



**UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FISICAS Y MATEMATICAS
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACION**

**HERRAMIENTA DE GESTION CUANTITATIVA DE PROYECTOS DE SOFTWARE
ORIENTADA POR UN PROCESO DE DESARROLLO**

**TESIS PARA OPTAR AL GRADO DE MAGISTER EN
TECNOLOGIAS DE LA INFORMACION**

DANIEL ALBERTO RAMIREZ ABARCA

**PROFESOR GUIA:
MARIA CECILIA BASTARRICA PIÑEYRO**

**MIEMBROS DE LA COMISIÓN:
JOHAN FABRY
SERGIO OCHOA DELORENZI
GONZALO ROJAS DURÁN**

**SANTIAGO DE CHILE
2014**

Resumen

Esta tesis tiene como objetivo definir y diseñar una herramienta de administración de proyectos de software orientada por un proceso de desarrollo, que permita obtener en forma oportuna un conjunto de métricas que apunten a mejorar la calidad del proceso y productos de la organización.

Este proyecto está contextualizado en la experiencia de Amisoft Ingeniería Ltda., la cual ha implementado un proceso denominado “Amisoft Process Framework” (APF), que enmarca las actividades de gestión, ingeniería y soporte, necesarias para guiar cada proyecto en su ejecución. Este proceso de desarrollo y mantención de software ha sido un gran paso para la organización en busca de la mejora continua y dar cumplimiento a la política de calidad de la empresa. No obstante, este paso no es suficiente para garantizar el éxito de la organización en la entrega de productos y servicios de calidad, siendo necesario llevar a cabo una medición del proceso de desarrollo para entender, monitorear, controlar y predecir el desarrollo y mantención de los proyectos y tomar buenas decisiones en su gestión.

El trabajo consistió en proveer una herramienta de gestión de proyectos de software integrada, orientada por el proceso de desarrollo APF, que permita, a partir de los datos de los proyectos ejecutados por la organización, realizar la medición de indicadores y el cálculo de métricas para un mejor entendimiento de la pertinencia del proceso durante el desarrollo y mantenimiento de los proyectos de Amisoft, y provea de la información necesaria para mejorar los procesos y productos de la organización. En particular, se incorporaron a la herramienta las métricas del valor ganado y la volatilidad de requerimientos.

Como parte de este trabajo, también se destaca la identificación y propuesta de las características y funcionalidades fundamentales para una herramienta de este tipo, así como también la ejecución de un proyecto piloto que permitiera validar el correcto funcionamiento del sistema y evaluar la calidad de las métricas obtenidas desde el mismo, en comparación con las obtenidas actualmente, a fin de validar su efectividad respecto a los objetivos definidos por la empresa y para este trabajo.

Agradecimientos

Dedico este trabajo con especial cariño a mi señora y mis hijos, Catalina y Sebastián, por todo el apoyo que me han entregado a lo largo de este proceso, siendo fuente de motivación y superación en los momentos más difíciles.

A mis padres, por inculcar en mí la pasión por aprender, por sus recomendaciones, preocupaciones, y todos esos detalles que me han convertido en lo que soy hoy. A mi hermana, dándose el tiempo de leer y corregir la primera versión de este trabajo.

Gracias a mi profesora guía, María Cecilia Bastarrica, por darme la posibilidad de trabajar junto a ella, brindándome su apoyo, tiempo, paciencia y dedicación.

Agradezco también a los profesores del Magíster, por la motivación y conocimientos entregados, siempre con la mejor disponibilidad para enseñar y resolver mis inquietudes. También quiero mencionar a la coordinación del magíster por su trato cordial y estar siempre disponibles en los momentos requeridos.

Finalmente, quiero agradecer a Amisoft, y a todos sus colaboradores, por el apoyo y la colaboración entregada para que este trabajo resultara exitoso.

Tabla de contenido

Capítulo I.....	1
1 Introducción.....	1
1.1 Contexto.....	1
1.2 Motivación.....	3
1.3 Oportunidad de Mejora.....	5
1.4 Objetivo general.....	6
1.5 Objetivos específicos.....	6
1.6 Descripción de la solución.....	6
1.7 Metodología.....	7
Capítulo II.....	9
2 Marco Teórico.....	9
2.1 Estándares para la Gestión de Proyectos.....	9
2.2 Administración de Proyectos en CMMI-DEV.....	9
2.3 Administración de Proyectos en RUP.....	13
2.4 Guía PMBOK para la Gestión de Proyectos.....	16
2.5 Conclusiones acerca de los estándares.....	17
2.6 El Proceso APF.....	19
2.7 Métricas para la toma de decisiones.....	20
2.8 Métricas de Amisoft.....	24
2.9 Herramientas de Gestión Cuantitativa de Proyectos.....	25
Capítulo III.....	29
3 Desarrollo de la Herramienta de Gestión de Proyectos.....	29
3.1 Introducción.....	29
3.2 Lista de características.....	29
3.3 Priorización de las características.....	31
3.4 Modelo de Dominio.....	33
3.5 Arquitectura del sistema.....	36
3.6 Modelo de Casos de Uso del Sistema.....	42

3.7 Primera Iteración.....	45
3.8 Segunda Iteración.....	51
3.9 Tercera Iteración	60
Capítulo IV.....	69
4 Implementación y Resultados Obtenidos.....	69
4.1 Metodología de implementación	69
4.2 Experimento Piloto.....	70
4.3 Resultados obtenidos	71
4.4 Comparación con el enfoque manual.....	75
4.5 Conclusiones acerca de los resultados obtenidos	81
4.6 Impacto económico del proyecto.....	81
4.7 Implementación de nuevas funcionalidades	84
Capítulo V.....	86
5 Conclusiones y Trabajo Futuro	86
5.1 Conclusiones	86
5.2 Lecciones Aprendidas	87
5.3 Trabajo Futuro.....	88
Bibliografía	90
Anexos	93
Anexo A - Áreas de Proceso APF.....	93
Procesos de Gestión	93
Procesos de Ingeniería	94
Procesos de Soporte.....	95
Uso del Proceso APF.....	96
Anexo B – Métricas de Amisoft.....	102
Adherencia al proceso.....	102
Pruebas Funcionales.....	103
Volatilidad de Requerimientos.....	103
Valor Ganado	104

Índice de Tablas

TABLA 1. ÁREAS DE PROCESO DE LA ADMINISTRACIÓN DE PROYECTOS	9
TABLA 2. ÁREAS COMUNES DE LOS ESTÁNDARES	18
TABLA 3. MÉTRICAS PARA CONTROL DEL PROYECTO	21
TABLA 4. MÉTRICAS DEFINIDAS EN MOTOROLA	22
TABLA 5. MEDICIONES PARA LAS ÁREAS DE PROCESO DE CMMI	23
TABLA 6. DESCRIPCIÓN DE LAS CARACTERÍSTICAS PROPUESTAS PARA UN SISTEMA DE GESTIÓN DE PROYECTOS	30
TABLA 7. PRIORIZACIÓN DE LAS CARACTERÍSTICAS	32
TABLA 8. RESUMEN VARIABLES VALOR GANADO DESDE SISTEMA	72
TABLA 9. RESUMEN VARIABLES ÍNDICES CPI-SPI DESDE SISTEMA	74
TABLA 10. RESUMEN VARIABLES VALOR GANADO DESDE PLANILLA	77
TABLA 11. RESUMEN VARIABLES ÍNDICES CPI-SPI DESDE PLANILLA	78
TABLA 12. COSTO ANALISTA DE MÉTRICAS EN HH	82
TABLA 13. COSTO RECOLECCIÓN Y PROCESAMIENTO DE MÉTRICAS	82
TABLA 14. COSTO IMPLEMENTACIÓN DEL SISTEMA EN UF	83
TABLA 15. CARACTERÍSTICAS INCORPORADAS EN EL SISTEMA	85
TABLA 16. MÉTRICA ADHERENCIA AL PROCESO	102
TABLA 17. MÉTRICA PRUEBAS FUNCIONALES	103
TABLA 18. INDICADOR VOLATILIDAD REQUERIMIENTOS DEL PLAN DE MEDICIÓN DE AMISOFT	103
TABLA 19. MÉTRICA VALOR GANADO	104

Índice de Figuras

FIGURA 1. PROCESO DE DESARROLLO APF	2
FIGURA 2. ÁREAS DE PROCESO BÁSICAS DE LA PLANIFICACIÓN DE PROYECTOS [4]	10
FIGURA 3. ÁREAS DE PROCESO AVANZADAS DE LA PLANIFICACIÓN DE PROYECTOS [4]	12
<i>FIGURA 4. ELEMENTOS DE RUP Y SUS RELACIONES</i>	13
<i>FIGURA 5. FLUJO DE TRABAJO ADMINISTRACIÓN DE PROYECTOS EN RUP [15]</i>	15
<i>FIGURA 6. RELACIÓN ENTRE ROLES, TAREAS Y ARTEFACTOS</i>	20
FIGURA 7. MODELO DE DOMINO – PROCESO DE DESARROLLO	34
FIGURA 8. MODELO DE DOMINO – PLANIFICACIÓN DEL PROYECTO	35
FIGURA 9. MODELO DE DOMINIO – MONITOREO Y CONTROL DEL PROYECTO	36
FIGURA 10. MODELO DE DOMINIO – ADMINISTRACIÓN DE REQUERIMIENTOS	36
FIGURA 11. VISTA DE CASOS DE USO Y SUS RELACIONES	38
FIGURA 12. VISTA LÓGICA	39
FIGURA 13. VISTA DE IMPLEMENTACIÓN	41
FIGURA 14. VISTA DE DESPLIEGUE	42
FIGURA 15. ACTORES DEL SISTEMA	43
FIGURA 16. DIAGRAMA DE CASOS DE USO DEL SISTEMA	44
FIGURA 17. DIAGRAMA DE SECUENCIA DEL CASO DE USO CREAR PROYECTO	46
FIGURA 18. DIAGRAMA DE SECUENCIA DEL CASO DE USO IDENTIFICAR PROYECTO	47
FIGURA 19. DIAGRAMA DE SECUENCIA DEL CASO DE USO ASIGNAR RECURSO	48
FIGURA 20. DIAGRAMA DE CLASES ITERACIÓN 1.	49
FIGURA 21. PÁGINA WEB IMPLEMENTACIÓN CREAR PROYECTO	50
FIGURA 22. PÁGINA WEB IMPLEMENTACIÓN ASIGNAR RECURSO	51
FIGURA 23. DIAGRAMA DE SECUENCIA DEL CASO DE USO CREAR TAREA	53
FIGURA 24. DIAGRAMA DE SECUENCIA DEL CASO DE USO REGISTRAR TRABAJO	55
FIGURA 25. DIAGRAMA DE SECUENCIA DEL CASO DE USO CONSULTAR ESTADO DE AVANCE	56
FIGURA 26. DIAGRAMA DE CLASES ITERACIÓN 2	57
FIGURA 27. PÁGINA WEB IMPLEMENTACIÓN CREAR TAREA	58
FIGURA 28. PÁGINA WEB IMPLEMENTACIÓN REGISTRAR TRABAJO	58
FIGURA 29. PÁGINA WEB IMPLEMENTACIÓN CONSULTAR ESTADO DE AVANCE	59
FIGURA 30. PÁGINA WEB IMPLEMENTACIÓN CONSULTAR ESTADO DE AVANCE	59
FIGURA 31. DIAGRAMA DE SECUENCIA DEL CASO DE USO CREAR REQUERIMIENTO	61
FIGURA 32. DIAGRAMA DE SECUENCIA DEL CASO DE USO CONSULTAR VOLATILIDAD DE REQUERIMIENTOS	64
FIGURA 33. DIAGRAMA DE CLASES ITERACIÓN 3	65

FIGURA 34. PÁGINA WEB IMPLEMENTACIÓN CASO DE USO CREAR REQUERIMIENTO	66
FIGURA 35. PÁGINA WEB IMPLEMENTACIÓN CASO DE USO ASOCIAR REQUERIMIENTO	66
FIGURA 36. PÁGINA WEB IMPLEMENTACIÓN CASO DE USO IDENTIFICAR RELEASE	67
FIGURA 37. PÁGINA WEB IMPLEMENTACIÓN CASO DE USO CONSULTAR VOLATILIDAD	68
FIGURA 38. CARTA GANTT ITERACIÓN N° 2 SITCI	70
FIGURA 39. RELEASE Y REQUERIMIENTOS ASOCIADOS DEL PROYECTO SITCI	71
FIGURA 40. GRÁFICO VALOR GANADO DESDE SISTEMA	73
FIGURA 41. GRÁFICO VALOR ÍNDICES CPI-SPI DESDE SISTEMA	74
FIGURA 42. VOLATILIDAD DE REQUERIMIENTOS PROYECTO SITCI – SISTEMA GESTIÓN	75
FIGURA 43. PLANILLA DE REGISTRO DE HORAS PROYECTO SITCI	76
FIGURA 44. PLANILLA EVALUACIÓN DE ESTADO PROYECTO SITCI	76
FIGURA 45. GRÁFICO VALOR GANADO PROYECTO SITCI DESDE PLANILLA	77
FIGURA 46. GRÁFICO VALOR ÍNDICES CPI-SPI PROYECTO SITCI DESDE PLANILLA	79
FIGURA 47. VOLATILIDAD DE REQUERIMIENTOS PROYECTO SITCI DESDE PLANILLA	79
FIGURA 48. EVOLUCIÓN DE COSTOS EN LA RECOLECCIÓN DE MÉTRICAS DEL ÁREA DE M&A VERSUS EL COSTO DE IMPLEMENTACIÓN DEL SISTEMA	84
<i>FIGURA 49. FLUJO DE TRABAJO DE LA FASE DE INICIO</i>	97
<i>FIGURA 50. FLUJO DE TRABAJO DE LA FASE DE ELABORACIÓN</i>	99
<i>FIGURA 51. FLUJO DE TRABAJO DE LA FASE DE CONSTRUCCIÓN</i>	100
<i>FIGURA 52. FLUJO DE TRABAJO DE LA FASE DE TRANSICIÓN</i>	101

Capítulo I

1 Introducción

1.1 Contexto

La presente tesis se contextualiza en la experiencia de Amisoft Ingeniería Ltda., una empresa de servicios de tecnología de la información con más de 14 años de experiencia. Está localizada en Santiago, y su misión es “*ser aliados de negocio de nuestros clientes en el área TI, mediante un alto compromiso con la calidad y la innovación*”.

Amisoft se ha propuesto como política de calidad satisfacer las necesidades y expectativas de sus clientes potenciando la agilidad y eficiencia de sus procesos de negocio. Para ello se compromete a proveer productos y servicios de TI confiables y eficientes mediante las siguientes estrategias:

- Promover la excelencia técnica y humana de la organización.
- Innovar en los procesos y en el uso de TI, que permitan mejorar los negocios actuales y futuros de la organización y de sus clientes.
- Implementar, mantener y mejorar continuamente sus procesos, en el marco de su sistema de gestión de la calidad.

En línea con su política de calidad, Amisoft ha implementado un proceso denominado “Amisoft Process Framework” (APF), que enmarca las actividades de gestión, ingeniería y soporte, necesarias para guiar a cada proyecto en su ejecución. La definición de estos procesos, se ha llevado a cabo siguiendo las mejores prácticas del modelo de calidad CMMI-DEV, la norma ISO 9001:2000 y el proyecto Tutelkan¹, que abarcan los procesos involucrados en el desarrollo de productos de software. El proceso APF se ha implementado en todos los proyectos de desarrollo y mantención de la organización en el transcurso de los últimos dos años. Algunas de las características del proceso APF son:

- *Está basado en desarrollo iterativo incremental.* Iterativo ya que se encuentra definido en un ciclo de desarrollo, en el cual cada iteración es vista como un mini proyecto que abarca todas las áreas de proceso definidas. Incremental, porque en cada iteración se alcanza una mayor cercanía con los objetivos finales del proyecto, es decir, se agrega nuevo valor (funcionalidades) al producto final.

¹ www.tutelkan.info

- *Es orientado y dirigido por Casos de Uso.* Los Casos de Uso son utilizados para obtener una mejor especificación en los requerimientos de usuario, permitiendo establecer una relación entre ellos y un mejor seguimiento de los mismos. Con los Casos de Uso definidos se puede crear el diagrama de Casos de Uso y su especificación, además de los diferentes diagramas que ofrece UML y que serán desarrollados según especificaciones del proyecto.
- *Tiene cuatro fases de desarrollo.* Al igual que RUP [1], el desarrollo de software se encuentra enmarcado en cuatro fases de desarrollo: Inicio, Elaboración, Construcción y Transición. Cada fase se compone de iteraciones, enfocándose en mayor o menor medida en cada una de las áreas de proceso, dependiendo de la etapa y de la iteración en que se encuentre el desarrollo del proyecto.

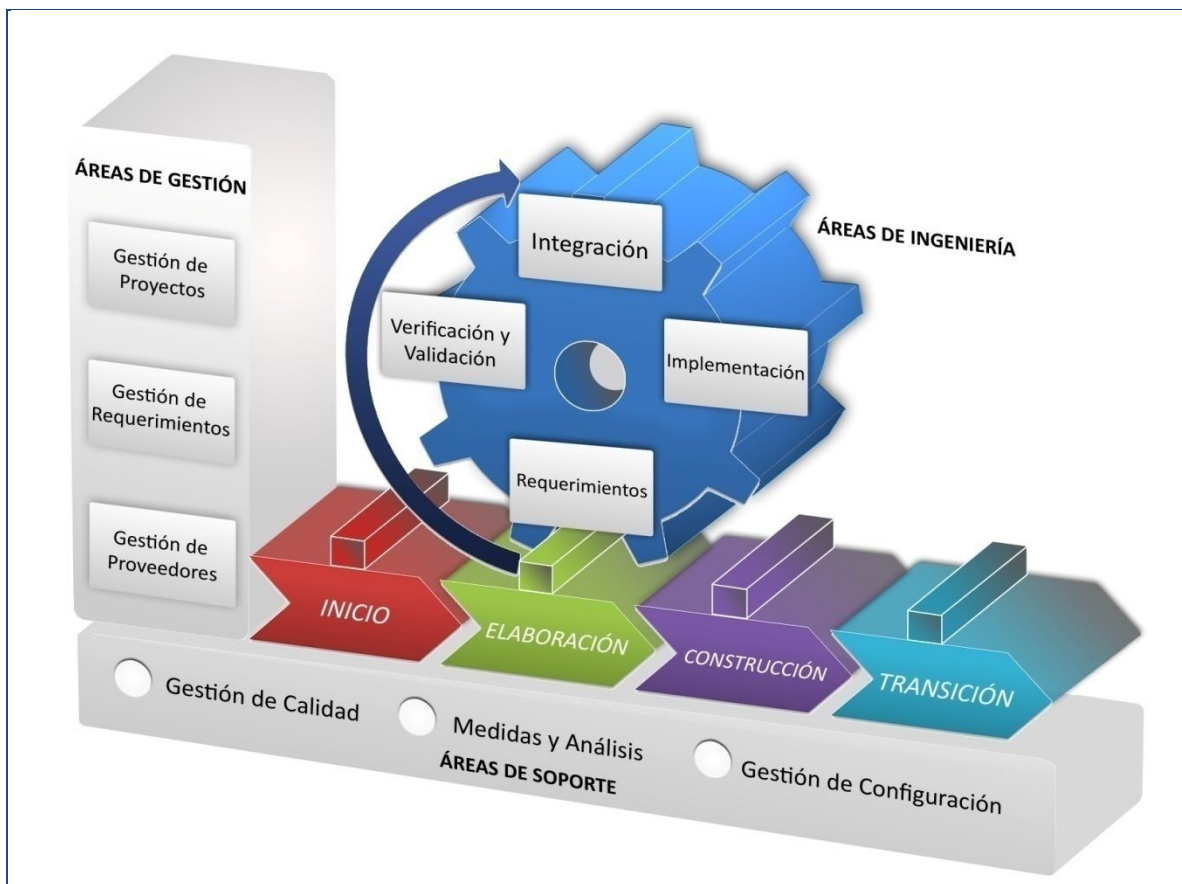


Figura 1. Proceso de Desarrollo APF

Dentro de APF las áreas de proceso se encuentran enmarcadas dentro de tres categorías: Procesos de Gestión, Procesos de Ingeniería y Procesos de Soporte.

- *Procesos de Gestión:*

Estos se encuentran enfocados directamente en el proyecto, cubriendo las necesidades relacionadas con la planificación, seguimiento y control del mismo. En esta categoría se encuentran las siguientes áreas de proceso: **Planificación de Proyectos (PP)**, **Monitoreo y Control de Proyecto (PMC)** y **Administración de Requerimientos (REQM)**.

- *Procesos de Ingeniería:*

Estos procesos dan soporte al ciclo de vida de desarrollo del producto; abarcan desde la toma de requerimientos hasta cuando el producto se encuentra totalmente operativo. En esta categoría se encuentran las siguientes áreas de proceso: **Desarrollo de Requerimientos (RD)**, **Solución Técnica (TS)**, **Verificación (VER)**, **Validación (VAL)** e **Integración de producto (PI)**.

- *Procesos de Soporte:*

En esta categoría se encuentran los procesos esenciales para soportar el desarrollo y mantenimiento del producto. Son transversales a todas las áreas de proceso definidas anteriormente, ya que éstas ocupan los resultados obtenidos por estos procesos. En esta categoría se encuentran las siguientes áreas de proceso: **Administración de la Configuración (CM)**, **Aseguramiento de Calidad de Procesos y Productos (PPQA)** y **Medición y Análisis (M&A)**.

1.2 Motivación

En el transcurso de los últimos dos años, el proceso APF ha sido adoptado por todos los proyectos de la organización, tanto en los proyectos de desarrollo nuevos como los proyectos de mantención. Durante este período el grupo de procesos ha realizado un plan de seguimiento con el objetivo de garantizar que las actividades definidas sean entendidas y ejecutadas por todos los involucrados según lo establecido. Uno de los principales indicadores utilizados para este objetivo ha sido la adherencia al proceso, el cual se mide a partir de los artefactos generados por los proyectos según cada una de las fases del proceso (artefacto completo, incompleto o no existente).

Aún cuando este indicador ha sido útil para establecer la internalización del proceso por parte de los distintos proyectos, no es el único ni el más adecuado para medir la adherencia a las actividades del proceso en la ejecución de los mismos, ya que no permite concluir si se están realizando todas las actividades definidas. Esto ha motivado

la necesidad de generar nuevas métricas que permitan medir más eficazmente la adherencia y calidad del proceso de desarrollo de software, que permitan tomar mejores decisiones acerca de su configuración y uso. Por otro lado, en el afán por mejorar el control de los proyectos y mejorar la calidad de los productos, se ha establecido un conjunto de métricas que permiten gestionar los proyectos.

Uno de los inconvenientes que se ha identificado es que la información generada por los proyectos de desarrollo y mantención se encuentra dispersa en distintas fuentes, tales como artefactos del proyecto, artefactos del proceso y distintas herramientas tecnológicas de apoyo utilizadas en la organización. Este aspecto dificulta la recolección de los datos, lo que implica a su vez un mayor esfuerzo por parte de los involucrados para generar las mediciones y una mayor posibilidad de introducir errores en los cálculos, ya que en muchas ocasiones los datos no son homologables y deben ser inferidos. En este punto influye que la mayoría de las herramientas disponibles en el mercado están enfocadas principalmente en la planificación y seguimiento de proyectos de software, y no así en la mejora de procesos de desarrollo de software y la gestión cuantitativa de los mismos [2], por lo que una única herramienta no satisface todas las necesidades de la organización.

El problema mencionado anteriormente se debe principalmente a que no todas las planificaciones de los proyectos usan las mismas definiciones y formato para las actividades y sus entregables, por lo que la información generada en los distintos proyectos no es homogénea. Estas diferencias ocasionan que, en muchas oportunidades, no es posible usar esta información como un input válido para mejorar el proceso, así como tampoco como información histórica que permita inferir el comportamiento de futuros proyectos de similares características, ya que realizar este análisis tendría un alto costo para la organización.

Dado lo anterior, existe una serie de interrogantes a las cuales se desearía dar respuesta, pero que en el escenario actual es difícil de determinar. Algunas de estas interrogantes son:

- ¿Se ajusta el proceso actual a las necesidades de los distintos proyectos?
- ¿Cuál es el porcentaje de adherencia de los proyectos de la organización a las actividades definidas por el proceso de desarrollo para cada una de las áreas de proceso?
- ¿Son útiles todas las actividades definidas en el proceso para los proyectos tanto de desarrollo como de mantención?
- ¿Cuáles son las principales actividades desarrolladas en los proyectos?

- ¿Cuáles son las actividades menos desarrolladas? ¿Son estas necesarias?
- ¿Cuánto tiempo y esfuerzo se invierte para realizar las actividades de una fase del proceso por cada área de proceso?

1.3 Oportunidad de Mejora

Contar con un proceso para el desarrollo y mantención de software ha sido un gran paso para la organización en busca de la mejora continua y dar cumplimiento a la política de calidad de Amisoft. No obstante lo anterior, este paso no es suficiente para garantizar el éxito de la organización en la entrega de productos y servicios de calidad. Es necesario llevar a cabo una medición del proceso de desarrollo para entender, monitorear, controlar y predecir el desarrollo y mantención de los proyectos y tomar buenas decisiones en su gestión. [17]

La oportunidad de mejora propuesta en esta tesis se enfoca en proveer una herramienta de gestión de proyectos de software integrada, orientada por el proceso de desarrollo APF, que permita generar, de forma oportuna, métricas previamente definidas. Mediante esta herramienta, se pretende dar respuesta a las interrogantes presentadas anteriormente, y obtener un mejor entendimiento de la pertinencia del proceso durante el desarrollo y mantenimiento de los proyectos de Amisoft, y que además, provea de la información necesaria para mejorar los procesos y productos de la organización.

Una herramienta que apoye la gestión de proyectos desde la base de un proceso definido, orientado a la entrega de métricas de forma rápida y oportuna, permitiría dirigir la gestión desde la arista del proceso, generando un vínculo entre el proceso, las actividades planificadas, las tareas asignadas, los artefactos generados y los roles del proyecto. Mediante esta vinculación se podrá generar información que no sólo permita guiar al jefe de proyecto en su gestión, sino también entregar información histórica para futuras estimaciones y un input a los encargados del proceso de la organización para determinar el comportamiento del mismo, y la necesidad de futuras mejoras. Todas estas características se encontrarían contenidas en una única fuente de información, lo que simplificaría además la gestión de los datos.

1.4 Objetivo general

El objetivo general de este trabajo de tesis es definir y diseñar una herramienta de administración de proyectos de software orientada por un proceso de desarrollo que permita obtener en forma oportuna un conjunto de métricas de proceso y proyectos que apunten a mejorar la calidad del proceso y productos de la organización.

1.5 Objetivos específicos

- Determinar las características mínimas de una herramienta de apoyo a la gestión de proyectos dirigida por un proceso utilizando como referencia el estudio de metodologías y estándares utilizados en la gestión de proyectos.
- Establecer las métricas de proceso y proyecto que serán obtenidas a partir de la herramienta de apoyo para el proceso de gestión de proyectos de la empresa.
- Diseño y análisis de la herramienta de apoyo a la gestión de proyectos.
- Implementar un prototipo de la herramienta que permita obtener de forma inmediata al menos dos de las métricas definidas.
- Validar la usabilidad de la herramienta y comprobar que no entorpece el normal desarrollo de los proyectos.

1.6 Descripción de la solución

Dentro de los resultados esperados a partir del desarrollo de la solución propuesta en este trabajo de tesis se encuentran los siguientes:

- Gestión y control de los proyectos de desarrollo y mantención en forma cuantitativa.
- Evaluación del progreso de los proyectos, con respecto a la estimación versus el esfuerzo.
- Evaluación del tiempo y esfuerzo real para realizar las actividades del proceso de desarrollo de software.
- Evaluación del tiempo y esfuerzo real para el desarrollo del proyecto.
- Evaluación de la adherencia de los proyectos al proceso definido por la organización.
- Generación de información para facilitar la mejora y adaptación de los procesos de la organización.
- Generación de información histórica para facilitar la evaluación y estimación de futuros proyectos.

- Mayor entendimiento y utilización del proceso organizacional por parte de todos los involucrados.

1.7 Metodología

Para cumplir con los objetivos de esta tesis se hace necesario examinar las necesidades de la organización y realizar un análisis del estado del arte en los temas relativos a la gestión de proyectos y la mejora de procesos de software. Con este fin, se ha definido el siguiente plan de trabajo:

- Estudio de estándares para procesos de software, tales como la norma ISO/IEC 15504, el modelo de madurez CMMI-DEV, la guía PMBOK para la administración de proyectos, proyecto Tutelkan, el proceso APF, entre otros, que permitan determinar características generales y buenas prácticas en la gestión de proyectos, que podrían servir de referencia para una herramienta de gestión.
- Estudio del estado del arte para la definición de métricas de software. Esto permitirá entender y orientar las características de las métricas que debería incluir una herramienta de estas características.
- Definir las características y funcionalidades de la herramienta de apoyo a la gestión cuantitativa de proyectos orientada por un proceso.
- Levantamiento de requerimientos para la herramienta de gestión de proyectos de software, tomando como referencia las necesidades de la empresa Amisoft.
- Definición de las métricas para la evaluación y mejora del proceso de la organización que serán consideradas en la herramienta de gestión, basándose en los objetivos organizacionales y el estado del arte anteriormente mencionado.
- Definición de métricas para la evaluación y control de los proyectos de la organización, considerando las métricas ya definidas por la organización y los estudios realizados anteriormente.
- Analizar y diseñar la herramienta de apoyo a la gestión de proyectos utilizando una metodología ad-hoc para este proyecto.
- Implementar un prototipo con las funcionalidades mínimas requeridas para la gestión de proyectos, incluyendo la recolección de datos para dos de las métricas definidas para la herramienta. En particular, interesa incorporar la medición del valor ganado y la volatilidad de los requerimientos.

- Utilizar el prototipo de la herramienta en un proyecto de desarrollo de Amisoft. Analizar los resultados. Comparar los resultados obtenidos por la herramienta con los resultados obtenidos a través del proceso actual.
- Concluir si la herramienta desarrollada cumple con los resultados propuestos en el presente trabajo.

Capítulo II

2 Marco Teórico

2.1 Estándares para la Gestión de Proyectos

Es necesario obtener un entendimiento común acerca de la gestión de proyectos, y cómo diferentes estándares reconocidos en el mercado plantean prácticas y actividades para llevarla a cabo. En particular nos enfocaremos en el modelo CMMI, el proceso RUP y las prácticas de la guía PMBOK.

De este entendimiento se podrán desprender cuáles son los principales aspectos que se debe tener en consideración para una gestión de proyectos efectiva, y cuáles de estas características serían útiles de implementar en una herramienta de gestión.

2.2 Administración de Proyectos en CMMI-DEV

CMMI-DEV, desarrollado por el Software Engineering Institute, es un modelo de referencia que cubre las actividades para desarrollar productos y servicios. Está compuesto de 22 áreas de procesos agrupadas en cuatro categorías: Administración de Procesos, Administración de Proyectos, Ingeniería y Soporte. [4]

La categoría de Administración de Proyectos tiene un total de siete áreas de proceso que describen las actividades relacionadas a la planificación, monitoreo y control del proyecto, las que a su vez son divididas en básicas y avanzadas. Las áreas de proceso para la Administración de Proyectos son presentadas en la *Tabla 1*.

Tabla 1. Áreas de proceso de la administración de Proyectos

Área de Proceso Administración de Proyectos		Nivel de Madurez	
Planificación de proyectos	PP	2	Básica
Administración de requerimientos	REQM	2	Básica
Monitoreo y control de proyectos	PMC	2	Básica
Administración de acuerdos con proveedores	SAM	2	Básica
Administración de riesgos	RSKM	3	Avanzada
Administración integrada de proyectos	IPM	3	Avanzada
Administración cuantitativa de proyectos	QPM	4	Avanzada

Las áreas de proceso básicas para la administración de proyectos incluyen actividades enfocadas en establecer y mantener el plan del proyecto y los compromisos con los involucrados, monitorear el progreso del proyecto y su correlación a lo planificado, tomar acciones correctivas y administrar los acuerdos con los proveedores. Como se muestra en la Figura 2, estas áreas de proceso interactúan entre sí, así como también interactúan con áreas de proceso de otras categorías.

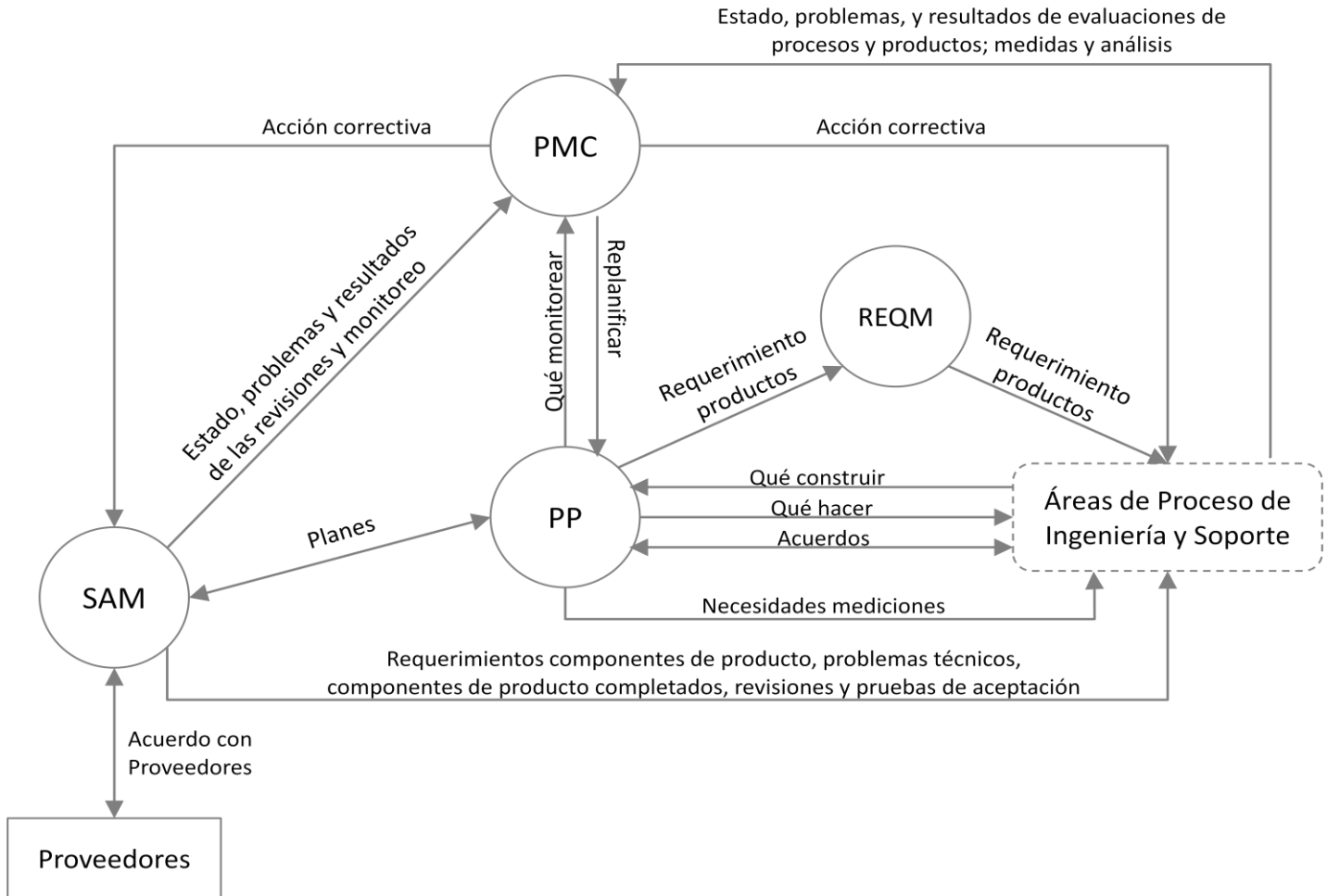


Figura 2. Áreas de proceso básicas de la planificación de proyectos [4]

El área de proceso Planificación de Proyectos incluye el desarrollo del plan del proyecto, involucrando a todos los interesados relevantes y obteniendo un compromiso de estos con el plan del proyecto, así como mantener el plan actualizado. El inicio de la planificación comienza con los requerimientos, que definen el producto y el proyecto. El plan del proyecto incorpora todas las actividades de administración y desarrollo que se

realizarán a lo largo del proyecto. De esta forma también se debe revisar otros planes que afecten al proyecto, tales como los planes de gestión de la configuración, verificación, medición y análisis.

El área de proceso Monitoreo y Control del Proyecto contiene prácticas para monitorear y controlar las actividades planificadas y tomar acciones correctivas en caso de desviaciones significativas. El plan del proyecto especifica la frecuencia de las revisiones y las medidas usadas para monitorear el progreso, el cual es determinado principalmente mediante la comparación del estado actual del proyecto respecto a lo planificado inicialmente. Cuando el estado actual se desvía significativamente de los valores esperados, se considerarán las acciones correctivas necesarias. Estas acciones pueden incluir la re-planificación, la cual requerirá las prácticas usadas en Planificación de Proyectos.

El área de proceso de Administración de Requerimientos mantiene los requerimientos. Describe las actividades que se realizarán para obtener y controlar los cambios en los requerimientos, y asegura que otros planes que pudieran verse afectados sean actualizados adecuadamente. Además se encarga de proveer la trazabilidad de los requerimientos, partiendo por las necesidades de los usuarios hasta los requerimientos del producto y de sus componentes.

Además la Administración de Requerimientos se asegura que los cambios a los requerimientos sean reflejados en los planes del proyecto, actividades y productos de trabajo. Este ciclo de cambios puede afectar a las áreas de proceso de Ingeniería; por ende, la administración de requerimientos es una actividad dinámica, a menudo recursiva, siendo fundamental para mantener los procesos de ingeniería controlados y disciplinados. El área de procesos de Administración de Acuerdos con Proveedores se encarga de las necesidades del proyecto, y de adquirir productos o servicios a través de proveedores.

Las áreas de proceso avanzadas para la Administración de Proyectos se enfocan en actividades tales como: definir un proceso ajustado para el proyecto a partir de los procesos estándares de la organización, establecer el ambiente de trabajo del proyecto a partir de los ambientes estándares de la organización, coordinar y colaborar con los involucrados relevantes, conformar y mantener los equipos de trabajo para la realización de los proyectos, gestionar cuantitativamente el proyecto, gestionar los riesgos.

Como se muestra en la Figura 3, todas las áreas de proceso avanzadas dependen de la habilidad de planificar, monitorear, y controlar el proyecto, las cuales son provistas por las áreas de proceso básicas.

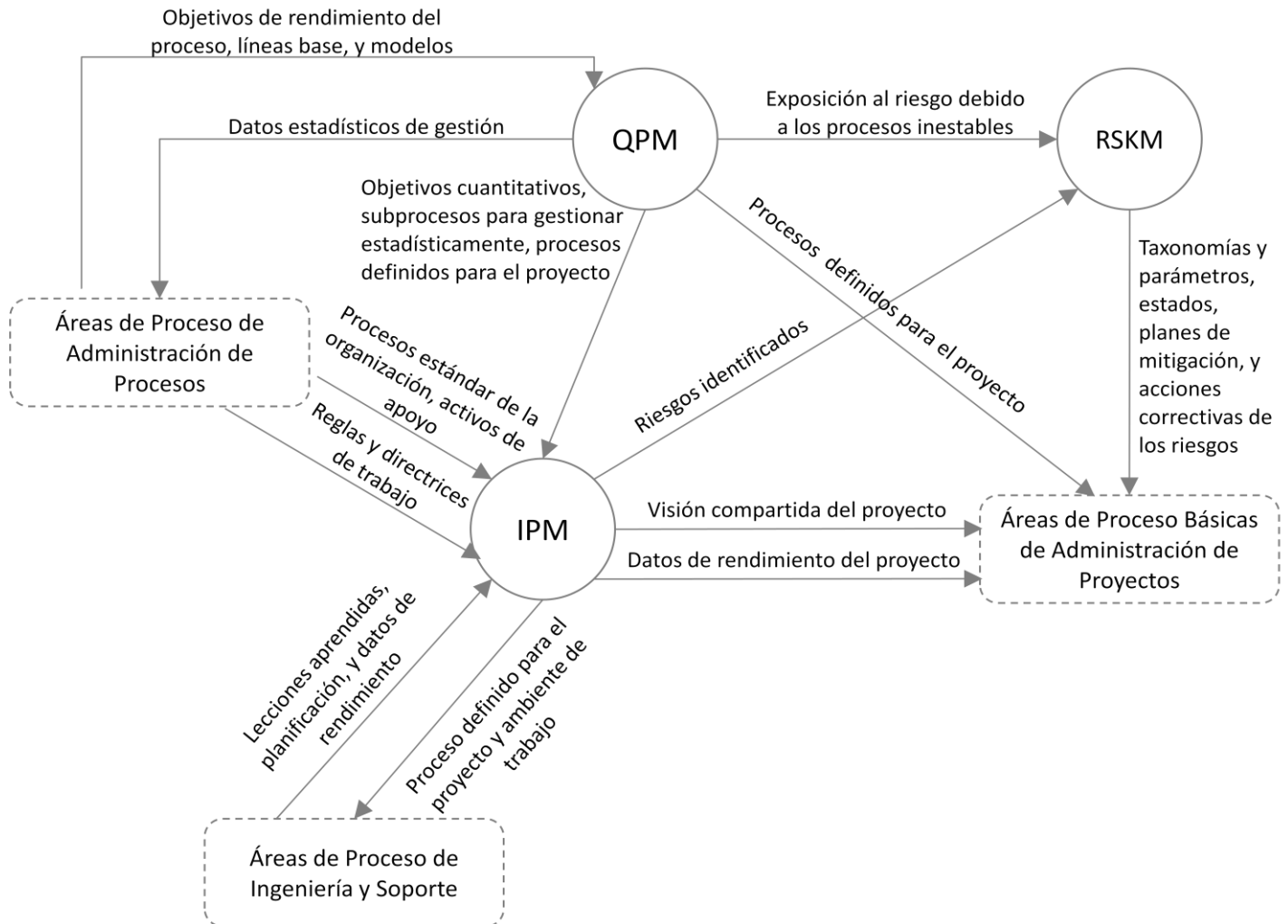


Figura 3. Áreas de proceso avanzadas de la planificación de proyectos [4]

La identificación y monitoreo de los riesgos son cubiertos en las áreas de proceso Planificación del Proyecto y Monitoreo y Control del Proyecto. El área de proceso Administración de Riesgos permite que esta actividad adquiera un enfoque de largo plazo en la administración de los riesgos, con actividades que incluyen la identificación de parámetros de riesgos, evaluaciones del riesgo y mitigaciones del riesgo.

El área de proceso llamado Administración Cuantitativa de Proyectos establece objetivos enfocados en la calidad y rendimiento del proceso y administra cuantitativamente el proyecto. Los objetivos de calidad y rendimiento del proceso del proyecto están basados en los objetivos establecidos por la organización y el cliente. El proceso definido del proyecto es compuesto usando estadísticas y otras técnicas cuantitativas. Este tipo de análisis permite al proyecto predecir si alcanzará sus objetivos de calidad y rendimiento

del proceso. Basado en la predicción, el proyecto puede ajustar el proceso definido o puede negociar cambios a los objetivos de calidad y rendimiento del proceso. En la medida que el proyecto progresa, el rendimiento de los subprocesos seleccionados es monitoreado cuidadosamente para ayudar a evaluar si el proyecto está en buen camino para alcanzar sus objetivos.

2.3 Administración de Proyectos en RUP

RUP es un proceso de ingeniería de software iterativo, centrado en la arquitectura, y guiado por los casos de uso, que define responsables de qué, cómo y cuándo se hacen las cosas [15]. Se basa en una serie de principios que apoyan el desarrollo iterativo, los cuales son implementados en su proceso a través de dos dimensiones:

- 1 Estructura dinámica o **dimensión de tiempo del proceso**, expresado en términos de ciclos, fases, iteraciones e hitos a lo largo del ciclo de vida de un proyecto.
- 2 Estructura estática o **dimensión de disciplinas**, que describe una agrupación lógica de los elementos del proceso (roles, actividades, artefactos y de las guías y conceptos asociados), y cómo estos interactuarán en un flujo de trabajo, presentados en la *Figura 4*.



Figura 4. Elementos de RUP y sus relaciones

RUP propone un conjunto de nueve disciplinas divididas en disciplinas técnicas y disciplinas de soporte. Entre las disciplinas de soporte se encuentra la administración de proyectos, que provee una guía para administrar proyectos.

A diferencia de otros frameworks, RUP se enfoca puntualmente en aspectos propios del proceso de desarrollo de software iterativo, y no cubre otros aspectos típicos de la administración de proyectos, tales como la administración de recursos humanos (contratación, entrenamiento, evaluación), la gestión de presupuestos y contratos (clientes y proveedores), o la gestión financiera.

Los objetivos principales de la disciplina de gestión de proyectos en RUP están:

- **Planificar un proyecto iterativo**

Los objetivos de la planificación son asignar tareas y responsabilidades a los recursos del equipo a lo largo del proyecto, y monitorear el progreso relativo a la planificación y detectar potenciales problemas. RUP recomienda que el desarrollo se base en dos tipos de planes: plan de la fase y planes de iteraciones. La planificación de las fases es de alto nivel y sirve para resumir el alcance y supuestos del proyecto; mientras la planificación por iteraciones contiene un mayor detalle de las actividades que se realizarán. La idea detrás de esto es que no se desperdicie tiempo tratando de planificar más allá de lo que se conoce en el momento.

- **Administrar los Riesgos**

La administración de riesgos busca hacerse cargo de todos aquellos aspectos que son desconocidos para el proyecto. Por cada riesgo percibido se debe decidir con anticipación qué se va a hacer y conducir la planificación de cada iteración para mitigar o eliminar los riesgos lo más temprano posible.

- **Monitorear y Controlar el proyecto**

El monitoreo y control del proyecto cubre aspectos tales como:

- Enfrentar las peticiones de cambio, y programar estos cambios en la iteración actual o futuras.

- Constantemente monitorear el proyecto en términos de los riesgos activos, objetivos de progreso y calidad.
- Reportar el estado del proyecto y enfrentar problemas que vayan ocurriendo y resolverlos de acuerdo a un plan de resoluciones.

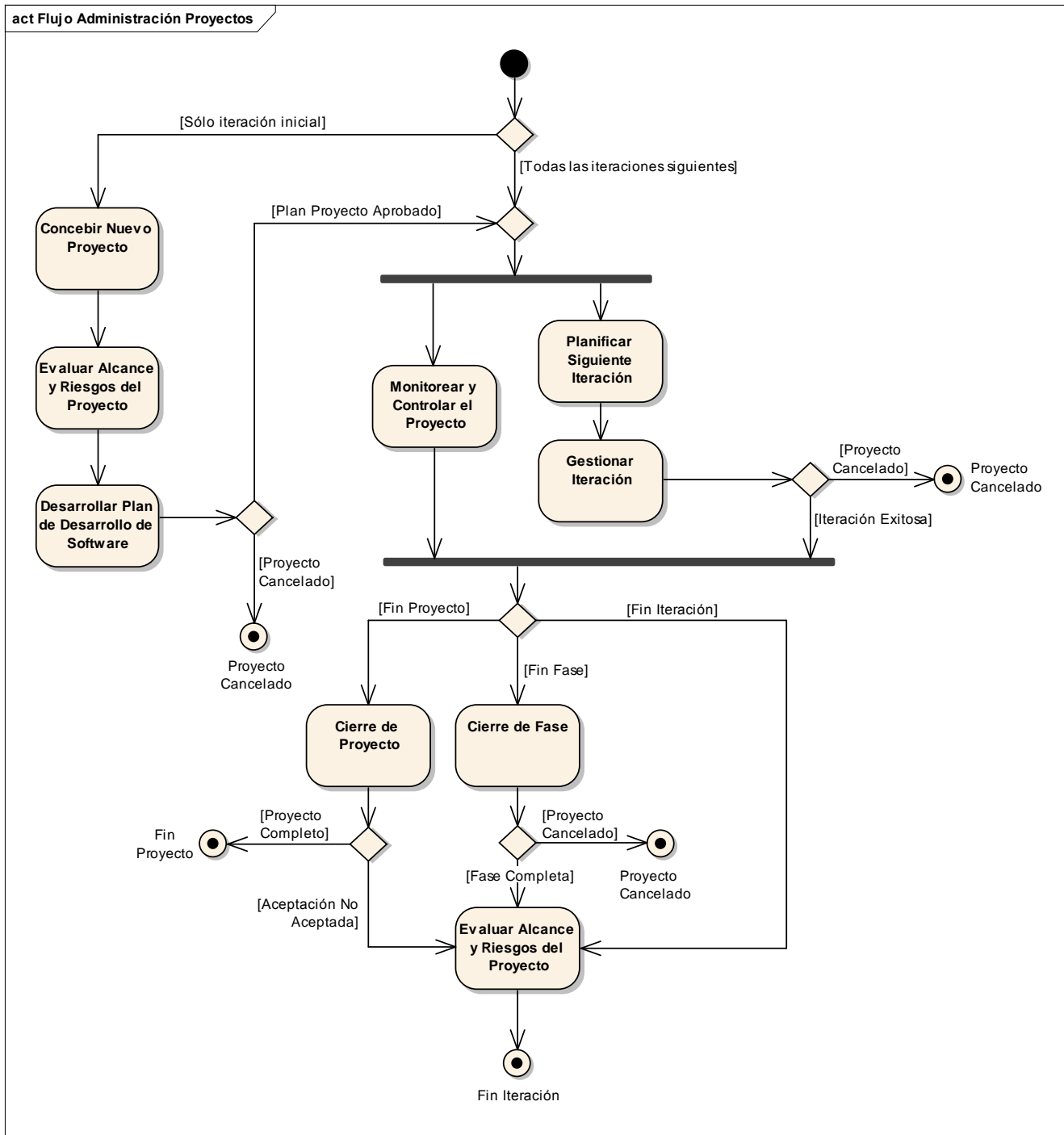


Figura 5. Flujo de Trabajo Administración de Proyectos en RUP [15]

Para llevar a cabo estas actividades es vital realizar mediciones al proyecto. Las medidas permiten evaluar qué tan cerca o lejos se está de los objetivos trazados en el plan (completitud, calidad, conformidad con los requerimientos), así como también permite mejorar las planificaciones futuras basados en la experiencia.

Los aspectos mencionados anteriormente son materializados a través de un workflow de iteración utilizando la notación UML, donde cada actividad representa un detalle de trabajo en RUP, el cual es presentado en la *Figura 5*.

2.4 Guía PMBOK para la Gestión de Proyectos

La guía PMBOK, desarrollada por el Project Management Institute, proporciona una guía para la gestión de proyectos, definiendo métodos, procesos y buenas prácticas que buscan aumentar las posibilidades de éxito en los proyectos [20]. A diferencia de CMMI-DEV y RUP, el PMBOK aporta un conjunto de buenas prácticas para la gestión de proyecto de todo tipo, y no está enfocado meramente en los proyectos de desarrollo de software.

La guía PMBOK propone nueve áreas de conocimiento. Estas son:

1. **Alcance:** Define los procesos para asegurar que el proyecto cumpla con todos los objetivos definidos y que incluya todo el trabajo necesario para completarlo. También define y controla todo aquello que se incluye y que no se incluye en el proyecto.
2. **Tiempo:** Define los procesos necesarios para asegurar el oportuno término del proyecto.
3. **Calidad:** Define los procesos y actividades para determinar responsables, objetivos y políticas de calidad de acuerdo a las necesidades del proyecto.
4. **Costos:** Incluye los procesos requeridos para asegurar que el proyecto es completado conforme al presupuesto aprobado; para ello define procesos para estimar, presupuestar y controlar los costos en el transcurso del proyecto.
5. **Integración:** Describe los procesos y actividades necesarios para asegurar que los distintos elementos del proyecto estén adecuadamente identificados, definidos, combinados, unificados y coordinados. Esto implica la asignación de recursos, balance de objetivos y administrar las interdependencias entre las áreas de conocimiento.
6. **Comunicaciones:** Describe los procesos relacionados con la generación, recolección, distribución, almacenamiento y disposición final de la información del proyecto, en

tiempo y forma. La comunicación efectiva conecta a todos los interesados del proyecto para compartir las experiencias, perspectivas e intereses en la ejecución y resultados del proyecto.

7. **Recursos Humanos:** Se preocupa de organizar y gestionar el equipo del proyecto, compuesto por todos aquellos con un rol asignado y responsabilidades para completar el proyecto.
8. **Riesgos:** Define los procesos relacionados con la gestión de riesgos del proyecto. Su objetivo es aumentar la probabilidad e impacto de los eventos positivos y disminuir la probabilidad e impacto de los eventos negativos del proyecto.
9. **Adquisiciones:** Procesos requeridos para comprar o adquirir bienes, servicios o resultados, así como para contratar procesos de dirección

2.5 Conclusiones acerca de los estándares

CMMI-DEV, RUP y la guía PMBOK comparten una serie de características en torno a la gestión de proyectos. Todos ellos reconocen la gestión de proyectos como una tarea iterativa, debido a su elaboración progresiva, y proponen una serie de buenas prácticas para una gestión efectiva.

CMMI-DEV provee una guía para mejorar el proceso y mide la capacidad y madurez de adopción de la misma. Para ello, se concentra en la descripción de los objetivos de una correcta gestión y desarrollo de proyectos de software, vale decir determina el “qué” a través de objetivos específicos. En tanto, RUP se enfoca en las mejores prácticas de la gestión de proyectos en el contexto del desarrollo de proyectos de software, y apunta a detallar el “cómo” con definiciones de procesos listas para ser utilizadas. Por otro lado, la guía PMBOK propone prácticas genéricas y aplicables a la gestión de proyectos en cualquier dominio de aplicación.

Existen análisis en los cuales se realizan comparaciones entre los tres modelos, y en ellos se ha llegado a la conclusión de que no existen mayores incompatibilidades entre estos estándares. Por ejemplo, Uttangi y Azeem proponen que RUP y CMMI pueden ser integrados de manera de maximizar la sinergia entre ambos, minimizar la redundancia, y cubrir los vacíos en el proceso que ambos definen [27].

De esta forma se puede concluir que estos estándares son complementarios, pues en los aspectos que alguno de los estándares no profundiza, se puede recurrir a las prácticas sugeridas en otro de ellos.

En la *Tabla 2* se presenta un resumen de los temas abordados por los estándares CMMI-DEV, RUP y PMBOK, agrupados por aquellos que abordan áreas comunes o que tienen similitudes en la gestión de proyectos.

Tabla 2. Áreas comunes de los estándares

CMMI-DEV Área de Proceso	RUP Disciplina	PMBOK Área de Conocimiento
<ul style="list-style-type: none"> • Planificación de Proyectos • Monitoreo y Control de Proyectos • Administración de Requerimientos 	<ul style="list-style-type: none"> • Administración de Proyectos • Requerimientos • Despliegue • Gestión del cambio y configuraciones 	<ul style="list-style-type: none"> • Integración
<ul style="list-style-type: none"> • Administración de Requerimientos 	<ul style="list-style-type: none"> • Administración de Proyectos • Requerimientos • Gestión del cambio y configuraciones 	<ul style="list-style-type: none"> • Alcance
<ul style="list-style-type: none"> • Planificación de Proyectos • Monitoreo y Control de Proyectos 	<ul style="list-style-type: none"> • Administración de Proyectos 	<ul style="list-style-type: none"> • Tiempo
<ul style="list-style-type: none"> • Planificación de Proyectos • Monitoreo y Control de Proyectos 	<ul style="list-style-type: none"> • Administración de Proyectos 	<ul style="list-style-type: none"> • Costo
<ul style="list-style-type: none"> • Aseguramiento de Calidad de Proceso y Producto 	<ul style="list-style-type: none"> • Administración de Proyectos • Gestión del cambio y configuraciones 	<ul style="list-style-type: none"> • Calidad
<ul style="list-style-type: none"> • Planificación de Proyectos 	<ul style="list-style-type: none"> • Administración de Proyectos 	<ul style="list-style-type: none"> • Recursos Humanos
<ul style="list-style-type: none"> • Planificación de Proyectos 	<ul style="list-style-type: none"> • Administración de Proyectos 	<ul style="list-style-type: none"> • Comunicaciones
<ul style="list-style-type: none"> • Administración de Riesgos 	<ul style="list-style-type: none"> • Administración de Proyectos 	<ul style="list-style-type: none"> • Riesgos
<ul style="list-style-type: none"> • Administración de acuerdos con proveedores 	<ul style="list-style-type: none"> • Requerimientos 	<ul style="list-style-type: none"> • Adquisiciones

2.6 El Proceso APF

El proceso APF (Amisoft Process Framework) de desarrollo de software de Amisoft ha sido definido tomando en consideración las mejores prácticas de la industria, entre las cuales se encuentran las prácticas del modelo CMMI-DEV, la norma internacional ISO 9001:2000 y el Proceso de Referencia Tutelkan (TRP). El proceso APF incorpora además conceptos de RUP, tales como su enfoque iterativo incremental, su orientación a los casos de uso y sus fases de desarrollo, lo que permite describir claramente quién hace qué, cuándo y cómo en un proyecto de desarrollo o mantenimiento de software. De esta forma, al igual que RUP, el proceso APF está expresado en términos de roles (*quién* realiza la tarea), actividades (*cómo* ellos realizan estas tareas) y artefactos (*qué* se obtiene al realizar la actividad), que en su conjunto definen el proceso.

Dentro de la definición del proceso APF se encuentran los siguientes elementos:

Área de proceso: Corresponde a la definición formal de cómo realizar las actividades que involucra cada proyecto. Dentro de APF, las áreas se encuentran enmarcadas dentro de tres categorías: Procesos de gestión, Procesos de ingeniería y Procesos de soporte.

Actividades: Cada actividad corresponde a un conjunto de tareas que definen una parte específica del flujo de trabajo. Por ejemplo, el área de Planificación de proyecto está compuesta de dos actividades: Preparar la Propuesta y Planificar el Proyecto.

Tareas: Son pequeños trabajos que deben realizar personas específicas en una cierta instancia de tiempo para generar evidencia. Esta evidencia se materializa en los artefactos.

Artefactos: Los artefactos son productos de trabajo tangibles (documentos, planillas, presentaciones, componentes, entre otros) resultantes de cada tarea realizada.

Roles: Son los actores que ejecutan las tareas y, por lo tanto, los realizadores de cada artefacto resultante.

En la *Figura 6* se ejemplifica la relación entre los roles, las tareas y los artefactos, en este caso para el rol de analista de sistemas, del proceso APF.



Figura 6. Relación entre roles, tareas y artefactos

Áreas de Proceso

APF se compone de 11 áreas de proceso que se encuentran agrupadas en tres categorías: procesos de gestión, procesos de ingeniería y procesos de soporte.

Procesos de Gestión

Se encuentran enfocados directamente en el proyecto, cubriendo las necesidades relacionadas con la planificación, seguimiento y control del mismo.

Procesos de Ingeniería

Estos procesos dan soporte al ciclo de vida de desarrollo del producto. Abarcan desde la captura de requerimientos hasta el despliegue del producto en un ambiente productivo y totalmente operativo.

Procesos de Soporte

En esta categoría se proporcionan los procesos esenciales para soportar el desarrollo y mantenimiento del producto. Son transversales a todas las áreas de proceso definidas anteriormente, ya que éstas ocupan los resultados obtenidos por estos procesos.

2.7 Métricas para la toma de decisiones

De acuerdo a Pfleeger es necesario llevar a cabo el proceso de medición para entender qué ocurre durante el desarrollo y mantenimiento de software, controlar lo que ocurre en los proyectos y mejorar los procesos y productos [17]. Esta información es utilizada por distintos miembros de la organización para tomar mejores decisiones.

Ebert y Dumke plantean que las actividades de gestión de proyectos pueden ser agrupadas en seis categorías, en las cuales deberían existir métricas relacionadas. A estas actividades las denominan “Control del Proyecto”, y son: administración de requerimientos, estimación y costos, administración de recursos, planificación y calendarización, monitoreo y revisión, y control del producto. A partir de este conjunto de actividades se propone un conjunto de mediciones significativas que pueden servir de punto de referencia para la medición y análisis [6]. Algunos ejemplos de estas métricas se pueden ver en la *Tabla 3*.

Tabla 3. Métricas para Control del Proyecto

Tipo	Medición
Tamaño Proyecto	<ul style="list-style-type: none"> • Líneas de código • Medición funcional, por ejemplo puntos de función
Planificación	<ul style="list-style-type: none"> • Precisión (duración total / duración comprometida)
Esfuerzo	<ul style="list-style-type: none"> • Esfuerzo total proyecto • Costo para completar • Costo al completar
Productividad	<ul style="list-style-type: none"> • Costo por actividad comparado a lo estimado • Defectos detectados por caso de prueba
Progreso	<ul style="list-style-type: none"> • Estado de los requerimientos • Requerimientos completados • Progreso de las pruebas • Valor ganado
Calidad	<ul style="list-style-type: none"> • Efectividad en la detección de defectos • Fuga de defectos • Volatilidad de los requerimientos
Éxito del Negocio	<ul style="list-style-type: none"> • Beneficios alcanzados / Beneficios previstos • Ingreso alcanzados / Ingreso previstos

Putnam y Myers identifican cinco métricas que definen aspectos a medir para la gestión de proyectos [23]:

- *Tiempo planificado*
- *Esfuerzo*
- *Funcionalidad*
- *Tasa de defectos*
- *Productividad del proceso*

Daskalantonakis presenta la experiencia de la empresa Motorola, que implementó un programa de medición de software en el cual se definieron métricas para la mejora del proceso y el control del proyecto. Para derivar las métricas se utilizó la técnica

“Goal/Question/Metric”, tomando como punto de partida los objetivos de mejora identificados en la organización [5]. En la *Tabla 4* se resumen las métricas identificadas en Motorola.

De la misma manera, otros estudios han trabajado en definir un conjunto de métricas para modelos de proceso de software, que permitan discutir cómo estas métricas pueden ser usadas como indicadores de mantenibilidad [3].

Tabla 4. Métricas definidas en Motorola

Objetivo	Métrica
Mejorar planificación del Proyecto	<ul style="list-style-type: none"> • Precisión en la estimación del calendario • Precisión en la estimación del esfuerzo
Incrementar contención de defectos	<ul style="list-style-type: none"> • Efectividad en la detección de defectos totales • Efectividad en la detección de defectos por fase
Incrementar confiabilidad del software	<ul style="list-style-type: none"> • Tasa de fallos
Reducir densidad de defectos	<ul style="list-style-type: none"> • Fallas durante el proceso • Defectos durante el proceso • Total defectos liberados • Total defectos liberados en incremento • Total defectos detectados por el cliente • Total defectos detectados por el cliente en incremento
Mejorar servicio al cliente	<ul style="list-style-type: none"> • N° Nuevos problemas abiertos • N° Total problemas abiertos • Promedio tiempo problemas abiertos • Promedio tiempo problemas cerrados
Reducir costo de las no-conformidades	<ul style="list-style-type: none"> • Costo de arreglar problemas
Incrementar productividad	<ul style="list-style-type: none"> • Productividad total del software • Productividad del software en incremento

Ebert y Dumke, en un análisis para mejorar los procesos y los productos, proponen una serie de mediciones útiles para modelos de procesos que permitirían determinar la efectividad, eficiencia y rendimiento de estos. En particular, presenta resumidamente mediciones para cada una de las áreas de proceso del modelo de madurez CMMI [6].

En la *Tabla 5*, a modo de referencia, se presenta un resumen de las métricas propuestas por Ebert y Dumke para aquellas áreas de proceso de CMMI que coinciden con las definidas en el proceso APF de Amisoft, y que podrían ser útiles para una herramienta de gestión cuantitativa de proyectos.

Tabla 5. Mediciones para las áreas de proceso de CMMI

Área de Proceso	Medición
Monitoreo y Control del Proyecto (PMC)	<ul style="list-style-type: none"> • N° de proyectos que usan el proceso PMC. • Esfuerzo y costo en actividades de PMC. • N° de hitos alcanzados comparado con hitos planificados. • Valor Ganado. • Diferencia de lo actual respecto a lo planificado (por ejemplo: calidad, planificado, contenido, esfuerzo, costo).
Planificación del Proyecto (PP)	<ul style="list-style-type: none"> • N° de proyectos que usan el proceso PP. • Esfuerzo y costo en actividades de PP. • Precisión de la estimación. • Número de revisiones al plan inicial. • Porcentaje de distribución por razones de cambio en el plan.
Administración de Requerimientos (REQM)	<ul style="list-style-type: none"> • N° de proyectos que usan el proceso REQM. • Esfuerzo y costo en actividades de REQM. • Volatilidad de los requerimientos (porcentaje de requerimientos nuevos y cambios en requerimientos comparado con la línea base inicial del proyecto). • N° de requerimiento en un cierto estado (por ejemplo: elicitados, definidos, especificados, comprometidos, probados, entregados). • Tiempo de ciclo por requerimiento entre estados relevantes (por ejemplo, duración desde inicio hasta la entrega del requerimiento). • N° de puntos de traza por requerimiento (por ejemplo, requerimientos al plan de proyecto, a los casos de prueba).
Desarrollo de Requerimientos (RD)	<ul style="list-style-type: none"> • Grado de uso del proceso RD. • Esfuerzo, costo y beneficios medibles para las actividades de RD. • Esfuerzo, costo y tiempo gastado en re-trabajo de requerimientos. • Cambios por categoría de cambios por requerimiento. • Defectos por categoría por requerimiento. • N° de requerimientos asignados a un release comparado con requerimientos abiertos.
Solución Técnica (TS)	<ul style="list-style-type: none"> • Grado de uso del proceso TS. • Esfuerzo, costo y beneficios medibles para las actividades de TS. • Esfuerzo, costo y tiempo gastado en re-trabajo de diseño y desarrollo. • Tamaño y complejidad de los productos de trabajo (especialmente código). • Defectos por categoría por producto de trabajo.
Validación (VAL)	<ul style="list-style-type: none"> • Grado de uso del proceso de validación. • Esfuerzo, costo y beneficios medibles para las actividades de VAL. • N° de defectos detectados por actividad de validación. • Esfuerzo por defecto durante la validación (desde la detección hasta la corrección y la re-entrega). • Tiempo de espera promedio para corregir un defecto durante la validación. • Porcentaje de productos de trabajo probado comparado a

	lo planificado.
Verificación (VER)	<ul style="list-style-type: none"> • Grado de uso del proceso de verificación. • Esfuerzo, costo y beneficios medibles para las actividades de VER. • N° de defectos detectados por actividad de verificación. • Esfuerzo por defecto durante la verificación (desde la detección hasta la corrección y la re-entrega). • Tiempo de espera promedio para corregir un defecto durante la verificación. • Porcentaje de productos de trabajo revisados comparado a lo planificado.
Integración del Producto (PI)	<ul style="list-style-type: none"> • Grado de uso del proceso PI. • Esfuerzo, costo y beneficios medibles para las actividades de PI. • Porcentaje de componentes del producto integrados actualmente comparado a los componentes del producto diseñados. • Defectos encontrados durante la integración del producto.
Administración de la Configuración (CM)	<ul style="list-style-type: none"> • N° de proyectos que usan el proceso CM. • Esfuerzo y costo de las actividades de CM. • N° de cambios a una línea base inicial. • Costo de los cambios a la línea base inicial. • Cantidad de solicitudes de cambio procesadas por semana. • N° de cambios por categoría (por ejemplo: defectos, mal entendidos, entradas erróneas).
Medición y Análisis (MA)	<ul style="list-style-type: none"> • N° de proyectos que usan el proceso MA. • Esfuerzo y costo de las actividades de MA. • N° de proyectos que entregan las mediciones según lo solicitado. • Distribución porcentual de las causas de variación de los planes actuales contra el plan inicial.
Aseguramiento de Calidad de Procesos y Productos (PPQA)	<ul style="list-style-type: none"> • N° de proyectos que usan el proceso PPQA. • Esfuerzo y costo de las actividades de PPQA. • N° de productos y procesos auditados por proyecto. • Esfuerzo y costo gastado en re-trabajo. • N° de defectos por categoría. • N° de deficiencias por auditoría. • Porcentaje de deficiencias resueltas dentro del mismo proyecto. • N° de escalaciones.

2.8 Métricas de Amisoft

Conforme a su política de calidad, Amisoft ha establecido un plan de medición de la organización que permite determinar en forma cuantificable que se cumplan los objetivos de calidad. En el plan organizacional se define un conjunto de indicadores y métricas que son capturados periódicamente; de este conjunto de indicadores, sólo un pequeño conjunto está relacionado directamente con el proceso de desarrollo. Otros indicadores, relacionados a objetivos de calidad, están relacionados indirectamente, como por ejemplo

la satisfacción del cliente, ya que se puede deducir que el proceso mejora la calidad del producto, y que un producto de calidad aumenta la satisfacción del cliente.

Pese a la existencia de este plan de medición para tomar decisiones, aún no era suficiente para contar con información de sus procesos y de cuál es el desempeño de los proyectos de tal modo de permitir mejorar la toma de decisiones y acceder realmente a la mejora continua planteada por la organización.

En un afán por responder a estas necesidades de información la organización trabajó en la propuesta de un conjunto de métricas que dieran respuesta a las necesidades de información de la empresa, para lo cual presentó procesos para la definición, recolección, almacenamiento y análisis de las métricas [29]. Estas métricas fueron definidas considerando los estándares de la industria, abordados en la sección anterior. Estas métricas corresponden a:

- Adherencia al proceso
- Valor Ganado
- Pruebas funcionales
- Volatilidad de requerimientos

El detalle de las métricas se encuentra en el Anexo B del presente trabajo, donde son presentadas en mayor profundidad.

Dado el alcance del presente trabajo, no se abordó la definición de nuevas métricas, sino que se trabajó en base a las métricas ya definidas por la organización. Como se presentará más adelante, se contemplará en una primera versión de la herramienta la inclusión de las métricas *volatilidad de requerimientos* y *valor ganado* de acuerdo a las definiciones presentadas anteriormente, y abordadas en el trabajo de Vidal [29]. Esto permitirá analizar la capacidad de la herramienta de proveer métricas, que son utilizadas actualmente por la organización, en forma oportuna y medir su impacto real en comparación a lo existente hoy en día.

2.9 Herramientas de Gestión Cuantitativa de Proyectos

Según el Software Engineering Institute, el propósito de la gestión cuantitativa de proyectos es alcanzar los objetivos de calidad y rendimiento del proceso que han sido establecidos para el proyecto [25]. Para ello se plantean las siguientes actividades:

- Establecer y mantener los objetivos de calidad y rendimiento del proceso para el proyecto.

- Usar estadísticas y otras técnicas cuantitativas, para componer un proceso definido que permita al proyecto alcanzar sus objetivos de calidad y rendimiento del proceso.
- Seleccionar subprocesos y atributos críticos para evaluar el rendimiento y que ayuden al proyecto a alcanzar sus objetivos de calidad y rendimiento del proceso.
- Seleccionar medidas y técnicas analíticas para ser usadas en la administración cuantitativa.
- Monitorear el rendimiento de los subprocesos seleccionados usando estadísticas y otras técnicas cuantitativas.
- Administrar el proyecto usando estadísticas y otras técnicas cuantitativas para determinar si los objetivos del proyecto sobre calidad y rendimiento del proceso serán o no satisfechos.
- Ejecutar un análisis de causas para problemas seleccionados, que tienen que ver con deficiencias en el logro de los objetivos del proyecto, para calidad y rendimiento del proceso.

Según lo anterior, una herramienta de gestión cuantitativa de proyectos debe permitir la medición de los subprocesos seleccionados conforme el proceso es usado en el proyecto. La recolección de la información se debe realizar usando definiciones operacionales bien entendidas y reglas de recolección. Con la información obtenida en los proyectos anteriores, o con los problemas descubiertos en actividades previas, debe ser posible contestar preguntas acerca del rendimiento del proceso. Es decir, las medidas en la gestión cuantitativa tienen como objetivo fundamental estabilizar el proceso organizacional para que se obtengan los niveles de productividad y calidad esperados.

En el estudio realizado en el presente trabajo, se ha detectado la poca existencia de herramientas de gestión de proyectos que estén orientadas hacia la gestión cuantitativa. De acuerdo a Calvo-Manzano et al., en un acercamiento hacia la gestión cuantitativa de proyectos, la mayoría de las herramientas para la gestión de proyectos disponibles en el mercado están enfocadas principalmente en elementos de planificación y seguimiento de proyectos de software [2]. Ejemplo de este tipo de herramientas son Primavera², Open Plan Professional³, Microsoft Project⁴, Project.net⁵, entre otras. Sin embargo, estas herramientas de gestión de proyectos no han considerado otros aspectos tales como:

² <http://www.oracle.com/us/products/applications/primavera/index.html>

³ <http://www.delttek.com/products/ipm/openplan.aspx>

⁴ <http://www.microsoft.com/project/en-us/project-management.aspx>

- Funcionalidades para la administración de activos de procesos de la organización, como por ejemplo procesos estándares, modelos de ciclo de vida, criterios de adaptación de los procesos, medidas organizacionales, entre otras.
- Adaptación de los procesos para cada proyecto, a partir del conjunto de procesos estándares de la organización.
- Modelos cuantitativos que permitan predecir el comportamiento futuro de los proyectos, basándose en información histórica.

En general, estos aspectos no son considerados fundamentales en la gestión de proyectos, por lo que son considerados y resueltos en otras áreas de la ingeniería de software, tales como la mejora de procesos, la medición y estimación del software, entre otras.

Desde el punto de vista de la gestión cuantitativa, existen herramientas que proponen un marco de trabajo para la gestión cuantitativa de proyectos, que están orientadas a la mejora de procesos de software, pero que no consideran aspectos de planificación y seguimiento de los proyectos. Por ejemplo, Sejum propone una herramienta para la mejora de procesos aplicando Six Sigma en integración con el Personal Software Process (PSP) y el Team Software Process (TSP), basada en la existencia de un repositorio de información [26].

Considerando el proceso de desarrollo, Frappier y Richard proponen una herramienta de gestión para proyectos orientados por un proceso de software, que integra seis características principales: modelado de procesos, estimación de esfuerzo, planificación, seguimiento, control y mediciones [8].

La herramienta se basa en tres principios básicos:

- i. Un plan de proyecto debería partir de un modelo de proceso.
- ii. La estimación de esfuerzo debería basarse en datos históricos de los proyectos.
- iii. El control del proyecto debería basarse en unidades a las cuales se les puede hacer seguimiento a lo largo de las fases y actividades.

Esta propuesta permite descomponer el esfuerzo del proyecto en tareas administrables siguiendo el modelo de proceso, mejorando el cambio organizacional para adoptar y entender el proceso. Sin embargo carece de métricas que permitan la evolución del proceso.

Tomando en consideración lo expuesto anteriormente, y dada la relación existente entre el proceso de desarrollo de software y la construcción de productos de software de

⁵ <http://www.project.net/>

calidad, además de la importancia de medir y administrar el proceso software para la mejora continua del proceso [9], sería interesante contar con una herramienta integrada de gestión cuantitativa de proyectos orientada por un proceso que considerará los elementos mencionados previamente.

Capítulo III

3 Desarrollo de la Herramienta de Gestión de Proyectos

3.1 Introducción

Para realizar el desarrollo de la herramienta de gestión de proyectos, se ha decidido utilizar un enfoque iterativo incremental que permita dividir el trabajo en iteraciones, que definan los resultados esperados y los entregables al término de cada una de ellas, y que en su conjunto, constituyan micro-incrementos en la entrega del sistema en la medida que se cumplen los objetivos de las mismas.

Antes de iniciar el desarrollo, y tomando como referencia el marco teórico expuesto en el capítulo anterior, se propondrá y priorizarán las características de un sistema de gestión de proyectos y se describirá el dominio del problema. A partir de esto, se identificarán los casos de uso del sistema, los cuales dirigirán el proceso de desarrollo y permitirán definir las iteraciones del proyecto. Los casos de uso del sistema serán detallados y se realizará el análisis y diseño del sistema utilizando diagramas de secuencia y de clases antes de iniciar la implementación en el lenguaje de programación.

En este desarrollo se deben considerar los objetivos propuestos inicialmente. En particular, interesa que la solución cumpla con tres aspectos fundamentales, los cuales son:

- Permitir la gestión de proyectos de desarrollo de software
- Orientar la gestión de dichos proyectos mediante el proceso de desarrollo definido por la organización
- Permitir la obtención de métricas para facilitar el monitoreo y control de los proyectos, que permitan la toma de decisiones en forma oportuna

3.2 Lista de características

La primera actividad en el desarrollo del sistema corresponde a identificar las características y funcionalidades fundamentales en la gestión de proyectos. Las características propuestas en este trabajo han sido seleccionadas considerando principalmente las definiciones del proceso APF definido por Amisoft, que es utilizado por

la totalidad de los proyectos de desarrollo de software de la organización, además de aquellas que se han podido obtener a partir del estudio realizado en este trabajo.

En la *Tabla 6* se presentan las características que debería considerar un sistema de gestión de proyectos, que representan la base para el proceso de análisis, diseño e implementación del sistema. Las características propuestas han sido agrupadas en categorías para una mayor claridad.

Tabla 6. Descripción de las características propuestas para un sistema de gestión de proyectos

N°	Característica	Descripción
Planificación de proyectos		
1	Administrar proyecto	Permite realizar la generación de los proyectos y definir su alcance.
2	Administrar recursos	Permite registrar los recursos asignados al proyecto y definir la participación y roles de cada uno de ellos.
3	Establecer estimaciones	Permite registrar y estimar el tamaño y complejidad de los productos y actividades del proyecto.
4	Gestionar actividades	Permite definir y secuenciar las actividades del proyecto, además de asignar los recursos y duración de las actividades
5	Administrar involucrados	Permite registrar a los interesados (stakeholders) de un proyecto y registrar sus comunicaciones mediante minutas de reuniones.
6	Gestionar documentación	Permite realizar el registro, almacenamiento y disposición de la información del proyecto.
Monitoreo y control de proyectos		
7	Monitorear proyecto	Permite revisar el estado del proyecto derivado de los planificado versus lo real.
8	Registrar acciones correctivas	Permite registrar actividades o acciones en caso de desviaciones en la planificación.
Administración de requerimientos		
9	Administrar requerimientos	Permite realizar el registro de todos los requerimientos asociados al proyecto y otra información relevante para su posterior gestión.
10	Administrar solicitudes cambio	Permite registrar las solicitudes de cambios a los

		requerimientos.
11	Mantener trazabilidad	Permite registrar la trazabilidad de los requerimientos desde las necesidades de los usuarios hasta los componentes del producto.
Administración de proveedores		
12	Gestionar proveedores	Permite realizar el registro de productos o servicios que son adquiridos a través de proveedores para satisfacer los requerimientos del proyecto.
Administración de riesgos		
13	Administrar riesgos	Permite realizar el registro y mantención de los riesgos de un proyecto para una posterior evaluación de los mismos.
14	Mitigar riesgos	Permite el registro de actividades o acciones tendientes a mitigar un riesgo del proyecto, y los resultados de las mismas.
Administración proceso de desarrollo		
15	Gestionar proceso	Permite registrar y mantener actualizado el proceso de desarrollo definido por la organización para su uso en la gestión de los proyectos.
Administración del sistema		
16	Administrar usuarios	Permite al administrador del sistema y a los usuarios del mismo mantener actualizados sus datos y privilegios.
17	Login	Permite a los usuarios iniciar una sesión de trabajo en el sistema y acceder a las características que cada uno tenga habilitadas según su rol en los proyectos.

3.3 Priorización de las características

Para establecer una correcta planificación del proyecto, cada una de las características definidas anteriormente ha sido priorizada. Para ello se ha utilizado el método MoSCoW, descrito por el International Institute of Business Analysis, que permite decidir qué requerimientos deben ser implementados inicialmente y cuáles pueden incluirse en versiones posteriores, además de aquellos que no se realizarán [10].

La priorización de las características se realizó en conjunto con la organización, evaluando la relación e impacto de cada una de las características con respecto a las actividades definidas en el proceso APF y considerando las necesidades detectadas por la organización, en base a la experiencia obtenida en la gestión de proyectos. Dado que el proceso APF recoge las mejores prácticas del modelo CMMI-DEV, se dio especial importancia a las actividades de las áreas de proceso básicas para la administración de proyectos: planificación de proyectos, administración de requerimientos y monitoreo y control del proyecto.

En la *Tabla 7* se muestra las características priorizadas de acuerdo a los siguientes criterios:

M – Debe estar

S – Debería estar

C – Podría estar

W – No estará

Tabla 7. Priorización de las características

N°	Característica	M	S	C	W
1	Administrar proyecto	X			
2	Administrar recursos	X			
3	Establecer estimaciones			X	
4	Gestionar actividades	X			
5	Administrar involucrados				X
6	Gestionar documentación		X		
7	Monitorear proyecto	X			
8	Registrar acciones correctivas			X	
9	Administrar requerimientos	X			
10	Administrar solicitudes cambio		X		
11	Mantener trazabilidad		X		
12	Gestionar proveedores				X
13	Administrar riesgos			X	
14	Mitigar riesgos			X	
15	Gestionar proceso			X	
16	Administrar usuarios			X	
17	Login	X			

De acuerdo a esta priorización, y en virtud del alcance de este proyecto de tesis, se implementarán en la herramienta de gestión de proyectos las características que se han identificado como indispensables, que corresponde a aquellas marcadas con una “X” en columna M de la *Tabla 7*.

3.4 Modelo de Dominio

El modelo de dominio para la herramienta de gestión de proyectos se ha dividido, de acuerdo a la definición de características expuesto anteriormente, en cuatro categorías relacionadas. Éstas son:

- Planificación de proyectos
- Monitoreo y control de proyectos
- Administración de requerimientos
- Administración del sistema

Dentro del dominio de la planificación de proyectos, se incluirá brevemente el proceso de desarrollo. El proceso de desarrollo es importante porque la planificación del proyecto es guiada por un proceso, que en el caso de Amisoft corresponde al proceso APF.

En la Figura 7 se presentan los objetos del modelo de dominio del proceso de desarrollo. El objeto “Proceso” tiene una colección de “Fases” y “Actividades” que deben ser ejecutadas en un proyecto. Una “Fase” representa un periodo de tiempo en el proyecto, para el cual se establecen objetivos bien definidos (como por ejemplo definir el alcance del proyecto o especificar los requerimientos) los cuales son ejecutados a través de actividades. La “Actividad” representa un grupo de trabajo, el cual es llevado a cabo por un rol (individual o grupal), y que produce un producto que agrega valor al proyecto, estos son representados a través de las colecciones “Rol” y “Artefacto” respectivamente, las cuales son contenidas por el objeto “Actividad”.

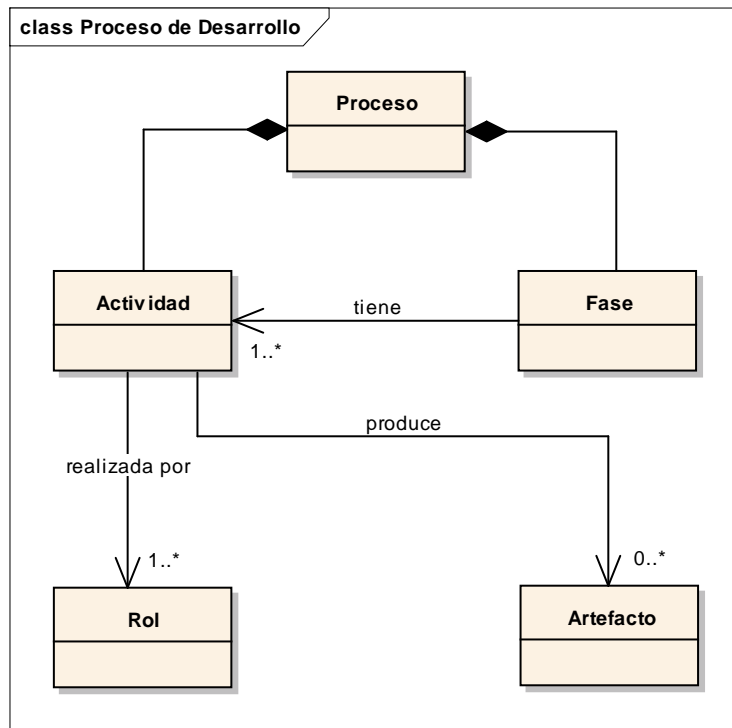


Figura 7. Modelo de domino – proceso de desarrollo

En la Figura 8 se muestra la relación entre la planificación del proyecto y el proceso de desarrollo utilizado para el proyecto. Cada proyecto de la organización, en base a su contexto, se basa en un proceso que guía su planificación de actividades. Esto está representado a través del objeto “Proyecto”, el cual está relacionado con el objeto “Proceso”. Además, el objeto “Proyecto” contiene una colección de “Iteraciones”, que a su vez contiene una colección de “Tareas”. Cada “Iteración” está relacionada a una “Fase” del proceso del cual se basa el proyecto. A su vez, cada “Iteración” contiene una colección de “Tareas”, las cuales están relacionadas a una “Actividad” del proceso. Esto significa que el “Proceso” relacionado al objeto “Proyecto” determina las “Fases” y “Actividades” relacionadas a las “Iteraciones” y “Tareas” respectivamente. El “Proyecto” contiene además una colección de “Personas”, donde cada “Persona” está relacionada a un “Rol” del “Proceso”. Las “Tareas” están relacionadas a las “Personas” a través de asignaciones.

De acuerdo a lo anterior, se puede resumir que un proyecto se divide en fases. A su vez, las fases son descompuestas en iteraciones, y cada iteración se compone de un conjunto de tareas acordes con las actividades de la fase. Por cada tarea de la iteración se debe estimar el esfuerzo para completarla junto a una fecha estimada de inicio y término de la

misma, el cual se deriva de la complejidad relativa de cada tarea. A cada una de estas tareas se le asignan los recursos necesarios para su realización.

Que el plan de proyecto esté basado en un proceso de desarrollo asegura que todos los proyectos de la organización estén estructurados de manera consistente, que todas las actividades sean consideradas, y que el monitoreo del proyecto y la obtención de datos para las métricas, sean también consistentes en todos los proyectos. Esto permitiría, a su vez, que la estimación de esfuerzo basada en información histórica sea más confiable y, facilitaría a su vez, la mejora del proceso basado en métricas.

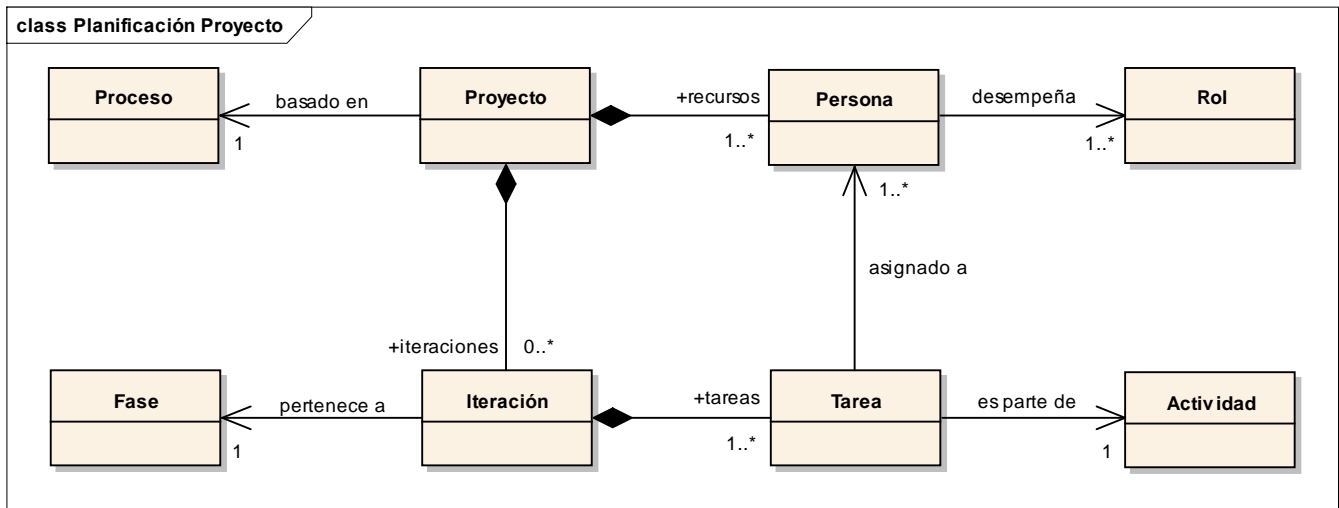


Figura 8. Modelo de domino – planificación del proyecto

En la medida que avanza la iteración, los recursos del proyecto informan diariamente el esfuerzo actual en cada tarea. Al fin de la semana, se puede determinar el progreso real para cada tarea y sus recursos. Como se ilustra en la Figura 9, cada objeto “Tarea” tiene una colección de objetos “RegistroTrabajo” los cuales representan el trabajo real realizado para ejecutar una tarea. Cada uno de estos registros de trabajo son reportados por un recurso del proyecto, representado mediante la asociación con el objeto “Persona”.

El control del proyecto permite medir el progreso real contra lo planificado y tomar acciones correctivas cuando sea necesario en orden de cumplir con los objetivos del proyecto en términos de esfuerzo y calendario. Normalmente, al comienzo de cada semana la iteración actual es evaluada, generalmente mediante una reunión de equipo utilizando métricas adecuadas. En particular, para realizar la evaluación de los proyectos,

Amisoft utiliza la métrica del valor ganado, analizando la variación de sus variables y los índices de desempeño SPI/CPI.



Figura 9. Modelo de dominio – monitoreo y control del proyecto

En el transcurso del proyecto se definen y administran los requerimientos del mismo. Como se muestra en la Figura 10, esto se representa a través del objeto “Proyecto” que tiene una colección de “Requerimiento”, el cual está asociado a una “Persona”, que representa al responsable del requerimiento. Así mismo, el “Proyecto” tiene una colección de “Release”, que representa las distintas versiones del software que es entregada al cliente. Cada una de estas versiones está asociada a uno o más objetos “Requerimiento”, que representan los distintos requerimientos que se están incluyendo en dicha versión.

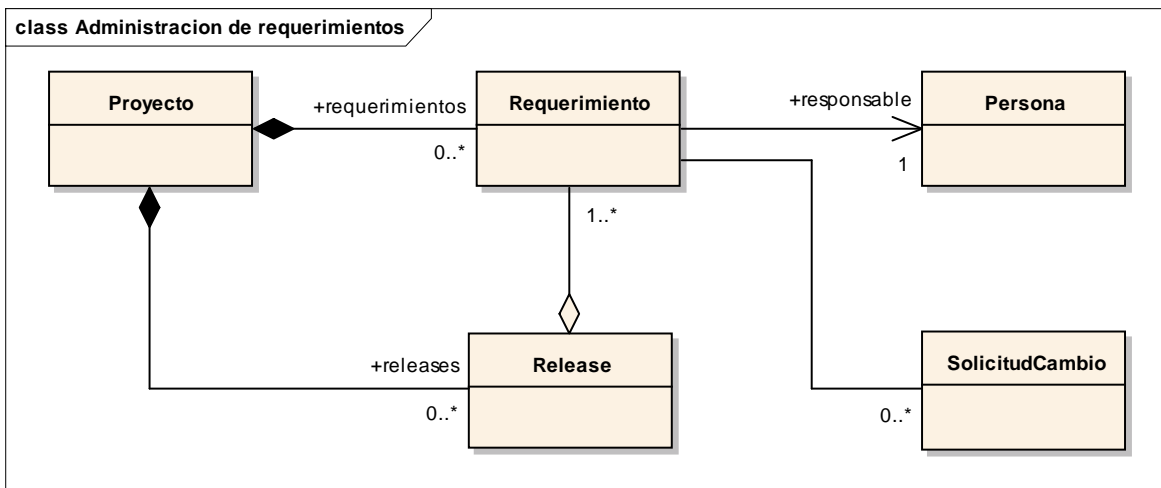


Figura 10. Modelo de dominio – administración de requerimientos

3.5 Arquitectura del sistema

Para la representación de la arquitectura se utilizará el modelo de vistas 4 + 1 sugerido por Kruchten, que propone un conjunto de cinco vistas para describir el sistema desde distintas perspectivas. Cada una de estas perspectivas se centra en un punto de vista particular del sistema, permitiendo descubrir los elementos relevantes y sus relaciones,

así como a su vez, omitir aquellos elementos que no son importantes [14]. Las vistas sugeridas son:

- **Vista lógica**, la cual se enfoca en los requerimientos funcionales del sistema, siendo una abstracción del modelo de diseño, identificando los paquetes de diseño principales, subsistemas y clases.
- **Vista de implementación**, que describe la organización de los módulos estáticos del software en términos de empaquetamientos y capas, y en términos de administración de la configuración.
- **Vista de procesos**, enfocado en los aspectos concurrentes del sistema, tales como tareas, hilos o procesos y sus interacciones.
- **Vista de despliegue**, muestra como los componentes del sistema son mapeados en las distintas plataformas y nodos computacionales.
- **Vista de casos de uso**, contiene los casos de uso claves, que guían el diseño de la arquitectura en las primeras fases de desarrollo y que permiten validar las diversas vistas de la arquitectura.

Cada una de estas vistas puede ser representada en diferentes niveles de abstracción utilizando distintos diagramas según las necesidades y complejidad de cada sistema.

Si bien este modelo no es una metodología, se sugiere guiar su desarrollo en forma incremental a partir de la vista de los casos de uso. Con la definición de los casos de uso principales se puede definir la vista lógica, y con esta, la vista de implementación. Dependiendo de la complejidad del sistema, se puede definir la vista de procesos, para finalmente concluir con la vista de despliegue.

Vista de casos de uso

Esta vista ha sido representada mediante el modelo de casos de uso. Un caso de uso representa la interacción de un actor con el sistema para llevar a cabo una unidad de trabajo significativa.

De acuerdo a las características y el modelo de dominio del problema, se ha concluido que las interacciones ocurren entre los distintos actores y el sistema de gestión de proyectos sin existir otros sistemas involucrados. Estas interacciones se producen mediante una interfaz de usuario, que correspondería al sistema de gestión de proyecto.

En la Figura 11, se muestra como dos casos de uso implementan parte de una de las características del sistema correspondiente a “administrar proyecto”. La interacción ocurre entre el actor “jefe de proyecto” y los casos de uso “crear proyecto” y “configurar

proyecto”. Esta vista guía la elaboración de la arquitectura porque permite trazar y validar la relación de los casos de uso con los distintos componentes de la arquitectura. Por ejemplo, muestra cómo los componentes “active record” y “action view” implementan parte de la funcionalidad de crear un proyecto, y cómo, a su vez, estos están relacionados con el objeto “proyecto” y la interfaz de usuario que corresponde a una página web.

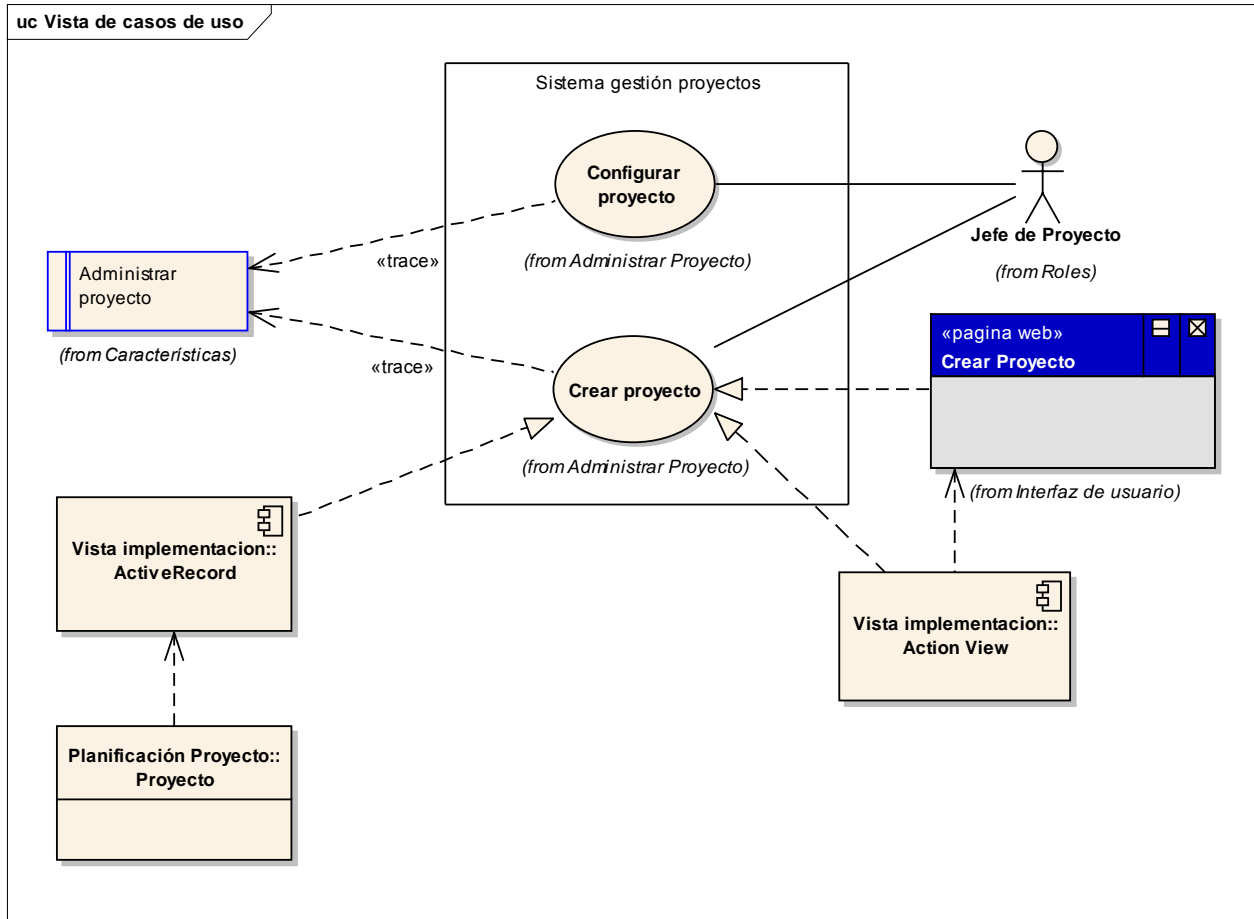


Figura 11. Vista de casos de uso y sus relaciones

Vista lógica

Esta vista se enfoca en la realización de las funcionalidades de la aplicación en términos de sus elementos estructurales. Para ello, ha sido representada mediante una división horizontal, en una arquitectura por capas, que permite distribuir las responsabilidades en cada una de estas capas.

A continuación se describe cada una de las capas mostradas en la Figura 12.

- **Presentación.** Esta capa contiene los componentes que implementan y muestran la interfaz de usuario y controlan las interacciones del mismo con la aplicación, así como también componentes para organizar la interacción y controlar la entradas de datos de

los usuarios. Esta capa de presentación se separa en *Vistas*, que corresponden a los elementos visuales de la aplicación utilizados para desplegar información y aceptar entradas de los usuarios, y *Controladores*, que definen el comportamiento lógico y estructural de la aplicación .

- **Negocio.** Contiene los componentes que implementan la lógica y reglas de negocio. Es la encargada de obtener, procesar, transformar y gestionar la información de la aplicación, además de asegurar y validar la consistencia de los datos. En general son unidades básicas de diseño, implementación, despliegue, mantenimiento y gestión del ciclo de vida de la aplicación.

Se pueden identificar dos categorías:

- Componentes de negocio: Encargados en encapsular la lógica de negocio y los datos necesarios para representar los elementos del mundo real en la aplicación, como por ejemplo un Usuario o un Proyecto. Además contiene y gestiona los datos de negocio que utiliza la aplicación; y facilita el acceso a los datos y las funcionalidades relacionadas.
- Flujos de negocio: Encargados de procesar los datos de los componentes del negocio para llevar a cabo procesos de negocio.

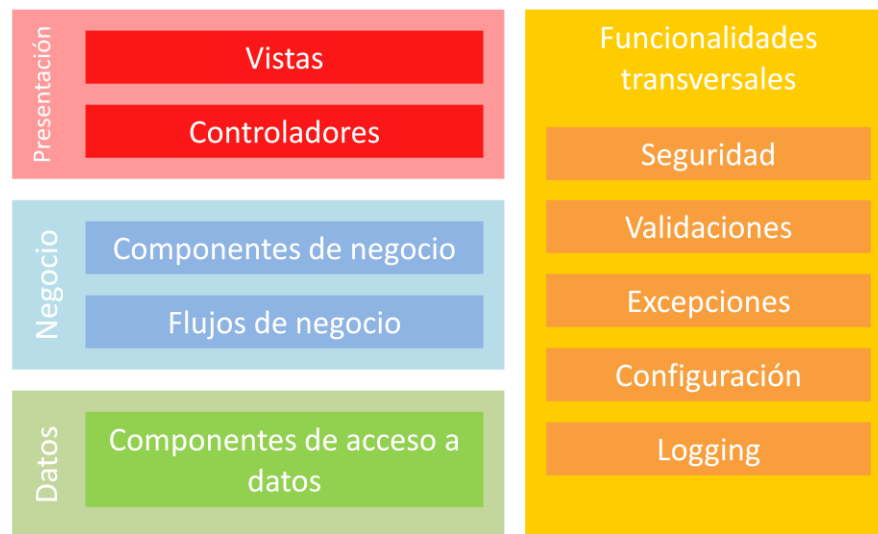


Figura 12. Vista Lógica

- **Datos.** Esta capa contiene componentes que implementan la lógica necesarios para acceder a los datos almacenados persistentemente, centralizando las funcionalidades

de acceso a los datos con el fin de hacer la aplicación más fácil de configurar y mantener. Existen *frameworks* de acceso a datos, incluyendo muchos Object/Relational Mapping (ORM), que ya implementan dichos componentes de forma automática, lo que reduce la cantidad de código de acceso a datos que se debe escribir.

- **Funcionalidades transversales.** Esta capa corresponde a funcionalidades comunes que atraviesan las distintas capas de la aplicación. Estas funcionalidades generalmente incluyen acciones como la autenticación, autorización, almacenamiento en caché, manejo de excepciones, configuración y validación. Estas funcionalidades son transversales, ya que afecta a toda la aplicación, y están centralizadas en un solo lugar.

Vista de implementación

Esta vista comprende los componentes utilizados para implementar y entregar el sistema físico enfocándose en la organización del software. Mientras la vista lógica descrita anteriormente se presenta a un nivel conceptual, esta vista muestra los artefactos a un nivel físico.

La aplicación de gestión de proyectos será implementada utilizando el *framework* de desarrollo de aplicaciones web *Rails*, escrito en el lenguaje de programación *Ruby*, el cual se basa en el patrón Modelo Vista Controlador (MVC).

En la Figura 13 se presentan los componentes principales de la vista de implementación, los cuales se describen a continuación:

- **Action Dispatch:** Componente encargado de manejar el enrutamiento de las solicitudes realizadas desde el cliente. Determina qué controlador debe manejar la solicitud, así como la lista de parámetros asociados.
- **Action Controller:** Este componente provee las funcionalidades para acceder a los datos necesarios desde el negocio, y controlar la presentación y re direccionamiento de las vistas. Además, permite gestionar las sesiones de usuario, el flujo de la aplicación, las características de almacenamiento en caché, entre otras opciones. Provee un controlador base desde el cual todos los controladores de la aplicación pueden heredar. Este componente corresponde al controlador de la capa de presentación.

- **Action View:** Este componente es el encargado de proveer todas las funcionalidades para presentar la información solicitada, generalmente generando código HTML, XML o Javascript hacia el cliente. Como su nombre lo indica, este componente corresponde a la vista de la capa de presentación.
- **Active Record:** Este componente es el encargado de representar los datos y la lógica del negocio, facilitando la creación y uso de los objetos del negocio cuyos datos necesitan ser almacenados persistentemente en la base de datos. Es una implementación del patrón Active Record a través de un sistema ORM [7]. Permite, entre otras cosas, representar los modelos y sus datos, representar las asociaciones entre estos modelos, así como jerarquías y herencias entre estos. Además permite validar los modelos antes de hacerlos persistentes en la base de datos y realizar operaciones en la base de datos en un estilo orientado a objetos sin la necesidad de escribir sentencias SQL. El Active Record corresponde a los componentes de negocio y acceso a datos de las capas de negocio y datos respectivamente.

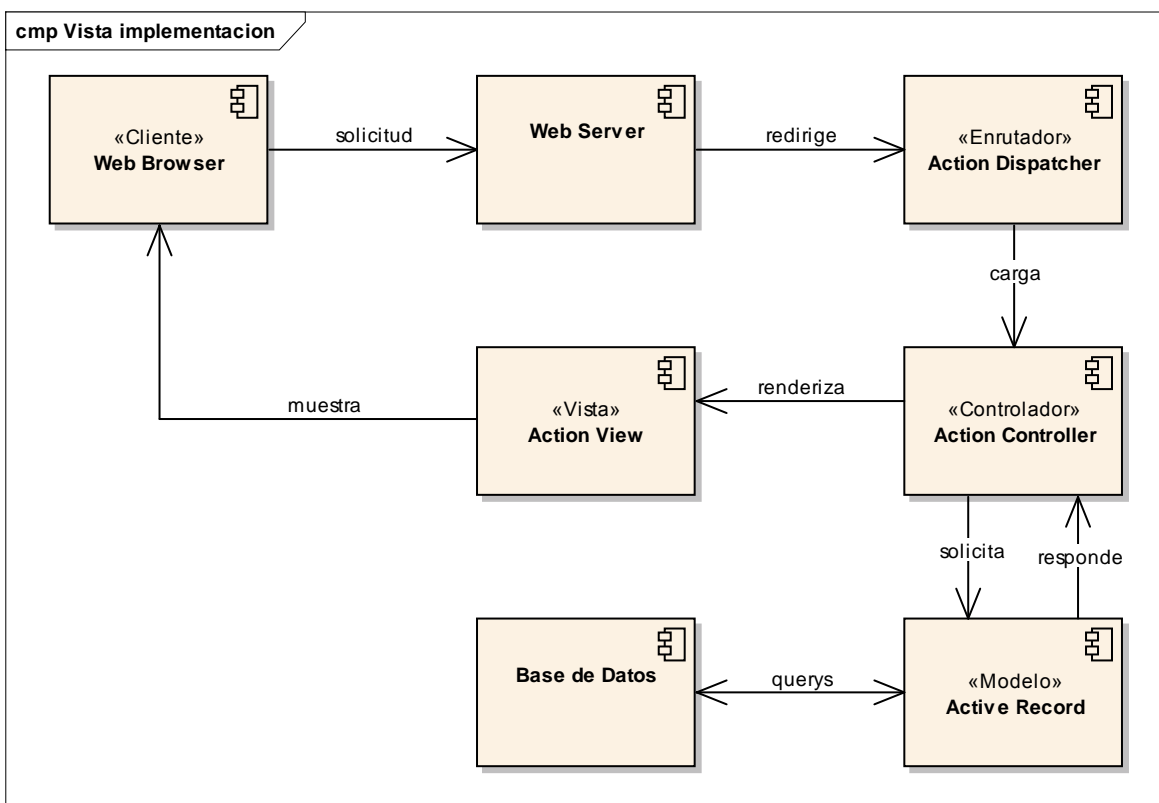


Figura 13. Vista de implementación

Vista de despliegue

La vista de despliegue representa la estructura física necesaria para la implementación del sistema, abarcando los nodos que forman la topología de hardware del sistema.

La Figura 14 muestra la configuración de los nodos en que se ejecuta el sistema; ésta se compone de los siguientes nodos:

- Cliente: PC con Web Browser (Internet Explorer, Google Chrome y FireFox u otro) desde donde se realizan las solicitudes.
- Servidor Web: El Servidor de Aplicaciones contiene los componentes de presentación (IU) y las capas de negocio y acceso a datos. Se utilizará una configuración NGINX y Unicorn, donde se configurará la aplicación Ruby on Rails de gestión de proyectos.
- Servidor de Base de Datos: Encargada de almacenar los datos en forma persistente. El motor de base de datos que se utilizará es PostgreSQL.

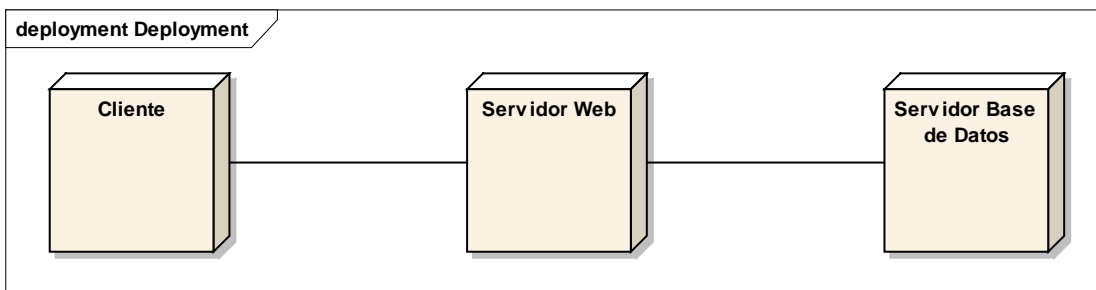


Figura 14. Vista de despliegue

3.6 Modelo de Casos de Uso del Sistema

Conforme a la descripción realizada anteriormente, se presentarán los actores del sistema y la estructura del modelo de casos de uso.

Actores

Existen dos tipos de actores principales, los cuales son especializaciones del rol usuario, que interactúan en el sistema de gestión de proyectos, los cuales se muestran en la Figura 15, y que son descritos a continuación.

Administrador de las métricas. Es el encargado de acceder a las métricas de la organización y obtener reportes acerca de la *performance* de los distintos proyectos.

Recurso. Corresponde a usuarios que son participantes de un proyecto. El sistema les provee el acceso a toda la información relevante del proyecto al cual pertenecen. Una especialización en este rol es el jefe de proyecto, quién es el encargado de mantener y gestionar el proyecto.

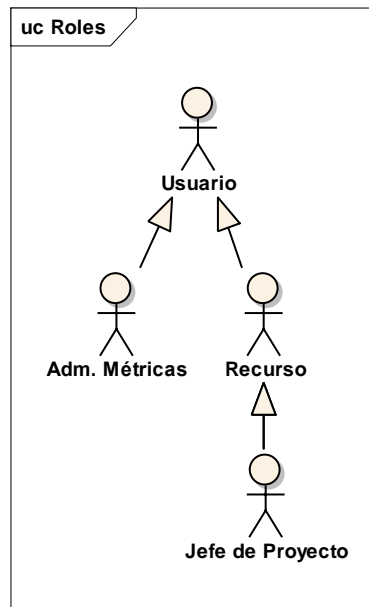


Figura 15. Actores del sistema

Estructura

A continuación, se presentan los casos de uso identificados, las relaciones entre ellos y la participación de los actores en estos. En la Figura 16 se presentan los casos de uso, que han sido organizados en paquetes de acuerdo a las características identificadas anteriormente.

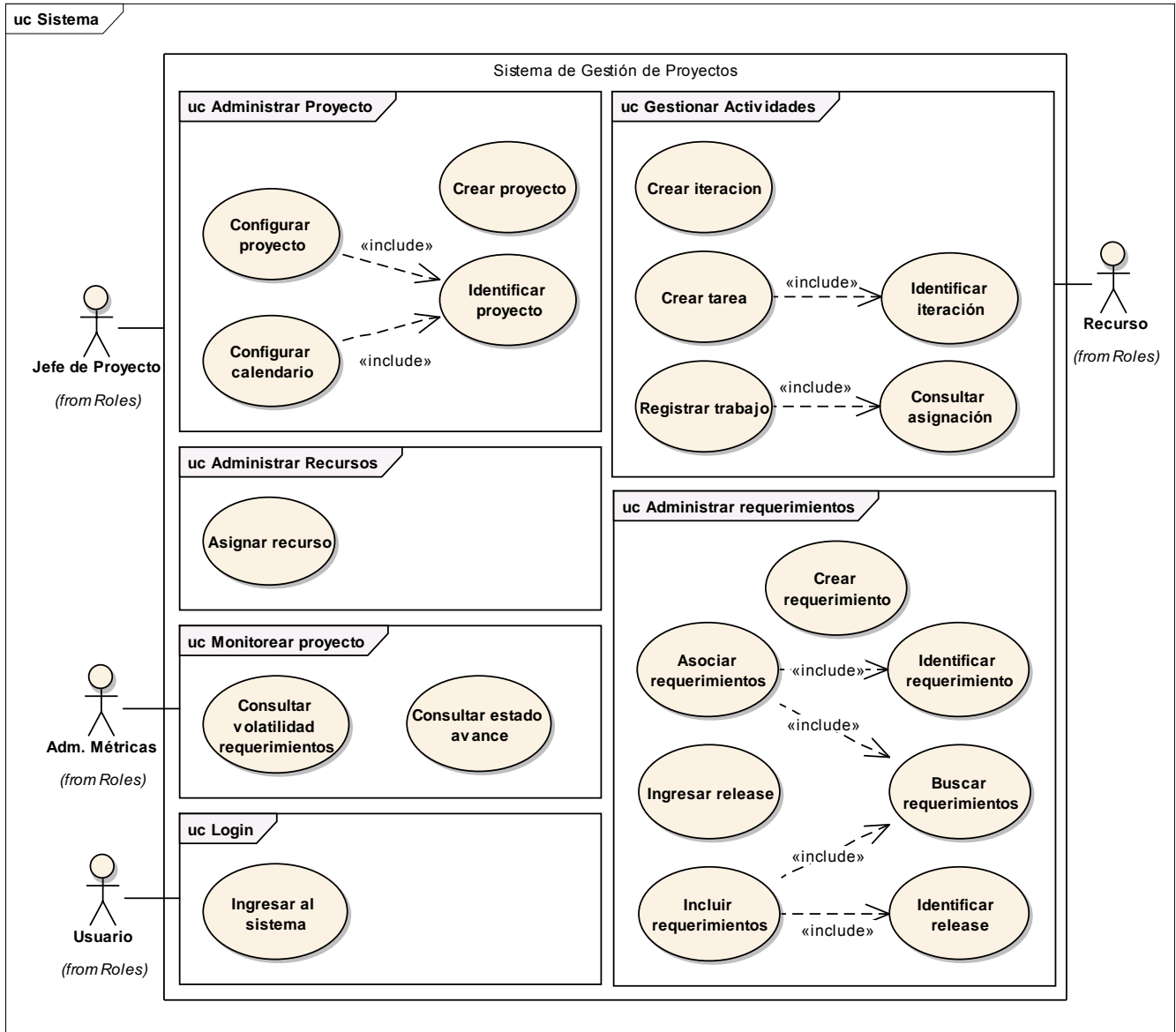


Figura 16. Diagrama de Casos de Uso del Sistema

Para la implementación de los casos de uso identificados, se han definido tres iteraciones en las cuales se han implementado los siguientes casos de uso:

Iteración 1	Iteración 2	Iteración 3
Crear proyecto	Crear iteración	Crear requerimiento
Identificar proyecto	Identificar iteración	Identificar requerimiento
Configurar proyecto	Crear tarea	Buscar requerimientos
Configurar calendario	Consultar asignación	Asociar requerimiento
Asignar recurso	Registrar trabajo	Ingresar release

Ingresar al sistema	Consultar estado de avance	Identificar release
		Incluir requerimientos
		Consultar volatilidad de requerimientos

3.7 Primera Iteración

En la primera iteración se abordaron las funcionalidades que permiten la administración de un proyecto y sus recursos, además de la funcionalidad que permite ingresar al sistema. Estos casos de uso presentan la base de lo que es la administración del proyecto y sus recursos.

La configuración del proyecto y el calendario constituyen los parámetros de referencia para realizar la definición de las actividades del proyecto y los cálculos de las métricas de valor ganado, ya que a partir de ellas se podrá calcular el presupuesto del proyecto.

Nombre	Crear proyecto
Actores	Jefe de Proyecto
Sinopsis	Este caso de uso comienza cuando el jefe de proyecto solicita crear un nuevo proyecto en el sistema. Se solicitan los datos y, en caso de éxito, se registra y confirma el nuevo proyecto.
Escenario típico	<ol style="list-style-type: none"> 1. Incluir "Ingresar al sistema" 2. El sistema lista las categorías para los tipos de proyecto disponibles 3. El sistema lista los ciclos de vida disponibles 4. El usuario ingresa los datos del proyecto. Los datos disponibles son: <ul style="list-style-type: none"> • Nombre • Descripción • Fecha de inicio • Fecha de término • Cliente • Categoría • Ciclo de Vida 5. El sistema registra el proyecto. 6. El sistema despliega mensaje indicando que proyecto fue creado exitosamente y muestra los datos del proyecto.
Precondición	El usuario debe tener una sesión válida con el perfil de jefe de proyecto
Escenarios alternativos	<ol style="list-style-type: none"> 5a. El usuario no ha ingresado todos los datos del proyecto <ol style="list-style-type: none"> 1. El sistema notifica el error indicando los datos pendientes 2. El usuario retoma en punto 4.

En la Figura 17 se presenta el diagrama de secuencia del curso de eventos para el caso de uso *crear proyecto*.

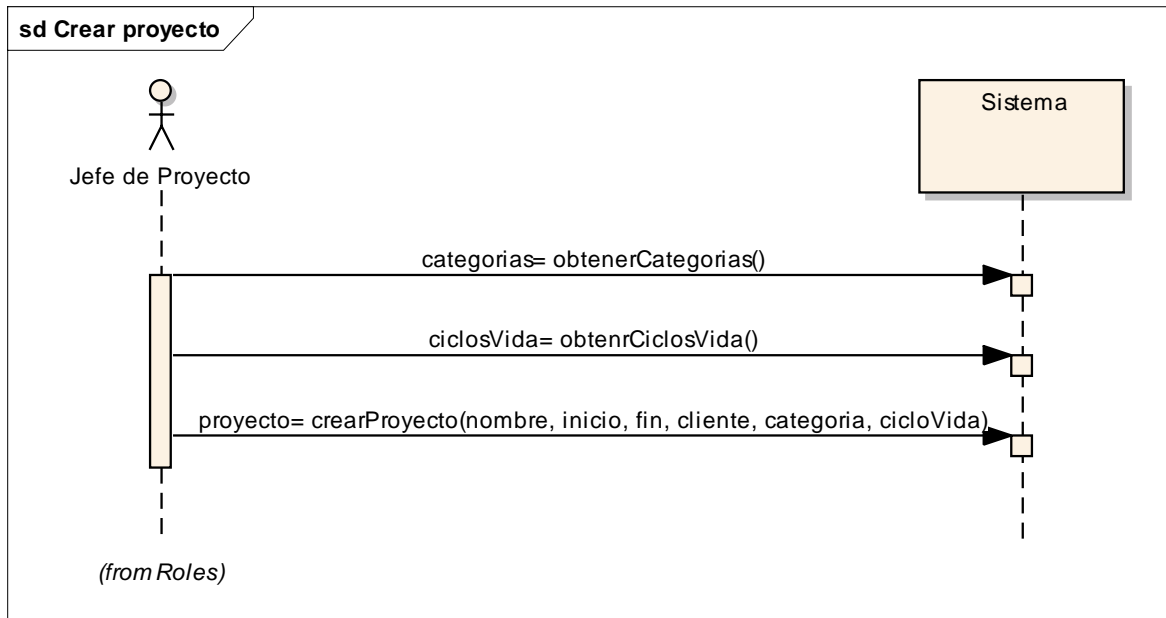


Figura 17. Diagrama de secuencia del caso de uso Crear proyecto

Nombre	Configurar Proyecto
Actores	Jefe de Proyecto
Sinopsis	Este caso de uso comienza cuando el jefe de proyecto solicita configurar los parámetros del proyecto. El usuario indica los días laborables de la semana y las horas de trabajo diarias. El sistema confirma la configuración del proyecto.

Nombre	Configurar calendario
Actores	Jefe de Proyecto
Sinopsis	Este caso de uso comienza cuando el jefe de proyecto solicita registrar un nuevo día no laborable para el proyecto. El sistema lista los días no laborables registrados actualmente en el proyecto. Si no hay conflicto en las fechas, el sistema ingresa el nuevo día no laborable y actualiza la lista de días no laborables.

Nombre	Identificar proyecto
Actores	Jefe de Proyecto
Sinopsis	Este caso de uso comienza cuando el jefe de proyecto solicita acceder a un proyecto activo del sistema.
Escenario	1. El sistema muestra los proyectos activos del sistema

típico	2. El usuario selecciona un proyecto de la lista 3. El sistema muestra el proyecto seleccionado
Escenarios alternativos	1a. No existen proyectos activos 1. El sistema muestra mensaje indicando que no existen proyectos 2a. Proyecto buscado no aparece en la lista

En la Figura 18 se presenta el diagrama de secuencia del caso de uso *Identificar proyecto*. El sistema realizará la búsqueda entre aquellos proyectos en los cuales el usuario está participando.

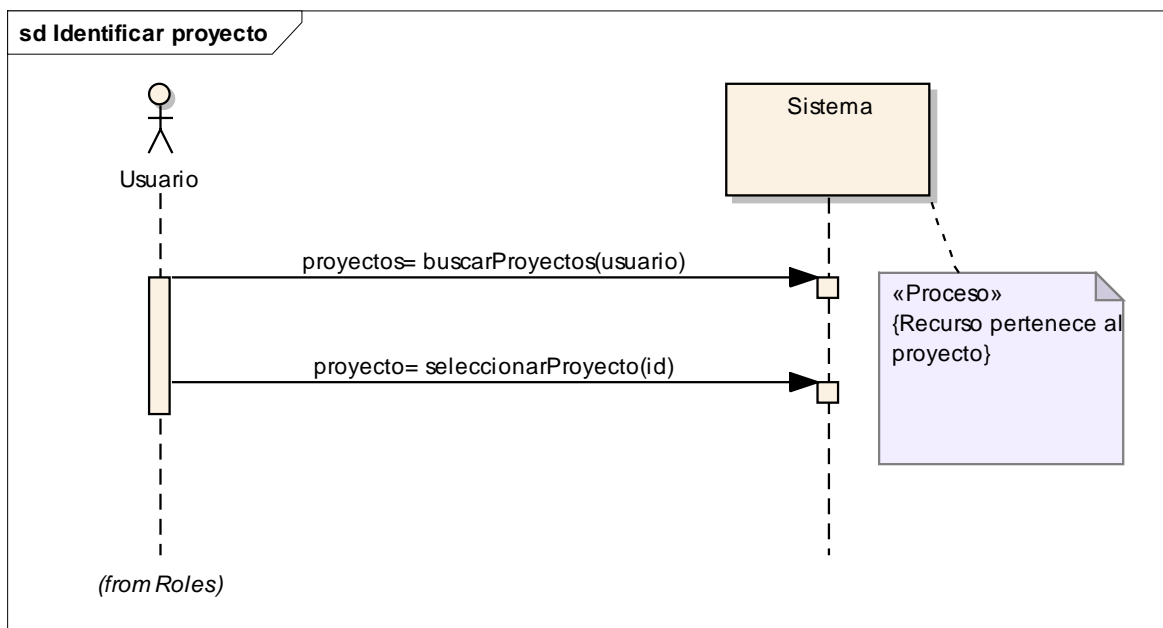


Figura 18. Diagrama de secuencia del caso de uso *Identificar proyecto*

Nombre	Asignar recurso
Actores	Jefe de Proyecto
Sinopsis	Este caso de uso comienza cuando el jefe de proyecto solicita agregar un recurso al proyecto.
Escenario típico	<ol style="list-style-type: none"> 1. Incluir "Identificar proyecto" 2. El sistema lista los roles disponibles. 3. El usuario ingresa los datos del recurso. Los datos disponibles son: <ul style="list-style-type: none"> ○ Email ○ Porcentaje de dedicación ○ Roles 4. El sistema registra el recurso 5. El sistema despliega mensaje indicando que el recurso fue agregado exitosamente y muestra los datos del recurso.

Escenarios alternativos	6. El sistema notifica al nuevo recurso vía email que ha sido agregado al proyecto.
	4a. El usuario no ha ingresado todos los datos del recurso <ul style="list-style-type: none"> 1. El sistema notifica el error indicando los datos pendientes 2. El usuario retoma en punto 3.

En la Figura 19 se presenta el diagrama de secuencia del caso de uso *asignar recurso*. Al momento de asignar un nuevo recurso al proyecto se le envía una notificación a la persona mediante un correo electrónico.

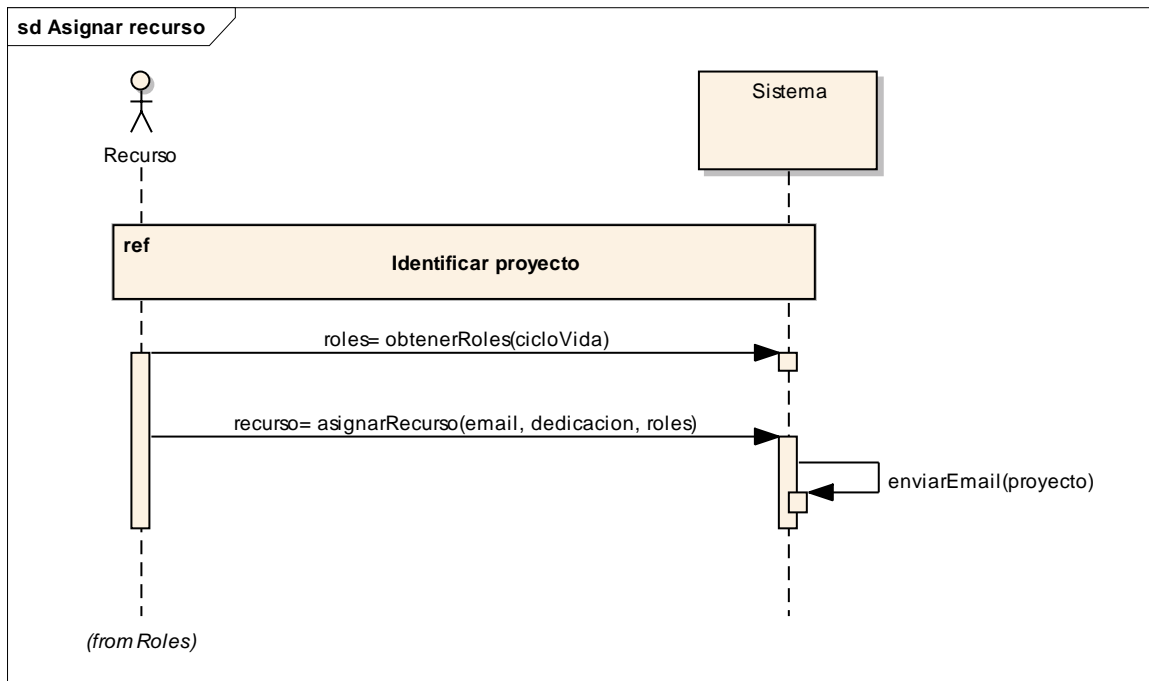


Figura 19. Diagrama de secuencia del caso de uso *Asignar recurso*

Nombre	Ingresar al sistema
Actores	Usuario
Sinopsis	Este caso de uso permite identificar al actor como un usuario del sistema registrado mediante la comprobación de un nombre de usuario y password.

Diagrama de clases

El diagrama de clases permite ver una vista estática del modelo, o parte del modelo, que describe los atributos y el comportamiento que este tiene. Este diagrama de clases es útil para ilustrar las relaciones entre las clases.

En la Figura 20 se presentan las clases identificadas en esta primera iteración, en donde las clases principales corresponden a “Proyecto” y “Persona”, las cuales se conectan a través de la clase “Participante” que define los recursos de un proyecto. Un “Proyecto” además se asocia a una instancia de la clase “Ciclo de Vida”, que permite, al momento de definir un participante, seleccionar uno o más roles de acuerdo al ciclo de vida seleccionado. Además se muestran las relaciones con las clases “Calendario” y “DiasNoLaborables” que permiten realizar la configuración del proyecto.

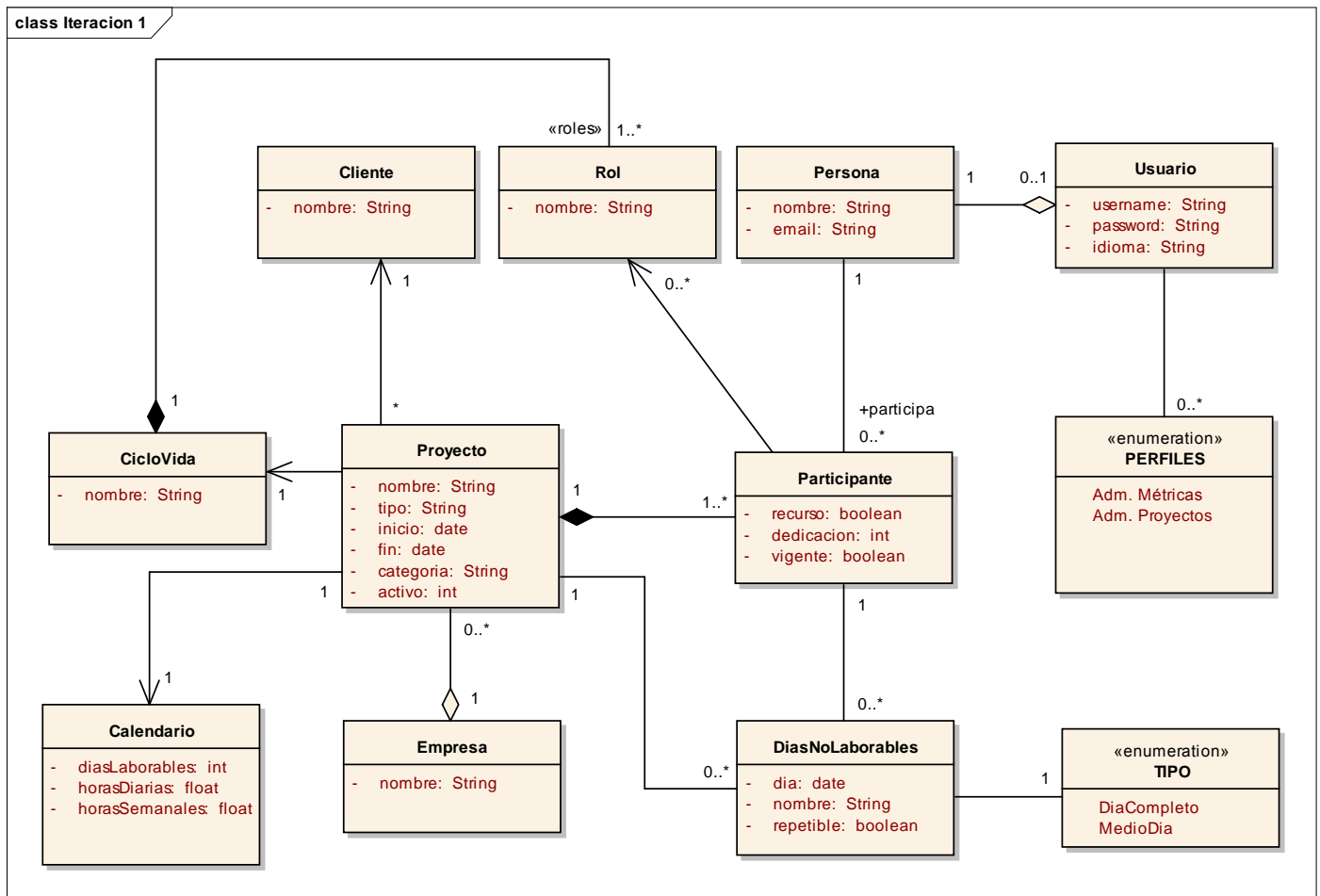


Figura 20. Diagrama de clases Iteración 1.

Implementación

La implementación del sistema de gestión de proyectos consistió en traducir el diseño realizado al código del lenguaje de programación Ruby. El entorno de desarrollo utilizado es el framework Ruby on Rails y la herramienta de desarrollo es Sublime Text 2.

En esta primera iteración se inició la creación de la aplicación utilizando un script provisto por la aplicación Ruby on Rails que genera la estructura de directorios y archivos inicial. A partir de aquí, se trabajó en la implementación de los casos de uso.

En la Figura 21 se presenta la implementación del caso de uso *crear proyecto*, en la cual se puede ingresar la información del nuevo proyecto y se destaca la selección del ciclo de vida que este proyecto utilizará.

Nuevo Proyecto

Nombre SITCORTE

Descripción Sistema de tramitación para las Cortes de Apelaciones

Inicio 02-01-2012

Fin

Cliente PJUD

Categoría Mantención

Ciclo de Vida APF

Guardar

Figura 21. Página web implementación crear proyecto

En la Figura 22 se presenta la implementación del caso de uso asignar recurso el cual permite agregar una nueva persona al proyecto e indicar su dedicación y roles en el mismo. Los roles disponibles dependen del ciclo de vida seleccionado para el proyecto. Por otro lado, la dedicación es importante para calcular las métricas de valor ganado, ya que, en base a la configuración del proyecto y la dedicación de los recursos, se pueden calcular, entre otras cosas, el presupuesto en horas hombre del proyecto.

SITCORTE Documentos Requerimientos Iteraciones Formularios Admin

Nuevo Participante

Email Persona Recurso

Dedicación %

Roles

- Administrador de la configuración
- Analista de sistemas
- Analista PPQA**
- Arquitecto de Software
- Jefe de proyecto
- Programador
- Tester
- Analista QA
- Analista de métricas e infraestructura
- Gerente de proyectos

Participantes Actuales

Andres Molina (3%)
Último ingreso: miércoles 14 de mayo de 2014 11:52

Victor Herrera Robledo (100%)
Último ingreso: lunes 12 de mayo de 2014 18:07

Dayana Nieto
Último ingreso: jueves 08 de mayo de 2014 12:04

Guillermo Sandoval
Último ingreso: miércoles 07 de mayo de 2014 14:10

Daniel Ramirez (100%)
Último ingreso: miércoles 14 de mayo de 2014 13:55

Daniela Quiroz
Último ingreso: lunes 10 de febrero de 2014 13:40

Participantes anteriores

Victor Solari Orellana (50%)
Último ingreso: martes 11 de marzo de 2014 10:42

sujej Lopez (5%)
Último ingreso: martes 06 de mayo de 2014 17:38

Figura 22. Página web implementación asignar recurso

3.8 Segunda Iteración

En esta segunda iteración se realizó el diseño e implementación del conjunto de casos de uso que permiten gestionar las actividades del proyecto, así como monitorear su estado de avance. Además se diseñó e implementó la primera métrica del sistema correspondiente a la métrica de valor ganado.

Nombre	Crear iteración
Actores	Jefe de proyecto
Sinopsis	Este caso de uso comienza cuando el jefe de proyecto solicita crear una nueva iteración en el proyecto actual. Se solicitan los datos y, en caso de éxito, se registra y confirma la nueva iteración.

Nombre	Identificar iteración
Actores	Jefe de proyecto
Sinopsis	Este caso de uso permite al usuario seleccionar, desde una lista de iteraciones, una iteración activa del proyecto. El sistema retorna y muestra la iteración seleccionada.

Nombre	Crear tarea
Actores	Jefe de proyecto
Sinopsis	Permite a un recurso listar todas las tareas asignadas a su persona para cada uno de los proyectos en los cuales participa.
Escenario típico	<ol style="list-style-type: none"> 1. Incluir "Identificar iteración" 2. El sistema lista los recursos actuales del proyecto 3. El usuario ingresa los datos de la tarea. Los datos son: <ul style="list-style-type: none"> ○ Nombre ○ Fecha de inicio ○ Duración en horas 4. El sistema sugiere una fecha de término para la tarea 5. El usuario selecciona los recursos que se le asignan a la tarea 6. El sistema registra la tarea 7. El sistema actualiza las métricas del sistema 8. El sistema despliega mensaje indicando que la tarea fue creada exitosamente y muestra los datos de la tarea.
Escenarios alternativos	<ol style="list-style-type: none"> 6a. El usuario no ha ingresado todos los datos de la tarea <ol style="list-style-type: none"> 1. El sistema notifica el error indicando los datos pendientes 2. El usuario retoma en punto 3.

En la Figura 23 se presenta el diagrama de secuencia del caso de uso *crear tarea*. Cuando el jefe de proyecto ingresa la duración y fecha de inicio de una tarea, el sistema propone, a modo de facilitar esta actividad, una fecha de término de la tarea considerando la configuración del proyecto. Al momento de crear la nueva tarea, el sistema actualiza las métricas del proyecto, actualizando las horas planificadas.

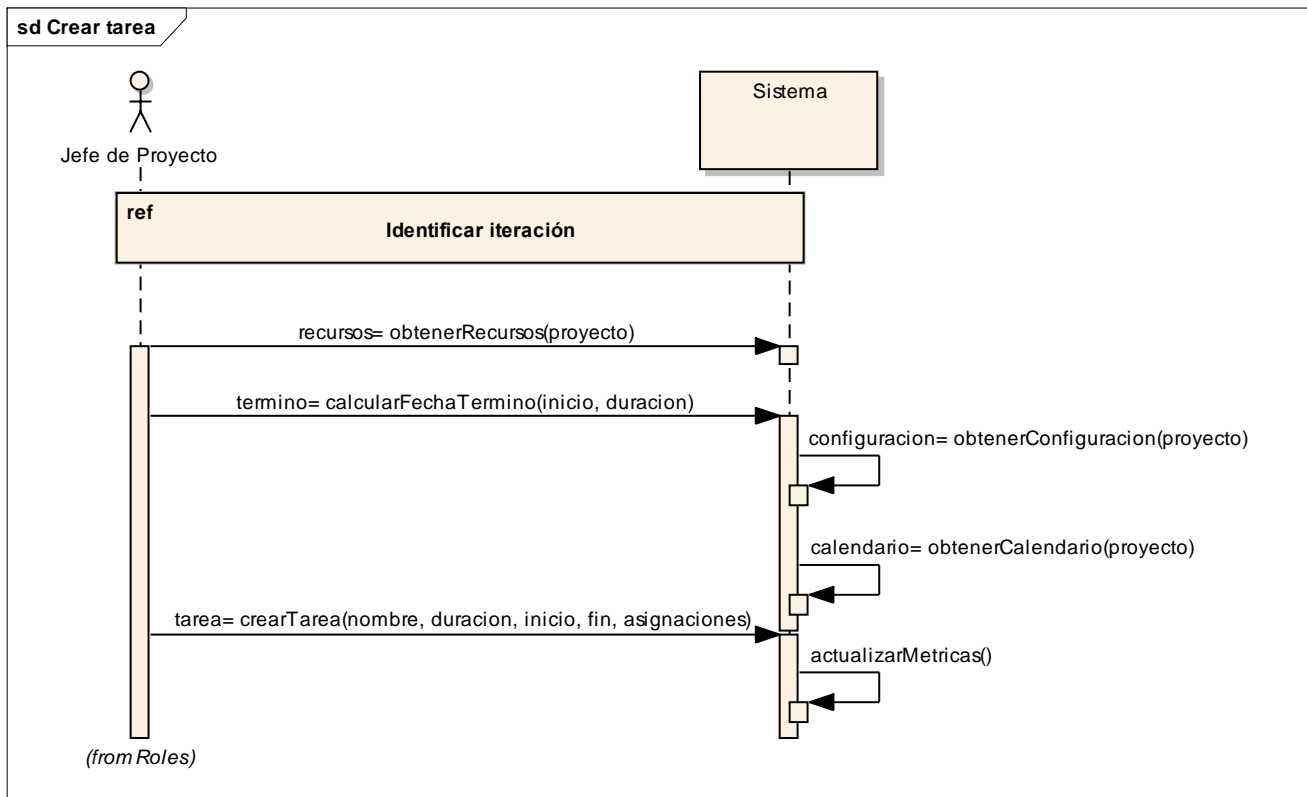


Figura 23. Diagrama de secuencia del caso de uso Crear tarea

Nombre	Consultar asignación
Actores	Recurso
Sinopsis	Permite a un recurso listar todas las tareas asignadas a su persona, para cada uno de los proyectos en los cuales participa.

Nombre	Registrar trabajo
Actores	Recurso
Sinopsis	Permite a un recurso registrar las horas de trabajo incurridas en una tarea que tenga asignada.
Escenario típico	<ol style="list-style-type: none"> 1. Incluir "Consultar asignación" 2. El sistema lista las horas de trabajo previamente registradas en la tarea 3. El usuario ingresa las horas de trabajo 4. El sistema registra las horas de trabajo 5. El sistema despliega mensaje indicando que las horas de trabajo fueron registradas exitosamente y actualiza lista de horas de trabajo

Escenarios alternativos	<ol style="list-style-type: none"> 6. Usuario solicita cerrar tarea <ol style="list-style-type: none"> 1. El usuario solicita cerrar la tarea 2. El sistema registra el cierre de la tarea 3. El sistema despliega mensaje indicando que la tarea fue creada exitosamente y muestra los datos de la tarea. 2a. No existen horas de trabajo previamente registradas <ol style="list-style-type: none"> 1. El sistema muestra mensaje indicando que no existen horas registradas 2. Usuario retoma en punto 3 4a. El usuario no ha ingresado todos los datos del registro de horas <ol style="list-style-type: none"> 1. El sistema notifica el error indicando los datos pendientes 2. El usuario retoma en punto 3.
-------------------------	--

En la Figura 24 se presenta el diagrama de secuencia del caso de uso *registrar trabajo*. Al momento de registrar el trabajo el sistema actualiza las métricas de la tarea. Opcionalmente, el recurso puede solicitar cerrar la asignación, en cuyo caso, una vez cerrada, el sistema actualiza las métricas del proyecto actualizando las horas ganadas y el costo.

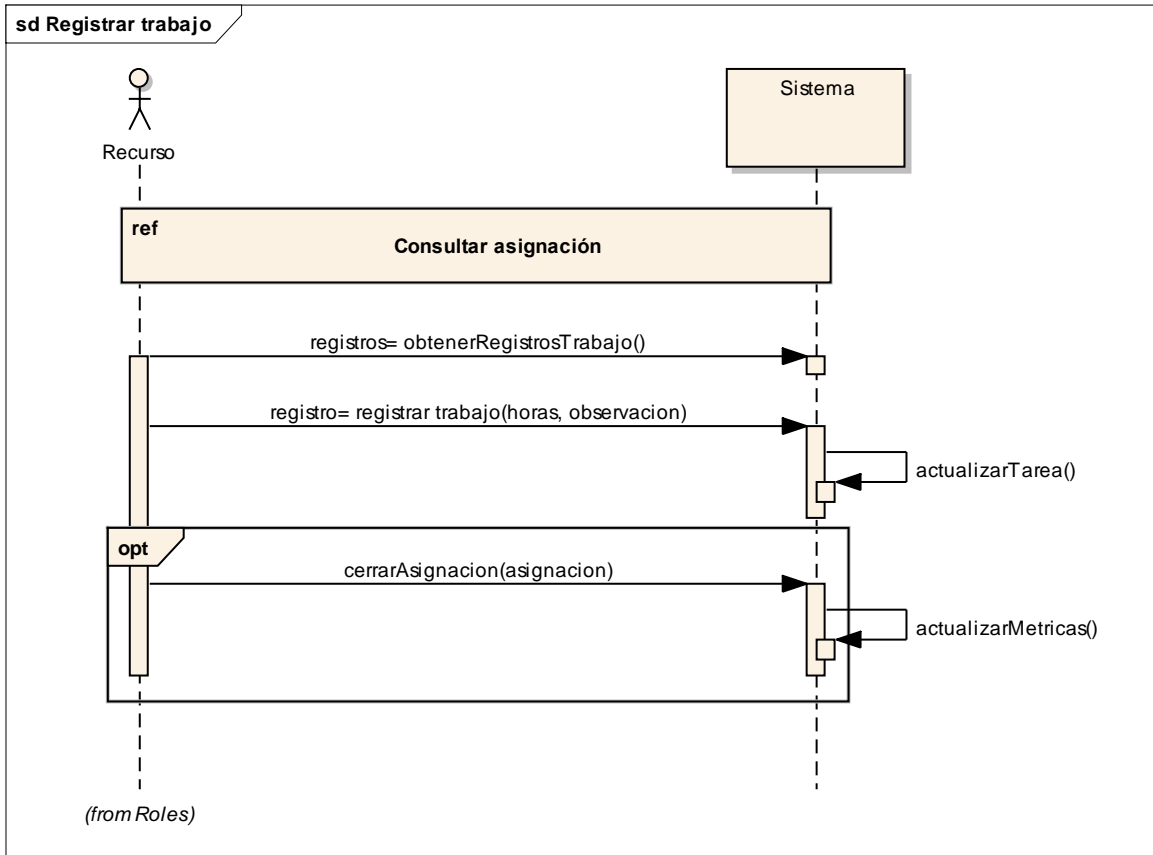


Figura 24. Diagrama de secuencia del caso de uso Registrar trabajo

Nombre	Consultar estado de avance
Actores	Jefe de Proyecto, Administrador de métricas
Sinopsis	Este caso de uso comienza cuando el usuario solicita revisar las métricas de estado de avance de los proyectos.
Escenario típico	<ol style="list-style-type: none"> 1. El sistema lista los proyectos de la empresa 2. El sistema lista los clientes de la empresa 3. El sistema lista las personas de la empresa 4. El sistema lista las categorías de la empresa 5. El usuario selecciona los criterios de consulta 6. El sistema registra las horas de trabajo 7. El sistema despliega las métricas de estado de avance
Escenarios alternativos	N/A

En la Figura 25 se presenta el diagrama de secuencia del caso de uso *consultar estado de avance*. Esto permite obtener la métrica de valor ganado en torno a distintos parámetros, como por ejemplo, obtener la métrica para un cliente en particular, para un proyecto en particular o una persona en particular, además de las combinaciones entre estas.

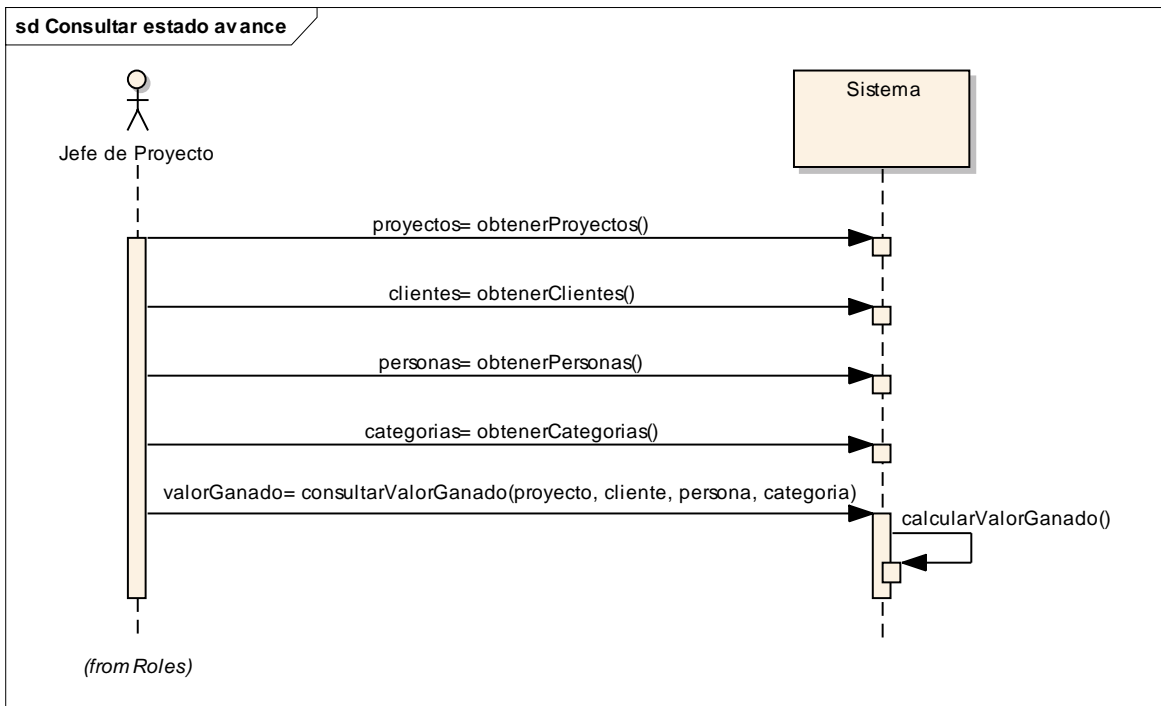


Figura 25. Diagrama de secuencia del caso de uso Consultar estado de avance

Diagrama de clases

En la segunda iteración se identificaron las clases que participan en la realización de los casos de uso además de las asociaciones de estas con las clases identificadas en la primera iteración. Un proyecto se compone de un conjunto de iteraciones, las cuales se componen a su vez en un conjunto tareas. Estas tareas pueden ser asignadas a uno o más participantes del proyecto. La clase "Asignacion" está asociada con la clase "ObservadorAsignacion", que es el encargado monitorear los cambios en las asignaciones y actualizar las métricas del valor ganado.

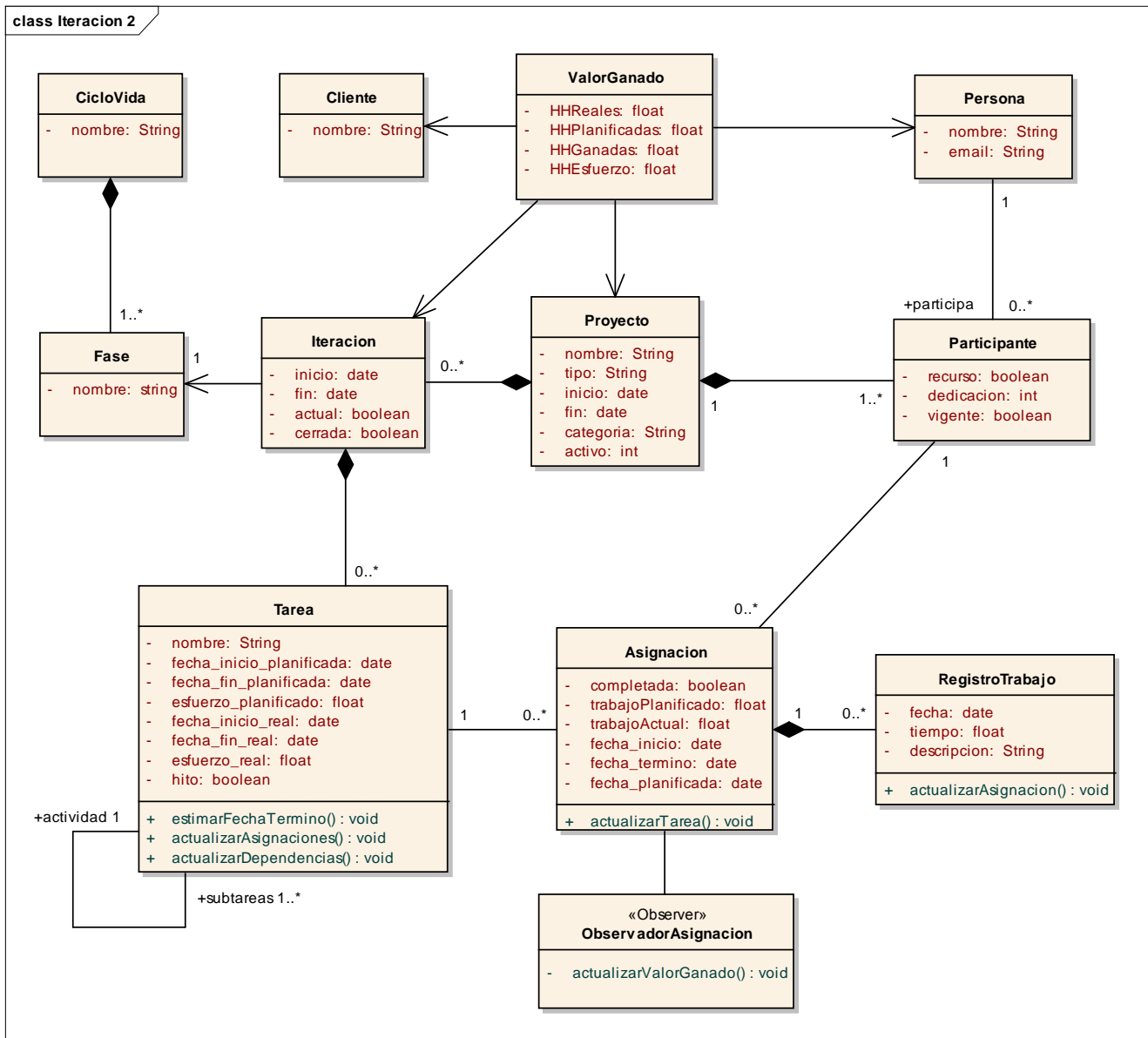


Figura 26. Diagrama de clases Iteración 2

Implementación

En la Figura 27 se presenta la implementación del caso de uso *crear tarea*, en la cual se puede ingresar la información de la tarea y sus responsables. Se destaca la selección de los recursos, los cuales corresponden a los participantes actuales del proyecto. Además se muestra que, al momento de ingresar la duración y fecha de inicio el sistema, sugiere una fecha de término.

COUGAR Mis asignaciones Mis proyectos Procesos Métricas Administración Registrado como Daniel Ramirez

DIGIPROT Documentos Requerimientos Iteraciones Formularios Admin Agregar elemento

Nueva Tarea

Nombre Hito? Duración

Inicio Fin Recursos

- Carla Tarifeño
- Alexis Tapia
- Daniel Ramirez
- Andres Molina
- Francisco Norambuena

Figura 27. Página web implementación crear tarea

En la Figura 28 se presenta la implementación del caso de uso *registrar trabajo*, en la cual un usuario puede ingresar las horas de trabajo dedicadas a una tarea asignada a su persona. El usuario puede ingresar múltiples horas a una misma asignación y cerrar la asignación una vez que considere que ha finalizado la actividad.

Registro de Horas

Se ha ingresado el registro de trabajo

Fecha Tiempo

Observación

Mis horas registradas

Fecha	Tiempo	Observación
28/04/2014	2	análisis preliminar

Figura 28. Página web implementación registrar trabajo

En la Figura 29 y Figura 30 se muestra la implementación del caso de uso *consultar estado de avance*, en donde el encargado de las métricas puede realizar la consulta del estado de avance, utilizando los distintos criterios de consultas propuestos por la aplicación. Dependiendo de los criterios seleccionados el sistema mostrará los resultados

mediante un gráfico lineal y una tabla de valores. En ambos casos, la métrica de valor ganado corresponderá a los valores acumulados.

COUGAR Mis asignaciones Mis proyectos Procesos Métricas Administración

Valor Ganado

Reporte Por fecha Por proyecto

Cliente

Proyecto

Persona

Categoría

Rango -

Buscar

Figura 29. Página web implementación consultar estado de avance

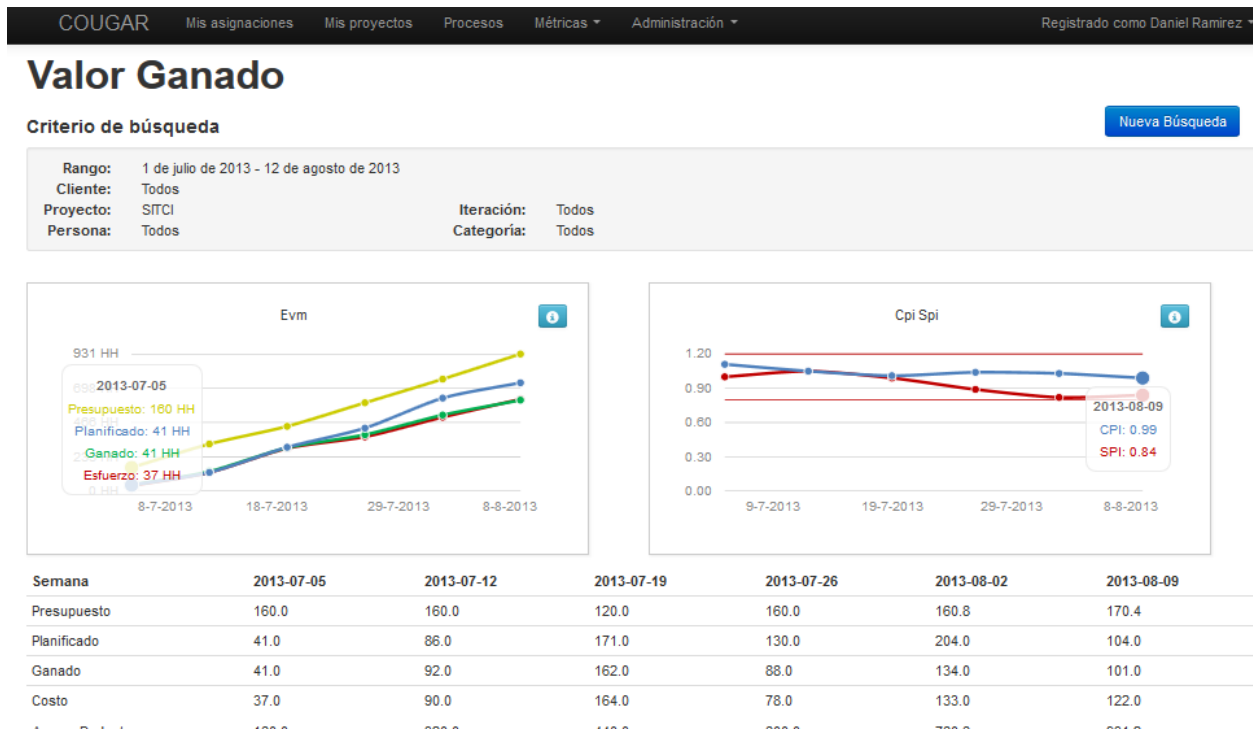


Figura 30. Página web implementación consultar estado de avance

3.9 Tercera Iteración

En la tercera iteración se realizó el diseño e implementación del conjunto de casos de uso que permiten gestionar los requerimientos y versiones del proyecto, así como monitorear los requerimientos a través de la métrica *volatilidad de requerimientos*.

Nombre	Crear requerimiento
Actores	Jefe de proyecto
Sinopsis	Este caso de uso comienza cuando el usuario solicita crear un nuevo requerimiento en el proyecto actual. Se solicitan los datos y, en caso de éxito, se registra y confirma el nuevo requerimiento.
Escenario típico	<ol style="list-style-type: none"> 1. Incluir “Identificar proyecto” 2. El sistema lista las categorías del requerimiento 3. El sistema lista las prioridades del requerimiento 4. El sistema lista los recursos del proyecto 5. El sistema lista las complejidades del requerimiento (sólo en el caso de un requerimiento de software) 6. El usuario ingresa los datos del requerimiento. Los datos son: <ul style="list-style-type: none"> ○ Código ○ Nombre ○ Descripción ○ Categoría ○ Prioridad ○ Responsable ○ Complejidad ○ Origen (sólo en el caso de un requerimiento de negocio o usuario) 7. El sistema agrega el requerimiento 8. El sistema despliega un mensaje indicando que el requerimiento fue creado exitosamente y muestra los datos del requerimiento
Escenarios alternativos	<ol style="list-style-type: none"> 6a. El usuario no ha ingresado todos los datos del requerimiento <ol style="list-style-type: none"> 1. El sistema notifica el error indicando los datos pendientes 2. El usuario retoma en punto 5. 9. El usuario cambia el estado del requerimiento <ol style="list-style-type: none"> 1. El sistema lista los estados disponibles para el requerimiento 2. El usuario selecciona el nuevo estado del requerimiento 3. El sistema registra el estado del requerimiento

4. El sistema despliega mensaje indicando que el estado del requerimiento ha sido actualizado.

En la Figura 31 se presenta el diagrama de secuencia del caso de uso *crear requerimiento*. En el caso de un requerimiento de software, el sistema retornará una lista de complejidades para indicar la complejidad del requerimiento. Opcionalmente, una vez creado el requerimiento se puede cambiar el estado de éste desde un estado inicial a alguno de los estados posibles.

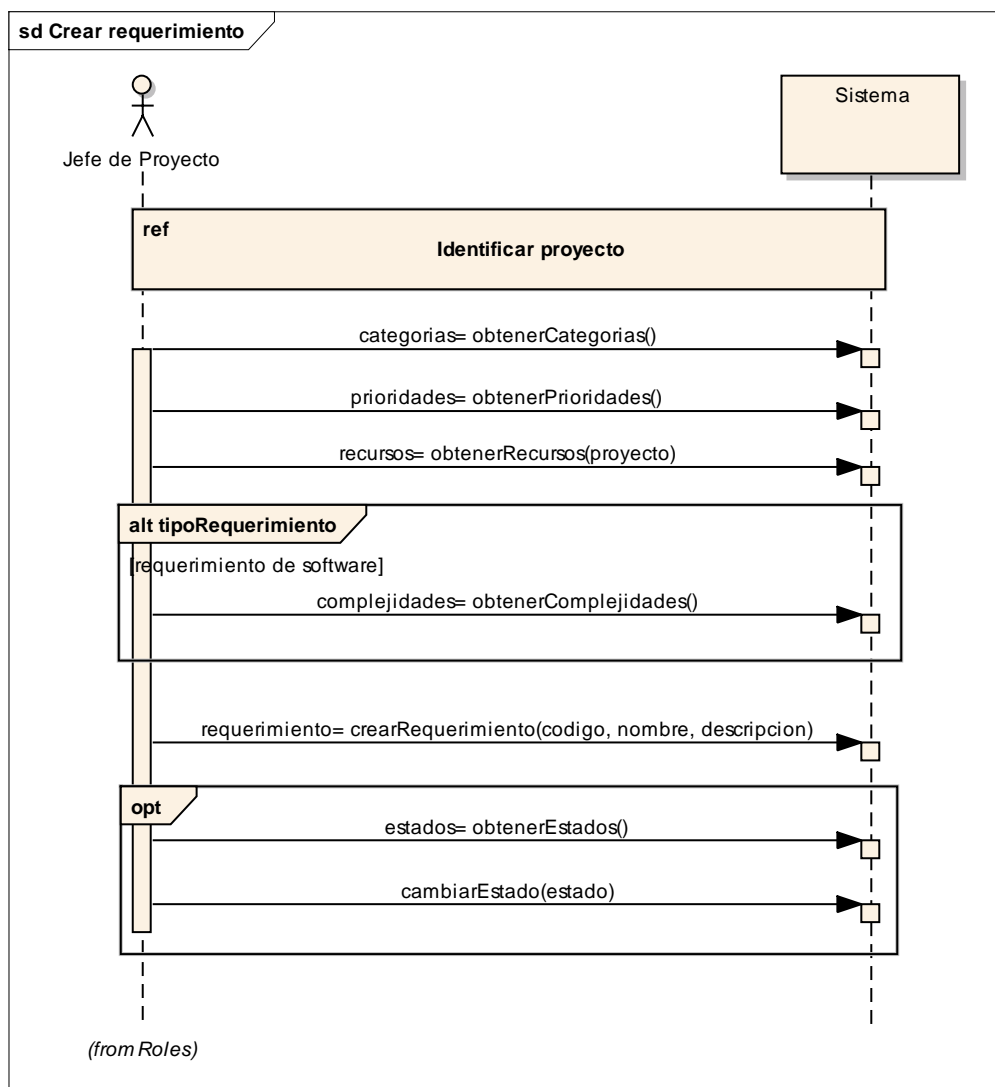


Figura 31. Diagrama de secuencia del caso de uso *Crear requerimiento*

Nombre	Identificar requerimiento
Actores	Jefe de proyecto
Sinopsis	Este caso de uso permite al usuario seleccionar, desde una lista de requerimientos, un requerimiento del proyecto. El sistema retorna y muestra el requerimiento seleccionado.

Nombre	Buscar requerimiento
Actores	Jefe de proyecto
Sinopsis	Este caso de uso comienza cuando el usuario solicita buscar un requerimiento. El sistema lista los requerimientos del proyecto que cumplen con el criterio de búsqueda.

Nombre	Asociar requerimiento
Actores	Jefe de proyecto
Sinopsis	Este caso de uso comienza cuando el usuario solicita asociar uno o más requerimientos al requerimiento actual. El sistema lista todos los requerimientos del proyecto y el usuario selecciona aquellos que desea asociar al requerimiento actual. Esto permite establecer la trazabilidad entre los requerimiento de negocio, usuario y software.

Nombre	Ingresar release
Actores	Jefe de proyecto
Sinopsis	Este caso de uso comienza cuando el usuario solicita registrar una nueva versión del sistema en el proyecto actual. Se solicitan los datos y, en caso de éxito, se registra y confirma la nueva versión.

Nombre	Identificar release
Actores	Jefe de proyecto
Sinopsis	Este caso de uso permite al usuario seleccionar, desde una lista de versiones, una versión del proyecto. El sistema retorna y muestra la versión seleccionada.

Nombre	Incluir requerimientos
Actores	Jefe de proyecto
Sinopsis	Este caso de uso comienza cuando el usuario solicita asociar uno o más requerimientos a la versión actual. El sistema lista todos los requerimientos del proyecto que están en un estado posible para ser asociados. El usuario selecciona aquellos que desea asociar a la versión. Esto permite establecer la trazabilidad entre los requerimientos y las versiones del sistema.

Nombre	Consultar volatilidad de requerimientos
Actores	Jefe de proyecto, Administrador de métricas
Sinopsis	Este caso de uso comienza cuando el usuario solicita revisar las métricas de volatilidad de los requerimientos.
Escenario típico	<ol style="list-style-type: none"> 1. El sistema lista los clientes de la empresa 2. El sistema lista las categorías de la empresa 3. El sistema lista los proyectos de la empresa 4. El sistema lista los releases del proyecto 5. El usuario selecciona los criterios de consulta 6. El sistema despliega las métricas volatilidad de requerimientos
Escenarios alternativos	N/A

En la Figura 34 se muestra el diagrama de secuencia del caso de uso *consultar volatilidad de requerimientos*. De forma similar a la métrica de valor ganado el sistema retorna múltiples opciones de consulta con las cuales se puede ver la información.

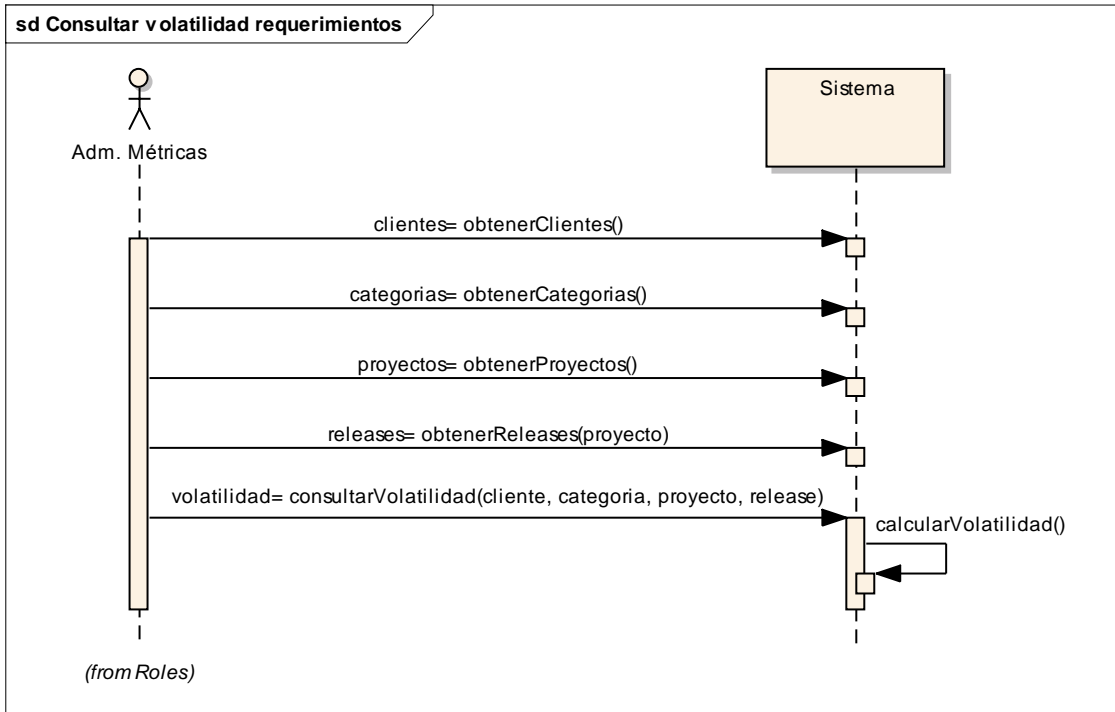


Figura 32. Diagrama de secuencia del caso de uso Consultar volatilidad de requerimientos

Diagrama de clases

En la tercera iteración se identificaron las clases que intervienen en la implementación de la aplicación. Destaca la clase “Requerimiento” de la cual heredan tres clases que corresponden a los tres tipos de requerimientos que maneja la aplicación: requerimientos de negocio, de usuario, y de software; cada uno de los cuales comparten atributos y comportamiento común, así como a su vez tienen atributos y comportamiento propio. La clase “Proyecto” definida en la primera iteración es la que se compone de los requerimientos, así como también se compone de versiones a través de la clase “Release”. Estas versiones, que corresponden a las versiones del sistema entregadas al cliente, se relacionan con los requerimientos de software lo que permite configurar una traza entre ambos.

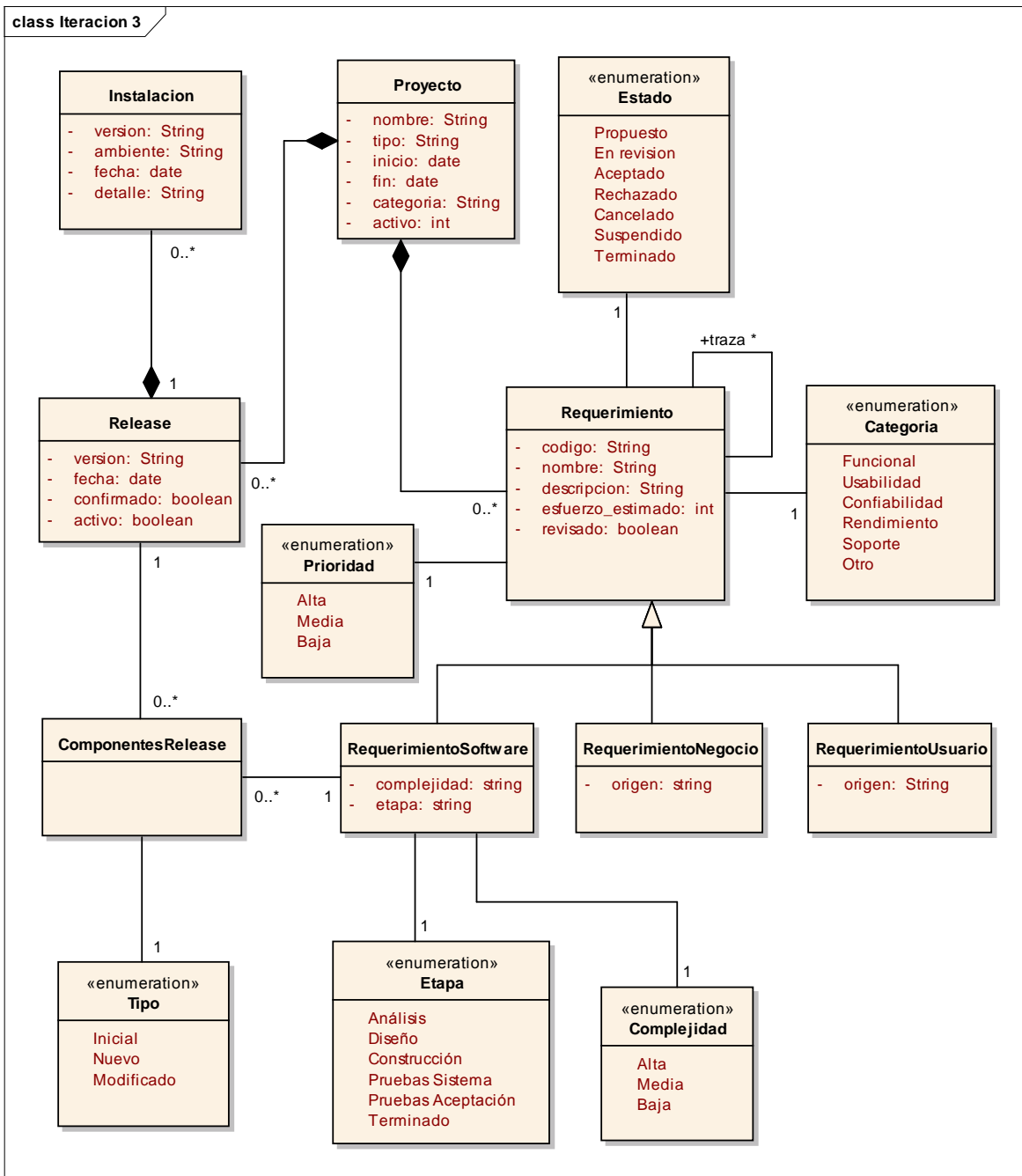


Figura 33. Diagrama de clases Iteración 3

Implementación

En la Figura 34 se presenta la implementación del caso de uso *crear requerimiento*, en la cual un usuario puede registrar un nuevo requerimiento al sistema asociado al proyecto. Como se puede apreciar en la figura, dependiendo del tipo de requerimiento se puede solicitar información adicional, como en el caso del *requerimiento de software* en el cual se solicita indicar la complejidad asociada al requerimiento.

SITCORTE Documentos Requerimientos Iteraciones Formularios Admin Agregar elemento

Nuevo requerimiento de software

Código: Nombre:

Descripción:

Categoría: Prioridad: Complejidad: Responsable:

Figura 34. Página web implementación caso de uso crear requerimiento

Los requerimientos se pueden asociar a una versión del proyecto. En la Figura 35 se muestra cómo se listan los requerimientos del proyecto agrupados por requerimientos de usuario y sus requerimientos de software, los cuales se pueden seleccionar para asociarlos a la versión.

DIGIPROT Documentos Requerimientos Iteraciones Formularios Admin Agregar elemento

Asociar requerimientos a release

Requerimientos de software relacionables al release.

Código	Nombre	Estado	Tipo	Categoría	Prioridad
<input type="checkbox"/> UR-22	Agrupación de recursos de protección	Propuesto	Requerimiento de usuario	Funcional	Alto
<input type="checkbox"/> RS-113	Agregar filtros de búsqueda en bandeja de distribución	Aceptado	Requerimiento de software	Funcional	Alto
<input type="checkbox"/> RS-114	Agrupar de recursos	Aceptado	Requerimiento de software	Funcional	Alto
<input type="checkbox"/> RS-115	Consultar recursos agrupados	Aceptado	Requerimiento de software	Funcional	Alto
<input type="checkbox"/> RS-116	Ver Recursos anidados	Aceptado	Requerimiento de software	Funcional	Alto

Figura 35. Página web implementación caso de uso asociar requerimiento

Una vez asociados los requerimientos estos son listados en la versión seleccionada. Como se puede apreciar en la Figura 36, se muestra para el *release* 1.41.0.0 del proyecto SITCORTE cuáles son los requerimientos asociados a este, y cuál es el estado de los requerimientos. Adicionalmente, para mejor información se muestra un gráfico circular indicando la volatilidad, complejidad y etapa de los requerimientos asociados a esa versión en particular.

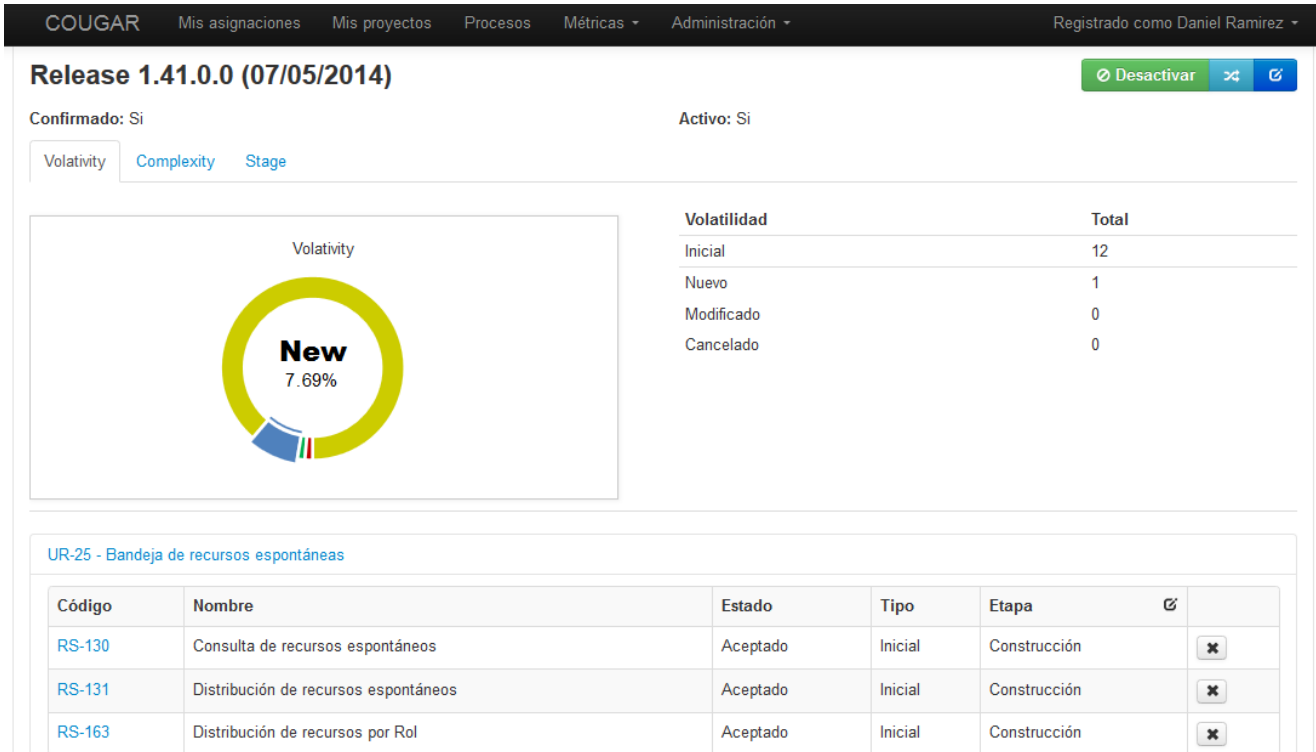


Figura 36. Página web implementación caso de uso identificar release

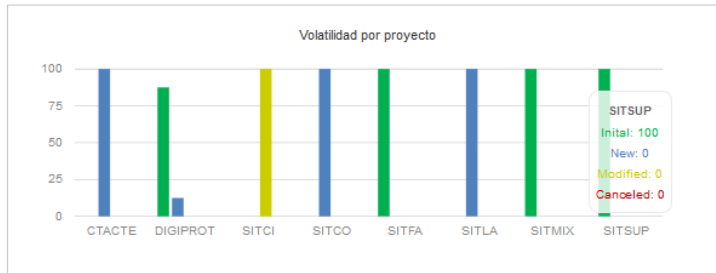
En la Figura 37 se muestra la implementación del caso de uso *consultar volatilidad* en donde el encargado de las métricas puede realizar la consulta utilizando los distintos criterios de consulta propuestos por la aplicación. Dependiendo de los criterios seleccionados el sistema mostrará los resultados mediante un gráfico de barras y una tabla de valores.

Volatilidad de Requerimientos

Criterio de Búsqueda

Nueva Búsqueda

Cliente: PJUD Categoría: Todos
 Proyecto: Todos Release: Todos



CTACTE - 1.27.0.0

UR-66	UR-66 Modificar formato numérico Informe Excel Hist. Depósitos con Saldo	Nuevo
UR-64	UR-64 Modulo Ingreso Talonario Cheques	Nuevo
UR-67	UR-67 Modulo Ingreso Comprobante de Traspaso	Nuevo
UR-60	UR-60 Validación Ingreso Depósitos con Fecha Futura	Nuevo
UR-69	UR-69 Caducación de Cheques 60 días	Nuevo
UR-68	UR-68 Auditoría e Informe MAO Comprobantes de Traspaso Ingresados	Nuevo

Figura 37. Página web implementación caso de uso consultar volatilidad

Capítulo IV

4 Implementación y Resultados Obtenidos

En este capítulo se presenta la estrategia utilizada para implementar el sistema de gestión de proyectos en Amisoft, y los resultados obtenidos durante este proceso.

4.1 Metodología de implementación

La estrategia de implementación del sistema de gestión de proyecto consistió en definir, en conjunto con la organización, una planificación para migrar los proyectos desde el sistema actual al nuevo sistema.

Inicialmente se definió un proyecto piloto que permitiera validar el correcto funcionamiento del sistema y evaluar la calidad de las métricas obtenidas desde el mismo, en comparación con las obtenidas actualmente, a fin de validar su efectividad respecto a los objetivos definidos por la empresa y para este trabajo.

El proyecto piloto continuó la marcha blanca del sistema durante una segunda iteración, con el fin de evaluar las mejoras y monitorear los aspectos nuevos que no pudieran haber sido capturados durante la primera iteración.

Una vez finalizada la marcha blanca del proyecto piloto, se compararon los resultados y se evaluó el trabajo realizado. La organización, en base a esta información, tomó la decisión de incorporar todos los proyectos al sistema de gestión.

Para contrarrestar la resistencia al cambio, se definió un calendario para ir incorporando paulatinamente los distintos proyectos de la organización al nuevo sistema de gestión, hasta que la totalidad de éstos estuvieran utilizando el sistema. Desde entonces, el proyecto piloto dejó de trabajar en paralelo y comenzó a utilizar exclusivamente el sistema nuevo.

La planificación consideró reuniones con todos los jefes de proyecto para coordinar la incorporación de sus proyectos al sistema y, además, reuniones para capacitar a todos los miembros del equipo en uso del mismo. En la incorporación de cada proyecto se realizó un seguimiento diario para garantizar el correcto uso y entendimiento del sistema, así como para aclarar dudas y recoger posibles mejoras en futuras versiones.

A la fecha actual, todos los proyectos de Amisoft son gestionados a través de la herramienta de gestión de proyectos desarrollada en el presente trabajo.

4.2 Experimento Piloto

El proyecto piloto seleccionado correspondió al “Sistema de tramitación de causas civiles” (SITCI), que corresponde a un proyecto de mantención en el cual se implementan nuevas funcionalidades requeridas por los tribunales civiles, así como la resolución de incidencias. Se hizo coincidir el uso del sistema de gestión con el inicio de una nueva iteración del proyecto SITCI, que tuvo una duración de aproximadamente 4 semanas, periodo en el cual el proyecto trabajó utilizando ambos sistemas de gestión paralelamente.

Para el proyecto SITCI, durante la etapa piloto, se registró en el sistema la planificación (carta Gantt) correspondiente a la iteración N°2 del proyecto. De esta forma, como se puede apreciar en la Figura 38, se registraron las actividades y tareas de la iteración y se registraron las asignaciones a cada uno de los recursos del equipo. Durante este periodo cada participante fue registrando sus horas de trabajo en la aplicación. A partir de esta información se pudo calcular las horas planificadas de trabajo, las horas ganadas y el esfuerzo realizado.

203	RS-94 Validar permiso de clacte en el modulo del mismo nomb	17/07/2013	18/07/2013	16 hrs	16 hrs	-	-		
213	RS-95 Agregar validacion al firmar ncl orden de giro	19/07/2013	24/07/2013	27 hrs	27 hrs	-	-		
223	RS-96 Consultar giros realizados de causas en tramitacion	29/07/2013	31/07/2013	20 hrs	20 hrs	30/07/2013	12/08/2013	53 hrs	
233	RS-97 Generar excel al consultar giros de causas	31/07/2013	02/08/2013	23 hrs	23 hrs	12/08/2013	13/08/2013	11 hrs	
243	RS-98 Enviar giro a cuenta corriente	05/08/2013	06/08/2013	12 hrs	12 hrs	14/08/2013	16/08/2013	11 hrs	
253	RS-99 Informe de depositos por fecha	16/07/2013	17/07/2013	12 hrs	12 hrs	16/06/2013	17/07/2013	9 hrs	
263	RS-100 Informe de giros por fecha	18/07/2013	19/07/2013	12 hrs	12 hrs	18/07/2013	19/07/2013	9 hrs	
273	RS-101 Consultar giros causa tramitacion	24/07/2013	29/07/2013	31 hrs	23 hrs	24/07/2013	05/08/2013	16 hrs	
283	RS-102 Consultar depositos causa tramitacion	29/07/2013	31/07/2013	19 hrs	11 hrs	29/07/2013	31/08/2013	8 hrs	
293	RS-103 Crear auditoria de estado diario generacion y desbloq	17/07/2013	23/07/2013	33 hrs	33 hrs	-	-		
303	RS-104 Automatizacion de generacion de estado diario	24/07/2013	05/08/2013	66 hrs	58 hrs	24/07/2013	25/09/2013	28 hrs	
313	Integracion de producto	22/07/2013	26/08/2013	204 hrs	12 hrs	22/07/2013	26/08/2013	13 hrs	
322	(VAL) Planificar Pruebas	29/07/2013	30/07/2013	12 hrs	0 hrs	31/07/2013	31/07/2013	6 hrs	
325	(VER) Generar plan de revisiones	31/07/2013	31/07/2013	3 hrs	3 hrs	05/08/2013	05/08/2013	5 hrs	
326	Ejecutar pruebas	01/08/2013	01/08/2013	8 hrs	3 hrs	31/07/2013	02/08/2013	11 hrs	
331	RS-105 Modificar efecto de girese	01/08/2013	08/08/2013	48 hrs	48 hrs	06/08/2013	12/08/2013	34 hrs	
341	RS-106 Guardar PDF al generar estado diario	12/08/2013	26/08/2013	88 hrs	48 hrs	08/08/2013	22/10/2013	44 hrs	
351	Integracion de producto	13/08/2013	14/08/2013	10 hrs	6 hrs	13/08/2013	19/08/2013	6 hrs	
357	(VAL) Planificar Pruebas	16/08/2013	16/08/2013	3 hrs	3 hrs	20/08/2013	20/08/2013	3 hrs	
359	(VAL) Validar	16/08/2013	19/08/2013	6 hrs	6 hrs	20/08/2013	20/08/2013	6 hrs	
362	RS-106 Agregar Agrupacion por beneficiario en Actuación Automatica	16/08/2013	21/08/2013	26 hrs	26 hrs	19/08/2013	20/08/2013	16 hrs	

Figura 38. Carta Gantt iteración N° 2 SITCI

De manera similar a la planificación, se registraron en el sistema los requerimientos que se estaban desarrollando en la iteración N°2 del proyecto SITCI. Como se puede apreciar en la Figura 39, también se registró el *release* de la aplicación SITCI en la cual se estaba trabajando, correspondiente a la versión 2.7.0.5, y se le asociaron los requerimientos incluidos en dicha versión.

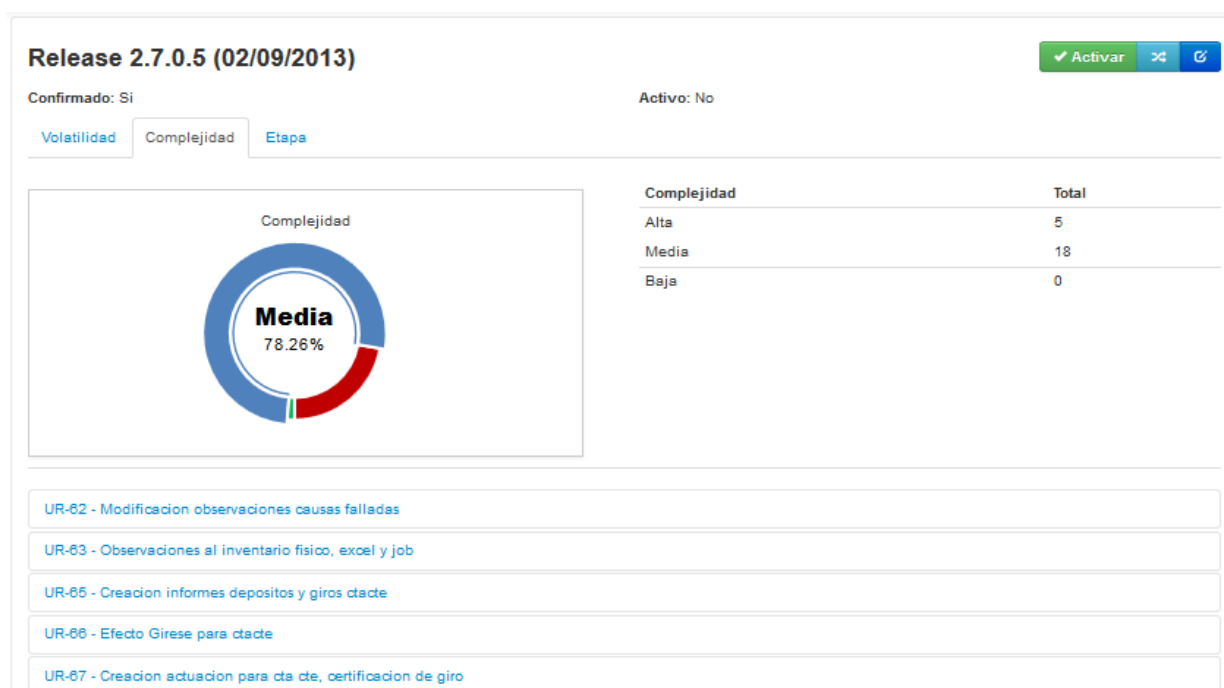


Figura 39. Release y requerimientos asociados del proyecto SITCI

En la medida que la iteración del proyecto SITCI avanzaba, se revisó de forma semanal que las horas obtenidas por el sistema coincidieran con las horas planificadas y las horas registradas por los participantes en el sistema. Además, se revisó que todos los requerimientos estuvieran asociados al *release* en curso, y que la línea base de requerimientos para el *release* se estableciera en el momento oportuno. Así, se quería tener certeza de que la información entregada por el sistema fuera la correcta.

Una vez terminada la iteración se evaluaron los resultados obtenidos, así como se trabajó en implementar algunas mejoras y correcciones en el sistema.

4.3 Resultados obtenidos

Uno de los objetivos principales en el desarrollo de este trabajo era la implementación de métricas que se obtuvieran de forma oportuna en el sistema. Por eso una de las

actividades principales durante la marcha blanca fue evaluar las métricas del sistema. A continuación se presenta un resumen del análisis de las métricas, las cuales corresponden a: valor ganado y la volatilidad de los requerimientos.

Valor Ganado

Para realizar el análisis de la métrica del valor ganado se consideró evaluar: las variables que permiten medir el estado de un proyecto, y los índices de desempeño que se pueden obtener a partir de estas variables para determinar el desempeño del proyecto en costo y tiempo.

Variabes de Análisis de EVM

De acuerdo a lo presentado en el capítulo anterior, las variables para el análisis de la métrica del valor ganado son obtenidas a través del sistema de acuerdo a la información ingresada al sistema.

En la *Tabla 8* se muestra el detalle de las variables obtenidas para el análisis del valor ganado obtenido desde el sistema para cada una de las semanas de la iteración del proyecto.

Tabla 8. Resumen variables valor ganado desde sistema

Horas Hombre / Semana	19-Julio	26-Julio	02-Agosto	09-Agosto	16-Agosto
Presupuesto (BAC)	120	160	160.8	170.4	186.4
Planificado (PV)	171	130	204	104	80
Ganado (EV)	162	88	133	101	105
Esfuerzo (AC)	164	78	132	122	75
Presupuesto Acum. (BAC)	120	280	440.8	611.2	797.6
Planificado Acum. (PV)	171	301	505	609	689
Ganado Acum. (EV)	162	250	383	484	589
Esfuerzo Acum. (AC)	164	242	374	496	571

En la Figura 40 se muestra la gráfica con la evolución del valor ganado a lo largo de la iteración N° 2 del proyecto SITCI. Como se puede apreciar, la gráfica permite determinar, en cada punto de control semanal, cuál era el presupuesto en horas disponible para el proyecto, la cantidad de horas planificadas, las horas ganadas correspondientes a las tareas terminadas y el esfuerzo requerido en horas para finalizar las tareas.

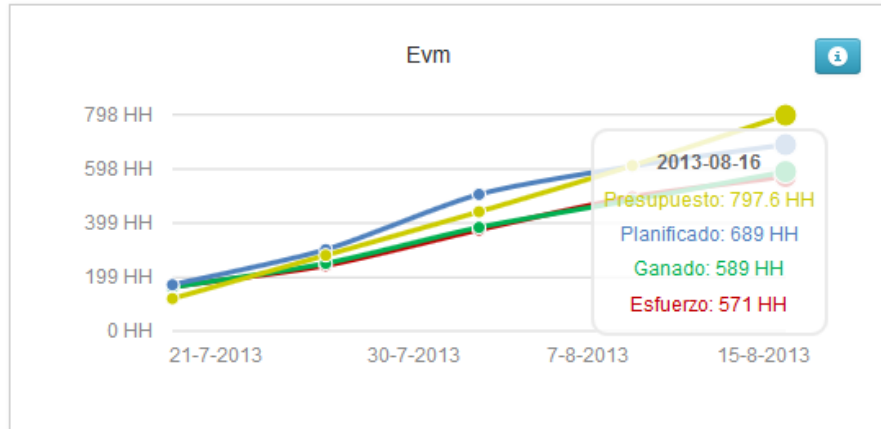


Figura 40. Gráfico Valor Ganado desde Sistema

Una vez finalizada la iteración del proyecto SITCI y realizado el análisis de las variables se pudo concluir que: desde el punto de vista del sistema, éste fue consistente con respecto a lo registrado en el sistema y lo planificado por el jefe de proyecto, descartando inconsistencia en las variables por posibles errores en el sistema.

Índices de desempeño CPI/SPI

Una vez analizadas las variables de la métrica del valor ganado se continuó con el análisis de los índices de desempeño CPI y SPI, que permiten comunicar el estado de un proyecto. El índice de desempeño del costo (CPI) es una medida del valor ganado de un proyecto comparada a los costos reales incurridos, mientras que el índice de desempeño del cronograma (SPI) es una medida de progreso real del cronograma del proyecto [20]. Al repasar estos cálculos, los índices de desempeño deben ser cercanos o superiores a 1. Al igual que con las variables del valor ganado, se quiere validar que los valores entregados por el sistema sean confiables y consistentes. Estos indicadores son calculados a partir de los datos obtenidos por las variables del valor ganado, por lo que no fue necesario validar que los datos de entrada fueran correctos, sino que el cálculo en sí fuera el correcto, para lo cual se realizó un trabajo semanal de cotejar los datos entregados por el sistema contra un cálculo manual de los mismos.

En la *Tabla 9* y *Figura 41* se muestran los resultados obtenidos por el sistema durante las semanas de trabajo del proyecto SITCI de la iteración N° 2.

Tabla 9. Resumen variables índices CPI-SPI desde sistema

Horas Hombre / Semana	19-Julio	26-Julio	02-Agosto	09-Agosto	16-Agosto
CPI	0.99	1.13	1.01	0.83	1.40
SPI	0.95	0.68	0.65	0.97	1.31
CPI Acumulado	0.99	1.03	1.02	0.98	1.03
SPI Acumulado	0.95	0.83	0.76	0.79	0.85

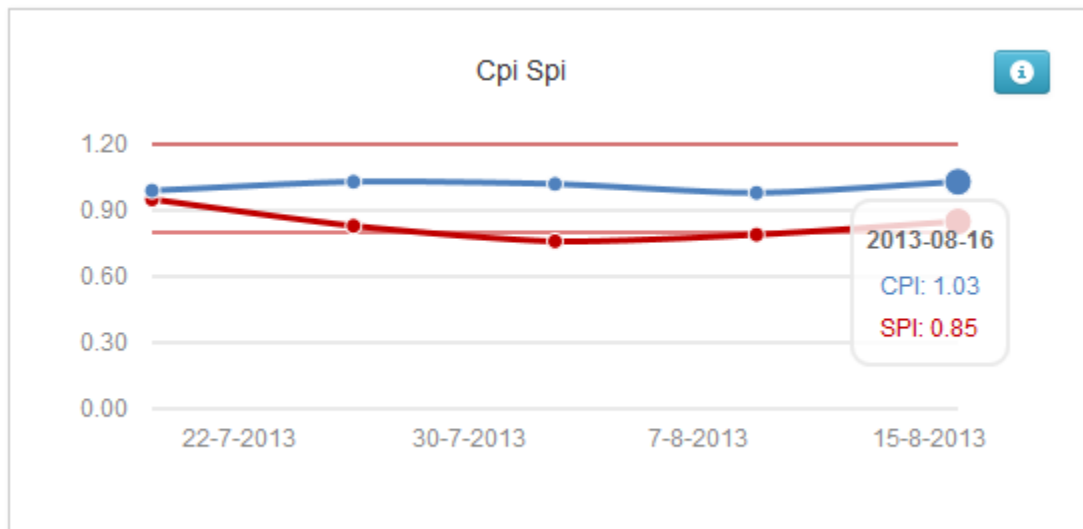


Figura 41. Gráfico valor índices CPI-SPI desde sistema

El cotejo de la información fue exitoso, concluyendo que la información entregada por el sistema es consistente con la información generada por el proyecto.

Volatilidad de Requerimientos

Para la métrica de volatilidad de requerimientos, se realizó una revisión semanal en conjunto con el jefe de proyecto del sistema SITCI para validar que la información entregada por el sistema fuera consistente.

Durante la planificación de la iteración, el jefe de proyecto, en acuerdo con su contraparte, acordaron la implementación de un conjunto de requerimientos iniciales para la versión 2.7.0.5 de la aplicación SITCI, con los cuales se estableció la línea base de requerimientos. En el transcurso de la iteración se añadieron nuevos requerimientos, los cuales, en conjunto con los requerimientos iniciales, sufrieron modificaciones en su totalidad. Como se puede apreciar en la Figura 42, al finalizar la iteración N°2 del proyecto

SITCI, la totalidad de los requerimientos para la versión 2.7.0.5 habían sido modificados respecto a sus versiones iniciales.

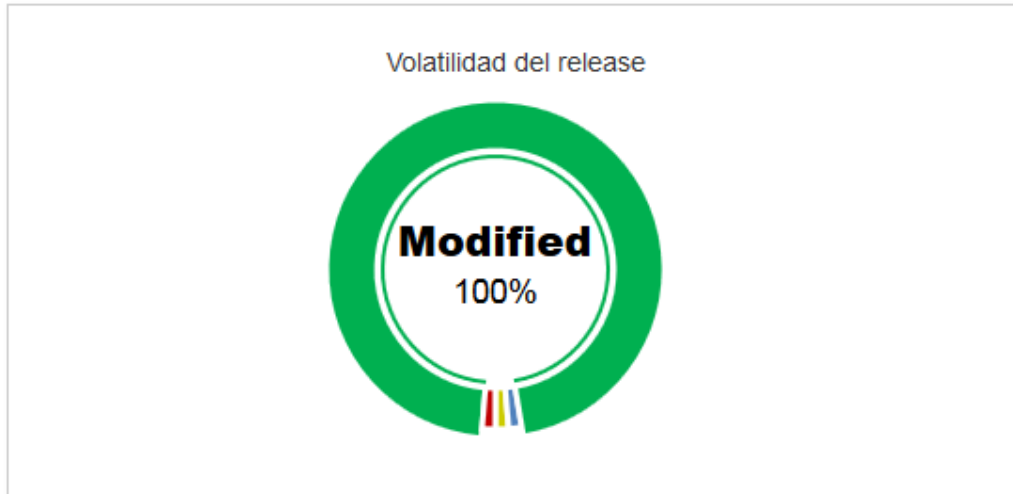


Figura 42. Volatilidad de requerimientos proyecto SITCI – Sistema gestión

4.4 Comparación con el enfoque manual

No obstante fue posible validar las métricas obtenidas desde el sistema, se consideró prudente analizar los datos obtenidos mediante el método manual que era utilizado hasta esa fecha por todos los proyectos, a modo de poder compararlos y sacar algunas conclusiones acerca de ambos enfoques.

Valor Ganado

El proceso de obtención de las métricas actual considera como punto de partida la carta Gantt de la iteración, al igual que lo hace el sistema, sin embargo, como se puede apreciar en la Figura 43, el registro de las horas de trabajo se realiza a partir de una planilla de registro, la cual debe ser llenada por todos los participantes del proyecto y consolidada en un único documento por el jefe de proyecto. A partir de esta información el jefe de proyecto semanalmente realiza una evaluación de estado cotejando las horas planificadas en la carta Gantt contra las horas ganadas en la planificación y el esfuerzo realizado obtenido de la planilla de registro de horas. En la Figura 44 se puede ver el resumen de evaluación de estado del proyecto SITCI.

FECHA	AREA DE PROCESO	ACTIVIDAD	ROL	RESPONSABLE	TIEMPO [H:MM]	OBSERVACIONES
29-07-2013	RD	Documentar requerimientos	Analista de sistemas	Mauro Contreras	1:00	formateo del documento de requeri
29-07-2013	NA	Otro	0	Mauro Contreras	1:00	revisión de tareas generales
30-07-2013	RD	Documentar requerimientos	Analista de sistemas	Mauro Contreras	2:30	formatos del documento de ERS
30-07-2013	NA	Otro	0	Mauro Contreras	0:40	evaluación recurso
31-07-2013	NA	Reunión	0	Mauro Contreras	1:00	reunión PMO
31-07-2013	NA	Reunión	0	Mauro Contreras	1:20	reunión por tema RD
31-07-2013	NA	Reunión	0	Mauro Contreras	1:30	reunión con Nathaly
01-08-2013	NA	Reunión	0	Mauro Contreras	1:30	reunión
29-07-2013	TS	Desarrollar componentes y programas	Programador	Scarlett Morey	3:30	Desarrollo ctacte
29-07-2013	PMC	Registrar incidentes	Jefe de Proyecto/Analista/Programador	Scarlett Morey	3:30	Registro de incidencias desde 11:1
30-07-2013	NA	Otro	0	Scarlett Morey	2:00	Explicar Desarrollo a suje y Instala
30-07-2013	TS	Desarrollar componentes y programas	Programador	Scarlett Morey	2:50	Desarrollo a contar de las 11
30-07-2013	NA	Reunión	0	Scarlett Morey	1:30	Reunión de avance
30-07-2013	PMC	Registrar incidentes	Jefe de Proyecto/Analista/Programador	Scarlett Morey	0:30	Registro de incidencias
31-07-2013	PMC	Registrar incidentes	Jefe de Proyecto/Analista/Programador	Scarlett Morey	4:00	Registrar incidencias
31-07-2013	TS	Desarrollar componentes y programas	Programador	Scarlett Morey	3:15	RS- 101 y 102
01-08-2013	TS	Desarrollar componentes y programas	Programador	Scarlett Morey	3:00	RS- 101 y 102
01-08-2013	NA	Otro	0	Scarlett Morey	1:06	Pruebas varias
01-08-2013	PMC	Registrar incidentes	Jefe de Proyecto/Analista/Programador	Scarlett Morey	3:30	Registrar incidencias
02-08-2013	NA	Otro	0	Scarlett Morey	2:15	implementar Observaciones
02-08-2013	TS	Desarrollar componentes y programas	Programador	Scarlett Morey	1:15	RS- 101 y 102
02-08-2013	NA	Otro	0	Scarlett Morey	0:35	sin vpn
02-08-2013	PRUEBAS	Validar	Tester	Scarlett Morey	1:00	pruebas unitarias req. 101 y 102
02-08-2013	PMC	Registrar incidentes	Jefe de Proyecto/Analista/Programador	Scarlett Morey	2:15	Registrar incidencias

Figura 43. Planilla de registro de horas proyecto SITCI

Resumen Evaluación de Estado										
SITCI										
Estado del proyecto - tiempo										
Fecha	Nº Iteración	Área de proceso	Total Gantt	Planificado (PV)	Ganado (EV)	Esfuerzo/ Costo actual (AC)	Índice de desempeño de esfuerzo CPI (EV/AC)	Índice de desempeño de cronograma SPI (EV/PV)	Línea Base	Presupuesto
19-07-2013	2	RD		31	21	1	2100%	67,7%	31	
19-07-2013	2	TS		98	71	63	113%	72,4%	98	
19-07-2013	2	PI		0	0	0	-	-	0	
19-07-2013	2	Validación		0	0	0	-	-	0	
19-07-2013	2	Verificación		0	0	0	-	-	0	
19-07-2013	2	PP		0	0	0	-	-	0	
19-07-2013	2	REQM		0	0	4	0%	-	0	
19-07-2013	2	PMC		26	26	23	113%	100,0%	26	
19-07-2013	2	NA		15	15	13,2	114%	100,0%	15	
19-07-2013	2	Total semana 1		170,00	133,00	104,20	1,28	0,78	170	159,8
26-07-2013	2	RD		24	22	9,3	237%	91,7%	24	
26-07-2013	2	TS		38	72	60,57	119%	189,5%	38	
26-07-2013	2	PI		6	6	4,5	133%	100,0%	6	
26-07-2013	2	Validación		0	0	0	-	-	0	
26-07-2013	2	Verificación		0	0	0	-	-	0	
26-07-2013	2	PP		0	0	0	-	-	0	
26-07-2013	2	REQM		0	0	4	0%	-	0	
26-07-2013	2	PMC		26	26	28,5	91%	100,0%	26	
26-07-2013	2	NA		15	15	22,39	67%	100,0%	15	
26-07-2013	2	Total semana 2		109,00	141,00	129,26	1,09	1,29	109	212,5
26-07-2013	2	Acumulado semana 2		279,00	274,00	233,46	1,17	0,98	279	372,3

Figura 44. Planilla evaluación de estado proyecto SITCI

Variables de Análisis de EVM

En la *Tabla 10* se muestra el detalle de las variables obtenidas para el análisis del valor ganado obtenido desde la planilla de evaluación de estado para cada una de las semanas de la iteración del proyecto.

Tabla 10. Resumen variables valor ganado desde planilla

Horas Hombre / Semana	19-Julio	26-Julio	02-Agosto	09-Agosto	16-Agosto
Presupuesto (BAC)	159.8	212.5	212.5	170	170
Planificado (PV)	170	109	165	145	104
Ganado (EV)	133	141	108	139	104
Esfuerzo (AC)	104.2	129.26	106	116	110
Presupuesto Acum. (BAC)	159.8	372.3	584.8	754.8	924.8
Planificado Acum. (PV)	170	279	444	589	693
Ganado Acum. (EV)	133	274	382	521	625
Esfuerzo Acum. (AC)	104.2	233.46	339.46	455	565

En la Figura 45 se muestra la gráfica con la evolución del valor ganado a lo largo de la iteración N° 2 del proyecto SITCI obtenida a partir de los datos de la planilla de evaluación de estado.

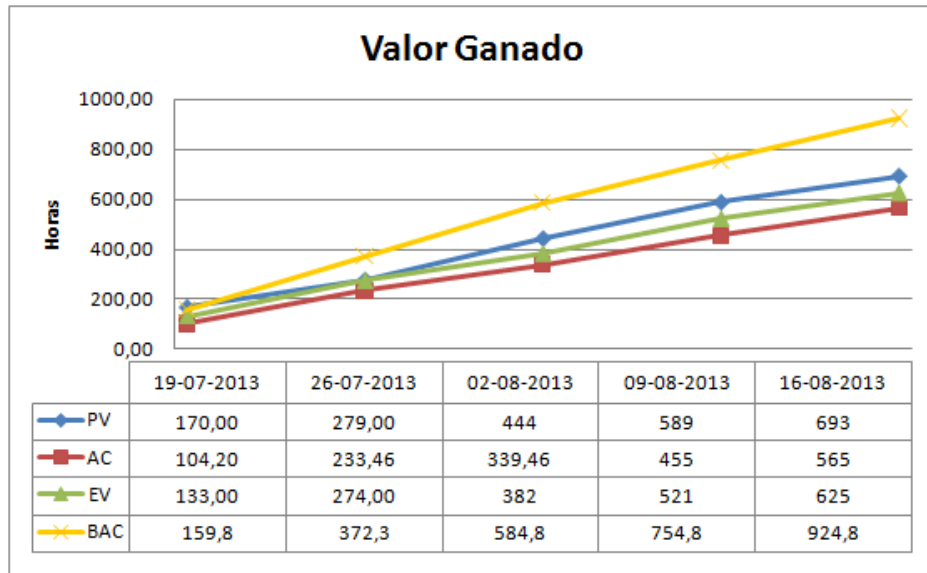


Figura 45. Gráfico valor ganado proyecto SITCI desde planilla

Una vez realizado el análisis de las variables del valor ganado, obtenido desde ambas fuentes, se pudo concluir que, si bien ambos no coinciden en los valores obtenidos, en general ambos no presentan diferencias significativas. La mayor diferencia se presentó en el presupuesto del proyecto, la cual se originó en la falta de comunicación entre el jefe de proyecto y el analista de métricas acerca de un participante que no fue integrado al proyecto.

Las diferencias entre ambos proyectos dejaron en evidencia la introducción de errores en las métricas manuales producto de las distintas fuentes de datos y el cruce de información que se debe realizar para obtener las métricas.

No obstante, cabe destacar que ambos métodos dependen significativamente de que los participantes ingresen la información requerida para obtener las métricas. Vale decir, el sistema está evitando la fuente de error humana y disminuyendo los tiempos requeridos para obtener las métricas.

Índices de desempeño CPI/SPI

Debido a que las variables del valor ganado obtenidas desde la planilla de evaluación de estado eran distintas a las entregadas por el sistema, se sabía a priori que los índices de desempeño CPI-SPI serían diferentes. No obstante lo anterior se realizó una verificación para validar que no existieran diferencias significativas.

En la *Tabla 11* y *Figura 46* se muestran los resultados obtenidos desde la planilla durante la iteración N° 2.

Tabla 11. Resumen variables índices CPI-SPI desde planilla

Horas Hombre / Semana	19-Julio	26-Julio	02-Agosto	09-Agosto	16-Agosto
CPI	1.28	1.09	1.02	1.20	0.95
SPI	0.78	1.29	0.65	0.96	1.00
CPI Acumulado	1.28	1.17	1.13	1.14	1.11
SPI Acumulado	0.78	0.98	0.86	0.88	0.90

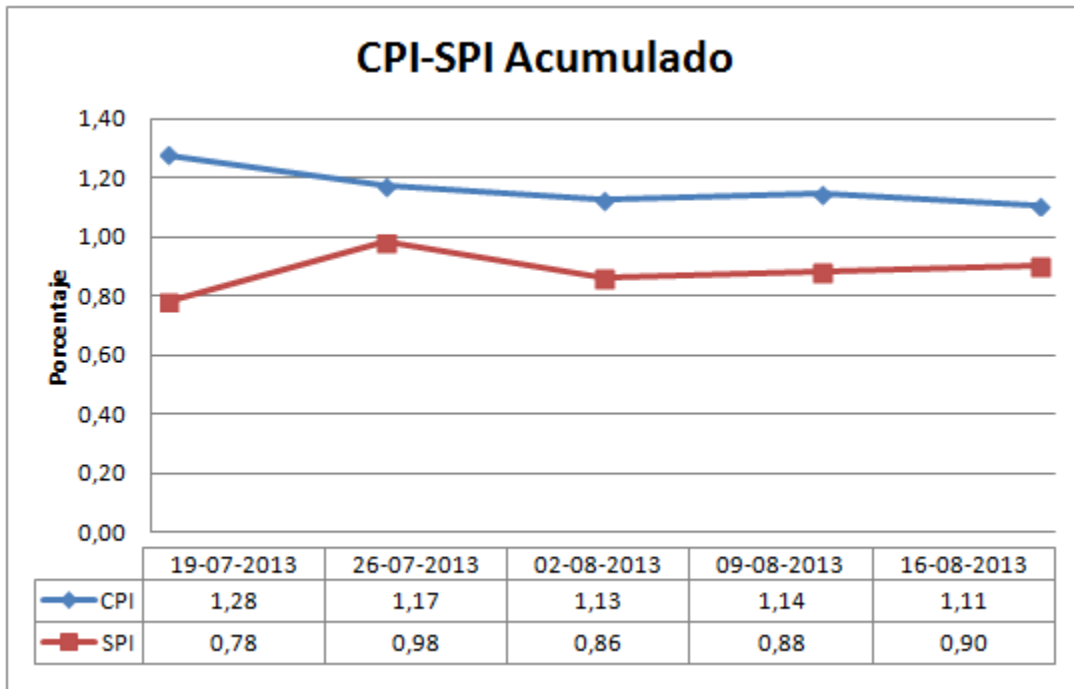


Figura 46. Gráfico valor índices CPI-SPI proyecto SITCI desde planilla

Volatilidad de requerimientos

En la métrica de *volatilidad de requerimientos* nos encontramos con una de las primeras diferencias. Actualmente, la métrica era obtenida manualmente por el analista de métricas a partir de los requerimientos considerados dentro de una iteración de trabajo, y registrada en una planilla de evaluación de estado. Como se puede apreciar en la Figura 47, la métrica de volatilidad de requerimientos obtenida desde la planilla indica que existen un total de 89% de requerimientos nuevos y un 11% de requerimientos terminados.

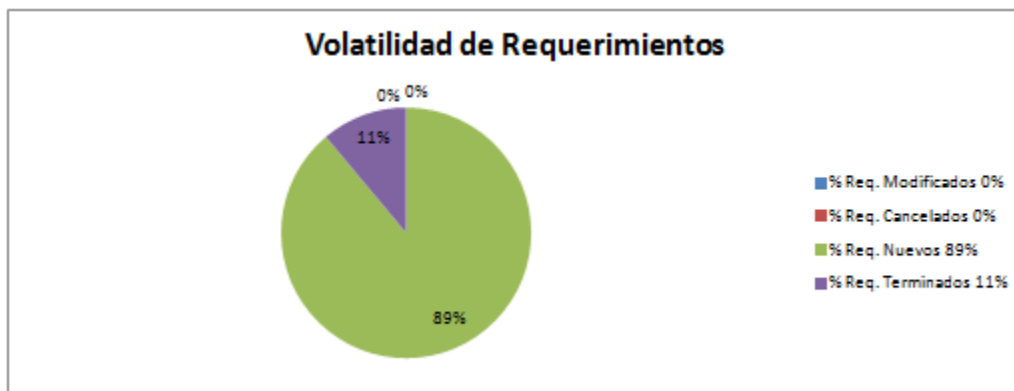


Figura 47. Volatilidad de requerimientos proyecto SITCI desde planilla

A diferencia de las métricas obtenidas desde la planilla, el sistema de gestión indica que la totalidad de los requerimientos han sido modificados, tal como se puede apreciar en la Figura 42.

Como se puede apreciar, existe una diferencia total entre las métricas capturadas. Esto se debe principalmente a que el método manual no considera una línea base sobre la cual calcular la métrica de forma tal que, el cambio depende de la rigurosidad del jefe de proyecto para actualizar los registros necesarios para que el analista de métricas detecte el cambio. Esto conlleva dos posibles entradas de error: que el jefe de proyecto no actualice el registro, y que el analista de métricas tome conocimiento del cambio en el momento de la recolección de la métrica. Considerando que el proceso de recolección es manual, el error humano es un factor a considerar. El sistema, por el contrario, propone una línea base para los requerimientos del *release*, sobre la cual calcular la métrica, a medida que se agregan los requerimientos al *release*.

Realizado el análisis de los requerimientos en conjunto con el jefe de proyecto de SITCI, se pudo validar que efectivamente durante la planificación de la iteración existía un conjunto de requerimientos iniciales a los cuales se les añadió un importante conjunto de requerimientos nuevos. No obstante, en el transcurso de la iteración para entregar una nueva versión del sistema, la cual ya se encontraba confirmada, estos requerimientos sufrieron modificaciones en su totalidad. Se puede concluir de esta manera que, si bien la métrica manual reflejaba la realidad del proyecto en un punto de la iteración, no es la métrica precisa al finalizar la misma.

Es importante destacar, respecto a la medición de la métrica, un cambio que podría pasar desapercibido, pero que tiene un alto impacto en aquellos proyectos que definen más de una iteración para hacer entrega una misma versión del sistema. Mediante el método manual, al definir una nueva iteración con nuevos requerimientos para la misma versión, éstos serían considerados como requerimientos iniciales, ya que el cálculo se realiza por iteración, sin embargo, en la realidad éstos son requerimientos nuevos del sistema. Esto no ocurre en el sistema, ya que la métrica de volatilidad de los requerimientos es considerada a nivel de la versión entregable del sistema, y no de la iteración del proyecto.

4.5 Conclusiones acerca de los resultados obtenidos

Una vez analizados los resultados de ambos enfoques, se puede concluir que, una de las ventajas del sistema, con respecto al método manual, es que la información está disponible para el jefe de proyecto en todo momento, por lo que se pueden tomar acciones correctivas a partir de esta información. Con el método manual, la información no está disponible sino una semana después, cuando las métricas son elaboradas por el analista de métricas.

Otro de los aspectos destacables del sistema es que, al centralizar la información y automatizar las métricas, se disminuye la posibilidad de introducir inconsistencias por errores humanos en los cálculos. Sin embargo, y como se mencionó anteriormente, ambos métodos dependen significativamente de que los participantes registren la información requerida en forma correcta y consistente, lo cual no es posible de controlar por el sistema, sino que depende de la capacidad de la organización para capacitar y promover su correcto uso y las ventajas que eso conlleva.

4.6 Impacto económico del proyecto

A medida que el proceso de desarrollo de proyectos (APF) se fue institucionalizando en la organización, se detectó la necesidad estratégica de contar con información de sus procesos y saber cuál es el desempeño de los proyectos que permitan a la gerencia mejorar su toma de decisiones. Con este afán parte importante de los esfuerzos en el último periodo estuvieron abocados a la implementación de un área de proceso de Medidas y Análisis. Mediante un análisis costo-beneficio, la empresa evaluó positivamente esta implementación y estimó los costos anuales del área de Medidas y Análisis en 1.056 UF, considerando entre el costo de un analista de métricas y el costo anual de obtención de las métricas por parte de los jefes de proyecto [29].

Como se puede apreciar en la *Tabla 12*, una parte importante de las actividades realizadas por el analista de métricas está dedicada exclusivamente a la recolección de los datos y actualizar documentos, por lo que el foco estaba concentrado prioritariamente en la captura y no en el análisis.

Tabla 12. Costo analista de métricas en hh

Actividades	Frecuencia	Tiempo aprox	Tiempo mensual
Recolectar datos de las mediciones	Semanal	8	32
Analizar y registrar los datos recolectados	Semanal	3	12
Comunicar los resultados	Semanal	0.5	2
Actualizar planilla de métricas	Mensual	2	2
Realizar resumen de métricas por proyecto	Semanal	5	20
Monitorear plan de medición	Mensual	1	1
		Total	69

En la *Tabla 13* se muestra una estimación de los costos mensuales incurridos por la organización relacionados exclusivamente a la recolección de métricas.

Tabla 13. Costo recolección y procesamiento de métricas

	Tiempo Mes HH	Valor HH UF	Total UF
Costos analista de métricas	64	0,6	38,4
Costos jefes de proyecto	60	0,7	42
		Total	80,4

Por este motivo, se considero necesario conocer el impacto económico que tendría el proyecto para la organización y cómo afectará el desempeño de la empresa. Para ello se realizó un también un análisis del costo-beneficio de implementar el proyecto. A continuación se describe el análisis realizado y cómo se recogieron los datos e interpretaron para llegar a un análisis económico financiero del proyecto.

De acuerdo a la planificación de las iteraciones presentadas en el capítulo anterior, en la *Tabla 14* se presenta un resumen del costo en horas-hombre y valor en UF de la implementación del sistema, considerando para ello el trabajo de un analista programador que tiene un costo de 0,6 UF la hora. De esta forma la implementación del sistema se ha realizado en un periodo de tres meses con una inversión de 271,8 UF.

Tabla 14. Costo implementación del sistema en UF

Actividad	Duración HH	Total UF
Crear proyecto	24,00	14,4
Identificar proyecto	16,00	9,6
Configurar proyecto	16,00	9,6
Configurar calendario	24,00	14,4
Asignar recurso	32,00	19,2
Ingresar al sistema	16,00	9,6
Crear iteración	20,00	12
Identificar iteración	16,00	9,6
Crear tarea	32,00	19,2
Consultar asignación	16,00	9,6
Registrar trabajo	24,00	14,4
Consultar estado de avance	45,00	27
Crear requerimiento	30,00	18
Identificar requerimiento	16,00	9,6
Buscar requerimientos	16,00	9,6
Asociar requerimiento	24,00	14,4
Ingresar release	24,00	14,4
Identificar release	16,00	9,6
Incluir requerimientos	16,00	9,6
Consultar volatilidad de requerimientos	30,00	18
	TOTAL	271,8

Considerando los costos del área de proceso de Medidas y Análisis, y como se puede apreciar en la Figura 48, la organización recuperaría la inversión al cabo del cuarto mes logrando un ahorro significativo al cabo de un año. Por otro lado las horas-hombre ahorradas tanto por parte del analista de métricas como de los jefes de proyecto pueden ser invertidas en otras actividades pertinentes para la organización.

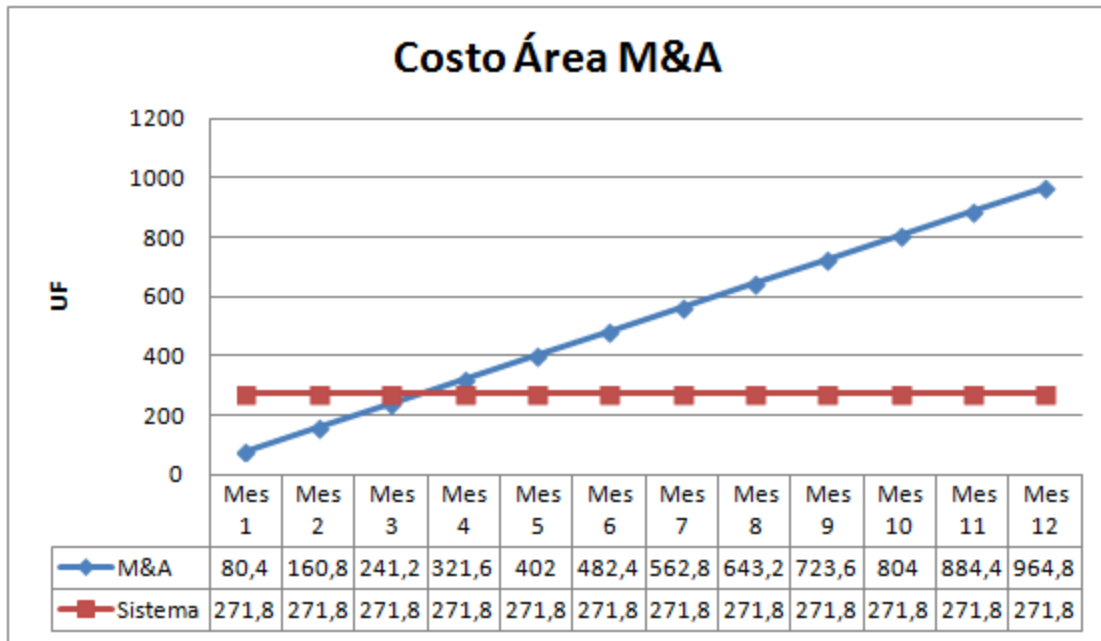


Figura 48. Evolución de costos en la recolección de métricas del área de M&A versus el costo de implementación del sistema

Además del beneficio económico, se puede concluir que contar con la herramienta de apoyo a la gestión permite:

- Posibilita la obtención de las métricas sin mayor esfuerzo
- Minimiza los errores en el cálculo de las métricas, logrando una mayor exactitud en sus valores
- Permite que la empresa se concentre en el análisis de los resultados de la medición y no en la etapa de captura de la información

4.7 Implementación de nuevas funcionalidades

Después de incluidos todos los proyectos en el nuevo sistema de gestión de proyectos presentado en el presente trabajo, se detectaron, en conjunto con la organización una serie de características que serían importantes de incorporar considerando que ya se estaba dejando de utilizar el sistema anterior y que además eran importantes de tener en miras de la certificación ISO-9001:2000 y la evaluación CMMI-DEV.

En la *Tabla 15* se listan las características que se decidió incorporar al sistema de gestión de proyectos, muchas de las cuales ya habían sido detectadas en la etapa de análisis del capítulo anterior.

Tabla 15. Características incorporadas en el sistema

N°	Característica	Descripción
1	Gestionar documentación	Permite realizar el registro, almacenamiento y disposición de la información del proyecto.
2	Administrar solicitudes cambio	Permite registrar las solicitudes de cambios a los requerimientos.
3	Gestionar proceso	Permite registrar y mantener actualizado el proceso de desarrollo definido por la organización para su uso en la gestión de los proyectos.
4	Administrar usuarios	Permite al administrador del sistema y a los usuarios del mismo mantener actualizados sus datos y privilegios.
5	Administración de incidencias	Permite registrar las incidencias reportadas en el proyecto y registrar el estado y resolución de las mismas.
6	Administrar involucrados	Permite registrar a los interesados (stakeholders) de un proyecto y registrar sus comunicaciones mediante minutas de reuniones.

De esta manera se realizó una planificación para implementar dichas funcionalidades en el sistema y ponerlas a disposición de los proyectos. El objetivo principal de este nuevo ciclo de desarrollo era cumplir con las exigencias, a nivel de registros, de las normas antes mencionadas, así como centralizar las fuentes de información de los proyectos en un único repositorio, que permitiera, además, la incorporación de nuevas métricas en el sistema a futuro.

Capítulo V

5 Conclusiones y Trabajo Futuro

En este capítulo se presenta un resumen de las conclusiones, lecciones aprendidas y trabajos futuros sobre la herramienta de gestión cuantitativa de proyectos de software.

5.1 Conclusiones

El proyecto de tesis “HERRAMIENTA DE GESTION CUANTITATIVA DE PROYECTOS DE SOFTWARE ORIENTADA POR UN PROCESO DE DESARROLLO” ha permitido determinar las características más relevantes que un sistema de apoyo a la gestión de proyectos debería proveer, a partir de las cuales se ha desarrollado una primera versión del sistema, que ha sido puesto en marcha en la empresa Amisoft. Este sistema ha facilitando la gestión de los proyectos de desarrollo de software, permitiendo obtener en forma oportuna un conjunto de métricas que apunten a mejorar la calidad del proceso y productos de la organización.

Así se puede concluir que se cumplió con el objetivo central del presente trabajo y con los objetivos específicos del mismo, entre los que se puede mencionar:

- Determinar las características mínimas de una herramienta de apoyo a la gestión de proyectos dirigida por un proceso utilizando como referencia el estudio de metodologías y estándares utilizados en la gestión de proyectos.
- Establecer un conjunto inicial de métricas que son obtenidas a partir de la herramienta de apoyo para el proceso de gestión de proyectos de la empresa.
- Se ha diseñado y analizado la herramienta apoyo a la gestión de proyectos.
- Se ha implementado una primera versión de la herramienta que permite obtener de forma oportuna las métricas definidas y que es utilizada por la totalidad de los proyectos en la organización.

Al cumplir los objetivos propuestos se puede concluir además que se han alcanzado los resultados esperados

- Gestión y control de los proyectos de desarrollo y mantención en forma cuantitativa.
- Evaluación del progreso de los proyectos, con respecto a la estimación versus el esfuerzo.

- Evaluación del tiempo y esfuerzo real para realizar las actividades del proceso de desarrollo de software.
- Evaluación del tiempo y esfuerzo real para el desarrollo del proyecto.
- Evaluación de la adherencia de los proyectos al proceso definido por la organización.
- Generación de información para facilitar la mejora y adaptación de los procesos de la organización.
- Generación de información histórica para facilitar la evaluación y estimación de futuros proyectos.
- Mayor entendimiento y utilización del proceso organizacional por parte de todos los involucrados.

5.2 Lecciones Aprendidas

A lo largo de este proyecto de tesis, y producto del trabajo realizado, se han podido recoger una serie de lecciones aprendidas y buenas prácticas que pueden ser utilizadas como un activo significativo de conocimiento para futuros proyectos de estas características. A continuación se presentan algunas de estas experiencias.

- Antes de comenzar este proyecto de tesis era oportuno evaluar con la organización la pertinencia de desarrollar una herramienta de gestión de proyecto en forma interna, versus alternativas tales como adquirir un software existente en el mercado, o incorporar nuevas funcionalidades a herramientas de software libre. Si bien esta evaluación no es parte del presente trabajo, es oportuno mencionar algunas de las razones por las cuales se tomó la decisión de desarrollar la herramienta y cuál fue su impacto posterior.
 - El principal beneficio era la posibilidad de personalizar y ajustar las funcionalidades de la herramienta a las necesidades y cultura de la organización, entregando mayor control sobre aquellos aspectos que eran más relevantes.
 - Por otro lado, la implementación del proceso APF ha permitido a la organización adquirir un nivel de conocimiento y experiencia significativos respecto de los diferentes estándares, prácticas y actividades referentes a la gestión de proyectos de software. Esto, sumado a la experiencia en el desarrollo de software, da la posibilidad de desarrollar una herramienta a la medida, en plazos acotados y a bajo riesgo.

- Finalmente, el desarrollo de una herramienta propia abre la posibilidad a futuro de transformarlo en una oportunidad de negocio para la organización, a través de un producto comercial.
- Uno de los aspectos fundamentales para el éxito del proyecto fue la adecuada difusión del proyecto y el involucramiento de todas las personas de la organización. Si bien esto representó un esfuerzo extra para la organización, repercutió positivamente en la disposición de las personas hacia la adopción de la herramienta y la mitigación de la resistencia al cambio.
- En este trabajo se logró concluir el aporte de la herramienta en la entrega de métricas en forma oportuna. Sin embargo, quedó en evidencia la necesidad de crear un hábito y disciplina por parte de los participantes para que registren la información requerida por la herramienta en forma correcta y consistente. Esto no es posible de controlar por el sistema, sino que depende exclusivamente de la capacidad de la organización para capacitar y promover su correcto uso y las ventajas que eso conlleva.
- Además de promover y fomentar el uso de la herramienta en forma constante, es necesario implementar formalmente un adecuado seguimiento de estas actividades. Por ejemplo, en momentos de estrés es frecuente que se dejen de lado actividades que no son percibidas como críticas, tales como el registro de horas de las actividades desarrolladas en el día. Esto repercute directamente en las métricas entregadas por el sistema y el aporte que estas puedan realizar en la toma de decisiones.

5.3 Trabajo Futuro

Aún cuando se logró cumplir con los objetivos propuestos, quedan pendientes un conjunto de temas, que dado el alcance de este trabajo, no fue posible abordar, y que podrían ser tomados en consideración a futuro, a modo de mejorar el sistema propiamente tal y prestar un mejor apoyo en la gestión de proyectos de la organización. A continuación se presentan algunos de estos temas y su posible impacto.

Implementación de nuevas funcionalidades

Si bien en este trabajo se implementaron varias de las características sugeridas para una herramienta de este tipo, aún queda un conjunto importante de características que podrían ser consideradas en futuras versiones del sistema. En particular aquellas referidas a la administración de riesgos y establecer estimaciones.

Por otra parte, en la medida que el uso del sistema se fue haciendo extensivo por parte de todos los proyectos de la organización, se ha logrado obtener una vasta retroalimentación

de la cual han surgido una serie de sugerencias de mejoras y nuevas funcionalidades que son factibles de implementar, entre las que se destacan las siguientes:

- Importar y exportar la carta Gantt en formato Microsoft Project
- Registrar detalle de la instalación de versiones del sistema en cada ambiente
- Reporte de horas por integrantes
- Incluir filtros de consulta en listado de requerimientos e incidencias
- Exportar a Excel listado de requerimientos e incidencias

Esta información es muy valiosa, ya que la fuente de estas sugerencias son los mismos usuarios del sistema.

Incorporar capacidades predictivas

En la medida que el sistema es utilizado, se está generando una cantidad importante de información que, analizada correctamente, permitiría entender el rendimiento del proceso APF y la calidad de los productos entregados. Esto permitiría predecir el rendimiento del proceso y la calidad de los productos en el futuro.

Implementación de nuevas métricas

Una de las ventajas que ofrece el sistema es que, al contemplar un amplio espectro de funcionalidades de la gestión, se convierte además en un repositorio de datos que puede ser utilizado para obtener nuevas métricas y a distintos niveles de granularidad. Esto se convierte en una oportunidad para implementar nuevas métricas orientadas a la mejora del proceso.

Creación de nuevas métricas

Complementando el punto anterior, sería interesante abordar la posibilidad de que los usuarios del sistema pudieran crear nuevas métricas en la herramienta, dependiendo de sus necesidades, sin la necesidad de que estas sean implementadas por un desarrollador. Existen herramientas, tales como *MicroStrategy Analytics*⁶, que permiten la creación de métricas directamente en la aplicación a partir de plantillas o funciones matemáticas.

⁶ <http://www.microstrategy.com>

Bibliografía

- [1] BROWNSWORD L. y GALLAGHER B. 2001. The Rational Unified Process and the Capability Maturity Model - Integrated Systems/Software Engineering. [en línea] <<http://resources.sei.cmu.edu/library/asset-view.cfm?assetid=19908>> [consulta: 23 septiembre 2014].
- [2] CALVO-MANZANO J., GARCÍA I. y ARCILLA M. 2008. Hacia la gestión cuantitativa en la gestión de proyectos en el ámbito de las pymes. REICIS, Revista Española de Innovación, Calidad e Ingeniería del Software 4(2):7-19.
- [3] CANFORA G., GARCÍA F., PIATTINI M., RUIZ F. y VISAGGIO C. A. 2005. A family of experiments to validate metrics for software process models. Journal of Systems and Software 77(2):113-129.
- [4] CMMI PRODUCT TEAM. 2010. CMMI for Development, Version 1.3. Pittsburgh, USA, Software Engineering Institute, Carnegie Mellon University. 482p.
- [5] DASKALANTONAKIS M. 1992. A Practical View of Software Measurement and Implementation Experiences Within Motorola. IEEE Transactions on Software Engineering 18(11):998-1010.
- [6] EBERT C. y DUMKE R. 2007. Software Measurement: Establish – Extract – Evaluate - Execute. Secaucus, NJ, USA, Springer-Verlag New York, Inc. 561p.
- [7] FOWLER M. 2002. Patterns of Enterprise Application Architecture. Boston, MA, USA, Addison-Wesley Longman Publishing Co., Inc. 560p.
- [8] FRAPPIER M. y RICHARD M. 2004. SMP: A Process-Driven Approach to Project Management. En: PROCEEDINGS OF the Proceedings of the 37th Annual Hawaii International Conference on System Sciences: 5 – 8 enero de 2004. IEEE Computer Society, Washington, DC, USA.
- [9] HUMPHREY W. 1989. Managing the Software Process. Boston, MA, USA, Addison-Wesley Longman Publishing Co., Inc. 512p.
- [10] INTERNATIONAL INSTITUTE OF BUSINESS ANALYSIS. 2009. A Guide to the Business Analysis Body of Knowledge (BABOK Guide), 2da ed. Toronto, Canada. International Institute of Business Analysis. 272p.
- [11] ISO. 2008. Norma Internacional ISO 9001:2008, Sistemas de gestión de calidad – Requisitos. 4ta ed. Ginebra, Secretaria General de ISO. 27p.
- [12] JACOBSON I., BOOCH G. y RUMBAUGH J. 1999. El Proceso Unificado de Desarrollo de Software. Boston, MA, USA Addison-Wesley Longman Publishing Co., Inc. 512p.

- [13] KROLL, P. y KRUCHTEN, P. 2003. The Rational Unified Process Made Easy: A Practitioner's Guide to the RUP. Boston, MA, USA, Addison-Wesley Longman Publishing Co., Inc. 464p.
- [14] KRUCHTEN P. 1995. The 4 + 1 View of Architecture. Software, IEEE 12(6):42-50.
- [15] KRUCHTEN, P. 2003. The Rational Unified Process: An Introduction. 3rd ed. Boston, MA, USA, Addison-Wesley Longman Publishing Co. 336p.
- [16] OKTABA H., PIATTINI M., PINO F., OROZCO M.J. y ALQUICIRA C. 2008. Competisoft: Mejora de procesos de software para pequeñas y medianas empresas y proyectos. Madrid, Ra-Ma Editorial. 284p.
- [17] PFLEEGER S. 1997. Assessing Software Measurement. IEEE Software Magazine 14(2):25-26.
- [18] PIATTINI M., GARCÍA F., GARZÁS J. y GENERO M. 2008. Medición y estimación del software – Técnicas y métodos para mejorar la calidad y la productividad. Madrid, Ra-Ma Editorial. 332p.
- [19] PINTO F. 2011. Introducción al CMMI. [diapositiva] 1 apunte del curso “Introducción al modelo de madurez CMMI” del diplomado de Calidad de Software del Departamento de Ciencias de la Computación de la Universidad de Chile.
- [20] PROJECT MANAGEMENT INSTITUTE. 2008. Guía de los Fundamentos para la Dirección de Proyectos (Guía del PMBOK). 4ta ed. Project Management Institute. 467p.
- [21] PROYECTO TUTELKAN. 2009. Tutelkan Reference Process (TRP), parte 1: Conceptos y Guía Introductoria [en línea] < <http://www.tutelkan.info> > [consulta: 23 septiembre 2014].
- [22] PROYECTO TUTELKAN. 2009. Tutelkan Reference Process (TRP), parte 2: TRP Básico [en línea] < <http://www.tutelkan.info> > [consulta: 23 septiembre 2014].
- [23] PUTNAM L. y MYERS W. 2003. Five Core Metrics – The Intelligence Behind Successful Software Management. New York, NY, USA, Dorset House Publishing. 328p.
- [24] SHERER W. y THRASHER S. 2005. Contrasting CMMI and the PMBOK. [en línea] <<http://dtic.mil/ndia/2005cmmi/>> [consulta: 23 septiembre 2014].
- [25] SOFTWARE ENGINEERING INSTITUTE. 2011. CMMI for Development v1.3. Carnegie Mellon University. 482p.

- [26] SEJUN K., OKJOO C. y JONGMOON B. 2001. Quantitative Project Management Framework via Integrating Six Sigma and PSP/TSP. *CrossTalk - The Journal of Defense Software Engineering* 24(4):6-10.
- [27] UTTANGI R. y AZEEM R.S.A. 2007. Fast track to CMMI implementation: Integrating the CMMI and RUP process frameworks. [en línea] IBM Developer Works <http://www.ibm.com/developerworks/rational/library/oct07/uttangi_rizwan/> [consulta: 23 septiembre 2014].
- [28] VÁSQUEZ D., PARDO C., COLLAZOS C. y PINO F. 2010. Modelo liviano de medidas para evaluar la mejora de procesos de desarrollo de software MLM-PDS. *Ingeniería y Ciencia* 6(12):171-202.
- [29] VIDAL R. 2013. Métricas para los Procesos de Gestión e Ingeniería de Proyectos de Desarrollo de Software para una PYME. Magister en Tecnologías de la Información. Santiago, Universidad de Chile, Facultad de Ciencias Físicas y Matemáticas. 78p.

Anexos

Anexo A - Áreas de Proceso APF

Procesos de Gestión

En esta categoría se encuentran las siguientes áreas de proceso:

I. Administración de Requerimientos (REQM)

El propósito de este proceso es administrar los requerimientos de los productos del proyecto y los componentes de los productos, e identificar las inconsistencias entre esos requerimientos, los planes del proyecto y los productos de trabajo. Para determinar el estado de las actividades de la Administración de Requerimientos se debe medir y analizar:

- El esfuerzo y tiempo utilizados por cada tarea de Administración de Requerimientos.
- El estado de los requerimientos asignados.
- El número de nuevos requerimientos y cambios generados durante el proyecto.

Objetivos

- Administrar los requerimientos e identificar las inconsistencias de éstos con el plan de proyecto y los productos de trabajo.

II. Planificación de Proyecto (PP)

Este proceso busca establecer planes razonables para la ejecución de las tareas y la administración del proyecto, estimando los recursos necesarios, estableciendo los compromisos y definiendo el plan para realizar el trabajo. Dentro de la planificación del proyecto se establece un plan para la actualización, distribución y el control de todos los documentos del proyecto.

Objetivos

- Establecer y mantener parámetros de planificación del proyecto.
- Establecer y mantener un plan de proyectos como la base para administrar el proyecto.
- Establecer y mantener compromisos con el plan de proyecto.

III. Monitoreo y Control del Proyecto (PMC)

El propósito de este proceso es otorgar una adecuada visibilidad respecto al avance del proyecto para permitir la toma de acciones correctivas cuando el proyecto se desvía significativamente de los planes.

Objetivos

- Monitorear el rendimiento real y el avance del proyecto contra el plan del proyecto.
- Gestionar acciones correctivas cuando el rendimiento del proyecto o los resultados se desvían significativamente del plan.

IV. Administración de Acuerdo con Proveedores (SAM)

Es la administración de todo producto o servicio adquirido a través de un proveedor.

Objetivos

- Establecer y mantener acuerdos con los proveedores.
- Satisfacer, tanto internamente como con el proveedor, los acuerdos establecidos entre las partes.

Procesos de Ingeniería

En esta categoría se encuentran las siguientes áreas de proceso:

I. Desarrollo de Requerimientos (RD)

Este proceso captura y entiende las necesidades del cliente para poder determinar claramente el deseo y las condiciones que los involucrados poseen respecto al producto a desarrollar para traducirlo a requerimientos de productos y componentes de productos, estableciendo el alcance y los límites del sistema a construir.

Objetivos

- Definir y profundizar el alcance del proyecto.
- Analizar, describir y validar los requerimientos del proyecto.
- Evaluar y estimar cambios de requerimientos y su impacto en el proyecto.

II. Solución Técnica (TS)

Diseñar, desarrollar e implementar (incluyendo pruebas unitarias) soluciones a los requerimientos del cliente.

Objetivo

- Evaluar y seleccionar soluciones que potencialmente satisfagan un conjunto adecuado de requerimientos.
- Desarrollar detalladamente los diseños para las soluciones seleccionadas.
- Implementar los diseños del producto o componente.

III. Integración del Producto (PI)

Indicar y planificar las actividades que son necesarias para ensamblar el sistema a partir de sus componentes, asegurando que el sistema una vez integrado funciona correctamente. Se debe definir la estrategia del paso a producción y poner el sistema a disposición de los usuarios.

El alcance de esta área de proceso es lograr la integración del sistema completo a través de un ensamblaje progresivo de los componentes, en una etapa o varias etapas incrementales, de acuerdo a una secuencia y procedimientos de integración definida.

Objetivos

- Planificar y preparar el proceso de instalación.
- Identificar los componentes de software que serán integrados para la instalación.
- Integrar los componentes y probar su acoplamiento.
- Configurar y preparar el ambiente donde se instalarán los componentes.
- Instalar los componentes integrados en producción.

IV. Pruebas

El propósito de este proceso es evaluar que los productos o componentes satisfacen los requerimientos (verificación) y demostrar que cumple con el uso propuesto cuando es entregado al cliente (validación).

Objetivos

- Seleccionar los productos a ser verificados y los métodos utilizados para verificar cada uno
- Seleccionar los productos a ser validados y los métodos utilizados para validar cada uno

Procesos de Soporte

En esta categoría se encuentran las siguientes áreas de proceso:

I. Medición y Análisis (MA)

Desarrollar y mantener una capacidad de medición para dar soporte a la administración de información necesaria, y así contar con datos relevantes a la hora de tomar decisiones.

Objetivos

- Alinear los objetivos y actividades de medición con las necesidades y objetivos Identificados de información.
- Proveer resultados de las mediciones que se enfoquen en las necesidades y objetivos de información.

II. Administración de Configuraciones (CM)

Establecer y mantener la integridad de los productos de trabajo identificando las líneas base de la configuración, llevando controles sobre ésta y sobre el estado de la misma, y además realizando auditorias que permitan servir como apoyo a los procesos de creación de software.

Objetivos

- Establecimiento de las líneas base de la configuración del sistema.
- Mantener el control sobre la configuración y realizar seguimientos a los cambios que ésta pueda sufrir.
- Establecer y mantener la integridad de las líneas base.

III. Aseguramiento de Calidad de Procesos y Productos (PPQA)

Proveer al personal y a los administradores (o la gerencia) de una visión objetiva del proceso y los productos de trabajo asociados, revisando y auditando los productos y actividades para verificar el cumplimiento con los procedimientos y estándares e informando de los resultados de las revisiones y auditorias.

Objetivos

- Evaluar objetivamente procesos y productos de trabajo y servicios.
- Comunicar y asegurar la resolución de los incumplimientos y establecer registros.

Uso del Proceso APF

El ciclo de vida del proceso APF define cuatro fases de desarrollo: Inicio, Elaboración, Construcción y Transición. En cada una de ellas se encuentran descritos los objetivos que deben cumplirse y las actividades definidas para ello. Independiente de si el proyecto es nuevo o de mantención, debe seguirse la misma estructura de trabajo. Basado en un desarrollo iterativo incremental, el propósito es que cada incremento pase por todas las fases de desarrollo, o bien que cada fase se componga de iteraciones en las cuales cada incremento es una mejora de la iteración precedente.

Fase de Inicio

Durante la fase de inicio se define el modelo del negocio, el alcance del proyecto y el dominio del problema. Se identifican todos los actores y Casos de Uso, y se diseñan, de acuerdo a las prioridades del negocio y desde lo técnico, los Casos de Uso más esenciales. En base a éstos, se deben proporcionar arquitecturas posibles de desarrollo. Se evalúa la estimación de riesgos y todos los elementos relacionados con estimación de costos. Se desarrolla la planificación del proyecto para determinar cuáles son las

actividades y qué recursos deben ser asignados. Se comunica el plan a todas las partes involucradas.

Hay cinco objetivos para la fase de Inicio:

- **Entender qué construir.** Establecer el ámbito y la visión del proyecto a construir y sus límites, además de identificar los interesados (*stakeholders*) del proyecto.
- **Identificar la funcionalidad clave del sistema.** Encontrar los Casos de Uso críticos del sistema, los escenarios básicos que definen la funcionalidad.
- **Planificar.** Iniciar la planificación del proyecto con todas las actividades involucradas en el desarrollo del software.
- **Determinar al menos una posible solución.** Mostrar al menos una arquitectura candidata para los escenarios principales y su factibilidad.
- **Estimar el costo.** Estimar el costo en recursos y tiempo de todo el proyecto, estimar los riesgos y estrategias de contingencia y mitigación.

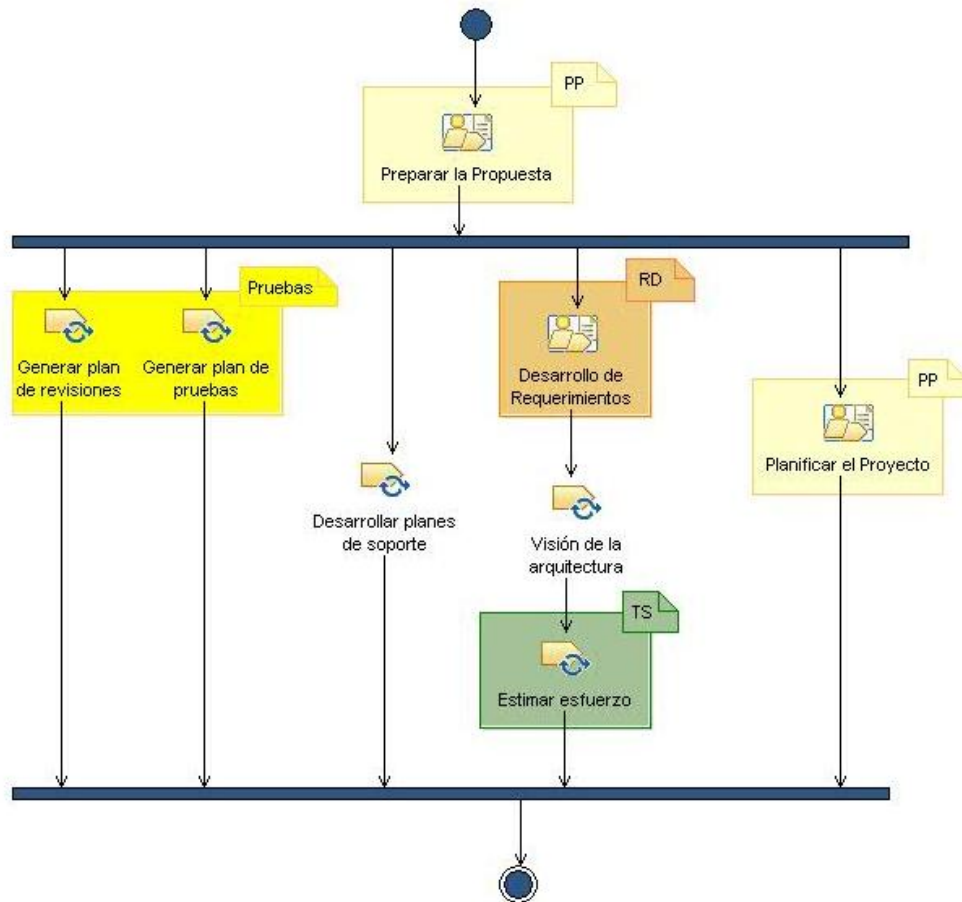


Figura 49. Flujo de Trabajo de la Fase de Inicio

Fase de Elaboración

En esta fase se agrupan las actividades tendientes a diseñar y construir el sistema. Se tiene como prioridad identificar el modelo del negocio y el dominio del problema. Se identifican los principales casos de uso, se actualizan los riesgos, los costos y se complementa el plan de proyecto. Además, se debe proporcionar arquitecturas posibles de desarrollo y a partir de éstas, construir los prototipos pertenecientes a los módulos principales del sistema. Los prototipos son realizados utilizando los casos de uso seleccionados (sin perjuicio que aparezcan otros), y se abordan los riesgos más críticos.

Hay tres objetivos para la fase de Elaboración:

- **Obtener un entendimiento más detallado de los requerimientos.** Asegurarse de ganar profundidad en el entendimiento de los requerimientos más críticos a ser validados por la arquitectura.
- **Describir la arquitectura del software.** Definir y validar la arquitectura. Diseñar, implementar y probar un esqueleto estructural del sistema. Demostrar que la arquitectura propuesta soportará la visión y los requerimientos del producto.
- **Mitigar los riesgos críticos.** De acuerdo a la lista de riesgos definida en la fase de inicio, se debe actualizar la magnitud de éstos en base al desarrollo de la arquitectura, análisis y diseño.

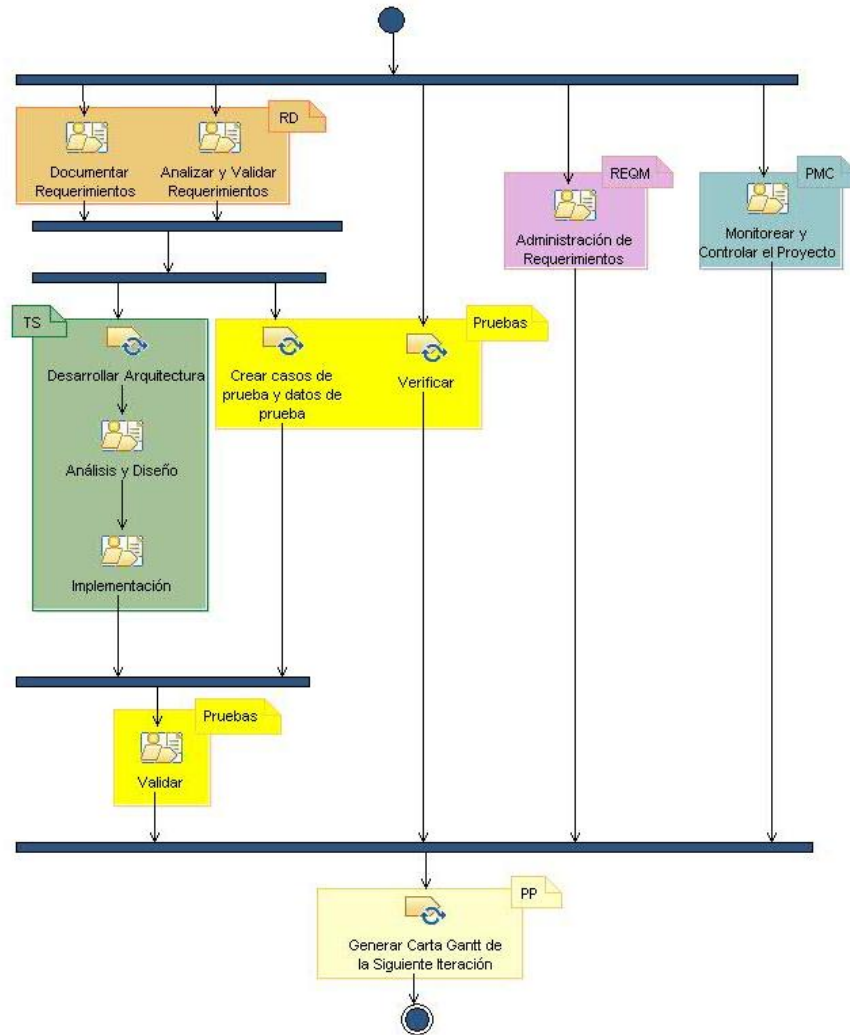


Figura 50. Flujo de Trabajo de la Fase de Elaboración

Fase de Construcción

El objetivo de esta fase es implementar el producto de forma incremental a través de sucesivas iteraciones. Durante esta fase todos los componentes, características y requisitos deben ser implementados, integrados y probados en su totalidad, obteniendo una versión aceptable del producto.

Hay dos objetivos para la fase de Construcción:

- **Desarrollar iterativamente un producto.** Conseguir versiones funcionales tan rápido como sea práctico. Describir los requerimientos restantes, completar en detalles los diseños, completar la implementación y pruebas del software.

- **Minimizar los costos de desarrollo.** Mediante la optimización de recursos y evitar tener que rehacer un trabajo o incluso desecharlo. Asignar componentes que puedan ser desarrollados independientemente uno del otro.

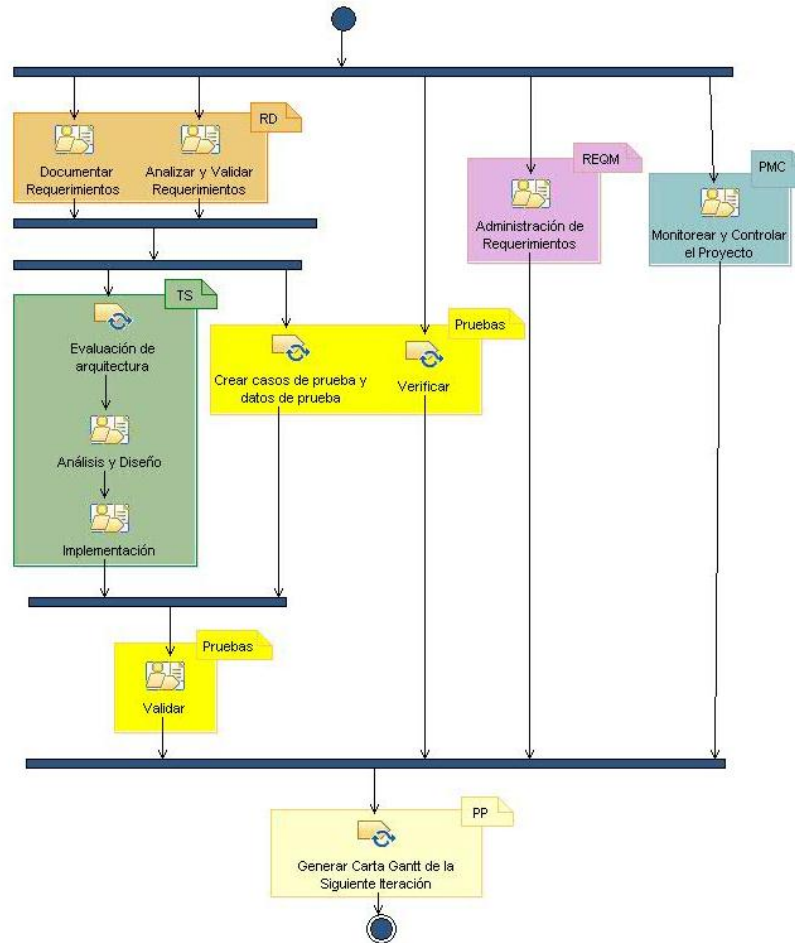


Figura 51. Flujo de Trabajo de la Fase de Construcción

Fase de Transición

La finalidad de esta fase es poner el producto en manos de los usuarios finales, para lo que se requiere desarrollar nuevas versiones actualizadas del producto, completar la documentación, entrenar al usuario en el manejo del producto, y, en general, tareas relacionadas con el ajuste, configuración, instalación y facilidad de uso del producto.

Hay dos objetivos para la fase de Transición:

- **Producto final que opere de acuerdo a los deseos del usuario.** Un producto final que cumpla los requerimientos esperados, que funcione y satisfaga suficientemente al usuario final.

- **Mejorar el desempeño en futuros proyectos.** Documentar las lecciones aprendidas y mejorar el ambiente de los procesos y las herramientas para el desarrollo del proyecto.

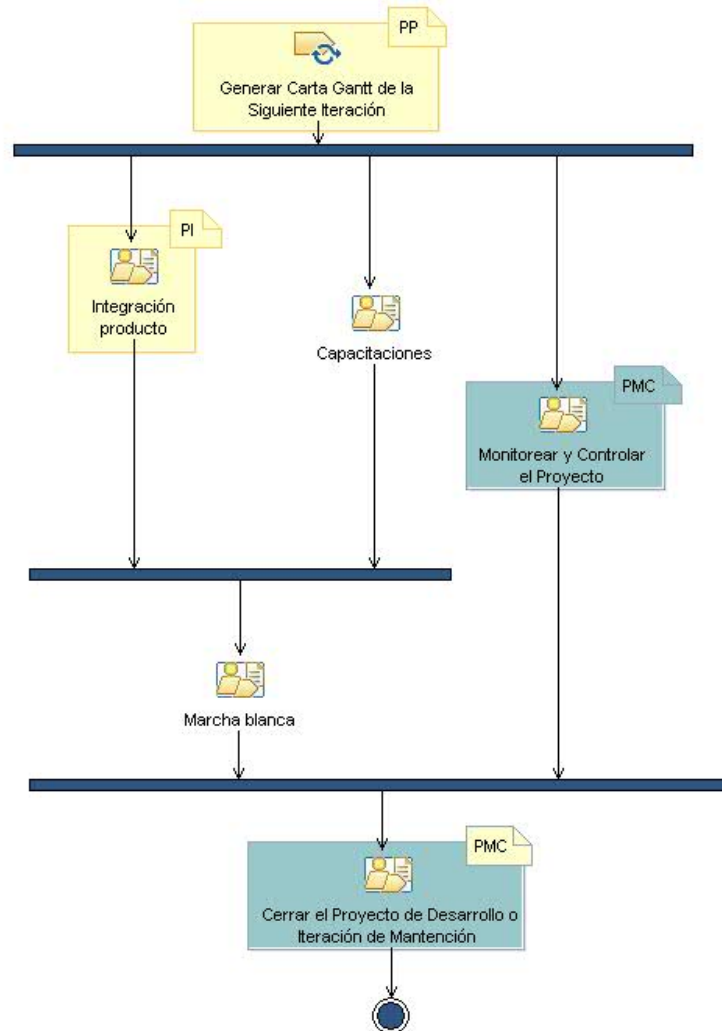


Figura 52. Flujo de Trabajo de la Fase de Transición

Anexo B – Métricas de Amisoft

A continuación se presenta la definición de las métricas propuestas por Vidal, y que son actualmente utilizadas en Amisoft [29].

Adherencia al proceso

Tabla 16. Métrica adherencia al proceso

Identificador	Porcentaje de adherencia al proceso
Definición	La adherencia al proceso de acuerdo a las revisiones realizados por el PPQA.
Metas	Determinar la adherencia a las áreas de procesos donde se aplicarán las métricas. Se pretende poder llegar a obtener al menos un 90% de adherencia.
Métricas	Ea: estado de los artefactos revisados: Completado, Incompleto, No existe y Persiste Incumplimiento. Ta: número de artefactos comprometidos para la revisión
Periodicidad	Semanalmente se revisan los artefactos obligatorios y aquellos generados de acuerdo a la planificación, al final de cada fase se realiza una auditoría completa de todos los artefactos a fin de mantener su consistencia.
Procedimiento de Análisis	Calcular los % de completitud de los artefactos comprometidos por el proyecto, solo se consideran para efectos de la métrica de adherencia aquellos que están en estado completado: $\% = (Ea / Ta) * 100$
Forma de Presentación	Gráfico circular por proyecto y gráfico de barras para la comparación de los proyectos pilotos para la tesis.
Herramientas	Planilla de Verificación de proceso (PPQA)
Responsabilidades	Jefe de Proyectos, Analista de Métricas

Pruebas Funcionales

Tabla 17. Métrica pruebas funcionales

Identificador	Pruebas Funcionales
Definición	Demostrar que el producto cumple con los requisitos de usuario al ejecutar los casos de pruebas planificados para el producto.
Metas	Alcanzar el 90% de éxito de las pruebas funcionales en los requisitos del
Métricas	<ul style="list-style-type: none"> Número de casos de pruebas ejecutados Número de defectos reportados
Periodicidad	Cada vez que se realice un paso al ambiente de testing.
Procedimiento de análisis	<p>Calcula el porcentaje de los casos de prueba ejecutados y que no tengan un defecto reportado.</p> $\%PF = 100 - (Nd * 100) / Ncpe$ <p>Donde, Nd es el número de defectos reportados y Ncpe es el número de casos de prueba ejecutados.</p> <p>PF: Porcentaje de pruebas funcionales</p>
Forma de presentación	Gráfico circular para la iteración y gráfico de barra cuando exista más de una iteración en el proyecto
Herramientas	Planilla de Métricas, casos de prueba
Responsabilidades	Analista QA, Analista de Métricas

Volatilidad de Requerimientos

Tabla 18. Indicador Volatilidad Requerimientos del Plan de Medición de Amisoft

Indicador	Volatilidad de los requerimientos
Definición	Medir la forma como han variado en porcentaje los requerimientos originalmente propuestos a los ejecutados en el proyecto.
Metas	Determinar la volatilidad de los requerimientos (Cambios de los requerimientos)
Métricas	TR : número total de requerimientos

Periodicidad	TN : número total de requerimientos nuevos TM : número total de requerimientos modificados TC : número total de requerimientos cancelados TT : número total de requerimientos terminados
	En proyectos de desarrollo se hará al término de la fase de elaboración de cada proyecto, para los proyectos de mantención se realizará al término de la iteración.
Procedimiento de análisis	Calcular el % de variabilidad de los requerimientos $\%RN = (TN/TR)*100$ indica el % de requerimientos nuevos $\%RM = (TM/TR)*100$ indica el % de requerimientos modificados $\%RC = (TC/TR)*100$ indica el % de requerimientos cancelados $\%TT = (TT/TR)*100$ indica el % de requerimientos terminados
Forma de Presentación	Visualizar los resultados a nivel de gráfico circular y tabla.
Herramientas	Planilla de Métricas, Formulario de Requerimiento de Usuario (Project.net)
Responsabilidades	Jefe de Proyecto, Analista de Métrica

Valor Ganado

Tabla 19. Métrica valor ganado

Indicador	Valor Ganado
Definición	Compara la cantidad de trabajo ya completado con respecto a lo planificado para un momento específico del proyecto. Se contabiliza además el esfuerzo en HH realizado para lograr esa cantidad de trabajo terminado.
Metas	Facilitar la evaluación del progreso del proyecto de acuerdo al plan estimado. Conocer el esfuerzo realizado para lograr los avances del proyecto a fin de mejorar las estimaciones.
Métricas	BAC : línea base de presupuesto de HH PV: valor presupuestado en HH de todas las tareas que fueron

	<p>planeadas empezar y terminar en el momento del análisis.</p> <p>EV: valor ganado en HH que representa la suma de todas las tareas terminadas en el momento del análisis.</p> <p>AC: esfuerzo o costo en HH realizado para terminar las tareas al momento del análisis.</p>
<p>Periodicidad</p>	<p>Para proyectos de mantención el punto de control será semanal.</p> <p>Para proyectos de desarrollo se definirá de acuerdo al tamaño y complejidad del proyecto.</p>
<p>Procedimiento de análisis</p>	<p>Variaciones</p> <p>Las variaciones nos muestran cuánto nos estamos desviando del cronograma y costos presupuestados para una fecha de control específica.</p> <p>Variación del costo CV</p> <p>Nos permite identificar si estamos por encima o por debajo del valor planeado en HH a la fecha de control y en qué cuantía. Un valor negativo indica que nos estamos excediendo en el presupuesto y un valor positivo indica lo contrario.</p> <p>$CV = EV - AC$</p> <p>Variación del Cronograma SV</p> <p>Nos indica qué tan adelantados o atrasados estamos en nuestro cronograma. SV compara el trabajo realizado EV con el valor planeado PV. Un valor negativo indica que estamos atrasados y un valor positivo indica lo contrario.</p> <p>$SV = EV - PV$</p> <p>CV y SV son valores absolutos y si bien indican condiciones favorables y desfavorables, nos dan idea de magnitud. Las fórmulas siguientes indican el valor relativo de la situación:</p> <p>$SV\% = SV/PV$, este % indica cuánto atraso o adelanto llevamos</p>

con respecto al cronograma planteado.

$CV\% = CV/EV$, este % nos indica cuán excedidos o por debajo de la línea de base del presupuesto estamos.

Índices

La medición del valor ganado define índices de rendimiento normalizado para que se pueda comparar valores en distintos tiempos en proyectos o valores para proyectos de tamaños distintos.

Los índices muestran qué estamos logrando hacer con nuestros recursos para una fecha de control específica. Son útiles para poder comunicar una valoración objetiva del estado del proyecto. Los índices mayores a la unidad son favorables y los menores a la unidad son desfavorables.

Índice de desempeño del Costo (presupuesto)

Es una medida del valor ganado de un proyecto comparada a los costos reales incurridos. Si el índice es igual a 1, entonces el proyecto está de acuerdo al presupuesto planeado. Si el índice es mayor a 1 entonces el proyecto está por debajo del presupuesto planeado. Si el índice es menor a 1 el proyecto está por encima del presupuesto planeado.

$$CPI = EV/AC$$

Índice de desempeño del Cronograma

Es una medida de progreso real del cronograma del proyecto. Si el índice es igual a 1, entonces el proyecto está en el cronograma. Si el índice es mayor que 1 entonces el proyecto está adelantado de acuerdo al cronograma. Si el índice es menor que 1 entonces el proyecto está atrasado de acuerdo al cronograma.

$$SPI = EV/PV$$

Proyecciones

La proyección estimada para concluir el proyecto a la fecha de control será:

$$EAC = AC + ETC, \text{ donde}$$

$$ETC = (BAC - EV) / CPI$$

Índice TCPI

El índice TCPI nos da una estimación de cuál debería ser nuestro índice de desempeño para terminar como presupuestamos originalmente.

$$TCPI = (BAC - EV) / (BAC - AC)$$

Índice Cronograma SPI	Costo-CPI x	Cronograma		
		SPI > 1.0	SPI = 1.0	SPI < 1.0
Esfuerzo	CPI > 1.0	<i>Adelantado en plazo y bajo el esfuerzo</i>	<i>En el plazo y bajo el esfuerzo</i>	<i>Atrasado en plazo y bajo el Esfuerzo</i>
	CPI = 1.0	<i>Adelantado en plazo y dentro del esfuerzo</i>	<i>En plazo y dentro del esfuerzo</i>	<i>Atrasado en plazo y dentro del esfuerzo</i>
	CPI < 1.0	<i>Adelantado en plazo y sobre el esfuerzo</i>	<i>En plazo y sobre el esfuerzo</i>	<i>Atrasado en plazo y sobre el esfuerzo</i>

Forma de Presentación

Visualizar los resultados a nivel de cuadro resumen y gráfico de líneas.

Herramientas

Planilla de métricas y planilla de evaluación de estado.

Responsabilidades

Jefe de Proyecto, Analista de Métrica