales modelos de proceso que se utilizan habitualmente y su correcto uso para alcanzar los mejores resultados. Además, debemos tener en cuenta que si no todos, casi todos los modelos de proceso que existen pretenden ser la solución para cualquier organización, aunque nosotros haremos algunas diferenciaciones que se deben tener en cuenta a la hora de seleccionar un modelo frente a otro.

2.1. ¿Por qué usar modelos de procesos?

Para gestionar cualquier proyecto que incluya procesos software, como puede ser el desarrollo de software, es necesario contar con un método que nos ayude a alcanzar nuestros objetivos de la mejor forma posible, reduciendo al mínimo el nivel de incertidumbre que nos encontramos al afrontar un nuevo proyecto. Es decir, es necesaria una metodología o modelo que nos ayude a **[2]**:

- Optimizar los procesos implicados en el proyecto.
- Usar métodos que guían en la planificación.
- Definir qué hacer, como y cuando.

Debemos tener en cuenta que el uso de una metodología no es garantía en la conquista de nuestros objetivos, por lo que necesitamos como punto de partida contar con algunos aspectos importantes:

- La selección del Modelo de Proceso adecuado.
- La correcta aplicación del Modelo seleccionado.
- Medios materiales (tecnología, herramientas, etc).
- Recursos humanos.
- Compromiso de todo el personal implicado.
- Objetivos claros y alcanzables.

Al aplicar una metodología de forma adecuada, es decir, teniendo en cuenta los aspectos antes mencionados, desde el punto de vista de la gestión obtendremos como ventajas:

- Facilitar la tarea de planificación.
- Facilitar la tarea del control y seguimiento de un proyecto.
- Mejorar la relación coste/beneficio.
- Optimizar el uso de recursos disponibles
- Facilitar la evaluación de resultados y cumplimiento de los objetivos
- Facilitar la comunicación efectiva entre usuarios/clientes y desarrolladores

Desde el punto de vista de la ingeniería de software las ventajas al aplicar una metodología son:

- Ayudar a la comprensión del problema
- Optimizar el conjunto y cada una de las fases del proceso de desarrollo
- Facilitar el mantenimiento
- Permitir la reutilización

Desde el punto de vista del cliente o usuario:

- Garantía de un determinado nivel de calidad en el producto final
- Confianza en los plazos de tiempo fijados en la definición del proyecto
- Definir el ciclo de vida que más se adecue a las condiciones y características del producto.

2.2. Selección del modelo de procesos adecuado

Nosotros fijaremos el foco del trabajo en la selección del modelo de proceso más adecuado para nuestra organización. Pero para poder seleccionar el modelo de proceso adecuado no basta con elegir un modelo al azar e implantarlo, antes necesitamos realizar en la organización un análisis de su situación en cuanto a la gestión de las áreas de proceso que deseamos mejorar, en el que nos apoyaremos para poder seleccionar el modelo de procesos. Es el principal motivo por el cual es tan importante hacer un estudio franco que describa la realidad de las áreas de proceso y sus procesos dentro de la organización antes de poder seleccionar e implantar cualquier modelo.

La selección de un modelo de procesos podría llegar a ser un problema para nuestra organización si:

2.2.1. Modelo de procesos inadecuado

Esta situación se da cuando seleccionamos un modelo de proceso como si se tratase del culto a una metodología en vez de tomar una decisión técnica en la que basar nuestra decisión al seleccionar un modelo [3]. Muchas empresas ni siquiera tratan de evaluar los métodos sino que se limitan a adoptar el modelo de proceso más popular, por lo tanto, seleccionan modelos inadecuados para la organización.

La existencia de más de 50 metodologías de desarrollo de software, cada una de ellas con sus leales seguidores, nos indica claramente que ninguna de estas metodologías es capaz de manejar todos los tamaños y tipos de aplicaciones de software. Algunos modelos funcionan mejor para aplicaciones pequeñas y equipos pequeños, otros modelos trabajan bien para los grandes sistemas y grandes equipos, algunos van bien para desarrollar aplicaciones embebidas complejas y otros modelos funcionan bien para las aplicaciones militares de alta seguridad.

¿Cómo es posible seleccionar la mejor metodología para proyectos específicos?

2.2.2. Desconocemos la variedad de modelos

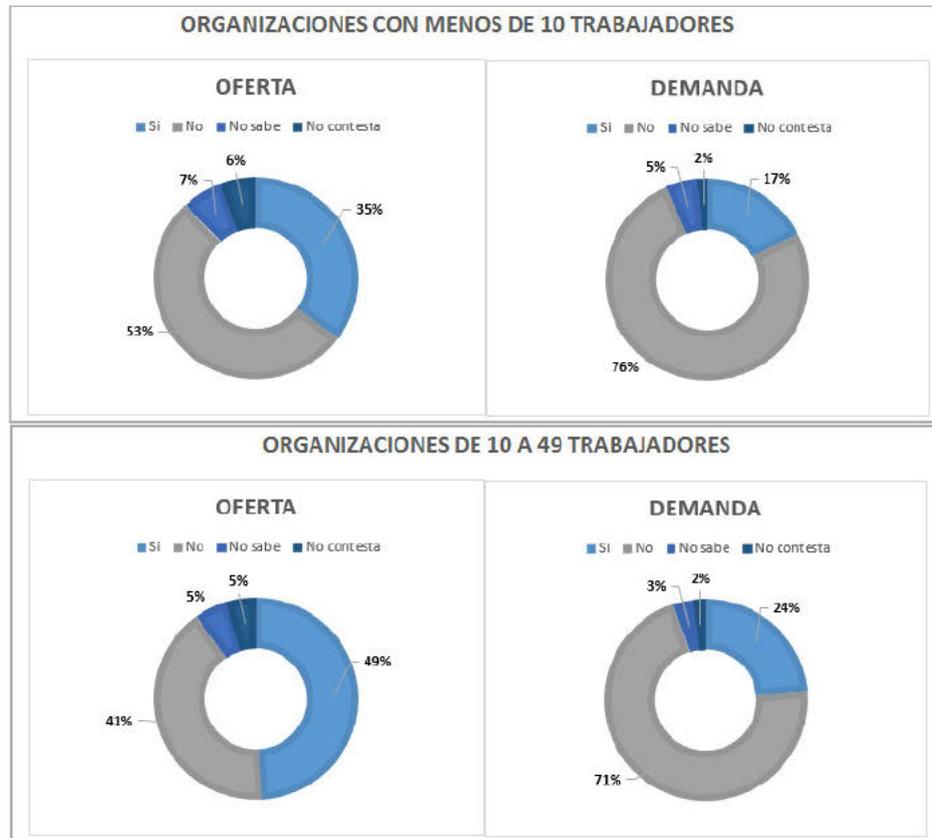
Un estudio [4] que se hizo en el año 2008, ofrecía una visión del estado de salud de las empresas españolas a nivel de calidad del software, en el uso y conocimiento tanto de metodologías de trabajo, como de herramientas, desde la perspectiva de la oferta y la demanda.

Ámbito	Oferta	Demanda
Sectorial	Actividades informáticas Investigación y desarrollo	Industria Construcción Servicios Administración pública, defensa y seguridad social obligatoria Otras actividades profesionales
Profesional	Ingenieros superiores Profesionales de la informática de nivel superior, medio y técnico	Responsables de compras

Tabla 1. Ámbito de estudio.

Del análisis de los resultados de la investigación se deduce un amplio desconocimiento de las propuestas metodológicas existentes. Así, dos de cada

tres encuestados de la demanda (64,8%) y uno de cada tres de la oferta (37,6%) afirman no tener conocimiento alguno sobre los estándares, las normas oficiales ni las metodologías orientadas a la calidad.



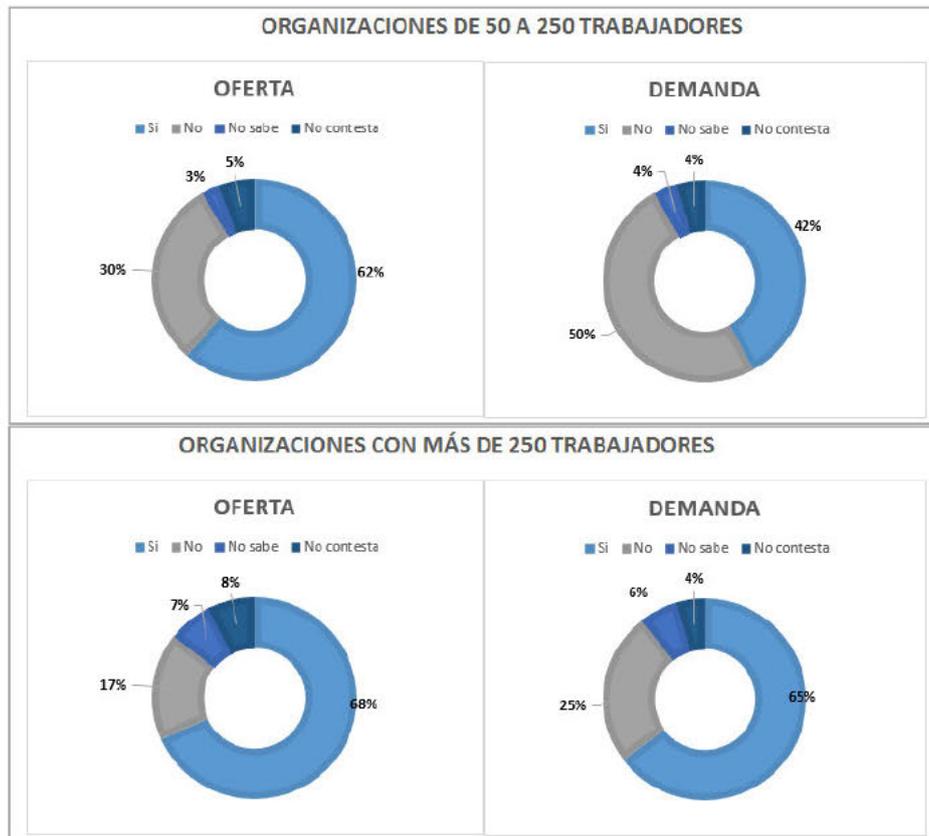


Figura 1. Conocimiento de estándares, normas oficiales y metodologías, según el tamaño de la organización, tanto en oferta como en demanda (%).

El modelo con un mayor grado de penetración en las pymes es CMMI (7,9%), al igual que en la gran empresa, donde alcanza el 23.9%. Las demás metodologías, modelos y estándares (ITIL, RUP, IEEE, etc.) no suponen un porcentaje representativo, de modo que en algunos casos no llegan a representar más del 1% del total de los resultados cuantitativos obtenidos. Es decir, en general no se conoce la variedad de metodologías existentes para la gestión y mejora de procesos, por lo que difícilmente podremos seleccionar el modelo más adecuado si desconocemos que existe.

2.2.3. Desconocemos la estimación de beneficios

Para poder generar un adecuado entendimiento de implicaciones y beneficios asociados a un programa de mejora, el análisis del Retorno de Inversión o ROI es una herramienta de utilidad, permite apoyar la justificación de la inversión en términos tangibles para los tomadores de decisiones en la organización [5]. Podremos realizar estimaciones de los beneficios antes, durante y después de la ejecución del proyecto, siempre y cuando contemos con los datos adecuados para ello:

- Hay que determinar el alcance de la iniciativa de mejora.

- Estimar el esfuerzo y los costes asociados.
- Dimensionar los beneficios.
- Calcular el ROI.
- Presentación de resultados.

Para calcular el ROI es necesario realizar la siguiente ecuación que nos dará el resultado en % de los beneficios obtenidos:

$$\text{ROI} = [(\text{retorno de la inversión} - \text{inversión inicial}) / (\text{inversión inicial})] * 100$$

Este cálculo podría ayudarnos en la selección del modelo de proceso más adecuado para nuestra organización, es decir, deberíamos utilizar este dato si hay resistencia a la implantación de un modelo de proceso en nuestra organización. Este dato podría despejar dudas al respecto, cuando haya que tomar la decisión al seleccionar un modelo de proceso por el que debemos de realizar una inversión elevada de dinero inicialmente frente a otro que no la necesite.

Capítulo 3

Estado de la Cuestión

En la actualidad podemos dividir los modelos de procesos en dos grupos; tradicionales y ágiles. Las metodologías tradicionales también son conocidas como pesadas, esto se debe al grado de flexibilidad a la hora de adaptar un modelo a circunstancias particulares de una organización, frente a las metodologías ágiles que como podremos comprobar ofrecen mayor grado de flexibilidad en las circunstancias que así lo requieran.

Analizaremos los principales modelos de proceso tradicionales y ágiles. Gracias a este análisis podremos realizar una comparativa entre los modelos tradicionales y ágiles que nos ayudara a entender el estado de la cuestión.

Además, descubriremos que el uso de los distintos modelos existentes dentro de una misma organización, no solo es posible, sino que es lo más apropiado para gestionar y mejorar los procesos software de nuestra organización.

A continuación, definiremos algunos conceptos generales para entender el contexto y el propio problema planteado.

3.1. Conceptos generales

3.1.1. Procesos software

Se entiende por “proceso” [1] la serie de pasos utilizados para producir un resultado deseado. En el caso del software, proceso es el conjunto de actividades, métodos, prácticas y transformaciones que utilizan las personas para desarrollar, mantener el software y sus productos asociados.

Así un proceso en el ámbito del software podrá ser la gestión de requisitos, la definición de requisitos, el aseguramiento de la calidad, la gestión de configuración, la planificación del proyecto, el seguimiento y el control de dicho proyecto, las pruebas, el mantenimiento y los servicios asociados a él, etc. A medida que una empresa software madura su proceso software, institucionaliza su proceso mediante directrices, estándares y estructuras organizativas. La institucionalización implica la construcción de una infraestructura y de una cultura corporativa que soporte los métodos, prácticas y procedimientos de la empresa, para que estos perduren después de que aquellos que los definieron hayan abandonado la organización.

3.1.2. Áreas de proceso

Son un conjunto de prácticas relacionadas a un área determinada que cuando se implementa colectivamente, cumple una serie de objetivos considerados importantes para hacer mejoras en esa área de proceso. Las áreas de proceso cubren desde el desarrollo del producto y de los servicios, hasta el mantenimiento de los mismos.

La definición de las áreas de proceso dentro de nuestra organización facilitará la elaboración del análisis con respecto a la situación actual de la misma y la aplicación de cualquier modelo de proceso que deseemos implantar entre otras ventajas frente a las organizaciones que no tienen bien definidas las áreas de proceso.

3.1.3. Modelos de procesos

Los modelos de procesos son pautas a seguir en cuanto al conjunto de actividades, métodos, prácticas y transformaciones que utilizan las personas para conseguir un resultado deseado. Si hablamos de modelos de proceso relacionados con el software, estos nos ayudaran a desarrollar, mantener el software y sus productos asociados mediante una metodología definida y previamente probada.

En definitiva, un modelo de proceso pretende ayudarnos a gestionar cualquier proceso de forma ordenada y controlada, indicándonos las pautas generales a seguir para lograr nuestros objetivos optimizando los recursos disponibles.

3.1.4. Ingeniería software

Ingeniería del Software es el estudio de los principios y metodologías para desarrollo y mantenimiento de sistemas de software. [Zelkovitz, 1978].

Ingeniería del Software es la aplicación práctica del conocimiento científico en el diseño y construcción de programas de computadora y la documentación asociada requerida para desarrollar, operar y mantenerlos. Se conoce también como desarrollo de software o producción de software. [Bohem, 1976].

La Ingeniería del software trata del establecimiento de los principios y métodos de la ingeniería a fin de obtener software de modo rentable que sea fiable y trabaje en máquinas reales. [Bauer, 1972].

Es una disciplina que comprende todos los aspectos de la producción de software desde las etapas iniciales de la especificación del sistema, hasta el mantenimiento de éste después de que se utiliza. [Sommerville, 2001]

3.2. El método en las organizaciones

En las organizaciones se usa el método como modo estructurado y ordenado de proceder para obtener un resultado, es por ello que adquiere mayor relevancia el uso de métodos que garanticen dentro de las organizaciones un funcionamiento coordinado básico, que nos permita alcanzar los objetivos deseados de la mejor forma. Aquí es donde entran en juego los modelos de procesos.

3.2.1. Proyectos gestionados sin modelos de procesos

En la actualidad, podremos encontrarnos con proyectos gestionados sin seguir una metodología clara y bien definida. En consecuencia, los resultados obtenidos probablemente no serán los deseados [1]:

- Roles del equipo, incluido el de jefe de proyecto, poco definidos o no coordinados.
- Improvisación sobre la marcha.
- Pérdidas de rendimiento del equipo, que se encuentra sometido a crisis constantes (apaga fuegos).
- Adecuación inexistente entre la capacidad y habilidades del equipo y la tarea asignada
- Falta de un seguimiento adecuado del avanza del desarrollo y del punto exacto en que se está dentro del proceso
- Estimaciones de fechas de entrega irreales, sin justificación.
- Hay retrasos en el plazo de entrega
- Es usual la existencia de sobre-costes de todo tipo
- Las actividades dedicadas a garantizar la calidad (revisiones y pruebas) suelen recortarse o eliminarse directamente.
- Elevada rotación del personal.
- Poca motivación por realizar las funciones asignadas.

Estos proyectos no fracasan en su totalidad pero su éxito se apoya en el esfuerzo extra del propio equipo (héroes), capacidad individual y no de la empresa. Por tanto, aunque el proyecto cumpla en plazo y costes, la empresa no puede asegurar ese mismo éxito en proyectos sucesivos.

Si el motivo por el cual no se aplica una metodología en la gestión de un proyecto fuese la complejidad a la hora de aplicar la metodología o la complejidad del proyecto en sí, es muy probable que ocurra alguna de las siguientes situaciones:

- No se ha seleccionado la metodología más adecuada
- No hay unos criterios claros
- Los objetivos son muy ambiciosos

- No hay personal cualificado o debidamente formado al frente del proyecto
- No existe compromiso con el método

Por todo ello, creemos que uno de los aspectos fundamentales, es saber seleccionar el Modelo adecuado, pero no el único, para cubrir nuestras necesidades y que más se ajuste a nuestras circunstancias para alcanzar los objetivos deseados.

3.2.2. Proyectos gestionados con modelos de procesos

Cualquier proyecto gestionado mediante modelos de procesos directamente se incluye en la mejora del proceso software, un mecanismo de mejora continua de la calidad. Se puede utilizar para evaluar [1] la capacidad de los proveedores, para auditar desarrollos software internos y para planificar la estrategia de ingeniería del software de la empresa. Según un estudio llevado a cabo sobre un grupo de empresas implicadas en la mejora del proceso software, se han obtenido en una media de tres años y medio retornos de inversión de 7 veces el coste de las mejoras, debido a la reducción de los costes de producción, a una mejora de la calidad de los productos y a conseguir ajustarse a las necesidades de los clientes, entre otros logros. También indican estas empresas como beneficios menos tangibles un entorno de trabajo más estable, una reducción de la tasa de rotación del personal y una mejora en las relaciones de trabajo con los clientes.

La evidencia del valor de la mejora del proceso software está clara. Entre los beneficios encontrados en las empresas que se han embarcado en la mejora de su proceso software se puede destacar que han obtenido:

- Ganancia media en productividad de un 39%
- Reducción media de tiempo de puesta en marcha del 19%
- Retorno de la inversión entre 4 y 9 veces la inversión inicial

Otros beneficios fueron una mayor detección de errores antes de que los productos saliesen al mercado (22%) e incluso, algunas empresas han alcanzado un mecanismo de planificación de proyectos tan preciso que ahora sus proyectos están a tiempo y dentro del presupuesto.

En esencia, la mejora del proceso es simple, consiste en aplicar las prácticas que proporcionan buenos resultados y cambiar o eliminar las practicas que causan problemas. Así, al iniciar un programa de mejora del proceso software, es necesario tener en cuenta los siguientes objetivos principales de la mejora del proceso:

- Comprender el estado actual de las prácticas de gestión y de ingeniería del software en la empresa.

- Seleccionar las áreas de mejora donde los cambios puedan producir los máximos beneficios a medio y largo plazo.
- Centrarse en añadir valor al negocio, no en alcanzar la utopía del mejor proceso posible.
- Combinar procesos eficaces con personas con habilidades, motivadas y creativas.

Sin embargo, aunque muchas empresas de desarrollo se están subiendo al tren de la mejora del proceso software, no obstante, un número importante de ellas no consiguen arrancar definitivamente. Lo más general en estas últimas, es que lleven a cabo las siguientes actividades:

- Gastar dinero y tiempo en la evaluación del proceso, servicios de consultoría y seminarios de formación.
- Crear un gran manual de procedimientos y decir a todo el personal que tiene que seguir todos los procedimientos, comenzando inmediatamente.
- Escuchar el mandato de la alta dirección de “hágalo”, pero sin comprometerse (generalmente en forma de no proporcionar recursos)
- Ver como los manuales recogen polvo a medida que el personal no cambia realmente el modo en que trabajan.

3.3. 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 [6].

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 esta 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:

- CMMI
- Iconix
- IDEAL
- ITIL
- ISO 9001

- MSF (Microsoft Solution Framework)
- RUP (Rational Unified Process)
- SPICE
- Win-Win Spiral Model

Las metodologías tradicionales seguirán siendo útiles para el desarrollo de software en aquellos sectores en los que sea precisa la construcción de software crítico o con un nivel alto de calidad, como puede ser el sector aeronáutico, espacial, ferroviario, bancario, etc. Son sectores en los que es precisa una planificación controlada, asegurando cada uno de los pasos que se dan para la correcta creación de software y no solo en sectores críticos son necesarias las metodologías tradicionales, sino que también deben usarse en factorías software donde son necesarios unos estándares de calidad para producir productos competitivos y satisfactorios para el cliente final.

A continuación, describiremos tres metodologías tradicionales utilizadas en el sector de la industria del software; CMMI, ITIL y SPICE.

3.3.1. CMMI

❖ Historia

CMMI (Capability Maturity Model Integration) es un conjunto de modelos para las buenas prácticas que ayudan a las organizaciones a mejorar sus procesos. Estos modelos son desarrollados por equipos de producto con los miembros de la industria, gobierno y el Instituto de Ingeniería de Software (SEI) [7]. En el caso de la industria relacionada con las tecnologías de la información un modelo específico, este modelo denominado CMMI para el Desarrollo (CMMI-DEV), proporciona un amplio conjunto de directrices para el desarrollo de productos y servicios.

CMMI for Development es un modelo que cubre las actividades del desarrollo de los productos y servicios. Organizaciones de muchas industrias, incluyendo la aeroespacial, la banca, hardware, software, defensa, la fabricación de automóviles y las telecomunicaciones utilizan CMMI for Development. Contiene prácticas que abarcan la gestión de proyectos, gestión de procesos, ingeniería de sistemas, ingeniería hardware, ingeniería software y otros procesos de apoyo utilizados en el desarrollo y mantenimiento.

El proyecto de integración de CMM se creó para solucionar el problema de la utilización de múltiples CMM. La combinación de los modelos seleccionados en un solo marco de mejora fue diseñada para ser utilizado por las organizaciones en su búsqueda de la mejora de procesos en toda la empresa. Para conseguir la integración de un conjunto de modelos en uno necesito de algo más que simplemente combinar los modelos ya existentes. Usando procesos que

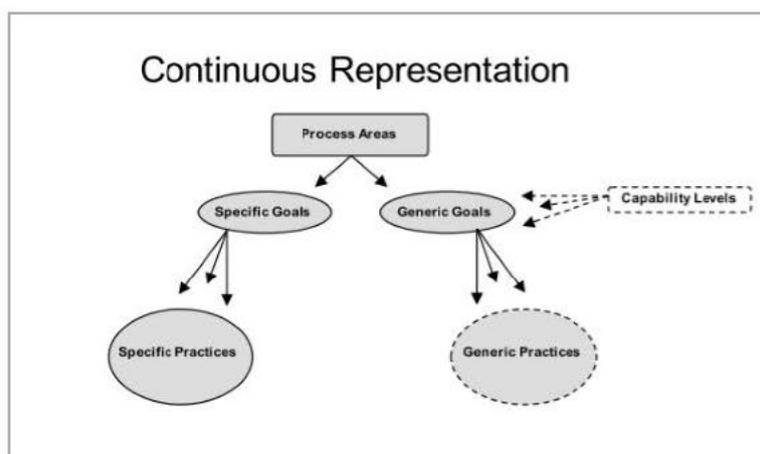
todos los modelos. La Version 1.3 del CMMI for Acquisition, del CMMI for Development y CMMI for Services fueron lanzados en 2010.

❖ Niveles

Los niveles se utilizan en CMMI-DEV para describir un cambio evolutivo recomendado para una organización que quiere mejorar los procesos que utiliza para desarrollar productos o servicios. También pueden ser el resultado de la actividad de clasificación en las evaluaciones, siete evaluaciones pueden aplicarse a la totalidad de organizaciones o grupos más pequeños, como grupos de proyectos o divisiones.

CMMI soporta dos partes de mejoras usando niveles. Una parte permite a la organización mejorar de forma incremental procesos correspondientes a un área de proceso individual o un grupo de áreas de proceso, seleccionadas por la organización. La otra parte permite a las organizaciones mejorar un conjunto de procesos relacionados abordando gradualmente conjuntos sucesivos de áreas de proceso.

Estas dos partes para la mejora se asocian con los dos tipos de niveles; niveles de capacidad y niveles de madurez. Estos niveles se corresponden a dos enfoques para la mejora del proceso llamado representaciones. Las dos representaciones se llaman; Continuous Representation (representación continua) y Staged Representation (representación por etapas). El uso de la representación continua nos permite alcanzar los niveles de capacidad y el uso de la representación por etapas nos permite alcanzar los niveles de madurez.



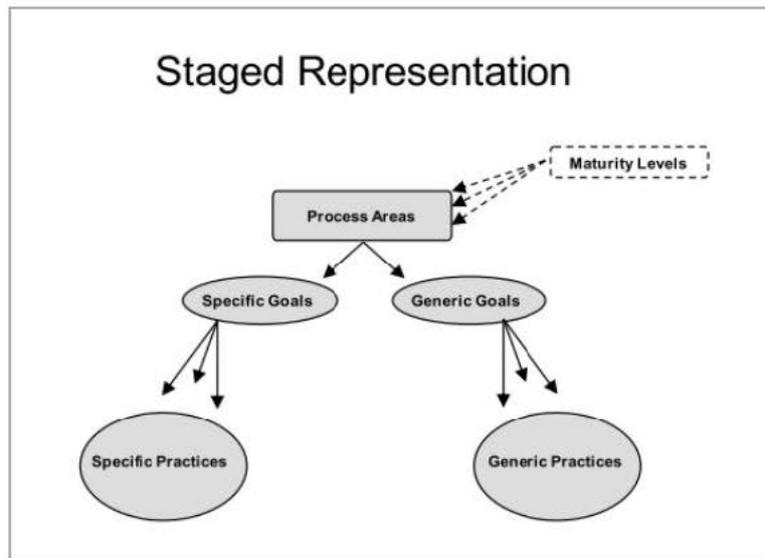


Figura 3. Diagrama conceptual de la representación continua en la parte superior y por etapas en la parte inferior de la figura.

Para llegar a un nivel particular, una organización debe satisfacer todos los objetivos del área de proceso o del conjunto de áreas de proceso que están enfocados a la mejora, independientemente de si se trata de una capacidad o un nivel de madurez. Ambas representaciones proporcionan maneras de mejorar sus procesos para alcanzar los objetivos de negocio y ofrecen el mismo contenido esencial y uso de los mismos componentes de modelo.

Las diferencias entre las estructuras son sutiles pero significativas. La representación por etapas utiliza niveles de madurez para caracterizar el estado general de los procesos de la organización en relación con el modelo como un conjunto, mientras que la representación continua utiliza los niveles de capacidad para caracterizar el estado de los procesos de la organización con respecto a un área de proceso individual.

La representación continua se centra en la capacidad de área de proceso para medida por los niveles de capacidad y la representación por etapas se centra en la madurez general, medida por los niveles de madurez. CMMI utiliza estos niveles para comparar y evaluar actividades, así como orientar la mejora de los esfuerzos de la organización.

Los niveles de capacidad se aplican a la mejora de procesos de la organización en las áreas de proceso individuales. Estos niveles son un medio para mejorar los procesos incrementalmente correspondientes a un área de proceso. Los cuatro niveles de capacidad están numerados del 0 al 3.

Los niveles de madurez se aplican a la mejora de procesos de la organización a través de múltiples áreas de proceso. Estos niveles son un medio de la mejora de los procesos correspondientes a un determinado conjunto de áreas

de proceso. A continuación se muestra una tabla con los distintos niveles de capacidad y madurez:

Nivel	Niveles de Capacidad, Representación Continua	Niveles de Madurez, Representación por Etapas
Nivel 0	Incompleto	-
Nivel 1	Realizado	Iniciado
Nivel 2	Administrado	Administrado
Nivel 3	Definido	Definido
Nivel 4	-	Gestionado Cuantitativamente
Nivel 5	-	Optimizado

Tabla 2. Niveles CMM.

❖ Áreas de proceso

Las 22 áreas de proceso se presentan en orden alfabético por acrónimo:

- Causal Analysis and Resolution (CAR)
- Configuration Management (CM)
- Decision Analysis and Resolution (DAR)
- Integrated Project Management (IPM)
- Measurement and Analysis (MA)
- Organizational Process Definition (OPD)
- Organizational Process Focus (OPF)
- Organizational Performance Management (OPM)
- Organizational Process Performance (OPP)
- Organizational Training (OT)
- Product Integration (PI)
- Project Monitoring and Control (PMC)
- Project Planning (PP)
- Process and Product Quality Assurance (PPQA)
- Quantitative Project Management (QPM)
- Requirements Development (RD)
- Requirements Management (REQM)
- Risk Management (RSKM)
- Supplier Agreement Management (SAM)
- Technical Solution (TS)
- Validation (VAL)
- Verification (VER)

Las áreas de proceso son vistas de manera diferente según las dos representaciones existentes. A continuación podremos comparar los puntos de vista con los que se utilizan las áreas de proceso en la representación continua y la representación por etapas:

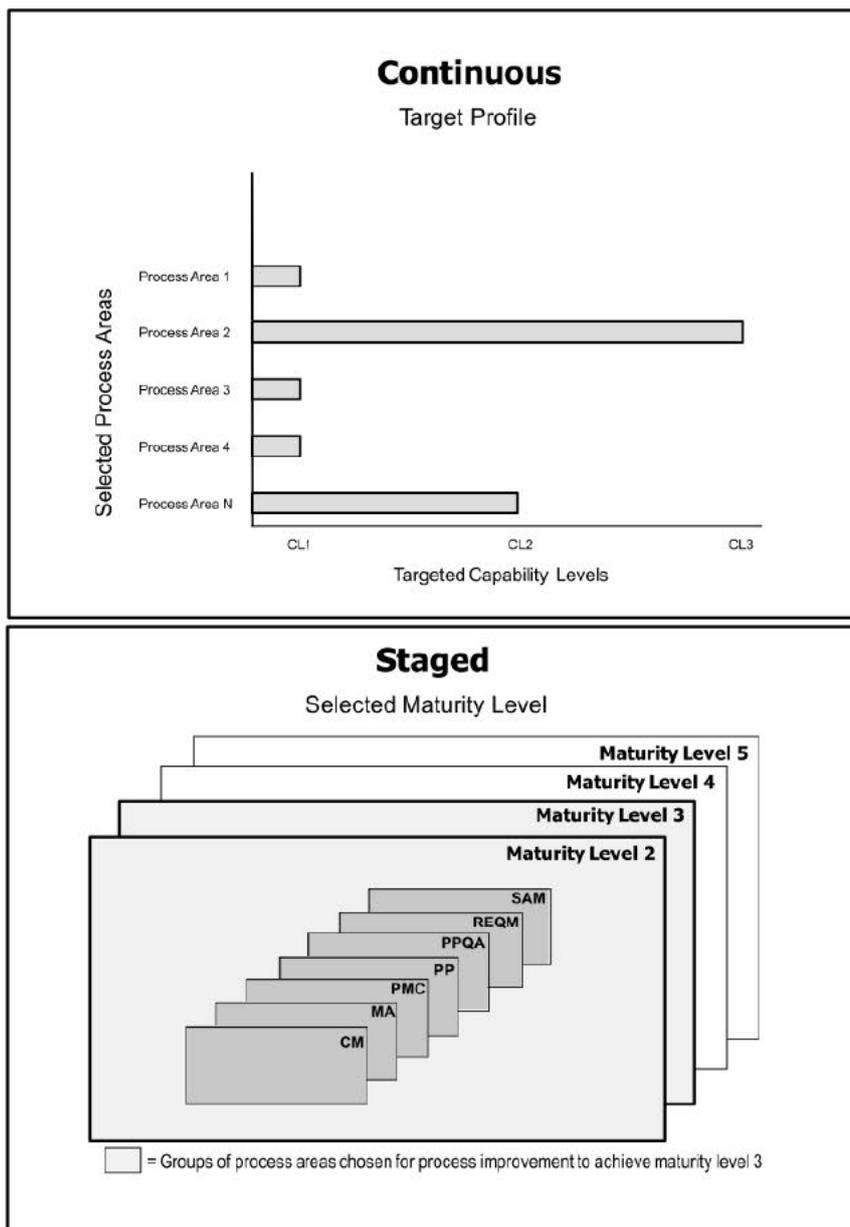


Figura 4. Diagrama conceptual de las áreas de proceso y la representación continua en la parte superior y por etapas en la parte inferior de la figura.

La representación continua permite a la organización elegir el foco de sus esfuerzos de mejora de procesos eligiendo aquellas áreas de proceso o conjuntos de áreas de procesos interrelacionados para mejorar los beneficios de la organización y sus objetivos de negocio. Aunque hay algunos límites cuando las organizaciones seleccionan un área de proceso ya que estas tienen dependencias entre áreas, la organización cuenta con una gran libertad en su selección.

Para utilizar la representación continua, las áreas de proceso se organizan en cuatro categorías: Gestión de Procesos, Gestión de Proyectos, Ingeniería Y

Soporte. A veces, un grupo informal de áreas de proceso se denominan como áreas de proceso de alta madurez, estas áreas son 4; OPP, QPM, OPM y CAR. Estas áreas de proceso se centran en mejorar el rendimiento de los procesos implementados que se relacionan más estrechamente a la organización de los objetivos de negocio.

❖ **Relación entre áreas**

Las relaciones entre las múltiples áreas de proceso, incluyendo la información y artefactos que fluyen desde un área de proceso a otro, ayudan a ver de forma más amplia la implementación y mejora de procesos.

Ser consciente de las relaciones fundamentales que existen entre las áreas de proceso de CMMI nos ayudara a aplicar CMMI de una manera útil y productiva. Estas relaciones las encontraremos en las descripciones que CMMI específica para cada una de las 22 áreas de proceso mencionadas en el apartado anterior.

Como ya hemos mencionado estas áreas se agrupan en cuatro categorías para entender mejor la relación entre ellas:

- **Gestión de Procesos:** Contienen las actividades entre proyectos relacionados con la definición, planificación, implementación, ejecución, seguimiento, control, evaluación, medición y mejora de los procesos:
 - Organizational Process Definition (OPD)
 - Organizational Process Focus (OPF)
 - Organizational Performance Management (OPM)
 - Organizational Process Performance (OPP)
 - Organizational Training (OT)

- **Gestión de Proyectos:** Abarca la gestión de proyectos y las actividades relacionadas con la planificación, el seguimiento y el control del proyecto:
 - Integrated Project Management (IPM)
 - Project Monitoring and Control (PMC)
 - Project Planning (PP)
 - Quantitative Project Management (QPM)
 - Requirements Management (REQM)
 - Risk Management (RSKM)
 - Supplier Agreement Management (SAM)

- **Ingeniería:** Las áreas de proceso de ingeniería se aplican al desarrollo de cualquier producto o servicio en el ámbito del desarrollo, por ejemplo, productos software, hardware, servicios o procesos:
 - Product Integration (PI)

- Requirements Development (RD)
 - Technical Solution (TS)
 - Validation (VAL)
 - Verification (VER)
- **Soporte:** Cubren las actividades que apoyan productos de desarrollo y mantenimiento, también se utilizan en el contexto del soporte de otras áreas de proceso:
 - Causal Analysis and Resolution (CAR)
 - Configuration Management (CM)
 - Decision Analysis and Resolution (DAR)
 - Measurement and Analysis (MA)
 - Process and Product Quality Assurance (PPQA)

3.3.2. ITIL

❖ Historia

ITIL (Information Technology Infrastructure Library) puede ser definido como un conjunto de buenas prácticas destinadas a mejorar la gestión y provisión de servicios TI [8]. Su objetivo último es mejorar la calidad de los servicios TI ofrecidos, evitar los problemas asociados a los mismos y en caso de que estos ocurran ofrecer un marco de actuación para que estos sean solucionados con el menor impacto y a la mayor brevedad posible.

Sus orígenes se remontan a la década de los 80 cuando el gobierno británico, preocupado por la calidad de los servicios TI de los que dependía la administración, solicitó a una de sus agencias, la CCTA acrónimo de Central Computer and Telecommunications Agency, para que desarrollara un estándar para la provisión eficiente de servicios TI.

En la actualidad es la OGC (Office of Government Commerce) el organismo encargado de velar por este estándar y la responsable de la última versión de ITIL (v3) que data del año 2007 (versión que nosotros utilizaremos).

La OGC cuenta con la colaboración de varias organizaciones para el mantenimiento de ITIL:

- **itsMF:** el Information Technology Management Forum es una organización independiente y reconocida internacionalmente que tiene como principal objetivo impulsar la adopción de las mejores prácticas ITIL para la gestión de servicios TI.
- **APM Group:** es una organización comercial encargada por la OGC de definir, publicar y gestionar las certificaciones ITIL así como de acreditar a los organismos examinadores.

- **Organismos examinadores:** en la actualidad existen varios organismos examinadores acreditados por APMG entre los que se encuentran EXIN, BCS/ISEB y LCS.

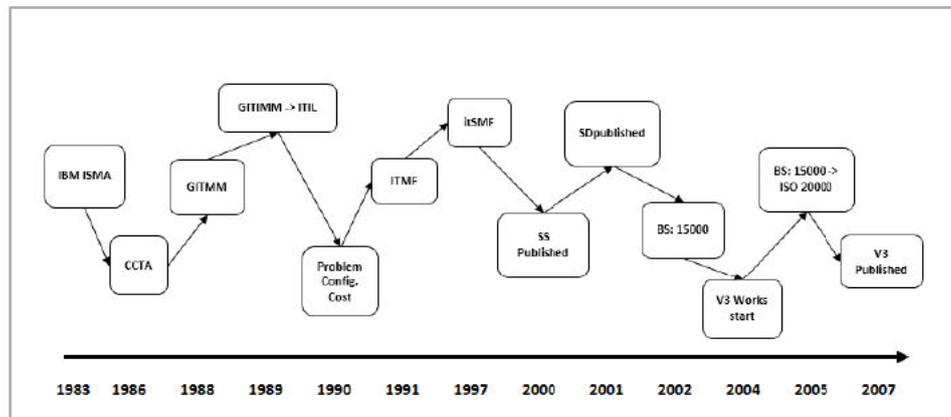


Figura 5. Línea temporal sobre la historia de ITIL.

❖ Fases

ITIL estructura la gestión de los servicios TI sobre el concepto de Ciclo de Vida de los Servicios. El Ciclo de Vida del Servicio consta de cinco fases que se corresponden con los libros de ITIL:

1. **Estrategia del Servicio:** propone tratar la gestión de servicios no sólo como una capacidad sino como un activo estratégico.
2. **Diseño del Servicio:** cubre los principios y métodos necesarios para transformar los objetivos estratégicos en portafolios de servicios y activos.
3. **Transición del Servicio:** cubre el proceso de transición para la implementación de nuevos servicios o su mejora.
4. **Operación del Servicio:** cubre las mejores prácticas para la gestión del día a día en la operación del servicio.
5. **Mejora Continua del Servicio:** proporciona una guía para la creación y mantenimiento del valor ofrecido a los clientes a traves de un diseño, transición y operación del servicio optimizado.

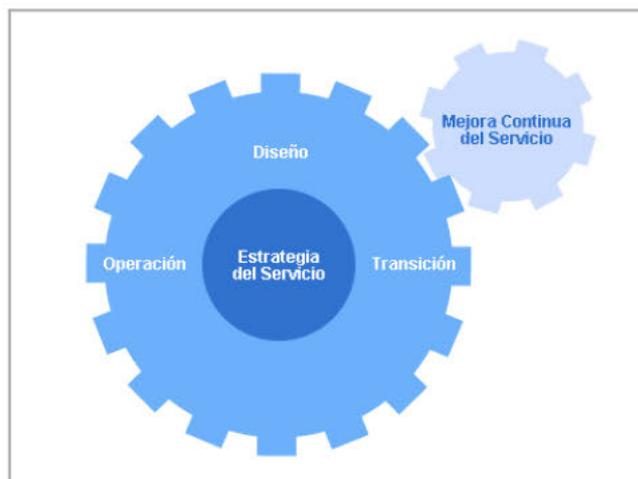


Figura 6. Las 5 fases ITIL relacionadas entre sí.

Las distintas fases tienen múltiples procesos que en ocasiones están relacionados con otros procesos de otras fases. Ninguno de los ciclos de la fase de vida de los servicios debe ser considerado como un compartimento estanco pues sus interrelaciones con las otras fases son de vital importancia para la correcta Gestión del Servicio.

❖ Procesos

La fase de **Estrategia del Servicio** es central al concepto de Ciclo de vida del servicio y tiene como principal objetivo convertir la Gestión del Servicio en un activo estratégico. Para conseguir este objetivo es imprescindible determinar en primera instancia qué servicios deben ser prestados y por qué han de ser prestados desde la perspectiva del cliente y el mercado. Los procesos asociados directamente a la fase de Estrategia son:

- **Gestión Financiera:** responsable de garantizar la prestación de servicios con unos costes controlados y una correcta relación calidad-precio.
- **Gestión del Portfolio de Servicios:** responsable de la inversión en servicios nuevos y actualizados que ofrezcan el máximo valor al cliente minimizando a su vez los riesgos y costes asociados.
- **Gestión de la Demanda:** responsable de la armonización de la oferta de los servicios ofrecidos con las demandas del mercado.

La fase de **Diseño del Servicio** es la responsable de diseñar nuevos servicios o modificar los ya existentes para su incorporación al catálogo de servicios y su paso al entorno de producción. El Diseño del Servicio debe seguir las directrices establecidas en la fase de Estrategia y debe a su vez colaborar con ella para que los servicios diseñados se adecuen a las necesidades del mercado, sean eficientes en costes y rentables, cumplan los estándares de

calidad adoptados y aporten valor a clientes y usuarios. Los procesos de la fase de Diseño son:

- **Gestión del Catálogo de Servicios:** responsable de crear y mantener un catálogo de servicios de la organización TI que incluya toda la información relevante: gestores, estatus, proveedores, etcétera.
- **Gestión de Niveles de Servicio:** responsable de acordar y garantizar los niveles de calidad de los servicios TI prestados.
- **Gestión de la Capacidad:** responsable de garantizar que la organización TI dispone de la capacidad suficiente para prestar los servicios acordados.
- **Gestión de la Disponibilidad:** responsable de garantizar que se cumplen los niveles de disponibilidad acordados en los SLA.
- **Gestión de la Continuidad de los Servicios TI:** responsable de establecer planes de contingencia que aseguren la continuidad del servicio en un tiempo predeterminado con el menor impacto posible en los servicios de carácter crítico.
- **Gestión de la Seguridad de la Información:** responsable de establecer las políticas de integridad, confidencialidad y disponibilidad de la información.
- **Gestión de Proveedores:** responsable de la relación con los proveedores y el cumplimiento de los UCs.

La fase de **Transición del Servicio** es hacer que los productos y servicios definidos en la fase de Diseño del Servicio se integren en el entorno de producción y sean accesibles a los clientes y usuarios autorizados. Los procesos de la fase de Transición son:

- **Planificación y soporte a la Transición:** responsable de planificar y coordinar todo el proceso de transición asociado a la creación o modificación de los servicios TI.
- **Gestión de Cambios:** responsable de supervisar y aprobar la introducción o modificación de los servicios prestados garantizando que todo el proceso ha sido convenientemente planificado, evaluado, probado, implementado y documentado.
- **Gestión de la Configuración y Activos del Servicio:** responsable del registro y gestión de los elementos de configuración (CIs) y activos del servicio. Este proceso da soporte a prácticamente todos los aspectos de la Gestión del Servicio.
- **Gestión de Entregas y Despliegues:** Responsable de desarrollar, probar e implementar las nuevas versiones de los servicios según las directrices marcadas en la fase de Diseño del Servicio.
- **Validación y pruebas:** responsable de garantizar que los servicios cumplen los requisitos preestablecidos antes de su paso al entorno de producción.
- **Evaluación:** responsable de evaluar la calidad general de los servicios, su rentabilidad, su utilización, la percepción de sus usuarios, etcétera.
- **Gestión del Conocimiento:** gestiona toda la información relevante a la prestación de los servicios asegurando que esté disponible para los agentes implicados en su concepción, diseño, desarrollo, implementación y operación.

La fase de **Operación del Servicio** es, sin duda, la más crítica entre todas. La percepción que los clientes y usuarios tengan de la calidad de los servicios prestados depende en última instancia de una correcta organización y coordinación de todos los agentes involucrados. Los principales objetivos de la fase de Operación del Servicio incluyen; coordinar e implementar todos los procesos, actividades y funciones necesarias para la prestación de los servicios acordados con los niveles de calidad aprobados, dar soporte a todos los usuarios del servicio y gestionar la infraestructura tecnológica necesaria para la prestación del servicio. Los procesos de la fase de Operación son:

- **Gestión de Eventos:** responsable de monitorizar todos los eventos que acontezcan en la infraestructura TI con el objetivo de asegurar su correcto funcionamiento y ayudar a prever incidencias futuras.
- **Gestión de Incidencias:** responsable de registrar todas las incidencias que afecten a la calidad del servicio y restaurarlo a los niveles acordados de calidad en el más breve plazo posible.
- **Petición de Servicios TI:** responsable de gestionar las peticiones de usuarios y clientes que habitualmente requieren pequeños cambios en la prestación del servicio.
- **Gestión de Problemas:** responsable de analizar y ofrecer soluciones a aquellos incidentes que por su frecuencia o impacto degradan la calidad del servicio.
- **Gestión de Acceso a los Servicios TI:** responsable de garantizar que sólo las personas con los permisos adecuados pueda acceder a la información de carácter restringido.

Y por último la fase de **Mejora Continua del Servicio** que exige continuos cambios y éstos deben tener un solo objetivo en el campo de la gestión de servicios TI: ofrecer mejores servicios adaptados a las siempre cambiantes necesidades de nuestros clientes y todo ello mediante procesos internos optimizados que permitan mayores retornos a la inversión y mayor satisfacción del cliente. Pero este objetivo de mejora sólo se puede alcanzar mediante la continua monitorización y medición de todas las actividades y procesos involucrados en la prestación de los servicios TI. Los procesos de la fase de Mejora son:

- **Proceso de Mejora:** este es un proceso que consta de 7 pasos que describen como se deben medir la calidad y rendimiento de los procesos para generar los informes adecuados que permitan la creación de un Plan de Mejora del Servicio (SIP).
- **Informes de Servicios TI:** es el responsable de la generación de los informes que permitan evaluar los servicios ofrecidos y los resultados de las mejoras propuestas.

❖ Relación entre fases

A continuación resumimos las principales interdependencias.

Estrategia y Diseño: El principal input ofrecido a la fase de Diseño del Servicio por la fase de Estrategia es un Portfolio de Servicios orientado a cada segmento del mercado. La estrategia debe aportar al diseño del servicio:

- Modelos de servicio que ofrezcan una guía sobre como aportar valor a los servicios propuestos.
- Información sobre restricciones derivadas de los clientes o política de precios, etcétera.

Estrategia y Transición: A la hora de establecer una correcta Estrategia del Servicio es necesario conocer en profundidad sus implicaciones en la fase de Transición del Servicio. Cada cambio y evolución implica costes e inevitablemente tiene un impacto en clientes y usuarios.

Es indispensable sopesar los riesgos y potenciales beneficios asociados para establecer una estrategia que minimice los primeros maximizando a su vez los segundos.

Por otro lado la Transición del Servicio debe colaborar, en aquello que le corresponde, a dar soporte a la perspectiva y posicionamiento del servicio establecidos en la fase de estrategia.

Diseño y Transición: La fase de transición debe disponer de toda la documentación necesaria para elaborar los planes de cambio y realizar el despliegue del servicio:

- Planes de capacidad y disponibilidad
- Paquetes de servicio
- SLAs
- Planes de continuidad TI

A su vez la fase de Transición debe asesorar al Diseño sobre los riesgos y posibles impactos del cambio en la calidad del servicio.

Diseño y Operación: La fase de operación es la más crítica y de ella depende la percepción final del cliente sobre la calidad del servicio. Por lo tanto un factor esencial en el diseño del servicio es tener en cuenta la operativa del mismo.

El diseño debe:

- Ser usable.
- Ser sostenible y escalable.
- Ofrecer la funcionalidad requerida.
- Ser eficiente.
- Cumplir los protocolos de seguridad requeridos.

- Permitir el acceso sólo al personal autorizado.

Transición y Operación: La Operación del Servicio debe suministrar información relevante sobre:

- El entorno de producción
- El conocimiento asociado (incidencias, percepción de clientes y usuarios...) a servicios similares a los que se han de desplegar.

La Transición del Servicio debe poner a disposición de la fase de Operación:

- Toda la documentación necesaria asociada al uso y mantenimiento de los nuevos o actualizados servicios.
- La información relativa a los procesos de prueba y evaluación.

Transición y Mejora Continua: La principal misión de la fase de Mejora Continua es mejorar todos los procesos y tareas involucrados en la prestación del servicio con el objetivo último de mejorar la calidad, rendimiento y rentabilidad de estos y la consecuente percepción de clientes, usuarios y organización TI.

La fase de Transición es clave en este aspecto. Los cambios son la fuente principal de incidencias y problemas tanto a nivel interno (componente tecnológica) como a nivel externo (calidad del servicio).

La fase de Mejora Continua es por sí misma una de las principales fuentes de cambio introduciendo mejoras en los procesos y ajustando la calidad y rentabilidad de los servicios.

Operación y Estrategia: La fase de Operación es la más importante desde el punto de vista del cliente, los servicios pueden ser adecuados y estar bien diseñados pero si el eslabón de la operación falla los resultados no serán los buscados y la percepción del cliente será negativa.

Por lo tanto un factor esencial en el enfoque estratégico de los servicios es asegurar que son operacionalmente viables.

Recíprocamente, la Operación del Servicio debe de resultar en la fuente más fiable sobre las demandas y restricciones de los clientes que servirán de guía para dar forma a la estrategia más adecuada.

Operación y Mejora Continua: La fase de Mejora Continua del Servicio depende directamente de la fase de Operación pues ésta representa la principal fuente de información para la optimización de los procesos y actividades involucrados en la prestación del servicio.

Los informes generados en la fase de Operación del Servicio deben, en particular, incorporar información detallada sobre:

- Incidencias en la prestación del servicio.
- Soluciones propuestas a los problemas detectados en la fase de operación.
- Peticiones de los usuarios y clientes.

Mejora Continua y Estrategia: En un mundo en constante desarrollo tecnológico las estrategias no deben ser inamovibles. La estrategia debe ser continuamente rediseñada atendiendo a múltiples factores.

La Mejora del Servicio debe ofrecer información a la fase de Estrategia sobre aspectos que pueden ser optimizados, tales como calidad y rendimiento, pero esto siempre debe hacerse partiendo de la perspectiva de negocio establecida durante la fase de estrategia.

Mejora Continua y Diseño: La satisfacción de los clientes depende en gran medida de los procesos y actividades desarrolladas en la fase de diseño:

- ¿Resultado la capacidad suficiente?
- ¿Se cumplieron los SLAs?
- ¿Se tuvieron en cuenta los requisitos del cliente?

Si esto no fuera así es necesario introducir planes de mejora que minimicen o eliminen los problemas encontrados y aporten una guía para las mejoras necesarias en las soluciones y arquitecturas empleadas.

3.3.3. SPICE, ISO 15504

❖ Historia

En 1991 la Organización de Estandarización internacional ISO aprobó la realización de un estudio sobre la necesidad de crear un estándar internacional para la evaluación de procesos (software), entonces se creó el proyecto SPICE (Software Process Improvement and Capability dEtermination) [9]. En 1992 se realizó el desarrollo por pasos de un estándar para la evaluación de procesos. Los pasos fueron los siguientes:

- Publicación inicial como Informe Técnico 'Technical Report' (borrador de estándar).
- Después pasa a revisión y publicación como estándar internacional IS ISO/IEC 15504 – Tecnologías de la Información – Evaluación de Procesos ('ISO/IEC 15504 – Information Technology – Process Assessment').

El proyecto SPICE tenía tres objetivos principales:

- Desarrollar un borrador de trabajo para un estándar de evaluación de procesos de software.
- Llevar a cabo los ensayos de la industria de la norma emergente.
- Promover la transferencia de tecnología de la evaluación de procesos de software a la industria del software a nivel mundial.

En 1995 el primer objetivo del proyecto se logró y tres años más tarde se publicó la primera familia de estándares ISO TR 15504, ya en 2003 el proyecto SPICE se cerró oficialmente. Con el fin de apoyar la excelencia y la coherencia de la formación de los evaluadores, el proyecto SPICE también desarrolló y lanzó un Plan de Estudios de formación de los evaluadores SPICE que es utilizado actualmente por el Esquema de Registro Internacional de Evaluadores.

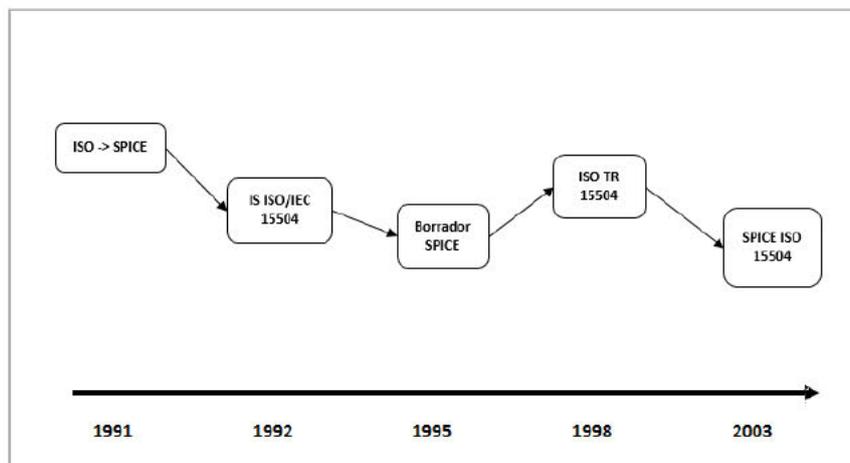


Figura 7. Línea temporal sobre la historia de SPICE.

❖ Partes

Es un estándar internacional que es aplicable a cualquier organización/empresa que quiera conocer y mejorar la capacidad de sus procesos, independientemente del tipo de organización, del modelo del ciclo de vida adoptado, de la metodología de desarrollo y de la tecnología utilizada [10].

La primera versión estructuraba el modelo en nueve partes, pero en el curso de los debates y votaciones, en aras de reducir el tamaño del estándar, se decide que se divida en cinco partes:

- **Parte 1.** Conceptos y Vocabulario: Representa una introducción general a la norma, proporcionando una guía de uso de la misma. En esta parte se incluye el conjunto de términos definidos específicamente para la norma.
- **Parte 2.** Realizando una Evaluación (Requisitos, normativa): Define los requisitos que debe cumplir una evaluación para que produzca resultados repetibles, fiables y consistentes.

- **Parte 3.** Guía para Realización de Evaluaciones: Establece una guía para la realización de evaluaciones de procesos, interpretando los requisitos de las partes normativas para diferentes contextos de evaluación.
- **Parte 4.** Guía para el Uso de Resultados de Evaluaciones: Proporciona una guía para poder utilizar los resultados de una evaluación en la mejora de los procesos evaluados. La guía incluye ejemplos de la aplicación de mejoras en una gran variedad de situaciones.
- **Parte 5.** Un Modelo de Evaluación de Procesos Ejemplar: Proporciona un modelo totalmente compatible con la parte normativa, que incluye un conjunto de indicadores que facilitan el cálculo de la capacidad de los procesos.

❖ Niveles

El estándar no pretende fijar la manera de realizar los procesos dentro de una organización, sino que valora su capacidad y ayuda a proponer mejoras que aumenten esta capacidad [11]. La manera de llevar a cabo estas mejoras no está incluida en la norma. El modelo de referencia agrupa los procesos que se realizan durante el ciclo de vida del software en tres niveles:

1. Procesos primarios.
 - Adquisición
 - Suministro
 - Ingeniería
 - Operación
2. Procesos de soporte.
 - Soporte
3. Procesos de la organización.
 - Gestión
 - Mejora del proceso
 - Recursos e Infraestructura
 - Reutilización

El estándar no indica en ningún momento en que orden y de qué forma deben llevarse a cabo los procesos, se limita a definirlos y a caracterizarlos. Para analizar la capacidad de los procesos hay una escala de valoración que consta de seis niveles, del 0 al 5. Los distintos niveles están caracterizados por unos atributos bien definidos

- **Nivel 0.** Incompleto: El proceso no está implementado.
- **Nivel 1.** Realizado: Existe evidencia de la realización del proceso.

- **Nivel 2.** Gestionado: El proceso se gestiona y los procesos de trabajo se establecen, controlan y mantienen
- **Nivel 3.** Establecido: Se utiliza un proceso adaptado basado en un proceso estándar.
- **Nivel 4.** Predecible: El proceso se gestiona usando técnicas cuantitativas.
- **Nivel 5.** En optimización. El proceso se mejora continuamente para cumplir los objetivos de negocios actuales y futuros.

El nivel de capacidad que tiene un proceso depende de los atributos que posee y del grado con el cual se alcanzan dichos atributos. La escala para la valoración de estos atributos se compone de cuatro valores o grados distintos de posesión del mismo:

- (N) No alcanzado (0% - 15%)
- (P) Parcialmente alcanzado (16% - 50%)
- (L) Ampliamente alcanzado (51% - 85%)
- (F) Completamente alcanzado (86% - 100%)

3.4. Metodologías Ágiles

Las metodologías ágiles son una serie de técnicas para la gestión de proyectos que han surgido como contraposición a los métodos clásicos de gestión o metodologías tradicionales como CMMI, ITIL y SPICE. Aunque surgieron en el ámbito del desarrollo de software, también han sido exportadas a otro tipo de proyectos.

Todas las metodologías que se consideran ágiles cumplen con el manifiesto ágil que no es más que una serie de principios que se agrupan en 4 valores [12]:

- Los individuos y su interacción, por encima de los procesos y las herramientas.
- El software que funciona, frente a la documentación exhaustiva.
- La colaboración con el cliente, por encima de la negociación contractual.
- La respuesta al cambio, por encima del seguimiento de un plan

Los principios que se han definido en el manifiesto son:

- Nuestra mayor prioridad es satisfacer al cliente mediante la entrega temprana y continua de software con valor.
- Aceptamos que los requisitos cambien, incluso en etapas tardías del desarrollo. Los procesos Ágiles aprovechan el cambio para proporcionar ventaja competitiva al cliente.
- Entregamos software funcional frecuentemente, entre dos semanas y dos meses, con preferencia al periodo de tiempo más corto posible.

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

Cualquier metodología que cumpla con estos principios será considerada como ágil. A priori, sin hacer un análisis detallado podemos comprobar que existen grandes diferencias entre las metodologías tradicionales y las ágiles. Una vez que veamos tres de las metodologías ágiles más comunes en la actualidad, veremos una comparativa entre ellas para comprender los objetivos que cubren cada una de las metodologías que mencionamos en este trabajo, ayudándonos a seleccionar la metodología más adecuada para nuestra organización.

3.4.1. SCRUM

❖ Historia

El concepto de Scrum tiene su origen en un estudio de 1986 [13] sobre los nuevos procesos de desarrollo utilizados en productos exitosos en Japón y los Estados Unidos (cámaras de fotos de Canon, fotocopiadoras de Xerox, automóviles de Honda, ordenadores de HP y otros). 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).

En 1993 se realizó el primer Scrum para desarrollo de software y **en 1995 el proceso fue formalizado**. En 2001 un grupo de personas muy relevantes en lo que empezaba a ser el desarrollo ágil escribieron los valores fundamentales de los procesos ágiles.

En la actualidad, Scrum se está utilizando en diferentes tipos de negocio y, especialmente, en el desarrollo de software. La Scrum Alliance es la organización sin ánimo de lucro que se encarga de difundir Scrum en este ámbito.

❖ Fundamentos

Scrum se basa en [14]:

- El desarrollo incremental de los requisitos del proyecto en bloques temporales cortos y fijos (iteraciones de un mes natural y hasta de dos semanas, si así se necesita)
- La priorización de los requisitos por valor para el cliente y coste de desarrollo en cada iteración.
- El control empírico del proyecto. Por un lado, al final de cada iteración se demuestra al cliente el resultado real obtenido, de manera que pueda tomar las decisiones necesarias en función de lo que observa y del contexto del proyecto en ese momento. Por otro lado, el equipo se sincroniza diariamente y realiza las adaptaciones necesarias.
- La potenciación del equipo, que se compromete a entregar unos requisitos y para ello se le otorga la autoridad necesaria para organizar su trabajo.
- La sistematización de la colaboración y la comunicación tanto entre el equipo como con el cliente.
- El timeboxing (fijar el tiempo máximo para conseguir unos objetivos, tomar una decisión o realizar unas tareas) de las actividades del proyecto, para ayudar a la toma de decisiones y conseguir resultados.

❖ El proceso

En Scrum un proyecto se ejecuta en bloques temporales cortos y fijos (iteraciones o sprints de entre dos semanas y un mes natural, en función de las necesidades del proyecto). Cada iteración tiene que proporcionar un resultado completo, un incremento de producto final que sea susceptible de ser entregado que suponga el mínimo esfuerzo para el cliente cuando lo solicite.

El proceso parte de la lista de objetivos para crear el producto, que actúa como plan del proyecto. En esta lista el cliente prioriza los objetivos balanceando el valor que le aportan respecto a su coste y quedan repartidos en iteraciones y entregas. De manera regular el cliente puede maximizar la utilidad de lo que se

desarrolla y el retorno de inversión mediante la re-planificación de objetivos del producto, que realiza durante la iteración con vista a las siguientes iteraciones.

Las actividades que se llevan a cabo en Scrum son las siguientes:

- **Planificación de la iteración:** El primer día de la iteración se realiza la reunión de la iteración:

1. **Selección de requisitos** (4 horas máximo). El cliente presenta al equipo la lista de requisitos priorizada del producto o proyecto. El equipo pregunta al cliente las dudas que surgen y selecciona los requisitos más prioritarios que se compromete a completar en la iteración, de manera que puedan ser entregados si el cliente lo solicita.
2. **Planificación de la iteración** (4 horas máximo). El equipo elabora la lista de tareas de la iteración necesarias para desarrollar los requisitos a que se ha comprometido. La estimación de esfuerzo se hace de manera conjunta y los miembros del equipo se auto asignan las tareas.

- **Ejecución de la iteración:** Cada día el equipo realiza una reunión de sincronización (15 minutos máximo). Cada miembro del equipo inspecciona el trabajo que el resto está realizando (dependencias entre tareas, progreso hacia el objetivo de la iteración, obstáculos que pueden impedir este objetivo) para poder hacer las adaptaciones necesarias que permitan cumplir con el compromiso adquirido. En la reunión cada miembro del equipo responde a tres preguntas:

1. ¿Qué he hecho desde la última reunión de sincronización?
2. ¿Qué voy a hacer a partir de este momento?
3. ¿Qué impedimentos tengo o voy a tener?

Durante la iteración el responsable de liderar el equipo (Scrum Master) se encarga de que el equipo pueda cumplir con su compromiso y de que no se merme su productividad.

1. Elimina los obstáculos que el equipo no puede resolver por sí mismo.
2. Protege al equipo de interrupciones externas que puedan afectar su compromiso o su productividad.

- **Inspección y adaptación:** El último día de la iteración se realiza la reunión de revisión de la iteración. Tiene dos partes:

1. **Demostración** (4 horas máximo). El equipo presenta al cliente los requisitos completados en la iteración, en forma de incremento de producto preparado para ser entregado con el mínimo esfuerzo. En función de los resultados mostrados y de los cambios que haya habido en el contexto del proyecto, el cliente realiza las adaptaciones necesarias de manera objetiva, ya desde la primera iteración, re-planificando el proyecto.

2. **Retrospectiva** (4 horas máximo). El equipo analiza cómo ha sido su manera de trabajar y cuáles son los problemas que podrían impedirle progresar adecuadamente, mejorando de manera continua su productividad, El Scrum Master se encargará de ir eliminando los obstáculos identificados.

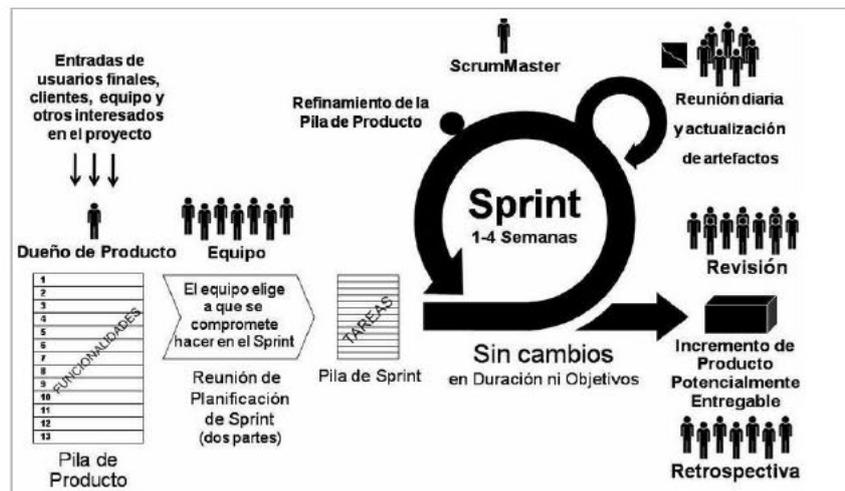


Figura 8. Esquema conceptual del proceso SCRUM.

3.4.2. KANBAN

❖ Historia

Durante las pasadas tres décadas Kanban que se define como “un sistema de producción altamente efectivo y eficiente”, ha contribuido a generar un panorama manufacturero óptimo y competitivo [15]. El origen de la metodología Kanban debemos buscarlo en los procesos de producción “just-in-time” (JIT) ideados por Toyota, en los que se utilizaban tarjetas para identificar necesidades de material en la cadena de producción. Kanban es una palabra japonesa que significa “tarjetas visuales”.

Las principales ventajas de esta metodología es que es muy fácil de utilizar, actualizar y asumir por parte del equipo. Además, destaca por ser una técnica de gestión de las tareas muy visual, que permite ver a golpe de vista el estado de los proyectos, así como también pautar el desarrollo del trabajo de manera efectiva.

❖ Fundamentos

La metodología Kanban se basa en una serie de principios que la diferencian del resto de metodologías ágiles [16]:

1. No se debe mandar material defectuoso a los procesos posteriores.
2. Los procesos requerirán solo lo esencial.
3. Procesar la cantidad exacta requerida por el proceso siguiente.
4. Balancear la producción.
5. Kanban es un medio para evitar especulaciones.
6. Estabilizar y racionalizar el proceso.

Estos puntos llevados a entornos de proyectos software se traducirían en:

- **Calidad garantizada.** Todo lo que se hace debe salir bien a la primera, no hay margen de error. De aquí a que en Kanban no se premie la rapidez, sino la calidad final de las tareas realizadas. Esto se basa en el hecho que muchas veces cuesta más arreglarlo después que hacerlo bien a la primera.
- **Reducción del desperdicio.** Kanban se basa en hacer solamente lo justo y necesario, pero hacerlo bien. Esto supone la reducción de todo aquello que es superficial o secundario.
- **Mejora continua.** Kanban no es simplemente un método de gestión, sino también un sistema de mejora en el desarrollo de proyectos, según los objetivos a alcanzar.
- **Flexibilidad.** Lo siguiente a realizar se decide del backlog (o tareas pendientes acumuladas), pudiéndose priorizar aquellas tareas entrantes según las necesidades del momento (capacidad de dar respuesta a tareas imprevistas)

❖ Reglas

Principalmente son tres reglas las que hacen que Kanban destaque como metodología de desarrollo de proyectos y procesos software [17]:

1. **Mostrar el proceso:** Kanban se basa en el desarrollo incremental, dividiendo el trabajo en partes. Una de las principales aportaciones es que utiliza técnicas visuales para ver la situación de cada tarea. Normalmente cada una de esas partes se escribe en un post-it y se pega en una pizarra, estos post-it suelen tener información variada, aparte de la descripción, debieran tener la estimación de la duración de la tarea.

La pizarra tiene tantas columnas como estados por los que puede pasar la tarea.

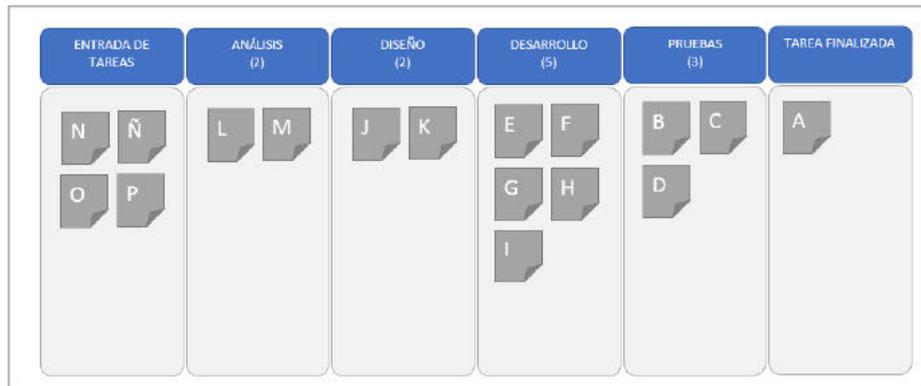


Figura 9. Muro de tareas KANBAN.

El objetivo de esta visualización es que quede claro el trabajo a realizar, en qué está trabajando cada persona, que todo el mundo tenga algo que hacer y el tener clara la prioridad de las tareas.

2. **Limitar el trabajo en curso (WIP):** Quizás una de las principales ideas del Kanban es que el trabajo en curso (Work In Progress) debería estar limitado, es decir, que el número máximo de tareas que se pueden realizar en cada fase debe ser algo conocido. A ese número de tareas se le llama límite del WIP, además, para empezar con una nueva tarea alguna otra tarea previa debe haber finalizado.

En la anterior figura de ejemplo el número límite del WIP se ha colocado entre paréntesis debajo del nombre de cada tarea. Por ejemplo, ver que el límite del WIP para las pruebas es 3.

La idea es centrarse en cerrar tareas y no en comenzar tareas. Por ello limitar el WIP impide empezar cosas hasta que se han cerrado aquellas en las que se está trabajando. Lo complicado es encontrar el mejor límite WIP.

3. **Optimizar el flujo de trabajo:** El tiempo que se tarda en terminar cada tarea se debe medir, a ese tiempo se le denomina "lead time" y cuenta desde que se hace una petición hasta que se hace la entrega.

Gracias a estas tres reglas podremos controlar los procesos y detectar los posibles cuellos de botella y solventarlos en la medida de lo posible sin que afecte demasiado a la dinámica de los proyectos que se llevan a cabo usando Kanban.

3.4.3. XP

❖ Historia

La programación extrema o eXtreme Programming (XP) es un enfoque de la ingeniería de software formulado por Kent Beck, autor del primer libro sobre la materia, *Extreme Programming Explained: Embrace Change* (1999) [18]. Es el más destacado de los procesos ágiles de desarrollo de software. Al igual que éstos, la programación extrema se diferencia de las metodologías tradicionales principalmente en que pone más énfasis en la adaptabilidad que en la previsibilidad.

Los defensores de XP consideran que los cambios de requisitos sobre la marcha son un aspecto natural, inevitable e incluso deseable del desarrollo de proyectos. Creen que ser capaz de adaptarse a los cambios de requisitos en cualquier punto de la vida del proyecto es una aproximación mejor y más realista que intentar definir todos los requisitos al comienzo del proyecto e invertir esfuerzos después en controlar los cambios en los requisitos.

❖ Fases

El ciclo de vida ideal de XP tiene seis fases [19]:

1. **Exploración:** En esta fase, los clientes plantean a grandes rasgos las historias de usuario que son de interés para la primera entrega del producto. Al mismo tiempo el equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto. Se prueba la tecnología y se exploran las posibilidades de la arquitectura del sistema construyendo un prototipo. La fase de exploración toma de pocas semanas a pocos meses, dependiendo del tamaño y familiaridad que tengan los programadores con la tecnología.
2. **Planificación de la Entrega:** En esta fase el cliente establece la prioridad de cada historia de usuario, y correspondientemente, los programadores realizan una estimación del esfuerzo necesario de cada una de ellas. Se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente. Una entrega debería obtenerse en no más de tres meses. Esta fase dura unos pocos días.
3. **Iteraciones:** Esta fase incluye varias iteraciones sobre el sistema antes de ser entregado. El Plan de Entrega está compuesto por iteraciones de no más de tres semanas. En la primera iteración se puede intentar establecer una arquitectura del sistema que pueda ser utilizada durante el resto del proyecto. Esto se logra escogiendo las historias que fueren la creación de esta arquitectura, sin embargo, esto no siempre es posible ya que es el cliente quien decide qué historias se implementarán en cada iteración (para maximizar el valor de negocio). Al final de la última iteración el sistema estará listo para entrar en producción.
4. **Producción:** La fase de producción requiere de pruebas adicionales y revisiones de rendimiento antes de que el sistema sea trasladado al entorno del cliente. Al mismo tiempo, se deben tomar decisiones sobre la inclusión de nuevas características a la versión actual, debido a cambios durante esta fase. Es posible que se rebaje el tiempo que toma cada iteración, de tres a

una semana. Las ideas que han sido propuestas y las sugerencias son documentadas para su posterior implementación (por ejemplo, durante la fase de mantenimiento).

5. **Mantenimiento:** Mientras la primera versión se encuentra en producción, el proyecto XP debe mantener el sistema en funcionamiento al mismo tiempo que desarrolla nuevas iteraciones. Para realizar esto se requiere de tareas de soporte para el cliente. De esta forma, la velocidad de desarrollo puede bajar después de la puesta del sistema en producción. La fase de mantenimiento puede requerir nuevo personal dentro del equipo y cambios en su estructura.
6. **Muerte del Proyecto:** Es cuando el cliente no tiene más historias para ser incluidas en el sistema. Esto requiere que se satisfagan las necesidades del cliente en otros aspectos como rendimiento y confiabilidad del sistema. Se genera la documentación final del sistema y no se realizan más cambios en la arquitectura. La muerte del proyecto también ocurre cuando el sistema no genera los beneficios esperados por el cliente o cuando no hay presupuesto para mantenerlo.

❖ Prácticas

- **El juego de la planificación:** Es un espacio frecuente de comunicación entre el cliente y los programadores. El equipo técnico realiza una estimación del esfuerzo requerido para la implementación de las historias de usuario y los clientes deciden sobre el ámbito y tiempo de las entregas y de cada iteración. Esta práctica se puede ilustrar como un juego, donde existen dos tipos de jugadores: Cliente y Programador. El cliente establece la prioridad de cada historia de usuario, de acuerdo con el valor que aporta para el negocio. Los programadores estiman el esfuerzo asociado a cada historia de usuario. Se ordenan las historias de usuario según prioridad y esfuerzo, y se define el contenido de la entrega y/o iteración, apostando por enfrentar lo de más valor y riesgo cuanto antes. Este juego se realiza durante la planificación de la entrega, en la planificación de cada iteración y cuando sea necesario reconducir el proyecto.
- **Entregas pequeñas:** La idea es producir rápidamente versiones del sistema que sean operativas, aunque obviamente no cuenten con toda la funcionalidad pretendida para el sistema pero si que constituyan un resultado de valor para el negocio. Una entrega no debería tardar más de 3 meses.
- **Metáfora:** En XP no se enfatiza la definición temprana de una arquitectura estable para el sistema. Dicha arquitectura se asume evolutiva y los posibles inconvenientes que se generarían por no contar con ella explícitamente en el comienzo del proyecto se solventan con la existencia de una metáfora. El sistema es definido mediante una metáfora o un conjunto de metáforas compartidas por el cliente y el equipo de desarrollo. Una metáfora es una historia compartida que describe cómo debería funcionar el sistema. Este conjunto de nombres ayuda a la nomenclatura de clases y métodos del sistema.

- **Diseño simple:** Se debe diseñar la solución más simple que pueda funcionar y ser implementada en un momento determinado del proyecto. La complejidad innecesaria y el código extra debe ser removido inmediatamente. Kent Beck dice que en cualquier momento el diseño adecuado para el software es aquel que: supera con éxito todas las pruebas, no tiene lógica duplicada, refleja claramente la intención de implementación de los programadores y tiene el menor número posible de clases y métodos.
- **Pruebas:** La producción de código está dirigida por las pruebas unitarias. Las pruebas unitarias son establecidas antes de escribir el código y son ejecutadas constantemente ante cada modificación del sistema. Los clientes escriben las pruebas funcionales para cada historia de usuario que deba validarse. En este contexto de desarrollo evolutivo y de énfasis en pruebas constantes, la automatización para apoyar esta actividad es crucial.
- **Refactorización (Refactoring):** La refactorización es una actividad constante de reestructuración del código con el objetivo de remover duplicación de código, mejorar su legibilidad, simplificarlo y hacerlo más flexible para facilitar los posteriores cambios. La refactorización mejora la estructura interna del código sin alterar su comportamiento externo.
- **Programación en parejas:** Toda la producción de código debe realizarse con trabajo en parejas de programadores. Las principales ventajas de introducir este estilo de programación son: muchos errores son detectados conforme son introducidos en el código (inspecciones de código continuas), por consiguiente la tasa de errores del producto final es más baja, los diseños son mejores y el tamaño del código menor (continua discusión de ideas de los programadores), los problemas de programación se resuelven más rápido, se posibilita la transferencia de conocimientos de programación entre los miembros del equipo, varias personas entienden las diferentes partes sistema, los programadores conversan mejorando así el flujo de información y la dinámica del equipo, y finalmente, los programadores disfrutan más su trabajo.
- **Propiedad colectiva del código:** Cualquier programador puede cambiar cualquier parte del código en cualquier momento. Esta práctica motiva a todos a contribuir con nuevas ideas en todos los segmentos del sistema, evitando a la vez que algún programador sea imprescindible para realizar cambios en alguna porción de código.
- **Integración continua:** Cada pieza de código es integrada en el sistema una vez que esté lista. Así, el sistema puede llegar a ser integrado y construido varias veces en un mismo día. Todas las pruebas son ejecutadas y tienen que ser aprobadas para que el nuevo código sea incorporado definitivamente. La integración continua a menudo reduce la fragmentación de los esfuerzos de los desarrolladores por falta de comunicación sobre lo que puede ser reutilizado o compartido.
- **40 horas por semana:** Se debe trabajar un máximo de 40 horas por semana. No se trabajan horas extras en dos semanas seguidas. Si esto ocurre, probablemente está ocurriendo un problema que debe corregirse. El trabajo extra desmotiva al equipo. Los proyectos que requieren trabajo extra para intentar cumplir con los plazos suelen al final ser entregados con

retraso. En lugar de esto se puede realizar el juego de la planificación para cambiar el ámbito del proyecto o la fecha de entrega.

- **Cliente in-situ:** El cliente tiene que estar presente y disponible todo el tiempo para el equipo. Gran parte del éxito del proyecto XP se debe a que es el cliente quien conduce constantemente el trabajo hacia lo que aportará mayor valor de negocio y los programadores pueden resolver de manera inmediata cualquier duda asociada. La comunicación oral es más efectiva que la escrita, ya que esta última toma mucho tiempo en generarse y puede tener más riesgo de ser mal interpretada.
- **Estándares de programación:** XP enfatiza la comunicación de los programadores a través del código, con lo cual es indispensable que se sigan ciertos estándares de programación (del equipo, de la organización u otros estándares reconocidos para los lenguajes de programación utilizados). Los estándares de programación mantienen el código legible para los miembros del equipo, facilitando los cambios.

3.5. Modelos Tradicionales vs Agiles

Los modelos tradicionales tienen muchas cosas en común pero cada uno de ellos tiene sus propias características que lo hacen particular como veremos a continuación con la comparativa entre CMMI, ITIL y SPICE [20] [21]:

CARACTERÍSTICAS	MODELOS		
	CMMI DEV	ITIL	SPICE
Desarrollador	Software Engineering Institute (SEI)	Office of Government Commerce (OGC)	International Organization for Standardization (ISO)
Versión vigente	V 1.3	V 3	15504-9
Arquitectura	CMMI DEV es una de las 3 constelaciones CMMI	5 libros o fases	5 partes
Procesos	22 áreas de procesos agrupadas en 4 categorías	24 procesos agrupados en 5 libros	9 procesos agrupados en 3 categorías
Certificación	Evaluación por evaluadores acreditados por SEI	Evaluación por evaluadores acreditados por OGC	Evaluación por evaluadores acreditados por ISO
Objetivo	Mejora del proceso,	Gestión y mejora de procesos del ciclo de	Valoración del proceso y guía para

	determinación de la capacidad del contratista	vida de servicios software	la mejora
--	---	----------------------------	-----------

Tabla 3. Características generales de 3 modelos tradicionales; CMMI-DEV, ITIL y SPICE.

Los modelos ágiles SCRUM, KANBAN y XP también comparten características generales aunque tienen algunas diferencias que los hacen diferenciarse entre ellos como podremos ver en la siguiente tabla comparativa:

CARACTERÍSTICAS	MODELOS		
	SCRUM	KANBAN	XP
Facilitador	Scrum Master	N/A	Coach, Big boss
Administrador Requerimientos	Product Owner	N/A	Cliente
Equipo Proyecto	Equipo de Desarrollo	N/A	Programador y Tester
Prácticas/Reglas	9	3	12
Limitación Work In Progress	Limitación por iteración	Limitación por estado	Limitación por iteración
Estimación	Obligatoria	Opcional	Obligatoria
Objetivo	Gestión y mejora de procesos estableciendo un marco de trabajo	Finalizar tareas prioritarias y evitar cuellos de botella	Gestión del desarrollo del software

Tabla 4. Características generales de 3 metodologías ágiles; SCRUM KANBAN y XP.

Hemos visto tres metodologías tradicionales y tres metodologías ágiles, estas metodologías actualmente son de las más utilizadas en la industria del software, aunque no por ello deben ser las mejores o las más apropiadas para nuestra organización como veremos más adelante. Para poder seleccionar la metodología más apropiada para nuestra organización debemos tener en cuenta las principales características que encontraremos al poner en práctica un tipo de metodología frente a otro.

A continuación mostramos una tabla resumen [22] con las principales características de los dos tipos de metodologías que hemos estudiado, a tener

en cuenta a la hora de seleccionar el modelo de proceso más adecuado para nuestra organización:

METODOLOGÍA	
ÁGIL	TRADICIONAL
Pocos artefactos. El modelado es prescindible, modelos desechables	Más artefactos. El modelado es esencial, mantenimiento de modelos
Pocos Roles, mas genéricos y flexibles	Más Roles, más específicos
No existe un contrato tradicional, debe ser bastante flexible	Existe un contrato prefijado
Cliente es parte del equipo de desarrollo (además in-situ)	El cliente interactúa con el equipo de desarrollo mediante reuniones
Orientada a proyectos pequeños. Corta duración (o entregas frecuentes), equipos pequeños (<10 integrantes) y trabajando en el mismo sitio	Aplicable a proyectos de cualquier tamaño, pero suelen ser especialmente efectivas en proyectos grandes y con equipos posiblemente dispersos
La arquitectura se va definiendo y mejorando a lo largo del proyecto	Se promueve que la arquitectura se defina tempranamente en el proyecto
Énfasis en los aspectos humanos, el individuo y el trabajo en equipo	Énfasis en la definición del proceso, roles, actividades y artefactos
Basadas en heurísticas provenientes de prácticas de producción de código	Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo
Se esperan cambios durante el proyecto	Se espera que no ocurran cambios de gran impacto durante el proyecto

Tabla 5. Comparativa de las características generales de todos los modelos tradicionales y ágiles.

Capítulo 4

Resolución

Para poder seleccionar el modelo de proceso más adecuado para nuestra organización no es suficiente con conocer la existencia de algún modelo, sino que debemos tener en cuenta una serie de aspectos importantes que desarrollaremos a continuación. También debemos tener en cuenta la comparativa entre metodologías tradicionales y ágiles para la selección del modelo que más se ajusta a las características de nuestra organización. Por último veremos la combinación de modelos como una solución a la gestión y mejora de procesos para alcanzar los objetivos de nuestra organización con éxito, ya que no siempre un único modelo podrá adaptarse a las necesidades de la organización.

4.1. Siete aspectos importantes a tener en cuenta

Debemos hablar de 7 aspectos importantes que debemos tener en cuenta para poder seleccionar el modelo de proceso más adecuado para nuestra organización:

1. Analizar la gestión actual en la organización.
2. Definir los objetivos de la organización.
3. Los modelos de proceso disponibles.
4. Seleccionar el modelo según los objetivos fijados.
5. Recibir formación sobre el modelo seleccionado.
6. Comprometerse con el método.
7. La implantación del modelo seleccionado.

4.1.1. Analizar la gestión actual en la organización

Al iniciar el cambio en cualquier organización que supone implantar un modelo de proceso, debemos hacer un análisis o diagnóstico previo dentro de la organización para conocer de una forma objetiva el estado actual en cuanto a las prácticas que se llevan a cabo en la gestión de los procesos software. Este análisis, debe realizarse no solo para comprender el estado actual de la organización, sino que también nos ayudara a comparar el resultado obtenido una vez hayamos implantado el modelo de proceso. Con un análisis posterior a la implantación del modelo de proceso seleccionado, tendremos la oportunidad de comparar los resultados obtenidos al evaluar las prácticas dentro de la organización antes y después de la implantación del modelo seleccionado.

Es obvio que el análisis de las prácticas actuales en la organización debe ser franco y no debe buscar la benevolencia, ni satisfacer la opinión de ningún

miembro de la organización, sino que debe reflejar la realidad de la organización para poder definir el contexto al que debemos hacer frente a la hora de implantar un modelo de proceso. Este análisis o diagnóstico [23] puede hacerse mediante:

- **Entrevistas:** deben enfocarse de tal forma que el entrevistado entienda la importancia del cambio que se desea realizar en la organización hacia la gestión y mejora de procesos para que pueda aportar su punto de vista crítico en cuanto a las prácticas actuales. Además, el entrevistado podrá contribuir a la mejora de procesos comprendiendo la importancia del análisis que se está llevando a cabo. Estas entrevistas deben realizarse a todos los niveles de la organización.
- **Método de análisis en grupo:** nos permitirá conocer la aprobación o conformidad de las ideas sobre la mejora de procesos, además de la inconformidad o desacuerdo de los distintos grupos en la organización, puesto que lo planteado en estas dinámicas de grupo serán aceptadas o rechazadas no solo por una persona, sino por todo el grupo.
- **Cuestionarios:** El analista puede utilizar un cuestionario en varios momentos, no solo para el diagnóstico de la situación actual, sino que puede ser utilizado en el trabajo de sistemas para obtener un consenso para identificar una dirección o un área para un estudio más profundo, para realizar una auditoría posterior a la implementación y para identificar requerimientos más específicos.
- **Observaciones:** sobre las personas en el momento de realizar su trabajo. La observación es una técnica para descubrir hechos que tienen una gran aceptación por parte de los usuarios. Puede hacerse de varias formas, se pueden recorrer las áreas de trabajo y tomar notas al lado de las personas, cosas y actividades, se puede observar a una persona o a una actividad sin que se percate de la presencia del observador y sin que haya interacción con el usuario, también puede observar una operación sin que haya interacción pero la persona que está siendo observada es plenamente consciente de este hecho y finalmente se puede observar e interactuar con las personas que están siendo observadas.

4.1.2. Definir los objetivos de la organización

Para seleccionar el modelo de proceso adecuado en nuestra organización necesitamos definir los objetivos que pretendemos alcanzar con la implantación de un modelo de proceso. Nos podremos apoyar en el análisis de la gestión actual para enfocar la selección del modelo y solventar las debilidades detectadas en dicho análisis. Los resultados obtenidos nos indicarán el tipo de modelo que necesitaremos implantar, bien sea del tipo tradicional o ágil, si necesitaremos implantar un modelo para conseguir una certificación o para

mejorar las distintas prácticas que se realizan en la organización para conseguir un producto de calidad.

La definición de los objetivos en la organización nos permitirá compartirlos con todos los miembros del equipo que participará en la implantación del modelo seleccionado, de forma clara y tangible. Es muy importante que el personal de la organización comprenda la situación actual de la empresa, las desventajas de las prácticas llevadas a cabo hasta la fecha frente a la implantación de un modelo de proceso. Esto es fundamental para alcanzar los objetivos fijados por la organización, las personas se deben implicar en el desarrollo de las nuevas prácticas marcadas por el modelo de proceso que vamos a seleccionar. Esto se consigue haciendo al equipo participe de los objetivos a alcanzar, involucrándoles en los resultados, logrando así que todos los miembros estén en la misma línea de actuación.

Definir claramente los objetivos de nuestra organización es fundamental para las prácticas llevadas a cabo en todos sus niveles, es por ello, que debemos redactar correctamente estos objetivos, con argumentos sólidos sin ambigüedades. Una vez que hemos determinado las metas que deseamos alcanzar, ya podemos concretar cómo hacerlo mediante la selección del modelo de proceso que más se ajuste a los objetivos marcados. Ya hemos visto los dos tipos de modelos, alguna muestra de ellos y sus principales características para ayudarnos a seleccionar el modelo de proceso más adecuado para nuestra organización.

4.1.3. Modelos de Proceso disponibles

Una vez que en la organización se decide implantar un modelo de proceso para obtener productos de calidad optimizando los recursos disponibles, nos encontramos con uno de los principales problemas a la hora de seleccionar un modelo, ¿qué modelo de proceso es el más adecuado? Para poder responder a esta pregunta correctamente, deberíamos conocer las distintas opciones que tenemos a nuestra disposición, algo que aparentemente es obvio.

Es decir, si no conocemos las metodologías existentes, difícilmente podremos aplicar el modelo de proceso más adecuado para nuestra organización. Por ello, es necesario realizar un ejercicio de investigación bien sea con los propios medios de la organización o a través de personal cualificado en el área de gestión y mejora de procesos, como podrían ser los profesionales del área de consultoría especializados en la gestión y mejora de procesos software.

4.1.4. Seleccionar el Modelo según los objetivos fijados

En principio, todos los modelos de procesos que existen fueron creados para poder utilizarlos en cualquier organización, pero podremos comprobar que

existen modelos de procesos más apropiados para según qué circunstancias y características tenga la organización donde pretendemos implantarlo. Según las necesidades y los objetivos que vayamos a cubrir, tendremos que escoger entre dos grupos de modelos que hemos explicado con detalle en este trabajo, las metodologías tradicionales, también conocidas como pesadas por su poca flexibilidad ante proyectos en constante cambio que exigen una adaptabilidad mayor en plazos cortos de tiempo, y las metodologías ágiles enfocadas precisamente a la flexibilidad y adaptabilidad al constante cambio en los proyectos.

Entender el contexto donde pretendemos implantar el modelo de proceso nos ayudara en la correcta selección. El contexto vendrá definido por el análisis inicial donde se especifica la situación actual de la organización y los objetivos que se han definido. Esto significa que debemos tener en cuenta la situación en la que se encuentra la organización y hacia donde se pretende llegar con la implantación del modelo de proceso, dos factores importantes a la hora de seleccionar el modelo que pretendemos implantar en nuestra organización.

Es posible que nuestra organización necesite implantar distintos modelos de proceso para cubrir las necesidades y poder alcanzar los objetivos fijados. Comprobaremos que la convivencia de modelos de procesos puede ayudarnos a utilizar las ventajas que presentan unos modelos frente a otros en nuestro beneficio, superando así los distintos problemas relacionados con la gestión y mejora de procesos en la organización.

4.1.5. Formarse en el Modelo seleccionado

Una vez seleccionado el modelo de proceso, deberán formarse todos los empleados que vayan a estar implicados en la implantación y utilización de dicho modelo. Es importante que los colaboradores en la organización comprendan el impacto que tendrá el uso del modelo de proceso que se ha seleccionado y que mejor forma que contar con la formación que podrán recibir de profesionales con experiencia en la implantación y uso del modelo que hemos seleccionado. Además de comprender el impacto que tendrá el uso del modelo seleccionado, la formación deberá ayudar a los colaboradores a adquirir los conocimientos fundamentales para el correcto uso del modelo.

La formación no tiene por qué ser igual para todos los colaboradores, puesto que no todos realizaran las mismas funciones, aunque si deben tener una visión general del modelo y cada colaborador deberá recibir una formación más específica en cuanto a su área de trabajo, detallando las implicaciones del trabajo que realizaran como parte de los procesos descritos en el modelo. Con los conocimientos generales y específicos adquiridos en la formación, los colaboradores están preparados para formar parte de los cambios que se pretenden implantar en la organización e iniciarse en las nuevas prácticas corporativas.

4.1.6. Compromiso con el método

Con la correcta aplicación de una metodología conseguimos establecer un marco de trabajo común a todas las personas que participan en los proyectos de nuestra organización.

Si en nuestra organización, hasta la fecha, no se aplicaba una metodología como marco de trabajo, nos encontramos con una tarea que puede ser compleja y el grado de complejidad vendrá marcado fundamentalmente por el grado de compromiso con el método de cada una de las personas que participaran en la implantación, seguimiento y defensa del mismo, puesto que supone un cambio importante en las prácticas y procesos dentro de la organización con todo lo que ello supone.

Para lograr el éxito mediante una metodología, el compromiso por parte del personal involucrado y principalmente por las personas que dirigen el cambio, es vital, por lo que es necesario comprender el verdadero alcance al implantar una metodología en cualquier organización.

Las personas que dirigen el cambio deben velar por:

- El desarrollo de la misión, la visión y los valores, y su papel de modelo de referencia de una cultura de excelencia.
- Su compromiso personal con el desarrollo, puesta en práctica y mejora continua del sistema de gestión de la organización.
- Su implicación con clientes, asociados y representantes de la organización.
- La motivación, el apoyo y el reconocimiento al personal

4.1.7. La implantación del Modelo seleccionado

Para implantar el modelo nos apoyaremos en el estudio de situación realizado inicialmente, de tal forma que seleccionaremos una de las áreas de la organización donde sea más necesaria la mejora de los procesos que se venían haciendo hasta el momento. Dentro del área a mejorar se implantará el modelo gradualmente comenzando con un proyecto para poder medir y controlar el impacto de la implantación del nuevo modelo de procesos, de este modo será más evidente el cambio que se introduce en la organización.

Es importante seleccionar correctamente el área y el proyecto donde comenzar la implantación puesto que servirá de referencia para el resto de áreas, por lo que debe hacerse un trabajo minucioso y controlado para poder comparar los resultados entre el antes y el después de la implantación del modelo de procesos seleccionado.

4.2. La selección del Modelo adecuado.

Para poder seleccionar el modelo adecuado para nuestra organización y establecer un marco de trabajo corporativo, un buen comienzo es decantarnos por un tipo de metodología; tradicional o ágil. Para elegir un tipo de metodología frente a otro debemos apoyarnos en los objetivos fijados por la organización y los recursos que esta dispone para gestionar los proyectos:

- **Acreditar un estándar de calidad:** Si pretendemos conseguir una certificación que acredite un estándar de calidad reconocido internacionalmente para poder presentarnos a licitaciones, por ejemplo, hoy en día debemos fijarnos en las metodologías tradicionales, puesto que son las más extendidas y aceptadas. De lo contrario si la organización no pretende certificar un estándar podremos elegir cualquier tipo de metodología que nos convenga según sus características.
- **Establecer buenas prácticas:** Con la implantación de cualquiera de los dos tipos de metodologías podremos lograr establecer una serie de buenas prácticas que nos ayudara a obtener una estructura bien organizada, con la que poder planificar los distintos procesos involucrados.
- **Mejorar los procesos existentes:** Cualquier de los dos tipos de metodología debería de mejorar los procesos existentes puesto que es uno de los principales objetivos por los cuales se crean los modelos de procesos.
- **Gestionar cualquier tipo de proyecto:** A priori, los dos tipos de metodologías podrían gestionar todo tipo de proyectos, pero hemos visto que los modelos tradicionales están enfocados a la gestión de proyectos grandes o complejos y los modelos ágiles se enfocan más a la gestión de proyectos pequeños o poco complejos. La complejidad está relacionada con el número de componentes que deben intervenir para poder crear soluciones, por lo que un número elevado de componentes implicará un número elevado de relaciones entre ellos creando arquitecturas complejas en la mayoría de los casos.
- **Roles y jerarquías:** Los dos tipos de metodologías necesitan roles y jerarquías, pero las metodologías tradicionales acentúan más la necesidad en cuanto a la definición de roles y jerarquías para usar correctamente sus procesos y normas frente a los modelos ágiles que por el contrario no requieren de roles bien diferenciados.
- **Número de empleados:** Los equipos formados por menos de 10 personas deberían enfocarse más hacia la aplicación de metodologías ágiles y por el contrario los equipos formados por más de 10 personas deberían seleccionar modelos tradicionales. Aunque no siempre será así debido al

contexto de cada equipo, en líneas generales, los equipos pequeños seleccionarían un modelo ágil y los equipos grandes seleccionarían modelos tradicionales.

- **Contratos flexibles:** Si es necesario para el correcto desarrollo de un proyecto que el tipo de contrato sea lo más flexible posible, permitiendo constantes adaptaciones en los requisitos iniciales, parece obvio declinarse por metodologías ágiles para gestionar estos proyectos y por el contrario, si los contratos son poco flexibles con requisitos estables deberíamos seleccionar un modelo de proceso del tipo tradicional.
- **La relación con el cliente:** Si la relación con el cliente es periódica, mediante reuniones porque no se encuentra en nuestra organización, y no contamos con la presencia del cliente casi como parte del equipo que desarrolla el proyecto solicitando correcciones, mejoras o adaptaciones, deberíamos seleccionar una metodología tradicional y de lo contrario seleccionaríamos una metodología ágil.

CUESTIONARIO			RESPUESTAS	
			SI	NO
6	¿Necesitamos certificar los procesos software involucrados en nuestra organización?			
7	¿Buscamos establecer buenas prácticas sin certificación?			
8	¿Queremos mejorar los procesos existentes en nuestra organización?			
9	¿Gestionaremos cualquier tipo de proyecto sin importar la complejidad del mismo?			
10	¿Disponemos de suficientes recursos como para establecer roles y jerarquias?			
11	¿Hay mas de 10 empleados en nuestra organización?			
12	¿Habrá requisitos definidos inicialmente con apenas variaciones?			
13	¿El cliente interactua habitualmente como parte del equipo?			

Figura 10. Ejemplo del cuestionario para ayudar en la selección del tipo de metodología.

Estos aspectos nos guiarán en la selección del modelo adecuado para nuestra organización, pero son orientativos y no debemos aferrarnos a ellos para justificar nuestra selección, aunque si nos sirven para ayudarnos a entender mejor los tipos de modelos existentes y tomar una decisión lo más acertada posible a partir de los 7 aspectos importantes del apartado anterior y estos rasgos generales que mencionamos a la hora de seleccionar un modelo de procesos.

4.3. La convivencia de Modelos

¿Qué pasa si después de analizar nuestra organización y algunos modelos de proceso, llegamos a la conclusión de que ningún puede cubrir todas las necesidades para llegar a los objetivos marcados? Podría suceder, que en vez de necesitar un único modelo de procesos, necesitésemos dos o más.

Cada vez son más las organizaciones que aplican varios modelos para la gestión y mejora de procesos, ya que en muchas de ellas es necesario gestionar todo tipo de proyectos asegurando niveles de calidad competitivos. Para ello, es necesaria la aplicación de diferentes modelos en los distintos departamentos que componen la organización. Nos centraremos en la convivencia de dos modelos para entender mejor el concepto de convivencia de modelos.

La idea es confeccionar un puzle de modelos dentro de la organización que nos permita aprovechar las metodologías existentes para gestionar y mejorar los procesos software, mediante la combinación de ventajas que nos ofrecen los modelos implicados.

Actualmente es corriente la convivencia entre modelos tradicionales, modelos ágiles y modelos tradicionales y ágiles. Siempre que exista una convivencia de modelos de procesos es habitual encontrar un modelo establecido a nivel corporativo que será utilizado como marco de trabajo corporativo y podremos usar otro modelo para gestionar todos los procesos involucrados en la organización. El tipo de modelo que se establecería como marco de trabajo corporativo podría ser de cualquiera de los dos tipos, tradicional o ágil, según las características de la propia organización.

Por ejemplo, si la organización busca obtener una certificación que acredite el cumplimiento de un estándar determinado como pudiera ser ITIL, se establecería dicho modelo como marco de trabajo corporativo y podríamos utilizar un modelo ágil para gestionar ciertos procesos de la organización sin por ello dejar de cumplir los procesos y procedimientos del estándar corporativo establecido (en este caso ITIL).

4.3.1. CMMI-DEV e ITIL

Dos de los modelos tradicionales más usados, el modelo CMMI-DEV gestiona y mejora los procesos de ingeniería de sistemas, ingeniería del software, desarrollo de productos integrados y adquisición de software. Mientras que con ITIL podremos garantizar la explotación del producto software, logrando una gestión eficiente de la infraestructura y los servicios IT.

La convivencia de estos dos modelos es posible puesto que las desventajas de un modelo se cubren con las ventajas del otro. En este caso CMMI-DEV se enfocaría más al desarrollo del software e ITIL se enfocaría más a los servicios IT que la organización pueda ofrecer como producto. Además, la organización podría lograr certificarse en la aplicación de los dos modelos si así lo decidiera,

acreditando las buenas practicas que recogen estos modelos para crear productos y servicios de calidad.

Otro motivo por el cual una organización pudiera decidir implantar dos modelos como estos, seria por el hecho de abarcar el mayor número de licitaciones posibles. Esto se debe a la asociación entre los modelos y regiones donde se aceptan como estándar “valido”, es decir, hay unos modelos validos frente a otros a pesar de contar con certificaciones reconocidas internacionalmente. Principalmente encontramos esta diferenciación entre regiones en dos zonas, EEUU y Europa, CMMI es más común encontrarla como requisito de licitaciones en EEUU, mientras que ITIL y SPICE se solicita en licitaciones en Europa.

4.3.2. CMMI-DEV y SPICE

El modelo CMMI-DEV proporciona un conjunto de buenas prácticas con las que poder desarrollar y crear productos software de calidad, mientras que SPICE proporciona un marco de trabajo para la evaluación de los procesos y establece un modelo de referencia para el ciclo de vida del software.

A pesar de que los dos modelos no son equivalentes, si cuentan con ciertas similitudes y es posible combinarlos gracias a las fuertes relaciones que existen entre ellos, aunque serán necesarios procesos de adaptación para compatibilizarlos. Ambos modelos nos ofrecen visibilidad de los proyectos, comunicación, planificación, calidad, coordinación de proyectos, aumento de satisfacción del cliente y competitividad entre muchos otros factores. Al igual que en el caso anterior, podremos certificar los procesos en los que hemos implantado los dos modelos, otorgando a la organización mayor nivel de competitividad frente a las organizaciones que no están certificadas.

4.3.3. CMMI-DEV y SCRUM

En este caso vamos a combinar un modelo tradicional y uno ágil, por lo que cabe esperar que el objetivo será establecer como marco de trabajo el modelo CMMI-DEV y para implantarlo usaremos SCRUM como apoyo, logrando así establecer un conjunto de buenas prácticas dentro de nuestra organización. Como ya hemos visto, SCRUM es una metodología ágil enfocada a la gestión de proyectos, es por ello que deberíamos usarla para las áreas de proceso especificadas en CMMI-DEV relacionadas con la gestión de proyectos.

Al igual que en casos anteriores donde convivían dos modelos tradicionales estableciendo uno de ellos como marco de trabajo corporativo, combinando CMMI-DEV y SCRUM, podremos conseguir una certificación en nuestra organización al contar con CMMI-DEV como marco de trabajo corporativo.

4.3.4. ITIL y XP

A priori son dos metodologías con filosofías muy distintas, ITIL es un modelo que establece unas prácticas siguiendo una estructura ordenada y bien definida, mientras que XP nos da una serie de pautas para poder adaptar los constantes cambios en la dirección de los proyectos dando sentido a las prácticas que se llevan a cabo que puedan parecer desordenadas. Es evidente que las dos metodologías parten de filosofías distintas, ITIL da un enfoque estructurado y ordenado en contraposición a XP que se enfoca a la adaptación al cambio sin necesidad de contar con una estructura rígida al cambio. Por ello, son dos metodologías que podremos combinar, porque donde ITIL no puede establecerse como modelo debido a su naturaleza estructurada, XP logra los mejores resultados.

En este caso ITIL haría de marco de trabajo corporativo y XP se utilizaría en las áreas de gestión y desarrollo de proyectos, ofreciendo a la organización la capacidad de gestionar cualquier tipo de proyecto software. Además, la organización podría conseguir una certificación ITIL apoyándose en los procesos definidos en XP, logrando un mayor nivel de competitividad.

4.3.5. SCRUM y KANBAN

Entre todas las combinaciones de metodologías que mencionamos en este trabajo, esta es la más extendida y aceptada. De hecho se utiliza una palabra específica para denominar esta combinación de metodologías, SCRUMBAN. Al igual que con el resto de combinación de modelos, en este caso se persigue el objetivo de combinar las prácticas que otorgan mayor beneficio a la organización desde el punto de vista de la gestión y mejora de procesos.

Utilizaríamos SCRUM como marco de trabajo puesto que entre las dos metodologías es la más estricta en cuanto a la organización y planificación, especificando la duración de las reuniones y la frecuencia de las mismas entre otras prácticas descritas. Sin embargo, KANBAN no especifica roles ni reuniones que deberíamos aplicar para llevar a cabo la gestión de proyectos, estos son los principales factores que hacen de la combinación de estas dos metodologías tan favorables. Con SCRUM conseguimos gestionar los proyectos usando una serie de prácticas definidas en la metodología y con KANBAN establecemos un flujo de trabajo continuo y constante.

4.3.6. SCRUM y XP

La combinación de estas metodologías es posible siempre que establezcamos como marco de trabajo SCRUM y utilicemos XP para la gestión de procesos

responsables del desarrollo de software. La metodología SCRUM está enfocada a la gestión de proyectos con unas reglas bien definidas para llevar a cabo una serie de prácticas estructuradas, mientras que XP se enfoca a la gestión del desarrollo de software.

Capítulo 5

Conclusión y Líneas Futuras

Para acertar en la selección del modelo de procesos más adecuado para nuestra organización es importante definir el contexto donde pretendemos implantarlo. Hemos visto 7 aspectos generales a tener en cuenta para ayudarnos a definir el contexto, debemos conocer los objetivos que deseamos alcanzar con la implantación del modelo que vamos a seleccionar, los tipos de proyectos que deseamos gestionar, los medios de los que dispone nuestra organización y tantas variables como seamos capaces de definir y concretar. Por ello, es tan importante realizar un análisis en profundidad del contexto donde pretendemos implantar el modelo que vayamos a seleccionar, es decir, debemos definir las características de nuestra organización antes de poder seleccionar un modelo de procesos.

Dependiendo de las características de la organización y sus objetivos, será más fácil o difícil seleccionar el modelo de procesos, ya que habrá organizaciones que por el tipo de proyectos que gestiona deberá implantar un modelo en concreto como pueden ser aquellas organizaciones que se enfocan casi en exclusiva a participar en licitaciones que piden específicamente un modelo en particular. Pero este no es el caso más genérico, de ahí la problemática existente a la hora de seleccionar un estándar apropiado para nuestra organización.

La industria del software tiene más de 50 metodologías de desarrollo diferentes, es un número demasiado grande para compararlo en nuestro trabajo. Los seis métodos que hemos visto y comparado aquí, han tenido algunos éxitos y han tenido algunos fracasos también. En general, la familia Ágil y los métodos que hacen hincapié en la velocidad han logrado su objetivo y son bastante rápidos. Los métodos que hacen hincapié en la calidad, como las metodologías tradicionales, también han logrado sus objetivos y entregan muy pocos defectos.

Ningún método parece ser una panacea universal que puede tener éxito en todos los tamaños y tipo de aplicación de software. Una vez realizado el análisis del estado y características de nuestra organización, deberíamos poder seleccionar entre los dos tipos de metodologías, tradicionales o ágiles. Este sería un buen punto de partida para seleccionar el modelo de procesos más adecuado para nuestra organización.

Por último, hemos visto que la convivencia de modelos es usada para lograr gestionar todo tipo de proyecto, por este motivo en la actualidad la combinación de modelos y metodologías está más extendida. En esta línea enfocaremos el próximo estudio, la combinación de modelos y sus beneficios, analizando su situación actual y las ventajas al implantar en las organizaciones este “puzle de modelos y metodologías”.

BIBLIOGRAFÍA Y REFERENCIAS

- Roger S. Pressman. 2010. *Ingeniería del software: Un enfoque práctico, 7ma Edición.*
 - Jan van Bon; Arjen de Jong; Axel Kolthof; Mike Pieper; Ruby Tjassing; Annelies van der Veen; Tienneke Verheijen. *Fundamentos de la Gestión de Servicios de TI basada en ITIL v3.* Editorial ITSM Library (Enero 2008)
 - J. Tuya, I. R. Román, J. D. Cosín (2007). *Técnicas Cuantitativas para la Gestión en la Ingeniería del Software.* La Coruña, España: Netbiblo S. L.
1. Cuevas, G.; Amescua, A.; Cerrada, J.A.; San Feliu, T.; Calvo-Manzano, J.A.; Arcilla, M.; García, M. *Gestión del Proceso Software.* Editorial Centro de Estudios Ramón Areces S.A. 2002.
 2. *Ingeniería del software: metodologías y ciclos de vida, Marzo 2009, INTECO (Instituto Nacional de Tecnologías de la Comunicación.* https://www.incibe.es/file/N85W1ZWFHifRgUc_oY8_Xg
 3. Capers Jones, *Evaluating Agile and Scrum with Other Software Methodologies.* Mar 20, 2013. <http://www.infoq.com/articles/evaluating-agile-software-methodologies>
 4. Elena Argüelles, Antonio Sepúlveda. *Laboratorio Nacional de Calidad del Software, Diagnóstico de la situación de la calidad del software en la industria española.* INTECO (Instituto Nacional de Tecnologías de la Comunicación). Volumen 4, Número 2 (especial X JICS), septiembre, 2008. REICIS (Revista Española de Innovación, Calidad e Ingeniería del Software).
 5. Alfredo Calvo, Angélica Su, *Análisis del ROI: Una Herramienta para Justificar la Mejora de Procesos.* Enero-febrero 2007. http://sg.com.mx/revista/13/analisis-del-roi-una-herramienta-para-justificar-la-mejora-procesos#.VU-WY_ntmko
 6. Karenny Brito Acuña, *Selección de metodologías de desarrollo para aplicaciones web en la facultad de informática de la universidad de Cienfuegos, 2009*
 7. About CMMI for Development. 06/03/2014. <http://www.sei.cmu.edu/reports/10tr033.pdf>
 8. About ITIL V3. 10/02/2014. <http://www.osiatis.es/blogTI/2010/07/26/nuevo-curso-online-til-v3/>
 9. EcuRed (n.d.). ISO 15504. Consultado el 22 de Noviembre de 2013 en http://www.ecured.cu/index.php/ISO_15504
 10. L. R. Corral (n.d.). ISO 15504 Software Process Improvement Capability dEtermination. Consultado el 22 de Noviembre de 2013 en http://www.luiscorral.webs.com/3_2.pdf

11. ISO (2004). ISO/IEC 15504-4:2004(en). Consultado el 22 de Noviembre de 2013 en <https://www.iso.org/obp/ui/#iso:std:iso-iec:15504:-4:ed-1:v1:en>
12. About Agile Manifesto. 17/06/2015. <http://agilemanifesto.org/iso/es/>
13. The New New Product Development Game, por Hirotaka Takeuchi (Hitotsubashi University) y Ikujiro Nonaka. Harvard Business Review, Enero-Febrero de 1986
14. About SCRUM. 17/06/2015. <http://www.proyectosagiles.org/que-es-scrum>
15. About Kanban. 23/04/2015. https://docs.google.com/presentation/d/1d4ZqcQ9s4S_EaATPue5LKs0Ymqc7s_q0belzlrhraKs/embed?slide=id.i65
16. About Kanban. 24/04/2015. <http://kanban-pinkys.es/tl/Definici%F3n-y-Funci%F3n-.htm>
17. About Kanban. 24/04/2015. <http://www.javiergarzas.com/2011/11/kanban.html>
18. About XP. 09/04/2015. <http://www.extremeprogramming.org/>
19. Kent Beck, Addison Wesley. 2000. Extreme Programming Explained.
20. Modelos de Evaluación y Mejora de Procesos: Análisis Comparativo, Manuel de la Villa¹, Mercedes Ruiz² e Isabel Ramos
21. La madurez de los servicios TI, Antoni Lluís Mesquida, Antonia Mas, Esperança Amengual, Departamento de Matemáticas e Informàtica, Universitat de les Illes Balears
22. Patricio Letelier, M^a Carmen Penadés, Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP). Volumen 5, Número 26, abril-junio 2006. <http://www.cyta.com.ar/ta0502/v5n2a1.htm>
23. Argueta Cortés Jairo Isacar, Estudio General del Sistema. 2009. <http://www.openboxer.260mb.com/tutorialSW/tema1.2.html>

ACRONIMOS

CAR: Causal Analysis and Resolution
CCTA: Central Computer and Telecommunications Agency
CIs: Configuration Items
CM: Configuration Management
CMMI: Capability Maturity Model Integration
CMMI-DEV: Capability Maturity Model Integration for Development
CMMI-ACQ: Capability Maturity Model Integration for Acquisition
CMMI-SVC: Capability Maturity Model Integration for Services
DAR: Decision Analysis and Resolution
EXIN: Exameninstituut voor Informatica
IDEAL: Initiating Diagnosing Establishing Acting Learning
IEEE: Institute of Electrical and Electronics Engineers
IPM: Integrated Project Management
ISEB: Information Systems Examination Board
ISO: International Organization for Standardization
ISO IEC: International Organization for Standardization International Electrotechnical Commission
ISO TR: International Organization for Standardization Technical Report
ITIL: Information Technology Infrastructure Library
itSMF: Information Technology Management Forum
JIT: Just In Time
MA: Measurement and Analysis
MSF: Microsoft Solution Framework
LCS: Loyalist Certification Services
OGC: Office of Government Commerce
OPD: Organizational Process Definition
OPF: Organizational Process Focus
OPM: Organizational Performance Management
OPP: Organizational Process Performance
OT: Organizational Training
PI: Product Integration
PMC: Project Monitoring and Control
PP: Project Planning
PPQA: Process and Product Quality Assurance
QPM: Quantitative Project Management
RD: Requirements Development
REQM: Requirements Management
ROI: Return On Investment

RSKM: Risk Management
RUP: Rational Unified Process
SAM: Supplier Agreement Management
SECAM: Systems Engineering Capability Assessment Method
SEI: Software Engineering Institute
SIP: Service Improvement Plan
SLA: Service Level Agreement
SPICE: Software Process Improvement and Capability dEtermination
TI: Tecnología de la Información
TIC: Tecnologías de la Información y Comunicaciones
TS: Technical Solution
UCs: Underpinning Contract
VAL: Validation
VER: Verification
WIP: Work In Progress
XP: eXtreme Programming