



UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

FORMALIZACIÓN DE UN PROCESO DE DESARROLLO DE
SOFTWARE PARA UNA PEQUEÑA EMPRESA

TESIS PARA OPTAR AL GRADO DE MAGISTER EN TECNOLOGÍAS
DE LA INFORMACIÓN

OMAR ANTONIO NEYRA RODRÍGUEZ

PROFESOR GUÍA:
ALEJANDRO LAGOS SANTELICES
PROFESOR GUÍA 2:
JOSÉ PINO URTUBIA

MIEMBROS DE LA COMISIÓN:
MARÍA CECILIA BASTARRICA PIÑEYRO
NELSON BALOIAN TATARYAN
LUIS SILVESTRE QUIROGA

SANTIAGO DE CHILE
2021

Resumen

Una de las características diferenciadoras clave de toda empresa pasa por la calidad de los productos que entregue a sus clientes. La obtención de un producto de calidad es el resultado de variadas características propias de cada empresa. Obtener este tipo de productos puede ser aún más complejo si, entre otros, la cantidad de participantes durante el proceso se incrementa, el tamaño de los proyectos crece o se requiere abordar múltiples proyectos de forma paralela.

Nexo-IT Consultores SpA es una PYME que desarrolla software a medida en distintas áreas y para diversos clientes, durante los últimos años dentro de la compañía se han tornado evidentes algunos problemas internos que se reflejan en el atraso en la entrega de las soluciones y en la cantidad de incidencias que se levantan durante las pruebas de éstas. Lo anterior impacta directamente en los plazos comprometidos incrementando los costos asociados y reduciendo así las ganancias proyectadas.

Debido a la cantidad y magnitud de los proyectos abordados en la actualidad es que la gerencia decidió iniciar un proceso mediante el cual se defina y adopte un proceso de trabajo que permita abordar de la mejor forma los proyectos requeridos por sus clientes y que es el objetivo abordado en esta Tesis.

Al iniciar el trabajo fue necesario regenerar los datos de proyectos históricos a partir de la información dentro de los sistemas del cliente, esto debido a la inexistencia de estos datos dentro de la compañía. También se trabajó catalogando las actividades ejecutadas en la actualidad, recolectadas a través de reuniones de trabajo junto a los jefes de proyectos. Luego de lo anterior, y según lo planificado, se procedió a estudiar el estándar propuesto en la ISO/IEC 29110 y el proceso definido en OpenUP que fueron tomados como base durante la construcción del proceso desarrollado.

Luego de haber definido el conjunto de actividades, tareas y roles se optó por especificar el proceso usando SPEM 2.0 mediante el uso de EPF Composer. Durante el transcurso del desarrollo se volvió evidente la necesidad de evaluar el proceso diseñado, actividad que no se había planificado inicialmente, por lo que se optó usar la herramienta AVISPA, herramienta elegida debido a la abundante documentación existente y a su característica de tomar como base de información datos obtenidos desde empresas chilenas.

Al final de este trabajo se obtuvo un proceso especificado formalmente, proceso que en el transcurso de este trabajo debió ser ajustado debido a las alarmas levantadas durante la etapa de evaluación y obtenidas gracias a la herramienta AVISPA. Finalmente se ejecutaron tres proyectos piloto que estuvieron disponibles al momento de este trabajo.

Si bien los objetivos definidos en este documento fueron conservadores estos no fueron logrados en su totalidad. Luego de la evaluación se concluye que el proceso definido se ajusta a la realidad de la compañía, aunque está abierto a mejoras, y que más bien la causa de este resultado se debe a la poca representatividad de los proyectos piloto abordados, esto debido principalmente a la indisponibilidad de conocimiento específico dentro de los equipos de trabajo en el momento oportuno.

Tabla de contenido

Resumen.....	i
1. Introducción.....	1
1.1 Contexto	1
1.2 Problema	1
1.3 Objetivos	2
1.4 Metodología a utilizar	3
1.4.1 Etapa de levantamiento	3
1.4.2 Etapa de construcción.....	3
1.4.3 Etapa de validación	4
2. Marco teórico.....	5
2.1 Proceso de desarrollo de software.....	5
2.2 ISO/IEC 29110 - Perfiles de ciclo de vida para entidades muy pequeñas	6
2.3 OpenUP.....	8
2.3.1 Principios.....	9
2.3.2 Componentes.....	10
2.3.3 Fases.....	10
2.4 Spem.....	12
2.4.1 Marco de uso conceptual	12
2.4.2 Arquitectura del metamodelo SPEM 2.0.....	13
2.5 EPF Composer	15
2.6 AVISPA (Analysis and Visualization for Software Process Assessment)	16
3. Definición del proceso de desarrollo.....	17
3.1 Etapa de levantamiento.....	17
3.1.1 Levantamiento de datos históricos	17
3.1.2 Identificación y catalogación de actividades.....	20
3.1.3 Análisis de las prácticas y procesos	22
3.2 Etapa de construcción	22
3.2.1 Construcción del proceso de desarrollo y prácticas asociadas	22
3.2.2 Análisis y publicación del resultado además de entregar apoyo técnico.....	24
4. Validación del proceso.....	33
4.1 Ejecución de proyectos adoptando el proceso y prácticas definidas	33
4.2 Validación de los resultados	42

4.3 Análisis de los resultados	44
5. Conclusión	46
Bibliografía.....	48
Anexos	50
Anexo A.....	50

Índice de tablas

Tabla 1: Indicadores de proyectos históricos	2
Tabla 2: Regeneración de datos históricos	18
Tabla 3: Información del proyecto Dashboard	34
Tabla 4: Información del proyecto Prepago	37
Tabla 5: Información del proyecto Excedentes	39
Tabla 6: Tabulación de proyectos históricos	42
Tabla 7: Tabulación de datos recolectados nuevos proyectos	43
Tabla 8: Resumen de los resultados	44

Índice de ilustraciones

Figura 1: Series ISO/IEC 29110.....	8
Figura 2: Procesos ISO/IEC 29110	8
Figura 3: Marco conceptual de SPEM.....	14
Figura 4: Estructura del metamodelo SPEM	15
Figura 5: Desviación estándar de incidencias para proyectos medianos	19
Figura 6: Desviación estándar de incidencias para proyectos pequeños	19
Figura 7: Desviación estándar de controles de cambio para proyectos medianos	20
Figura 8: Desviación estándar de controles de cambio para proyectos pequeños	20
Figura 9: Proceso estándar ejecutado por DA	22
Figura 10: Proceso estándar ejecutado por JZ.....	23
Figura 11: Proceso propuesto	24
Figura 12: Actividades definidas para el proceso.....	25
Figura 13: Roles aislados.....	26
Figura 14: Roles sobrecargados.....	27
Figura 15: Sub-proyectos independientes.....	28
Figura 16: Tareas de propósito múltiple.....	29
Figura 17: Sub-proyectos independientes.....	30
Figura 18: Artefactos de entrada para muchas tareas	31
Figura 19: Artefactos residuales	32
Figura 20: Proceso proyecto Dashboard.....	34
Figura 21: Tareas seleccionadas para el proyecto Dashboard	35
Figura 22: Roles involucrados en el proyecto Dashboard	35
Figura 23: Planificación proyecto Dashboard	36
Figura 24: Proceso proyecto Prepago	37
Figura 25: Tareas seleccionadas para el proyecto Prepago	38
Figura 26: Roles involucrados en el proyecto Prepago	38
Figura 27: Planificación proyecto Prepago	39
Figura 28: Proceso proyecto Excedentes.....	40

Figura 29: Tareas seleccionadas para el proyecto Excedentes	41
Figura 30: Roles involucrados en el proyecto Excedentes	41
Figura 31: Planificación proyecto Excedentes	42
Figura 32: Bitácora proyecto Dashboard.....	43
Figura 33: Bitácora proyecto Prepago	43
Figura 34: Bitácora proyecto Excedentes	44
Figura 35: Tareas dentro de Gerencia de proyectos	50
Figura 36: Detalle de la actividad “Gerencia de proyectos”	51
Figura 37: Tareas dentro de Generación de propuesta	52
Figura 38: Detalle de la actividad Generación de propuesta	52
Figura 39: Tareas dentro de Planificación de proyecto	53
Figura 40: Detalle de la actividad Planificación del proyecto	54
Figura 41: Tareas dentro de Estructuración del equipo de trabajo	54
Figura 42: Detalle de la actividad Estructuración del equipo de trabajo	55
Figura 43: Tareas dentro del Kick off	55
Figura 44: Detalle de la actividad Kick off	55
Figura 45: Tareas dentro de la Planificación del sprint	56
Figura 46: Detalle de la actividad Planificación de sprint	57
Figura 47: Tareas dentro de Cambios del sistema	58
Figura 48: Detalle de la actividad Cambios del sistema.....	58
Figura 49: Tareas dentro de Controles de cambio	59
Figura 50: Detalle de la actividad Cambios del sistema.....	59
Figura 51: Tareas dentro de Cambio de alcance.....	60
Figura 52: Detalle de la actividad Cambio de alcance	60
Figura 53: Tareas dentro de Ejecución del sprint	61
Figura 54: Detalle de la actividad Ejecución del sprint.....	61
Figura 55: Tareas dentro de Pruebas integrales.....	62
Figura 56: Detalle de la actividad Pruebas integrales.....	62
Figura 57: Tareas dentro de Paso a producción.....	63

Figura 58: Detalle de la actividad Paso a producción.....	63
Figura 59: Tareas dentro de Retroalimentación.....	64
Figura 60: Detalle de la actividad Retroalimentación	64
Figura 61: Tareas dentro de Cierre	64
Figura 62: Detalle de la actividad Cierre.....	65

1. Introducción

1.1 Contexto

Implementar las mejores prácticas reconocidas en la industria del desarrollo de software es un objetivo que muchas compañías buscan pero que puede ser difícil de lograr cuando, por ejemplo, éstas no cuentan con procesos formales establecidos que formen parte de su cultura. Es más, el forzar la implementación de estas prácticas puede provocar un desequilibrio interno que puede llevar a la compañía a una situación más compleja de la que tenía antes de iniciar el proceso.

Nexo-IT Consultores SpA es una PYME nacida el año 2007 que desarrolla software a medida en distintas áreas y para diversos clientes como bancos, cajas de compensación, entidades de servicios financieros entre otros. En la actualidad cuenta con 33 empleados de los cuales el 58% pertenecen al área de desarrollo de sistemas. En esta área la estructura organizacional es clara en cuanto a la definición de cargos, se cuenta con un gerente de operaciones, 2 jefes de proyectos y 16 implementadores.

Nexo-IT ha logrado mantener relaciones comerciales con la mayoría de sus clientes durante largo tiempo, la cartera actual está compuesta por cinco grandes empresas y una PYME, clientes que constantemente solicitan nuevos desarrollos o mantenciones a sus sistemas. La compañía entiende que la estabilidad de las relaciones comerciales se debe a las buenas interacciones personales, buenos tiempos de respuesta de soporte y altos niveles de satisfacción por parte de los clientes. La anterior evaluación es subjetiva ya que no se tienen herramientas que permitan verificar esto, siendo un tema pendiente por desarrollar.

1.2 Problema

Junto al crecimiento de los equipos de trabajo y al aumento del tamaño de las soluciones desarrolladas se han incrementado los problemas con los que la compañía debe convivir. El principal problema detectado se debe a que en la actualidad no existe una forma única de llevar adelante los proyectos, dejando esta definición en manos del jefe de proyecto a cargo, tomando éste el control total de las actividades a realizar. Este problema se refleja cuando, por ejemplo, es necesario reemplazar a los integrantes de los equipos ya que las diferencias de los procesos dificultan la adopción de estos.

También se ha detectado una baja en la calidad de los productos desarrollados, evidenciado por un incremento en la cantidad de incidencias por resolver luego de entregadas las soluciones. Esto repercute en las relaciones con los clientes ya que va en desmedro de la confianza y en la imagen de la compañía.

Otro problema que considerar son falencias en el relevamiento y explicitación de los requisitos. Esta es una tarea sensible dado que, cuando no se ejecuta con el rigor adecuado, impacta directamente en la programación del proyecto y sus costos asociados exigiendo recursos programados para otras tareas. Es común que fallas en este proceso lleven a una definición ambigua del alcance dificultando así su cierre.

La baja precisión en la estimación de esfuerzo de las tareas es otra área débil. Actualmente este proceso se lleva a cabo en base a la experiencia del equipo de trabajo a cargo, lo que en la actualidad no es suficiente para lograr control y la adecuada gestión de los proyectos.

Los puntos anteriores afectan tanto a los desarrollos nuevos como a las mantenciones de sistemas existentes y si bien ambas facetas tienen similar beneficio para la compañía es en los nuevos desarrollos donde más se nota el impacto adverso.

El problema que esta tesis aborda es la indefinición de un proceso de desarrollo de software y prácticas asociadas que presenta Nexo-IT Consultores SpA, indefinición provocada por lo expuesto anteriormente y que requiere resolución para permitir la producción de sistemas de calidad de forma controlada y predecible.

Para abordar este problema se deben considerar las siguientes definiciones:

- Los proyectos pequeños son los que tienen como máximo una duración programada de seis semanas.
- Los proyectos medianos son los que tienen una duración programada que va desde las siete hasta las dieciséis semanas.
- Tanto las incidencias como los controles de cambio se miden en puntos que representan la media de incidencias reportadas por el cliente.

Los datos presentados en la Tabla 1: Indicadores de proyectos históricos, muestran los datos regenerados de proyectos históricos.

Tipo de proyecto	Incidencias	Controles de cambio	Desviación de esfuerzo
Medianos	3	5	35%
Pequeños	1	5	26%

Tabla 1: Indicadores de proyectos históricos

1.3 Objetivos

El objetivo general de este proyecto es definir un proceso de desarrollo y prácticas asociadas que permitan la sistematización de nuevos productos con el objeto de entregar un marco de trabajo único para los proyectos de la compañía, esperando de esta forma reducir los atrasos en las entregas de estos.

Los objetivos específicos se describen a continuación y son los que permitirán evaluar el resultado de este trabajo.

1. Para los proyectos pequeños, reducir el atraso en no más de un 10% según lo planificado inicialmente.
2. Para los proyectos medios, reducir el atraso de forma que este no supere un 20% del tiempo planificado inicialmente.

3. En general, mantener o reducir en 1 punto la mediana de las incidencias.
4. En general, mantener o reducir en 1 punto la mediana de los controles de cambio.

1.4 Metodología a utilizar

Este proyecto se divide en 3 etapas las que en conjunto contienen un total de 6 actividades. Estas se describen a continuación:

1.4.1 Etapa de levantamiento

Levantamiento de datos históricos

Durante esta actividad se regeneró la información de proyectos históricos respecto a los plazos e incidencias generados, esto con el fin de realizar una comparación de resultados durante la etapa de validación. Este proceso fue necesario debido a la inexistencia de datos históricos tabulados dentro de la compañía.

Identificación y catalogación de actividades actuales

El objetivo de esta actividad es identificar y catalogar las actividades que los equipos de trabajo realizan al día de hoy durante el desarrollo de sus proyectos. También se busca identificar el ciclo de vida y el cómo estas actividades están organizadas. Estas tareas son desarrolladas en reuniones de trabajo que darán como resultado un catálogo de actividades y una descripción de los procesos formulados en el lenguaje SPEM 2.0 [15].

Análisis de las prácticas y procesos

Durante esta actividad se analizaron las prácticas recolectadas antes con el objeto de encontrar similitudes que permitan consolidarlas y así tener como resultado un catálogo consolidado. De forma similar, los procesos de cada equipo fueron comparados con el objeto de obtener un marco de trabajo general que servirá para definir un proceso formal de desarrollo que los equipos deberán adoptar en futuros desarrollos.

1.4.2 Etapa de construcción

Construcción del marco de trabajo

Tomando como entrada el catálogo consolidado de prácticas y como modelos el estándar propuesto en ISO/IEC 29110 [5] y el proceso definido en OpenUP [6] se generó un conjunto de roles, artefactos, tareas y orden en el cual los desarrollos serán ejecutados. El objetivo de esta actividad es la de generar un proceso adaptable que permita ser ajustado según los nuevos desarrollos los requieran, definiendo así un marco de trabajo común. La metodología de ciclo de vida, sea tradicional o ágil, también dependerá del tipo de proyecto y experiencia del equipo de trabajo, por lo que la elección dependerá de estas variables. La especificación del proceso será mediante el uso del lenguaje SPEM 2.0.

Análisis y publicación del resultado

Durante esta actividad se evaluó el proceso desarrollado usando la herramienta AVISPA [16], el objetivo de este análisis es detectar posibles errores o deficiencias en el proceso, características que pueden ser abordadas en caso sea necesario. Además, se presenta el resultado de la actividad de construcción a los equipos de trabajo, con el objeto de discutir éste y aplicar ajustes menores si es que se estima necesario. El objetivo principal de esta actividad es entregar el conocimiento necesario a los equipos para que éstos puedan adoptar las nuevas prácticas de la mejor forma, adaptando el nuevo proceso a sus proyectos según sea necesario. Esta actividad tuvo un tiempo acotado de ejecución ya que luego fue necesario prestar asistencia durante la ejecución de los proyectos.

1.4.3 Etapa de validación

Ejecución de proyectos adoptando el proceso y prácticas definidas

Durante esta actividad el nuevo proceso y sus prácticas fueron ejecutados sobre los nuevos proyectos por cada uno de los equipos de trabajo, registrando los tiempos por etapas y prácticas adoptadas por estos, además de los controles de cambio e incidencias levantadas por el cliente para cada proyecto. También se registraron las observaciones sobre el nuevo proceso, información que podrá ser usada para trabajos futuros.

Validación de los resultados

Una vez terminado el plazo de ejecución los datos recolectados fueron comparados contra la información histórica regenerada, el resultado del análisis permitió evaluar el real impacto del trabajo abarcando características sobre las cuales se puedan obtener datos históricos comparativos.

La propuesta es medir el impacto del trabajo como sigue:

- A. Mejora de los productos desarrollados: comparando la cantidad de incidencias levantadas por el cliente luego de la entrega del producto, entendiendo que una disminución de estas implicaría una mejora en la calidad del producto.
- B. Calidad en la especificación de requisitos: midiendo la cantidad de controles de cambios solicitados, ya sea durante el desarrollo o luego de entregado el producto y entendiendo que una disminución indicará una mejora de la actividad.
- C. Desviación del esfuerzo planificado: comparando los plazos planificados de los proyectos históricos contra los nuevos proyectos trabajados bajo el nuevo proceso y entendiendo que una reducción en los atrasos de entrega indicará una mejora en el proceso.

Análisis de los resultados

En esta última actividad se describen los resultados obtenidos y un análisis de estos. El objetivo es dejar en evidencia el resultado obtenido y las razones por las cuales se obtuvo este.

2. Marco teórico

El software que desarrollamos no sólo es un conjunto de líneas de código que soluciona una problemática determinada, sino más bien es una colección de programas, configuraciones, datos, documentación, modelos, reglas, etc. que en su conjunto permiten apoyar y satisfacer las necesidades de las personas.

Para desarrollar y mantener software es que se han desarrollado metodologías compuestas por conjuntos de actividades que han demostrado ser eficaces en la práctica. La característica de estos modelos es que especifican lo que se debe hacer y no la forma en la cual se deben ejecutar las actividades propuestas [1]. Su beneficio principal radica en que entregan un marco de trabajo validado en base a multitud de experiencias.

Existen variados procesos de desarrollo de software ampliamente conocidos en la industria que proponen metodologías que facilitan la obtención de software de alta calidad. Inicialmente estos procesos se enfocan en la gestión realizada en grandes empresas [2], por lo que su adopción en pequeñas o medianas se dificulta al tratar de ser implementadas debido a su complejidad. Procesos de desarrollo como CMMI-DEV [3] o ISO/IEC 15504 [4] son ejemplos de procesos enfocados en las grandes empresas cada uno con sus particularidades.

Considerando distintas variables como la falta de tiempo, solvencia económica, informalidad de los procesos, entre otros, se han desarrollado procesos enfocados en las pequeñas empresas que ponen foco en la formalización de sus procesos y como los equipos de trabajo hacen uso de estos [5]. Ejemplos de procesos para pequeñas empresas son ISO/IEC 29110 y OpenUP.

La norma ISO/IEC 29110 [5] se caracteriza por estar enfocada a “entidades muy pequeñas” de no más de 30 personas además de no obligar la operación bajo un ciclo de vida particular, permitiendo a cada empresa adaptar las prácticas al ciclo de vida que estime conveniente. Por su parte, OpenUP [6] toma la filosofía ágil basado en la colaboración durante el proceso de software. Ambos definen actividades que pueden ser adaptadas según la cultura de la organización, lo que permite mayor flexibilidad y menor impacto en su adopción. También entregan artefactos asociados a las tareas que pueden ser adaptados o usados como guía para iniciar un proceso de estandarización.

2.1 Proceso de desarrollo de software

Cuando la complejidad de los proyectos de desarrollo de software aumenta se hace necesario seguir alguna especificación que permita al equipo de trabajo tener un marco común de forma que todas las actividades asociadas al desarrollo del sistema sean coherentes y formales.

Podemos visualizar un proceso de desarrollo de software como una serie de pasos que una persona u organización sigue para producir un sistema de software. Un proceso de software no es abstracto, es el proceso utilizado por un individuo u organización en un proyecto específico enfocado en la necesidad particular de un usuario [11].

La definición de un proceso de desarrollo de software es una descripción de este proceso en la que se identifican roles, artefactos y tareas y que permiten guiar a los ingenieros durante el desarrollo de

software [12]. Podemos detallar lo anterior como sigue:

- Roles: asociados a los participantes del proceso.
- Artefactos: es lo que debe construirse como modelos, documentos, guías, etc.
- Tareas: que deben realizarse y que incluyen técnicas y herramientas.
- Orden de tiempo: en el cual deben organizarse las actividades.

El objetivo de un proceso de desarrollo de software es el garantizar la construcción y la calidad del software resultante [13]. La ingeniería de procesos de software consiste en modelar, diseñar, mejorar y aplicar procesos utilizando lenguajes de alto nivel para el modelado de procesos. Si bien existe una amplia variedad de lenguajes para el modelado de procesos, aquellos basados en productos de trabajo son los más adecuados para modelar los procesos de desarrollo de software [14].

2.2 ISO/IEC 29110 - Perfiles de ciclo de vida para entidades muy pequeñas

La ISO (Organización internacional para la normalización) y la IEC (Comisión electrotécnica internacional) buscan llegar al mercado de las pequeñas y microempresas o VSEs (Very Small Entities) dado que los estándares tradicionales no son fáciles de adoptar por este tipo de entidades, es así como en el año 2011 publican la versión definitiva de la ISO/IEC 29110 en sus 5 series [7]. La ISO/IEC declara que la motivación de este estándar está dada por la importancia de las pequeñas empresas que se estima generan entre el 95% y el 99% de los negocios de TI (dependiendo de cada país). Además, declara que este grupo generalmente no es reconocido internacionalmente como generador de software de calidad dado que la mayoría de las normas ISO/IEC no se ajustan a las necesidades propias de este tipo de empresas.

El conjunto de documentos de la ISO/IEC 29110 han sido desarrollados con el objeto de mejorar la calidad de los procesos y los productos para ser usada con cualquier ciclo de vida, teniendo esto en mente es que se generaron los documentos según el tipo de audiencia interesada [8]:

ISO/IEC 29110 – Parte 1: Este documento define los conceptos asociados al negocio de las VSEs, introduce conceptos, presenta características y requisitos de éstas, además de aclarar definiciones de los perfiles. Este documento está dirigido a toda persona que tenga interés en esta norma.

La ISO/IEC 29110 – Parte 2: Presenta el Framework y su taxonomía, introduce conceptos de normalización de la ingeniería de software para las VSEs. Este documento está dirigido a empresas que producen estándares y comercializadoras de normas o herramientas relacionadas.

La ISO/IEC 29110 – Parte 3: Define las directrices para la evaluación del proceso y los requisitos necesarios para cumplir el propósito de los perfiles definidos. Este documento está dirigido a quienes evaluarán el proceso y su implementación.

La ISO/IEC 29110 – Parte 4-m: Genera los lineamientos y definiciones de todos los perfiles aplicables a las VSEs que no desarrollan productos de software crítico. Este documento está dirigido a empresas que producen estándares y comercializadoras de normas o herramientas relacionadas.

La ISO/IEC 29110 – Parte 5-m-n: Proporciona las guías de administración e ingeniería para perfiles por grupo, proporcionando la información de las actividades, procesos, roles y artefactos requeridos para la gestión de los proyectos. Este documento está dirigido a las VSEs y sus clientes.

La ISO/IEC 29110 – Parte 6-x: (Aún no publicada) Proporciona las guías de administración e ingeniería no asociadas a un perfil específico. Este documento está dirigido a las VSEs y a sus clientes.

En la Figura 1: Series ISO/IEC 29110, se presenta gráficamente la norma ISO/IEC 29110 y las series de posiciones de las partes que componen el marco de referencia.

En particular, para el desarrollo de este trabajo fue necesario profundizar en el documento ISO/IEC 29110-5-6-2 dado que es en éste donde se proponen las guías y roles que permiten la implementación del proceso para un perfil básico. Este documento define 2 procesos de trabajo, Gestión de proyecto (GP) y la Definición y ejecución del sistema (DE).

El proceso de gestión de proyecto (GP) tiene por objeto establecer y llevar a cabo de manera sistémica las tareas del desarrollo del sistema, lo que permite cumplir con los objetivos del proyecto respecto a sus tiempos, costos y calidad esperados. Este proceso está compuesto por 4 actividades:

- **Planeación:** Se define, acuerda y estima el alcance del proyecto.
- **Plan de ejecución:** Implementa el documento del plan del proyecto, también se planifica el plan para abordar las solicitudes de cambios.
- **Evaluación y control:** Se evalúa el plan ejecución del proyecto.
- **Cierre:** Provee la documentación del proyecto y productos de acuerdo con los requisitos definidos.

El proceso de definición del sistema y ejecución (DE) tiene por objeto la ejecución sistemática de las actividades de análisis, diseño, construcción, integración, verificación y validación para sistemas nuevos o mantenciones de acuerdo con los requisitos especificados. Este proceso está compuesto por 6 actividades:

- **Definición del sistema e inicio de ejecución:** Esta actividad asegura que el equipo de trabajo ejecuta el plan de proyecto establecido.
- **Ingeniería de requisitos:** Se definen, aprueban y comunican los requisitos a las partes involucradas.
- **Diseño:** Se describen los componentes e interfaces del software transformando los requisitos en software funcional.
- **Construcción:** Se construye de acuerdo con el diseño ejecutando pruebas unitarias por componente.
- **Integración de sistema, verificación y validación:** Se asegura que las piezas integradas satisfagan los requisitos del sistema.
- **Despacho del producto:** Entrega del producto al cliente o a alguno de los involucrados.

La relación entre estos procesos es graficada en la Figura 2: Procesos ISO/IEC 29110.

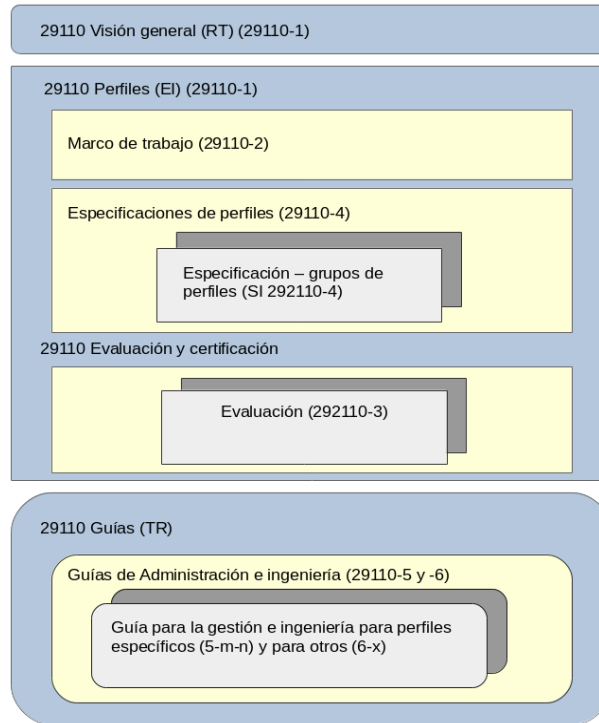


Figura 1: Series ISO/IEC 29110

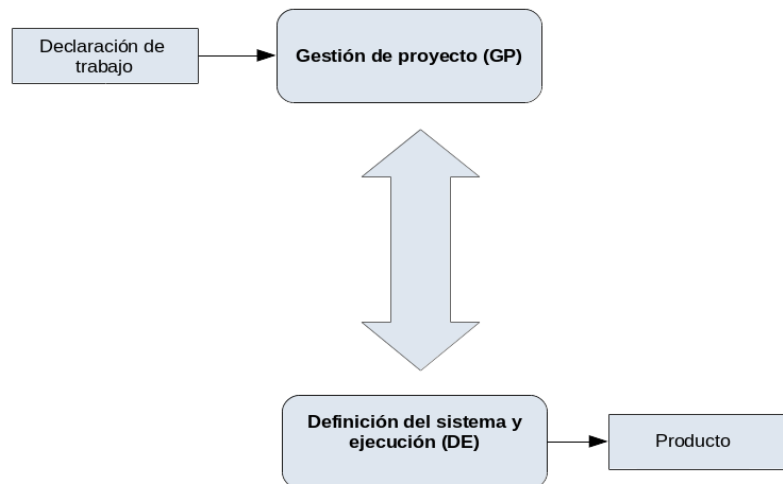


Figura 2: Procesos ISO/IEC 29110

2.3 OpenUP

OpenUP es una metodología de desarrollo de software basada en RUP (Rational Unified Process) que contiene el conjunto mínimo de prácticas que tienen por objeto ayudar a los equipos de desarrollo de software a realizar un producto de alta calidad de forma eficiente. Esta metodología fue propuesta por el

grupo de empresas conformado por: IBM Corp, Telelogic AB, Armstrong Process Group Inc., Number Six Software Inc. y Xansa; quienes la donaron a la Fundación Eclipse que la ha publicado bajo licencia libre. OpenUP es un proceso de desarrollo de software iterativo que es mínimo, completo y extensible [9].

OpenUP adopta un enfoque ágil para el desarrollo, enfocado en la colaboración del equipo y en los beneficios para las partes interesadas por sobre la formalidad y entregables improductivos. El proceso proporciona un enfoque progresivo para construir sistemas dentro de un ciclo de vida comprobado y estructurado.

Si bien OpenUP ya está definido y listo para usarse, es posible elegir extender el proceso o modificar los artefactos para cubrir necesidades particulares:

- Es posible ampliar el nivel de detalle de cada artefacto, con el fin de satisfacer las necesidades propias.
- Si se requiere, se puede modificar el proceso agregando prácticas ya conocidas por el equipo de trabajo.

2.3.1 Principios

OpenUP es impulsado por los siguientes principios básicos [9]

- **Equilibrar las prioridades para maximizar el valor de los interesados:** Para tener éxito las partes interesadas y los participantes del proyecto deben converger en una comprensión clara y un acuerdo de los siguientes factores:
 - Problema que resolver.
 - Restricciones impuestas al equipo de desarrollo (costo, cronograma, recursos, etc.).
 - Restricciones impuestas en la solución.

El desafío para todos los participantes del proyecto es crear una solución que maximice el valor entregado a los interesados, sujeto a las restricciones. Mantener el equilibrio implica realizar concesiones críticas de costo-beneficio entre las características deseadas y las decisiones de diseño posteriores que definen la arquitectura del sistema.

- **Colaborar para alinear intereses y compartir el conocimiento:** Desarrollar prácticas que fomenten un grato ambiente de equipo. Un equipo saludable permite la colaboración efectiva que alinea los intereses de los participantes del proyecto y ayuda a desarrollar una comprensión compartida.
- **Centrarse tempranamente en la arquitectura para minimizar riesgos y organizar el desarrollo:** Sin una base arquitectónica clara un sistema evolucionará de forma ineficiente. Un sistema de este tipo a menudo resulta difícil de ampliar, reutilizar o integrar sin un trabajo no contemplado inicialmente. También es difícil organizar el equipo o comunicar ideas sin el enfoque técnico común que brinda la arquitectura.
- **Evolucionar continuamente para obtener retroalimentación y mejorar:** La intención detrás de este principio es obtener retroalimentación continua y mejorar tanto el producto como el proceso del equipo del proyecto. Cuando se proporciona una estructura y se crea una mentalidad

para obtener retroalimentación y mejora continuas los cambios se acomodan más fácilmente. Además, la retroalimentación se captura más temprano y con mayor frecuencia y los riesgos de mayor impacto se enfrentan antes. Al identificar y atacar constantemente los riesgos, hay más confianza en el progreso y la calidad del proyecto.

2.3.2 Componentes

Para adoptar OpenUP es que esta metodología exige que los involucrados en el proceso tomen familiaridad con los siguientes elementos:

- **Artefacto:** Un artefacto es todo lo que se produce. Pueden ser documentos, modelos, bases de datos, fuentes ejecutables, etc.
- **Tarea:** Una tarea se define como una serie de pasos que involucra crear o modificar uno o más artefactos. Una tarea puede ser la creación de un documento o definir el plan de iteración entre otros.
- **Rol:** Un rol define el comportamiento y responsabilidades de un participante o de un conjunto de estos trabajando como equipo. Es necesario destacar que un rol no define a un participante sino más bien las responsabilidades que este adquiere. Es común que un participante tome más de un rol de forma paralela, por lo que el participante cambiará su rol durante la ejecución del proyecto. Los roles pueden definirse como analista, desarrollador, etc.
- **Proceso:** Los procesos toman los artefactos, tareas y roles y los relacionan entre sí dentro de secuencias temporales que satisfacen las necesidades del proyecto. Además, las tareas y/o artefactos pueden ser agrupados en actividades.

Los elementos anteriores son organizados usando los siguientes conceptos:

Prácticas: Una práctica es un enfoque documentado para resolver uno o varios problemas comunes, estas están pensadas como "partes" del proceso de adopción, habilitación y configuración.

Configuraciones: La mayoría de las configuraciones consisten en una selección de prácticas y más contenido para vincular éstas.

OpenUP declara que el ciclo de vida de un proyecto permite que el equipo de desarrollo aporte con los artefactos necesarios para generar micro incrementos, lo que permite visualizar periódicamente el progreso ya que la aplicación va evolucionando en función de estos.

El objetivo de OpenUP es ayudar al equipo de desarrollo durante todo el ciclo de vida para que éste sea capaz de añadir valor de negocio a los clientes, con la entrega de un software operativo y funcional con cada iteración. El ciclo de vida del proyecto provee a los clientes una visión del proyecto para que sea más fácil controlar el riesgo, los tiempos y cualquier otro problema durante el desarrollo del proyecto.

2.3.3 Fases

OpenUP es un proceso iterativo con iteraciones distribuidas a través de cuatro fases:

Fase de inicio: El propósito de esta fase es lograr que las partes interesadas acuerden y tomen conocimiento de los objetivos del proyecto. Esta fase tiene cuatro objetivos:

- **Comprender lo que se construirá:** Determinar la visión general, incluyendo el alcance del sistema. Además, se requiere identificar las partes interesadas y cuáles son sus criterios de aceptación.
- **Identificar las funcionalidades clave:** Definir los requisitos críticos.
- **Determinar a lo menos una posible solución:** Evaluar si la visión descrita es factible técnicamente. Esto puede involucrar el desarrollo de una arquitectura a alto nivel o generar pruebas de conceptos.
- **Viabilidad:** Evaluar un costo estimado del proyecto, su cronograma y los riesgos asociados.

Fase de elaboración: El propósito de esta fase es generar la arquitectura base para soportar la siguiente fase. Durante esta fase se ejecutan las tareas de análisis del dominio y definición de la arquitectura del sistema. Se debe elaborar un plan de proyecto, estableciendo los requisitos y la arquitectura. Además, se deben definir el proceso de desarrollo, el entorno de desarrollo y las herramientas y la infraestructura a utilizar. Al final de la fase se debe tener una definición clara y precisa de los casos de uso, los actores y la arquitectura del sistema. Los objetivos de esta fase ayudan a abordar los riesgos asociados con los requisitos, la arquitectura, los costos y el cronograma:

- **Permitir el entendimiento detallado de los requisitos:** Tener buena comprensión de la mayoría de los requisitos permite generar un plan detallado y así obtener el apoyo de las partes interesadas.
- **Diseño, implementación, validación y establecimiento de la arquitectura base:** Diseñar, implementar y probar la estructura del sistema.
- **Mitigación de los principales riesgos:** Es común que los principales riesgos aparezcan como resultado del trabajo detallado de los requisitos.

Fase de construcción: El propósito de esta fase es construir todos los componentes y funcionalidades del sistema que aún no han sido implementados, probados e integrados. Los objetivos de esta fase ayudan a generar un desarrollo eficiente:

- **Desarrollo iterativo:** Generar un producto que pueda ser publicado y validado por el usuario final.
- **Minimizar los costos de desarrollo y lograr cierto grado de paralelismo:** Optimizar los recursos y propiciar el trabajo paralelo entre el equipo de desarrollo.

Fase de transición: El propósito de esta fase corresponde a la introducción del producto a los usuarios siempre que esté lo suficientemente maduro. La fase de la transición consta de las sub-fases de pruebas de versiones beta, marcha blanca y capacitación de los usuarios finales. En función de la respuesta obtenida por los usuarios puede ser necesario realizar cambios en las entregas finales. Los objetivos de esta fase son:

- **Pruebas integrales para validar que los requisitos sean cumplidos:** Es común que durante esta etapa sea necesario aplicar arreglos o hacer mejoras de rendimiento y usabilidad.
- **Lograr que la aceptación de las partes interesadas:** Puede involucrar set de pruebas para la lograr la aceptación del producto.

Mejorar la ejecución del proyecto a través de las lecciones aprendidas: Documentar todas las actividades que generaron impacto (positivo o negativo) en el desarrollo del proyecto para así mejorar el proceso en el futuro.

2.4 Spem

SPEM (Software Process Engineering Metamodel) es una especificación de OMG (Object Management Group) y es un lenguaje de alto nivel para el modelado de procesos que está basado en MOF (MetaObject Facility) y es un metamodelo UML (Uniform Model Language) que utiliza esta notación permitiendo visualizar, especificar, construir y documentar sistemas orientados a objetos [12]. También provee un conjunto de elementos de modelado de procesos para describir cualquier proceso de desarrollo de software sin agregar modelos o restricciones de áreas o disciplinas específicas. SPEM proporciona una sintaxis y estructura para cada aspecto de los procesos desarrollo, incluyendo:

- Roles.
- Tareas.
- Artefactos.
- Lista de verificación
- Productos de trabajo.
- Técnicas y herramientas.
- Estructuras de trabajo.
- Capacidad de rastreo y refinamiento.
- Ayuda sensible al contexto, guía y lineamientos.
- Descripción textual de elementos.

2.4.1 Marco de uso conceptual

En la industria del software existen muchas visiones y conocimientos disponibles sobre cómo desarrollar software de manera efectiva. Los equipos de desarrollo no solo necesitan adquirir información detallada sobre tecnologías de desarrollo específicas, así como diversos entornos de desarrollo y herramientas, sino que también deben descubrir cómo organizar su trabajo a través de las mejores prácticas de desarrollo moderno, como el desarrollo de software ágil, iterativo, centrado en la arquitectura, basado en el riesgo y la calidad.

Permitir que los participantes del equipo de desarrollo definan las prácticas con las cuales abordar el proceso de forma independiente presenta algunos problemas [12]:

- Los miembros del equipo no tendrán acceso fácil y centralizado a la misma fuente de información cuando sea necesario, es decir, los desarrolladores pueden recolectar información desde distintas versiones o fuentes de datos.
- Es difícil integrar artefactos y procesos de desarrollos que han sido desarrollados con distintos formatos, esto ya que cada integrante puede haber utilizado un estilo de representación distinto.
- Es difícil definir y organizar un enfoque de desarrollo sistemático implementado por cada uno de los desarrolladores.

SPEM es un metamodelo de ingeniería de procesos, así como también un marco de trabajo conceptual. Figura 3: Marco conceptual de SPEM, que puede proporcionar los conceptos necesarios para modelar, documentar, presentar, gestionar, intercambiar y aplicar métodos y procesos de desarrollo. Una implementación de este metamodelo podría ser dirigida a ingenieros de procesos, líderes de proyectos, gerentes de proyectos que son responsables de mantener e implementar procesos para sus organizaciones de desarrollo o proyectos individuales [12].

Un proceso especificado en SPEM permite [12]:

- Proveer librerías de elementos estandarizados y reutilizables.
- Apoyar la elaboración, administración y crecimiento sistemático de procesos.
- Apoyar el uso de contenidos y procesos específicos a cada proyecto.
- Apoyar la ejecución de los procesos en los proyectos de desarrollo.

2.4.2 Arquitectura del metamodelo SPEM 2.0

SPEM 2.0 se utiliza para definir los procesos de desarrollo de software y sus componentes. El alcance de SPEM se limita a los elementos mínimos necesarios para definir cualquier proceso de desarrollo de software sin agregar características específicas para dominios o disciplinas de desarrollo particulares. El objetivo es adaptarse a una amplia gama de métodos y procesos de desarrollo de diferentes estilos, variables culturales, niveles de formalismo y modelos de ciclo de vida, aunque el foco de SPEM apunta a los proyectos de desarrollo.

SPEM 2.0 no pretende ser un lenguaje genérico de modelado de procesos, tampoco proporciona sus propios conceptos de modelado. SPEM 2.0 define la capacidad del implementador para elegir el enfoque de modelado que mejor se adapte a sus necesidades. También proporciona estructuras específicas para mejorar dichos modelos genéricos que son característicos para describir procesos de desarrollo. En otras palabras, SPEM 2.0 se enfoca en proporcionar las estructuras de información adicional que necesita para los procesos modelados con actividades UML 2.0 o BPMN / BPDM para describir un proceso de desarrollo real [12].

SPEM 2.0 está estructurado en siete paquetes principales que dividen el modelo en unidades lógicas. Cada unidad extiende de las unidades de las que depende proporcionando estructuras y capacidades adicionales. Como regla general, el mecanismo de combinación de paquetes aplicado realiza una extensión gradual de las propiedades e interrelaciones modeladas unidad por unidad permitiendo a las unidades definidas en las capas inferiores usar las características de las unidades de nivel superior, Figura 4: Estructura del metamodelo SPEM. En muchos casos, las clases de metamodelos se introducen en un paquete de nivel inferior de la manera más simple posible y luego se extienden en unidades de nivel superior a través del mecanismo de combinación de paquetes con propiedades y relaciones adicionales para cumplir con los requisitos de modelado de procesos más complejos. Esto permite utilizar diferentes niveles de capacidades, conjuntos de conceptos, y niveles de formalismo para expresar los procesos utilizando unos u otros paquetes.

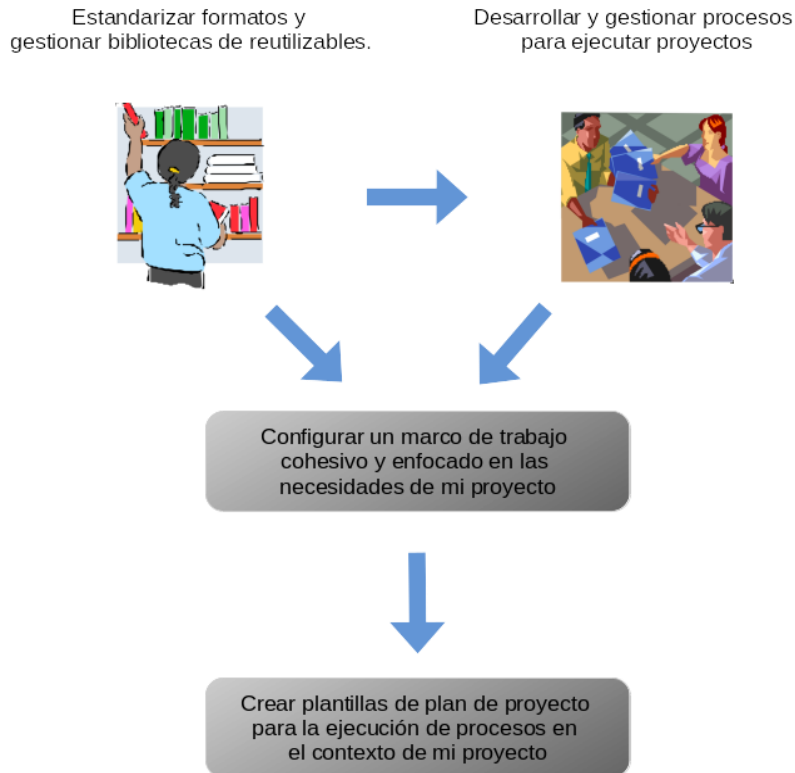


Figura 3: Marco conceptual de SPEM

Los contenidos de cada paquete se pueden resumir como sigue [15]:

- Core: Contiene las clases y abstracciones que sirven de base para las clases de los demás paquetes.
- Process Structure: Define la base para la creación de modelos de procesos. Define la estructura de desglose de trabajo estática mediante anidamiento de actividades y dependencias entre ellas.
- Process Behavior: Permite extender las estructuras del paquete anterior con modelos de comportamiento.
- Managed Content: Permite incorporar y gestionar descripciones en lenguaje natural, documentos y otras informaciones útiles para la comprensión por humanos.
- Method Content: Incluye los métodos y técnicas fundamentales que sirven de base para el ensamblado de procesos, metodologías, ciclos de vida, etc.
- Process with Methods: Permite integrar los procesos definidos en el paquete “Process Structure” con los métodos y técnicas del paquete “Method Content”. Al asociar elementos de método a partes específicas de procesos, se crean nuevas clases (tarea en uso, rol en uso, etc.) que heredan de los elementos de método, pero pueden tener cambios individualizados.

Method Plugin: Incorpora conceptos para diseñar y gestionar bibliotecas o repositorios de contenidos de método y de procesos, que sean mantenibles a gran escala, reutilizables y configurables.

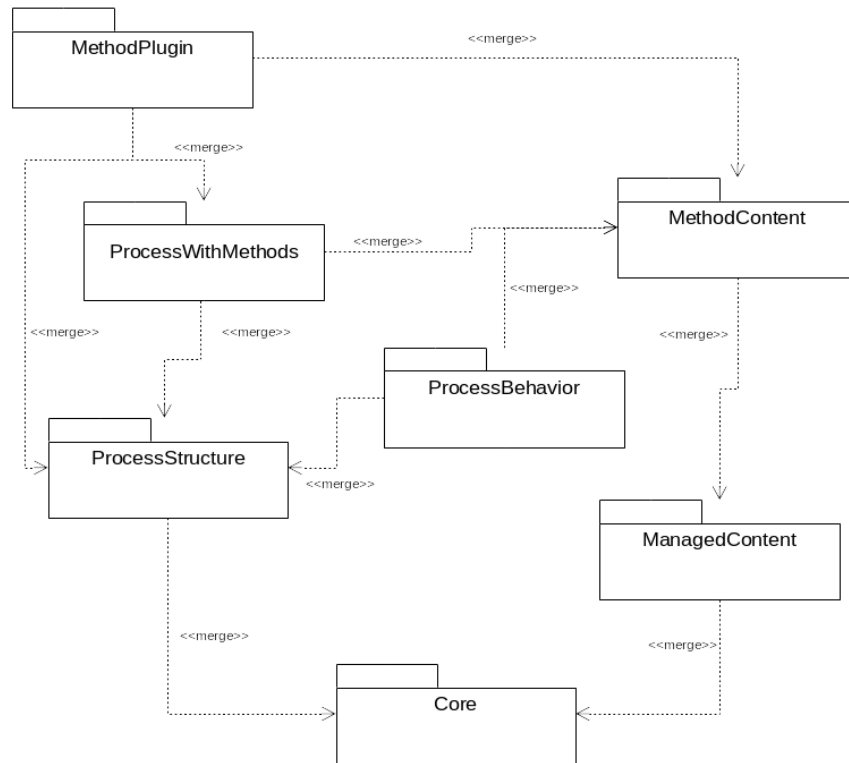


Figura 4: Estructura del metamodelo SPEM

2.5 EPF Composer

Eclipse Process Framework (EPF) es un proyecto de código abierto que es administrado por la Fundación Eclipse, sus principales objetivos son **[10]**:

- Proporcionar un marco de trabajo y de herramientas extensibles para la ingeniería de procesos de software: creación de métodos y procesos, administración de bibliotecas, configuración y publicación de un proceso.
- Proporcionar contenido de procesos validado y extensible para una variedad de procesos de desarrollo y administración de software que soportan el desarrollo iterativo, ágil e incremental y que se aplican a un amplio conjunto de plataformas y aplicaciones de desarrollo.

Este marco de trabajo está compuesto por los siguientes componentes **[10]**:

- Metamodelo: el contenido y los procesos se estructuran según el metamodelo SPEM 2.0. El metamodelo será extensible mediante el uso de atributos personalizados y elementos de proceso personalizados.
- Marco de herramientas de proceso central extensible: se proporcionan las API de funcionalidad básica y punto de extensión para permitir la creación de métodos, creación de procesos, administración de bibliotecas y configuración y publicación.

2.6 AVISPA (Analysis and Visualization for Software Process Assessment)

AVISPA es una herramienta que permite construir planos que resaltan patrones de error para un modelo de proceso dado. Los patrones de error se identifican mediante elementos que son anormalmente diferentes, gráficamente, en comparación al resto de los elementos. Entre los beneficios que entrega esta herramienta es permitir a los usuarios enfocarse sólo en los elementos que presentan problemas. AVISPA provee tres vistas distintas que se enfocan en un aspecto particular del modelo de proceso [16]:

- **Plano de roles:** Cada nodo representa un rol y su tamaño representa el número de tareas en las cuales éste está involucrado. Los arcos entre los nodos representan colaboración entre los distintos roles.
- **Plano de tareas:** Los nodos representan tareas en donde el alto representa la cantidad de entradas y el ancho la cantidad de productos de trabajo de salida. Los arcos entre nodos representan precedencia entre las tareas. El color de los nodos representa la cantidad de roles involucrados en la tarea.
- **Plano de productos de trabajo:** cada nodo representa los distintos productos de trabajo, la altura del nodo representa la cantidad de tareas que escriben mientras que el ancho la cantidad de tareas que leen el producto de trabajo. Los arcos entre estos nodos representan que tareas usan y producen los grupos de trabajo relacionados.

AVISPA define los patrones de error como el conjunto de anomalías detectados durante la especificación de procesos de software que frecuentemente se identifican durante este proceso, estos patrones son [16]:

- **Rol aislado:** es un rol definido que no colabora dentro del proceso. Se puede identificar dentro del plano de roles como un nodo no conectado.
- **Subproyectos independientes:** Ocurre cuando la definición presenta subgrafos independientes. Se identifica dentro del plano de tareas y del plano de productos de trabajo como subgrafos no conectados.
- **Rol sobrecargado:** es un rol definido con muchas tareas asociadas. Se identifica dentro del plano de roles como nodos de mayor tamaño que el resto.
- **Tareas multipropósito:** Son tareas que generan como salida un alto número de productos de trabajo. Se visualizan en el plano de tareas como nodos de mayor tamaño que el resto.
- **Productos de trabajo altamente usados:** Son productos de trabajo usados por un alto número de tareas. Se visualiza en el plano de productos de trabajo como nodos de mayor tamaño que el resto.
- **Roles, tareas y productos de trabajo sin guías:** Son los elementos que no poseen guías asociadas.
- **Productos de trabajo residuales:** Son productos de trabajo entregables de cara al cliente o innecesarios. Se visualiza en el plano de productos de trabajo elementos que no sirven de entrada a ninguna tarea.

3. Definición del proceso de desarrollo

La complejidad de las nuevas necesidades de los clientes de Nexo-IT exige a la empresa tener la capacidad de manejar el crecimiento de los equipos de trabajo y la correcta gestión de estos. El no tener una definición clara de los procesos ha provocado problemas tanto en calidad como en los tiempos de respuesta que los nuevos desarrollos exigen. Dado lo anterior, el objetivo de este trabajo es el formalizar un marco de trabajo propio de la compañía con el objeto de mantener control sobre las actividades que se ejecutan dentro de los distintos proyectos, buscando con esto cumplir con los compromisos adquiridos con los clientes.

La metodología usada durante esta etapa consta de dos etapas generales:

1. Etapa de levantamiento
 - a. Levantamiento de datos históricos de planificación de proyectos necesario para la evaluación del marco de trabajo propuesto
 - b. Identificación y catalogación de las actividades ejecutados actualmente por los jefes de proyectos.
 - c. Análisis de prácticas y procesos con el objeto de consolidar éstas.
2. Etapa de construcción
 - a. Construcción del marco de trabajo general.
 - b. Análisis y publicación del resultado.

3.1 Etapa de levantamiento

3.1.1 Levantamiento de datos históricos

Durante esta actividad se trabajó recopilando la información histórica almacenada para un cliente particular, los datos fueron tabulados de forma que luego puedan ser usados para evaluar el proceso propuesto. El listado de proyectos recolectado se presenta en la Tabla 2: Regeneración de datos históricos, donde las columnas de duración planificada y real están medidas en meses, mientras que las incidencias y controles de cambio se presentan en cantidad de casos. Para luego facilitar la evaluación de los resultados se generan cuatro gráficos que representan la desviación estándar de los proyectos, medianos y pequeños para las incidencias y controles de cambio.

También se presentan cuatro gráficos que muestran como es que el número de incidencias y el de controles de cambio se distribuyen según el tipo de proyecto, con eso se busca visualizar el nivel de control de las actividades que actualmente se ejecutan.

La Figura 5: Desviación estándar de incidencias para proyectos medianos muestra el rango dentro del cual varía las incidencias ocurridas para los proyectos históricos de tamaño mediano.

La Figura 6: Desviación estándar de incidencias para proyectos pequeños también representa la distribución de las incidencias ocurridas, pero para los proyectos de tamaño pequeño.

Proyecto	Tipo	Duración planificada (meses)	Duración real (meses)	Nro. de incidencias	Controles de cambio
Facturación electrónica	Mediano	3.5	4.5	6	10
Satisfacción compras	Mediano	1.5	2	1	4
PecCam	Mediano	2.5	4	4	5
Gestor documental	Pequeño	1	1.2	3	10
Integración BP	Mediano	3.5	4.5	1	3
AD onpremise/sync	Pequeño	1	1.2	1	0
Beneficios	Mediano	3	3.5	1	4
Tramos	Pequeño	1	1.2	1	0
Portal público	Pequeño	0.75	1	3	5
Directorio	Pequeño	0.75	0.8	1	0
Migración convivencia	Mediano	3	3.5	3	4
Socios	Mediano	2.5	4	3	11
Comisiones	Mediano	2.5	5	3	13
Convivencia/beneficios	Mediano	2	3	1	2
Servicios PO	Mediano	3	3.2	6	12
BT-Servipag	Pequeño	1.5	1.8	1	14
BT-Reembolsos	Pequeño	1.5	1.6	1	8

Tabla 2: Regeneración de datos históricos

La distribución de los controles de cambio para los proyectos históricos de tamaño mediano se muestra en la Figura 7: Desviación estándar de controles de cambio para proyectos medianos.

Finalmente, la Figura 8: Desviación estándar de controles de cambio para proyectos pequeños, muestra la distribución de los controles de cambio ocurridos en los proyectos históricos de tamaño pequeño.

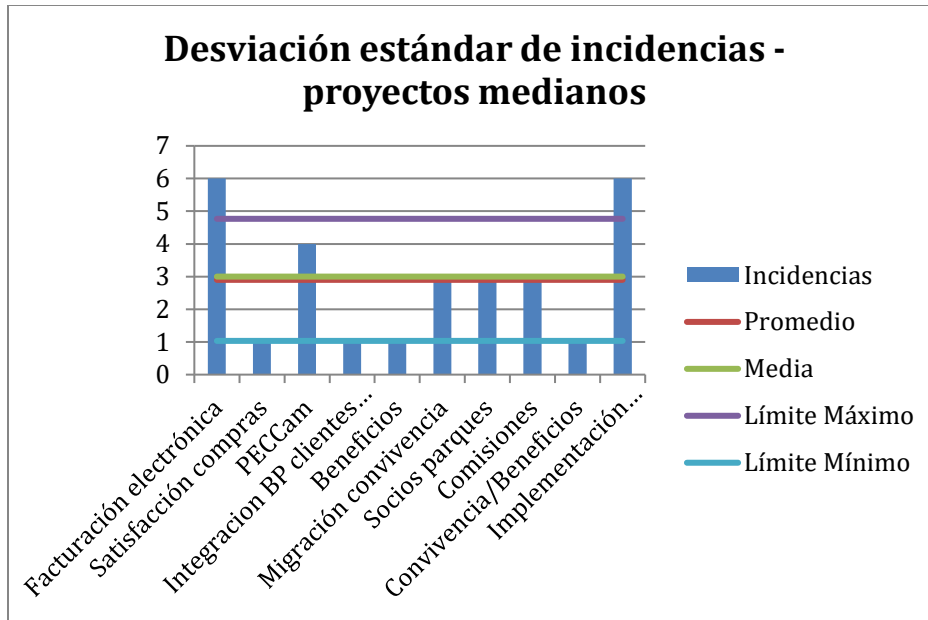


Figura 5: Desviación estándar de incidencias para proyectos medianos

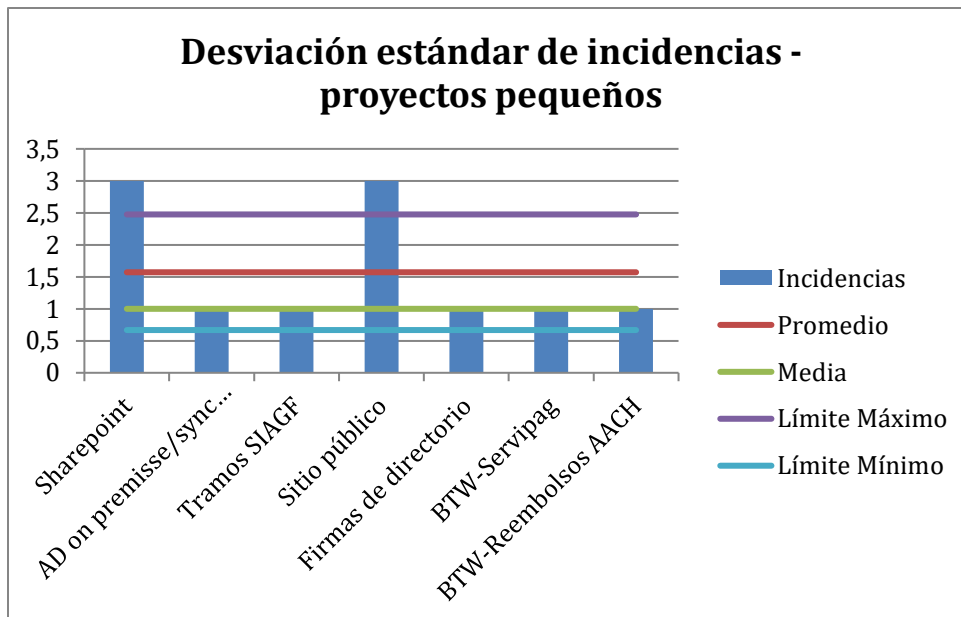


Figura 6: Desviación estándar de incidencias para proyectos pequeños

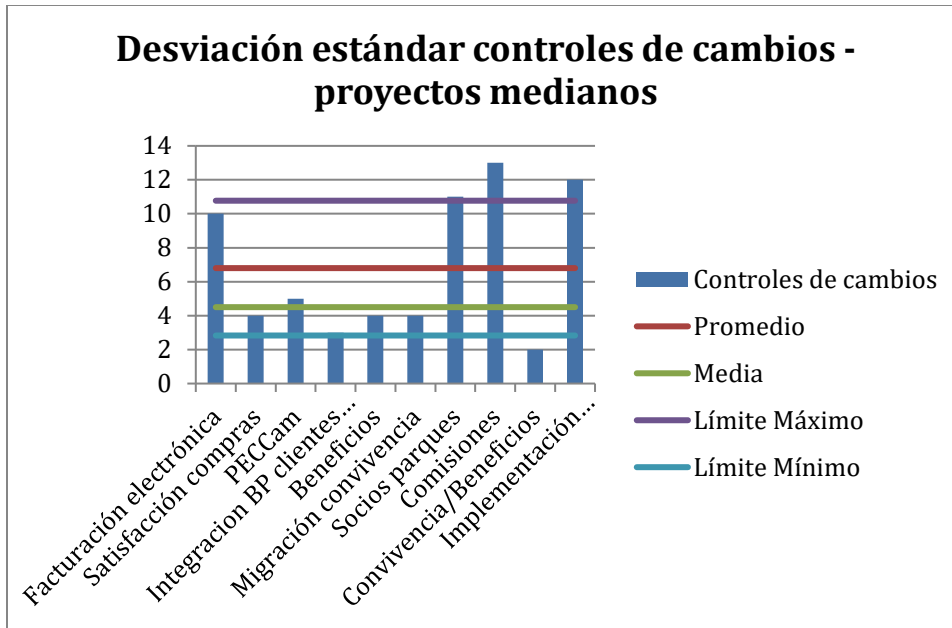


Figura 7: Desviación estándar de controles de cambio para proyectos medianos

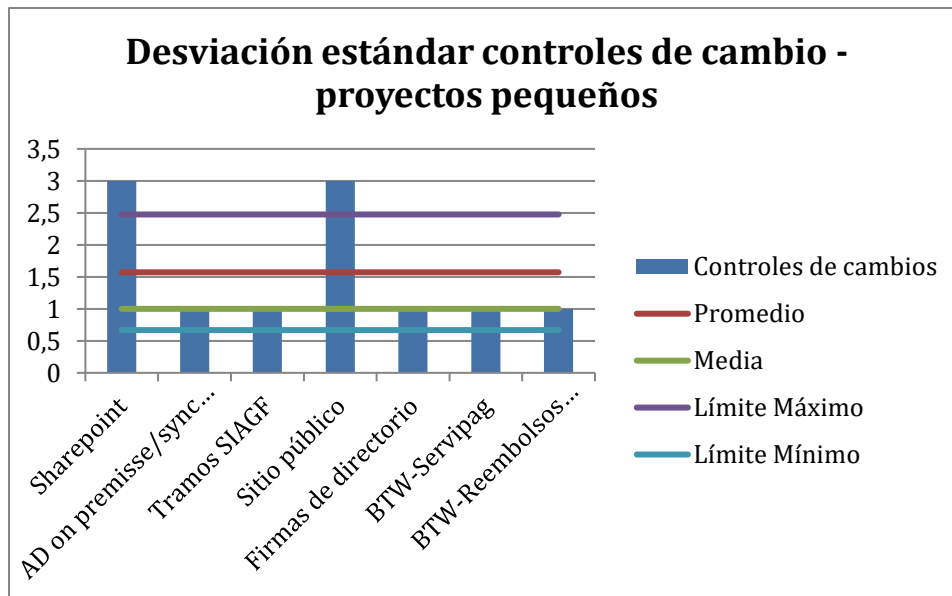


Figura 8: Desviación estándar de controles de cambio para proyectos pequeños

3.1.2 Identificación y catalogación de actividades

Durante esta etapa se identificaron y catalogaron las actividades que los dos equipos de trabajo realizan en la actualidad durante la ejecución de sus proyectos. También se identificaron los ciclos de vida y la forma en la cual estas actividades están organizadas. El levantamiento de los datos se realizó a través de entrevistas con cada uno de los jefes de proyectos, entrevistas que dan como resultado un listado de

actividades por cada uno obteniendo los siguientes modelos.

Equipo 1 (DA)

DA participa dentro de Nexo-IT como jefe de proyectos desde hace cinco años, por las características de su rol DA tiene dentro de sus responsabilidades participar en la generación de la propuesta que será enviada al cliente. Este declara que todos los proyectos que lidera son abordados bajo el modelo cascada ya que es el que mejor se adapta a sus proyectos y a los procesos de sus clientes.

DA trabaja de forma permanente en las instalaciones del cliente. Además, tiene contacto directo con las jefaturas internas teniendo visibilidad de los proyectos internos dando consultoría siempre que le es requerido, por lo anterior es que DA en la actualidad no genera documentación de viabilidad de los nuevos proyectos, DA es quien filtra los proyectos que son licitados, de cara a Nexo-IT, lo que puede provocar la pérdida de oportunidades al no llegar éstas a la etapa de evaluación.

El mayor problema que DA identifica en sus proyectos es el manejo de los controles de cambios, por temas asociados a la relación con el cliente y respecto a los tiempos asociados a la generación de la propuesta, por carga de trabajo.

La Figura 9: Proceso estándar ejecutado por DA, muestra el proceso bajo el que comúnmente ejecuta sus proyectos, para obtener este fueron necesarias tres sesiones de trabajo.

Equipo 2 (JZ)

JZ participa dentro de Nexo-IT como jefe de proyectos desde hace dos años. JZ sólo participa en la ejecución del proyecto, no tiene tareas asociadas en la creación de la propuesta. Dentro del cliente en donde se desempeña declara que el 90% de los proyectos que lidera son abordados bajo un modelo iterativo ya que es el que mejor se adapta a sus proyectos, además de ser el modelo impulsado por el cliente.

Los pocos proyectos que no son abordados bajo el modelo iterativo son trabajados bajo un modelo cascada y generalmente son proyectos pequeños que no superan las 3 semanas de trabajo.

Estas actividades fueron desarrolladas en tres sesiones de trabajo y dieron como resultado el proceso formalizado en EPF Composer mostrado en la Figura 10: Proceso estándar ejecutado por JZ.

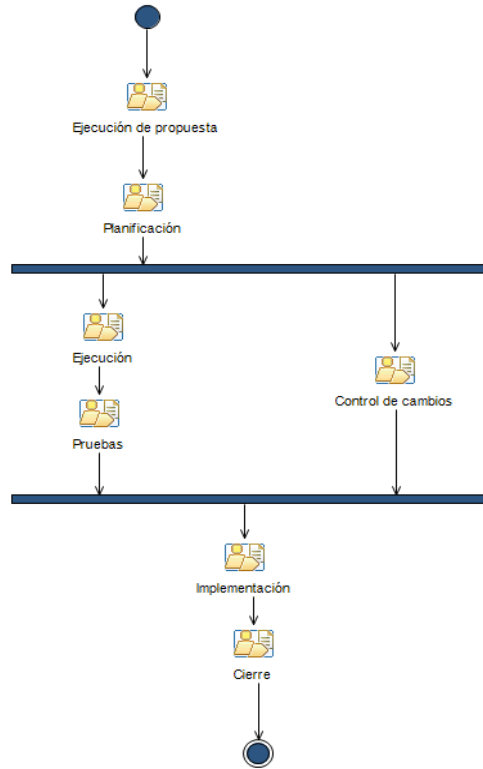


Figura 9: Proceso estándar ejecutado por DA

3.1.3 Análisis de las prácticas y procesos

Durante esta actividad se analizaron las prácticas recolectadas con el objeto de encontrar similitudes que permitan consolidar éstas y así obtener un catálogo consolidado. De forma similar, los procesos de cada equipo fueron comparados con el objeto de obtener un marco de trabajo general que sirvió para definir un proceso formal de desarrollo que los equipos adoptarán en los futuros desarrollos. El resultado de esta etapa se refleja en la construcción del proceso de desarrollo descrito en la siguiente sección.

3.2 Etapa de construcción

3.2.1 Construcción del proceso de desarrollo y prácticas asociadas

Tomando como entrada el catálogo consolidado de prácticas levantadas en la etapa de análisis y como modelos el estándar propuesto en ISO/IEC 29110 y el proceso definido en OpenUP se generó un conjunto de roles, artefactos, tareas y orden en el cual los desarrollos serán ejecutados. El objetivo de esta actividad es generar un proceso adaptable que permite ser ajustado según los proyectos lo requieran, definiendo así un marco de trabajo general y común que sea lo suficientemente genérico como para abordar proyecto con metodologías ágiles o tradicionales. La especificación del proceso se realizó

mediante el uso del lenguaje SPEM 2.0.

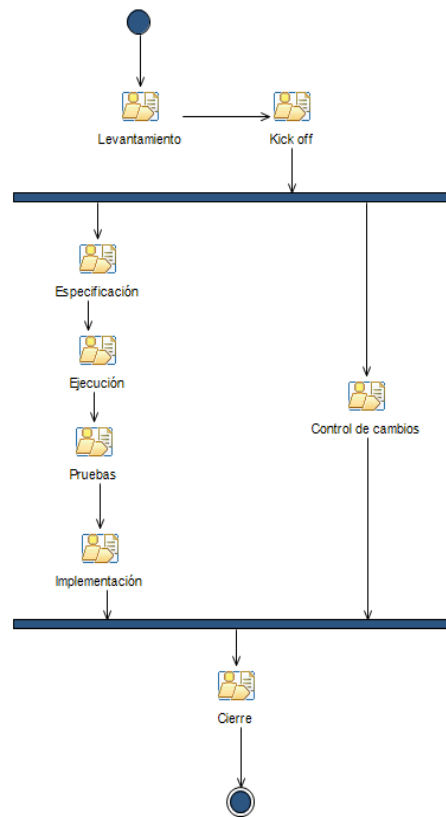


Figura 10: Proceso estándar ejecutado por JZ

El proceso propuesto consta de cuatro fases agrupadas en dos procesos según se describe a continuación:

1. Proceso 1: Gestión de proyectos (GP)
 - a. Fase 1. Desarrollo de propuesta
 - b. Fase 2. Planificación general
2. Proceso 2: Ejecución del proyecto (EP)
 - a. Fase 3: Planificación detallada
 - b. Fase 4: Ejecución

La división del proceso tiene por objeto principal separar las actividades comerciales y las de ejecución del proyecto, diferenciando de mejor manera las responsabilidades por roles. Además, se espera que esta separación permita:

- Que la GP pueda gestionar los recursos necesarios para la ejecución del proyecto, permitiendo así al equipo de ejecución enfocarse en las actividades relacionadas.
- Que la EP pueda tomar control sobre el desarrollo del proyecto, seleccionando la mejor forma de abordar éste según sus características, necesidades, expectativas del cliente y experiencia del equipo de trabajo. También se espera que las tareas comerciales que puedan ser necesarias

durante los controles de cambio sean abordadas por el área comercial separando así las responsabilidades.

Los procesos y fases definidos deben ser ejecutados sin excepción, siempre que el flujo lo requiera, ya que son los elementos base del marco de trabajo definido, estos se presentan en la Figura 11: Proceso propuesto.



Figura 11: Proceso propuesto

Además, con el detalle de las actividades recolectadas y analizadas se generó un marco de trabajo que consta de 13 actividades agrupadas en las 4 fases definidas presentadas en la Figura 12: Actividades definidas para el proceso.

El detalle del proceso definido puede ser consultado en el Anexo A.

3.2.2 Análisis y publicación del resultado además de entregar apoyo técnico

Si bien la construcción del proceso de desarrollo permite declarar el marco general dentro del cual los equipos ejecutarán sus labores, existe la posibilidad que éste contenga errores o deficiencias en su construcción. Es por lo anterior que la definición del proceso continúa con una evaluación de este, actividad que permite identificar errores o deficiencias para así contar con información que permita ejecutar correcciones o mejoras en caso sea necesario.

Para esta actividad se buscó apoyo de la herramienta AVISPA [16], herramienta que permite analizar gráficamente la calidad de un proceso declarado con SPEM 2.0, permitiendo identificar errores comunes entregando para esto 3 distintos planos: plano de roles, plano de tareas y plano de productos de trabajo. El proceso construido fue analizado y el resultado se describe a continuación:

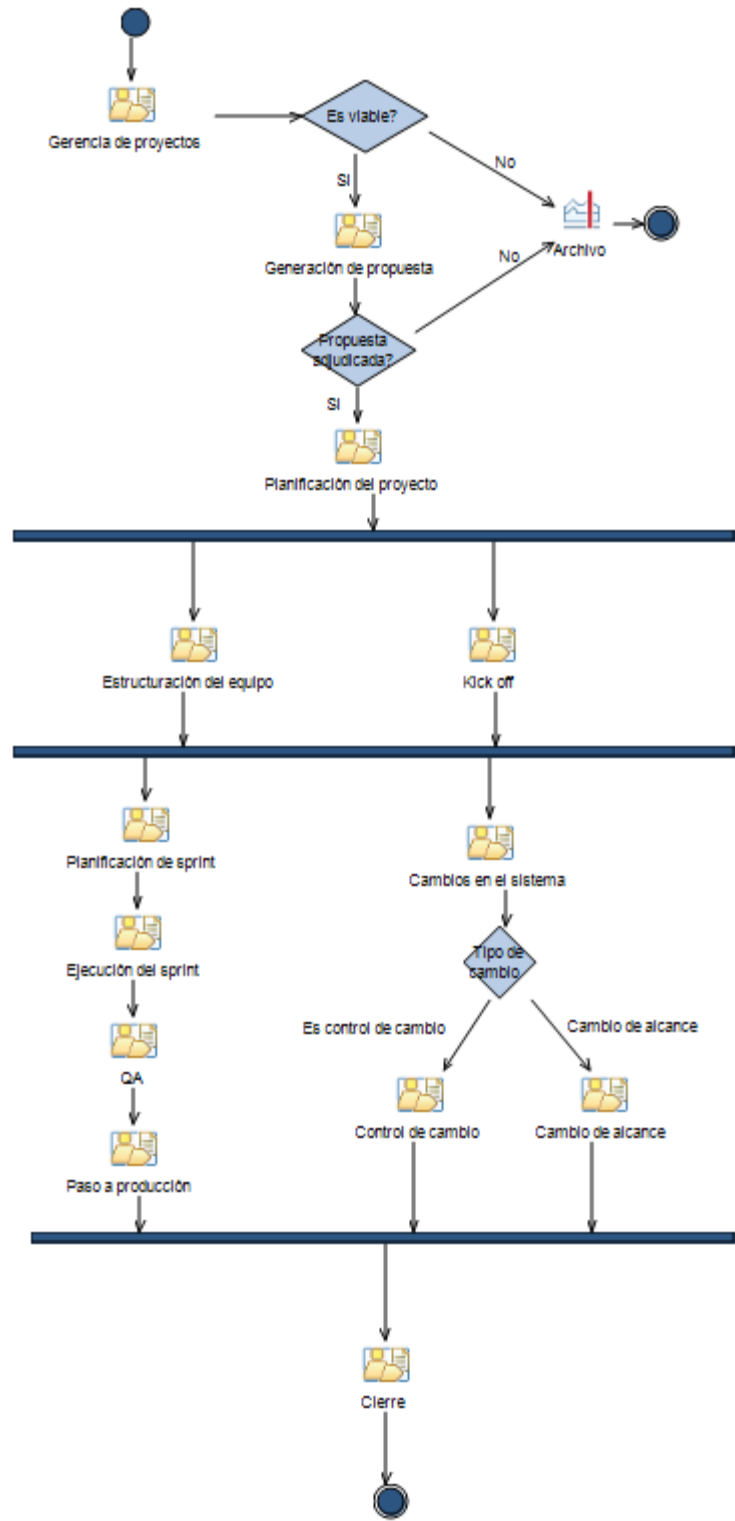


Figura 12: Actividades definidas para el proceso

Plano de roles

Este plano permite visualizar los roles participantes del proceso, como se relacionan entre sí además de la carga de trabajo por cada uno de estos, los dos patrones analizados son:

Roles aislados

Identifica casos en los cuales los roles no colaboran en la ejecución de tareas. Si bien esto por sí puede no ser un error, generalmente muestra un problema de especificación. La Figura 13: Roles aislados, muestra que en este caso no se presenta esta situación, cada nodo representa un rol y los arcos la colaboración entre estos.

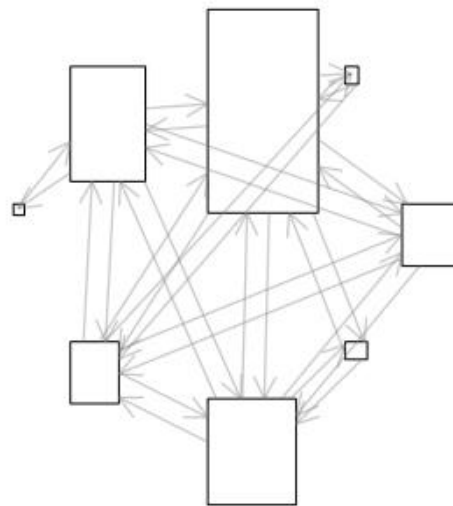


Figura 13: Roles aislados

Roles sobrecargados

Los roles con alta carga de trabajo corren el riesgo de comprometer la ejecución de las tareas en la cuales participa. El análisis del proceso construido evidencia problemas de sobrecarga en prácticamente todos los roles, Figura 14: Roles sobrecargados, por lo anterior, la redistribución de tareas no parece ser la solución al problema. Una posible solución es la especialización de roles repartiendo responsabilidades, lo que necesariamente requiere el aumento de personal en el equipo de trabajo.

Cada nodo representa un rol, el ancho y alto representa la cantidad de tareas y de artefactos relacionados respectivamente.

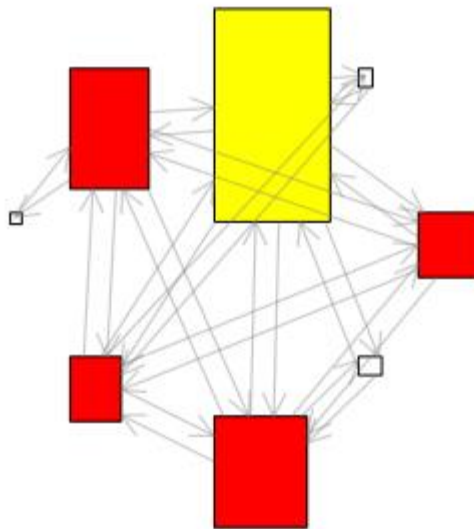


Figura 14: Roles sobrecargados

Plano de tareas

Este plano permite visualizar las tareas a ejecutar en función de sus relaciones, de los productos de trabajo generados y los roles que participan en cada una de estas, los dos patrones analizados son:

Sub-proyectos independientes

El proceso define la forma en la cual las tareas se interrelacionan por lo que no es normal encontrar tareas aisladas, siendo lo anterior, síntoma de una mala especificación. Durante la construcción del trabajo se identificaron cinco tareas aisladas, ver Figura 15: Sub-proyectos independientes:

- Ejecución del sprint/Documentación
- Cambios en el sistema/Análisis del cambio
- Gerencia de proyectos/Envío confirmación participación
- Gerencia de proyectos/Recepción de requisitos
- Cambios en el sistema/Recepción de solicitud de cambio

El análisis de esta situación permitió identificar problemas durante la especificación del proceso dentro de EPF Composer las que luego fueron corregidas. En el plano cada nodo representa una tarea del proceso, su alto y ancho representan la cantidad de productos de trabajo como entradas y salidas respectivamente, mientras que los distintos colores permiten identificar los sub-proyectos identificados.

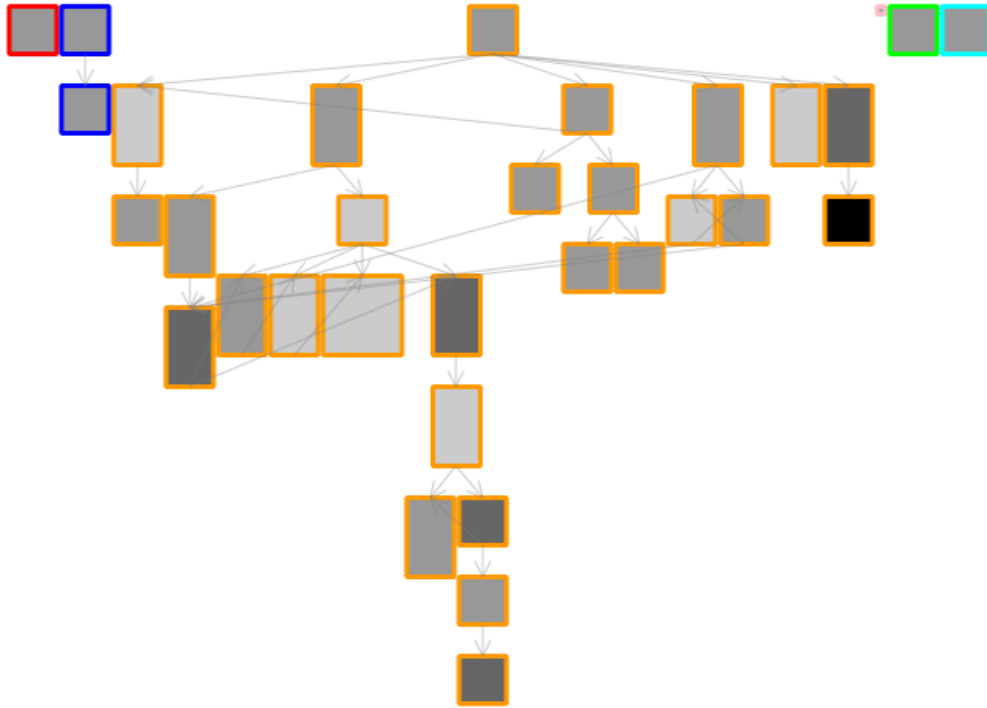


Figura 15: Sub-proyectos independientes

Tareas de propósito múltiple

Este patrón busca identificar tareas de alta complejidad, definiendo que estas son las que no tienen un objetivo único. La forma en la cual se identifican estas tareas es a través de la cantidad de artefactos que generan, un número alto podría indicar la necesidad de replantear la tarea siendo necesario rediseñar esta. La Figura 16: Tareas de propósito múltiple, muestra que la tarea “Ejecución de pruebas unitarias” presenta diferencias con respecto al resto de los elementos, este problema fue abordado y corregido luego de su detección.

Cada nodo representa una tarea del proceso, su alto y ancho representan la cantidad de productos de trabajo como entradas y salidas respectivamente mientras que los colores identifican las tareas multi-propósito.

Plano de productos de trabajo

Este plano permite visualizar los artefactos generados durante el proceso, los patrones analizados son:

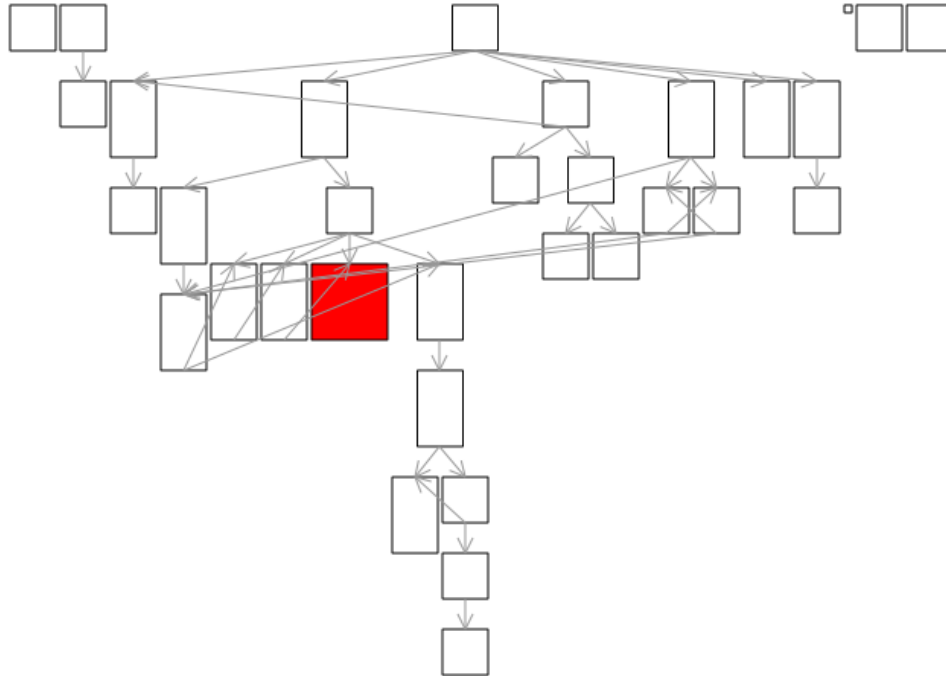


Figura 16: Tareas de propósito múltiple

Sub-proyectos independientes

Al igual que el plano de tareas, los productos de trabajo deben estar interconectados ya que la ausencia de relación indica un problema de definición. Durante la evaluación de este patrón se identificaron algunos productos de trabajo con problemas:

- Sistema productivo
- Propuesta técnica
- Propuesta económica
- Manual de usuario
- Evidencia de pruebas
- Bitácora de paso a producción

Luego del análisis se identificó que:

- Los artefactos “Sistema productivo” y “Evidencia de pruebas” se encuentran empaquetados dentro del entregable “Sistema validado” que es un producto de la tarea “Ejecución de pruebas”.
- Los artefactos “Propuesta técnica” y “Propuesta económica” se encuentran empaquetados dentro del entregable “Propuesta”.
- Los artefactos “Bitácora de paso a producción” y “Manual de usuario” se encuentran empaquetados dentro del entregable “Sistema”.

En la Figura 17: “Sub-proyectos independientes” cada elemento representa un producto de trabajo mientras que el ancho y alto representa la cantidad de tareas que usan y generan estos respectivamente.

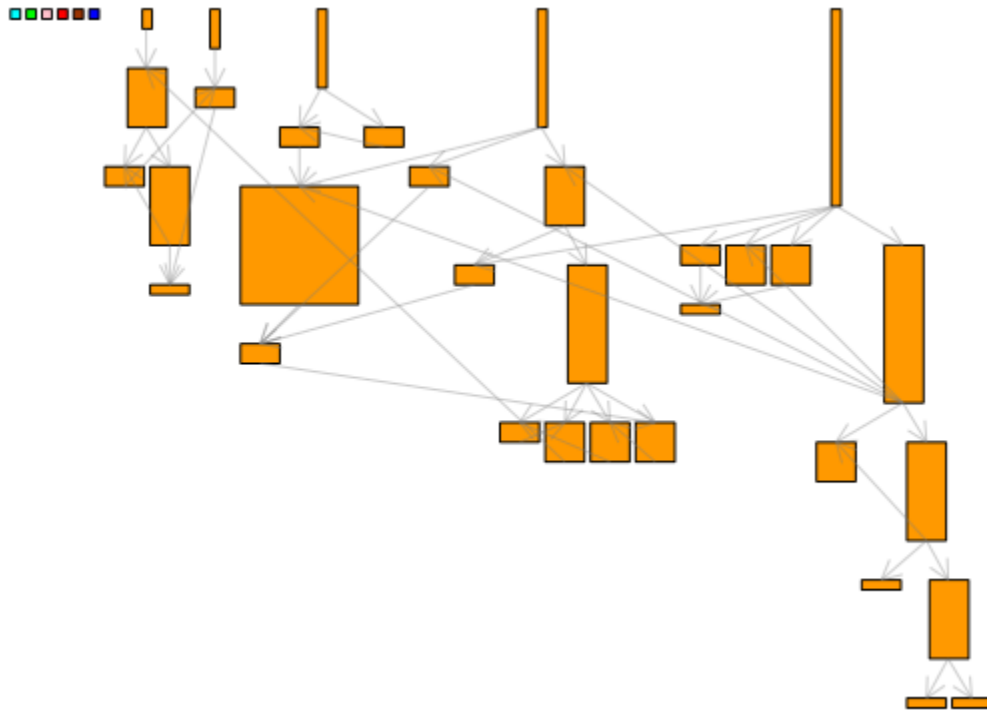


Figura 17: Sub-proyectos independientes

Artefactos de entrada para muchas tareas

Cuando un artefacto es requerido por un número alto de tareas puede entenderse como un posible problema debido al efecto que puede causar su no disponibilidad. Durante el análisis se identificaron seis artefactos con esta característica:

- Diseño detallado
- Documento de diseño general
- Requerimiento del cliente
- Plan del proyecto (backlog)
- Documento de requisitos general
- Estimación de esfuerzo

Mientras el artefacto “Diseño detallado” forma parte del proceso “Ejecución del proyecto”, los demás están contenidos en el proceso “Gestión de proyectos”.

En la Figura 18: Artefactos de entrada para muchas tareas, cada elemento representa un producto de trabajo mientras que su ancho y alto representa la cantidad de tareas que usan y generan estos respectivamente. Los colores muestran los elementos que presentan problemas.

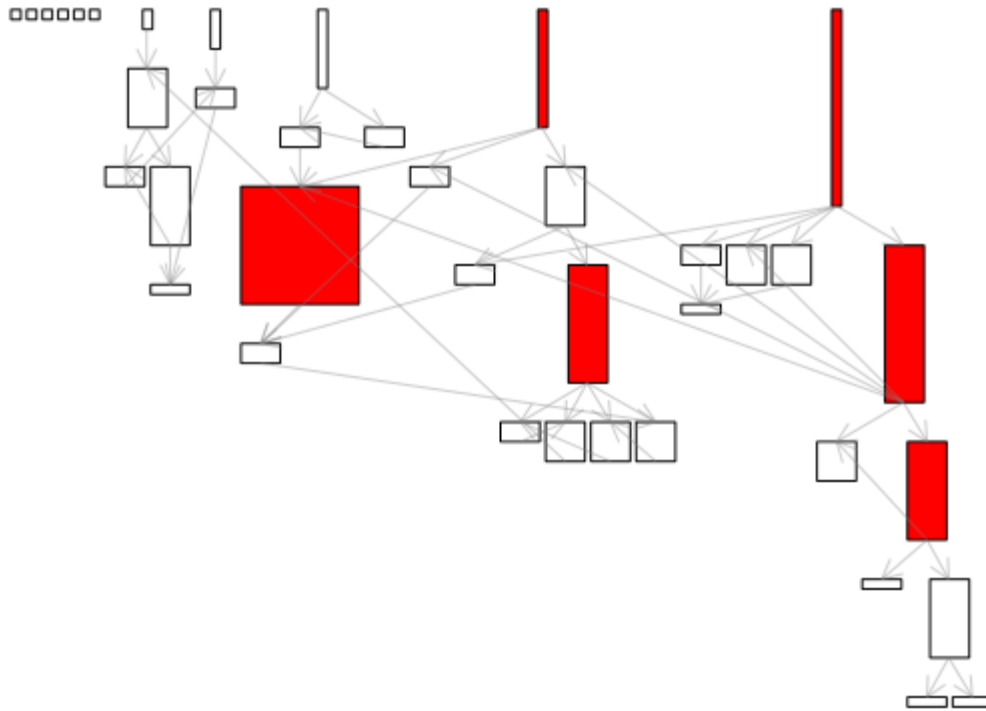


Figura 18: Artefactos de entrada para muchas tareas

El artefacto “Diseño detallado” fue analizado y se identificó su relación con cuatro tareas, tareas con relación de dependencia entre sí por lo que el efecto cuello de botella se puede descartar.

El análisis del resto de los artefactos permitió confirmar el riesgo en la aparición del efecto cuello de botella durante el proceso de gestión de proyecto, por lo anterior es que se mantiene especial cuidado y monitoreo durante las tareas afectadas.

Artefactos residuales

Los artefactos que no son usados como entrada para tareas deben ser de tipo entregables, en caso existan artefactos que no cumplan con la condición anterior deben ser considerados productos de trabajo residuales y debe considerarse analizar posibles problemas de diseño. Durante el análisis se detectaron cuatro productos de trabajo que tienen esta característica:

- Estado participación.
- Asignación del jefe de proyecto.
- Incorporación nuevo personal.
- Incorporación personal de planta.

El artefacto “Estado participación” corresponde a la confirmación que la compañía envía al cliente con el objeto de declarar la participación en el proceso de licitación, mientras que los tres restantes artefactos “Asignación jefe de proyecto”, “Incorporación nuevo personal” e “Incorporación personal de planta” son tareas relacionadas a la construcción del equipo de trabajo y son hitos que deben ser cubiertos dentro del proyecto.

En la Figura 19: Artefactos residuales, los elementos coloreados en azul representan los artefactos residuales identificados en el proceso.

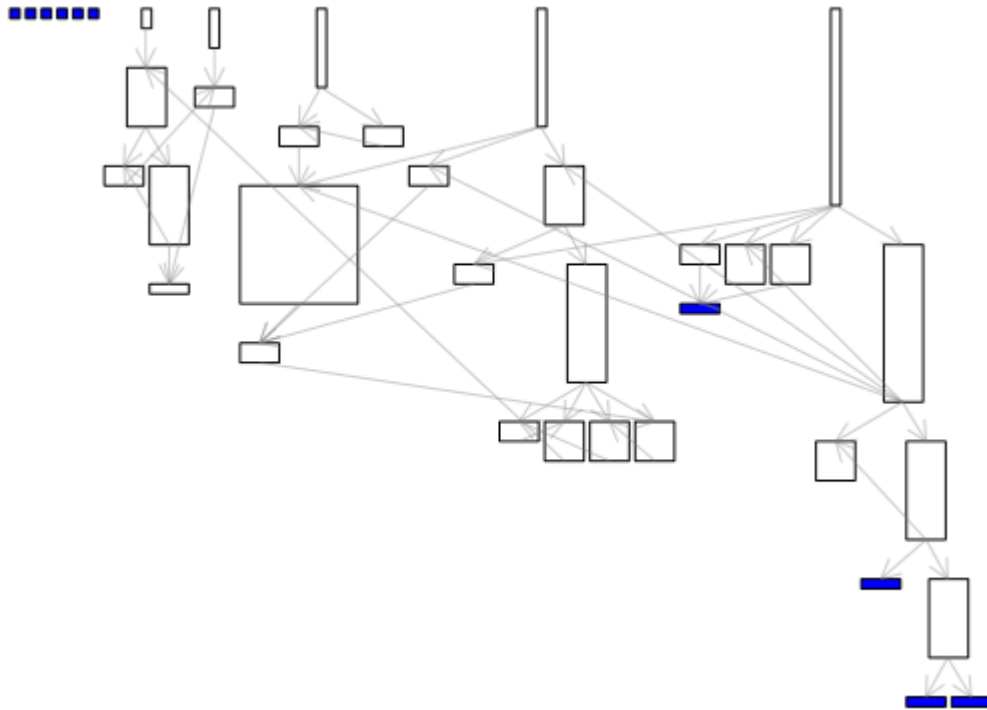


Figura 19: Artefactos residuales

Los demás artefactos identificados forman parte de entregable por lo que no fueron considerados en el análisis:

- Sistema productivo
- Propuesta técnica
- Propuesta económica
- Manual de usuario
- Evidencia de pruebas
- Bitácora de paso a producción

Por último, durante esta actividad se presentó el resultado obtenido a los equipos de trabajo con el objeto de discutir este. El objetivo principal de esta actividad fue entregar el conocimiento necesario a los equipos de trabajo para que estos puedan adoptar las nuevas prácticas y proceso de la mejor forma, adaptando el nuevo proceso a sus proyectos según sea necesario. Esta actividad tuvo tiempos de ejecución acotados teniendo en consideración que luego sería necesario prestar asistencia durante la ejecución de los proyectos.

4. Validación del proceso

Ya construido el proceso y junto al resultado de su evaluación es que este fue tomado como modelo para la ejecución de tres proyectos reales. Durante esta etapa se busca evaluar el real impacto del proceso desarrollado.

4.1 Ejecución de proyectos adoptando el proceso y prácticas definidas

Usando el proceso desarrollado se ejecutaron tres proyectos con el objetivo de comparar los resultados con proyectos históricos que no se regían por un proceso formal. Los tres proyectos fueron ejecutados para un único cliente, lo anterior debido a disponibilidad y al profundo conocimiento que se tiene respecto a la forma de trabajo que éste implementa.

Durante esta actividad se ejecutaron las siguientes tareas:

1. Análisis del proyecto, buscando contexto para la definición del proceso con el que se abordará el proyecto.
2. Definición del proceso a implementar, seleccionando los roles, actividades y tareas que se ejecutarán y que se encuentran definidos en el proceso de desarrollo construido.
3. Planificación de la ejecución del proyecto, usando el proceso definido y registrando los datos de ejecución en la bitácora del proyecto para dejar evidencia de los tiempos, incidencias y controles de cambio que se levantaron durante esta actividad.

Los datos obtenidos serán usados como entrada por la siguiente actividad, la que busca evaluar estos resultados.

Proyecto Dashboard

Este proyecto consta del desarrollo de un dashboard Web que tiene por objeto permitir a los ejecutivos de la compañía tener una vista consolidada del estado comercial de sus clientes. Para presentar la información también es necesario el desarrollo de tres servicios en el sistema CORE que entregan información comercial, además de otros tres servicios que permiten comunicar el sistema CORE con la plataforma del cliente. Además de lo anterior, el dashboard se comunica con otros dos sistemas, los que en conjunto permiten consolidar la información principal de los clientes. Las características del proyecto se detallan en la Tabla 3: Información del proyecto.

La propuesta de trabajo consta de dos iteraciones de tres semanas cada una y el proceso definido para el proyecto se muestra en la Figura 20: Proceso proyecto Dashboard, esta definición fue acordada entre la gerencia de proyectos y el equipo de trabajo asignado.

Las tareas abordadas en este proyecto se muestran en la Figura 21: Tareas seleccionadas para el proyecto Dashboard, mientras que los roles participantes se muestran la Figura 22: Roles involucrados en el proyecto Dashboard.


Característica	Valor
Tipo de proyecto	Mediano
Jefe de proyecto	JZ
Desviación del esfuerzo	22%
Metodología usada	Ágil, dos sprints
Incidencias levantadas	3
Controles ca cambio	3

Tabla 3: Información del proyecto Dashboard

Presentation Name	Index	Predecessors
Proceso dashboard	0	
Gerencia de proyectos	1	
Recepción de requisitos.	2	
Análisis de factibilidad.	3	2
Generación de propuesta	4	
Análisis de requisitos	5	
Desarrollo propuesta	6	5
Estimación de esfuerzo	7	5
Envío propuesta	8	6,7
Planificación del proyecto	9	
Generación del backlog	10	
Diseño de la solución	11	
Gestión de riesgos	12	10,11
Definición del proceso de desarrollo	13	12
Estructuración del equipo	14	9
Evaluación necesidades	15	
Asignación jefe de proyecto	16	15
Asignación de personal planta	17	16
Archivo	18	
Planificación de sprint	19	14
Planificación sprint	20	21
Diseño del sprint	21	
Ejecución del sprint	22	19
Coordinación de tareas	23	
Desarrollo	24	23
Ejecución de pruebas unitarias	25	23
Documentación	26	23
Cambios en el sistema	27	14
Recepción de solicitud de cambio.	28	
Análisis de cambio.	29	28
Control de cambio	30	
Selección de funcionalidad a reemplazar	31	29
Cambio de alcance	32	
Agregar nueva funcionalidad al proyecto	33	29
Paso a producción	34	22
Planificación paso a producción	35	
Cierre	36	30,32,34
Generar documento de cierre	37	
Aprobación de cierre	38	37

Figura 20: Proceso proyecto Dashboard

Tasks


 This is a system created custom category for grouping same type elements.

Relationships

Contents	Relationships
	<ul style="list-style-type: none"> • Análisis de requisitos • Asignación jefe de proyecto • Aprobación de cierre • Análisis de factibilidad • Ejecución de pruebas unitarias • Selección de funcionalidad a reemplazar • Desarrollo propuesta • Recepción de requisitos. • Coordinación de tareas • Definición del proceso de desarrollo • Gestión de riesgos • Diseño del sprint • Evaluación necesidades • Envío propuesta • Planificación sprint • Diseño de la solución • Agregar nueva funcionalidad al proyecto • Generar documento de cierre • Asignación de personal planta • Estimación de esfuerzo • Documentación • Recepción de solicitud de cambio. • Análisis de cambio. • Generación del backlog • Desarrollo • Planificación paso a producción

Figura 21: Tareas seleccionadas para el proyecto Dashboard

Roles

 This is a system created custom category for grouping same type elements.

Relationships

Contents	Relationships
	<ul style="list-style-type: none"> • Desarrollador • Operaciones • Gerente de proyecto • Jefe de proyecto • Cliente • Analista.

Figura 22: Roles involucrados en el proyecto Dashboard

Si bien el proyecto fue ejecutado bajo un proceso ágil compuesto de dos sprints, se generó una carta Gantt para cubrir las necesidades de gestión del cliente, la programación consta de 283 horas lineales totalizando 493 horas de trabajo.

Nombre de tarea	Duración	Comienzo	Fin
▲ Proyecto DashBoard	283 horas	lun 6/10/19	mar 7/30/19
▲ Planificación general	47 horas	lun 6/10/19	lun 6/17/19
Generación del backlog	16 horas	lun 6/10/19	mar 6/11/19
Diseño general	27 horas	mié 6/12/19	lun 6/17/19
Listado de Riesgos	4 horas	lun 6/17/19	lun 6/17/19
▲ Sprint 1	140 horas	lun 6/17/19	jue 7/11/19
Diseño del sprint	16 horas	lun 6/17/19	mié 6/19/19
Planificación de tareas	8 horas	mié 6/19/19	jue 6/20/19
Coordinación de tareas	4 horas	jue 6/20/19	vie 6/21/19
Desarrollo servicios sistema CORE	40 horas	vie 6/21/19	vie 6/28/19
Desarrollo dashboard - Panel	24 horas	vie 6/21/19	mié 6/26/19
Desarrollo dashboard - Información cliente	24 horas	mié 6/26/19	lun 7/1/19
Desarrollo dashboard - Promociones	24 horas	lun 7/1/19	jue 7/4/19
Pruebas integrales	16 horas	jue 7/4/19	lun 7/8/19
Documentación	16 horas	lun 7/8/19	mié 7/10/19
Planificación paso a producción	8 horas	mié 7/10/19	jue 7/11/19
▲ Sprint 2	192 horas	vie 6/21/19	vie 7/26/19
Diseño del sprint	16 horas	vie 6/21/19	mar 6/25/19
Planificación de tareas	8 horas	mar 6/25/19	mié 6/26/19
Coordinación de tareas	4 horas	mié 6/26/19	mié 6/26/19
Desarrollo servicios en API	32 horas	vie 6/28/19	jue 7/4/19
Desarrollo dashboard - Excesos	24 horas	lun 7/8/19	jue 7/11/19
Desarrollo dashboard - Licencias médicas	24 horas	jue 7/11/19	mié 7/17/19
Desarrollo dashboard - Créditos	16 horas	mié 7/17/19	vie 7/19/19
Pruebas integrales	16 horas	vie 7/19/19	mar 7/23/19
Documentación	16 horas	mar 7/23/19	jue 7/25/19
Planificación paso a producción	8 horas	jue 7/25/19	vie 7/26/19
▲ Cierre	16 horas	vie 7/26/19	mar 7/30/19
Documentación de cierre	16 horas	vie 7/26/19	mar 7/30/19

Figura 23: Planificación proyecto Dashboard

Proyecto Prepago

Este proyecto tiene por objetivo el permitir al cliente entregar un nuevo producto, de tipo bancario, a sus afiliados. Para cumplir con lo anterior es necesario ampliar el CORE financiero con que el cliente opera, necesitando para esto conocimiento específico que como empresa poseemos. Dentro del proyecto existen otros proveedores por lo que la gestión entre estos es clave para cumplir con los plazos estipulados. La construcción necesaria consta en las transacciones estándar necesarias para operar con cuentas bancarias. Las principales características de este proyecto se muestran en la Tabla 4: Información del proyecto Prepago.

La propuesta de trabajo fue acordada con el cliente definiendo un modelo de tipo cascada, por lo anterior es que las actividades y tareas definidas para este proyecto se muestran a continuación.

El proceso definido para este proyecto se muestra en la Figura 24: Proceso proyecto Prepago, y este fue acordado entre la gerencia de proyectos y el equipo de trabajo asignado.

Característica	Valor
Tipo de proyecto	Mediano
Jefe de proyecto	DA
Desviación del esfuerzo	40%
Metodología usada	Tradicional, cascada
Incidencias levantadas	4
Controles ca cambio	10

Tabla 4: Información del proyecto Prepago

Presentation Name	Ind...	Predecessors
Proceso	0	
Gerencia de proyectos	1	
Recepción de requisitos.	2	
Análisis de factibilidad.	3	2
Generación de propuesta	4	
Análisis de requisitos	5	
Desarrollo propuesta	6	5
Estimación de esfuerzo	7	5
Envío propuesta	8	6,7
Planificación del proyecto	9	
Generación del backlog	10	
Diseño de la solución	11	
Gestión de riesgos	12	10,11
Definición del proceso de desarrollo	13	12
Estructuración del equipo	14	9
Evaluación necesidades	15	
Asignación jefe de proyecto	16	15
Asignación de personal planta	17	16
Archivo	18	
Ejecución del sprint	19	14
Coordinación de tareas	20	
Desarrollo	21	20
Ejecución de pruebas unitarias	22	20
Documentación	23	20
QA	24	19
Planificación de prueba	25	
Ejecución de pruebas	26	25
Cambios en el sistema	27	14
Recepción de solicitud de cambio.	28	
Análisis de cambio.	29	28
Control de cambio	30	
Selección de funcionalidad a reemplazar	31	29
Cambio de alcance	32	
Agregar nueva funcionalidad al proyecto	33	29
Paso a producción	34	24
Planificación paso a producción	35	
Implementación en producción	36	35
Cierre	37	30,32,34
Generar documento de cierre	38	
Aprobación de cierre	39	38

Figura 24: Proceso proyecto Prepago

Tasks



This is a system created custom category for grouping same type elements.

Expand All Sections

Collapse All Sections

Relationships	
Contents	<ul style="list-style-type: none">• Planificación de prueba• Análisis de requisitos• Asignación jefe de proyecto• Aprobación de cierre• Análisis de factibilidad• Ejecución de pruebas unitarias• Selección de funcionalidad a reemplazar• Desarrollo propuesta• Recepción de requisitos.• Coordinación de tareas• Definición del proceso de desarrollo• Gestión de riesgos• Evaluación necesidades• Ejecución de pruebas• Envío propuesta• Diseño de la solución• Agregar nueva funcionalidad al proyecto• Implementación en producción• Generar documento de cierre• Asignación de personal planta• Estimación de esfuerzo• Documentación• Recepción de solicitud de cambio.• Análisis de cambio.• Generación del backlog• Desarrollo• Planificación paso a producción

Figura 25: Tareas seleccionadas para el proyecto Prepago

Roles



This is a system created custom category for grouping same type elements.

Expand All Sections

Collapse All Sections

Relationships	
Contents	<ul style="list-style-type: none">• Desarrollador• Operaciones• Gerente de proyecto• Jefe de proyecto• Cliente• Analista.• Analista de QA

Figura 26: Roles involucrados en el proyecto Prepago

La propuesta de trabajo consta de 284 horas lineales, con un total de 476 horas totales de trabajo.

Nombre de tarea	Duración	Comienzo	Fin
Proyecto Prepago	284 horas	lun 7/29/19	lun 9/16/19
Planificación general	76 horas	lun 7/29/19	vie 8/9/19
Generación del backlog	24 horas	lun 7/29/19	mié 7/31/19
Diseño general	24 horas	jue 8/1/19	lun 8/5/19
Listado de riesgos	8 horas	mar 8/6/19	mar 8/6/19
Planificación de tareas	16 horas	mié 8/7/19	jue 8/8/19
Coordinación de tareas	4 horas	vie 8/9/19	vie 8/9/19
Ejecución	144 horas	vie 8/9/19	mié 9/4/19
Construcción integración OIDC	32 horas	vie 8/9/19	jue 8/15/19
Construcción componente puente	40 horas	jue 8/15/19	jue 8/22/19
Construcción panel consulta de cuentas	24 horas	vie 8/9/19	mié 8/14/19
Construcción panel estado resultado	16 horas	mié 8/14/19	vie 8/16/19
Construcción componente desembolso	16 horas	vie 8/16/19	mar 8/20/19
Construcción componente de cargo	16 horas	mar 8/20/19	jue 8/22/19
Construcción componente consulta cuentas	8 horas	jue 8/22/19	vie 8/23/19
Construcción componente estado	8 horas	vie 8/23/19	lun 8/26/19
Integración de componentes	24 horas	lun 8/26/19	jue 8/29/19
Pruebas integrales	32 horas	jue 8/29/19	mié 9/4/19
QA	32 horas	mié 9/4/19	mar 9/10/19
Creación plan de pruebas	16 horas	mié 9/4/19	vie 9/6/19
Ejecución de pruebas	16 horas	vie 9/6/19	mar 9/10/19
Paso a producción	16 horas	mar 9/10/19	jue 9/12/19
Consolidación de artefactos	8 horas	mar 9/10/19	mié 9/11/19
Apoyo paso a producción	8 horas	mié 9/11/19	jue 9/12/19
Cierre	16 horas	jue 9/12/19	lun 9/16/19
Documentación de cierre	16 horas	jue 9/12/19	lun 9/16/19

Figura 27: Planificación proyecto Prepago

Proyecto Excedentes

El objetivo de este proyecto es permitir al cliente alinearse con las normativas gubernamentales que buscan la devolución de dineros pertenecientes a sus afiliados. Para esto se requiere la construcción de un proceso autónomo que se encargue de la gestión de estos dineros.

Los detalles del proyecto se muestran en la Tabla 5: Información del proyecto Excedentes.

Característica	Valor
Tipo de proyecto	Pequeño
Jefe de proyecto	DA
Desviación del esfuerzo	18%
Metodología usada	Tradicional, cascada
Incidencias levantadas	2
Controles ca cambio	0

Tabla 5: Información del proyecto Excedentes

En coordinación con el cliente se decidió trabajar bajo un modelo cascada, las actividades y tareas

definidas se muestran en la Figura 28: Proceso proyecto Excedentes.

Presentation Name	Ind...	Predecessors
Proceso	0	
Gerencia de proyectos	1	
Recepción de requisitos.	2	
Análisis de factibilidad.	3	2
Generación de propuesta	4	
Análisis de requisitos	5	
Desarrollo propuesta	6	5
Estimación de esfuerzo	7	5
Envío propuesta	8	6,7
Planificación del proyecto	9	
Generación del backlog	10	
Diseño de la solución	11	
Gestión de riesgos	12	10,11
Definición del proceso de desarrollo	13	12
Estructuración del equipo	14	9
Evaluación necesidades	15	
Asignación jefe de proyecto	16	15
Asignación de personal planta	17	16
Archivo	18	
Planificación de sprint	19	14
Planificación sprint	20	21
Diseño del sprint	21	
Ejecución del sprint	22	19
Coordinación de tareas	23	
Desarrollo	24	23
Ejecución de pruebas unitarias	25	23
Documentación	26	23
Cambios en el sistema	27	14
Recepción de solicitud de cambio.	28	
Análisis de cambio.	29	28
Control de cambio	30	
Selección de funcionalidad a reemplazar	31	29
Cambio de alcance	32	
Agregar nueva funcionalidad al proyecto	33	29
Paso a producción	34	22
Planificación paso a producción	35	
Cierre	36	30,32,34
Generar documento de cierre	37	
Aprobación de cierre	38	37

Figura 28: Proceso proyecto Excedentes

Para este proyecto se seleccionaron las tareas mostradas en la Figura 29: Tareas seleccionadas para el proyecto Excedentes, mientras que los roles involucrados se muestran en la Figura 30: Roles involucrados en el proyecto Excedentes.

Tasks



This is a system created custom category for grouping same type elements.

[Expand All Sections](#) [Collapse All Sections](#)

Relationships	
Contents	<ul style="list-style-type: none">• Análisis de requisitos• Asignación jefe de proyecto• Aprobación de cierre• Análisis de factibilidad• Ejecución de pruebas unitarias• Selección de funcionalidad a reemplazar• Desarrollo propuesta• Recepción de requisitos.• Coordinación de tareas• Definición del proceso de desarrollo• Gestión de riesgos• Diseño del sprint• Evaluación necesidades• Envío propuesta• Planificación sprint• Diseño de la solución• Agregar nueva funcionalidad al proyecto• Generar documento de cierre• Asignación de personal planta• Estimación de esfuerzo• Documentación• Recepción de solicitud de cambio.• Análisis de cambio.• Generación del backlog• Desarrollo• Planificación paso a producción

Figura 29: Tareas seleccionadas para el proyecto Excedentes

Roles



This is a system created custom category for grouping same type elements.

[Expand All Sections](#) [Collapse All Sections](#)

Relationships	
Contents	<ul style="list-style-type: none">• Desarrollador• Operaciones• Gerente de proyecto• Jefe de proyecto• Cliente• Analista.

Figura 30: Roles involucrados en el proyecto Excedentes

La propuesta de trabajo consta de 156 horas lineales, totalizando 228 horas de trabajo. La planificación generada se muestra a continuación.

Nombre de tarea	Duración	Comienzo	Fin
▲ Proyecto Excedentes	156 horas	lun 9/2/19	mié 10/2/19
▲ Planificación general	32 horas	lun 9/2/19	jue 9/5/19
Generación del backlog	16 horas	lun 9/2/19	mar 9/3/19
Diseño general	8 horas	mié 9/4/19	mié 9/4/19
Listado de riesgos	4 horas	jue 9/5/19	jue 9/5/19
Coordinación de tareas	4 horas	jue 9/5/19	jue 9/5/19
▲ Ejecución	88 horas	vie 9/6/19	mié 9/25/19
Ampliación componente puente	16 horas	vie 9/6/19	lun 9/9/19
Construcción proceso recolección de cuentas	24 horas	mar 9/10/19	jue 9/12/19
Construcción componente mandato	16 horas	vie 9/13/19	lun 9/16/19
Integración módulo abono en cuenta	8 horas	mar 9/17/19	mar 9/17/19
Pruebas integrales	24 horas	lun 9/23/19	mié 9/25/19
Documentación	8 horas	lun 9/23/19	lun 9/23/19
▲ QA	32 horas	mar 9/24/19	vie 9/27/19
Creación plan de pruebas	16 horas	mar 9/24/19	mié 9/25/19
Ejecución de pruebas	16 horas	jue 9/26/19	vie 9/27/19
▲ Paso a producción	12 horas	lun 9/30/19	mar 10/1/19
Consolidación de artefactos	8 horas	lun 9/30/19	lun 9/30/19
Apoyo paso a producción	4 horas	mar 10/1/19	mar 10/1/19
▲ Cierre	8 horas	mar 10/1/19	mié 10/2/19
Documentación de cierre	8 horas	mar 10/1/19	mié 10/2/19

Figura 31: Planificación proyecto Excedentes

4.2 Validación de los resultados

Para la validación de los resultados se tabularon los datos históricos regenerados, datos que servirán para evaluar los objetivos de este trabajo. Es necesario aclarar que la desviación del esfuerzo considera los tiempos imputados en la resolución de las incidencias levantadas, mientras que los controles de cambios no afectan los tiempos del proyecto planificado. La Tabla 6: Tabulación de proyectos históricos muestra los datos promedios de los datos históricos según el tipo de proyecto y que se calculan a partir de los datos de la Tabla 2: Regeneración de datos históricos.

Tipo de proyecto	Incidencias	Controles de cambio	Desviación de esfuerzo
Medianos	3	5	35%
Pequeños	1	5	26%

Tabla 6: Tabulación de proyectos históricos

Una vez terminada la ejecución de los tres proyectos, los datos recolectados fueron tabulados para ser comparados con la información histórica regenerada, el objetivo de esta actividad es evaluar el real impacto del trabajo realizado y el resultado se presenta en la Tabla 7: Tabulación de datos recolectados nuevos proyectos.

Proyecto	Incidencias	Controles de cambio	Desviación de esfuerzo
Dashboard	3	3	22%
Prepago	4	10	40%
Excedentes	2	0	18%

Tabla 7: Tabulación de datos recolectados nuevos proyectos

Las bitácoras resumidas por proyecto son las siguientes:

Bitácora resumida proyecto Dashboard

La desviación del esfuerzo para este proyecto, de tamaño mediano, fue de un 22%, atraso que supera el máximo de 20% definido en el objetivo de este trabajo pero que refleja una reducción considerable respecto al atraso medio histórico. Según se notó durante la ejecución del proyecto, la desviación se produce debido a diferencias de alcance en requerimientos específicos, siendo necesario ejecutar nuevas tareas de análisis y su posterior ejecución. La ejecución planificada y real se muestra en la Figura 32: Bitácora proyecto Dashboard.

Nombre de tarea	Duración	Comienzo	Fin	HH Reales
Proyecto DashBoard	407	lun 10/06/19	mar 30/07/19	495
Planificación general	47	lun 10/06/19	lun 17/06/19	63
Sprint 1	180	lun 17/06/19	jue 11/07/19	196
Sprint 2	164	vie 21/06/19	vie 26/07/19	212
Cierre	16	vie 26/07/19	mar 30/07/19	24

Figura 32: Bitácora proyecto Dashboard

Bitácora resumida proyecto Prepago

La desviación del esfuerzo para este proyecto, de tamaño mediano, fue de un 40%, atraso que supera el máximo de 20% definido en el objetivo de este trabajo y también situándose por sobre el atraso medio histórico. La desviación nace a partir de una mala especificación inicial, que involucró cambios en la tecnología a usar y también debido a problemas de coordinación entre las distintas partes involucradas. La cantidad de controles de cambios del proyecto también refleja las dificultades que el cliente tuvo durante su etapa de análisis, lo que también impactó en las planificaciones de los proveedores. La ejecución planificada y real se muestra en la Figura 33: Bitácora proyecto Prepago.

Nombre de tarea	Duración	Comienzo	Fin	HH Reales
Proyecto Prepago	356	lun 29/07/19	lun 16/09/19	500
Planificación general	76	lun 29/07/19	vie 09/08/19	92
Ejecución	216	vie 09/08/19	mié 04/09/19	264
QA	32	mié 04/09/19	mar 10/09/19	64
Paso a producción	16	mar 10/09/19	jue 12/09/19	40
Cierre	16	jue 12/09/19	lun 16/09/19	40

Figura 33: Bitácora proyecto Prepago

Bitácora resumida proyecto Excedentes

La desviación del esfuerzo para este proyecto, de tamaño pequeño, fue de un 18%, atraso que supera el

máximo de 10% definido en el objetivo de este trabajo y muy cerca de la media histórica. La desviación de este proyecto fue causada principalmente por la falta de conocimiento que el equipo de trabajo tenía sobre el producto afectado, problemática levantada durante la construcción del equipo de trabajo y que, si bien fue abordada durante la ejecución, queda claro que las medidas tomadas no fueron suficientes. La ejecución planificada y real se muestra en la Figura 34: Bitácora proyecto Excedentes.

Nombre de tarea	Duración	Comienzo	Fin	HH Reales
Proyecto Excedentes	180	lun 02/09/19	mié 02/10/19	212
Planificación general	32	lun 02/09/19	jue 05/09/19	36
Ejecución	96	vie 06/09/19	mié 25/09/19	116
QA	32	mar 24/09/19	vie 27/09/19	40
Paso a producción	12	lun 30/09/19	mar 01/10/19	12
Cierre	8	mar 01/10/19	mié 02/10/19	8

Figura 34: Bitácora proyecto Excedentes

4.3 Análisis de los resultados

En base al levantamiento de los datos históricos y junto a las bitácoras de los proyectos ejecutados bajo el nuevo proceso de desarrollo es que se genera la Tabla 8: Resumen de los resultados, que contiene la evaluación de los resultados obtenidos.

Tipo de proyecto	Incidencias (1)	Controles de cambio (2)	Desviación de esfuerzo (3)	Incidencias (4)	Controles de cambio (5)	Desviación de esfuerzo (6)	Desviación de esfuerzo máxima esperada
Medianos	3	5	35%	3	6	31%	20%
Pequeños	1	5	26%	2	0	18%	10%

Tabla 8: Resumen de los resultados

Dónde:

- (1) Incidencias históricas medias.
- (2) Controles de cambios históricos medios.
- (3) Desviación de esfuerzo histórico.
- (4) Incidencias medias bajo el proceso de desarrollo.
- (5) Controles de cambios medios bajo el proceso de desarrollo.
- (6) Desviación de esfuerzo bajo el proceso de desarrollo.

De lo anterior y en base a los resultados históricos se destaca lo siguiente:

- La cantidad de incidencias para los proyectos medianos y pequeños se mantiene dentro de la desviación estándar histórica, según lo anterior se entiende que la adopción del proceso no ha provocado mejora o perjuicio en la calidad de los productos desarrollados.
- La cantidad de controles de cambios para los proyectos medianos se mantienen dentro de la desviación estándar histórica, entendiendo a partir de esto que las tareas de especificación de requisitos requieren una mayor profundidad o mejoras en sus prácticas para lograr reducir este valor.
- Respecto a los controles de cambios para proyectos pequeños, durante el desarrollo de este trabajo no fue necesario abordar tareas de este tipo y si bien puede reflejar una mejora es

necesario obtener más datos para ejecutar una evaluación precisa.

- Se puede notar una leve reducción en la desviación en la estimación de esfuerzo, algo más pronunciada para los proyectos pequeños. La claridad y el orden de las tareas y actividades a ejecutar, entre otros, han impactado positivamente en el desempeño por lo que se nota el beneficio de haber adoptado la formalidad del nuevo proceso. También se espera que la experiencia obtenida en cada caso permita mejorar los tiempos y prácticas.

Si bien la versión inicial del proceso desarrollado mostró algunos problemas de especificación, la mayoría de estos fueron resueltos dejando sólo pendiente el patrón de sobrecarga de roles, esto debido a que en la actualidad no es posible aumentar la cantidad de roles involucrados en los proyectos. Teniendo esto en cuenta se descarta que el resultado obtenido haya sido provocado por esta, o al menos se entiende que no es la causa principal del resultado, esto también fundamentado bajo lo declarado por los grupos de trabajo y también considerando que las tareas y artefactos definidos no cambiaron en mayor medida respecto a las actividades que históricamente se han ejecutado.

El otro tema evaluado es el nivel de representatividad de los proyectos piloto abordados, en este caso es necesario considerar que los tres proyectos requirieron de conocimiento específico que en su momento no estuvo disponible. Además, los tres proyectos tuvieron que ser implementados bajo tecnologías y plataformas nuevas tanto para el cliente como para la compañía, y si bien esta condición estuvo contemplada inicialmente, causó problemas no contemplados tanto en la definición de las arquitecturas como en la estabilización de la plataforma. Esta situación provocó atrasos en las entregas durante todas las actividades realizadas por lo que se entiende que el no cumplimiento de los objetivos propuestos se debe principalmente a esta situación.

Si bien existen mejoras en los resultados respecto a los datos históricos, estos reflejan el incumplimiento de los objetivos declarados en este trabajo. En general y luego de evaluar la situación se determinó, junto a la gerencia de proyectos, mantener la ejecución del proceso y aplicar las mejoras necesarias con el objeto de, en el mediano plazo, obtener resultados ajustados a la realidad. Todo lo anterior entendiendo que este es un proceso evolutivo de mejora continua que requiere de plazos mayores para su total adopción.

5. Conclusión

En el desarrollo de este documento se describe la situación de Nexo-IT, PYME que desarrolla software a medida. Luego se declaran los problemas detectados, como lo son la disparidad en la ejecución de los proyectos entre los equipos de trabajo, baja calidad de los productos desarrollados y la imprecisión en la tarea de estimación de esfuerzo.

El trabajo propone como objetivo reducir los atrasos en la ejecución de los proyectos, clasificando estos según la duración estimada. Además, se busca reducir la cantidad de cambios de alcance y controles de cambios que tengan relación con tareas de especificación de requisitos mal generados.

Para cumplir con los objetivos propuestos es que se propone la adopción de un proceso de desarrollo que se acomode a la cultura organizacional de Nexo-IT, se evaluaron las opciones y se optó por definir un proceso propio que tenga como base las características de otros estándares validados en la industria, como lo son ISO/IEC 29110 y OpenUP. El proceso definido fue modelado mediante el uso de la herramienta EPF Composer y luego analizado con la herramienta AVISPA.

Luego de ejecutado el proceso de desarrollo en tres proyectos piloto y en base al análisis de los resultados obtenidos es que queda de manifiesto que los objetivos no fueron logrados, concluyendo que existen dos motivos principales por los cuales se obtuvieron estos resultados:

- Motivos comerciales, que tienen relación a la premura con la que el cliente requiere el inicio de los proyectos, situación que reduce considerablemente los tiempos en las tareas de especificación de requisitos provocando finalmente las desviaciones declaradas. Es necesario notar que en este punto también existe responsabilidad por parte de la compañía dado que somos quienes tomamos la decisión de abordar los proyectos bajo las condiciones que el cliente declara.
- Problemas de coordinación en la disponibilidad del personal, para los tres proyectos piloto fue necesario contar con conocimiento específico, conocimiento que si bien existe dentro de la compañía no estuvo a disposición en las fechas coordinadas. Para estos casos fue necesario iniciar las actividades con personal sin experiencia acabada en las tecnologías ni negocios propios de los proyectos, impactando en los atrasos registrados.

También cabe destacar que las actividades de adopción del nuevo proceso de desarrollo implicaron consumo de tiempos que inicialmente no fueron considerados y que impactaron negativamente en la desviación de tiempos en los proyectos. El problema gatillante de esto fue la resistencia al cambio que algunos participantes de los equipos de trabajo presentaron, en particular al tener que tomar responsabilidad en tareas que anteriormente no eran consideradas como carga de trabajo propias y que son vistas como trabajo extra.

Luego de haber puesto en marcha el proceso y aun cuando el objetivo no fue logrado se pueden destacar los siguientes beneficios:

- La formalidad del proceso permite ejecutar tareas que de otra forma no son ejecutadas objetivamente, un ejemplo de esto es la ejecución de las pruebas unitarias. Durante la ejecución de los proyectos se obtuvieron resultados que permiten identificar posibles problemas, esto con el objetivo de mejorar continuamente las actividades y/o tareas.
- Relacionado al punto anterior, para el equipo de desarrollo se torna más natural y simple la

ejecución de los proyectos debido a que todas las actividades y tareas son conocidas y transversales entre los proyectos. Con esto se evita la sensación de desorientación que los equipos podían sentir cuando debían encarar nuevos proyectos sin un proceso definido.

- El proceso permite transparentar a la gerencia de proyectos la real magnitud de las actividades y tareas necesarias para la ejecución de los proyectos. Este es un punto importante dentro de la compañía ya que evita subestimar el real esfuerzo de trabajo necesario por proyecto.

Como trabajo futuro se declara:

- El compromiso, por parte de la gerencia de proyectos, en la mantención continúa del proceso de desarrollo de forma que este se pueda mantener actualizado y/o complementado siempre que aparezcan nuevas características que sean de importancia para la compañía.
- También se ve necesario generar una instancia de inducción que permita a los nuevos colaboradores tomar conocimiento del proceso, esto para transparentar la responsabilidad basada en roles, actividades y tareas a abordar.

Las principales lecciones aprendidas son:

- Es necesario generar acuerdos de transparencia junto a los clientes con el objetivo de reservar tiempos de trabajo acordados para las tareas de especificación de requisitos.
- Es necesario especificar las guías dado que permiten la continuidad de las tareas y evitan malentendidos entre los participantes del equipo de trabajo. Durante el desarrollo de este trabajo no se definieron la totalidad de las guías siendo luego evidente que es una tarea clave para el equipo de trabajo.
- Definir un conjunto de herramientas colaborativas institucional para el desarrollo del proyecto permite visibilidad y conocimiento general de la situación de los proyectos, a la par que permite compartir conocimiento entre los equipos de trabajo.

Bibliografía

- [1] Margaret K. Kulpa, Kent A. Johnson, 2008. Documentation Guidelines. En: Interpreting the CMMI (R): A Process Improvement Approach. 2ª ed. RC Press. pp. 177-182
- [2] Batista J. and Dias de Figueiredo A., 2000. “SPI in a Very Small Team: a Case with CMM”. En: Software Process Improvement and Practice. pp.1-4
- [3] CMMI Product Team, 2010, CMMI for Development, Version 1.3 [en línea] <http://resources.sei.cmu.edu/library/asset-view.cfm?assetid=9661> [consulta: 22 Junio 2017]
- [4] International Organization for Standardization, 2012, Information technology -- Process assessment - - Part 5: An exemplar software life cycle process assessment model [en línea] <https://www.iso.org/standard/60555.html> [consulta: 20 Junio 2017]
- [5] International Organization for Standardization, 2014, Systems and software engineering — Lifecycle profiles for Very Small Entities (VSEs) [en línea] <https://www.iso.org/obp/ui/#iso:std:iso-iec:tr:29110:-5-6-1:ed-1:v1:en> [consulta: 20 Junio 2017]
- [6] **OpenUP Team**, Introduction to OpenUP (Open Unified Process): [en línea] <http://epf.eclipse.org/wikis/openup/> [consulta: 16 Junio 2017]
- [7] ISO/IEC TR 29110-1:2016(E) Part 1: Overview. pp 12-14
- [8] ISO/IEC TR 29110-5-6-2:2014(E) Part 5: Systems engineering — Management and engineering guide: Generic profile group: Basic profile. pp. 5-7
- [9] Eclipse Foundation [en línea] https://www.eclipse.org/epf/openup_component/openup_vision.php [consulta: 6 Abril 2019]
- [10] Eclipse Foundation [en línea] <https://projects.eclipse.org/projects/technology.epf> [consulta: 6 Abril 2019]
- [11] Yingxu Wang, Graham King, 2000, Software Engineering Processes: Principles and Applications. s.l.: CRC Press. pp 15-19
- [12] Object Management Group, 2002, “Software Process Engineering Metamodel Specification”; An Adopted Specification of the Object Management Group, Inc; Version 1.0. pp. 1-10
- [13] B. Combemale, A. Caplain, X. Crégut y B. Coulette, 2006, “Towards a rigorous use of SPEM”, ICEIS'06, Paphos(Cyprus), INSTICC. pp. 1-4
- [14] A. Järvi y T. Mäkilä, 2005, “Observations on modeling software processes with SPEM process components”, Proceedings of The 9th Symposium on Programming Languages and Software Tools. pp. 2-3
- [15] Object Management Group® (OMG®) [en línea] <https://www.omg.org/spec/SPEM/2.0/PDF> [consulta: 27 Abril 2019]
- [16] Julio A. Hurtado Alegría, María Cecilia Bastarrica, Alexandre Bergel, 2014, “AVISPA: a tool for

analyzing software process models”, JOURNAL OF SOFTWARE-EVOLUTION AND PROCESS

Anexos

Anexo A

Proceso 1: Gestión de proyectos (GP)

Este proceso tiene por objeto principal abordar las tareas comerciales para reducir este tipo de tareas durante la ejecución del proyecto. Las fases por ejecutar son:

Fase 1: Desarrollo de la propuesta

Durante esta fase se evaluará la factibilidad técnica y económica de la compañía para abordar el proyecto requerido por el cliente. Además, se genera un plan de proyecto a alto nivel, ejecutando un levantamiento general de los requisitos que permita estimar el esfuerzo de trabajo necesario, las actividades ejecutadas dentro de esta fase son:

Gerencia de proyectos

Esta actividad tiene por objetivo evaluar las nuevas oportunidades de negocio. Dentro de esta actividad se evalúa la factibilidad técnica y económica del proyecto dentro del contexto de la compañía, la Figura 35: Tareas dentro de Gerencia de proyectos, muestra el diagrama de esta actividad.

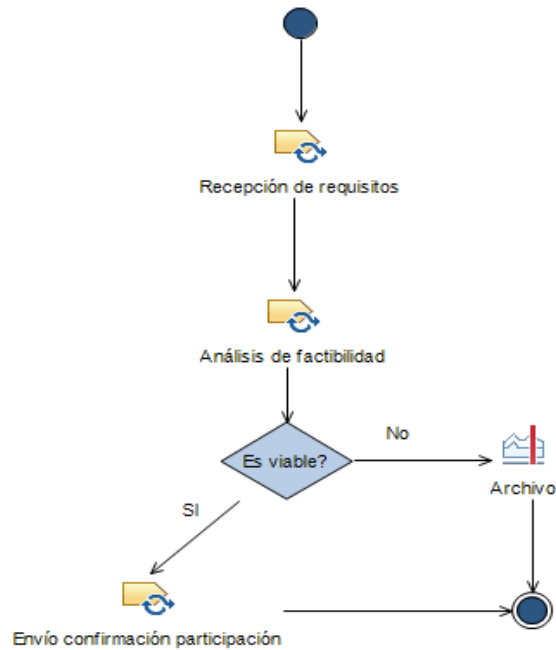


Figura 35: Tareas dentro de Gerencia de proyectos

La ejecución de esta actividad permite, principalmente, reducir el riesgo que conlleva abordar proyectos alejados a la experiencia técnica y capacidad económica que la compañía posee. En caso de que el proyecto no sea viable se responde al cliente y el proyecto pasa al archivo común.

La Figura 36: Detalle de la actividad “Gerencia de proyectos”, muestra los roles, tareas y artefactos involucrados en esta actividad.

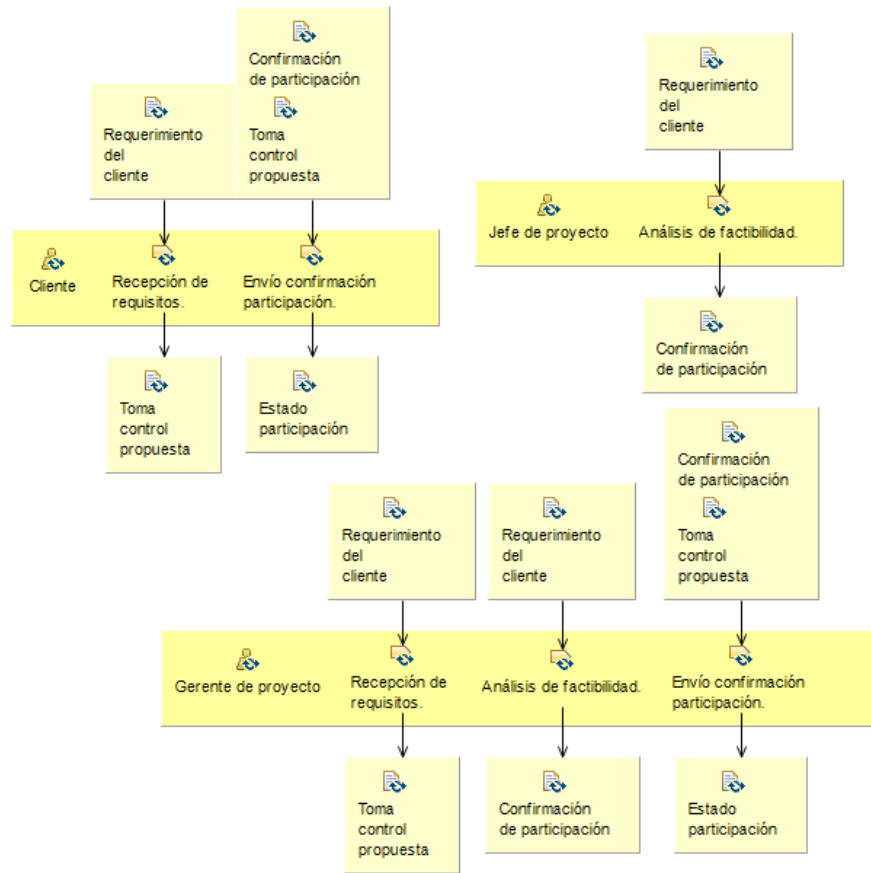


Figura 36: Detalle de la actividad “Gerencia de proyectos”

Generación de propuesta

En caso el proyecto sea viable se procede a generar la propuesta que define el alcance del proyecto y que luego será enviada al cliente. Las tareas dentro de esta actividad están enfocadas en estimar el esfuerzo y costos del proyecto, la estructura de la actividad se presenta en la Figura 37: Tareas dentro de Generación de propuesta. La Figura 38: Detalle de la actividad Generación de propuesta muestra el detalle de esta, roles, tareas y artefactos involucrados.

contratación de nuevo personal. La especificación de los requisitos ejecutada en esta etapa sirve como entrada para la ejecución del proyecto. Las actividades dentro de esta fase son:

Planificación del proyecto

Las tareas de esta actividad están enfocadas en entregar al equipo de desarrollo la información necesaria para abordar el proyecto y así tener claridad del objetivo de este. La Figura 39: Tareas dentro de Planificación de proyecto, representa esta actividad.

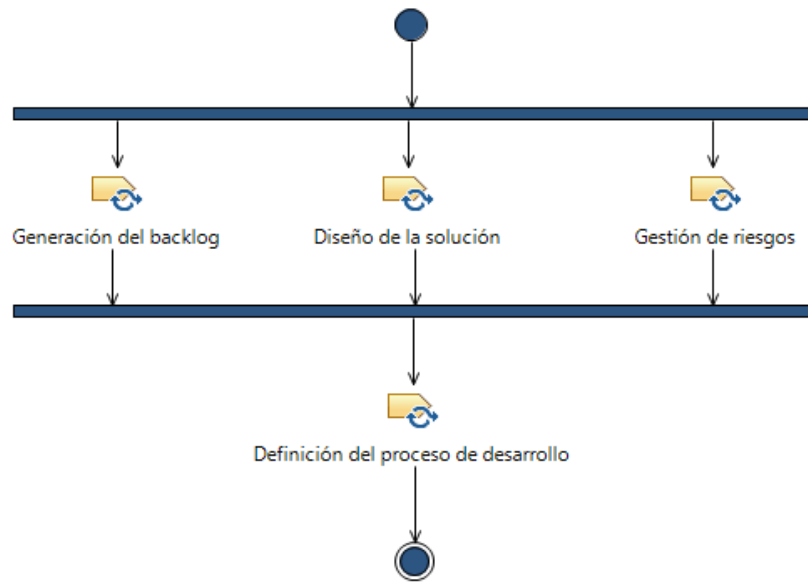


Figura 39: Tareas dentro de Planificación de proyecto

La Figura 40: Detalle de la actividad Planificación del proyecto, muestra los roles involucrados en esta actividad junto a las tareas y artefactos asociados.

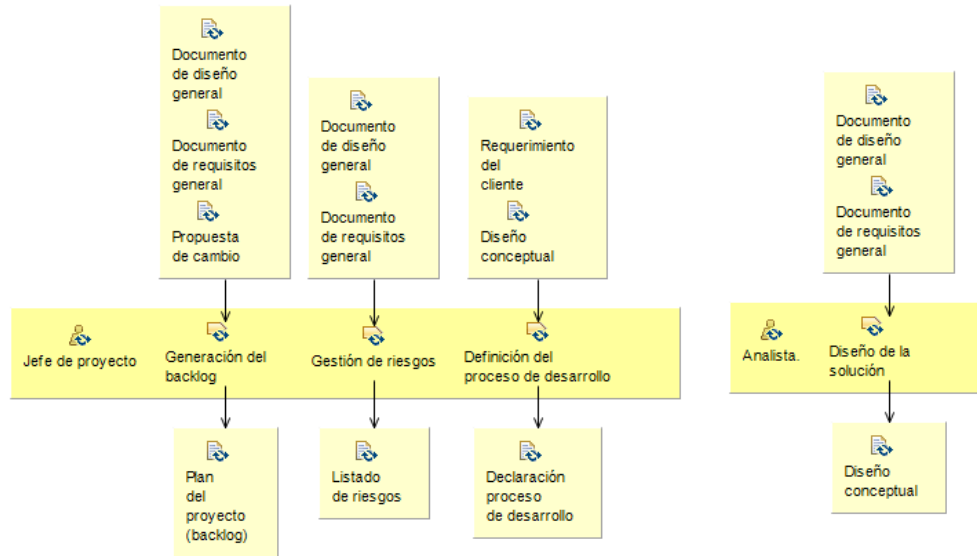


Figura 40: Detalle de la actividad Planificación del proyecto

Estructuración del equipo de trabajo

El objetivo de esta actividad es conformar el equipo de trabajo según las necesidades del proyecto. Lo primero es evaluar la necesidad de adquirir componentes de terceros para luego proceder a la asignación del jefe de proyectos. Luego se evalúa la situación actual del personal de planta asignando según disponibilidad, si fuese necesario se procede a la contratación de nuevo personal según las características del proyecto. La Figura 41: Tareas dentro de Estructuración del equipo de trabajo, muestra el diagrama de esta actividad.

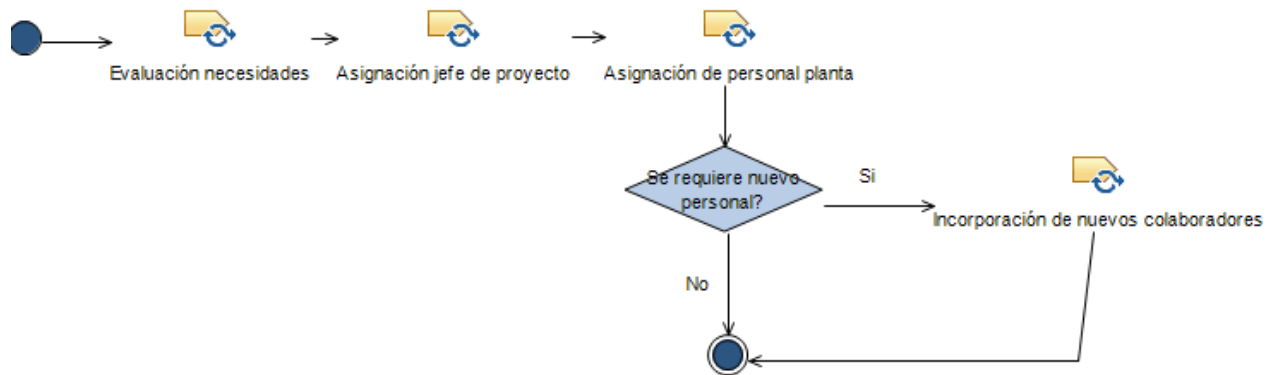


Figura 41: Tareas dentro de Estructuración del equipo de trabajo

La Figura 42: Detalle de la actividad Estructuración del equipo de trabajo, muestra el detalla de esta actividad.

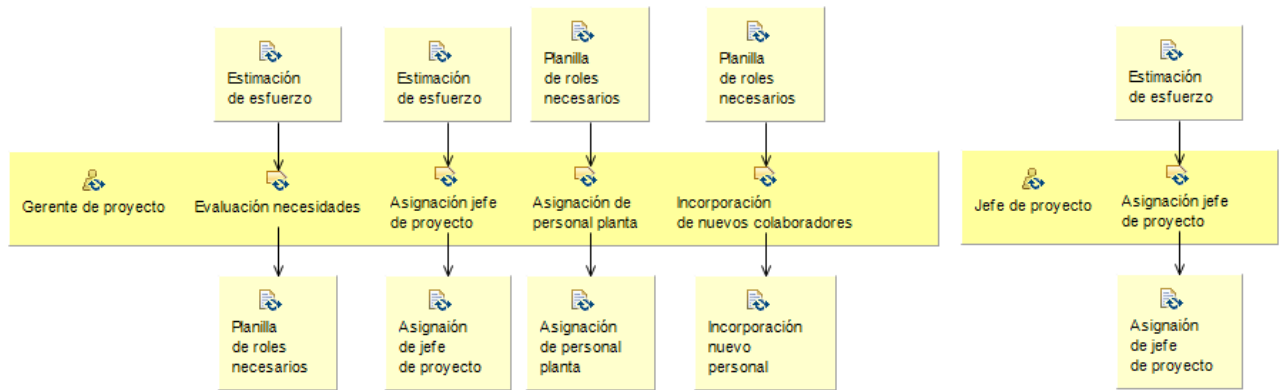


Figura 42: Detalle de la actividad Estructuración del equipo de trabajo

Kick off

Esta actividad, Figura 43: Tareas dentro del Kick off, tiene como propósito iniciar formalmente la ejecución del proyecto. Esto se materializa través de la planificación y ejecución de una reunión en donde se presenta la planificación del proyecto y del equipo de trabajo. La Figura 44: Detalle de la actividad Kick off, muestra los roles, tareas y artefactos involucrados en esta actividad.

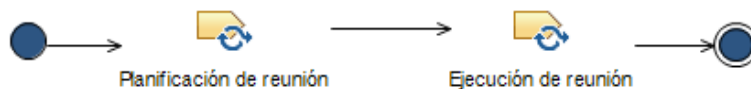


Figura 43: Tareas dentro del Kick off

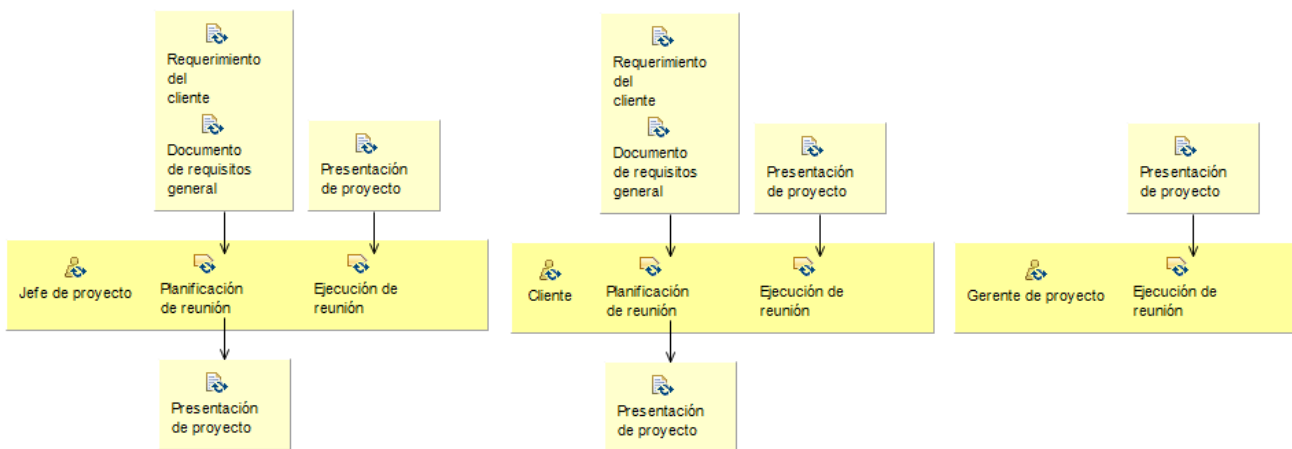


Figura 44: Detalle de la actividad Kick off

Proceso 2: Ejecución del proyecto (EP)

Durante este proceso se realizan las tareas de ejecución del proyecto y tiene por finalidad implementar las funcionalidades explicitadas durante la fase de generación de propuesta.

Fase 3: Planificación detallada

Esta fase se enfoca en priorizar las funcionalidades a implementar según las necesidades del cliente y la disponibilidad de los requisitos técnicos, además de hacer gestión de los cambios requeridos por el cliente. Esta fase puede ser repetitiva y dependerá del proceso de desarrollo acordado. Las actividades dentro de esta fase son:

Planificación de sprint

Durante esta actividad, Figura 45: Tareas dentro de la Planificación del sprint, se define el alcance del desarrollo (total o parcial según el proceso de desarrollo) seleccionando las tareas a abordar. Todas las tareas deben ser asignadas y calendarizadas para luego ser actualizadas en el backlog del proyecto. También es necesario notificar el acuerdo a los involucrados para clarificar el objetivo de esta etapa.

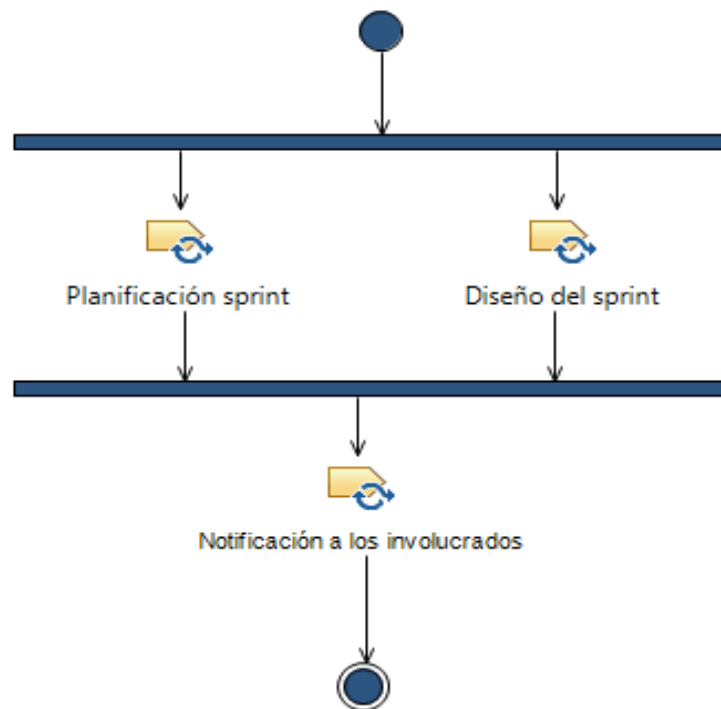


Figura 45: Tareas dentro de la Planificación del sprint

La Figura 46: Detalle de la actividad Planificación de sprint, muestra los roles, tareas y artefactos involucrados en esta actividad.

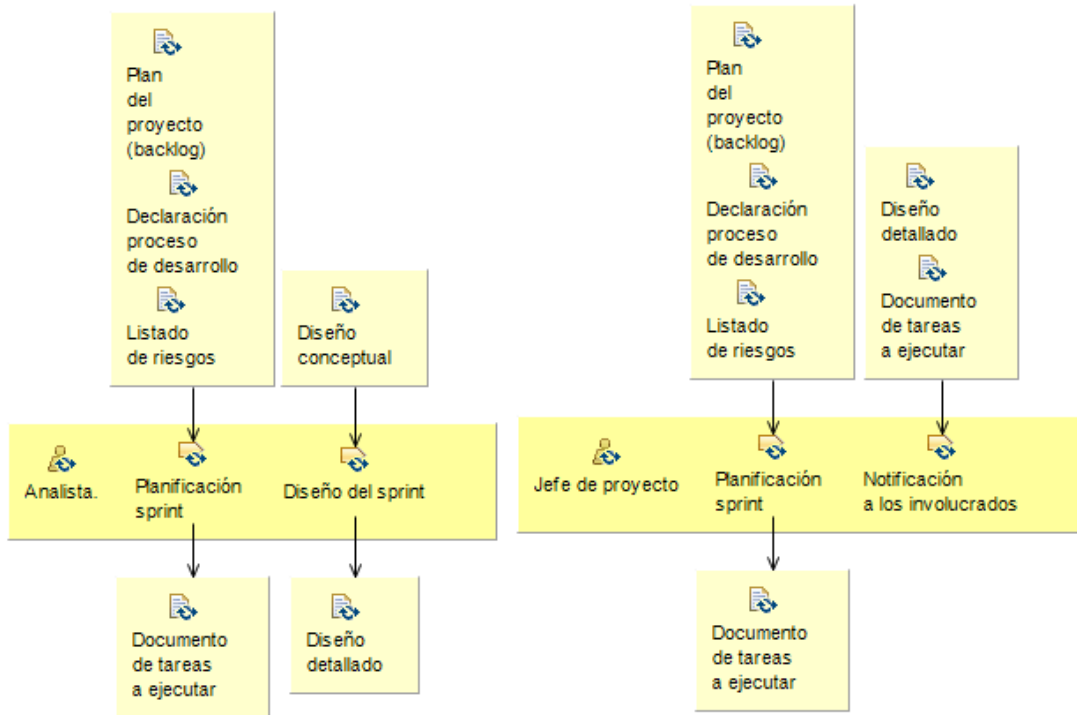


Figura 46: Detalle de la actividad Planificación de sprint

Cambios del sistema

Las tareas que abordan los cambios requeridos por el cliente se trabajan durante las tareas de planificación ya que afectan el desarrollo del proyecto. Antes de proceder con la aplicación de los cambios es necesario analizar estos y para así identificar si corresponden a cambios de priorización de tareas o a cambios en el alcance del proyecto.

La Figura 47: Tareas dentro de Cambios del sistema, muestra las tareas de esta actividad y como estas están organizadas y el detalle de esta se muestra en la Figura 48: Detalle de la actividad Cambios del sistema.

Si los cambios sólo implican el reemplazo de tareas entonces esto se abordará como un control de cambio, Figura 49: Tareas dentro de Controles de cambio, para esto se requiere analizar el backlog buscando las tareas a ser reemplazadas. El diagrama de esta actividad se muestra en la figura y su detalla en la Figura 50: Detalle de la actividad Cambios del sistema.

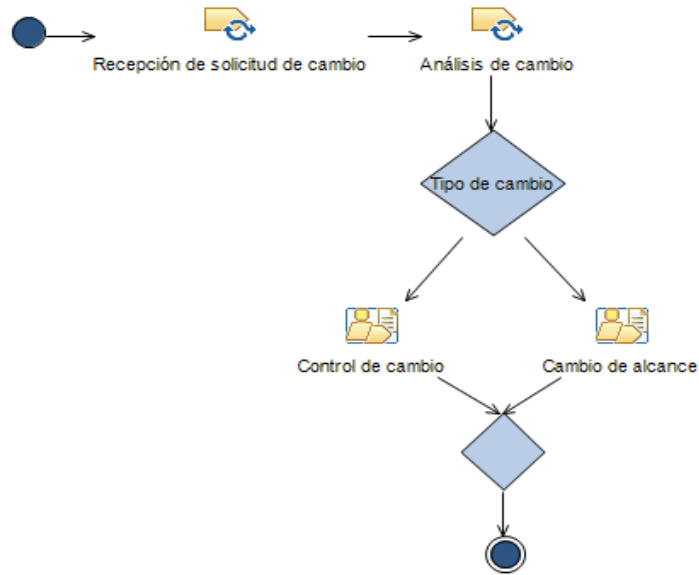


Figura 47: Tareas dentro de Cambios del sistema

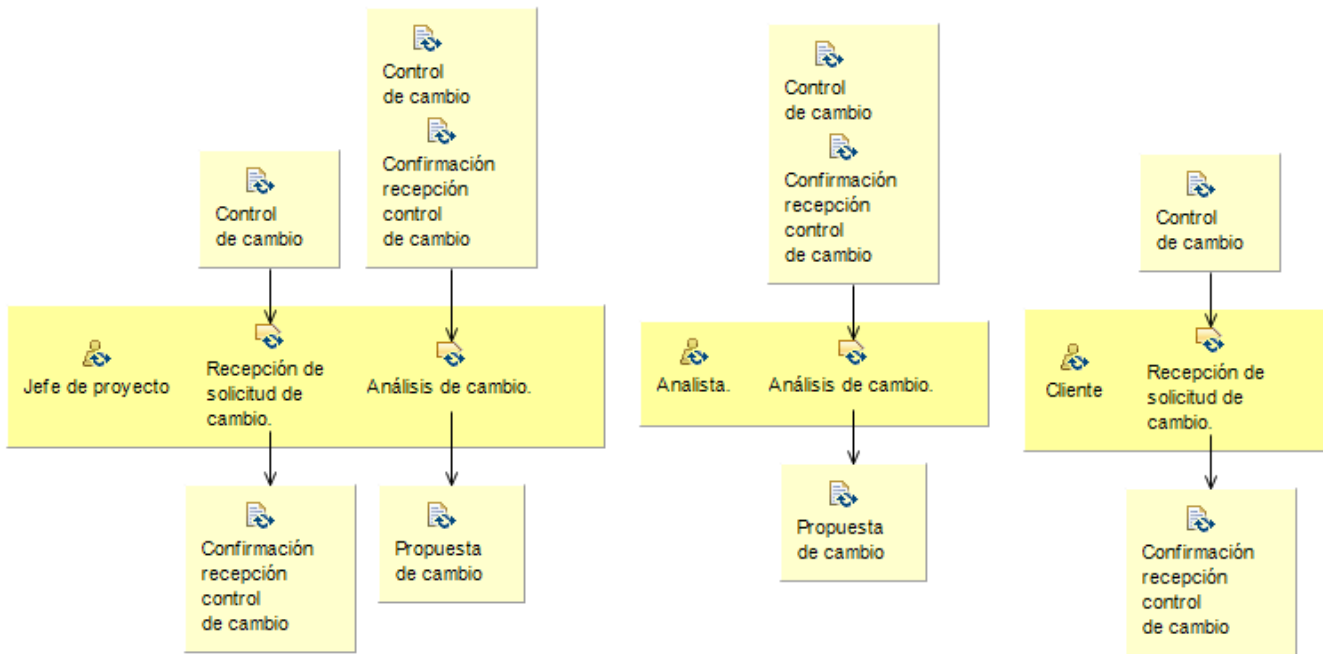


Figura 48: Detalle de la actividad Cambios del sistema

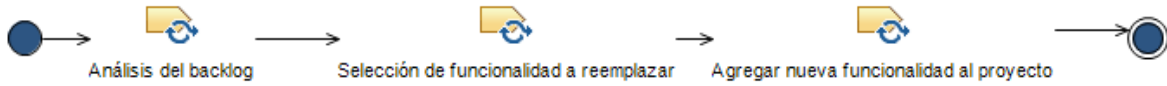


Figura 49: Tareas dentro de Controles de cambio

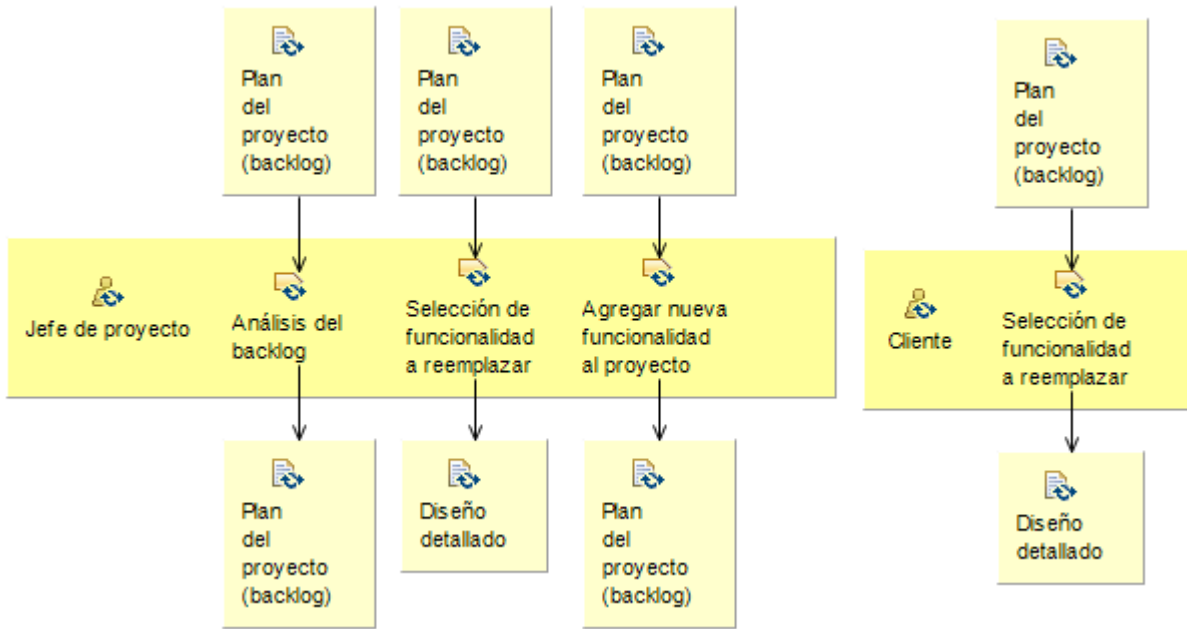


Figura 50: Detalle de la actividad Cambios del sistema

Si, por el contrario, las nuevas necesidades implican agregar nuevas funcionalidades será necesario involucrar a la gerencia de proyectos para que ésta se encargue de las tareas comerciales. Luego de tener la aceptación de los cambios por parte del cliente las tareas deben ser agregadas al proyecto actualizando su planificación, esta actividad se muestra en la Figura 51: Tareas dentro de Cambio de alcance, y los roles, tareas y artefactos se muestran en la Figura 52: Detalle de la actividad Cambio de alcance.

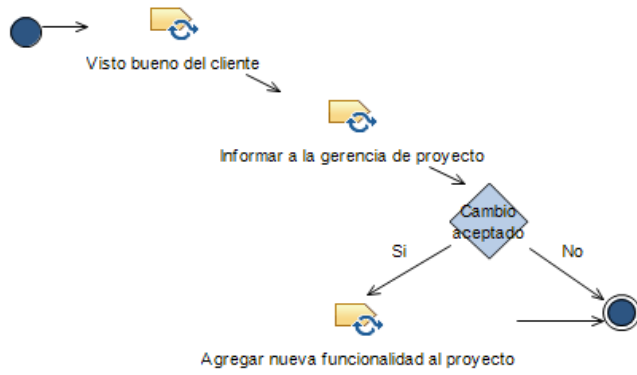


Figura 51: Tareas dentro de Cambio de alcance

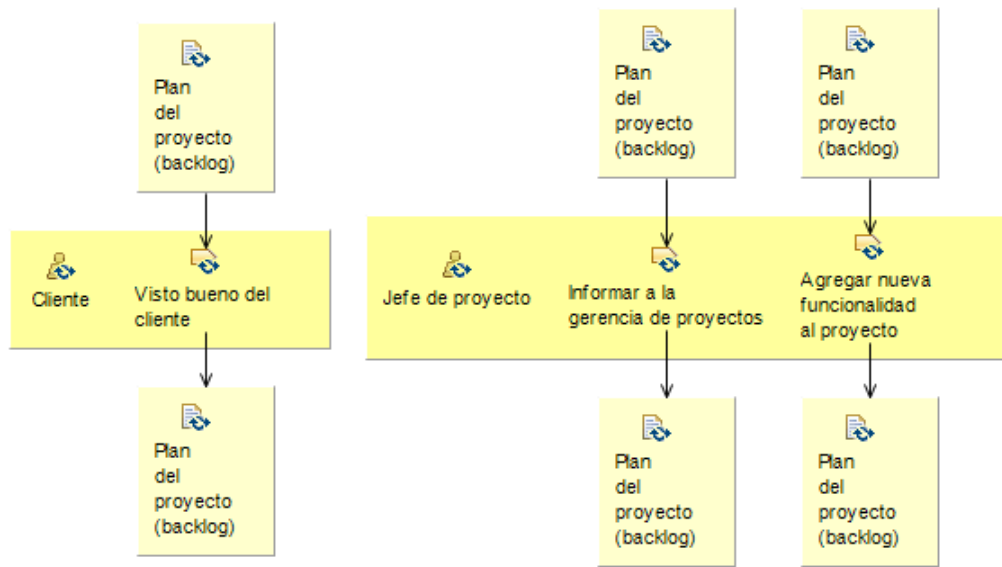


Figura 52: Detalle de la actividad Cambio de alcance

Fase 4: Ejecución

Durante esta fase se ejecutan las tareas seleccionadas y priorizadas en la planificación detallada teniendo por objetivo implementar las funcionalidades descritas en la propuesta entregada, disponiendo luego del sistema en el ambiente productivo del cliente. Las actividades que se abordan durante esta fase son:

Ejecución del sprint

Durante la ejecución es que se desarrolla el sistema, para esto primero es necesario coordinar las tareas entre los integrantes del equipo, ejecutar el desarrollo, pruebas unitarias y documentar las tareas abordadas, Figura 53: Tareas dentro de Ejecución del sprint. Además, esta actividad puede ser ejecutada de forma reiterada y fuera de planificación si es que se detectan errores en actividades posteriores. El resultado de esta actividad sirve como entrada para la ejecución de las pruebas integrales.

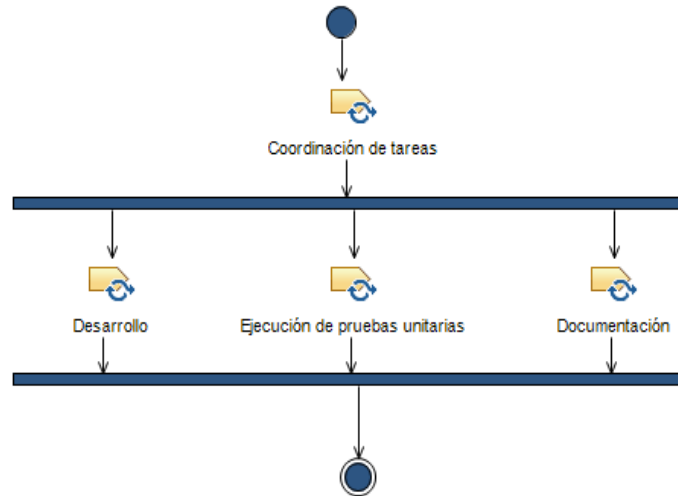


Figura 53: Tareas dentro de Ejecución del sprint

Los roles, tareas y artefactos involucrados en esta actividad se muestran en la Figura 54: Detalle de la actividad Ejecución del sprint.

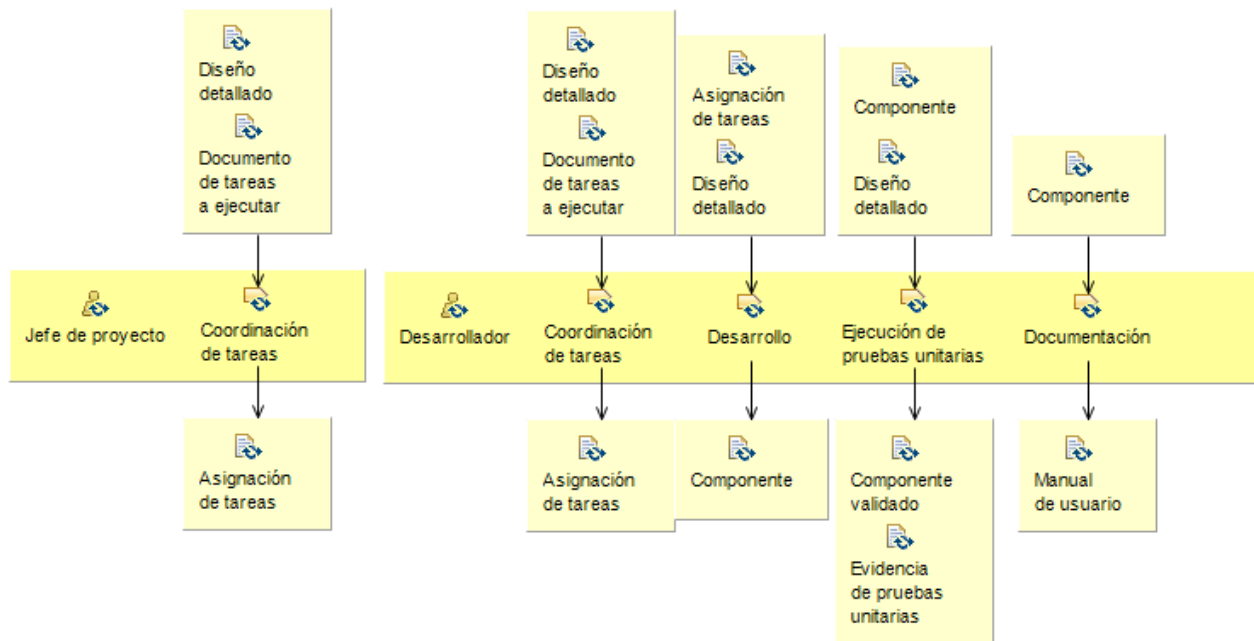


Figura 54: Detalle de la actividad Ejecución del sprint

Pruebas integrales

Durante esta actividad, Figura 55: Tareas dentro de Pruebas integrales, se coordinan las pruebas que validan la funcionalidad del sistema desarrollado. Lo primero es coordinar las pruebas ya que éstas dependen de la ejecución del sprint, con esta entrada se ejecutan las pruebas que permiten evidenciar los resultados. Si luego de ejecutada esta tarea se detectan errores es necesario solicitar la corrección de

estos al jefe de proyectos, si por el contrario las pruebas se completan sin errores se informa de esto a los involucrados y se continúa con el paso a producción.

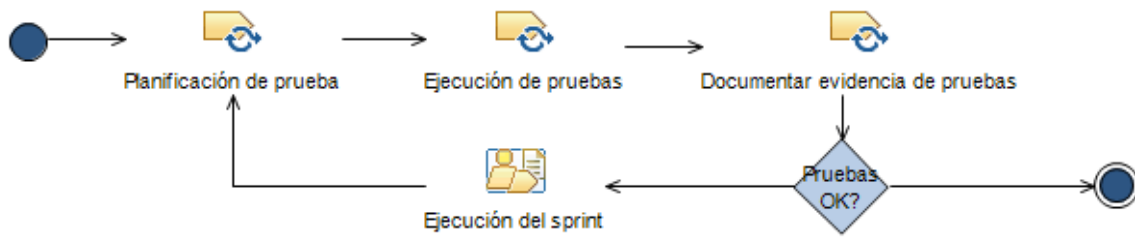


Figura 55: Tareas dentro de Pruebas integrales

La Figura 56: Detalle de la actividad Pruebas integrales, muestra los roles, tareas y artefactos involucrados en la actividad de pruebas integrales.

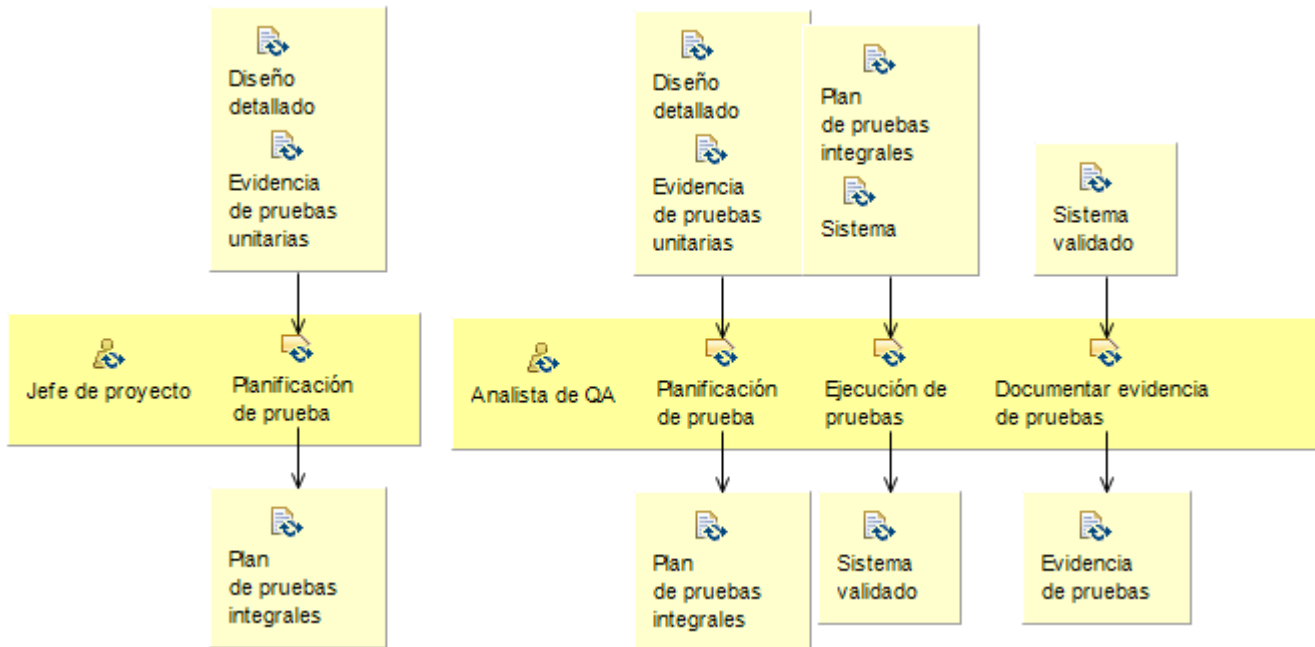


Figura 56: Detalle de la actividad Pruebas integrales

Paso a producción

Luego de la ejecución exitosa de las pruebas y junto al visto bueno del cliente se procede a la instalación del sistema en el ambiente productivo. Una vez planificada la tarea se reúne la documentación necesaria y se instala el sistema. Para validar la correcta aplicación en el ambiente se realizan las pruebas de continuidad, pruebas que permiten validar la continuidad operacional del negocio. En caso se detecten problemas, el sistema es reversado a la última versión funcional registrada y luego se informa de esto al equipo de trabajo. En caso las pruebas de continuidad resulten exitosas se cierra el ciclo de trabajo, Figura 57: Tareas dentro de Paso a producción.

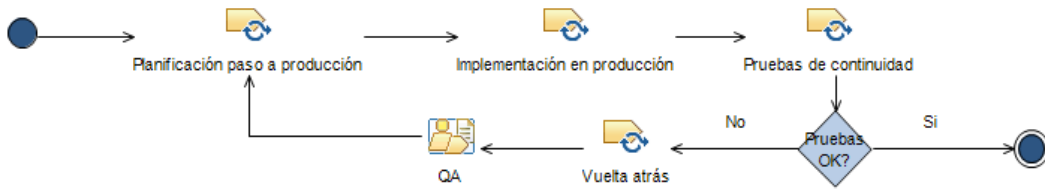


Figura 57: Tareas dentro de Paso a producción

La Figura 59: Tareas dentro de Retroalimentación, muestra los roles, tareas y artefactos involucrados en esta actividad.

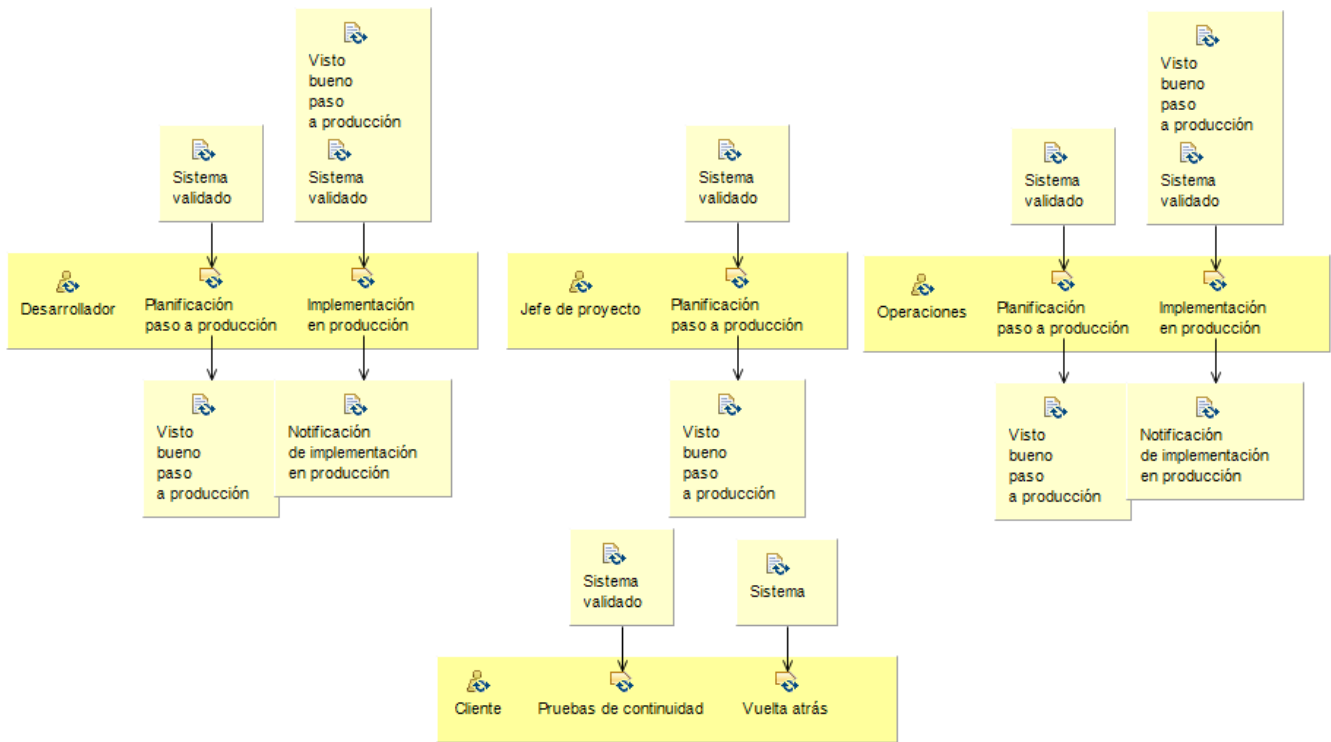


Figura 58: Detalle de la actividad Paso a producción

Retroalimentación

Durante esta actividad se espera que el equipo de trabajo transparente las experiencias, tanto positivas como negativas, de forma que éstas sirvan de apoyo a nuevas iteraciones o proyectos. La actividad debe ser planificada de forma que el equipo de trabajo pueda exponer de la mejor manera posible sus experiencias, de esta forma se espera que la ejecución de la reunión sea más productiva.

La Figura 59: Tareas dentro de Retroalimentación, muestra el proceso de la actividad mientras que la Figura 60: Detalle de la actividad muestra los roles, tareas y artefactos involucrados.

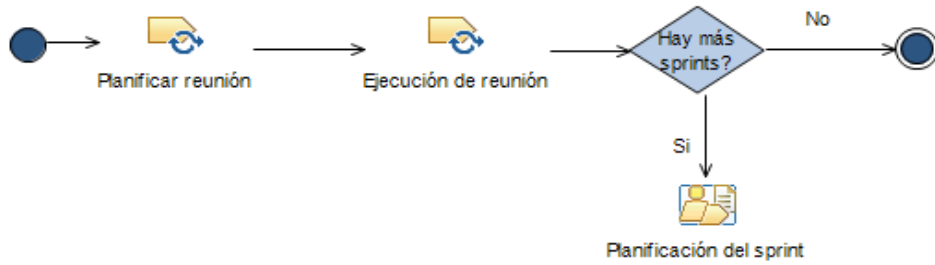


Figura 59: Tareas dentro de Retroalimentación

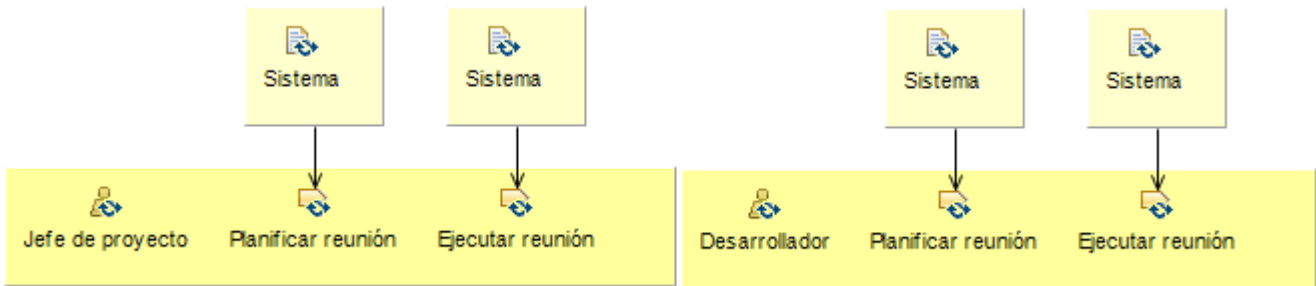


Figura 60: Detalle de la actividad Retroalimentación

Cierre

Luego que el sistema ha sido completamente instalado en producción es que se procede a generar el documento de cierre en el que se deja constancia del acto, Figura 61: Tareas dentro de Cierre. Este documento se genera bajo acuerdo del cliente quien luego aprueba formalmente este documento. La Figura 62: Detalle de la actividad Cierre, muestra los roles, tareas y artefactos involucrados en esta actividad.

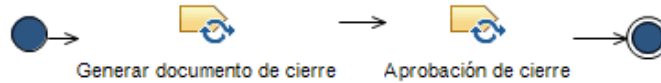


Figura 61: Tareas dentro de Cierre

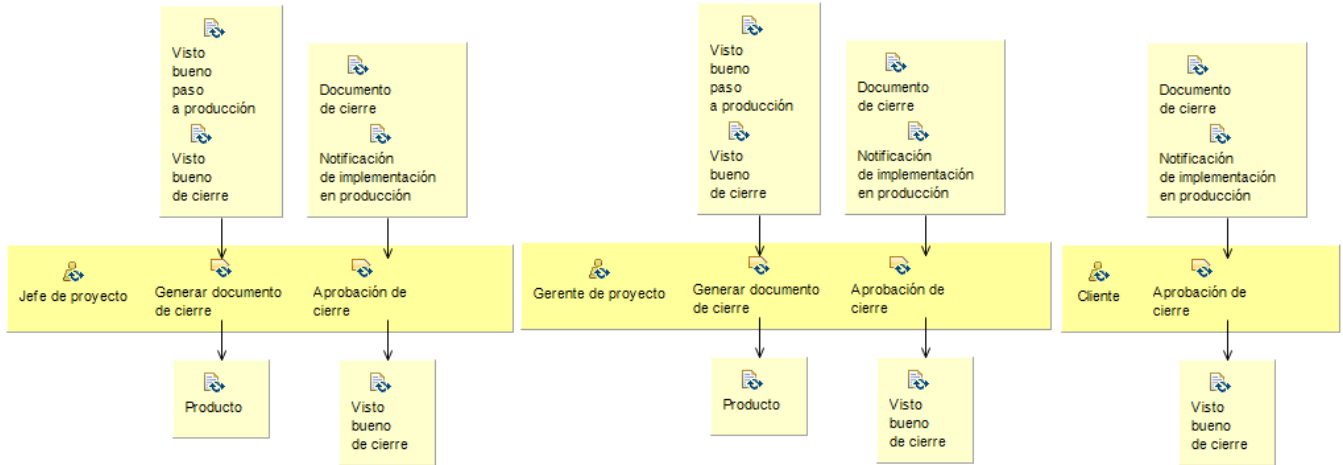


Figura 62: Detalle de la actividad Cierre