



**Marco de trabajo para la gestión de configuración en cadena de versiones de artefactos
software en repositorios locales de la empresa INNOVASYS SA**

Moreno Iza, Gino Paúl

Vicerrectorado de Investigación, Innovación y Transferencia de Tecnología Centro de
Posgrados

Maestría en Ingeniería de Software

Trabajo de titulación, previo a la obtención del título de Magíster en Ingeniería de Software

Ing. Rea Guamán, Ángel Marcelo, PhD.

12 de diciembre del 2022

Latacunga



Reporte de Verificación de Contenido

COPYLEAKS

Capítulos Totales Final v 2.docx

Scanned on: 0:42 December 3, 2022 UTC



Overall Similarity Score



Results Found



Total Words in Text

Identical Words	773
Words with Minor Changes	442
Paraphrased Words	282
Omitted Words	0

Firma:



Firmado electrónicamente por:
ANGEL MARCELO
REA GUAMAN

.....

Ing. Rea Guamán, Ángel Marcelo, PhD.

Director

C.C.: 0201068772



Vicerrectorado de Investigación, Innovación y Transferencia de Tecnología

Centro de Posgrados

Certificación

Certifico que el trabajo de titulación: **“Marco de trabajo para la gestión de configuración en cadena de versiones de artefactos software en repositorios locales de la empresa INNOVASYS SA”** fue realizado por el señor **Moreno Iza, Gino Paúl**; el mismo que cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, además fue revisado y analizado en su totalidad por la herramienta de prevención y/o verificación de similitud de contenidos; razón por la cual me permito acreditar y autorizar para que se lo sustente públicamente.

Latacunga, 12 de diciembre del 2022

Firma:



Firmado electrónicamente por:
**ANGEL MARCELO
REA GUAMAN**

.....
Ing. Rea Guamán, Ángel Marcelo, PhD.

Director



Vicerrectorado de Investigación, Innovación y Transferencia de Tecnología

Centro de Posgrados

Responsabilidad De Autoría

Yo, **Moreno Iza, Gino Paúl**, con cédula de ciudadanía n° 050389177-2, declaro que el contenido, ideas y criterios del trabajo de titulación: **Marco de trabajo para la gestión de configuración en cadena de versiones de artefactos software en repositorios locales de la empresa INNOVASYS SA** es de mi autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Latacunga, 12 de diciembre del 2022

Firma:

Moreno Iza, Gino Paúl

C.C.: 0503891772



Vicerrectorado de Investigación, Innovación y Transferencia de Tecnología

Centro de Posgrados

Autorización de Publicación

Yo, **Moreno Iza, Gino Paúl**, con cédula de ciudadanía n° 050389177-2, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de titulación: **Marco de trabajo para la gestión de configuración en cadena de versiones de artefactos software en repositorios locales de la empresa INNOVASYS SA** en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi responsabilidad.

Latacunga, 12 de diciembre del 2022

Firma:

Moreno Iza, Gino Paúl

C.C.: 0503891772

DEDICATORIA

A mi amado hijo Jair por esa alegría que trasmite a mi vida, fuente de amor e inocencia, aquel ser iluminado, tierno y cariñoso que me llena de vitalidad para seguir consiguiendo metas, sobre él pongo todo mi amor incondicional.

Latacunga, 12 de diciembre del 2022

AGRADECIMIENTO

Al ser parte de esta noble y respetada institución académica en donde crecí como persona y profesional, expreso que el presente trabajo de investigación está dedicado a mis padres por su guía, amor y apoyo para lograr recorrer este sendero de conocimientos; a mi esposa e hijo por ser la fortaleza, inspiración y el motor que alimenta mis ganas de superación; al ingeniero Marcelo Rea por su inmensa vocación de trabajo quien con su experiencia, conocimiento y motivación me orientó en la investigación; y a todo el personal directivo y académico del programa de Maestría en ingeniería de Software de la UFA – ESPE, sede Latacunga, quienes me brindaron la oportunidad de obtener calidad en aprendizaje, únicas experiencias y amistades inolvidables.

Latacunga, 12 de diciembre del 2022

ÍNDICE DE CONTENIDO

Carátula	1
Reporte de Verificación de Contenido	2
Certificación	3
Responsabilidad de Autoría	4
Autorización de Publicación.....	5
Dedicatoria	6
Agradecimiento.....	7
Índice de contenido	8
Índice de figuras.....	17
Índice de tablas	19
Resumen	20
Abastract	21
Capítulo I: Aspectos Generales	22
Introducción	22

Planteamiento del problema	22
Antecedentes	23
Justificación.....	23
Importancia	23
Objetivos	24
<i>Objetivo General</i>	<i>24</i>
<i>Objetivos Específicos</i>	<i>24</i>
Hipótesis de investigación	25
Variable de investigación	25
<i>Variable independiente</i>	<i>25</i>
<i>Variable dependiente.....</i>	<i>25</i>
Metodología de investigación	26
<i>Tipo de investigación</i>	<i>26</i>
<i>Niveles de investigación.....</i>	<i>26</i>
Métodos y técnicas de investigación.....	27

<i>Método de Análisis-Síntesis</i>	27
<i>Método Sistemico</i>	27
Capítulo II: Marco Teórico	28
Antecedentes Históricos	28
Primera Etapa: Cronología (1998-1999).....	28
<i>Estándar 1219d de IEEE (1998)</i>	28
<i>ISO/IEC 15504 (1998)</i>	28
<i>ISO/IEC 14764 (1999)</i>	29
Segunda Etapa: Cronología (2010-2012)	29
<i>CMMI-DEV v1.3 (2010)</i>	29
<i>IEEE 828-2012 (2012)</i>	29
<i>IEEE 828-2012 (2012)</i>	30
Antecedentes conceptuales y referenciales	30
<i>Caracterización del proceso de desarrollo de software</i>	30
<i>Implementación de GCS en empresas pequeñas con IS/IEC 29110</i>	37

<i>Factores de éxito</i>	38
<i>Generación de informes</i>	39
<i>Antecedentes contextuales</i>	39
Metodología de desarrollo del proyecto	42
<i>Fuentes y técnica de recopilación de información y análisis de datos</i>	43
Estado del arte	43
Marco de trabajo para gestión de configuración de software (Revisión Sistemática) 44	
<i>RS de marcos de trabajo de GCS.</i>	44
<i>Identificación de las necesidades.</i>	44
<i>Protocolo de Revisión.</i>	44
<i>Ejecución.</i>	45
<i>Evaluación de la calidad de los estudios.</i>	47
<i>Análisis, Interpretación e Informe de Resultados.</i>	47
<i>Conclusiones de la Revisión Sistemática de Marcos de Trabajo para la gestión de Configuración de Software</i>	51
Modelos Referenciales	51

CMMI (Gestión de Configuración)	52
<i>Propósito</i>	52
<i>CM 1.1. Realizar el control de versiones</i>	55
<i>CM 2.1 Identificar los elementos a ser incluidos en la Gestión de Configuración..</i>	56
<i>CM 2.2 Desarrollar, actualizar y usar un sistema de configuración y de Gestión del cambio</i>	58
<i>CM 2.3 Desarrollar o liberar líneas base para uso interno o para su entrega al cliente</i>	59
<i>CM 2.4 Gestionar los cambios en los elementos de la gestión de la configuración</i>	61
<i>CM 2.5 Establecer los registros de gestión de configuración</i>	65
<i>CM 2.5 Establecer los registros de gestión de configuración</i>	66
ITIL	68
<i>Gestión de la Configuración</i>	68
<i>Planificación</i>	70
<i>Identificación</i>	71
<i>Monitorización</i>	72

Control	72
Verificación y auditorías	72
Gestión de Versiones y despliegue.....	74
Política y Planificación de Nuevas Versiones.....	74
Diseño, Desarrollo y Configuración	75
Plan de Recuperación	75
Prueba y Validación de la Versión.....	75
Planificación de la Implantación.....	76
Comunicación, Formación y Capacitación.....	77
Distribución de Versiones	78
Consideraciones	78
Consideraciones Finales	79
Diseño, Desarrollo y Configuración	80
Plan de Recuperación	80
Prueba y Validación de la Versión.....	81

Consideraciones Finales	81
Capítulo III: Resolución	83
Revisión sistemática de Gestión de Configuración.....	83
Marco de trabajo de Gestión de Configuración de Software.....	85
<i>Elementos del marco para la Gestión de Configuración de Software</i>	<i>85</i>
<i>De los roles y responsabilidades</i>	<i>87</i>
<i>Creación o actualización de la CMDB.....</i>	<i>89</i>
<i>Descripción del sub proceso</i>	<i>91</i>
<i>Procedimientos para creación o mejora de la CMDB</i>	<i>91</i>
<i>Proceso para Registro y/o actualización de los elementos de configuración.....</i>	<i>96</i>
<i>Procedimiento para Monitoreo de la Configuración</i>	<i>103</i>
Automatización del marco de trabajo para la Gestión de Configuración de Software	106
<i>Gestionar los proyectos Software.....</i>	<i>107</i>
Capítulo IV: Experimentación	111
Método de diseño de experimentos de ingeniería de software.....	111

Caso de estudio.....	113
Línea base inicial de los activos de la organización.....	114
<i>Evidencia de ejecución de Pruebas Unitarias.....</i>	<i>114</i>
<i>Conclusiones</i>	<i>117</i>
<i>Resumen de la línea base inicial</i>	<i>118</i>
<i>Actividades del marco de trabajo para la Gestión Configuración de Software.....</i>	<i>118</i>
<i>Identificar los cambios de los elementos de configuración</i>	<i>121</i>
<i>Figura 21 Ingreso del documento de una nueva versión de un elemento de configuración</i>	<i>122</i>
<i>Actividad 2: Actualizar el estado actual del elemento de configuración.....</i>	<i>123</i>
<i>Actividad 3: Registrar elemento de configuración y notificar la acción realizada</i>	<i>123</i>
<i>Actividad 4: Revisar la actualización del CI en CMDB.....</i>	<i>123</i>
<i>Actividad 5: Notificar Novedades encontradas</i>	<i>124</i>
Comparación resultados línea base inicial y marco de trabajo de Gestión de Configuración	124
Conclusiones	125

Validación de las hipótesis de trabajo.....	125
Resumen del capítulo IV.....	126
Bibliografía.....	128
Anexos.....	141

ÍNDICE DE FIGURAS

Figura 1 <i>El flujo de información del proceso de GIEC</i>	39
Figura 2 <i>Frecuencia del nombrado de los Marcos de Trabajo para la Gestión de Configuración de Software</i>	50
Figura 3 <i>Relaciones con otros procesos de la gestión de la configuración</i>	70
Figura 4 <i>Diagrama de Creación Mejora de la CMDB</i>	91
Figura 5 <i>Diagrama de Registro y/o Actualización de los Elementos de Configuración</i>	98
Figura 6 <i>Diagrama de Monitoreo de la Configuración</i>	103
Figura 7 <i>Gestionar los proyectos de Software</i>	107
Figura 8 <i>Agregar nueva versión a un artefacto</i>	108
Figura 9 <i>Agregar nuevo artefacto a un proyecto</i>	108
Figura 10 <i>Carga de la nueva versión del artefacto</i>	109
Figura 11 <i>Mensaje de Confirmación</i>	109
Figura 12 <i>Formulario para un nuevo proyecto</i>	110
Figura 13 <i>Comparación de versiones</i>	110
Figura 14 <i>Características de los estudios de caso. (Fuente: Wohlin, Höst, & Henningson, 2003)</i>	112
Figura 15 <i>Desarrollo de formulario compuesto</i>	115
Figura 16 <i>Configuración del formulario en la actividad</i>	116
Figura 17 <i>Solicitud de Prórroga 288401 creada</i>	116
Figura 18 <i>Solicitud en buzón de tareas del oficial</i>	117
Figura 19 <i>UF Prórrogas I en buzón de tareas del oficial</i>	117
Figura 20 <i>Ingreso de una nueva versión de un elemento de configuración</i>	122
Figura 21 <i>Ingreso del documento de una nueva versión de un elemento de configuración</i>	122

Figura 22 *Verificación del Artefacto anterior y la nueva versión*..... 124

ÍNDICE DE TABLAS

Tabla 1 <i>Criterios de inclusión y exclusión (RS Marco de trabajo para le gestión de configuración de software – RS MGCS)</i>	45
Tabla 2 <i>Distribución de los Estudios por Fuente (RS MGCS)</i>	47
Tabla 3 <i>Estudios clasificados en base a la categorización de Peterson (RS MGCS)</i>	48
Tabla 4 <i>Marcos de Trabajo para la Gestión de Configuración de Software Identificados (RS MGCS)</i>	49
Tabla 5 <i>CMMI 2.0 Prácticas Recomendadas Sobre la Gestión de la Configuración (CM)</i>	54
Tabla 6 <i>CMMI 2.0 Prácticas Sobre la Gestión de la Configuración (CM)</i>	84
Tabla 7 <i>CMMI 2.0 Prácticas Sobre la Gestión de la Configuración (CM)</i>	87
Tabla 8 <i>Descripción, creación o mejora de CMDB</i>	89
Tabla 9 <i>Procedimiento para Creación o Mejora de CMDB</i>	91
Tabla 10 <i>Descripción, creación o mejora de CMDB</i>	96
Tabla 11 <i>Registro y/o Actualización de ítems de configuración</i>	99
Tabla 12 <i>Proceso de Monitoreo de la configuración</i>	101
Tabla 13 <i>Procedimiento de Monitoreo de la configuración</i>	103
Tabla 14 <i>Prueba unitaria</i>	114
Tabla 15 <i>Objetivo de prueba</i>	114
Tabla 16 <i>Actividades del Marco de Trabajo de GC</i>	119

RESUMEN

La GCS es un proceso de ingeniería que define los procedimientos técnicos utilizados para controlar y mejorar la calidad del software, sin embargo, en gran parte de empresas las áreas de desarrollo de software no emplean SCM a la mayoría de artefactos generados en el proceso de desarrollo de aplicaciones como documentos de especificación de requisitos, diseños de interfaces, modelamiento de bases de datos, entre otros. El problema se genera por las existencias de múltiples versiones de artefactos que tienen contenidos similares, ocasionando caos en la administración de los artefactos porque no es posible conocer información sobre su estado en el tiempo. El presente proyecto de titulación elabora un marco de trabajo de GCS de cadena de versiones en artefactos software para identificar contenidos similares almacenados en repositorios locales, el cual servirá para optimizar la cadena de versiones, el control de configuraciones y el estado de cualquier artefacto de software con contenidos similares almacenados en repositorios locales. Un marco de referencia de GCS es una herramienta que contiene un conjunto de características y medidas que consideran criterios de seguimiento y control que permiten evaluar el proceso de desarrollo de software, mediante cálculos, criterios de aprobación y ponderaciones establecidas. Para el desarrollo de la investigación se inicia con la construcción del marco teórico, una revisión sistemática con la finalidad de conocer el estado actual del tema a investigar, así como los elementos que participan y su relación con el tema de la investigación y un estudio de los modelos referenciales para obtener elementos principales que ayuden a la construcción del marco de referencia y estos resultados se reflejan en un aplicativo prototipo. Siendo evaluado las líneas base inicial y final (marco de trabajo), se compara los resultados obtenidos y se concluye el cumplimiento de los objetivos esperados validando las hipótesis del trabajo.

Palabras clave: marco de trabajo, Gestión de Configuración de Software (GCS), revisión sistemática, modelos de referencia.

ABSTRACT

SCM is an engineering process that defines the technical procedures used to control and improve the quality of software, however, in most companies, software development areas do not use SCM at most. artifacts generated in the application development process such as requirements specification documents, interface designs, database modeling, among others. The problem is generated by the existence of multiple versions of artifacts that have similar contents, causing chaos in the administration of the artifacts because it is not possible to know information about their state over time. This degree project develops a version chain GCS framework in software artifacts to identify similar content stored in local repositories, which will serve to optimize the version chain, configuration control, and status of any software artifact. with similar content stored in local repositories. A GCS reference framework is a tool that contains a set of characteristics and measures that consider monitoring and control criteria that allow evaluating the software development process, through calculations, approval criteria, and established weightings. For the development of the research, it begins with the construction of the theoretical framework, a systematic review in order to know the current state of the subject to be investigated, as well as the elements that participate and their relationship with the subject of the investigation and a study of the referential models to obtain main elements that help to build the reference framework and these results are reflected in a prototype application. After evaluating the initial and final baselines (work framework), the results obtained are compared and compliance with the expected objectives is concluded, validating the work hypotheses.

Keywords: framework, Software Configuration Management (GCS), systematic review, reference models.

Capítulo I

Aspectos Generales

Introducción

La calidad del software es asociada con defectos y factores de calidad, se miden de forma diferente según proyecto o métricas. La Gestión de Configuración de Software (GCS) puede definirse como la capacidad de controlar y gestionar los cambios en un proyecto, se utiliza principalmente para controlar la evolución de los sistemas de software. GCS se ha aplicado a otras áreas desde el cambio de siglo y la gestión del cambio se ha convertido en una herramienta orientada al proceso, en lugar de un proceso orientado a la gestión. El papel de humanos en GCS aún no se ha estudiado ampliamente en comparación a otras áreas de la ingeniería de software”. (Fahmy, 2020)

La presente investigación tiene como finalidad explorar horizontes no investigados dentro de la GCS, para elaborar un marco de trabajo que tendrá como finalidad tomar las mejores prácticas para garantizar el cumplimiento de las normas internacionales en el cumplimiento de la calidad de software.

Planteamiento del problema

Las áreas de desarrollo de empresas de software comúnmente presentan problemas en la organización de los artefactos que se generan en el proceso de desarrollo de aplicaciones software, como documentos de especificación de requisitos, diseños de interfaces, modelamiento de bases de datos, código, entre otros. Este problema se genera por las existencias de múltiples versiones de artefactos que tienen contenidos similares, que son resultado de los cambios que se introducen en algunos de estos, porque no existe un proceso definido para la gestión de las versiones que incluye un repositorio organizado que incluya una herramienta de gestión de la configuración. Este incremento de versiones ocasiona caos en la

administración de los artefactos, porque no es posible conocer información sobre su estado en el tiempo. En base a las limitaciones antes descritas se plantea el siguiente problema:

¿Cómo optimizar la cadena de versiones de los artefactos generados en el desarrollo de aplicaciones software almacenados en repositorios locales en las empresas desarrolladoras de software?

Antecedentes

La Gestión de la Configuración del Software (GCS) es un proceso de ingeniería, que indica procedimientos técnicos para el control y mejora de la calidad del software. Los subprocesos o fases que lo conforman son: Identificación de la Configuración, Control de Cambios, Control de Versiones, Informes de Estado y Auditoría de la Configuración. Cada una de las fases tiene parámetros para procesos de control que buscan la integración de las actividades relacionadas con el desarrollo del software desde las primeras fases, asignando roles y responsabilidades al personal de trabajo, para llegar a obtener un producto software de calidad; de esta manera se puede realizar un cambio o mantenimiento eficiente. (Paredes et al., 2011)

Justificación

Comúnmente en la actualidad las empresas de software están compuesta por varios departamentos: técnico, comercial, administrativo y gerencial, en donde el personal de desarrollo de software del departamento técnico genera artefactos software resultado de sus actividades planificadas, en el cual son responsables de las nuevas versiones de archivos creadas a partir de un cambio, esto viene a representar un desorden que se debe ordenar cronológicamente los registros de los documentos.

Importancia

Se hace evidente la necesidad de implementar un marco referencial de gestión de configuración de software que se enfoque a identificar el estado y traza de archivos alojados en

repositorios locales.

Es por ello que, se realiza la presente propuesta cuyos fines inmediatos consisten tanto en la investigación, diseño e implementación de un marco de referencia de gestión de configuración de software para identificar el estado y traza de software alojados en repositorios locales, mismo que será validado mediante un caso práctico.

La presente investigación contribuirá en primer lugar con tener un marco de referencia de gestión de la configuración de software en el estado y traza de software alojados en repositorios locales, hasta el punto de generar un conocimiento base para el desarrollo de propuestas tecnológicas mediante este principio.

En el mercado se conoce acerca de herramientas para llevar el control únicamente de código fuente de programas informáticos, pero no existe un marco de trabajo sencillo para gestionar el estado en el tiempo o las versiones de los artefactos creados durante el desarrollo de aplicaciones software.

Objetivos

Objetivo General

- Construir un marco de trabajo de GCS de cadena de versiones en artefactos software, para identificar contenidos similares almacenados en repositorios locales en la empresa INNOVASYS SA.

Objetivos Específicos

- Construir el marco teórico que fundamente el marco de referencia de GCS, para identificar estado y traza de software alojados en repositorios locales.
- Desarrollar el marco de referencia de GCS para identificar estado y traza de software alojados en repositorios locales.
- Implementar el marco de referencia de GCS para identificar estado en el tiempo de artefactos software alojados en repositorios locales.

- Desarrollo de un prototipo como esquema del sistema a representar.
- Validar el modelo de referencia de GCS para identificar estado y traza de software alojados en repositorios locales.

Hipótesis de investigación

Si se elabora un marco de trabajo se optimiza la cadena de versiones, el control de configuraciones y el estado de cualquier artefacto de software con contenidos similares almacenados en repositorios locales.

Variable de investigación

Variable independiente

Marco de referencia de Gestión de la Configuración de Software para artefactos resultantes en el desarrollo de aplicaciones.

Contextualización de la variable independiente

Un marco de referencia de Gestión de la Configuración de Software es una herramienta que contiene un conjunto de características y medidas, que consideran criterios de seguimiento y control que permiten evaluar el proceso de desarrollo de software mediante cálculos, criterios de aprobación y ponderaciones establecidas.

Variable dependiente

Optimizar la Gestión de Configuración de Software en el desarrollo de aplicaciones para la empresa INNOVASYS SA.

- Operacionalización de la variable.
- Cambios durante el desarrollo.
- Reducción de costo de mantenimiento.
- Seguimiento de los Cambios.
- Aprobación y Cierre de la Petición.

Metodología de investigación

Tipo de investigación

Los métodos teóricos se utilizarán para el cumplimiento de las tareas, se utilizaron los siguientes métodos teóricos de investigación.

- **Histórico Lógico:** Para determinar los antecedentes históricos de los modelos de Gestión de Configuración de Software enfocados al artefacto.
- **Analítico Sintético:** Para desarrollar la situación problemática, marco teórico, además se utiliza para procesar la información obtenida en la investigación de campo.
- **Hipotético–Deductivo:** Constituye un eje transversal de la investigación, estará presente en todo el proceso desde la descripción de la situación problémica, y en base al planteamiento de la hipótesis.
- **Sistémico:** Utilizado para el diseño tanto del modelo de evaluación como para la construcción de la herramienta tecnológica que lo operacionalizará de forma más adecuada.

Por otra parte, los Métodos Empíricos como la encuesta, la entrevista, análisis de documentos, y observación directa, se emplearán para recolectar información y datos que servirán para medir el impacto y validar el modelo a través de los indicadores.

Niveles de investigación

Exploratoria

Como primer nivel, se realiza una revisión y esquematización de varios trabajos de investigación realizados acerca del tema, con el fin de clarificar los conceptos y definir el alcance del modelo que se propone.

Descriptiva

Describe una parte de la realidad en la que se desarrolla la industria del software en donde se ha adoptado la MSA (definición de sigla), como un nuevo estándar, donde se revisan sus factores y características más representativas, para construir una herramienta y predecir su aplicabilidad en un nuevo proyecto software.

Métodos y técnicas de investigación

Método de Análisis-Síntesis

Para realizar juicios críticos fundamentados, y poder evaluar, decidir cuáles factores de una arquitectura de micro-servicios son los más determinantes.

Método Sistémico

Utilizado para el diseño tanto del modelo de decisión, como para la construcción de la herramienta tecnológica que lo operacionalizará de forma adecuada.

Capítulo II

Marco Teórico

Antecedentes Históricos

La Gestión de la Configuración del Software (GCS) es un proceso de ingeniería, que indica procedimientos técnicos para el control y mejora de la calidad del software. Los subprocesos o fases que lo conforman son: Identificación de la Configuración, Control de Cambios, Control de Versiones, Informes de Estado, Auditoría de la Configuración. Cada una de las fases tiene parámetros para procesos de control que buscan la integración de las actividades relacionadas con el desarrollo del software desde las primeras fases, asignando roles y responsabilidades al personal de trabajo, para llegar a obtener un producto software de calidad; de esta manera se puede realizar un cambio o mantenimiento de una manera eficiente. (Paredes et al., 2011)

Primera Etapa: Cronología (1998-1999)

Estándar 1219d de IEEE (1998)

El estándar 1219 de IEEE (Institute of Electrical and Electronics Engineers) define a la gestión de configuración del software como la modificación de un producto software tras ser liberado para corregir defectos, mejorar la ejecución de otros atributos, o adaptar el producto a un nuevo entorno.

ISO/IEC 15504 (1998)

La norma IS/UEC 15504 define un área de proceso completo sobre gestión de la configuración cuyo propósito es: establecer y mantener la integridad de todos los ítems de configuración en un proceso o proyecto y hacerlos disponibles a las partes involucradas. Esta norma define 5 niveles de capacidad: incompleto, realizado, gestionado, establecido, predecible, optimizado. (Alain April, 2018)

ISO/IEC 14764 (1999)

La ISO/IEC 14764 define mantenimiento del software o gestión de configuración del software como el proceso por el cual, un producto sufre modificaciones en su código y documentación asociada para solucionar un problema o mejorar.

Segunda Etapa: Cronología (2010-2012)**CMMI-DEV v1.3 (2010)**

Este modelo CMMI (Capability Maturity Model Integration), resalta la importancia de la gestión de configuración en entornos ágiles, debido a que estos normalmente se enfrentan a cambios frecuentes dentro de su desarrollo, compilación e integración continua. Además, resalta la importancia de definir un rol responsable para llevar a cabo el seguimiento de la implementación de la gestión de configuración. Este modelo para manejo en entornos ágiles de la configuración de software propuso las historias de usuario y la pila de iteraciones (CMMI, 2010).

IEEE 828-2012 (2012)

El modelo 828-2012 es un estándar desarrollado por la IEEE (Institute of Electrical and Electronics Engineers) que establece los requerimientos mínimos para llevar a cabo la gestión de configuración en sistemas e ingeniería del software. Este estándar define: cuales actividades de la gestión de configuración deben ser llevadas a cabo, cuando deben suceder en el ciclo de vida del software y cuales recursos o planeación son requeridos. Este modelo plantea un plan para llevar a cabo la gestión de configuración que son siete procesos de bajo nivel y dos casos especiales en los que se debe aplicar dicho plan.

Los siete procesos de bajo nivel son: planeación, gestión, identificación, control de cambios, reporte de estado, auditoria y gestión de los lanzamientos de la configuración. Los casos especiales son: control de los proveedores y control de interfaces (Engineering Standards Committee of the IEEE Computer Society, 2012).

IEEE 828-2012 (2012)

El modelo 828-2012 es un estándar desarrollado por la IEEE (Institute of Electrical and Electronics Engineers) que establece los requerimientos mínimos para llevar a cabo la gestión de configuración en sistemas e ingeniería del software. Este estándar define: cuales actividades de la gestión de configuración deben ser llevadas a cabo, cuando deben suceder en el ciclo de vida del software y cuales recursos o planeación son requeridos.

Este modelo plantea un plan para llevar a cabo la gestión de configuración que son siete procesos de bajo nivel y dos casos especiales en los que se debe aplicar dicho plan.

Los siete procesos de bajo nivel son: planeación, gestión, identificación, control de cambios, reporte de estado, auditoria y gestión de los lanzamientos de la configuración. Los casos especiales son: control de los proveedores y control de interfaces (Engineering Standards Committee of the IEEE Computer Society, 2012).

Antecedentes conceptuales y referenciales

Caracterización del proceso de desarrollo de software

Desarrollo de software

El software es el equipamiento lógico e intangible de un sistema informático, que comprende el conjunto de los componentes lógicos necesarios que hacen posible la realización de tareas específicas, en contraposición a los componentes físicos que son llamados hardware. En otras palabras, es el conjunto de los programas de cómputo, procedimientos, reglas, documentación y datos asociados, que forman parte de las operaciones de un sistema de computación (Maida & Pacienza, 2015).

Proceso de desarrollo de software

Es la descripción de una secuencia de actividades que deben ser seguidas por un equipo de trabajadores para generar un conjunto coherente de productos, uno de los cuales es el programa del sistema deseado. El objetivo es la formalización de las actividades

relacionadas con el desarrollo del software de un sistema informático. (Drake, 2020)

El proceso de desarrollo de software define el qué, quién, cuándo y cómo del desarrollo de software, teniendo cuatro actividades fundamentales que son comunes para todos los procesos de desarrollo de software.

- Especificación del software: se debe definir la funcionalidad y restricciones operacionales que debe cumplir el software.
- Diseño e Implementación del software: se diseña y construye el software de acuerdo con la especificación.
- Validación del software: el software debe validarse, para asegurar que cumpla con lo
- Evolución del software: el software debe evolucionar, para adaptarse a las necesidades del cliente.

Además de estas actividades fundamentales, Pressman menciona un conjunto de “actividades protectoras”, que se aplican a lo largo de todo el proceso del software (Universidad de la República de Uruguay, 2015), las mismas que se señalan a continuación:

- Seguimiento y control de proyecto de software.
- Revisiones técnicas formales
- Garantía de calidad del software
- Gestión de configuración del software
- Preparación y producción de documentos
- Gestión de reutilización
- Mediciones
- Gestión de riesgos

Un proceso de desarrollo de software, también conocido como un ciclo de desarrollo de software (SDLC), es una estructura impuesta en el desarrollo de software del producto.

Términos similares incluyen el ciclo de vida del software y el proceso de software.

Gestión de configuración de software

La gestión de la configuración de software permite llevar a cabo un control sistemático de los cambios en el sistema durante todo el proceso de desarrollo, y en general, no importa qué tan grande pueda llegar a ser un proyecto, GCS (Gestión de Configuración de Software) tiene un efecto crítico en la calidad. Una buena gestión de la configuración permite llevar un control minucioso de los cambios en todo el sistema con el paso del tiempo, lo que asegura que la integridad del software nunca se verá comprometida, por lo cual, la implementación SMC (siglas) en los proyectos de software trae varios beneficios a corto y largo plazos (Orozco Eduardo, 2017). Entre los más importantes:

- Integridad del sistema a lo largo de todo el proceso de desarrollo.
- Reducción de costos en mantenimiento.
- Mayor control de incertidumbre y la complejidad del producto.
- Trazabilidad completa del sistema.
- Escalabilidad de todo el sistema a través del tiempo.
- Ayuda a minimizar el impacto de los cambios financieros de planeación.

Estudios empíricos sobre la gestión de configuración de software

Recientemente, la comunidad de gestión de configuración de software (GCS) ha reconocido el potencial de la conciencia, y existe un creciente cuerpo de investigación que construye herramientas centradas en la conciencia conceptos para gestionar la coordinación en el desarrollo de software.

Las herramientas en particular están explorando la noción de conciencia del espacio de trabajo (como surgió por primera vez en los sistemas de software colaborativo), para apoyar coordinación entre múltiples desarrolladores que trabajan en paralelo en la misma base de código. La intención es que los desarrolladores estén continuamente informados de los cambios en curso en otros espacios de trabajo, así como, los efectos anticipados de esos

cambios, para que puedan detectar cambios potencialmente conflictivos y responden proactivamente. Ejemplo, las respuestas pueden incluir contactar a la otra parte para discutir, posponer los cambios hasta que otro desarrollador haya verificado en la suya, usando el sistema GCS para ver el de otro desarrollador espacio de trabajo para determinar el alcance de un conflicto y otras acciones afines. Por lo tanto, los cambios conflictivos pueden abordarse antes que se vuelvan demasiado severos. Incluso pueden evitarse por completo, cuando los desarrolladores reconsideran si editar un artefacto que ellos o alguien más está modificando en ese momento. (Sarma et al., 2016)

ISO 12207

La norma ISO 12207, es una metodología que se utiliza para la gestión del ciclo de vida del software. Este ciclo de vida del software se define como el proceso mediante el cual, en función de una serie de entradas que se procesarán, se obtendrán unas salidas. ISO 12207 se trata de un estándar elaborado de una manera flexible y modular, de tal modo que pueda adaptarse al usuario que lo utilice. El estándar 12207 se apoya en dos pilares principales: modularidad y responsabilidad. La modularidad es la capacidad de que el sistema tenga el máximo de afinidad y empaque con el menor esfuerzo posible. La responsabilidad tratará de que en cada etapa del proceso exista un encargado, de tal modo que, en la organización el trabajo final obtenido se vea representado en cada fase por un responsable del desarrollo software, dando de este modo cohesión y unidad al proyecto global.

Objetivos

El objetivo principal sería que ISO 12207 se pudiese instaurar en una organización de una manera natural, de acuerdo al grado de madurez que cada organización disponga, así como disponer de unos cimientos adecuados, y que ellos redunden en mejoras en los procesos de desarrollo de software.

Por tanto, se pudiera decir, que el propósito de ISO 12207, fuese el de establecer un

marco con el cual poder desarrollar un software de una calidad adecuada, aplicando para ello, metodologías de desarrollo de software y metodologías de mantenimiento, lo cual provocaría una evidente mejora en la calidad del producto, siempre bajo un estricto control en los proyectos. Por otro lado, ISO 12207, tiene claro que persigue lo que se acaba de describir, sin embargo, no dice cómo se debe hacer, ni cuáles son las herramientas a utilizar, ni a qué lenguajes o entornos afecta, ni tampoco es un estándar que aplica métricas de valoración de la calidad del software.

Procesos

Los procesos se clasifican en tres tipos:

- Procesos principales.
- Procesos de soporte.
- Procesos de la organización

Los procesos principales no se aplicarán a nivel global, sino con total independencia del proyecto software que se desee desarrollar. Sin embargo, los procesos de soporte y de organización, deben guiarse por una metodología impuesta, genérica, sin tener ninguna vinculación con la aplicación a desarrollar o de la propia empresa que lo realiza. (Raúl Calderón, 2016)

Los procesos principales implican:

- Adquisición.
- Suministro.
- Desarrollo.
- Operación.
- Mantenimiento

Los procesos principales implican:

- Documentación.

- Gestión de la configuración.
- Aseguramiento de calidad.
- Verificación.
- Validación.
- Revisión conjunta.
- Auditoría.
- Resolución de problemas.

Y los procesos de la organización:

- Gestión.
- Infraestructura.
- Mejora.
- Recursos Humano.

Línea base (LB)

Una línea base es una especificación o producto que ha sido revisado formalmente, sobre el que se ha llegado a un acuerdo, y que de ahí en adelante servirá como base para un desarrollo posterior que puede cambiarse solamente a través de procedimientos formales de control de cambios. Actualmente existen muchos proyectos que ponen en práctica la creación de las líneas bases ya que las mismas hacen que estos sean terminados con la calidad y eficiencia requerida debido a su capacidad de evaluar el desempeño y la estimación exacta del futuro mejorado. (Rivero, 2011)

Instrumento para la identificación de Línea Base

Según (Espinosa, 2014) una línea base es un punto de referencia en el proceso de desarrollo del software que queda marcado por la aprobación de uno o varios EEC (Elemento Experimental de Configuración) mediante una revisión técnica formal.

En la construcción del instrumento que permite el registro de las diferentes líneas base

para gestión de configuración de experimentos consideramos:

- El proceso de experimentación, tiene un conjunto de fases establecidas con un orden instituido, dicho orden no implica que una actividad tiene que terminar para iniciar con la siguiente.
- En cada una de las fases se desarrollan o se generan ciertos productos y resultados experimentales.
- Es importante contar con puntos de control o hitos que permitan establecer el momento de transición de una fase a otra.
- Los puntos de control se utilizan en la GEC (Gestión experimental de Configuración) para aprobar formalmente los EEC desarrollados en las diferentes fases del proceso de experimentación.

Control de cambios

El control de cambios es un conjunto de prácticas para la evaluación y aprobación de los cambios realizado a distintos elementos de la configuración software durante el ciclo de vida del software.

Las actividades del control de cambios en GCS (Espinosa, 2014), incluyen:

- Identificación y registro de peticiones de cambio.
- Análisis y evaluación de los cambios.
- Aprobación o rechazo de la petición de cambio.
- Implementación del cambio.
- Verificación y liberación del EC modificado del proyecto software.

Repositorio de software y sus ramas

Hay muchas opciones para las herramientas de repositorio de GCS, a menudo llamadas software de control de versiones o repositorio de software. Han sido creadas para facilitar el trabajo en grupo sobre desarrollo, mantenimiento y producción de software, donde el código y

la documentación necesitan ser compartidos y actualizados.

El repositorio de software es fundamental para el desarrollo, el mantenimiento y el lanzamiento de actividades de gestión durante un proyecto.

Un elemento clave del plan GCS es la propuesta de una estrategia de ramificación que será adecuada para el proyecto. Cada herramienta de repositorio de software tiene algún tipo de soporte de bifurcación que son medios de ramificación que se desvían de la línea principal de desarrollo (también llamada tronco), para realizar desarrollo de trabajo sin afectación.

En muchas herramientas, este es un proceso algo costoso que a menudo requiere que cree una nueva copia de su directorio de código fuente, que puede llevar mucho tiempo para proyectos grandes (Alain April, 2018).

Auditoría de configuración de software

El estándar IEEE 1028 define las auditorías como un examen independiente de un producto de software, un proceso de software o un conjunto de procesos de software realizados por un tercero, para evaluar el cumplimiento de especificaciones, normas, acuerdos contractuales, u otros criterios.

Se podría solicitar una auditoría de GCS para un proyecto de software, para evaluar cómo los CI (Ítems de Configuración) satisfacen los características funcionales y físicas necesarias y también para evaluar cómo el plan GCS fue implementado en el proyecto. Normalmente se realizan dos tipos de auditorías formales: una auditoría de configuración funcional (FCA) y la auditoría de configuración física. (PCA)(Alain April, 2018)

Implementación de GCS en empresas pequeñas con IS/IEC 29110

El estándar ISO 29110 describe un perfil básico para organizaciones pequeñas, un enfoque simple para CM (Gestión de Configuración) es realizar las siguientes tareas:

- Identificar componentes.
- Describir los estándares que se utilizarán durante un proyecto típico.

- Formalizar las revisiones antes de que un elemento de configuración sea depositado en el repositorio.
- Establecer un proceso de control de cambios simple.
- Establecer la biblioteca y controlar el acceso.
- Gestionar solicitudes de cambio.
- Comprobar de vez en cuando que las copias de seguridad en el repositorio se han realizado correctamente.

ISO 29110 recomienda que se establezca un repositorio para almacenar productos de trabajo y sus versiones. (Alain April, 2018)

Factores de éxito

Los siguientes son algunos factores relacionados con las prácticas de CM de una organización que pueden promover o desalentar el desarrollo de software de calidad. (Alain April, 2018)

Factores que fomentan la calidad del software:

- Cultura organizacional y soporte de gestión para SCM.
- Presencia de una visión, misión y políticas de SCM.
- Adecuada asignación de recursos y apoyo SQA.
- Planificación temprana de CM y comunicaciones efectivas.
- Conjunto de herramientas estable y profesionales de CM competentes
- Certificación y capacitación SCM disponibles

Factores que pueden afectar negativamente la calidad del software:

- Sin soporte de gestión para SCM.
- No hay capacitación o certificación de SCM disponible.
- Proceso SCM rígido y complejo.

- Falta de recursos humanos requeridos y presupuestos necesarios para la ejecución de SCM durante el proyecto.
- Comunicaciones descoordinadas con la CCB.

Generación de informes

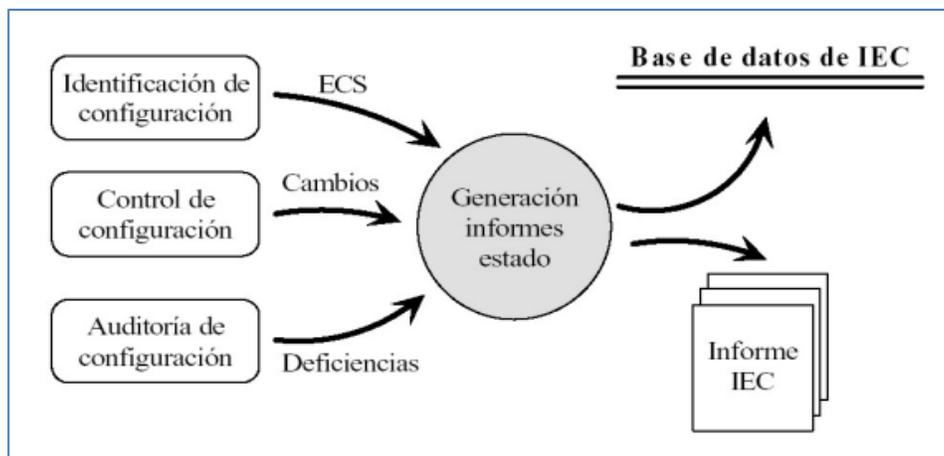
La generación de informes de estado de la configuración (GIEC) responde a las preguntas.

- ¿Qué pasó?
- ¿Quién lo hizo?
- ¿Cuándo paso?
- ¿Qué más se vio afectado?

El flujo de información del proceso de GIEC se puede apreciar en la siguiente figura:

Figura 1

El flujo de información del proceso de GIEC



Nota. En el gráfico se puede observar la representación del flujo del proceso GIEC. Tomado de (Orozco Eduardo, 2017)

Antecedentes contextuales

INNOVASYS SA es una empresa multinacional de software bancario especializada en el negocio financiero desde 2004. INNOVASYS SA une su experiencia con instituciones

financieras para identificar las necesidades tecnológicas y las tendencias de la banca, para de esta manera desarrollar un software financiero que satisfaga esas necesidades. El sistema es utilizado por más de 70 instituciones financieras y tenemos más de 700 especialistas en tecnología. Estamos en Estados Unidos de América, El Salvador, Panamá, Colombia, Ecuador, Perú, España y Argentina.

De acuerdo a su amplia experiencia ofrece:

- Consultoría comprometida a apoyar la transformación de los negocios para satisfacer las nuevas necesidades del mercado y crear ventajas competitivas a través de frameworks ágiles, desarrollo de proyectos, conceptos de producto, prototipado y consultoría sobre ecosistemas digitales
- Los servicios de implementación de INNOVASYS se basan en la integración transparente y oportuna de nuestros productos y/o servicios en su organización. Una implementación de INNOVASYS podría incluir: software, hardware, datos, documentación, capacitación, cambios organizacionales y de proceso necesarios para asegurar el éxito del proyecto
- Las integraciones de soluciones INNOVASYS y de otras soluciones no INNOVASYS, incorporan soluciones integrales de software en los negocios con el objetivo de lograr un mayor crecimiento para los clientes
- Software como Servicio – SaaS, se encuentra nativa en la nube con todas las ventajas que esta tecnología ofrece: seguridad, mayor escalabilidad, menores costos y tiempos de implementación, accesibilidad y compatibilidad
- El equipo de expertos en la fábrica de INNOVASYS ofrece desarrollo de software continuo y orientado al valor en el negocio para mejorar las soluciones, personalizarlas y satisfacer las necesidades más exigentes de los clientes de una manera productiva, consistente y de alta calidad

- INNOVASYS Outsourcing incluye un amplio portafolio de servicios de TI como gestión de arquitectura, gestión de infraestructura, gestión de procesamiento por lotes, desarrollo evolutivo, gestión de mesa de ayuda y soporte, entre otros; orientados a generar valor a los clientes

Para identificar la existencia del problema se elaboraron y aplicaron encuestas orientadas al personal de la empresa INNOVASYS. De las cuales se obtuvieron los siguientes resultados:

¿Con qué frecuencia realiza cambios sobre la redacción de sus archivos (Word, Excel, PowerPoint, Scripts, Código Fuente), en si todos los artefactos que se pueden generar en el proceso de desarrollo de software dentro de la empresa?

- El 60% del personal tienen una frecuencia de 1 a 3 veces a la semana, el 20% del personal tiene una frecuencia de 4 a 6 veces a la semana, mientras que el 20% del personal tiene una frecuencia de registro mayor a 6 veces a la semana.

¿Cuenta con una aplicación informática que le permita visualizar el orden cronológico de los cambios realizados en los archivos?

- El 100% del personal dice que no cuentan con una aplicación.

¿Con que frecuencia se le ha perdido documentos importantes de sus repositorios?

- El 30% del personal tienen una frecuencia “nunca”, el 50% del personal tiene una frecuencia “a veces”, mientras que el 20% del personal tiene una frecuencia de “siempre”.

¿Le causa confusión identificar los cambios realizados sobre documentos por su contenido?

- El 30% del personal dice que no, mientras que el otro 70% dice que sí.

¿Le gustaría tener un aplicativo para centralizar sus documentos y que este le indique didácticamente sus cambios en el tiempo?

- El 10% del personal dice que no, mientras que el otro 90% dice que sí.

¿Considera que el desarrollo de un marco de referencia para la gestión de control de cambios en archivos con contenidos similares puede facilitar su organización, localización y recuperación en una carpeta o repositorio local o remoto?

- El 10% del personal dice que sí, el 80% del personal dice que no, mientras que el 10% del personal dice que tal vez.

¿Piensa usted que en la empresa será bien recibido la implementación y ejecución de un marco de referencia de gestión de cambios en repositorios locales?

- El 100% del personal dice que sí.

Metodología de desarrollo del proyecto

Los métodos teóricos utilizados para el cumplimiento de las tareas en la investigación son los siguientes:

- Método Histórico Lógico: Para determinar los antecedentes históricos de los modelos de calidad externa enfocados al producto.
- Método Analítico Sintético: Para desarrollar la situación problemática, marco teórico, además, se utiliza para procesar la información obtenida en la investigación de campo.
- Método Hipotético–Deductivo: Constituye un eje transversal de la investigación, estará presente en todo el proceso desde la descripción de la situación problemática, y en base al planteamiento de la hipótesis.
- Método Sistémico: Utilizado para el diseño tanto del modelo de evaluación como para la construcción de la herramienta tecnológica que lo operacionalizará de forma más adecuada.

Por otra parte, los Métodos Empíricos como la encuesta, la entrevista, análisis de documentos, y observación directa, se emplearán para recolectar información y datos que

servirán para medir el impacto y validar el modelo a través de los indicadores.

Fuentes y técnica de recopilación de información y análisis de datos

La presente investigación hará uso de una revisión sistemática para determinar que marcos de trabajo de gestión de la configuración son utilizados y nombrados en los artículos científicos, que se muestran en el siguiente capítulo, también se usaran fuentes primarias (información obtenida de la investigación) para la sustentación de los problemas encontrados y su solución, también se usaran fuentes secundarias (bibliográficas) para la determinación de los factores técnicos determinantes que serán el eje central de modelo a proponer.

Las herramientas para analizar la información a utilizar son métodos estadísticos, entre otros.

Estado del arte

El objetivo de este capítulo es presentar el estado del arte del tema de esta tesis de grado y marco de trabajo para la gestión de configuración de software, para ello, lo haremos mediante la aplicación del proceso de Revisión Sistemática (RS), así como de la meta-narrativa (consulta, análisis e interpretación de la información obtenida), con la finalidad de conocer el estado actual del tema a investigar, así como los elementos que participan y su relación con el tema de la investigación de esta tesis de grado.

Este capítulo pretende dar a conocer los conceptos y elementos necesarios para poder entender el tema principal de esta tesis de grado. Es decir, entender los trabajos previos realizados que contribuirán a los temas que tienen relación con el contexto de la investigación.

Una revisión sistemática de la literatura existente es un medio para identificar, evaluar e interpretar toda la investigación disponible relevante para una determinada pregunta de investigación, área temática o fenómeno de interés. El proceso de RS está definido en el Anexo I de este documento.

Se realiza una búsqueda de los principales marcos de trabajo para gestión de

configuración de software (véase sección 3.1). En la sección 3.2 se presentan los conceptos de gestión de configuración de software. Finalmente, en la sección 3.3 se incluyen las consideraciones finales de lo ejecutado.

Marco de trabajo para gestión de configuración de software (Revisión Sistemática)

Para la revisión sistemática se ha considerado el proceso de revisión propuesto por Kitchenham (B. A. Kitchenham & Jorgensen, 2004) (B. Kitchenham & Charters, 2007) descrito en el Anexo I.

RS de marcos de trabajo de GCS.

La industria tecnológica presenta una serie de marcos de trabajo para la gestión de configuración de software, que sirven de punto de referencia de la industria para la gestión de configuración de software (GCS) en las empresas. Por ello, se pretende, a través de una revisión sistemática de la literatura científica, responder a la siguiente pregunta:

¿Cuáles son los principales marcos de trabajo de gestión de configuración de software que existen en el mercado?

A continuación, se presenta el proceso de revisión sistemática para los marcos de trabajo de GCS siguiendo el proceso indicado en el Anexo I.

Identificación de las necesidades.

La revisión sistemática nos permitirá responder a las siguientes inquietudes: (1) identificar los diferentes marcos de trabajo para GCS que existen en el mercado, (2) identificar los marcos de trabajo más nombrados en los estudios revisados.

Protocolo de Revisión.

En este apartado se definen las siguientes actividades: las preguntas a realizar, las fuentes de bases de datos que deben usarse para realizar la búsqueda, la elaboración de las cadenas de búsqueda según los criterios definidos, y que incluye las palabras claves de las cadenas de búsqueda.

1. **Preguntas:** las preguntas a responder son: ¿Cuáles son los marcos de trabajo para GCS que existen en el mercado?, ¿Cuáles son los principales marcos de trabajo para GCS que se utilizan en los estudios de investigación?
2. **Fuentes seleccionadas:** IEEE Computer, Science Direct, SpringerLink, Google Scholar.
3. **Las cadenas de búsqueda** fueron las siguientes:
 - a. “Software Configuration Management”.
 - b. “Framework”.
4. **Palabras claves:** las palabras usadas en la cadena de búsqueda incluyen: “Software Configuration Management”, “Framework”.

Ejecución.

En este apartado se describe la búsqueda realizada de los artículos en las bases de datos seleccionadas con las cadenas de búsquedas predefinidas. A la búsqueda, se les aplican los criterios de inclusión y exclusión establecidos a continuación.

Criterios de selección de estudios, y procedimientos para inclusión y exclusión:

En la Tabla 1 se presentan los criterios de inclusión y exclusión que se han aplicado a los resultados de la búsqueda inicial.

Tabla 1

Criterios de inclusión y exclusión (RS Marco de trabajo para le gestión de configuración de software – RS MGCS)

Inclusión (I)	Exclusión (E)
I1. Estudios empíricos de gestión de configuración de software y marco de trabajo. Artículos que tratan sobre los factores que condicionan la gestión de	E1. Artículos que están basados sólo en una opinión particular que no aborde la gestión de configuración de software.

Inclusión (I)	Exclusión (E)
configuración de software en las organizaciones.	E2. Artículos cortos.
I2. Artículos que exponen sobre la gestión de configuración de software y marcos de trabajo.	E3. Estudios que no son relevantes para las preguntas de investigación o no están relacionados con el estudio en particular.
I3. Artículos que incluyen en su redacción las palabras claves.	E4. Estudios que no son claros o presentan ambigüedad.
I4. Artículos cuyo título, resumen o contenido está relacionado con el tema.	E5. Publicaciones duplicadas.
	E6 Estudios anteriores a 2010 debido a las nuevas tendencias tecnológicas.

Nota. En la tabla se presentan los trabajos criterios a ser revisados

- 1) En la primera fase de la revisión sistemática, usando los motores de búsqueda de las bases de datos identificadas y colocando la cadena de búsqueda elaborada en el protocolo de revisión, se encontraron un total de 4.719 estudios.
- 2) En la segunda fase, se han revisado los estudios tomando en consideración los criterios de inclusión y exclusión anteriores, obteniendo un total de 36 estudios relevantes.
- 3) En la tercera fase, se identificaron 13 estudios llamados primarios. Para este fin, se ha realizado la lectura de los estudios relevantes y se han seleccionado aquellos que respondían a las preguntas formuladas en la parte inicial. En la **¡Error! No se encuentra el origen de la referencia.** se muestran las fuentes seleccionadas, el número total de artículos, y el número de artículos relevantes y primarios seleccionados por fuente.

Tabla 2*Distribución de los Estudios por Fuente (RS MGCS)*

Fuente	Total	Relevantes	Primarios
IEEE Explore	11	6	1
Science Direct	153	6	1
SpringerLink	275	4	0
Google Scholar	4280	20	11
Total	4719	36	13

Nota. En la tabla se cuantifica los artículos investigados

Evaluación de la calidad de los estudios.

Para proceder a evaluar la calidad de los estudios, se realizaron las siguientes preguntas:

- ¿Es el estudio primario importante para la investigación que se está realizando?
- ¿Los estudios primarios proveen la suficiente información como para que los resultados se beneficien de la revisión sistemática?
- ¿Los estudios revisados dan un valor agregado a la investigación que se está realizando?

De las preguntas realizadas, se pudo comprobar que los 13 estudios primarios seleccionados son importantes, proveen suficiente información y dan un valor agregado a la revisión sistemática.

Análisis, Interpretación e Informe de Resultados.

Al término de la revisión sistemática, los resultados se han consolidado y analizado. En el análisis, se usan herramientas para identificar la frecuencia del nombrado de marcos de

trabajo para la gestión de configuración de software. Para el análisis de los estudios se han clasificado en: (1) marcos de trabajo para la gestión de configuración de software, (2) modelos más nombrados en los estudios y (3) variantes de los marcos de trabajo para la gestión de configuración de software:

Los artículos contienen casos de estudio, experimentos, encuestas, y se presentan en la Tabla 3, y se clasifican de acuerdo a los criterios de (Peterson et al., 2008).

Tabla 3

Estudios clasificados en base a la categorización de Peterson (RS MGCS)

Clasificación	Referencia
Propuestas de solución	(Bartusevics & Novickis, 2015a), (Bartusevics & Novickis, 2015b), (Schwägerl et al., 2015)
Búsquedas de evaluación	(Saleem & Burney, 2019), (Fahmy et al., 2018), (Cederbom, 2015), (Jendraszyk, 2021)
Artículos de opinión	(Fahmy et al., 2020), (Durrani et al., 2014), (Chou, 2011)
Artículos de experiencia	(Bartusevics & Novickis, 2015a), (Mohd Fauzi et al., 2010)
Artículos filosóficos	(Silva & Da Cunha, 2012)

Nota. En la tabla se indica la base de categorización de Peterson

Una vez seleccionados y analizados cada uno de los estudios primarios, se identificaron los marcos de trabajo para la gestión de configuración de software más nombrados en los estudios de investigación y se elaboró una matriz de marcos de trabajo por número de veces nombrados en los estudios.

A continuación, se presentan los resultados de la revisión sistemática.

- 1) Marcos de Trabajo para la Gestión de Configuración de Software utilizados en los artículos: con el fin de conocer los marcos de trabajo que se utilizan en los estudios de investigación, se seleccionaron los modelos identificados en los estudios primarios (véase Tabla 4; **Error! No se encuentra el origen de la referencia.**).

Tabla 4

Marcos de Trabajo para la Gestión de Configuración de Software Identificados (RS MGCS)

Modelos	Referencias
CCMMI (Capability Maturity Model Integration)	(Bartusevics & Novickis, 2015a), (Uskarıcı & Demirörs, 2017), (Saleem & Burney, 2019), (Fahmy et al., 2020), (Mohd Fauzi et al., 2010), (Chou, 2011)
ITIL (Information Technology Infrastructure Library)	(Bartusevics & Novickis, 2015a),(Cederbom, 2015)

Nota. En la tabla se indica las investigaciones de marcos de GCS

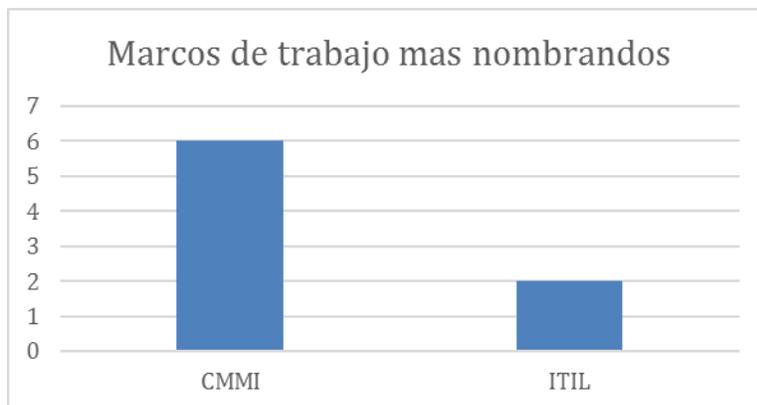
- 2) Frecuencia de Nombrado de los Marcos de Trabajo para la Gestión de Configuración de Software:

Se determinó el número de veces que el marco de trabajo se repite. Al tabular los datos, se obtiene la lista de marcos de trabajo para la gestión de configuración de software que son más nombrados en los estudios, obteniéndose los marcos de trabajo empleados en la gestión de configuración de software, ordenados por la frecuencia con que han sido mencionados por los autores. Se identificaron los principales marcos de trabajo, de acuerdo a la distribución de frecuencias. En la Figura 2. Frecuencia del nombrado de los Marcos de Trabajo para la Gestión de Configuración de Software, se muestra la frecuencia de nombrado de los marcos de trabajo. En los estudios de investigación, se

puede observar que el marco de trabajo más utilizado es el CMMI, es el marco de trabajo más nombrado.

Figura 2

Frecuencia del nombrado de los Marcos de Trabajo para la Gestión de Configuración de Software



Nota. En el gráfico se puede observar la frecuencia con que se nombre los marcos de trabajo de GCS

Del análisis realizado de los marcos de trabajo para la gestión de la configuración, se ha observado lo siguiente:

- “CMMI” Los modelos CMMI (Capability Maturity Model Integration) son colecciones de buenas prácticas que ayudan a las organizaciones a mejorar sus procesos. Estos modelos son desarrollos por equipos de producto con miembros procedentes de la industria, del gobierno y del Software Engineering Institute (SEI).
- “ITIL” (IT Infrastructure Library) es un estándar público que describe las Mejores Prácticas para Gestión de Servicios TI. Proporciona un marco de trabajo para el Gobierno de TI, centrándose en la medición y la mejora continua de la calidad de servicio entregado, tanto desde un punto de vista empresarial como del destinatario del servicio.

Conclusiones de la Revisión Sistemática de Marcos de Trabajo para la gestión de Configuración de Software

De los resultados obtenidos, podemos determinar que:

- El modelo CMMI se enfoca en las buenas prácticas del proceso de gestión de cambios en la gestión de la configuración.
- ITIL, describe las buenas prácticas y mejora continua del proceso de gestión de la configuración y gestión de versiones.

Del análisis realizado, se desprende que el modelo CMMI e ITIL son una buena referencia para la descripción de un marco de trabajo para la gestión de la configuración, en donde se enfatiza el manejo del proceso de gestión de configuración y puntualizando la importancia de la gestión del versionamiento de los artefactos del proceso de desarrollo de software como, por ejemplo:

- Requerimiento funcional,
- Casos de Uso (Negocio, Sistema),
- Diagramas de flujo de Datos,
- Modelos Entidad Relación,
- Diccionario de Datos,
- Documento de Arquitectura,
- Código fuente y demás documentos que se generen en este proceso.

Modelos Referenciales

Esta tesis se enfoca en el diseño de un marco de trabajo para la gestión de la configuración, pues la gestión de la configuración es una disciplina especializada, brinda soporte a los procesos de ingeniería de sistemas y de software. Su propósito es identificar y documentar cada elemento de la configuración (CI), mantener un control de los cambios en

tales características, registrar y reportar cada cambio y estado de la configuración, hasta el soporte de auditoría. (IEEE 828, 2012)

De la revisión sistemática realizada, se pudo encontrar que, los modelos de gestión más nombrados son los que a continuación se describen.

CMMI (Gestión de Configuración)

Propósito

Según el CMMI Institute (2018), la gestión de la configuración es un proceso de soporte de la ingeniería del software. Sus principales objetivos son: gestionar la integridad de los productos de trabajo, controlar versiones, cambios y realización de auditorías. La implementación de dicho proceso de soporte, permite a las organizaciones: ofrecer la versión correcta de la solución al cliente y reducir la pérdida de trabajo.

El propósito de la Gestión de Configuración (CM) es establecer y mantener la integridad de los productos de trabajo utilizando la identificación de la configuración, el control de la configuración, el informe del estado de la configuración y las auditorías de la configuración.

El área de proceso de Gestión de Configuración implica las siguientes actividades:

- Identificar la configuración de los productos de trabajo seleccionados que componen las líneas base en puntos determinados en el tiempo.
- Controlar los cambios a los elementos de configuración.
- Construir o proporcionar las especificaciones para construir los productos de trabajo a partir del sistema de gestión de configuración.
- Mantener la integridad de las líneas base.
- Proporcionar a los desarrolladores, usuarios finales y clientes datos precisos del estado y de la configuración actual.

Los productos de trabajo puestos bajo gestión de configuración incluyen los productos que se entregan al cliente, los productos de trabajo internos seleccionados, los productos

adquiridos, las herramientas y otros elementos utilizados para crear y describir estos productos de trabajo.

Algunos ejemplos de productos que pueden ponerse bajo gestión de configuración son:

- Hardware y equipamiento.
- Diagramas.
- Especificaciones de producto.
- Configuraciones de herramientas.
- Compiladores.
- Herramientas de pruebas y scripts de pruebas.
- Registros de instalación.
- Ficheros de datos de producto.
- Publicaciones técnicas de producto.
- Planes.
- Historias de usuario.
- Backlogs de iteración.
- Descripciones de proceso.
- Requisitos.
- Documentación de la arquitectura y datos de diseño.
- Planes de líneas de producto, procesos y activo base.

Las líneas base proporcionan una base estable para la evolución continua de los elementos de configuración. Las líneas base se añaden al sistema de gestión de configuración a medida que se desarrollan. Los cambios a las líneas base y a las versiones de los productos de trabajo construidos a partir del sistema de gestión de configuración, se controlan y monitorizan sistemáticamente mediante: el control de la configuración, la gestión de cambios y las funciones de auditoría de configuración de gestión de configuración.

La gestión de configuración se centra en el control riguroso de los aspectos de gestión y técnicos de los productos de trabajo, incluyendo el producto o servicio entregado.

En entornos de desarrollo de software ágiles, la gestión de la configuración (CM) es importante debido a la necesidad de soportar cambios y ensamblados frecuentes (normalmente a diario), múltiples líneas base, y múltiples espacios de trabajo soportados por CM (p.ej., para individuos, equipos, e incluso para la programación por pares). Al inicio de cada iteración, es necesario reconfirmar las necesidades de soporte de CM. CM esta cuidadosamente integrada en los ritmos de cada equipo centrándose en minimizar la distracción del equipo para realizar el trabajo.

El CMMI 2.0 por medio de la integración de buenas prácticas, recomienda la implementación del proceso de soporte fundamental a la gestión de la configuración, mejorar y mantener la capacidad de este proceso requiere las siguientes prácticas recomendadas, resumidas a continuación:

Tabla 5

CMMI 2.0 Prácticas Recomendadas Sobre la Gestión de la Configuración (CM)

Práctica Recomendada por el CMMI 2.0	Nivel de Capacidad
CM 1.1 Realizar el control de versiones.	1
CM 2.1 Identificar los artículos a ser incluidos en la Gestión de la configuración.	2
CM 2.2 Desarrollar, actualizar y usar un sistema de configuración y de Gestión del cambio.	2
CM 2.3 Desarrollar o liberar líneas base para uso interno o para su entrega al cliente	2
CM 2.5 Desarrollar, mantener actualizados y utilizar registros que	2

Práctica Recomendada por el CMMI 2.0	Nivel de Capacidad
describen los elementos de la gestión de la configuración.	
CM 2.6 Realizar auditorías de configuración para mantener la integridad de las líneas base de configuración, cambios y contenido del sistema de gestión de la configuración.	2

Nota. En la tabla se indica las prácticas recomendadas para CM

CM 1.1. Realizar el control de versiones

Los cambios a la configuración en los productos de software son comunes en la industria del software. Principalmente, porque, las empresas con el paso del tiempo comprenden de mejor manera su modelo de negocio. Por lo que, cambian sus requerimientos para adaptarse al contexto en que se desempeñan. Este hecho afecta de manera directa a la configuración.

Los miembros de un equipo de desarrollo de software realizan diferentes cambios al software respectivamente en sus espacios de trabajo, los conflictos aparecen cuando se intenta fusionar las diferentes instancias en una misma versión de software, debido a la falta de coordinación en las tareas a realizar durante el desarrollo de una solución. (Kaur y Singh, 2017)

Las organizaciones para mantener su competitividad, debido a la alta demanda de software, mantener la calidad del software y que aporte valor al cliente al mismo tiempo, opta por, automatizar sus procesos de desarrollo de software, utilizando sistema de gestión de versiones como: GitLab, GitHub, Bitbucket.

La implementación de estos sistemas trae consigo las siguientes ventajas:

- Automatización de los procesos de desarrollo.
- Automatización de los procesos de operaciones.
- Control de cambios.

- Comunicación entre los miembros del equipo de desarrollo.
- Trazabilidad de los elementos de configuración.
- Incorporación de datos estadísticos de los proyectos.
- Distribución de la gestión de la configuración.

La innovación en las tecnologías para el desarrollo de software es crucial en una economía creciente, que demandan una mayor tasa de entregas de software. La continua integración afecta directamente en la calidad del software y la productividad de las organizaciones. (Kaur y Singh, 2017)

Las organizaciones son capaces de alcanzar el nivel 1 de capacidad de la gestión de la configuración, simplemente por implementar un sistema de versiones basado en Git. Expandir su capacidad mediante el uso de Herramientas adicionales como JIRA y Travis. También se debe aclarar que la implementación de un sistema de versiones implica tiempo de capacitación al personal, coordinar actividades. En medio de un ambiente ágil de desarrollo de software.

CM 2.1 Identificar los elementos a ser incluidos en la Gestión de Configuración

Antes que determinemos los elementos de la configuración (CI) que serán incluirse a la gestión de la configuración. Es importante, el contexto y la cultura organizacional en la cual se desarrollan las organizaciones.

El contexto organizacional de una organización comprende, como intenta lograr sus objetivos respecto a la gestión de la configuración (Abran et al., 2004). Para que, las versiones correctas sean entregadas a los clientes (Engineering Standards Committee of the IEEE Computer Society, 2012). Hoy en día muchas empresas están utilizando contextos ágiles para el desarrollo de software versus los tradicionales.

La identificación de la configuración consiste en la selección y especificación de:

- Productos entregados al cliente.
- Productos de trabajo internos seleccionados.

- Productos adquiridos.
- Herramientas y otros activos esenciales del entorno de trabajo del proyecto.
- Otros elementos usados en la creación y la descripción de estos productos de trabajo.

Los elementos de configuración pueden incluir el hardware, el equipamiento y los activos tangibles, así como el software y la documentación. La documentación puede incluir especificaciones de requisitos y documentos de interfaz. También se pueden incluir otros documentos que sirven para identificar la configuración del producto o servicio, tales como los resultados de las pruebas.

Un “elemento de configuración” es una entidad seleccionada para la gestión de configuración, que puede consistir en varios productos de trabajo relacionados que forman una línea base. Esta agrupación lógica proporciona facilidad de identificación y acceso controlado. La selección de los productos de trabajo para la gestión de configuración debería basarse en criterios establecidos durante la planificación, para ello se debe considerar las siguientes actividades:

- Seleccionar los elementos de configuración y los productos de trabajo que los componen, basándose en criterios documentados.
- Asignar identificadores únicos a los elementos de configuración.
- Especificar las características importantes de cada elemento de configuración.
- Especificar, cuándo se pone bajo gestión de configuración cada elemento de configuración.
- Identificar al propietario responsable de cada elemento de configuración.
- Especificar las relaciones entre los elementos de configuración. La incorporación de los tipos de relaciones (p. ej., padre-hijo, dependencia), que existen entre los

elementos de configuración en la estructura de gestión de configuración (p. ej., base de datos de gestión de configuración), ayuda en la gestión de los efectos e impactos de los cambios

CM 2.2 Desarrollar, actualizar y usar un sistema de configuración y de Gestión del cambio

Un sistema de gestión de configuración incluye los medios de almacenamiento, procedimientos y herramientas para acceder al sistema. Un sistema de gestión de configuración puede constar de múltiples subsistemas con diferentes implementaciones que son apropiadas para cada entorno de gestión de configuración.

Un sistema de gestión de cambios incluye los medios de almacenamiento, los procedimientos y las herramientas para registrar y acceder a las peticiones de cambio, que por lo general incluye:

- Sistema de gestión de configuración con productos de trabajo controlados.
- Procedimientos de control de acceso al sistema de gestión de configuración.
- Base de datos de peticiones de cambio.
- Sistema de gestión de configuración con productos de trabajo controlados.

Para ello se debe realizar las siguientes actividades:

- Establecer un mecanismo para gestionar múltiples niveles de control. El nivel de control se selecciona normalmente en base a los objetivos, los riesgos y los recursos del proyecto. Los niveles de control pueden variar en relación al ciclo de vida del proyecto, al tipo de sistema bajo desarrollo y a los requisitos específicos del proyecto.
- Proporcionar control de acceso para asegurar el acceso autorizado al sistema de gestión de configuración.

- Almacenar y recuperar los elementos de configuración en un sistema de gestión de configuración.
- Compartir y transferir los elementos de configuración entre los niveles de control en el sistema de gestión de configuración.
- Almacenar y recuperar versiones archivadas de elementos de configuración.
- Almacenar, actualizar y recuperar los registros de gestión de configuración.
- Crear informes a partir del sistema de gestión de configuración.
- Preservar los contenidos del sistema de gestión de configuración.
- Modificar la estructura de gestión de configuración según sea necesario.

CM 2.3 Desarrollar o liberar líneas base para uso interno o para su entrega al cliente

En el desarrollo software, la comprensión de los conceptos de alcance, costo y tiempo es muy importante para la gestión de un proyecto. El Project Management Institute (2017) señala que las líneas base mantienen una relación directa a la planificación u hojas de ruta del proyecto.

Por lo que, la falta de esta práctica compromete el desarrollo de un proyecto de software exitoso.

En consecuencia, tres factores evidencian el desarrollo de líneas base son sus productos de trabajo como son:

- Su uso interno o externo.
- Realización de hojas de ruta.
- La estructura de desglose de trabajo (EDT).

Crear o liberar las líneas base para uso interno y para la entrega al cliente.

Una línea base se representa mediante la asignación de un identificador a un elemento de configuración o una colección de elementos de configuración y entidades asociadas en un momento determinado en el tiempo. A medida que evoluciona un producto o servicio, se pueden utilizar múltiples líneas base para controlar el desarrollo y las pruebas (véase la definición de “línea base” en el glosario).

Los productos hardware, así como el software y la documentación, se deberían incluir también en las líneas base para las configuraciones relativas a la infraestructura (p. ej., software, hardware) y en la preparación para las pruebas del sistema que incluyen interfaces del hardware y software.

Un conjunto común de líneas base incluye los requisitos de nivel de sistema, los requisitos de diseño a nivel de elementos del sistema y la definición del producto al final del desarrollo/inicio de la puesta en producción. Estas líneas base se conocen normalmente como “línea base funcional,” “línea base asignada” y “línea base del producto”, respectivamente.

Una línea base de software puede ser un conjunto de requisitos, de diseño, de archivos de código fuente y su código ejecutable asociado, de archivos de construcción y de la documentación de usuario (entidades asociadas) a los que se ha asignado un identificador único, como, por ejemplo:

- Líneas base.
- Descripción de las líneas base.

Para ello se debe realizar las siguientes actividades:

- Obtener la autorización del Comité de Control de Configuración (CCB) antes de crear o liberar las líneas base de elementos de configuración.
- Crear o liberar líneas base sólo de los elementos de configuración en el sistema de gestión de configuración.

- Documentar el conjunto de elementos de configuración que están contenidos en una línea base.
- Poner a disposición el conjunto actual de líneas base.

Para dar seguimiento y controlar los cambios, se siguen y se controlan los productos de trabajo bajo gestión de configuración.

Las prácticas específicas de esta meta sirven para mantener las líneas base después de que estén establecidas por las prácticas específicas de la meta específica Establecer las líneas base.

CM 2.4 Gestionar los cambios en los elementos de la gestión de la configuración

Los cambios en el software son frecuentes. Normalmente impactan en el desarrollo y operaciones de la organización. Esto implica que, una coordinación de un proceso de cambio es necesaria para gestionar los cambios a los elementos de la configuración y solo los cambios establecidos sean cumplidos. (Hüttermann, 2012)

Esta práctica es muy importante en cualquier organización, por esta razón, muchas de ellas, establecen algún tipo de proceso o mecanismo para atender los cambios los requerimientos del software. Estos mecanismos incluyen: Uso de Correos electrónicos hasta sistemas de tiquetes.

Según Akbar, Sang, Khan, Mahmood, Qadri y Xiang (2019) asegura que, en las organizaciones dedicadas al negocio del desarrollo de software, el cambio dinámico de los requerimientos es esencial. Por otra parte, las empresas manifiestan dificultad en atender los cambios. Tienden a introducir los cambios al finalizar una iteración del desarrollo, en algunos casos delimitando del número de cambios que el interesado o dueño del producto puede realizar.

En cuanto a esta práctica se encontró que las pequeñas organizaciones:

- No todas tienen un proceso establecido para atender los cambios.
- Si registran peticiones de cambios solicitadas por los clientes, así como su priorización, pero no existen documentación con formato establecido.
- Realizan inspecciones para medir el estado de las solicitudes de cambio.
- Casi todas realizan pruebas unitarias o revisiones que garanticen consecuencias no deseadas.

Para poder solventar estas falencias se debe tomar en cuenta lo siguiente:

Seguir las peticiones de cambio a los elementos de configuración. Las peticiones de cambio no sólo tratan los requisitos nuevos o modificados, sino también los fallos y los defectos en los productos de trabajo.

Las peticiones de cambio se analizan para determinar el impacto que tendrá el cambio en el producto de trabajo, en los productos de trabajo relacionados, en el presupuesto y en el calendario.

Para ello se debe gestionar las peticiones de cambio, cabe mencionar las sub prácticas a ser consideradas:

- Iniciar y registrar las peticiones de cambio en la base de datos de peticiones de cambio.
- Analizar el impacto de los cambios y de las correcciones propuestas en las peticiones de cambio. Los cambios se evalúan mediante actividades que aseguren que son consistentes con todos los requisitos técnicos y del proyecto. Los cambios se evalúan por su impacto más allá de los requisitos inmediatos del proyecto o del contrato. Los cambios a un elemento utilizado en múltiples productos pueden resolver una cuestión inmediata a la vez que causar un problema en otras aplicaciones. Los cambios se evalúan por su impacto en los planes liberados.

- Clasificar y priorizar las peticiones de cambio. Las peticiones urgentes se identifican y remiten a una autoridad competente para este tipo de peticiones, si es necesario. Los cambios se asignan a las líneas base futuras.
- Revisar las peticiones de cambio a tratar en la siguiente línea base con las partes interesadas relevantes y llegar a un acuerdo.
- Realizar la revisión de la petición de cambio con los participantes apropiados.
- Registrar el orden de cada petición de cambio y la razón fundamental de la decisión, incluyendo criterios de éxito, un breve plan de acción si procede y las necesidades satisfechas o no satisfechas por el cambio. Realizar las acciones requeridas e informar de los resultados a las partes interesadas relevantes.
- Seguir el estado de las peticiones de cambio hasta su cierre. Las peticiones de cambio llevadas al sistema se deberían manejar de manera eficiente y oportuna. Una vez que se ha procesado la petición de cambio, es crítico cerrarla con la acción apropiada aprobada tan pronto como sea factible. Las acciones que se han dejado abiertas dan como resultado listas de estado más grandes de lo necesario, que a su vez dan como resultado costes añadidos y confusión.

Controlar los cambios a los elementos de configuración. Se mantiene el control sobre la configuración de la línea base del producto de trabajo. Este control incluye el seguimiento de la configuración de cada elemento de configuración, aprobando una nueva configuración, en caso de ser necesario, y actualizando la línea base, considerando el historial de revisiones de los elementos de configuración, y los archivos de líneas base.

Las sub prácticas a ser consideradas son las siguientes:

- Controlar los cambios a los elementos de configuración a lo largo de la vida del producto o servicio.

- Obtener la autorización apropiada antes que los elementos de configuración modificados sean introducidos en el sistema de gestión de configuración.
- Realizar actividades de check-in y check-out de los elementos de configuración en el sistema de gestión de configuración para la incorporación de los cambios, de forma que se mantenga la exactitud y la integridad de los elementos de configuración.
 - a) Confirmar que las revisiones están autorizadas.
 - b) Actualizar los elementos de configuración.
 - c) Archivar la línea base reemplazada y recuperar la nueva línea base.
 - d) Comentar los cambios realizados al elemento.
 - e) Vincular los cambios a los productos de trabajo relacionados, tales como requisitos, historias de usuarios y pruebas
- Realizar revisiones para asegurar que los cambios no hayan causado efectos no deseados en las líneas base (p. ej., asegurar que los cambios no hayan comprometido la seguridad o la protección del sistema).
- Registrar los cambios a los elementos de configuración y las razones de los cambios, según sea apropiado.
- Si se acepta un cambio propuesto al producto de trabajo, se identifica un calendario para incorporar el cambio al producto de trabajo y a otras áreas afectadas.

Los mecanismos de control de configuración se pueden adaptar a las categorías de los cambios. Por ejemplo, las consideraciones para la aprobación podrían ser menos rigurosas para los cambios de componentes que no afectan a otros componentes.

Los elementos de configuración modificados se liberan después de la revisión y de la aprobación de los cambios de configuración. Estos no son oficiales hasta que se liberen.

CM 2.5 Establecer los registros de gestión de configuración

El estado de la configuración requiere la actualización y mantenimiento de registro que describan los elementos de la configuración para revisar y reportar los activos bajo la gestión de la configuración del proyecto. (Engineering Standards Committee of the IEEE Computer Society, 2012)

En la misma norma se recomienda que los elementos de la configuración deben contener la siguiente información (CMMI 2.0, 2018):

- Fecha de origen.
- Localización.
- Versión aprobada.
- Estado.
- Trazabilidad.

De tal modo que, cuando las organizaciones logran el cumplimiento de esta práctica son capaces de realizar reportes en base a la planificación y la información de los elementos de la configuración. Por lo que, mantienen mecanismos y medios para la extracción de información, entre ellos:

- Revisiones y registro de los elementos.
- Registro de solicitud de cambio.
- Estado de elementos de la configuración.
- Revisión de las líneas base.

Establecer y mantener los registros que describen los elementos de configuración, para lo cual se debe incluir los siguiente:

- Historial de revisiones de los elementos de configuración.
- Registro de cambios.
- Registros de peticiones de cambio.
- Estado de los elementos de configuración.
- Diferencias entre líneas base.

Para una adecuada gestión se debe mantener las siguientes sub prácticas:

- Registrar las acciones de gestión de configuración con suficiente detalle para que se conozca el contenido y el estado de cada elemento de configuración, y para que se puedan recuperar versiones anteriores.
- Asegurar que las partes interesadas relevantes tengan acceso y conocimiento del estado de configuración de los elementos de configuración.
- Especificar la última versión de las líneas base.
- Identificar la versión de los elementos de configuración que constituyen una línea base particular.
- Describir las diferencias entre líneas base sucesivas.
- Modificar, si procede, el estado y la historia (es decir, cambios y otras acciones) de cada elemento de configuración.

CM 2.5 Establecer los registros de gestión de configuración

Según Bryan, Siegel y Whiteleather (1982) dice, las metodologías ágiles son una combinación de varios métodos de desarrollo entre incrementales e iterativas. Destacan seis factores que deben revisarse periódicamente: el cumplimiento, el flujo de trabajo, la seguridad, el proceso gestión, revisión de códigos, gestión de la construcción y el trabajo en equipo.

Dai y Vasarhelyi (2016), en cambio, proponen una auditoría acorde a la industria 4.0 basados en seis principios de diseño e implementación: interoperabilidad, virtualización, descentralización, capacidad en tiempo real, orientación a los servicios y modularidad. Debido a la presión que reciben las organizaciones para entregar con mayor frecuencia productos de software.

El incremento de la frecuencia de liberaciones de software incluye mayor incremento de auditorías, revisiones o inspecciones por el equipo de desarrollo, que garanticen la calidad del software. Sin embargo, el 80% de las pequeñas organizaciones analizadas no realiza esta práctica.

Realizar auditorías de configuración para mantener la integridad de las líneas base de configuración.

Las auditorías de configuración confirman que las líneas base y la documentación resultante se ajustan a un estándar o requisito especificado. Los registros relativos a los elementos de configuración pueden existir en múltiples bases de datos o sistemas de gestión de configuración. En tales circunstancias, las auditorías de configuración se deberían ampliar a las otras bases de datos, según proceda, para asegurar la precisión, consistencia y completitud de la información del elemento de configuración (véase la definición de “auditoría de la configuración” en el glosario).

Algunos ejemplos de tipos de auditoría son:

- Auditoría de configuración funcional (FCA): las auditorías que se llevan a cabo para verificar que el desarrollo de un elemento de configuración ha sido satisfactoriamente completado, que el elemento cumple con las características funcionales y de atributos de calidad especificados en la línea base funcional o asignada y que sus documentos operativos y de soporte están completos y son satisfactorios.
- Auditorías de configuración física (PCA): auditorías que se llevan a cabo para verificar

que un elemento de configuración, tal como fue construido, es conforme con la documentación técnica que lo define y describe.

- Auditorías de gestión de la configuración: auditorías que se llevan a cabo para confirmar que los registros de gestión de configuración y los elementos de configuración son completos, consistentes y precisos.

Como resultado de este punto se debe obtener, resultados de la auditoría de configuración y elementos de acción, en los cuales se debe considerar las siguientes sub prácticas:

- Evaluar la integridad de las líneas base.
- Confirmar que los registros de gestión de configuración identifican correctamente los elementos de configuración.
- Revisar la estructura y la integridad de los elementos en el sistema de gestión de configuración.
- Confirmar que los elementos en el sistema de gestión de configuración son completos, correctos y consistentes. El contenido del sistema de gestión de configuración completo, correcto y consistente se basa en los requisitos indicados en el plan, así como en las peticiones de cambio aprobadas.
- Confirmar el cumplimiento con los estándares y procedimientos aplicables de gestión de configuración.
- Seguir los elementos de acción desde la auditoría hasta su cierre.

ITIL

Gestión de la Configuración

Para cualquier organización es importante disponer de información sobre su

infraestructura TI, y mantenerla lo más actualizada posible.

La información que este proceso ha de mantener actualizada, tiene que ver con los elementos de configuración (CI) y sus relaciones en la infraestructura. Estos CI se componen de los siguientes elementos:

- Hardware.
- Software.
- Personas.
- Componentes de red.
- Líneas de negocio

Los informes han de aportar, al menos, la siguiente información:

- Financiera y política del producto: valor y depreciación de los componentes; licencias; nivel de estandarización de la infraestructura.
- Evaluación de impacto: componentes afectados por el despliegue de nuevas actuaciones y cambios; componentes necesarios frente a desastres y contemplados en el plan de recuperación.
- Provisión de servicios y precio: configuraciones de TI necesarias para ofrecer servicio; componentes necesarios.

El objetivo de la gestión de la configuración es, por lo tanto, divisible en tres partes:

- Proveer información y documentación suficiente para colaborar e interactuar con el resto de procesos de gestión de servicios.
- Establecer un control objetivo de los componentes de la infraestructura y de los servicios, generando informes de detalle de éstos, que permitan identificar, mantener

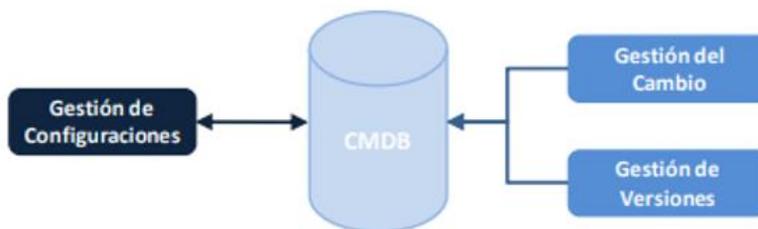
y verificar las versiones de los elementos de configuración (CI).

- Monitorizar periódicamente la configuración de los sistemas para contrastarla con la almacenada en la CMDB.

Las relaciones del proceso con la gestión de cambios y de versiones puede verse en la siguiente figura, donde la gestión de configuraciones se alimenta de la información cargada en la CMDB, para realizar los ajustes necesarios que los otros procesos han definido, a su vez que retroalimenta esta base de datos con la información más actualizada de las nuevas configuraciones de los componentes de la infraestructura TI.

Figura 3

Relaciones con otros procesos de la gestión de la configuración



Nota. En el gráfico se puede observar la relación de CM con otros procesos

La ejecución de este proceso se establece en diversas actividades que se desarrollan en los siguientes apartados: planificación, identificación, monitorización, control, verificación e informes.

Planificación

La planificación de esta actividad es, determinar fundamentalmente ciertos aspectos a tener en cuenta, para conseguir una adecuada gestión de la configuración que sea eficaz y para evitar las consecuencias que una mala gestión de este proceso puede traer para la organización, tanto en temas económicos como en la prestación de los servicios a los clientes. Para ello será necesario al menos:

- Definir un responsable.
- Analizar concienzudamente la información disponible, los recursos, licencias, componentes, etc., ya existentes.
- Establecer una política y estrategia de configuraciones que aclare el alcance de las actuaciones.
- Establecer un medio de comunicación activo con los procesos de gestión de cambios y versiones (como se ha visto reflejado en la Fig. 3)

Identificación

Para la identificación de los componentes y de sus configuraciones es necesario establecer un procedimiento que defina los pasos para generar informes en la CMDB y cómo mantenerla actualizada.

Este procedimiento debe incluir el desarrollo de una nomenclatura específica, las relaciones existentes entre los componentes, responsables y usuarios asignados, el estado actual, documentación y manuales disponibles de esos componentes.

En este procedimiento debe definirse el alcance y el nivel de detalle:

- **Alcance.** En primer lugar, habremos de determinar qué componentes TI van a ser incluidos en la CMDB, para posteriormente definir las áreas incluidas en el alcance, así como la documentación que se solicitará y los datos asociados a los componentes.

Cualquier servicio o proceso es susceptible de ser incluido en la CMDB, pero unos objetivos muy ambiciosos resultan contraproducentes, por el enorme trabajo que puede suponer para este proceso en sus comienzos.

- **Nivel de detalle.** Definir el nivel de detalle es establecer una jerarquía o un árbol de componentes y elementos que conforman la infraestructura, de manera que, se pueda

determinar qué tipo de actuación hay que realizar para que la gestión de cambios tenga éxito en la interacción con este proceso.

En este subproceso se incluye la gestión de la librería de *software* definitivo (*Definitive Software Library – DSL*) y el almacén de hardware definitivo (*Definitive Hardware Storage – DHS*). Estos dos elementos son los depósitos que contienen las versiones definitivas de *software* o *hardware* que se emplea en la infraestructura en un momento determinado.

Monitorización

El ciclo de vida de un componente puede dividirse en fases o etapas, que pueden ser denominadas con una codificación que indique su estado en cada momento.

Este tipo de registro permite disponer de un indicador muy útil para monitorizar el ciclo de vida de un componente, además de facilitar información de interés para otros procesos, tales como gestión de cambios, gestión de la disponibilidad, etc.

Control

El objetivo del control de la gestión de configuraciones, es asegurar que sólo quedan registrados en la base de datos los CI que hayan sido autorizados, cuando se realiza un cambio en las características registradas o en las relaciones de éste con los demás elementos de la infraestructura.

Verificación y auditorías

Las auditorías son procesos de verificación del sistema de gestión de configuraciones (todas las CMDB unidas de las que disponga la organización), de manera que se compruebe que ésta refleja la realidad. Una auditoría ha de realizarse, al menos, cuando ocurra alguna de las situaciones siguientes:

- Cuando ocurra un desastre.

- Cuando se lleven a cabo cambios.
- Periódicamente.

Después de modificar o implantar una nueva CMDB.

Los beneficios de aplicar este proceso en la organización son múltiples, sobre todo porque facilita la actuación de otros procesos relacionados con el mantenimiento del servicio, lo que implica una provisión de servicios constante y de alta calidad por lo que a continuación se menciona:

- Se gestionan los componentes de la infraestructura individualmente para su mantenimiento.
- Se incrementa la eficacia en la solución de problemas y gestión de incidencias. Por un lado, éstas disminuyen y, el cliente queda más satisfecho al ver la pronta respuesta.
- Se facilita el análisis de impacto, debido a la cantidad de información de valor existente.
- Existe un mayor control del hardware y el software que hay en la organización.

Relaciones de este proceso con otros de ITIL.

- **Gestión de cambios:** cualquier cambio ha de estar registrado en la CMDB correspondiente, para que la gestión de configuraciones pueda obtener información de cómo actuar.
- **Gestión de versiones:** lo mismo que en el caso anterior, actualización de CMDB.
- **Gestión de niveles de servicio:** cualquier modificación en la configuración puede ocasionar cambios en los SLA, ofreciendo mejoras en diversos aspectos del servicio.

Gestión de Versiones y despliegue

El objetivo de este proceso es controlar, distribuir y poner en marcha el software y el hardware en el entorno de producción que ha sido autorizado en el proceso de gestión del cambio.

La gestión de versiones debe encargarse de dirigir e implementar los cambios en los servicios TI. Es responsabilidad de este proceso:

- Planificar, coordinar y poner en marcha el software (SW) y el hardware (HW).
- Asegurar que las nuevas instalaciones de SW y HW han sido autorizadas y son las correctas.
- Establecer canales adecuados de comunicación con el cliente.
- Cooperar con los procesos de gestión del cambio y de configuración para mantener actualizada la CMDB.

Las principales actividades de la gestión de versiones son las siguientes:

- Definir una política de nuevas versiones.
- Probar las nuevas versiones en un entorno de pruebas.
- Validar las nuevas versiones.
- Implementar las nuevas versiones en el entorno de producción.
- Comunicar a los clientes y usuarios las funcionalidades de la nueva versión.

Política y Planificación de Nuevas Versiones

La política de versiones se define para conocer cómo y en qué momento se despliegan las nuevas versiones. Esta planificación sólo es útil para SW básico, ya que, cuando se trata de desarrollos complejos, cada proyecto dispone de una planificación y un paso al entorno de producción propio.

Esta política, además, debe definir los CI que pueden ser distribuidos con

independencia de los demás, según:

- El impacto que generaría la versión sobre otros componentes.
- El coste económico asociado a las pruebas previas al despliegue.
- La dificultad y resistencia al cambio.
- Los responsables de las diferentes etapas del proceso.
- El ciclo de vida estimado de la nueva versión.
- Otros factores.

Diseño, Desarrollo y Configuración

Un aspecto importante a tener en cuenta es quién realiza el desarrollo, si la misma organización o una empresa externa. En el caso de que se trate de un desarrollo externo, habrá que tener muy en cuenta que se cumple con lo establecido en los RFC (siglas) correspondientes, y que los proveedores de servicio hayan sido evaluados previamente.

Así, se deben definir procedimientos que sirvan para desarrollar configuraciones estándar; es decir, establecer cómo se diseña el SW, cómo, dónde y quién desarrolla el SW, cómo y dónde se prueba el SW y el HW previo al cambio.

Uno de los aspectos que han de ser definidos en los procedimientos es el plan de recuperación o restauración a una versión previa, en el caso de que el cambio no haya sido satisfactorio.

Plan de Recuperación

El plan de recuperación en este caso será gestionado junto con el proceso de gestión del cambio, ya que, será necesario disponer de la posibilidad de volver a la versión anterior al despliegue, en el caso en que la implantación haya sido fallida.

Prueba y Validación de la Versión

El objetivo de la ejecución de pruebas es comprobar la validez de la versión previa al

despliegue.

Antes de la implantación de un nuevo SW o HW, éste ha de ser sometido a diversas pruebas funcionales por parte del equipo de desarrollo y de los usuarios futuros. Deben considerarse pruebas tales como:

- Pruebas unitarias.
- Pruebas de funcionalidad.
- Pruebas de operación.
- Pruebas de integración.
- Pruebas de rendimiento.

Además, debe generarse la siguiente documentación para conocer el estado del proceso, así como los resultados orientados a la mejora continua:

- Listas de errores.
- Pruebas de los planes de recuperación.
- Documentación para usuarios y personal de servicio referente a la funcionalidad y a la usabilidad de las nuevas versiones.

Se encarga de dar la validación final a la versión, para que se ponga en marcha la implantación, será el CAB (gestión del cambio). Para llegar a esta validación será necesaria la verificación del equipo de desarrollo.

Planificación de la Implantación

Existen dos maneras de desplegar una nueva versión:

- Por etapas: dividida en fases temporales y de localización.
- Integral: de manera completa y todos los puntos al mismo tiempo.

La planificación del despliegue o implantación debe definir los siguientes aspectos:

- Cronograma.

- Tareas.
- Responsabilidades.
- Personas necesarias, CI a instalar, modificar o borrar.
- Notificaciones.
- Actualización de la CMDB.
- Programación de reuniones de coordinación y comunicación

Tras la puesta en marcha comenzarán, con toda seguridad, a recibirse incidencias y comentarios, por lo que el centro de servicios debe mantener constantemente informada a la gestión de versiones para que esta información sea analizada. El análisis será parte de la mejora y en las próximas versiones serán incorporadas las sugerencias más interesantes.

Comunicación, Formación y Capacitación

- **Comunicación.** El personal al que afectará la nueva versión debe estar informado de los planes de implantación y cómo incidirán éstos en su trabajo diario. Asimismo, el personal del centro de servicios también debe conocer cómo se están llevando a cabo los cambios para poder mantener informado al cliente y a los usuarios.
- **Formación.** Existen multitud de buenas prácticas para una correcta formación de los usuarios de las nuevas versiones desplegadas, pero las más comunes son realizar cursos de sensibilización on-line previos, para más tarde poner en marcha cursos de formación presencial, reuniones de trabajo, discusión o foros de comunicación y resolución de dudas. Todo ello debe ser dimensionado dependiendo de la medida que tenga el cambio que va a producir en la organización la(s) nueva(s) versión(es).
- **Capacitación.** La capacitación será el último paso en el que, siempre que sea necesario, se plantearán cursos prácticos que podrán ser evaluados, sobre el uso de la nueva versión implantada.

Distribución de Versiones

La distribución de versiones no complejas puede realizarse de dos maneras:

- Manual.
- Automática.

Es una buena práctica desarrollar herramientas automáticas de instalación de versiones (distribución) para el software. Existe la posibilidad de disponer distribuciones de los dos tipos, por localizaciones de la organización:

- Automatizadas, donde no existan procesos críticos.
- Manuales, donde una versión fallida pueda provocar una disminución en la calidad o una interrupción en la prestación del servicio.

Estas prácticas permiten verificar el éxito de las versiones de una manera más rápida y menos costosa, por consiguiente, ahorro en costes para la empresa.

Los beneficios de disponer de un proceso de gestión de versiones es que garantiza la implantación, con menor riesgo, de un software y un hardware, cuya producción es de mayor calidad, así como menor cantidad de problemas derivados, menos implantaciones fallidas y, por lo tanto, ahorro en costes para la organización.

Consideraciones

Para recordar:

- Los cambios ocurren.
- La gestión del cambio trata de asegurar una continuidad y calidad del servicio cuando se están llevando a cabo cambios en él.
- Los cambios pueden ser de índole documental, de software, hardware, componentes y personas.

- Las configuraciones de los componentes de la infraestructura provienen de la información que los RFC aportan, para que los CI se ajusten a las necesidades del servicio.
- La puesta en marcha de nuevo software o hardware ha de realizarse con cuidado, ya que un despliegue en la infraestructura TI que no haya sido comprobado previamente, puede ocasionar fallos en la prestación del servicio. De esto se encarga la gestión de versiones.
- La gestión del conocimiento es un proceso que se integra dentro de éstos y otros procesos de prestación del servicio. Gracias a ella, podremos hacer mejor uso de la información de la que disponemos y la capacitación de las personas será más eficaz; entre otras cosas, porque todas las personas se sienten implicadas en el aporte de datos para la mejora del conocimiento de la organización.
- La CMDB (Change Management Data Base) o base de datos de gestión del cambio es un lugar donde se localiza toda la información relativa a los tres procesos (cambio, versiones y configuración). Dicha base de datos tiene que ofrecer información y debe ser retroalimentada constantemente para estar siempre actualizada.
- Pueden existir múltiples CMDB, una para cada necesidad normalmente, si se trata de grandes organizaciones. En tal caso, el conjunto de las bases de datos se llama CMS, Change Management System

Consideraciones Finales

La política de versiones se define para conocer cómo y en qué momento se despliegan las nuevas versiones. Esta planificación sólo es útil para SW básico, ya que, cuando se trata de desarrollos complejos, cada proyecto dispone de una planificación y un paso al entorno de producción propio.

Esta política, además, debe definir los CI que pueden ser distribuidos con independencia de los demás, según:

- El impacto que generaría la versión sobre otros componentes.
- El coste económico asociado a las pruebas previas al despliegue.
- La dificultad y resistencia al cambio.
- Los responsables de las diferentes etapas del proceso.
- El ciclo de vida estimado de la nueva versión.
- Otros factores

Diseño, Desarrollo y Configuración

Un aspecto importante a tener en cuenta es quién realiza el desarrollo, si la misma organización o una empresa externa. En el caso de que se trate de un desarrollo externo, habrá que tener muy en cuenta que se cumple con lo establecido en los RFC (siglas) correspondientes, y que los proveedores de servicio hayan sido evaluados previamente.

Así, se deben definir procedimientos que sirvan para desarrollar configuraciones estándar; es decir, establecer cómo se diseña el SW, cómo, dónde y quién desarrolla el SW, cómo y dónde se prueba el SW y el HW previo al cambio.

Uno de los aspectos que han de ser definidos en los procedimientos es el plan de recuperación o restauración a una versión previa, en el caso de que el cambio no haya sido satisfactorio.

Plan de Recuperación

El plan de recuperación en este caso será gestionado junto con el proceso de gestión del cambio, ya que, será necesario disponer de la posibilidad de volver a la versión anterior al despliegue, en el caso en que la implantación haya sido fallida.

Prueba y Validación de la Versión

El objetivo de la ejecución de pruebas es comprobar la validez de la versión previa al despliegue.

Antes de la implantación de un nuevo SW o HW, éste ha de ser sometido a diversas pruebas funcionales por parte del equipo de desarrollo y de los usuarios futuros. Deben considerarse pruebas tales como:

- Pruebas unitarias.
- Pruebas de funcionalidad.
- Pruebas de operación.
- Pruebas de integración.
- Pruebas de rendimiento.

Además, debe generarse la siguiente documentación para conocer el estado del proceso, así como los resultados orientados a la mejora continua:

- Listas de errores.
- Pruebas de los planes de recuperación.
- Documentación para usuarios y personal de servicio referente a la funcionalidad y a la usabilidad de las nuevas versiones.

Se encarga de dar la validación final a la versión, para que se ponga en marcha la implantación, será el CAB (gestión del cambio). Para llegar a esta validación será necesaria la verificación del equipo de desarrollo.

Consideraciones Finales

En este capítulo se hace referencia a la revisión de la literatura de los diferentes artículos científicos y no científicos relacionados con marcos de trabajo para la gestión de configuración, considerando los elementos más importantes que deben ser analizados.

Los principales resultados que podemos destacar de este capítulo son los siguientes:

- La revisión sistemática realizada se ha enfocado en los marcos de trabajo para la gestión de la configuración, en donde se ha podido observar que hacen referencia a CMMI e ITIL.
- La primera se centra en los procesos asociados al desarrollo de software, mientras que la segunda enfatiza en el enfoque a servicios.
- En ITIL, la gestión de la configuración proporciona información sobre los elementos de configuración que contribuyen a cada servicio y sus relaciones.

Capítulo III

Resolución

En el presente capítulo, se realiza una descripción detallada del marco de trabajo para la gestión de configuración de software propuesta en el presente trabajo de investigación, detallando para ello el conjunto de elementos que lo conforman, así como los procesos que se han seguido para su definición y diseño. Para ello en los siguientes apartados se describen los pasos seguidos, identificando los elementos describiendo un modelo de proceso y describiendo las actividades a ser consideradas.

En cada uno de los apartados mencionados se especifican los factores que conforman el marco propuesto, su diseño y el proceso que se debe seguir para la implementación. Así también se describe el proceso que es seguido en la etapa de revisión de la literatura del tema abordado.

Revisión sistemática de Gestión de Configuración

Durante la fase de identificación de los elementos a ser considerados en la gestión de configuración, de la revisión Sistemática realiza y descrita en el capítulo 2, se pudo identificar que los más nombrados son el CMMI e ITIL, de los cuales, las características y particularidades se describen en el siguiente cuadro descriptivo que identifica las características más importantes a ser considerados y quienes son los comunes de los marcos de trabajo.

- Gestión de las líneas base de los elementos de configuración.
- Seguimiento y control de los cambios de los elementos de configuración.
- Gestión de auditorías y establecer los registros de gestión de configuración.

Tabla 6*CMMI 2.0 Prácticas Sobre la Gestión de la Configuración (CM)*

Característica	CMMI	ITIL
Identificar los elementos de configuración	X	X
Establecer un sistema de gestión de configuración.	X	
Crear o liberar las líneas base.	X	X
Seguir las peticiones de cambio.	X	X
Controlar los elementos de configuración.	X	X
Establecer los registros de gestión de configuración.	X	X
Realizar auditorías de configuración.	X	X
Comunicación, formación y Capacitación.		X
Relación entre elementos de configuración.		X

Nota. En la tabla se indica las características para emplearse en un marco de trabajo.

De este cuadro se pudo desprender que las características que debe ser consideradas en la descripción de un marco de trabajo para la gestión de configuración, tomando en cuenta que, CMMI está enfocado en los procesos de la gestión del desarrollo de software e ITIL en la descripción de los elementos de configuración que apoyan a los servicios de TI de las organizaciones.

Las características a ser consideradas son los siguientes:

Marco de trabajo de Gestión de Configuración de Software

Elementos del marco para la Gestión de Configuración de Software

En este punto se describen los elementos y relaciones que se identificaron que se deben ser considerados para proponer el marco de trabajo de gestión de configuración de software.

- **Elementos de Configuración.** – Son todos los artefactos que se van a considerar en la gestión de configuración de software.
- **Actores:** Personas o departamentos que participan en el proceso de gestión de la configuración de software.
- **Jefe de TI.** – Responsable de un área de tecnología por medio del cual se canalizan los pedidos o solicitudes de gestión de configuración.
- **Desarrollo.** - Área de un departamento de tecnología encargados de la gestión del ciclo de vida del desarrollo de software.
- **Administrador Configuración.** – Personal encargado y responsable de gestionar los elementos de configuración.
- **Versionador.** - Persona encargada de administrar las versiones de los artefactos que están siendo considerados como parte de la gestión de configuración de software.

- **Control de Calidad.** - Persona encargada de verificar que los artefactos de la gestión de configuración cumplan con las especificaciones, contenido y formatos establecidos y definidos en a la organización.
- **Actividades de Gestión de la Configuración:** Las tareas a ser ejecutadas que han sido consideradas en la gestión de configuración de software.
- **Solicitar Documentos.** – En esta actividad se solicita un artefacto para su utilización y actualización de ser necesaria.
- **Verificar Solicitud.** – Se verifica que la solicitud de pedido de un artefacto se encuentre definida según lo establecido por la organización.
- **Verificar Línea Base.** – Determinar el estado de la línea base del artefacto, que incluye los siguientes estados, creado, actualizado, por crear, y se responde según el estado de la línea base.
- **Enviar Documentos.** – El área de desarrollo envía los artefactos creados al versionar para ser gestionados y crear la línea base.
- **Crear Línea Base.** – Se crea o actualiza la línea base del artefacto que se está gestionando.
- **Entregar Información.** – Determinar el estado de la línea base del artefacto, que incluye los siguientes estados, creado, actualizado, por crear, y se responde según el estado de la línea base.
- **Enviar Documentos.** – Una vez determinado el estado de la línea base se envía el artefacto al solicitante según la solicitud enviada.
- **Revisar Información.** – Se valida la información a ser entregada según solicitud realizada, cuando se trata de pedidos de auditoria o de entes de control.
- **Actualizar Documentos (Desarrollo).** – Se actualiza los artefactos solicitados por desarrollo con las nuevas especificaciones o cambios realizados.

- **Verificar Documentación.** – Se verifica actualiza los artefactos solicitados por desarrollo con las nuevas especificaciones o cambios realizados.
- **Actualizar Línea Base.** – Se actualiza la línea base de los artefactos solicitados que incluya las modificaciones y actualizaciones realizadas.
- **Informar Actualización Línea Base.** – Se informa a todos los interesados de la actualización efectiva de la línea base en el sistema de gestión de configuración de software.

De los roles y responsabilidades

Las responsabilidades generales a nivel de los actores que son parte del presente procedimiento se establecen de la siguiente manera:

Tabla 7

CMMI 2.0 Prácticas Sobre la Gestión de la Configuración (CM)

Rol	Responsabilidades
Director de TI	<ul style="list-style-type: none"> • Velar por el cumplimiento de la normativa legal vigente • Gestionar la planificación, identificación, control y monitoreo de todos los elementos de configuración en la CMDB (Configuration Management DataBase) • Promover el uso efectivo de la CMDB dentro de la Dirección. • Asegurar la consistencia e integridad de la información de la CMDB a través de la ejecución de procedimientos de monitoreo. • Aprobar cambios estructurales en la CMDB
Sub direcciones	<ul style="list-style-type: none"> • Revisar y aprobar la estructura de la configuración de la CMDB de acuerdo al ámbito de su competencia

Rol	Responsabilidades
Responsable de la configuración	<ul style="list-style-type: none">• Definir alcance y nivel de detalle de la configuración• Establecer la estructura de la configuración• Establecer plan de monitoreo de la configuración• Elaborar cronograma de monitoreo de la configuración• Revisión de productos generados• Reportar cualquier discrepancia o no conformidad en los elementos de configuración.• Liderar las actividades de evaluación del proceso: revisar tipos de elementos de configuración, relaciones, atributos y valores asociados, estructura de la base de datos, derechos de acceso, entre otros• Monitorizar y reportar los cambios no autorizados sobre los elementos de configuración• Asegurar que todos los elementos de configuración se encuentran registrados de forma adecuada en la CMDB.• Participar en la mejora continua del proceso de Gestión de la Configuración.• Revisar que los elementos de configuración se encuentren registrados en la CMDB con el estado y datos de configuración apropiados.
Analista(s) de la	<ul style="list-style-type: none">• Levantar la información de la estructura de la configuración de

Rol	Responsabilidades
configuración	<p>acuerdo al ámbito de su competencia</p> <ul style="list-style-type: none"> • Realizar el registro y/o actualización de los elementos de configuración en la CMDB una vez que los productos sean liberados a producción • Elaborar informe de productos generados • Registrar línea base CMDB, basada en el levantamiento de información realizado por las diferentes subdirecciones • Asegurar la idoneidad e integridad de los elementos de configuración, mediante la ejecución de monitoreo de la CMDB, siempre y cuando no corresponda a la misma subdirección solicitante del monitoreo. • Trabajar conjuntamente con el responsable de la Configuración para identificar las causas de cualquier problema identificado en los monitoreos e implementar las acciones correctivas

Nota. En la tabla se indica los roles y responsabilidades del marco de trabajo

Creación o actualización de la CMDB

Tabla 8

Descripción, creación o mejora de CMDB

Ficha Técnica	
NOMBRE DEL SUB	Creación o mejora de la CMDB
PROCESO	

 Ficha Técnica

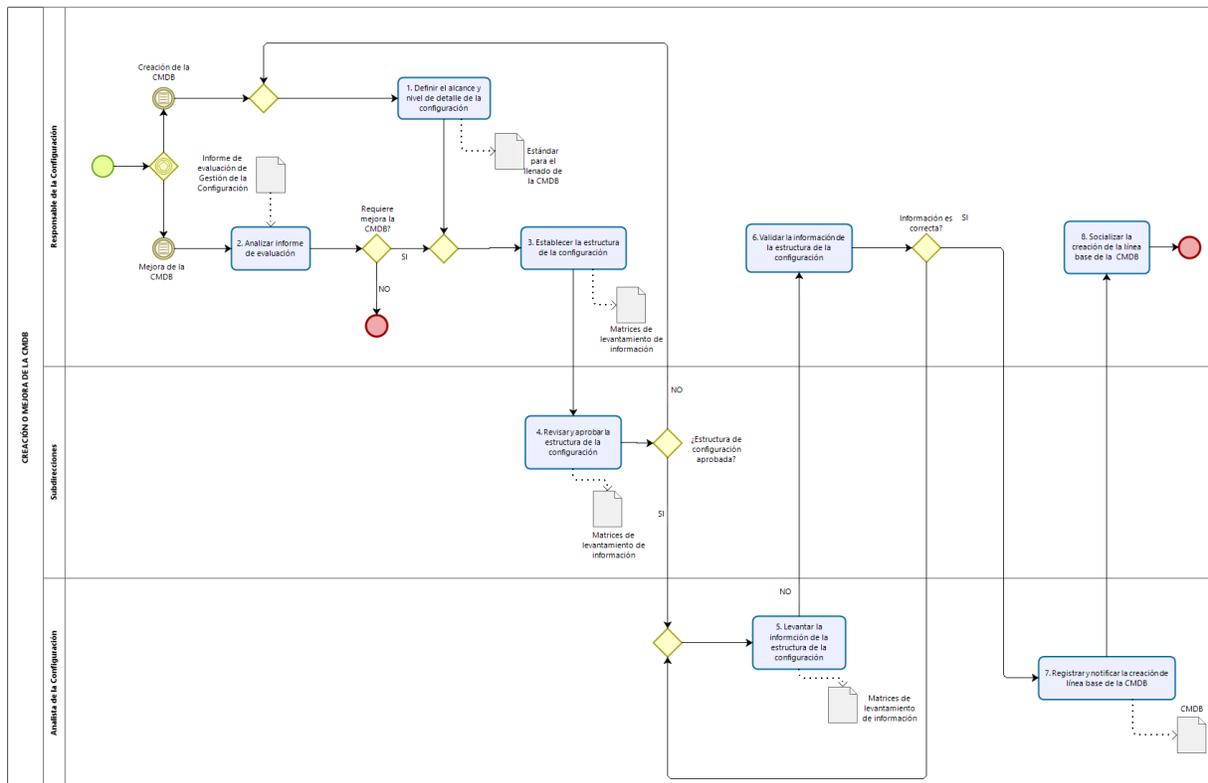
RESPONSABLE(S) DEL SUB PROCESO	Director de TI
OBJETIVO GENERAL	Establecer y mantener la estructura de configuración de servicios e infraestructura tecnológica; registro de los elementos de configuración (CI) y las relaciones entre estos.
ALCANCE	Desde el análisis para la creación/mejora de la estructura de la CMDB hasta el registro de la CMDB.
ÁMBITO DE APLICACIÓN	Todas las unidades de negocio que prestan servicios a través de sistemas tecnológicos administrados por la Dirección de TI.
ENTRADAS	Nuevo: Necesidad de la Dirección de TI Actualización: Informe de monitoreo del Proceso de Gestión de Configuración
PRODUCTOS / SERVICIOS DEL SUBPROCESO	CMDB actualizada (Arquitectura de datos y componentes de software)
TIPO DE CLIENTE	Internos: <ul style="list-style-type: none"> • Subdirecciones existentes, Directivos Externos: <ul style="list-style-type: none"> • N/A

Nota. En la tabla se la descripción, creación o mejora de CMDB

Descripción del sub proceso

Figura 4

Diagrama de Creación Mejora de la CMDB



Nota. En el gráfico se puede observar el diagrama de mejora de la CMDB

Procedimientos para creación o mejora de la CMDB

Tabla 9

Procedimiento para Creación o Mejora de CMDB

No.	Actividad	Descripción de actividad	Responsable	Documento/Sistema
	Disparador	Corresponde a Creación de CMDB: Continúa con la actividad 1.		N/A

No.	Actividad	Descripción de actividad	Responsable	Documento/Sistema
		Corresponde a Mejora de la CMDB: Continúa con la actividad 2		
1	Definir el alcance y nivel de detalle de la configuración	Se debe establecer los elementos configurables de servicios e infraestructura tecnológica que van a ser incluidos en la CMDB junto con las características de dichos elementos.	Responsable de la Configuración	Estándar para el llenado de la CMDB
2	Analizar informe de monitoreo	Se debe realizar una revisión y análisis del informe generado en el monitoreo de la configuración, con la finalidad de determinar: Si la CMDB requiere de mejoras para el cumplimiento de su objetivo en base a la definición del alcance y nivel de detalle establecido	Responsable de la Configuración	Informe de monitoreo del Proceso de Gestión de Configuración

No.	Actividad	Descripción de actividad	Responsable	Documento/Sistema
	¿Requiere actualización de CMDB?	<p>Respuesta Afirmativa:</p> <p>Continúa en la actividad 3.</p> <p>Respuesta Negativa:</p> <p>Finaliza el sub proceso</p>		N/A
3	Establecer la estructura de la configuración	<p>Consiste en construir una estructura basada en el alcance establecido en la actividad 1, la cual contemple la información de los tipos de elementos de configuración, atributos, tipos de relaciones, atributos de relaciones y códigos de estado.</p>	Responsable de la configuración	Matrices de levantamiento de información
4	Revisar y aprobar la estructura de la configuración	<p>Consiste en la verificación de la información establecida en la estructura de la configuración de acuerdo al ámbito de competencia, es decir se valida si dicha información cumple con el propósito definido para el proceso de</p>	Subdirectores de la Dirección de TI	Matrices de levantamiento de información

No.	Actividad	Descripción de actividad	Responsable	Documento/Sistema
		Gestión de la Configuración.		
		Una vez validada la estructura de la configuración y la misma se encuentre correcta se procede con la aprobación de la misma.		
	¿Estructura de configuración aprobada?	<p>Respuesta Afirmativa: Continúa con la siguiente actividad Nro. 4.</p> <p>Respuesta Negativa: Continúa con la actividad Nro. 1</p>		N/A
5	Levantar información de la estructura de la configuración	Se debe obtener la información requerida en cada una de las estructuras de configuración establecidas y proceder a registrarlos en los mismos.	Analista de la Configuración	Matriz de levantamiento de información (actualizada)
6	Validar la información de la	Consiste en constatar físicamente la información	Responsable de la	N/A

No.	Actividad	Descripción de actividad	Responsable	Documento/Sistema
	estructura de la configuración	ingresada en las matrices de levantamiento de información contra lo existente en la Dirección de TI	Configuración	
	¿Información es correcta?	<p>Respuesta Afirmativa: Continúa con la siguiente actividad 6.</p> <p>Respuesta Negativa: Continúa con la actividad 4.</p>		N/A
7	Registrar y notificar la creación de línea base de la CMDB	<p>Consiste en ingresar la información de las estructuras de la configuración validadas en la actividad anterior en la CMDB, con la finalidad de contar con una línea base y a continuación proceder a comunicar al responsable de la Configuración la creación de la estructura de la</p>	Analista de la Configuración	CMDB V.1.0

No.	Actividad	Descripción de actividad	Responsable	Documento/Sistema
		configuración.		
8	Socializar la creación de la línea base de la CMDB	Poner en conocimiento a las subdirecciones de la Dirección de TI, que se ha establecido y creado la CMDB para que puedan hacer uso de la misma, la cual se puede realizar mediante correo electrónico institucional o reunión de trabajo	Responsable de la Configuración	N/A

Nota. En la tabla se observa el Procedimiento para Creación o Mejora de CMDB

Proceso para Registro y/o actualización de los elementos de configuración

Tabla 10

Descripción, creación o mejora de CMDB

Ficha Técnica	
NOMBRE DEL SUB PROCESO	Registro y/o actualización de los elementos de configuración
RESPONSABLE(S) DEL SUB PROCESO	Director de TI
OBJETIVO GENERAL	Mantener actualizado los elementos de configuración (CI's).
ALCANCE	Desde la identificación de los cambios realizados en los

 Ficha Técnica

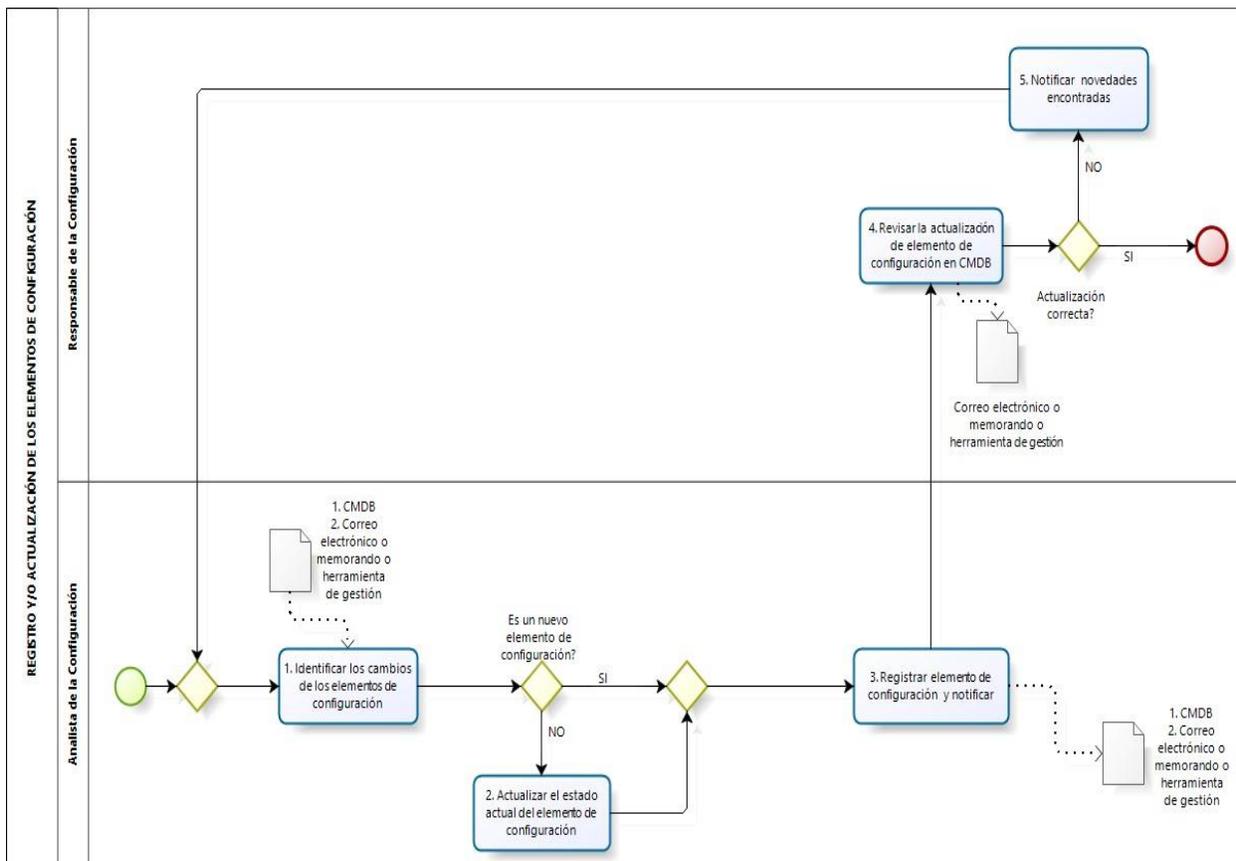
	elementos de configuración, manejo de históricos de los elementos de configuración y registro/actualización de los elementos de configuración en la CMDB
ÁMBITO DE APLICACIÓN	Todas las unidades de negocio que prestan servicios a través de sistemas tecnológicos administrados por la Dirección de TI.
ENTRADAS	CMDB
PRODUCTOS / SERVICIOS DEL SUBPROCESO	CMDB actualizada (Arquitectura de datos y componentes de software)
TIPO DE CLIENTE	<p>Internos:</p> <ul style="list-style-type: none"> • Subdirecciones existentes, Directivos <p>Externos:</p> <ul style="list-style-type: none"> • N/A

Nota. En la tabla se observa la descripción, creación o mejora de CMDB

Descripción del sub proceso

Figura 5

Diagrama de Registro y/o Actualización de los Elementos de Configuración



Nota. En la figura se observa el flujo de registro y/o actualización de los Elementos de Configuración

Registro y/o actualización de ítems de configuración

Tabla 11

Registro y/o Actualización de ítems de configuración

No.	Actividad	Descripción de actividad	Responsable	Documento/Sistema
				CMDB actualizada
1	Identificar los cambios de los elementos de configuración.	Determinar los cambios que se han realizado en los elementos de configuración, ya sea a nivel de servicio o infraestructura tecnológica.	Analista de la Configuración	Correo electrónico o memorando o herramientas de gestión
	¿Es un nuevo elemento de configuración?	<p>Respuesta Afirmativa:</p> <p>Continúa con la actividad 3.</p> <p>Respuesta Negativa:</p> <p>Continua con la siguiente actividad 2.</p>		N/A
2	Actualizar el estado actual del elemento de configuración.	Consiste en cambiar a Inactivo el estado del CI, con la finalidad de mantener un histórico de los cambios realizados en la CMDB.	Analista de la Configuración	N/A

No.	Actividad	Descripción de actividad	Responsable	Documento/Sistema
3	Registrar elemento de configuración y notificar la acción realizada.	Se debe registrar el elemento de configuración en la CMDB para que la misma se encuentre actualizada y a continuación se notifica la/las acciones/es realizadas al responsable de la Configuración.	Analista de la Configuración	CMDB Actualizada Correo electrónico o memorando o herramienta de gestión
4	Revisar la actualización de CI en CMDB	Revisar que la información del cambio realizado al elemento de configuración se encuentre registrado en la CMDB, en base a la información proporcionado por el Analista de la Configuración	Responsable de la Configuración	Correo electrónico o memorando o herramienta de gestión
	¿Actualización correcta?	Respuesta Afirmativa: El subproceso finaliza. Respuesta Negativa: Continúa con la siguiente actividad 5.		N/A
5		Notificar novedades	Para los casos	Responsable de la

No.	Actividad	Descripción de actividad	Responsable	Documento/Sistema
		encontradas	en los cuales se han encontrado novedades en la actualización de los elementos de configuración en la CMDB, se informa las novedades encontradas durante la verificación por el mismo medio que se recibió la información.	Configuración

Nota. En la tabla se observa el Registro y/o Actualización de ítems de configuración

Tabla 12

Proceso de Monitoreo de la configuración

Ficha Técnica	
NOMBRE DEL SUB	Monitoreo de la configuración.
PROCESO	

 Ficha Técnica

RESPONSABLE(S) DEL Director de TI.

SUB PROCESO

OBJETIVO GENERAL Revisar periódicamente el repositorio de configuración y verificar su integridad y completitud del CMDB de acuerdo con el alcance definido.

ALCANCE Desde la planificación de las actividades, ejecución del monitoreo hasta la elaboración de informes del monitoreo realizado de acuerdo a las novedades obtenidas en el monitoreo.

ÁMBITO DE APLICACIÓN Todas las unidades de negocio que prestan servicios a través de sistemas tecnológicos administrados por la Dirección.

ENTRADAS CMDB.

PRODUCTOS / SERVICIOS DEL SUBPROCESO Informe de Monitoreo del Proceso Gestión de la Configuración (Análisis de impacto en los componentes aplicativos)

TIPO DE CLIENTE Internos:

- Subdirecciones existentes, Directivos

Externos:

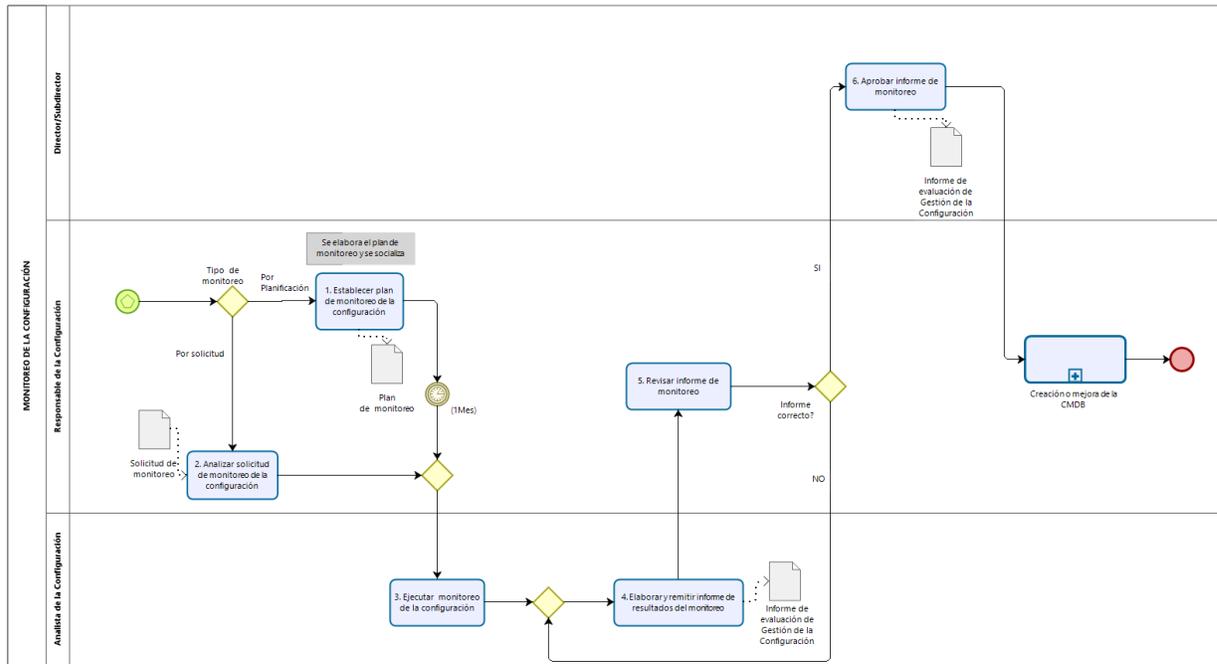
- N/A

Nota. En la tabla se observa la descripción, creación o mejora de CMDB

Descripción del sub proceso

Figura 6

Diagrama de Monitoreo de la Configuración



Nota. En la figura se observa el Monitoreo de la Configuración

Procedimiento para Monitoreo de la Configuración

Tabla 13

Procedimiento de Monitoreo de la configuración

No.	Actividad	Descripción de actividad	Responsable	Documento/Sistema
	Tipo de monitoreo	Por Planificación:		N/A
		Continúa en la actividad		
		1.		
		Por solicitud: Continúa		
		en la actividad 2.		

No.	Actividad	Descripción de actividad	Responsable	Documento/Sistema
1	Establecer plan de monitoreo de la configuración.	Elaborar una estrategia para la verificación de la utilización y control de la CMDB y la completitud de la misma de acuerdo con el alcance definido. Una vez definida la estrategia, se iniciará la ejecución de la misma al mes siguiente de formalizado el mismo.	Responsable de la Configuración	Plan de monitoreo
2	Analizar solicitud de monitoreo de la configuración.	Identificar los elementos de configuración a ser verificados dentro de la CMDB.	Responsable de la Configuración	Solicitud de monitoreo
3	Ejecutar monitoreo de la configuración	Verificación de los elementos de configuración de lo puesto en marcha y en producción contra la información registrada en la CMDB y/o completitud de la misma de acuerdo con el alcance definido.	Analista(s) de la Configuración	N/A

No.	Actividad	Descripción de actividad	Responsable	Documento/Sistema
4	Elaborar y remitir informe de resultados del monitoreo.	Identificación, análisis de los hallazgos encontrados durante la ejecución del monitoreo plasmados en un documento de evaluación.	Analista de la Configuración	Informe de monitoreo del Proceso de Gestión de Configuración
5	Revisar informe de monitoreo.	Verificar que el informe de monitoreo generado (Informe de monitoreo del Proceso de Gestión de Configuración), contiene todos los insumos necesarios para que pueda ser emitido.	Responsable de la Configuración	N/A
	¿Informe correcto?	Respuesta Afirmativa: Continúa con la siguiente actividad 6. Respuesta Negativa: Continúa con la actividad 4.		N/A
6	Aprobar informe de monitoreo.	Validar que el Informe de monitoreo del Proceso de Gestión de Configuración	Subdirector de la Dirección de TI	Informe de monitoreo del Proceso de Gestión de

No.	Actividad	Descripción de actividad	Responsable	Documento/Sistema
		es correcto.		Configuración Aprobado
7	Creación o actualización de la CMDB.	Ejecución del subproceso de la CMDB Creación o actualización de la CMDB	Responsable de la Configuración	N/A

Nota. En la tabla se observa el Procedimiento de Monitoreo de la configuración

Automatización del marco de trabajo para la Gestión de Configuración de Software

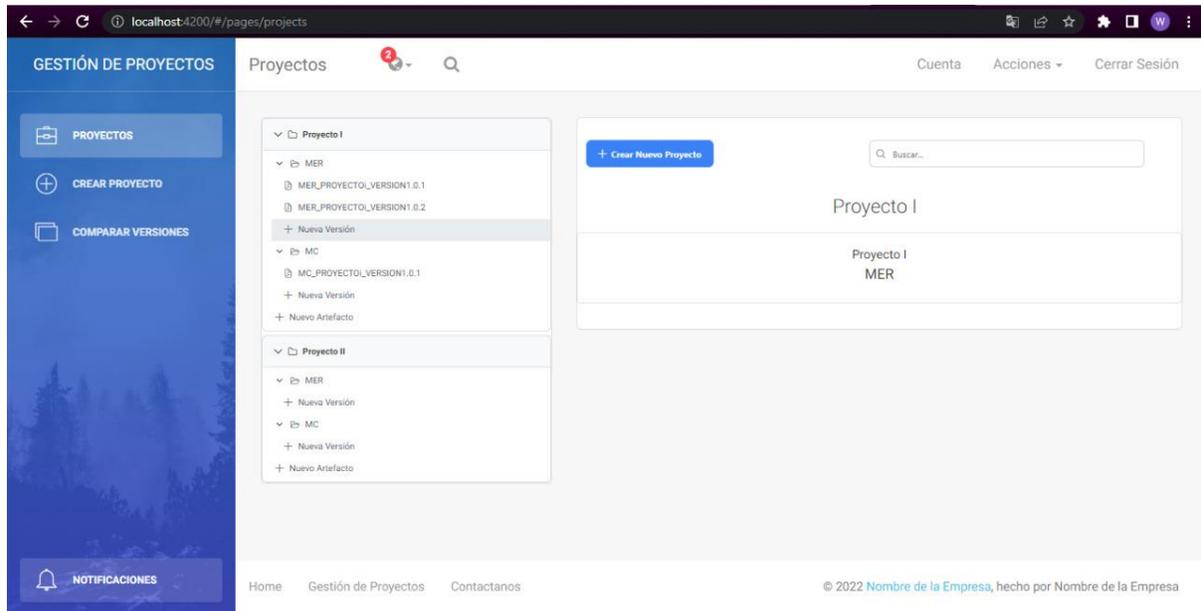
Para la automatización del marco de trabajo de gestión de configuración de software se ha desarrollado un prototipo de software, el cual verifica las actividades de validación de los artefactos modificados y que serán incluidos y actualizados en las líneas base.

A continuación, se adjunta las pantallas del prototipo elaborado:

Gestionar los proyectos Software

Figura 7

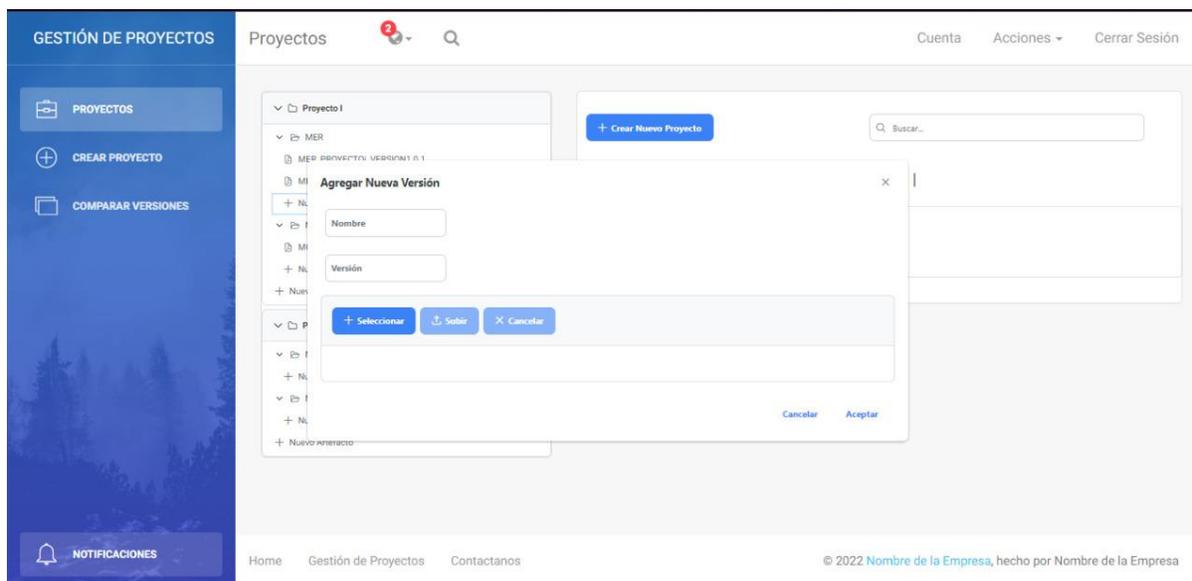
Gestionar los proyectos de Software



Nota. En el gráfico se puede observar la interfaz principal del prototipo web en donde se plasma el Marco de trabajo elaborado.

Figura 8

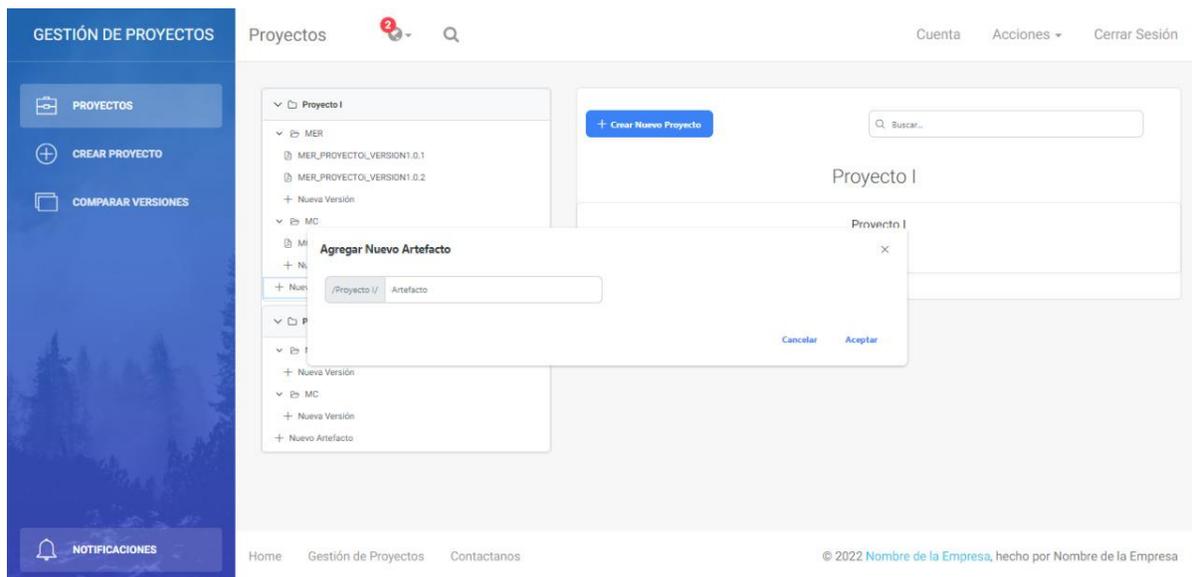
Agregar nueva versión a un artefacto



Nota. En el gráfico se puede observar la opción de generar una nueva versión a un artefacto

Figura 9

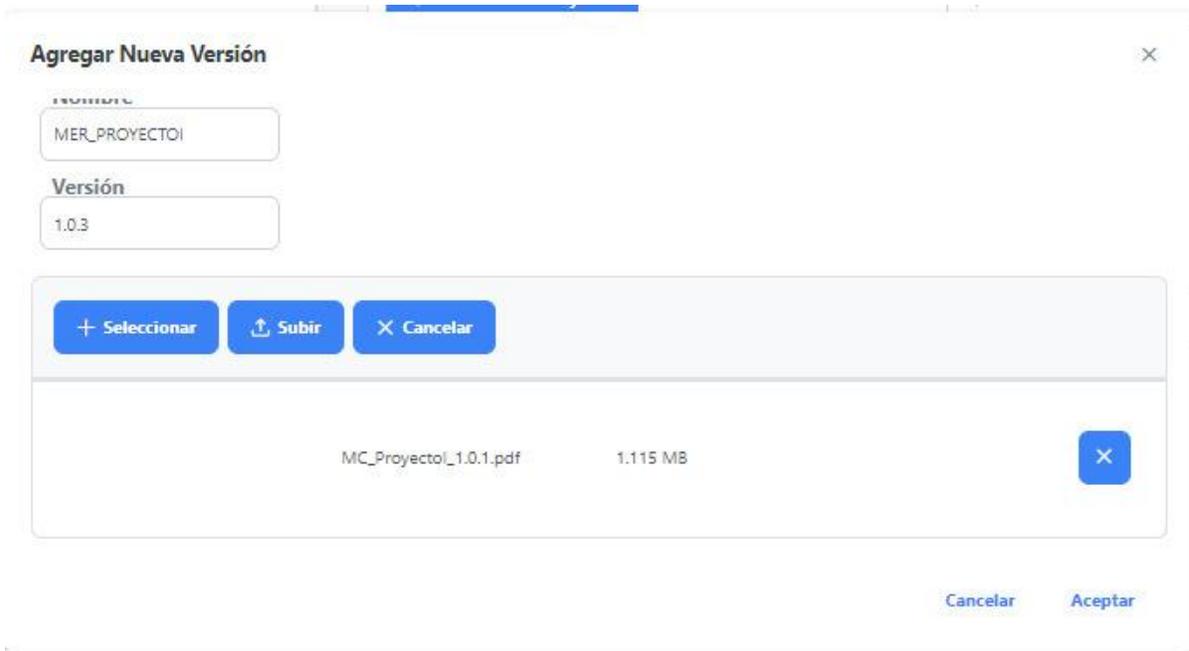
Agregar nuevo artefacto a un proyecto



Nota. En el gráfico se puede observar la opción de agregar un nuevo artefacto al proyecto

Figura 10

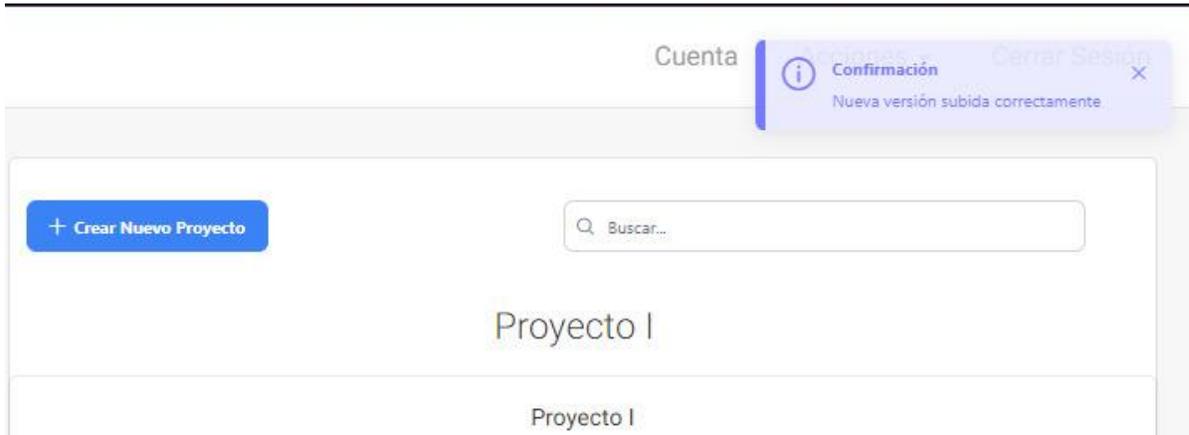
Carga de la nueva versión del artefacto



Nota. En el gráfico se puede observar la carga de una nueva versión de un artefacto

Figura 11

Mensaje de Confirmación



Nota. En el gráfico se puede observar el mensaje exitoso de carga

Figura 12

Formulario para un nuevo proyecto

GESTIÓN DE PROYECTOS Crear Proyecto 2

Cuenta Acciones Cerrar Sesión

PROYECTOS

CREAR PROYECTO

COMPARAR VERSIONES

NOTIFICACIONES

Home Gestión de Proyectos Contactanos © 2022 Nombre de la Empresa, hecho por Nombre de la Empresa

Recomendaciones

- Establecer objetivos específicos
- Planificación del alcance
- Comprender el propósito del proyecto
- Hacer un cronograma del proyecto
- Visión del presupuesto necesario
- Reuniones periódicas con el equipo
- especificar las fortalezas y debilidades del equipo
- Automatizar algunas tareas
- Evaluar el proyecto

Crear Nuevo Proyecto

TÍTULO DEL PROYECTO

DESCRIPCIÓN

FECHA DE INICIO

ARTEFACTO 1

ARTEFACTO 2

✓ Crear Proyecto ✕ Cancelar

Nota. En el gráfico se puede observar el formulario para la creación de un nuevo proyecto

Figura 13

Comparación de versiones

GESTIÓN DE PROYECTOS Comparar Versiones 2

Cuenta Acciones Cerrar Sesión

PROYECTO I
MER

Seleccionar versión

MER_PROYECTO_LVERSION_1.0.2 ✕

pdf

MER_PROYECTO_LVERSION_1.0.2

Seleccionar versión

MER_PROYECTO_LVERSION_1.0.3 ✕

pdf

MER_PROYECTO_LVERSION_1.0.3

REALIZAR COMPARACIÓN

Completado. Los documentos tienen un 89.13% de similitud.

Home Gestión de Proyectos Contactanos © 2022 Nombre de la Empresa, hecho por Nombre de la Empresa

Nota. En el gráfico se puede observar la comparación de 2 archivos

Capítulo IV

Experimentación

En este capítulo se presenta el trabajo de experimentación realizado con el fin de validar las hipótesis de trabajo planteadas en el capítulo 1 de esta tesis. La contrastación de las hipótesis se realizará mediante técnicas de diseño de experimentos para la ingeniería de software. Es por ello que, en la siguiente sección, se presenta el método seguido para su posterior aplicación. Posteriormente, se desarrollan los casos de estudio que permitan validar las hipótesis, para generar una línea base.

Se procederá a aplicar la metodología definida siguiendo cada una de sus actividades y tareas propuestas, para recopilar la información generada.

Al final, se analizará toda la información obtenida comparando con la información de la línea base inicial y se valorará la utilidad del modelo conceptual definido, la metodología utilizada y demás artefactos elaborados.

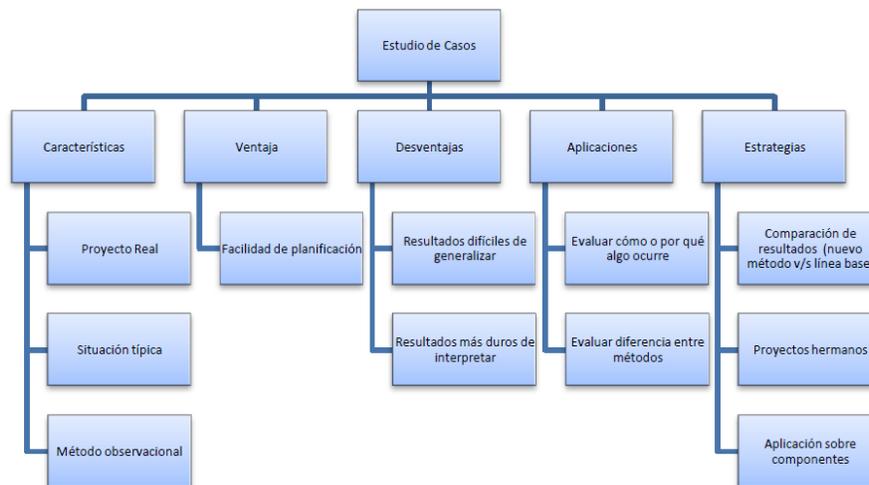
Método de diseño de experimentos de ingeniería de software

El único modo para contrastar las creencias y las opiniones para convertirlas en hechos es la experimentación. El fin de la experimentación es identificar las causas por las que se producen determinados resultados. Según (Wohlin, Runeson, Höst, Ohlsson, & Reg, 2000), la ingeniería de software (IS) debe tratarse como una ciencia y considera la experimentación como una parte importante del conocimiento científico (Juristo & Moreno, 2001) y una de las formas de validación de las propuestas. La IS hará posible la comprensión e identificación de las variables que entran en juego en la construcción de software y las conexiones que existen entre ellas. Las estrategias empíricas existentes para el desarrollo de experimentos en ingeniería de software incluyen encuestas, estudios de caso y experimentos. (Wohlin, Höst, & Henningsson, 2003)

La figura 13 muestra las características de los estudios de caso como método de investigación según se propone en (Wohlin, Höst, & Henningsson, 2003).

Figura 14

Características de los estudios de caso. (Fuente: Wohlin, Höst, & Henningsson, 2003)



Nota. En el gráfico se puede observar las características de los estudios de caso. Tomado de (Fuente: Wohlin, Höst, & Henningsson, 2003)

Según estos autores, hay tres formas de plantear los estudios de casos:

- Comparar los resultados entre una línea base y la nueva propuesta.
- Desarrollar dos proyectos simultáneamente tomando uno de ellos como base.
- Aplicar la nueva propuesta sobre unos determinados componentes y comparar los resultados obtenidos con los de aquellos componentes sobre los que no se haya aplicado la nueva propuesta.

En esta tesis se ha optado por la primera forma. Para establecer la línea base (inicial) se ha procedido a obtener la matriz de riesgos según la metodología definida actualmente en la organización. Después de implantar el modelo conceptual y el marco de trabajo definido, los resultados obtenidos constituirán la línea base (final) frente a que comparar la nueva propuesta.

Después de definir la línea base inicial se ha implementado el marco de trabajo para la gestión de configuración de software definido en el capítulo 4, tomando como referencia el modelo conceptual y ejecutando las actividades y tareas propuestas. Luego de la ejecución de las actividades y tareas definidas, se han identificado las diferencias entre el caso anterior y luego implementar el marco de trabajo de gestión de configuración de software.

Caso de estudio

En la actualidad el control de versiones sobre artefactos manejados en documentos es completamente manual, la persona que realiza la actualización, edición o cambios debe contemplar la lectura, modificación de la información y el versionamiento se lo coloca al final del documento, en el cual se ingresa los siguientes datos: versión, responsable, revisado por, aprobado por. La edición de un documento significa sobre escribir un fichero, una vez sobre escrito, por supuesto, no hay camino de vuelta para recuperar la versión anterior a los cambios realizados.

Este proceso tiende a fallar, ya que, está expuesto a que el encargado de la revisión del artefacto cometa errores, se entregue al cliente y este al momento de realizar las pruebas se tope con las novedades de que existe fallos, lo que conlleva a notificar de nuevo a la empresa INNOVASYS SA, implicando pérdida de tiempo en revisión de la programación y documentación, si se maneja políticas puede existir sanciones económicas, la desestimación, etc.

Si se requiere acceder a una versión anterior del documento se revisa la información del pie en donde se indica la persona que realizó la modificación, se evidencia que la persona que realizó el cambio simplemente coloca sus iniciales por lo que se vuelve aún más complicado encontrar a la persona y solicitarle la versión anterior, en el mejor de los casos, en el peor de los casos tal vez la persona ya no se encuentre trabajando en la empresa.

Línea base inicial de los activos de la organización

Se tomó como línea base el proyecto del Banco Agrario de Colombia, para el cual se elaboraron las pruebas unitarias según el proceso definido y se recuperó la siguiente información:

Evidencia de ejecución de Pruebas Unitarias

Tabla 14

Prueba unitaria

Datos generales	
PROYECTO	Banco Agrario de Colombia
HISTORIA DE USUARIO	COB-S655359-ACT-UNIDAD FUNCIONAL Prórrogas FWPRORROGA
PREPARADO POR	Paul Moreno
FECHA ACTUAL	10/06/2022
FECHA ÚLTIMO INFORME	

Nota. En la tabla se observa la descripción de la prueba unitaria

Evidencias de las Pruebas

Pantalla o método a probar. Se crea un formulario compuesto para llamar al formulario simple “Solicitud de Prórrogas”, en la siguiente tabla:

Tabla 15

Objetivo de prueba

Datos generales	
OBJETIVO DE LA	Configurar en Workflow la Unidad Funcional Prórrogas ICR

 Datos generales

PRUEBA

EJECUTADO POR Paul Moreno

OBSERVACIONES Pruebas Realizadas en ambiente de Desarrollo

Nota. En la tabla se observa el objetivo de la prueba unitaria

Figura 15*Desarrollo de formulario compuesto*

ICR REQUEST / FUNCTIONALITIES / ICRExtension

....

Datos del Trámite

Tipo Documento	Número de Documento	Nombre Cliente
<input type="text"/>	<input type="text"/>	<input type="text"/>
N° de Trámite ICR	N° ICR Finagro	N° Operación Activa
<input type="text"/>	<input type="text"/>	<input type="text"/>
Llave de Redescuento	Fecha Vencimiento	Estado
<input type="text"/>	<input type="text"/>	<input type="text"/>

[+ New](#)

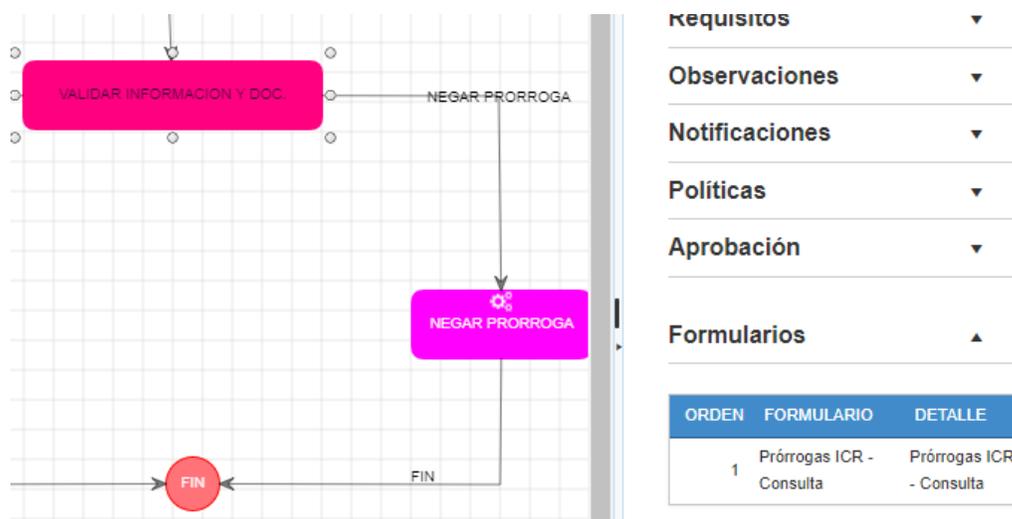
Nro. Solicitud	Tipo Prórroga	Estado	Fecha Ingreso	Fecha Venc...	Prórroga	Ampliación	Ola Invernal	
<input type="text"/>	Detalle							

Nota. En el gráfico se puede observar el desarrollo de un formulario compuesto

Se registra la nueva pantalla en un componente “Prórrogas ICR – Consulta” y al momento de configurar la actividad en workflow se muestra el nombre del nuevo formulario, en la siguiente figura:

Figura 16

Configuración del formulario en la actividad



Nota. En el gráfico se puede observar la configuración del formulario en la actividad

Se debe crear una nueva solicitud de Prórroga en la siguiente figura:

Figura 17

Solicitud de Prórroga 288401 creada

Inicio / Solicitud de Prórrogas

Datos del Trámite

Tipo Documento CEDULA DE CIUDADANIA	Número de Documento 2764144493	Nombre Cliente nombreLargo 2764144493
N° de Trámite ICR 93	N° ICR Finagro 14-29418	N° Operación Activa 725013010168242
Llave de Redescuento 1450233836	Fecha Vencimiento 24/08/2015	Estado ELEGIBLE AUTORIZADO

+ Nuevo Exportar: [X]

NRO. SOLICITUD	TIPO PRÓRRO...	ESTADO	FECHA INGR...	FECHA VENG...	PRÓRROGA	AMPLIACIÓN	OLA INVERNAL	
288401	PRO	TRA	24/02/2022	23/08/2015	T	N	N	Detalle

Elementos mostrados 1 - 1 de 1

Imprimir

Nota. En el gráfico se puede observar la creación de la solicitud de Prórroga 288401 creada

Se muestra la nueva solicitud en el buzón de tareas del oficial, en la siguiente figura:

Figura 18

Solicitud en buzón de tareas del oficial

Cliente: nombreLargo 2764144493 - 5532318 A tiempo
SOLPRO.1301.22.000005-SOLICITUD PRORROGA-VALIDAR INFORMACION Y DOC.
 Enviado el 10/06/2022 08:47:39, Asignado a ALBARRACIN ROA ORLANDO

Nota. En el gráfico se puede observar la tarea asignada al oficial

Se debe crear una nueva solicitud de Prórroga en la siguiente figura:

Figura 19

UF Prórrogas I en buzón de tareas del oficial

SOLPRO.1301.22.000005 SOLICITUD PRORROGA (Version 3) / VALIDAR INFORMACION Y DOC. Observaciones Requisitos Acciones-

Datos del Trámite

Tipo Documento	Número de Documento	Nombre Cliente
CEDULA DE CIUDADANIA	2764144493	nombreLargo 2764144493
N° de Trámite ICR	N° ICR Finagro	N° Operación Activa
93	14-29418	725013010168242
Llave de Redescuento	Fecha Vencimiento	Estado
1450233836	08/24/2015	ELEGIBLE AUTORIZADO

Exportar:

NRO. SOLICITUD	TIPO PRORROGA	ESTADO	FECHA INGRESO	FECHA VENCIMIE...	PRORROGA	AMPLIACIÓN	OLA INVERNAL	
288401	PRORROGA	EN TRAMITE	02/24/2022	08/23/2015	EN CURSO	NO TIENE	NO TIENE	Detalle

Elementos mostrados 1 - 1 de 1

Nota. En el gráfico se puede observar la consulta de datos de la prórroga creada

Conclusiones

Los siguientes puntos resumen las conclusiones de la línea base del caso de estudio:

- En las pruebas del software siempre existe la posibilidad de generar prórrogas por la cantidad de errores que se describen en el aplicativo, de aquí se demuestra que los errores son continuos.
- Los errores son por la mala gestión de los artefactos del desarrollo del aplicativo de software.

Resultado del presente análisis, las prórrogas son una constante en todos por la falta de

una adecuada gestión de la configuración.

Resumen de la línea base inicial

En esta subsección se han plasmado las actividades realizadas cuando se realiza el control de la calidad de software, y se puede observar lo siguiente:

- Excesivos errores por falta de una gestión adecuada de líneas base y control de versiones.
- Pérdida de información y volver rehacer los artefactos.
- No se puede dar continuidad, ni mantenimiento de los aplicativos.

Como conclusión podemos indicar que la gestión de configuración no es adecuada en la empresa INNOVASYS SA, generando retraso en los proyectos y mala gestión de versiones por la falta de un adecuado marco de trabajo para la gestión de configuración de software.

Actividades del marco de trabajo para la Gestión Configuración de Software

En esta sección se describe como se ha implementado el marco de trabajo propuesto para la gestión de configuración de software, que es parte de la resolución del problema, en la empresa objeto del estudio experimental, que en este caso particular es INNOVASYS SA.

Para poder aplicar el marco de trabajo referencial definido, y poder comparar con la línea base elaborada, se ha tomado como ejemplo particular de un proyecto de software el cual lo vamos aplicar el Marco de Trabajo.

El proceso desarrollado para la gestión de configuración está compuesto por tres subprocesos:

- Creación o mejora de la CMDB.
- Registro y/o actualización de los elementos de configuración.
- Monitoreo de la Configuración.

De estos tres procesos para aplicar el Marco de trabajo elaborado se ha tomado el subproceso “Registro y/o actualización de los elementos de configuración”, pues este

subproceso, permite identificar el gran problema de la gestión de los elementos generados en el proceso de desarrollo de software.

A continuación, se describen las actividades a desarrollar en el marco de trabajo para la gestión de configuración de software descrito en el capítulo 4. También, se indica como se ha realizado la implementación de las diferentes actividades definidas en el marco de trabajo.

Tabla 16

Actividades del Marco de Trabajo de GC

Detalle de actividades				
1	Identificar los cambios de los elementos de configuración.	Determinar los cambios que se han realizado en los elementos de configuración, ya sea a nivel de servicio o infraestructura tecnológica.	Analista de la Configuración	Correo electrónico o memorando o herramientas de gestión
	Es un nuevo elemento de configuración?	<p>Respuesta Afirmativa: Continúa con la actividad 3.</p> <p>Respuesta Negativa: Continua con la siguiente actividad 2.</p>		N/A
2	Actualizar el estado actual del elemento de configuración.	Consiste en cambiar a Inactivo el estado del CI, con la finalidad de	Analista de la Configuración	N/A

 Detalle de actividades

		mantener un histórico de los cambios realizados en la CMDB.		
3	Registrar elemento de configuración y notificar la acción realizada.	Se debe registrar el elemento de configuración en la CMDB para que la misma se encuentre actualizada y a continuación se notifica la/las acciones/es realizadas al responsable de la Configuración.	Analista(s) de la Configuración	CMDB actualizada Correo electrónico o memorando o herramienta de gestión
4	Revisar la actualización de CI en CMDB	Revisar que la información del cambio realizado al elemento de configuración se encuentre registrado en la CMDB, en base a la información proporcionado por el Analista de la Configuración	Responsable de la Configuración	Correo electrónico o memorando o herramienta de gestión
	¿Actualización	Respuesta Afirmativa: El		N/A

Detalle de actividades			
	correcta?	subproceso finaliza. Respuesta Negativa: Continúa con la siguiente actividad 5.	
5	Notificar novedades encontradas	Para los casos en los cuales se han encontrado novedades en la actualización de los elementos de configuración en la CMDB, se informa las novedades encontradas durante la verificación por el mismo medio que se recibió la información.	Responsable de la Configuración N/A

Nota. En la tabla se observa el Actividades del Marco de Trabajo de GC

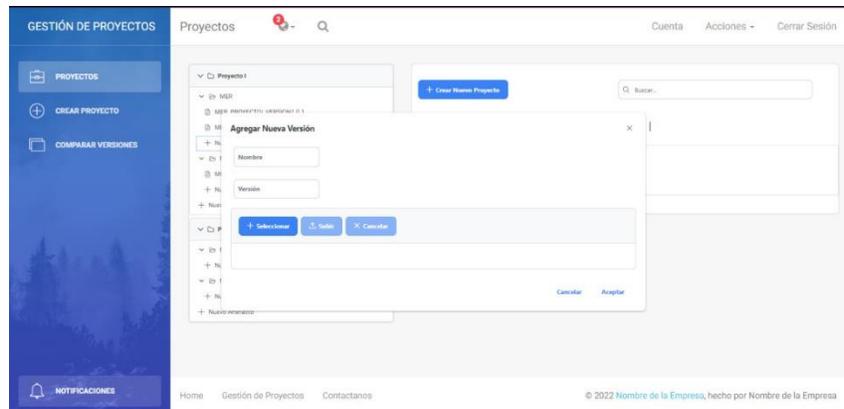
Identificar los cambios de los elementos de configuración

En esta actividad se determinan los cambios que se han realizado en los elementos de configuración, ya sea a nivel de servicio o infraestructura tecnológica, como los criterios de valoración con los que se va a realizar la experimentación. En este caso particular, se ha determinado lo siguiente:

- Al Analista de la Configuración recibe por la herramienta de gestión de configuración el pedido de revisión del elemento (artefacto de configuración, para verificar su contenido).

Figura 20

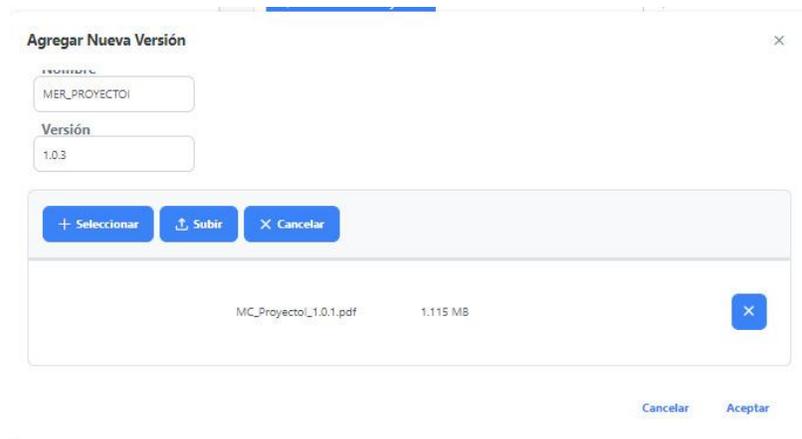
Ingreso de una nueva versión de un elemento de configuración



Nota. En el gráfico se puede observar el ingreso de una nueva versión de un elemento de configuración

Figura 21

Ingreso del documento de una nueva versión de un elemento de configuración



Nota. En el gráfico se puede observar el ingreso del documento de una nueva versión de un elemento de configuración.

- Se identifica si es un nuevo elemento de configuración.

- En caso de ser positivo se ejecuta la actividad 3.
- Caso contrario se ejecuta la actividad 2.

Actividad 2: Actualizar el estado actual del elemento de configuración

Esta actividad consiste en cambiar a **Inactivo** el estado del CI, con la finalidad de mantener un histórico de los cambios realizados en la CMDB.

- El analista de configuración mantiene actualizado al elemento de configuración.

Actividad 3: Registrar elemento de configuración y notificar la acción realizada

En esta actividad se debe registrar el elemento de configuración en la CMDB para que la misma se encuentre actualizada y a continuación se notifica la/las acciones/, al responsable de la Configuración.

- El responsable de la configuración registra el nuevo elemento de la configuración.

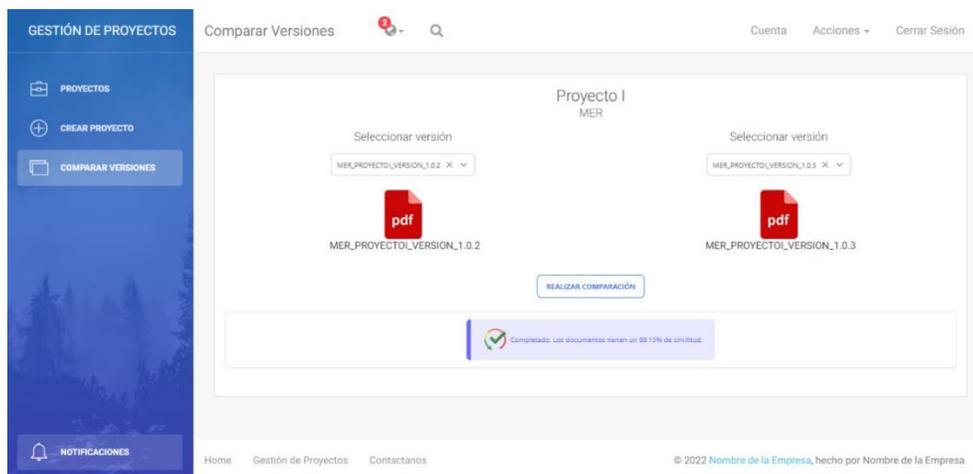
Actividad 4: Revisar la actualización del CI en CMDB

En esta actividad se revisa que la información del cambio realizado al elemento de configuración se encuentre registrado en la CMDB, en base a la información proporcionado por el analista de configuración.

- El responsable de la Configuración revisa que el elemento de la configuración, es decir el artefacto elaborado no solo se encuentre registrado en la CMDB, sino también, se verifique que el documento actual tenga los cambios realizados al documento originalmente enviado, para ello realiza la verificación en el sistema de Gestión de configuración de software (Prototipo), como se puede ver en el grafico 22

Figura 22

Verificación del Artefacto anterior y la nueva versión



Nota. En el gráfico se puede observar la Verificación del Artefacto anterior y la nueva versión

Se verifica que la información este correcta:

- En caso de ser positivo se termina el subprocesso.
- En caso contrario de envía a la actividad 5.

Actividad 5: Notificar Novedades encontradas

En esta actividad, para los casos en los cuales se han encontrado novedades en la actualización de los elementos de configuración en la CMDB, se informa las novedades encontradas de la verificación por el sistema de gestión de configuración.

Comparación resultados línea base inicial y marco de trabajo de Gestión de Configuración

En la comparación se analiza la gestión de configuración como se estaba realizando en lo que llamamos línea base y luego de aplicar el marco de trabajo de Gestión de Configuración planteado, para poder identificar los puntos fuertes y débiles de la identificación y evaluación en la Gestión de Configuración, asociación de los elementos, y resultados obtenidos.

La revisión de los resultados entre las líneas base inicial y final se describen a continuación:

- Los elementos de gestión de la configuración han permitido que no exista la posibilidad de cambiar y realizar modificaciones previas por la buena gestión.
- Incidencias de cambios en línea base son efectivos con el nuevo modelo de gestión.
- Los artefactos utilizados son los mínimos necesarios, pues permiten la trazabilidad del software desarrollado.
- Tiempo que se utiliza en modificar los artefactos que son los elementos de configuración fueron mínimos por la buena gestión de los mismos.
- Las utilizaciones adecuadas de los recursos humanos se ven evidenciados con el nuevo marco de gestión.

Conclusiones

En este capítulo se han presentado los requisitos y el proceso del prototipo elaborado, como una alternativa para la gestión de configuración para las organizaciones.

Un elemento importante que identifica a este prototipo es la gestión de los documentos con la comparativa entre el documento original y el nuevo que se generan en el proceso de desarrollo de software dentro de la organización.

El prototipo se ha desarrollado en un framework de desarrollo de software muy fácil de implementar, de gran escalabilidad en un entorno web y fácil de dar mantenimiento, también se ha buscado que fuera muy sencillo en su uso. Se contemplan desarrollar versiones periódicas para mejorar y mantener la aplicación

Validación de las hipótesis de trabajo

Un marco de trabajo no es una herramienta que asegure el éxito de aquella actividad a la que se aplique, sino una guía de ayuda para mejorar los resultados y reducir la dependencia del criterio experto de las personas. El marco de trabajo propuesto en esta tesis no pretende resolver todos los problemas a los que se enfrenta la organización para la gestión de configuración, sino guiarle en las actividades oportunas para la gestión de ellos artefactos que se

utilizan en la gestión de configuración en el desarrollo de un aplicativo de software. Si se logra este cometido, es factible mencionar que el marco de trabajo es aplicable y ha cumplido las expectativas planteadas. Y verificar, si las hipótesis que se plantearon han sido validadas.

Después de haber realizado las evaluaciones de las líneas base inicial y final (marco de trabajo), y comparar los resultados obtenidos, se puede concluir que se han cumplido los objetivos esperados y se pueden validar las hipótesis del trabajo.

- *“Si se elabora un marco de trabajo se optimiza la cadena de versiones, el control de configuraciones y el estado de cualquier artefacto de software con contenidos similares almacenados en repositorios locales”*. Esta hipótesis queda validada con la gestión de los artefactos que son generados en el desarrollo del aplicativo de software, pues permite una trazabilidad de los cambios que ha sufrido el documento para evitar información duplicada y adecuada gestión de versionamiento de los artefactos.

Resumen del capítulo IV

En este capítulo se ha desarrollado el caso de estudio con la finalidad de validar el marco de trabajo elaborado, sus actividades y tareas descritas. Para ello, se ha definido hacerlo en la empresa INNOVASYS SA, que es una empresa estatal ubicada en la ciudad de Quito Ecuador, la cual se encarga de desarrollar aplicaciones de software a medida.

Además, se ha realizado la validación de las hipótesis planteadas en el Capítulo 3.

Los principales resultados que podemos destacar de este capítulo son los siguientes:

- Se ha desarrollado la línea base inicial del caso de estudio, siguiendo el procedimiento establecido por la empresa INNOVASYS SA, el cual toma como referencia una metodología propia de gestión de configuración de software.

- Se ha implementado el marco de trabajo propia elaborada en los artefactos generados en el desarrollo de aplicaciones de software (línea base final), siguiendo las actividades y tareas definidas para la gestión de configuración de software.
- Con los resultados obtenidos de la line base inicial y final (implementación del marco de trabajo) en el desarrollo de un aplicativo de software, se ha realizado la comparación de los resultados obtenidos, y se han descrito las diferencias encontradas.
- Se han validado las hipótesis planteadas en el capítulo 3, con la descripción de las actividades realizadas y los resultados obtenidos.
- Como conclusión, podemos indicar que la comparación entre las líneas base inicial y final (implementación del marco de trabajo) ha permitido identificar la importancia de una adecuada gestión de la gestión de configuración de software y específicamente en la gestión de las versiones de los artefactos de forma adecuada.

Bibliografía

- ✓ ADDIN Mendeley Bibliography CSL_BIBLIOGRAPHY 71504, U. (2008). Metodología de análisis y gestión de riesgos para los sistemas de información.
- ✓ Adler, R. M. (2013). A dynamic capability maturity model for improving cyber security. 2013 IEEE International Conference on Technologies for Homeland Security (HST), 230–235. <https://doi.org/10.1109/THS.2013.6699005>
- ✓ Barclay, C. (2014). Sustainable security advantage in a changing environment: The cybersecurity capability maturity model (CM2). Proceedings of the 2014 ITU Kaleidoscope Academic Conference: Living in a Converged World - Impossible Without Standards?, K 2014, 275–282. <https://doi.org/10.1109/Kaleidoscope.2014.6858466>
- ✓ Bartusevics, A., & Novickis, L. (2015a). Model-based Approach for Implementation of Software Configuration Management Process. 5.
- ✓ Bartusevics, A., & Novickis, L. (2015b). Models for implementation of software configuration management. *Procedia Computer Science*, 43(C), 3–10. <https://doi.org/10.1016/j.procs.2014.12.002>
- ✓ Bazaz, A., & Arthur, J. D. (2007). Towards a taxonomy of vulnerabilities. Proceedings of the Annual Hawaii International Conference on System Sciences, c, 1–10. <https://doi.org/10.1109/HICSS.2007.566>
- ✓ Bazaz, A., Arthur, J. D., & Tront, J. G. (2006). Modeling security vulnerabilities: A constraints and assumptions perspective. Proceedings - 2nd IEEE International Symposium on Dependable, Autonomic and Secure Computing, DASC 2006, c, 95–102. <https://doi.org/10.1109/DASC.2006.35>
- ✓ Bergmann, F. B., & Silveira, M. S. (2014). “My life doesn’t have to be an open book”: A model to help designers to enhance privacy controls on social network sites. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8533 LNCS, 387–399. <https://doi.org/10.1007/978->

3-319-07620-1_34

- ✓ Berkley Research Group. (2016). Estudio comparativo de los niveles de madurez en ciberseguridad. 1–45.
- ✓ Bishop, M., Miloslavskaya, N., & Theocharidou, M. (2015). Information Security Education Across the Curriculum: 9th IFIP WG 11.8 World Conference, WISE9 Hamburg, Germany, May 26-28, 2015 Proceedings. *IFIP Advances in Information and Communication Technology*, 453, 53–63. <https://doi.org/10.1007/978-3-319-18500-2>
- ✓ Bompard, E., Huang, T., Wu, Y., & Cremenescu, M. (2013). Classification and trend analysis of threats origins to the security of power systems. *International Journal of Electrical Power and Energy Systems*, 50(1), 50–64.
<https://doi.org/10.1016/j.ijepes.2013.02.008>
- ✓ Cebula, J. J., & Young, L. R. (2010). A Taxonomy of Operational Cyber Security Risks. Carnegie-Mellon Univ Pittsburgh Pa Software Engineering Inst, December, 1–47.
<https://doi.org/10.1007/978-1-4419-7133-3>
- ✓ Cederbom, M. (2015). Institutionen för datavetenskap Improving software configuration management across by Martin Cederbom.
- ✓ Chou, I. H. (2011). Secure Software Configuration Management Processes for nuclear safety software development environment. *Annals of Nuclear Energy*, 38(10), 2174–2179. <https://doi.org/10.1016/j.anucene.2011.06.016>
- ✓ Christopher, J. D. (2014). Cybersecurity Capability Maturity Model (C2M2). Department of Homeland Security, February. <https://energy.gov/oe/cybersecurity-critical-energy-infrastructure/cybersecurity-capability-maturity-model-c2m2-program>
- ✓ CMU. (2003). System Security Engineering Capability Maturity Model (SSE-CMM). Proceedings of the 19th International Conference on Software Engineering ICSE 97, 35, 566–567. <https://doi.org/10.1145/253228.253454>
- ✓ Cohen, F., Lowrie, J., & Preston, C. (n.d.). Chapter 1 THE STATE OF THE SCIENCE

OF DIGITAL EVIDENCE EXAMINATION. Ifip International Federation For Information Processing, 3–21.

- ✓ Delmee, F. (2016). the Structure of a Cyber Risk a Scenario Based Approach in Cyber Risk.
- ✓ Department of Energy, U. (2019). CYBERSECURITY CAPABILITY MATURITY MODEL (C2M2). June.
- ✓ Donaldson, S. E., Siegel, S. G., Williams, C. K., & Abdul, A. (2015). Enterprise Cybersecurity. Apress.
- ✓ Durrani, U., Pita, Z., Richardson, J., & Lenarcic, J. (2014). Association for Information Systems AIS Electronic Library (AISeL) AN EMPIRICAL STUDY OF LEAN AND AGILE INFLUENCES IN SOFTWARE CONFIGURATION MANAGEMENT.
<http://aisel.aisnet.org/pacis2014%0Ahttp://aisel.aisnet.org/pacis2014/320>
- ✓ Elnagdy, S. A., Qiu, M., & Gai, K. (2016). Understanding Taxonomy of Cyber Risks for Cybersecurity Insurance of Financial Industry in Cloud Computing. Proceedings - 3rd IEEE International Conference on Cyber Security and Cloud Computing, CSCloud 2016 and 2nd IEEE International Conference of Scalable and Smart Cloud, SSC 2016, 295–300. <https://doi.org/10.1109/CSCloud.2016.46>
- ✓ Fahmy, S., Deraman, A., & Yahaya, J. H. (2018). The role of human in software configuration management. ACM International Conference Proceeding Series, 56–60. <https://doi.org/10.1145/3185089.3185117>
- ✓ Fahmy, S., Deraman, A., Yahaya, J., Nasir, A., & Shamsudin, N. (2020). The evolution of software configuration management. International Journal of Advanced Trends in Computer Science and Engineering, 9(1 Special Issue 3), 50–63. <https://doi.org/10.30534/ijatcse/2020/0891.32020>
- ✓ Farooq, A., Kakakhel, S. R. U., Virtanen, S., & Isoaho, J. (2016). A taxonomy of perceived information security and privacy threats among IT security students. 2015

- 10th International Conference for Internet Technology and Secured Transactions, ICITST 2015, 280–286. <https://doi.org/10.1109/ICITST.2015.7412106>
- ✓ Goksen, Y., Cevik, E., & Avunduk, H. (2015). A Case Analysis on the Focus on the Maturity Models and Information Technologies. *Procedia Economics and Finance*, 19(15), 208–216. [https://doi.org/10.1016/S2212-5671\(15\)00022-2](https://doi.org/10.1016/S2212-5671(15)00022-2)
 - ✓ Greitzer, F. L., Strozer, J. R., Cohen, S., Moore, A. P., Mundie, D., & Cowley, J. (2014). Analysis of unintentional insider threats deriving from social engineering exploits. *Proceedings - IEEE Symposium on Security and Privacy*, 2014-Janua, 236–250. <https://doi.org/10.1109/SPW.2014.39>
 - ✓ Gupta, M., Rees, J., Chaturvedi, A., & Chi, J. (2006). Matching information security vulnerabilities to organizational security profiles: A genetic algorithm approach. *Decision Support Systems*, 41(3), 592–603. <https://doi.org/10.1016/j.dss.2004.06.004>
 - ✓ Halvorsen, C. P., & Conradi, R. (2001). A Taxonomy to Compare SPI Frameworks. *Software Process Technology*, 2077(0212), 217–235. https://doi.org/10.1007/3-540-45752-6_17
 - ✓ Herzfeldt, A., Hausen, M., Briggs, R. O., & Krcmar, H. (2012). Developing a Risk Management Process and Risk Taxonomy for Medium-Sized It Solution Providers. *Ecis*, 2012, Paper 165.
 - ✓ Hohan, A. I., Olaru, M., & Pirnea, I. C. (2015). Assessment and Continuous Improvement of Information Security Based on TQM and Business Excellence Principles. *Procedia Economics and Finance*, 32(15), 352–359. [https://doi.org/10.1016/S2212-5671\(15\)01404-5](https://doi.org/10.1016/S2212-5671(15)01404-5)
 - ✓ ISACA. (2016). *Glosary* (2016). <https://www.isaca.org/Pages/Glossary.aspx?tid=1784&char=R>
 - ✓ Jendraszyk, D. (2021). Requirements Analysis of Approvable Software for a GLASS

Ground Station according to DO-278A.

- ✓ Kamesh, & Sakthi Priya, N. (2014). Security enhancement of authenticated RFID generation. *International Journal of Applied Engineering Research*, 9(22), 5968–5974.
<https://doi.org/10.1002/sec>
- ✓ Kishor S. Trivedi, Dong Seong Kim, A. R. (2004). Dependability and Security. 1–10.
- ✓ Kitchenham, B. A., & Jorgensen, T. D. M. (2004). Evidence-based Software Engineering. *Proceedings of the 26th International Conference on Software Engineering*.
<https://doi.org/10.1109/ICSE.2004.1317449>
- ✓ Kitchenham, B., & Charters, S. (2007). Guidelines for performing Systematic Literature reviews in Software Engineering Version 2.3. *Engineering*, 45(4ve), 1051.
<https://doi.org/10.1145/1134285.1134500>
- ✓ Lai, Y. P., & Hsia, P. L. (2007). Using the vulnerability information of computer systems to improve the network security. *Computer Communications*, 30(9), 2032–2047.
<https://doi.org/10.1016/j.comcom.2007.03.007>
- ✓ Laribbe, L. (2006). *NAVAL POSTGRADUATE* by (Issue December). Naval Postgraduate School.
- ✓ Le, N. T. (2016). Can maturity models support cyber security?
- ✓ Lee, S., Chung, T., & Choi, M. (2006). An empirical study of quality and cost based security engineering. *Information Security Practice and Experience*, 379–389.
http://link.springer.com/chapter/10.1007/11689522_35
- ✓ Li, Z., Shahidehpour, M., and Aminifar, F. (2017). Cybersecurity in Distributed Power Systems. *Proceedings of the IEEE*, PP(99), 1–22.
<https://doi.org/10.1109/JPROC.2017.2687865>
- ✓ Li, Xiaodan, Chang, X., Board, J. A., & Trivedi, K. S. (2017). A novel approach for software vulnerability classification. *Proceedings - Annual Reliability and Maintainability Symposium*. <https://doi.org/10.1109/RAM.2017.7889792>

- ✓ Li, Xu, Liang, X., Lu, R., Shen, X., Lin, X., & Zhu, H. (2012). Securing smart grid: Cyber attacks, countermeasures, and challenges. *IEEE Communications Magazine*, 50(8), 38–45. <https://doi.org/10.1109/MCOM.2012.6257525>
- ✓ Linares-Vasquez, M., Bavota, G., & Escobar-Velasquez, C. (2017). An empirical study on android-related vulnerabilities. *IEEE International Working Conference on Mining Software Repositories*, i, 2–13. <https://doi.org/10.1109/MSR.2017.60>
- ✓ Liu, L., De Vel, O., Han, Q.-L., Zhang, J., & Xiang, Y. (2018). Detecting and Preventing Cyber Insider Threats: A Survey. *IEEE Communications Surveys & Tutorials*, c, 1–1. <https://doi.org/10.1109/COMST.2018.2800740>
- ✓ Miloslavskaya, N., Tolstoy, A., & Zapechnikov, S. (2016a). Taxonomy for unsecure big data processing in security operations centers. *Proceedings - 2016 4th International Conference on Future Internet of Things and Cloud Workshops, W-FiCloud 2016*, 154–159. <https://doi.org/10.1109/W-FiCloud.2016.42>
- ✓ Miloslavskaya, N., Tolstoy, A., & Zapechnikov, S. (2016b). Taxonomy for unsecure digital information processing. *2016 3rd International Conference on Digital Information Processing, Data Mining, and Wireless Communications, DIPDMWC 2016*, 81–86. <https://doi.org/10.1109/DIPDMWC.2016.7529368>
- ✓ Ministerio de Administraciones Públicas. (2006). *MAGERIT.v.2. Metodología de Análisis y Gestión de Riesgos de los Sistemas de Información. 3 - Guías de Técnicas. 3*, 72.
- ✓ Mohd Fauzi, S. S., Bannerman, P. L., & Staples, M. (2010). Software Configuration Management in Global Software Development: A systematic map. *Proceedings - Asia-Pacific Software Engineering Conference, APSEC*, 3(1), 404–413. <https://doi.org/10.1109/APSEC.2010.53>
- ✓ Ouedraogo, M., Mouratidis, H., Dubois, E., & Khadraoui, D. (2010). Information systems security criticality and assurance evaluation. *Lecture Notes in Computer Science (Including Subseries Lecture*

- ✓ Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 6059 LNCS, 38–54.
https://doi.org/10.1007/978-3-642-13577-4_4
- ✓ Payette, J., Anegebe, E., Caceres, E., & Muegge, S. (2015). Secure by Design: Cybersecurity Extensions to Project Management Maturity Models for Critical Infrastructure Projects. *Technology Innovation Management Review*, 5(6), 26.
<http://proxy1.ncu.edu/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=edb&AN=108596380&site=eds-live>
- ✓ Petersen, K., Feldt, R., Mujtaba, S., & Mattsson, M. (2008). Systematic Mapping Studies in Software Engineering. *12Th International Conference on Evaluation and Assessment in Software Engineering*, 17, 10. <https://doi.org/10.1142/S0218194007003112>
- ✓ Pitropakis, N., Darra, E., Vrakas, N., & Lambrinoudakis, C. (2013). It's all in the cloud: Reviewing cloud security. *Proceedings - IEEE 10th International Conference on Ubiquitous Intelligence and Computing, UIC 2013 and IEEE 10th International Conference on Autonomic and Trusted Computing, ATC 2013*, 1, 355–362.
<https://doi.org/10.1109/UIC-ATC.2013.13>
- ✓ Rea-Guaman, A. M., Sanchez-Garcia, I. D., Feliu, T. S., & Calvo-Manzano, J. A. (2017). Modelos de Madurez en Ciberseguridad: una revisión sistemática. *Iberian Conference on Information Systems and Technologies, CISTI*.
<https://doi.org/10.23919/CISTI.2017.7975865>
- ✓ Rot, a, & Sobinska, M. (2013). IT security threats in cloud computing sourcing model. *2013 Federated Conference on Computer Science and Information Systems, FedCSIS 2013*, 1153–1156. <http://www.scopus.com/inward/record.url?eid=2-s2.0-84892536420&partnerID=40&md5=5ed58482479a8c1e0dc64e972e727d01>
- ✓ Saleem, H., & Burney, S. M. A. (2019). Imposing Software Traceability and Configuration Management for Change Tolerance in Software Production. *IJCSNS International Journal of Computer Science and Network Security*, 19(1), 145–154.

- ✓ Schwägerl, F., Buchmann, T., Uhrig, S., & Westfechtel, B. (2015). Towards the integration of model-driven engineering, software product line engineering, and software configuration management. MODELSWARD 2015 - 3rd International Conference on Model-Driven Engineering and Software Development, Proceedings, 5–18.
<https://doi.org/10.5220/0005195000050018>
- ✓ Shamala, P., & Ahmad, R. (2014). A proposed taxonomy of assets for information security risk assessment (ISRA). 2014 4th World Congress on Information and Communication Technologies, WICT 2014, 29–33.
<https://doi.org/10.1109/WICT.2014.7077297>
- ✓ Shameli-Sendi, A., Aghababaei-Barzegar, R., & Cheriet, M. (2016). Taxonomy of information security risk assessment (ISRA). *Computers & Security*, 57, 14–30.
<https://doi.org/10.1016/j.cose.2015.11.001>
- ✓ Silva, C., Batista, R., Queiroz, R., Garcia, V., Silva, J., Gatti, D., Assad, R., Nascimento, L.,
- ✓ Brito, K., & Miranda, P. (2016). Towards a taxonomy for security threats on the web ecosystem. Proceedings of the NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium, Noms, 584–590.
<https://doi.org/10.1109/NOMS.2016.7502862>
- ✓ Silva, G. B. E., & Da Cunha, A. M. (2012). Extending an GCS ontology for configuration status reporting. Proceedings of the 2012 IEEE 16th International Enterprise Distributed Object Computing Conference Workshops, EDOCW 2012, 156–164.
<https://doi.org/10.1109/EDOCW.2012.29>
- ✓ Siponen, M. T. (2002). MATURITY CRITERIA FOR DEVELOPING SECURE IS AND SW. Proceedings of the IFIP TC11 17th International Conference on Information Security. Visions and Perspectives, 1–18. <https://doi.org/10.1007/978-0-387-3558>

- ✓ Tripathi, A., & Singh, U. K. (2010). Towards standardization of vulnerability taxonomy. ICCTD 2010 - 2010 2nd International Conference on Computer Technology and Development, Proceedings, ICCTD, 379–384.
<https://doi.org/10.1109/ICCTD.2010.5645826>
- ✓ Tripathi, A., & Singh, U. K. (2012). Taxonomic Analysis of Classification Schemes in Vulnerability Databases. 2011 6Th International Conference on Computer Sciences and Convergence Information Technology (Iccit), 686–691.
- ✓ UNE-ISO GUIA 73, U.-I. G. 73:2010. (2010). Gestión del riesgo. Vocabulario.
- ✓ US Department of Defense. (2019). Cybersecurity Maturity Model Certification. Cmmc.
- ✓ US Department of Homeland Security. (2014). Department of Homeland Security Cybersecurity Capability Maturity Model White Paper.
[file:///C:/Users/Marcelo/Downloads/Capability Maturity Model White Paper.pdf](file:///C:/Users/Marcelo/Downloads/Capability%20Maturity%20Model%20White%20Paper.pdf)
- ✓ Uskarci, A., & Demirörs, O. (2017). Do staged maturity models result in organization-wide continuous process improvement? Insight from employees. Computer Standards and Interfaces, 52(January), 25–40. <https://doi.org/10.1016/j.csi.2017.01.008>
- ✓ von Känel, J., Cope, E. W., Deleris, L. A., Nayak, N., & Torok, R. G. (2010). Three key enablers to successful enterprise risk management. IBM Journal of Research and Development, 54(3), 231–245. <https://doi.org/10.1147/JRD.2010.2043973>
- ✓ Von Solms, S. H. B. (2015). A maturity model for part of the African Union Convention on Cyber Security. Proceedings of the 2015 Science and Information Conference, SAI 2015, 1316–1320. <https://doi.org/10.1109/SAI.2015.7237313>
- ✓ White, G. B. (2011). The community cyber security maturity model. 2011 IEEE International Conference on Technologies for Homeland Security, HST 2011, 173–178.
<https://doi.org/10.1109/THS.2011.6107866>
- ✓ Yahya, F., Walters, R. J., & Wills, G. B. (2015). Analysing threats in cloud storage. 2015 World Congress on Internet Security, WorldCIS 2015, 44–48.

<https://doi.org/10.1109/WorldCIS.2015.7359411>

- ✓ Yazid, A. I. S., Faizal, M. A., Rabiah, A., Shahrin, S., & Solahuddin, S. (2012). Enhancement of Asset value classification for Mobile devices. Proceedings 2012 International Conference on Cyber Security, Cyber Warfare and Digital Forensic, CyberSec 2012, 106–110. <https://doi.org/10.1109/CyberSec.2012.6246097>
- ✓ Yu, Z., Thomborson, C., Wang, C., Fu, J., & Wang, J. (2009). A security model for VoIP steganography. 1st International Conference on Multimedia Information Networking and Security, MINES 2009, 1, 35–40. <https://doi.org/10.1109/MINES.2009.227>
- ✓ Zhao, Z., & Dai, Y. (2012). A New Method of Vulnerability Taxonomy Based on Information Security Attributes. 2012 IEEE 12th International Conference on Computer and Information Technology, 739–741. <https://doi.org/10.1109/CIT.2012.152>
- ✓ Alain April. (2018). Software Configuration Management.
- ✓ Bartusevics, A., & Novickis, L. (2015a). Model-based Approach for Implementation of Software Configuration Management Process. 5.
- ✓ Bartusevics, A., & Novickis, L. (2015b). Models for implementation of software configuration management. Procedia Computer Science, 43(C), 3–10. <https://doi.org/10.1016/j.procs.2014.12.002>
- ✓ Cederbom, M. (2015). Institutionen för datavetenskap Improving software configuration management across by Martin Cederbom.
- ✓ Chou, I. H. (2011). Secure Software Configuration Management Processes for nuclear safety software development environment. Annals of Nuclear Energy, 38(10), 2174–2179. <https://doi.org/10.1016/j.anucene.2011.06.016>
- ✓ CMMI. (2010). CMMI ® para Desarrollo, Versión 1.3 Equipo del Producto CMMI. <http://www.sei.cmu.edu>
- ✓ Durrani, U., Pita, Z., Richardson, J., & Lenarcic, J. (2014). Association for Information

Systems AIS Electronic Library (AISeL) AN EMPIRICAL STUDY OF LEAN AND AGILE INFLUENCES IN SOFTWARE CONFIGURATION MANAGEMENT.

- ✓ Engineering Standards Committee of the IEEE Computer Society, S. (2012). IEEE Standard for Configuration Management in Systems and Software Engineering Sponsored by the Software & Systems Engineering Standards Committee.
- ✓ Espinosa, G. (2014). GESTIÓN DE CONFIGURACIÓN Y LÍNEA DE PRODUCTOS PARA MEJORAR EL PROCESO EXPERIMENTAL EN INGENIERÍA DEL SOFTWARE.
- ✓ Fahmy, S. (2020). The Evolution of Software Configuration Management. International Journal of Advanced Trends in Computer Science and Engineering, 9(1.3), 50–63.
<https://doi.org/10.30534/ijatcse/2020/0891.32020>
- ✓ Fahmy, S., Deraman, A., & Yahaya, J. H. (2018). The role of human in software configuration management. ACM International Conference Proceeding Series, 56–60.
<https://doi.org/10.1145/3185089.3185117>
- ✓ Fahmy, S., Deraman, A., Yahaya, J., Nasir, A., & Shamsudin, N. (2020). The evolution of software configuration management. International Journal of Advanced Trends in Computer Science and Engineering, 9(1 Special Issue 3), 50–63.
<https://doi.org/10.30534/ijatcse/2020/0891.32020>
- ✓ Jendraszyk, D. (2021). Requirements Analysis of Approvable Software for a GLASS Ground Station according to DO-278A.
- ✓ Kitchenham, B. A., & Jorgensen, T. D. M. (2004). Evidence-based Software Engineering. Proceedings of the 26th International Conference on Software Engineering.
<https://doi.org/10.1109/ICSE.2004.1317449>
- ✓ Kitchenham, B., & Charters, S. (2007). Guidelines for performing Systematic Literature reviews in Software Engineering Version 2.3. Engineering, 45(4ve), 1051.
<https://doi.org/10.1145/1134285.1134500>
- ✓ Maida, E., & Pacienza, J. (2015). Metodologías de desarrollo de software.

- ✓ Mohd Fauzi, S. S., Bannerman, P. L., & Staples, M. (2010). Software Configuration Management in Global Software Development: A systematic map. Proceedings - Asia-Pacific Software Engineering Conference, APSEC, 3(1), 404–413.
<https://doi.org/10.1109/APSEC.2010.53>
- ✓ Orozco Eduardo. (2017). Proceso ágil para la gestión de la configuración de software en micro, pequeñas y medianas empresas de software.
- ✓ Paredes, S., Hinojosa, C., & Ruiz, J. (2011). La importancia de la Gestión de la Configuración del Software, en. In Tendencias en Computación (Vol. 1, Issue 3).
- ✓ Petersen, K., Feldt, R., Mujtaba, S., & Mattsson, M. (2008). Systematic Mapping Studies in Software Engineering. 12Th International Conference on Evaluation and Assessment in Software Engineering, 17, 10. <https://doi.org/10.1142/S0218194007003112>
- ✓ Rivero, A. (2011). Propuesta de una línea base de requisitos genéricos para la Línea de Productos de Software SCADA del Centro de Desarrollo de Informática Industrial.
- ✓ Saleem, H., & Burney, S. M. A. (2019). Imposing Software Traceability and Configuration Management for Change Tolerance in Software Production. IJCSNS International Journal of Computer Science and Network Security, 19(1), 145–154.
- ✓ Sarma, A., Chen, X., Kuttal, S., Dabbish, L., & Wang, Z. (2016). Hiring in the global stage: Profiles of online contributions. Proceedings - 11th IEEE International Conference on Global Software Engineering, ICGSE 2016, 1–10.
<https://doi.org/10.1109/ICGSE.2016.35>
- ✓ Schwägerl, F., Buchmann, T., Uhrig, S., & Westfechtel, B. (2015). Towards the integration of model-driven engineering, software product line engineering, and software configuration management. MODELSWARD 2015 - 3rd International Conference on Model-Driven Engineering and Software Development, Proceedings, 5–18.
<https://doi.org/10.5220/0005195000050018>

- ✓ Silva, G. B. E., & Da Cunha, A. M. (2012). Extending an SCM ontology for configuration status reporting. Proceedings of the 2012 IEEE 16th International Enterprise Distributed Object Computing Conference Workshops, EDOCW 2012, 156–164.
<https://doi.org/10.1109/EDOCW.2012.29>
- ✓ Universidad de la República de Uruguay. (2015). Proceso Software y Ciclo de Vida.
<https://www.fing.edu.uy/tecnoinf/maldonado/cursos/2015/rpyl/desarrolloSoftware.pdf>
- ✓ Uskarci, A., & Demirörs, O. (2017). Do staged maturity models result in organization-wide continuous process improvement? Insight from employees. Computer Standards and Interfaces, 52(January), 25–40. <https://doi.org/10.1016/j.csi.2017.01.008>
- ✓ UNE-ISO Guía 73:2010, “Gestión del riesgo. Vocabulario”. ISO/IEC 27000:2009, “Information technology — Security techniques — Information security management systems — Overview and vocabulary”.
- ✓ ISO 73 ISO Guide 73:2009, “Risk management — Vocabulary”. UNE-ISO Guía 73:2010, “Gestión del riesgo. Vocabulario”.
- ✓ ISO/IEC 27000:2009, “Information technology — Security techniques — Information security management systems — Overview and vocabulary”.

Anexos