



TESIS DOCTORAL

K-ETeC: Adaptación del método Lean Kanban, e inclusión de técnicas de coaching e inteligencia emocional grupal, para la mejora de la productividad en la gestión del servicio de mantenimiento de software

Autor:

D. Roberto Hens Pato

Directora:

Dra. Esperanza Marcos Martínez

Codirector:

Dr. Juan Manuel Vara Mesa

Programa de Doctorado en Tecnologías de la Información y
las Comunicaciones

Escuela Internacional de Doctorado

2020

La Dra. Da. Esperanza Marcos y el Dr. D. Juan M. Vara, profesores del Departamento de Ciencias de la Educación, Lenguaje, Cultura y Artes, Ciencias Histórico-Jurídicas y Humanísticas y Lenguas Modernas de la Universidad Rey Juan Carlos, directora y codirector, respectivamente, de la Tesis Doctoral: **“K-ETeC: Adaptación del método Lean Kanban, e inclusión de técnicas de coaching e inteligencia emocional grupal, para la mejora de la productividad en la gestión del servicio de mantenimiento de software”** realizada por el doctorando D. Roberto Hens Pato,

HACEN CONSTAR QUE:

Esta Tesis Doctoral reúne los requisitos para su defensa y aprobación.

En Madrid, a 21 de septiembre de 2020.

Fdo.: Dra. Da. Esperanza Marcos Martínez



Fdo.: Dr. D. Juan Manuel Vara Mesa



“ Research is to see what everybody else has seen,
and to think what nobody else has thought.

— ALBERT SZENT-GYÖRGYI

Resumen

El principal objetivo de esta Tesis Doctoral es la mejora de la productividad de los equipos de mantenimiento de software a través de la adaptación y aplicación del método Lean Kanban, mediante la implantación de las prácticas que se consideren más oportunas atendiendo a las características del equipo, así como la sostenibilidad de la cultura de mejora continua a través de la aplicación de técnicas de coaching e inteligencia emocional grupal.

En el siglo actual, gran parte de la economía de los países desarrollados se basa en la prestación de servicios de alto nivel, como el caso de los profesionales de las tecnologías de la información y las comunicaciones. Esto ha generado la necesidad de aumentar la productividad de estas actividades al igual que ocurriera en su momento con las actividades asociadas al sector primario y secundario. En este sentido, los profesionales del desarrollo del software llevan muchos años aplicando diferentes alternativas para mejorar su eficiencia.

Durante la primera década del presente siglo la ingeniería del software ha evolucionado sensiblemente tras la firma, en el año 2001, del Manifiesto Ágil para Desarrollo de Software que facilitó el desarrollo de diversas metodologías dentro del denominado *paraguas ágil*. Una de estas propuestas metodológicas que surgió a mediados de la década es el método Lean Kanban para desarrollo de software, que permite mejorar la productividad de los equipos, mostrando especial atención al entorno multi-proyecto tan habitual en equipos que prestan servicios de mantenimiento de software.

En este contexto, la Tesis Doctoral que se presenta expone los resultados de la adaptación y aplicación del método Lean Kanban en un equipo que ofrece el servicio de mantenimiento de software, a través de la monitorización de una iniciativa durante nueve meses. Se trata de un caso de estudio en el que se ha realizado una adopción progresiva en un equipo, analizando la evolución de los principales indicadores del método a lo largo de dicho periodo y prestando especial interés, no sólo a la mejora del proceso de desarrollo, sino también a la mejora de la productividad del equipo.

A lo largo de la investigación llevada a cabo como base de la presente Tesis Doctoral, se ha podido constatar una mejora significativa en diferentes aspectos asociados al proceso de desarrollo, a la organización del equipo de trabajo, a las prácticas de ingeniería y a la calidad del producto software.

Las mejoras citadas anteriormente adquieren mayor relevancia si tenemos en cuenta que trabajan simultáneamente en distintos frentes como son la mejora de la calidad interna del producto o la mejora de la organización del equipo de trabajo. Mientras, los indicadores básicos por los que podemos medir y analizar el rendimiento de un equipo de trabajo que presta servicios de mantenimiento de software, se mantienen constante.

De este modo, se ha podido constatar que gracias al continuo análisis del proceso de desarrollo y la implementación de bucles de retroalimentación, se ha llevado a cabo un elevado número de acciones que permiten la optimización del proceso, sin desviar al equipo de su misión original de prestación de un servicio de mantenimiento de software. Podemos concluir que dedicar los esfuerzos para iniciar ese motor de cambio continuo ha tenido un resultado más positivo que la definición de un exhaustivo plan de mejora global. En este sentido, los cambios fueron apareciendo gradualmente, como posibles soluciones a los problemas planteados en las retrospectivas semanales.

Uno de los objetivos claves en este tipo de iniciativas es evitar que, transcurrido un tiempo desde su aplicación, el equipo entre en la rutina perdiendo con ello la inercia en la cultura de mejora continua alcanzada. La selección de un equipo de *outsourcing* nos hacía partir de la premisa de que estos equipos, habitualmente aislados dentro de la organización cliente y alejados de su empresa matriz, acostumbran a tener problemas para identificar objetivos alineados con los de estas dos organizaciones. Consideramos que esta desconexión provoca una baja motivación evitando que la iniciativa de mejora continua persista en el tiempo.

Para solucionar este problema, consideramos que el coaching y la inteligencia emocional grupal pueden servir de ayuda, por lo que diseñamos un proceso que combina ambas técnicas al que hemos denominado *Emotional Team Coaching process* (ETeC). Por una parte, el coaching de equipos nos permitió actuar desde fuera, a modo de espejo para el equipo. Por otra parte, trabajar sobre la emocionalidad grupal permitió transformar las emociones negativas en positivas, empoderando al equipo y enfocándolo hacia una visión y meta común. Gracias a la aplicación del proceso ETeC se consiguieron establecer objetivos comunes que generaron una mayor cohesión y motivación en los miembros del equipo.

Abstract

The main objective of this Doctoral Thesis is to improve the productivity of the software maintenance teams through the adaptation and application of the Lean Kanban method, by the implementation of the practices that are considered the most appropriate according to the characteristics of the equipment, as well as the sustainability of the culture of continuous improvement through the application of coaching techniques and group emotional intelligence.

In the current century, a large part of the economy of developed countries is based on the provision of high-level services, as is the case for professionals in information and communication technologies. This has generated the need to increase the productivity of these activities, as occurred at the time with activities associated with the primary and secondary sector. In this sense, software development professionals have been applying different alternatives for many years to improve their efficiency.

During the first decade of this century, software engineering has evolved significantly after the signing, in 2001, of the Agile Manifesto for Software Development, which facilitated the development of various methodologies within the so-called *agile umbrella*. One of these methodological proposals that emerged in the middle of the decade is the Lean Kanban method for software development, which allows to improve the productivity of the teams, paying special attention to the multi-project environment so common in teams that provide software maintenance services.

In this context, the Doctoral Thesis presents the results of the adaptation and application of the Lean Kanban method in a team that offers the software maintenance service through the monitoring of an initiative over nine months. This is a case study in which a progressive adoption has been carried out in a team, analysing the evolution of the main indicators of the method throughout the period and paying special interest, not only to the improvement of the development process, but also to the improvement of team productivity.

Throughout the research carried out as the basis of this Doctoral Thesis, a significant improvement has been found in different aspects associated with the development process, the organization of the work team, engineering practices and product quality.

The improvements mentioned above acquire greater relevance if we take into account that they work simultaneously on different aspects, such as improving the

internal quality of the product or improving the organization of the work team. Meanwhile, the basic indicators by which we can measure and analyse the performance of a team that provides software maintenance services remains constant.

In this way, it has been found that thanks to the continuous analysis of the development process and the implementation of feedback loops, a large number of actions have been carried out that allow the optimization of the process, without diverting the team from its original mission of provision of a software maintenance service. We can conclude that dedicating the efforts to start this engine of continuous change has had a more positive result than the definition of a comprehensive global improvement plan. In this sense, the changes appeared gradually, as possible solutions to the problems raised in the weekly retrospectives.

One of the key objectives in this type of initiative is to prevent the team from getting into a routine after some time has elapsed, thereby losing inertia in the culture of continuous improvement achieved. The selection of an outsourcing team made us start from the premise that these teams, usually isolated within the client and far from their parent company, tend to have problems identifying objectives aligned with those of these two organizations. We consider that this disconnection causes low motivation, preventing the continuous improvement initiative from persisting over time.

To solve this problem, we consider that group coaching and emotional intelligence can be helpful, so we designed a process that combines both techniques, which we have called Emotional Team Coaching process (ETeC). On the one hand, team coaching has acted from the outside, as a mirror for the team. On the other hand, working on group emotionality to transform negative emotions into positive ones, empowering the team and focusing it towards a common vision and goal. Thanks to the application of the ETeC process, it was possible to establish common objectives that generated greater cohesion and motivation in the team members.

AGRADECIMIENTOS

Quiero tener un agradecimiento con mis directores ya que, sin su aliento, consejos, ayuda y participación habría sido imposible llegar a escribir las últimas líneas de esta memoria. Un proyecto de estas dimensiones compartido con otras responsabilidades se hace tan duro que, si no hubiera sido por vuestros apoyos, en algún momento habría tirado la toalla.

Me hace una gran ilusión finalizar esta Tesis bajo la dirección de Esperanza puesto que me lleva acompañando académicamente desde mi segundo año de Universidad... allá... a principios de los 90, cuando empezamos a colaborar en el Laboratorio de Bases de Datos Avanzadas de la Universidad Carlos III. Durante tres décadas, a su lado he sido: alumno de grado, colaborador, profesor asociado, doctorando, alumno de master y de nuevo doctorando. Qué poder decir, su nombre está tan ligado a mi relación con la Universidad que no la entendería sin ella. Y es por esto que mi agradecimiento va más allá de la dirección de esta Tesis, se remonta a tres décadas.

Por otro lado, Juancho, quiero agradecerle la ayuda que me has prestado a lo largo de este, en ocasiones, tortuoso camino. Ha sido todo un placer que fueras codirector de mi Tesis. Son también muchos años juntos, de hecho, compartimos *pupitre* como alumnos de doctorado, allá por 2004, cuando lancé mi primer asalto al título de Doctor. Otros compromisos profesionales me llevaron a aparcar los estudios hasta volver a retomarlos en el 2015, donde, convertido en un experimentado investigador me has arropado hasta alcanzar la meta. Tú y yo sabemos que, si no hubieras estado a mi lado, no la habría cruzado.

El trabajo de investigación que ha concluido con la presentación de esta Tesis Doctoral ha sido fruto de la colaboración de muchas personas, por lo que, me gustaría agradecer el esfuerzo de todas las que, de una u otra forma, han formado parte de esta investigación. Por un lado, mi equipo de desarrollo, que se embarcó en esta aventura de 17 meses en la que hemos sufrido, avanzado, retrocedido, pero sobre todo aprendido muchísimo. Igualmente, a todas las personas de mi empresa y de la organización cliente que facilitaron la puesta en marcha de esta investigación. Un caso de estudio de esta envergadura no puede llevarse a cabo sin el visto bueno y el apoyo de las personas que, indirectamente, podían verse afectadas por su resultado. Gracias por haber confiado lo suficiente en mí como para dejar adentrarme en terreno desconocido, es todo un lujo difícil de encontrar en el entorno empresarial real.

Como no puede ser de otra forma, mis agradecimientos especiales son para mi mujer, Susana. Corazón, tú has vivido a mi lado todas mis etapas como estudiante: la ingeniería técnica, la superior, el primer master, mis primeros estudios de doctorado, el segundo master y finalmente, este último tramo que concluye en esta Tesis Doctoral. No entendería mi vida sin tu compañía y de igual forma creo que no habría concluido mi trayectoria académica sin tu apoyo incondicional. Soy consciente que cada uno de los caminos académicos que he iniciado ha supuesto un esfuerzo para los dos. Todas las horas que he tenido que dedicar a esta investigación ha sido a costa de *robártelas* a ti, ya sea para poder disfrutar de un mayor tiempo de ocio o para ayudarte con nuestros hijos. Sin embargo, no ha existido ni una queja durante estas tres décadas de estudios, sino todo lo contrario, nada más que ayuda incondicional y palabras de aliento. Igualmente, tengo que agradecer a nuestros hijos la paciencia que han tenido conmigo. El trabajo, las clases y para colmo el doctorado... estoy convencido que en ocasiones tanta acumulación, sin pretenderlo, habrán hecho cambiar mi carácter y eso lo sufren los que están más cerca. Espero poder recompensároslo a los tres como os lo merecéis.

No quiero dejar de tener unas palabras para mis padres. Si bien es cierto que durante esta última etapa prácticamente no han sido conscientes de lo que tenía entre manos, el ejemplo de esfuerzo y sacrificio que me han dado durante toda la vida, creo que ha sido fundamental para alcanzar todos mis logros, pasados y futuros.

Gracias de todo corazón.

R.

Índice

1. INTRODUCCIÓN	1
1.1 Planteamiento del problema y enfoque.....	1
1.2 Hipótesis y Objetivos.....	5
1.3 Contexto de la investigación.....	6
1.3.1 <i>Proyectos y etapas de la investigación</i>	7
1.4 Método de Investigación	9
1.4.1 <i>Ejecución de la investigación: caso de estudio</i>	11
1.4.2 <i>Definición del caso de estudio</i>	13
1.5 Esquema de la Tesis	16
2. CONCEPTOS PREVIOS Y ESTADO DEL ARTE.....	19
2.1 Contexto del método Lean Kanban	19
2.1.1 <i>Lean Product Development</i>	23
2.1.2 <i>Theory of Constraints (TOC)</i>	24
2.1.3 <i>Metodologías ágiles</i>	28
2.2 <i>Outsourcing</i> de mantenimiento de software	29
2.3 Coaching e inteligencia emocional grupal	33
2.4 Trabajos relacionados	35
3. APLICACIÓN DEL MÉTODO LEAN KANBAN	41
3.1 Protocolo de recogida de datos	42
3.2 Ejecución del caso de estudio	43
3.2.1 <i>Puesta en marcha de la iniciativa</i>	45
3.2.2 <i>Métricas utilizadas en el proceso de mejora</i>	47
3.2.3 <i>Análisis de las fuentes de variabilidad</i>	49
3.2.4 <i>Visualización</i>	52
3.2.5 <i>Gestionar el flujo</i>	56
3.2.6 <i>Implementar bucles de retroalimentación y mejorar colaborativamente</i>	58
3.3 Análisis de resultados	60
3.4 Lecciones aprendidas.....	74
4. EL PROCESO ETEC	81
4.1 Protocolo de recogida de datos	81
4.2 Ejecución del caso de estudio	83

4.2.1	<i>Fase de Diagnóstico</i>	86
4.2.2	<i>Fase de Intervención</i>	94
4.3	Análisis de resultados	101
4.4	Lecciones aprendidas.....	105
5.	CONCLUSIONES	111
5.1	Contribuciones principales y resultados científicos.....	111
5.2	Análisis de la consecución de los objetivos	113
5.3	Trabajos futuros.....	119
	APÉNDICE A. BIBLIOGRAFÍA	123
	APÉNDICE B. ACRÓNIMOS	137

Lista de Figuras

Figura F-1.1 Definición del caso de estudio.....	14
Figura F-2.1 Principios y prácticas del método Lean Kanban.....	20
Figura F-2.2 Influencias sobre el método Lean Kanban.....	20
Figura F-3.1. PDSA Cycle.....	41
Figura F-3.2. Ejemplo del tablero Kanban en sus primeras iteraciones	47
Figura F-3.3. Primera versión del ticket	53
Figura F-3.4. Segunda versión del ticket	53
Figura F-3.5. Tercera versión del ticket.....	54
Figura F-3.6. Versión estandarizada del ticket	54
Figura F-3.7. Representación de autoevaluación del equipo	59
Figura F-3.8. Cycle-Time (Todas las clases excepto Ligeras y Bugs)	62
Figura F-3.9. Cycle-Time (Clase Estándar).....	62
Figura F-3.10. Lead Time (Clase Estándar)	66
Figura F-3.11. Salida en función de la estimación relativa.....	67
Figura F-3.12. Comparación peticiones en espera.....	70
Figura F-3.13. Distribución recepción de Peticiones.....	70
Figura F-3.14. Tendencia del Lead-Time y Cycle-Time	71
Figura F-3.15. Tickets terminados en un ciclo por Clase de Servicio	72
Figura F-3.16. Encuesta de consecución de los objetivos	73
Figura F-4.1. Distribución de sesiones del proceso ETeC.....	86
Figura F-4.2. Percepción del entorno de trabajo percibida por el equipo.	88
Figura F-4.3. Abstracción de la percepción del entorno de trabajo	89
Figura F-4.4. Valoración de las dimensiones	90
Figura F-4.5. Plan de acción.....	100
Figura F-4.6. Alineación de metas en diferentes niveles organizativos.....	102
Figura F-4.7. Alineación de metas personales dentro de un equipo	103

Lista de Tablas

Tabla T-1.1. Factores de éxito y fallo de los proyectos	2
Tabla T-1.2. Detalles del Proyecto ELASTIC	7
Tabla T-1.3. Detalles del Proyecto SICOMORo-CM.....	8
Tabla T-1.4. Detalles del Proyecto MADRID	8
Tabla T-2.1. Comparación de los trabajos analizados	36
Tabla T-3.1. Métricas definidas para el caso de estudio.....	49
Tabla T-3.2. Lead-Time asociado a cada clase de servicio	50
Tabla T-3.3. Clases de servicio y ANS asociados	51
Tabla T-3.4. Ejemplos de políticas del tablero	56
Tabla T-4.1. FCE principales en la implementación de la Mejora Continua.....	82
Tabla T-4.2. Preguntas evaluadas por debajo de la media.....	91
Tabla T-4.3. Respuestas a la pregunta de objetivos de la reunión	92
Tabla T-4.4. Respuestas a las preguntas abiertas.....	93
Tabla T-4.5. Conclusiones de la fase de evaluación	94
Tabla T-4.6. Factores de mejora para aumentar la motivación.....	95
Tabla T-4.7. Resultado de Philips 6-3-5.....	98

Introducción

En la presente Tesis Doctoral se propone la aplicación del método Lean Kanban para la mejora de la productividad de los equipos a través de un enfoque ágil de desarrollo y el uso de técnicas de coaching e inteligencia emocional grupal para la sostenibilidad de la cultura de mejora continua.

En la sección 1.1 presentamos las principales motivaciones que condujeron a la decisión de realizar esta investigación. En la sección 1.2 se exponen la hipótesis y los objetivos que se derivan directamente de ella, y la sección 1.3 describe el contexto en el que se ha desarrollado este trabajo. En la sección 1.4 se resume el método de investigación y, por último, la sección 1.5 proporciona una visión general sobre el resto de esta Tesis Doctoral.

1.1 Planteamiento del problema y enfoque

A lo largo de la breve historia de la ingeniería del software, los investigadores han mantenido una **búsqueda** constante de **nuevos métodos** y **técnicas** que han ido permitiendo solucionar los numerosos problemas que recurrentemente aparecían en la gran mayoría de los proyectos de desarrollo de software y que provocó que en la década de 1960 se acuñara el término de “crisis del software”. En aquella época, ya existía la percepción de la importancia del software en muchas de las actividades de la sociedad (Mens, 2008), sin embargo, la detección de un elevado número de problemas en los proyectos de desarrollo dejaba entrever la necesidad urgente de darles solución.

Una de las causas que motivó la crisis del software podemos encontrarla en que, durante aquellos primeros años de progresos y avances en el campo de la computación, se requería de un impulso tan rápido que prácticamente la totalidad de las *nuevas ideas* eran aceptadas como aportaciones interesantes (Glass, 1994).

Dado el enorme gasto (y el vertiginoso ritmo de crecimiento) que empezaban a requerir los proyectos asociados con el desarrollo de software, desde la OTAN se organizó una conferencia con el objetivo de arrojar luz sobre los numerosos problemas existentes en el área y debatir las posibles técnicas, métodos y aproximaciones que podrían conducir a encontrar una solución o aliviar el problema (Naur et al., 1968). Así, el principal objetivo de esta conferencia fue el establecimiento y definición de principios sólidos para el desarrollo de software que ayudasen a producir software confiable, eficiente y económicamente viable.

Como resultado, se impulsó la idea de adaptar los procesos de desarrollo software a los definidos en otras ingenierías que venían demostrando excelentes

2 K-ETeC

resultados de aplicación. En este contexto, fue el proceso de desarrollo en cascada, identificado por la clara separación entre las diferentes fases del proyecto, el enfoque mayoritariamente adoptado (Royce, 1987).

En la misma línea, el Departamento de Defensa (DoD) de Estados Unidos, emprendió un doble esfuerzo para acelerar las mejoras a corto y largo plazo en el desarrollo de software para las aplicaciones de sus sistemas de defensa. Para ello, se establecieron políticas y prácticas de gestión del ciclo de vida del software (De Roze & Nyman, 1978). De estos esfuerzos surgió CMMI, un modelo que permitía evaluar a los proveedores de software para, de algún modo, asegurar la calidad del software que producían (Carnegie Mellon University, 2009).

Sin embargo, el paso del tiempo acabó mostrando en muchos casos que altos niveles de madurez no garantizaban el éxito de los proyectos. En (Verner et al., 1999) se analizaban los procesos de desarrollo de 20 empresas que habían alcanzado niveles de madurez CMMI entre 1 y 4, concluyendo que los proyectos desarrollados por estas organizaciones adolecían de problemas serios, algunos de los cuales eran evitables mediante la correcta aplicación de técnicas de ingeniería del software. La tabla T-1.1 muestra los nueve factores principales localizados como causantes de los éxitos/fracasos de los proyectos analizados en este estudio.

Proyectos exitosos	Proyectos fallidos
Apoyo de la alta dirección	Falta de apoyo de la alta dirección
Implicación de los interesados	Falta de implicación de los interesados
Requisitos negociados y bien definidos	Requisitos no negociados y poco precisos
Director de Proyecto con experiencia	Director de Proyecto inexperto
Precisión de las estimaciones de los desarrolladores	Imprecisión de las estimaciones de los desarrolladores
Modelos de ciclo de vida adecuados	Modelos de ciclo de vida inapropiados
Gestión de riesgos	Falta de gestión de riesgos
Alta comunicación dentro de los equipos	Falta de comunicación dentro de los equipos
Baja rotación del personal	Alta rotación del personal

Tabla T-1.1. Factores de éxito y fallo de los proyectos

Por ello, en el año 2001 se plantea una nueva aproximación: el Manifiesto Ágil para Desarrollo del Software (Beck et al., 2001). Surge como respuesta a los problemas de las aproximaciones determinísticas anteriores, donde el excesivo control, la obligación del cierre de requisitos en fases tempranas y la planificación

descendente acababa generalmente en proyectos con muchísimo esfuerzo dedicado a la gestión pero con un desempeño desastroso (Appelo, 2011).

Alineado con el enfoque de desarrollo promovido por el Manifiesto Ágil, David J. Anderson adaptó y aplicó la Teoría de las Restricciones (TOC) (Goldratt & Cox, 1984) a un equipo de mantenimiento de software (Anderson & Dumitriu, 2005). Incorporó posteriormente elementos visuales como las tarjetas Kanban, desarrollando como resultado un nuevo marco de trabajo, el método Lean Kanban para el desarrollo de software, que desde entonces ha ido evolucionando y perfeccionándose.

Generalmente, la principal motivación para afrontar un cambio metodológico como la adopción del método Lean Kanban, proviene de la búsqueda de una mejora en la productividad del equipo, lo que revertirá en la mejora de la prestación del servicio.

En situaciones de recesión económica, como la sufrida en el 2008 y que debido al carácter cíclico de las crisis volverá a repetirse en los próximos años, en este caso agravada por la crisis sanitaria asociada a la pandemia que estamos padeciendo, la mejora de la productividad que pueden proporcionar los métodos tradicionales no es suficiente para paliar el deterioro que suelen experimentar los equipos de mantenimiento de software. El descenso en la capacidad productiva no se debe únicamente a la reducción del número de ingenieros del software que componen los equipos, sino que hay que añadir la pérdida de conocimiento provocada por la marcha de aquellos empleados que ya no forman parte de los equipos, así como las relaciones que mantenían con el resto de la organización, con la consecuente y drástica pérdida de eficiencia (Parise et al., 2006).

Metodológicamente hablando, una de las mejores opciones para dar respuesta a esta necesidad de mejora de la productividad es la adopción del **método Lean Kanban**. Este se caracteriza por perseguir la mejora continua, apoyándose para ello en ciclos de retroalimentación y en la entrega de valor al cliente con una frecuencia mayor (Leopold & Kaltenecker, 2015). Si bien es cierto que en la literatura podemos encontrar numerosos trabajos sobre la aplicación de los principios Lean en la industria manufacturera (Bhamu & Sangwan, 2014), no es así para el caso de su aplicación al mundo del desarrollo software. De este modo, uno de los objetivos de esta Tesis es proporcionar directrices y lecciones aprendidas sobre la adaptación y adopción del método Lean Kanban por equipos de mantenimiento de software, siguiendo el enfoque adoptado en su primera aplicación (Anderson & Dumitriu, 2005).

Aunque el método Lean Kanban es apropiado para la mayor parte de los proyectos (Anderson, 2011), la presente Tesis Doctoral analiza concretamente el **impacto sobre la productividad** por parte de los equipos de mantenimiento del software que trabajan bajo una modalidad de *outsourcing*. A su vez, propone el uso de técnicas de coaching e inteligencia emocional grupal para la **sostenibilidad de la mejora continua** que promueve el método y que, por el contexto en que desarrollan su labor estos equipos, suelen perder pasado el tiempo.

Por otro lado, el mantenimiento de software es una actividad que realizan trabajadores del conocimiento (ingenieros del software), por lo que el aumento de la productividad de los equipos no puede provenir, o al menos no sólo, de una mejora o mayor automatización de los procesos, sino que exige que impongamos la responsabilidad de su productividad en los propios trabajadores del conocimiento, gestionándose a sí mismos a través de su propia autonomía (Drucker, 1999).

A su vez, en la actualidad existe una creciente utilización del *outsourcing* como herramienta de gestión que surge, en respuesta a las demandas de formas más eficientes para abordar la competitividad de las empresas. Las organizaciones, reconociendo que no pueden ser altamente eficientes en todas las actividades y funciones involucradas en la producción de sus productos y servicios, están evolucionando sus estrategias basadas en *competencias centrales*: un conjunto de *habilidades y conocimientos* que ayudan a mantener su ventaja competitiva. Las actividades que no se encuentran dentro de sus competencias centrales son llevadas a cabo mediante la modalidad de *outsourcing*. Es decir, pueden ser proporcionados en su totalidad o en parte por un proveedor externo cuyas competencias centrales son las actividades y funciones destinadas a la subcontratación (Sharpe, 1997).

Sin embargo, a pesar de su crecimiento, la contratación externa a menudo está mal controlada, tiene un alto costo y una pérdida de calidad y rendimiento del servicio (Jacobs, 1994). El hecho de tener las actividades poco estratégicas subcontratadas, hace que la empresa pierda cierto control en su ejecución y, por supuesto, en los trabajadores que las llevan a cabo. La modalidad de *outsourcing* incorpora varios riesgos para el contratista: posibilidad de una gestión débil, costes ocultos, falta de aprendizaje organizacional y pérdida de capacidad de innovación entre otras (Earl, 1996). En particular, se produce cierta falta de alineación entre la visión del cliente y la de la consultora tecnológica que actúa como proveedora del personal. Ambas organizaciones deben ser capaces de entender y respetar las metas y requerimientos de la otra parte (Sparrow, 2012). En este escenario, la meta del equipo de desarrollo no suele encontrar reflejo en ninguna de las dos organizaciones y sin embargo es un elemento clave en la productividad organizacional (Pritchard et al., 1988).

Por ello, en esta Tesis Doctoral se aborda este problema mediante la definición de un **proceso que combina varias técnicas de coaching e inteligencia emocional grupal** para mejorar la motivación y cohesión de equipos de mantenimiento de software en modalidad *outsourcing*. La aplicación de este proceso permitirá paliar los problemas derivados de la falta de una misión clara y metas comunes, lo que ocasiona un estancamiento en el proceso de sostenibilidad de la mejora continua que promueve el método Lean Kanban.

1.2 Hipótesis y Objetivos

A continuación, se especifica la hipótesis principal de la presente Tesis Doctoral, así como los objetivos que se derivan de ésta.

La hipótesis formulada en esta Tesis Doctoral es que *“es posible **mejorar la productividad** de los equipos de mantenimiento de software en modalidad de *outsourcing* mediante la aplicación del método **Lean Kanban** y facilitar la **sostenibilidad** de la cultura de mejora continua, impulsada por la adopción del método, mediante la aplicación de técnicas de **coaching** e inteligencia emocional grupal”*.

Por lo tanto, el **principal objetivo** de esta Tesis Doctoral, directamente derivado de la hipótesis, es *la mejora de la productividad de un equipo de mantenimiento de software a través de la aplicación del método Lean Kanban, mediante la implantación de las prácticas que se consideren más oportunas atendiendo a las características del equipo, así como la sostenibilidad de la cultura de mejora continua a través de la aplicación de técnicas de coaching e inteligencia emocional grupal*.

Este objetivo principal se puede descomponer en los siguientes objetivos parciales:

- O1.** Estudios de trabajos previos
 - O1.1.** Estudio del método Lean Kanban.
 - O1.2.** Estudio de los equipos de mantenimiento de software en modalidad *outsourcing*.
 - O1.3.** Estudio de técnicas de coaching e inteligencia emocional grupal.
- O2.** Formación del equipo en el método Lean Kanban.
- O3.** Adaptación y aplicación del método Lean Kanban.
 - O3.1.** Análisis y selección de cambios a introducir en el equipo.
 - O3.2.** Impulso de los cambios seleccionados.

O3.3. Medición y análisis del impacto de la transformación sobre los indicadores de rendimiento.

O3.4. Análisis de resultados y lecciones aprendidas.

O4. Aplicación de técnicas de coaching e inteligencia emocional grupal.

O4.1. Ejecución de la fase de Diagnóstico.

O4.2. Ejecución de la fase de Intervención.

O4.3. Análisis de resultados y lecciones aprendidas.

1.3 Contexto de la investigación

El trabajo de investigación presentado en esta memoria se ha desarrollado en el seno del grupo de investigación Kybele-ISE, de la Universidad Rey Juan Carlos, al que pertenecen tanto el doctorando como los directores de la presente Tesis Doctoral. Para contextualizar su contenido, se destacan a continuación algunas de las líneas de investigación en las que el grupo viene desarrollando su actividad en los últimos años:

- **Ingeniería del software.** El grupo de investigación tiene intereses en diferentes áreas, como el desarrollo de herramientas y aplicaciones, la mejora de la calidad del software, las metodologías ágiles y la ingeniería dirigida por modelos. De entre estas áreas, la presente tesis doctoral se enmarca en las áreas de **calidad y metodologías ágiles**.
- **Ingeniería de servicios.** En los últimos años la actividad investigadora del grupo se ha centrado especialmente en la Ingeniería de Servicios, siendo varios los proyectos de investigación que el grupo ha desarrollado en esta línea. De entre las áreas temáticas que comprende esta línea, la presente tesis doctoral se enmarca fundamentalmente en el área de la **provisión de servicios TIC**.
- **Habilidades personales y mejora de equipos.** Dentro de esta línea de investigación, el grupo ha trabajado fundamentalmente en el diseño de nuevos métodos docentes para el desarrollo de habilidades personales y la definición de procesos que integren nuevas técnicas (técnicas teatrales, coaching, gamificación, etc.) para el desarrollo de habilidades personales y la gestión de equipos de mantenimiento de software. La presente Tesis Doctoral es, de hecho, uno de los principales trabajos del grupo en esta última área.

1.3.1 *Proyectos y etapas de la investigación*

Tal y como se expone en este mismo capítulo, la presente Tesis Doctoral se ha desarrollado aplicando el método de investigación en acción, donde el grupo de referencia ha sido un equipo de mantenimiento de software al que el doctorando pertenece, desempeñando el rol de jefe de proyecto. Esta vinculación ha permitido controlar en todo momento la aplicación de las propuestas formuladas en el contexto de la Tesis y recabar toda la información relacionada con sus resultados.

Además, el doctorando ha participado en varios de los proyectos de investigación desarrollados por el grupo durante este periodo. En particular, la tesis doctoral se enmarca fundamentalmente en los proyectos ELASTIC, SICOMORo-CM y MADRID.

El proyecto ELASTIC, financiado por el Ministerio de Economía y Competitividad, tuvo como objetivo crear una metodología ligera y basada en modelos, para la sistematización del proceso de servitización, la cual es una problemática a la que se enfrentan las organizaciones al afrontar un cambio de enfoque, desde una organización tradicional orientada a producto, a una organización moderna enfocada a servicios. En el contexto de este proyecto, el doctorando ha trabajado en el desarrollo de herramientas destinadas a la servitización de las TIC, en concreto a los equipos que prestan servicios de mantenimiento de software. Los detalles del proyecto se muestran en la Tabla 1.2.

<i>Título</i>	<i>ELASTIC: proceso ligero de servitización basado en modelos – aplicación a las TIC</i>
<i>Entidad financiadora</i>	Ministerio de Economía y Competitividad
<i>Referencia</i>	TIN2014-52938-C2-1-R
<i>Entidades participantes</i>	Universidad Rey Juan Carlos, IBM
<i>Inicio</i>	Enero de 2015
<i>Finalización</i>	Diciembre de 2017
<i>Investigador principal</i>	Dra. Esperanza Marcos

Tabla T-1.2. Detalles del Proyecto ELASTIC

Por su parte, el proyecto SICOMORo-CM financiado por la Comunidad de Madrid y el Fondo Social Europeo, tuvo como objetivo principal la construcción de metodologías, apoyadas con herramientas, que permitan el desarrollo de software confiable y de alta calidad utilizando un proceso riguroso que cubra todas sus fases. Los detalles de este proyecto se muestran en la Tabla T.1.3.

<i>Título</i>	<i>SICOMORo-CM: desarrollo de Sistemas Confiables mediante MOdelos y heRramientas avanzadas</i>
<i>Entidad financiadora</i>	Comunidad de Madrid y Fondo Social Europeo
<i>Referencia</i>	TIN2014-52938-C2-1-R
<i>Entidades participantes</i>	U. Complutense de Madrid, U. Rey Juan Carlos, U. Autónoma de Madrid, U. Politécnica de Madrid
<i>Inicio</i>	Octubre de 2014
<i>Finalización</i>	Septiembre de 2017
<i>IP proyecto coordinado</i>	Dr. Manuel Núñez
<i>IP nodo URJC</i>	Dra. Esperanza Marcos

Tabla T-1.3. Detalles del Proyecto SICOMORo-CM

Por último, el proyecto MADRID, continuación del proyecto ELASTIC, tiene como objetivo la definición de una metodología centrada en el usuario para el desarrollo de servicios. Para ello, se identifican una serie de técnicas y prácticas, adaptadas de métodos como el *Design Thinking* o la *gamificación*, orientadas a potenciar la creatividad y la participación del usuario en el diseño temprano de la solución. A continuación, tomando como entradas los artefactos generados en la fase creativa, se define un proceso de desarrollo ágil, adaptado a cualquier sector, que permita obtener un diseño de bajo nivel del servicio. Como parte del proyecto se pretende mostrar su especialización para el sector TIC y el desarrollo de un entorno de modelado que dará soporte a las diferentes fases del proceso propuesto. Los detalles de este proyecto se muestran en la tabla T.1.4.

<i>Título</i>	<i>MADRID - Marco metodológico centrado en el usuario para el diseño de servicios en la nueva sociedad digital</i>
<i>Entidad financiadora</i>	Ministerio de Educación y Ciencia
<i>Referencia</i>	TIN2017-88557-R
<i>Entidades participantes</i>	Universidad Rey Juan Carlos, IBM
<i>Inicio</i>	Enero de 2018
<i>Finalización</i>	Diciembre de 2020
<i>Investigador principal</i>	Dr. Juan Manuel Vara Dra. Valeria de Castro

Tabla T-1.4. Detalles del Proyecto MADRID

1.4 Método de Investigación

La ingeniería del software se trata de una disciplina relativamente nueva e inmadura en comparación con otras disciplinas de ingeniería. Dadas sus diferencias significativas con respecto al resto de ciencias empíricas y formales (al igual que les ocurre a otras ramas de ingeniería), provoca una importante dificultad para la aplicación directa de métodos clásicos en su investigación.

La International Software Engineering Research Network en su manifiesto específica (ISERN Manifiesto, 1993): “Como comunidad, hemos comenzado a reconocer que el software no puede producirse con una tecnología estándar, sino que debe desarrollarse con tecnologías adaptadas a los objetivos y características de proyectos particulares. En consecuencia, la investigación de ingeniería del software debe realizarse en un contexto experimental que nos permita observar y experimentar con las tecnologías en uso, comprender sus debilidades y fortalezas, adaptar las tecnologías a los objetivos y características de proyectos particulares, y agruparlos con evidencia obtenida empíricamente para mejorar su potencial de reutilización en proyectos futuros”.

Hasta el momento, la ingeniería del software ha hecho suyas muchas de las ideas, conceptos y métodos de investigación de otras disciplinas como la arquitectura o la ingeniería. Sin embargo, los sistemas de información disponen de unas propiedades de mutabilidad y adaptabilidad que no encontramos en las otras disciplinas, generando gran cantidad de problemas de diseños únicos y desafiantes que, con el tiempo, ha dejado ver la necesidad de la aplicación de ideas nuevas y creativas.

Aunque la aplicación del enfoque experimental en la ingeniería del software se remonta a los comienzos de la disciplina, fue en la década de los ochenta cuando se enfatiza la necesidad de su aplicación centrándose en experimentos que involucran sistemas software (productos, procesos y recursos de tipo software) (Šmite et al., 2010).

A continuación, se presenta el método de investigación utilizado para la realización de la presente Tesis Doctoral. En concreto, el método utilizado ha sido **Technical Action Research** (TAR) (Wieringa & Morali, 2012), una variante del método **Investigación-Acción** (Wood-Harper, 1985). TAR es el intento de escalar una propuesta sobre una determinada situación práctica al usarlo realmente en un problema particular.

Podemos definir el método de Investigación-Acción, de forma muy general, como la intervención en una situación real para mejorar esta situación y aprender de ella (Susman & Evered, 1978). Además del diagnóstico, Investigación-Acción

incluye una **intervención de los investigadores** para mejorar la situación y aprender de lo sucedido en el experimento.

Por su parte, TAR amplía las condiciones de una propuesta utilizándola realmente en un problema particular, lo que permite mostrar si un artefacto es aplicable en la práctica y, al ser utilizado en una empresa cliente, muestra que el artefacto es al menos relevante para esa empresa. Parte de una serie de asunciones:

- El problema para el que se diseña el artefacto no es único.
- Existe un modelo conceptual de estos problemas.
- El mundo real es lo suficientemente estable como para permitir la predicción de lo que sucedería si se usara el artefacto.

TAR aborda la resolución de problemas en la forma de un ciclo de ingeniería en el que los artefactos se comparan y evalúan contra objetivos estables (Wood-Harper, 1985). El ciclo de ingeniería es un proceso iterativo de decisión racional en el que el diseñador genera y evalúa diseños alternativos comparándolos con los objetivos del diseño. En las primeras iteraciones a través del ciclo de ingeniería, el diseñador hace supuestos de idealización para que sea más fácil encontrar un diseño.

Una vez que se ha dado una prueba de concepto, se mejora el diseño iterando a lo largo del ciclo, relajando gradualmente los supuestos, hasta que todos se puedan satisfacer fácilmente en la práctica. TAR utiliza el ciclo de ingeniería en las últimas etapas de este proceso iterativo, es decir, cuando el investigador prueba un artefacto usándolo para resolver el problema de un cliente.

De esta forma, TAR permite validar nuevos artefactos en condiciones de práctica. Para ello se deben poder responder preguntas de eficacia y utilidad sobre el artefacto en un contexto determinado, e investigar la solidez de las respuestas ante determinados cambios sobre el artefacto y el contexto inicial. TAR incorpora un ciclo de investigación para dar respuestas a las preguntas de validación enfocadas a la predicción del efecto del artefacto si fuera transferido al mundo real. Este ciclo de investigación tiene las siguientes fases:

Investigación del problema. En esta fase se determina cuál es la unidad de estudio, qué conceptos se utilizan para formular las preguntas de investigación sobre la unidad de estudio y lo que ya se sabe sobre estas preguntas. En TAR se parte de casos claros de problemas, es decir, se aplica la técnica en una empresa cliente de la que creemos que claramente cae dentro de la población.

Diseño de la investigación. El diseño de investigación en TAR consiste en adquirir acceso a una empresa cliente, acordar un objetivo de mejora para el ciclo del cliente, acordar qué hará el investigador por la empresa y cómo recopilará los

datos. Todo el ejercicio de TAR se basa en la premisa de que lo que el investigador aprende en este caso particular, proporcionará lecciones aprendidas que serán utilizables en el próximo caso. Durante el diseño de la investigación, el investigador debe pensar en cómo extraer las lecciones aprendidas que serán aplicables también a otros casos.

Validación del diseño de investigación. Es una evaluación del riesgo latente de no poder responder las preguntas de la investigación tras la ejecución del diseño de la investigación realizado en la fase anterior. Para realizar este análisis se deben responder a preguntas del tipo:

- ¿Las preguntas de investigación podrán ser respondidas a través de los datos producidos?
- ¿Los problemas de otras empresas serán lo suficientemente similar al de la empresa cliente?
- ¿Podrán beneficiarse el resto de empresas de las lecciones aprendidas tras la ejecución?

Las respuestas a este tipo de preguntas pueden orientar al investigador hacia la necesidad de adaptación de su diseño de investigación.

Ejecución de la investigación. Durante esta fase se lleva a cabo la ejecución del ciclo del cliente, para lo que será necesario acordar lugares, tiempo y recursos (humanos, materiales y técnicos) para realizar las tareas de la propuesta. Antes de finalizar la ejecución será necesaria una evaluación con el cliente para determinar el grado de consecución de sus objetivos. En nuestro caso, hemos optado por la realización de un caso de estudio para llevar a cabo este ciclo del cliente. En la sección 1.4.1 se explica el método de caso de estudio en sí, mientras que, la definición de nuestro caso de estudio particular se detalla en la sección 1.4.2.

Evaluación de resultados. Después de completar el ciclo del cliente, el investigador regresa al ciclo de investigación y analiza los resultados.

1.4.1 Ejecución de la investigación: caso de estudio

Un proceso experimental proporciona una base para el avance necesario del conocimiento. De este modo, la experimentación se realiza para ayudarnos a evaluar, predecir, comprender, controlar y mejorar el desarrollo de software, tanto en lo relativo al proceso, como al producto (Basili et al., 1986).

El objetivo de la experimentación es prescribir soluciones que puedan estar soportadas por un cuerpo de conocimiento validado por evidencia científica, de modo que la experimentación consiste en identificar las causas por las que se

producen determinados resultados. Al aplicarla a la ingeniería del software, permite identificar y facilitar la comprensión de distintas variables que entran en juego en la construcción de sistemas de información. A partir de la identificación de estas variables y sus conexiones, se pueden ratificar o refutar la validez de las prácticas en las que nos basamos para construir software.

En concreto, en la disciplina de la ingeniería del software, es necesario validar las teorías propuestas, identificar sobre qué escenarios funcionan satisfactoriamente, conocer sus limitaciones y descubrir posibles puntos de mejora mediante el diseño de experimentos que así lo permitan (Kish, 1959). De este modo, en la presente Tesis Doctoral nos hemos basado en el método del caso de estudio, que es uno de los tres métodos empíricos más utilizados (junto con las encuestas y los experimentos), para la ejecución de la investigación. En concreto, ha sido planificado y llevado a cabo de acuerdo a las directrices propuestas en (Runeson & Höst, 2009) y, en las que destacan las siguientes fases:

1. *Definición del caso de estudio.* — Se debe llevar a cabo a partir de una propuesta para la cual se quiere analizar su comportamiento en un contexto real, estableciendo la secuencia lógica que conecta los datos empíricos a las preguntas iniciales de investigación y, en última instancia, a sus conclusiones (Yin, 2017). En este sentido, resulta importante que los investigadores revisen la literatura relevante e incluyan proposiciones teóricas con respecto al caso de estudio antes de comenzar a realizar cualquier recolección de datos. De este modo, se hace imprescindible dominar con un elevado nivel de detalle la propuesta a validar, el contexto en el que se enmarca ésta y los aspectos que se pretenden analizar con la ejecución del caso de estudio (Baxter & Jack, 2008).

En el capítulo 2 se lleva a cabo la revisión de la literatura relevante sobre el marco tecnológico que afecta a nuestra investigación. Por otra parte, en la siguiente sección (1.4.2) se muestra la definición de nuestro caso de estudio.

2. *Protocolo de recogida de datos.* — En un caso de estudio es importante tener en cuenta los puntos de vista de diferentes roles e investigar diferencias. Dado que nuestro caso de estudio se divide en dos fases, se han definido dos protocolos de recogida de datos que pueden encontrarse en la sección 3.1 para la fase 1 (adaptación y aplicación del método Lean Kanban) y sección 4.1 para la fase 2 (aplicación de técnicas de coaching e inteligencia emocional grupal).
3. *Análisis de resultados.* — El objetivo básico del análisis es derivar conclusiones de los datos, manteniendo una clara cadena de evidencia. La cadena de evidencia significa que un lector debe poder seguir la derivación de los resultados y las conclusiones de los datos recopilados. Esto significa que el

investigador debe presentar la información suficiente de cada paso del estudio y de cada decisión tomada (Yin, 2017).

En las secciones 3.3 y 4.3 se realiza el análisis de resultados de la fase 1 y fase 2, respectivamente.

4. *Lecciones aprendidas.* — Se puede enumerar una serie de características que debe tener un informe de resultados de un caso de estudio (Robson, 2002):
 - Indicar de qué trataba el estudio.
 - Comunicar un sentido claro del caso de estudio.
 - Proporcionar una "historia de la investigación" para que el lector pueda ver lo que se hizo, por quién y cómo.
 - Proporcionar datos básicos de forma enfocada, para que el lector pueda asegurarse de que las conclusiones son razonables.
 - Articular las conclusiones del investigador y ubicarlas en el contexto al que afectan.

En las secciones 3.4 y 4.4 se presentan las lecciones aprendidas de cada fase de nuestro caso de estudio.

1.4.2 Definición del caso de estudio

El **objetivo principal** de nuestro caso de estudio es validar la hipótesis presentada en esta Tesis Doctoral: se propone la mejora de la productividad de un equipo de mantenimiento de software a través de la aplicación de un marco de referencia, el método Lean Kanban, mediante la implantación de las prácticas que se consideren más oportunas atendiendo a las características del equipo, así como la sostenibilidad de la cultura de mejora continua a través de la aplicación de técnicas de coaching e inteligencia emocional grupal.

Para ello, se debe llevar a cabo una transformación en sus métodos de trabajo de modo que el rendimiento del equipo acabe siendo mayor al finalizar el caso de estudio. Por otra parte, es necesario que una vez se haya avanzado en la transformación, el equipo participe en un proceso de coaching e inteligencia emocional grupal. Tanto los integrantes del equipo como la organización a la que prestan servicio deben dar su consentimiento explícito a la participación en el proceso.

La figura F-1.1 muestra la estructura básica del caso de estudio definido. En una **primera fase**, que tiene una duración de 9 meses, se afronta la adaptación y aplicación del **método Lean Kaban** mediante la implantación de las prácticas que se consideren más oportunas atendiendo a las características del equipo. Para ello,

se lleva a cabo una transformación en sus métodos de trabajo de modo que el rendimiento del equipo acabe siendo mayor al finalizar el periodo. Durante la ejecución del caso de estudio se irán registrando indicadores cuantitativos de rendimiento a través del análisis del trabajo realizado por el equipo, así como otras medidas cualitativas procedentes de las reuniones de retrospectivas que se llevarán a cabo.

Posteriormente se deben realizar las **actividades preparatorias para el cambio de fase** con una duración de **2 meses**. Entre las actividades a realizar se encuentran la valoración de la ejecución del caso de estudio, la confirmación con todos los interesados (equipo de mantenimiento, organización cliente, consultora TI y equipo investigador) del interés en continuar con el caso de estudio, la planificación de agendas y la asignación de recursos: **tiempo, económicos, espacios**. En este aspecto se debe garantizar, por parte de la organización cliente, la obtención de los espacios necesarios, comprometiendo salas de reuniones alejadas del centro habitual de trabajo, en las que el equipo pudiera centrarse durante las sesiones. Igualmente, se debe obtener el compromiso de garantizar el tiempo necesario para el desarrollo de las sesiones, evitando cualquier tipo de interrupción.

Nótese que este compromiso no es fácil de adquirir sin un apoyo de la dirección, dadas las labores requeridas por un equipo de mantenimiento de software, con un gran número de sistemas de información en producción que pueden requerir de intervenciones urgentes. En cuanto al aspecto económico, es importante asegurar el compromiso de garantizar la facturación de todas las horas dedicadas al proceso de coaching.

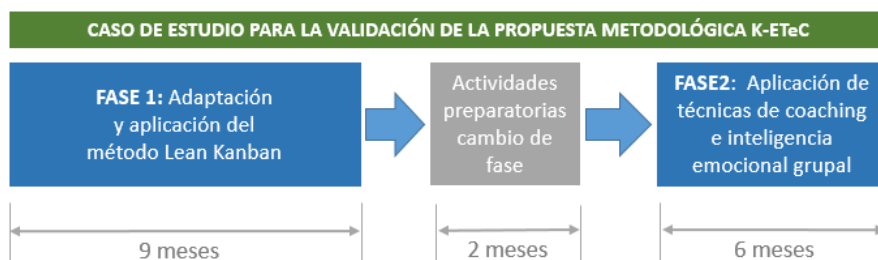


Figura F-1.1 Definición del caso de estudio.

Por último, en la segunda fase se afronta la aplicación de las **técnicas de coaching e inteligencia emocional grupal** con una duración de **6 meses**. Durante esta fase se lleva a cabo un diagnóstico del equipo de trabajo que facilite la identificación de los puntos de mejora sobre los que llevar a cabo la intervención.

Esta, ha de finalizar con la realización de un plan de acción que permita al equipo alcanzar los objetivos definidos.

El periodo de ejecución del caso de estudio completo comprende 17 meses. Se trata de un plazo de tiempo elevado, máxime teniendo en cuenta que ha de aplicarse en un entorno empresarial que dificulta el mantenimiento de equipos estables. Aun así, dada la naturaleza y complejidad de las transformaciones que debían llevarse a cabo, consideramos que es un tiempo razonable para su ejecución.

Descripción del equipo de trabajo

El equipo de trabajo cumplía las condiciones que consideramos imprescindibles para la buena ejecución del caso de estudio:

- Se trataba de un equipo de **mantenimiento del software** que prestaba sus servicios en un entorno empresarial real bajo la modalidad de contratación de **outsourcing**.
- Su marco metodológico **no incluía la metodología Lean**. Puesto que el caso de estudio se fundamenta en la transformación del equipo de mantenimiento de software, es necesario que el equipo parta de un marco metodológico tradicional para poder llevar a cabo su transformación.
- Tenía la suficiente **autonomía** para la incorporación de nuevas prácticas y técnicas organizativas. Nótese la importancia de esta característica ya que se requería que el equipo de investigación pudiera tomar decisiones que afectaran directamente en su rendimiento.
- En cuanto a la **participación** en el **proceso de coaching** se contaba con el interés de todos los miembros del equipo. Este era un aspecto crítico en la definición del caso de estudio puesto que un proceso de coaching que pretenda tener las mínimas oportunidades de éxito debe contar con el interés real del *coachee* (persona que hace y se responsabiliza del proceso de cambio para conseguir su objetivo). Es por esto que antes de la ejecución de la primera fase, el equipo fue informado del caso de estudio al completo, explicándoles que en la segunda fase se llevaría a cabo un proceso de coaching e indicándoles que la participación en el proceso era del todo voluntaria. Una encuesta previa mostró un 100% de interés en la participación del caso de estudio.

Por otra parte, queremos hacer notar la analogía entre la composición y funciones del equipo de Microsoft descrito en (Anderson & Dumitriu, 2005) que llevó a cabo la primera implementación del método: formado por un jefe de proyecto, cinco desarrolladores y un técnico de soporte que hacía las funciones de atención al usuario y resolvía parte de las incidencias.

Nuestro equipo era responsable del mantenimiento de aplicaciones internas de gestión por lo que debía compatibilizar el mantenimiento evolutivo, el mantenimiento correctivo y la resolución de incidencias del entorno de producción. En total tenía bajo su responsabilidad alrededor de 25 sistemas heterogéneos: aplicaciones web, aplicaciones cliente/servidor, servicios Web, así como una variedad de procesos de consolidación de datos que se ejecutan en distintas tecnologías: Jobs de Oracle, DTS, servicios Windows y tareas programadas.

1.5 Esquema de la Tesis

Los siguientes capítulos están organizados como se explica a continuación:

- El **capítulo 2**, proporciona una visión completa sobre los **conceptos previos y el estado del arte**. Para ello, la sección 2.1 introduce el método Lean Kanban para el desarrollo del software contextualizándolo con las principales líneas de pensamiento sobre las que se apoya. En la sección 2.2 se estudia la situación de los equipos de mantenimiento de software que trabajan bajo una modalidad de *outsourcing*. En la sección 2.3 se introduce el *coaching* e inteligencia emocional grupal. Por último, en la sección 2.4 se muestran los trabajos relacionados con el caso de estudio desarrollado en la presente Tesis Doctoral, clasificando los diferentes enfoques en la implantación del método.
- El **capítulo 3** está centrado en la presentación de las diferentes iniciativas llevadas a cabo en la primera fase de nuestro caso de estudio, poniendo especial foco en las diferentes decisiones que se tomaron durante la adaptación y aplicación del **método Lean Kanban** al equipo de servicio de mantenimiento de software. Se finaliza el capítulo con el análisis de los resultados de esta fase del caso de estudio y la exposición de las lecciones aprendidas.
- El **capítulo 4** describe la segunda fase de nuestro caso de estudio, la presentación de nuestra propuesta de proceso de **coaching e inteligencia emocional grupal** (proceso ETeC) y su aplicación al equipo de mantenimiento de software. De forma análoga al capítulo 3, se finaliza con el análisis de los resultados de esta fase del caso de estudio y las lecciones aprendidas durante su aplicación.
- El **capítulo 5** concluye resumiendo las **principales contribuciones** de esta Tesis Doctoral. Para este propósito, proporciona un análisis de los resultados. Por otra parte, plantea una serie de preguntas para **futuras investigaciones**.
- Finalmente, el **apéndice A** recoge las referencias bibliográficas y electrónicas utilizadas a lo largo de esta Tesis Doctoral y el **apéndice B** resume las siglas distribuidas en el texto.

*Conceptos Previos y
Estado del Arte*

En este segundo Capítulo se ofrece una visión global del estado del arte de las diferentes áreas relacionadas con la cuestión de investigación que se aborda en esta Tesis Doctoral. En primer lugar, la sección 2.1 presenta el contexto del método Lean Kanban, profundizando en las líneas de pensamiento que han influido en el método: Lean Product Development, la teoría de las restricciones y las metodologías ágiles. En la sección 2.2 se estudia el efecto que produce la prestación de servicios en modalidad de *outsourcing* sobre los equipos de trabajo, y en especial, en los de desarrollo de software. En la sección 2.3 se introduce el *coaching* e inteligencia emocional grupal. Por último, la sección 2.4 presenta las principales contribuciones científicas relacionadas con la aplicación del método Lean Kanban en escenarios reales.

2.1 Contexto del método Lean Kanban

El término Lean fue referenciado por primera vez a principios de la década de los noventa (Womack et al. 1990) convirtiéndose en la forma occidental de llamar al Toyota Production System (TPS), que fue posteriormente adaptado para su aplicación al desarrollo de software en el movimiento actualmente conocido como Lean Software Development.

Aunque ofrecer una única definición para referirse al *pensamiento Lean* aplicado al desarrollo del software no es una tarea sencilla, sí es cierto que podemos diferenciar dos enfoques principales a la hora de adaptar los principios Lean en este entorno: aquellos que se basan en la eliminación del desperdicio (Poppendieck & Poppendieck, 2007) y los que se apoyan en el concepto de flujo (Reinertsten, 2009). El método Lean Kanban propuesto por David J. Anderson pertenece al segundo enfoque y está basado en la filosofía *Just in Time*: producir los elementos que se necesitan, en las cantidades que se necesitan y en el momento en que se necesita (Womack & Jones, 1996). El método fue aplicado por primera vez en un equipo que trabajaba en el mantenimiento de aplicaciones internas de Microsoft (Anderson & Dumitriu, 2005).

El método Lean Kanban se compone de principios que se alcanzan a través de una serie de actividades esenciales definidas las seis prácticas (Anderson & Carmichael, 2016) que se muestran en la figura 2.1.

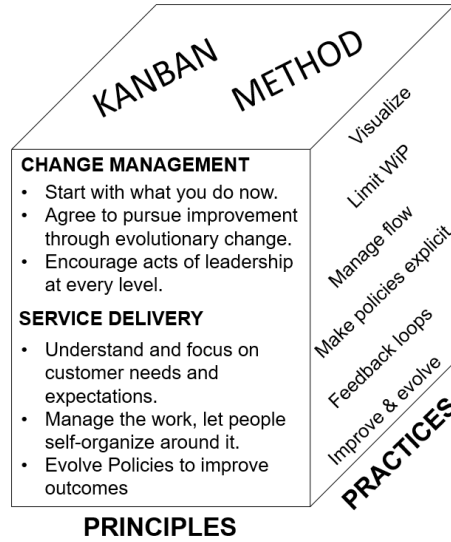


Figura F-2.1 Principios y prácticas del método Lean Kanban.

Adaptada de (Anderson, D. & Carmichael, 2015)

La definición de estos principios y prácticas provienen de las influencias de diferentes *líneas de pensamiento* entre las que destacan el Lean Product Development, Theory of Constraints (TOC) y las metodologías ágiles. Las principales aportaciones de las diferentes líneas de pensamiento sobre el método Lean Kanban se muestra en la figura F-2.2.

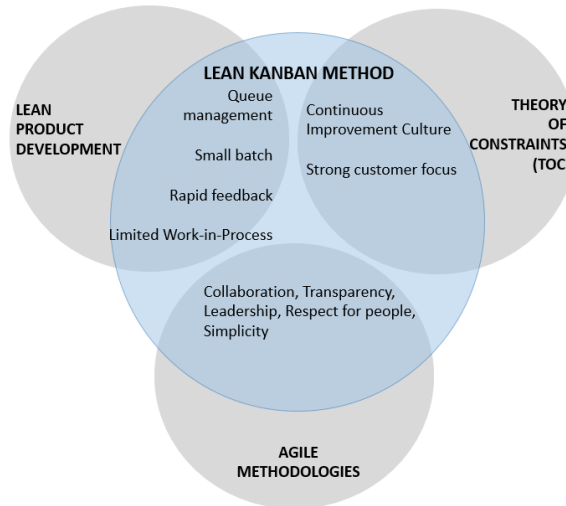


Figura F-2.2 Influencias sobre el método Lean Kanban

Por último, el método Lean Kanban promueve los siguientes valores (Burrows, 2014): transparencia, equilibrio, colaboración, enfoque al cliente, flujo, liderazgo, entendimiento, acuerdo y respeto

La **transparencia** es el primero de los valores y se ve fuertemente apoyado por la primera práctica: visualizar. De hecho, esta práctica se encuentra en la gran mayoría de las implementaciones de Kanban. Es habitual que siempre exista un tablero (físico o virtual) que permite la visualización del trabajo del equipo. Es conveniente recordar que la traducción del japonés de Kanban es “señal visual” lo que viene a denotar la importancia de esta práctica y su aportación a la transparencia.

Otra de las prácticas que mejora la transparencia del proceso de desarrollo es “hacer explícitas las políticas”. La necesidad de esta práctica es porque no todo lo que se necesita visualizar puede hacerse mediante tarjetas, colores y avatares. En ocasiones, una breve descripción de las políticas asociadas a cierta parte del sistema, pueden hacer que las personas que trabajen en él entiendan en mayor medida su funcionamiento.

El **equilibrio** es otro de los valores buscados por el método Lean Kanban y para alcanzarlo se requiere anticipación, vigilancia y esfuerzo. Es importante encontrar el equilibrio de la carga de trabajo frente a la capacidad disponible, el equilibrio de la toma de decisiones basadas en la urgencia frente a las basadas en la gestión de fechas y plazos, así como el equilibrio entre las necesidades de los diferentes interesados. La principal práctica para introducir equilibrio en el sistema es *limitar el trabajo en curso* (WIP) lo que permite la implementación de un sistema *Pull* y facilita la distribución uniforme del trabajo a través del sistema.

La **colaboración**, al igual que la transparencia y el equilibrio cataliza el cambio e incrementando la colaboración facilitaremos que el cambio llegue a ser más sostenible. La práctica *mejorar colaborativamente, evolucionar experimentalmente* ofrece sus servicios a este valor. Cuando se dispone de equipos auto-organizados, las mejoras y otras oportunidades de innovación se desarrollarán desde la colaboración que, facilitada por la propia auto-organización, surge naturalmente si se involucra a los grupos de personas apropiados. Por otra parte, el concepto de evolucionar experimentalmente trata de cómo es dirigido el cambio. En Kanban la base científica de esta experimentación reposa en el método PDCA (Deming, 1993).

El **enfoque al cliente** es otro de los valores del método Lean Kanban, potenciado a través de la práctica *Gestionar el flujo*. A primera vista puede no

comprenderse la relación existente entre la gestión del flujo y el enfoque al cliente, aunque están muy relacionados. Tiene que ver con las necesidades del cliente y la forma de poder llegar a anticiparnos a ellas. Para ello, es importante que todo el equipo sea capaz de contestar a las siguientes preguntas: ¿Qué estás entregando? ¿Para qué? ¿Por qué? Tener claras las respuestas y ser capaz de organizar el flujo de trabajo atendiendo a estas preguntas, permitirá alinear los procesos del equipo con los usuarios/clientes.

El **flujo** es otro de los valores que se alcanzan a través de la práctica *Gestionar el flujo*. Toda implantación del método Lean Kanban se debe orientar a la gestión del flujo de trabajo, manteniéndolo lo más sencillo posible, potenciando su transparencia a través de la gestión visual y dejando al descubierto los principales impedimentos que evitan que el trabajo fluya. Otra técnica utilizada para potenciar el flujo es dividir las tareas grandes en varias más pequeñas. De este modo, se evitará una foto monolítica que permanece por varios días puesto que los miembros del equipo serán capaces de avanzar trabajo varias veces en el día (al ser el tamaño de las tareas reducido), consiguiendo que exista flujo.

El **liderazgo** es el puente que permite enlazar las prácticas y los principios de los que se compone el método Lean Kanban. Ahora bien, el foco impuesto en el liderazgo no implica necesariamente que se valoren las jerarquías, de hecho, las compañías Lean acostumbran a tener estructuras eminentemente planas. El motivo de potenciar el liderazgo es la estrecha relación existente entre liderazgo y cambio: un método orientado al cambio no puede ignorar el aspecto del liderazgo. Si bien es cierto que el cambio en las organizaciones puede surgir de modo espontáneo (e incluso accidental), lo normal es que se produzca fruto de un liderazgo que permite y potencia dicho cambio.

El **entendimiento** está asociado con el primer principio fundacional: *comenzar con lo que se hace en la actualidad*. Esto no implica que se deba evitar llevar a cabo ningún cambio hasta que se finalice un *proyecto de análisis de la situación*. Unos mínimos de acuerdo deben ser suficientes para catalizar los cambios necesarios. Este acuerdo debe enfocarse en aspectos concretos: el propósito del sistema, cómo presta servicio al cliente, cómo se puede dejar a los clientes insatisfechos y a los trabajadores frustrados.

El **acuerdo** se encuentra en el segundo principio fundacional del método: *acordar perseguir el cambio evolutivo*. Esto implica el acuerdo de que el cambio es necesario y fomentarlo a través de una estrategia evolutiva. La gestión del cambio se deberá apoyar en los *agentes de cambio*, que son aquellas personas que hacen

que el cambio ocurra ya sea siguiendo algún plan de implementación del cambio o catalizando el cambio a través de la clara comunicación de la visión.

El **respeto** es el último de los valores que promueve el método y podemos encontrarlo en el tercer principio fundacional: *inicialmente, respetar los procesos, roles, responsabilidades y títulos de trabajo actuales*. La promesa de unas mejores condiciones en el futuro, no debe repercutir a que en el corto plazo las personas se vean afectadas por decisiones que no les respetan. Detrás de este principio subyace uno de los pilares del Toyota Way: *respect for people* (Liker, 2004).

Tal como hemos dicho, la definición de los principios y prácticas descritos provienen de las influencias de diferentes líneas de pensamiento entre las que destacan el Lean Product Development, la Teoría de las Restricciones (TOC) y las metodologías ágiles. A continuación, se analizan cada una de ellas, así como su relación con el método Lean Kanban.

2.1.1 *Lean Product Development*

El concepto de Lean Product Development, que tiene una gran influencia sobre el método Lean Kanban, es considerablemente más reciente que el Lean Manufacturing sobre el que se basa (Womack & Jones, 1996). A continuación, se describen algunos de los pensadores más significativos dentro del Lean Product Development, así como su línea de pensamiento:

- Donald G. Reinertsen, que se centra en los primeros principios Lean como la **teoría de colas y economía de costes**, permitiendo que las ideas y técnicas, tanto de Lean Manufacturing como de Lean Product Development, sean aplicados fuera de su dominio original de forma más eficiente (Reinertsen, 2009). De hecho, el método Lean Kanban como se conoce en la actualidad no existiría si no es por el interés personal de Reinertsen en la primera implantación (aplicando TOC) de David J. Anderson (Anderson & Dumitriu, 2005).
- Michael N. Kennedy, por su parte, pone su foco en la diferencia entre cómo Toyota gestiona el desarrollo de productos (Toyota Product Development) y la fabricación (Toyota Production System) mediante **set-based learning**, en lugar del flujo de una sola pieza (Kennedy, 2003). Las limitaciones físicas presentes en los procesos de fabricación no tienen reflejo en el desarrollo del software, lo que facilita el principio de **retrasar la decisión** permitiendo explorar la solución desde diferentes ángulos, lo cual ofrece flexibilidad a un coste muy bajo.

- Peter Middleton y James Sutton se centran en el **compromiso con las necesidades del cliente**, asegurando que un entendimiento profundo de las necesidades de los clientes posibilita, no solamente unas relaciones más fructíferas, sino también unos productos (y arquitecturas) más duraderos (Middleton & Sutton, 2005).
- Mary y Tom Poppendieck (Poppendieck & Poppendieck, 2007) parten de los principios Lean, los **desperdicios** y otros conceptos del Lean Manufacturing y los adaptan a las particularidades del desarrollo del software.

La evolución futura del Lean Product Development se apoyará en las siguientes líneas base (Schmidt & Rosenberg, 2014):

1. Prácticas extraídas de Toyota y otras compañías líderes como Google, que implementadas fuera de estas compañías puedan ser adaptadas con la suficiente confiabilidad.
2. Prácticas desarrolladas y encontradas para trabajar en contextos particulares, y que con el tiempo se puedan difundir más ampliamente.
3. Impulso del respaldo teórico, parte extraído de las prácticas anteriores, parte importado de otros campos y el resto derivado de primeros principios y probados en el campo del desarrollo del software.

2.1.2 *Theory of Constraints (TOC)*

La Teoría de las Restricciones fue enunciada por Eli Goldratt, comenzando por su primera publicación de la década de los 80, *The Goal* (Goldratt & Cox, 1984), en la que se centra en la aplicación de TOC en el entorno industrial. Posteriormente expande sus ideas al marketing y la distribución con *It's not luck* (Goldratt, 1994) y cierra la serie en la publicación *Critical Chain* (Goldratt, 1997) en la que aplican sus ideas al campo de la dirección de proyectos.

TOC es un marco de trabajo conceptual que pretende ayudar tanto a individuos como a organizaciones a preparar y enfocar los **cambios** desde una perspectiva de **mejora continua**. El objetivo de TOC es ayudar a despejar las siguientes variables: qué cambiar, para qué cambiar y cómo causar el cambio.

Las teorías de Goldratt han alcanzado la suficiente madurez como para ser incorporadas en la guía PMBoK® del Project Management Institute (Rose, 2013), actualmente la base de conocimiento más importante para la dirección de proyectos, que incorpora CCPM (uno de los conceptos de TOC) como una técnica del proceso del desarrollo del cronograma.

La esencia de TOC se basa en los 5 pasos secuenciales que se enumeran a continuación:

1. Identificar la restricción (el cuello de botella) del sistema.
2. Decidir cómo explotarlo.
3. Subordinar todo a la decisión anterior.
4. Superar la restricción del sistema (aumentar su capacidad).
5. Si en los pasos anteriores se ha roto una restricción, regresar al primer paso, pero **no permitir la inercia** (para facilitar la mejora continua).

A diferencia de otros métodos de gestión de proyectos, que se focalizan sobre todo en ciertos aspectos técnicos, la aplicación de TOC se centra en cómo los directivos gestionan el comportamiento humano, en términos de construir la red del proyecto y en cómo se gestiona posteriormente. Goldratt argumenta que la principal razón para el retraso de los proyectos se debe al abuso de los *tiempos de seguridad* creados en cada actividad (Rand, 2000). Este es un factor humano, en lugar de técnico y la solución del mismo requiere el cambio de comportamiento de las personas que planifican y estiman sus proyectos.

A continuación, se resumen algunos de los aspectos importantes de la aplicación de TOC al desarrollo del software:

- La importancia de las **métricas**. Goldratt habitualmente refleja la necesidad de tomar medidas a lo largo de toda la cadena productiva, tanto en tiempos de procesado como en calidad del producto; sin una referencia clara del punto actual es muy difícil decidir si aplicando determinadas prácticas se está avanzando al objetivo marcado. En los proyectos de desarrollo del software es habitual encontrar una falta de medición del rendimiento del proceso incluso de la calidad del producto. Ahora bien, cuando se plantea la incorporación de métricas, dada la facilidad para generar/manejar datos, es habitual equivocarse en exceso. De este modo, sólo se debería registrar una métrica después de haberse planteado por qué tiene importancia en el contexto del proyecto. A continuación, se enumeran algunas métricas interesantes en entornos de desarrollo del software: el tiempo de entrega, el porcentaje de errores, el acoplamiento de clases, la complejidad ciclomática del código, la profundidad de herencia o la adecuación a normas de programación (análisis estático de código).
- "El **sentido común** es el menos común de los sentidos". Goldratt reflexiona habitualmente en que la mayor parte de los planes de mejora se basan en tomar

decisiones de "sentido común" nada excepcionales a simple vista. Por otra parte, como señala Anderson, no se necesita de mentes prodigiosas ni de disponer del mejor personal técnico del mundo (algo importante en cualquier caso al tratarse de trabajadores del conocimiento). Por el contrario, la mayor parte de las mejoras introducidas son fruto de aplicar medidas de sentido común que habitualmente no se aprecian porque chocan con la cultura preestablecida. De este modo, además del sentido común, convendría añadir valentía para cambiar la forma habitual de realizar el trabajo (Anderson, 2010).

- Respecto a la **innovación y el rechazo al cambio**, Goldratt muestra la dificultad que presentan determinadas organizaciones para introducir un cambio de pensamiento incluso en situaciones críticas. Esta dificultad puede ser capaz de llevar a la quiebra a una organización. El hecho de *no tener nada que perder* es lo que en muchas ocasiones impulsa el cambio. De existir alguna posibilidad de salvar la situación con las normas preestablecidas, en muchas ocasiones no se tomarían las medidas que mejoran el rendimiento de equipos y organizaciones. Esta idea es el eje central de *El dilema del Innovador* (Christensen, 2000), donde se pueden encontrar documentados casos concretos de agentes dominantes de un mercado que acaban desapareciendo ante su incapacidad de adaptación a determinadas innovaciones. Se detallan dos casos en sectores muy diferentes: las maquinarias de perforación con la aparición de los cilindros hidráulicos en sustitución de las poleas y las empresas tecnológicas de fabricación de dispositivos de almacenamiento (discos duros) con su necesidad continua de duplicación de capacidad.
- En un entorno de producción industrial la **reducción del tamaño** de los lotes repercute en una reducción del inventario en los almacenes intermedios, disminuyendo, por tanto, el tiempo perdido en estos almacenes a la espera de capacidad productiva. Esto deriva en una condensación del tiempo transcurrido desde la aceptación del pedido hasta la entrega. Lo anterior tiene un punto débil: implica invertir más tiempo en configuración de la maquinaria. Ahora bien, si la maquinaria afectada no son cuellos de botella del proceso, el rendimiento global no se verá penalizado. Adaptado al entorno del desarrollo del software, podemos decir que cuanto menores sean nuestros trabajos (tareas de ingeniería) menor será el tiempo de proceso y de espera a que alguien pueda hacerse cargo de ellos, por lo que el tiempo de proceso disminuye. Nótese la relación de este concepto con uno de los valores del método Lean Kanban enunciados al comienzo de esta sección: el **flujo**.

- En líneas generales, los trabajos de desarrollo del software **consumen tiempo** en alguna de las siguientes actividades:
 - Ingeniería: análisis, diseño, construcción, pruebas, etc.
 - Configuración del entorno.
 - Tiempo de espera por falta de capacidad productiva en una determinada actividad de ingeniería.
 - Tiempo de espera debido dependencias con otros desarrollos relacionados (en el caso de trabajos más grandes).

Si los trabajos son muy grandes desperdiciaremos tiempo principalmente en los dos últimos puntos. El problema es que disminuyendo el tamaño aumentamos el tiempo de configuración (puesto que tendremos que preparar el entorno más veces). En el mundo industrial este tiempo se pierde configurando la maquinaria para que se comience a trabajar con una pieza determinada. En desarrollo de software se *pierde* en la configuración del entorno de pruebas y en el despliegue de los sistemas, lo que lleva a la necesidad de automatizar las pruebas y el proceso de despliegue.

- Un último aspecto extrapolable desde el entorno industrial al desarrollo del software es el **tiempo ocioso**. Goldratt señala la conveniencia de detener una infraestructura productiva (y sus operarios) en determinadas circunstancias en beneficio de la productividad general. El sentido común y las normas básicas de contabilidad apuntan a que, para acelerar su amortización, la máquina debe trabajar al 100% de su capacidad. Pero en realidad, esto genera una optimización local que no redundará en la optimización global puesto que produce un inventario innecesario de productos semielaborados (aumentando con ello el tiempo de ciclo y el coste del inventario). Goldratt concluye que detener maquinaria y personal bajo determinadas circunstancias puede resultar rentable.

Un ejemplo de adaptación de este concepto en el entorno del software, en los que la utilización de estos tiempos ociosos puede resultar rentable es el modelo Spotify (Kniberg & Ivarsson, 2012). En este modelo todo el personal dispone de un 10% de su tiempo para dedicarse a actividades que consideran interesantes, pero no directamente asociadas a compromisos de sus proyectos. Generalmente suelen agrupar ese 10% y cada dos semanas disponen de un *hack-day* que pueden dedicar a investigar cualquier tecnología que les resulte de interés y que pueda llegar a ser aplicado en su trabajo en un futuro. Si bien en TOC el tiempo ocioso busca eliminar inventario intermedio, en

Spotify se busca generar innovación. Aun así, en ambos entornos resulta rentable que los recursos no trabajen continuamente al límite de su capacidad evitando optimizaciones locales que no optimizan el proceso completo. En esta línea, Anderson aboga por aprovechar los bloqueos producidos por los *WIP* para generar estos espacios de innovación.

2.1.3 Metodologías ágiles

El manifiesto para el desarrollo ágil de software (Beck et al., 2001) firmado en 2001 por 17 personas de reconocido prestigio internacional ha influenciado enormemente en el desarrollo del software del siglo XXI.

Después de casi dos décadas de la firma del citado manifiesto se pueden seguir encontrando diferentes definiciones del concepto de agilidad: capacidad de crear y responder, con rapidez y flexibilidad, a los cambios que provienen de un, cada vez más dinámico, entorno de negocios (Dingsøyr et al., 2012). Desde un punto de vista más formal la podríamos definir como la capacidad del equipo de software para responder de manera eficiente y eficaz e incorporar cambios a los requerimientos del usuario durante el ciclo de vida del proyecto (Lee & Xia, 2010).

Desde la firma del manuscrito la comunidad académica ha profundizado en los principales principios del manifiesto, los individuos, las interacciones, el *software funcionando*, así como el papel de la documentación en el proceso de desarrollo, pero en menor medida sobre el preámbulo del manifiesto: *uncovering better ways of developing software*. El método Lean Kanban es, de hecho, la aplicación de dicho preámbulo del manifiesto, al encontrarse las diferentes metodologías en continua evolución.

Tanto los métodos ágiles como el método Lean Kanban comparten su enfoque sobre la planificación adaptativa y tienen una aproximación muy centrada en las personas. Los valores y principios ágiles y los del método Lean Kanban son tan parecidos que en ocasiones cuesta diferenciarlos: colaboración, liderazgo, respeto por las personas, transparencia y simplicidad son, entre otros, valores reconocidos por una o varias de las metodologías ágiles principales como FDD, XP y Scrum.

Así pues, parece adecuado afirmar que el método Lean Kanban puede trabajar perfectamente en conjunción con otros métodos ágiles, de hecho, en el caso de Scrum se llega a plantear una fusión de ambas metodologías bajo el nombre de Scrumban (Nikitina et al., 2012).

2.2 *Outsourcing* de mantenimiento de software

El *outsourcing* se define con carácter general como "el uso de agentes externos para realizar una o más actividades organizacionales". Esta definición infiere en la posible subcontratación de cualquier función organizacional (Brooks, Nita, 2006). En investigaciones relacionadas específicamente con la externalización de TI, encontramos una definición más específica como "la contribución significativa de proveedores externos en los recursos físicos y humanos asociados con los componentes completos o específicos de la infraestructura de TI en la organización" (Loh & Venkatraman, 1991).

A principios de la década de los 90, muchas compañías de desarrollo de software optaron por esta modalidad de contratación. Los enfoques varían desde la subcontratación de partes de proyectos de desarrollo de software, hasta el establecimiento de equipos de desarrollo verdaderamente virtuales (Kobitzsch et al., 2001).

Lo cierto es que, el *outsourcing* se acabó convirtiendo en una *bala de plata* que a menudo se espera resuelva una multitud de problemas. Muchas decisiones ejecutivas vienen motivadas porque los costes crecen demasiado rápido, la tecnología cambia con demasiada frecuencia o los clientes no están siendo atendidos adecuadamente por las capacidades técnicas de sus empresas (Benko, 1992).

En este contexto, el ***outsourcing de mantenimiento de software*** se está convirtiendo en una alternativa popular en la industria. Las empresas de productos de software están adoptando la subcontratación en prácticamente todas las fases del ciclo de vida del software, considerándola como una opción estratégica a largo plazo. Las funciones de mantenimiento subcontratado pueden incluir todos los tipos clásicos de mantenimientos (es decir, preventivo, correctivo, adaptativo y perfectivo). En concreto, mantenimiento de código, corrección de errores, mejora de funciones, control de versiones, actualización de software, optimización de software, pruebas, auditoría, portabilidad a diferentes plataformas de hardware/software, localización, mejora de la interfaz gráfica del usuario, etc. (Ahmed, 2006).

La estrategia de *outsourcing* puede ser diferente atendiendo al nivel de internalización de los recursos técnicos y humanos incorporados en la organización contratante. Podemos enumerar como los tipos de *outsourcing* más populares: la subcontratación completa, la subcontratación de gestión de instalaciones y la subcontratación de integración de sistemas (Loh & Venkatraman, 1991).

Todas las modalidades anteriores están sujetas a las ventajas e inconvenientes del modelo de subcontratación. Las **principales ventajas** ofrecidas por el modelo son las siguientes (Antonucci et al., 1998):

- *Ahorro de costes.* La dura competencias de muchos mercados hace que las empresas reestructuren sus plantillas con el objetivo de ahorrar costes. La empresa subcontratada ofrece este ahorro de costes apoyándose en las siguientes ventajas competitivas:
 - o Dispone de estructuras organizativas más delgadas y de un control más estricto de los beneficios adicionales.
 - o Utiliza personal de bajo costo de manera más agresiva. Además, con la ayuda de las telecomunicaciones, puede trasladar los centros de datos a áreas de bajo coste.
 - o Utiliza acuerdos de compra de hardware y licenciamiento de software más efectivos.
- *Acceso a personal cualificado en la última tecnología.* La volatilidad de las tecnologías de la información hace que los perfiles se queden rápidamente obsoletos. Esta velocidad provoca que el retorno de la inversión en formación del personal interno no sea el adecuado. El personal subcontratado, por el contrario, es seleccionado con las habilidades y competencias requeridas en cada momento.
- *Flexibilidad.* Las empresas deben ser lo suficientemente flexibles para sobrevivir en un entorno empresarial en continuo cambio, por lo que sus departamentos de TI deben responder lo suficientemente rápido a los cambios demandados. A través del *outsourcing* pueden acceder rápidamente a una amplia gama de recursos, competencias y capacidades difícilmente accesible, en el corto plazo, a través de medios propios.
- *Seguridad laboral para empleados internos.* La subcontratación de empleados parte habitualmente de la premisa de una relación contractual corta. Es factible el aumento o disminución del número de empleados subcontratados sin llegar a tener una mala reputación como empleador estable (facilitando la captación de talento en las áreas de actividad principal). Además, el hecho de disponer de un número de empleados subcontratados, permite a los empleados internos disponer de una mayor seguridad en sus puestos de trabajo ante fluctuaciones de mercado.

A su vez, el modelo de *outsourcing* no está libre de ciertos **riesgos y desventajas** (Antonucci et al., 1998):

- *Exageración de los beneficios de la subcontratación.* Es habitual que los directivos que han optado por el modelo de *outsourcing* informen únicamente de situaciones de éxito. Las noticias emitidas suelen enfocarse a los ahorros proyectados, mientras que las tarifas exorbitantes por enmiendas a los contratos no llegan a hacerse públicas ya que pocas empresas desean publicitar problemas de gestión.
- *El área de TI no es sencilla de subcontratar.* Debido a que la tecnología se infiltra por toda la organización, puede no ser como otras áreas que una empresa subcontratara con éxito en el pasado.
- *Las tecnologías de la información evolucionan rápidamente.* Esto implica que hacer predicciones a más de 3 años puede ser altamente especulativo lo que dificulta la firma de contratos de *outsourcing* a largo plazo.
- *Pérdida de control.* Los detractores argumentan que un proveedor externo no ofrece la misma responsabilidad y nivel de servicio que un área de TI interna. A su vez, existen susceptibilidades en aspectos de confidencialidad de los datos, las aplicaciones estratégicas o el aprovisionamiento de mecanismos de recuperación ante desastres.
- *Malo para la moral de los empleados.* El modelo de *outsourcing* a menudo resulta en despidos. Tal situación puede perjudicar la moral del personal interno y hacer peligrar la retención de talento ante una falta de seguridad laboral.

Hasta la fecha, se han realizado numerosos estudios que examinan el impacto del *outsourcing* de TI dentro y entre las organizaciones. Los investigadores han centrado principalmente sus esfuerzos en comprender por qué las organizaciones subcontratan y las motivaciones detrás de los acuerdos de subcontratación (Lacity & Hirschheim, 1995), las relaciones entre empresas clientes y proveedores (Kim & Chung, 2003) o cómo se puede utilizar el modelo de aceptación de la tecnología para examinar las intenciones de subcontratar (Benamati & Rajkumar, 2002).

Considerando la necesidad de las organizaciones de **contratar y retener** profesionales de TI con **talento**, sorprende la falta de atención al profesional de TI dentro de esta línea de investigación. En (Brooks et al., 2009) se analiza el resultado de una encuesta a más de 450 profesionales de TI de varias organizaciones que muestra las diferentes percepciones de cómo el *outsourcing* impacta a estos profesionales. El estudio concluye que las percepciones del profesional de TI sobre el impacto del *outsourcing* tienen implicaciones en la satisfacción y en la intención de rotación. Los resultados indican que, tanto personal como profesionalmente, se percibe que el *outsourcing* tiene un **impacto negativo**. Esta conclusión proporciona

una evidencia de que se deberían mejorar los esfuerzos de reclutamiento y retención para atenuar estos sentimientos. El hecho de no abordar los problemas detectados, puede provocar una caída en la satisfacción de los profesionales y, en última instancia, un descenso en su productividad.

La mayor parte de la literatura existente sobre *outsourcing* desde los años 90 se centra en el desplazamiento de empleados de TI al proveedor, principalmente a subcontratistas fuera del país del cliente. Por el contrario, nuestra investigación se refiere a una relación de servicios gestionados en la que la organización cliente (administración pública) contrata los servicios de un equipo completo de la empresa consultora para trabajar en sus propias instalaciones. Esta es una modalidad de relación de *bodyshoping* muy frecuente en España. El libro de Zalewski es una de las mejores fuentes que encontramos para reflejar con mayor precisión los sentimientos de los profesionales de TI en escenarios similares a los del equipo de nuestra investigación. En su libro, Zalewski analiza las experiencias de los profesionales de TI y RRHH como ilustran los siguientes extractos (Zalewski, 2018):

- “(...) la separación **limita** las oportunidades para construir conexiones sociales que llevarían a **promociones laborales internas**”.
- “Debido a que sus trabajos implicaban optimizar los servicios basados en el modelo del proveedor para los procesos de TI o de recursos humanos y brindar a los clientes el producto que estaban buscando, estos profesionales se **sienten exprimidos** tanto por su empresa como por la organización cliente”.
- “A menudo tenían un apoyo **limitado para resolver conflictos** entre las expectativas y la realidad de la organización cliente y la consultora de TI”.
- “Para la mayoría, el empleo continuo significaba hacer lo mismo en diferentes organizaciones clientes, en lugar de la **oportunidad de aprender** diversos aspectos dentro de su campo”.
- “A menudo, se atrofia su oportunidad de **crecimiento profesional**”.

Asimismo, los siguientes extractos de (Brooks, Nita, 2006) giran en torno a algunos problemas relacionados con el *outsourcing* de equipos:

- “El **compromiso** es mayor entre los empleados que creen que se les trata como recursos para desarrollar en lugar de productos para comprar y vender”.
- “El *outsourcing* también representa una **amenaza** para las oportunidades percibidas de un individuo para el **progreso** y la **consecución de objetivos**”.
- “A medida que las organizaciones aumentan su participación en *outsourcing* al participar en tipos que son más integrales, el empleado se enfrenta a un aumento de los niveles de **inseguridad laboral**, así como a una **disminución** de los niveles de **motivación y satisfacción**”.

- *“También se espera que el impacto en el individuo sea mucho mayor. Una vez más, a medida que las funciones se vuelven más inclusivas de todas las actividades de TI, la probabilidad de mantener altos niveles de **motivación, satisfacción y compromiso disminuye.**”*

Por otro lado, los expertos industriales encuestados en (Ali et al., 2019) coincidieron en que la brecha de comunicación y la mala coordinación cliente-proveedor, así como el riesgo relacional y la mala gestión de las relaciones, son barreras críticas para el éxito de las relaciones de *outsourcing*.

En (Delen et al., 2016) se revisaron 30 acuerdos de *outsourcing* para encontrar tres factores controlables estadísticamente significativos que discriminaban entre el fracaso y el éxito en una etapa temprana. Sorprendentemente, uno de ellos fue la comunicación dentro de consultora de TI (no entre el cliente y el proveedor). Esta observación coincide con el escenario de nuestra investigación, donde los integrantes del equipo tienen poco o ningún contacto con su empresa.

Los ejecutivos de TI entrevistados en (Corbett, 1994) indicaron que la **comunicación** juega un papel clave en el *outsourcing* e identificaron la gestión de las **relaciones externas** como la segunda parte más importante de su trabajo después de que se lleva a cabo el *outsourcing*. Igualmente identificaron “un aumento en la importancia de lo que podría llamarse gestión general o habilidades no basadas en la tecnología” entre las habilidades necesarias para enfrentar el impacto del *outsourcing*.

Cabe recordar que el equipo de **nuestra investigación** sirve para ilustrar una situación común en los departamentos de TI de la administración pública española: equipos subcontratados de larga duración cuyos miembros no se sienten pertenecientes ni a la consultora de TI ni (obviamente) a la administración pública que actúa como contratista (Zalewski 2018).

2.3 Coaching e inteligencia emocional grupal

La mejora de la productividad es uno de los principales retos a los que tradicionalmente se han enfrentado los diferentes sectores productivos de la economía. En la época industrial, la productividad se centraba principalmente en la optimización de procesos y en los beneficios obtenidos de las economías de escala (la fabricación en grandes lotes). Sin embargo, la sociedad del conocimiento y la servitización, requieren la modificación de los factores con los que se mide la productividad. Fue en la segunda mitad del siglo XX, cuando se definió por primera vez el término “trabajador del conocimiento” (Drucker, 1999) y fue entonces cuando se planteó la necesidad de abordar la mejora de la productividad en base a la eficiencia de los trabajadores; la industria del software es un claro ejemplo de productividad ligada a la eficiencia de los equipos de trabajo. En este ámbito, el

coaching de equipos y la inteligencia emocional grupal juegan un importante papel en la mejora de la productividad de los equipos.

El coaching se refiere al proceso de acompañamiento a una persona o equipo hacia sus objetivos y metas. La primera aplicación del coaching fue en el ámbito deportivo y, a la vista de los buenos resultados, su uso se fue extendiendo a otros ámbitos, como el empresarial, convirtiéndose en una herramienta básica en los entornos ejecutivos de grandes corporaciones. Actualmente se ha generalizado su aplicación a mandos intermedios y a equipos de trabajo. Existen diversas escuelas de coaching; la aproximación utilizada durante el desarrollo de la presente Tesis Doctoral, se sustenta principalmente en el coaching ontológico (Echeverría, 2017) y en el coaching de equipos (Cardón, 2007).

Hasta la aparición del primer libro de Goleman sobre Inteligencia Emocional en el año 1995, se consideraba que los “trabajadores estrella” eran aquellos con una mayor formación y un mayor cociente intelectual. Sin embargo, Goleman puso de manifiesto (Goleman, 1998) que lo que realmente marcaba a los trabajadores estrella eran otros factores, como las habilidades sociales, la empatía o las habilidades de comunicación, todos ellos, factores en el ámbito de la inteligencia emocional. La aceptación de sus ideas fue generalizada provocando que, en las últimas décadas, la mayoría de las empresas han incorporado este tipo de habilidades en sus programas formativos.

Hoy, es posible afirmar que también existe una **inteligencia emocional** colectiva que afecta a la **productividad** de los equipos. Basándonos en Druskat y Wolf, “definimos la inteligencia emocional grupal como la capacidad de un grupo para generar un conjunto compartido de normas que gestionan el proceso emocional de una manera que genera confianza, identidad grupal y eficacia grupal.” (Druskat & Wolff, 2001). En (Elfenbein et al., 2006), Elfenbein argumenta que “los equipos con mayor inteligencia emocional promedio tienen un mejor desempeño de equipo que aquellos grupos con menor inteligencia emocional”. Además, en un equipo, “la capacidad de comprender las expresiones emocionales de los demás explicaba el 40% de la variación en el rendimiento del equipo”. Elfenbein describe la *Team EI* como la inteligencia emocional que se exhibe cuando los miembros del equipo interactúan entre sí.

Estudio tras estudio ha demostrado que los equipos son más creativos y productivos cuando pueden lograr altos niveles de participación, cooperación y colaboración entre los miembros. Tres condiciones son esenciales: confianza entre los miembros, sentido de identidad grupal -orgullo en el grupo, sentido de eficacia grupal- y la creencia de que son más efectivos trabajando juntos que separados (Druskat & Wolff, 2001).

2.4 Trabajos relacionados

No existe una guía estándar de aplicación de los principios Lean Kanban en el desarrollo del software, aunque los principales estudios realizados de casos de aplicación práctica, sí suelen compartir ciertos aspectos en la aplicación del método. En la presente sección se comparan varios artículos basados en la aplicación de iniciativas Lean Kanban y se realiza una comparativa con el enfoque utilizado en nuestro caso de estudio.

En (Ikonen et al., 2011), los autores centran la aplicación de la iniciativa en aspectos relacionados con el trabajo llevado a cabo habitualmente en los proyectos de desarrollo de software: la cantidad de documentación, los mecanismos de resolución de problemas, la gestión visual, la comprensión del todo (principio Lean), la adaptación intuitiva del método, la auto-asignación del trabajo, el proceso de “aprobación” (cadena de mando) y el proceso de comunicación. En cuanto a los resultados obtenidos, los autores muestran una mejora en la motivación del equipo y el control sobre las actividades de desarrollo. A su vez, muestra las limitaciones que la gestión visual aislada tiene sobre los resultados si no es para dar soporte a prácticas adicionales.

Por su parte, en (Middleton & Joyce, 2011), los autores se centran en la mejora de la métrica del Lead-Time, el tiempo de desarrollo, en la reducción de los defectos encontrados por los usuarios, el número de entregas por mes y el número de propuestas de mejora sugeridas por los equipos mensualmente. La implantación de Lean se basa en la aplicación conjunta de varios métodos, entre ellos, la gestión visual, la resolución de problemas, la reducción del tamaño de los lotes y el control estadístico del proceso.

En (Rutherford et al., 2010), los autores muestran la transición de dos equipos que trabajaban sin ningún tipo de planificación a equipos Scrum que posteriormente se convierten en un equipo Lean. En dicha transformación se centraron inicialmente en la mejora de las técnicas de ingeniería y organización del equipo, la mejora del flujo de valor que, como apuntan los autores, tiene como efecto colateral una mayor atención a la calidad. Otro factor de importancia en el caso de estudio es el efecto positivo en el enfoque a la resolución de problemas en equipo.

Uno de los aspectos diferenciadores de nuestra propuesta metodológica es la aplicación de técnicas de coaching e inteligencia emocional grupal en equipos de mantenimiento de software. Si bien es cierto que existen numerosos estudios sobre los factores que hacen que un equipo sea exitoso desde una gran variedad de campos, el desarrollo de software sigue siendo un proceso poco conocido dentro del

campo del comportamiento organizacional. En particular, es difícil encontrar, en la bibliografía existente, estudios sobre la influencia de la inteligencia emocional de los equipos en los resultados de los proyectos software.

En (Hiroi et al., 2005) se propone un método para evaluar la inteligencia emocional de los miembros del proyecto mediante un sistema para recopilar datos sobre competencias e inteligencia emocional.

En (Günsel & Açikgöz, 2013) los autores analizan el efecto de la inteligencia emocional en los equipos de desarrollo de software sobre los resultados del proyecto. En este caso, los resultados del análisis apoyan la relación entre el reconocimiento emocional y la velocidad de comercialización, así como la funcionalidad del nuevo producto de software. Por tanto, se concluye que el entendimiento mutuo y la empatía en los equipos de desarrollo de software resultan en acortar la duración de los proyectos y el desarrollo de productos de software que son más funcionales.

Si bien es cierto que ambos estudios afrontan la influencia de la inteligencia emocional en los resultados del desarrollo del software, ninguno de ellos realiza propuestas para mejorar la situación del equipo a partir de la evaluación obtenida.

En nuestra investigación nos hemos centrado en ocho aspectos que resultaba importante tener en cuenta en una implantación del método Lean Kanban como son: la gestión visual, el análisis y posterior reducción de las fuentes de variabilidad, la identificación de las clases de servicio y sus acuerdos de nivel de servicio (ANS) asociados, la prioridad de analizar y optimizar las principales métricas Lean, la gestión del flujo de trabajo, la importancia de utilizar mecanismos para hacer las políticas explícitas, la implantación de bucles de retroalimentación y el establecimiento de una cultura que facilite la mejora continua.

	[1]	[2]	[3]	[4]	[5]	Nuestra propuesta
Gestión visual	X	X	X			X
Gestión del flujo		X	X			X
Bucles de retroalimentación	X		X			X
Métricas Lean del proceso		X				X
Políticas explícitas		X				X
Mejorar colaborativamente		X				X
Fuentes de Variabilidad						X
Clases de servicio						X
Inteligencia emocional				X	X	X
[1] Ikonen et al. [2] Middleton y Joyce [3] Rutherford, K. et al. [4] Günsel y Açikgöz [5] Hiroi et al.						

Tabla T-2.1. Comparación de los trabajos analizados

Los diferentes estudios analizados comparten muchos de los aspectos aplicados en nuestro caso. Por ejemplo, la gestión visual denota una gran importancia en la aplicación del método como demuestra que los tres estudios analizados la tenían en cuenta. A su vez, la gestión del flujo de trabajo y el establecimiento de bucles de retroalimentación son tratados en dos de los tres estudios y la definición, análisis y estudio de métricas Lean, la definición explícita de las políticas, así como la mejora colaborativa del proceso fue tratado en otro de los estudios.

Como se puede observar en el resumen presentado en la tabla T-2.1, entre nuestro caso de estudio y el resto de trabajos analizados, existe una diferencia en el enfoque presentado en estos trabajos y el nuestro. El análisis y reducción de las **fuentes de variabilidad** y la clasificación de las **clases de servicio** no se encuentran reflejados en ningún trabajo excepto en el nuestro.

Estos dos aspectos mencionados tienen una gran importancia en la implantación del método Lean Kanban. Consideramos que la comprensión de la variabilidad y cómo impacta en el rendimiento de un proceso son herramientas esenciales de cualquier gestor, por lo que la aplicación de las acciones oportunas basadas en el estudio de la variabilidad debe ser uno de los elementos centrales de cualquier programa de mejora continua. Por otra parte, la definición de las diferentes clases de servicios permite el establecimiento de diferentes acuerdos de niveles de servicios. Gracias a esto podremos obtener una ventaja tanto en la agilidad aportada al negocio como a la propia gestión del riesgo de los proyectos, ofreciendo una mayor flexibilidad.

Otro aspecto diferenciador de nuestra investigación, es el hecho de centrarnos en el estudio de equipos que trabajan bajo una modalidad de *outsourcing*, dado que las circunstancias ambientales de estos equipos influyen sensiblemente en su estabilidad **emocional**.

Para concluir, hemos de indicar que **no hemos encontrado estudios** que agrupen la **visión global** de la investigación de la presente Tesis Doctoral, combinando la mejora gradual de un equipo con el mantenimiento de su cultura de mejora continua a través de la aplicación de técnicas de coaching e inteligencia emocional grupal.

Aplicación del método Lean
Kanban

En el presente capítulo se explican los pasos llevados a cabo para la adaptación y aplicación del método Lean Kanban durante nuestro caso de estudio, analizando las propuestas de cambio realizadas durante todo el proceso. Comenzamos estableciendo el **protocolo de recogida de datos** (sección 3.1). Posteriormente, se explica la **ejecución del caso de estudio** (sección 3.2) en la que se analiza la secuencia cronológica del proceso llevado a cabo. Para mostrar con mayor claridad la ejecución, esta sección se estructura como sigue: en la sección 3.2.1 se repasan las primeras decisiones necesarias para la puesta en marcha de la iniciativa, en la sección 3.2.2 se explican las métricas que facilitaron la medición del rendimiento del equipo de trabajo, en la sección 3.2.3 se analizan las fuentes de variabilidad que pueden disminuir el rendimiento y las medidas oportunas para su reducción, en la sección 3.2.4 se indican las acciones llevadas a cabo para la mejora de la visualización del trabajo del equipo, en la sección 3.2.5 se explican las medidas tomadas para la gestión del flujo de trabajo y en la sección 3.2.6 se incide en la implementación de bucles de retroalimentación que permitieran la mejora continua del proceso de desarrollo. Una vez finalizada la ejecución del caso de estudio se analizan los **resultados obtenidos** (sección 3.3) y, por último, se presentan las **lecciones aprendidas** (sección 3.4).

El proceso de mejora que se implantó para la transformación del equipo de desarrollo siguió el ciclo PDSA de W. Edwards Deming definido en (Deming, 1993) la última evolución de su clásico ciclo de aprendizaje y mejora, tal como se muestra en la figura F-3.1.

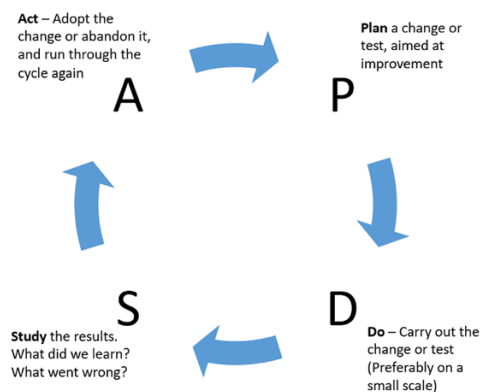


Figura F-3.1. PDSA Cycle (Deming, 1993)

Durante la fase de planificación, y siempre bajo la visión Lean de introducir pequeños cambios, se valoraban las diferentes opciones y cambios a incorporar en el proceso de desarrollo y en la organización del equipo de trabajo. Una vez ejecutado el experimento se evaluaban los resultados atendiendo a las métricas de rendimiento del equipo. Si los resultados eran favorables (se había mejorado o mantenido el rendimiento) se adoptaba el cambio y en caso contrario se descartaba. A partir de ese momento se planificaba una nueva iniciativa de mejora.

3.1 Protocolo de recogida de datos

El protocolo de recogida de datos, en el contexto de la validación de la hipótesis de la presente Tesis Doctoral, consiste en la definición de una serie de preguntas que deben ser respondidas después de llevar a cabo la ejecución del caso de estudio.

En la primera fase de nuestro caso de estudio, la validación tiene por objetivo comprobar **la mejora de la productividad del equipo de desarrollo mediante la transformación de sus métodos de trabajo.**

A continuación, se presentan una serie de preguntas del caso de estudio (PCE) alineadas con dicho objetivo, a las que se dará respuesta en la sección 3.3 *Análisis de resultados*. Las preguntas presentadas, son planteadas por el impulsor del método Lean Kaban y propuestas en su libro de referencia (Anderson, 2010) como objetivos alcanzables a través de la aplicación del método:

- PCE.1 ¿Se ha conseguido optimizar el proceso de desarrollo de partida?
- PCE.2 ¿Se ha incrementado la calidad de las entregas llevadas a cabo por el equipo?
- PCE.3 ¿Se ha mejorado la predictibilidad de los tiempos de entrega del equipo?
- PCE.4 ¿Se ha mejorado la satisfacción del equipo?
- PCE.5 ¿Se han podido crear espacios para la innovación dentro del equipo?
- PCE.6 ¿Se ha conseguido simplificar las tareas de priorización de los trabajos que debe realizar el equipo?
- PCE.7 ¿Se ha incrementado la transparencia sobre el proceso y su funcionamiento?

PCE.8 ¿Se ha conseguido establecer una cultura de mejora continua?

En cuanto a la recolección de datos que permitirá obtener evidencias con las que dar respuesta a las anteriores preguntas, se ha decidido utilizar las siguientes fuentes de información:

- La observación directa de la ejecución del caso de estudio.
- Las métricas obtenidas del registro de los datos llevado a cabo a lo largo del caso de estudio.
- Medidas de rendimiento del proceso de desarrollo tomadas durante la ejecución del caso de estudio.
- Entrevistas con los miembros del equipo.

3.2 Ejecución del caso de estudio

El primer paso del caso de estudio consistía en iniciar la visualización del trabajo en curso del equipo. Para ello se creó un **tablero Kanban físico** que nos permitió conocer fácilmente todo el trabajo que el equipo tenía asignado. A su vez, se pudo establecer al instante cuánto de ese trabajo podía ser considerado como “trabajo en curso” del equipo. Esto es, todos los encargos de trabajo sobre los que se había realizado alguna acción (aunque actualmente el equipo no trabajara directamente sobre ello). El enfoque utilizado fue dejar fluir el trabajo, observando cómo entraba en el *backlog* y cómo el equipo lo movía *aguas abajo*.

Una vez realizada esta primera aproximación, el siguiente paso consistió en la **reducción del trabajo en curso** del equipo. Dado que la limitación del WIP es una de las piedras angulares del método Lean Kanban, ésta fue una de las primeras acciones que se llevaron a cabo en el caso de estudio.

El análisis de las **fuentes de variabilidad** es un aspecto importante dentro del método Lean Kanban. En concreto, en nuestro caso de estudio dicho análisis reflejó que existían dos fuentes principales. Por un lado, el **tamaño** de los trabajos, debido a que en el tablero se mezclaban encargos que podían requerir varias semanas de esfuerzo con otros que podrían requerir sólo unos días u horas.

La segunda fuente de variabilidad era la **tipología** de los encargos de trabajo. Se observó que el tablero incorporaba encargos asociados a solicitudes realizadas directamente por los usuarios, con trabajos relacionados con aspectos de migraciones tecnológicas o reducción de deuda técnica. Dado que enfrentarse a la primera fuente de variabilidad (tamaño de los trabajos) requeriría introducir una fase previa de estimación (práctica que inicialmente no realizaba el equipo), nos

centramos en reducir la segunda fuente de variabilidad, incluyendo para ello el concepto de clase de servicio (Anderson, 2010). De este modo, se dividieron los encargos de trabajo en dos categorías:

- Trabajo Estándar: aquellas peticiones recibidas directamente de los usuarios.
- Trabajo Técnico o de Calidad: encargos que no tenían que ver con ningún aspecto funcional de los sistemas de información.

Una vez introducido el concepto de clase de servicio y dejando pasar un tiempo prudencial para que el equipo asimilase su uso, se expandió el concepto añadiendo nuevas clases de servicio que permitirían clasificar en un mayor nivel de detalle los encargos de trabajo del equipo: Bug, Urgente, Ligera y Fecha límite (en la sección 3.2.3 *Análisis de las fuentes de variabilidad* se describen con mayor detalle las diferentes clases de servicio).

Por otra parte, la definición de clases de servicio facilitó la definición de diferentes **políticas** por parte del equipo que debían ser explicitadas. De este modo, todos los miembros del equipo conocían las reglas del juego que guiaban el proceso de desarrollo. Esto, facilitó en gran medida la atención a los acuerdos de nivel de servicio definidos para cada clase de servicio, dirigiendo el trabajo del equipo sin la necesidad de una estructura de control y mando clásica.

Dentro del protocolo de recogida de datos del caso de estudio, se encontraba la definición y captura de las **métricas** principales del proceso. Tratándose de un método enmarcado en la filosofía Lean, las principales métricas recogidas estaban asociadas a los tiempos de desarrollo de los trabajos afrontados por el equipo (Corona & Pani, 2013). De este modo, se definieron como métricas principales el Lead-Time y el Cycle-Time. Dado que la incorporación de métricas implicaba un esfuerzo adicional en su captura, el resto de indicadores se incorporaron paulatinamente tras un análisis detallado de su validez en el proceso. En la sección 3.2.2 *Métricas utilizadas en el proceso de mejora* se describen las motivaciones que nos llevaron en la elección de cada métrica.

La práctica de **gestión del flujo** va más allá de permitir que el trabajo fluya por el proceso de desarrollo, sino que se debe buscar la uniformidad y suavidad del mismo, la búsqueda de buenos resultados económicos y anticiparnos a las necesidades de los clientes (Anderson, 2010). Conseguir que el trabajo fluya a través del proceso de desarrollo de una forma uniforme y suave requirió en nuestro caso la puesta en marcha de varias iniciativas. Nos enfocamos en reducir los perfiles especialistas y amplificar el conocimiento de cada miembro del equipo con respecto a todos los proyectos de los que eran responsables. Reduciendo los perfiles especialistas, conseguimos minimizar los cuellos de botella que ocasionan estos

roles y facilitamos el balanceo de recursos para distribuirlos allá donde eran más necesarios. Para obtener el máximo rendimiento de la redistribución de recursos es necesario disponer de un equipo que conoce y comparte conocimiento, de modo que puedan ser balanceados a los proyectos que están ocasionando problemas de flujo. Para ello, se pusieron en marcha iniciativas para potenciar la compartición de conocimiento, explicadas con mayor detalle en la sección 3.2.5 *Gestionar el flujo*.

El método Lean Kanban dispone de un mecanismo que facilita **la inspección y adaptación** del propio proceso, factor que comparte con las metodologías ágiles. El equipo debe reservar un tiempo para extraer lecciones aprendidas a través del análisis del proceso, con el objetivo de mejorarlo constantemente (Stellman & Greene, 2015). En nuestro caso de estudio, se definieron reuniones semanales para llevar a cabo una inspección detallada de los aspectos organizativos y técnicos que habían afectado al proceso de desarrollo.

El objetivo final de la puesta en marcha de estos bucles de retroalimentación es arraigar en el equipo el concepto de mejora colaborativa. Aunque en nuestro caso de estudio fue alcanzado dicho objetivo, consideramos que su **sostenibilidad** a largo plazo es más difícil de lograr. Por ello, nuestra propuesta incorpora la aplicación de técnicas de coaching emocional e inteligencia emocional grupal, llevadas a cabo en la segunda fase de nuestro caso de estudio y explicadas en detalle en el Capítulo 4.

A lo largo de las secciones 3.2.1 a 3.2.6 se analizarán en detalle las principales acciones, métricas y decisiones llevadas a cabo durante la ejecución del caso de estudio. Están estructuradas atendiendo a los aspectos que consideramos de mayor interés en la aplicación del método: la puesta en marcha, las métricas del proceso, las fuentes de variabilidad, la visualización, la gestión del flujo y los bucles de retroalimentación.

3.2.1 Puesta en marcha de la iniciativa

El primer paso llevado a cabo fue la definición de un **tablero Kanban** lo más simple posible, partiendo para ello de la definición más sencilla que se puede plantear sobre cualquier proceso: Trabajo Pendiente (Pila) – Trabajo en Curso (En Desarrollo) – Trabajo Finalizado (Finalizado). Este primer tablero se rellenó con la **situación real** de los encargos de trabajo del equipo. Para ello, se colocó en la *Trabajo Pendiente (Pila)* todos los encargos de trabajo registrados en las diferentes herramientas y formatos que utilizaba el equipo: la herramienta ALM, los correos electrónicos con marcas de seguimiento, las peticiones abiertas en el sistema de gestión de incidencias y las diferentes notas tomadas en las últimas reuniones. Todo este trabajo fue reescrito en los tickets (las tarjetas Kanban que se moverán a través

del tablero) permitiendo visualizar todo el trabajo al que hacía frente el equipo de desarrollo.

El siguiente paso consistió en visualizar el *Work-in-progress* del equipo, por lo se desplazó a la columna *En Desarrollo* todos los tickets sobre los que se había realizado algún trabajo. No únicamente los que se estuviera trabajando en este momento, sino aquellos que habían sido iniciados, independientemente de su estado actual. El objetivo fue definir claramente el **trabajo en curso del equipo**, tareas que de una u otra forma se habían iniciado, pero no habían concluido. La lista obtenida fue suficientemente dispar, desde trabajos asociados a proyectos de semanas de duración hasta incidencias que podrían requerir un esfuerzo de menos de una hora.

Uno de los objetivos del método Lean Kanban es obtener un flujo de trabajo constante y suave lo cual es difícil de conseguir cuando el trabajo en curso es muy elevado, algo que en este momento inicial se pudo observar gracias al soporte de la gestión visual. Aunque inicialmente al equipo le parecía un concepto abstracto, se limitó desde un primer momento el trabajo de la columna *En Desarrollo*. Para establecer los límites iniciales se utilizó la siguiente regla (Kniberg & Skarin, 2010):

$WIP = 2n - 1$ <p>donde n equivale al número de personas del equipo</p>
--

La utilización de esta fórmula facilita la **limitación del trabajo en curso**, de una forma paulatina y sin estresar en exceso al equipo de trabajo. Se permite que cada miembro del equipo tenga dos encargos de trabajo activos, pero al restarle uno, se obliga a los miembros del equipo a interactuar entre sí, requiriendo una coordinación durante la selección de los trabajos a iniciar.

Así el WIP objetivo para la columna *En Desarrollo*, siguiendo la fórmula descrita, debía ser 9 (al contar con 5 desarrolladores). En cualquier caso, el propio equipo decidió que para potenciar la participación en diferentes proyectos sería preferible bajar el límite para tensionar la selección de trabajos. De este modo, en caso de bloqueos, los desarrolladores no tendrían más remedio que seleccionar tareas que afectaban a proyectos menos conocidos, potenciando con ello la práctica de *Propiedad colectiva de código* de XP (Beck, 2000). Por tanto, se estableció un WIP inicial de 7.

Tanto en (Kniberg & Skarin, 2010) como en (Anderson, 2010) se concluye que el valor inicial del límite no es muy significativo, en realidad se trata de un valor empírico que debe ir regulándose atendiendo al comportamiento del equipo. En las primeras semanas de funcionamiento se ajusta de una forma natural y se mantiene

en continua evolución a medida que se mejora el tablero, en paralelo con la mejora del propio proceso de desarrollo.

El equipo de desarrollo provenía de un modelo de gestión tradicional *Command and Control*, opuesto a la gestión de un equipo Lean Kanban que debe buscar la auto-organización. Para ello, nos marcamos el objetivo a medio plazo de transformar el rol de jefe de proyecto a un enfoque *agile coach* (Adkins, 2010). Ahora bien, siguiendo uno de los principios fundacionales del método Lean Kanban *Respect the current process, roles, responsibilities & titles* (Anderson, 2010), en una primera instancia se evitó realizar ningún cambio en los roles definidos, por lo que se optó por incorporar una nueva columna donde visualizar el trabajo priorizado. De este modo, se mantuvo la gestión *Command and Control* que venía llevándose a cabo, a la vez que se visualizaba la organización del trabajo. La figura F-3.2 muestra el aspecto de la primera versión del tablero.



Figura F-3.2. Ejemplo del tablero Kanban en sus primeras iteraciones

Las primeras iteraciones trascurrieron incidiendo en que el equipo ajustase sus dinámicas diarias para cumplir las (intencionadamente reducidas) normas del tablero: iniciar trabajo desde la columna *Seleccionado*, no sobrepasar los límites establecidos e intentar que la realidad del equipo se represente fielmente en el tablero (para ello era necesario adquirir el hábito de mover los tickets antes de iniciar o al finalizar una tarea).

3.2.2 Métricas utilizadas en el proceso de mejora

Probablemente la frase más popular que justifica la necesidad de definir métricas para conducir las iniciativas de mejora de procesos sea el Principio de Kelvin, enunciado originariamente por el físico y matemático William Thomson (Lord Kelvin): *“If you cannot measure it, you cannot improve it...”* (Gill & Finzi, 1988).

Atendiendo a las razones y el efecto del uso de cada indicador podemos dividir el conjunto de métricas en 5 categorías: planificación, seguimiento del progreso, mejora de la calidad, solución de problemas de procesos de software y motivación de las personas (Kupiainen et al., 2015).

Por nuestra parte, seleccionamos el Lead-Time (plazo de entrega) y el Cycle-Time (tiempo de proceso) dado que la medición del tiempo de espera requeriría una mayor captura de datos, complicando la implantación del método (motivo por la que fue descartada). Atendiendo a (Kniberg & Skarin, 2010) podemos definir estas métricas como sigue:

- **Lead-Time:** Tiempo desde que se recibe un encargo de trabajo de cualquier tipo en el equipo (momento en el que entra en el *backlog*) hasta que se resuelve (generalmente implica el despliegue de una versión de software que satisface dicha solicitud).
- **Cycle-Time:** Tiempo de proceso, esto es, tiempo que un encargo de trabajo tarda en recorrer todo el proceso de desarrollo. El tiempo se inicia en el momento de la selección del trabajo (puesto que se considera que la priorización de dicho trabajo ya implica en sí un esfuerzo por parte del equipo) hasta que se resuelve.

Dado que la transformación del equipo requería la incorporación de cambios en su forma de trabajo, para poder evaluar si dichos cambios mejoraban o degradaban el comportamiento del proceso era necesario conocer el **rendimiento** del mismo: número de encargos de trabajo llevados a cabo en un determinado tiempo (*Throughput*). En nuestro caso, este indicador puede ser contabilizado como la suma del esfuerzo requerido en cada ticket finalizado durante dicho periodo de tiempo. De esta forma, podíamos conocer si el proceso mantenía un rendimiento constante o si, por algún motivo, se estaba degradando.

El servicio de mantenimiento del software tiene unas características muy distintas al desarrollo de un nuevo producto. Los requisitos generalmente son menos ambiguos en un servicio de mantenimiento puesto que existe un producto real utilizado por los usuarios. Esto facilita definir las nuevas características que les interesaría incorporar o los cambios a implementar en la funcionalidad actualmente desplegada. Por contra, se genera un continuo flujo de encargos de trabajo (provenientes de distintos niveles organizativos) que se van acumulando y que en ocasiones son difícil de priorizar. Por otra parte, un sistema en producción requiere solucionar las incidencias que no fueron detectadas en la fase de pruebas. Estas incidencias pueden llegar a ser bloqueantes, por lo que se deberán poder atender de urgencia. Además, mientras se mantiene el sistema se producen cambios

tecnológicos, transparentes a los usuarios, que obligan a realizar adaptaciones en el código fuente. Por último, cuanto más se conocen los sistemas, más debilidades veremos en ellos. Los riesgos y la deuda técnica asociada serán por tanto otra fuente generadora de encargos de trabajo.

Los diferentes tipos de trabajo anteriormente descritos son importantes para una buena prestación del servicio, pero el objetivo final de cualquier sistema es que sus usuarios encuentren una utilidad práctica que resuelva determinados problemas. El **número de peticiones en espera** de los usuarios (dejando aparte riesgos, aspectos relacionados con la calidad o cambios de infraestructura) será un *radiador de información*¹ de vital importancia para evaluar el rendimiento del equipo, desde la perspectiva de la calidad del servicio percibida por el usuario.

La tabla T-3.1 resume las métricas utilizadas durante la ejecución del caso de estudio.

Métrica
Plazo de entrega (<i>Lead-Time</i>)
Tiempo de proceso (<i>Cycle-Time</i>)
Rendimiento (<i>Throughput</i>)
Peticiones en espera

Tabla T-3.1. Métricas definidas para el caso de estudio

3.2.3 Análisis de las fuentes de variabilidad

Cuando las métricas comenzaron a mostrar suficiente información, se pudo observar que la **distribución** del Lead-Time era muy elevada, dificultando la **predictibilidad** del proceso. Aunque se disponía de un valor medio de Lead-Time, su dispersión impedía predecir con un nivel aceptable el tiempo de entrega.

En este punto, un análisis de variabilidad nos permitió identificar dos fuentes principales. Por un lado, el tamaño de los trabajos a realizar, puesto que existían encargos que, al ser significativamente grandes, sus tiempos se alejaban de la media. La segunda fuente de variabilidad eran los diferentes tipos de encargos de trabajo que se introducían en el tablero.

¹ Radiador de información: Cualquier artefacto que transmita información del equipo, y que se encuentra visible tanto para el equipo como para la organización (Cockburn, 2006).

Optamos por afrontar en primera instancia la segunda fuente de variabilidad puesto que, el problema del tamaño, requería una estimación de todos los encargos que entrarán en el tablero y, además, la descomposición de aquellos que demandaran un considerable esfuerzo de desarrollo. Dado que en aquel momento el equipo no realizaba ninguna estimación, este cambio implicaba una importante adaptación, por lo que aplazamos su implantación.

Para reducir la segunda fuente de variabilidad se incluyeron en el proceso varias clases de servicio que permitieran clasificar, más adecuadamente, los encargos de trabajo. Las empresas de prestación de servicios están muy habituadas a trabajar con estos conceptos puesto que acostumbran a ofrecer un nivel de servicio acomodado a cada clase de servicio (véanse las clases *business* o turista en las aerolíneas, por ejemplo). De igual modo, la prestación de un servicio de mantenimiento de software debe acomodar el nivel del servicio prestado a cada tipo de trabajo recibido. Clasificarlos mediante una **clase de servicio** concreta, permite establecer diferentes objetivos de tiempos de entrega.

Durante un tiempo, el equipo trabajó exclusivamente con las clases de servicio *Estándar* y *Técnica/Calidad*, buscando con ello una distribución más uniforme del Lead-Time:

- Clase Estándar: trabajos que aportan valor directo al usuario, básicamente ampliando funcionalidad o solucionando alguna limitación.
- Clase Técnica o de Calidad (ToQ): trabajos derivados de cambios de infraestructura o relacionado con la calidad –por ejemplo, *refactorizar* la implementación de una parte de código de difícil mantenibilidad–.

Puesto que la clase Estándar era la encargada de ofrecer **valor** al usuario, se **priorizaba** en el tablero en detrimento de la clase ToQ. De este modo, de forma natural, estas últimas deberían distribuirse en la parte superior del Lead-Time mientras que la clase Estándar conseguiría mejorar sus tiempos de entrega. Atendiendo a esta estrategia, con el transcurso de los ciclos, se fueron distanciando estabilizándose alrededor de los valores que se muestran en la tabla T-3.2.

CLASE DE SERVICIO	LEAD-TIME
Estándar	11,5 días
Técnica o Calidad	16,7 días

Tabla T-3.2. Lead-Time asociado a cada clase de servicio

Una vez que el equipo se acostumbró a trabajar con estas clases de servicio, se ampliaron en base a las expuestas en (Anderson, 2010). La ampliación del número de clases de servicio permitió mejorar la clasificación de los encargos de trabajo. Por otra parte, se definieron acuerdos de nivel de servicio (ANS) asociados a cada clase, facilitando con ello la tarea de priorización del equipo. Por ejemplo, no tiene sentido optimizar el Lead-Time de un trabajo que tiene una fecha límite de entrega cuando, la expectativa del servicio por parte del usuario, se basa en no sobrepasar dicha fecha.

En la tabla T-3.3 se describen las distintas clases de servicio definidas en nuestro caso de estudio y sus acuerdos de nivel de servicio asociados.

Clases de servicio	Acuerdo de Nivel de Servicio (ANS)
ESTÁNDAR: Cualquier encargo de trabajo solicitado por un usuario que requiera modificar código fuente.	<ul style="list-style-type: none"> Al tratarse de una petición de usuario el objetivo será minimizar el Lead-Time.
TÉCNICA O CALIDAD (ToQ): Encargos de trabajo derivados de cambios de infraestructura o relacionados con la calidad de producto.	<ul style="list-style-type: none"> Al tratarse de asuntos que generalmente no aportan valor a los usuarios, se sacrificará su Lead-Time en favor de la clase Estándar. No se incluirá más de un trabajo ToQ simultáneamente en el tablero. Siempre debe existir un trabajo ToQ en el tablero.
BUG: Defecto detectado en la fase de pruebas.	<ul style="list-style-type: none"> Se corregirá en el momento de la detección si el WIP lo permite.
URGENTE: Encargo de trabajo urgente que debe ser atendido de inmediato.	<ul style="list-style-type: none"> Se priorizará en detrimento de cualquier otro encargo de trabajo. Esta clase de servicio no está sujeta a las restricciones del WIP del tablero.
LIGERA: Encargo de trabajo que requiere un esfuerzo mínimo de resolución sin necesidad de modificar código fuente.	<ul style="list-style-type: none"> Al tratarse de una petición de usuario el objetivo será minimizar el Lead-Time.
FECHA LÍMITE: Cualquier encargo de trabajo en el que existe claramente definida una fecha límite.	<ul style="list-style-type: none"> No preocupa el Lead-Time, únicamente entregar antes de la fecha límite.

Tabla T-3.3. Clases de servicio y ANS asociados

La segunda fuente de variabilidad que afectaba a la distribución del Lead-Time era la diferencia existente en el **tamaño** de los encargos de trabajo. Para poder ajustar el tamaño de todos los encargos que se introducen en el tablero era necesaria la estimación previa de los mismos. Se decidió utilizar la estimación relativa (Cohn, 2005) que permite valorar el esfuerzo que supondrá llevar a cabo un determinado

encargo de trabajo, utilizando como unidad de medida, otro trabajo que nos resulte familiar. En lugar de estimar en horas o días, cuando llevamos a cabo una estimación relativa utilizamos un valor que aproxime el tamaño de dicho encargo.

Existen diferentes formas de definir este valor: puede ser numérico (Puntos) o utilizando el tamaño de las tallas de camisetas (*T-Shirt Sizing*). En nuestro caso utilizamos puntos de historia para definir el tamaño de los trabajos y utilizamos la técnica de *Planning Poker* para su estimación (Grenning, 2002). Al tratarse de un proceso de estimación grupal facilita, como beneficio secundario, la transferencia de conocimiento entre los miembros del equipo.

Para evitar que se introdujeran encargos de trabajo demasiado grandes, se estableció la política de que cualquier trabajo igual o superior a 20 puntos debía ser dividido en varios tickets. Los datos históricos mostraban que a partir de este tamaño el trabajo aumentaba considerablemente su Lead-Time. De esta forma, se consiguió reducir la segunda fuente de variabilidad del proceso y se mejoró la predictibilidad de sus tiempos de entrega.

3.2.4 Visualización

La gestión visual es uno de los elementos más diferenciadores del método Lean Kanban y es habitual que sea de las primeras cuestiones a tener en cuenta tras la decisión de aplicar el método. Para ello, Lean Kanban incorpora un gran número de posibilidades de visualización, aunque siempre se debe tener en cuenta que el objetivo final de estas señales visuales es la toma de decisiones. Con este objetivo en mente se implementaron los siguientes mecanismos de visualización:

Información contenida en los tickets

Los tickets son el soporte utilizado para registrar y visualizar el trabajo que transcurre a lo largo del proceso. En nuestro estudio fueron evolucionando significativamente con el tiempo. En una primera aproximación el equipo utilizó *Post-it* como soporte físico y se especificó un conjunto de información mínimo (ver figura F-3.3.):

- Descripción del trabajo a realizar por los ingenieros de software (centro del ticket).
- Fecha de entrada en el *backlog* (esquina superior izquierda).
- Fecha de entrega (esquina inferior derecha) que se rellenaba cuando se completaba el trabajo.

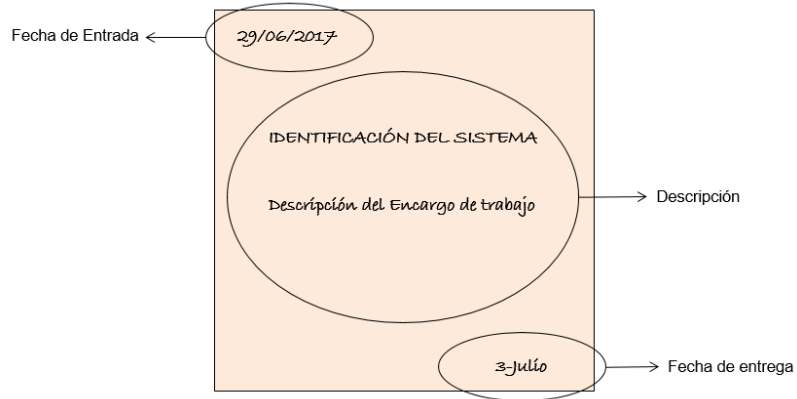


Figura F-3.3. Primera versión del ticket

Transcurrido los primeros ciclos, como se explicó en la sección 3.2.3 *Análisis de las fuentes de variabilidad*, se fueron clasificando los diferentes encargos de trabajo atendiendo a su tipología. Por ello, se introdujo un elemento visual que permitiera identificarlos fácilmente (en la figura F-3.4 el símbolo “E” representa el tipo de trabajo “Estándar”). Por otra parte, ante la necesidad de obtener información para el cálculo del *Cycle-Time* (ver sección 3.2.2 *Métricas utilizadas en el proceso de mejora*) se requirió añadir la fecha en la que se comenzaba a trabajar con cada ticket (esquina superior derecha).

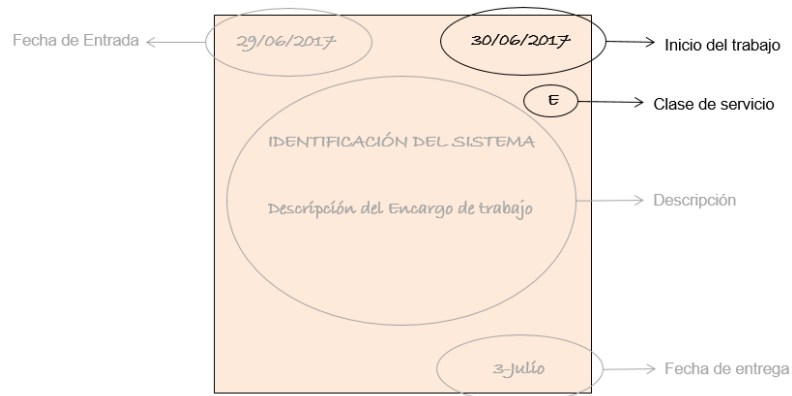


Figura F-3.4. Segunda versión del ticket

Posteriormente (ver figura F-3.5) se incluyó la estimación relativa de los trabajos, como se comentó en la sección 3.2.3 *Análisis de las fuentes de variabilidad* (véase el “2” en la parte inferior que indica que el trabajo asociado a este ticket tiene estimado un esfuerzo de 2 puntos de historia). Igualmente, se añadió la referencia a la herramienta de seguimiento de tareas *Team Foundation Server* (esquina inferior izquierda).

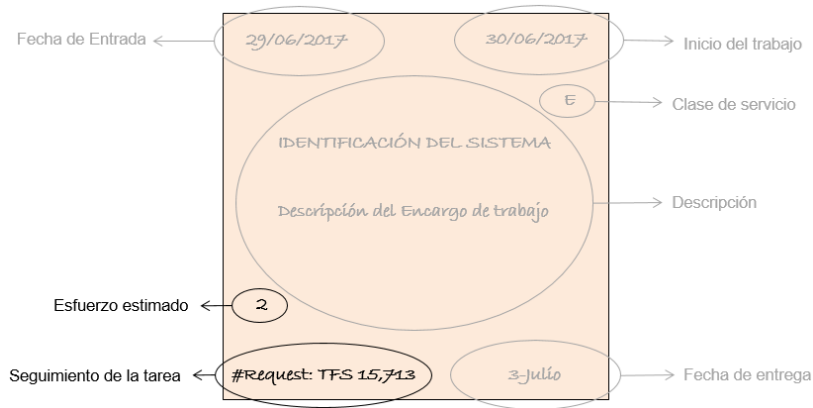


Figura F-3.5. Tercera versión del ticket

Como se puede observar de la figura anterior, según se avanzó en la aplicación del método, la información de los tickets aumentó considerablemente. El hecho de utilizar post-it estándar, sin ningún formato, provocó que en determinadas ocasiones la información no se distribuía con el mismo criterio, provocando errores en el tratamiento de los datos. Para evitar este problema, se diseñó un ticket con un formato predefinido que permitía estructurar mejor la información.

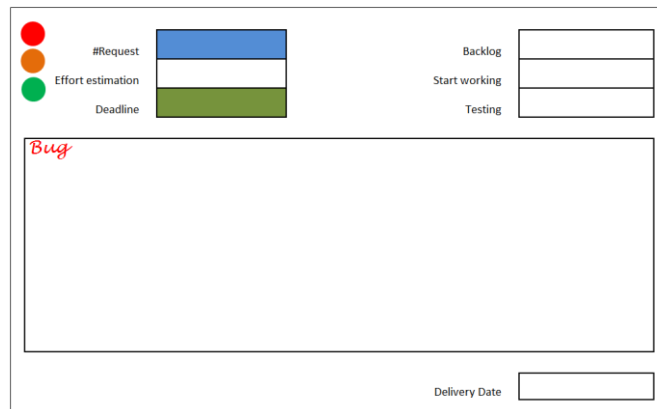


Figura F-3.6. Versión estandarizada del ticket

En la figura F-3.6 se puede observar dicho formato en el que, además de la información comentada, existen campos para definir fechas límites (si la clase de servicio lo requería) y la fecha de entrada en la columna de *Aceptación* (con el objetivo de tomar medidas de la fase de pruebas). Para diferenciar las clases de servicio se utilizaron códigos de colores en los tickets:

- Punto rojo: clase de servicio urgente.
- Punto naranja: clase de servicio técnica o relacionada con la calidad.
- Punto verde: clase de servicio ligera (requiere poco esfuerzo).
- Marca azul: refleja que la clase de servicio es creada a petición de un usuario permitiendo visualizar claramente el trabajo por el que los usuarios finales están esperando.
- Marca verde: permite visualizar claramente las tareas de la clase de servicio que tienen asociada una fecha límite para evitar sobrepasar dichas fechas.
- Bug: si la tarea proviene de un error detectado en la fase de aceptación se incluye la palabra *Bug* al comienzo de la descripción.

A través de estas referencias visuales se permitió conocer, de forma inmediata, todos los encargos de trabajo sobre los que estaba trabajando el equipo, facilitando con ello la toma de decisiones de selección de nuevos encargos y permitiendo que todo el equipo conociera la situación actual del trabajo en curso.

Otra forma de ganar visibilidad propuesta por el método es hacer las políticas explícitas (Anderson, 2010), para asegurar que en todo momento las personas que participan en el proceso tengan claras las normas que rige el flujo de trabajo. Esto es especialmente importante cuando el trabajo *cambia de manos* a lo largo del proceso, por ejemplo, en la entrada de la fase de aceptación.

Nótese que, en un entorno industrial, se cuenta con la propia visualización física del producto y la maquinaria mientras que, en las tareas llevadas a cabo por trabajadores del conocimiento, el proceso en sí no tiene un rastro físico a lo largo de la organización. De ahí que a medida que los procesos incrementan su complejidad, se producen situaciones en que no está claro cómo proceder. La tabla T-3.4 muestra varios ejemplos de políticas asociadas a las columnas del tablero que fueron definidas para nuestro caso de estudio.

Es importante pensar en un proceso como un conjunto de **políticas que gobiernan el comportamiento** (Anderson, 2010). Haciendo **explícitas** estas políticas conseguimos, ya no sólo visualizar el flujo de trabajo, sino que dotamos de visibilidad al mismo proceso de desarrollo.

Política asociada a la columna “Petición a Operaciones”
<p>Sólo se permite dejar un ticket en esta columna si está preparado para el pase a producción.</p> <p>No implica que tenga que ser desplegado inmediatamente, podría requerir la llegada de otros tickets para un despliegue conjunto.</p> <p>Esta columna no consume WIP.</p> <p>Precaución: Funciona como un <i>buffer</i> por lo que es importante tener en cuenta que ante una espera no obligatoria se extenderá innecesariamente el Lead-Time.</p>
Política asociada a la columna “En producción”
<p>Sólo se permite dejar un ticket en esta columna si la funcionalidad está desplegada en producción y visible por el usuario (puede ser interno en tarjetas Técnicas o de Calidad).</p> <p>Es posible que exista una Petición registrada en el buzón, en tal caso se debe comunicar su resolución y cerrar la Petición.</p> <p>Esta columna no consume WIP.</p> <p>Las tarjetas son retiradas de esta columna cuando finaliza el ciclo (semanal).</p>
Política asociada a la columna “Buffer de Aceptación”
<p>Sólo se permite dejar un ticket en esta columna si forma parte de una épica o tiene algún otro tipo de relación con otras historias que están en desarrollo y requieren ser probadas simultáneamente.</p> <p>Esta columna no consume WIP de Aceptación.</p>
Política asociada a la columna “Hecho”
<p>Sólo se permite dejar un ticket en esta columna si:</p> <ul style="list-style-type: none"> - se ha lanzado la <i>build</i> de Pase a Producción, - se ha verificado que compila y pasa las pruebas automatizadas, - y la <i>build</i> ha desplegado la aplicación automáticamente en los servidores de pruebas, permitiendo ser probada sin ninguna configuración adicional por parte del <i>tester</i>.

Tabla T-3.4. Ejemplos de políticas del tablero

3.2.5 Gestionar el flujo

La práctica de gestión del flujo va más allá de permitir que el trabajo fluya por el proceso de desarrollo. Se debe buscar la uniformidad y suavidad del flujo de trabajo, la búsqueda de buenos resultados económicos y la antelación a las necesidades de los usuarios (Anderson, 2010).

En determinadas situaciones puede ser necesario **sustituir los perfiles especialistas** por otros que ayuden a suavizar el flujo, evitando que existan personas que sufren grandes fluctuaciones en la carga de trabajo en su día a día. De hecho, esta es una característica clave del pensamiento Lean y generalmente suele ser uno de los principales problemas cuando las organizaciones deciden poner en marcha

una iniciativa de este tipo. Un ejemplo clásico en este sentido es la experiencia del constructor de automóviles Porsche, cuya adopción de los métodos Lean permitió salvar la compañía ante una situación financiera complicada. Uno de los principales problemas durante la implantación de la iniciativa Lean fue el miedo a perder el alto grado de especialización de sus ingenieros (mucho más elevado que en el resto de constructores de automóviles). Una vez superada esta transformación, la compañía mejoró en su capacidad para balancear recursos facilitando de este modo su adaptación a la demanda (Womack & Jones, 1996).

En nuestro estudio, el equipo contaba con un ingeniero especialista en bases de datos. Cuando se desarrollan nuevos sistemas de información, este perfil puede ser imprescindible, pero en un servicio de mantenimiento de software descende notablemente su necesidad. De este modo, se llevó a cabo una transformación de sus funciones cambiando y asumiendo nuevas competencias. Así, tomó un mayor protagonismo en el área de aceptación donde realizó tareas de automatización de pruebas. Además, como se consiguió reducir el tamaño de los trabajos (sección 3.2.3 *Análisis de las fuentes de variabilidad*) se invertía más tiempo en preparar los entornos de pruebas. Por ello, se responsabilizó de la automatización de los despliegues, lo que mejoraba el rendimiento del resto del equipo.

El especialista en bases de datos continuó haciendo tareas asociadas a su especialidad, pero cuando realmente aportaba valor, dada la complejidad de la tarea asociada. En caso contrario, el resto de los miembros del equipo pasaron a asumir tareas que antes eran de exclusiva competencia del especialista. Al eliminar la limitación de asignación de tareas relacionadas con la base de datos, se dotó al equipo de una mayor flexibilidad a la hora de balancear recursos, facilitando un flujo de trabajo suave ante tensiones puntuales en el proceso.

Compartir código/conocimiento de los proyectos fue otro factor que influyó positivamente en el establecimiento de un flujo suave de trabajo. De esta forma, se evitaron los *proyectos silos* que únicamente podían ser mantenidos por un integrante del equipo. Esto facilitó que, ante un pico en la demanda, se pudiera gestionar adecuadamente la carga de trabajo mediante la redistribución de recursos.

Para alcanzar este objetivo se llevaron a cabo varias acciones, en primer lugar, la estimación de los encargos de trabajo (sección 3.2.3 *Análisis de las fuentes de variabilidad*) permitía que cada miembro del equipo conociera las necesidades de todos los sistemas de información. Por otra parte, se explicitó una política de selección de trabajos (sección 3.2.4 *Visualización*) que primaba la selección de aquellos sistemas de información sobre los que los técnicos no tenían demasiada

experiencia, esta política no aplicaba a los encargos de trabajo de la clase de servicio *Urgente* (ver sección 3.2.3 *Análisis de las fuentes de variabilidad*) que debían ser atendidas por el técnico con mayor experiencia para acelerar su resolución.

Todas estas iniciativas llevadas a cabo en el equipo, facilitaron la gestión de un flujo suave. De este modo, nos encontramos en disposición de emprender, con mayor garantía de éxito, cambios organizativos, así como adquirir nuevas habilidades y herramientas que complementaran al equipo.

3.2.6 *Implementar bucles de retroalimentación y mejorar colaborativamente*

Un excelente mecanismo introducido por el método para la inspección y adaptación del propio proceso son las reuniones de retrospectiva, utilizadas para evaluar cómo se viene trabajando y facilitar espacios de innovación a fin de mejorar desde la situación actual. Para ello, se decidió establecer reuniones semanales en las que poder revisar los aspectos organizativos y técnicos del último periodo.

Según se fue avanzando en la implantación del método, durante estas reuniones el equipo fue capaz de variar el comportamiento y salida del proceso de desarrollo a través de modificaciones en los WIP (sección 3.2.1 *Puesta en marcha de la iniciativa*), los tamaños relativos del ticket (sección 3.2.4 *Visualización*), así como por la incorporación de políticas (sección 3.2.5 *Gestionar el flujo*).

A medio plazo, para facilitar espacios de innovación dentro del equipo, resultó interesante la incorporación de una auto-evaluación sobre determinados aspectos de interés para sus integrantes (Kniberg & Ivarsson, 2012).

Como agentes de cambio, consideramos que no es suficiente hacer que el equipo tome conciencia de nuestra idea, sino que los mensajes racionales debían complementarse con disparadores emocionales de cambio (Appelo, 2011). Para ello, se elaboró una lista con los intereses personales de cada integrante y después de alinear éstos con el interés general del equipo (y la organización a la que prestaba servicio), se definieron seis áreas a las que prestar especial atención. El enfoque básicamente era el siguiente: identificar las debilidades del equipo, que a su vez coinciden con intereses personales y aprovecharlos como lanzaderas para generar espacios de innovación. Todos los técnicos tenían asignada un área de interés y cuando se producía un bloqueo en el tablero (debido a los WIP) se podían dedicar a mejorar sus competencias en esa área.

En la figura F-3.7 se puede observar un ejemplo de auto-evaluación que se decidió realizar con una periodicidad de tres ciclos. Para cada área se identificaba su estado actual, representado por el color del círculo de la última columna. Si la competencia en un área se consideraba adquirida se representaba con un círculo verde, si estaba en proceso con un círculo amarillo y si existían carencias importantes en esa área, se indicaba con un círculo rojo.

Por otra parte, se analizaba la tendencia apreciada durante los tres últimos ciclos (positiva, negativa o neutra). Se trata de una valoración subjetiva, pero tanto el análisis de las tendencias como el debate generado durante las retrospectivas resultaban ser de gran utilidad para introducir una cultura de mejora continua dentro del equipo. En el ejemplo de la figura F-3.7, el área de auto-organización parte de una evaluación amarilla (en proceso) y en la primera y tercera evaluación la tendencia es positiva por lo que en el último ciclo se da por adquirida la competencia de auto-organización pasando a una evaluación verde.

ÁREA	Ciclo x	Ciclo x+3	Ciclo x+6	Ciclo x+9
Mejora Conocimiento Testing	● ↘	● ↘	● ↘	● ↘
UX	● ↗	● ↗	● ↗	● →
Enfoque a la Calidad Interna del Producto	● ↘	● →	● ↗	● ↗
Participación en la definición del Proceso	● ↗	● →	● ↗	● ↗
Auto-Organización	● ↗	● →	● ↗	● →
Nuevas tecnologías	● →	● →	● →	● →

Figura F-3.7. Representación de autoevaluación del equipo

El aspecto más complicado de la mejora continua en las organizaciones es cambiar el comportamiento de las personas (Appelo, 2011). Nuestra propuesta para establecer una cultura de mejora continua es la generación de *tiempo libre* entre los miembros del equipo.

En el entorno industrial, resulta difícil hacer entender a la dirección que parar una máquina (y sus operarios) en situaciones concretas de la producción pueda ser rentable a la empresa (Goldratt & Cox, 1984). El sentido común aconseja que, para ser amortizadas, estas máquinas deben trabajar al 100% de su capacidad. En realidad, esta decisión genera una cantidad de inventario innecesario entre los diferentes subprocesos de la producción (creando buffers intermedios) y colateralmente alarga el tiempo de ciclo. Ambos efectos son negativos para el

funcionamiento del sistema por lo que, detener una máquina bajo determinadas circunstancias puede aumentar la productividad global de una planta de producción.

De forma similar, en los trabajadores del conocimiento existe una relación entre la productividad global de un equipo y el *tiempo libre* de sus integrantes. Para analizar este tema es necesario definir los parámetros con los que medir la productividad para un determinado equipo puesto que, atendiendo a su misión específica, puede medirse de diferente forma. En nuestro caso de estudio se manejaron las métricas explicadas en la sección 3.3 *Métricas utilizadas en el proceso de mejora*.

Cada equipo deberá tener una serie de parámetros por los que valorar su desempeño a lo largo del tiempo, siendo imprescindible tomar las métricas necesarias para poder garantizar que la iniciativa de mejora lleva la dirección adecuada o variar el rumbo si algún cambio está perjudicando el desempeño total del equipo. De otro modo, se estarán tomando decisiones sin garantizar que ayudan a alcanzar los objetivos definidos.

En este sentido, los bloqueos que generan las políticas del tablero (WIP principalmente), puede requerir el análisis del desempeño global a medio plazo para garantizar que no se está degradando.

Ante determinadas situaciones del tablero puede darse la circunstancia que un miembro del equipo no pueda iniciar nuevo trabajo ni ayudar a otro compañero sin caer en la Ley de Rendimientos Decrecientes (Brue, 1993) por lo que, hasta que se resuelva el bloqueo del tablero, el recurso bloqueado estará temporalmente *ocioso*. Ahora bien, gestionando esos bloqueos adecuadamente como tiempos de innovación dedicados a las áreas de interés compartido, se puede redundar en el bien general aumentando el desempeño global del equipo.

Desde nuestro punto de vista, lo importante en este asunto es conseguir alinear los tiempos de innovación con las deficiencias del proceso de desarrollo. De este modo, a medio plazo los indicadores de rendimiento no sólo no se degradarán, sino que mejorarán, como pudimos observar en el presente caso de estudio y detallamos en la próxima sección de análisis de resultados.

3.3 Análisis de resultados

Una vez llevada a cabo la ejecución del caso de estudio, nuestro siguiente paso consiste en dar respuesta a las preguntas planteadas en la sección 3.1 *Protocolo de recogida de datos*.

PCE.1 ¿Se ha optimizado el proceso existente?

Respuesta: **Sí.**

A lo largo de los diferentes ciclos y a través de las reuniones de retrospectiva se ha mejorado el proceso de desarrollo de software que utilizaba el equipo. Existe una gran variedad de mejoras implementadas desde diferentes puntos de vista: mejoras organizativas, técnicas y orientadas a la mejora de la calidad. En la sección *3.4 Lecciones aprendidas*, se enumeran las diferentes prácticas puestas en marcha. Todas las mejoras implementadas orientadas al proceso de desarrollo fueron clave en la ejecución del caso de estudio puesto que, en la premisa de transformación de un equipo, siempre subyace la mejora su proceso de trabajo. A lo largo del caso de estudio hemos utilizado exitosamente el método Lean Kanban como agente facilitador del cambio, apoyándonos en sus dinámicas para optimizar gradualmente el proceso del equipo, sin cambios disruptivos en la forma de trabajar ni costosos procesos de reingeniería.

Un indicador clave durante la optimización del proceso es el Cycle-Time ya que está asociado directamente con los tiempos del proceso de desarrollo. Si bien es cierto que no se puede afirmar directamente que a menor tiempo de ciclo mejor proceso, sí resulta interesante este indicador porque permite valorar en el corto plazo los cambios (técnicos, metodológicos u organizativos) introducidos en el proceso, analizando rápidamente sus efectos en la velocidad de desarrollo del equipo. Desde nuestro punto de vista, es un indicador a tener muy en cuenta en cada cambio en el sistema Kanban y en nuestro caso de estudio su variación ha sido analizada en detalle en cada mejora incorporada al proceso.

Para facilitar el estudio del Cycle-Time se utilizaron prácticamente todas las clases de servicio, quedando únicamente excluidas las clases Ligeras y Bugs dado que, al no transitar por todas las columnas del tablero, alterarían (a la baja) la media de este indicador. En la figura F-3.8 se puede observar la evolución de esta métrica para las clases de servicio Estándar, ToQ, Urgente y Fecha límite. Nótese que la fluctuación del Cycle-Time refleja con mayor rapidez los efectos de los cambios introducidos en el tablero.

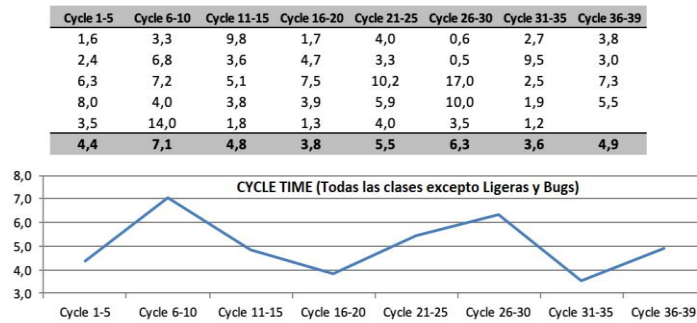


Figura F-3.8. Cycle-Time (Todas las clases excepto Ligeras y Bugs)

Se debe tener en cuenta que el Cycle-Time incluye tanto el tiempo en que un ticket está en proceso como los tiempos de bloqueos. De ahí que si comparamos el tiempo de ciclo de todos los trabajos con el de la clase Estándar veamos una diferencia importante, dado que, como se comentó en la sección 3.4 *Lecciones aprendidas*, al tratarse de una petición de usuario el ANS de esta clase de servicio se enfoca a minimizar su tiempo de entrega.

La figura F-3.9 refleja el Cycle-Time para esta clase de servicio. En general mantiene una tendencia decreciente para la clase Estándar en comparación con los datos de la figura F-3.8 que agrupa todas las clases de servicios (excepto ligeras y bugs). Esto es algo lógico dado que el objetivo de nuestra investigación era llevar a cabo una transformación Lean, por lo que el equipo debía mantener claramente definido el foco principal en la entrega del mayor valor posible a los usuarios (representado en la clase estándar).

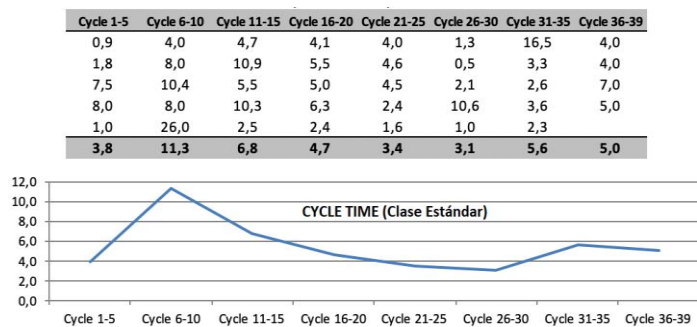


Figura F-3.9. Cycle-Time (Clase Estándar)

De este modo, podemos afirmar que no sólo se mejoró el tiempo medio de proceso del equipo, sino que a su vez se optimizó con la incorporación de un importante número de mejoras (ver sección 3.4 *Lecciones aprendidas*).

Aunque el método Lean Kanban no requiere estrictamente de ciclos puesto que se enfoca en el flujo continuo de trabajo, desde nuestro punto de vista introducir el concepto de ciclo permite tener puntos de control en los que establecer medidas y analizar tendencias para conducir la optimización del proceso.

En nuestra propuesta de implantación del método hacemos coincidir la finalización del ciclo con la reunión de retrospectiva (sección 3.7 *Implementar bucles de retroalimentación y mejorar colaborativamente*). Todo el equipo analiza los indicadores y la situación del trabajo llevado a cabo, planteando nuevas iniciativas de mejora o retirando las que no están funcionando.

Por último, para ilustrar el efecto del proceso en términos de mejoras tangibles, pasemos a uno de los resultados: la decisión de celebrar reuniones de *Lean Coffee* regularmente. Esta práctica es interesante para potenciar la cultura del cambio y la mejora continua dentro de las organizaciones a través de conversaciones informales. Estas sesiones generaron un buen número de iniciativas metodológicas y técnicas.

Por ejemplo, se establecieron dos políticas de prueba: 1) las pruebas automatizadas deberían lograr al menos un 60% de cobertura de código en nuevos desarrollos; 2) cada solicitud de cambio de un sistema en mantenimiento debe incluir de 1 a 10 pruebas automatizadas. No tenemos datos sobre el resultado de la primera aplicación de la política ya que no se abordaron nuevos desarrollos durante esta experiencia. Sin embargo, la implementación de la segunda política resultó en un crecimiento relevante en el número de pruebas automatizadas. Antes de su aplicación, menos del 7% de los proyectos en mantenimiento incluían pruebas automatizadas, mientras que más del 50% de los proyectos las habían incorporado al final de la experiencia.

PCE.2 ¿Se ha incrementado la calidad de las entregas llevadas a cabo por el equipo?

Respuesta: **Sí.**

El método Lean Kanban permite establecer claros criterios de calidad como políticas definidas en el propio tablero. Obviamente, no todos los cambios se pudieron llevar a cabo en la primera semana de aplicación, pero gradualmente se fueron incluyendo políticas que, en cierta forma, incitaron a todos los miembros del equipo a realizar entregas con mayor calidad.

En la sección 3.2.4 *Visualización*, se especifican varios ejemplos de las políticas del tablero, entre las que se pueden encontrar algunas muy asociadas a la calidad del producto software, como, por ejemplo, las políticas asociadas a la columna *Hecho*:

Sólo se permite dejar un ticket en esta columna si:

- *se ha lanzado la build de Pase a Producción,*
- *se ha verificado que compila y pasa las pruebas automatizadas,*
- *y la build ha desplegado la aplicación automáticamente en los servidores de pruebas, permitiendo ser probada sin ninguna configuración adicional por parte del tester.*

Otra política incorporada en este sentido fue la obligatoriedad de que cualquier cambio introducido en un sistema de información debía descender o al menos mantener las alertas de análisis estático de código previas, facilitando la mejora gradual de la deuda técnica de los proyectos responsabilidad del equipo.

Por otra parte, para compensar la tendencia natural de los equipos Lean a la entrega de valor a los usuarios se explicitó la política de que siempre debía existir en el tablero algún ticket de la clase Técnica o de Calidad. La implementación de esta política permitió subir el número de trabajos asociados con la calidad de unos niveles en torno al 6% iniciales al 17% de los encargos de trabajo. Como se desprende del análisis del Lead-Time y de las peticiones en espera (tiempos de entrega menores y peticiones en espera dentro de los límites de control), invertir en este tipo de trabajos incide en un buen rendimiento global y finalmente redundando en el usuario.

PCE.3 ¿Se ha mejorado la predictibilidad de los tiempos de entrega del equipo?

Respuesta: **Sí.**

El mero hecho de limitar el trabajo en curso –WIP- acorta los tiempos medios de entrega, como se desprende de la aplicación del principio de la Ley de Little (Nord et al., 2012). Pero si además de limitar el WIP, se reducen suficientemente los tamaños de los trabajos, y se dispone de una estimación razonable del esfuerzo de desarrollo (sección 3.2.3 *Análisis de las fuentes de variabilidad*), una rápida gestión visual sobre la situación del tablero (sección 3.2.4 *Visualización*) teniendo en cuenta los tiempos medios de entrega y de proceso permitieron predecir un rango de entrega suficientemente razonable.

Para poder calcular el Cycle-Time y Lead-Time, fue necesario incorporar en los tickets las fechas de entrada y finalización. Dado que sólo suponía un pequeño cambio en las dinámicas del equipo, se consideró adecuado afrontarlo en los primeros ciclos.

En la figura F-3.10 se puede observar la evolución media del Lead-Time de la clase estándar que como se ha comentado anteriormente, es la clase más enfocada a la entrega de valor al usuario. Por ello, acelerando su desarrollo se acelera la entrega de valor.

Los primeros 5 ciclos no son significativos debido a que el tablero se inició con todos los trabajos en una fecha concreta. A partir del sexto ciclo los tiempos alcanzaron sus valores reales, llegando a picos cercanos a los 26 días desde la solicitud a la entrega. Superado el decimosexto ciclo observamos que se estabiliza en valores medios cercanos a los 10 días. Esto implica que a partir de dicho ciclo se consiguió **reducir a la mitad** los tiempos medios de entrega.

Cycle 1-5	Cycle 6-10	Cycle 11-15	Cycle 16-20	Cycle 21-25	Cycle 26-30	Cycle 31-35	Cycle 36-39
2,3	14,2	27,3	9,2	13,0	10,8	17,0	5,0
1,8	15,5	18,3	18,5	14,1	21,5	6,5	9,8
10,8	26,2	8,0	6,0	12,3	2,7	5,9	18,0
9,5	22,0	21,0	10,0	4,6	18,6	18,3	5,0
1,4	26,5	11,7	8,6	9,0	1,0	3,5	
5,2	20,9	17,3	10,5	10,6	10,9	10,2	9,4

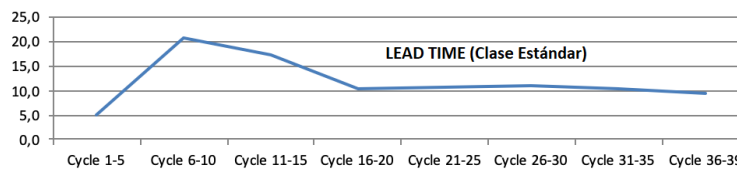


Figura F-3.10. Lead Time (Clase Estándar)

Desde nuestro punto de vista, esta mejora proviene de varios factores que simultáneamente influyen de forma positiva en este indicador:

- Definir claramente unos objetivos y visualizar las políticas oportunas que permitan dirigir los esfuerzos de desarrollo en la dirección adecuada.
- Limitar el trabajo en curso, obligando a finalizar las tareas actuales antes de asumir otras nuevas. Si no se inicia nuevo trabajo en paralelo se reducirán los tiempos medios de entrega.
- La priorización de los trabajos atendiendo a los acuerdos de nivel de servicio definidos (sección 3.2.3 *Análisis de las fuentes de variabilidad*). En los primeros ciclos se llegó a la conclusión de que no tenía sentido esforzarse en reducir el Lead-Time de todos los encargos de trabajo. Los trabajos Técnicos o de Calidad tienen importancia y ante determinadas circunstancias pueden requerir ser priorizadas frente a cualquier otra clase de servicio, pero generalmente pueden llevar un ritmo menor al de las peticiones de los usuarios.
- Todos los ciclos finalizan en retrospectivas en las que se analizan los indicadores principales y se plantean opciones de mejora.
- Uno de los principios del Lean Software Development aconseja retrasar lo más posible las decisiones irreversibles (Poppendieck & Poppendieck, 2007) de modo que, cuando se tomen esas decisiones, se disponga de más información y se tenga más posibilidad de no errar en el diseño o implementación. De este principio se beneficia la clase de servicio “Fecha límite” puesto que al disponer de unos tiempos medios de procesamiento (Cycle-Time) y de una estimación relativa de los trabajos, se estará en disposición de retrasar lo más posible su implementación (controlando el riesgo de sobrepasar la fecha límite). No tiene sentido centrarse en reducir el Lead-Time de este tipo de trabajos puesto que su ANS asociado se focaliza en no sobrepasar la fecha límite (sección 3.2.3 *Análisis de las fuentes de variabilidad*). Adicionalmente, al retrasar estos trabajos conseguiremos mejoras en el Lead-Time de otras clases de servicio. A su vez, es muy habitual en un entorno de mantenimiento de software que las especificaciones cambien e incluso se anulen determinados encargos de trabajo. De este

modo, si se retrasa la implementación hasta el último momento responsable es posible que no llegue a implementarse, evitando incurrir en un nuevo desperdicio.

Por otra parte, el rendimiento del sistema (*Throughput*) fue otro indicador que nos permitió mejorar la predictibilidad del proceso de desarrollo. Como se comentó en la sección 3.2.3 *Análisis de las fuentes de variabilidad*, para facilitar el cálculo de este indicador se empleó la estimación relativa (Cohn, 2005) mediante la técnica de Planning Poker (Grenning, 2002). Por supuesto, el hecho de contabilizar la salida atendiendo a la estimación inicial implica que, en algún caso, ante errores de estimación, el rendimiento de salida no sea real. Ahora bien, el análisis de esta desviación se realizaba en las sesiones de retrospectivas lo que, por otra parte, permitía acumular experiencia para mejorar las futuras estimaciones. Esta es una diferencia significativa con respecto a la primera implantación del método Lean Kanban (Anderson & Dumitriu, 2005) puesto que Anderson proponía la predicción estadística (evitando todo esfuerzo de estimación por parte de los desarrolladores) mientras que en nuestro caso **proponemos la estimación relativa** como método que permite afinar el rendimiento de salida del proceso y distribuir el conocimiento de los proyectos.

Como se puede apreciar en la figura F-3.11, con el transcurso de los ciclos se fueron minimizando los picos de producción. Un elemento clave en este logro, fue la reducción del tamaño de los encargos de trabajos (tamaño del lote), lo que permitió reducir el tiempo medio de entrega evitando así las situaciones de los ciclos con salida estimada de 97, 70 y 66 que son precedidos por ciclos valles. En realidad, en estos ciclos se liberaba trabajo retenido demasiado tiempo en el proceso, de ahí que el rendimiento de salida no es el asociado a dicho periodo sino al desarrollado en ciclos anteriores.

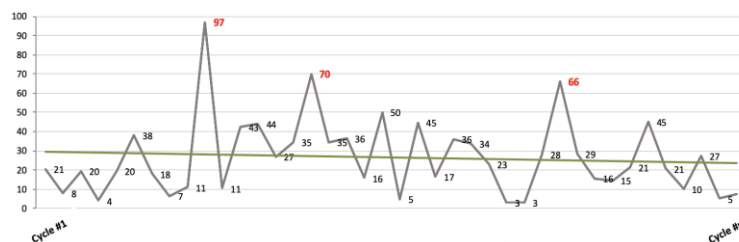


Figura F-3.11. Salida en función de la estimación relativa

El rendimiento medio de salida para el periodo que duró nuestro caso de estudio se situó en 27 puntos de historia lo que permite predecir, el tiempo de entrega de un determinado encargo de trabajo si se inicia de inmediato. Si el rendimiento medio es de 27 puntos, un trabajo estimado en 40 puntos de historia será imposible de entregar en una semana ni tan siquiera clasificándolo bajo la clase de servicio Urgente y asignando a todo el equipo. La posibilidad de ofrecer esta información a la dirección facilitó la búsqueda de alternativas en lugar de estresar a un equipo con mínimas posibilidades de éxito.

PCE.4 ¿Se ha mejorado la satisfacción del equipo?

Respuesta: **Parcialmente.**

Ciertamente la medición del nivel de satisfacción de un individuo es una tarea que entraña gran dificultad y por lo tanto, intentar asignar un valor al nivel de satisfacción de un equipo se convierte en una labor si cabe más compleja. Aunque varios de los objetivos alcanzados por el método (mejora de la predictibilidad y de la calidad del producto y del proceso) repercuten positivamente en el descenso de estrés de los equipos, desde nuestro punto de vista existen determinados factores emocionales sobre los que el método no actúa y que tienen un importante efecto en la motivación y satisfacción del equipo.

De este modo, la segunda parte de este caso de estudio analizado en detalle en el capítulo 4 consiste en la aplicación de técnicas de coaching e inteligencia emocional grupal que permiten reforzar la consecución de este objetivo.

PCE.5 ¿Se han podido crear espacios para la innovación dentro del equipo?

Respuesta: **Sí.**

La definición de políticas de asignación, la limitación del trabajo en curso y la gestión de los cuellos de botella, permitieron la generación de *tiempo libre* en el equipo. La propuesta del método Lean Kanban, llevada a cabo exitosamente en nuestro caso de estudio, es una gestión adecuada de este *tiempo libre*, de modo que repercuta en el beneficio del equipo, de sus integrantes a nivel individual y de la organización a la que prestan servicio. En la sección 3.2.6 *Implementar bucles de retroalimentación y mejorar colaborativamente*, se describe el radiador de información utilizado para permitir que, de una forma auto-organizada, el equipo sea capaz

de compaginar estos intereses de modo que todos se vean beneficiados por la innovación generada.

Si bien es cierto que inicialmente se encontró dificultad para que los miembros del equipo asumieran tareas de innovación, fuera de los encargos de trabajo registrados en el tablero, con el transcurso de los ciclos sí fueron capaces de incorporar esta dinámica en las situaciones de bloqueo habilitando espacios de innovación que repercutieron positivamente en el equipo.

PCE.6 ¿Se ha conseguido simplificar las tareas de priorización de los trabajos que debe realizar el equipo?

Respuesta: **Sí.**

La visualización del trabajo (en curso, bloqueado y pendiente), las políticas establecidas en cuanto a los límites del trabajo en curso y la clasificación de los distintos tipos de trabajo permitió, utilizando los indicadores de rendimiento del equipo, priorizar los trabajos de forma prácticamente automática. Además, gracias a la transparencia y visibilidad impulsada por el método y a que los objetivos y políticas se habían definidos previamente y eran conocidas por todos los miembros del equipo, esta priorización se pudo incluso delegar al propio equipo. De este modo se resolvió uno de los principales cuellos de botella existentes en los equipos de mantenimiento de software que siguen un modelo de *Command and Control*: la priorización de los trabajos por parte del jefe de proyecto.

En este tipo de equipos, el jefe de proyecto es consciente de ser un cuello de botella, y para evitar que ningún miembro del equipo se quede sin trabajo ante una indisponibilidad suya, acostumbra a priorizar más trabajo del estrictamente necesario. Esto impide que dicha priorización se ajuste exactamente a las necesidades reales del equipo en el momento de iniciar el trabajo priorizado.

Otro aspecto importante, que facilitó la priorización del trabajo, es la reducción de las peticiones en espera. Como se puede comprender, priorizar un *backlog* de 30 encargos de trabajos pendientes requiere mayor esfuerzo que un *backlog* de 10.

El equipo de nuestro caso de estudio disponía de un sistema de recepción de solicitudes atendido por personal dedicado exclusivamente a recibir, distribuir y contestar dichas peticiones. Toda solicitud realizada por un usuario era registrada en dicho sistema y la única forma de dar de baja una petición era solucionarla o descartarla

de forma motivada. Hasta la aplicación del método Lean Kanban el número de peticiones activas se concentraban en el intervalo [20-30].

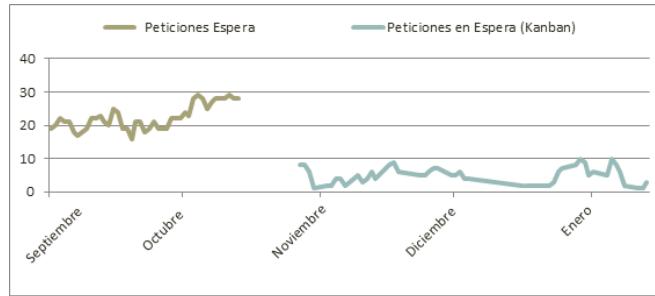


Figura F-3.12. Comparación peticiones en espera

Transcurridos los seis primeros meses desde la incorporación del método Lean Kanban se llegó a alcanzar una tasa inferior a 10 peticiones en espera durante varios días consecutivos. En ese momento, se estableció el límite superior de control a 10 peticiones en espera. Como se puede observar en la figura F-3.12, este compromiso se mantuvo por debajo del límite de control sostenidamente durante meses.

Inicialmente, se atribuyó la mejora de este indicador a un descenso significativo en el número de peticiones. Para descartar esta hipótesis se analizó un histograma con los datos de las peticiones recibidas distribuidas por meses (figura F-3.13). Exceptuando los periodos vacacionales, se puede observar que el flujo de peticiones era razonablemente estable y, de hecho, se había conseguido la mejora del indicador en noviembre, uno de los meses con mayor número de peticiones.

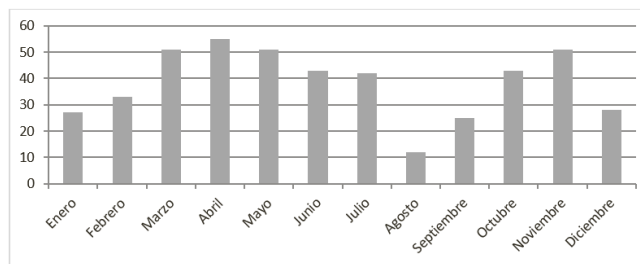


Figura F-3.13. Distribución recepción de Peticiones

PCE.7 ¿Se ha incrementado la transparencia sobre el proceso y su funcionamiento?

Respuesta: **Sí.**

Es importante que el equipo tenga en todo momento presente las métricas a las que debe prestar atención, recordándole, de este modo, la orientación de la iniciativa de mejora. Desde nuestro punto de vista, es una buena práctica que el equipo pueda visualizar fácilmente la tendencia de estos indicadores. En nuestro caso de estudio, optamos por visualizar en la parte superior del tablero los tiempos del Lead-Time y Cycle-Time conseguidos en los 3 últimos ciclos, así como la media acumulada (ver figura F-3.14).



Figura F-3.14. Tendencia del Lead-Time y Cycle-Time

El objetivo inicial de la visualización de los tiempos, era mostrar al equipo permanentemente los valores de rendimiento obtenidos. De este modo, eran conscientes de que las acciones diarias sobre el tablero, tenían un efecto en el corto plazo (siguiente ciclo). Dado que el arranque de la iniciativa supuso introducir todos los tickets en el tablero en el mismo día, se requería que fueran pasando varios ciclos hasta que los indicadores encontraran su nivel natural, a partir de ese momento, se comenzaron a introducir iniciativas de mejora.

A la vez que se visualizan los tiempos del proceso se deben tener en cuenta las clases de servicio de los tickets finalizados durante el ciclo. Dado que el equipo seleccionaba el trabajo a realizar, era importante evaluar qué tipo de trabajo se realizaba habitualmente. Obviamente los trabajos Urgentes no son negociables, pero en el resto debe existir un equilibrio, de no ser así, se estaría poniendo un excesivo foco en el valor entregado a los usuarios a costa de olvidarnos de calidad. El diagrama de la figura F-3.15 era otro de los radiadores de información a los que tenía acceso el equipo.

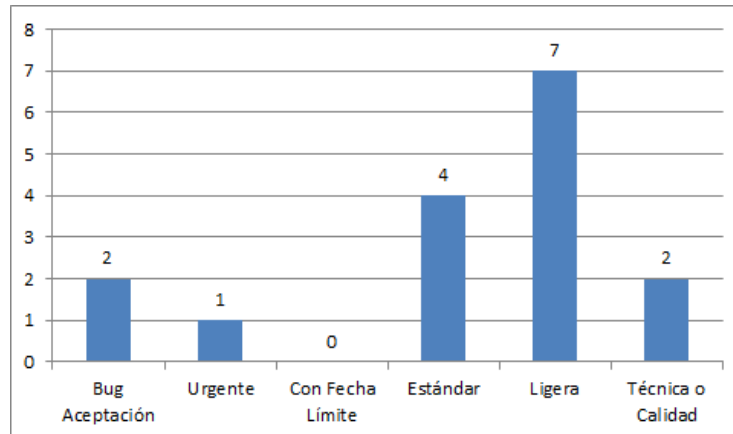


Figura F-3.15. Tickets terminados en un ciclo por Clase de Servicio

Consideramos que la implementación del método en nuestro caso de estudio ha sido correcta, por lo que, tanto el equipo de mantenimiento como el resto de agentes que se relacionaban con él (gerencia, operaciones u otros equipos de mantenimiento de software), comprendieron el resultado de sus acciones y evitaron interferir en el funcionamiento interno del equipo, mejorando de esta forma el desempeño global. La dirección llegó a comprender que, en determinadas ocasiones, añadir un trabajo extra al equipo, implicaba superar sus límites de trabajo y evitó hacerlo excepto en contadas ocasiones. Esto permitió que el equipo se mantuviera focalizado en las tareas que tenía iniciadas en cada momento. El resto de equipos que se relacionaban con nuestro equipo de mantenimiento, podían comprobar fácilmente las implicaciones que el retraso de alguna de sus tareas acarrearía en nuestro equipo al observar los bloqueos existentes. De hecho, solucionar estos conflictos se convirtió en una tarea prioritaria para el director de proyectos.

PCE.8 ¿Se ha conseguido establecer una cultura de mejora continua?

Respuesta: **Sí.**

La visualización del proceso y la constante revisión de indicadores, políticas y prácticas, unido con el poder otorgado al equipo para definir y modificar su propio proceso, hizo emerger una cultura de mejora continua. El equipo ya no sólo se preocupa de completar tareas de mantenimiento, sino que continuamente se replanteaba el proceso buscando prácticas que resolvieran los

problemas detectados o eliminando aquello que no aportaba valor al producto. Pequeños pero continuos cambios que hicieron que el proceso de desarrollo fuera cada vez más eficiente.

Ahora bien, desde nuestro punto de vista, la sostenibilidad de esta cultura de mejora continua es difícil de mantener en el tiempo. Principalmente en equipos de mantenimiento de software que trabajan en modalidad de *outsourcing* dado que encuentran dificultades para alinear sus metas personales y de equipo con la organización a la que prestan servicio. Es por este motivo que tras la finalización de esta primera fase de nuestro caso de estudio afrontamos la incorporación de técnicas de coaching e inteligencia emocional grupal con el objetivo de facilitar la sostenibilidad de esta cultura en el tiempo.

Con el fin de validar el análisis de resultados expuesto, una vez finalizado el caso de estudio, se llevó a cabo una encuesta a todo el equipo de trabajo en la que se preguntó por el grado de consecución de los objetivos. El resultado de la encuesta se muestra en la figura F-3.16. Como se puede observar, desde la perspectiva del equipo de trabajo, el grado de consecución de consecución de los objetivos fue alcanzado de forma notable.

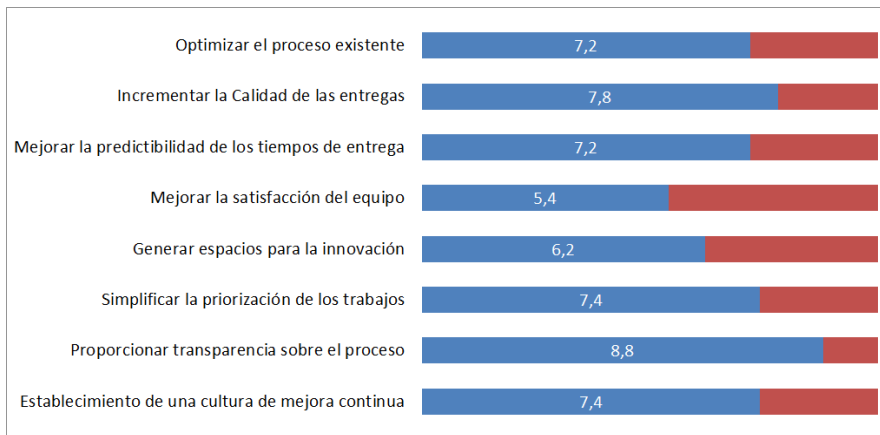


Figura F-3.16. Encuesta de consecución de los objetivos

Cabe destacar la máxima puntuación (8.8) sobre la *transparencia* del proceso de desarrollo, algo razonable teniendo en cuenta que es uno de los nueve valores del método. Por otra parte, la menor valoración corresponde al objetivo de **mejora de la satisfacción**. Aunque el método incide en determinados aspectos que pueden mejorar la satisfacción, aumentando su liderazgo, facilitando la toma de decisiones

y reduciendo el estrés, en un equipo de estas características (mantenimiento del software en modalidad de *outsourcing*), el impacto del *outsourcing* tienen implicaciones negativas en la satisfacción y en la intención de rotación del personal (Brooks et al., 2009). La valoración obtenida, proporciona una evidencia de que se deberían mejorar los esfuerzos para la mejora de la satisfacción por lo que proponemos la aplicación de técnicas de coaching e inteligencia emocional grupal.

3.4 Lecciones aprendidas

Durante la elaboración del caso de estudio pudimos observar que el **esfuerzo** requerido para la **mejora de un proceso de desarrollo** (en un grado aceptable) es muy elevado. Nuestra conclusión es que puede llegar a requerir más esfuerzo que una gestión tradicional del equipo puesto que mantener un ritmo constante de mejora requiere un cambio en la cultura de los equipos para mantenerse enfocados en esta mejora continua.

Es importante disponer de un conjunto de indicadores (sección 3.2.2 *Métricas utilizadas en el proceso de mejora*) para la mejora del proceso. Esto permite conocer su **estado de salud**, visualizando la evolución del rendimiento general del equipo. Desde nuestro punto de vista, para ajustar el proceso es imprescindible comprender y vigilar todos estos indicadores.

En nuestro caso de estudio, las **métricas** pudieron comenzar a recogerse cuando se había alcanzado un mínimo de madurez en la implantación del método. De este modo, tuvimos que esperar a la visualización del proceso, a la adquisición del hábito de trabajar con el tablero y alcanzar unos WIP que se ajustaban razonablemente al equipo. La reflexión en este sentido es que observar los indicadores sin tener en cuenta el contexto del ciclo en que se recogen los datos es poco constructivo. De ahí que optáramos por definir **periodos de ciclo muy cortos** facilitando el análisis de correlación entre las situaciones acontecidas durante el ciclo y las tendencias reflejadas por los indicadores.

Desde nuestra perspectiva, es importante que estas **retrospectivas** se realicen nada más finalizar el ciclo puesto que existen un elevado número de factores que alteran los indicadores de rendimiento, pero que tras un análisis detallado, no implican que el proceso esté fuera de control. El trabajo de las retrospectivas es, a nuestro modo de ver, de vital importancia para sostener la mejora continua en el tiempo.

Gracias al continuo análisis del proceso de desarrollo se han llevado a cabo un importante número de acciones que permiten mejorarlo sin desviar al equipo de

su misión. Nuestra reflexión es que es más importante dedicar los esfuerzos para iniciar ese motor de cambio, que definir un exhaustivo plan de mejora global. En nuestro caso de estudio, los **cambios fueron emergiendo gradualmente**, ante necesidades puestas al descubierto en las retrospectivas semanales.

A continuación, se listan las mejoras introducidas más significativas a lo largo del caso de estudio. Se puede observar que son un mix de acciones técnicas, organizativas y enfocadas a la calidad que comparten el mismo objetivo: corregir errores cometidos, reducir las ineficiencias y optimizar el proceso de desarrollo.

- **Medidas Organizativas**

- Priorizar el trabajo atendiendo a los ANS definidos en las diferentes clases de servicio.
- Llevar a cabo reuniones de retrospectivas semanales.
- Incrementar la auto-organización del equipo mediante la eliminación gradual de la columna de selección del trabajo dentro del tablero, impulsando a los miembros del equipo a seleccionar ellos mismos los encargos de trabajo atendiendo a las políticas definidas.
- Compartir código y responsabilidad. Como se comentó en la sección 3.2.5 *Gestionar el flujo*, el hecho de evitar proyectos *silos* facilita la gestión del flujo de trabajo, uno de los nueve valores del método (Burrows, 2014).
- Compartir información: para alcanzar el punto anterior es imprescindible que el grado de transparencia dentro del equipo sea elevado, el método Lean Kanban facilita este aspecto a través de la visualización y las reuniones diarias de seguimiento.

- **Medidas Técnicas**

- Practicar la integración continua en todos los proyectos.
- Continuous Delivery: Orientación del proceso de desarrollo y del equipo en su conjunto a dichas prácticas. Para mejorar el flujo de trabajo se dedicaron esfuerzos extra en la configuración y automatización de los entornos para el despliegue automatizado de las aplicaciones. Se estableció la obligatoriedad de desplegar desde las máquinas de compilación remota (evitar llevarlo a cabo desde el entorno de desarrollo local).
- Estimación relativa de todos los encargos de trabajo.

- **Medidas orientadas a la mejora de la Calidad**

- Revisiones técnicas formales de partes de código críticas.
- Automatización de pruebas.
- Análisis estático de código.
- Métricas de código (Complejidad ciclomática, líneas de código x método, etc.). El mantener una base de código heterogénea y de diferente

procedencia se hizo imposible igualar las métricas en todos los proyectos. Para ello se decidió automatizar estos cálculos, permitiendo comparar fácilmente el impacto de un determinado cambio en la calidad del producto software. Se estableció la política de que el *tester* debiera evaluar estos indicadores e informar al desarrollador si la tendencia era negativa.

- Pruebas (*testing*) cruzadas de los trabajos. Cualquier cambio desplegado en producción debía ser probado por un ingeniero que no participe en la implementación.

El grado de consecución de estas prácticas fue diferente a lo largo del caso de estudio. De hecho, algunas de ellas implican días de trabajo para ponerlas en marcha, mientras que otras requieren de meses hasta su completa adopción. Por otra parte, no todas las iniciativas funcionaron adecuadamente. Se llegó a abandonar, por ejemplo, un intento de programación por parejas porque los miembros del equipo no se encontraban cómodos con esta técnica. Nuestra reflexión en este sentido es que, con la inclusión de todos estos cambios, no sólo se mejoraron los indicadores de rendimiento, sino que, a su vez, se mejoró sensiblemente el proceso de desarrollo y la calidad interna de los productos desarrollados.

La hipótesis inicial en el presente caso de estudio fue que *es posible mejorar la productividad de los equipos de mantenimiento de software en modalidad de outsourcing mediante la aplicación del método Lean Kanban y facilitar la sostenibilidad de la cultura de mejora continua, impulsada por la adopción del método, mediante la aplicación de técnicas de coaching e inteligencia emocional grupal*. En cuanto a la primera parte de nuestra hipótesis, desde nuestro punto de vista, la mejora substancial de la productividad debe provenir de cambios que van más allá de la aplicación de buenas prácticas de ingeniería que, si bien importantes, por sí solas no podrían ofrecer una mejora de tal envergadura.

La reflexión en este sentido es que el método Lean Kanban es una buena aproximación dado que, si bien tiene en cuenta la ingeniería utilizada durante la construcción del producto, con especial énfasis en la mejora continua de la calidad, **permite focalizarse en medidas concretas del proceso** como los tiempos medios de entrega, la reducción de la variabilidad, el ciclo de mejora continua global, así como la reducción drástica del trabajo en curso. Todo ello facilita la incorporación de medidas, tanto técnicas como organizativas, que permiten la transformación monitorizada del proceso, muy orientada a la mejora de la productividad del equipo, dado que los indicadores seleccionados (sección 3.2.2 *Métricas utilizadas en el proceso de mejora*) están orientados a potenciar la productividad.

Del análisis de los indicadores podemos concluir que la aplicación de Lean Kanban a este tipo de entornos resulta altamente recomendable. Los tiempos de entrega se redujeron drásticamente por lo que aumentó la frecuencia en que se entregaba valor al cliente, lo que, a su vez, repercutió en la reducción del ciclo de *feedback* sobre los productos entregados, facilitando su evolución. El tiempo de ciclo, medido como tiempo de ingeniería de los trabajos realizados, igualmente decreció significativamente gracias a la reducción del WIP del equipo y a la reducción del tamaño del lote.

La tendencia positiva de los indicadores, ofreció un espacio para la mejora continua que pudo ser aprovechado por el equipo para mejorar el proceso en los niveles comentados previamente: acciones organizativas, técnicas y orientadas a la mejora de la calidad del producto.

Aun así, una de las principales problemáticas que se presenta en toda iniciativa Lean Kanban es la inercia del proceso de mejora, que pasado un tiempo puede entrar en un punto muerto. En la Teoría de las Restricciones (Goldratt & Cox, 1984), el quinto paso de su proceso es evitar dicha inercia, de modo que la mejora deje de tener un efecto real en la organización. La segunda fase de nuestro caso de estudio se introduce en esta línea de investigación. La aplicación de técnicas de coaching e inteligencia emocional grupal, permitió a los miembros del equipo definir y mantener unos objetivos comunes que evitaran caer en dicha inercia, facilitando, de este modo, sostener los efectos positivos de la mejora continua con el paso del tiempo. En el siguiente capítulo presentamos la aplicación de dicho proceso y nuestras reflexiones al respecto.

El proceso ETeC

En el presente capítulo se expone la ejecución de la segunda fase del caso de estudio orientada a la validación de la hipótesis formulada en esta Tesis Doctoral: “es posible mejorar la productividad de los equipos de mantenimiento de software en modalidad de outsourcing mediante la aplicación del método Lean Kanban y **facilitar la sostenibilidad de la cultura de mejora continua, impulsada por la adopción del método, mediante la aplicación de técnicas de coaching e inteligencia emocional grupal**”.

Tras la exitosa transformación del equipo mediante la aplicación del método Lean Kanban y una vez constatado el bajo nivel de motivación y satisfacción de sus integrantes, como se desprende de la encuesta de consecución de objetivos presentada en la sección 3.3 *Análisis de resultados*, nuestra propuesta es complementar la aplicación del método con el despliegue de técnicas de coaching. Para ello se definió nuestro propio proceso que se analiza en el presente capítulo.

4.1 Protocolo de recogida de datos

El protocolo de recogida de datos, en el contexto de la validación de la segunda parte de la hipótesis de la presente Tesis Doctoral, consiste en la definición de una serie de preguntas que deben ser respondidas después de llevar a cabo la ejecución del caso de estudio.

En la primera parte de nuestro caso de estudio, la validación tuvo por objetivo comprobar la mejora de la productividad del equipo de mantenimiento mediante la transformación de sus métodos de trabajo. En esta segunda fase, a través de nuestro caso de estudio, pretendemos validar la **sostenibilidad de la mejora continua mediante la aplicación de técnicas de coaching e inteligencia emocional grupal**.

Como se expone en (Dabhilkar & Bengtsson, 2004) nuestra premisa conceptual reivindica una relación entre la motivación del equipo de trabajo y la mejora continua. Por ello, hemos optado por desarrollar más este aspecto con respecto a la sostenibilidad de la cultura de mejora continua. Consideramos que no es suficiente desplegar políticas de mejora continua alineadas con los planes estratégicos generales de las organizaciones, sino que también se requieren trabajadores motivados para impulsar los esfuerzos requeridos por estas políticas. En la teoría del rediseño del trabajo (Hackman & Oldham, 1980), se especifica que la motivación para mejorar continuamente los procesos a nivel operativo, se

fortalece cuando los trabajadores tienen una autonomía considerable para determinar los medios por los cuales se realiza el trabajo y cuando reciben retroalimentación sobre los resultados de su trabajo.

Atendiendo al estudio empírico (Suarez & Miguel, 2009) en 49 industrias diferentes, entre las principales causas del abandono de la cultura de la mejora continua se encuentra, según el 50% de los encuestados, en la **falta de motivación** de los empleados para participar. Una revisión de la literatura (García et al., 2014) encontró que hay muchos elementos que contribuyen a la implementación exitosa de la mejora continua. La tabla T-4.1 muestra un resumen de los principales Factores Críticos de Éxito (FCE) relacionados con la implementación de la cultura de mejora continua: aquellos que han sido citados en más de 5 de las referencias bibliográficas analizadas en el estudio.

Factor Crítico de Éxito	Nº de veces citado
Compromiso y motivación del personal	10
Apoyo de la alta dirección	7
Recursos asignados (tiempo, económicos, espacios)	7
Liderazgo	6
Desarrollar una cultura de mejora continua	5
Establecer metas para programas de mejora	5

Tabla T-4.1. FCE principales en la implementación de la Mejora Continua

Adaptada de (García et al., 2014)

Consideramos importante reseñar que de los seis factores más citados se encuentra el **compromiso y motivación** del equipo de trabajo, así como el establecimiento de **metas** para la mejora continua.

A continuación, se presentan una serie de preguntas del caso de estudio (PCE) alineadas con la **sostenibilidad de la mejora continua**, a las que se dará respuesta en la sección 4.3 *Análisis de resultados*.

- PCE.1 ¿Ha mejorado la motivación del equipo de trabajo?
- PCE.2 ¿Se han conseguido definir e implementar adecuadamente una meta común del equipo de trabajo?
- PCE.3 ¿Los resultados obtenidos sugieren un retorno de la inversión adecuado para la aplicación del proceso ETeC?

En cuanto a la recolección de datos que permitirá obtener evidencias con las que dar respuesta a las anteriores preguntas, se ha decidido utilizar las siguientes fuentes de información:

- Resultados de las dinámicas adoptadas durante la ejecución del caso de estudio.
- La observación directa del equipo en su entorno de trabajo.
- Entrevistas con los miembros del equipo.
- Retroalimentación recibida durante la fase de intervención.

4.2 Ejecución del caso de estudio

Como se explicó en la sección *1.4.2 Definición del caso de estudio*, el equipo seleccionado requería de una aceptación voluntaria del proceso de coaching. Dado que nuestro caso de estudio se compone de dos fases secuenciales, y que la primera de ellas (adaptación y aplicación del método Lean Kanban) requiere un esfuerzo importante, llegado el momento de iniciar la segunda fase de estudio nos debíamos asegurar que todos los integrantes seguían interesados en la iniciativa. Es por esto que, antes de la incorporación de las *coaches*, se volvió a evaluar su interés en el proceso.

Una vez garantizado el compromiso del equipo con el proceso de coaching, se procedió a vigilar los otros FCE identificados en la tabla T-4.1 que, si bien habían sido alcanzados, consideramos importante su fortalecimiento antes del inicio del proceso.

De este modo, quisimos evidenciar la importancia del **apoyo de la dirección** al proceso asegurándonos su presencia durante la primera fase de diagnóstico. Para ello, dos responsables de la organización cliente asistieron a la primera sesión con el compromiso de que se retirarían en las siguientes sesiones, sobre todos en la fase de intervención para no coaccionar al equipo.

Otro FCE a tener en cuenta es el **Liderazgo**. En nuestro caso de estudio, el liderazgo técnico asociado a la adopción eficiente de métodos y prácticas de trabajo (Roth, 2006), no era discutible puesto que el proceso ETeC se iniciaba tras una exitosa transformación Lean del equipo, conducida por el director de proyectos.

El último FCE analizado es el **desarrollo de una cultura de mejora continua**. Como se ha comentado con anterioridad, en el momento de iniciar la segunda fase del caso de estudio, este factor había sido logrado exitosamente si bien es cierto que, desde nuestro punto de vista, estaba comenzando a verse

comprometido. Una segunda encuesta al equipo sobre la consecución del objetivo de mejora continua ratificó esta percepción, al obtener un valor de 6,2 frente al 7,4 alcanzado tras la finalización de la primera fase. Esta valoración del equipo vino a confirmar la percepción del director de proyecto en relación con la participación del equipo en las reuniones de retrospectiva de ese periodo (espacio de tiempo transcurrido desde la finalización de la primera fase hasta el momento de inicio de la segunda).

A este respecto, existen evidencias de que los equipos entran en algún tipo de rutina en las retrospectivas después de cierto tiempo (McHugh et al., 2011), lo que les impide alcanzar el nivel deseado de reflexión durante estas sesiones. Esto conlleva una reducción de la mejora continua (Andriyani et al., 2017) o “algunos fenómenos varios que podrían afectar al resultado y eficiencia de las retrospectivas a nivel de equipo” (Lehtinen et al., 2017). Este parecía ser el escenario de nuestra investigación. El equipo se estancó en una especie de inercia negativa después de que la transformación se consolidó y estaba obstaculizando la cultura de mejora continua que trajo la adopción del método Lean Kanban. Desde nuestra perspectiva, pudo tener que ver con que las retrospectivas se enfocan principalmente en analizar problemas técnicos y metas a corto plazo, o tal vez el ambiente del equipo no favorecía una comunicación sincera.

Como se comentó en la sección 2.3 *Coaching e inteligencia emocional grupal*, en la segunda mitad del siglo XX, con el establecimiento de la sociedad del conocimiento, la mejora de la productividad no se debe tanto a los procesos (como ocurrió en la época industrial) como a la eficiencia de las personas. La industria del software es un claro ejemplo de productividad ligada a la eficiencia de los equipos de mantenimiento de software.

Por aquel entonces Toyota, una pequeña empresa de fabricación de automóviles, basaba su supervivencia frente a los gigantes del sector en una serie de principios entre los que la gestión de personas y el equipo era clave. Cincuenta años después, aquel pequeño fabricante japonés se convirtió en el mayor fabricante a nivel mundial y revolucionó la forma de trabajo de la industria (Womack et al., 1990). Los principios de Toyota se aplican en la actualidad en diferentes sectores; Lean Software Development (Poppendieck & Poppendieck, 2007) o el método Lean Kanban (Anderson & Dumitriu, 2005) son ejemplos exitosos de su aplicación en el desarrollo de software.

Sin embargo, existe una diferencia significativa entre la relación laboral de los empleados de Toyota y la de los empleados del sector de tecnologías de la

información cuando se trabaja con un modelo de prestación de servicios de *outsourcing*. La aplicación del método Lean Kanban en proyectos de *outsourcing* plantea limitaciones derivadas de las particularidades de este tipo de equipos debido a que la visión de la empresa cliente no está alineada con la de la consultora TI (proveedora del servicio/equipo). Por ello, la misión de los equipos de desarrollo en *outsourcing* habitualmente no encuentra reflejo en ninguna de las dos partes. Desde nuestro punto de vista, esta falta de misión redundante en una baja motivación y cohesión del equipo.

Una empresa que priorice a sus empleados invertirá recursos en la creación de un ambiente óptimo para el equipo. Sin embargo, es muy habitual que en una relación de *outsourcing* la empresa proveedora del servicio no perciba como propia esta necesidad. Durante nuestra investigación, pudimos constatar este problema al implantar el método Lean Kanban en la primera fase del caso de estudio. La falta de una misión clara y de metas comunes provocó un estancamiento en el proceso de sostenibilidad de la mejora continua.

Para solventar esta dificultad consideramos que el coaching emocional grupal e inteligencia emocional son herramientas especialmente potentes para la intervención en equipos, por lo que diseñamos un proceso que combinara ambas técnicas; nos referiremos a él como **Emotional Team Coaching process (ETeC)** (ver figura F-4.1). Por una parte, el coaching de equipos permite actuar desde fuera, a modo de espejo para el equipo. Por otra parte, trabajar sobre la emocionalidad grupal permite transformar las emociones negativas en positivas, empoderando al equipo y enfocándolo hacia una visión y meta común. Gracias a la aplicación del proceso ETeC, se consiguieron establecer objetivos comunes que generaron una mayor cohesión y motivación en los integrantes del equipo.

El proceso de coaching (ver figura F-4.1) dentro de nuestro caso de estudio tuvo una **duración de 6 meses**. Cuenta con **dos fases** claramente diferenciadas: diagnóstico e intervención. Para su realización se llevaron a **cabo cinco sesiones**, con una periodicidad aproximada de un mes, en las oficinas de la organización cliente. Las sesiones se llevaron a cabo en una sala de reuniones tranquila y alejada de la zona habitual de trabajo.

El proceso fue conducido por dos *coaches*, ambas expertas en inteligencia emocional, con certificación en coaching personal y ejecutivo y coaching de equipos. Una de ellas psicóloga con amplia experiencia en formación y RRHH, y la otra ingeniera de software, contando con experiencia en investigación, consultoría y formación.

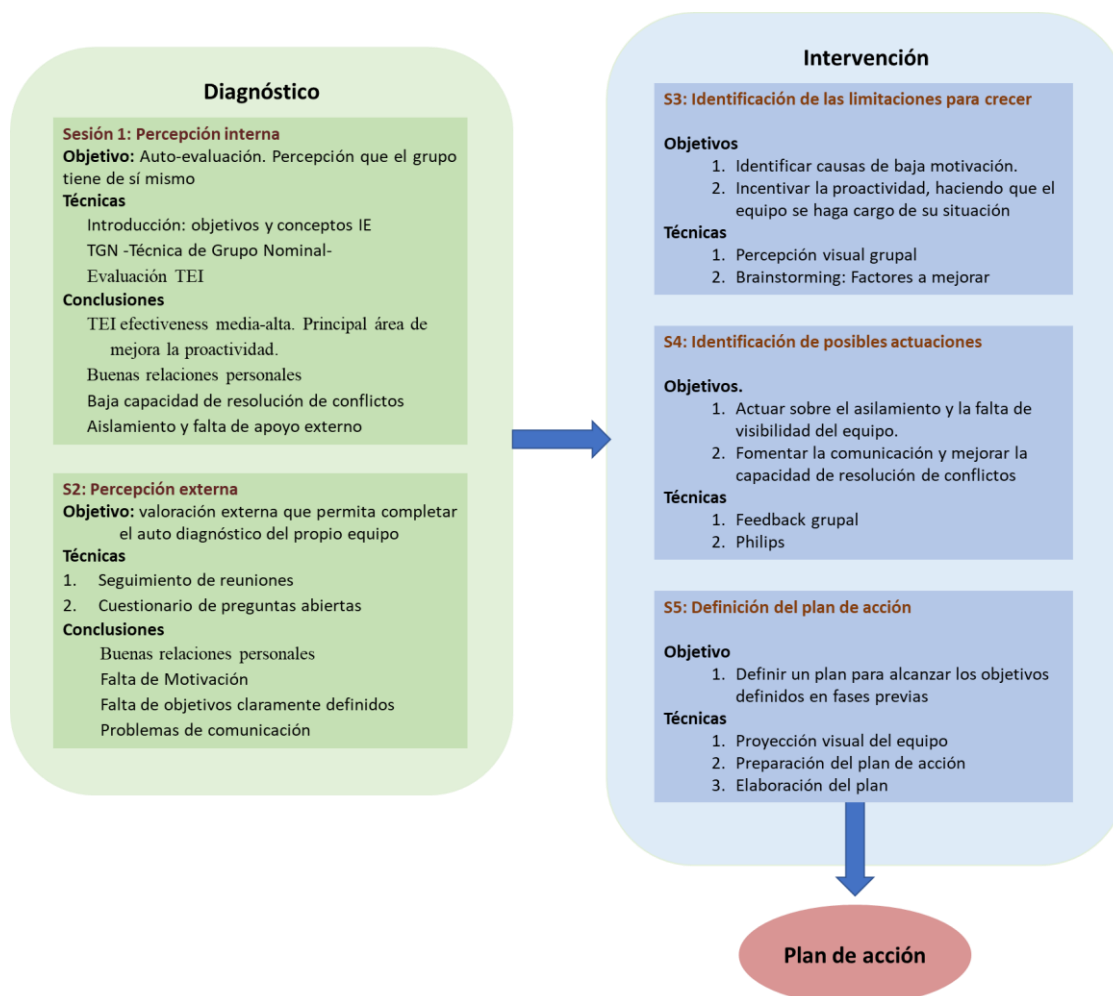


Figura F-4.1. Distribución de sesiones del proceso ETeC

4.2.1 Fase de Diagnóstico

La fase de diagnóstico tiene como objetivo realizar una **evaluación inicial** del equipo que permita a las *coaches* identificar puntos de mejora. El diagnóstico se hace en dos sesiones ya que se tienen en cuenta dos puntos de vista: interno, o percepción que el propio equipo tiene de sí mismo y externo, mediante la observación del funcionamiento en reuniones y valoración por parte de las *coaches*.

S1: PERCEPCIÓN INTERNA

El objetivo de la primera sesión es realizar un diagnóstico inicial teniendo en cuenta la propia **perspectiva del equipo**. Los resultados de esta sesión, y la siguiente, permitieron a las coaches identificar puntos de mejora del equipo y organizar el resto de la intervención.

La sesión consta de una **primera parte introductoria**, en la que se explica la dinámica de la intervención, y una segunda parte de diagnóstico propiamente dicho. En la primera parte, se introduce también los conceptos de inteligencia emocional y coaching grupal, haciendo especial énfasis en su importancia en los equipos. Se pretende con ello motivar a los miembros del equipo para que comprendan la importancia del proceso y de lo que les puede aportar. Buscamos su implicación de manera activa, ya que su participación es esencial para que el proceso pueda dar mejores resultados.

En la segunda parte de la sesión se aplican **2 técnicas de diagnóstico**: una adaptación de la TGN (Delbecq & Van de Ven, 1971) o técnica de grupo nominal y una encuesta de evaluación de Inteligencia Emocional Grupal (Wolff, 2017).

Adaptación de TGN para diagnóstico grupal

La TGN es una técnica grupal de creatividad que facilita la generación de ideas y la toma de decisiones. Se realiza en seis pasos: planteamiento del objetivo, generación de ideas y su escritura individual por cada miembro del equipo, puesta en común de las ideas, coloquio, votación y debate sobre el resultado.

En este caso, se adaptó esta técnica con el fin de conocer la percepción de los propios integrantes del equipo sobre el entorno de trabajo dentro y fuera del equipo. Los seis pasos de la TGN se realizaron del siguiente modo:

1. Planteamiento del objetivo: obtener la idea de equipo que tienen los mismos integrantes: analizar coincidencias y discrepancias.
2. Generación de ideas: cada integrante del equipo, de modo individual debía pintar un dibujo en un folio tamaño DinA4. Se les ofreció diferente material de pintura y libertad en cuanto al formato del dibujo. Aunque la temática del dibujo era la misma para todos –representar el equipo de trabajo y el entorno con el que se relacionaba– ellos no conocían la temática de los compañeros. A ellos se les indica que el objetivo es analizar el comportamiento en distintos tipos de equipos y que cada uno dibujará el equipo que le toque.

4. Concluidos los dibujos se requirió que rotaran los dibujos comenzando por la persona sentada a la derecha. Cada uno escribe en la parte de atrás del dibujo una palabra con la primera idea que les sugiere el dibujo al observarlo. Esta dinámica finalizaba cuando el dibujo original le llegaba de vuelta a cada autor.
5. Una vez que cada dibujo llega a su autor, éste debe seleccionar las 3 ideas clave (las que más se repiten o las más representativas). Las escribe en una tarjeta y son pegadas en la pared.
6. A partir de aquí se comienza un debate y, con la dirección de las coaches, se van agrupando por similitudes (figura F-4.2).



Figura F-4.2. Percepción del entorno de trabajo percibida por el equipo.

7. Por último, se llevó a cabo un proceso de abstracción con el fin identificar las principales percepciones del equipo. La figura F-4.3 muestra la clasificación obtenida, identificando a los aspectos positivos y negativos del equipo, así como el número de miembros del grupo que identifica esta característica.

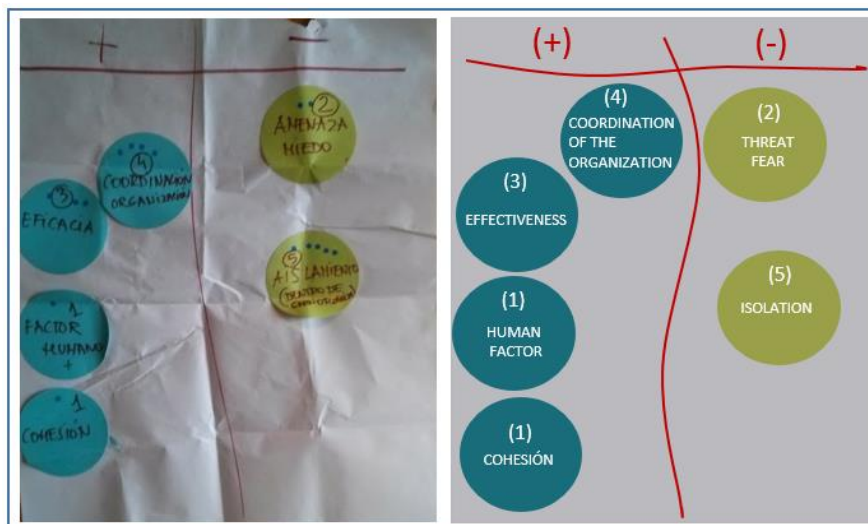


Figura F-4.3. Abstracción de la percepción del entorno de trabajo

De los resultados obtenidos, se pudo concluir que nos encontrábamos ante un equipo que se percibe bien organizado y con una elevada eficacia, con una valoración positiva de las relaciones personales. Sin embargo, se sienten aislados y amenazados. La apreciación de las coaches es que dicho aislamiento se debe a dos factores: relación con el cliente y relación con la consultora tecnológica.

Team Emotional Intelligent Effectiveness (TEI)

La evaluación de la Inteligencia Emocional Grupal (TEI) se llevó a cabo mediante un cuestionario de 61 preguntas, basado en el modelo de Druskat y Wolf (Druskat & Wolff, 2001; Elfenbein et al., 2006). El modelo propone 6 dimensiones de TEI y 9 normas que guían la interacción del equipo con sus miembros (nivel individual), el equipo en su conjunto (nivel de equipo) y otros fuera del equipo (nivel transfronterizo). En cada uno de estos niveles hay normas que crean conciencia de la emoción en el equipo y normas que regulan su comportamiento (Wolff, 2017).

La figura F-4.4 muestra los valores medios alcanzados para cada dimensión por el conjunto de miembros del equipo:

1. Conciencia grupal/ Team Awareness of Members
2. Relación grupal/ Team Management of Members

3. Autoevaluación grupal/ Team self-assesment
4. Retroalimentación y expresión de emociones/ Seeking feedback and support emotions
5. Creación de ambiente/ Creating and affirmative enviroentment
6. Proactividad en problemas/ Proactive problem solving
7. Relación intergrupala/ Team social awareness
8. Apoyo externo/ External relationships building

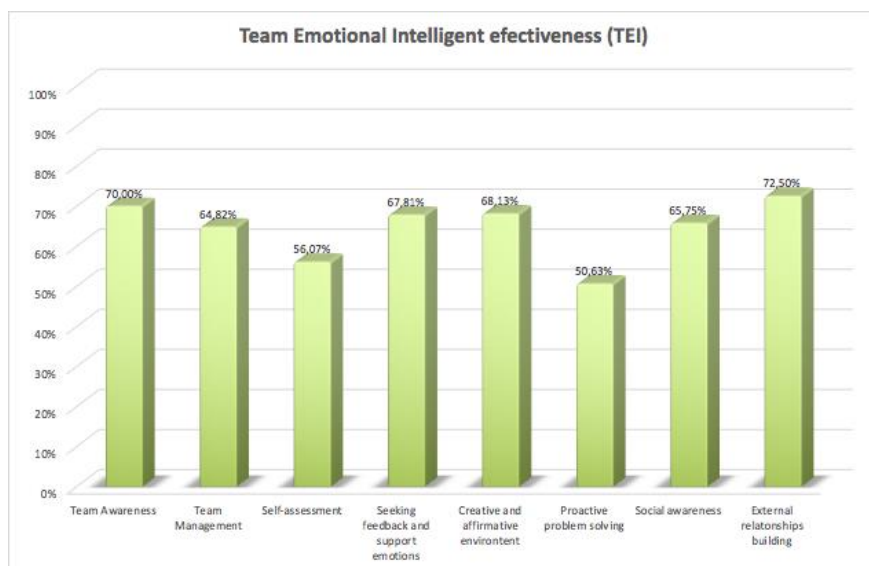


Figura F-4.4. Valoración de las dimensiones

Es importante recalcar que cada equipo tiene su estilo propio y una dinámica interna que se refleja en este tipo de cuestionarios. En este sentido, no existen respuestas mejores o peores y el resultado de este cuestionario refleja, las tendencias y conductas propias que describen a cada equipo de trabajo. No obstante, se concluye que la auto-evaluación de la inteligencia emocional del equipo se encuentra en un valor positivo. De hecho, sólo en 3 de las 61 preguntas realizadas se obtiene un valor inferior a la media (ver tabla T-4.2) y dos de estas tres preguntas están relacionadas con la interacción con otros equipos de la empresa, algo habitual en equipos de *outsourcing*.

Pregunta	Valor
En el equipo tenemos conductas consideradas como prohibidas	2,1
Para evaluar nuestro desempeño tomamos como referencia el de otros equipos de trabajo y comparamos	2,3
Conocemos y entendemos lo que les ocurre a otros equipos de esta empresa	2,4

Tabla T-4.2. Preguntas evaluadas por debajo de la media

Como **resultado** de esta sesión se observó que los miembros mantienen unas buenas relaciones y se concluye que el equipo tiene un valor medio/alto en cuanto a TEI, siendo su principal debilidad la **proactividad**, que fue apreciada baja también en las observaciones de las *coaches*. Como aspectos de mejora las *coaches* advierten baja capacidad del equipo para la **resolución de conflictos** y un cierto **aislamiento** y **falta de apoyo externo**.

S2: PERCEPCIÓN EXTERNA

Aunque en ambas sesiones hay parte de autovaloración y de análisis externo, esta sesión tiene como objetivo obtener una valoración del equipo por parte de las *coaches*, que permita completar el auto diagnóstico ya realizado por el equipo.

En esta sesión se realiza un seguimiento de reuniones, técnica habitual en el coaching de equipos (Cardón, 2007) y un cuestionario de preguntas abiertas basado en las propuestas de diagnóstico en coaching de equipos (Martínez, 2009).

Se concluye en esta sesión que el equipo no tiene claramente definidos sus objetivos a medio y largo plazo, lo que probablemente influye también en la falta de motivación. Por otra parte, también existen carencias en la organización, derivados de problemas de comunicación.

Coaching grupal: seguimiento de reuniones

Esta técnica consiste en la observación por parte de las *coaches* de una reunión del equipo. En este caso se trataba de una reunión rutinaria en la que el equipo analizaba retrospectivamente el trabajo realizado en un periodo anterior.

Las *coaches* observaron las interacciones entre los miembros del equipo durante la reunión, su lenguaje verbal y no verbal, con el objetivo de analizar las relaciones entre los miembros del equipo, así como sus procesos de toma de decisiones grupales. Otro objetivo era hacer consciente al grupo de las posibles deficiencias en su organización y en el método de trabajo.

Así pues, la intervención de las *coaches* se basa en la escucha, limitándose a hacer de espejo al equipo para mostrarles las interacciones que no les son útiles, así como otras potenciadoras. Algunas de estas intervenciones fueron encaminadas a

resaltar los estados de ánimo del equipo y sus miembros que, por actitud o lenguaje no verbal, podían influir en la dinámica general.

Al finalizar la reunión se pide a cada miembro del equipo que escriba en un papel, de manera anónima, cuál era su idea del objetivo que tenía esta reunión de trabajo. La tabla T-4.3 muestra las respuestas.

¿Podrías definir claramente el objetivo de la reunión?	
1.	Aún no lo sé.
2.	Comunicación entre el equipo / Ver la perspectiva del equipo.
3.	Para recordar compromisos.
4.	No me ha valido para mucho.
5.	Me sirve para valorar los puntos débiles y fuertes del equipo.
6.	La reunión de hoy me ha servido para retomar el pulso al grupo y retomar las conversaciones menos software.

Tabla T-4.3. Respuestas a la pregunta de objetivos de la reunión

De la observación del equipo se pudo concluir que su interacción se centraba en aspectos puramente técnicos. Así, en la reunión, el equipo llega a varios compromisos en cuanto a la organización del trabajo a nivel técnico, algo que era habitual. Sin embargo, se obvian totalmente aquellas cuestiones relacionadas con el funcionamiento como equipo: motivación, estado de ánimo, cohesión, etc.

Se aprecia, por tanto, la necesidad de **mejora en habilidades personales e interpersonales**, ya que el funcionamiento del equipo se ve más comprometido por estas cuestiones que por sus capacidades puramente técnicas. Se trata de circunstancias habituales en los equipos de desarrollo de software, que suelen ser juzgados en función de criterios basados principalmente en indicadores técnicos (Wilson, 1999). Esto conlleva a que este tipo de equipos releguen a un segundo plano todo lo relacionado con la inteligencia emocional y las habilidades personales.

Cuestionario de preguntas abiertas

Para finalizar esta sesión se pasó un cuestionario que los miembros del equipo debían contestar de manera individual y anónima. Se trata de una serie de preguntas basadas en una técnica de diagnóstico en coaching grupal propuesta por Peñalver (Martínez, 2009). La tabla T-4.4 muestra el resultado del cuestionario.

Las respuestas obtenidas, coincidiendo con las conclusiones de la primera sesión, indican que se trata de un equipo con buenas relaciones entre sus miembros, pero que a su vez tiene una clara **falta de motivación**. Esta falta de motivación puede deberse a las barreras burocráticas e institucionales (característica de equipos de *outsourcing*) y a la falta de proyectos nuevos y motivadores.

RESPUESTAS A LAS PREGUNTAS ABIERTAS

<p>Q1. ¿Cómo definirías el ambiente del equipo, y de tu área/departamento?</p> <p>Equipo Bueno (6)</p> <p>Departamento Mal / No encaja</p> <p>Baja motivación</p> <p>Compromiso con el trabajo</p> <p>Divertido</p>	
<p>Q2. ¿Qué es lo mejor de este equipo?</p> <p>Relaciones personales (3)</p> <p>Cohesión</p> <p>Diversidad Caracteres</p> <p>Ambiente</p> <p>La antigüedad</p> <p>La organización y compartir</p>	
<p>Q3. ¿Cuál es vuestro *Talón de Aquiles?</p> <p>Falta de Motivación (2)</p> <p>Aplicación metódica</p> <p>Aislamiento con otros</p>	
<p>Q4. ¿Existen barreras entre el equipo y algún otro colectivo (central, mandos intermedios, proveedores, clientes, etc.)? Si así fuera, ¿de qué tipo son?</p> <p>Sí, institucionales (3)</p> <p>No.</p> <p>Sí, burocráticas (3)</p>	
<p>Q5. ¿Qué crees que necesita este equipo?, ¿Qué sugieres, propones?</p> <p>Más motivación</p> <p>Reconocimiento</p> <p>Proyectos de Nuevos Desarrollos (2)</p> <p>Recuperar el ámbito individual</p> <p>Evitar los problemas externos y ayudarnos en el equipo para facilitar el desarrollo personal</p> <p>Cambio en el trato como "Personal Externo"</p>	

Tabla T-4.4. Respuestas a las preguntas abiertas

Conclusiones del diagnóstico

Como conclusiones más relevantes de la fase de diagnóstico podemos identificar las debilidades y fortalezas que se resumen en la tabla T-4.5.

Fortalezas	Debilidades
<ul style="list-style-type: none"> • TEI eficacia media-alta. • Buenas relaciones personales • Grupo con cohesión • Buen ambiente 	<ul style="list-style-type: none"> • Baja capacidad para la resolución de conflictos • Baja proactividad de los miembros del equipo • Aislamiento y falta de apoyo externo • Falta de motivación • Falta de definición de objetivos a medio y largo plazo

Tabla T-4.5. Conclusiones de la fase de evaluación

Un segundo análisis nos lleva a concluir que existe mucha relación entre las distintas debilidades del equipo. La falta de motivación es un factor clave en el desempeño del equipo. Esa **falta de motivación** procede de factores como el aislamiento y la carencia de una visión compartida de objetivos. Por otra parte, al ser un grupo poco proactivo y con dificultad para la resolución de problemas, están **estancados** con esta situación. Por ello, las *coaches* deciden plantear la intervención orientada a una mejora de la motivación del equipo.

4.2.2 Fase de Intervención

Esta fase se vertebra en tres sesiones. Las dos primeras tienen como objetivo abordar directamente el problema motivacional, incidiendo en el aislamiento, a la vez que se trabaja la proactividad y la comunicación y resolución de problemas. La última sesión se centra en la elaboración de un plan de trabajo para alcanzar los objetivos definidos en fases previas.

S3: IDENTIFICACIÓN DE LIMITACIONES PARA CRECER

En esta sesión las *coaches* trabajan con el equipo para buscar la principal causa de la baja motivación de sus integrantes, ya que este se aprecia como la principal limitación para el desarrollo del equipo. A la vez, se trata de que el equipo tome la responsabilidad de lo que les ocurre, haciéndose cargo de, bien aceptar la situación, o bien tomar las riendas para cambiarla. Para ello, se realizan 2 dinámicas: percepción visual grupal y otra de *brainstorming*.

Dinámica 1: Percepción visual grupal

En esta dinámica se solicita a los miembros del equipo que se coloquen espacialmente de acuerdo a tres criterios. En primer lugar, deben hacer una fila

colocándose **libremente** en ella. Posteriormente se les pide que verbalicen en voz alta el estado anímico en el que se encuentran en el equipo de trabajo. En segundo lugar, deben hacer una fila ordenada de mayor a menor **antigüedad** de cada miembro. Nuevamente se les pide que verbalicen el estado de ánimo y se aprecian diferencias, pudiendo observar el efecto que ha causado el paso del tiempo en la motivación de los individuos, y les hace conscientes de cómo ha influido en ellos.

De la dinámica de percepción visual se puede concluir que los miembros del equipo que llevaban más tiempo tenían una menor motivación que los que se habían incorporado más recientemente. Esto implica que **el equipo se siente agredido**, en un **ambiente tóxico** que les va disminuyendo la motivación y la energía para innovar y desarrollarse.

Dinámica 2: Brainstorming

Esta dinámica se planta con el objetivo de identificar qué es lo que causa esta desmotivación paulatina que sufre el equipo. Para ello, se pide al equipo que reflexione sobre cuáles son los factores que, en su opinión, les afectan más negativamente. Cada miembro del equipo escribe en diferentes post-it tres aspectos (no técnicos) del equipo que le gustaría mejorar. Los post-it se colocan en una pared visible y una vez agrupados y eliminados los duplicados, se realizó una votación, para la cual, a cada miembro del equipo se le concedieron dos *gomets* que podía libremente asignar a cada uno de los factores ambientales. La tabla T-4.6 muestra los resultados de esta votación.

Factores de mejora para aumentar la motivación	Votos
1. Mejorar la visibilidad del equipo	5
2. Eliminar burocracia	2
3. Desarrollar nuevos proyectos	1
4. Incorporar nueva tecnología	0
5. Mejorar la política del equipo con el personal externo (cliente)	0

Tabla T-4.6. Factores de mejora para aumentar la motivación

Las votaciones de los objetivos evidenciaron claramente la necesidad de poner en marcha acciones orientadas a la obtención de una mayor **visibilidad externa**. Por otra parte, como resultado del debate establecido en esta sesión se constata, casi por unanimidad, que el principal **foco de inestabilidad** y malestar en el equipo proviene de la **relación con su propia empresa** y con el **cliente** (algo que venía observándose desde la segunda sesión).

Por lo tanto, por una amplia mayoría se decidió establecer como **meta de equipo la mejora de la visibilidad** tanto en la consultora tecnológica como en el cliente.

Desde nuestro punto de vista, la falta de visibilidad tiene que ver con el contexto del equipo: el sentimiento de los miembros del equipo era que ni la consultora TI ni la organización cliente les tenían en cuenta. Esta modalidad de contratación lleva a la organización cliente a restringir al mínimo la interacción de sus trabajadores con el equipo de *outsourcing*: solo pueden interactuar con el director de proyectos, e incluso esto debe evitarse si es posible.

Por otro lado, la consultora de TI que desplaza un equipo de trabajo al cliente, no muestra un interés en sus empleados mientras el cliente siga pagando por sus servicios. Pueden pasar años desde que la empresa se pone en contacto con un trabajador subcontratado determinado, por ejemplo. Esto da como resultado el sentimiento antes mencionado: los miembros del equipo eran prácticamente invisibles para la organización cliente y la consultora tecnológica.

S4: IDENTIFICACIÓN DE POSIBLES ACTUACIONES

Esta sesión tiene como **objetivo identificar actuaciones** posibles para mejorar la falta de visibilidad del equipo, tanto con el cliente como con la consultora. A la vez, las *coaches* trabajan la proactividad y creatividad para la resolución de problemas y la comunicación para lograr que tengan una relación más fluida en la generación de ideas entre ellos, así como para mejorar las relaciones con el cliente y consultora. Para ello, se realizan dos dinámicas: Feedback grupal y Philips 3-6-3.

Dinámica 1: Feedback Grupal

Las *coaches* comienzan dando al equipo una pequeña píldora sobre cómo dar feedback: en positivo, al hacer y no al ser, algo que nos gusta y algo que mejorar. Posteriormente, se propone una rueda de feedback con la siguiente dinámica:

- Cada integrante del equipo escribe en una tarjeta para cada uno de sus compañeros, en la que debe escribir dos aspectos positivos, un aspecto a mejorar y un consejo.
- Al ser el primer feedback grupal se decide hacerlo anónimo para que se sientan más libres. Por ello, una vez terminada esta fase, cada miembro del equipo recoge sus tarjetas y las lee.
- Para finalizar se hace una rueda en la que cada uno expresa cómo se ha sentido al leer sus tarjetas.

En la puesta en común se constata que hay muchas **percepciones** que los miembros del equipo tienen entre ellos que son totalmente desconocidas. Y, fundamentalmente, en cuanto a los aspectos positivos. Así, varios miembros aseguran no ser conscientes de que sus compañeros apreciaran, por ejemplo, su generosidad ayudando a los demás, o el ambiente y clima positivo que generaban. Eran en general más conscientes de sus aspectos de mejora, por lo que no les afectaron a su autoestima o a su relación de equipo, sino que más bien, los consejos les abrían un campo de mejora. Se creó un **clima positivo y de confianza** por lo que decidieron que debían ser más claros en su comunicación y entrenar su asertividad entre ellos, lo que les permitiría también ser más asertivos en sus reclamos a la consultora y al cliente.

Dinámica 2: Philipps 6-3-5

Esta técnica (Denton & Denton, 1999) es ampliamente usada como dinámica grupal para promover la participación de todos los miembros del grupo, fomentar la proactividad, y para la generación de ideas creativas en la resolución de problemas e innovación. Por ello, las *coaches* proponen esta dinámica enfocada a generar ideas que permitan obtener al grupo una mayor visibilidad, a la vez que incentiva al equipo en la proactividad y solución creativa de problemas.

Las *coaches* proponen la siguiente situación a analizar: **¿qué podemos hacer para mejorar la visibilidad en nuestra empresa y en el cliente?** El grupo debe responder a esta pregunta usando una dinámica basada en realizar 6 rondas de ideas, generando 3 ideas en cada ronda y en un máximo de 5 minutos. Comienza la primera ronda y pasados los 5 minutos, cada integrante del equipo pasa la hoja a su compañero de la derecha y coge la hoja del de su izquierda. En la segunda ronda, y a partir de las 3 ideas escritas, debe generar otras 3 que pueden ser completamente nuevas o complementarias a las ya escritas. Esta dinámica se repite hasta concluir las 6 rondas.

De la dinámica surgieron aproximadamente 30 ideas que iban desde hacer cursos de certificación, hasta ponerse camisetas descriptivas de su situación, o entrar con tambores en la oficina. Se procedió a votar en dos rondas. En la primera cada miembro del grupo vota libremente entre una o varias. De ahí se seleccionan 8 que son las más votadas (entre 4 y 6 votos cada una, frente a 0, 1 o 2 votos del resto). A continuación, se repite la votación sólo entre estas 8 acciones, para lo que cada miembro contaba con tres puntos que podía otorgar a las ideas que considerara más conveniente. El resultado de la votación se muestra en la tabla T-4.7. Sólo se muestran las propuestas votadas y, a modo de ejemplo, 2 que obtuvieron 0 votos.

Acción	Puntuación
Uso de nuevas Metodologías /Tecnologías	6
Realización de Lean Coffee /Mesa de trabajo	4
Rotación de área dentro del cliente	2
Aumentar la empatía en todo el entorno (cliente y consultora)	1
Rotación dentro de la empresa (especializarse)	1
Proponer ascenso del responsable directo	0
Actitud positiva grupal e individual	0

Tabla T-4.7. Resultado de Philips 6-3-5

S5: DEFINICIÓN DE UN PLAN DE ACCIÓN

Esta sesión se plantea como de cierre de la intervención y debe finalizarse con un plan de acción que permita al equipo alcanzar el objetivo de aumentar su visibilidad. Para ello, las *coaches* proponen tres dinámicas: proyección visual del equipo a medio plazo, preparación del plan de acción y elaboración del mismo.

Dinámica 1: Proyección visual del equipo a medio plazo

Se utilizan técnicas proyectivas que permitan al grupo visualizar la imagen deseada del equipo. Trabajar la visión es una técnica ampliamente usada en **psicología positiva** para aumentar la **autoestima** y **motivación** en la consecución de objetivos. Se utilizan en este caso de manera grupal, incidiendo en la autoestima y comportamiento del equipo. El objetivo es construir una visión compartida e ilusionante del equipo.

Comienzan trabajando la imagen de equipo, en lo referente al aspecto de visibilidad externa, de manera individual. Para ello se solicita a cada miembro que visualice al equipo dentro de 1 año y que plasmen la imagen en papel, bien a través de un dibujo, una expresión, o cualquier otra metáfora que lo represente adecuadamente. Deben intentar contestar a las siguientes preguntas: ¿Cómo ves al equipo? ¿Cómo te sientes?

Posteriormente, todos los integrantes pasan a trabajar en equipo y elaboran la imagen conjunta del equipo dentro de un año. Igualmente, se les pide consensuar una imagen que deben plasmar en un dibujo, o en una metáfora que lo represente y responder de manera individual: ¿cómo ves ahora al equipo? ¿Cómo te sientes?

Dinámica 2: Preparación del plan de acción

Con una imagen compartida e ilusionante del futuro como equipo, se procede a establecer la hoja de ruta que les permita alcanzar esta visión. Para ello, las *coaches*, proponen una dinámica con el fin de concretar y definir **objetivos SMART**. La dinámica consiste en trabajar en equipo y responder de manera conjunta a las siguientes cuestiones:

- **Objetivo:** ¿Qué queremos hacer? ¿Qué significa exactamente esto que queremos?
- **Specific:** ¿Es específico? ¿Si lo enseñamos a una persona desconocida lo entiende?
- **Medible:** ¿Cómo vamos a medir que nos acercamos a su consecución?
- **Alcanzable:** ¿Es alcanzable? Dad razones por las que es alcanzable.
- **Relevante:** ¿Es un objetivo retador? ¿Qué nos aporta? ¿Cómo nos veremos recompensados cuando lo consigamos?
- **Time:** ¿Cuándo lo conseguiremos? ¿En qué fecha?

Dinámica 3: Elaboración del plan

Para elaborar el plan de acción se usa una técnica proyectiva de la línea del tiempo, visualizando al equipo en distintas etapas en el camino:

- ¿Cómo seremos dentro de diez meses? ¿Qué hemos tenido que lograr para ser así?
- ¿Y dentro de ocho?, ¿y dentro de seis?, ¿y dentro de cuatro?, ¿y dentro de tres?, ¿y dentro de dos?, ¿y dentro de uno?

De cada una de estas respuestas surge el **objetivo** a lograr en un mes, en dos, en tres, etc. hasta conseguir el objetivo a diez meses, que es el plazo que se fijaron. Esta técnica permite iniciar un debate a partir del cual el equipo especifica el plan de acción. Las *coaches* les piden especificar:

- Una **hoja de ruta** con acciones para conseguir el objetivo: calendario con hitos, tiempos, actores y responsables si los hubiera.
- Una **alianza** de equipo que refleje, a modo de decálogo, el modo en que van a trabajar e interactuar entre ellos. Son las normas de trabajo se dan a ellos mismos para conseguir el objetivo. Esta alianza debe de ser concreta y precisa.

El resultado del plan de acción se muestra en la figura F-4.5.

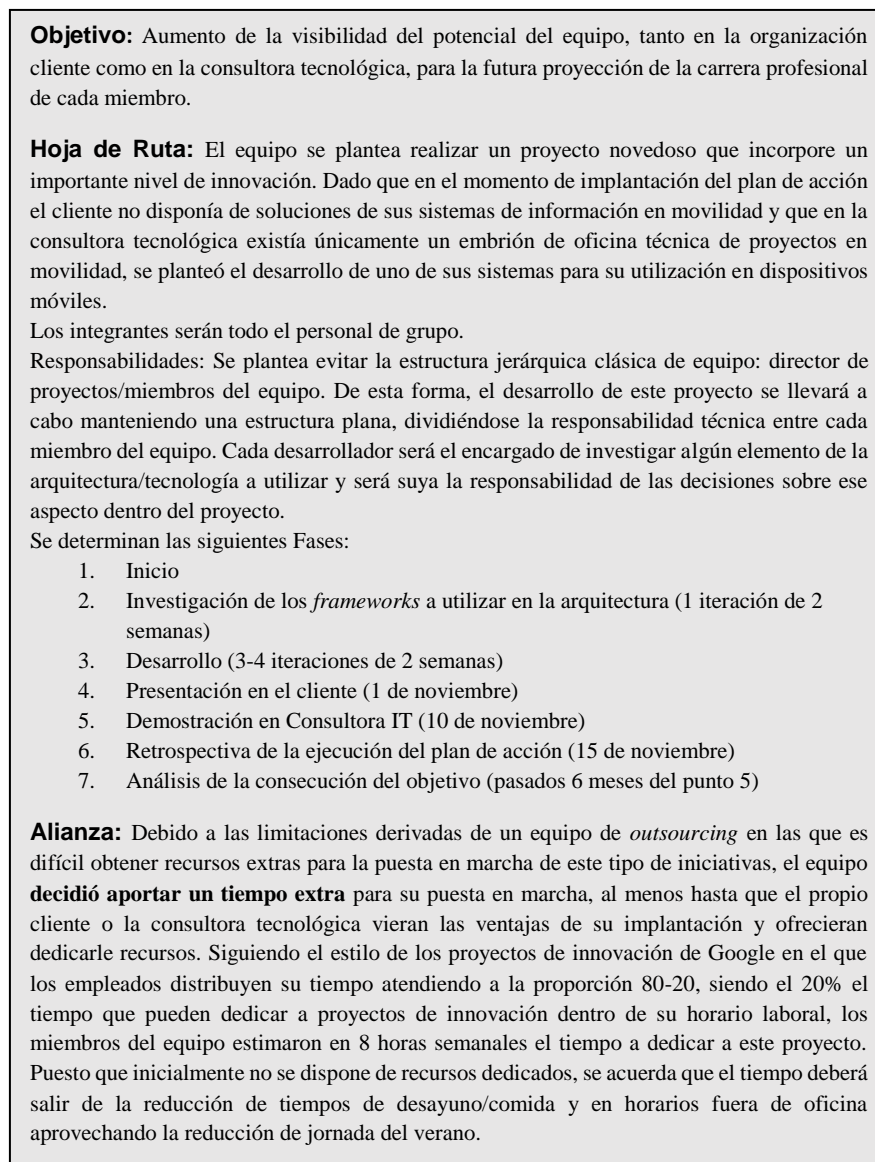


Figura F-4.5. Plan de acción

4.3 Análisis de resultados

Una vez llevada a cabo la ejecución del caso de estudio, nuestro siguiente paso será dar respuesta a las preguntas planteada en la sección 4.1 *Protocolo de recogida de datos*.

PCE.1 ¿Ha mejorado la motivación del equipo de trabajo?

Respuesta: **Sí.**

Entre las conclusiones obtenidas sorprende la variedad de formas en que evolucionó la iniciativa entre los diferentes integrantes del equipo. Transcurridos dos meses de aplicación del plan, dos de los miembros más antiguos decidieron emprender cambios profesionales que les llevaron a abandonar el equipo. La amplificación del objetivo común de aumento de la visibilidad les llevó a afrontar nuevos retos profesionales. Dejaron una posición que ocupaban desde hacía más de 8 años y que tras la aplicación de *ETeC* fueron conscientes que no les motiva lo suficiente. Otro de los miembros del equipo encontró, gracias a la mejora de su **visibilidad**, una posibilidad de cambiar de posición, asumiendo funciones de *agile coach* dentro de la oficina Lean que el mismo cliente decidió crear ante los buenos resultados del presente caso de estudio.

Mientras tanto, el resto del equipo continuó ejecutando su plan de acción, lo cual hizo que miembros con una motivación media se sintieran más motivados aumentando, con ello, la cohesión del equipo. Por otro lado, se incorporaron nuevos ingenieros del software permitiendo que un equipo estancado y con problemas de motivación se renovara con nuevo personal que, si bien no podían mejorar la productividad de los miembros seniors por su falta de experiencia en el contexto del proyecto, sí que aportaron una mejora en el clima emocional del equipo al no estar afectado por los factores de frustración del pasado.

Nuestra conclusión es que el equipo se benefició, puesto que su motivación colectiva, entendida como la media agrupada de cada integrante del equipo, había aumentado al haber sustituido, a través del proceso de coaching y de forma *no traumática*, a los integrantes más desmotivados. A nivel personal, llevaron a cabo las acciones necesarias para solucionar su situación y encontraron mejoras en sus carreras profesionales. Por otra parte, las personas que permanecieron en el equipo se beneficiaron a nivel individual al encontrarse con un

ambiente laboral menos tóxico y tener la posibilidad de trabajar en solucionar factores bloqueantes de su propia motivación.

PCE.2 ¿Se han conseguido definir e implementar adecuadamente una meta común del equipo de trabajo?

Respuesta: **Sí.**

Dentro de nuestro planteamiento, se encontraba la búsqueda de metas comunes que permitieran focalizar el esfuerzo colectivo en determinados aspectos, sobre los que se percibían la necesidad de mejora desde una perspectiva de equipo, no individual. Desde nuestro punto de vista, el elemento clave en esta cuestión es ofrecer la libertad a un equipo auto-organizado para encontrar su propia meta. Sin embargo, en la modalidad de prestación de servicios de *outsourcing* es muy habitual encontrarse con cuatro metas diferentes que, en ocasiones, pueden ser contrapuestas: la meta individual de cada integrante del equipo, la meta del equipo, la meta de la consultora tecnológica y, por último, muy específica de equipos de *outsourcing*, la meta del propio cliente. Esta fue la situación de partida al inicio del proceso ETeC, como representa la figura F-4.6. Al finalizar el proceso, están metas estaban alineadas, lo que, desde nuestro punto de vista, es un factor clave para constituir un equipo de alto rendimiento.

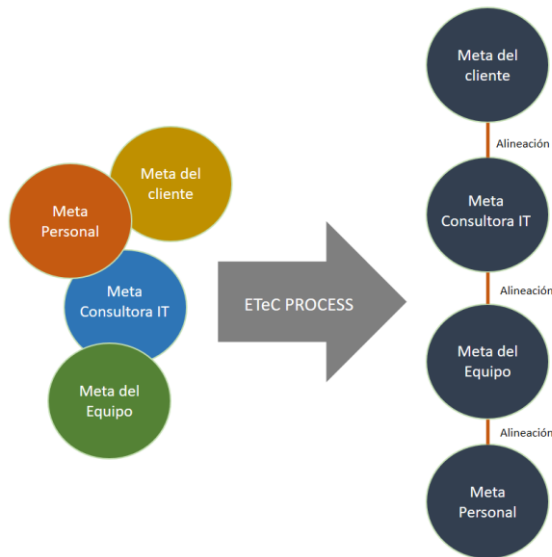


Figura F-4.6. Alineación de metas en diferentes niveles organizativos

Como norma general, en escasas ocasiones los equipos dedican tiempo en la definición de su propia meta. Nuestra reflexión es que un equipo auto-organizado sin una meta común no es más que un conjunto de personas que trabajan juntas para desarrollar un producto. Sin una meta común es difícil contar con un equipo de alto rendimiento, dado que los objetivos personales de cada integrante del equipo generarán unas fuerzas contrapuestas que desembocarán en una pérdida de la eficiencia global.

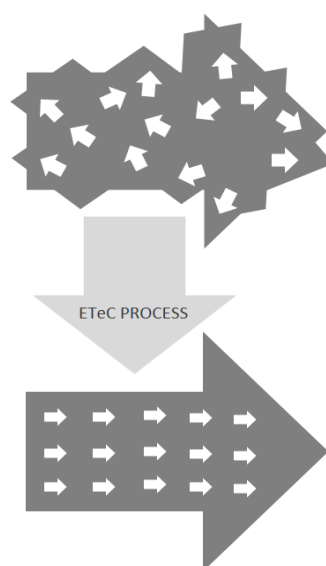


Figura F-4.7. Alineación de metas personales dentro de un equipo

PCE.3 ¿Los resultados obtenidos sugieren un retorno de la inversión adecuado para la aplicación del proceso ETeC?

Respuestas: **Sí.**

En lo referente a la variable económica del ROI, si bien es cierto que, al tratarse de un proceso de investigación la organización cliente no asumía ningún coste asociado a la contratación de las *coaches*, sí se asumió el coste de las horas dedicadas por el equipo a las sesiones presenciales. Aun así, para facilitar la aplicación práctica de ETeC, se estima el coste completo que implicaría el proceso y se detalla a continuación.

Coste asociado al *coach*:

- Adaptación de las técnicas de diagnóstico; Revisión y análisis de resultados: 2 sesiones de 8 horas.
- Diseño del programa de intervención; Selección y adaptación de técnicas para el trabajo en grupo: 1 sesión de 8 horas.
- Sesiones presenciales: 5 sesiones de 3 horas.

Coste asociado al profesional de Coaching²= 39horas x Precio/hora de su tarifa.

Coste asociado al equipo:

Se trata de una inversión que implica 15 horas en un período de 6 meses. Considerando 22 días hábiles al mes y 8 horas al día, lo que equivale a 1056 horas laborables en todo el proceso. Por tanto, durante el período en el que se ejecutó la experiencia, la intervención consumió el 0,023% del tiempo de trabajo de cada ingeniero de software.

Desde nuestro punto de vista, se trata de una **inversión rentable**. Aunque no debemos olvidar que este tipo de experiencias implican un coste importante. Sin embargo, los beneficios a medio y largo plazo lo compensan.

En cuanto a determinados aspectos más cualitativos del proceso, además de los resultados analizados anteriormente, nos gustaría poner el acento en los siguientes retornos de la inversión, directamente derivados de la intervención:

- **Ahorro de tiempo en reuniones futuras**: un mejor clima dará como resultado con seguridad que las reuniones sean más productivas. Además, algunas reuniones pueden ser directamente prescindibles o al menos acortadas.
- Efectos sobre la **productividad** y la **calidad** de un mejor clima, que ya han sido documentados en la literatura existente (Acuña, et al., 2008; Capretz, 2014; Ganesh, 2013; Sumner & Molka-Danielsen, 2010).

² Téngase en cuenta que, aunque en nuestro caso de estudio participaron dos *coaches*, se estima únicamente el coste de uno. Esto se debe a que se desarrolló en el contexto de un proyecto de investigación mientras que la aplicación práctica de este tipo de intervenciones únicamente necesitará de un *coach*.

- **Disminución** de los niveles de **absentismo**, que también se ha reconocido como un factor positivo que puede derivarse de una mejor motivación (Cohen & Bailey, 1997; Da Silva & França, 2012).

El equipo investigador estaba interesado en demostrar que la aplicación de estas técnicas a los equipos de desarrollo de software podría ofrecer resultados inmediatos, y fomentar el interés de los profesionales del software en la aplicación de estas técnicas a la ingeniería de software. Creemos que estos objetivos se han logrado. Desde un punto de vista práctico, la propia organización cliente quedó convencida de que la inversión resultaba rentable, potenciando la creación de una oficina Lean en la que actualmente trabajan 5 personas (un coste muy superior al indicando en esta sección).

Nuestra conclusión es que los resultados de la experiencia pagarán sus costos generados de su aplicación. De hecho, actualmente las grandes empresas contratan con frecuencia servicios de consultoría de coaching para ejecutivos. Igualmente, en el contexto del desarrollo de software, en los últimos años ha crecido sustancialmente el número de empresas que ha contratado servicios de *Agile Coaching* para facilitar la transición a marcos de trabajo ágiles/Lean. En nuestra humilde opinión, contemplamos un futuro en el que la contratación de consultores de coaching emocional grupal para ayudar a los equipos de desarrollo se convierta en algo habitual. Aún mejor, para aquellas organizaciones que buscan crear equipos de alto rendimiento, consideramos que estos *coaches* podrían (y deberían) convertirse en empleados internos en el medio plazo.

4.4 Lecciones aprendidas

Al finalizar la segunda fase de nuestro caso de estudio, expuesto a lo largo del presente capítulo, se pudo llegar a extraer una serie de reflexiones y conclusiones importantes de reseñar en esta sección puesto que pueden ser elementos de base a utilizar en futuras investigaciones. Consideramos que estas reflexiones y conclusiones pueden ayudar a los investigadores (en particular, la comunidad de *agile coaches*) dispuestos a aplicar técnicas de coaching e inteligencia emocional grupal en sus equipos de mantenimiento de software.

La aplicación de *ETeC* partía de un equipo ya habituado a las dinámicas de desarrollo del software a través del método Lean Kanban al haber trabajado con este

mismo equipo en la primera fase del caso de estudio. Como se comentó en la sección 3.2.6 *Implementar bucles de retroalimentación y mejorar colaborativamente*, entre estas dinámicas se encontraban las reuniones de retrospectiva en las que el equipo analiza su forma de trabajo buscando oportunidades de mejora. Con el transcurso del tiempo, es habitual entrar en una cierta **inercia en el proceso de mejora** que, desde nuestro punto de vista, se ve agravada en equipos de *outsourcing* que prestan sus servicios en las instalaciones del cliente, fuera de las oficinas de la consultora tecnológica. Para contrarrestar este efecto, se debe ampliar la motivación y cohesión del equipo. Para ello, se puede utilizar las técnicas de coaching e inteligencia emocional grupal y en concreto el proceso ETeC definido.

La aplicación de ETeC facilitará la identificación de los principales factores de **frustración colectiva**, permitiendo la definición de un **plan de acción** que aproveche el **interés común** como palanca de cambio para resolver la frustración detectada.

En cuanto a la extrapolación de la propuesta a otros equipos de diferentes campos, el coaching de equipos en la actualidad se aplica ampliamente en otros entornos como el empresarial o el marketing, como se mencionó anteriormente. De hecho, la experiencia de las *coaches* que participaron en nuestra investigación, aplicando técnicas de coaching de equipos en otros campos, es una de las garantías de la propuesta. La novedad aquí proviene de la combinación de técnicas de coaching e inteligencia emocional grupal con el desarrollo de software, que es básicamente una tarea humana realizada por equipos y, por tanto, sujeta a la mejora mediante mejores técnicas de motivación y gestión de equipos. Esta es básicamente nuestra propuesta y, por tanto, creemos que se puede extrapolar a otros equipos de desarrollo de software.

Ahora bien, existen alternativas a algunas de las técnicas que se utilizaron en nuestra investigación. En este sentido, la existencia de otras técnicas, el *Brainstorming* (Osborn, 2012) y los Grupos Focales no estructurados (Merton, 1987) merecen ser considerados como otras opciones para impulsar la toma de decisiones en grupo. Asimismo, la Escala del Perfil de Inteligencia Emocional del Grupo de Trabajo (WEIP-S, del inglés Work Group Emotional Intelligence Profile - Scale) (Jordan & Lawrence, 2009) podría usarse en lugar de la prueba TEI. A continuación, explicamos las razones de las elegidas para nuestra investigación.

NGT es básicamente un refinamiento de *Brainstorming* que promueve acciones individuales para generar más ideas. Su advenimiento se relacionó con algunos resultados que indican que las personas suelen autocensurarse en grupos

para evitar críticas o valoraciones, o que pueden estar dominadas por extrovertidos (Sample, 1984). Algunos trabajos también concluyeron que la participación grupal cuando se usa en las sesiones de *Brainstorming* inhibe el pensamiento creativo en comparación con el uso de técnicas silenciosas (Taylor et al., 1958). Además, se prefiere NGT cuando a las partes interesadas les gusta una (o alguna) salida cuantitativa del proceso.

En comparación con los grupos focales no estructurados, en (Delbecq et al., 1975) se resumieron algunas ventajas de los grupos nominales relacionadas con mayores niveles de satisfacción, cantidad y calidad de ideas generadas y con un estatus igual para todos los participantes. Además, el hecho de que las NGT estén más estructuradas implica menores necesidades de formación para el moderador (Delbecq & Van de Ven, 1971).

Una de las principales desventajas de NGT es que el método carece de flexibilidad porque solo puede tratar un problema a la vez. Además, los miembros involucrados en NGT deben estar de acuerdo hasta cierto punto y todos deben sentirse cómodos con la estructura involucrada. Otra desventaja es la cantidad de tiempo necesario para prepararse para la actividad: no hay espontaneidad involucrada con este método. Las instalaciones deben organizarse y planificarse cuidadosamente. Es posible que las opiniones no converjan en el proceso de votación, la fertilización cruzada de ideas puede verse limitada y el proceso puede parecer demasiado mecánico.

En resumen, dado que la votación es anónima en NGT, se facilita la participación equitativa de los miembros del grupo y se reducen las distracciones relacionadas con la charla presente en otros métodos. Por otro lado, es posible que el proceso de votación no converja y que el proceso parezca demasiado mecánico. Dado todo esto, se descubrió que la naturaleza estructurada de NGT encaja mejor con la mente estructurada de los ingenieros de software que componen el equipo.

Por otra parte, al tratarse de un tema mucho más reciente, no existe mucha literatura sobre la evaluación de la **inteligencia emocional en equipo**. La elección del test TEI frente a WEIP-S estuvo más relacionada con la naturaleza del equipo en estudio. La literatura existente aclara las diferencias entre los empleados internos y los subcontratados. Las diferentes condiciones contribuyen a un clima organizacional negativo que conduce a una menor (o nula) identificación y sentido de pertenencia de los profesionales subcontratados y, en consecuencia, impacta en la eficiencia del equipo (Brooks et al., 2018). Por ello, optamos por utilizar el Test

TEI de (Druskat & Wolff, 2001), ya que el sentido de pertenencia e identificación juegan un papel clave en los equipos de alto rendimiento según su modelo de TEI.

Por lo tanto, podría darse el caso de que el *coach* que aplica ETeC prefiera utilizar otras técnicas, ya sea porque se sienten más seguros con esa técnica o por alguna característica particular del equipo en estudio. Sin embargo, el proceso (fases e incluso sesiones) podría seguir siendo exactamente el mismo. En el peor de los casos, el *coach* podría optar por diseñar un proceso ETeC completamente diferente. Aun así, estaría aplicando técnicas de coaching e inteligencia emocional grupal al desarrollo del software. De este modo, nuestro trabajo habría servido entonces para inspirar, tanto a profesionales del coaching como a los del software, al mostrar que la aplicación de estas técnicas produce resultados reales en contextos industriales.

Por todo lo anterior, podemos concluir tras la experiencia con este equipo de *outsourcing*, que las técnicas de coaching e inteligencia emocional grupal son beneficiosas para todos los interesados en esta modalidad de prestación de servicio: cliente, consultora, equipos y personas. El proceso permitió **descubrir frustraciones colectivas** que, de no actuar sobre ellas, habrían acabado generando una desmotivación personal que conduce a la reducción de la productividad del equipo y al estancamiento del desarrollo personal de sus integrantes. Justo el interés contrario que cualquier consultora tecnológica y cliente al que presta sus servicios puedan tener: contar con equipos de alto rendimiento.

Conclusiones

Para finalizar, durante el presente capítulo se resumen las contribuciones principales y se contrastan los objetivos alcanzados con los establecidos al comienzo de la Tesis Doctoral. Además, se plantean una serie de preguntas para futuras investigaciones.

5.1 Contribuciones principales y resultados científicos

En esta sección se muestran las principales contribuciones de la presente Tesis Doctoral.

La **principal contribución** de nuestra investigación es la definición de una propuesta metodológica para la mejora de la productividad de los equipos de mantenimiento de software. Para ello, se lleva a cabo la adaptación del método Lean Kanban y su novedosa combinación con un proceso de coaching e inteligencia emocional grupal.

Algunos de los aspectos aplicados durante nuestra investigación, son compartidos con los estudios previos analizados. Ahora bien, identificamos dos aspectos sobre los que, a nuestro entender, no existen estudios prácticos aplicados y que nuestra investigación contribuye a acentuar su importancia durante la implantación del método Lean Kanban: la identificación de las **fuentes de variabilidad** del proceso y la definición de las **clases de servicio**. Consideramos que la comprensión de la variabilidad y cómo impacta en el rendimiento de un proceso debe ser una herramienta esencial de cualquier gestor. De esta forma, la aplicación de las acciones oportunas, basadas en el estudio de la variabilidad, debe ser uno de los elementos centrales de cualquier programa de mejora continua. Por otra parte, la definición de las diferentes clases de servicios permite el establecimiento de diferentes acuerdos de niveles de servicios. Esto ofrece una ventaja tanto en la agilidad aportada al negocio como a la propia gestión del riesgo de los proyectos.

Otra de nuestras contribuciones ha sido la incorporación al método de determinadas prácticas que, desde nuestra perspectiva, mejoran su rendimiento. Por un lado, la **estimación** de todos los encargos de trabajo y por otro la reducción de la **periodicidad de los ciclos de retroalimentación** que proponemos establecer en semanales.

Para concluir, hemos de indicar que, tras la revisión de la bibliografía disponible, no encontramos estudios que agrupen la **visión global** de la investigación de la presente Tesis Doctoral, combinando la **mejora gradual** de un equipo con el mantenimiento de su cultura de mejora continua a través de la aplicación de **técnicas de coaching e inteligencia emocional grupal**.

Algunos de los **resultados** de esta Tesis Doctoral han sido publicados en diferentes foros y se enumeran a continuación:

ARTÍCULOS EN REVISTAS INTERNACIONALES

- Esperanza Marcos, Roberto Hens, Teresa Puebla y Juan Manuel Vara (2020). *Applying Emotional Team Coaching to Software Development*, IEEE Software. DOI: <https://doi.org/10.1109/MS.2020.3007658>

ARTÍCULOS EN CONFERENCIAS INTERNACIONALES

- Roberto Hens, Esperanza Marcos, Juan Manuel Vara (2015). *Improving ICT services management through the adoption of Lean Kanban*. 4th International Business Servitization Conference, Madrid, Spain. DOI: <https://doi.org/10.3926/serv2015>
- Roberto Hens, Esperanza Marcos, Juan Manuel Vara, David Granada (2020). *Lean Kanban in an industrial context: a success story*. International Conference Software Engineering 2020, Seoul, South Korea.

PONENCIAS EN CONFERENCIAS NACIONALES

- Roberto Hens (2019), *Transformación Agile en un Ministerio*. IT Agile Enterprise | Agile 19, Madrid, España.

CONTRIBUCIONES A LA INDUSTRIA

- El caso de estudio expuesto en la presente Tesis Doctoral también tuvo una contribución directa sobre el entorno real. En concreto, las lecciones aprendidas derivadas de su ejecución tuvieron una importante repercusión en la organización cliente sobre la que el equipo prestaba sus servicios. Dado los resultados obtenidos tras la aplicación de K-ETeC, la organización decidió extender la iniciativa a sus 16 equipos de desarrollo. Para ello, decidió poner en marcha una oficina Lean dirigida por el doctorando. En la actualidad, la oficina cuenta con 5 personas que se encargan de extender las lecciones aprendidas de nuestra investigación enfocándose sistemáticamente en la mejora de la productividad de los equipos de desarrollo. Dada la importancia de sus

logros, la organización cliente ha sido invitada a la presentación de la oficina Lean en congresos nacionales:

- *Oficina de Mejora Continua: Transformación Agile en un Ministerio.* Tech Management Day, 2019, Madrid, España. (Valorada por los asistentes como la mejor ponencia del evento).

5.2 Análisis de la consecución de los objetivos

Al comienzo de esta Tesis Doctoral, en el capítulo 1, se definió un objetivo principal: *la mejora de la productividad de un equipo de mantenimiento de software mediante la aplicación del método Lean Kanban, mediante la implantación de las prácticas que se consideren más oportunas atendiendo a las características del equipo, así como la sostenibilidad de la cultura de mejora continua a través de la aplicación de técnicas de coaching e inteligencia emocional grupal.* Junto a este objetivo principal se definieron un conjunto de objetivos parciales que permitirían alcanzarlo. A continuación, se analiza el grado de cumplimiento de cada uno de los objetivos parciales definidos.

01. Estudios de trabajos previos

Para el contexto de la presente Tesis Doctoral se estudiaron los conceptos previos requeridos para afrontar con éxito la investigación, así como los trabajos relacionados disponibles en la bibliografía.

Por un lado, se estudió la **situación del método Lean Kanban**, desde la perspectiva de su fundador, David J. Anderson y la relación con las diferentes metodologías y líneas de pensamiento que influyeron en la definición del método tal como lo propone: Lean Software Development, la Teoría de las Restricciones y las Metodologías Ágiles.

A su vez, se estudió el efecto que produce la prestación de servicios en modalidad de **outsourcing** sobre los equipos de trabajo, y en especial, en los de desarrollo de software. Se prestó especial atención en el efecto que puede tener esta modalidad de contratación en la motivación de los integrantes del equipo de trabajo.

Por último, se analizó la situación en lo que respecta a **coaching de equipos e inteligencia emocional**. Si bien es cierto que en la actualidad hay una gran cantidad de literatura relacionada con el *Agile Coaching*, en realidad este tipo de coaching está muy orientado a la facilitación y despliegue de prácticas ágiles en equipos de desarrollo. Sin embargo, en nuestra

investigación nos centramos en aquellas cuestiones relacionadas con el funcionamiento como equipo: motivación, estado de ánimo, cohesión, etc.

Por último, se analizaron los **trabajos previos existentes** sobre la aplicación práctica de los elementos básicos del método propuesto por Anderson, encontrando ciertas diferencias con nuestra propuesta. A su vez se buscaron, sin éxito, trabajos previos que incorporaran, en la gestión de equipos de desarrollo, técnicas de coaching e inteligencia emocional.

O2. Formación del equipo en el método Lean Kanban.

De cara a la ejecución de nuestro caso de estudio se planteó como objetivo de la Tesis Doctoral la **formación del equipo** en el método Lean Kanban. Lo cierto es que, si bien considerábamos importante que el equipo no tuviera experiencia previa en el desarrollo Lean, en ningún caso fue condición imprescindible la falta de conocimientos en metodologías ágiles o el pensamiento Lean de sus integrantes. Aun así, pudimos concluir que no disponían de conocimientos básicos de agilidad y que ninguno de los integrantes había tenido contacto con el método Lean Kanban.

De este modo, se desarrollaron una serie de **seminarios de formación** en los que se impartieron conocimientos teóricos sobre agilidad en general y el método Lean Kanban en particular. Igualmente, se llevaron a cabo diferentes talleres para poner en práctica los fundamentos teóricos tratados: talleres de historias de usuario y el **juego colaborativo *getKanban*** (Heikkilä et al., 2016).

O3. Adaptación y aplicación del método Lean Kanban.

Este objetivo parcial lo dividimos a su vez en cuatro objetivos. Por un lado, la selección de cambios a introducir en el equipo, el impulso de las prácticas a introducir, la medición y valoración del impacto sobre los indicadores de rendimiento y el análisis de resultados y lecciones aprendidas.

Uno de los objetivos principales en la presente Tesis Doctoral era la aplicación del método Lean Kanban. De hecho, éste se puede considerar el aspecto principal de la primera fase de nuestro caso de estudio. Los anteriores objetivos parciales estaban enfocados en establecer una línea base que nos permitiera afrontar este tercer objetivo con garantías.

Como en cualquier proceso de cambio, la **puesta en marcha** de la iniciativa Lean Kanban propuesta en nuestra investigación era una de las fases

con mayor riesgo de fracaso. Dado que se debía contar con el compromiso del equipo, fue necesario trabajar el aspecto motivacional. Se evitaron las decisiones autoritarias y se argumentaron, con datos objetivos de los trabajos previos analizados, los beneficios esperados a través de la aplicación del método propuesto. Tras la puesta en marcha, el equipo tenía clara la posibilidad de mejorar sensiblemente su productividad sin necesidad de incrementar el estrés por la incorporación de nuevas dinámicas de trabajo.

Como se comentó en la sección 3.4 *Lecciones aprendidas*, durante la aplicación del caso de estudio se incorporó un elevado número de cambios en el proceso de desarrollo que afectaron a diferentes aspectos: organizativos, técnicos y de calidad. Dado que el objetivo de nuestra investigación es la mejora del proceso en general, no entramos a valorar las diferentes prácticas incorporadas. Nuestra propuesta no pretende prescribir combinación de prácticas idónea para la mejora de la productividad de los equipos, sino que se basa en la aplicación del método Lean Kanban como facilitador del cambio que permite alcanzar estos grados de mejora.

Nótese que determinadas prácticas que el equipo incorporó en nuestro caso de estudio pueden no ser idóneas para otros equipos, dado que, por diferencias en su contexto de trabajo, no permiten su aplicación o no retornan los mismos beneficios. De esta forma, las prácticas incorporadas atienden únicamente al contexto del caso de estudio de nuestra investigación, por lo que no pretendemos concluir su idoneidad para todos los equipos. Será responsabilidad del responsable de la transformación de cada equipo, la selección de las prácticas más ajustadas a sus circunstancias.

Ahora bien, sí hay ciertos puntos de la aplicación original que **adaptamos** dadas las ventajas que, a nuestro entender, aportan en el proceso. En la primera aplicación del método (Anderson & Dumitriu, 2005) la **estimación** consumía hasta el 40% de la capacidad productiva del equipo. Una de las mejoras planteadas por Anderson fue la eliminación de la estimación, lo que, atendiendo a los resultados del estudio produjo, una mejora importante e inmediata de la productividad; al eliminar la estimación, se reducen los costes de la priorización facilitando la celebración de reuniones de priorización con mayor frecuencia. Ahora bien, dado que en la mayoría de los proyectos sigue siendo preciso facilitar el pronóstico del tiempo de entrega de los servicios, Anderson propone el pronóstico probabilístico como método alternativo para

su elaboración. Este método brinda la oportunidad de basar el pronóstico en el flujo de valor observado (Anderson & Carmichael, 2016).

En la propuesta **K-ETeC**, sí optamos por incorporar la **estimación**. Como se ha comentado previamente en la sección 3.2 *Ejecución del caso de estudio*, nuestra situación de partida fue diferente a la primera aplicación del método. Ellos disponían de un equipo de mantenimiento de software con un nivel de madurez CMMI-5 y utilizaban como método de trabajo el *Team Software Process / Personal Software Process (TSP/PSP)*, mientras que, en nuestro caso, partíamos de un equipo sin un método de trabajo claramente definido, que no realizaba estimación alguna de sus encargos de trabajo. Cuando nos decidimos a incluir la estimación en el proceso, pudimos observar que los indicadores de rendimiento no se resentían, y menos, en unos valores del 40%.

Desde nuestro punto de vista, la diferencia fue nuestra apuesta por métodos de estimación ágiles (Planning Poker, por ejemplo) que ofrecen la precisión necesaria para facilitar la planificación y predicción requerida sin un coste elevado (Cohn, 2005).

Entre las **ventajas** obtenidas de la estimación se encuentran la facilidad de las labores de priorización, la mejora de la medición del rendimiento (*Throughput*) del proceso y la transferencia de conocimiento entre los miembros del equipo. Esta última, a su vez, aporta unos beneficios adicionales como es la disposición del equipo para evitar los proyectos silos, aumentando con ello a facilidad de redistribución de recursos ante variaciones puntuales de la demanda, evitando la generación de cuellos de botella. Estos afectan negativamente al rendimiento del proceso (aumentando los tiempos de entrega) y a la motivación del equipo debido al incremento del estrés.

La segunda **adaptación** realizada sobre el método se refiere a los ciclos de **retroalimentación**. Si bien es cierto que la propuesta inicial del método Lean Kanban incorpora las retrospectivas como circuitos básicos de retroalimentación, que permiten el crecimiento de la madurez organizacional y la mejora continua a nivel de la organización, en nuestra propuesta reducimos el intervalo entre estas reuniones bajándolo, de una **periodicidad** mensual (Anderson, 2010), a una **semanal**.

Se buscaron varios **objetivos** en esta adaptación. Por un lado, mantener al equipo muy focalizado en la iniciativa. Es un error habitual que las propuestas de cambio arranquen con planes de transformación a cinco años y que, pasados pocos meses, la mayor parte de la organización no es consciente de

la situación real de la iniciativa. En nuestra propuesta, al mantener una frecuencia semanal, los miembros del equipo llevan a cabo un seguimiento riguroso de los cambios. Esto facilita, atendiendo a los indicadores de rendimiento, la toma de decisiones en cuanto a la conveniencia de los cambios incorporados o su retirada del proceso.

Consideramos que el grado de consecución de este objetivo de adaptación y aplicación del método es elevado como se desprende de la encuesta de consecución de objetivos mostrada en la figura F-3.16 de la sección 3.3 *Análisis de resultados*.

O4. Aplicación de técnicas de coaching e inteligencia emocional grupal.

Este objetivo parcial lo dividimos a su vez en tres objetivos. Por un lado, la ejecución de la fase de Diagnóstico, posteriormente la ejecución de la fase de Intervención y finalmente el análisis de resultados y lecciones aprendidas.

Como se comentó en la sección 4.2.1 *Fase de Diagnóstico*, en una primera instancia se pretendía realizar una evaluación inicial del equipo de trabajo teniendo en cuenta un punto de vista interno (perspectiva del equipo) y uno externo (observación del funcionamiento en reuniones y valoración por parte de las *coaches*). Esta evaluación inicial fue llevada a cabo con éxito lo que permitió observar que la falta de motivación era un factor clave en el desempeño del equipo. Se pudo concluir que la falta de motivación procedía de factores como el aislamiento y carencia de una visión compartida de los objetivos. A su vez, permitió clasificar al equipo como poco proactivo y con dificultad para la resolución de problemas, dejando al descubierto que se encontraban en una situación de estancamiento. Por todo ello, como **resultado** de esta fase de diagnóstico se planteó la intervención orientada a una **mejora de la motivación** del equipo.

La **fase de intervención** propuesta en el proceso ETeC se desarrolla en tres sesiones. Las dos primeras tienen como objetivo abordar las principales debilidades detectadas en la fase de diagnóstico, en nuestro caso el problema motivacional, incidiendo en el aislamiento. A su vez se trabajó la proactividad, la comunicación y la resolución de problemas. Durante la última sesión se propone pasar a la acción mediante el desarrollo de un plan de trabajo para alcanzar los objetivos definidos en las sesiones previas.

El **análisis de resultados** mostró que la **motivación del equipo** de trabajo había aumentado al finalizar el proceso. Desde nuestra perspectiva, esto se

debió a la suma de dos circunstancias principales. Por un lado, el cambio profesional voluntario de dos miembros del equipo con un alto grado de desmotivación. Esto facilitó la incorporación, de forma no traumática, de personal que arrastraba años de desmotivación en el cliente, haciendo un porte positivo a la motivación media del equipo. Por otro lado, el resto de integrantes continuaron ejecutando el plan de acción lo cual facilitó que se sintieran más motivados, aumentando con ello, la motivación media. Concluimos, por tanto, que el equipo se benefició, puesto que su motivación colectiva, entendida como la media agrupada de cada integrante del equipo había aumentado tras la ejecución de ETeC.

Otro de los resultados alcanzados fue la implementación de una **meta común** en el equipo de trabajo. Esta era una cuestión importante de nuestro caso de estudio ya que permitía focalizar el esfuerzo colectivo, donde se percibía la necesidad de mejora, desde una perspectiva de equipo. Hay que tener en cuenta que en una modalidad de prestación de servicios de *outsourcing* es muy habitual la desorientación de los equipos de trabajo en la definición de metas que, en ocasiones, pueden ser contrapuestas: la meta individual de cada integrante, la meta del equipo, la meta de la consultora tecnológica y, por último, la meta de la organización cliente. Al finalizar el proceso, estas metas habían sido alineadas.

Por último, analizamos el **retorno de la inversión** de la aplicación del proceso ETeC. Este análisis lo llevamos a cabo enfrentando la inversión económica requerida con los beneficios esperados. En cuanto al impacto **económico** directo, se llevó a cabo un estudio de los costes del proceso asociados tanto al *coach* como al tiempo dedicado por cada ingeniero del software (fácilmente traducible a precio/hora en un entorno de *outsourcing*). En cuanto a los beneficios esperados, se detectaron ventajas como el ahorro de tiempo en reuniones futuras, los efectos positivos sobre la productividad y la calidad derivados de un mejor clima laboral, así como la disminución de los niveles de absentismo. Nuestra conclusión es que los resultados de la experiencia amortizan holgadamente los costes generados de su aplicación.

En cuanto a las **lecciones aprendidas** cabe destacar que la aplicación de ETeC permite la identificación de los principales **factores de frustración colectiva** facilitando la definición de un **plan de acción** que posibilita el aprovechamiento del interés común como palanca de cambio para resolver la frustración detectada durante el diagnóstico.

Por último, queremos subrayar que nuestra propuesta se circunscribe a la definición del proceso en sí, no las técnicas de coaching llevadas a cabo en nuestro caso de estudio. De este modo, la elección de las técnicas a aplicar durante las sesiones definidas en ETeC son elección del profesional que esté ejecutando el proceso y atenderá a factores ambientales del equipo, así como el propio conocimiento y experiencia de este profesional con estas u otras técnicas. De este modo, las técnicas pueden variar mientras que el proceso propuesto seguiría siendo exactamente el mismo.

5.3 Trabajos futuros

A partir de las contribuciones alcanzadas en la presente Tesis Doctoral, se han detectado varios caminos para seguir trabajando. Algunos de ellos no fueron considerados como objetivos de esta Tesis Doctoral, mientras que otros han surgido durante la evaluación del caso de estudio. A continuación, resumimos algunos de ellos.

La investigación llevada a cabo se ha centrado en una tipología muy específica de equipos de trabajo, sobre los que habíamos anticipado una serie de problemas que preveíamos poder solucionar. En concreto, nos centramos en equipos que prestan un servicio de mantenimiento de software. A su vez, nos interesaba analizar el efecto que tiene en la productividad del equipo el tipo de relación laboral. Para ello nos centramos en la modalidad de *outsourcing*, actualmente muy extendida a nivel global y en España en particular. Ahora bien, consideramos que el proceso K-ETeC puede ser utilizado en **equipos con otras características**. Por esto, pensamos que puede resultar interesante confirmar esta reflexión mediante la aplicación de nuestra propuesta en los siguientes equipos:

- **Equipos de desarrollo de productos software.** Dado que la tipología de los trabajos a realizar difiere significativamente frente a un equipo de mantenimiento de software, sería interesante verificar las adaptaciones al método Lean Kanban propuestas en K-ETeC. Por otra parte, consideramos que el aspecto motivacional igualmente se diferencia del equipo de nuestro caso de estudio, debido al proceso creativo existente en la construcción de nuevos productos, por lo que sería interesante verificar igualmente la aplicación del proceso de coaching a equipos de estas características.
- **Equipos *in-house* de mantenimiento software.** Un aspecto clave de la evaluación de nuestro caso de estudio fue la modalidad de contratación. De este modo, consideramos de interés extender la aplicación del método K-ETeC

a equipos contratados directamente por la organización en la que prestan servicios, extrayendo la modalidad de *outsourcing* de la ecuación.

- **Equipos de prestación de otros servicios.** Como se comentó en el capítulo 2, el pensamiento Lean, procedente del entorno industrial, actualmente está siendo aplicado en la gestión de una gran variedad de servicios. Es por esto que consideramos de interés, para la ciencia y gestión de los servicios, la validación de nuestra propuesta en la prestación de otros tipos de servicios que no tengan relación con la ingeniería del software. Los principios Lean y las adaptaciones realizadas al método Lean Kanban en K-ETeC sostenemos que pueden ser de aplicación en otros sectores de actividad.

Otro aspecto de posible interés para continuar la presente investigación es la construcción de una **herramienta software** que de soporte al método. Si bien es cierto que actualmente existe una gran cantidad de herramientas para la automatización del método Lean Kanban y el cálculo de sus métricas principales, la presión del mercado ha hecho que ninguna de las ellas implemente de forma rigurosa los principios y valores del método original y, por extensión, el método K-ETeC. Además, en ningún caso, dada la novedad incorporada en nuestra investigación, se cubre el proceso de coaching emocional grupal. Una futura línea de investigación sería la implementación de una herramienta que facilitara el seguimiento de las sesiones, sirviera de repositorio documental de la evolución de las mismas y facilitara la evolución temporal de la situación de los equipos.

Dada la extensión del caso de estudio llevado a cabo en nuestra investigación, para su presentación nos ha sido imposible **contrastar los resultados** con otros equipos de iguales características. De esta forma, consideramos de interés la reproducción de nuestro caso de estudio con varios equipos de trabajo similares, con el fin de comparar los resultados obtenidos y extender las lecciones aprendidas. Un aspecto que nos ha quedado pendiente de resolver y que igualmente puede constituir una futura línea de investigación, es la definición de los **plazos para el desarrollo del proceso ETeC de forma periódica**. Desafortunadamente en la actualidad no disponemos del conocimiento para ofrecer una recomendación válida para todos los casos. Una base anual podría funcionar, pero es algo que dependerá del desempeño del equipo. Ante un retorno de la inercia negativa, es el responsable del proyecto quien deberá decidir si ha llegado el momento de impulsar una nueva intervención para abordar este problema. Una futura investigación podría dotarle de unos parámetros válidos que facilite esta toma de decisión.

Bibliografía

- Acuña, S. T., Gómez, M., & Juristo, N. (2008). Towards understanding the relationship between team climate and software quality -a quasi- experimental study. *Empirical Software Engineering*, 13(4), 401.
- Adkins, L. (2010). *Coaching agile teams: A companion for ScrumMasters, agile coaches, and project managers in transition*. Boston, USA: Pearson Education.
- Agusa, K. (2004). Software engineering evolution. Paper presented at the *7th International Workshop on Principles of Software Evolution*, 3-8.
- Ahmed, R. E. (2006). Software maintenance outsourcing: Issues and strategies. *Computers & Electrical Engineering*, 32(6), 449-453.
- Ali, S., Ullah, N., Abrar, M. F., Majeed, M. F., Umar, M. A., & Huang, J. (2019). Barriers to software outsourcing partnership formation: An exploratory analysis. *IEEE Access*, 7, 164556-164594.
- Alturki, A., Gable, G. G., & Bandara, W. (2011). A design science research roadmap. Paper presented at the *International Conference on Design Science Research in Information Systems*, 107-123.
- Anderson, D. J. (2010). *Kanban: Successful evolutionary change for your technology business*. Sequim, Washington, USA: Blue Hole Press.
- Anderson, D. J. (2011). Kanban, when is it not appropriate. Retrieved June 25, 2017, from <https://es.slideshare.net/AGILEMinds/david-anderson-kanban-when-is-it-not-appropriate>.
- Anderson, D. J., & Carmichael, A. (2016). *Essential kanban condensed*. Seattle, Washington, USA: Lean Kanban University Press.
- Anderson, D. J., & Dumitriu, D. (2005). From worst to best in 9 months: Implementing a drum-buffer-rope solution at Microsoft's IT department. Paper presented at the *TOC ICO World Conference*, Barcelona.
- Andriyani, Y., Hoda, R., & Amor, R. (2017). Reflection in agile retrospectives. Paper presented at the *Agile Processes in Software Engineering and Extreme Programming. XP 2017*, Cologne (Germany), 283.

- Antonucci, Y. L., Lordi, F. C., & Tucker III, J. J. (1998). The pros and cons of IT outsourcing. *Journal of Accountancy*, 185(6), 26.
- Appelo, J. (2011). *Management 3.0: Leading agile developers, developing agile leaders*. Indiana, USA: Pearson Education.
- Basili, V. R., Selby, R. W., & Hutchens, D. H. (1986). Experimentation in software engineering. *IEEE Transactions on Software Engineering*, SE-12(7), 733-743.
- Baxter, P., & Jack, S. (2008). Qualitative case study methodology: Study design and implementation for novice researchers. *The Qualitative Report*, 13(4), 544-559.
- Beck, K. (2000). *Extreme programming explained: Embrace change*. Addison-Wesley.
- Beck, K., Beedle, M., Van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M.,...Jeffries, R. (2001, February). Manifesto for agile software development. Retrieved May 10, 2014, from <https://agilemanifesto.org>.
- Benamati, J., & Rajkumar, T. M. (2002). The application development outsourcing decision: An application of the technology acceptance model. *Journal of Computer Information Systems*, 42(4), 35-43.
- Benko, C. (1992). If information system outsourcing is the solution, what is the problem? *Journal of Systems Management*, 43(11), 32.
- Bhamu, J., & Sangwan, K. S. (2014). Lean manufacturing: Literature review and research issues. *International Journal of Operations & Production Management*.
- Blaga, P. (2020). The importance of human resources in the continuous improvement of the production quality. *Procedia Manufacturing*, 46, 287-293.
- Brooks, F. P. (1975). *Thy mythical man month, essays on software engineering*. Massachusetts, USA: Addison-Wesley.
- Brooks, F. P., & Bullet, N. S. (1987). Essence and accidents of software engineering. *IEEE Computer*, 20(4), 10-19.

- Brooks, N. (2006). Understanding IT outsourcing and its potential effects on IT workers and their environment. *Journal of Computer Information Systems*, 46(4), 46-53.
- Brooks, N. G., Miller, R. E., & Korzaan, M. L. (2009). IT workers on outsourcing: What about me? What about the profession? *Academy of Information & Management Sciences Journal*, 12(1-2).
- Brue, S. L. (1993). Retrospectives: The law of diminishing returns. *Journal of Economic Perspectives*, 7(3), 185-192.
- Bunge, M. (1989). *La investigación científica*. Barcelona: Ariel.
- Burrows, M. (2014). *Kanban from the inside*. Washington, USA: Blue Hole Press.
- Capretz, L. F. (2014). Bringing the human factor to software engineering. *IEEE Software*, 31(2), 104.
- Cardón, A. (2007). *Coaching de equipos*. Barcelona: Ediciones Gestión 2000.
- Carnegie Mellon University. (2009). Brief history of CMMI. Retrieved May 2018, from https://resources.sei.cmu.edu/asset_files/Brochure/2009_015_001_28416.pdf
- Christensen, C. M. (2000). *The innovator's dilemma*. New York: Harper Business.
- Cockburn, A. (2006). *Agile software development: The cooperative game*. Pearson Education.
- Cohen, S. G., & Bailey, D. E. (1997). What makes teams work: Group effectiveness research from the shop floor to the executive suite. *Journal of Management*, 23(3), 239-290.
- Cohn, M. (2005). *Agile estimating and planning*. Nueva Jersey, USA: Pearson Education.
- Corbett, M. F. (1994). Outsourcing and the new it executive a trends report. *Information Systems Management*, 11(4), 19-22.
- Corona, E., & Pani, F. E. (2013). A review of lean-kanban approaches in the software development. *WSEAS Transactions on Information Science and Applications*, 10(1), 1-13.

- Da Silva, F. Q., & França, A. C. C. (2012). Towards understanding the underlying structure of motivational factors for software engineers to guide the definition of motivational programs. *Journal of Systems and Software*, 85(2), 216-226.
- Dabhilkar, M., & Bengtsson, L. (2004). Balanced scorecards for strategic and sustainable continuous improvement capability. *Journal of Manufacturing Technology Management*, 15(4), 350-359.
- De Roze, B. C., & Nyman, T. H. (1978). The software life cycle- A management and technological challenge in the department of defense. *IEEE Transactions on Software Engineering*, SE-4(4), 309-318.
- Delbecq, A. L., & Van de Ven, Andrew H. (1971). A group process model for problem identification and program planning. *The Journal of Applied Behavioral Science*, 7(4), 466-492.
- Delbecq, A. L., Van de Ven, Andrew H., & Gustafson, D. H. (1975). *Group techniques for program planning: A guide to nominal group and delphi processes*. Scott, Foresman.
- Delen, G. P., Peters, R. J., Verhoef, C., & Van Vlijmen, S. (2016). Lessons from dutch IT-outsourcing success and failure. *Science of Computer Programming*, 130(130), 37-68.
- Deming, W. E. (1993). The new economics. Massachusetts Institute of Technology. *Center for Advanced Engineering Study*. Cambridge, 240p.
- Denton, D. K., & Denton, R. A. (1999). *The toolbox for the mind: Finding and implementing creative solutions in the workplace*. McGraw-Hill.
- Dingsøyr, T., Nerur, S., Balijepally, V., & Moe, N. B. (2012). A decade of agile methodologies: Towards explaining agile software development. *The Journal of Systems and Software*, 85(6), 1213-1221.
- Drucker, P. (1959). *Landmarks of tomorrow: A report on the new post-modern world*. Transaction Publishers.
- Drucker, P. F. (1999). Knowledge-worker productivity: The biggest challenge. *California Management Review*, 41(2), 79-94.
- Druskat, V. U., & Wolff, S. B. (2001a). Building the emotional intelligence of groups. *Harvard Business Review*, 79(3), 80-91.

- Druskat, V. U., & Wolff, S. B. (2001b). Group emotional intelligence and its influence on group effectiveness. *The emotionally intelligent workplace: How to select for, measure, and improve emotional intelligence in individuals, groups and organizations* (pp. 132-155) Jossey-Bass.
- Earl, M. J. (1996). The risks of outsourcing IT. *Sloan Management Review*, 37, 26-32.
- Echeverría, R. (2017). *Ontología del lenguaje*. Ediciones Granica.
- Eisenhardt, K. M. (1989). Building theories from case study research. *Academy of Management Review*, 14(4), 532-550.
- Elfenbein, H. A., Druskat, V. U., Sala, F., & Mount, G. (2006). *Team emotional intelligence: What it can mean and how it can affect performance*. New Jersey, USA: Lawrence Erlbaum Mahwah.
- Ganesh, M. P. (2013). Climate in software development teams: Role of task interdependence and procedural justice. *Asian Academy of Management Journal*, 18(1), 55.
- García, J. L., Maldonado, A. A., Alvarado, A., & Rivera, D. G. (2014). Human critical success factors for kaizen and its impacts in industrial performance. *The International Journal of Advanced Manufacturing Technology*, 70(9-12), 2187-2198.
- Garzías Parra, J., Enríquez de S, J A, & Irrazábal, E. (2012). *Gestión ágil de proyectos software*. Madrid, Spain: Kybele Consulting.
- Gilb, T., & Finzi, S. (1988). *Principles of software engineering management*. Michigan, USA: Addison-Wesley Publishing Company.
- Glass, R. L. (1994). The software-research crisis. *IEEE Software*, 11(6), 42-47.
- Goldratt, E. M. (1994). *It's not luck*. England: The North River Press.
- Goldratt, E. M. (1997). *Critical chain*. Great Barrington, Massachusetts, USA: The North River Press.
- Goldratt, E. M., & Cox, J. (1984). *The goal: A process of ongoing improvement* (First ed.). Great Barrington, Massachusetts, USA: North River Press.
- Goleman, D. (1998). *Working with emotional intelligence*. Bantam.

- Grenning, J. (2002). Planning poker or how to avoid analysis paralysis while release planning. *Hawthorn Woods: Renaissance Software Consulting*, 3, 22-23.
- Günsel, A., & Açikgöz, A. (2013). The effects of team flexibility and emotional intelligence on software development performance. *Group Decision and Negotiation*, 22(2), 359-377.
- Hackman, J. R., & Oldham, G. (1980). Work redesign. *Reading*, 7(1), 121-124.
- Heikkilä, V. T., Paasivaara, M., & Lassenius, C. (2016). Teaching university students kanban with a collaborative board game. Paper presented at the *Proceedings of the 38th International Conference on Software Engineering Companion*, 471-480.
- Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science information systems research MIS quarterly. *MISQ Discovery*, 28(1).
- Hevner, A., & Chatterjee, S. (2010). Design science research in information systems. *Design research in information systems* (pp. 9-22) Springer.
- Hiroi, S., Miyazaki, M., Koizumi, Y., & Konosu, T. (2005). Evaluating the competence and emotional intelligence of members in a software development project. *Journal of Managerial Psychology*, 20, 178-187.
- Ikonen, M., Pirinen, E., Fagerholm, F., Kettunen, P., & Abrahamsson, P. (2011). On the impact of kanban on software project work: An empirical case study investigation. Paper presented at the *2011 16th IEEE International Conference on Engineering of Complex Computer Systems*, 305-314.
- ISERN Manifesto. (1993). International software engineering research network. Retrieved February 21, 2019, from <https://isern.iese.de/isern-manifesto>.
- Jacobs, R. A. (1994). The invisible workforce: How to align contract and temporary workers with core organizational goals. *National Productivity Review*, 13(2), 169-183.
- Jacobson, I., Booch, G., & Rumbaugh, J. (1998). *The unified software development process*. Massachusetts, USA: Addison-Wesley.
- Järvinen, P. (2007). Action research is similar to design science. *Quality & Quantity*, 41(1), 37-54.
- Jordan, P. J., & Lawrence, S. A. (2009). Emotional intelligence in teams: Development and initial validation of the short version of the workgroup

- emotional intelligence profile (WEIP-S). *Journal of Management & Organization*, 15(4), 452-469.
- Kennedy, M. N. (2003). *Product development for the lean enterprise: Why Toyota's system is four times more productive and how you can implement it*. Richmond, Virginia, USA: Oaklea Press.
- Kim, S., & Chung, Y. (2003). Critical success factors for IS outsourcing implementation from an interorganizational relationship perspective. *Journal of Computer Information Systems*, 43(4), 81-90.
- Kish, L. (1959). Some statistical problems in research design. *American Sociological Review*, 24(3), 328-338.
- Kniberg, H., & Ivarsson, A. (2012). Scaling agile with tribes, squads, chapters & guilds. Paper presented at the *London Lean Kanban Day 2013*. Retrieved April 21, 2018, from <http://www.bcs.org/content/conWebDoc/49927>.
- Kniberg, H., & Skarin, M. (2010). *Kanban and scrum: Making the most of both*. USA: C4Media, Publisher of infoQ.com.
- Kobitzsch, W., Rombach, D., & Feldmann, R. L. (2001). Outsourcing in india [software development]. *IEEE Software*, 18(2), 78-86.
- Kuechler, B., & Vaishnavi, V. (2008). On theory development in design science research: Anatomy of a research project. *European Journal of Information Systems*, 17(5), 489-504.
- Kupiainen, E., Mäntylä, M. V., & Itkonen, J. (2015). Using metrics in agile and lean software Development– A systematic literature review of industrial studies. *Information and Software Technology*, 62, 143-163.
- Lacity, M. C., & Hirschheim, R. (1995). *Beyond the information systems outsourcing bandwagon: The insourcing response*. USA: John Wiley & Sons.
- Lee, G., & Xia, W. (2010). Toward agile: An integrated analysis of quantitative and qualitative field data on software development agility. *MIS Quarterly*, 34(1).
- Lee, M. K. (1996). IT outsourcing contracts: Practical issues for management. *Industrial Management & Data Systems*, 96(1), 15-20.
- Lehman, M. M. (1980). Programs, life cycles, and laws of software evolution. *Proceedings of the IEEE*, 68(9), 1060-1076.

- Lehtinen, T. O., Itkonen, J., & Lassenius, C. (2017). Recurring opinions or productive improvements— What agile teams actually discuss in retrospectives. *Empirical Software Engineering*, 22(5), 2409-2452.
- Leopold, K., & Kaltenecker, S. (2015). *Kanban change leadership: Creating a culture of continuous improvement*. USA: Wiley Online Library.
- Liker, J. K. (2004). *The Toyota way: 14 management principles from the world's greatest manufacturer*. McGraw-Hill Education.
- Loh, L. W., & Venkatraman, N. (1991). *'Outsourcing' as a mechanism of information technology governance: A test of alternative diffusion models*. Cambridge, Massachusetts, USA: MIT Sloan School of Management.
- Marcelo García, C. (1992). *El estudio de casos: Una estrategia para la formación del profesorado y la investigación didáctica*. Secretariado de Publicaciones de la Universidad de Sevilla.
- Marcos, E., & Marcos, A. (1999). An aristotelian approach to the methodological research: A method for data model construction. Paper presented at the *4th UKAIS Conference*, University of York, 532-543.
- Martínez, O. P. (2009). *Emociones colectivas: La inteligencia emocional de los equipos*. Grupo Planeta (GBS).
- Matteson, S. (2018). Report: Software failure caused \$1.7 trillion in financial losses in 2017. Retrieved March 15, 2020 from <https://www.techrepublic.com/article/report-software-failure-caused-1-7-trillion-in-financial-losses-in-2017>.
- McHugh, O., Conboy, K., & Lang, M. (2011). Agile practices: The impact on trust in software project teams. *IEEE Software*, 29(3), 71-76.
- Mens, T. (2008). Introduction and roadmap: History and challenges of software evolution. *Software evolution* (pp. 1-11). Springer, Berlin, Heidelberg.
- Merriam, S. B. (1998). *Qualitative research and case study applications in education*. San Francisco, USA: Jossey-Bass.
- Merton, R. K. (1987). The focussed interview and focus groups: Continuities and discontinuities. *The Public Opinion Quarterly*, 51(4), 550-566.

- Middleton, P., & Joyce, D. (2011). Lean software management: BBC worldwide case study. *IEEE Transactions on Engineering Management*, 59(1), 20-32.
- Middleton, P., & Sutton, J. (2005). *Lean software strategies: Proven techniques for managers and developers*. United States: CRC Press.
- Moen, R. D., & Norman, C. L. (2009). The history of the PDCA cycle. *Proceedings from the Seventh Asian Network for Quality Congress*, Tokyo.
- Nadler, R. (2017, July 6). 10 reasons why teams need emotional intelligence. Actions to raise the EI of your team. Retrieved October 23, 2018, from <https://www.psychologytoday.com/intl/blog/leading-emotional-intelligence/201707/10-reasons-why-teams-need-emotional-intelligence>.
- Naur, P., & Randell, B. (1969). Software Engineering: Report of a conference sponsored by the NATO Science Committee, Garmisch, Germany, 7th-11th October 1968.
- Nikitina, N., Kajko-Mattsson, M., & Stråle, M. (2012). From Scrum to Scrumban: A case study of a process transition. Paper presented at the *2012 International Conference on Software and System Process (ICSSP)*, 140-149.
- Nord, R. L., Ozkaya, I., & Sangwan, R. S. (2012). Making architecture visible to improve flow management in lean software development. *IEEE Software*, 29(5), 33-39.
- Osborn, A. (2012). *Applied imagination-principles and procedures of creative writing*. Read Books Ltd.
- Parise, S., Cross, R., & Davenport, T. H. (2006). Strategies for preventing a knowledge-loss crisis. *MIT Sloan Management Review*, 47(4), 31.
- Peffer, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of Management Information Systems*, 24(3), 45-77.
- Perry, D., Porter, A., & Votta, L. (May 1, 2000). Empirical studies of software engineering. Paper presented at the *International Conference on Software Engineering*, Limerick, Ireland. 345-355.
- Poppendieck, M., & Poppendieck, T. (2007). *Implementing lean software development: From concept to cash*. Boston, USA: Pearson Education.

- Pritchard, R. D., Jones, S. D., Roth, P. L., Stuebing, K. K., & Ekeberg, S. E. (1988). Effects of group feedback, goal setting, and incentives on organizational productivity. *Journal of Applied Psychology*, 73(2), 337.
- Rand, G. K. (2000). Critical chain: The theory of constraints applied to project management. *International Journal of Project Management*, 18(3), 173-177.
- Reinertsten, D. G. (2009). *The principles of product development flow: Second generation lean product development*. California, USA: Celeritas.
- Robson, C. (2002). *Real world research: A resource for social scientists and practitioner-researchers* (Vol. 2). Oxford: Blackwell.
- Rose, K. H. (2013). A guide to the project management body of knowledge (PMBOK® guide) –Fifth edition. *Project Management Journal*, 3(44).
- Roth, G. (2006). Distributing leadership practices for lean transformation. *Reflections*, 7(2), 15-29.
- Rowley, J. (2002). Using case studies in research. *Management Research News*, 25(1), 16-27.
- Royce, W. W. (1987). Managing the development of large software systems: Concepts and techniques. In *Proceedings of the 9th International Conference on Software Engineering* (328-338).
- Runeson, P., & Höst, M. (2009). Guidelines for conducting and reporting case study research in software engineering. *Empirical Software Engineering*, 14(2), 131.
- Rutherford, K., Shannon, P., Judson, C., & Kidd, N. (2010). From chaos to kanban, via scrum. Paper presented at the *International Conference on Agile Software Development*, 344-352.
- Sample, J. A. (1984). Nominal group technique: An alternative to brainstorming. *Journal of Extension*, 22(2), 1-2.
- Schmidt, E., & Rosenberg, J. (2014). *How google works*. Great Britain: Hachette UK.
- Sharpe, M. (1997). Outsourcing, organizational competitiveness, and work. *Journal of Labor Research*, 18(4), 535-549.
- Simon, H. A. (2019). *The sciences of the artificial*. MIT press.

- Šmite, D., Wohlin, C., Gorschek, T., & Feldt, R. (2010). Empirical evidence in global software engineering: A systematic review. *Empirical Software Engineering, 15*(1), 91-118.
- Smith, A. P., & Smith, H. (2017). An international survey of the wellbeing of employees in the business process outsourcing industry. *Psychology, 8*(01), 160-167.
- Soy, S. (2015). *The case study as a research method*. University of Texas, Graduate School of Library and Information Science.
- Sparrow, E. (2012). *Successful IT outsourcing: From choosing a provider to managing the project*. London, UK: Springer Science & Business Media.
- Stake, R. E. (1995). *The art of case study research*. Sage.
- Stellman, A., & Greene, J. (2014). *Learning agile: Understanding scrum, XP, lean, and kanban*. O'Reilly Media, Inc.
- Suarez, B., & Miguel, J. (2009). In search of an area of sustainability: An empirical study of the implementation of the continuous improvement process in spanish town councils. *INNOVATE Journal of Administrative and Social Sciences, 19*(35), 47-64.
- Sumner, M., & Molka-Danielsen, J. (2010). Global IT teams and project success. In *Proceedings of the 2010 Special Interest Group on Management Information System's 48th Annual Conference on Computer Personnel Research*, Vancouver, Canada. 34-42.
- Susman, G. I., & Evered, R. D. (1978). An assessment of the scientific merits of action research. *Administrative Science Quarterly, 58*2-603.
- Taylor, D. W., Berry, P. C., & Block, C. H. (1958). Does group participation when using brainstorming facilitate or inhibit creative thinking? *Administrative Science Quarterly, 23*-47.
- Verner, J. M., Overmyer, S. P., & McCain, K. W. (1999). In the 25 years since the mythical man-month what have we learned about project management? *Information and Software Technology, 41*(14), 1021-1026.
- Wieringa, R., & Morali, A. (2012). Technical action research as a validation method in information systems design science. Paper presented at the *International Conference on Design Science Research in Information Systems*, 220-238.

- Wilson, G. (1999). Is the open-source community setting a bad example? *IEEE Software*, 16(1), 23-25.
- Wolff, S. B. (2017). *Bringing the team EI survey in-house accreditation program*. Massachusetts: GEI Partners. Retrieved January 13, 2019, from http://www.profwolff.org/GEIPartners/index_files/Articles/GEI%20Technical%20Manual.pdf
- Womack, J. P., & Jones, D. T. (1996). *Lean thinking*. New York, USA: Simon & Schuster.
- Womack, J. P., Jones, D. T., & Roos, D. (1990). *Machine that changed the world*. New York, USA: Simon and Schuster.
- Wood-Harper, T. (1985). Research methods in information systems: Using action research. *Research Methods in Information Systems*, 169, 191.
- Yazan, B. (2015). *Three approaches to case study methods in education: Yin, merriam, and stake*. Florida, USA: Nova Southeastern University, Inc.
- Yin, R. (2017). *Case study research and applications: Design and methods*. Sage publications.
- Yin, R. (1981). The case study as a serious research strategy. *Science Communication*, 3(1), 97-114.
- Yin, R. (1981). The case study crisis: some answers. *Administrative science quarterly*, 26(1), 58-65.
- Zalewski, J. M. (2018). *Working lives and in-house outsourcing: Chewed-up by two masters*. London, UK: Routledge.
- Zelkowitz, M. V., & Wallace, D. R. (1998). Experimental models for validating technology. *Computer*, 31(5), 23-31.

Acrónimos

Tabla de Acrónimos

Acrónimo	Descripción
ALM	Application Lifecycle Management
ANS	Acuerdos de Nivel de Servicio
CMMI	Capability Maturity Model Integration
DoD	Departamento de Defensa de los Estados Unidos
DSRM	Design Science Reseach Methodology
ETeC	Emotional Team Coaching process
ETeC	Emotional Team Coching process
FCE	Factores Críticos de Éxito
ISERN	International Software Engineering Research Network
OTAN	Organización del Tratado del Atlántico Norte
PDSA	Plan-Do-Study-Act
TEI	Inteligencia Emocional Grupal
TGN	Técnica de Grupo Nominal
TOC	Theory of Constraints
ToQ	Clase de Servicio Técnica o de Calidad
TPS	Toyota Production System
WEIP-S	Work Group Emotional Intelligence Profile - Scale
WIP	Work-In-Progress

