



UNIVERSIDAD NACIONAL  
AUTÓNOMA DE  
MÉXICO

**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**  
**Posgrado en Ciencias e Ingeniería de la Computación**

**“Integración de métodos ágiles a procesos de desarrollo de software en una empresa de nivel 5 CMMI-DEV: Caso de Estudio”.**

**TRABAJO DE TESIS  
QUE PARA OPTAR POR EL GRADO DE:  
MAESTRO EN INGENIERÍA (COMPUTACIÓN)**

**PRESENTA:  
GABRIELA ARAUZ ORTIZ**

**TUTORA DRA. HANNA JADWIGA OKATABA**  
**Facultad de Ciencias UNAM, Departamento de Matemáticas**

**CO-TUTORA MTRA. ELSA RAMIREZ HERNANDEZ**  
**Posgrado en Ciencias e Ingeniería de la Computación**

**MÉXICO, D. F. A JUNIO DEL 2013**



Universidad Nacional  
Autónoma de México

Dirección General de Bibliotecas de la UNAM

**Biblioteca Central**



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

## AGRADECIMIENTOS

*Primeramente agradezco a Dios por iluminar mi camino y darme la fortaleza para iniciar y completar este anhelo.*

*Al CONACYT por la confianza para fortalecer y concluir exitosamente mi formación profesional.*

*A mi padre <sup>f</sup> por el apoyo incondicional a mis sueños.*

*A mi hermoso esposo Julio Cesar por su amor infinito, comprensión, consejos, acompañamiento y el calor de sus brazos que confortan mi corazón y mi alma.*

*A mi generosa madre por convertirse en mi mejor amiga y cuidar de mi familia.*

*A mi preciosa hija Anita por ayudarme a madurar y brindarme el ejemplo de su fortaleza.*

*A mi linda y dulce hija Jully por su amor, ternura y apapachitos, que me han ayudado a salir adelante en momentos difíciles.*

*A mis entrañables amigos Araceli y Andrés, por su apoyo y ayuda al emprender esta maravillosa aventura que tanto anhelaba.*

*A mis profesores de la gloriosa Facultad de Ingeniería Mtro. Lauro Santiago, Mtro. Larry Escobar y Mtro. Enrique Arenas, por sus hermosas palabras de aliento, confianza y apoyo.*

*A la Dra. Hanna por compartir su conocimiento, experiencia y humanidad durante mi trabajo de tesis, por dejar huella en nosotros como sus alumnos y ser un gran ejemplo a seguir.*

*A la Mtra. Elsa por compartir su experiencia y permitirme conocer su maravilloso mundo.*

*A Fernando Rueda, Juan Carlos Barrientos, Carlos García, Cristina Dávila, Javier Flores, Rubén Murguía y Maribel pertenecientes al equipo de trabajo de Praxis por apoyarme en la recolección de la información, compartir su experiencia y brindarme una parte de su tiempo y dedicación.*

*A la Mtra. Lupita Ibarguengoitia por su apoyo, orientación y consejos para encontrar mi camino.*

*Al Mtro. Gustavo y al Dr. Fernando por sus correcciones, mejorando mi trabajo de investigación.*

*A Lulú, Amalia, Cecilia, Diana y al Dr. Arámbula por su confianza, apoyo, comprensión y hermosas palabras que impulsaron día a día el duro y bello camino de la maestría.*

*A mis compañeros porque juntos luchamos por alcanzar una meta común y compartimos sueños, esperanzas, crecimientos y pesares.*

*A cada uno de los miembros de mi familia que esperaron pacientemente durante mi ausencia y que recibieron con mucho cariño mi regreso.*

## **DEDICATORIA**

Dedico mis estudios de Maestría a mi padre †, porque él fue la inspiración y estímulo de seguir estudiando. Así como también a mis hijas, para que nunca dejen de luchar por sus sueños y anhelos, porque la mayor fuerza es la que pueden encontrar en sí mismas.

El trabajo de tesis se lo dedicó a mi esposo Julio César, porque sin su apoyo no hubiera concluido mi objetivo, por estar al pie del cañón al lado de mis hermosas hijas y mantener unida a mi pequeña familia. Esta dedicación se extiende a mi madre, por cuidar y amar a mi familia mientras me esmeraba por alcanzar mis metas.

“Yo no estimo tesoros ni riquezas,  
y así, siempre me causa más contento  
poner riquezas en mi entendimiento  
que no mi entendimiento en las riquezas.”

**Sor Juana Inés de la Cruz**

# INDICE

Capítulo 1.....	10
1.1. Objetivo de la investigación .....	11
1.2. Objetivos Específicos .....	11
1.3. Metodología para la investigación.....	12
1.4. Relevancia y contribución del trabajo.....	12
1.5. Breve descripción de cada capítulo .....	12
Capítulo 2.....	14
2.1. Desarrollo de software internacional.....	14
2.2. Evaluaciones CMMI-DEV Ver 1.3 a nivel mundial.....	15
2.3. Experiencias de integración de técnicas ágiles con CMMI-DEV Nivel 5 .....	17
2.3.1. Empresas de Desarrollo de Software Offshore .....	17
2.3.2. Empresa mexicana CMMI-DEV Nivel 5 adopta ágiles .....	18
2.4. Empresa Px.....	18
2.4.1. Arquitectura de procesos CMMI en Px.....	20
Capítulo 3.....	23
3.1. CMMI-DEV .....	23
3.1.1 Componentes y representación de CMMI-DEV .....	24
3.1.2 Nivel de madurez .....	25
3.1.2.1 Nivel Inicial .....	25
3.1.2.2 Nivel Gestionado .....	25
3.1.2.3 Nivel Definido.....	26
3.1.2.4 Nivel Gestionado Cuantitativamente.....	26
3.1.2.5 Nivel Optimizado .....	26
3.2 RUP .....	27
3.2.1 Estructura de RUP .....	27
3.2.2 Ciclo de vida de RUP .....	28
3.2.2.1 Fase Inception .....	28

3.2.2.2	Fase Elaboration .....	29
3.2.2.3	Fase Construction.....	29
3.2.2.4	Fase Transition.....	29
Capítulo 4.....		30
4.1.	Scrum.....	30
4.1.1.	Equipo Scrum.....	31
4.1.1.1.	Dueño del Producto o Product Owner.....	31
4.1.1.2.	Equipo de Desarrollo o Development Team.....	31
4.1.1.3.	Scrum Master .....	31
4.1.2.	Eventos Scrum.....	32
4.1.2.1.	Sprint.....	32
4.1.2.2.	Reunión de Planeación de Sprint o Sprint Planning Meeting.....	32
4.1.2.3.	Scrum Diario o Daily Scrum .....	32
4.1.2.4.	Reunión de Revisión de Sprint o Sprint Review .....	33
4.1.2.5.	Reunión de Retrospectiva o Sprint Retrospective.....	33
4.1.3.	Artefactos de Scrum.....	33
4.1.3.1.	Product Backlog.....	33
4.1.3.2.	Sprint Backlog .....	34
4.2.	Extreme Programming.....	34
4.2.1.	Características de XP durante el desarrollo.....	35
4.2.2.	Valores .....	35
4.2.3.	Principios.....	36
4.2.4.	Prácticas.....	36
4.3.	Kanban.....	37
4.3.1.	Forma de trabajar con Kanban.....	37
Capítulo 5.....		39
5.1.	Metodología de caso de estudio.....	39
5.1.1.	Fase 1 – Diseño de la investigación .....	40
5.1.2.	Fase 2 – Colección de datos .....	41

5.1.3.	Fase 3 – Análisis e Interpretación.....	43
5.1.4.	Fase 4 – Elaboración de reporte para su difusión.....	43
5.2.	Estrategias de investigación.....	44
5.3.	Perspectiva de la investigación.....	45
5.4.	Características de la investigación.....	45
5.5.	Criterios de calidad y cuidados éticos de la investigación.....	47
Capítulo 6.....		49
6.1.1.	Motivación.....	49
6.1.2.	Objetivo de la investigación .....	49
6.1.3.	Descripción de la investigación .....	50
6.1.4.	Relevancia de la investigación .....	50
6.1.5.	Preguntas de investigación .....	50
6.1.6.	Hipótesis de la investigación.....	51
6.2.	Unidades de Análisis .....	52
6.2.1.	Unidad I: Técnicas ágiles integradas con CMMI-DEV.....	53
6.2.2.	Unidad II: Métricas de proyectos para determinar beneficios .....	54
6.2.3.	Unidad III: Adaptación del grupo de trabajo .....	54
Capítulo 7.....		56
7.1.	Unidad de análisis I: Técnicas ágiles integradas con CMMI-DEV .....	56
7.2.	Unidad de análisis II: Métricas de proyectos para determinar beneficios .....	59
7.3.	Unidad de análisis III: Adaptación del grupo de trabajo .....	61
Capítulo 8.....		62
8.1.	Unidad de análisis I: Técnicas ágiles integradas con CMMI-DEV .....	62
8.1.1.	Codificación de entrevistas .....	63
8.1.2.	Triangulación de las entrevistas.....	64
8.2.	Unidad de análisis II: Métricas de proyectos para determinar beneficios.....	64
8.2.1.	Elementos de una gráfica para medir un indicador.....	65
8.2.2.	Mejora de velocidad de implementación.....	66

8.2.3.	Disminución de tiempo en elaboración de documentación.....	68
8.2.4.	Disminución de defectos en el producto final de un proyecto .....	69
8.3.	Unidad de análisis III: Adaptación del grupo de trabajo.....	70
Capítulo 9.....		72
9.1.	Unidad de análisis I: Técnicas ágiles integradas con CMMI-DEV .....	72
9.1.1.	Desarrollo de software usando metodología RUP con CMMI-DEV .....	73
9.1.2.	Integración de técnicas ágiles con CMMI-DEV.....	76
9.1.3.	Comprobación de la hipótesis de la unidad de análisis I .....	88
9.2.	Unidad de análisis II: Métricas de proyectos para determinar beneficios.....	89
9.2.1.	Comprobación de la hipótesis de la unidad de análisis II .....	92
9.3.	Unidad de análisis III: Adaptación del grupo de trabajo.....	94
9.3.1.	Comprobación de la hipótesis de la unidad de análisis III .....	105
Capítulo 10 .....		108
Beneficios y áreas de oportunidad.....		108
10.1.	Beneficios .....	108
10.1.1.	Enfoque en producto y no en documentación.....	109
10.1.2.	Enfoque en la recolección de requerimientos .....	110
10.1.3.	Selección de las tareas a realizar .....	110
10.1.4.	Descripción de tareas.....	111
10.1.5.	Mayor control de actividades .....	111
10.1.6.	Corrección oportuna de defectos .....	112
10.1.7.	Aumento de productividad.....	113
10.1.8.	Conservar la calidad.....	114
10.1.9.	Mejora profesional de los integrantes del equipo de trabajo .....	114
10.1.10.	Mejoras personales de los integrantes.....	115
10.1.11.	Mejoras en el ambiente de trabajo.....	115
10.1.12.	Re-certificación en Nivel 5 de CMMI-DEV .....	116
10.1.13.	Satisfacción del cliente.....	116
10.2.	Áreas de oportunidad.....	117

10.2.1. Mejorar el proceso de cambios empleando técnicas ágiles .....	117
10.2.2. Desventajas para los integrantes del equipo.....	117
Capítulo 11 .....	118
Conclusiones.....	118
11.1. Debilidades de la investigación .....	119
11.2. Trabajo futuro .....	120
Apéndice A .....	121
Apéndice B .....	122
Apéndice C .....	123
Apéndice D .....	125
Apéndice E .....	126
Apéndice F.....	127
Bibliografía.....	128

# Capítulo 1

## Introducción

Las empresas líderes en desarrollo de software buscan colocarse en el mercado nacional e internacional alcanzando un alto rendimiento en sus proyectos con herramientas robustas, consistentes y eficientes para aumentar la producción de software de calidad [1].

En el desarrollo de software, los modelos de procesos sirven de guía para establecer un modo de trabajo con prácticas y técnicas robustas, mejorando el desempeño del grupo en los proyectos. Por su parte, las metodologías y métodos para desarrollar software establecen la estructura, herramientas a utilizar, planeación y control de las actividades y del personal que participa en un proyecto.

Los modelos CMMI®<sup>1</sup> son colecciones de buenas prácticas para direccionar el desarrollo y mantenimiento de productos y servicios en su ciclo de vida [2]. CMMI proporciona un enfoque para mejorar procesos de una organización [3].

El modelo CMMI para Desarrollo denominado CMMI-DEV®<sup>2</sup>, proporciona un conjunto completo e integrado de guías para desarrollar productos y sistemas de software [2].

---

<sup>1</sup> CMMI es una marca registrada por el Software Engineering Institute (SEI) de la Carnegie Mellon University en Estados Unidos de Norteamérica. A lo largo de toda la tesis se hace mención a CMMI, por lo que en las siguientes apariciones se omitirá el símbolo de Derechos Reservados ®.

Los métodos ágiles son una forma de desarrollar software funcional, valorando a las personas, respondiendo ante el cambio y con trabajo colaborativo [4].

Al integrar métodos ágiles con CMMI-DEV, permiten aprovechar una combinación innovadora de dos técnicas aparentemente contrarias para obtener beneficios y aprovechar la sinergia que tienen mejorando la productividad del negocio [5].

## **1.1. Objetivo de la investigación**

El objetivo de la investigación es realizar un caso de estudio dentro de una empresa que integró técnicas ágiles con CMMI-DEV Nivel 5; con la finalidad de documentar beneficios cuantitativos y cualitativos en el desempeño de los proyectos bajo esta forma de trabajar el desarrollo de software.

## **1.2. Objetivos Específicos**

Durante la investigación se perseguirán objetivos específicos que permitirán alcanzar el objetivo principal de la misma investigación y delimitarán el alcance del caso de estudio. Los objetivos específicos son:

- Identificar las técnicas de los métodos ágiles y la manera de cómo se integraron con CMMI-DEV Nivel 5 en la empresa Px.
- Realizar la comparación de métricas de productividad y calidad en proyectos que utilizaron metodología RUP con CMMI-DEV contra los que integraron técnicas ágiles con CMMI-DEV.
- Identificar por medio de testimonios, los beneficios del uso de técnicas ágiles en la mejora del desempeño laboral de los participantes y la integración de los equipos de trabajo en los proyectos.
- Identificar y sugerir mejoras a los procesos de trabajo establecidos.

---

<sup>2</sup> CMMI-DEV es una marca registrada por el SEI. A lo largo de todo el trabajo de tesis se hace mención a CMMI-DEV, por lo que en lo sucesivo se omitirá el símbolo de Derechos Reservados ®.

### **1.3. Metodología para la investigación**

La investigación se realizará con la estrategia comparativa Caso de Estudio, para indagar la forma de trabajo y conocer la manera como se llevó a cabo la integración de técnicas ágiles con CMMI-DEV dentro de una empresa de desarrollo de software CMMI-DEV con Nivel 5.

### **1.4. Relevancia y contribución del trabajo**

Los resultados de la investigación proporcionarán un ejemplo documentado de un caso de estudio que presenta los beneficios de incorporar técnicas ágiles con CMMI-DEV en una empresa con Nivel 5, conservando la calidad y aumentando la productividad de la empresa.

Este ejemplo puede servir de referencia a empresas mexicanas que opten por integrar técnicas ágiles a su forma de trabajo empleando procesos innovadores con CMMI-DEV, o bien, a empresas que inicien el trabajo con procesos y quieran desarrollar software con métodos ágiles con un modelo evaluable.

### **1.5. Breve descripción de cada capítulo**

La tesis se encuentra dividida en 11 capítulos, cuyo contenido se describe a continuación:

El capítulo 1 **Introducción**, menciona el tema de tesis, motivos para elegir este tema de tesis, objetivo principal y objetivos específicos que guiarán la investigación, metodología de investigación a utilizar y la relevancia que podría tener la investigación en empresas que desarrollan software.

El capítulo 2 **Contexto del caso de estudio**, describe el desarrollo de software en los países con mayor reconocimiento en el área, muestra el conteo de empresas evaluadas por país en CMMI-DEV Nivel 5 Ver 1.3 a nivel internacional y Latinoamérica, experiencias de empresas que han desarrollado software empleando técnicas ágiles con CMMI-DEV Nivel 5 y presentación de la empresa donde se realizará la investigación del caso de estudio.

El capítulo 3 **Modelos de referencia estructurados**, describe los aspectos teóricos de los modelos de referencia relacionados al tema de la investigación, tal como CMMI-DEV y RUP<sup>3</sup>.

El capítulo 4 **Métodos ágiles**, describe los métodos ágiles utilizados para seleccionar las técnicas ágiles empleadas en la integración con CMMI-DEV de la empresa estudiada.

El capítulo 5 **Metodología de Investigación**, detalla la estrategia metodológica que sigue la investigación del caso de estudio, breve descripción de las fases de la metodología, estrategia, perspectiva y características de la investigación.

El capítulo 6 **Diseño del Caso de Estudio**, muestra el diseño de la investigación, características, hipótesis, unidades de análisis, así como las preguntas de investigación que guiaron el camino del caso de estudio.

El capítulo 7 **Colección de datos**, detalla las técnicas utilizadas para recabar los datos en cada una de las unidades de análisis de investigación, criterios considerados de selección de las personas que participan en la investigación y la forma como se almacenó la información.

El capítulo 8 **Análisis de datos**, presenta las técnicas para analizar los datos y evidencias recolectadas para cada unidad de análisis.

El capítulo 9 **Interpretación**, muestra los resultados de la investigación, responde las preguntas de investigación y determina si las hipótesis iniciales son aceptadas o rechazadas.

El capítulo 10 **Beneficios y áreas de oportunidad**, extraen los beneficios e identifica las áreas de oportunidad de los resultados de la investigación.

En el capítulo 11 **Conclusiones** expresa las conclusiones del trabajo de investigación a partir de los resultados obtenidos, debilidades de la investigación y trabajo a futuro.

En los apéndices se muestra información complementaria a la investigación que puede resultar de interés al lector.

---

<sup>3</sup> Proceso de desarrollo de software desarrollado por la empresa Rational® Software propiedad de IBM®.

# Capítulo 2

## Contexto del caso de estudio

Este capítulo describe el estado del desarrollo de software a nivel mundial, países que han tenido buenos resultados en exportación de software, ubicación de México a nivel mundial y de América Latina en evaluaciones CMMI-DEV Nivel 5 Ver 1.3, incluye casos reales de empresas con CMMI-DEV Nivel 5 que han integrado técnicas ágiles a su forma de trabajo y finalmente describe la empresa donde se realizará el trabajo de investigación para el caso de estudio.

### 2.1. Desarrollo de software internacional

El progreso tecnológico es uno de los principales motores que han transformado el mundo, marcando un capitalismo informático mundial [6]. Los cambios tecnológicos se han extendido rápidamente y el software juega un papel estratégico en la industria de todos los países para manejar información [7], aumentando la dependencia tecnológica y en sistemas informáticos [8].

A nivel mundial, Estados Unidos ha impulsado la industria de software [7], al generar productos innovadores y otros desarrollos apoyados por el gobierno para resguardar la seguridad nacional [9]. Japón destaca por aplicaciones en la industria pesada y adherirse con fábricas nacionales de computadoras, en tanto China exporta software principalmente a Japón [9].

Algunas naciones de reciente industrialización como India, Israel e Irlanda han destacado en producción de software e inserción en otras naciones [9]. Pocos países, entre ellos India han implementado programas exitosos de calidad basados en estándares internacionales [7] y es pionero en exportar servicios informáticos [9].

Irlanda desarrolla y traduce exitosamente software a la Unión Europea gracias a sus políticas industriales [9], mientras Israel exporta software a Estados Unidos y Europa Occidental [10].

Países como Brasil, Argentina, México, Uruguay y Costa Rica se encuentran en proceso de maduración con crecimiento constante [7]. Estos países latinoamericanos han producido software a nivel internacional con sus propias limitaciones, gracias a sus capacidades, infraestructura [7] y apoyos del gobierno para impulsar la industria de software [7] [9].

Brasil y Argentina se han centrado en el mercado interno y recientemente han exportado software. A diferencia Uruguay y Costa Rica se han centrado en exportar software, dejando a un lado el mercado interno [7]. El desarrollo de la industria de software en México ha destacado en América Latina por ser el más dinámico en tecnología de la información y conocimiento, aunque todavía es incipiente [7].

## **2.2. Evaluaciones CMMI-DEV Ver 1.3 a nivel mundial**

La competencia aumenta en la industria de software, forzando a las empresas a mejorar sus prácticas y procesos de desarrollo de software [8]. Seguir un modelo de calidad basado en buenas prácticas y principios, permite a la industria de software aumentar la productividad y calidad del producto final administrando el tiempo y presupuesto establecidos [11] [12]. Para acotar la investigación, nos enfocaremos en el modelo de madurez CMMI-DEV de Nivel 5.

Implementar un modelo de calidad requiere de invertir tiempo, dinero y recursos, disminuyendo la cantidad de organizaciones capaces de incorporarlos [13] [14]. Bajo estas consideraciones se presenta una breve investigación de los países con mayor número de empresas evaluadas con CMMI-DEV Ver 1.3 Nivel 5, utilizando dos referencias de diferentes años.

La empresa Px obtuvo la evaluación CMMI-DEV Nivel 5 Versión 1.3 en diciembre del 2012, por lo que se seleccionaron periodos de medición para evaluaciones del 2013 y 2014.

Se considera primero el número de empresas evaluadas al finalizar el año 2013, según el artículo “Ranking mundial en certificaciones CMMI-DEV ver. 1.3 año 2013” de la “Revista Iberoamericana de Producción Académica y Gestión Educativa” [15]. Posteriormente, se presenta el número de empresas evaluadas con CMMI-DEV Ver 1.3 Nivel 5 al 9 de Septiembre del 2014, extraídas de la página oficial “CMMI Institute, Published Apraisal Results” [16]. Los países evaluados CMM-DEV Nivel 5 Ver 1.3 se encuentran en el Apéndice A.

La Figura 2-1 muestra los países con mayor número de empresas evaluadas con CMMI-DEV Ver 1.3 Nivel 5, resaltando el aumento de más del 100% de evaluaciones del 2013 al 2014.

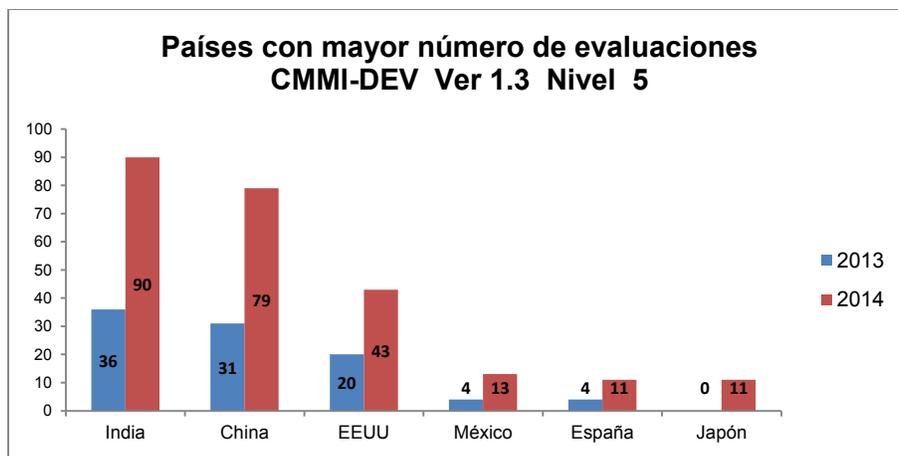


Figura 2-1 Países con el mayor número de empresas evaluadas CMMI-DEV Ver 1.3 Nivel 5 (Análisis estadístico de la implementación de buenas prácticas en organizaciones desarrolladoras de software, 2013) (Institute, 2014).

Podemos observar que India se mantiene a la vanguardia no solamente en la industria del software [7], sino también en 1er lugar en evaluaciones CMMI-DEV Ver 1.3 Nivel 5 obtenidas por empresas de todo el país [16]. Empresas de China y Estados Unidos están presentes en desarrollo de software [7] [9] y en evaluaciones con el modelo CMMI-DEV Ver 1.3 Nivel 5 [16]. México ocupa el 4to lugar a nivel mundial en empresas evaluadas pasando de solo 4 a 13 en un año y Japón logro evaluar 11 empresas con Nivel 5, colocándose en 5to lugar junto con España.

Grandes exportadores de software como Irlanda actualmente no cuentan con evaluaciones CMMI-DEV Nivel 5, por la poca importancia que da su mercado objetivo a esta evaluación [7]. A diferencia de Israel, que logró aumentar las empresas evaluadas con CMMI-DEV Ver 1.3 Nivel 5 en el año 2014 [11] [16]. Países Latinoamericanos con alguna evaluación CMMI-DEV Ver 1.3 Nivel 5 se muestran en la Figura 2-2.

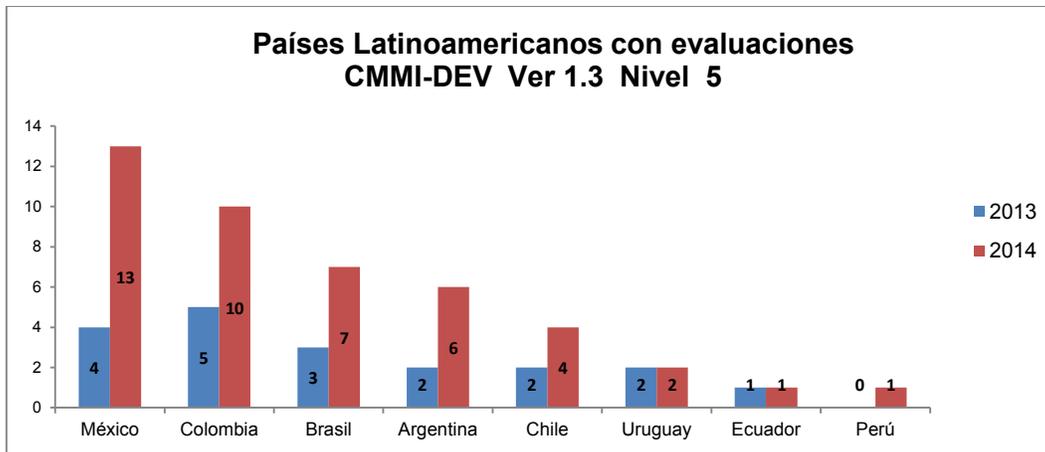


Figura 2-2 Países latinoamericanos con al menos una evaluación CMMI-DEV Ver 1.3 Nivel 5 (Análisis estadístico de la implementación de buenas prácticas en organizaciones desarrolladoras de software, 2013) (Institute, 2014).

México ocupa el 1er lugar. Brasil y Argentina han incrementado las empresas evaluadas, pero se colocan debajo de Colombia. En tanto, Uruguay y Ecuador se han mantenido constantes con sus empresas evaluadas, mientras Perú evaluó una empresa en el 2014.

## 2.3. Experiencias de integración de técnicas ágiles con CMMI-DEV Nivel 5

Los métodos ágiles han venido evolucionado [10] y se incorporan rápidamente a la industria de software por su flexibilidad y capacidad de respuesta [17]. CMMI-DEV incrementa el rendimiento organizacional al integrar métodos ágiles [18], centrado en los objetivos que se deben alcanzar y no en la forma de lograrlo [19]. Esta sección describe dos casos de empresas con evaluación CMMI-DEV Nivel 5 que integraron métodos ágiles para desarrollar software.

### 2.3.1. Empresas de Desarrollo de Software Offshore<sup>4</sup>

Este caso fue extraído del artículo “The impact of process choice in high maturity environments: An empirical analysis” [14]. Fue una investigación de tres años en dos empresas *offshore* con características similares. Ambas tenían evaluación CMMI Nivel 5 y un Grupo de Procesos de Ingeniería de Software (SEPG). La investigación abarcó 112 proyectos de inicio a fin.

<sup>4</sup> Empresas que desarrollan productos en un lugar diferente al origen de la empresa, para disminuir los costos de producción.

Cada equipo decidía si utilizaba métodos ágiles o no; las técnicas empleadas requerían aprobación del SEPG y el Grupo de Aseguramiento de la Calidad intervenía para corregir las desviaciones en los proyectos. Proyectos más pequeños, con clientes no involucrados o que requería nueva tecnología fueron más propensos a utilizar procesos ágiles.

El trabajo con métodos ágiles tuvo un impacto positivo en el desempeño de los proyectos aumentando la reutilización y productividad, disminuyó el retrabajo y la densidad de defectos. Ambas empresas estandarizaron e institucionalizaron los procesos con técnicas ágiles para una posterior evaluación con CMMI-DEV y re-certificación con Nivel 5.

### **2.3.2. Empresa mexicana CMMI-DEV Nivel 5 adopta ágiles**

Esta experiencia se extrajo del artículo “Applying Agile and Lean Practices in a Software Development Project into a CMMI Organization” [3]. Es una empresa con evaluación CMMI-DEV Nivel 5 Ver 1.2, que promueve reuniones internas, en búsqueda de nuevas propuestas para aumentar la agilidad sin perder la disciplina de los procesos.

La empresa integró procesos ágiles a lo largo de todo el ciclo de vida de dos desarrollos de software. Con Scrum<sup>5</sup> se administró el proyecto, aplicaron técnicas ágiles en sus prácticas y diseñaron plantillas para controlar todo el proceso.

En el artículo se analizan evidencias sobre cómo la empresa aumentó la satisfacción del cliente, mejoró la comunicación del equipo de trabajo, redujo el tiempo de entrega del producto y tuvo buena administración de cambios. El inconveniente fue la resistencia de los equipos de desarrollo a la forma de trabajo por la falta de conocimiento, lo que fue solucionado con capacitación.

## **2.4. Empresa Px**

La empresa Px es una organización reconocida en el área de Tecnologías de la Información, que desarrolla software con altos estándares de calidad y cuenta con buena posición en el mercado internacional.

---

<sup>5</sup> Marco de trabajo ágil para administrar el desarrollo del software eficientemente [25].

La empresa inició la adopción de modelos de calidad en enero del 2003, tal como CMMI-DEV, PMBoK<sup>6</sup> e ISO 9001. El objetivo era definir un modelo de operación basado en las mejores prácticas internacionales que le permitieran estandarizar los flujos de trabajo y obtener las ventajas prometidas por aquellas. Los procesos implementados por la empresa Px han utilizado dos ciclos de vida para desarrollar software: Modelo Cascada y Rational® Unified Process (RUP).

El Grupo de Procesos de la empresa Px está formado por expertos en Ingeniería de Procesos, Aseguramiento de la Calidad de Software, Ingeniería de Software, Pruebas, Administración de la Configuración, Estadística, Administración de Proyectos y de la Capacitación Organizacional quienes garantizan la calidad de los procesos.

El Centro de Excelencia para el Desarrollo de Software (CEDS) es un área de desarrollo de software de la empresa Px. El CEDS logró que en el año 2005, la empresa Px tuviera la evaluación CMM Nivel 3, en el 2007 alcanzó CMMI-DEV Nivel 4 y en el 2009 logró CMMI-DEV Versión 1.2 Nivel 5 por primera ocasión.

En el año 2010, para mejorar la productividad en los desarrollos de software e innovar los procesos, surgió la propuesta de incorporar métodos ágiles. Para tal fin, los expertos de Ingeniería de Procesos de la empresa Px participaron en cursos, conferencias, asesorías e investigación de métodos ágiles para fundamentar la propuesta de trabajo con estas técnicas.

El Plan de Mejora de la empresa Px para el año 2010 consideró la incorporación de técnicas ágiles en un proyecto piloto realizado en el área del CEDS. El éxito de los productos obtenidos en el proyecto piloto provocó que la empresa Px incorporara técnicas ágiles a sus procesos organizacionales.

En el año 2012 la empresa Px ratificó su evaluación CMMI-DEV Versión 1.3 Nivel 5 al innovar sus procesos con técnicas ágiles, lo que le ha permitido aumentar su participación en el mercado, aumentar la productividad manteniendo su nivel de calidad y mejorar los resultados del negocio. Fue de las primeras empresas en evaluarse con la Versión 1.3 de CMMI-DEV en México.

---

<sup>6</sup> Estándar internacional para la administración de proyectos, proporciona las prácticas fundamentales necesarias para lograr excelentes resultados organizacionales [70].

El modelo de calidad CMMI-DEV fue adoptado inicialmente para desarrollar productos de software nuevos y posteriormente dado el éxito de los resultados lo integro para el desarrollo de sistemas a la medida.

Hoy, la empresa Px cuenta con la implementación de técnicas ágiles como su tercer ciclo de vida para desarrollar software, donde las técnicas ágiles son utilizadas en entornos complicados para reducir tiempo de desarrollo y mejorar la productividad de los proyectos. Por esta razón, la presente investigación se realizará en la empresa Px, con el objetivo de documentar y compartir su experiencia exitosa al incorporar técnicas ágiles con procesos CMMI-DEV Nivel 5.

### **2.4.1.Arquitectura de procesos CMMI en Px**

La Arquitectura de Procesos usada en la empresa Px y utilizada para obtener su primer CMMI-DEV Nivel 5, se encuentra definida en un documento del activo organizacional de la empresa con la siguiente representación gráfica (Figura 2-3). Esta representación incluye tanto áreas de proceso de CMMI como procesos internos de la empresa Px.

La administración de los proyectos la cubrían con los procesos: Administración de Estrategias (SPP) y Administración del Rendimiento del Proyecto (SPTO), cubriendo además la Administración integrada del software (ISM) y la administración cuantitativa. Los procesos de desarrollo estaban considerados en el grupo de procesos Ingeniería de Producto de Software (SPE) considerando las disciplinas de RUP: Requerimientos, Análisis y diseño, Implementación, Pruebas y Despliegue.

Para realizar la integración de modelos ágiles, la arquitectura fue modificada y el documento del activo organizacional fue ajustado al ciclo de vida del proyecto (Figura 2-4). Se agregaron las alternativas de técnicas ágiles tanto en procesos de administración como ingeniería (desarrollo).

De acuerdo a las entrevistas, los procesos de administración fueron afectados en un 70%, ingeniería en un 50% y soporte un 30%. La siguiente figura muestra la arquitectura de procesos modificada.

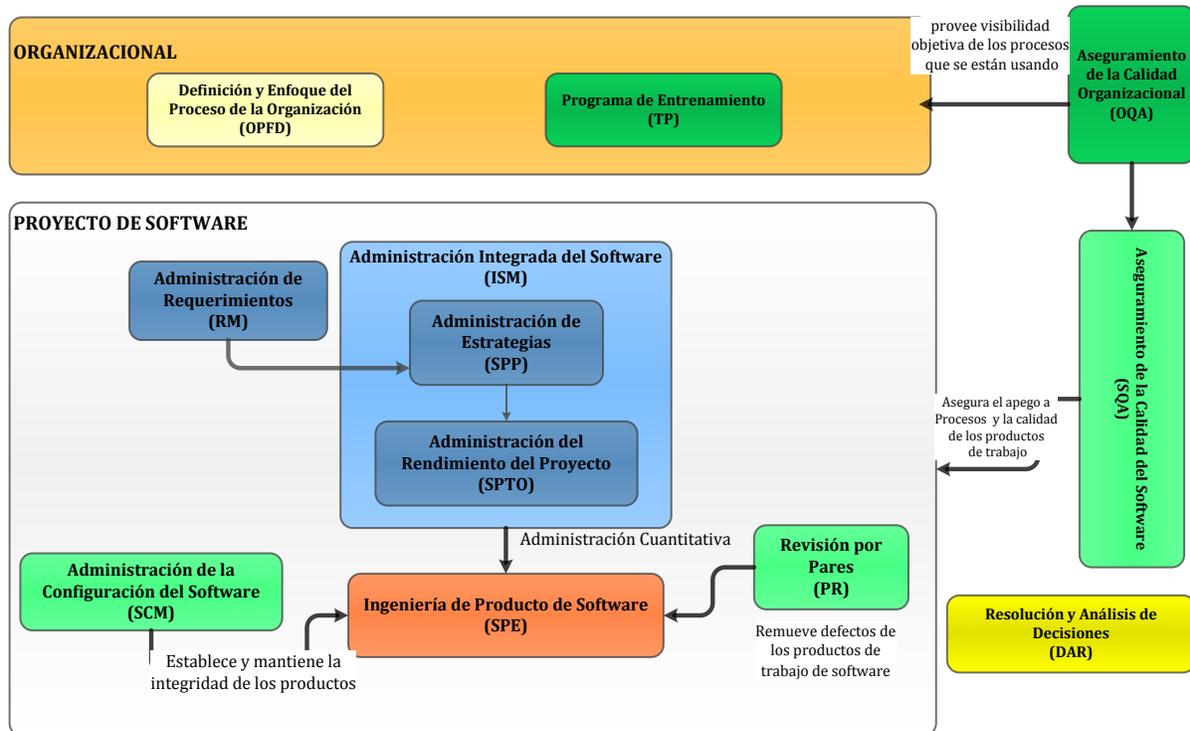


Figura 2-3 Arquitectura de procesos de Px al obtener la primera evaluación CMMI-DEV Nivel 5.

Scrum y Kanban contienen algunas de las mejores prácticas para cubrir el nivel de madurez 2 de CMMI. Principalmente las relacionadas con Administración de Requerimientos (RM), Planificación de Proyectos (PP) y Monitorización y Control del Proyecto (PMC).

Tanto XP como RUP, aplican prácticas para la Definición de Requerimientos (RD), monitoreo de prioridades en la Administración de Requerimientos (RM). FDD y TDD cubren varias actividades de las solicitadas para los procesos de ingeniería.

Los procesos de soporte como Aseguramiento de la Calidad del Proceso y del Producto (PPQA), y Administración de la Configuración (CM) sufren modificaciones, sin dejar de obtener ventajas con su implementación.

El Aseguramiento de la Calidad del Producto, se acopló el proceso de auditorías al dinamismo y practicidad para documentar y revisar los aspectos técnicos que usa el equipo de trabajo. La planeación de las revisiones se vió afectada también pues éstas ya no se realizan en periodos semanales o quincenales, se sugiere usar listas de verificación más simples aplicables diariamente.

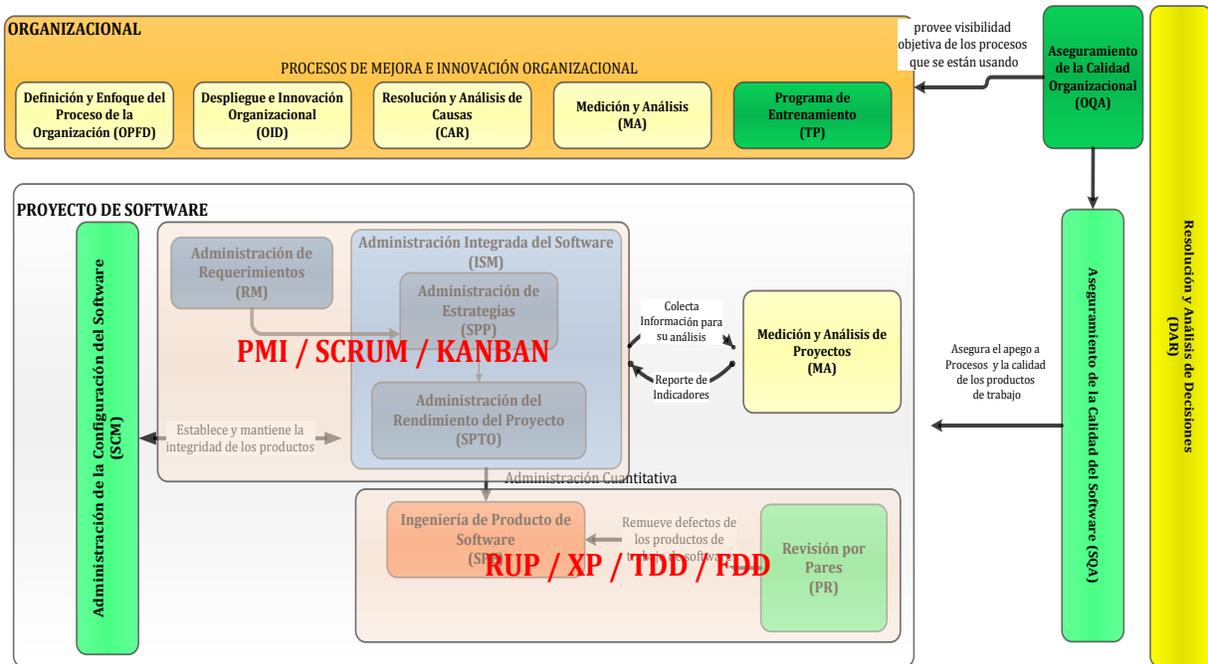


Figura 2-4 Arquitectura de procesos de Px al integrar métodos ágiles.

La Administración de la Configuración de Software se adaptó a la velocidad de desarrollo, implementando técnicas de integración continua que permitían ser más veloces en la realización de liberaciones parciales y formales diariamente.

Por los objetivos de la investigación de la tesis, el enfoque está dirigido a los métodos ágiles que se integraron para desarrollar software con procesos de CMMI-DEV.

## Capítulo 3

### Modelos de referencia estructurados

Los modelos de madurez, estándares y metodologías pueden ayudar a las empresas a mejorar el negocio, siendo CMMI quién cubre todo el ciclo de vida del producto desde su concepción hasta su entrega y mantenimiento [20]. Así mismo, RUP es una metodología clara que indica responsabilidades y guía el desarrollo de software sin importar el tamaño del proyecto, asegurando la calidad de los productos generados [21].

El presente capítulo muestra los modelos de referencia para trabajar estructuradamente el desarrollo de software: CMMI-DEV y RUP, ya que estos modelos eran la base del trabajo de la empresa Px que permitieron obtener resultados al integrar técnicas ágiles a CMMI-DEV en su forma de desarrollar. Por lo que, este trabajo no se indagará la alternativa del modelo cascada.

#### 3.1. CMMI-DEV

La información e imágenes de esta sección fueron extraídas de la página web del Software Engineering Institute<sup>7</sup> que proporciona información oficial de CMMI-DEV Ver 1.3 [20].

---

<sup>7</sup> Organismo internacional que promueve la mejora de procesos de la industria para producir productos y servicios de calidad.

Capability Maturity Model Integration (CMMI), es un modelo de mejora de procesos para desarrollar productos y servicios. Creado por el Software Engineering Institute (SEI) junto con expertos de la industria y del gobierno.

El Marco de CMMI proporciona una estructura para crear múltiples modelos y componentes de evaluación, contiene metas y prácticas para generar todos los modelos CMMI. Una constelación es una colección de componentes comunes CMMI y componentes de un área específica. CMMI cuenta con 3 constelaciones publicadas desarrollo, servicios y adquisiciones.

Por motivos de la investigación nos enfocaremos en la constelación de desarrollo denominada CMMI-DEV, que mejora procesos de desarrollo y servicios de software en las empresas [18].

### **3.1.1 Componentes y representación de CMMI-DEV**

Un área de proceso está formada por un conjunto de prácticas que satisfacen una serie de metas de mejora para el área de proceso. CMMI-DEV consta de 22 áreas de proceso, de las cuales 16 son áreas base<sup>8</sup>, 1 área de proceso compartida<sup>9</sup> y 5 áreas de proceso de desarrollo.

El cumplimiento de las metas genéricas<sup>10</sup> determinan la institucionalización de los procesos y las metas específicas las características únicas a cumplir por un área de proceso, ambas determinan si un área de proceso fue satisfecha. Las prácticas genéricas y específicas describen las actividades para lograr las metas correspondientes.

CMMI da soporte a dos representaciones de mejora de procesos, ambas representaciones mejoran los procesos para lograr los objetivos del negocio. La Representación por Etapas emplea niveles de madurez del 1 al 5, para alcanzar un nivel particular debe satisfacer todas las metas del conjunto de áreas de proceso asociadas al nivel.

Por cuestiones de la investigación la sección está centrada en la representación por etapas, ya que es la utilizada por la empresa Px.

---

<sup>8</sup> Una parte del material del área es el mismo en todas las constelaciones, la otra parte se ajusta para guiar el área específica de interés [20].

<sup>9</sup> Está presente en al menos dos modelos CMMI, pero no en todos [20].

<sup>10</sup> Son genéricas por aplicarse a múltiples áreas de proceso.

### 3.1.2 Nivel de madurez

Los niveles CMMI-DEV marcan el camino para la maduración de los procesos de una organización y mejorar los resultados de los proyectos al tomar decisiones con datos estadísticos de otros proyectos. El nivel de madurez indica el estado de los procesos de la organización, donde cada nivel trabaja un conjunto específico de áreas de proceso que se requieren implementar. Las áreas de proceso asignadas a cada nivel se encuentran en el Apéndice B.

Un nivel se adquiere al alcanzar las metas de las áreas de proceso correspondientes y como resultado de evaluar toda la organización o una sección de la misma.

Las áreas de proceso se dividen en 4 categorías, que dan soporte integral a CMMI-DEV:

- **Gestión de procesos.** Tiene actividades para mejorar calidad y rendimiento de procesos.
- **Gestión de proyectos.** Cubre actividades de planeación, monitoreo y control de proyectos.
- **Ingeniería.** Abarca estrategias de mejora de procesos orientada al producto.
- **Soporte.** Dan soporte a todas las demás áreas de proceso.

#### 3.1.2.1 Nivel Inicial

Primer nivel de CMMI-DEV, describe el perfil de cualquier empresa que desarrolle software:

- Los procesos son “ad-hoc” y caóticos, son abandonados durante las crisis.
- El éxito del proyecto depende de habilidades y acciones de empleados actuales.
- Los proyectos exceden frecuentemente presupuesto y tiempo esperado.

#### 3.1.2.2 Nivel Gestionado

El Nivel 2 de CMMI-DEV establece las primeras características a cumplir por una empresa:

- Visibilidad de productos por la Dirección, respeto a políticas del negocio.
- Monitoreo, cumplimiento y apego a procesos todo el tiempo.

- No hay estandarización en proceso y procedimientos.
- Empleados competentes y recursos disponibles para generar resultados controlados.
- Trabajo disciplinado con planes documentados y control de productos de trabajo.
- Interesados involucrados.

### 3.1.2.3 Nivel Definido

Corresponde al Nivel 3 de CMMI-DEV, describe una organización con procesos maduros con:

- Procesos estandarizados, detallados estrictamente, consistentes y en mejora constante.
- Empleados identifican bien los procesos.
- Comprensión de relaciones entre actividades, productos de trabajo y medidas del proceso.

### 3.1.2.4 Nivel Gestionado Cuantitativamente

CMMI-DEV Nivel 4 es un nivel alto de madurez en procesos, marca el inicio del análisis estadístico para mejorar los procesos de la organización, realizando:

- Comprensión de las relaciones entre procesos e impacto de la calidad y rendimiento.
- Análisis y monitoreo estadístico de calidad y rendimiento del proceso en todo el proyecto.
- Los proyectos tienen objetivos cuantitativos para medir calidad y rendimiento del proceso.
- Administración de proyectos en base a los resultados del rendimiento y calidad del proceso.
- Objetivos cuantitativos respecto clientes, usuarios finales, organización y procesos.

### 3.1.2.5 Nivel Optimizado

El nivel 5 es el nivel más alto de CMMI-DEV, representa a una organización con madurez en todas sus áreas de proceso, mejorando e innovando sus procesos constantemente, donde:

- Administrar proyectos con resultados del rendimiento global de múltiples proyectos.
- Identificar deficiencias cuantitativas en rendimiento y sus causas para mejora de procesos.
- Mejorar, innovar y medir constante procesos, estándares y tecnología de soporte.

- Comparar objetivos de calidad contra rendimiento del proceso.
- Actualizar objetivos de calidad y rendimiento tanto del proceso como de organización.

Se mencionan las características de todos los niveles de CMMI-DEV, porque la empresa Px ha alcanzado todas las particularidades que requiere cada nivel de madurez. Es necesario conocer el modelo de trabajo y los detalles que se deben respetar para realizar la integración de técnicas ágiles como una nueva forma de trabajo sin dejar de cumplir con el modelo de calidad actual.

## **3.2 RUP**

Esta subsección se obtuvo del libro “The Rational Unified Process an introduction” [22] y con algunos artículos vinculados con el tema. El Proceso Unificado de Rational® (RUP) es una metodología estándar, incremental<sup>11</sup> e iterativa<sup>12</sup> para desarrollar software bajo un enfoque disciplinado. Utilizado para asignar responsabilidades, asegurar la calidad, cubrir las necesidades del usuario final en un plazo establecido y con presupuesto predecible.

RUP integra 6 buenas prácticas de desarrollo de software a su metodología: desarrollo de software iterativo, administración de requerimientos, diseño de arquitectura, modelo de la estructura del software, aseguramiento de la calidad y control de cambios.

### **3.2.1 Estructura de RUP**

La arquitectura global de RUP consta de dos dimensiones (Figura 3-1):

- Eje horizontal contiene fases, ciclos, iteraciones e hitos del ciclo de vida del desarrollo.
- Eje vertical describe actividades, artefactos, flujos de trabajo e involucrados del proyecto.

RUP tiene 9 flujos de trabajo básicos, que ayudan a dividir las actividades en grupos lógicos para desarrollar el proyecto, se realiza en cada ciclo de vida con el énfasis e intensidad que necesite el proyecto.

---

<sup>11</sup> Prioriza requerimientos y determinar los que serán entregados primero para construir el proyecto completo.

<sup>12</sup> Repetir el ciclo completo las veces que sea necesario hasta generar el producto final completo y ejecutable.

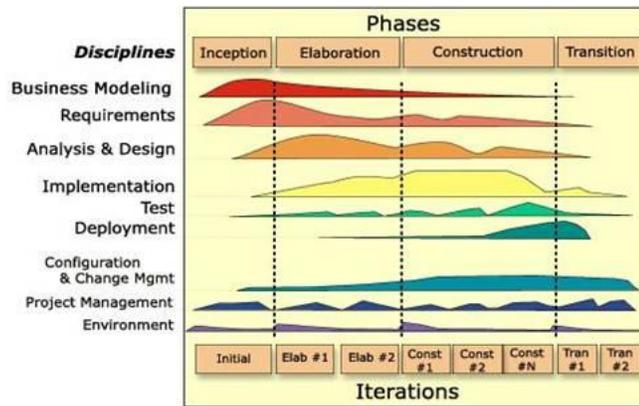


Figura 3-1Ciclo de vida de RUP [22].

Los flujos de trabajo son:

1. Modelado del negocio. Plantea las reglas del negocio y entendimiento del sistema.
2. Requerimientos. Descripción del problema a solucionar, funcionalidad e interacción.
3. Análisis y diseño. Especifica actividades y satisfacer requerimientos.
4. Implementación. Organiza código e integra en sistema ejecutable.
5. Pruebas. Verifica integración, checa requisitos y trata defectos.
6. Desarrollo. Produce productos y entrega a usuarios finales.
7. Administración de proyecto. Planifica, ejecuta y monitorea proyectos.
8. Administración de cambios y configuración. Controla cambios, artefactos y conflictos.
9. Ambiente. Proporciona programas, directrices, plantillas y herramientas para el proceso

### 3.2.2 Ciclo de vida de RUP

En el ciclo de vida RUP es una implementación de desarrollo iterativo en espiral, maneja el proceso en cuatro fases consecutivas que concluyen con un hito<sup>13</sup>. Las fases son Inception, Elaboration, Construction y Transition.

#### 3.2.2.1 Fase Inception

Durante la fase Inception o Inicial se modela el negocio, define alcance, riesgos, planeación, calendario, prototipos y documentación necesaria. El proyecto puede ser cancelado o repensado si no pasa el hito.

<sup>13</sup> Punto bien definido en el tiempo, donde los objetivos principales debieron haber sido alcanzados.

### 3.2.2.2 Fase Elaboration

El propósito de la fase de Elaboration o Elaboración es analizar el problema y conocer que tan grade es el proyecto aunque no se conozca la profundidad de los elementos. El proyecto se somete a la decisión de comprometerse con la construcción y transición, por lo que puede ser abortado o repensado si no pasa el hito.

### 3.2.2.3 Fase Construction

Durante la fase de Construction o Construcción, se desarrollan todos los componentes y características de aplicación, integrándose al producto y probando todas las funciones a fondo. Al finalizar, se decide si software, sitios y usuarios están listos para iniciar la operación sin altos riesgos. La transición puede posponerse de no estar listos.

### 3.2.2.4 Fase Transition

En la fase de Transition o Transición el producto de software ha sido entregado a los usuarios y los problemas surgen, obligando a desarrollar nuevas versiones, corregir problemas o terminar características pospuestas. Decide si se cumplieron los objetivos o se comienza con un nuevo ciclo de desarrollo.

RUP es uno de los ciclos de vida para desarrollar software en la empresa Px, pero la organización ha adaptado la metodología a sus propias necesidades. Tomando de RUP aquello que requiere para cumplir con el modelo de calidad CMMI-DEV y eliminando lo innecesario.

Las técnicas ágiles no cubren en su totalidad el modelo CMMI-DEV, por lo que la empresa Px conserva algunas prácticas de RUP. Conocer la metodología RUP permite entender mejor cuales son los aspectos que pueden ser abarcados por métodos ágiles y cuáles no.

# Capítulo 4

## Métodos ágiles

Los métodos ágiles motivan a los equipos de trabajo a desarrollar software en tiempos más cortos respecto a otras metodologías [23]. Muchos equipos de desarrollo han adoptado Scrum y Kanban<sup>14</sup> como método principal, porque se centran en las tareas de un proyecto [24].

Este capítulo muestra de manera general los métodos ágiles que la empresa Px tomó como base para seleccionar las técnicas ágiles que serían integradas con el modelo CMMI-DEV.

### 4.1. Scrum

La información de esta subsección fue extraída del “La guía de Scrum” [25], fortaleciendo la investigación con artículos científicos.

Scrum es un marco de trabajo iterativo e incremental para administrar el desarrollo y mantenimiento del software eficazmente. Está diseñado para aumentar la velocidad de desarrollo y la calidad del producto, fomenta la colaboración y comunicación entre los integrantes del equipo. [26]. Define responsabilidades y tareas por medio de roles [27]. Scrum consta de Equipo Scrum, Roles, Eventos, Artefactos y las reglas que los relacionan.

---

<sup>14</sup> Método ágil para desarrollar software, basado en tableros para administrar tareas o actividades [33].

### 4.1.1. Equipo Scrum

Los equipos Scrum son auto-organizados, productivos, multifuncionales, creativos y maximizan la oportunidad de retroalimentación. Tienen las capacidades y habilidades necesarias para no depender de personas externas al equipo. El Equipo Scrum está formado por el Dueño del Producto, Equipo de Desarrollo y Scrum Master.

#### 4.1.1.1. Dueño del Producto o Product Owner

Actúa como intermediario entre cliente y equipo de desarrollo, sus decisiones deben ser respetadas por toda la organización. Es el responsable del:

- Trabajo realizado por el Equipo Scrum y del proyecto frente al cliente.
- Contenido y disponibilidad del Product Backlog<sup>15</sup>, priorizar, comparar actividades planeadas contra las realizadas para tomar decisiones.

#### 4.1.1.2. Equipo de Desarrollo o Development Team

Es responsable de desarrollar y estimar los elementos de la lista de pendientes del Sprint<sup>16</sup>, llamada Sprint Backlog<sup>17</sup> para entregar avances funcionales del proyecto al final de cada Sprint. Tienen que ser los integrantes suficientes para completar el trabajo con calidad durante el Sprint. Es recomendable un promedio de 5 integrantes [28], para una comunicación apropiada.

#### 4.1.1.3. Scrum Master

Es el encargado de verificar que el Equipo de Desarrollo conozca y trabaje bajo el marco de Scrum, eliminar impedimentos y facilitar el entorno de trabajo para avanzar con el proyecto. Administra y organiza el Sprint Backlog<sup>18</sup> efectivamente para maximizar el valor del desarrollo.

---

<sup>15</sup> Lista de requerimientos que debe cumplir el proyecto completo.

<sup>16</sup> Periodo de tiempo en el cual se desarrollará una parte del producto.

<sup>17</sup> Lista de requerimientos a desarrollar en un Sprint.

<sup>18</sup> Lista de requerimientos a realizar durante el Sprint.

## 4.1.2.Eventos Scrum

Existen eventos predefinidos por Scrum, con la finalidad de regular las reuniones. Todos los eventos tienen definida una función y duración máxima, pero pueden terminar en cuanto se alcanza el objetivo. Dichos eventos son: Sprint, Reunión de Planeación de Sprint, Scrum Diario, Revisión de Sprint y Retrospectiva de Sprint.

### 4.1.2.1. Sprint

En un Sprint es un periodo de tiempo para desarrollar una parte del producto final, considerado como un pequeño proyecto independiente, funcional y desplegable. Cada Sprint tiene su propio objetivo<sup>19</sup>, alcance<sup>20</sup> y planeación. El Sprint contiene al resto de los eventos y en cuanto finaliza un Sprint inicia el siguiente. La duración puede ser 1 mes o menos.

En un Sprint no se realizan cambios que afecten los objetivos funcionales ni de calidad, esto podría provocar su cancelación e iniciar una revaloración. El único que puede cancelar un Sprint es el Dueño del Producto.

### 4.1.2.2. Reunión de Planeación de Sprint o Sprint Planning Meeting

Es una reunión colaborativa para definir el objetivo y alcance del Sprint, selecciona la lista de productos a desarrollar llamada Sprint Backlog, así como planea el trabajo a realizar durante el Sprint. El Scrum Master asegura que todo el Equipo de Desarrollo entienda el trabajo a realizar en el Sprint, está presente el Equipo Scrum, algún asesor técnico o personal relacionado con el negocio. La duración máxima es de 8 horas para Sprints de un mes.

### 4.1.2.3. Scrum Diario o Daily Scrum

El Scrum Diario es una reunión para organizar el trabajo del día correspondiente. Durante la reunión cada integrante del Equipo de Desarrollo expresa su trabajo con tres preguntas: ¿Qué hice ayer?, ¿Qué voy a hacer hoy?, ¿Qué impedimentos están en mi camino? [29]. Todos los

---

<sup>19</sup> Es la meta a ser alcanzada al implementar el Sprint Backlog por el equipo de desarrollo.

<sup>20</sup> Delimita el trabajo y funcionalidad del software a realizar durante el Sprint.

integrantes del equipo buscan entender y solucionar los problemas de los compañeros para poder avanzar. Generan gráficas de trabajo pendiente [29] y se toman decisiones rápidas.

Únicamente participa el Equipo Scrum. Es recomendable realizar la junta de pie [28] y a la misma hora todos los días. La duración es de 15 minutos aprox.

#### **4.1.2.4. Reunión de Revisión de Sprint o Sprint Review**

La reunión se realiza al final del Sprint, permite inspeccionar el entregable y adaptar el Product Backlog de ser necesario. La duración es de 4 horas aproximadamente para Sprint de 1 mes, participa el Equipo Scrum e invitados clave.

El Dueño del Producto verifica el estado actual del Sprint, comparando elementos terminados y faltantes del Sprint Backlog y proyecta fechas para finalizar. El Equipo de Desarrollo expresa lo que hizo bien, problemas que surgieron, soluciones y muestra el entregable para retroalimentación. El Equipo Scrum planifica el trabajo del siguiente Sprint y aspectos que pueden traer cambios.

#### **4.1.2.5. Reunión de Retrospectiva o Sprint Retrospective**

La reunión se realiza después de la Reunión de Revisión del Sprint y antes iniciar el siguiente Sprint. El Equipo Scrum inspecciona como le fue en el último Sprint, identifica elementos positivos, posibles mejoras, crea un plan para integrarlas y mejorar el desempeño del equipo. El Scrum Master anima al equipo a mejorar para el siguiente Sprint. Dura 3 horas aprox.

### **4.1.3. Artefactos de Scrum**

Los artefactos de Scrum son el trabajo materializado: Product Backlog y Sprint Backlog.

#### **4.1.3.1. Product Backlog**

Es la lista ordenada de todos los requisitos para desarrollar el producto completo, incluye características, funcionalidades, requisitos, mejoras y correcciones del producto final. Los

cambios solicitados, en caso de ser aceptados son registrados. Refleja los requisitos iniciales, pero evoluciona junto con el producto y su entorno, por lo que puede actualizarse en cualquier momento. Es revisada y refinada por el Equipo Scrum.

#### 4.1.3.2. Sprint Backlog

El Sprint Backlog es la lista de actividades necesarias para generar un producto funcional y útil durante el Sprint, seleccionadas por el Equipo de Desarrollo del Product Backlog. El Equipo de Desarrollo es el único que puede modificar esta lista, eliminando actividades innecesarias o agregando aquellas que se requieran. Los elementos de la lista se dividen en pequeñas tareas.

La empresa Px adoptó Scrum para complementar la administración de los proyectos basados en técnicas ágiles, al implementar este marco de trabajo algunos procesos fueron sustituidos por actividades ágiles que antes se realizaban con metodología basadas en PMBoK y algunas disciplinas de RUP. Scrum también fue considerado como el ciclo de vida de los proyectos ágiles.

## 4.2. Extreme Programming

Esta subsección fue extraída del libro “Extreme Programming Explained” [30] , fortaleciendo la investigación con artículos relacionados al tema.

Extreme Programming (XP) es una metodología ágil basada en el conocimiento técnico, aspectos humanos y cambios frecuentes por conceptos vagos de los clientes, por lo que adopta prácticas sensibles al ser humano y respeto al trabajo. La forma de trabajo incluye valores, metas alcanzables, comunicación clara, trabajo en equipo y excelentes técnicas de programación.

El equipo de trabajo debe ser de individuos involucrados, con apertura, dispuestos a realizar su mejor esfuerzo en el trabajo y asumir las consecuencias con responsabilidad. El sentido humanista busca modificar el comportamiento de las personas. El éxito de los proyectos radica en la confianza en el trabajo que realiza el equipo.

XP cuenta con la participación continua del cliente en el proyecto, motivando la retroalimentación en los requerimientos generando cambios oportunos [31].

### **4.2.1. Características de XP durante el desarrollo**

Al momento, XP está basado en 5 valores, 11 principios y 13 prácticas, donde los valores tienen un propósito dentro de la práctica y los principios son la directriz de valores y prácticas.

Algunas actividades que se realizan con XP durante el desarrollo de software, son:

- Al iniciar un proyecto se realiza una pequeña versión del software para prevenir gastos por cancelación del proyecto.
- Planear con tareas cortas para resolver problemas durante el ciclo y refinar constantemente las especificaciones.
- El calendario de actividades es en ciclos cortos para tener listo algún entregable y canalizando primero las actividades con mayor prioridad para el negocio.
- Iteraciones semanales para tener retroalimentación detallada del cliente.
- Los programadores estiman individualmente con responsabilidad.
- Ejecución de pruebas durante todo el proyecto para garantizar la calidad y evitar acumulación de problemas.
- El proyecto se entrega en un solo lanzamiento.

### **4.2.2. Valores**

Los valores dan dirección a las prácticas para desarrollar software, ya que aun cometiendo un error se debe valorar el aprendizaje. La falta de conocimiento no es problema, pero si hacer las cosas mal a sabiendas que se está haciendo mal.

Son 5 valores fundamentales que guían el desarrollo de software con esta metodología: comunicación efectiva entre los miembros del equipo de desarrollo, simplicidad en las soluciones, retroalimentación oportuna para adecuar rápidamente, respeto a la contribución de los integrantes del equipo y valor para enfrentar problemas en busca de soluciones óptimas.

### 4.2.3.Principios

Los principios ayudan a entender el objetivo de las prácticas. Algunos principios que rigen XP:

- La humanidad fortalece al ser humano para aumentar su productividad.
- Mejora permanente tanto en individuos como en resultados.
- Diversidad en los equipos brinda enriquecimiento de habilidades en los equipos de trabajo.
- La reflexión honesta ayuda a lograr la excelencia.
- Tener un problema es una oportunidad para adquirir nuevo conocimiento.
- El fracaso impulsa triunfos cuando se asimila lo ocurrido evitando que vuelva a suceder.
- El flujo de trabajo frecuente brinda beneficio constante.
- La redundancia en soluciones puede prevenir problemas costosos futuros.
- La calidad desde el inicio de las actividades da predictibilidad en comportamientos futuros.

### 4.2.4.Prácticas

Las prácticas son actividades claras, útiles y con una finalidad, que se realizarán durante el desarrollo de software. Las prácticas pueden ser modificadas conforme las necesidades del proyecto u organización [32], siempre y cuando se basen en valores.

Las prácticas de XP que han sido aprovechadas por la empresa Px para desarrollar software bajo el modelo de calidad de CMMI-DEV, son:

- Los miembros del equipo comparten un espacio para trabajar juntos y facilitar la comunicación.
- El área de trabajo debe ser cómoda, limpia, con luz y espacio suficiente para comunicarse.
- El equipo de trabajo cuenta con las habilidades y preparación para cumplir con el proyecto.
- La holgura de actividades pequeñas, permite aumentar las actividades a trabajar o completar tareas más grandes.
- Realizar metáforas sobre posibles supuestos del proyecto para encontrar soluciones.
- Los requerimientos utilizan tarjetas físicas para estimar el tiempo de su desarrollo.
- La integración continua disminuye posibles problemas futuros y altos costos al corregirlos.
- El diseño incremental permite madurar las necesidades del proyecto y permite mejorar.

Estas dos prácticas se pueden fomentar dentro de los equipos, pero inicialmente depende de las costumbres y hábitos de los integrantes del equipo:

- Respetar el espacio personal entre compañeros, ya sea por higiene o respeto.
- Los individuos deben estar descansados, sin distractores y administrar bien el tiempo.

Las siguientes prácticas de XP están formalizadas como procesos organizacionales alternativos de Px para desarrollar software, pero pueden ser llevadas a cabo por los desarrolladores como parte del trabajo en equipo y comunicación que predominan en los métodos ágiles:

- Construir un pequeño software en 10 minutos puede aclarar aspectos del proyecto y recibir retroalimentación inmediata por parte del cliente.
- Programar en parejas utilizando una misma computadora.
- Establecer algunas pruebas antes del desarrollo puede aclarar el panorama del proyecto.

### **4.3. Kanban**

Esta subsección se obtuvo del libro “Kanban: Successful Evolutionary Change for Your Technology Business” [33] y se complementó con artículos relacionados con el tema.

La idea fundamental de Kanban es visualizar el flujo de trabajo por medio de un tablero e informar a todo el equipo de trabajo sobre el estado del producto, sincronizar y controlar el trabajo respecto a la capacidad de producción en diferentes momentos del desarrollo, así como actualizar y dar seguimiento al trabajo [24] [34] [23].

#### **4.3.1. Forma de trabajar con Kanban**

Los tableros de Kanban esquematizan el trabajo, tienen divisiones verticales a manera de columnas que representan las fases del ciclo de vida del desarrollo de software o los estados por los que puede pasar una actividad o tarea.

Junto con el cliente se crea una lista de actividades para desarrollar el producto completo, las actividades son desglosadas en tareas y se priorizan.

Los miembros del equipo representan cada tarea en una tarjeta o postick que incluye una breve descripción con el tiempo estimado para su desarrollo. Las tarjetas se colocan en la primera columna del tablero, expresando que son tareas pendientes por realizar.

Kanban permite a los desarrolladores seleccionar libremente las tareas que van a realizar [23]. El desarrollador toma la tarjeta de la actividad que va a realizar y la cambian de columna, para especificar que ya no está en pendiente y fue seleccionada para realizarla. De esa manera la van pasando de estado en estado, hasta que llega a la última columna expresando que esa tarea ya completo las fases del ciclo de desarrollo y está lista.

Kanban maneja el “Trabajo en Progreso”, en donde limita cada columna o fase con un número de actividades máximas que se pueden realizar, para evitar la saturación. Visualizar el trabajo del proyecto permite medir los tiempos de espera y detectar rápidamente problemas por cambios repentinos evitando bloqueos en el tablero [23] [24].

Los equipos de trabajo de Kanban son auto-organizados, colaborativos, organizados, responsables, trabajan por entregas e integran continuamente. Lo que trae como consecuencia flujo permanente de trabajo, aumentar la calidad del producto, disminuir tiempos de entrega y aumentar la productividad.

Kanban no requiere grandes cambios ni estructuras organizacionales específicas como otros métodos ágiles [34]. La simplicidad y claridad de Kanban faculta a las personas a seguir pocas reglas [23], pero no es suficiente para administrar un proyecto y dificulta visualizar proyectos complejos por lo que requiere de prácticas de apoyo [23].

La empresa Px utiliza Kanban para llevar el control de sus actividades por medio de tarjetas y tableros digitales, si el equipo requiere mayor información visual tiene espacios destinados para los tableros físicos.

# Capítulo 5

## Metodología de la investigación

Este capítulo describe las fases y elementos de la metodología del caso de estudio, estrategia, enfoque, características y criterios de calidad de la investigación sobre las cuales se basará la investigación de esta tesis.

En el área de ingeniería de software los casos de estudio ayudan a ver sucesos específicos o investigar alguna área de la organización [35]. Cada aspecto a investigar es llamado unidad de análisis, donde la cantidad de unidades varían respecto a las preguntas de investigación [35].

Hay diversas metodologías para realizar la investigación de un caso de estudio en el área de ingeniería de software [35] [36]. La tesis se basará en la metodología que presenta el libro “Case Study Research in Software Engineering. Guidelines an Examples” [35].

### 5.1. Metodología de caso de estudio

La metodología que se utilizó en la investigación del caso de estudio de la empresa Px, establece 4 fases de investigación: Diseño, Colección de Datos, Análisis e Interpretación y Reporte para Difusión, como se muestra en la Figura 5-1 [35].

Esta metodología es iterativa, es decir, las fases se pueden ir repitiendo conforme se vaya requiriendo en riguroso orden ascendente o descendente y sin saltarse fases [35]. Los resultados del análisis determinan si se realiza nuevamente una o más fases.

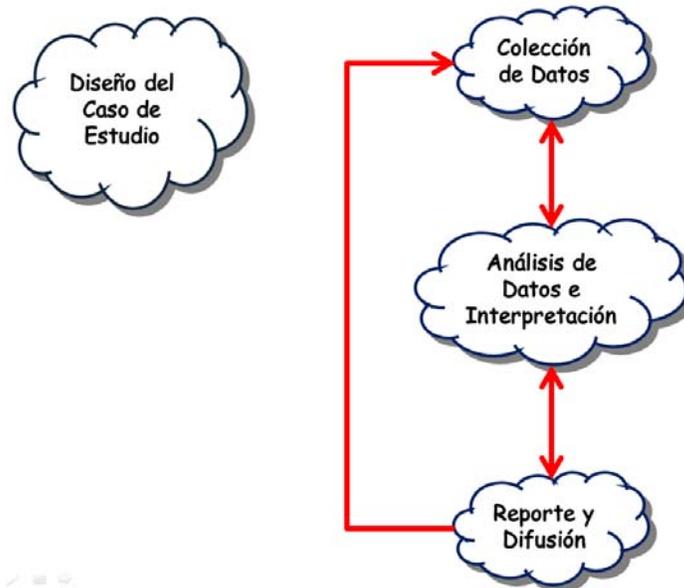


Figura 5-1 Metodología de caso de estudio de 4 fases.

Para considerar una investigación válida, se requiere confiabilidad en todas las fases del caso de estudio, los resultados no deben ser afectados por la perspectiva del investigador y las evidencias deben ser coherentes y fiables [35]. Los documentos generados durante la investigación fueron versionados para controlar el seguimiento de las acciones realizadas.

### 5.1.1. Fase 1 – Diseño de la investigación

El diseño de la investigación establece el plan a seguir [37] o la secuencia de pasos lógicos que unen los resultados de la investigación con las conclusiones [36].

El protocolo del Diseño debe ser dinámico para ser modificado conforme avanza el caso de estudio, es importante comparar constantemente la información obtenida con la esperada, para actualizar el protocolo y obtener resultados benéficos a la organización.

El protocolo del Diseño debe incluir los elementos que se describen en la Tabla 5-1 [35]:

Elemento del estudio	Descripción
<b>Razón que motiva la investigación</b>	Es la justificación o razones claras que motivan realizar una investigación, como : <ul style="list-style-type: none"> <li>• Realizar alguna mejora dentro de una organización.</li> <li>• Evaluar determinada tecnología.</li> <li>• Preparar un despliegue a mayor escala de una tecnología o proyecto piloto.</li> <li>• Contribuir con una teoría o hipótesis, así como identificar huecos en la literatura disponible.</li> </ul>
<b>Objetivo de la investigación</b>	El objetivo del estudio es aquello que se espera alcanzar con la investigación, aunque el resultado puede ser diferente a lo previsto inicialmente. En una investigación los objetivos académicos difieren de los objetivos de la industria.
<b>Marco teórico que da origen a la investigación</b>	Contar con el marco teórico de referencia tanto para investigadores como para revisores de la investigación. Este marco se puede fundamentar investigando en: revistas de literatura científica, investigaciones anteriores que influyen al actual, datos empíricos, ausencia y presencia de información, ventajas y problemas identificados [37] [36]. Al detallar la fase de diseño no se incluirá el marco teórico, porque se encuentra en los Capítulos 2,3 y 4 del trabajo de tesis.
<b>Preguntas de investigación</b>	Conjunto de preguntas que se formulan en base al objetivo e hipótesis de la investigación, donde al encontrar respuestas indica que se cumplió con el objetivo de la investigación. Estas preguntas se pueden perfeccionar conforme se avanza la investigación. Al iniciar el caso el investigador tiene cierto conocimiento y de ahí surgen las preguntas de investigación, que deben ser claras y precisas.
<b>Hipótesis o propuestas</b>	Las propuestas son predicciones lógicas del mundo real acerca del tema, también pueden considerarse como puestas en práctica de las preguntas de investigación o declaraciones sobre el conocimiento que se está buscando. Las hipótesis surgen de las propuestas y ayudan a definir los conceptos y procesos de la investigación
<b>Unidades de Análisis en que se divide la investigación</b>	Las unidades de análisis son secciones de la investigación limitadas por criterios de inclusión o exclusión [35] [37]. Hay 2 tipos de casos de estudio según las unidades de análisis que lo formen (Figura 5-2) [37]: <ul style="list-style-type: none"> <li>• Holístico. Es el estudio de un solo caso centrado en una sola unidad de análisis, busca preservar el carácter unitario del caso. Los casos de estudio múltiple de tipo holístico, tienen más de un caso de estudio y cada caso contiene una unidad de análisis.</li> <li>• Embebidos. Estudia varias unidades de análisis dentro de un mismo caso de estudio. Los casos embebidos múltiples, contienen más de un caso y cada caso contiene más de una unidad de análisis.</li> </ul>

Tabla 5-1 Elementos del protocolo del Diseño del caso de estudio.

## 5.1.2. Fase 2 – Colección de datos

La colección de datos consiste en recoger la información y evidencia que permiten corroborar las hipótesis establecidas en el diseño [36], así como determinar la forma en que se almacenarán los datos [35]. Una investigación con varias fuentes da fiabilidad y confianza, mostrando diversas perspectivas del mismo fenómeno [35].

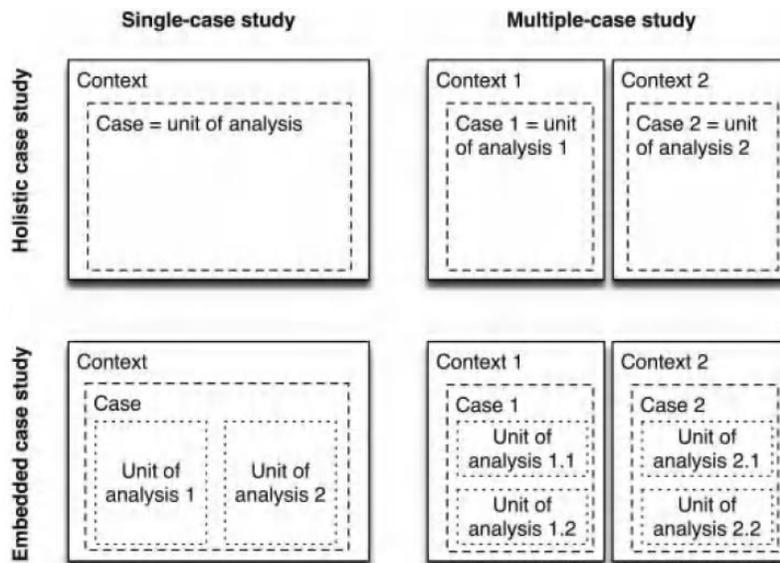


Figura 5-2 Tipos de casos de estudio [37].

Para realizar una recolección de datos adecuada es recomendable utilizar múltiples fuentes de datos, crear una base de datos para tener disponible la información cuando se requiera, validar los datos y versionar los documentos utilizados [44] [45].

Los métodos de recolección de datos pueden ser:

- Directa, por medio de entrevistas u observación.
- Indirecta, con instrumentación de herramientas.
- Independiente, al realizar análisis de documentación o artículos.

Las entrevistas son de carácter exploratorio y pueden ser de tipo ser no estructurada, semi-estructurada o estructurada, según su objetivo. La entrevista no estructurada indaga preocupaciones generales e intereses del investigador.

La semi-estructurada puede tener tanto preguntas cerradas como abiertas, el orden puede ser diferente al planeado, permite improvisar y explorar puntos planeados en la conversación. En tanto, que la estructurada está formada por preguntas cerradas<sup>21</sup>. Ambas tienen un objetivo descriptivo y exploratorio.

<sup>21</sup> Buscan conocer información específica, no requiere de detalles del tema abordado.

Cuidar la fuente de la información es prioritario, ya que se asegura la calidad de los datos recolectados. Verificar la viabilidad de los datos es importante, ya que en muchas ocasiones los datos son recolectados para otros propósitos. También, se debe cuidar el recabado excesivo de datos o recolectar información innecesaria [35].

### **5.1.3. Fase 3 – Análisis e Interpretación**

La fase de análisis e interpretación, especifica los métodos que se utilizaron para analizar los datos recolectados en cada unidad de análisis. Confrontando las fundamentos teóricos que motivaron el estudio con las evidencias recolectadas, determinando si fueron aceptadas o rechazadas las hipótesis [37]. El tipo de análisis sobre los datos depende de la cantidad y tipos de datos recogidos.

El análisis permite saber que está pasando con el caso de estudio, entender los detalles y encontrar patrones en los datos o saber si la investigación va por el camino correcto [35]. El análisis no solamente se lleva a cabo durante su propia fase, sino en el diseño, al recolectar datos, al transcribir una entrevista [35] o en algún otro momento donde el investigador pueda reflexionar sobre los datos recabados y los objetivos a alcanzar en la investigación.

Al realizar el análisis y descubrir conocimiento puede requerir modificar herramientas de colección de datos o el mismo diseño para mantener coherencia en la evidencia [35].

El trabajo de esta fase se dividirá en dos capítulos independientes pero relacionados. El Capítulo 8 corresponde al Análisis, que describirá las acciones realizadas con la información recolectada hasta llegar a la interpretación de la información. Mientras que, el Capítulo 9, se encargará de mostrar la Interpretación de la información, responder las preguntas de investigación con los datos analizados y aceptar o rechazar las hipótesis.

### **5.1.4. Fase 4 – Elaboración de reporte para su difusión**

El reporte escrito de la investigación es el resultado del trabajo realizado al finalizar la investigación. El informe contiene información específica según el medio de difusión, como revistas científicas, seminarios o talleres a estudiantes, reporte organizacional, entre otros [35].

No toda la información es publicable por intereses de la organización participante. Esto va de la mano con la ética profesional de los investigadores, así como los aspectos legales que implica difundir información sin autorización de la empresa [35].

En esta fase se registran objetivos, alcance, hipótesis, preguntas de investigación, diseño de la investigación, procedimientos de colección y análisis de datos, relaciones encontradas y validadas, conclusiones e implicaciones, así como la posibilidad de trabajos futuros [37].

El reporte de esta investigación es el mismo trabajo de la tesis, donde los resultados obtenidos a lo largo del caso de estudio se encuentran en el Capítulo 9 correspondiente a la Interpretación de los datos analizados y aceptación o rechazo de las hipótesis.

Los capítulos 6, 7 y 8 describen las fases de diseño, colección de datos y análisis de la investigación. El capítulo 9 muestra los resultados obtenidos en el caso de estudio respondiendo a las preguntas de investigación, acepta o rechaza hipótesis. Mientras en el capítulo 10 se recopilan los beneficios y áreas de mejora detectados en la investigación. Finalmente, el capítulo 11 contiene conclusiones, debilidades y fortalezas de la investigación, así como trabajos futuros.

## **5.2. Estrategias de investigación**

La investigación en un caso de estudio requiere seguir una estrategia durante la investigación, tal como la entrevista, experimentación y acciones sobre la investigación [35].

La entrevista provee una vista general del problema [35], utilizando cuestionarios en una población para investigar una situación especial [38], pueden ser individuales o grupales. La experimentación mide los efectos de una variable bajo un experimento y puede tener características similares a un caso de estudio [35]. La acción de investigación es una contribución o cambio en algún aspecto de la investigación [38].

Otra estrategia es la observación, que no altera el comportamiento del proceso o fenómeno en estudio [35]. El análisis de archivo y datos históricos son considerados como estrategias de investigación [37], aunque podrían ser catalogados como métodos de recolección de datos [35].

La entrevista fue la herramienta fundamental en la investigación sobre la empresa Px, para contestar las preguntas de investigación e indagar aspectos específicos de la integración de técnicas ágiles. También se usaron encuestas, datos históricos y documentos oficiales de la empresa Px.

Durante la investigación al ir obteniendo información, se agregaron, modificaron y adecuaron algunas preguntas de investigación de la fase de diseño. Esta acción se realizó para evitar redundancia en los datos, así como para fortalecer y enriquecer el resultado de la investigación.

### 5.3. Perspectiva de la investigación

La perspectiva de la investigación puede ser de tres tipos positivista, crítica e interpretativa [39].

El caso de estudio en la empresa Px cuenta con las tres perspectivas de investigación:

- Positivista al buscar evidencia formal, medir variables y cuestionar la hipótesis.
- Interpretativa por lograr aprendizaje al interpretar el fenómeno desde el punto de vista de los participantes.
- Crítica al indagar el impacto de la forma de trabajo sobre las relaciones sociales de los equipos y la mejora del individuo, así como los valores y preparación que se requieren con esta nueva forma de trabajar.

### 5.4. Características de la investigación

Las características del caso de estudio guían el rumbo de la investigación [35]. Las particularidades del caso de estudio de la empresa Px se muestran en la Tabla 5-1.

Característica	Descripción	Caso de Estudio en Px
<b>Proceder para llevar el control de la investigación</b>	<p>El control del caso de estudio puede ser de dos tipos, según los objetivos a analizar [35]:</p> <ul style="list-style-type: none"> <li>• Flexible. Los parámetros u objetivos de la investigación cambian durante el estudio.</li> <li>• Fija. Los parámetros u objetivos de la investigación no cambian.</li> </ul>	<p>El control es flexible, debido a que en el diseño inicial no se tiene totalmente claro el camino que puede tomar la investigación. Podrían cambiar en base a los resultados que se obtienen en el proceso [35] [36].</p>

<b>Característica</b>	<b>Descripción</b>	<b>Caso de Estudio en Px</b>
<b>Propósito de la investigación</b>	<p>Finalidad para realizar la investigación y descubrir lo que sucede con un fenómeno a través de las preguntas de investigación [36]. Los propósitos se pueden clasificar en 4 [38] [36]:</p> <ul style="list-style-type: none"> <li>• Explicación. Busca una causa del fenómeno.</li> <li>• Exploración. Genera hipótesis e investiga lo que sucede con el fenómeno estudiado.</li> <li>• Descripción. Detalla el estado actual del fenómeno.</li> <li>• Mejoramiento. Intenta mejorar el estado del fenómeno.</li> </ul>	<p>Los propósitos empleados en la investigación, son:</p> <ul style="list-style-type: none"> <li>• Explicación. Busca los motivadores de generar una nueva forma para desarrollar software.</li> <li>• Exploración. Genera hipótesis y preguntas de investigación descritas en la fase de diseño<sup>22</sup>.</li> <li>• Descripción. Incluye la forma de trabajo actual y la influencia en el ambiente de trabajo.</li> <li>• Mejoramiento. Intentar detectar aspectos de mejora en la forma de trabajo.</li> </ul>
<b>Tipos de datos a utilizar en la investigación</b>	<p>Los tipos de datos a coleccionar pueden ser [35]:</p> <ul style="list-style-type: none"> <li>• Cuantitativos. Incluye números y clases, se analiza estadísticamente lo que ayuda a la exactitud.</li> <li>• Cualitativos. Incluye palabras, descripción, dibujos y diagramas, se realiza clasificación y categorización.</li> </ul>	<p>Los datos a procesar en el caso de estudio, son:</p> <ul style="list-style-type: none"> <li>• Cuantitativa, incluye encuestas y documentos históricos.</li> <li>• Cualitativa, incorpora entrevistas, documentos históricos y descriptivos.</li> </ul>
<b>Tipos de triangulación a realizar para la extracción de resultados</b>	<p>Emplear diversas estrategias de investigación permite evaluar de diferentes puntos de vista, provocando resultados enriquecedores [40].</p> <p>Se tienen 4 tipos de triangulación [40]:</p> <ul style="list-style-type: none"> <li>• Datos, verifica y compara datos en diferentes momentos de diferentes fuentes.</li> <li>• Observadores, más de un observador en el estudio.</li> <li>• Metodología, estudia el mismo fenómeno desde diferentes enfoques, verificando que diferentes métodos arrojan diversos resultados.</li> <li>• Investigadores, al haber varias personas de diversas disciplinas apoyando la investigación.</li> </ul>	<p>La investigación incluye:</p> <ul style="list-style-type: none"> <li>• Datos. Al recabar información con encuestas y entrevistas y comparación de muestreos. Exponiendo diferentes puntos de vista.</li> <li>• Metodología. Los beneficios de la nueva forma de trabajo son extraídos de diversas fuentes, como entrevistas, encuestas y gráficas de métricas.</li> </ul>
<b>Enfoque del proceso de la investigación empírica</b>	<p>En la investigación empírica, el conocimiento se obtiene a través de la experiencia. Tiene 3 características:</p> <ul style="list-style-type: none"> <li>• Transparencia. Fase de investigación bien definidas y los resultados deben ser visibles.</li> <li>• Inspección. Vigilancia en todo momento para detectar variaciones.</li> <li>• Adaptación. Realizar los ajustes necesarios para evitar más desviación en suceso.</li> </ul> <p>Se tienen 2 enfoques en la investigación empírica [35]:</p> <p>Inductiva:</p> <ul style="list-style-type: none"> <li>• Inductiva. Se realiza observación, busca patrones, propone una hipótesis y compara o establece una teoría.</li> <li>• Deductiva. Iniciar con teoría existente, propone una hipótesis de investigación, hace observaciones y confirma o rechaza la hipótesis.</li> </ul>	<p>El enfoque a utilizar es deductivo, Figura 5-1:</p> <ol style="list-style-type: none"> <li>1. Inicia con dos teorías existentes para desarrollar software: CMMI-DEV Nivel 5 y Métodos ágiles.</li> <li>2. Se generan una hipótesis para la investigación, mostrada en la fase de diseño del capítulo 6.</li> <li>3. Realizar investigación, al: explorar, describir, analizar y explicar el fenómeno en cuestión para proponer mejoras.</li> <li>4. Sacar conclusiones y confirmar o rechazar hipótesis.</li> </ol>

**Tabla 5-2 Características de la investigación.**

<sup>22</sup> Primer Fase de la Metodología de caso de estudio a utilizar.

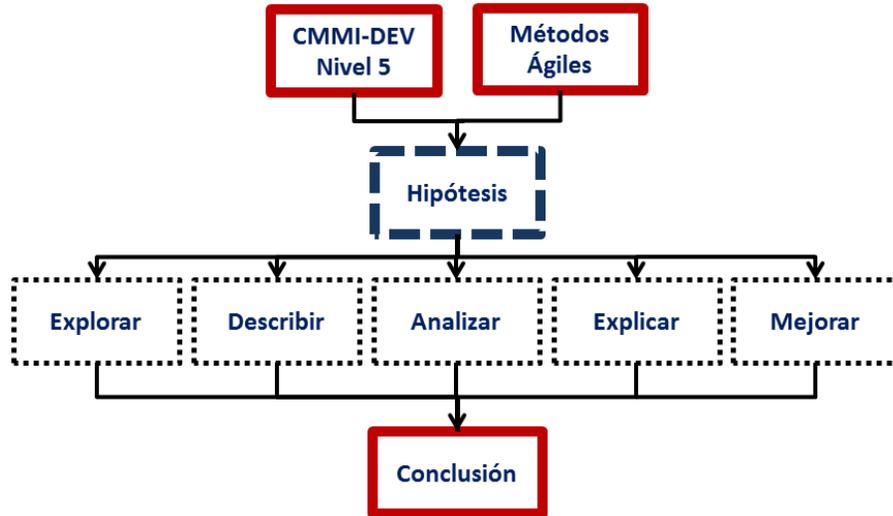


Figura 5-3 Enfoque de investigación Deductivo

## 5.5. Criterios de calidad y cuidados éticos de la investigación

La calidad de la investigación se cuidó en todo el caso de estudio y en cada actividad realizada, para lograr una investigación veraz y confiable. Los aspectos que se cuidaron en la investigación de la empresa Px, fueron:

- Antes de iniciar la investigación se indagaron los aspectos teóricos relacionados.
- La información del marco teórico orientó el diseño de la investigación.
- Los artículos científicos relacionados ayudaron a refinar el camino de las preguntas de investigación.
- Recopilar ordenada y coherentemente de los datos, por medio de carpetas y archivos versionados.
- Trazabilidad al coleccionar evidencias y obtener argumentos suficientes para responder a las preguntas de investigación.
- Emplear diversos métodos de triangulación para obtener los resultados de la investigación.
- Abordar conflictos que surgen en el camino de la investigación y aplicar los cambios requeridos al diseño de la investigación.
- Inferir nuevas preguntas de investigación con los datos recolectados.
- Utilizar metodologías adecuadas según los tipos de datos para obtener resultados válidos.

Durante el caso de estudio se manejó información confidencial, por lo que se tuvieron los cuidados éticos:

- Solicitar autorización para recopilar datos y realizar grabaciones, tanto a responsables de la información, como a los participantes.
- Las entrevistas y encuestas se realizaron de manera individual y confidencial.
- Manipular cuidadosamente los datos sensibles, sin difundir la información entre involucrados.
- Confidencialidad con la información proporcionada.
- Evitar interrumpir constantemente a los participantes en sus actividades laborales.
- Los entrevistados revisaron la información que proporcionaron al ser transcrita la entrevista.
- Cuidar la información publicada evitando perjudicar a la organización Px.
- El responsable empresarial determinó que información se podía publicar.

# Capítulo 6

## Diseño del caso de estudio

Este capítulo se enfoca en la fase de diseño del caso de estudio, mostrando el camino que siguió la investigación. El diseño acompañó las otras fases de la metodología a lo largo del caso de estudio y fue modificado conforme surgía la necesidad. Existe transversalidad entre los diferentes elementos del diseño de la investigación, ya que algunos dan origen a otros.

### 6.1.1.Motivación

Conocer las técnicas ágiles que incorporó la empresa Px con CMMI-DEV para establecer una nueva forma de desarrollar software que permitió mejorar la productividad, disminuir el tiempo para desarrollar un proyecto de software y mejorar el ambiente de trabajo de los equipos.

### 6.1.2.Objetivo de la investigación

El objetivo de la investigación, es realizar un caso de estudio dentro de una empresa que integró técnicas ágiles con CMMI-DEV Nivel 5; con la finalidad de documentar beneficios cuantitativos y cualitativos en el desempeño de los proyectos bajo esta forma de trabajar el desarrollo de software.

### **6.1.3.Descripción de la investigación**

La investigación se realizó en la empresa Px, específicamente con el Grupo de Procesos y el área del CEDS. Para realizar la investigación se requirió:

- Asesoría sobre la empresa Px y su forma de trabajo.
- Información del trabajo de primer proyecto piloto y la forma actual de desarrollar software.
- Documentación de la descripción de algunas herramientas y procesos de la empresa Px.
- Métricas de proyectos con metodología RUP y técnicas ágiles.
- Información del ambiente de trabajo de los equipos ágiles.

En el caso de estudio se analizaron los datos recolectados con métodos acordes a la cantidad y tipo de información. Se interpretaron los datos obtenidos para responder las preguntas de investigación, resolver hipótesis, confirmar objetivos e identificar beneficios y áreas de oportunidad.

### **6.1.4.Relevancia de la investigación**

Los resultados de la investigación proporcionarán un ejemplo documentado de un caso de estudio que presenta los beneficios de incorporar técnicas ágiles con CMMI-DEV en una empresa con Nivel 5, conservando la calidad y aumentando la productividad de la empresa.

Este ejemplo puede servir de referencia a empresas mexicanas que opte por integrar técnicas ágiles a su forma de trabajo empleando procesos innovadores con CMMI-DEV, o bien, a empresas que inicien el trabajo con procesos y quieran desarrollar software con métodos ágiles.

### **6.1.5.Preguntas de investigación**

Las preguntas de investigación tienen origen en los objetivos específicos descritos en el Capítulo 1, al investigar un aspecto particular del caso de estudio (Tabla 6-1).

<b>Pregunta de Investigación</b>	<b>Objetivo Específico</b>
¿Qué técnicas ágiles utilizó la empresa Px para modificar los procesos con metodología RUP, manteniendo el Nivel 5 de CMMI-DEV?	Identificar las técnicas de los métodos ágiles y la manera de cómo se integraron con CMMI-DEV Nivel 5 en la empresa Px.
¿Qué beneficios se obtuvieron al integrar técnicas ágiles con CMMI-DEV Nivel 5?	Realizar la comparación de métricas de productividad y calidad en proyectos que utilizaron metodología RUP con CMMI-DEV contra los que integraron técnicas ágiles con CMMI-DEV.
¿Cómo ha sido la adaptación de los grupos de trabajo al integrar técnicas ágiles con CMMI-DEV?	Identificar por medio de testimonios, los beneficios del uso de técnicas ágiles en la mejora del desempeño laboral de los participantes y la integración de los equipos de trabajo en los proyectos.

**Tabla 6-1 Preguntas de investigación asociadas a los objetivos específicos de la investigación.**

El objetivo específico ‘Identificar y sugerir mejoras a los procesos de trabajo establecidos’, se resolvió en función de las respuestas de las preguntas de investigación previas. De los resultados de las unidades de análisis fueron identificados beneficios y áreas de oportunidad.

### 6.1.6. Hipótesis de la investigación

Para cada pregunta de investigación se generó una hipótesis (Tabla 6-2). Las hipótesis están sustentadas en los aspectos teóricos que fundamentan la investigación.

<b>Pregunta de Investigación</b>	<b>Hipótesis</b>
¿Qué técnicas ágiles utilizó la empresa Px para modificar los procesos con metodología RUP, manteniendo el Nivel 5 de CMMI-DEV?	Se utilizaron las técnicas ágiles Scrum, XP y Kanban para administrar y desarrollar los proyectos de software, sin dejar de cumplir con el Nivel 5 de CMMI-DEV.
¿Qué beneficios se obtuvieron al integrar técnicas ágiles con CMMI-DEV Nivel 5?	Los beneficios de la integración de técnicas ágiles fueron: <ol style="list-style-type: none"> <li>1. Disminuir tiempos de desarrollo y costo.</li> <li>2. Mejorar la calidad del producto.</li> </ol>
¿Cómo ha sido la adaptación de los grupos de trabajo al integrar técnicas ágiles con CMMI-DEV?	Los grupos de trabajo responden satisfactoriamente, al manifestar: <ol style="list-style-type: none"> <li>1. Integrantes motivados al usar métodos de trabajo innovadores.</li> <li>2. Aumentó en el desempeño de su trabajo.</li> <li>3. Miembros del equipo más comprometidos.</li> <li>4. Mayor comunicación y retroalimentación entre los integrantes del equipo de trabajo.</li> </ol>

**Tabla 6-2 Hipótesis generadas a partir de las preguntas de investigación.**

## 6.2. Unidades de Análisis

Las unidades de análisis del caso de estudio se generaron a partir de las preguntas de investigación. El caso de estudio se realizó en una sola empresa, sin embargo, se distinguen varias unidades de análisis en la investigación, por lo que se tiene un caso de estudio simple embebido Figura 6-1.

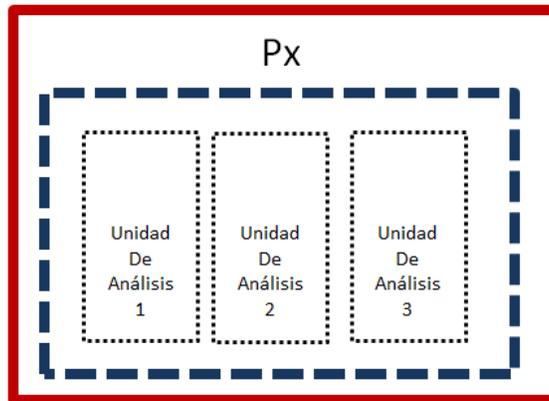


Figura 6-1 Representación de las unidades de análisis del caso de estudio en Px.

Para delimitar las unidades de análisis del caso de estudio se consideraron los objetivos específicos y en consecuencia las preguntas de investigación. Con base en ello decidimos definir una unidad de análisis por cada pregunta de investigación (Tabla 6-2).

Pregunta de Investigación	Unidad de Análisis
¿Qué técnicas ágiles utilizó la empresa Px para modificar los procesos con metodología RUP, manteniendo el Nivel 5 de CMMI-DEV?	Técnicas ágiles integradas con CMMI-DEV
¿Qué beneficios se obtuvieron al integrar técnicas ágiles con CMMI-DEV Nivel 5?	Métricas de proyectos para determinar beneficios
¿Cómo ha sido la adaptación de los grupos de trabajo al integrar técnicas ágiles con CMMI-DEV?	Adaptación del grupo de trabajo

Tabla 6-3 Unidades de análisis asociadas a las preguntas de investigación.

Para cada unidad de análisis se diseñaron preguntas de investigación adicionales, con la finalidad de indagar más afondo los detalles de las unidades.

## 6.2.1. Unidad I: Técnicas ágiles integradas con CMMI-DEV

Esta unidad de análisis tiene como finalidad conocer cuáles son las técnicas ágiles seleccionadas para desarrollar software CMMI-DEV y la manera como se fueron integrando al trabajo con CMMI-DEV.

Para tener información suficiente, la unidad de análisis se dividió en dos secciones. Una sección se encargó de investigar la forma de desarrollar software usando metodología RUP con CMMI-DEV para tener un conocimiento general de la forma de trabajo antes de integrar técnicas ágiles. En la otra sección se indagó la integración de técnicas ágiles con CMMI-DEV.

Las preguntas para conocer la forma de desarrollar software utilizando metodología RUP con CMMI-DEV, son:

- a) ¿Cuál es la estrategia que se sigue en un proyecto de desarrollo de software que utiliza metodología RUP con CMMI-DEV Nivel 5?
- b) ¿Cómo se desarrolla software usando metodología RUP con CMMI-DEV Nivel 5?
- c) ¿Cómo se realizan los cambios durante un proyecto de metodología RUP con CMMI-DEV Nivel 5?, ya que algunas de las principales características de las técnicas ágiles es la facilidad y factibilidad de realizar cambios.

Las preguntas para conocer la forma de desarrollar software empleando técnicas ágiles con CMMI-DEV, son:

- a) ¿Por qué motivo la empresa Px decide integrar técnicas ágiles con CMMI-DEV Nivel 5?
- b) ¿Cuál fue la estrategia que siguió la empresa Px para integrar técnicas ágiles con CMMI-DEV Nivel 5?
- c) ¿Cuáles fueron los criterios para seleccionar las técnicas ágiles que se incorporaron al desarrollo de software bajo el modelo CMMI-DEV Nivel?
- d) Las técnicas ágiles que se integraron con CMMI-DEV Nivel 5 para desarrollar proyectos de software, se descubren con las siguientes preguntas de investigación:
  - ¿Cómo se genera el plan de trabajo utilizando técnicas ágiles con CMMI-DEV Nivel 5?
  - ¿Cómo se implementaron las técnicas ágiles con CMMI-DEV Nivel 5?

- ¿Cómo se realizan los cambios durante un proyecto con técnicas ágiles con CMMI-DEV Nivel 5?
  - ¿La empresa Px cuenta con un plan de mejora para el desarrollo de software?
  - ¿Cómo se reestructuro la forma de trabajo de la empresa para integrar las técnicas ágiles con CMMI-DEV Nivel 5?
  - ¿Qué áreas de proceso y que actividades se eliminaron o implementaron para integrar las técnicas ágiles con CMMI-DEV Nivel 5?
- e) ¿Cuáles son las diferencias de trabajar CMMI-DEV Nivel 5 con metodología RUP respecto a las técnicas ágiles?
- f) ¿Cómo responde el cliente ante las iteraciones rápidas?

### **6.2.2. Unidad II: Métricas de proyectos para determinar beneficios**

La segunda unidad de análisis tiene la finalidad de conocer cuáles fueron los beneficios cuantitativos de emplear técnicas ágiles con CMMI-DEV, como nueva forma de trabajo. La extracción de estos beneficios se realizó al comparar las métricas de desarrollar software usando técnicas ágiles respecto a emplear metodología RUP durante los proyectos.

- a) ¿Cómo se determinó el aumento de productividad de un proyecto de software con técnicas ágiles respecto a uno con metodología RUP?
- b) ¿Cómo se determinó que se tuvo una reducción de defectos?
- c) ¿Cómo se determinó la reducción en los costos de desarrollo de un proyecto empleando técnicas ágiles?

### **6.2.3. Unidad III: Adaptación del grupo de trabajo**

Esta unidad de análisis consiste en indagar cómo responde el equipo de trabajo ante esta forma de desarrollar software y los beneficios que deja a los individuos que laboran bajo este criterio.

- a) Características del desarrollo de software usando técnicas ágiles con CMMI-DEV.
- ¿Cuál es la diferencia de trabajar CMMI-DEV utilizando técnicas ágiles respecto a emplear metodología RUP?

- ¿Cuáles son las ventajas de utilizar técnicas ágiles con CMMI-DEV?
- ¿Se disminuyó el tiempo para desarrollar software utilizando técnicas ágiles?
- ¿El trabajo con técnicas ágiles mejoró el ambiente de trabajo?

b) Adaptación del equipo de trabajo

- ¿Fue difícil la adaptación de los integrantes al desarrollar software utilizando técnicas ágiles con CMMI-DEV?
- ¿Se cuenta con asesorías sobre técnicas ágiles con CMMI-DEV?

c) Características de los integrantes del equipo de desarrollo y mejoras individuales

- ¿Cuáles son las características de los integrantes del grupo de trabajo?
- ¿Los integrantes del equipo han mejorado individualmente?
- ¿Cómo se sienten los integrantes del equipo de trabajo con esta forma de trabajar?
- ¿Cuál es la experiencia de los equipos de trabajo con técnicas ágiles?

# Capítulo 7

## Colección de datos

El objetivo de este capítulo es describir el tipo de datos que se coleccionaron, las técnicas utilizadas para recogerlos y los métodos empleados para almacenar la información. La descripción de la recolección se describe para cada unidad de análisis en secciones separadas.

Los datos fueron recolectados por unidad de análisis, después de recoger la información se sondeaba rápidamente que se tuviera las evidencias necesarias para responder las preguntas de investigación y comprobar el estado de las hipótesis. Las evidencias se almacenaron en una carpeta llamada “Recolección de Datos”, que tiene una subcarpeta por unidad de análisis.

### **7.1. Unidad de análisis I: Técnicas ágiles integradas con CMMI-DEV**

Esta unidad requiere conocer las técnicas ágiles que fueron incorporadas para desarrollar software bajo el modelo CMMI-DEV. Para ello se consideró necesario conocer tanto la experiencia de los participantes en el primer proyecto piloto con técnicas ágiles, como la de los que actualmente trabajan técnicas ágiles con CMMI-DEV. Logrando extraer las técnicas ágiles que fueron propuestas inicialmente y que actualmente prevalecen en el desarrollo de software.

La estrategia principal de recolección de datos de esta unidad de análisis fue el uso de las entrevistas, las cuales fueron grabadas (excepto en un caso) para conservar la integridad y confiabilidad de la información [35] [41].

La información recolectada para esta unidad de análisis fue cualitativa, bajo una perspectiva interpretativa al buscar el punto de vista de los encuestados sobre la integración y su contexto.

Otra característica de la investigación es la flexibilidad, permitiendo agregar, modificar y eliminar preguntas de investigación para evitar redundancia, así como tener información completa y de calidad en las entrevistas.

Esta unidad buscó los motivadores que dieron origen a esta unidad, donde las preguntas de investigación llevaron a la deducir la hipótesis planteada en el diseño. Realizando las fases bidireccionales de la metodología: colección de datos, análisis de datos e interpretación.

Para esta unidad de análisis se recabaron 5 entrevistas a:

1. Miembro directivo de la empresa Px.
2. Asegurador de Calidad de Software.
3. Desarrollador de proyecto piloto.
4. Asesor de proyecto piloto y miembro del Grupo de Procesos.
5. Desarrollador del CEDS.

Todas las personas entrevistadas han trabajado usando técnicas ágiles con CMMI-DEV. Los primeros 4 entrevistados han desarrollado software utilizando metodología RUP con CMMI-DEV. El desarrollador del proyecto piloto, trabajaba como desarrollador de software durante el primer proyecto piloto y actualmente es instructor en la empresa Px. El tiempo de las entrevistas fue de 30 minutos hasta 2 horas, dependiendo del número de preguntas de la entrevista y la disponibilidad del entrevistado.

La primera entrevista fue de tipo semi-estructurado y se realizó para conocer aspectos generales de la integración de técnicas ágiles con CMMI-DEV. Las siguientes 3 entrevistas fueron formuladas con base en las preguntas de investigación de la primer unidad de análisis y considerando el puesto de la persona entrevistada.

La segunda entrevista se realizó de manera remota y fue no estructurada, ya que su principal función era introductoria a la forma de trabajo de proyectos con metodología RUP y técnicas ágiles. La tercera entrevista fue remota y semi-estructurada, ya que se anexaron preguntas para conocer cómo fue el trabajo en el primer proyecto piloto con técnicas ágiles.

La cuarta entrevista fue presencial y tipo semi-estructurado, las preguntas estaban enfocadas en la forma de desarrollar software actualmente y algunas preguntas del primer proyecto piloto. La última entrevista fue presencial, no estructurada y no sé grabo, por surgir de manera espontánea en una visita al CEDS. Después de la entrevista se tomaron notas. Las preguntas se enfocaron en la forma de desarrollar software actualmente usando técnicas ágiles con CMMI-DEV.

Cada entrevista grabada fue procesada con las siguientes acciones:

1. Transcribir literalmente la grabación de la entrevista original.
2. Depurar la entrevista, eliminando información fuera del alcance de la investigación.
3. Dar formato a las preguntas en color azul marino y respuesta de color negro.

Al terminar de realizar la transcripción se envió el documento al entrevistado, dándole la oportunidad de modificar, corregir, aclarar o ampliar la información proporcionada [35]. Una vez recibida la respuesta de los entrevistados se verificaron e incorporaron las nuevas aportaciones o modificaciones solicitadas por el entrevistado. De esta manera se trató de disminuir el sesgo al ser un solo investigador y aumentar la consistencia de los hallazgos encontrados [42].

Todas las entrevistas fueron almacenadas en una carpeta llamada "Entrevistas de campo", ahí se generó una subcarpeta para cada entrevista realizada. El control y administración se realizó en un solo archivo, con una hoja de cálculo que contiene información básica de la entrevista.

La información que se incluyó fue [41]:

- Datos generales de la entrevista: fecha, duración, hora inicial y final.
- Datos personales del entrevistado: nombre, puesto, teléfono, e-mail.
- Objetivo, meta alcanzada y aportación de la entrevista.
- Dirección de la carpeta que almacena archivos de grabación, transcripción o notas.

- Nombres de los archivos de grabaciones y del archivo de la transcripción o nota.
- Lista de aspectos pendientes originados por la entrevista y estado del pendiente.

## **7.2. Unidad de análisis II: Métricas de proyectos para determinar beneficios**

Esta unidad de análisis tiene como finalidad identificar los beneficios que se originaron al integrar técnicas ágiles con CMMI-DEV. Estos beneficios serán extraídos de datos cuantitativos y cualitativos, generados de registros históricos de proyectos realizados por la empresa Px.

La perspectiva de investigación es positivista al buscar evidencia formal, medir variables y cuestionar la hipótesis de la investigación.

La unidad de análisis, utilizó la característica de flexibilidad al seleccionar los registros históricos proporcionados por la empresa Px, para comprobar los beneficios de la integración y responder las preguntas de investigación. Realizando las fases bidireccionales de la metodología: colección de datos, análisis de datos e interpretación.

Para esta unidad de análisis se recolectó la siguiente información:

- Entrevistas no estructuradas.
- La presentación con los objetivos de mejora organizacional.
- Reporte de métricas de primeros proyectos pilotos con técnicas ágiles.
- Documento con descripción de las gráficas utilizadas en el área de métricas.
- Presentación que documenta la mejora de la velocidad en primer proyecto piloto con técnicas ágiles.
- Documento de disminución del tiempo invertido para documentar proyecto piloto.
- Informe con disminución de densidad de defectos por producto.

Se realizaron 3 entrevistas. La primera entrevista fue con el responsable de elaborar los reportes de métricas para los proyectos, con la finalidad de conocer el proceso de medición, reportes de métricas, elementos, características e interpretación de los reportes. Esta entrevista fue grabada y procesada como las entrevistas de la unidad de análisis anterior.

La segunda y tercer entrevistas fueron no estructuradas, con el responsable del área de métricas. La segunda entrevista fue presencial y no se grabó, ya que tenía el objetivo de solicitar la información que se requería para comprobar los beneficios que trajo la integración de técnicas ágiles con CMMI-DEV.

La tercera entrevista fue grabada, transcrita, tenía la finalidad de recibir los documentos solicitados y breves comentarios del contenido de cada archivo. Todos los documentos proporcionados fueron almacenados en una carpeta llamada “Métricas”.

A continuación se describen los documentos que proporcionó la empresa Px para mostrar los beneficios logrados en los proyectos de desarrollo de software, por medio de métricas:

El archivo de la presentación, documenta los objetivos organizacionales de mejora y lista los objetivos que la empresa Px pretende mejorar. Donde destaca el aumento de la velocidad de implementación y la disminución de defectos en los productos generados en cada proyecto.

El reporte de métricas contiene información de los primeros proyectos pilotos con técnicas ágiles, incluye los datos de los indicadores para medir el estado de cada proyecto, desde que inició hasta que fue entregado al cliente.

El documento con descripción de las gráficas utilizadas en el área de métricas, incluye definiciones y descripción de la gráfica para representar el desempeño de los indicadores de un proyecto.

La presentación que documenta la mejora de velocidad en proyecto piloto con técnicas ágiles, contiene la explicación de cómo determinar la mejora de un proyecto. También, incluye los resultados obtenidos al medir la velocidad de implementación de los 2 primeros Sprints del primer proyecto piloto.

El documento de disminución del tiempo invertido para documentar proyecto piloto, contiene el muestreo de la documentación realizada durante la ejecución del proyecto piloto con técnicas ágiles e incluye muestreos de proyectos con metodología RUP, donde se midió el esfuerzo requerido para documentar el proyecto.

El informe con disminución de la densidad de defectos por producto, contiene los muestreos realizados en el primer proyecto piloto con ágiles y en un proyecto con metodología RUP, donde se midió la densidad de defectos por producto para cada proyecto.

### **7.3. Unidad de análisis III: Adaptación del grupo de trabajo**

La unidad de análisis requiere conocer como ha sido la adaptación de los equipo de trabajo al emplear técnicas ágiles con CMMI-DEV respecto a otras metodologías. Para recoger estos datos se utilizó una encuesta<sup>23</sup> dirigida a los desarrolladores del CEDS, incluyendo al desarrollador del primer proyecto piloto que entrevistamos en la primer unidad de análisis.

Los datos recolectados para esta unidad de análisis fueron cuantitativos y cualitativos, por medio de una encuesta. La perspectiva fue interpretativa y crítica, al considerar el punto de vista de los encuestados, en cuestión de relaciones sociales y laborales, así como la preparación y mejora de los encuestados.

El tener una investigación flexible, permitió modificar el instrumento utilizado para realizar la encuesta, conforme se adquiría conocimiento en el trayecto de la investigación. Esta unidad buscó los beneficios individuales y grupales de los equipos de trabajo, al utilizar la nueva estrategia para desarrollar software. Así como una comparación con otras formas de trabajo.

Las encuestas fueron impresas, aplicadas de manera presencial en el lugar de trabajo de cada desarrollador. Los datos de las encuestas fueron almacenadas en una hoja de cálculo, donde la respuesta a cada pregunta se asoció a una columna. A preguntas con más de una respuesta, se les asignó una columna de la hoja por cada posible respuesta. El nombre de cada columna, estaba relacionado con su contenido.

Una vez capturadas las respuestas de las encuestas, se generó el diccionario de datos con la información de las preguntas y todas sus posibles respuestas. Finalmente, la hoja de cálculo se exporto a un manejador de base de datos gratuito para poder realizar consultas.

---

<sup>23</sup> La encuesta se encuentra en el Apéndice C.

# Capítulo 8

## Análisis de datos

El capítulo detalla el proceso seguido para analizar los datos cualitativos y cuantitativos, respondiendo a las preguntas de investigación de cada unidad y verificación de las hipótesis. El análisis de datos de cada unidad se realizó con métodos específicos, al tipo de datos recolectados en cada unidad de análisis.

### 8.1. Unidad de análisis I: Técnicas ágiles integradas con CMMI-DEV

Los datos analizados en esta unidad, fueron las transcripciones de las entrevistas y las notas recabadas.

Antes de iniciar el análisis se asignó un color al texto de la transcripción de cada entrevista, para identificar al entrevistado que realizó la aportación de los datos, quedando la asignación de colores de la siguiente manera:

- Rosa - Directivo.
- Rojo - Asesor de proyecto piloto y miembro del Grupo de Procesos.
- Azul cielo – Desarrollador del primer proyecto piloto.
- Naranja – Asegurador de Calidad de Software.

Las notas de la entrevista del desarrollador actual del CEDS, no tiene asignado color porque no fueron grabadas. Pero fueron incluidas en la fase de interpretación, para confirmar el uso de técnicas ágiles en el desarrollo de software actual.

### **8.1.1. Codificación de entrevistas**

La información de las entrevistas requiere ser clasificada por aspectos comunes. Esta clasificación se puede realizar con el proceso de codificación, que es una manera de preparar la materia prima para el análisis [43]. La codificación agrupa los datos en bloques de información identificables que facilitan el análisis para una interpretación posterior [43].

A partir de las preguntas de investigación se creó un catálogo de códigos<sup>24</sup> para clasificar la información de las entrevistas. Al ir leyendo cada entrevista, se fue asignando el código correspondiente al tema abarcado por el texto o párrafo. Si el texto no estaba relacionado con algún código existente, se asociaba un código nuevo y se anexaba al catálogo de códigos.

En un nuevo documento de texto se escribieron todos los códigos del catálogo, con estilo de negritas y en color azul marino para distinguir del resto de la información. En este archivo se fue reuniendo la información asociada por código de todas las entrevistas, identificándolas por color.

Al terminar de reunir todas las entrevistas, se leyó nuevamente el documento verificando una asociación correcta del código. Posteriormente, la información de cada código fue reordenada en orden cronológico, prioritario o lógico. Si fue necesario que la información estuviera asociada a otro código, se realizó el cambio.

A continuación, los códigos identificados y su texto correspondiente se anexaron a la pregunta de investigación asociada.

Finalmente, se revisó el archivo resultante para verificar que la información recabada contribuya a dar respuesta a la pregunta de investigación. Si fue necesario, se reordenó la información de las pregunta para lograr una secuencia lógica y contexto adecuado para la respuesta.

---

<sup>24</sup> Ver catálogo de códigos en Apéndice D.

### **8.1.2. Triangulación de las entrevistas**

El siguiente paso que se realizó fue la triangulación, para ver la coincidencia de las aportaciones de los diferentes entrevistados a una misma pregunta de investigación.

La triangulación ofrece diversas perspectivas para mejorar la interpretación del fenómeno en estudio y mejora la calidad de un estudio cualitativo [40]. Utilizar la triangulación mejora la interpretación de las respuestas a cada pregunta de investigación, disminuyendo la posibilidad de malos entendidos [40].

La triangulación se realiza a las respuestas de cada pregunta de investigación, de tal manera que los textos que coincidan para más de un encuestado dan sustento a la respuesta [40]. En tanto, aquellos resultados que no coincidan, dan una mejor perspectiva de lo que ocurre con el fenómeno en estudio, considerando la influencia en el participante [40].

El investigador tiene que ser capaz de rescatar, manipular y recuperar la información más significativa de cada una de las secciones de información codificada [43]. La información ya analizada e interpretada, se reúne en el Capítulo 9 denominado Interpretación de los datos. El archivo se versionaba al procesar una entrevista o al terminar de realizar un nuevo proceso.

### **8.2. Unidad de análisis II: Métricas de proyectos para determinar beneficios**

La perspectiva considerada dentro de esta unidad fue positivista, al buscar evidencia formal, medir variables y cuestionar la hipótesis. El enfoque fue deductivo al explorar, analizar y explicar la información recabada para responder las preguntas de investigación.

A la primer y tercer entrevista se aplicó el proceso de codificación y triangulación, permitiendo conocer el proceso de métricas, determinar el estado de un proyecto a partir de los indicadores<sup>25</sup> y entender la información de los reportes de métricas para proyectos con RUP y técnicas ágiles.

---

<sup>25</sup> Un indicador es una característica del desarrollo de software que puede ser muestreada, medida y graficada para determinar el estado de esa característica en el proyecto. Determinando si el proceso está controlado o no.

## 8.2.1. Elementos de una gráfica para medir un indicador

El documento con la descripción de las gráficas utilizadas en el área de métricas, define los elementos de una gráfica para medir un indicador. Formada por los elementos de la Figura 8-1:

- El eje X representa Sprints, meses o muestreos en un tiempo específico.
- El eje Y representa el número de muestras encontradas en el tiempo específico.
- Línea central representa el valor promedio de las muestras tomadas ( $\mu$ ).
- Límite superior e inferior<sup>26</sup> rodean la línea media, a una distancia determinada respecto a la desviación estándar.

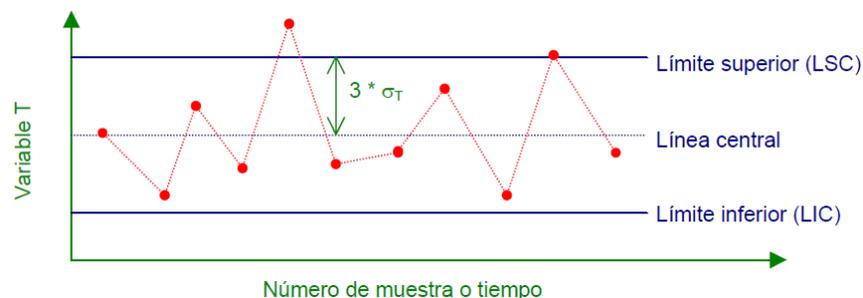


Figura 8-1 Gráfico de control del indicador o variable T.

Muestrear un indicador en un proceso activo, permite graficar y ver su comportamiento. Si la totalidad de puntos del muestreo están dentro de los límites de control, el proceso está controlado. Si un punto del muestreo está fuera de los límites, se dice que el proceso o indicador está fuera de control y es necesario indagar las causas que lo provocó.

Los procesos o indicadores fuera de control son alarmas, permiten detectar posibles problemas antes de aumentar el impacto en el proyecto y los costos por correcciones.

Se tienen dos tipos de gráficos que van juntos, el **Gráfico Individual** proviene de un muestreo individual y el **Gráfico Rango Móvil** consiste de 2 observaciones sucesivas. Ambas gráficas son de procesos activos y del mismo indicador para estimar la variabilidad del proceso.

<sup>26</sup> El límite superior e inferior depende del número de muestras trabajadas, distintos tamaños de muestras formarían límites escalonados y no una línea recta.

## 8.2.2. Mejora de velocidad de implementación

La mejora de la velocidad de implementación fue planteada como un objetivo organizacional. La presentación que documenta la mejora de la velocidad en primer proyecto piloto, fue analizada para comprender la manera como la empresa Px determinó el aumento de la velocidad de implementación.

El valor histórico de la velocidad de implementación de las Tablas de Dimensionamiento<sup>27</sup> de la empresa Px era 2.4654 cp/hr<sup>28</sup>.

Para determinar si hubo mejora en el desarrollo con técnicas ágiles, se realizaron muestreos al mismo indicador de cada Sprint de los primeros proyectos pilotos con técnicas ágiles.

Para una correcta interpretación de los indicadores es imprescindible que los datos muestreados vengan de una distribución aproximadamente normal. En caso de ser visiblemente asimétrica se emplea un método de transformación para aproximar a la normalidad el comportamiento muestreo del indicador para continuar analizando la información.

El muestreo de la velocidad de implementación del Sprint 1 del primer proyecto piloto, fue analizado para ver su distribución. Se utilizó un programa con la prueba de Anderson Darling<sup>29</sup>, para determinar si el comportamiento de las muestras se apegaba a una distribución normal.

El resultado de la prueba indicó que las muestras no cumplían con la distribución normal, lo que indicó que la velocidad real durante la implementación podía tener un comportamiento fuera de control durante el monitoreo, como se muestra en la Figura 8-2. Por lo que se aplicó a este muestreo un método de transformación, para apegar a un comportamiento normal y seguir analizando las muestras.

---

<sup>27</sup> Son tablas que ayudan a estimar el tiempo para realizar las actividades del ciclo de vida de un proyecto, estos valores se determinan en base a datos históricos de la empresa Px. Al realizar el primer proyecto piloto con técnicas ágiles, estos valores estaban basados en desarrollos con metodología RUP.

<sup>28</sup> Puntos de complejidad por hora.

<sup>29</sup> Es una prueba para determinar si una muestra viene de una distribución de probabilidad específica. Es una herramienta muy potente para detectar desviaciones de la normalidad, si el valor de  $P < 0.005$  no es normal [69].

También, se obtuvo el Grafico Individual y el Grafico de Rango Móvil, arrojando una velocidad de implementación promedio de 3.87 cp/hr, como se muestra en la Figura 8-3.

Al comparar la velocidad de implementación 2.47 cp/hr extraída de la Tabla de Dimensionamiento, contra la velocidad de implementación promedio 3.8684 cp/hr, se vió que la velocidad aumentó un 56.27 %. Se realizó el mismo análisis al muestreo del Sprint 2, obteniendo una velocidad de implementación promedio de 5.56 cp/hr mostrando un aumento de 125% en la velocidad.

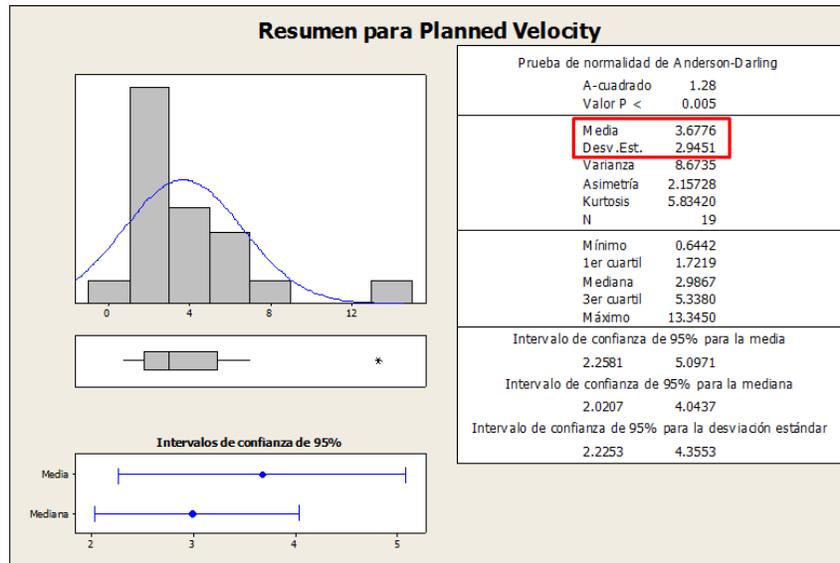


Figura 8-2 Resultado de la prueba Anderson Darling de Sprint 1, extraído de documentos de Px.

La empresa Px realizó la prueba estadística llamada ANOVA<sup>30</sup>, para verificar que la mejora en la velocidad de implementación en los proyectos pilotos no eran casos particulares ni aislados.

Para realizar la primera prueba ANOVA, se consideraron los muestreos de la velocidad de implementación de los Sprints del primer proyecto piloto y otros dos proyectos pilotos. Los resultados de la prueba ANOVA, demostraron que la mejora en la velocidad de implementación no fue un caso aislado.

<sup>30</sup> Análisis de varianzas, utilizado para probar que las medias de tres o más poblaciones son las mismas, contra la alternativa de que no todas las medias de las poblaciones son las mismas. En base a una hipótesis nula y una hipótesis alternativa definida por Px.

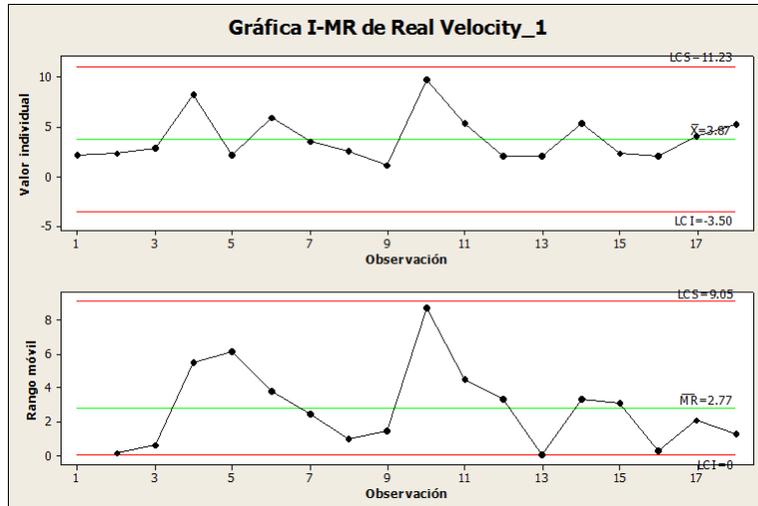


Figura 8-3 Análisis de empresa Px sobre indicador de velocidad de implementación del Sprint 1.

El segundo análisis con la prueba estadística ANOVA, se realizó respecto a un proyecto con procesos desplegados en la institución utilizando técnicas ágiles y utilizando dos proyectos pilotos. Comprobando que no existe diferencia entre la velocidad de implementación de proyectos pilotos con técnicas ágiles respecto a proyectos con procesos desplegados que incluyen técnicas ágiles.

Con los dos análisis se comprobó que la mejora en la velocidad de implementación se mantuvo en proyectos pilotos y proyectos desplegados, demostrando que dicha mejora no es un caso aislado ni particular al proyecto.

### 8.2.3. Disminución de tiempo en elaboración de documentación

El documento de la disminución del tiempo invertido para documentar el proyecto piloto fue analizado para entender cómo se determinó la disminución del tiempo que requiere un proyecto al ser documentado dentro de la empresa Px.

El valor histórico de las Tablas de Dimensionamiento sobre el tiempo requerido por día para documentar un proyecto en la empresa Px con metodología RUP fue 2.02 h. Para determinar si hubo mejora en proyectos ágiles, se realizaron muestreos al mismo indicador de un proyecto piloto con técnicas ágiles.

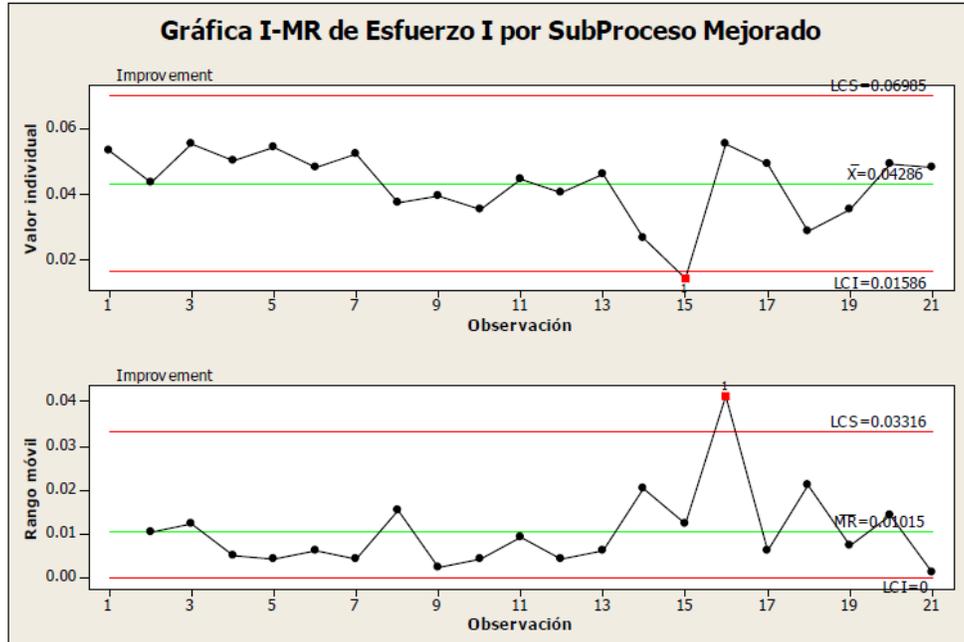


Figura 8-4 Análisis de empresa Px de horas dedicadas a documentar por día en proyecto con ágiles.

La Figura 8-4 visualiza el Gráfico Individual y Gráfico de Rango Móvil con el muestreo de las horas invertidas por día para documentar el proyecto piloto con técnicas ágiles, arrojando un tiempo promedio de 0.04286 h dedicadas a la documentación por día. Con lo que se concluye que se reduce de 2.02 horas a 3 minutos por día, dedicado a elaborar documentación.

#### 8.2.4. Disminución de defectos en el producto final de un proyecto

La presentación con objetivos de mejora organizacional muestra la disminución de defectos con técnicas ágiles como propuesta de mejora en los objetivos organizacionales, documentado la disminución en la ocurrencia de defectos por producto.

El valor histórico de la Tabla de Dimensionamiento sobre el número de defectos por productos finales en un proyecto de la empresa Px con metodología RUP fue de 0.054 defectos por producto. Al muestrear el indicador del proyecto piloto ágil, se obtuvo la gráfica de la Figura 8-5.

La gráfica muestra un valor medio de 0.016 defectos en productos finales, comparando con el valor de la línea base (0.054). Teniendo una reducción de 28.98%, respecto a la línea base de densidad de defectos de la empresa Px.

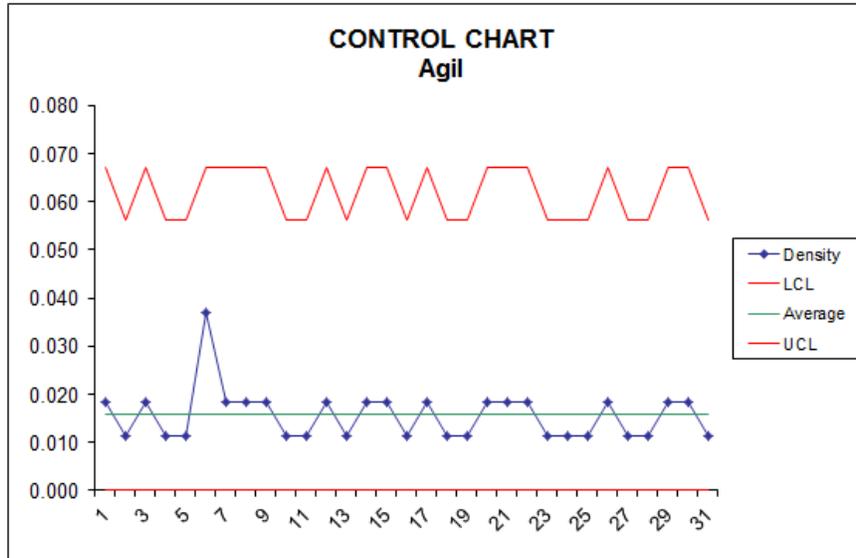


Figura 8-5 Muestreo de densidad de defectos por producto en el primer proyecto ágil.

### 8.3. Unidad de análisis III: Adaptación del grupo de trabajo

Esta unidad de análisis investiga la respuesta de los integrantes de los equipos de trabajo ante la integración de técnicas ágiles con CMMI-DEV para desarrollar software.

Para esta unidad de análisis fueron recabadas 9 encuestas, la cual permiten identificar o visualizar patrones dentro de la muestra [44]. Realizar un análisis sobre las frecuencias de respuesta para cada pregunta, resulta recomendable con la cantidad de muestras que fueron recolectadas [44].

La descripción informativa de un conjunto de datos está dada por la frecuencia con que se repite un evento<sup>31</sup>, denominada frecuencia de clase<sup>32</sup> [44]. En el análisis de la investigación la clase fue representada por cada una de las posibles respuestas de cada pregunta.

Mientras que la frecuencia relativa es la frecuencia de repetición respecto a toda la población y es quién hace evidente la existencia de patrones, dada por la fórmula [44]:

$$\text{Frecuencia relativa} = \frac{(\text{Frecuencia de clase})}{\text{Total de elementos de la muestra}}$$

<sup>31</sup> Un evento es cada una de las repuestas de la encuesta [44].

<sup>32</sup> Una clase representa a un conjunto de datos [44].

Por motivos de la investigación la frecuencia relativa será mostrada como porcentaje, representando el porcentaje de ocurrencia de la respuesta a una pregunta, en toda la población.

El histograma de frecuencias relativas se construye graficando en el eje horizontal las clases o respuestas de la pregunta analizada, y en el eje vertical la frecuencia relativa [44].

Las respuestas de las encuestas fueron almacenadas en una base de datos, lo que facilitó extraer la frecuencia de cada respuesta. Se realizó una consulta por cada pregunta, con la consulta:

```
SELECT respuesta1, COUNT(*)  
FROM encuestas  
GROUP BY respuesta1
```

El resultado de las consultas para todas las preguntas, fue almacenado en una hoja de cálculo. Una vez que se tuvo la frecuencia relativa de las respuestas de cada pregunta, se buscó responder las preguntas de investigación con las respuestas de una o más preguntas de la encuesta.

# Capítulo 9

## Interpretación de datos

La búsqueda de la información fue bajo criterio empírico, dando origen a investigar sobre dos teorías existentes para desarrollar software y que propiciaron la generación de tres hipótesis.

Este capítulo muestra el conocimiento adquirido durante la investigación, por medio de las respuestas a las preguntas de investigación de cada una de las unidades de análisis. Interpretando los acontecimientos ocurridos en Px y defendiendo el estado de cada una de las hipótesis.

### **9.1. Unidad de análisis I: Técnicas ágiles integradas con CMMI-DEV**

La primera unidad de análisis se divide en dos partes, la primera abarca el desarrollo de software utilizando metodología RUP con CMMI-DEV y la segunda comprende la integración de técnicas ágiles con CMMI-DEV.

La información presentada en este capítulo es el resultado del análisis (codificación y triangulación) descrito en el Capítulo 8 y se encuentra señalado entre comillas. También se

incorporó la interpretación de la información recabada sobre lo sucedido en la empresa Px, debajo de la información analizada.

Para hacer referencia a los participantes que aportaron información a la investigación de cada pregunta de investigación, se asignó un acrónimo a cada entrevistado:

1. Miembro directivo de la empresa Px, como Directivo.
2. Asegurador de Calidad de Software, denominado Calidad.
3. Desarrollador de proyecto piloto, llamado Programador1.
4. Asesor de proyecto piloto y miembro del Grupo de Procesos, es Asesor.
5. Desarrollador del CEDS, ubicado como Programador2.

### **9.1.1. Desarrollo de software usando metodología RUP con CMMI-DEV**

Esta sección describe la estrategia, la forma de desarrollar software y la gestión de cambios en los requerimientos, empleando metodología RUP con CMMI-DEV.

La estrategia para desarrollar software determina el estilo de trabajo durante los proyectos , la primera pregunta de investigación busca conocer una de las estrategias de desarrollo utilizadas por la empresa Px.

- a) ¿Cuál es la estrategia que se sigue en un proyecto de desarrollo de software que utiliza metodología RUP con CMMI-DEV Nivel 5?

*Respuesta de Directivo y Programador1*

“El desarrollo de los proyectos debe considerar las 4 fases de la metodología RUP: Inception, Elaboration, Construction y Transition, así como todas las disciplinas transversales para cada una de las fases de RUP. De CMMI-DEV se utiliza casi todas las áreas de proceso.”

“Se debe elaborar toda la documentación requerida para tener la evidencia completa, si alguien sale del equipo, la persona que se integra tiene las herramientas suficientes para continuar.”

Una de las estrategias de la empresa Px para desarrollar software se basa en emplear la metodología RUP bajo el modelo de calidad internacional CMMI-DEV, dando la pauta a un estilo de trabajo siguiendo buenas prácticas. La forma de implementar software utilizando RUP con CMMI-DEV fue definida en la siguiente pregunta.

b) ¿Cómo es el desarrollo de software usando metodología RUP con CMMI-DEV Nivel 5?

*Respuesta de Directivo y Programador1*

“El proyecto inicia con la fase Inception, realizando: definición del proyecto, generación de la línea base de requerimientos y productos, definir plataforma, servidor y lenguaje a usar.”

“Después, en la fase de Elaboration se genera la mayor parte del diseño y arquitectura del sistema con documentación, se realiza el refinamiento de la línea base de requerimientos y productos con el cliente, modelo Entidad-Relación y casos de uso.”

“La forma de trabajar se parece al Modelo en Cascada<sup>33</sup>, pero RUP es más flexible porque no te dedicas únicamente a análisis y diseño, sino que puedes definir casos de prueba a nivel de código, cascaron de la arquitectura o modificar o aclarar requerimientos.”

“La fase de Construction se encarga de implementar el proyecto. El administrador de proyecto asigna las actividades de forma arbitraria a los programadores, quienes trabajan en base a la línea de productos y requerimientos. La funcionalidad es descrita con texto tipo narrativa.”

“Las horas para desarrollar un producto se extraen de la Matriz de Tiempos<sup>34</sup>, considerando el tipo<sup>35</sup> y nivel de complejidad<sup>36</sup> del producto. El tiempo incluye horas asignadas a algunas áreas de procesos, por ejemplo un proceso mediano tiene asignadas 10 horas: 1 para administración de proyecto, 2 de pruebas, 3 de ingeniería de

---

<sup>33</sup> Enfoque tradicional para el ciclo de vida del desarrollo de software [68].

<sup>34</sup> Tabla con el tiempo estimado para desarrollar un producto, extraídos de datos estadísticos de la empresa Px.

<sup>35</sup> Tipos de productos: pantallas, procesos y reportes.

<sup>36</sup> Nivel de complejidad: Simple, Mediana y Compleja

requerimientos, 2 de análisis y diseño y 2 de implementación, por lo que el tiempo para la implementación es muy reducido.”

“En la fase de Transition se realiza el despliegue del producto a los usuarios finales, originando mejoras al producto.”

La forma de trabajar sigue una secuencia similar al Modelo en Cascada, aunque RUP tiene cierta flexibilidad en la ejecución del proyecto. Permite avanzar con actividades de la siguiente fase o regresar a una fase anterior para aclarar algún aspecto o requerimiento. Teniendo como desventaja el poco tiempo asignado a la implementación del producto. En la última pregunta de esta sección, se aborda la manera de realizar un cambio en un proyecto de software que emplea RUP con CMMI-DEV.

c) ¿Cómo se realizan los cambios durante un proyecto de metodología RUP con CMMI-DEV Nivel 5?

#### *Respuesta de Directivo y Calidad*

“El cambio solicitado por el cliente se gestiona con un Control de Cambios<sup>37</sup>, de manera paralela se avanza con los productos que no son afectados por dicho cambio, evitando re trabajar. Cuando el cambio es autorizado se modifica la planeación del proyecto para seguir avanzando.”

Al surgir un cambio por parte del cliente, el proceso con RUP busca afectar lo menos posible el desarrollo de los proyectos. En el momento que es autorizado el cambio, este se incorpora a la planeación.

Estas preguntas de investigación nos han permitido conocer a grandes rasgos cómo se desarrolla software empleando metodología RUP con CMMI-DEV en la empresa Px. Tomando un punto de partida en la forma que trabaja el desarrollo de software la empresa Px.

---

<sup>37</sup> Ver proceso de Control de Cambios en Apéndice E.

## 9.1.2. Integración de técnicas ágiles con CMMI-DEV

La segunda sección de la primera unidad de análisis, describe la manera cómo fueron integradas las técnicas ágiles con CMMI-DEV en el primer proyecto piloto. Rescatando las técnicas ágiles que se utilizan actualmente en el desarrollo ágil de la empresa Px.

La primera pregunta de esta sección busca conocer la motivación de la empresa Px para generar una nueva forma de desarrollar software.

a) ¿Por qué motivo la empresa Px decide integrar técnicas ágiles con CMMI-DEV Nivel 5?

*Respuesta de Directivo, Calidad, Asesor y Programador 1*

“La empresa Px necesitaba obtener la recertificación de CMMI-DEV Nivel 5 para ampliar su participación en licitaciones o RFP<sup>38</sup>, donde es requisito este nivel de calidad. También, buscaba disminuir los costos para bajar el precio final al cliente, mantener la calidad de los productos y colocarse mejor en el mercado.”

Podemos ver que el motivador principal era obtener la recertificación de CMMI-DEV Nivel 5, por el posicionamiento empresarial. Por lo que Px, planeó una estrategia para innovar los procesos en el desarrollo de software y al mismo tiempo disminuir los costos de la producción. La estrategia que empleó se encuentra en la siguiente respuesta.

b) ¿Cuál fue la estrategia que siguió la empresa Px para integrar técnicas ágiles con CMMI-DEV Nivel 5?

*Respuesta de Directivo, Ingeniería y Programador 1*

“Implementar un método de trabajo que se enfocará en el producto y no en la documentación, disminuyendo el tiempo de fabricación y trayendo como consecuencia reducir el costo del producto. Sin olvidar, conservar la evidencia de la ejecución de las prácticas para cumplir con CMMI-DEV.”

---

<sup>38</sup> Request For Proposal, propuesta inicial de requerimientos solicitando el servicio a empresas especializadas a presentar su propuesta técnica que cumpla con los requisitos, frecuentemente en licitación [67].

“Bajo el conocimiento de que CMMI-DEV no es solamente documentación, sino un modelo de control que permite hacer mejoras a los procesos de una organización. Así como, los métodos ágiles son simplemente una forma de trabajar.”

“Considerando estos dos aspectos, se realizó un compendio de técnicas ágiles que se podían aprovechar y adecuar a las prácticas de CMMI-DEV, realizando los ajustes necesarios conforme el negocio los vaya requiriendo. Adoptando técnicas ágiles en administración, planeación, desarrollo, monitoreo, control e ingeniería y se complementó con artefactos de RUP para cumplir con CMMI-DEV.”

Para establecer la estrategia de trabajo Px requería visualizar lo que tiene ahora, a donde quiere llegar y que necesita para lograr su objetivo. Esto le ayudo a encaminar una estrategia de trabajo enfocada en el uso de métodos ágiles, los cuales se enfocan en el producto más que en la documentación. Para implementar una forma de desarrollar software con ágiles, fue necesario seleccionar las técnicas ágiles adecuadas que cubrieran el modelo de CMMI-DEV.

Para institucionalizar los procesos de técnicas ágiles bajo CMMI-DEV en la empresa Px, fue necesario aplicar el Plan de Mejora<sup>39</sup>. Para el cual, una de las actividades es realizar un proyecto piloto con la propuesta. La siguiente pregunta se enfoca en la selección y justificación de técnicas ágiles para desarrollar software.

c) ¿Cuáles fueron los criterios para seleccionar las técnicas ágiles que se incorporaron al desarrollo de software bajo el modelo CMMI-DEV Nivel?

#### *Respuesta de Directivo e Ingeniería*

“Se realizó un mapeo de las prácticas de todas las áreas de proceso que se tienen que cumplir para el modelo CMMI-DEV y después se hizo un barrido para cubrir con técnicas ágiles ese mapa. Observando que utilizar solamente técnicas ágiles no sería suficiente para cubrir el modelo completo, por lo que se agregaron artefactos de RUP en las partes faltantes, logrando una adaptación total al modelo de la organización.”

---

<sup>39</sup> Ver Plan de Mejora en Apéndice F.

“La planeación y administración se realizaron con Scrum porque es una forma de dividir el trabajo de manera simple, identificando prioridades técnicas y del negocio. Esta metodología sufrió algunos cambios para ajustarse a la forma de trabajo de la empresa Px y poder ser adoptada en administración, planeación, retrospectiva, monitoreo y control.”

“Para la implementación se seleccionaron prácticas de integración continua, trabajo en el mismo lugar y programación entre pares de XP<sup>40</sup>. De RUP se conservaron únicamente los artefactos de la arquitectura, planes y pruebas que son básicos para ejecutar el proceso de manera adecuada y no hay una técnica ágil que cubra lo que requiere CMMI-DEV.”

Identificar las necesidades a cubrir con el modelo de calidad CMMI-DEV, permitió establecer las técnicas ágiles que podían ser utilizadas para desarrollar un proyecto de software. Básicamente, Scrum fue utilizado en la administración del proyecto y XP al aplicar prácticas durante la implementación del software. A continuación, se describe como se llevó a cabo la integración de técnicas ágiles manteniendo la coherencia con el modelo de calidad CMMI-DEV.

d) Integración de técnicas ágiles con CMMI-DEV Nivel 5 para desarrollar software.

Al desarrollar software es necesario establecer un plan de trabajo para cumplir con el objetivo del proyecto, abarcando este aspecto con la siguiente pregunta.

- ¿Cómo se genera el plan de trabajo utilizando técnicas ágiles con CMMI-DEV Nivel 5?

*Respuesta de Directivo e Ingeniería*

“Los procesos de técnicas ágiles seleccionadas se emplearon en un proyecto piloto, donde el equipo de trabajo podía seleccionar, reemplazar o eliminar aquellos procesos que les ayudarán a tener buenos resultados y hacer más ágil CMMI-DEV.”

---

<sup>40</sup> Programación Extrema, metodología ágil con prácticas basadas en valores para lograr un propósito específico del desarrollo de software. [45]

“La recolección de los requerimientos se hace en un Sprint, donde el cliente ayuda a recabar la información y documentar al momento, se dimensiona el problema analizando el alcance e inversión disponible, generando el Product Backlog y el objetivo del proyecto.”

“En el Sprint Planning se encuentra presente el equipo Scrum y un administrador de cada área de proceso de CMMI-DEV. En la reunión se dimensiona el proyecto para estimar el esfuerzo a invertir en productos y requerimientos, se define el objetivo del Sprint y se define el Sprint Backlog con productos de prioridad para el cliente y que generan valor al proyecto. Así, todas las áreas tienen la misma visión y realizan las actividades necesarias para lograr el objetivo.”

“El Scrum Master obtiene automáticamente el plan de trabajo, que es un proceso preestablecido con la lista de todas las actividades que se deben seguir durante el proyecto para cada área de proceso. Hay prácticas mínimas necesaria que se deben realizar para cumplir con el modelo, aun cuando el cliente no lo solicite.”

“Cada administrador de área planea, organiza y distribuye las horas que dedicará a cada Sprint y a sus actividades, respetando el tiempo inicial que le fue asignado. La administración de todas las áreas se realiza en la misma herramienta, logrando una administración integral del proyecto y cubriendo el modelo CMMI-DEV.”

En la planeación de un proyecto de software se utiliza la reunión Sprint Planning y los representantes de las áreas de proceso junto con el equipo de trabajo generan el Product Backlog y Sprint Backlog. Cada área de proceso administra su tiempo y actividades para lograr el objetivo del Sprint. La diferencia respecto al proyecto piloto fue la tolerancia de algunas áreas de proceso para apoyar y no interferir con el avance del proyecto piloto.

La ejecución de las técnicas ágiles en el desarrollo de proyecto se describe en la respuesta de la siguiente pregunta.

- ¿Cómo se implementaron las técnicas ágiles con CMMI-DEV Nivel 5?

*Respuesta de Directivo, Ingeniería, Programador1 y Programador 2*

“Inicialmente no se eliminaron procesos, pero se observó que había actividades que no generaban valor real al proyecto, ni le interesaban al cliente y tomaba bastante tiempo hacerlas, iniciando una tala regulatoria<sup>41</sup>. Evitando dejar de hacer documentación por el hecho de ser ágil.”

“Algunas actividades con metodología RUP que se eliminaron, fueron:”

- “Documentación para comprobar que el desarrollador realizó las pruebas unitarias.”
- “Documentación para indicar al equipo de trabajo las pruebas que tenían que realizar.”
- “Al realizar la retroalimentación directamente con el cliente, se disminuyó la triangulación y documentación de las reuniones. Donde, los desarrolladores tenían información de primera mano y no se invertía tanto tiempo en realizar, revisar y proporcionar los resultados de la retroalimentación”

“Al iniciar un proyecto, el equipo de trabajo conoce que”:

- “El equipo tiene asignada una sala para mejorar la comunicación y aclarar dudas.”
- “La documentación tiene un tiempo definido y no se realiza todos los días.”
- “La duración de un Sprint es de un mes (generalmente).”

“El desarrollo de los productos, se administran con las siguientes técnicas ágiles”:

- “La planeación la realiza el equipo de desarrollo con un tablero Kanban<sup>42</sup>.”
- “Los productos se descomponen en tareas más pequeñas.”
- “La estimación la realizan los desarrolladores con la estrategia Póker de Planeación<sup>43</sup>.”
- “Cada tarea tiene asignada un post-it.”
- “En el tablero se pegan todas las tareas de cada producto.”

---

<sup>41</sup> Aquellas cosas que se eliminan tiene que estar justificado.

<sup>42</sup> Los carriles estándar a utilizar son: ‘En espera’, ‘En proceso’, ‘En pruebas’ y ‘Terminado’. Si el equipo de desarrollo lo decide, pueden agregar algún carril adicional.

<sup>43</sup> Cada integrante tiene sus tarjetas de póker, al hacer la estimación levantan la tarjeta con el número de horas de su estimación, si son iguales queda estimada la tarea. En caso contrario, el valor más alto y el más pequeño explican el porqué de su estimación y se vuelve a estimar entre todo el equipo hasta llegar a un acuerdo.

- “El post-it de la tarea incluye nombre de la tarea, identificador del producto y del requerimiento<sup>44</sup>, horas estimadas, invertidas y totales reales.”
- “Al seleccionar las actividades se busca la no dependencia entre compañeros, para que el avance no dependa de una persona.”
- “Cada integrante selecciona las tareas con las que se sienta más cómodo y verifica que las tareas sumen 8 h.”
- “Cada quien toma sus tareas, las pone ‘En Proceso’ y trabaja con ellas durante el día.”
- “Al final del día se actualiza el tablero y se le toman fotos como evidencia diaria.”
- “La reunión Daily Scrum se realiza diariamente, donde se resuelven 3 cosas: ‘¿Qué es lo que hicimos ayer?’, ‘¿Qué vamos a hacer hoy?’ y ‘¿Tenemos algún problema?’.”
- “El tablero tiene una sección de Riesgos, diario se ve si hay algún riesgo, de ser así se registra en un post-it y se pega en el tablero.”

“Durante el Sprint se realizan historias de usuario, diseño, diagramas de clases, diagramas de secuencia, matriz de pruebas, manual de usuario, implementación y pruebas. Si no da tiempo de realizar un producto en ese Sprint, se retoma para el siguiente.”

“El monitoreo y control se realiza con los Daily Scrum y el tablero de Kanban, donde el Scrum Master monitorea el avance del equipo, actividades y riesgos que surjan.”

“Los Daily Scrum se documentan con correos electrónicos y el tablero digital se actualiza todos los días. Al actualizar la información en la herramienta de administración de proyectos, se genera automáticamente el Burn Down Chart <sup>45</sup> que da visibilidad del avance real del proyecto.”

“Al finalizar el Sprint, se realizan actividades para concluir de manera apropiada el ciclo”:

- “La tercera semana de cada Sprint está enfocada a realizar pruebas y correcciones.”
- “Mostrar el Demo funcionando con los requerimientos definidos en el Sprint.”

---

<sup>44</sup> Aportación del equipo piloto.

<sup>45</sup> Gráfica de avance del trabajo a realizar en función del tiempo, tareas pendientes, trabajo realizado, tareas completadas, pronóstico de tiempo en tareas faltantes considerando las tareas realizadas [71].

- “Se realiza el Sprint Retrospective, con lecciones aprendidas, anotando lo que funcionó para seguirlo haciendo, lo que se debe dejar de hacer y las propuestas de mejora.”

“Al terminar el primer proyecto piloto se definieron las técnicas que funcionaron y cuales no fueron útiles durante el desarrollo de software con CMMI-DEV. Las técnicas ágiles exitosas fueron documentadas y desplegadas todas al mismo momento en la organización.”

“La implementación del proyecto piloto se realizó con el apoyo del compendio de procesos de técnicas ágiles bajo CMMI-DEV. Scrum fue considerado el método ágil base para administrar los proyectos, el tablero de Kanban la herramienta de planeación y control de actividades, así como XP establece prácticas y principios para lograr el alcance del proyecto. Estas técnicas ágiles fueron aplicadas en el primer proyecto piloto de Px y actualmente forman parte de los procesos para el desarrollo de software de la empresa”.

Cada actividad que se realiza cumple una meta dentro de las prácticas de CMMI-DEV, al eliminar o sustituir algún proceso tiene que tener una finalidad, justificación o cumplir con la evidencia solicitada por el modelo de calidad.

Aplicar cambios dentro de los proyectos de desarrollo de software, es una característica propia de las técnicas ágiles. Los cambios en un proyecto se pueden presentar en cualquier parte del ciclo de vida de un proyecto y bajo diversas circunstancias, sin importar el momento en que se encuentre el desarrollo. De ahí surge la próxima pregunta.

- ¿Cómo se realizan los cambios durante un proyecto con técnicas ágiles con CMMI-DEV Nivel 5?

*Respuesta de Directivo, Ingeniería y Programador 1*

“Por esencia las técnicas ágiles están preparadas para hacer cambios controlados, pero antes de aplicarlos tienen que gestionarse con un Control de Cambio. El Product

Owner<sup>46</sup> es el responsable de determinar la prioridad del cambio, mientras que el equipo Scrum especifica la prioridad técnica.”

Algunos criterios que influyen al evaluar un cambio, considerando el impacto técnico:

- “Cambios mínimos como un campo o una regla de negocio que no se había entendido, se debe aplicar el cambio dentro del Sprint.”
- “Cambio menor, se deja terminar el Sprint y el cambio se incorpora al Product Backlog para realizar la planeación del siguiente SPRINT.”
- “Aumento del alcance, adicionando productos o módulos al proyecto, al no tener impacto en el Sprint actual se agrega al Product Backlog.”
- “Cambio de prioridades, fuera del alcance o de gran impacto, se detiene el Sprint.”

“La cuestión de cambios se está madurando en la organización Px. La única forma de que un cliente puede hacer un cambio cuando hay un Sprint en proceso, es parando el Sprint.”

El Control de Cambios es un proceso organizacional, es decir, está estandarizado independientemente del método para desarrollar software. Al surgir la necesidad de aplicar un cambio en el software se tiene que detener totalmente el Sprint, lo que podría resultar impactante en el avance del proyecto y en el ritmo de trabajo del equipo. Integrar el cambio de una manera más sutil podría mitigar las consecuencias y costos, pero favorecer el avance.

Al trabajar bajo un modelo de calidad se tiene la necesidad de realizar mejoras en los procesos organizacionales, originando a la pregunta siguiente.

- ¿La empresa Px cuenta con un plan de mejora para el desarrollo de software?

*Respuesta de Directivo e Ingeniería*

---

<sup>46</sup> Representa el cliente, tiene amplio conocimiento de la normativa y el dominio de negocios y experto en el uso del producto [72].

“La empresa Px cuenta con un Plan de Mejora<sup>47</sup> organizacional, que tiene como objetivo aumentar las habilidades de los miembros de la fábrica de software, manteniendo el nivel de calidad de los procesos, la mejora continua, promover la innovación y mantener el Nivel 5 de CMMI-DEV.”

“Otras sugerencias llegan de las juntas de lecciones aprendidas, al solicitar retroalimentación del proceso o de la Dirección de la empresa Px. Lo único que está prohibido es no seguir el proceso y ningún proceso aplica si no pasa por el ciclo de mejora. La integración de técnicas ágiles con CMMI-DEV, es un cambio que pasó por el ciclo de mejora.”

La empresa Px cuenta con un Plan de Mejora permanente a nivel organizacional. Al modificar o agregar procesos en la organización, se requiere permear al resto de la empresa para que funcione de manera apropiada. Este plan de mejora está enfocado en los procesos, pero va de la mano con optimizar y facilitar el trabajo realizado por los individuos, sin olvidar su apego al modelo de CMMI-DEV.

Haciendo necesario conocer si está nueva forma de desarrollar software reestructuro algún aspecto de la empresa Px.

- ¿Cómo se reestructuró la forma de trabajo de la empresa para integrar las técnicas ágiles con CMMI-DEV Nivel 5?

#### *Respuesta de Ingeniería*

“La empresa Px trabaja bajo procesos, lo que hacía falta conocer era la particularidad del proceso. Por lo que, no realizó una reestructuración como tal, porque únicamente se agregaron y modificaron procesos para desarrollar conservando las prácticas de CMMI-DEV. Lo que si se requirió integrar fue capacitación y coaching a las áreas involucradas.”

La integración de una nueva forma de trabajo no provoco cambios empresariales, porque la organización trabaja bajo procesos y fue como ampliar la cantidad de procesos que pueden ser

---

<sup>47</sup> Ver Proceso de Plan de Mejora en Apéndice F.

utilizados en el desarrollo. Esta innovación de procesos fue acompañada de capacitación para equipos de trabajo y áreas de proceso.

Las áreas de proceso relacionadas modificaron determinadas actividades relacionadas con el desarrollo, pero continuaba su presencia en el proyecto. Con la siguiente pregunta se busca identificar algunas áreas de proceso afectadas por la integración de técnicas ágiles.

- ¿Qué áreas de proceso y qué actividades se eliminaron o implementaron para desarrollar software que emplea técnicas ágiles con CMMI-DEV Nivel 5?

*Respuesta de Ingeniería, Calidad y Programador1*

“El área de proceso de Capacitación Organizacional, organizar capacitación para permear los procesos de todas las áreas involucradas, al personal que participa en los proyectos.”

“Del área de Administración de la Configuración, realiza diariamente versionado y respaldos de los productos generados. Las actividades asignadas se realizan con mayor frecuencia.”

“El área de proceso de Monitoreo y Control de Proyecto, modificación en los tiempos para emitir los reportes y conocer las alertas de indicadores que tienen mayor impacto en este tipo de proyectos. El Scrum Master, es quién implementa acciones correctivas en su plan de trabajo.”

“El área de proceso de Aseguramiento de Calidad de Procesos y Productos, acompaña todo el tiempo al equipo ágil en los Daily Scrum, donde realiza listas de verificación sobre lo que se expone y menciona los problemas o desviaciones que se están presentando para que el equipo esté enterado y establezca compromisos para resolverlas. Aumenta su presencia en los proyectos con técnicas ágiles, para verificar que el proyecto cumpla con la integridad y tiempo planeado, que generen la documentación correspondiente.”

“El área de proceso de Verificación, realiza pruebas desde el inicio al proyecto, define las Matrices de Prueba<sup>48</sup> para que el equipo construya con base en ellas y ejecute todos los caminos antes de liberar.”

Se puede observar que la función que realiza el área de proceso no cambia, sino que se modificó la manera o frecuencia con la que se cumplían ciertas actividades. Aunado a los programas de capacitación institucionales.

e) ¿Cuáles son las diferencias de trabajar CMMI-DEV Nivel 5 con metodología RUP respecto a las técnicas ágiles?

#### *Respuesta de Ingeniería y Programador 1*

“El uso de técnicas ágiles da una ventaja competitiva enorme ya que se están generando resultados satisfactorios en los proyectos conforme a lo planeado, el enfoque está sobre el producto y no tanto sobre la documentación. El desarrollo ágil permite tener visibilidad del proyecto y los riesgos se detectan todos los días.”

“RUP maneja las disciplinas muy marcadas, si no se tiene una actividad terminada no se puede pasar a la siguiente, comparado con ágiles que es más flexible en la realización de las actividades, aunque no mantiene un proceso tan ordenado como RUP.”

“El cliente ve el producto real y no un documento, dando retroalimentación al equipo de trabajo para tomar acciones sobre el producto final.”

“La organización definió una regla para determinar el método de trabajo en base al tamaño del proyecto, para proyectos con duración de 6 meses o menos utilizaría métodos ágiles y los mayores a 6 meses se trabajarían con RUP; en la parte de Ingeniería se da la opción a decidir con cuál de las dos metodologías se trabajará.”

Ambos métodos para desarrollar software tienen aspectos que pueden resultar beneficiosos para el producto final y el criterio de la organización para asignar una u otra metodología de

---

<sup>48</sup> Narran los caminos de éxito y los de fallos de los productos.

trabajo depende del tamaño del proyecto, permitiendo proteger el trabajo de todo el equipo y lograr cumplir con el objetivo del proyecto.

Aunque la retroalimentación oportuna y directa al equipo de trabajo es una ventaja de las técnicas ágiles sobre RUP, trayendo como consecuencia disminución de retrabajo por corregir defectos y aumento en la calidad de los productos generados.

Una parte de la satisfacción de los clientes se da por los resultados obtenidos con el proyecto, lo cual ha sido satisfactorio logrando un mejor posicionamiento en el mercado, tema que es abarcado por esta última pregunta de la investigación.

f) ¿Cómo responde el cliente ante las iteraciones rápidas?

#### *Respuesta de Directivo e Ingeniería*

“La respuesta de los clientes ante esta forma de trabajo ha rendido buenos frutos. Inclusive hay algunos que van a regresar por la misma forma de trabajo. Los clientes han felicitado a los grupos de trabajo, inclusive han comentado que se hará una segunda fase y que abrirán más aplicaciones por la satisfacción de los resultados.”

Delimitar el alcance del producto de software, permite acotar y cotizar un producto tangible, donde se resguarda la solicitud del cliente y el compromiso de la empresa de desarrollo. La entrega de productos funcionales al terminar un Sprint, brinda confianza por parte del cliente sobre el trabajo realizado.

Las preguntas de investigación no solamente nos han permitido conocer cómo se fue integrando una nueva manera de desarrollar software en la empresa Px con CMMI-DEV Nivel 5, sino la forma general de desarrollar software por medio de RUP. Permitted establecer diferencias, beneficios y aspectos importantes a conservar de ambos métodos de trabajo. De esta manera se logra concluir el análisis e interpretación de la información recolectada por medio de entrevistas para la *Unidad de Análisis I: Técnicas ágiles integradas con CMMI-DEV*.

### 9.1.3. Comprobación de la hipótesis de la unidad de análisis I

Por medio del enfoque deductivo se realizaron las siguientes actividades:

- Explorar los motivadores que influyeron para llevar a cabo la integración de técnicas ágiles dentro de la empresa Px. Desde el punto de vista de las personas que intervinieron en el proyecto piloto.
- Describir la forma como se llevó la integración en las diferentes fases del ciclo de vida del desarrollo de software.
- Analizar la información recabada para descubrir conocimiento sobre la integración de las técnicas ágiles.
- Explicar la información asociada a cada pregunta de investigación en base a las entrevistas, buscando aceptar o rechazar la hipótesis a partir del conocimiento obtenido.

El primer objetivo específico pertenece a esta unidad de análisis y consiste en: *Identificar las técnicas de los métodos ágiles y la manera de cómo se integraron con CMMI-DEV Nivel 5 en la empresa Px.*

La hipótesis asociada a este objetivo específico, dice que: *Se pueden modificar los procesos establecidos con metodología RUP utilizando las técnicas ágiles Scrum y Kanban sin dejar de cumplir con el Nivel 5 de CMMI-DEV.* En la respuesta de la pregunta: *¿Cuál fue la estrategia que siguió la empresa Px para integrar técnicas ágiles con CMMI-DEV Nivel 5?* El segundo párrafo menciona que CMMI-DEV es un modelo de control que permite hacer mejoras a los procesos de una organización y que los métodos ágiles son simplemente una forma de trabajar.

Para la pregunta: *¿Cuáles fueron los criterios para seleccionar las técnicas ágiles que se incorporaron al desarrollo de software bajo el modelo CMMI-DEV Nivel?* En el primer párrafo se mencionó que las técnicas ágiles cubren una parte del modelo de CMMI-DEV y con algunos artefactos de RUP se logra una adaptación total al modelo de la organización. En el segundo párrafo, se menciona que las características de Scrum permitieron realizar la planeación, administración, monitoreo y control de los proyectos.

Las preguntas: ¿Cómo se genera el plan de trabajo utilizando técnicas ágiles con CMMI-DEV Nivel 5? y ¿Cómo se implementaron las técnicas ágiles con CMMI-DEV Nivel 5? Explican la forma como se integraron técnicas ágiles de Scrum, Extreme Programming y Kanban en el primer proyecto piloto con ágiles.

En las respuestas de las preguntas: ¿Cómo se reestructuro la forma de trabajo de la empresa para integrar las técnicas ágiles con CMMI-DEV Nivel 5? y ¿Qué áreas de proceso y que actividades se eliminaron o implementaron para desarrollar software que emplea técnicas ágiles con CMMI-DEV Nivel 5? Se menciona que no hubo una reestructuración en Px al integrar las técnicas ágiles a sus procesos organizacionales, sino únicamente se realizó capacitación y adecuación de las actividades de las áreas de procesos.

Comprobando que las técnicas ágiles Scrum y Kanban pueden reemplazar algunos procesos establecidos con metodología RUP, sin dejar de cumplir con el Nivel 5 del modelo de calidad CMMI-DEV. Para cumplir con el modelo completo, se requirió utilizar también Extreme Programming y artefactos de RUP, por lo que se determinó que la hipótesis fue aceptada.

## **9.2. Unidad de análisis II: Métricas de proyectos para determinar beneficios**

En esta sección se muestra el resultado del análisis realizado por la empresa Px para determinar aumento en la productividad, disminución de tiempos para implementar software, disminución de defectos y reducción de los costos en todo el proyecto de software ágil.

d) ¿Cómo se determinó el aumento de productividad de un proyecto de software con técnicas ágiles respecto a uno con metodología RUP?

La línea base de desarrollo de software con metodología RUP, indica que la velocidad de implementación es de 2.47cp/h. En el primer Sprint se mejoró la velocidad aumentando a 3.87cp/h y en el segundo Sprint se aumentó la velocidad a 5.56cp/h.

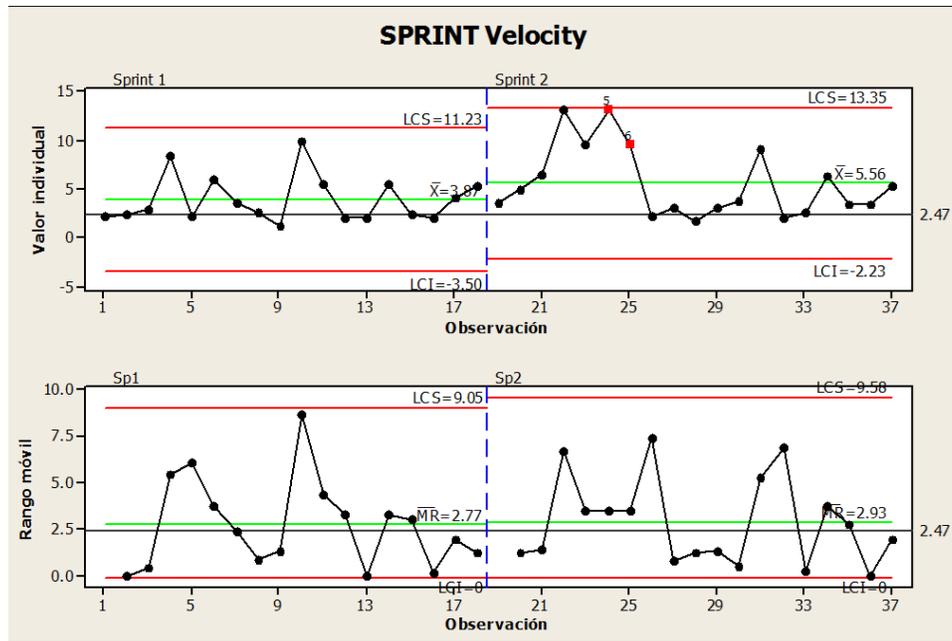


Figura 9-1 Análisis de velocidad de implementación de muestreo en Sprint 1 y 2.

En la Figura 9-1, la línea base aparece de forma horizontal en color negro, la velocidad de implementación promedio de cada Sprint está en color verde, la línea vertical punteada en color azul separa del lado izquierdo el muestreo del primer Sprint y del lado derecho la información del segundo Sprint.

La velocidad de implementación con los proyectos pilotos llegó hasta 7.07cp/h, lo que le permitió establecer a la empresa Px un rango en la velocidad de implementación, que iba desde 3.87cp/h hasta 7.07cp/h, donde la velocidad dependerá de las habilidades de los integrantes del equipo, de las características del proyecto y del mismo cliente.

El valor mínimo del rango tendría un aumento del 56% en la velocidad de implementación y podría aumentar a más del 100% si se trata de un equipo con mayor experiencia.

La ocurrencia de riesgos con el equipo de trabajo puede afectar la productividad de un proyecto, por lo que la empresa estableció su línea base en 3.87cp/h, utilizando esta velocidad en un desarrollo ágil para la planeación y cotización del proyecto.

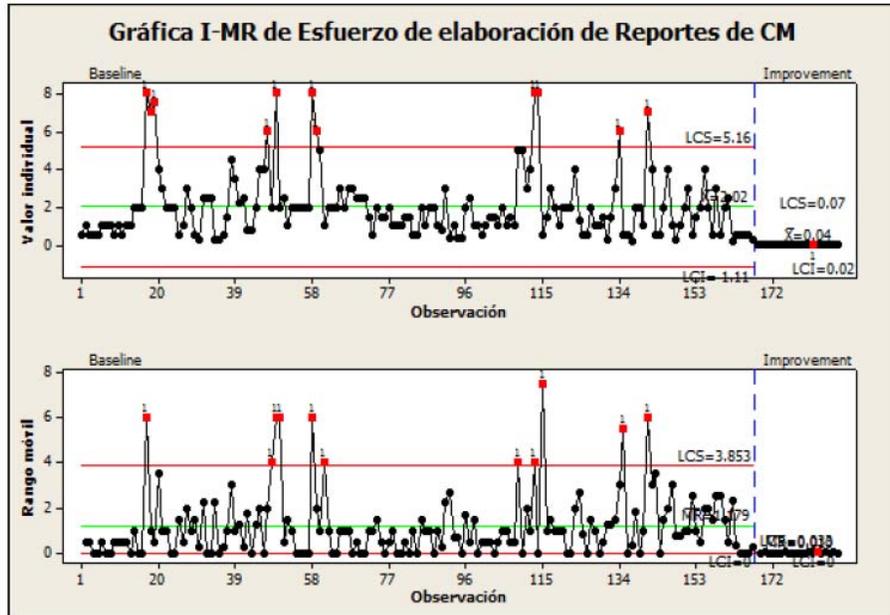


Figura 9-2 Análisis de horas dedicadas a la elaboración de los reportes de la documentación.

Otro aspecto que favoreció la productividad fue la disminución de los reportes de la documentación, donde la línea base de proyectos con RUP se encontraba en 2.02h dedicadas a documentar por día.

Al trabajar con técnicas ágiles el tiempo para documentar disminuyó a 0.04286h en promedio por día, como se muestra en la Figura 9-2. El disminuir la documentación con técnicas ágiles, permite que los equipos de trabajo dediquen mayor cantidad de tiempo a la implementación.

En la Figura 9-2, el lado derecho de la línea punteada color azul se encuentra el muestreo dedicado a elaborar reportes con técnicas ágiles y del lado izquierdo de la línea punteada se encuentra los reportes de la documentación con RUP.

El aumento de la velocidad de implementación y la disminución de la documentación, provocaron una disminución en el tiempo requerido para desarrollar un proyecto de software con técnicas ágiles y en consecuencia tener mayor productividad en el proyecto.

e) ¿Cómo se determinó que se tuvo una reducción de defectos?

En la Figura 9-3 se visualiza el muestreo del primer proyecto piloto ágil y el muestreo de un proyecto con metodología RUP, obteniendo una disminución de 28.98% en la densidad de defectos del proyecto ágil respecto a uno con RUP y la línea base.

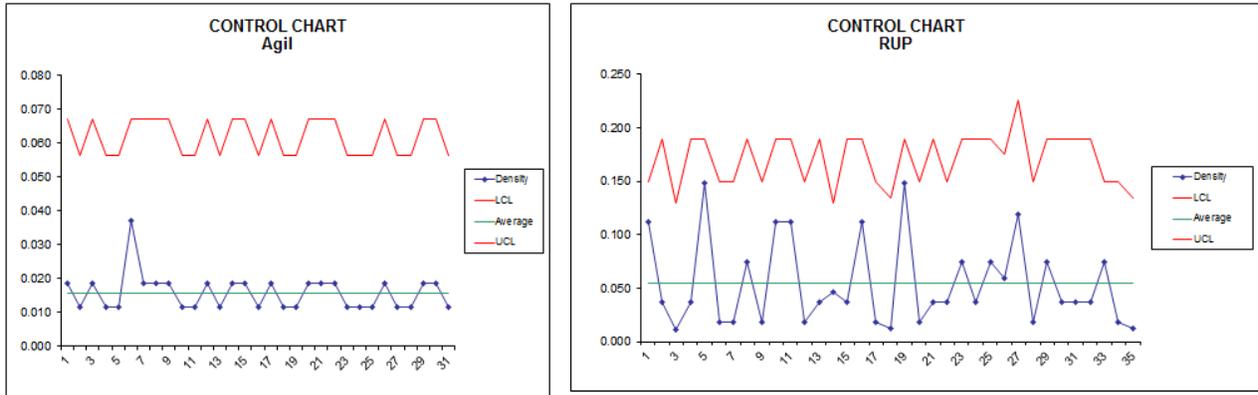


Figura 9-3 Cálculo de reducción de defectos en un proyecto con técnicas ágiles contra otro con RUP.

f) ¿Cómo se determinó la reducción en los costos de desarrollo de un proyecto empleando técnicas ágiles?

Al reducir la documentación, aumentar la velocidad de implementación y disminuir los defectos provoca una disminución en el tiempo que lleva desarrollar implementar el proyecto, lo que implica un decremento en los costos que se requieren para desarrollar software.

### 9.2.1. Comprobación de la hipótesis de la unidad de análisis II

El enfoque deductivo de la investigación oriento las siguientes actividades:

- Explorar los datos históricos que manejaba la empresa Px para representar la línea base de las métricas de interés.
- Describir la manera como se llevó a cabo el análisis estadístico sobre las métricas de los proyectos pilotos para poder determinar los beneficios.
- Analizar las gráficas obtenidas de los resultados probabilísticos de la información para generar conocimiento sobre esta unidad de análisis.

- Buscar la mejora de los procesos para desarrollar software utilizando técnicas ágiles bajo procesos CMMI-DEV, considerando las métricas seleccionadas.

El segundo objetivo específico de la investigación, está asociado a la unidad de análisis II, que dice: *Realizar la comparación de métricas de productividad y calidad en proyectos que utilizaron metodología RUP con CMMI-DEV contra los que integraron técnicas ágiles con CMMI-DEV.*

La hipótesis relacionada a este objetivo específico, es la siguiente.

*Los beneficios logrados con la integración de técnicas ágiles fueron:*

- A. Disminuir tiempos de desarrollo y costo.*
- B. Mejorar la calidad del producto.*

Las preguntas: ¿Cómo se determinó el aumento de productividad de un proyecto de software con técnicas ágiles respecto a uno con metodología RUP? y ¿Cómo se determinó la reducción en los costos de desarrollo de un proyecto empleando técnicas ágiles? Expresan una notoria reducción en la velocidad para implementar puntos de función, así como una disminución en el tiempo invertido para realizar los reportes de la documentación. Lo que reduce el tiempo necesario para desarrollar un proyecto de desarrollo de software y en consecuencia provoca disminuir los costos del software.

La respuesta a la pregunta: ¿Cómo se determinó que se tuvo una reducción de defectos? Nos menciona que se tuvo reducción de defectos en los productos finales, lo que nos indica que se mejoró la calidad del producto entregado respecto a la metodología RUP.

Esta misma respuesta nos dice que se tuvo disminución en el re-trabajo, al invertir menos tiempo para eliminar errores. Estas mejoras ayudan a entregar a tiempo los productos o disminuir el tiempo de dicha entrega, lo que también beneficia a reducir los costos que se invierten en un proyecto de software.

Al aumentar la velocidad de implementación, disminuir el tiempo dedicado a documentar y al re-trabajo, se genera un decremento tanto en el tiempo de desarrollo como en los costos del

desarrollo. Y no solo mantener la calidad, sino mejorarla al disminuir los defectos en los productos finales. Con lo que se comprueba la hipótesis de esta unidad de análisis.

### **9.3. Unidad de análisis III: Adaptación del grupo de trabajo**

Esta unidad de análisis responde a las preguntas de investigación referentes a la adaptación de los grupos de trabajo al desarrollar software utilizando técnicas ágiles con CMMI-DEV.

Tanto las técnicas ágiles como la metodología RUP trabajan con CMMI-DEV, por lo que se omitirá en las respuestas la referencia con CMMI-DEV por motivos de simplicidad. No significa que no se esté trabajando con procesos o no se estén considerando.

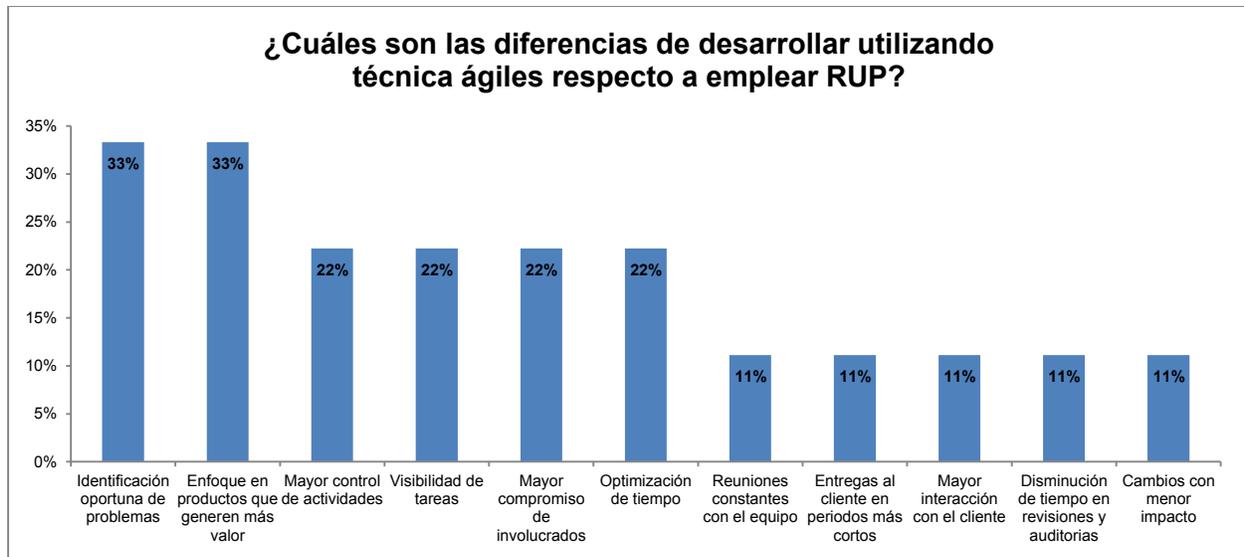
d) Características del desarrollo de software usando técnicas ágiles con CMMI-DEV.

Para obtener las características del trabajo empleando técnicas ágiles se realizó un comparativo con el desarrollo utilizando metodología RUP. Los aspectos abordados son diferencias y ventajas de desarrollar con técnica ágiles respecto con RUP.

- ¿Cuál es la diferencia de trabajar CMMI-DEV utilizando técnicas ágiles respecto a emplear metodología RUP?

En la Figura 9-4, se tienen las 11 respuestas de los encuestados para diferenciar entre ambas formas de trabajo. Las respuestas que tuvieron mayor frecuencia con un 33% fueron 'Identificación oportuna de problemas', 'Enfoque en productos que generan mayor valor', estas dos respuestas son características de los métodos ágiles que permiten aumentar el valor y calidad del producto final.

El grupo de respuestas con el 22% de frecuencia, 'Mayor control de actividades', 'Visibilidad de tareas', 'Mayor compromiso de los involucrados' y 'Optimización de tiempo', facilitan el trabajo de los integrantes del equipo para desarrollar las actividades necesarias del proyecto.



**Figura 9-4 Diferencias de desarrollar utilizando técnicas ágiles respecto a emplear RUP.**

- ¿Cuáles son las ventajas de integrar técnicas ágiles con CMMI-DEV?

Al abordar la sección de ventajas, lo primero que se preguntó fue ‘¿Tiene ventajas usar técnicas ágiles con procesos CMMI-DEV sobre otras metodologías?’. A esta pregunta el 100% de los encuestados contestó afirmativamente. Lo que indica aceptación a esta forma de trabajo y poca resistencia a la implementación de ágiles con CMMI-DEV.

En la Figura 9-5, muestra las 6 ventajas que han detectado los encuestados de incorporar técnicas ágiles para desarrollar software. La respuesta ‘Mayor control sobre actividades’ tuvo un 56% de menciones, lo que muestra que el equipo de trabajo conoce las actividades que tienen que realizar para completar los productos solicitados en el Sprint y pueden tener control sobre actividades conocidas. Las respuestas con el 22% de frecuencia van relacionadas con la ‘Visibilidad del cliente’ hacia el producto, lo que ayuda a tener ‘Menos errores’, y provocar equipos de desarrollo ‘Más eficientes’ y que el desarrollo sea ‘Más flexible’. El 11% considero que ‘Entregas tempranas’ del producto es una ventaja y está relacionada con brindar ‘Visibilidad del cliente’ hacia el producto.

Las dificultades encontradas al usar técnicas ágiles, ayudan a mejorar y lograr beneficios a esta forma de trabajo. El 78% de los encuestados ha detectado dificultades al usar técnicas ágiles y el 22% restante no encontró dificultad en la integración, como se muestra en la Figura 9-6.

Las personas que encontraron dificultad fueron los participantes en la integración de las técnicas ágiles o proyecto piloto. Mientras que las personas nuevas se han integrado a desarrollar con procesos ágiles ya establecidos y no han encontrado mayor dificultad con esta forma de trabajo.

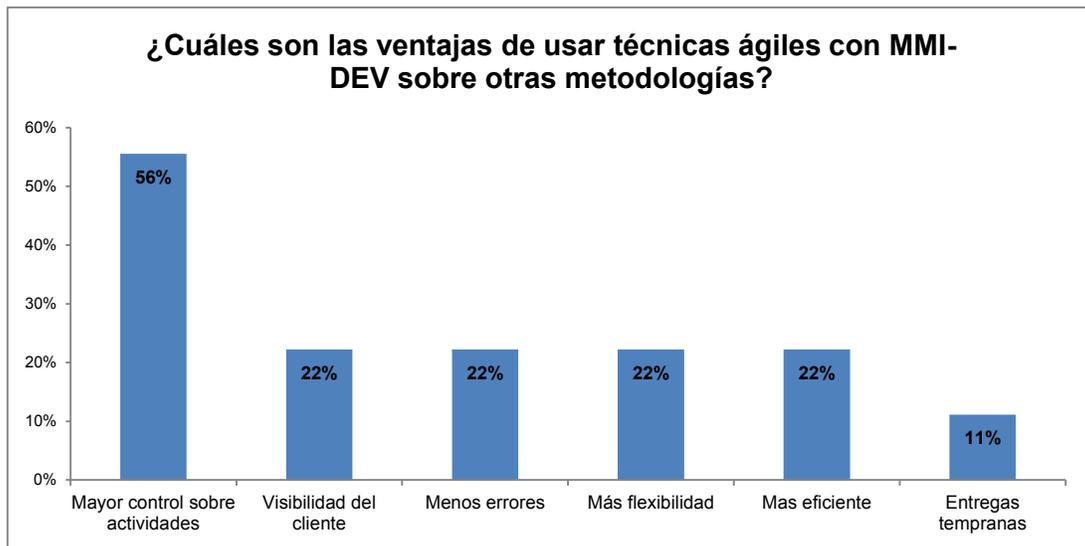


Figura 9-5 Ventajas de desarrollar software usando técnicas ágiles.

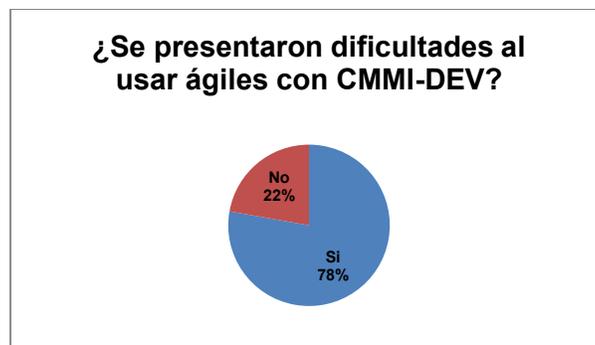


Figura 9-6 Presencia de dificultades al usar técnicas ágiles.

En la Figura 9-7 se visualizan las 4 respuestas sobre las dificultades encontradas en el trabajo con técnicas ágiles. Donde la respuesta con 78% de frecuencia es 'No es cotidiano, común o sencillo combinar técnicas ágiles con CMMI-DEV' y el 33% por la 'Adaptación al cambio' fueron dificultades encontradas durante el proceso de selección, integración e implementación de técnicas ágiles con CMMI-DEV. Las respuestas 'El trabajo depende de la sinergia del equipo' y 'Rechazo por falta de conocimiento' tuvieron un 11%, son dos aspectos a cuidar para que los integrantes del equipo optimicen el trabajo por medio de confianza y seguridad individual.

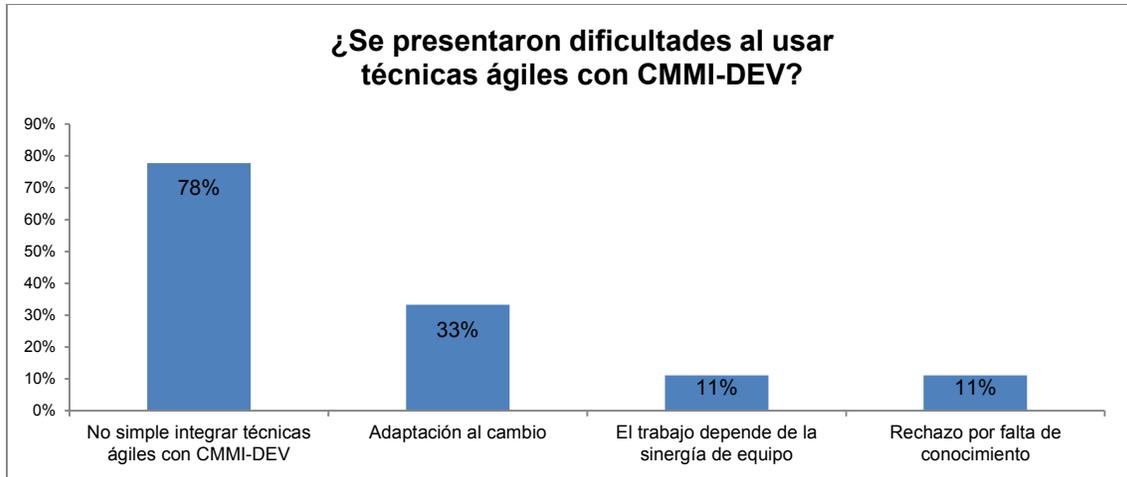


Figura 9-7 Dificultades al desarrollar software usando técnicas ágiles con CMMI-DEV.

- ¿Se disminuyó el tiempo para desarrollar software utilizando técnicas ágiles?

La Figura 9-8 muestra que el desarrollo con técnicas ágiles no solamente ha cumplido en el 100% de los casos con el tiempo planeado, sino que el 44% considera que se ha desarrollado más rápido que con otras metodologías. Aunque el 56% piensa que solamente se ha desarrollado un poco más rápido con técnicas ágiles.

En la Figura 9-9 se muestra que el 22% de los encuestados no acostumbran trabajar los fines de semana y el 56% trabaja de forma ocasional. Mientras que el 22% trabaja el fin de semana. Todas las personas que trabajan de manera ocasional los fines de semana asisten en promedio de 1 a 2 días mensuales. Ninguna persona encuestada trabaja de 7 a 8 días por mes, como se muestra en la Figura 9-9.

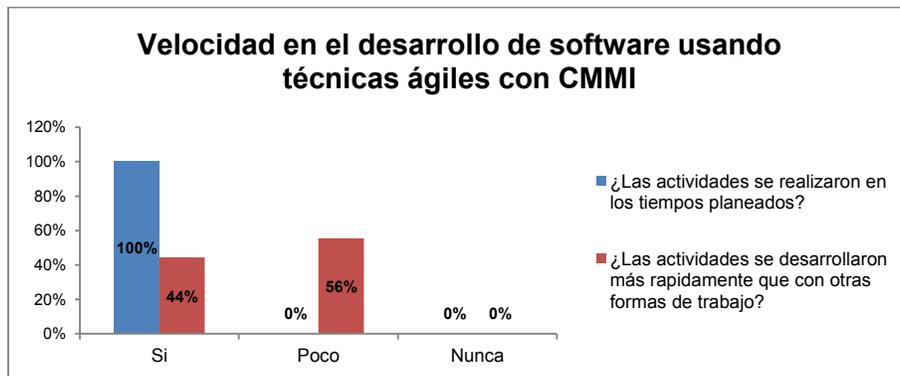


Figura 9-8 Velocidad en el desarrollo de software al usar técnicas ágiles con CMMI-DEV.

Al desarrollar con técnicas ágiles, los equipos de trabajo se organizan para desarrollar las actividades, estiman las actividades y son ellos quienes deciden si se requiere trabajar tiempo adicional o no. El 22% de los encuestados no trabaja los fines de semana, al organizar su tiempo durante la semana ayuda a mantener esta constante sin afectar la productividad de sus actividades.



Figura 9-9 Porcentaje de encuestados que trabajan los fines de semana con determinada frecuencia.

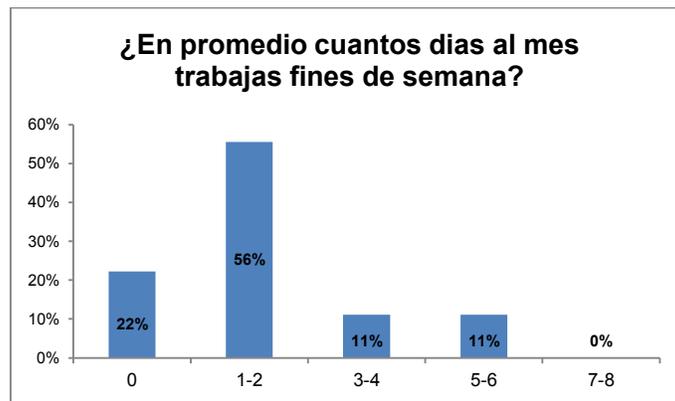


Figura 9-10 Promedio mensual de días trabajados durante el fin de semana.

- ¿El trabajo con técnicas ágiles mejoró el ambiente de trabajo?

La Figura 9-11 muestra que el 78% de los encuestados consideran que si hubo un aumento en la productividad y mejora del ambiente de trabajo, así como un 89% piensa que los integrantes de un equipo de trabajo se apoyan entre sí.

Únicamente el 22% consideran que hubo poco incremento tanto en la productividad como en mejorar el ambiente de trabajo y solamente el 11% piensa que los integrantes de un equipo de

trabajo se apoyan poco. Nadie piensa que la productividad y el ambiente de trabajo es igual que con otras metodologías.

Al observar la gráfica, se detecta que el apoyo entre los integrantes del equipo ayuda de manera proporcional a mejorar el ambiente de trabajo, estableciendo comunicación entre ellos que es una característica de los equipos ágiles. Estos dos factores nos traen como consecuencia aumentar la productividad del equipo, estableciendo una conexión entre los tres aspectos.

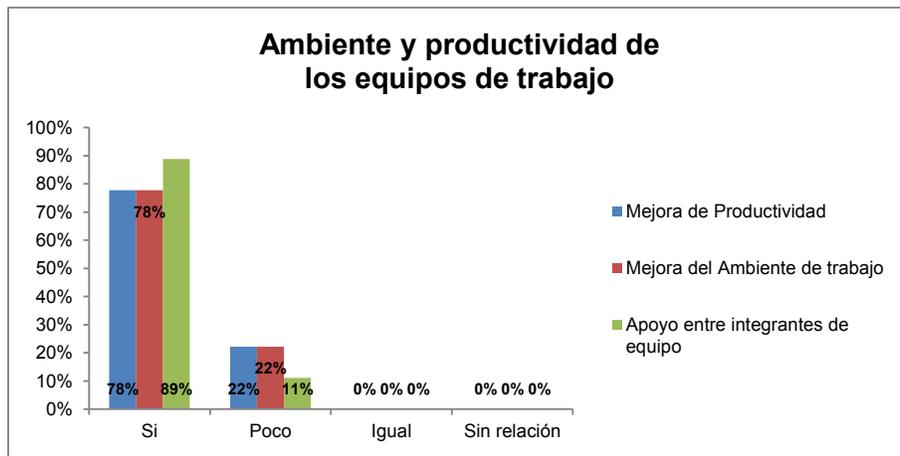


Figura 9-11 Ambiente y productividad de los equipos de trabajo.

Los aspectos que los encuestados consideran que han ayudado a mejorar el ambiente de trabajo se muestran en la Figura 9-12. Donde la respuesta con mayor frecuencia es 'Mejor comunicación con el equipo' con un 78%, seguida con un 33% de la respuesta 'Mejor definidas las actividades' y posteriormente con 22% 'Aumento de compromiso del equipo'.

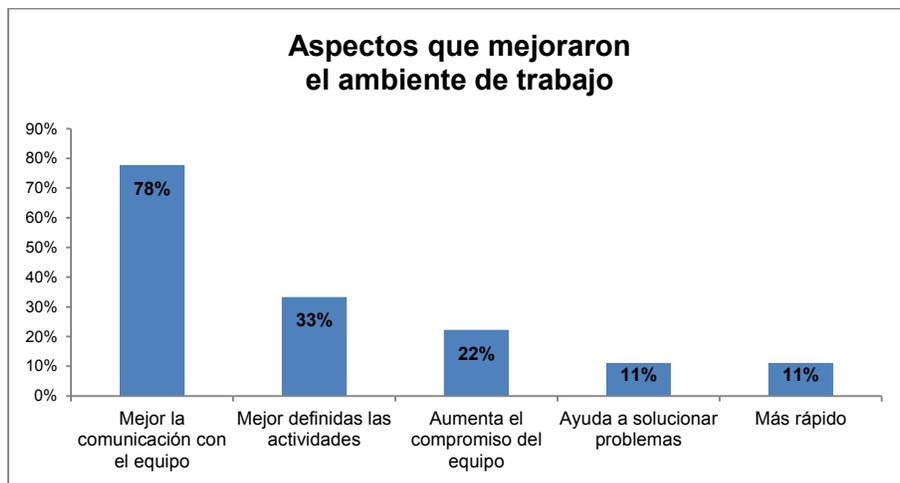


Figura 9-12 Aspectos que han mejorado el ambiente de trabajo en los equipos.

Cuando se realiza la planeación del proyecto está involucrado el equipo de trabajo, lo que mejora la comunicación e intervención para aclarar o detallar las actividades necesarias que se tiene que realizar. La intervención de los integrantes aumenta el compromiso no solamente con el trabajo, sino a buscar soluciones a los problemas y aumentar la velocidad de implementación de los productos.

e) Adaptación del equipo de trabajo

La buena adaptación de los integrantes al equipo de trabajo puede provocar mejoras en la productividad [45].

- ¿Fue difícil la adaptación de los integrantes al desarrollar software utilizando técnicas ágiles con procesos CMMI-DEV?

La Figura 9-12, muestra que ningún encuestado pensó que fue difícil la adaptación a esta forma de trabajo. El 56% piensa que si fue un 'Poco' complicada, mientras que el 44% opina que no tuvo dificultad la adaptación al trabajo con técnicas ágiles.

La adaptación a esta forma de trabajo les costó un poco más de tiempo y esfuerzo a las personas que participaron en la integración, porque reunieron dos formas de trabajo que se consideraban contrarias. Sin embargo, el 44% no le costó trabajo adaptarse a una forma de trabajo establecida y comprobada. La adaptación de técnicas ágiles se da por varios factores. El apoyo entre integrantes de un equipo y la asesoría brindada por la organización.

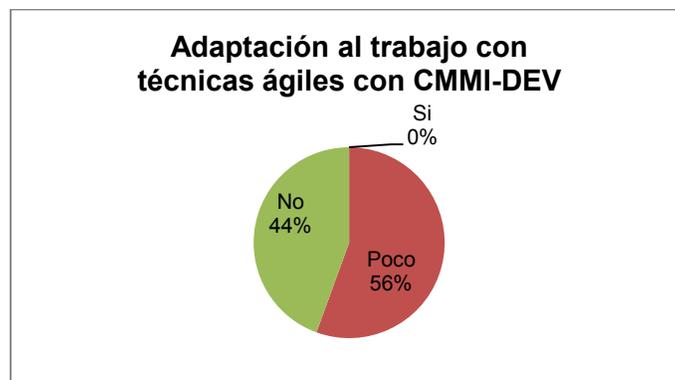


Figura 9-13 Adaptación de los integrantes al trabajo con técnicas ágiles.

En la Figura 9-14 se visualiza que el 89% de los encuestados se apoyan entre compañeros y el 11% piensan que los integrantes del equipo de trabajo se apoyan poco. Ninguno de los encuestados piensa que no hay apoyo entre integrantes de equipo. Lo que muestra que se práctica la colaboración entre compañeros propuesta por Scrum y XP.

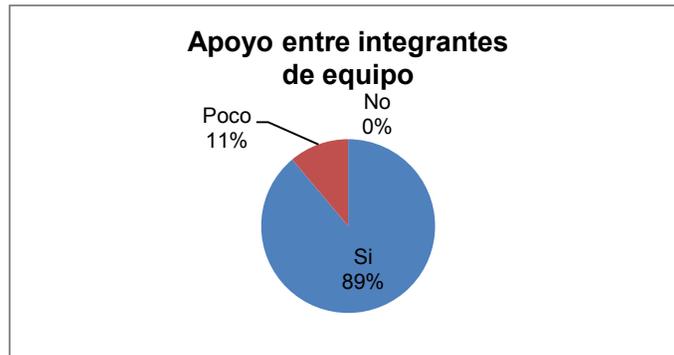


Figura 9-14 Adaptación de integrantes al trabajo con técnicas ágiles.

- ¿Se cuenta con asesorías sobre técnicas ágiles con procesos CMMI-DEV?

En la Figura 9-15. El 78% de los encuestados piensa que se aumentó tanto productividad como ambiente de trabajo, así como el 67% reciben asesorías todo el tiempo o cuando lo requieren.

La asesoría por parte de la organización es otro de los factores que ayuda a los equipos a tener mayor productividad y mejor ambiente de trabajo. Los integrantes de un equipo que están bien preparados y que cuentan con apoyo, pueden aumentar su desempeño individual en el equipo y la productividad de todo el equipo. La Figura 9-15 muestra una relación proporcional entre productividad, ambiente de trabajo y asesoría en la forma de trabajar.

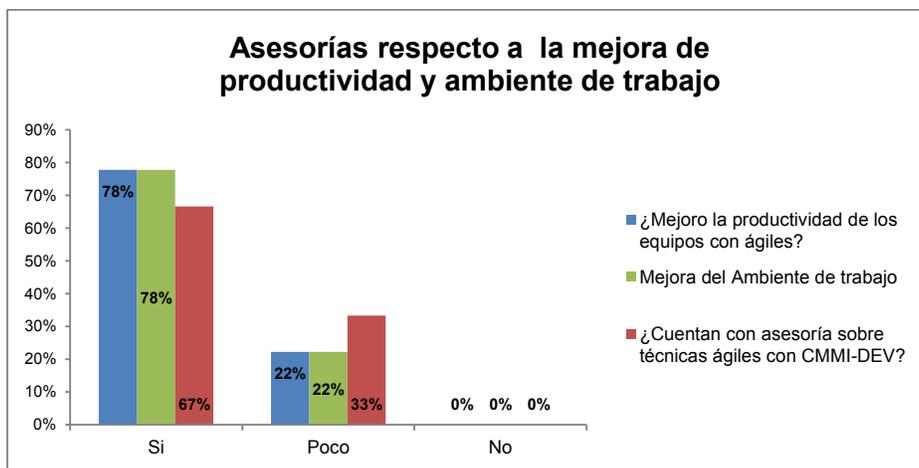


Figura 9-15 Asesoría al equipo de trabajo para mejorar la productividad y ambiente de trabajo.

f) Características de los integrantes del equipo de desarrollo y mejoras individuales.

Esta parte busca información individual de los integrantes de un equipo, desde las características que requieren tener a los beneficios que han logrado al trabajar con técnicas ágiles.

- ¿Cuáles son las características de los integrantes del grupo de trabajo?

Las características que se requiere un individuo para trabajar con técnicas ágiles con mayor porcentaje de frecuencia son: 'Trabajar en equipo' con 78%, 'Proactivos' y 'Buena formación académica' con 67%, seguidos por 'Comprometido' y 'Autodidactas' con 33%, 'Deseos de aprender' y 'Organizado, responsable y Ordenado' con 22%. Mostrado en la Figura 9-16.

Todas estas características son importantes en los individuos que pertenecen a un equipo de trabajo ágil, porque no solamente propiciarán un crecimiento individual por sus características individuales, sino porque el crecimiento será expandido entre los otros miembros del equipo.

Una de las características es tener una buena formación académica, lo que lleva al compromiso por parte de las universidades a fortalecer los planes de estudio para generar egresados con habilidades suficientes para integrarse a la industria con mayor seguridad.

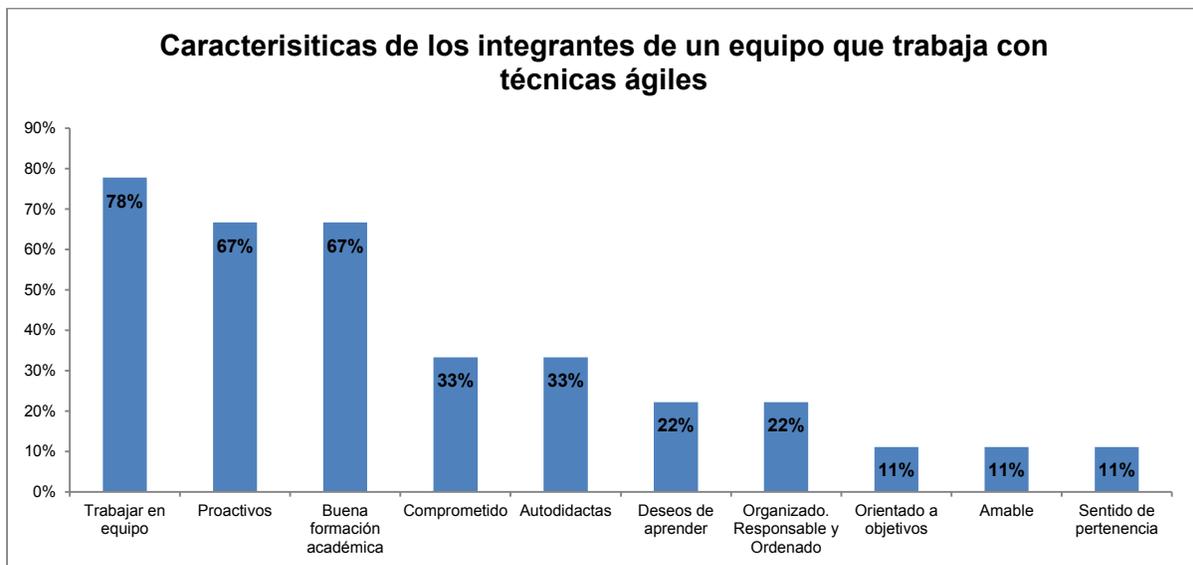


Figura 9-16 Características de un miembro para desarrollar con técnicas ágiles.

- ¿Los integrantes del equipo han mejorado individualmente?

Los aspectos individuales que se midieron con la encuesta fueron a nivel personal, profesional y el desempeño. La Figura 9-17 muestra un crecimiento en los tres aspectos, aunque de diferente magnitud.

Todos los encuestados tuvieron mejoras en aspectos profesionales y en su desempeño, donde el 89% de los encuestados tuvo una mejora notoria a nivel profesional y el 67% tuvo mejoras en el desempeño de su trabajo de manera considerable.

La mejora de aspectos personales fue menos notoria debido a que el 11% considera que su vida personal sigue igual que con otras formas de trabajo y el 22% consideran que no hay una relación entre el trabajo y la vida personal.

Recordando las prácticas de XP, donde establece que una vida personal sana produce una persona motivada que puede aumentar su productividad dentro de su trabajo. No significa que una dependa directamente de la otra.

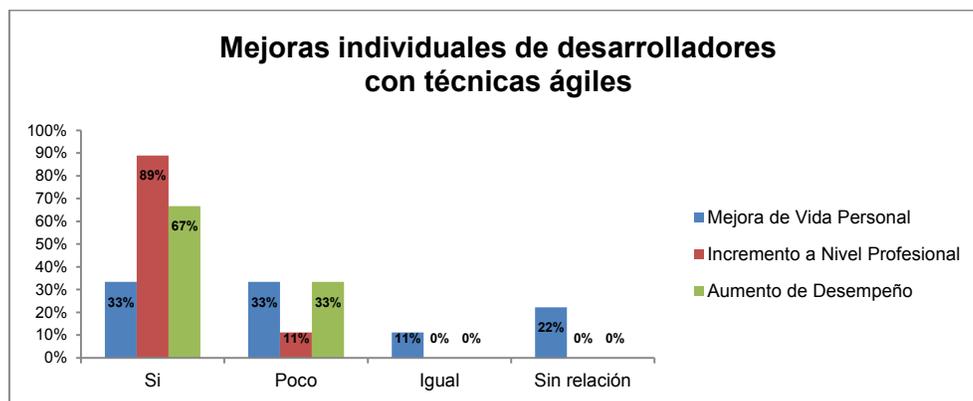
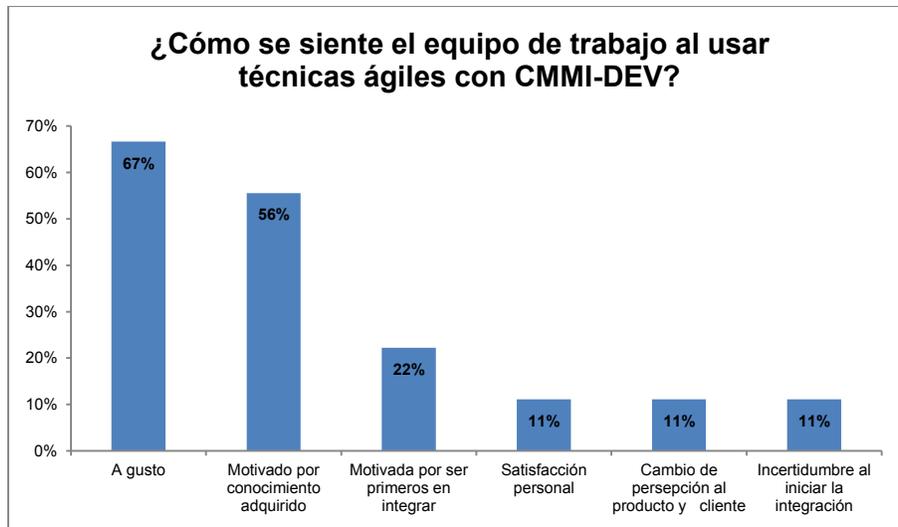


Figura 9-17 Mejoras de los integrantes que trabajan con técnicas ágiles.

- ¿Cómo se sienten los integrantes del equipo de trabajo con esta forma de trabajar?

Los logros que tenga una persona provocan cambios en su sentir y participación en actividades cotidianas [45]. La respuesta que tuvo mayor porcentaje de frecuencia en cuanto el sentir al trabajar con técnicas ágiles fue 'A gusto' con un 67%, seguido por 'Motivado por el conocimiento adquirido' con un 56%, mostrado en la Figura 9-18.



**Figura 9-18 Sentir al trabajar con técnicas ágiles.**

La mayoría de los encuestados se encuentran motivados, independientemente de cual sea el factor que estimula dicha motivación. Y gran parte de los encuestados se encuentra a gusto con su forma de trabajar, sin olvidar la satisfacción personal de algún encuestado. El sentir personal de cada individuo ha ayudado a que las técnicas ágiles dentro de la empresa Px rindan frutos positivos y aceptación.

El 11% de los encuestados considera un cambio al valorar el producto y al mismo cliente, así como una incertidumbre al surgir la propuesta de integración. Ambos aspectos ayudan a valorar el trabajo anterior que se realizaba, pero a vislumbrar una perspectiva favorable con el cambio.

- ¿Cuál es la experiencia de los equipos de trabajo con técnicas ágiles?

La experiencia de los integrantes se puede clasificar respecto a las metodologías trabajadas y el tiempo que tienen trabajando en el desarrollo de software. El 44% de los encuestados tienen de 5 a 10 años de experiencia en el desarrollo de software y el 33% tienen más de 10 años desarrollando software, como se muestra en la Figura 9-19.

La gráfica de la Figura 9-20 tiene la experiencia en la forma de trabajar antes de incorporarse a la empresa Px. Solamente el 22% había trabajado antes con CMMI-DEV, con técnicas ágiles el 44% y con RUP el 89%. Ninguno de los encuestados había desarrollado software utilizando técnicas ágiles con CMMI-DEV y el 22% utilizó RUP con CMMI-DEV.

Los equipos de trabajo cuentan con integrantes que tienen experiencia en el desarrollo de software, pero se está incorporando a personas con menos experiencia para permear esta forma de desarrollar software y poder tener equipos de trabajo con habilidades bastas para sacar los proyectos adelante.

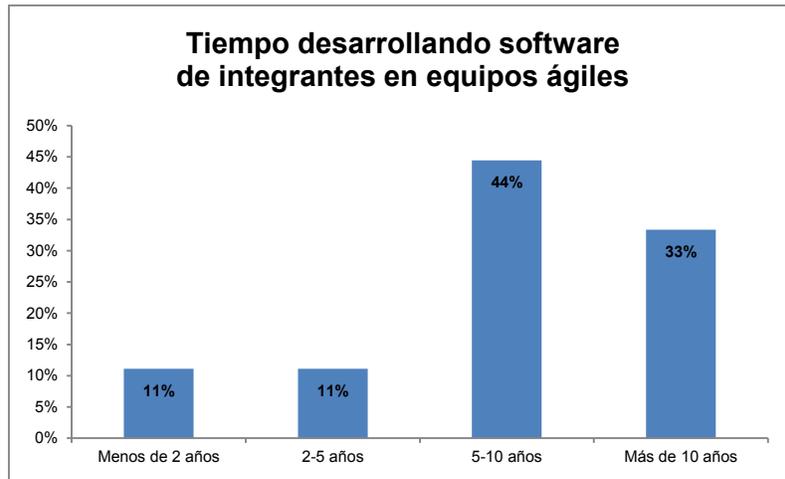


Figura 9-19 Experiencia en desarrollo de software de encuestados.

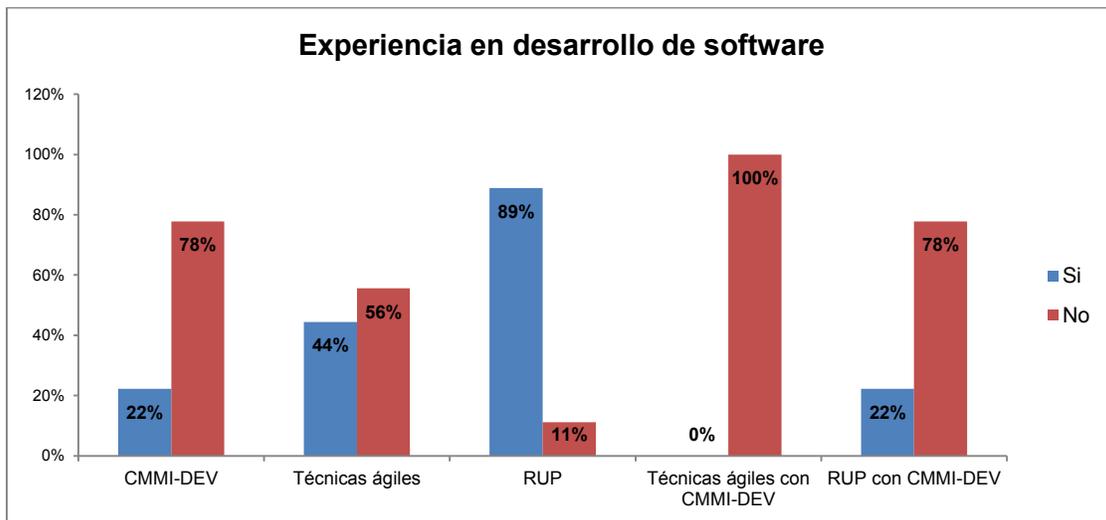


Figura 9-20 Experiencia con diferentes estrategias de desarrollo de encuestados.

### 9.3.1. Comprobación de la hipótesis de la unidad de análisis III

Utilizando el enfoque deductivo de la investigación, llevo a desarrollar las siguientes actividades:

- Explorar los antecedentes de la preparación laboral de algunos participantes en la integración de técnicas ágiles y personal que actualmente desarrolla bajo esta forma de desarrollo.
- Analizar los datos recolectados por medio de los encuestas, utilizando los métodos probabilísticos seleccionados acorde al tipo de información.
- Explicar la información asociada a cada pregunta de investigación en base a las gráficas obtenidas del análisis estadístico para aceptar o rechazar la hipótesis a partir del conocimiento obtenido.

El objetivo específico de la tercera unidad de análisis es: *Identificar por medio de testimonios, los beneficios del uso de técnicas ágiles en la mejora del desempeño laboral de los participantes y la integración de los equipos de trabajo en los proyectos.* La hipótesis relacionada a este objetivo específico se enuncia a continuación.

El grupo de trabajo ha respondido satisfactoriamente, manifestando:

- A. Integrantes motivados al usar métodos de trabajo innovadores.
- B. Aumentó en el desempeño de su trabajo.
- C. Miembros del equipo más comprometidos.
- D. Mayor comunicación y retroalimentación entre los integrantes del equipo de trabajo.

La sentencia A de la primera hipótesis de la unidad de análisis es confirmada por los resultados de la Figura 9-18, al verificar que los equipos de trabajo responden satisfactoriamente al sentirse 'A gusto' con esta forma de trabajo. Así como, el 56% 'Motivados por el conocimiento adquirido' y el 22% 'Motivados por ser los primeros en integrar técnicas ágiles con CMMI-DEV', ambas motivaciones son originadas por el trabajo con métodos innovadores. Mostrando que los integrantes de los equipos de trabajo no solamente están a gusto sino motivados al trabajar con un método de desarrollo de software innovador y que adjunta crecimiento profesional.

El 78% de los encuestados confirma que se ha aumentado la productividad de los equipos de trabajo y se detecta que el 67% de los encuestados ha tenido aumento en su desempeño laboral al trabajar con técnicas ágiles, ver Figura 9-16. Los integrantes han logrado un mejor

desempeño individual, lo que ha provocado un aumento de productividad para todo el equipo. Confirmando la sentencia B de la hipótesis.

El compromiso es considerado por el 33% de los encuestados como una característica que deben tener los integrantes de los equipos ágiles, Figura 9-16. Y el 22% considera que es una característica de la forma de trabajo, Figura 9-4. Por lo que la sentencia C de la hipótesis es confirmada, porque el ser comprometido es una cualidad que se debe tener desde el inicio que se trabaja con técnicas ágiles y va aumentando con la confianza que se deposita en cada uno de ellos al sentirse identificado o parte importante del equipo.

La Figura 9-14 muestra que el 89% de los encuestados recibe o apoya a los integrantes de su equipo de trabajo. La Figura 9-12 indica que el 78% de los encuestados consideran que la comunicación en el equipo ha mejorado el ambiente de trabajo. La Figura 9-4 indica que el 11% de los encuestados considera que reuniones constantes con el equipo es una característica del trabajo con técnicas ágiles. La comunicación no solamente logro un mejor entendimiento del proyecto completo, sino integración en la búsqueda de soluciones, apoyo entre compañeros y aumento de productividad. Con lo que se confirma la sentencia D de la hipótesis

Al confirmar las sentencias A, B, C y D de la hipótesis, se acepta la hipótesis de la tercera unidad de análisis.

Concluyendo la interpretación de los resultados de la investigación del caso de estudio con la aceptación de todas la hipótesis formuladas al inicio de la investigación.

# Capítulo 10

## Beneficios y áreas de oportunidad

Este capítulo documenta los beneficios y las áreas de oportunidad que surgen de la integración de técnicas ágiles con CMMI-DEV para desarrollar software. El origen de este capítulo se encuentra motivado por el objetivo específico: *Identificar y sugerir mejoras a los procesos de trabajo establecidos.*

Los resultados de las 3 unidades de análisis argumentaron los beneficios encontrados y las áreas de oportunidad identificadas.

Para referenciar el origen de la documentación de beneficios y áreas de oportunidad, se indicará la pregunta de dónde se extrajo la información para la unidad de análisis I, y el nombre de la Figura si se toman los resultados de la unidad de análisis II y III.

### 10.1. Beneficios

Los beneficios fueron establecidos como las ganancias obtenidas de la integración de técnicas ágiles con CMMMI-DEV. Desde los integrantes de los equipos de trabajo y el cliente, hasta para la misma organización Px.

### 10.1.1. Enfoque en producto y no en documentación

En la pregunta: ¿Cuál fue la estrategia que siguió la empresa Px para integrar técnicas ágiles con CMMI-DEV Nivel 5? Se informó que la estrategia se enfocará en el producto y no en la documentación para disminuir el tiempo de fabricación y reducir el costo del producto. Pero, conservando la evidencia solicitada por CMMI-DEV. Aprovechando que CMMI-DEV no es simplemente documentación, sino un modelo de control de calidad de productos de software.

La pregunta ¿Cuáles fueron los criterios para seleccionar las técnicas ágiles que se incorporaron al desarrollo de software bajo el modelo CMMI-DEV Nivel? Menciona, que de RUP únicamente se conservaron artefactos básicos cumplir el modelo CMMI-DEV y no había una técnica ágil que lo cubriera.

Una parte de la pregunta ¿Cuáles son las diferencias de trabajar CMMI-DEV Nivel 5 con metodología RUP respecto a las técnicas ágiles? Nos dice que, el cliente ve el producto y no la documentación, por lo que el equipo centra su atención en el desarrollo del producto.

La pregunta ¿Cómo se implementaron las técnicas ágiles con CMMI-DEV Nivel 5? Menciona que se eliminaron actividades de RUP que no generaban valor al proyecto y tomaba tiempo hacerlas. Algunas actividades eliminadas, fueron: documentación para comprobar que se hicieron pruebas unitarias y la descripción de las pruebas que tenían que hacer. Se organizaron los momentos para realizar documentación, se dejó de hacer todos los días.

La respuesta de ¿Cuáles son las diferencias de trabajar CMMI-DEV Nivel 5 con metodología RUP respecto a las técnicas ágiles? Se destaca que el enfoque está en el producto y no en la documentación.

Esta información resalta la trascendencia del desarrollo del producto sobre la importancia de la documentación, pero al estar guiado con un modelo de calidad como CMMI-DEV es necesario contar con la evidencia de que el trabajo fue realizado y que siguió un proceso establecido y ordenado.

### **10.1.2. Enfoque en la recolección de requerimientos**

La respuesta a la pregunta ¿Cómo es el desarrollo de software usando metodología RUP con CMMI-DEV Nivel 5?, indica que en la fase Inception se establece la línea base de requerimientos y productos, en la fase de Elaboración continúan las modificaciones o aclaraciones de los requerimientos.

En la pregunta ¿Cómo se genera el plan de trabajo utilizando técnicas ágiles con CMMI-DEV Nivel 5? Dice, que la recolección de los requerimientos se hace en un Sprint, el cliente ayuda a recabarlos y documentarlos, para establecer el objetivo y alcance del proyecto.

Las técnicas ágiles centran el inicio del proyecto en recabar los requerimientos junto con el cliente en un Sprint, mientras que con RUP se avanza en las fases y aún siguen modificándolos.

### **10.1.3. Selección de las tareas a realizar**

La pregunta ¿Cómo es el desarrollo de software usando metodología RUP con CMMI-DEV Nivel 5? Comentan que en el desarrollo con RUP, el administrador de proyecto asigna arbitrariamente las actividades a los desarrolladores.

Sin embargo, la pregunta ¿Cómo se implementaron las técnicas ágiles con CMMI-DEV Nivel 5? Menciona que cada integrante selecciona las tareas con las que se sienta más cómodo, así como son ellos quienes van administrando diariamente sus tareas en los tableros.

Con técnicas ágiles cada desarrollador elige las actividades a realizar y con RUP son asignadas, por lo que esto ayuda a los ágiles a terminar más rápido, que los integrantes del equipo se sientan motivados y con sentido de pertenencia a la organización.

#### **10.1.4. Descripción de tareas**

En la pregunta ¿Cómo es el desarrollo de software usando metodología RUP con CMMI-DEV Nivel 5? Explica que las funcionalidades de las actividades están descrita con texto narrativo, y queda sujeto a la interpretación de los programadores.

La respuesta de la pregunta ¿Cómo se genera el plan de trabajo utilizando técnicas ágiles con CMMI-DEV Nivel 5? Indica que el equipo de trabajo participa en reunión de planeación del Sprint, ayudan a dimensionar y estimar tanto las tareas como el esfuerzo que se requiere para alcanzar el objetivo del Sprint.

En la pregunta ¿Cómo se implementaron las técnicas ágiles con CMMI-DEV Nivel 5? Se menciona que los desarrolladores dividen los productos en tareas más pequeñas y participan tanto en la planeación del Sprint como en la estimación de las tareas a realizar.

Con técnicas ágiles los integrantes se encuentran involucrados en la planeación lo que mejora el entendimiento de los productos a implementar. Los integrantes del equipo conocen todo el proyecto y pueden apoyarse entre ellos para aclarar detalles de los productos. Mientras que con RUP, están más sujetos a cometer errores por utilizar su interpretación.

#### **10.1.5. Mayor control de actividades**

En la pregunta ¿Cómo se implementaron las técnicas ágiles con CMMI-DEV Nivel 5? Mencionan que para monitorear y controlar el avance del equipo se utilizan las reuniones diarias con el equipo de trabajo, el tablero Kanban y las herramientas de administración de proyectos.

La respuesta de la pregunta ¿Qué beneficios se obtuvieron al integrar técnicas ágiles con CMMI-DEV Nivel 5? Indica que se tiene mayor control del trabajo del equipo Scrum y la alta gerencia conoce el producto, ayuda a supervisar y evitar supuestos acerca del producto. Así como, los integrantes tienen un mayor control de los procesos al cubrir diversos roles.

La respuesta de ¿Cuáles son las diferencias de trabajar CMMI-DEV Nivel 5 con metodología RUP respecto a las técnicas ágiles? Nos dice que el desarrollo ágil permite mayor visibilidad del proyecto y de los riesgos diarios. Aunque, no es tan ordenado como RUP.

La Figura 9-4, indica que el 22% de los encuestados considera que se tiene un mayor control de las actividades con las técnicas ágiles. En la Figura 9-5, muestra que el 56% de los encuestados piensa que hay mayor control en las actividades.

Estos datos muestran un mayor control en monitoreo, control y visibilidad, tanto de las actividades como de los avances y riesgos, por parte de todo el equipo de trabajo. Aumentando las responsabilidades y el compromiso con el trabajo.

### **10.1.6. Corrección oportuna de defectos**

En la pregunta ¿Cómo se implementaron las técnicas ágiles con CMMI-DEV Nivel 5? Mencionan que, durante la tercera semana del Sprint se realizan pruebas y correcciones al producto. Al finalizar el Sprint el cliente puede ver un producto funcional, dando una retroalimentación oportuna a los desarrolladores. Las correcciones no solo se hacen a nivel del producto, sino también con el equipo de trabajo por medio de las reuniones de retrospectiva.

La pregunta ¿Cuáles son las diferencias de trabajar CMMI-DEV Nivel 5 con metodología RUP respecto a las técnicas ágiles? Aborda el punto de vista del cliente, donde indica que el cliente ve el producto real y da retroalimentación para tomar acciones sobre el producto final.

La Figura 9-4, indica que el 33% de los encuestados considera que la identificación oportuna de errores es una diferencia al desarrollar software con técnicas ágiles respecto al hacerlo con RUP.

La retroalimentación constante del cliente es uno de los factores que ayudan a disminuir el número de defectos del proyecto, así como la comunicación entre los miembros de un equipo para aclarar las funcionalidades de los productos o de las tareas a realizar, evitando la interpretación de un solo individuo.

### **10.1.7. Aumento de productividad**

La pregunta ¿Qué beneficios se obtuvieron al integrar técnicas ágiles con CMMI-DEV Nivel 5? Habla del incremento de la velocidad de implementación reduciendo el tiempo estimado para el desarrollo, al reducir la documentación innecesaria provocando menos interrupciones al programar.

En la Figura 9-2, se visualiza aumento en la velocidad de implementación con técnicas ágiles hasta de 5.56cp/hr respecto a la velocidad con RUP de 2.47 cp/hr. Lo que acelera el desarrollo del proyecto, disminuyo el costo del desarrollo y aumentando la productividad del proyecto completo.

La Figura 9-4, muestra que al desarrollar con técnicas ágiles se da prioridad a implementar productos de mayor valor, lo que genera mayor productividad del equipo a los ojos del cliente. El 22% de los encuestados considera que se optimiza el tiempo con las técnicas ágiles, lo que aporta que si hay disminución en los tiempos de desarrollo.

En la Figura 9-8 se observa que el 100% de los encuestados considera que las actividades se realizaron en el tiempo planeado al utilizar técnicas ágiles. Y el 44% piensa que las actividades se realizan más rápido respecto a otras metodologías.

Al observar la Figura 9-11, se identifica que el 78% de los encuestados considera que se ha aumentado la productividad de los equipos de trabajo.

La mejora de la productividad no solo depende de aumentar la velocidad de implementación, disminuir el tiempo de realizar documentación o generar menos defectos, sino de la comunicación entre miembros del equipo de trabajo, con el cliente, generar un ambiente de trabajo agradable, tener integrantes comprometidos, solidarios y motivados. Todos estos aspectos van de la mano para lograr una organización más productiva.

### **10.1.8. Conservar la calidad**

La Figura 9-3 muestra que se tuvo una reducción del 28.98% en los defectos generados al desarrollar el producto, indicando que se generaron productos con menos errores. En la Figura 9-5, el 22% indica que se tienen menos errores en el desarrollo con técnicas ágiles.

En la Figura 9-12, se muestra que el trabajo con técnicas ágiles ayuda a solucionar problemas durante el desarrollo, lo que provoca una disminución de defectos.

La calidad del producto no solamente se conservó, sino que fue mejorada con ágiles al terminar el proyecto con menos defectos que utilizando RUP.

### **10.1.9. Mejora profesional de los integrantes del equipo de trabajo**

La Figura 9-13 muestra que al 44% de los encuestados no les costó trabajo adaptarse al trabajo con ágiles, mientras que al 56% les costó un poco de trabajo la adaptación a esta forma de desarrollar software.

En la Figura 9-15 se observa que el 67% de los encuestados, reciben algún tipo de asesoría para desempeñar mejor su trabajo al tener un porcentaje proporcional en la productividad con 78%, respecto a los que piensan que cuentan con poca asesoría respecto a su trabajo.

Observando la Figura 9-16 se detecta que el 67% de los encuestados consideran deben tener buena formación académica, el 33% indica que deben ser autodidactas y el 22% piensa que deben tener deseos de aprender.

La Figura 9-17 muestra que el 89% piensa que desarrollar software con ágiles incrementó su nivel profesional y el 67% aumento su desempeño en el trabajo.

Cuando los desarrolladores cumplen con todos los roles se vuelven interdisciplinarios. Aunado a las cualidades de ser autodidacta, proactivo y trabajar en equipo aumenta el conocimiento durante cada proyecto.

### **10.1.10. Mejoras personales de los integrantes**

La Figura 9-9 indica que el 22% de los encuestados no trabajan los fines de semana, mientras que el 56% trabaja ocasionalmente y cuando lo hacen es de 1 a 2 días al mes.

En la Figura 9-17, se muestra que el 33% identifica mejoras en su vida personal con el trabajo ágil, el 33% detecta pocos beneficios personales y el 22% consideran que no tiene relación ambos aspectos.

Al observar la Figura 9-18, se detecta que el 67% de los integrantes se siente a gusto con el desarrollo ágil, el 56% se siente motivado por el conocimiento adquirido, en tanto el 22% está motivado por ser de los primeros en integrar técnicas ágiles con CMMI-DEV.

Una de las características del desarrollo con XP son los valores y la mejora personal de los individuos, algunas personas no detectan sus propias mejoras personales. Hay personas que no trabajan los fines de semana y esto es un factor positivo no solo en su vida personal, sino en el mismo trabajo. Afortunadamente, gran parte de las personas que trabajan el fin de semana solamente lo hacen de 1 a 2 días mensuales.

### **10.1.11. Mejoras en el ambiente de trabajo**

En la pregunta ¿Cómo se implementaron las técnicas ágiles con CMMI-DEV Nivel 5? Mencionan que en las reuniones diarias, entre todos los integrantes del equipo resuelven los problemas que tiene cada uno de ellos y puedan avanzar con su trabajo del día.

La Figura 9-11 muestra que el 89% de los encuestados consideran que los integrantes de un equipo de trabajo se apoyan entre sí y 78% considera que se ha mejorado el ambiente de trabajo en los equipos ágiles.

En la Figura 9-16, se indica que una de las características de los integrantes de equipos ágiles es trabajar en equipo con un 76% y el 67% piensa que deben ser proactivos.

En la Figura 9-12 se observa que el 78% piensa que la comunicación mejoró en los equipos y el 22% consideran que aumenta el compromiso del equipo. La Figura 9-14 indica que el 89% de los integrantes de un equipo se apoyan entre sí.

La comunicación es uno de los factores que más han ayudado en el ambiente de trabajo, debido al apoyo que se brindan los integrantes al resolver un problema avanzando más rápido.

### **10.1.12. Re-certificación en Nivel 5 de CMMI-DEV**

En la pregunta ¿Por qué motivo la empresa Px decide integrar técnicas ágiles con CMMI-DEV Nivel 5? Indica que aumentar la participación en licitaciones por ser un requisito cumplir con este nivel de calidad.

La evaluación de la empresa Px con CMMI-DEV, ayudó a mejorar la posición del negocio al lograr obtener el Nivel 5 de CMMI-DEV.

La re-certificación ayuda a las organizaciones a colocarse en el mercado específico de licitaciones, pero da el reconocimiento y confianza de cumplir con niveles de calidad en la industria de software.

### **10.1.13. Satisfacción del cliente**

La pregunta ¿Cómo responde el cliente ante las iteraciones rápidas? Hace buena referencia a la respuesta de los clientes con esta forma de trabajo por los resultados obtenidos, inclusive algunos comentaron que regresarán o solicitarán una segunda fase de aplicaciones. Ha habido felicitaciones a los grupos de trabajo.

En la Figura 9-4 y 9-5, el 11% de los encuestados considera que las entregas de los productos se realizan en menos tiempo con técnicas ágiles respecto a RUP.

Las técnicas ágiles han sido bien recibidas por los clientes, confiando en el trabajo realizado al ver productos funcionales rápidamente.

## **10.2. Áreas de oportunidad**

Las áreas de oportunidad fueron las actividades que no se realizan de manera adecuada o están faltando para asegurar un método de trabajo óptimo en el desarrollo de software.

### **10.2.1. Mejorar el proceso de cambios empleando técnicas ágiles**

En las preguntas de ¿Cómo se realizan los cambios durante un proyecto de metodología RUP con CMMI-DEV Nivel 5? y ¿Cómo se realizan los cambios durante un proyecto con técnicas ágiles con CMMI-DEV Nivel 5? Se responde la forma como se realiza el proceso de cambios con las diferentes formas de trabajo. Con la metodología RUP, mientras el cambio es aceptado se avanza el desarrollo de productos que no son afectados y cuando es aceptado se actualiza la planeación del proyecto. Mientras que al utilizar técnicas ágiles, la única forma de hacer un cambio cuando hay un Sprint en proceso, es deteniendo totalmente el Sprint.

Una posible solución, sería utilizar la misma estrategia de la metodología RUP, es decir, al surgir la solicitud de cambio se avanza con las tareas que no son afectadas y al ser autorizado se actualiza la planeación del proyecto, objetivo y alcance.

### **10.2.2. Desventajas para los integrantes del equipo**

En la pregunta ¿Se detectaron desventajas al integrar técnicas ágiles con CMMI-DEV Nivel 5? Manifiesta que el trabajo con técnicas ágiles puede provocar más presión en el equipo al buscar llegar al objetivo en la fecha próxima. Así como, la dificultad de acoplarse a un nuevo integrante pudiendo afectar el avance del proyecto.

La Figura 9-6 indica que el 78% de los encuestados si detectaron dificultades al trabajar técnicas ágiles con CMMMI-DEV. En la Figura 9-7, se muestra que el 78% consideraron que no es tan sencillo combinar técnicas ágiles con CMMI-DEV, el 33% piensan que fue un poco difícil la adaptación al cambio, mientras que el 11% considera que influye la sinergia del equipo para evitar dificultades al desarrollar software con ágiles. Las cualidades de los integrantes junto con el apoyo por parte de los compañeros del equipo y las asesorías de la organización, pueden ayudar a mitigar los problemas de integración y motivar el desarrollo con técnicas ágiles.

# Capítulo 11

## Conclusiones

El objetivo de la investigación fue documentar la manera como la empresa Px integro dos formas de desarrollar software y logró beneficios, así como detectar áreas de mejora en este método de trabajo.

Los beneficios no solamente fueron a nivel empresarial al conservar el Nivel 5 de la evaluación de CMMI-DEV y mejorar su posición en la industria del software. Sino, al enriquecer el método de trabajo de los equipos de desarrollo, optimizando los recursos para generar productos de calidad en menos tiempo.

El enfoque del trabajo no solamente fue en el producto de software, sino en los individuos que colaboran en el desarrollo del proyecto. Generando un ambiente de trabajo apropiado, que comprometa, responsabilice y motive a cada miembro del equipo o involucrado con el proyecto.

Los objetivos específicos fueron alcanzados al realizar las siguientes acciones:

1. Identificar las técnicas ágiles que se incorporaron con CMMI-DEV y la manera como fueron implementados en proyectos reales.
2. Comparar métricas de productividad y calidad, confrontando los resultados de proyectos ágiles respecto a proyecto que utilizan metodología RUP.

3. Identificar los beneficios de desarrollar con técnicas.
4. Detectar áreas de oportunidad con esta forma de desarrollar software.

CMMI-DEV es un modelo de calidad bastante amplio y con cierto nivel de complejidad, por todas las herramientas de soporte que requiere una empresa. Por lo que, seguir los niveles en estricto orden ayuda a que el nivel de calidad en los procesos de la empresa muestre una madurez real.

Integrar una nueva forma de desarrollar software, requiere de una investigación y planeación previa, para establecer la estrategia de trabajo e implementar en proyectos pilotos para facilitar y pulir su incorporación como proceso institucional. Así como, alcanzar los niveles de maduración de manera paulatina y consecutiva.

El resultado de la investigación puede ser una excelente guía para aquellas empresas que estén interesadas en adquirir un nivel de calidad de CMMI-DEV.

A nivel académico, se muestra que es importante la formación integral de los individuos que se incorporan a trabajar con modelos de calidad. Por lo que es necesario fortalecer los planes de estudio del área de ingeniería de software con materias relacionadas a la calidad, metodologías y métodos de desarrollo, administración de proyectos, técnicas de programación y estructuras de datos. Así como, fomentar la formación de alumnos autodidactas para una mejor incorporación a las nuevas tecnologías y retos que esta área ofrece.

### **11.1. Debilidades de la investigación**

Los proyectos utilizados en la investigación tenían periodos de duración contrastantes, el proyecto ágil fue de 6 meses en contraste al proyecto con RUP que fue de 2 meses. Lo que se contrapone con la estrategia para asignar el uso de alguna metodología respecto al tiempo de duración del proyecto.

La información para documentar el aumento de la productividad fue reducida, debido a que son datos históricos, confidenciales y que no se tienen a la mano. Evitando invadir el tiempo y espacio de las personas involucradas, para no perjudicar el desempeño de su trabajo.

El número de encuestas aplicadas fue pequeño, lo que impidió aplicar otros métodos estadísticos para analizar el comportamiento o influencia de las respuestas de la encuesta.

## **11.2. Trabajo futuro**

Algunas propuestas identificadas durante este caso de estudio para dar continuidad al trabajo de investigación:

- Mapear el modelo de calidad CMMI-DEV y asociar los procesos ágiles a las prácticas necesarias para desarrollar software.
- Profundizar el análisis de los indicadores del desempeño de los integrantes del equipo de trabajo e indagar los aspectos que aumentan su crecimiento profesional.
- Examinar y comparar las estimaciones de las tareas de proyectos ágiles contra los tiempos reales de ejecución.
- Mapear y aplicar técnicas ágiles en otros modelos de calidad enfocados a empresas más pequeñas.
- Investigar actividades que se llevan a cabo para la administración de un proyecto que utiliza métodos ágiles bajo procesos CMMI-DEV Nivel 5.

## Apéndice A

### Evaluaciones CMMI-DEV Ver 1.3 por país

La Tabla muestra los 35 países con al menos una empresas evaluadas con CMMI-DEV Ver 1.3 Nivel 5 en el año 2013 y 2014, ordenadas de mayor a menor número en el 2014.

Posición en Clasificación	País	CMMI-DEV Ver 1.3 Nivel 5	
		2013	2014
1	India	36	90
2	China	31	79
3	EEUU	20	43
4	México	4	13
5	España	4	11
6	Japón	0	11
7	Colombia	5	10
8	Brasil	3	7
9	República de Corea	2	7
10	Argentina	2	6
11	Israel	1	6
12	Tailandia	2	5
16	Filipinas	3	4
14	Portugal	2	4
14	Chile	2	4
14	Canadá	1	4
15	Vietnam	2	3
15	Rusia	2	3
16	Uruguay	2	2
17	Egipto	1	2
17	Australia	1	2
17	Hong Kong	1	2
18	Países Bajos	0	2
18	Arabia	0	2
19	Turquía	1	1
19	Malaysia	1	1
19	Singapur	1	1
19	Ecuador	1	1
19	Romania	1	1
20	Italia	0	1
20	Perú	0	1
20	Reino Unido	0	1
20	Polonia	0	1
20	Sir Lanka	0	1
20	Suiza	0	1

## Apéndice B

# Áreas de Proceso asociadas al Nivel de Madurez

La Tabla clasifica las Áreas de Procesos que se deben cumplir en cada uno de los Niveles de Madurez para lograr la evaluación en dicho nivel.

<b>Nivel</b>	<b>Categoría</b>	<b>Área de Proceso</b>
<b>Nivel 2</b>	<b>Gestión de Proyectos</b>	Monitoreo y Control del Proyecto Planificación del Proyecto Gestión de Requisitos Gestión de Acuerdos con Proveedores
	<b>Soporte</b>	Gestión de Configuración Medición y Análisis Aseguramiento de la Calidad del Proceso y del Producto
<b>Nivel 3</b>	<b>Gestión de procesos</b>	Definición de Procesos de la Organización Enfoque en Procesos de la Organización Formación en la Organización
	<b>Gestión de proyectos</b>	Gestión Integrada del Proyecto Gestión de Riesgos
	<b>Ingeniería</b>	Integración del Producto Desarrollo de Requisitos Solución Técnica Validación Verificación
	<b>Soporte</b>	Análisis de Decisiones y Resolución
<b>Nivel 4</b>	<b>Gestión de procesos</b>	Rendimiento de Procesos de la Organización Gestión del Rendimiento de la Organización
	<b>Gestión de proyectos</b>	Gestión Cuantitativa del Proyecto
<b>Nivel 5</b>	<b>Soporte</b>	Análisis Causal y Resolución

# Apéndice C

## Encuesta sobre ambiente de trabajo

Marca con una X la respuesta que consideres es la que corresponde a tu situación de cada pregunta o proporciona la información que se te solicita.

1.- Sexo Femenino  Masculino

2.- ¿Cuántos años de experiencia tiene trabajando en el área de desarrollo de software?

Más de 2 años  Entre 2-5 años  Entre 5-10 años  Más de 10 años

3.- ¿Cuántos años de experiencia tiene trabajando en Px desarrollando software?

Más de 2 años  Entre 2-5 años  Entre 5-10 años  Más de 10 años

4.- ¿El trabajo utilizando técnicas ágiles con procesos CMMI-DEV ha ayudado a mejorar tu vida personal?

Se trabaja únicamente el horario establecido, disminución de estrés, satisfacción personal, seguridad, bienestar familiar, hace ejercicio, mejorar en vida social.

Si  Poco  Sigue igual que antes  Fue perjudicial  No tiene relación

5.- ¿El trabajo utilizando técnicas ágiles con procesos CMMI-DEV ha ayudado a mejorar tu nivel profesional?

Desempeñas mejor tu trabajo, aumentaste tu conocimiento constantemente, te motiva a prepararte, mejora en la seguridad en el trabajo

Si  Poco  Sigue igual que antes  Fue perjudicial  No tiene relación

6.- ¿Mejoró tu desempeño en el desarrollo de software con las técnicas ágiles??

Aumento la velocidad al generar productos o artefactos, diseña o soluciona problemas con mayor facilidad, trabaja con mayor cantidad de herramientas.

Si  Poco  No  Igual que con otras metodologías

7.- ¿Te costó trabajo adaptarte al uso de técnicas ágiles para desarrollar software?

Si  Poco  No  Aún no me adapto totalmente

8.- ¿El trabajo con técnicas ágiles mejoró el ambiente de trabajo?

Si  Poco  No  Lo dificultó  Igual que con otras metodologías

¿Por qué?

---

9.- ¿Qué diferencias importantes detectó en el trabajo usando técnicas ágiles respecto a la metodología anterior?

---

10.- ¿Los integrantes del equipo de trabajo se apoyan entre sí?

Si  Poco  No  Nunca  Igual que con otras metodologías

11.- ¿Se cuenta con asesoría sobre el trabajo usando técnicas ágiles con procesos CMMI-DEV?  
Todo el tiempo  Nunca  Cuando se necesita  Cuando ingresas al área  En periodos establecidos

12.- ¿Mejoró la productividad de los equipos de trabajo al usar técnicas ágiles?  
Si  Poco  No  Igual que con otras metodologías

13.- ¿Se realizaron más rápido las actividades de desarrollo al usar técnicas ágiles?  
Si  Poco  No

14.- ¿Las actividades se realizan en los tiempos planeados?  
Si  Más rápido  Más lento

15.- Menciona 5 características que deben tener los integrantes de los equipos de trabajo de técnicas ágiles con procesos CMMI-DEV, ordena según la prioridad. Culturales, sociales, formación,...

---

16.- ¿Tiene ventajas usar técnicas ágiles con procesos CMMI-DEV sobre otras metodologías?

Si  Mencione cuales \_\_\_\_\_  
No  ¿Por qué? \_\_\_\_\_

---

17.- ¿Hubo dificultades en la forma de trabajar usando técnicas ágiles con procesos CMMI-DEV sobre otras metodologías?

Si  Mencione cuales \_\_\_\_\_  
No  ¿Por qué? \_\_\_\_\_

---

18.- ¿Cómo te sentiste con esta nueva forma de trabajo?  
Anímicamente, emocionalmente, profesionalmente,...

---

19.- ¿Trabajas los fines de semana?  
Si  No  Ocasionalmente

20.- Si tu respuesta fue Si u Ocasionalmente ¿Cuántas veces al mes?  
1-2  3-4  5-6  7-8

21.- ¿Antes de estar en Px llegaste a trabajar con procesos CMMI?  
Si  No

22.- ¿Antes de estar en Px llegaste a trabajar con técnicas ágiles?  
Si  No

23.- ¿Antes de estar en Px llegaste a usar técnicas ágiles con procesos CMMI?  
Si  No

24.- ¿Antes de estar en Px llegaste a trabajar con RUP?  
Si  No

25.- ¿Antes de estar en Px llegaste a usar RUP con procesos CMMI?  
Si  No

## Apéndice D

# Codificación empleada para clasificar las entrevistas de la unidad de análisis I

Código	Tema
Forma de trabajo en RUP	<ul style="list-style-type: none"> <li>¿Cuál es la estrategia que se sigue en un proyecto de desarrollo de software que utiliza metodología RUP con procesos CMMI-DEV Nivel 5?</li> <li>¿Cómo es la forma de trabajar con metodología RUP para el desarrollo de software con procesos CMMI-DEV Nivel 5?</li> </ul>
Cambios con RUP	<ul style="list-style-type: none"> <li>¿Cómo se realizan los cambios durante el proyecto de desarrollo de software con metodología RUP con procesos CMMI-DEV Nivel 5?</li> </ul>
Motivo para integrar	<ul style="list-style-type: none"> <li>¿Por qué la empresa Px decide integrar técnicas ágiles a procesos CMMI-DEV Nivel 5 para desarrollar software?</li> </ul>
Estrategia de integración	<ul style="list-style-type: none"> <li>¿Cuál fue la estrategia que siguió la empresa Px para integrar técnicas ágiles con procesos CMMI-DEV Nivel 5?</li> </ul>
Criterios para seleccionar técnicas ágiles	<ul style="list-style-type: none"> <li>¿Cuáles fueron los criterios para seleccionar las técnicas ágiles que se incorporaron a los procesos CMMI-DEV Nivel 5 para desarrollar software?</li> </ul>
Forma de trabajo técnicas ágiles y CMMI-DEV	<ul style="list-style-type: none"> <li>¿Cómo se realizó la integración de técnicas ágiles a procesos CMMI-DEV Nivel 5 para desarrollar software?</li> <li>¿Cómo se genera el plan de trabajo para desarrollar software con técnicas ágiles y procesos CMMI-DEV Nivel 5?</li> <li>¿Cómo se integraron las técnicas ágiles con procesos CMMI-DEV Nivel 5 al desarrollo de software?</li> </ul>
Plan de Mejora	<ul style="list-style-type: none"> <li>¿La empresa Px cuenta con Plan de Mejora para el desarrollo de software?</li> </ul>
Cambios en ágiles	<ul style="list-style-type: none"> <li>¿Cómo se gestionan los cambios solicitados por el cliente en un proyecto de desarrollo de software con técnicas ágiles con procesos CMMI-DEV Nivel 5?</li> </ul>
Relación con el resto de la empresa	<ul style="list-style-type: none"> <li>¿Cómo se reestructuro la forma de trabajo de la empresa para apoyar la integración de las técnicas ágiles a los procesos CMMI-DEV Nivel 5 en los proyectos de desarrollo de software?</li> <li>¿De qué manera interactúa los grupos de trabajo de las áreas de proceso en el desarrollo de software con técnicas ágiles con procesos CMMI-DEV Nivel 5?</li> </ul>
Beneficios de la integración	<ul style="list-style-type: none"> <li>¿Qué beneficios se obtuvieron al integrar Metodología RUP para el desarrollo de software con procesos CMMI-DEV Nivel 5?</li> <li>¿Cómo determinó la empresa Px que se tuvieron beneficios en el desarrollo de software al integrar técnicas ágiles a procesos CMMI-DEV Nivel 5?</li> <li>¿Qué beneficios se obtuvieron al integrar técnicas ágiles para desarrollar software a procesos CMMI-DEV Nivel 5?</li> </ul>
Desventajas de integración	<ul style="list-style-type: none"> <li>¿Se detectaron desventajas al integrar técnicas ágiles para desarrollar software a procesos CMMI-DEV Nivel 5?</li> </ul>
Diferencias entre ambas formas de trabajo	<ul style="list-style-type: none"> <li>¿Cuáles son las diferencias de trabajar procesos CMMI-DEV Nivel 5 con Metodología RUP y técnicas ágiles, para desarrollar software?</li> </ul>
El cliente	<ul style="list-style-type: none"> <li>¿Cómo responde el cliente ante las iteraciones rápidas de las técnicas ágiles?</li> </ul>

## Apéndice E

# Control de cambios

La gestión de Control de Cambios se tiene que realizar independientemente de la metodología que se siga para desarrollar el proyecto de software, para controlar el impacto en el negocio y en el desarrollo del proyecto, al no informar y evaluar de manera correcta, adicionando que es una práctica que se tiene que realizar para el modelo de calidad de CMMI-DEV.

El proceso de Control de Cambios necesita seguir actividades concretas:

- a) Documentar el cambio solicitado por medio de un correo electrónico o minuta y compartir esta información con el cliente y dirección.
- b) Determinar el impacto sobre lo que se tiene trabajado hasta el momento.
- c) Estimar el tiempo requerido para aplicar el cambio.
- d) Validar con la Dirección el cambio.
- e) Autorizar y firmar el cambio, tanto por parte de la dirección como del cliente.
- f) Modificar la lista de requerimientos del proyecto completo.
- g) Modificar o realizar nuevamente la planeación.

## Apéndice F

### Plan de mejora

El Plan de Mejora de procesos de desarrollo está a cargo de un área directiva y el Grupo de Procesos. Para integrar nuevos procesos, se debe cumplir el ciclo de mejora siguiente:

1. Investigar los aspectos teóricos
2. Adecuar una implementación para la empresa
3. Plantear hipótesis de los resultados
4. Hacer una primera definición
5. Implementar un piloto
6. Realizar mediciones respecto a la hipótesis inicial
7. Realizar ajustes
8. Corregir dentro del piloto
9. Terminar proceso piloto
10. Medir el resultado bajo un ambiente controlado
11. Se dice si funciona o no y que resultados se tuvieron
12. Determinar los resultados que se tendrían si se implementa a toda la institución

El Grupo de Procesos es responsable de estas mejoras, desplegar una nueva versión del proceso actualizado, capacita, asesora y aclara dudas sobre los procesos y plantillas que se estén integrando. El Grupo de Calidad es quien verifica que se cumpla de manera adecuada con los procesos y validan la calidad del producto.

## BIBLIOGRAFÍA

- [1] N. K. Lim, J. Ang y F. Pavri, «Diffusing software-based innovation with a software factory approach for software development,» *IEEE Xplore*, pp. 549-555, 2000.
- [2] M. B. Chrissis, M. Konrad y S. Shrum, *CMMI for Development: Guidelines for Process Integration and Product Improvement*, Addison-Wesley.
- [3] M. Morales Trujillo, H. Oktaba, F. J. Pino y M. J. Orozco, «Applying Agile and Lean Practices in a Software Development Project into a CMMI Organization,» *Springer*, pp. 17-29, 2011.
- [4] Varios Autores, «Manifiesto por el Desarrollo Ágil de Software,» 2001. [En línea]. Available: <http://agilemanifesto.org/iso/es/>. [Último acceso: 11 11 2013].
- [5] H. Glazer, J. Dalton, D. Anderson, M. Konrad y S. Shrum, *CMMI or Agile: Why Not Embrace Both!*, U.S.A.: Software Engineering Institute, 2008.
- [6] A. Dabat y J. Basave, *Globalización y alternativas incluyentes para el siglo XXI*, México: Porrúa, 2002.
- [7] P. Mochi Alemán, «La Industria del Software en México,» *Problemas del Desarrollo. Revista Latinoamericana de Economía*, vol. 35, nº 137, pp. 41-58, 2004.
- [8] R. Moreira de Castro, J. L. Braga y L. Santos Soares, «Selection of good practices for small software development teams: a knowledge-based approach,» de *ACM SIGSOFT Software Engineering Notes*, New York, E.E.U.U., 2013.
- [9] L. Yuhong, «Software Industry Competitive Strategy and Innovation, A Comparison Among Typical Nations,» de *International Conference of Information Technology, Computer Engineering and Management Sciences*, Nanjing, China, 2011.
- [10] P. E. McMahon, *Integrating CMMI and Agile Development*, Boston, E.E.U.U.: Addison-Wesley, 2011.
- [11] G. P. Gasca-Hurtado, L. M. Giraldo, J. Calvo Manzano y J. A. Echeverri Arias, «Análisis estadístico de la implementación de buenas prácticas en organizaciones desarrolladoras de software,» de *8a Conferencia Iberoamericana de Sistemas de Información y Tecnologías*, Lisboa, Portugal, 2013.
- [12] A. De Vasconcelos, J. De la Vara, J. Sánchez y O. Pastor, «Towards CMMI-compliant Business Process-Driven Requirements Engineering,» de *Conference on the Quality of Information and Communications Technology (QUATIC)*, Lisboa, Portugal, 2012.
- [13] J. M. Navarro y J. Garzás, «Experiencia en la implantación de CMMI-DEV v1.2 en una micropyme con metodologías ágiles y software libre,» *Revista Española de Innovación, Calidad e Ingeniería del*

*Software*, pp. 6-15, 2010.

- [14] N. Ramasubbu y R. K. Balan, «The impact of process choice in high maturity environments: An empirical analysis,» de *IEEE 31st International Conference on Software Engineering, 2009. ICSE 2009*, Vancouver, Canadá, 2009.
- [15] N. Y. León Padilla, Y. N. Gonzales Meneses, J. J. Hernández Mora y M. G. Medina Barrera, «Ranking mundial en certificaciones CMMI-DEV Ver. 1.3 Año 2013,» *Revista Iberoamericana de Producción Académica y Gestión Educativa*, vol. 1, nº 1, 2014.
- [16] C. Institute, «Published Appraisal Result,» 09 09 2014. [En línea]. Available: <https://sas.cmmiinstitute.com/pars/pars.aspx>.
- [17] T. Madi, Z. Dahalin y F. Bahar, «Content analysis on agile values: A perception from software practitioners,» de *5th Malaysian Conference in Software Engineering (MySEC)*, Malasia, 2011.
- [18] M. B. Chrissis, M. Konrad y S. Shrum, *CMMI for Development: Guidelines for Process Integration and Product Improvement*, Addison-Wesley, 2010.
- [19] S. W. Baker, «Formalizing agility: an agile organization's journey toward CMMI accreditation,» de *Proceedings of the Agile Development Conference (ADC'05)*, Colorado, E.E.U.U., 2005.
- [20] SEI, «CMMI-DEV Ver 1.3,» 24 10 2014. [En línea]. Available: <http://cmmiinstitute.com/assets/Spanish%20Technical%20Report%20CMMI%20V%201%203.pdf>.
- [21] I. P. E. Suardiyana Putra, A. Yuliyawati y P. Mursanto, «Industrial Extreme Programming practice's implementation in rational unified process on agile development theme,» de *International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, 2012, Depok, Indonesia, 2012.
- [22] P. Kruchten, *The Rational Unified Process an introduction*, Addison-Wesley, 2000.
- [23] M. Ikonen, E. Pirinen y F. Fagerholm, «On the Impact of Kanban on Software Project Work: An Empirical Case Study Investigation,» de *16th IEEE International Conference on Engineering of Complex Computer Systems (ICECCS)*, 2011, Las Vegas, E.E.U.U., 2011.
- [24] N. Oza, F. Fagerholm y J. Münch, «How does Kanban impact communication and collaboration in software engineering teams?,» de *6th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE)*, 2013, San Francisco, E.E.U.U., 2013.
- [25] K. Schwaber y J. Sutherland, «La Guía de Scrum,» 10 2014. [En línea]. Available: <http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-ES.pdf#zoom=100>. [Último acceso: 04 2014].

- [26] A. C. Sutherland, J. Sutherland y C. Hegarty, «Scrum in Church: Saving the World One Team at a Time,» de *Agile Conference, 2009*, Chicago, E.E.U.U., 2009.
- [27] J. Sutherland, C. Ruseng Jakobsen y K. Johnson, «Scrum and CMMI Level 5: The Magic Potion for Code Warriors,» de *Agile Conference, 2007*, Washington, E.E.U.U., 2007.
- [28] H. Guang-Yong, «Study and practice of import Scrum agile software development,» de *3rd International Conference on Communication Software and Networks*, Xi'an, China, 2011.
- [29] E. Hossain, M. Ali Babar y H.-y. Paik, «Using Scrum in Global Software Development: A Systematic Literature Review,» de *Fourth IEEE International Conference on Global Software Engineering, 2009*, Limerick, Irlanda, 2009.
- [30] K. Beck y C. Andres, *Extreme Programming Explained*, Boston, E.E.U.U.: Addison-Wesley, 2004.
- [31] W. Xiaohua, W. Zhi y Z. Ming, «The Relationship between Developers and Customers in Agile Methodology,» de *International Conference on Computer Science and Information Technology, 2008.*, Singapur, República de Singapur, 2008.
- [32] B. Xu, «Towards High Quality Software Development with Extreme Programming Methodology: Practices from Real Software Projects,» de *International Conference on Management and Service Science, 2009*, Wuhan, China, 2009.
- [33] D. J. Anderson, *Kanban. Successful Evolutionary Change for Technology Organizations*, E.E.U.U.: Blue Hole Press, 2010.
- [34] R. Turner, R. Madachy y D. Ingold, «Improving systems engineering effectiveness in rapid response development environments,» de *International Conference on Software and System Process (ICSSP), 2012*, Zurich, Suiza, 2012.
- [35] P. Runeson, M. Höst, A. Rainer y B. Regnell, *Case Study Research in Software Engineering. Guidelines and Examples*, New Jersey U.S.A / Canada: Wiley, 2012.
- [36] O. Villareal Larrinaga y R. J. Landeta, «El estudio de casos como metodología de investigación científica en dirección y economía de la empresa.,» *Investigaciones europeasa*, vol. 16, nº 3, pp. 31-52, 2010.
- [37] R. K. Yin, *Case Study Research: Design and Methods*. Design and Methods, SAGE Publications, 2003.
- [38] C. Robson, *Real world research*, Blackwell, 2002.
- [39] H. K. Klein y M. D. Myers, «A set of principles for conducting an evaluating interpretive field studies in information systems,» *MIS Quarterly*, vol. 23, nº 1, pp. 67-93, 1999.

- [40] M. Okuda Benavides y C. Gómez Restrepo, «Métodos de investigación cualitativa: triangulación,» *Revista Colombiana de Psiquiatría*, vol. XXXIV, nº 1, pp. 118-124, 2005.
- [41] S. Forrest , J. Singer y D. I. K. Sjoberg, *Guide to Advanced Empirical Software Engineering*, Londres: Springer, 2008.
- [42] C. Torrecillas Salinas, M. J. Escalona y M. Mejías, «A scrum-based approach to CMMI maturity level 2 in web development environments,» de *IIWAS '12 Proceedings of the 14th International Conference on Information Integration and Web-based Applications & Services*, New York, E.E.U.U., 2012.
- [43] A. Coffey y P. Atkinson, *Encontrar sentido a los datos cualitativos. Estrategias complementarias de investigación.*, Medellín, Colombia: Editorial Universidad Antioquia, 2003.
- [44] G. Canavos, *Probabilidad y estadística. Aplicaciones y métodos.*, México: Mc Graw-Hill, 1998.
- [45] K. Beck y C. Andres, *Extreme Programming Explained: Embrace Change*, Segunda Edición ed., U.S.A.: Addison Wesley Profesional, 2004, p. 224.
- [46] E.-D. Ahmed, «A view on the state of software engineering practice: experience from Egypt,» de *ACM SIGSOFT Software Ingeniería Notas*, New York, E.E.U.U., 2012.
- [47] Z. Bosheng, «The Status and Prospect of Process Improvements in China,» de *International Conference on Software and System Process*, New York, E.E.U.U., 2014.
- [48] M. Jenkins, «PRO-SOFTWARE: a government-industry-academia partnership that worked,» de *17th Conference on Software Engineering Education and Training, 2004.*, 2004.
- [49] K. Silva y C. Doss, «The Growth of an Agile Coach Community at a Fortune 200 Company,» de *Agile Conference (AGILE), 2007*, Washington, E.E.U.U, 2007.
- [50] C. Ruseng Jakobsen y K. A. Johnson, «Mature Agile with a twist of CMMI,» de *Conference Agile, 2008*, Toronto, Canadá, 2008.
- [51] P. Monteiro, R. J. Machado, R. Kazman, A. Lima, C. Simões y P. Ribeiro, «Mapping CMMI y RUP marcos de proceso para el contexto de las propuestas Elaboración de Proyectos de Software,» de *5ª Conferencia Internacional, SWQD 2013*, Viena, Austria, 2013.
- [52] R. Chaipraserth, A. Leelasantitham y S. Kiattisin, «A test automation framework in POCT system using TDD techniques,» de *13th International Symposium on Communications and Information Technologies (ISCIT)*, Surat Thani, 2013.
- [53] C. Zhuoyan, «The research of software development supporting environment based on UML,» de *Electronic and Mechanical Engineering and Information Technology (EMEIT), 2011 International*

*Conference on*, Harbin, China, 2011.

- [54] K. Beck, M. Beedle, A. Van Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, R. C. Martin, S. Mellor, K. Schwaber, J. Sutherland y D. Thomas, «Manifiesto por el Desarrollo Ágil de Software,» 2001. [En línea]. Available: <http://agilemanifesto.org/iso/es/>. [Último acceso: 11 11 2013].
- [55] J. Sutherland, N. Harrison y J. Riddle, «Teams That Finish Early Accelerate Faster: A Pattern Language for High Performing Scrum Teams,» de *47th Hawaii International Conference on System Sciences (HICSS), 2014*, Waikoloa, Hawaii, 2014.
- [56] F. Macias, M. Holcombe y M. Gheorghe, «A formal experiment comparing extreme programming with traditional software construction,» de *International Conference on Computer Science, 2003. Proceedings of the Fourth Mexican*, 2003.
- [57] K. Beck, *Test-Driven Development. By Example*, Addison-Wesley, 2002.
- [58] C. Blé Jurado, J. Gutiérrez Plaza, F. Reyes Perdomo y G. Mena, *Diseño ágil con TDD*, Creative Commons, 2010.
- [59] W. Shelton, N. Li y P. Ammann, «Adding Criteria-Based Tests to Test Driven Development,» de *Fifth International Conference on Software Testing, Verification and Validation (ICST), 2012 IEEE*, Montreal, C nada, 2012.
- [60] I. Benbasat, D. K. Goldstein y M. Mead, *The case research strategy in studies of information*, MIS Quaterly, 1987.
- [61] E. Yacuzzi, «El estudio de caso como metodolog a de investigaci n: Teor a, mecanismos causales, validaci n.».
- [62] F. Azam, H. Gull y S. Bibi, «Back and Forth (BnF) Software Process Model,» de *Computer Engineering and Applications (ICCEA), 2010 Second International Conference o*, La isla de Bali, Indonesia, 2010.
- [63] R. A. Espa ola, «Diccionario de la lengua espa ola,» [En l nea]. Available: <http://lema.rae.es/>. [ ltimo acceso: 28 10 2014].
- [64] S. N. Hussai, N. Hundewale, S. Aljahdali, Ashfaq Ahmed and K. Ashfaq Ahmed, "A Methodology for the Abstraction of Design Components from the Software Requirement Specification to the Object Oriented System," *IEEE*, pp. 311-315, 2012.
- [65] S. E. Institute, «Software Engineering Institute,» 24 10 2014. [En l nea]. Available: <http://cmmiinstitute.com/assets/Spanish%20Technical%20Report%20CMMI%20V%201%203.pdf>.

- [66] C. E. Paes de Barros y C. Massaki Hirata, «RUP Extension For the Software Performance,» de *Computer Software and Applications, 2008. COMPSAC '08. 32nd Annual IEEE International*, Turku, Finlandia, 2008.
- [67] Y. Saito, A. Monden y K. Matsumoto, «Evaluation of Non Functional Requirements in a Request For Proposal (RFP),» *IEEE*, pp. 106-111, 2012.
- [68] I. Sommerville, *Ingeniería del Software*, Madrid, España: Pearson Addison Wesley, 2005.
- [69] W. Zhang, G. Zhang y G. Qian , «Anderson-Darling Test based CFAR Detection,» de *1st International Conference on Information Science and Engineering (ICISE), 2009*, Nanjing, China, 2009.
- [70] PMI, «Institue, Project Management,» [En línea]. [Último acceso: 2015].
- [71] C. J. Woodward, A. Cain, S. Pace, A. Jones y Funke, «Helping Students Track Learning Progress Using Burn Down Charts,» *IEEE*, pp. 104-109, 2013.
- [72] B. Fitzgerald, K.-J. Stol, R. O'Sullivan y D. O. Briend, «Scaling agile methods to regulated environments: An industry case study,» *IEEE*, pp. 863-872, 2013.

## **DESARROLLO SOFTWARE CON MÉTODOS TRADICIONALES RUP Y MODELO CMMI-DEV**

La empresa Px empleaba la metodología tradicional RUP para el desarrollo de software durante todo el ciclo de vida del proyecto y CMMI-DEV como modelo para darle soporte a la forma de trabajar.

El desarrollo de los proyectos considera las 4 fases de trabajo de RUP: Inception, Elaboration, Construction y Transition, así como todas las disciplinas transversales para cada una de las fases de la metodología. De CMMI-DEV utiliza todas las áreas de proceso que fueran necesarias de la versión vigente.

Se realiza toda la documentación requerida para evitar cualquier problema y tener la evidencia completa. Si alguien repentinamente sale del equipo, la persona que se integre tenga todas las herramientas y pueda comenzar desde el punto donde se quedó el integrante que salió.

### **ACTIVIDADES REALIZADAS EN LAS FASES RUP**

#### **Inception**

En esta fase se hace la definición del proyecto en el Project Charter, para la disciplina de requerimientos se genera la línea base de requerimientos y línea base de productos para poder cotizar el costo del proyecto. En la parte de pruebas e implementación no hay gran avance, se define la plataforma, el servidor y el lenguaje a utilizar. Toda esta información se almacena en un conjunto de carpetas que son los entregables generados en esta etapa.

#### **Elaboration**

El objetivo de la fase de Elaboration es generar la mayor parte del Diseño y Arquitectura del sistema con su documentación, se refina la línea base de requerimientos y productos por medio de entrevistas con el cliente clarificando todos los requerimientos. Para los entregables se genera el Software Architecture Document, el modelo Entidad-Relación y casos de uso, para cada elemento se genera entregable y se suben al repositorio.

Con la creación del repositorio y la Administración de la Configuración se liga a CMMI y sus prácticas, ya que indican que se debe tener un versionado de toda la evidencia que se va haciendo en el proyecto. Se crean carpetas por fase y dentro de cada fase se encuentran todas las disciplinas con las carpetas de productos que se generan en cada disciplina.

## Construction

En la fase de Construction es donde realmente viene la implementación del proyecto, es parecido al Modelo en Cascada, pero RUP es más flexible porque no te dedicas únicamente a Análisis y Diseño, sino que puedes empezar a definir casos de prueba inclusive a nivel de código, puedes hacer el cascaron de la arquitectura o generar una serie de tareas que empieza a involucrar a los diversos responsables del proyecto. En este momento se pueden presentar modificaciones o aclarar requerimiento que no se entendieron, permitiendo realizar algún cambio. El Administrador de Proyecto asignaba las tareas a los desarrolladores aunque no fuera su fuerte o no le agrada la actividad asignada.

## Transition

La etapa final es la de Transition, que tiene que ver con la parte de despliegue o transición del producto a los usuarios finales del producto, lo que puede dar origen a mejoras del producto.

## PROCESO DE TRABAJO

Inicialmente los programadores trabajaban en base a la línea de productos y requerimientos que especificaban la funcionalidad en textos tipo narrativa, sin embargo, estaban sujetos a interpretación, ya que el programador lo entendía como venía escrito y así lo hacía.

Los requerimientos se clasifican en tres tipos de productos: pantallas, procesos y reportes, cualquier funcionalidad debería caer en uno de esas tres clasificaciones.

- Pantallas si tenías que hacer la creación o actualización de una pantalla.
- Un proceso es cualquier algoritmo que estuviera ligado a esa pantalla.
- Reporte cualquier reporte o gráfica que se tuviera que generar.

A su vez, los productos tienen un nivel de complejidad que puede ser Simple, Mediana y Compleja, considerando estos dos criterios se obtiene de la Matriz de Tiempos el número de horas asignadas para desarrollar el producto.

Cada producto incluye el tiempo total que toma realizar ese producto incluyendo el trabajo en algunas de las áreas de procesos de CMMI, entonces de 12 horas asignadas a un producto, donde 1 era de administración de proyecto, 2 de pruebas, 3 de ingeniería de requerimientos, 2 de análisis y diseño y 2 de implementación, por lo que para la programación el tiempo era muy reducido.

## CAMBIOS

Cuando surge un cambio por parte del cliente se tiene que gestionar un Control de Cambios, mientras tanto el proyecto con RUP sigue avanzando y en paralelo se gestiona el control de cambios, obviamente se continúa trabajando sobre aquello

que no impacta el cambio para no re TRABAJAR, en cuanto está autorizado el cambio la replaneación debe ser prácticamente inmediata para poder avanzar.

## **MOTIVACIÓN PARA INTEGRAR TÉCNICAS AGILES A CMMI-DEV**

Del lado del negocio se buscaba disminuir los costos para poder bajar el precio final al cliente y mantener la calidad actual, para posicionarse mejor en el mercado.

Surge la necesidad de implementar un proceso que permita enfocarse en el producto y no tanto en la documentación; pero cuidando generar la evidencia de las prácticas que se ejecutan para cumplir con el modelo de CMMI.

Lograr un ahorro en los gastos al quitar procesos innecesarios hasta cierto punto, disminuir los costos.

Para obtener la recertificación de CMMI-DEV Nivel 5 los métodos ágiles eran una aportación atractiva para lograr innovación en los procesos de la empresa.

## **ESTRATEGIA PARA IMPLEMENTAR TÉCNICAS AGILES CON CMMI**

La organización definió una regla para determinar la metodología de trabajo en base al tamaño del proyecto, para proyectos de menos de 6 meses se utilizarán métodos ágiles y los mayores a 6 meses se trabajan con RUP; en la parte de Ingeniería se da la opción a decidir con cuál de las dos metodologías se trabajará.

CMMI-DEV no es solamente documentación, sino que es un modelo de control que te permite hacer mejoras en los procesos de una organización; así como los métodos ágiles son simplemente una forma de trabajar. Trabajando estos dos aspectos metodológicos se ha logrado hacer un compendio de técnicas ágiles que

se pueden aprovechar y adecuar a las prácticas del modelo inicial realizando ajustes conforme el negocio va requiriendo.

Con estas consideraciones, se adoptaron técnicas ágiles para la administración, planeación, desarrollo, monitoreo, control e ingeniería y se complementó con artefactos de RUP para cumplir con el mapa trabajo de CMMI-DEV.

Para formalizar este método de trabajo como parte de la empresa se realizaron varios proyectos pilotos para pulir los procesos y poder documentarlos. CMMI no te dice 'Como hacerlo' sino te dice 'Que cosas no se deben olvidar hacer'.

## **PRÁCTICAS SELECCIONADAS Y CRITERIOS DE SELECCION**

La empresa hace un mapeo para cubrir todas las áreas de proceso que pide el modelo de CMMI-DEV, se fue viendo cómo cubrir con ágiles esa mapa y nos dimos cuenta que agiles puro no cubre el modelo completo, por lo que agregamos los artefactos que ya teníamos de RUP, adaptado totalmente el modelo a la organización.

Se tomaron las técnicas ágiles necesarias para cubrir el ciclo de vida completo para el desarrollo de software, dando la alternativa al grupo de trabajo a seleccionar, remplazar o eliminar aquellas les ayudaran a dar buenos resultados, logrando mejorar el proceso y hacer CMMI-DEV más ágil.

Se adoptó Scrum para la administración, desarrollo de planeaciones, retrospectivas, monitoreo y control de las actividades; para ingeniería se tomaron las prácticas de integración continua, programación entre pares y trabajo en el mismo lugar de Extreme Programming; de RUP se conservaron los artefactos de la arquitectura, planes y estrategia de prueba porque son básicos para la ejecutar el proceso de manera adecuada, logrando cubrir las prácticas de CMMI-DEV. De RUP se preservaron artefactos no metodología.

En un Sprint únicamente se trabaja con los requerimientos establecidos en el Sprint Backlog, se hace una vuelta completa a todas las disciplinas de RUP, trabajando con los requerimientos e historias de usuario, diseño, diagramas de clases básicos, diagramas de secuencia, matriz de pruebas, actualizar los datos de prueba, manual de usuario, implementación y ejecución de pruebas. Para el resto de los requerimientos se repite el proceso con el número de Sprints que sean necesarios para completar el proyecto, entonces, se ve como un proceso que mezcla todas las disciplinas y cada quién va ejecutando las actividades que le corresponden.

CMMI tiene que cubrir una serie de prácticas, se va armando el proceso con las técnicas ágiles apropiadas y se complementa con los artefactos de RUP, agregándolo al modelo. Por ejemplo:

- Para diseñar la arquitectura no hay una técnica ágiles para que cubra lo pide el modelo de CMMI, por lo que se toma esta parte de RUP.
- Se requiere monitorear el avance del proyecto paulatinamente, entonces se toma de Scrum las juntas diarias, lecciones aprendidas y retrospectivas.
- Necesitan hacer guías de programación, por lo que tomamos el artefacto de RUP.

## **PLAN DE MEJORA DE PROCESOS**

La empresa Px cuenta con un Plan de Mejora que tiene como objetivo aumentar las habilidades de los miembros de la fábrica de software, mantener el nivel de calidad de los procesos y la mejora continua para conservar el CMMI-DEV Nivel 5, a cargo de la “Dirección de Innovación y Calidad” y del Grupo de Procesos.

Todas las mejoras de la organización se piloteen en el CEDS, que es el Centro de Excelencia de Desarrollo de Software, con la finalidad de hacer los ajustes necesarios y después desplegar al resto de la organización.

La manera de integrar nuevas metodologías a la forma de trabajo actual, es bajo un ciclo de mejora que consta de siguientes pasos:

1. Investigar la parte teórica
2. Adecuar una implementación para la empresa
3. Plantear hipótesis de los resultados
4. Hacer una primera definición
5. Implementar un piloto
6. Realizar mediciones respecto a la hipótesis inicial
7. Realizar ajustes
8. Corregir dentro del piloto
9. Terminar proceso piloto
10. Medir el resultado bajo un ambiente controlado
11. Se dice si funciona o no y que resultados tuvo
12. Determinar los resultados que se tendrían si se implementa a toda la institución

Este es un proyecto de implementación de mejora permanente, por lo que se cuenta con un plan de mejora direccionado a los recursos del SEDS quienes llevan varios años operando con el proceso de CMMI-DEV Nivel 5 y proporcionan retroalimentación sobre los procesos, plantillas y artefactos que manejan, se concentran todas las mejoras en un archivo y después esas mejoras se evalúan.

Algunas aportaciones llegan cuando el Grupo de Procesos envía solicitudes de mejora o en las reuniones es común que las personas expresen si algo está mal o falta.

Otras sugerencias llegan de las juntas de lecciones aprendidas, al responder las preguntas “¿Que hicimos bien?”, “¿Qué debemos de seguir haciendo?”, “Qué podemos mejorar?” e incluso hay aportaciones de la misma Dirección General.

No todas las propuestas aplican, sino que se evalúan para saber cuáles son de mayor impacto monetario o afectación al proceso y Dirección General son los únicos que decide si aplica o no dicho cambio. Lo único que nosotros tenemos prohibido es no seguir el proceso y ningún proceso aplica sino pasa por el ciclo de mejora. La mejora inmediata no pasar por ciclo de mejora, siempre y cuando sea un cambio en una plantilla mal traducida o una sección que no sirve.

El Grupo de Procesos trabaja en esas mejoras, despliega una nueva versión del proceso ya actualizado, finalmente da capacitación, asesora y aclara dudas sobre los procesos y plantillas que se estén integrando. El Grupo de Calidad es quien verifica que se cumpla de manera adecuada con los procesos y validan la calidad del producto, con todo ello se logra la mejora continua.

El proceso siempre tiene mejoras, por lo que siempre se promueve a nivel organización la innovación, logrando una mejora continua en el proceso para mantener el CMMI-DEV Nivel 5.

## **INTEGRACIÓN DE TÉCNICAS ÁGILES A PROCESOS CMMI-DEV**

### **DESARROLLO DE SOFTWARE CON TÉCNICAS AGILES Y CMMI**

Para empezar a trabajar con el proyecto piloto, se tuvo asesoría y junto con el Grupo de Procesos de la empresa, se implementaron varias actividades útiles respetando las áreas de proceso y sin dejar de cumplir con CMMI-DEV.

La planeación y el marco de administración se realizaron con Scrum porque es una forma de dividir el trabajo de manera muy simple, identificando prioridades técnicas y del negocio; con base en estas prioridades se establecen los requerimientos a desarrollar durante el Sprint definiendo su objetivo. La metodología de Scrum sufrió algunos cambios para ajustarse a la forma de trabajo de la empresa Px.

La recolección de los requerimientos se hace en un Sprint, donde el cliente está involucrado ayudando a recabar la información y documentando al momento. Se dimensiona el problema, analizando hasta donde se quiere llegar y la inversión que se tiene para realizar la estimación alcance. En este Sprint se genera el Product Backlog y el objetivo del proyecto, alrededor de ello todas las áreas trabajaran para alcánzalos.

El Scrum Master del proyecto piloto venía de un esquema tradicional como Administrador de Proyecto y el Equipo Scrum tenía noción del trabajo con técnicas ágiles y procesos. Se asignó un espacio de trabajo y se comenzó el primer Spring.

Inicialmente no se eliminaron procesos pero se observó que había cosas que no generaban valor y fue cómo empezó a evolucionar el proceso de desarrollo. Considerando siempre que para eliminar algo se tenía que tener una justificación.

En cada Sprint se siguen una serie de pasos para analizar cuáles son los requerimientos que se van a trabajar, este ciclo se repite al iniciar cada Sprint hasta lograr terminar el producto completo.

- 1) Ver que productos del Product Backlog, se van a meter en el Sprint tomando aquellos productos que generan valor al permitir tener un entregable, demostrable y con funcionalidad.
- 2) Descomponer los productos en tareas más pequeñas antes de ver la complejidad de cada producto.

- 3) Estimar el tiempo para realizar cada tarea por medio de la estrategia de póker de planeación, donde cada punto equivale a una hora de trabajo para la tarea; de esa manera se estiman todos los productos.
- 4) El equipo Scrum no se casa con los tiempos de la Matriz de Tiempo del producto y usa la mayor cantidad de horas en la implementación que en otras áreas, porque los métodos ágiles se enfocan más en los productos y las personas que en los procesos
- 5) Seleccionar los productos, descomponer en tareas, ver cuantos productos se asignan a ese Sprint y sobre eso se trabaja, si hay un producto que no da tiempo de meterlo en ese Sprint se retoma para el siguiente.
- 6) Si el tiempo que tenía asociado a ese conjunto de productos para la implementación no alcanza, se reduce el tiempo asignado a otros procesos y se evalúa, logrando tener ajustado el Sprint.
- 7) El tiempo de duración de un Sprint es generalmente de un mes, donde 3 semanas son para desarrollo y 1 para pruebas como correcciones.

Durante los Sprints se lleva a cabo algunas actividades con técnicas ágiles:

- a) Todo el equipo trabaja en una sala para tener mejor comunicación y aclarar dudas del proyecto. Antes los compañeros de un mismo proyecto no trabajaban en lugares adyacentes.
- b) La planeación se hace con el equipo de trabajo, utilizando el tablero de Kanban donde se definen los carriles en “En espera”, “En proceso”, “En pruebas” y “Terminado”, los carriles los define el equipo de trabajo y ahí se monitorean las actividades.
- c) Durante la reunión de planeación se determinan los productos que se van a hacer, realizan la estimación y seleccionan los productos a realizar en un mes.
- d) En el tablero se pegan todas las tareas de cada uno de los productos, para tener trazabilidad.
- e) Cada tarea tiene un postick dentro del tablero.

- f) El postick de cada tarea tiene un identificador del producto, con lo que se puede ligar una tarea a un producto y el producto al requerimiento, esta fue una aportación del grupo piloto Scrum de Px.
- g) Cada postick incluye el identificador del producto, nombre de la tarea, número de horas estimadas para realizar la tarea y conforme pasa el tiempo se agregan las horas que se tarda en realizarla, teniendo finalmente el número de horas reales invertidas.
- h) Todos los días se tiene la reunión diaria, donde se ven 3 cosas: “¿Qué es lo que hicimos ayer?”, “¿Qué vamos a hacer hoy?” y “¿Tenemos algún problema? Estos problemas se resuelven para que cada quien avance con sus tareas.
- i) En Kanban se tiene la capacidad por día, donde cada quien tiene que cumplir con 8 horas de trabajo, se tienen que tomar posticks que sumados dieran 8 horas, cada quien elige los posticks con los que se siente más cómodo y con lo que puede trabajar.
- j) Al seleccionar cada quién sus actividades se busca la no dependencia de actividades entre compañeros, para que el objetivo del Sprint no dependa del avance de una persona, aquí es muy útil la descomposición detallada de las tareas porque el equipo sabe que tiene que hacer.
- k) Cada quien toma sus tareas, las pone en proceso y trabaja con ellas durante el día. Al final del día se actualiza el tablero y se le toman fotos para guardar la evidencia diaria de lo que se hace.
- l) El Scrum Master lleva ese mismo tablero pero en formato digital, para ir documentando la evidencia y cumplir con lo solicitado por CMMI-DEV,
- m) El Scrum Master, también lleva la gráfica de remanentes Born Down Chart, que indica el total de horas estimadas y devengadas.
- n) En el tablero hay una sección de Riesgos, diariamente se ve si se tiene algún riesgo, de ser así se registra en un postick y se pega en el tablero. El Scrum Master checa este aspecto todos los días y actualiza el tablero digital.
- o) Las actividades de documentación tienen un tiempo definido y no se realizaba todos los días.

La parte de monitoreo y control se realiza con Scrum, ya que el Scrum Master puede estar monitoreando tanto el avance del equipo como los riesgos que puedan surgir durante el proyecto, que es una práctica de CMMI. Esta actividad se realiza con el Daily Scrum, que es una reunión diaria que permite monitorear ambos aspectos, en esta reunión cada miembro del Scrum Team dice “Si puede avanzar”, “Si no pude avanzar”, y “Que riesgos y problemáticas tiene”, esto se documenta de manera simple por correo electrónico, logrando realizar la práctica de monitoreo y control de gente, actividades y riesgos.

Con el Daily Scrum se tiene visibilidad real del avance porque al actualizar la herramienta se genera el Burn Down Chart para la que se define un objetivo y reflejará automáticamente el esfuerzo total. Conforme avanzan los días mostrará como tendría que ser la productividad del equipo, la tendencia para cumplir el objetivo y cómo va el avance con el trabajo realizado hasta el momento.

Al estar finalizando el Sprint, se realizan actividades que permiten concluir de manera apropiada el proceso y con un producto funcionando, realizando:

- a) La tercera semana de cada Sprint la planeación está enfocada al área de Pruebas, ya que se libera el producto. Pruebas se encarga de probar el producto y se envían correcciones que se tienen que hacer.
- b) Se muestra el Demo del entregable funcionando con los requerimientos definidos en el Sprint.
- c) Si surgen cambios, se agrega al Product Backlog actualizando la línea base de productos y se hace la siguiente planeación.
- d) El área de Métricas lleva un control para ir midiendo la velocidad del equipo en cada Sprint, contando la cantidad de puntos de complejidad de los productos terminados y realizar un reporte al respecto como evidencia del proceso.
- e) Al final de cada SPRINT se hace la retrospectiva de lecciones aprendidas, se anota lo que funciono para seguirlo haciendo, lo que se debe dejar de hacer y las propuestas de mejora como aportación al proceso. El registro es individual,

se pone en posticks y se registra en un documento que proporciona el Scrum Master.

Se detectaron áreas de oportunidad en la forma de trabajo con RUP, ya que el realizar ciertas actividades tomaba bastante tiempo ejecutarlas y no generaban valor al proyecto, así como al cliente no le interesaban demasiado, por lo que se empezaron a quitar entregables en la parte de administración, realizando una tala regulatoria. Se tienen que administrar bien las actividades para conservar aquello que genera valor al proyecto y al cliente, sin dejar de hacer documentación por el hecho de ser ágil. En el área de ingeniería y pruebas también se eliminaron actividades justificadamente.

En la parte de Ingeniería se eliminó documentación que genera el programador como evidencia de haber realizado sus pruebas unitarias al terminar un componente o funcionalidad, esto era rudimentario, tedioso y no daba valor al proyecto. El programador realiza sus pruebas unitarias pero no documenta este proceso.

Para el área de pruebas, se elaboraba un script, que indicaba al área de desarrollo todas las funcionalidades que tenían que probar y revisar cada vez que se hacía una liberación. Esto servía como evidencia de que el programador había probado los programas o módulos antes de que lleguen a Pruebas. Los programadores hacen sus propias pruebas pero no siguen ningún script.

Se tienen dos maneras para hacer las Matrices de Pruebas, el diseñador de pruebas hace la matriz para realizar las pruebas unitarias, la otra opción, es que el desarrollador crea sus propios scripts de pruebas unitarias y las ejecuta. No se hacen las dos porque se duplica esfuerzos y perdería el concepto de ágiles.

Al terminar el proyecto piloto se definieron las técnicas que funcionaron y cuales no fueron útiles, para documentar las guías de los procesos con las técnicas ágiles exitosas y ser desplegadas todas al mismo momento en la empresa.

## PLAN DE TRABAJO

Se realiza la reunión llamada Sprint Planning, donde están las personas involucradas con el proyecto, incluyendo un administrador por disciplina. Ahí se realiza un dimensionamiento general, donde estiman el esfuerzo de productos y requerimientos ya sea por puntos de función o casos de uso. Al definir juntos el objetivo del Sprint y el Sprint Backlog, todas las disciplinas tienen la misma visión hacia donde van a trabajar y ponen las tareas necesarias para lograr ese objetivo.

El Scrum Master obtiene el número de horas que le toca a cada disciplina para todo el proyecto a partir de los puntos de función de manera automática, cada administrador de área planea y distribuye sus actividades dentro de la herramienta respetando el tiempo que le fue asignado. Cada administrador organizar el número de horas que dedicará a cada Sprint y a sus actividades, no puede planear más horas de las que tiene asignadas.

No solamente el equipo de ingeniería debe planear con la herramienta sino administración, pruebas, calidad, configuración y métricas logrando una administración integrada que cubre el modelo CMMI. El plan de trabajo es un proceso preestablecido que dice todas las actividades que se deben desarrollar durante el proyecto para cada una de las áreas.

Una junta de planeación puede llevar de 4 a 8 horas, 4 horas cuando el objetivo es claro o el equipo de trabajo es pequeño. El tamaño apropiado para un equipo Scrum es de 4 a 6 personas, si es más grande se sale de control aumentando los costos, por lo que es más apropiado subdividirlos y las reuniones se hacen con pequeños grupos, cuando se requiere una reunión con los diferentes equipos nada

más asiste un representante de equipo y se hace una junta llamada Scrum de Scrum.

La estrategia a seguir durante un proyecto se marca en la Guía de Ajuste donde se seleccionan las prácticas que se requiere para llevar a cabo el proyecto, hay prácticas que es necesario realizarlas aun cuando el cliente no las solicite para su proyecto, consideradas como prácticas mínimas necesaria para cumplir con el modelo. En adición, tiene la utilidad de saber que se cubre con las prácticas estipuladas por CMMI-DEV y asegurar que se está cumpliendo con lo contratado por el cliente.

Si el cliente quiere que vayas directamente al producto, entonces en tu Guía de Ajustes marcas las prácticas mínimas indispensables y el enfoque del proyecto es directo al producto. Si el cliente te pidió todos los artefactos disponibles en la Guía de Ajustes marcas todas las prácticas.

Las guías de ajuste proporcionan una dirección para el diseño, análisis y revisión de los procesos de software para mantener la coherencia con el modelo CMMI-DEV.

Actualmente en el SEDS no se trabaja la documentación en procesador de texto ni hoja de cálculo, sino que se ha creado una herramienta para hacer la planeación de todas las áreas involucradas en el proyecto; conforme se ha madurado el proceso se usan tableros digitales y dejado a un lado los tableros físicos. Los aspectos de cambios son importantes en ágiles y es algo que se está madurando, pero que ya está definido.

## CAMBIOS

El Product Owner teóricamente siempre debe estar cerca del desarrollo del proyecto, tener comunicación constante para resolver dudas, guiar la construcción

y supervisar el proyecto; es quien define la prioridad del cambio, las definiciones del negocio y sus prioridades, mientras que el equipo de trabajo da la prioridad técnica.

Por esencia las técnicas ágiles están preparadas para hacer cambios controlados, y el cliente puede solicitar un cambio en cualquier momento, los cuales tienen que ser evaluados según el impacto en el proyecto.

A continuación se muestran algunos criterios que pueden influir durante la evaluación de un cambio, recordando que en todos los casos se aplica la práctica de control de cambios.

- Cambios mínimos como un campo o una regla de negocio que no se había entendido, se debe aplicar el cambio dentro del Sprint.
- Cambio menor, se deja terminar el Sprint y el cambio se incorpora al Product Backlog para realizar la planeación del siguiente SPRINT.
- Cambio forzoso, se integra al Sprint actual y algún producto o tarea se quita de la planeación. No se tendrá el producto al 100% al final del Sprint como se planeó inicialmente.
- Aumento del alcance, adicionando productos o módulos para el proyecto, se agrega al Product Backlog si no se tiene impactó en el Sprint actual.
- Cambio de prioridades, fuera del alcance o de gran impacto, se detiene el Sprint.

En un control de cambios se tienen que realizar actividades necesarias para tener bajo control el impacto del cambio. Durante esta práctica se tiene que:

- a) Documentar el cambio solicitado por medio de un correo electrónico o minuta y compartir esta información con el cliente.
- b) Estimar el tiempo requerido para aplicar el cambio.

- c) Si el cambio aumento el alcance pero no se modificó lo que se tenía implementado hasta el momento, se deja terminar el Sprint y se planifican los cambios a partir del siguiente Sprint.
- d) Si el cambio afecta lo que se está haciendo en el Sprint actual, se para el Sprint y nuevamente se realiza la planeación para cumplir con el nuevo alcance solicitado.
- e) Validar con la Dirección el cambio.
- f) Autorizar y firmar el cambio, tanto por parte de la dirección como del cliente.
- g) Modificar el Backlog.
- h) Realizar planeación de Sprint.

## PRUEBAS

El área de pruebas entrega los resultados al probar el producto y poder realizar las correcciones pertinentes sobre el producto de trabajo, esto ocurre durante la tercera semana del Sprint.

El área de pruebas entra desde el inicio al proyecto y define las matrices de prueba y el equipo construye en base a estas matrices de pruebas, donde define lo que el sistema debería hacer cuando se haga una acción en especial o se introduzca algún valor, en lugar de narrarte el camino feliz o “happy path”. Se narran los caminos de éxito y los de fallos, entonces cuando el programador quiere liberar se tiene que ejecutar la matriz de pruebas antes, para asegurarse que todos los caminos están cubiertos.

El programador puede realizar las pruebas unitarias con base a la matriz, entonces, pruebas puede verificar otros caminos con la matriz base para los escenarios, pruebas funcionales o no funcionales.

Actualmente se está trabajando con una herramienta en la cual programas un script, lo ejecutas sobre una liberación y automáticamente te va dando el registro de los errores.

## **TRABAJO CON OTRAS ÁREAS**

El Scrum Master realiza una administración integral al incorporar todas las áreas de proceso en su plan de trabajo, donde las áreas ejecutan sus procesos en base a lo que tiene estipulado el Scrum Master. Inclusive lleva la administración de otros procesos que no están metidos en la parte ágil, como Calidad, Pruebas, Administración de la Configuración bajo esquema tradicional.

Hay procesos organizacionales como capacitación, que permite permear los procesos de todas las áreas a los proyectos, ya que las tablas de estimación tiene contemplado tiempo para todas las áreas de proceso.

La capacitación puede ser de manera personal como grupo de procesos u organizacional. Por ejemplo, todos los miembros de SEDS reciben capacitación dos veces al año independientemente de lo que se produzca en la fábrica.

Durante la ejecución del desarrollo del proyecto el área de Métricas emite un reporte con el estatus de cada indicador, cuando alguno tiene una alerta, tiene que ser atendida por el Scrum Master, quién implementa acciones correctivas en su plan de trabajo sobre dicho indicador. Con este reporte se va verificando y controlando que el proyecto cumpla con calidad y el tiempo establecido.

El área de calidad tiene sus propios procesos y documentos, lleva un control independiente, verifica que el proyecto cumpla con la integridad y tiempo planeado, que se genere la documentación correspondiente e intervienen en el proyecto si se tienen alertas en los análisis de métricas. El SQA acompaña todo el tiempo al Equipo Scrum durante las reuniones diarias, donde realiza listas de verificación sobre lo que se expone y menciona los problemas o desviaciones que

se están presentando para que el equipo este enterado de las desviaciones y establezca compromisos para resolverlas. El SCM realiza todos los días el versionado y respaldos de los productos generados en el día.

## **DIFERENCIAS ENTRE AMBAS FORMAS DE TRABAJO**

Algunas de las actividades que se realizaban en el desarrollo de software con RUP y se preservaron con la integración de las técnicas ágiles, son:

- Estándar de codificación y auto-documentación.
- Peer Review de los códigos, generalmente con un arquitecto, quien regresan un documento con observaciones que tienen que ser atendidas versionando con el estatus de esas observaciones.
- Integración continua, siempre sobre la última versión de todos los programadores.
- Respaldo y versionado del proyecto.

El uso de técnicas ágiles da una ventaja competitiva enorme ya que se están generando resultados satisfactorios en los proyectos conforme a lo planeado, el enfoque esta sobre el producto y no tanto sobre la documentación. Hay correcciones ágil no es perfecto pero permite tener visibilidad, los riesgos se levantan todos los días,

RUP maneja las disciplinas muy marcadas, si no se tiene una actividad no se puede pasar a la siguiente, comparado con ágiles que es más flexible en las actividades que puede realizar enfocándose en el producto y no tanto en la documentación, aunque no mantiene un proceso tan ordenado como RUP, los enfoques cambian totalmente.

En ágiles se definen los objetivos del Sprint junto con el Sponsor, Product Owner y el equipo de trabajo, cuidando el orden de las actividades a realizar considerando prioridades técnicas para realizar los productos, así como los tiempos para realizar

las actividades tienen el punto de vista técnico, esto es un cambio de paradigma total.

El cliente está más satisfecho porque siempre está viendo el producto real y no por medio de un documento, de tal manera que puede dar retroalimentación para tomar acciones sobre el producto.

## ÁREAS DE OPORTUNIDAD DE LA INTEGRACIÓN DE TÉCNICAS ÁGILES A CMMI

Han llegado recomendaciones para los cierres de proyectos, ya que son lo más complicado para la fábrica de software por que los clientes tienen miedo de firmar formalmente la entrega aun cuando ya la hayan aprobado N veces. Debido a la incertidumbre de si falla ¿Qué van a hacer?, ¿Van a estar aquí?, ¿Van a corregir? La mejor retroalimentación en este aspecto es la retroalimentación que puede dar al respecto el cliente.

Las técnicas ágiles de Test Drive Development y Feature Driven Development están en proceso de mejora.

## RESPUESTA DEL MERCADO A LA INTEGRACIÓN

La respuesta de los cliente ante esta forma de trabajo ha rendido buenos frutos, inclusive hay algunos van a regresar por la misma forma de trabajo. Los clientes han felicitado a los grupos de trabajo, inclusive han comentado que se hará una segunda fase y que abrirán más aplicaciones por los resultados satisfactorios.