



**FACULTAD DE INGENIERÍA Y ARQUITECTURA
SECCIÓN DE POSGRADO**

**DISEÑO DE UNA METODOLOGÍA DE CERTIFICACIÓN DE
PRODUCTOS DE SOFTWARE ORIENTADO AL SECTOR**

PÚBLICO

PRESENTADA POR

CARLOS JULIÁN BARZOLA MENDOZA

HÉCTOR HERNÁN HENRÍQUEZ TABOADA

**TESIS PARA OPTAR EL GRADO DE MAESTRO DE INGENIERÍA DE
COMPUTACIÓN Y SISTEMAS CON MENCIÓN EN
GESTIÓN DE TECNOLOGÍAS DE INFORMACIÓN**

LIMA – PERÚ

2014



Reconocimiento - No comercial - Sin obra derivada

CC BY-NC-ND

El autor sólo permite que se pueda descargar esta obra y compartirla con otras personas, siempre que se reconozca su autoría, pero no se puede cambiar de ninguna manera ni se puede utilizar comercialmente.

<http://creativecommons.org/licenses/by-nc-nd/4.0/>



USMP
UNIVERSIDAD DE
SAN MARTÍN DE PORRES

**FACULTAD DE
INGENIERÍA Y ARQUITECTURA**

SECCIÓN POSGRADO

**DISEÑO DE UNA METODOLOGÍA DE CERTIFICACIÓN DE
PRODUCTOS DE SOFTWARE ORIENTADO AL SECTOR**

PÚBLICO

TESIS

**PARA OPTAR EL GRADO DE MAESTRO DE INGENIERÍA DE
COMPUTACIÓN Y SISTEMAS CON MENCIÓN EN GESTIÓN DE
TECNOLOGÍAS DE INFORMACIÓN**

PRESENTADO POR

**BARZOLA MENDOZA, CARLOS JULIÁN
HENRÍQUEZ TABOADA, HÉCTOR HERNÁN**

LIMA – PERÚ

2014



Dedico este proyecto a mi familia que me ayudó con su apoyo incondicional a lograr mis metas profesionales. Gracias a Dios por otorgarme la sabiduría y la salud. Especialmente a Gabriela, mi esposa y a mis hijos Renzo y Carla,... ¡Dios los bendiga!



Quiero dedicarle este trabajo a Dios que me ha dado la vida y fortaleza para terminar este proyecto de investigación. A mi familia, y en especial a mi madre, por su ayuda y constante cooperación y, a mis hermanos por apoyarme en los momentos más difíciles.



ÍNDICE

RESUMEN	v
ABSTRACT	vi
INTRODUCCIÓN	vii
CAPÍTULO I DEFINICIÓN DEL PROBLEMA Y METODOLOGÍA	
1.1 Descripción de la situación problemática	1
1.2 Definición del problema de decisión	4
1.3 Supuestos con relación a condiciones del entorno	7
1.4 Objetivos tesis	7
1.5 Hipótesis	9
1.6 Identificación de variables	9
1.7 Matriz de consistencia	11
1.8 Metodología	14
1.9 Justificación e Importancia	17
1.10 Delimitación	18
CAPÍTULO II MARCO TEÓRICO CONCEPTUAL	
2.1 Antecedentes y modelos de solución previos	20
2.2 Bases y enfoques teóricos	87

2.3	Modelo teórico o conceptual	97
CAPÍTULO III DISEÑO DE UNA METODOLOGÍA DE CERTIFICACIÓN DE PRODUCTOS SOFTWARE ORIENTADO AL SECTOR PÚBLICO		
3.1	Descripción general	99
3.2	Método de elaboración de la metodología de certificación	100
CAPÍTULO IV APLICACIÓN DE LA METODOLOGÍA		
4.1	Instituciones públicas	157
4.2	Identificar y análisis de los factores críticos	158
4.3	Análisis interno	158
4.4	Análisis del contexto	163
4.5	Discusión de alternativas.	163
CAPÍTULO V RESULTADOS		
5.1	Introducción	171
5.2	Listado de formatos y documentos llenados	171
CAPÍTULO VI CONSTRATE DE HIPÓTESIS		
6.1	Prueba Chi Cuadrado	201
6.2.	Resultados	202
6.3.	Valor espero	203
6.4	Decisiones	204
CONCLUSIONES		206
RECOMENDACIONES		208
FUENTES DE INFORMACIÓN		210
GLOSARIO		216
ANEXOS		233

RESUMEN

En la actualidad el cliente es mucho más exhaustivo y crítico con los productos o servicios que adquiere u obtiene, entre ellos los servicios que ofrecen las soluciones de tecnología de información, y muchas veces son la cadena que ayuda a que determinada empresa tenga una mala reputación por la insatisfacción recibida.

Se realizó la investigación del porqué de los continuos errores en las aplicaciones que eran ofrecidas al usuario; se determinó que no se habían establecido procesos de desarrollo de software adecuados para poder asegurar la calidad del software que es implementado, y algunas de las razones por las que no lo habían efectuado eran por desconocimiento de estándares de calidad, modelos de proceso, metodologías de desarrollo de software, los cuales no habían sido tomados en cuenta en sus procesos actuales del ciclo de vida del software, basándose en la NTP ISO/IEC 12207.

Para dar una opción de cómo asegurar la calidad del software a implementar, se estudiaron algunos estándares, normas y modelos, de los cuales se pudo diseñar una metodología adecuada para ofrecer un software de calidad, y lograr, de esta manera, brindar soluciones al usuario que permitan a las empresas del estado, mejorar sus indicadores de apreciación.

Palabras claves: Estándares de Calidad, Modelos de Proceso, Metodologías de Desarrollo de Software.

ABSTRACT

Today the customer is much more comprehensive and critical to the goods or services purchased or obtained, including the services offered by information technology solutions, and often are the chain that helps certain company has a bad reputation by dissatisfaction received.

Investigation of why continuous errors in applications that were offered to the user is performed; was determined that it had not established adequate processes of software development to ensure the quality of the software is implemented, and some of the reasons why they had not done was due to ignorance of standards of quality, process models, methodologies software development, which had not been taken into account in their current processes of the software life cycle, based on the NTP ISO / IEC 12207.

To give an option of how to ensure the quality of the software to be implemented, some standards, rules and models were studied, of which it was possible to design an appropriate methodology to deliver software quality, and achieve, thus providing solutions to the user to enable state enterprises, improve their assessment indicators.

Keywords: Quality Standards, Process Models, Software Development Methodologies.

INTRODUCCIÓN

Una característica de nuestro mundo desarrollado es que todas las áreas de informática deben brindar productos de tecnología de información de calidad, que en su conjunto, proporcionen los bienes y servicios que son necesarios y que la sociedad demanda.

Para esto es necesario tener establecidas las políticas, normas, procedimientos y metodologías a aplicar para el desarrollo de las buenas prácticas que se tienen que aplicar para el logro de los objetivos planteados. En tal sentido, el presente trabajo, trae experiencias desarrolladas a lo largo de horas de investigación y reuniones de evaluación, las cuales permiten tener un control adecuado de la calidad del producto a brindar a los usuarios, considerando lo importante dentro de lo viable, de manera que no tenga un impacto sino gradual en cada una de las instituciones que puedan adoptarlos de manera flexible en su aplicación.

Actualmente todas las instituciones públicas del Estado están en un proceso de mejoramiento continuo de las Áreas de Tecnología de Información. Es por ello que están definiendo y desarrollando los procesos de TI relacionados con los procesos de Negocios, que son la base de toda institución que quiera lograr productos de TI de calidad. Para esto han implementado consultorías y estudios de investigación tendentes a la aplicación de metodologías y definición de procesos de planes estratégicos que permitan lograr los objetivos planteados.

Es por ello que la búsqueda de la excelencia y el mejoramiento continuo constituyen tareas fundamentales de estas instituciones, quienes trabajando en equipos multidisciplinarios proponen, planifican, desarrollan y ejecutan los cambios necesarios para ofrecer servicios de calidad acordes con las nuevas tendencias y los avances de la tecnología.

Dentro del marco expuesto, la presente tesis intitulada ***Diseño de una Metodología de Certificación de Productos de Software orientado al Sector Público*** se presenta con la finalidad de proponer una metodología acorde con las necesidades y adaptaciones que necesitan las instituciones públicas, donde a la par se dé cumplimiento a las normas, políticas y procedimientos establecidos por los diferentes órganos de control de las cuales dependen.



CAPÍTULO I

DEFINICIÓN DEL PROBLEMA Y METODOLOGÍA

1.1 Descripción de la Situación Problemática

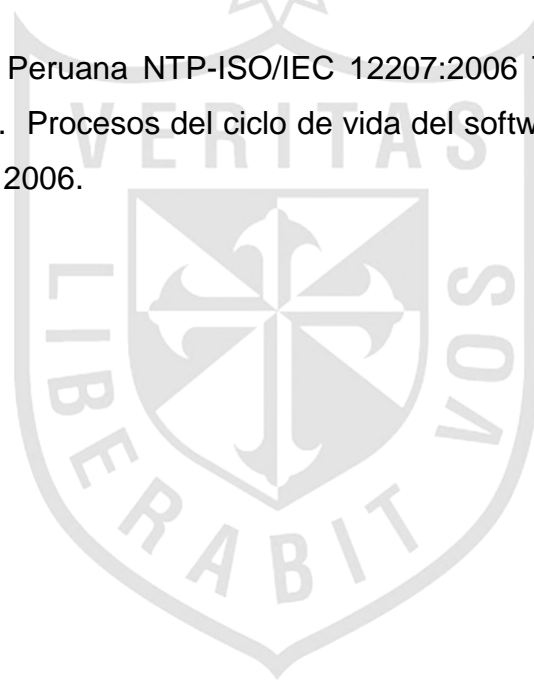
1.1.1 Principales hechos, síntomas y evidencias

Las entidades del Estado cuentan actualmente con una infraestructura tecnológica que permite el transporte de voz y datos. Gracias a ello se pueden brindar servicios en línea a los usuarios, permitiendo una mejor calidad en el servicio; sin embargo, las continuas fallas de las aplicaciones desarrolladas y/o customizadas, no permite tener la operatividad de los servicios dentro de los estándares de más del 99.9% de operatividad, causando esto malestar en el público usuario y por ende se ve afectada la calidad e imagen de las instituciones públicas, motivo por el cual las instituciones del Estado peruano, necesitan tener definida una metodología de Certificación de Productos de Software, que permita minimizar los errores de las aplicaciones finales que son puestas a explotación, la cual derivará en la calidad de los servicios brindados, y porqué no llevar adelante una certificación que asegure el correcto desarrollo de Productos de Software.

La Superintendencia de Banca y Seguros y AFP publica el 6 de Abril de 2009 en el diario El Peruano, la Circular G-140-2009

Gestión de la Seguridad de la Información la cual exige que las organizaciones que están sujetas a su mecanismo regulador, implanten un sistema orientado a controlar la operatividad de los aplicaciones que brindan las instituciones financieras así como minimizar el riesgo operativo de las instituciones, a través de un Sistema de Gestión de Seguridad de Información (SGSI), al igual que las normas siguientes:

- ✓ Norma Técnica Peruana "NTP ISO/IEC 27001:2008 EDI Tecnología de la Información. Técnicas de Seguridad. Sistemas de gestión de seguridad de la Información. Requisitos 1ª Edición", 11 de enero de 2009
- ✓ Norma Técnica Peruana NTP-ISO/IEC 12207:2006 TECNOLOGÍA DE LA INFORMACIÓN. Procesos del ciclo de vida del software. Perú. 2ª Edición, el 28 de julio de 2006.



Oportunidad

Situación actual	Situación deseable
<ul style="list-style-type: none"> • El constante mejoramiento de la situación actual de las áreas de TI de las Instituciones Públicas, muestran que existen requerimientos y necesidades no atendidas de los usuarios tanto internos como externos. 	<ul style="list-style-type: none"> • Las necesidades de otros departamentos deben ser priorizadas. • Se debe aprovechar el apoyo de la alta dirección para los proyectos que son innovadores.
<ul style="list-style-type: none"> • La importancia que le otorga la Alta Dirección de las Instituciones Públicas a proyectos relacionados con la modernización tecnológica, no está siendo correctamente utilizada. 	<ul style="list-style-type: none"> • Las Áreas de Tecnología de Información (Sistemas, Informática o afines) deben ser los pioneros en brindar los lineamientos de mejoras en los servicios que requieren de Tecnología de Información los usuarios de la Institución.
<ul style="list-style-type: none"> • La importancia del Área de Tecnología de Información, respecto al cumplimiento de los objetivos operativos y estratégicos de las Instituciones Públicas, muchas de ellas no son reflejadas en su total dimensión. 	<ul style="list-style-type: none"> • Aprovechar el conocimiento de la infraestructura de tecnologías de información para los futuros proyectos a realizarse, que poseen dichas áreas de TI.
<ul style="list-style-type: none"> • No cuenta con un estándar o modelo que sea replicable en cuanto al proceso de certificación de calidad de los software que construye. 	<ul style="list-style-type: none"> • El contar con asignaciones presupuestales que permitan un desarrollo adecuado. • Contar con un modelo de certificación de calidad.

1.2 Definición del Problema de Decisión.

Selección del Problema

Las diferentes áreas de Tecnología de Información de las Instituciones Públicas no cuentan con una Metodología de Certificación de Productos de Software establecido, que permita elaborar estrategias de desarrollo de software de calidad, que promuevan prácticas adaptativas en vez de predictivas, centradas en las personas o en los equipos, orientadas hacia la funcionalidad y la entrega, de comunicación intensiva y que requieren implicación directa, esto es una gran debilidad que adolecen las Áreas de Sistemas de las instituciones públicas, lo que impacta en la imagen institucional y por ende en el malestar general de los usuarios del sector público.

La información que mostramos en las tablas corresponde a una institución pública, tal como veremos:

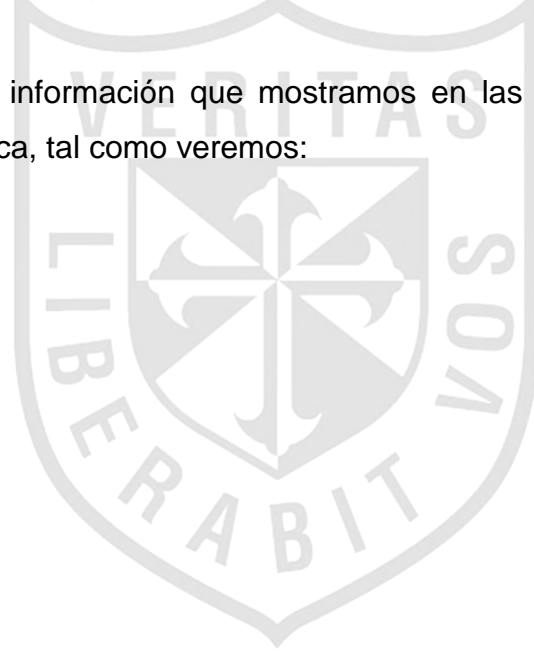


Tabla 01: Cantidad de mantenimientos y/o proyectos que se aprobaron en desarrollo
Para realizar las pruebas en certificación

	AÑO 2013										Total
	Enero	Febrero	Marzo	Abril	Mayo	Junio	Julio	Agosto	Setiembre	Octubre	
Mantenimiento	59	35	46	33	57	62	66	57	70	59	544
Proyecto	6	2	1	3	1	2	3	1	0	2	21

Tabla 02: Cantidad de mantenimientos y/o proyectos atendidos por la Sección Calidad de Soluciones

		2013										Total
		Enero	Febrero	Marzo	Abril	Mayo	Junio	Julio	Agosto	Setiembre	Octubre	
Certificados	Mantenimiento	43	31	40	29	28	58	49	44	41	43	406
	Proyecto	2	1	2	3	0	1	2	0	0	0	11
Rechazados	Mantenimiento	19	19	10	10	18	15	21	10	15	15	152
	Proyecto	5	3	4	0	1	1	2	0	1	1	18
Transitorio	Mantenimiento	64	47	20	21	40	7	1	4	9	4	217
Regularizado	Proyecto	0	0	0	1	0	1	0	0	0	0	2
Total General		133	101	76	64	87	83	75	58	66	63	806

Elaboración: Los autores

Certificadas : Actividad finalizada de la prueba y enviada a Producción

Rechazados : Actividad que se rechaza por no cumplir con los artefactos y por tener muchos errores.

Transitorias : *Actividad que no pasa por Calidad, las pruebas se realizan en Desarrollo y pasan a Producción directamente.*

AÑO 2011- 2012

Tabla 03: Cantidad de mantenimientos y/o proyectos atendidos en los años 2011 y 2012

		Rechazados		Certificados		TOTAL		TOTAL
		M	P	M	P	M	P	GENERAL
Años	2011	130	11	693	10	823	21	844
	2012	140	12	535	8	675	20	695

Elaboración: Los autores

1.3 Supuestos con relación a condiciones del entorno

Actualmente las entidades del sector público no poseen un Diseño de una Metodología de Certificación de Productos Software, originando esto una debilidad ya que genera constante problema en las definiciones de los requerimientos, mala planificación, inadecuada calidad de las pruebas unitarias, de integración y de usuario final, lo cual al final determina una ampliación de tiempo en el desarrollo y en consecuencia unos mayores costos y pérdida de oportunidad de negocio.

Los casos de éxito en empresas del mismo giro de negocio permitirán validar y dar soporte a estas nuevas tendencias metodológicas soportadas por herramientas de TI.

El Diseño de una Metodología de Certificación de Productos de Software no es de bajo costo, por cuanto involucra una inversión para la creación de documentos y el uso de computadoras para mantenerlos. Se requiere desarrollar todo el esquema de administración y distribución; asimismo, la combinación de personas y de tecnología, el ser humano es el que convierte la información en conocimiento. Los casos de éxito dan soporte al desarrollo de este proyecto.

1.4 Objetivos de la tesis

1.4.1 Objetivo general

Diseñar una Metodología de Certificación de Productos de Software orientado al sector público para planificar, desarrollar, verificar, validar y capturar el conocimiento de los especialistas y plasmarlo en una herramienta de TI, de tal forma que ante una incidencia se pueda recuperar el servicio en menor tiempo y a menor costo.

1.4.2 Objetivos específicos

- Definir la Metodología de Certificación de Productos Software Orientado al Sector Público, de acuerdo con el procedimiento de inspección, alineado a la NTP 12207: 2006, la cual permitirá definir los listado de comprobaciones que derivarán en:
 - ✓ Definir fases, métodos de acuerdo con el proceso de desarrollo de software.
 - Inicio
 - Elaboración
 - Construcción
 - Transición
 - ✓ Definir plantillas de documentos:
 - Definición funcional
 - Definición técnica
 - Definición de Arquitectura
 - ✓ Definir formularios de control estándar que permitan monitorear de acuerdo con la naturaleza del seguimiento respectivo.
 - Formulario de definición de recursos (BD, archivos, tablas, servidores, otros)
 - Formulario de revisión de pares
 - Formulario de `pase a Producción (donde incluye la migración a los diferentes ambientes que maneje la institución).
 - ✓ Definir planes de control para el seguimiento del proyecto y/o mantenimiento:
 - Hoja de control de documentación requerida
 - Hoja de seguimiento (Project) – Workflow
- **Identificar los factores críticos de éxito en la ejecución de la prueba piloto del Diseño de una Metodología de Certificación de Productos Software Orientado al Sector Público**, son aquellos factores o aspectos

que se han encontrado al aplicar la metodología de certificación del producto.

En la prueba piloto, se realizará, luego de implementar la Metodología de Certificación de Productos, la identificación de los factores críticos de éxito internos y externos, de esta manera estaríamos dando control de calidad al resultado de las pruebas requeridas.

1.5 Hipótesis

1.5.1 Hipótesis general

Desarrollo de un Diseño de una Metodología de Certificación de Productos de Software Orientado al Sector Público, que nos permitirá establecer un vínculo con el desarrollo con los diferentes parámetros de control establecidos por los órganos de control de las instituciones públicas.

1.5.2 Hipótesis específicas

- a) La metodología de certificación de productos de software facilita el conocimiento total de requerimiento logrando la calidad de concordancia solicitada, en la cual se establecerá una guía del desarrollo y/o mantenimiento de los productos de software.
- b) La metodología de certificación de productos de software permitirá ser posteriormente replicada a todas las áreas de TI, de manera que se pueda lograr si se requiere una certificación externa de los procesos efectuados.

1.6 Identificación de variables

1.6.1 Variable independiente

La creación de la Metodología de Certificación de Productos de Software orientado al Sector Público como fuente de referencia para el correcto cumplimiento de las necesidades de los usuarios de las aplicaciones.

1.6.2 Variable dependiente

El vínculo de la metodología desarrollada con los diferentes parámetros de control establecidos por los órganos de control de las instituciones públicas, permitirá identificar las debilidades y establecer las mejoras necesarias para el correcto cumplimiento de las normas y políticas de las instituciones públicas.



1.7 Matriz de Consistencia

1.7.1 Matriz principal

PROBLEMA	OBJETIVO	HIPÓTESIS	VARIABLES
Problema general	Objetivo general	Hipótesis general	
¿Cómo diseñar la metodología de certificación de productos de software orientado al sector público?	Diseñar una metodología de certificación de productos de software orientado al sector público, que permita certificar la calidad de las aplicaciones puestas en producción.	Diseñar una metodología de certificación de productos de software orientado al sector público permitirá establecer una guía de referencia que permita minimizar los errores de implementación de productos de software.	VI : La creación de la metodología de certificación de productos de software orientado al sector publico para planificar, desarrollar, verificar, validar y capturar el conocimiento de los especialistas. VD: El vínculo de la metodología desarrollada con los diferentes parámetros de control establecidos por los órganos de control de las instituciones públicas.

1.7.2 Matriz secundaria

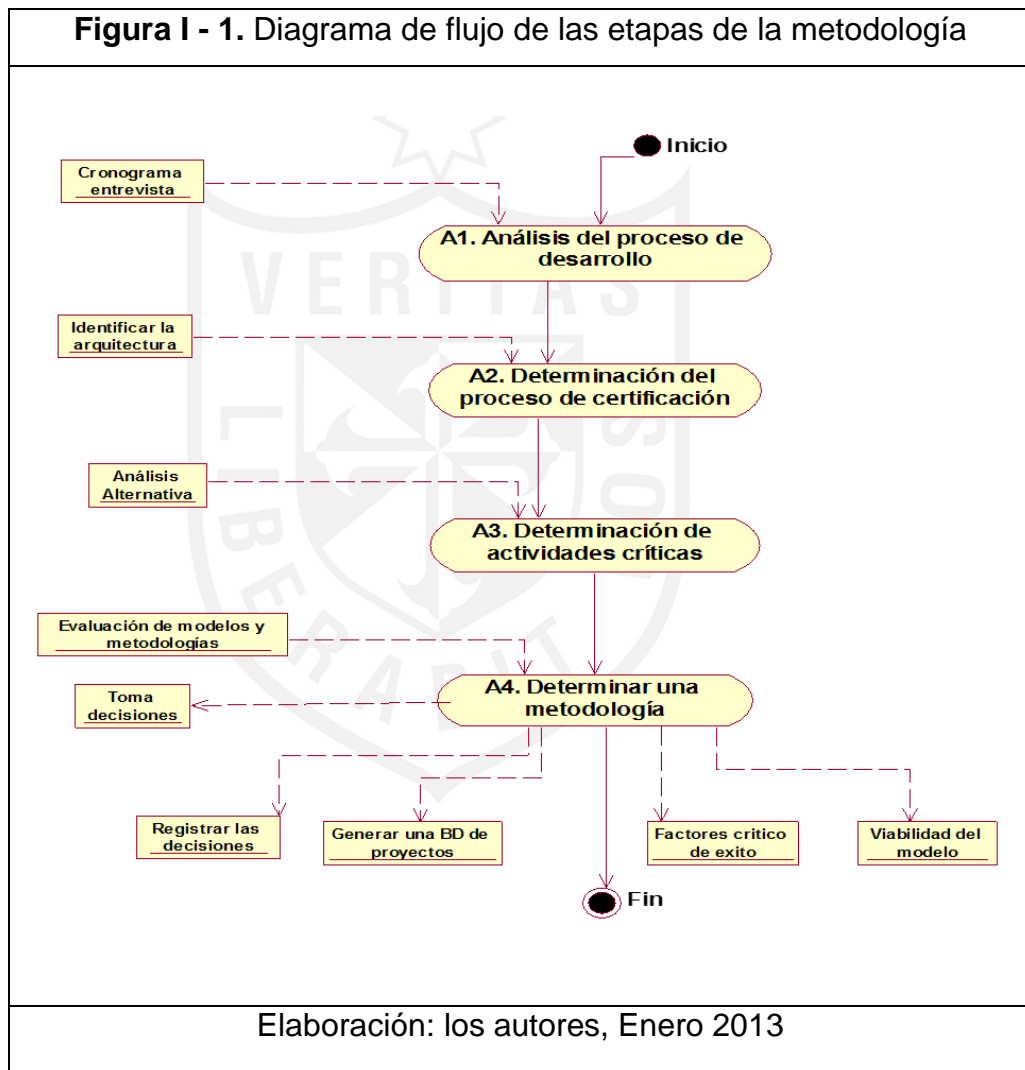
Problemas específicos	Objetivos específicos	Hipótesis específicos	Variables
<ul style="list-style-type: none"> ▪ ¿Cómo estructurar la metodología de certificación de productos de software orientado al sector público? 	<ul style="list-style-type: none"> ▪ Definir metodología de certificación de productos software orientado al sector público, de acuerdo al procedimiento de inspección, alineado a la NTP 12207:2006 	<ul style="list-style-type: none"> ▪ La metodología de certificación de productos de software, facilita el conocimiento total de requerimiento logrando la calidad de concordancia solicitada, lo cual establecerá una guía del desarrollo y/o mantenimiento de los productos de software. 	<p>VD1: La gestión del requerimiento pasando por sus fases logrando la calidad solicitada</p> <p>I1: Nivel de satisfacción de los usuarios de la institución. Tiempo</p> <p>VD2: Guía del desarrollo y/o mantenimiento</p> <p>I2: Grado de facilidad de desarrollo y/o mantenimiento de los productos de software. Desarrollo</p>

Problemas Específicos	Objetivos Específicos	Hipótesis Secundarias	Variables
<ul style="list-style-type: none"> ▪ ¿Cuáles son los factores críticos de éxito en la ejecución de la prueba de la metodología que permitirán establecer un producto de calidad en el sector público? 	<ul style="list-style-type: none"> ▪ Identificar los factores críticos de éxito en la ejecución de la prueba piloto del diseño de una metodología de certificación de productos software orientado al sector público. 	<ul style="list-style-type: none"> ▪ La metodología de certificación de productos de software, permitirá ser posteriormente replicadas a todas las áreas de TI, de manera tal de poder lograr si se requiere una certificación externa de los procesos efectuados. 	<p>VD2: Replicadas a todas las áreas de TI</p> <p>I2: Lograr si se requiere una certificación externa de los procesos efectuados. Verificación, Validación</p>

1.8 Metodología

1.8.1 Diseño(s) Metodológico(s)

Establecer y llevar a cabo las etapas, actividades que permitan cumplir con los objetivos de un proyecto en tiempo y costo esperados.



- **Análisis del Proceso de Desarrollo de Software en el sector público.**
En primera instancia, se trata de conocer el negocio para lo cual es necesario establecer un cronograma de entrevistas con los funcionarios principales de las áreas de negocio del sector público, así como con los principales funcionarios del Área de Informática que son partícipes del desarrollo de los proyectos de Software; Estas entrevistas permitirán conocer el grado de involucramiento así como las actividades que actualmente realizan dentro del desarrollo de los Productos de Software solicitados.
- **Determinación del Proceso de Certificación de Productos de Software.** Una vez identificada la arquitectura de proceso desarrollada, se deberá establecer el Proceso de Certificación de Productos de Software que se debe implementar, es necesario indicar que este debe ser actualizado constantemente por los especialistas responsables.
- **Determinación de actividades críticas de Tecnología de Información.**
De las actividades críticas identificadas para nuestro estudio solo tomaremos aquellas que formen parte de la Certificación de Productos de Software, el cual será analizado con todas las posibles alternativas de soluciones realizadas de acuerdo con estadísticas y evaluaciones que los especialistas conocen o tienen documentado en algunos casos, revisaremos toda la información existente para realizar la estructura adecuada y capturar el conocimiento de los especialistas que son los únicos conocedores de los eventos que se han presentado en toda la trayectoria del sector público.
- **Determinar una metodología (Modelo conceptual)**
De los estudios previos de los modelos y metodologías existentes debemos, analizar cada uno de ellos y desarrollar el modelo adecuado

alineado a nuestra realidad. Para ello debemos tener en cuenta lo siguiente:

✓ **Los fundamentos de la toma de decisiones.**

Debemos analizar cómo se toman las decisiones para la definición de los requerimientos, de qué manera se establecen las prioridades, cómo se controlan los avances y retrasos de los proyectos, de qué manera dan solución a los problemas que puedan suceder durante el desarrollo de los mismos, sobre todo cuando existe un cambio de alcance del proyecto.

✓ **Registrar las decisiones**

Para llevar el control de las decisiones tomadas ante las diversas acciones por los especialistas, se realizará el registro de cada uno de ellos, tomaremos en cuenta los fracasos y éxitos para luego poder clasificarlo adecuadamente y determinar los cambios, de acuerdo con los tipos de proyectos realizados.

✓ **Generar una Base de Datos de proyectos desarrollados**

Se generará una base de datos que contenga toda la información desarrollada por cada proyecto, de manera que permita un conocimiento de las acciones desarrolladas en la atención de los requerimientos efectuados. Cada suceso será almacenado en forma periódica.

✓ **Factores críticos de éxito del modelo propuesto**

Se analizarán los factores internos y externos que influyen en el diseño de la metodología propuesta. Este resultado nos permitirá conocer cuáles son los recursos necesarios para lograr el éxito del estudio.

✓ **Viabilidad del modelo propuesto para el sector público**

Una vez desarrollada todas las etapas anteriores se evaluará la viabilidad del diseño de la metodología propuesta para el sector público.

1.9 Justificación e importancia

La tendencia a la globalización de las economías nos exige crear nuevos enfoques y niveles de calidad que nos permitan posicionarnos dentro de la competitividad de forma individual y de grupos. Es cierto que la empresa es el campo propicio para desarrollar nuevos enfoques y teorías administrativas que nos permitan organizar de manera óptima los recursos de la misma. Para ello nos corresponde investigar y construir nuevas formas de organización, en nuestro caso, adaptar las existentes, de tal forma que nos permita generar un diseño de una metodología de Certificación de Productos de Software. Actualmente, la revolución tecnológica está llevando a los empresarios a formar parte del mundo de una sociedad informatizada, mediante el empleo y la aplicación de las nuevas tecnologías. Mediante la información, las computadoras y la Internet están creando nuevos paradigmas empresariales.

Aunque las ventajas son numerosas y deseables, los riesgos de seguridad tampoco tienen precedente y suponen un gran desafío en el campo de las Tecnologías de Información. Los procesos robustos y procedimientos de seguridad son tan importantes como el equipamiento y la tecnología al combinar las mejores competencias de muchas unidades dentro de la red; Cada organización es más poderosa y flexible de lo que puede ser individualmente.

1.10 Delimitación

Alcances

Por la dimensión de la información que se administra en la institución financiera de estudio, para nuestro estudio sólo nos centraremos en las áreas que formen parte del desarrollo de los proyectos de Software, siendo el área final del control la Sección de Certificación de Productos de Software.

Limitaciones

- **Acceso a la Información.** Es necesario tener en cuenta que la información que pueda ser utilizada para este estudio es confidencial por estar clasificada como activo de información, motivo por el cual el acceso a esta información es restringido.
- **La institución no toma conciencia de la importancia del Aseguramiento de la Calidad de los Software que debe brindar para uso de los usuarios finales, sean estos miembros de la institución o clientes virtuales.** Actualmente las plataformas tecnológicas son las adecuadas para brindar el mejor servicio a los clientes. Para que esto sea completo se debe minimizar la inoperatividad de los servicios o una mala atención a causa de problemas de malas definiciones, pruebas deficientes y/o implementaciones defectuosas.
- **Recurso humano.** Cuando nos referimos a los empleados que puedan participar en este proyecto, podemos definir que no podemos contar con ningún recurso ya que el recurso humano es lo que falta en estos momentos en el departamento de sistemas, este tema está limitado por el presupuesto asignado para el año 2013.

- **Tiempo.** Otra de las grandes limitaciones es el tiempo que se le puede dedicar a este proyecto, porque estaría siendo desarrollado por dos personas que verían este tema en sus tiempos libres lo cual va a dilatar el desarrollo.
- **Costo.** Exclusivamente para este proyecto no hay ningún presupuesto asignado para el año 2013.





CAPÍTULO II

MARCO TEÓRICO CONCEPTUAL

2.1 Antecedentes y modelos de solución previos

Actualmente en las entidades del Sector Público, no existe un diseño de una metodología de certificación de productos Software, si bien es cierto las arquitecturas tecnológicas son de última generación, se está perdiendo la importancia de dejar huellas que permitan la continuidad de los proyectos, y dentro de algunos factores que afectan, esto es la situación política.

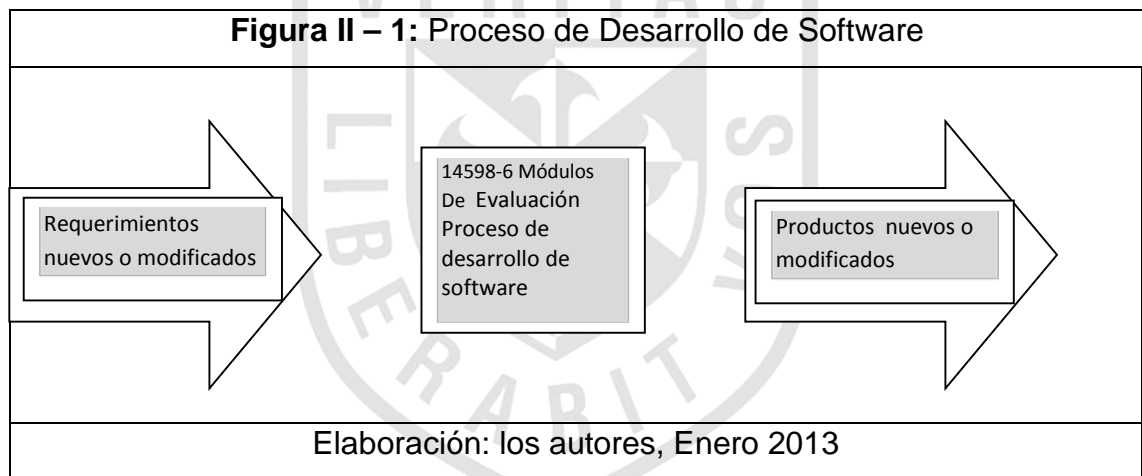
Para el desarrollo de esta tesis se toma como referencia a empresas del sector público, actualmente se está dando mucha importancia a los procesos que conforman la organización como punto de partida, de tal forma que todo esté alineado y represente mejor beneficio a la organización.

En este capítulo, y sin ánimo de abarcar todos los posibles modelos y metodologías que hayan desarrollado los agentes, se realizará una revisión de las aportaciones que se han considerado en estos últimos años como las más importantes.

En la práctica, muchas instituciones públicas complejas no están utilizando una Metodología de Certificación de Productos de Software que permitan minimizar los errores, de manera que se pueda tener una mayor seguridad de que las aplicaciones no fallarán salvo por el tema de código oculto u errores no previstos y fuera del alcance de la aplicación de la metodología propuesta.

2.1.1 Proceso de Desarrollo de Software

Un proceso de desarrollo de software se puede definir como un conjunto de actividades, métodos y prácticas que tienen como entrada los requerimientos del cliente, los cuales se utilizan para producir y mantener un software que los satisfaga.

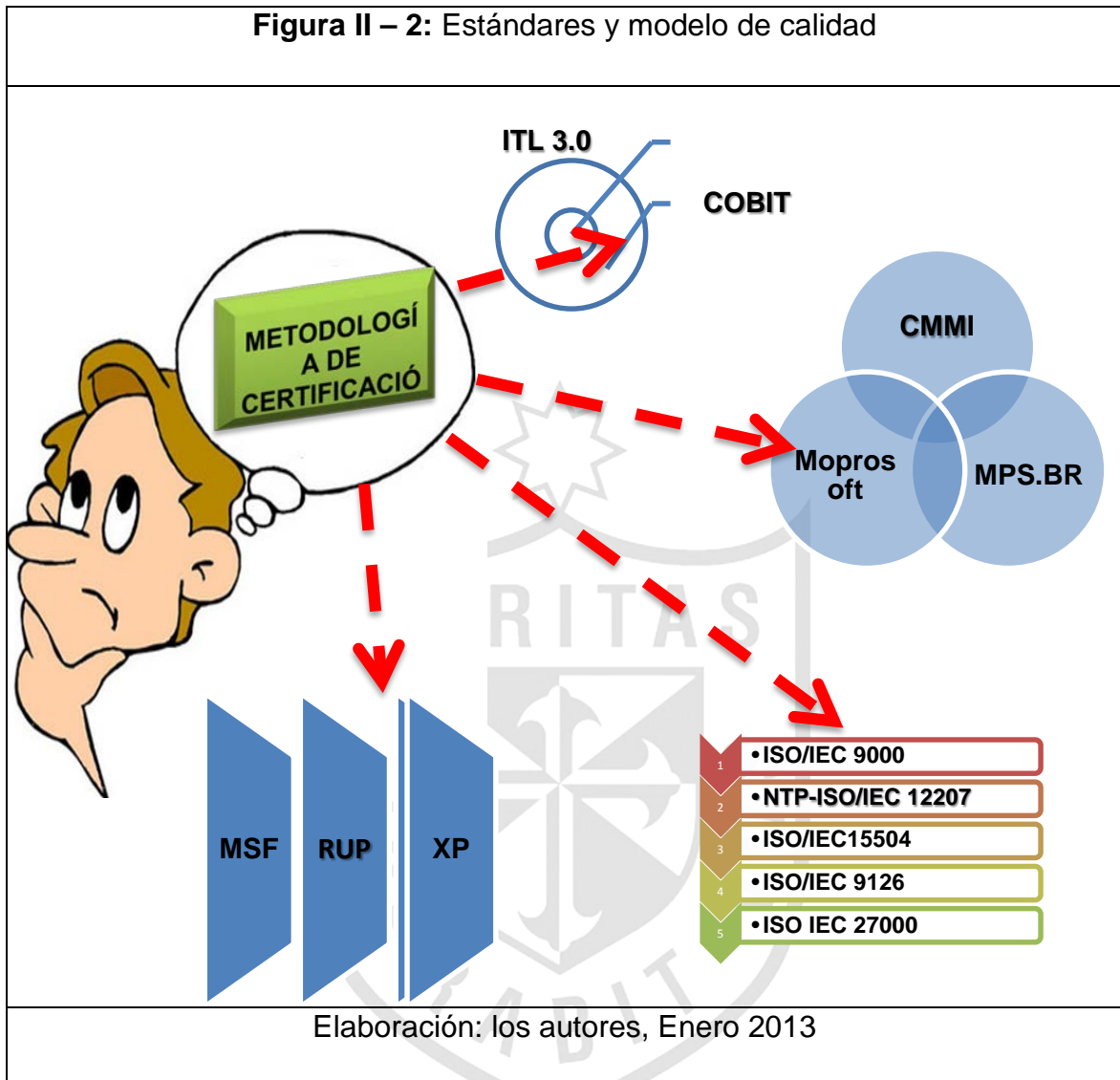


El proceso de desarrollo de software depende mucho de las habilidades de las personas involucradas y del modelo a seguir durante todas las etapas del desarrollo. A esto se le conoce como el ciclo de vida del software, que permite definir las herramientas y metodologías por utilizarse en las etapas del desarrollo y así obtener un software de calidad.

2.1.2 Estándares, metodologías y modelos de calidad

Los estándares, metodologías y modelos de calidad evaluados para el presente estudio son:

Figura II – 2: Estándares y modelo de calidad



2.1.3 Definiciones: ISO 9001:2000 NTP-ISO/IEC 12207: 2006, CMMI, SOA, MOPROSOFT, MPS.BR

Los Modelos de Calidad son herramientas que guían a las organizaciones a la mejora continua y elevan su competitividad. Son un conjunto de buenas prácticas que dicen **QUÉ** hacer pero no **CÓMO** hacerlas. Se utilizan en el ciclo de vida del software, y están enfocados a los procesos de gestión y desarrollo de proyectos. Por ello, los modelos de calidad contribuyen a generar una buena planificación, determinación de objetivos, formación y

coordinación de toda la organización y así llegar a consolidar un producto de calidad.

Uno de los modelos propuestos es el sistema de gestión de la calidad, que permite a las organizaciones convertirse en unidades más eficientes y efectivas. Asimismo, mejora la responsabilidad, motivación y compromiso del personal. Por este motivo, se está convirtiendo cada vez más en una condición necesaria la certificación del sistema de gestión de la calidad. El certificado demuestra a los clientes, quienes tienen en cuenta las consideraciones de la calidad que la empresa posee los sistemas necesarios para poder cumplir y satisfacer las obligaciones que asume frente a ellos.

Entre las organizaciones de certificaciones más importantes internacionalmente tenemos:

“ISO, International Organization for Standardization, establecida desde 1947, es una federación mundial que reúne organismos de normalización de más de 140 países. Su trabajo tiene como resultado acuerdos internacionales que se publican como estándares internacionales. La familia de estándares ISO representa un consenso internacional en buenas prácticas de gestión de la calidad con el objetivo de asegurar que las organizaciones ofrezcan productos o servicios que cumplan los requisitos de calidad vitales para los clientes.”(ISO 2004)

Entre sus certificaciones más relevantes de la ISO, con respecto a la gestión de calidad tenemos:

La norma ISO 9001:2000 “está orientada a la satisfacción de las necesidades del cliente y a la gestión de los procesos para lograr este objetivo, facilitando la toma de decisiones sobre la base de

información objetiva sobre el desempeño de los procesos y su impacto en las características de calidad, sirviendo de plataforma para el desarrollo de servicios, actividades y procesos según las necesidades de los clientes.” (SUNAT 2006)

Según la International Organization for Standardization:

“ISO 9001:2000 is the standard that provides a set of standardized requirements for a quality management system, regardless of what the user organization does, its size, or whether it is in the private, or public sector. It is the only standard in the family against which organizations can be certified – although certification is not a compulsory requirement of the standard.” (ISO 2008)

➤ **NORMA ISO 9001:2000**

Es un sistema de gestión de la calidad que, a través del cumplimiento de las buenas prácticas describe, permite mejorar la calidad y la satisfacción del cliente, y el funcionamiento de una organización mediante la mejora continua de sus procesos. Esta tendencia es global, y debe ser la aspiración de toda empresa competitiva, que quiera permanecer y sobrevivir en el exigente mercado actual.

➤ **NORMA TÉCNICA PERUANA NTP-ISO/IEC 12207:2006¹**

La **NTP-ISO/IEC 12207:2006**, establece un marco de referencia común para los procesos del ciclo de vida del software, con una terminología bien definida a la que puede hacer referencia la industria del software.

¹ URL: www.bvindicopi.gob.pe/normas/isoiec12207.pdf

Contiene procesos, actividades y tareas para aplicar durante la adquisición de un sistema que contiene software, un producto software puro o un servicio software, y durante el suministro, desarrollo, operación y mantenimiento de productos software.

Esta norma incluye también un proceso que puede emplearse para definir, controlar y mejorar los Procesos del ciclo de vida del software.

- ✓ La norma establece la arquitectura de alto nivel del ciclo de vida del software: los procesos y sus interrelaciones.
- ✓ El ciclo comienza con la idea y termina con la retirada del software.
- ✓ Se derivan los procesos considerando:
 - Modularidad del proceso: un proceso individual se dedica solamente a una única función. Alta cohesión y bajo acoplamiento.
 - Responsabilidad del proceso: un proceso individual es responsabilidad de una de las partes.
 - Parte: una organización (total o parcial) que entra en un contrato. La organización puede variar de una a muchas personas.
- ✓ Los procesos se organizan por actividades.
- ✓ Las actividades se implementan a través de las tareas.
- ✓ Una tarea es:
 - Un conjunto elemental o atómico de acciones.
 - Una tarea consume entradas (datos, información, control) y produce salidas (datos, información, control).
- ✓ Se consideran tres tipos de procesos:
 - **Primarios:** dan servicio a las partes principales, que son:
 - Adquiriente: el que adquiere u obtiene un sistema, producto o servicio software de un proveedor.
 - Desarrollador: organización que lleva a cabo actividades de desarrollo durante el proceso de ciclo de vida.
 - Operador: organización que opera el sistema

- Proveedor: organización que es contratada por el adquiriente para el suministro de un sistema, producto o servicio software bajo los términos de un contrato.
- Responsable de mantenimiento: organización que lleva a cabo tareas de mantenimiento.

Vemos que la norma se refiere a un sistema, producto o servicio software, esto es, considera un producto o servicio software como parte de un sistema más amplio (cuando esto es aplicable). Conforme a esto, veremos que los procesos contienen actividades y tareas relacionadas con el sistema.

a) Proceso de adquisición

Este proceso define las actividades y tareas del adquiriente. En este proceso se identifica la necesidad de adquirir, desarrollar o adaptar un sistema, producto o servicio software, preparar una solicitud y seleccionar un proveedor.

Actividades:

- Inicio
- Preparación de solicitud de propuestas
- Preparación y actualización del contrato
- Seguimiento del proveedor
- Aceptación y finalización

b) Proceso de suministro

Este proceso contiene las actividades y tareas del proveedor. Tiene tareas para determinar los procedimientos y recursos necesarios para gestionar el proyecto.

Actividades:

- Inicio
- Preparación de la respuesta

- Contrato
- Planificación
- Ejecución y control
- Revisión y evaluación
- Entrega y finalización

c) Proceso de desarrollo

Las actividades y tareas de este proceso son responsabilidad del desarrollador. Contiene las actividades de ingeniería de software para el producto software. Puede contener actividades a nivel de sistema si está especificado en el contrato.

Actividades:

- Implementación del proceso
- Análisis de requerimientos del sistema
- Diseño de la arquitectura del sistema
- Análisis de requerimientos de software
- Diseño de la arquitectura del software
- Diseño detallado del software
- Codificación y pruebas del software
- Integración del software
- Pruebas de calificación del software
- Integración del sistema
- Pruebas de calificación del sistema
- Instalación del software
- Apoyo a la aceptación de software

d) Proceso de operación

Cubre la operación del producto software y apoyo a los usuarios. Las actividades y tareas hacen referencia al sistema.

Actividades

- Implementación del proceso

- Pruebas de operación
- Operación del sistema
- Soporte al usuario

e) Proceso de mantenimiento

Modificar el producto software preservando su integridad. Incluye la migración y retirada del producto.

Actividades

- Implementación del proceso
- Análisis de problemas y modificaciones
- Implementación de las modificaciones
- Revisión/aceptación del mantenimiento
- Migración
- Retirada de software

- **Soporte.** El estándar contiene un grupo de 8 procesos de soporte, cuyo objetivo es, valga la redundancia brindar soporte y apoyar a los procesos primarios, teniendo como objetivo el de contribuir a la calidad y éxito del proyecto. Estos procesos pueden ser invocados tanto por procesos primarios como por otro proceso de soporte. El proceso de soporte comienza con un preámbulo, al que le pueden seguir un conjunto de acciones de nivel corporativo (no obligatorias), y continúa con un conjunto de actividades y tareas propias del proceso. Los 8 procesos de soporte son:

a) Documentación

El propósito de este proceso es obtener y persistir información. Este proceso define actividades las cuales planean, diseñan, desarrollan, editan, distribuyen y mantienen los documentos requeridos por todos los actores involucrados en el sistema (gerentes, ingenieros, usuarios).

Actividades:

- Implementación del proceso
- Diseño y desarrollo
- Producción
- Mantenimiento

b) Gestión de Configuración

El propósito de este proceso es identificar, definir y versionar, mediante líneas bases, los elementos del sistema, así como también asegurar que los elementos que pertenecen a la configuración estén completos y correctos, de controlar su manejo, persistencia y entrega de los mismos.

Actividades:

- Implementación del proceso
- Identificación de la configuración
- Control de la configuración
- Determinación del estado de la configuración
- Evaluación de la configuración
- Gestión de liberaciones y entregas

c) Aseguramiento de la Calidad

El propósito de este proceso es proveer de mecanismos para objetiva e independientemente, asegurar que los productos y/o servicios cumplan con los estándares y requerimientos establecidos, y que el desarrollo de otros procesos se alinie lo más posible a lo planificado originalmente.

Actividades:

- Implementación del proceso
- Aseguramiento del producto
- Aseguramiento del proceso
- Aseguramiento del sistema de calidad

d) Verificación

El propósito de este proceso es proveer las evaluaciones referentes a la verificación de un producto o servicio de una actividad dada.

Actividades:

- Implementación del proceso
- Verificación

e) Validación

El propósito de este proceso es determinar si un sistema ya construido cumple con las especificaciones y requerimientos para los cuales fue realizado.

Actividades:

- Implementación del Proceso
- Validación

f) Revisión conjunta

El propósito de este proceso es proveer de un marco que favorezca la integración entre inspector e inspeccionado.

Actividades:

- Implementación del proceso
- Revisiones de la gestión del proyecto
- Revisiones técnicas

g) Auditoría

El propósito de este proceso es proveer de un marco adecuado para establecer auditorías formales y contractuales sobre un determinado producto o servicio provisto.

Actividades:

- Implementación del proceso

- Auditoria

h) Resolución de problemas

El propósito de este proceso es proveer de mecanismos para la creación de procesos capaces de resolver problemas y tomar acciones correctivas para remover nuevos problemas detectados.

Actividades:

- Implementación del proceso
- Solución de problemas

- **Organizacionales.** Los procesos de la organización tienen como propósito establecer, controlar y mejorar otros procesos. Generalmente abarcan varios procesos juntos o son más bien genéricos y cada proceso los implementa y ajusta de acuerdo con sus necesidades.

Se llaman procesos organizacionales porque sus actividades y tareas son responsabilidad de la organización que usa dicho proceso. Es esta organización además quien debe asegurarse de que el proceso exista y esté operativo. El alcance de los procesos organizacionales normalmente no solo trasciende un proyecto en particular, sino que abarca a toda la organización. El estándar identifica cuatro procesos organizacionales:

a) Proceso de Gestión

El propósito de este proceso es proveer actividades y tareas genéricas que pueden emplearse y ajustarse para gestionar otros procesos. La norma pone al gerente como rol responsable de dicho proceso.

Cualquier proceso que requiera gestión, implementará y ejecutará el proceso de gestión, el mismo se adapta a procesos primarios

como actividades, por ejemplo, gestión del proyecto, proceso de adquisición, proceso de mantenimiento. Todos estos procesos implementan una instancia particular del proceso de gestión, tan compleja como sea necesario.

Actividades:

- Inicio y definición de alcance
- Planificación
- Ejecución y control
- Revisión y evaluación
- Terminación

b) Proceso de infraestructura

El propósito de este proceso es definir las actividades necesarias para establecer y mantener la infraestructura necesaria para poder ejecutar correctamente cualquier proceso del sistema, ya sea primario o de soporte. Dentro de la infraestructura entran aspectos como software, hardware, estándar, herramientas, técnicas, y facilidades.

Este proceso está presente en todos los procesos de la organización. Cuenta con tres actividades, implementación del proceso, establecimiento de la infraestructura y mantenimiento de la infraestructura.

Actividades:

- Implementación del proceso
- Establecimiento de la infraestructura
- Mantenimiento de la infraestructura

c) Proceso de mejora

Este proceso también está presente en todos los procesos de la organización, su propósito es proveer de actividades básicas y de alto nivel para establecer, evaluar, medir y mejorar un proceso de

ciclo de vida del software. El mismo está basado en el ciclo de Deming “Plan-Do-Check-Act”, plantea las dos (2) últimas actividades, dejando la planificación a los propios procesos.

Cuenta con tres actividades las cuales cubren el establecimiento del proceso, evaluación del proceso y mejora del mismo. Estas actividades se establecen a un nivel organizacional de forma que la mejora sea global en todos los proyectos.

Actividades:

- Establecimiento del proceso
- Evaluación del proceso
- Mejora del proceso

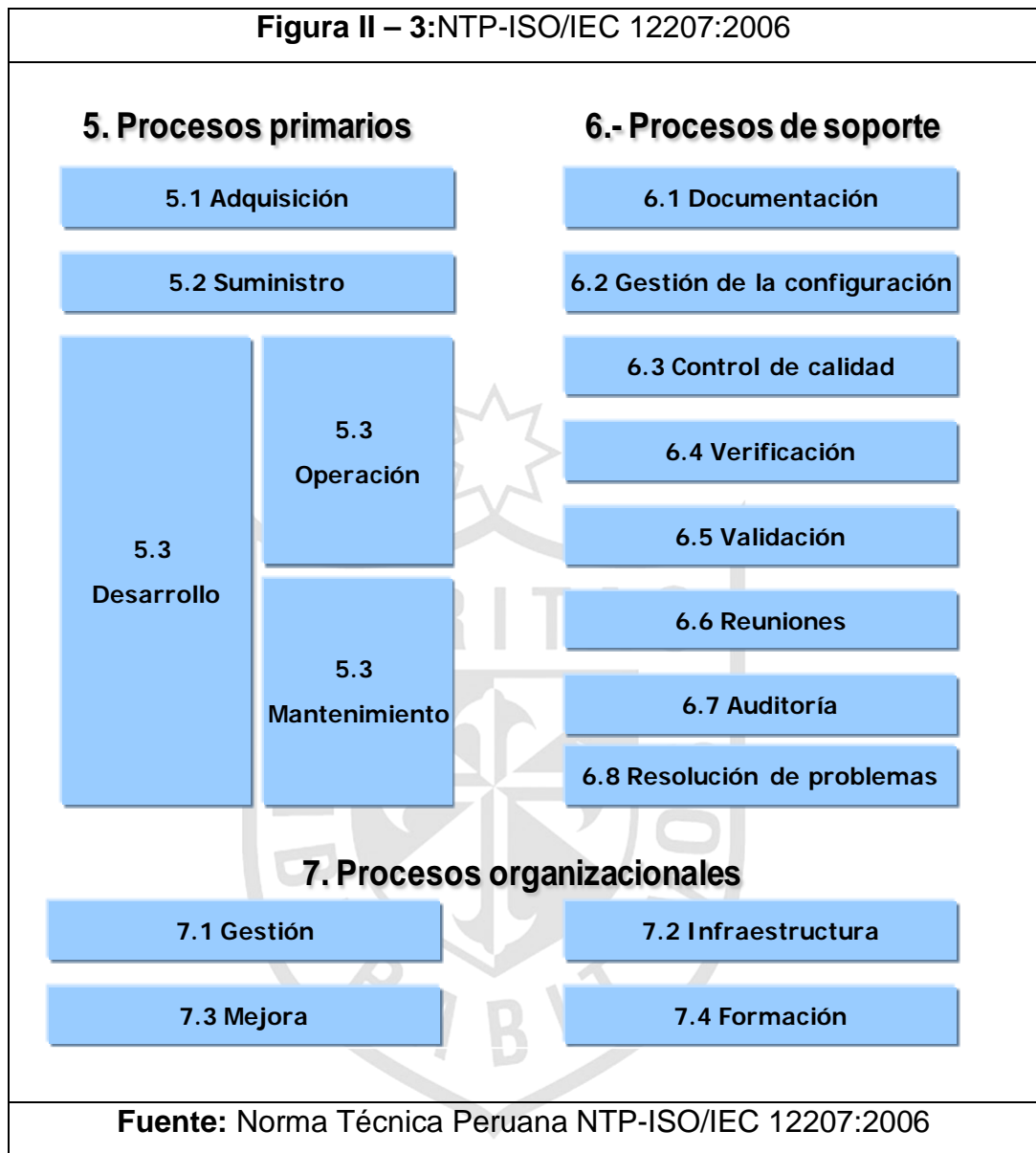
d) Proceso de formación

El propósito de este proceso es proporcionar y mantener al personal capacitado. Gran parte de la operativa de la organización como la ejecución de las tareas depende de un personal bien capacitado, tanto en aptitud como técnicamente.

Para lograr esto, existe el proceso de Recursos Humanos, que cuenta con tres actividades: implementación del proceso, desarrollo del material e implementación del plan. El estándar hace un énfasis en una buena planificación e implementación de la capacitación de forma de tener personal capacitado lo antes posible.

Actividades:

- Implementación del proceso
- Desarrollo del Material de formación
- Implementación del Plan de formación



➤ **CAPABILITY MATURITY MODEL INTEGRATION - CMMI**

El CMMI es un modelo de calidad del software que clasifica las empresas en niveles de madurez. Estos cinco niveles sirven para conocer la madurez de los procesos que se realizan para producir software.

El CMMI es un modelo creado por el SEI (Software Engineering Institute) de la CMU (Carnegie Mellon University) ha pedido por el DoD (Departamento de Defensa de los EEUU).

Permiten obtener un diagnóstico preciso de la madurez de los procesos relacionados con las tecnologías de la información de una organización y describe las tareas que se tienen que llevar a cabo para mejorar los procesos.

Abarca 4 categorías de procesos: Gestión de Procesos, Gestión de Proyectos, Ingeniería, Soporte.

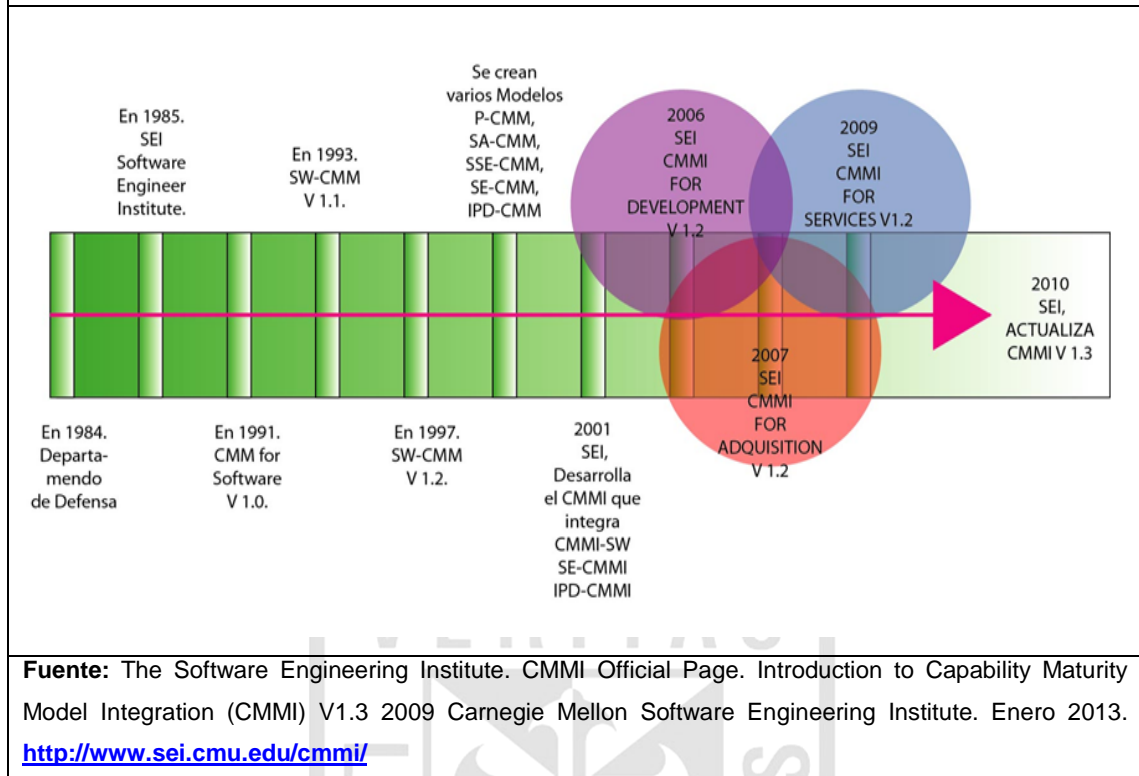
Es la aplicación de TQM (Total Quality Management) a la Ingeniería de Software y a la ingeniería de Sistemas.

Asimismo, el modelo de calidad para mejorar los procesos de software más reconocido a nivel mundial es el CMMI:

“Capability Maturity Model® Integration (CMMI) is a process improvement approach that provides organizations with the essential elements of effective processes. It can be used to guide process improvement across a project, a division, or an entire organization. CMMI helps integrate traditionally separate organizational functions, set process improvement goals and priorities, provide guidance for quality processes, and provide a point of reference for appraising current processes.” (SOFTWARE ENGINEERING INSTITUTE 2008).

Por lo tanto, el CMMI es un modelo de referencia o prácticas maduras utilizadas para mejorar la capacidad de un proceso o grupo de procesos de software con el fin de alcanzar la mejora continua en la organización.

Figura II – 4: Historia y Evolución



Los niveles CMMI. Son cinco:

- ✓ **Inicial o nivel 1 CMMI.** Este es el nivel en donde están todas las empresas que no tienen procesos. Los presupuestos se disparan, no es posible entregar el proyecto en fechas, se tiene que quedar durante noches y fines de semana para terminar un proyecto. No hay control sobre el estado del proyecto, el desarrollo del mismo es completamente opaco, no se sabe lo que pasa.
- ✓ **Repetible o nivel 2 CMMI.** Quiere decir que el éxito de los resultados obtenidos se puede repetir. La principal diferencia entre este nivel y el anterior es que el proyecto es gestionado y controlado durante el desarrollo del mismo. Este no es opaco y se puede saber el estado del

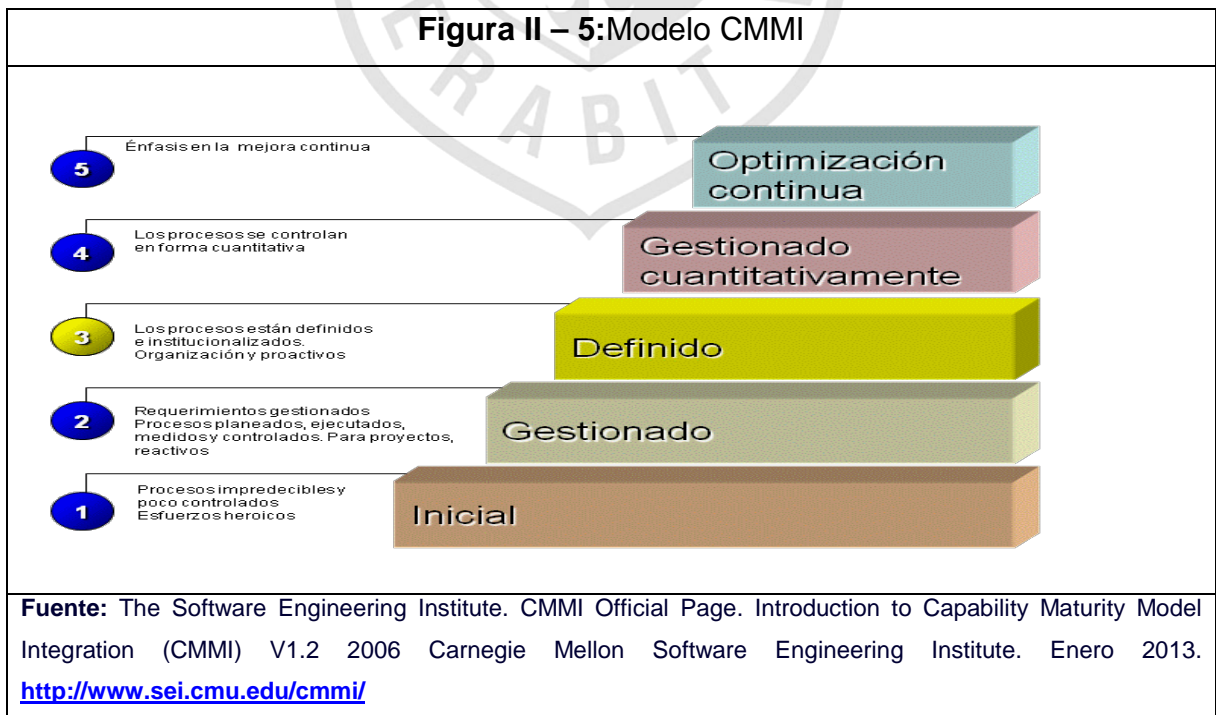
proyecto en todo momento. Los procesos que hay que implantar para alcanzar este nivel son:

- Planeamiento de proyecto – REQM
 - Control y seguimiento del proyecto – PP
 - Gestión de acuerdo con proveedores – PMC
 - Gestión de requerimientos – REQM
 - Aseguramiento de la calidad de productos y procesos – PPQA
 - Medición y Análisis – MA
 - Gestión de la configuración – CM
- ✓ **Definido o nivel 3 CMMI.** Resumiéndolo mucho, alcanzar este nivel significa que la forma de desarrollar proyectos (gestión e ingeniería) está definida, quiere decir que están establecida, documentada y que existen métricas (obtención de datos objetivos) para la consecución de objetivos concretos. Los procesos que hay que implantar para alcanzar este nivel son:
- Gestión integrada de proyectos – IPM
 - Gestión integrada de proveedores – ISM
 - Equipos integrados – IT
 - Gestión de riesgos – RSKM
 - Desarrollo de requerimientos – RD
 - Solución técnica – TS
 - Integración del producto – PI
 - Verificación – VER
 - Validación - VAL
 - Entrenamiento organizacional – OT
 - Foco en el proceso organizacional – OPF
 - Definición de proceso organizacional – OPD
 - Entorno organizativo para la integración – OEI
 - Análisis de decisiones y soluciones – DAR

- ✓ **Cuantitativamente Gestionado o nivel 4 CMMI.** Los proyectos usan objetivos medibles para alcanzar las necesidades de los clientes y la organización. Se usan métricas para gestionar la organización. Los procesos que hay que implantar para alcanzar este nivel son:
 - Desempeño de Proceso Organizacional – OPP
 - Gestión Cuantitativa de Proyecto – QPM

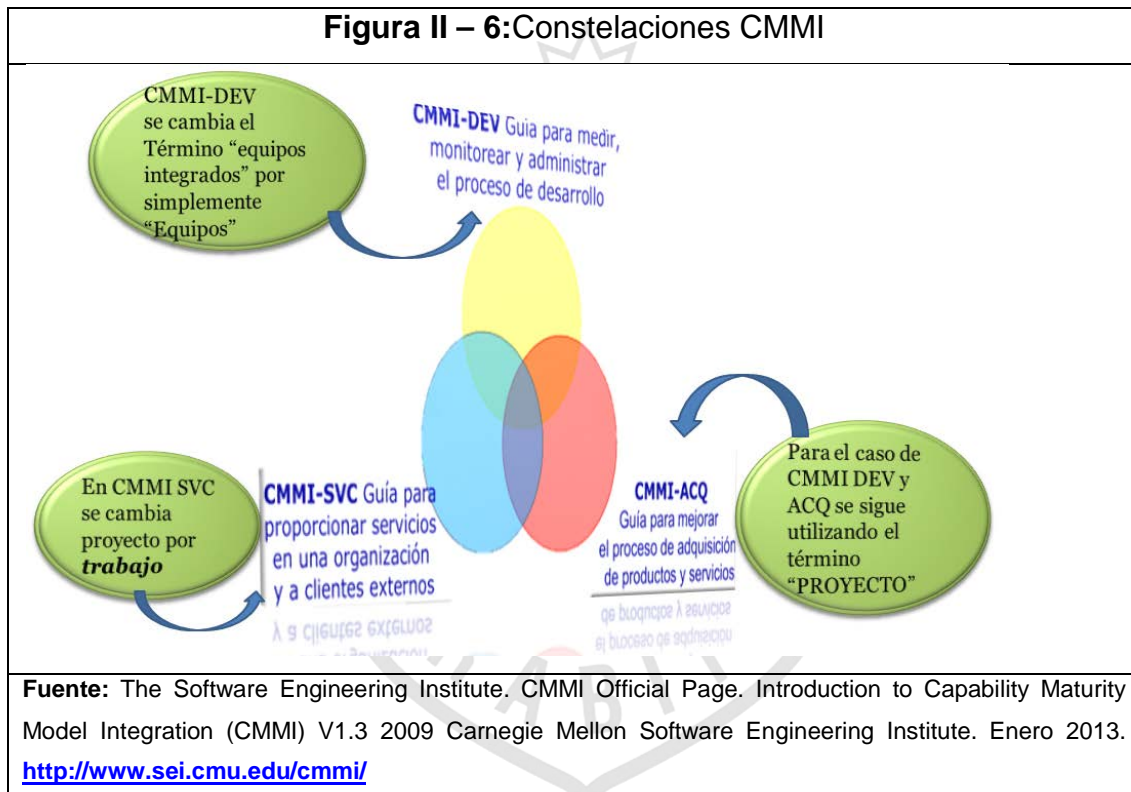
- ✓ **Optimizado o nivel 5 CMMI.** Los procesos de los proyectos y de la organización están orientados a la mejora de las actividades. Mejoras incrementales e innovadoras de los procesos que mediante métricas son identificadas, evaluadas y puestas en práctica. Los procesos que hay que implantar para alcanzar este nivel son:
 - Innovación organizacional y despliegue – OID
 - Análisis causal y soluciones – CAR
 -

Normalmente las empresas que intentan alcanzar los niveles 4 y 5 los realizan simultáneamente ya que están muy relacionados.



CMMI 1.3 Constelaciones

Las constelaciones en el modelo Capability Maturity Model Integration (CMMI) aparecen a partir de la versión 1.2 publicada en Agosto del 2006, como CMMI-DEV (CMMI for Development). Una constelación es una colección de componentes utilizados para construir modelos, materiales de capacitación y evaluación en un área de interés. Hasta la fecha existen tres constelaciones publicadas:



- ✓ **CMMI V.13 FOR DEVELOPMENT (CMMI-DEV):** Modelo para empresas que desarrollan y mantienen productos de software, ya sea para su uso o para la venta (fábrica de software o a la medida). Publicada en agosto del 2006.

Proporciona una oportunidad para evitar o eliminar estos nichos y barreras. CMMI para Desarrollo consta de buenas prácticas que tratan

las actividades de desarrollo aplicadas a productos y servicios. Aborda las prácticas que cubren el ciclo de vida del producto desde la concepción hasta la entrega y el mantenimiento. El énfasis está en el trabajo necesario para construir y mantener el producto completo.

CMMI-DEV contiene 22 áreas de proceso. De esas áreas de proceso, 16 son áreas de proceso base, 1 es un área de proceso compartida y 5 son áreas de proceso específicas de desarrollo.²

Todas las prácticas del modelo CMMI-DEV se centran en las actividades de la organización desarrolladora. Cinco áreas de proceso se centran en las prácticas específicas del desarrollo: tratando desarrollo de requisitos, solución técnica, integración del producto, verificación y validación.

El modelo se especifica que un proyecto u organización debe tener procesos que las prácticas de desarrollo relacionados con la dirección. Para determinar si estos procesos están en su lugar, los mapas de un proyecto u organización de sus procesos a las áreas de proceso en este modelo.

El mapeo de los procesos de las áreas de proceso permite a la organización realizar un seguimiento de sus progresos en relación con el modelo CMMI-DEV

²Un área de proceso base es un área de proceso que es común a todos los modelos CMMI. Un área de proceso compartida está presente en al menos dos modelos CMMI, pero no en todos

Categorías

Gestión de Proyectos	Gestión de Procesos
<ul style="list-style-type: none"> ▪ Gestión del proyecto integrado(IPM) ▪ Project Monitoring and Control (PMC) ▪ Planificación del proyecto (PP) ▪ Gestión Cuantitativa del proyecto (QPM) ▪ Gestión de Requerimientos (REQM)³ ▪ Gestión de Riesgos (RSKM) ▪ (+) Supplier Agreement Management (SAM) 	<ul style="list-style-type: none"> ▪ Innovación y Despliegue Organizativo (OID) ▪ Definición del Proceso Organizativo (OPD) ▪ Enfoque en el proceso Organizativo (OPF) ▪ Formación Organizativa (OT): ▪ Rendimiento del Proceso Organizativo (OPP) ▪ Formación Organizativa (OT)
Ingeniería	Soporte
<ul style="list-style-type: none"> ▪ Integración de Producto(PI) ▪ Requisitos de Desarrollo(RD) ▪ Solución Técnica(TS) ▪ Validación (VAL) ▪ Verificación (VER) 	<ul style="list-style-type: none"> ▪ Análisis y resolución causal (CAR). ▪ Gestión de la configuración (CM). ▪ Análisis y resolución de decisiones (DAR). ▪ Medición y Análisis (MA). ▪ Aseguramiento de la calidad del producto y proceso (PPQA).

³CMMI 1.3, REQM fue movido de "Ingeniería a "Gestión de Proyectos."

Características

- Para desarrolladores de productos y servicios.
 - Desarrollo y mantenimiento de productos y servicios
 - Cubre de manera implícita la parte de servicios en "producto"
 - Contiene áreas de proceso en la categoría de la ingeniería
 - Establece RD a nivel 3 en la representación escalonada
 - Categorizan REQM como área de proceso de ingeniería
 - Contiene productos de trabajo típicos
- ✓ **CMMI V.13– CMMI FOR ACQUISITION (CMMI-ACQ).** Publicada en noviembre del 2007 y sirve como guía para mejorar el proceso de adquisición de productos y servicios.

Es un modelo de mejores prácticas que pueden ayudar a mejorar las relaciones con sus proveedores, ayudando a mejorar sus propios procesos. Se puede utilizar para aumentar su control de los proyectos, gestionar mejor el abastecimiento global de productos y servicios, y con más éxito adquirir soluciones que satisfagan las necesidades de su organización.

Características

- Adquisición de Requisitos para el Desarrollo (ARD)
 - Desarrollo de Solicitud y Acuerdo con Proveedores (SSAD)
 - Contrato de Gestión (AM)
 - Adquisición de Gestión Técnica (ATM)
 - Verificación de Adquisición (AVER)
 - Validación de Adquisición (AVAL)
- ✓ **CMMI V.13– CMMI FOR SERVICE (CMMI-SVC).** Publicada en febrero del 2009 y sirve como guía para proporcionar servicios internos en una organización y a clientes externos.

CMMI-SVC modelo que proporciona una guía para aplicar las mejores prácticas de CMMI en un proveedor de servicios. Las mejores prácticas en el modelo enfocado hacia las actividades de prestación de servicios de calidad a clientes y usuarios finales. CMMI-SVC integra los cuerpos de conocimiento que son esenciales para un proveedor de servicios.

Características

- Cubre la administración, establecimiento y liberación de servicios, considerados como intangibles o productos no almacenables.
- Integra 6 nuevas áreas de proceso, además de ampliar REQM y utilizar el SAN de CMMI DEV.
- Considera un área de proceso opcional que amplía el alcance del modelo (SSD – SERVICE SYSTEM DEVELOPMENT).

Categorías / Nivel	Gestión proceso	Gestión proyectos	Definición entrega del Servicio	Soporte
5: Optimizado	OID			CAR
4: Administrativo Cuantitativa	OPP	QPM		
3: Definido	OPF, OPD OT	IPM, RSKM CAM, SCON	IRP, SST STSM,SSD	DAR
2: Gestionado		PP, PMC REQM,SAM	SD	MA, PPQA,CM

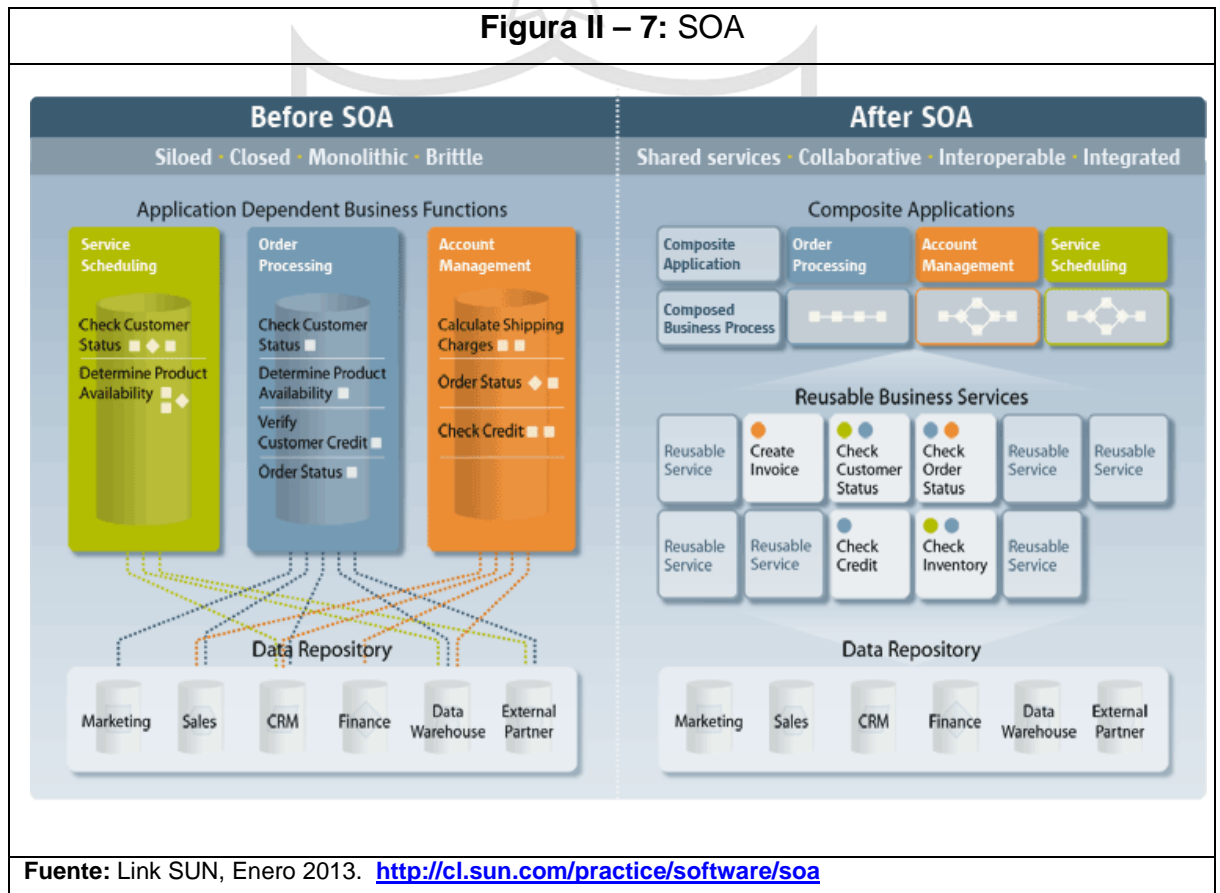
➤ ARQUITECTURA ORIENTADA A SERVICIOS - SOA

La Arquitectura SOA establece un marco de diseño para la integración de aplicaciones independientes de manera que desde la red pueda accederse a sus funcionalidades, las cuales se ofrecen como servicios. La forma más

habitual de implementarla es mediante Servicios Web, una tecnología basada en estándares e independiente de la plataforma, con la que SOA puede descomponer aplicaciones monolíticas en un conjunto de servicios e implementar esta funcionalidad en forma modular.

Con una Arquitectura Orientada a Servicios (SOA), los usuarios ya no tienen que iniciar sesión en varios sistemas, buscar los datos relevantes e integrar los resultados manualmente. Los datos de las actividades de los procesos de negocios se entregan como un servicio integrado, en una sola aplicación, en una sola pantalla, con un solo inicio de sesión.

Figura II – 7: SOA



Fuente: Link SUN, Enero 2013. <http://cl.sun.com/practice/software/soa>

✓ **Beneficios de SOA**

Los beneficios de SOA para una organización se plasman en dos niveles distintos: el del usuario corporativo y a nivel de la organización de IT. Desde el punto de vista de la empresa, SOA permite el desarrollo de una

nueva generación de aplicaciones dinámicas que resuelven una gran cantidad de problemas de alto nivel, fundamentales para el crecimiento y la competitividad. Las soluciones SOA permiten entre otras cosas:

- **Mejorar la toma de decisiones.** Al integrar el acceso a los servicios e información de negocio dentro de un conjunto de aplicaciones dinámicas compuestas, los directivos disponen de más información y de mejor calidad (más exacta y actualizada). Las personas, procesos y sistemas que abarcan múltiples departamentos pueden introducirse de forma más directa en una panorámica unificada, lo que permite conocer mejor los balances de costes y beneficios que se producen en las operaciones de negocio que se realizan a diario. Y al disponer de mejor información en un tiempo menor, las organizaciones pueden reaccionar de manera más ágil y rápida cuando surgen problemas o cambios.
- **Mejorar la productividad de los empleados.** Un acceso óptimo a los sistemas y la información y la posibilidad de mejorar los procesos permiten a las empresas aumentar la productividad individual de los empleados. Estos pueden dedicar sus energías a los procesos importantes, los que generan valor añadido y a actividades de colaboración, semiestructuradas, en vez de aceptar las limitaciones y restricciones impuestas por los sistemas de IT rígidos y monolíticos. Más aún: puesto que los usuarios pueden acceder a la información en los formatos y modalidades de presentación (web, cliente avanzado, dispositivo móvil), que necesitan, su productividad se multiplica en una gran cantidad de escenarios de uso, habituales o nuevos.
- **Potenciar las relaciones con clientes y proveedores.** Las ventajas de SOA trascienden las fronteras de la organización. Los beneficios

que ofrece SOA trascienden los límites de la propia organización. Los procesos de fusión y compra de empresas se hacen más rentables al ser más sencilla la integración de sistemas y aplicaciones diferentes.

La integración con partners comerciales y la optimización de los procesos de la cadena de suministro son, bajo esta perspectiva, objetivos perfectamente asequibles. Con SOA se puede conseguir mejorar la capacidad de respuesta a los clientes, habilitando por ejemplo portales unificados de servicios. Si los clientes y proveedores externos pueden disponer de acceso a aplicaciones y servicios de negocio dinámicos, no solamente se permite una colaboración avanzada, sino que se aumenta la satisfacción de clientes y proveedores.

SOA permite flexibilizar los procesos críticos de compras y gestión de pedidos habilitando modalidades como la subcontratación de ciertas actividades internas- superando las restricciones impuestas por las arquitecturas de IT subyacentes, y con ello consiguiendo un mejor alineamiento de los procesos con la estrategia corporativa.

➤ **MODELO DE PROCESOS PARA LA INDUSTRIA DEL SOFTWARE - MOPROSOFT**

Modelo para la mejora y evaluación de los procesos de desarrollo y mantenimiento de sistemas y productos de software. Desarrollado por la Asociación Mexicana para la Calidad en Ingeniería de Software a través de la Facultad de Ciencias de la Universidad Nacional Autónoma de México (UNAM) y a solicitud de la Secretaría de Economía para obtener una norma mexicana que resulte apropiada a las características de tamaño de la gran mayoría de empresas mexicanas de desarrollo y mantenimiento de software. Por otra parte, el modelo de procesos Moprosoft elaborado en México dice:

“Es un modelo de procesos para la industria de software nacional, que fomenta la estandarización de su operación a través de la incorporación de las mejores prácticas en gestión e ingeniería de software. La adopción del modelo permite elevar la capacidad de las organizaciones que desarrollan o mantienen software para ofrecer servicios con calidad y alcanzar niveles internacionales de competitividad.” (UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO 2005)

Según la Industria Mexicana de software el modelo de procesos Moprosoft es un modelo de calidad dirigido a las pequeñas y medianas empresas de desarrollo de software de México, que tiene como fin estandarizar sus operaciones a través de la introducción de las mejores prácticas, alcanzando niveles internacionales en capacidad de procesos⁴.

La calidad en el proceso de producción y en los productos de software es una exigencia creciente, dado que cada vez es más amplio el uso del software en los procesos críticos de las organizaciones.

Moprosoft es un modelo de procesos de Software desarrollado por un grupo de especialistas en calidad bajo la dirección de la Dra. Hanna Oktaba, presidenta de la AMCIS⁵, conjuntamente con la Facultad de Ciencias de la Universidad Nacional Autónoma de México (AMCIS) a solicitud de la Secretaría de Economía de México para ser usada como norma para la industria de desarrollo y mantenimiento de software.

Los criterios aplicados, para la elaboración de este modelo de proceso, son:

- ✓ La estructura de procesos resultante debe ser acorde con la estructura generalmente empleada por las organizaciones de la industria del software (alta dirección, gestión y operación).

⁴Cfr Software.net.mx, sitio oficial de la Industria Mexicana avalado por la Secretaría de Economía de México

⁵Asociación Mexicana para la Calidad en Ingeniería de Software

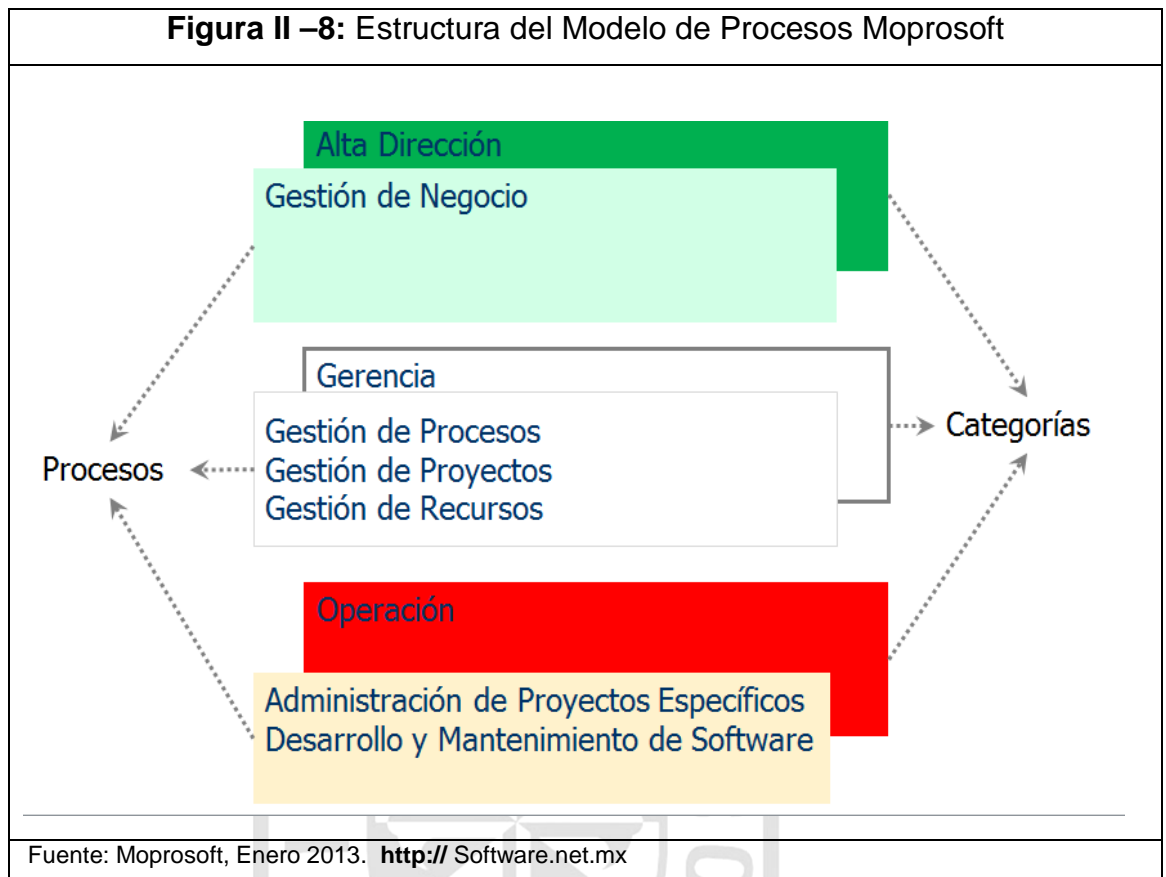
- ✓ La alta dirección tiene un papel importante a través de la planificación estratégica. Debe actuar como promotor del buen funcionamiento de la organización a través de su implicación en la revisión y mejora continua del modelo.
- ✓ En el modelo, se considera a la gestión como proveedora de recursos, procesos y proyectos; así como responsable de la vigilancia del cumplimiento de los objetivos estratégicos de la organización.
- ✓ En el modelo, se considera a la operación como ejecutora de los proyectos de desarrollo y mantenimiento de software.
- ✓ En el modelo, se integra con claridad y consistencia los elementos indispensables para la definición de los procesos y las relaciones entre ellos.
- ✓ En el modelo, se integran los elementos para realizar la administración de proyectos desde un sólo proceso.
- ✓ En el modelo, se integran los elementos para realizar la ingeniería de productos de software en un único marco que incluya los procesos precisos de soporte (verificación, validación, documentación y control de la documentación)
- ✓ En el modelo, se destaca la importancia de la gestión de recursos, con especial relevancia en aquellos que componen el conocimiento de la organización: productos generados por proyectos, datos de los proyectos, mediciones, documentación de procesos y datos cosechados a partir del uso y de las lecciones aprendidas.
- ✓ MOPROSOFT⁶ se basa en los modelos de procesos ISO 9001:2000, en las áreas de procesos de los niveles 2 y 3 de CMM-SW: CMMI v.1.2⁷., en el marco general ISO/IEC15504 y en prácticas y conceptos de PMBOKY SWEBOK.

⁶Modelo de Procesos para la Industria de Software Versión 1.3

⁷CMMI 1.2 (Capability Maturity Model Integration)

Procesos:

- Categoría alta dirección (DIR)
 - Gestión de Negocio
 - Categoría Gerencia (GER)
 - Gestión de Procesos
 - Gestión de Proyectos
 - Gestión de Recursos
 - Recursos Humanos y Ambiente de Trabajo
 - Bienes Servicios e Infraestructura
 - Conocimiento de la Organización
 - Categoría Operación (OPE)
 - Administración de Proyectos Específicos
 - Desarrollo y Mantenimiento de Software
- ✓ **Empresas Certificadas en Perú.** ACKLIS SAC y PUCP ejecutan el proyecto Moprosoft financiado por el Fancito (Fondo de Ciencia y Tecnología), que tiene como objetivo elaborar un modelo de evaluación para Moprosoft que cumpla con el estándar internacional ISO/IEC 15504 Evaluación de Procesos. Este modelo ya ha sido terminado y se emplea en las certificaciones realizadas por el Organismo de Certificación del Instituto para la Calidad de la Pontificia Universidad Católica del Perú. La primera empresa en obtener la certificación en Moprosoft en nuestro país es ASIS TP, dedicada al desarrollo de software en especial en el sector de Telecomunicaciones.



➤ **MEJORA DE PROCESO DEL SOFTWARE BRASILEÑO. INSTITUTO DEL SOFTWARE - MPS.BR**

Es un programa de Mejora de Proceso del Software Brasileño. Se encuentra en desarrollo desde el año 2003. Coordinado por la Asociación para Promoción del Software Brasileño (SOFTEX). Ministerio de Ciencia y Tecnología (MCT), Financiera de Estudios y Proyectos (FINEP) y el Banco Interamericano de Desarrollo (BID). La coordinación del programa MPS.BR cuenta con dos estructuras de apoyo: FCC (Foro de Acreditación y Control), ETM (Equipo Técnico del Modelo).

El modelo MPS⁸ establece un modelo de procesos de software y un método de evaluación de procesos. Esta estructura da sustentación al modelo MPS y asegura que esté siendo empleado de modo coherente con sus definiciones. El modelo MPS establece también un modelo de negocio para apoyar su adopción por las empresas desarrolladoras de software.

La base técnica para la construcción y perfeccionamiento de este modelo de mejora y evaluación de proceso de software está compuesta por las normas ISO/IEC 12207:2006 e ISO/IEC 15504-2 [ISO/IEC, 2003].

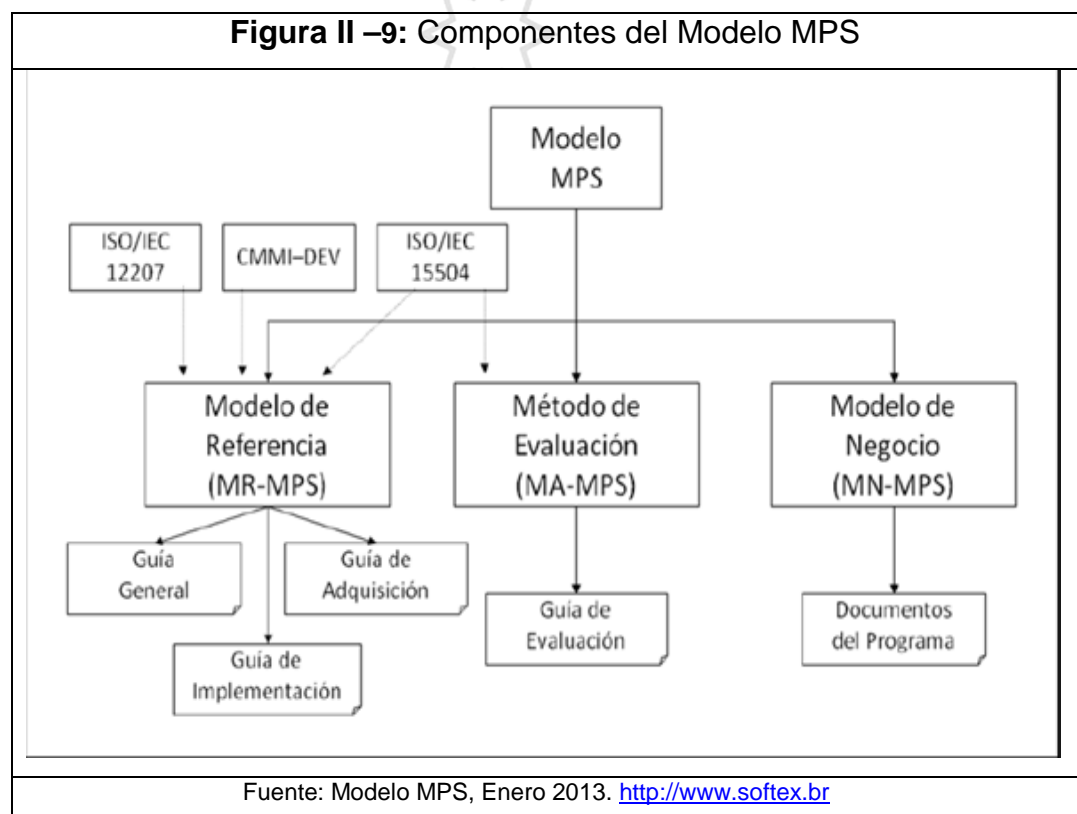
Una evaluación MPS es realizada utilizando el proceso y el método de evaluación MA-MPS. Una evaluación MPS verifica la conformidad de una organización/unidad organizacional a los procesos del MR-MPS.

El MR-MPS fue definido en conformidad con el CMMI-DEV [2006]. Para la definición y revisión del modelo de referencia, se hace una amplia consulta a la comunidad de implementadores y evaluadores MPS. La elaboración final es responsabilidad del ETM. El modelo MPS está dividido en tres (3) componentes: Modelo de Referencia (MR-MPS), Método de Evaluación (MA-MPS) y Modelo de Negocio (MN-MPS).

El Modelo de Referencia MR-MPS contiene los requisitos que los procesos de las unidades organizacionales deben cumplir para estar en conformidad con el MR-MPS. Contiene también las definiciones de los niveles de madurez, procesos y atributos del proceso. El método de evaluación MA-MPS contiene los requisitos para los evaluadores líderes, evaluadores adjuntos e instituciones evaluadoras.

⁸ MPS.BR, MR-MPS, MA-MPS y MN-MPS son marcas de la SOFTEX. La sigla MPS.BR está asociada al programa MPS.BR – Mejora del Proceso de Software Brasileño y la sigla MPS está asociada al modelo MPS – Mejora del Proceso de Software

El Modelo de Negocio MN-MPS describe reglas de negocio para implementación del MR-MPS por las Instituciones Implementadoras (II), evaluación siguiendo el MA-MPS por las Instituciones Evaluadoras (IA), organización de grupos de empresas por las Instituciones Organizadoras de Grupos de Empresas (IOGE) para implementación del MR-MPS y evaluación MA-MPS, certificación de Consultores de Adquisición (CA) y programas anuales de entrenamiento del MPS.BR por medio de cursos, pruebas y workshops.



Componentes

MPS.BR se basa en los conceptos de madurez y capacidad de proceso para la evaluación y mejora de la calidad y productividad de productos de software y servicios correlativos.

- ✓ Modelo de Referencia (MR-MPS)
- ✓ Método de Evaluación (MA-MPS)

- ✓ Modelo de Negocios (MN-MPS)

a. MPS.BR está descrito por medio de guías:

- Guía General
- Guía de Adquisición
- Guía de Evaluación

b. Descripción general

MPS.BR busca definir y perfeccionar un modelo de mejora y evaluación de procesos de software, enfocándose de manera especial en las micro, pequeñas y medianas empresas.

MPS.BR también establece un proceso y un método de evaluación de modo que se garantice su uso correcto.

Base técnica:

- Norma ISO / IEC 12207 – Proceso de Ciclo de Vida de Software y sus enmiendas 1 y 2.
- Norma ISO / IEC 15504 (SPICE) – Evaluación de Proceso.
- CMMI-SE/SW (Incluye procesos y resultados esperados adicionales)

c. Procesos MR-MPS

✓ **Procesos principales:**

Atienden el inicio y la ejecución del desarrollo, operación o mantenimiento de los productos software.

✓ **Procesos de apoyo:**

Auxilian otro proceso y contribuyen al éxito y calidad del proyecto de software

✓ **Procesos organizativos:**

Empleados por una organización para establecer, implementar o mejorar un proceso del ciclo de vida del software.

d. MR-MPS define siete niveles de madurez:

Los niveles de madurez establecen etapas de evolución de procesos identificando escalones de mejora de la implementación de procesos en la organización.

- A (En optimización)
- B (Gestionado cuantitativamente)
- C (Definido)
- D (Ampliamente definido)
- E (Parcialmente definido)
- F (Gestionado)
- G (Parcialmente gestionado)



2.1.4 Relación entre NTP-ISO/IEC 12207 CMMI 1.3 DEV, MOPROSOFT Y MPS.BR

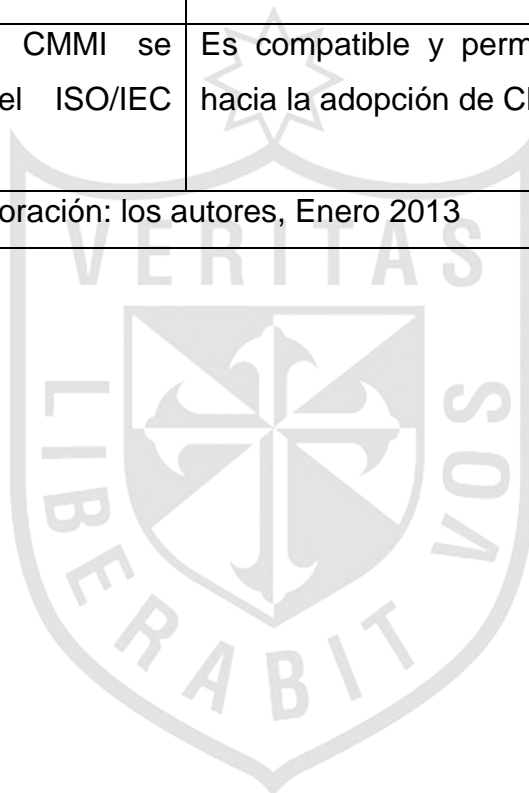
Cuadro II - 1: Relación entre NTP-ISO/IEC 12207 CMMI, MOPROSOFT y MPS.BR			
NTP-ISO/IEC 12207 (*)	CMMI 1.3 DEV (*)	MOPROSOFT (*)	MPS.BR
NTP ISO/IEC 12207 es la traducción de la norma ISO/IEC 12207. Con su promulgación se busca definir un marco y lenguaje común para los procesos software.	El CMMI es un modelo creado por el SEI (Software Engineering Institute) de la CMU (Carnegie Mellon University) ha pedido por el Departamento de Defensa de los EEUU.	Desarrollado por la Asociación Mexicana para la Calidad en Ingeniería de Software a través de la Facultad de Ciencias de la Universidad Nacional Autónoma de México (UNAM).	Una sociedad entre SOFTEX, el Gobierno y la Universidad, nació el proyecto MPS.Br (melhoria de processo de software brasileiro).
Periodo de tiempo que comienza a concebir la idea de un nuevo sistema de software, y termina cuando este se retira y deja de funcionar.	Permiten obtener un diagnóstico preciso de la madurez de los procesos relacionados con las tecnologías de la información de una organización y describe las tareas que se tienen	Permiten obtener un diagnóstico preciso de la madurez de los procesos relacionados con las tecnologías de la información	

Cuadro II - 1: Relación entre NTP-ISO/IEC 12207 CMMI, MOPROSOFT y MPS.BR			
NTP-ISO/IEC 12207 (*)	CMMI 1.3 DEV (*)	MOPROSOFT (*)	MPS.BR
	que llevar a cabo para mejorar los procesos.		
Se consideran tres tipos de procesos: Primarios, Soporte, Organizacionales.	Abarca cuatro categorías de procesos: Ingeniería, Gestión de Proyectos, Gestión de Procesos, Soporte.	Categoría de alta dirección (DIR). Categoría de Gerencia (GER). Categoría de Operación (OPE).	Componentes: Modelo de Referencia (MR-MPS) Método de Evaluación (MA-MPS) Modelo de Negocios (MN-MPS).
Presenta: 17 Procesos 74 Actividades 224 Tareas.	Define 5 niveles de madurez: 1 Inicial 2 Gestionado 3 Definido 4 Gestionado cuantitativamente 5 En optimización.	DIR: gestión de negocio. GER: gestión de procesos, gestión de proyectos, gestión de recursos, recursos humanos y ambiente de trabajo, bienes servicios e infraestructura,	Define siete niveles de madurez : A. En Optimización B. (Gestionado Cuantitativamente C. Definido D. Ampliamente Definido E. Parcialmente Definido F. Gestionado G. Parcialmente Gestionado.

Cuadro II - 1: Relación entre NTP-ISO/IEC 12207 CMMI, MOPROSOFT y MPS.BR			
NTP-ISO/IEC 12207 (*)	CMMI 1.3 DEV (*)	MOPROSOFT (*)	MPS.BR
		conocimiento de la organización OPE: administración de proyectos específicos, desarrollo y mantenimiento de software.	
¿Es útil para las organizaciones? Sí. Definitivamente si lo es. La estructura de los procesos y las actividades, si se ejecutan correctamente en toda la organización, conllevarían un control de documentación bastante útil, además de proveer de herramientas de trabajo estandarizadas.			
Permite obtener resultados con un esfuerzo, costo y tiempos razonables.	Permite obtener resultados con un gran esfuerzo, altos costos y largo tiempos.	Permite obtener resultados con un esfuerzo, costo y tiempos razonables.	
ISO/IEC 12207	Modelo basado en las mejores prácticas internacionales		
Implementaciones en empresas estatales.	Implementada y certificadas.	No se conoce	No se conoce
Se aplica a todas las empresas del sector del	Esta orientados a las grandes empresas.	Es un programa dirigido, aunque no de manera exclusiva, a las PYMES.	

Cuadro II - 1: Relación entre NTP-ISO/IEC 12207 CMMI, MOPROSOFT y MPS.BR			
NTP-ISO/IEC 12207 (*)	CMMI 1.3 DEV (*)	MOPROSOFT (*)	MPS.BR
estado.			
ISO/IEC 12207	El modelo CMMI se basa en el ISO/IEC 12207.	Es compatible y permite a una organización avanzar hacia la adopción de CMMI.	
Elaboración: los autores, Enero 2013			

(*) Modelo seleccionado

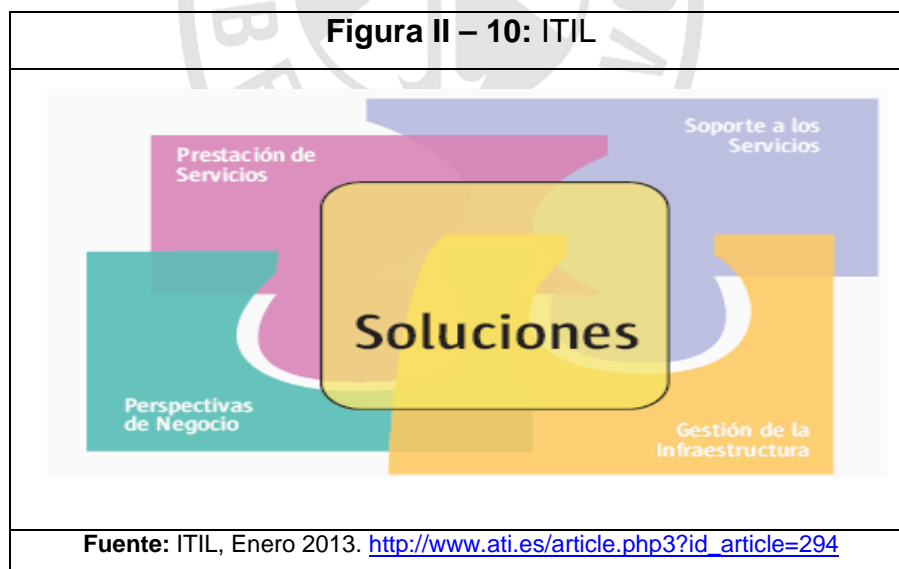


2.1.5 Metodologías de desarrollo

➤ **INFORMATION TECHNOLOGY INFRASTRUCTURE LIBRARY ITIL**

ITIL son las siglas de una metodología desarrollada a finales de los años 80's por iniciativa del gobierno del Reino Unido, específicamente por la OGC u Oficina Gubernativa de Comercio Británica (Office of Government Commerce). Las siglas de ITIL significan (Information Technology Infrastructure Library) o Librería de Infraestructura de Tecnologías de Información. Esta metodología es la aproximación más globalmente aceptada para la gestión de servicios de Tecnologías de Información en todo el mundo, ya que es una recopilación de las mejores prácticas tanto del sector público como del sector privado.

Estas mejores prácticas se dan con base en toda la experiencia adquirida con el tiempo en determinada actividad, y son soportadas bajo esquemas organizacionales complejos, pero a su vez bien definidos, y que se apoyan en herramientas de evaluación e implementación.



✓ **El objetivo de usar ITIL en Managed Services**

ITIL como metodología propone el establecimiento de estándares que nos ayuden en el control, operación y administración de los recursos (ya

sean propios o de los clientes). Plantea hacer una revisión y reestructuración de los procesos existentes en caso de que estos lo necesiten (si el nivel de eficiencia es bajo o que haya una forma más eficiente de hacer las cosas), lo que nos lleva a una mejora continua.

Otra de las cosas que propone es que para cada actividad que se realice se debe de hacer la documentación pertinente, ya que esta puede ser de gran utilidad para otros miembros del área, además de que quedan asentados todos los movimientos realizados, permitiendo que toda la gente esté al tanto de los cambios y no se tome a nadie por sorpresa.

En la documentación, se pone la fecha en la que se hace el cambio, una breve descripción de los cambios que se hicieron; quién fue la persona que hizo el cambio, así como quién autorizó el cambio, para que así se lleve todo un seguimiento de lo que ocurre en el entorno. Esto es más que nada como método, con el cual, se establece cierto control en el sistema de cambios, y así siempre habrá un responsable y se identificarán los procedimientos y cambios efectuados.

✓ **Forma de uso de ITIL en Managed Services**

En el modelo ITIL postula que el servicio de soporte, la administración y la operación se realiza a través de cinco procesos, a saber:

- Manejo de incidentes

Su objetivo primordial es restablecer el servicio lo más rápido posible para evitar que el cliente se vea afectado, esto se hace con la finalidad de que se minimicen los efectos de la operación. Se dice que el proveedor de debe de encargarse de que el cliente no debe percibir todas aquellas pequeñas o grandes fallas que llegue a presentar el sistema. A este concepto se le llama disponibilidad (que el usuario pueda tener acceso al servicio y que nunca se vea interrumpido).

- Manejo de problemas

El objetivo de este proceso es prevenir y reducir al máximo los incidentes, y esto nos lleva a una reducción en el nivel de incidencia. Por otro lado, nos ayuda a proporcionar soluciones rápidas y efectivas para asegurar el uso estructurado de recursos. En este proceso lo que se busca es que se pueda tener pleno control del problema, esto se logra dándole un seguimiento y un monitoreo al problema.

- Manejo de configuraciones

Su objetivo es proveer con información real y actualizada de lo que se tiene configurado e instalado en cada sistema del cliente. Este proceso es de los más complejos, ya que se mueve bajo cuatro vértices que son: administración de cambios, administración de liberaciones, administración de configuraciones y la administración de procesos diversos.

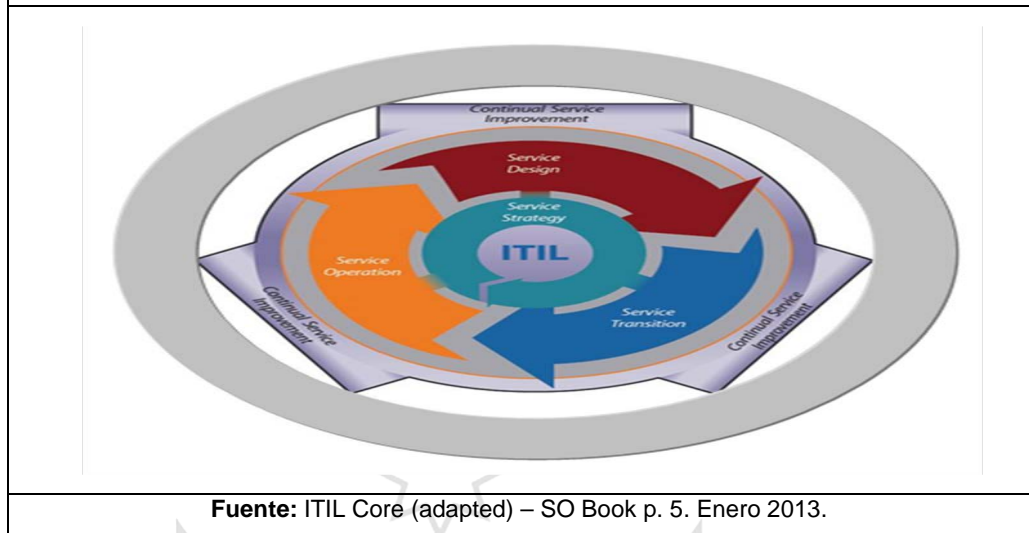
- Manejo de cambios

El objetivo de este proceso es reducir los riesgos tanto técnicos, económicos y de tiempo al momento de la realización de los cambios. Este diagrama al parecer es muy fácil de seguir, pero en realidad no lo es, ya que entre las etapas se da una fase de monitoreo para ver que no se han sufrido desviaciones de los objetivos.

- Manejo de entregas

Su objetivo es planear y controlar exitosamente la instalación de Software y Hardware bajo tres ambientes: ambiente de desarrollo, ambiente de pruebas controladas y ambiente real. Este proceso tiene un diagrama que marca la transición que se da de acuerdo a los ambientes por los que se va dando la evolución del proyecto.

Figura II – 11: ITIL



Fuente: ITIL Core (adapted) – SO Book p. 5. Enero 2013.

➤ **RATIONAL UNIFIED PROCESS (RUP)⁹**

El Proceso Unificado de Rational (Rational Unified Process en inglés, habitualmente resumido como RUP) es un proceso de desarrollo de software desarrollado por la empresa Rational Software, actualmente propiedad de IBM. Junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, diseño, implementación y documentación de sistemas orientados a objetos.

El RUP no es un sistema con pasos firmemente establecidos, sino un conjunto de metodologías adaptables al contexto y necesidades de cada organización.

También se conoce por este nombre al software, también desarrollado por Rational, que incluye información entrelazada de diversos artefactos y descripciones de las diversas actividades. Originalmente se diseñó un proceso genérico y de dominio público, el Proceso Unificado, y una especificación más detallada, el Rational Unified Process, que se vendiera como producto independiente.

⁹Rational Unified Process: 2004.<http://www-306.ibm.com/software/rational/>

Los elementos del RUP son:

- ✓ **Actividades**, Son los procesos que se llegan a determinar en cada iteración.
- ✓ **Trabajadores**, Vienen a ser las personas o entes involucrados en cada proceso.
- ✓ **Artefactos**, Un artefacto puede ser un documento, un modelo, o un elemento de modelo.

En la metodología RUP, llamada así por sus siglas en inglés Rational Unified Process, se divide en 4 fases el desarrollo del software:

- ✓ **Inicio**, Determinar la visión del proyecto.
- ✓ **Elaboración**, Determinar la arquitectura óptima.
- ✓ **Construcción**, Obtener la capacidad operacional inicial.
- ✓ **Transición**, Obtener el release del proyecto.

Cada una de estas etapas es desarrollada mediante el ciclo de iteraciones, la cual consiste en reproducir el ciclo de vida en cascada a menor escala. Los objetivos de una iteración se establecen en función de la evaluación de las iteraciones precedentes.

Vale mencionar que el ciclo de vida que se desarrolla por cada iteración, es llevada bajo dos disciplinas:

Disciplina de desarrollo

- ✓ Modelo de negocios: Entendiendo las necesidades del negocio.
- ✓ Requerimientos: Trasladando las necesidades del negocio a un sistema automatizado.
- ✓ Análisis y diseño: Trasladando los requerimientos dentro de la arquitectura de software.
- ✓ Implementación: Creando software que se ajuste a la arquitectura y que tenga el comportamiento deseado.

- ✓ Pruebas: Asegurándose que el comportamiento requerido es el correcto y que todo lo solicitado está presente.
- ✓ Despliegue: Hacer todo lo necesario para la salida del proyecto.

Disciplina de soporte

- ✓ Configuración y administración del cambio: Guardando todas las versiones del proyecto.
- ✓ Administrando el proyecto: Administrando horarios y recursos.
- ✓ Ambiente: Administrando el ambiente de desarrollo.

Es recomendable que a cada una de estas iteraciones se les clasifique y ordene según su prioridad, y que cada una se convierte luego en un entregable al cliente. Esto trae como beneficio la retroalimentación que se tendría en cada entregable o en cada iteración.

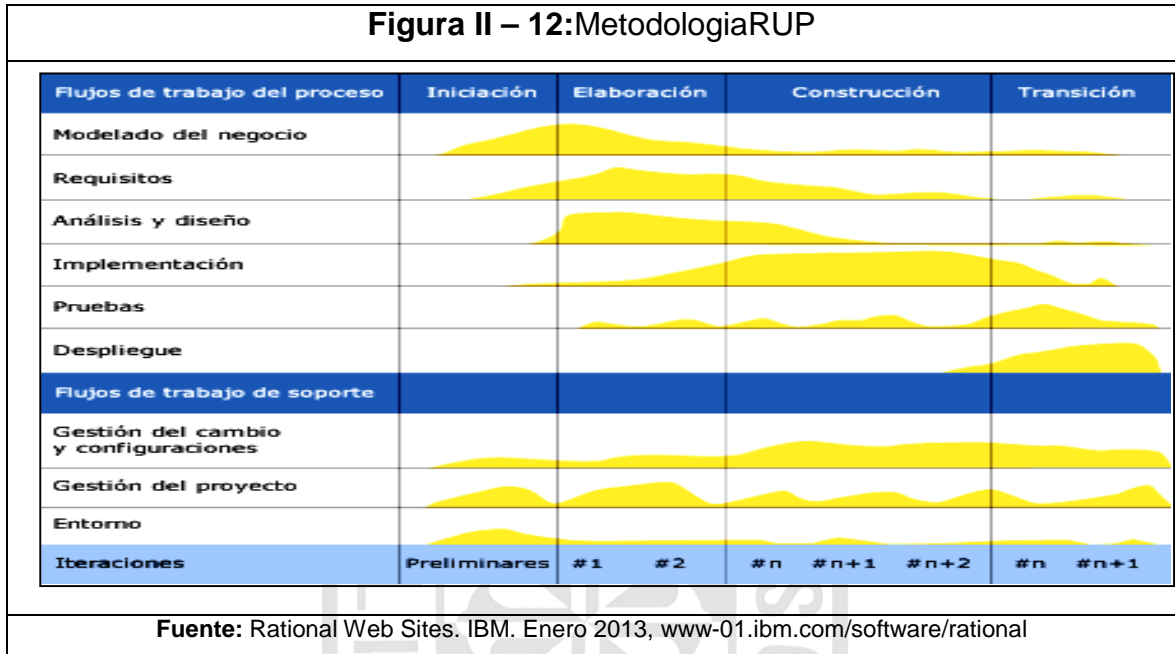
Una particularidad de esta metodología es que, en cada ciclo de iteración, se hace exigente el uso de artefactos, siendo por este motivo, una de las metodologías más importantes para alcanzar un grado de certificación en el desarrollo del software.

Por otro lado, en lo que se refiere a la metodología esta comprende tres principios claves: Dirigido por los casos de uso, centrado en la arquitectura, iterativo e incremental.

En lo referente a dirigido por los casos de uso, significa que los requerimientos están enfocados a dar valor al cliente y que el proceso debe garantizar que todo el desarrollo, pruebas, planeación, documentación etc, está orientado a cubrir estas expectativas del cliente y asegurar que los requerimientos de valor se ponen en producción.

En lo referente a centrado en arquitectura, significa que hay un énfasis a diseñar una arquitectura de calidad, y es la arquitectura también la que guía la forma cómo se debe planear y hacer el desarrollo.

En lo referente a iterativo e incremental, significa que el proyecto se divide en varios ciclos de vida (llamadas iteraciones) que deben dar como resultado un ejecutable. Por cada una de las iteraciones se va agregando requerimientos y sobre todo valor al cliente; por este motivo es incremental.



➤ **EXTREME PROGRAMING (XP)**

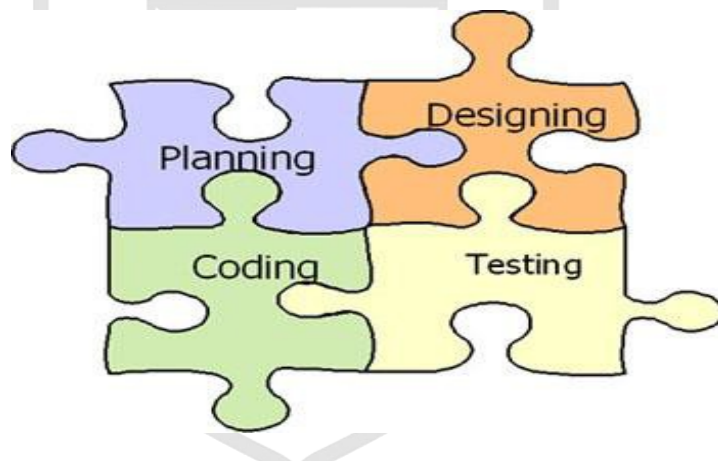
Es una de las metodologías de desarrollo de software más exitosas en la actualidad de las metodologías ágiles. El Extreme Programming¹⁰ se diferencia del resto de metodologías porque le pone mayor énfasis a la adaptabilidad. Los cambios de requisitos se pueden realizar en cualquier punto del proyecto y es un aspecto muy natural en esta metodología. El usuario final es parte del equipo de proyecto.

El Extreme Programming generalmente se utiliza para proyectos pequeños en la que el tiempo de presentación de los entregables es uno de los principales riesgos que se deben mitigar. La metodología XP se divide en 4 fases el desarrollo del software:

¹⁰ Extreme Programming: 2007. <http://www.extremeprogramming.com>

- **Planificación del proyecto:** Se ejecutan distintas actividades teniendo en cuenta las historias de usuario, release planning, iteraciones, velocidad del proyecto, programación en pareja y reuniones diarias.
- **Diseño:** Se elaboran diseños simples, glosario de términos, funcionalidad extra y refactoring.
- **Codificación:** La codificación debe hacerse atendiendo a estándares de codificación ya creados, asimismo, crear test que prueben el funcionamiento de los distintos códigos implementados.
- **Pruebas:** Para asegurar el funcionamiento final de una determinada historia de usuario se deben crear “Test de aceptación” y verificar cumplen su cometido.

Figura II – 13: Extreme Programming (XP)



Fuente: Metodologías desarrollo de software. Enero 2013. <http://www.informatizate.net>

Las características que posee la metodología XP. Son:

- **Respuesta ante el cambio:** Es más importante que el seguimiento de un plan.
- **Comunicación:** Usuarios y los desarrolladores.
- **Simplicidad:** Al desarrollar y codificar los módulos del software.
- **Retroalimentación:** Concreta y frecuente del equipo de desarrollo, el cliente y los usuarios finales..

- **Importancia:** Es más prioritario que el Software funcione a que exista una documentación exhaustiva.

Características de XP, la metodología se basa en:

- ✓ **Pruebas Unitarias:** se basa en las pruebas realizadas a los principales procesos, de tal manera que adelantándonos en algo hacia el futuro, podamos hacer pruebas de las fallas que pudieran ocurrir. Es como si nos adelantáramos a obtener los posibles errores.
- ✓ **Refabricación:** se basa en la reutilización de código, para lo cual se crean patrones o modelos estándares, siendo más flexible al cambio.
- ✓ **Programación en pares:** una particularidad de esta metodología es que propone la programación en pares, la cual consiste en que dos desarrolladores participen en un proyecto en una misma estación de trabajo. Cada miembro lleva a cabo la acción que el otro no está haciendo en ese momento. Es como el chofer y el copiloto: mientras uno conduce, el otro consulta el mapa.

¿Qué es lo que se propone con la metodología Extreme Programming?

- ✓ El inicio en pequeño y añadir funcionalidad con retroalimentación continua.
- ✓ El manejo del cambio se convierte en parte sustantiva del proceso.
- ✓ El costo del cambio no depende de la fase o etapa.
- ✓ No introduce funcionalidades antes que sean necesarias.
- ✓ El cliente o el usuario se convierte en miembro del equipo.

Derechos del cliente

- ✓ Decidir qué se implementa.
- ✓ Saber el estado real y el progreso del proyecto.
- ✓ Añadir, cambiar o quitar requerimientos en cualquier momento.
- ✓ Obtener lo máximo de cada semana de trabajo.
- ✓ Obtener un sistema funcionando cada 3 o 4 meses.

Derechos del desarrollador

- ✓ Decidir cómo se implementan los procesos.
- ✓ Crear el sistema con la mejor calidad posible.
- ✓ Pedir al cliente en cualquier momento aclaraciones de los requerimientos.
- ✓ Estimar el esfuerzo para implementar el sistema.
- ✓ Cambiar los requerimientos en base a nuevos descubrimientos.

Lo fundamental en este tipo de metodología es:

- ✓ La comunicación, entre los usuarios y los desarrolladores.
- ✓ La simplicidad, al desarrollar y codificar los módulos del sistema.
- ✓ La retroalimentación, concreta y frecuente del equipo de desarrollo, el cliente y los usuarios finales.

➤ **MICROSOFT SOLUTION FRAMEWORK (MSF)¹¹**

Esta es una metodología flexible e interrelacionada con una serie de conceptos, modelos y prácticas de uso, que controlan la planificación, el desarrollo y la gestión de proyectos tecnológicos. MSF se centra en los modelos de proceso y de equipo dejando en un segundo plano las elecciones tecnológicas.

MSF se compone de varios modelos encargados de planificar las diferentes partes implicadas en el desarrollo de un proyecto: Modelo de Arquitectura del Proyecto, Modelo de Equipo, Modelo de Proceso, Modelo de Gestión del Riesgo, Modelo de Diseño de Proceso y finalmente el modelo de Aplicación.

Las fases y algunos de los entregables de cada una que posee la metodología MSF son las siguientes:

¹¹Microsoft Solution Framework:2007

<http://www.microsoft.com/technet/solutionaccelerators/msf/default.aspx>

- **Visionado:** Elaboración de la visión, alcance, estructura del proyecto, planteamiento de los riesgos, lista Inicial de características a probar.
- **Planificación:** Se elabora perfiles de usuarios actualizados, requerimientos candidatos, casos de Uso detallados y escenarios de uso actuales.
- **Desarrollo:** Código fuente y archivo ejecutable, script de instalación, especificación funcional, especificación para pruebas y casos funcional.
- **Estabilización:** Se realiza la distribución final, notas de distribución, elementos de soporte de funcionamiento, código fuente y archivos ejecutables.
- **Distribución:** Sistemas de operación y soporte.

MSF tiene las características siguientes:

- ✓ **Adaptable:** es parecido a un compás, usado en cualquier parte como un mapa, del cual su uso es limitado a un específico lugar.
- ✓ **Escalable:** puede organizar equipos tan pequeños entre 3 o 4 personas, así como también, proyectos que requieren 50 personas a más.
- ✓ **Flexible:** es utilizada en el ambiente de desarrollo de cualquier cliente.
- ✓ **Tecnología agnóstica:** porque puede ser usada para desarrollar soluciones basadas sobre cualquier tecnología.

Figura II – 14:Microsoft Solution Framework (MSF)



Fuente: Link del, Microsoft, enero 2013.

http://www.mentores.net/articulos/intro_microsoft_sol_frame.htm

MSF se compone de varios modelos encargados de planificar las diferentes partes implicadas en el desarrollo de un proyecto: Modelo de Arquitectura del Proyecto, Modelo de Equipo, Modelo de Proceso, Modelo de Gestión del Riesgo, Modelo de Diseño de Proceso y finalmente el modelo de Aplicación.

- ✓ Modelo de arquitectura del proyecto: Diseñado para acortar la planificación del ciclo de vida. Este modelo define las pautas para construir proyectos empresariales a través del lanzamiento de versiones.
- ✓ Modelo de equipo: Este modelo ha sido diseñado para mejorar el rendimiento del equipo de desarrollo. Proporciona una estructura flexible para organizar los equipos de un proyecto. Puede ser escalado dependiendo del tamaño del proyecto y del equipo de personas disponibles.
- ✓ Modelo de proceso: Diseñado para mejorar el control del proyecto, minimizando el riesgo, y aumentar la calidad acortando el tiempo de entrega. Proporciona una estructura de pautas a seguir en el ciclo de

vida del proyecto, describiendo las fases, las actividades, la liberación de versiones y explicando su relación con el Modelo de equipo.

- ✓ Modelo de gestión del riesgo: Diseñado para ayudar al equipo a identificar las prioridades, tomar las decisiones estratégicas correctas y controlar las emergencias que puedan surgir. Este modelo proporciona un entorno estructurado para la toma de decisiones y acciones valorando los riesgos que puedan provocar.
- ✓ Modelo de diseño del proceso: Diseñado para distinguir entre los objetivos empresariales y las necesidades del usuario. Proporciona un modelo centrado en el usuario para obtener un diseño eficiente y flexible a través de un enfoque iterativo. Las fases de diseño conceptual, lógico y físico proveen tres perspectivas diferentes para los tres tipos de roles: los usuarios, el equipo y los desarrolladores.
- ✓ Modelo de aplicación: Diseñado para mejorar el desarrollo, el mantenimiento y el soporte, proporciona un modelo de tres niveles para diseñar y desarrollar aplicaciones software. Los servicios utilizados en este modelo son escalables, y pueden ser usados en un solo ordenador o incluso en varios servidores.

➤ **SCRUM**¹²

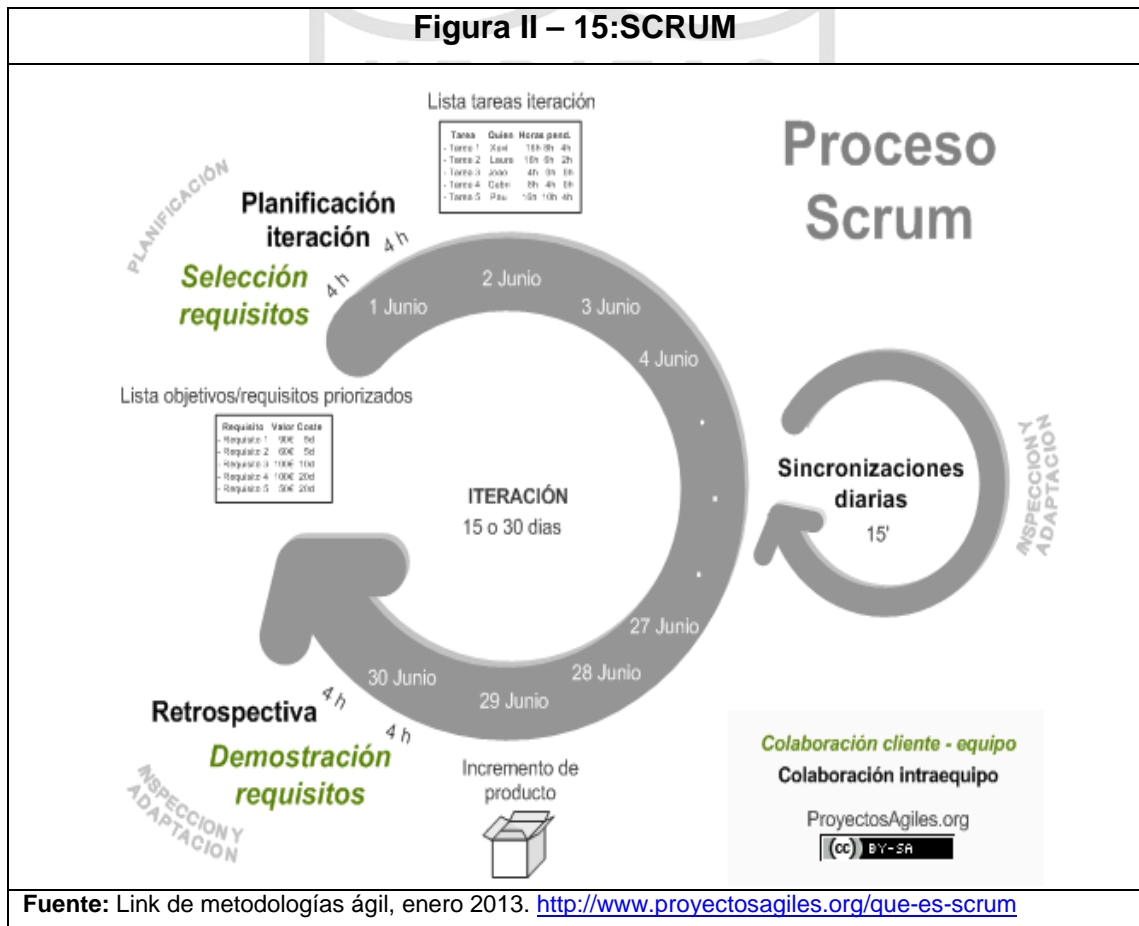
Es un proceso en el que se aplican de manera regular un conjunto de mejores prácticas para trabajar en equipo y obtener el mejor resultado posible de un proyecto. Estas prácticas se apoyan unas a otras y su selección tiene origen en un estudio de la manera de trabajar de equipos altamente productivos.

En Scrum se realizan entregas parciales y regulares del resultado final del proyecto, priorizadas por el beneficio que aportan al receptor del proyecto. Por ello, Scrum está especialmente indicado para proyectos en entornos complejos, donde se necesita obtener resultados pronto, donde los

¹² CONTROLCHAOS, 31 de enero de 2013<<http://www.scrum.org/scrum,quides/>>

requisitos son cambiantes o poco definidos, donde la innovación, la competitividad y la productividad son fundamentales.

Scrum también se utiliza para resolver situaciones en que no se está entregando al cliente lo que necesita, cuando las entregas se alargan demasiado, los costes se elevan o la calidad no es aceptable, cuando se necesita capacidad de reacción ante la competencia, cuando la moral de los equipos es baja y la rotación alta, cuando es necesario identificar y solucionar ineficiencias sistemáticamente o cuando se quiere trabajar utilizando un proceso especializado en el desarrollo de producto.



2.1.6 Relación entre XP, SCRUM, MSF, RUP

Cuadro II - 2: Relación entre XP, SCRUM, MSF, RUP				
Metodología	XP	SCRUM	MSF	RUP (*)
Característica				
Se adapta a proyectos de cualquier dimensión y de cualquier tecnología.	Corto plazo		Cualquier dimensión	
Gestión de proyecto	NO	SI		
Ámbito de aplicación	Genérico		Software / Sistemas	
Representación	Plana	Continua	Continua / etapas	Continua
Análisis del negocio	Poca		Medio	Completo
Análisis del Requerimiento	Completo			
Verifica la solución técnica	Completa		La necesaria	Completa
Validación	Completa		La necesaria	Completa
Gestión de cambio	Poca		La necesaria	Completa
Documentación generada	Poca		La necesaria	Completa
Elaboración: los autores, Enero 2013				

(*) Metodología seleccionada

2.1.7 Relación entre el Modelo y Metodología seleccionados

2.1.7.1 Relación NTP-ISO/IEC 12207:2006 con la Metodología RUP

Norma Técnica Peruana describe la arquitectura de los procesos del ciclo de vida del software, pero no especifica los detalles de cómo implementar o llevar a cabo las actividades y tareas incluidas en los procesos:

- **Proceso de gestión.** Define las actividades básicas de gestión, incluyendo la gestión de proyectos, durante un proceso del ciclo de vida, es ahí que lo relacionamos con el **RUP** que nos proporciona el artefacto plan de proyecto en **disciplina ámbito** y el plan de iteraciones en la disciplina **administración de proyecto**. El RUP proporciona los artefactos de plan de proyecto, plan de iteraciones.
- **Proceso de desarrollo.** Define las actividades del desarrollador, organización que define y desarrolla el producto software.
- **Proceso de mantenimiento.** Define las actividades del responsable de mantenimiento, organización que proporciona el servicio de mantenimiento del producto software; esto es, la gestión de las modificaciones al producto software para mantenerlo, actualizado y operativo. Este proceso incluye la migración y retirada del producto software.

El **RUP** se relaciona con el proceso de desarrollo y mantenimiento con las disciplina modelo de negocio (diagrama de caso de uso de negocio, diagrama de actividades), requerimiento (diagrama de casos de uso, plantilla de especificación de uso), análisis y diseño (diagrama de colaboración, secuencia, modelo físico) e implementación (diagrama de

componente, diagrama de despliegue), según los entregables que se planifiquen entregar.

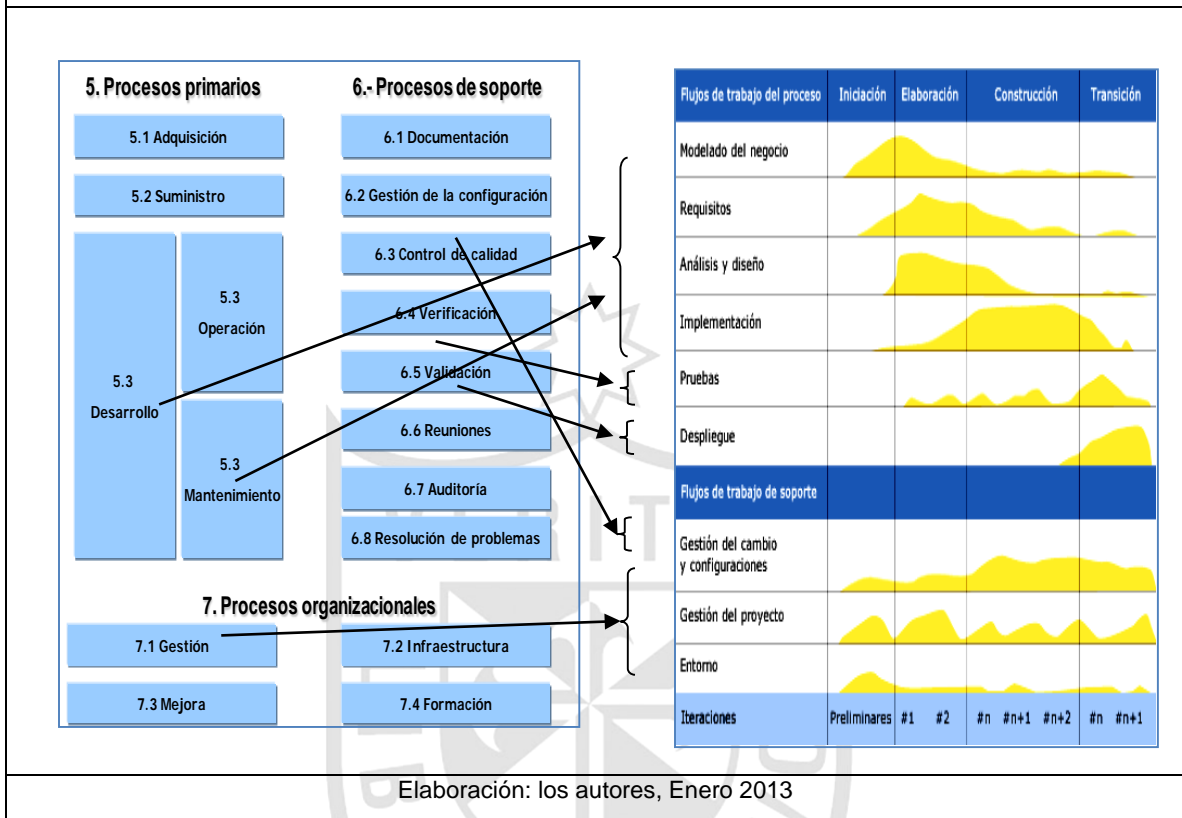
- **Proceso de gestión de la configuración.** Define las actividades de la gestión de la configuración.

El RUP está relacionado con la disciplina **administración y control de cambio** lo cual el **RUP** nos permite:

- ✓ Permitir, controlar y monitorear cambios para habilitar un desarrollo iterativo.
 - ✓ Controlar todos los artefactos de software - modelos, código, documentos, etc.
 - ✓ Administrar todas las versiones, con integración automática a los cambios realizados al software.
 - ✓ Establecer espacios de trabajo, seguros y aislados para cada desarrollador.
 - ✓ Contar con métricas de estado y avance.
- **Proceso de verificación.** Define las actividades (para el adquiriente, proveedor o una parte independiente) para verificar hasta un nivel de detalle dependiente del proyecto software, los productos software.
 - **Proceso de validación.** Define las actividades (para el adquiriente, proveedor o una parte independiente) para validar los productos software del proyecto software.

El **RUP** se relaciona con el proceso de verificación y validación con la disciplina **prueba**, nos permite:

Figura II – 16: Relación NTP-ISO/IEC 12207:2006 con la Metodología RUP



2.1.7.2 Relación Modelo CMMI con la Metodología RUP

CMMI es un modelo que define áreas de procesos (PA) en las que se deben llevar a cabo prácticas específicas o genéricas, por lo tanto el hecho de implementar RUP en el desarrollo de un proyecto implica que ciertas PA de CMMI sean alcanzadas y otras no.

CMMI nivel 2 y 3

- Requirements Management - REQM

RUP define claramente el proceso de administración de requerimientos y aporta los artefactos: diagrama de casos de uso, plantilla de especificación de caso de uso es una de las bases de RUP.

- **Project Planning - PP**

RUP habla de la planeación del proyecto de manera iterativa y del control de riesgos. RUP nos proporciona los artefactos: plan de proyecto, plan de iteraciones.

- **Project Monitoring and Control - PMC**

RUP define cómo debe ser el control del proyecto, con el artefacto lo planificado vs. lo avanzado.

- **Configuration Management - CM**

RUP es muy claro cuando se habla de administración de la configuración incluso es una de las mejores prácticas recomendada.

El RUP está relacionado con la disciplina **administración y control de cambio**, lo que nos permite:

- ✓ Controlar y monitorear cambios para habilitar un desarrollo iterativo.
- ✓ Controlar todos los artefactos de software - modelos, código, documentos, etc.
- ✓ Administrar todas las versiones, con integración automática a los cambios realizados al software.
- ✓ Establecer espacios de trabajo, seguros y aislados para cada desarrollador.
- ✓ Contar con métricas de estado y avance.

No son cubiertas por el RUP

- **Supplier Agreement Management -SAM**

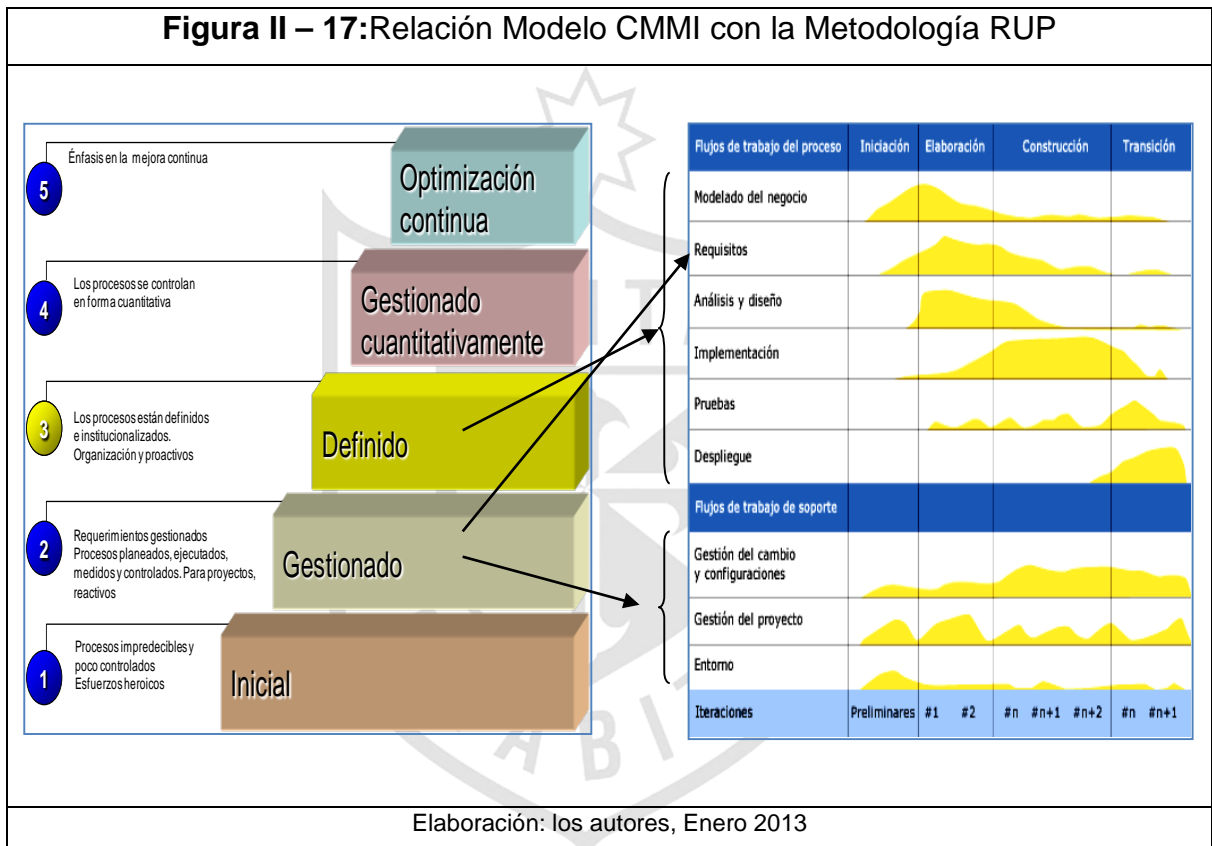
RUP no menciona nada sobre administración de acuerdos, es algo no considerado.

- **Measurement and Analysis - MA**

La medición y análisis no están contemplados detalladamente en RUP.

- **Process and Product Quality Ass. -PPQA**

En la etapa de transición, se lleva a cabo la verificación de la calidad aunque no tan detallada como lo exige CMMI. La verificación de calidad del producto está bien definida, pero la evaluación de calidad del proceso no está considerada.



2.1.7.3 Relación Modelo MOPROSOFT con la Metodología RUP

Es la mejora y evaluación de los procesos de desarrollo y mantenimiento de sistemas y productos de software. En la Categoría Operación (OPE) tenemos:

- Administración de Proyectos Específicos

- Desarrollo y Mantenimiento de Software

Es la que existe relación con el RUP:

- **Administración de Proyectos específicos**

Establecer y llevar a cabo sistemáticamente las actividades que permitan cumplir con los objetivos de un proyecto en tiempo y costo esperados.

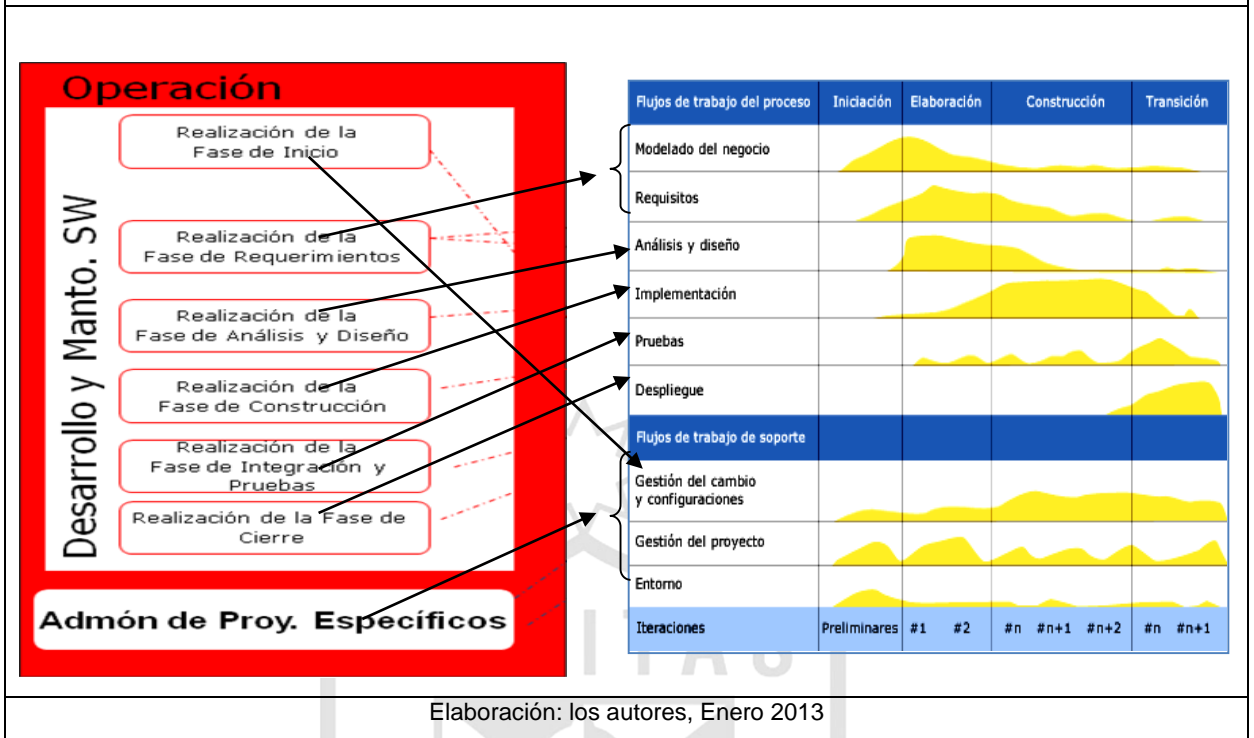
El **RUP** que nos proporciona el artefacto plan de proyecto en disciplina ámbito y el plan de iteraciones en la disciplina administración de proyecto.

- **Desarrollo y mantenimiento de software**

Es la realización sistemática de las actividades de análisis, diseño, construcción, integración y pruebas de productos de software nuevo o modificado cumpliendo con los requerimientos especificados.

El **RUP** se relaciona con el proceso de desarrollo y mantenimiento con las disciplina modelo de negocio (diagrama de caso de uso de negocio, diagrama de actividades), requerimiento (diagrama de casos de uso, plantilla de especificación de uso), análisis y diseño (diagrama de colaboración, secuencia, modelo físico), implementación (diagrama de componente, diagrama de despliegue), prueba (unitarias, funcionales e integración) y el despliegue (entrega del producto). Según los entregables que se planifiquen ofrecer.

Figura II – 18:Relación Modelo MOPROSOFT con la Metodología RUP



2.1.7.4 Entregables tomado del Modelo y Metodología

Cuadro II - 3:Entregables tomado del Modelo y Metodología

Fases/Flujos	Inicio	Elaboración	Construcción	Transición
Planificación	<ul style="list-style-type: none"> ▪ Lista de Proyecto aprobado y priorizados. ▪ Evaluación de riesgos. ▪ Estructura general del proyecto (WBS). ▪ Cronograma de trabajo. ▪ Monitorear el progreso y desempeño actual del proyecto vs el plan del proyecto. 	<ul style="list-style-type: none"> ▪ Monitorear el progreso y desempeño actual del proyecto vs el plan del proyecto. 	<ul style="list-style-type: none"> ▪ Monitorear el progreso y desempeño actual del proyecto vs el plan del proyecto. 	<ul style="list-style-type: none"> ▪ Monitorear el progreso y desempeño actual del proyecto vs el plan del proyecto.
	Modelo: MOPROSOFT basado en CMMI. Metodología RUP			

Cuadro II - 3:Entregables tomado del Modelo y Metodología

Fases/Flujos	Inicio	Elaboración	Construcción	Transición
Requerimiento	<ul style="list-style-type: none"> ▪ Solicitud de Req. Detallado. ▪ Cronograma detallado del proyecto. ▪ Asignación de recurso. 	<ul style="list-style-type: none"> ▪ Control y seguimiento. 	<ul style="list-style-type: none"> ▪ Control y seguimiento. 	<ul style="list-style-type: none"> ▪ Control y seguimiento.
	Modelo: MOPROSOFT basado en CMMI.			
Análisis		<ul style="list-style-type: none"> ▪ Elaboración Documento de Análisis Funcional: Doc. Análisis. ▪ Elaboración Documento Técnico. ▪ Especificaciones Requerimiento de software. ▪ Control y seguimiento. 	<ul style="list-style-type: none"> ▪ Revisión y Aprobación Doc. Funcional. ▪ Revisión y Aprobación Doc. Técnico. ▪ Control y seguimiento. 	
	Modelo: MOPROSOFT basado en CMMI. Metodología RUP			

Cuadro II - 3:Entregables tomado del Modelo y Metodología

Fases/Flujos	Inicio	Elaboración	Construcción	Transición
Diseño		<ul style="list-style-type: none"> ▪ Elaboración de pruebas técnicas y funcionales. ▪ Elaboración de documento a crear o modificar. ▪ Control y seguimiento. 	<ul style="list-style-type: none"> ▪ Elaboración: <ul style="list-style-type: none"> ☞ Manual de usuario. ☞ Manual de sistemas. ▪ Control y seguimiento. 	
		<p>Modelo: MOPROSOFT Metodología RUP.</p>	<p>basado en CMMI.</p>	
Implementación			<ul style="list-style-type: none"> ▪ Codificación <ul style="list-style-type: none"> ☞ Estándares. ▪ Documento de Implementación: <ul style="list-style-type: none"> ☞ Diagrama componente. ☞ Diagrama despliegue. ▪ Documento de pruebas 	

Cuadro II - 3: Entregables tomado del Modelo y Metodología

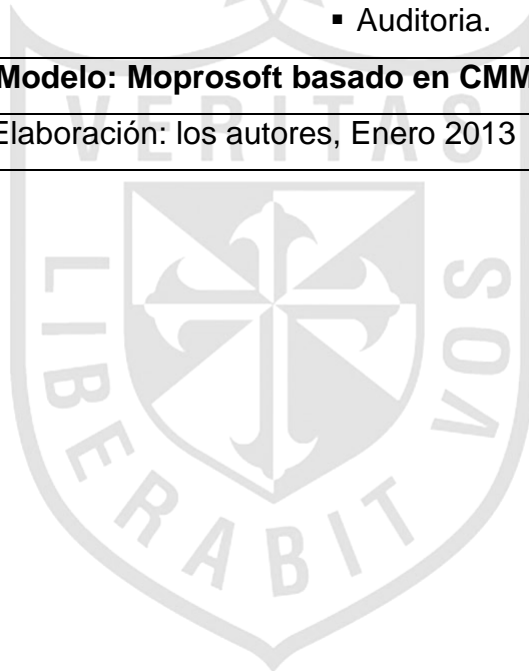
Fases/Flujos	Inicio	Elaboración	Construcción	Transición
			unitarias ▪ Rev. Pares. ▪ Control y seguimiento Modelo: Moprosoft basado en CMMI. Metodología RUP y XP.	
Prueba			▪ Aprobación del documento de pruebas. ▪ Ejecución de pruebas planificadas. <ul style="list-style-type: none"> ☞ Funcionales. ☞ Técnicas. ▪ Ejecución de pruebas de carga o volumen. ▪ Revisión de: <ul style="list-style-type: none"> ☞ Manual de usuario. ☞ Manual de sistemas. 	

Cuadro II - 3: Entregables tomado del Modelo y Metodología

Fases/Flujos	Inicio	Elaboración	Construcción	Transición
			<ul style="list-style-type: none"> ▪ Control y seguimiento. <p>Modelo: Moprosoft basado en CMMI. Metodología RUP y XP.</p>	
Despliegue				<ul style="list-style-type: none"> ▪ Pase a producción. ▪ Entrega: <ul style="list-style-type: none"> • Manual de usuario. • Manual de sistemas. ▪ Manual de capacitación. <p>Modelo: Moprosoft basado en CMMI. Metodología RUP.</p>

Cuadro II - 3: Entregables tomado del Modelo y Metodología

Fases/Flujos	Inicio	Elaboración	Construcción	Transición
Gestión de Cambio	<ul style="list-style-type: none"> ▪ Solicitud de Cambio. ▪ Lista de ítem de configuración. ▪ Auditoria. 			
<p align="center">Modelo: Moprosoft basado en CMMI. Metodología RUP</p>				
<p align="center">Elaboración: los autores, Enero 2013</p>				



2.2 Bases y enfoques teóricos

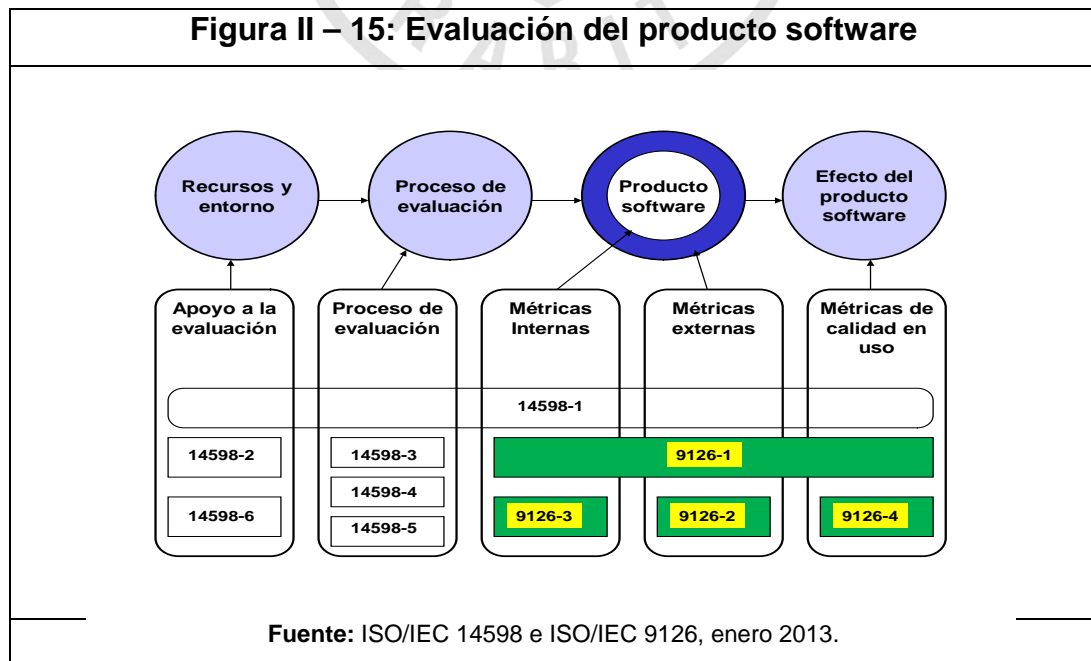
Diseño de una Metodología para la Certificación de Productos Software permite una valoración independiente que debe demostrar que la organización es capaz de desarrollar productos y servicios de calidad.

También es necesario considerar mediciones en el proceso empleado para diseñar, desarrollar, probar y controlar el producto. En esto juega un papel relevante la ISO/IEC 14598.

2.2.1 La ISO/IEC 14598

Ofrece una visión general, explica la relación entre su serie y el modelo de calidad de la ISO/IEC 9126, define los términos técnicos utilizados, contiene requisitos generales para la especificación y evaluación de la calidad del software y clarifica los conceptos generales.

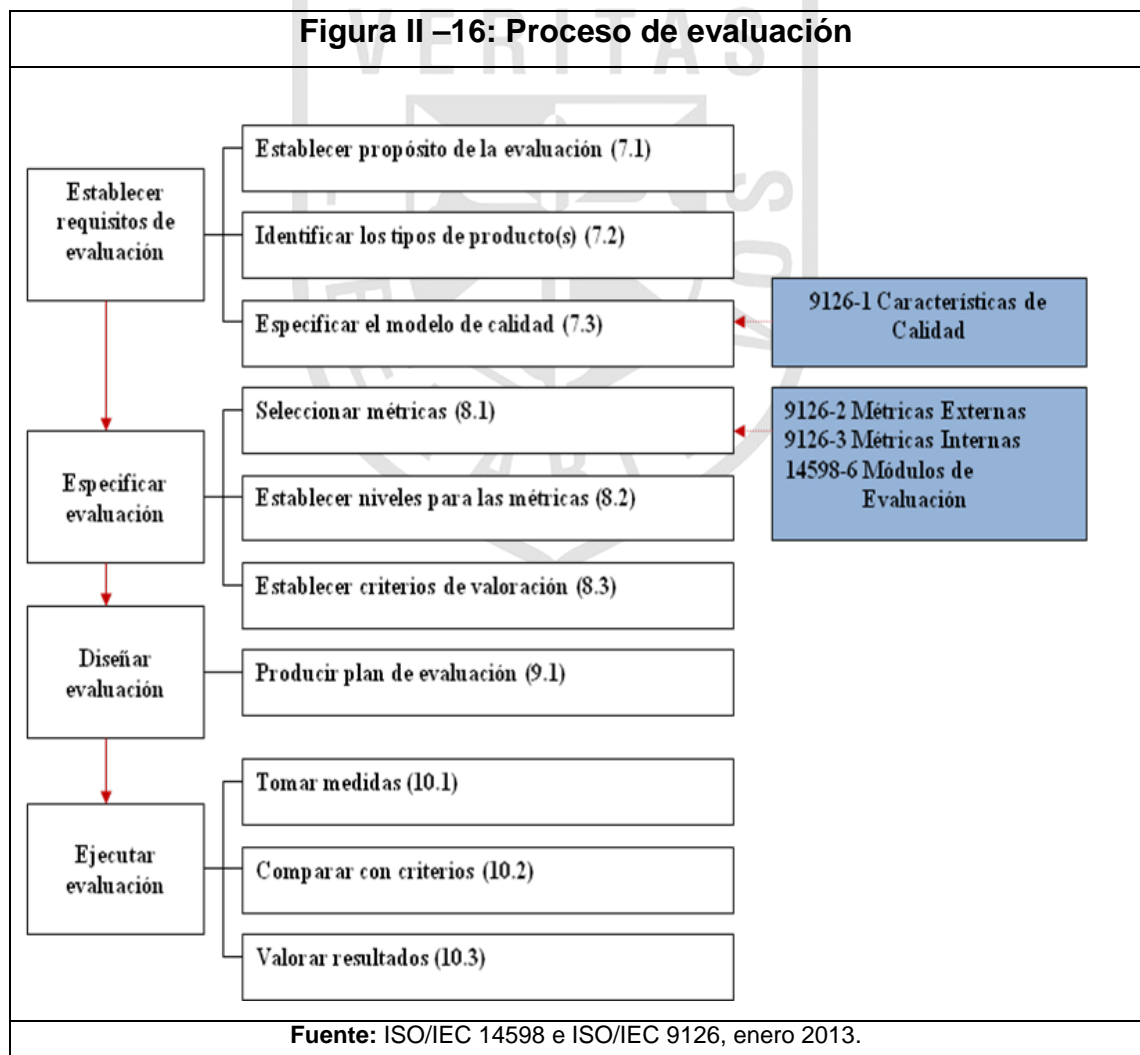
Además, provee un marco de trabajo para evaluar la calidad de todos los tipos de productos de software y establece requisitos para métodos de medición y evaluación de los productos de software.



➤ **ISO/IEC-14598 - Parte 1: Visión general**

Básicamente, provee una visión general de las otras cinco partes y explica la relación entre la evaluación del producto software y el modelo de calidad definido en el ISO/IEC 9126. Adicionalmente, hace la presentación del proceso de evaluación desglosado en los pasos siguientes:

- ✓ Establecer los requerimientos de evaluación
- ✓ Especificar la evaluación
- ✓ Planear la evaluación
- ✓ Ejecutar la evaluación.



➤ **ISO/IEC-14598 - Parte 2: Planificación y gestión**

Esta parte contiene los requerimientos y las guías para las funciones de soporte tales como el planeamiento y gestión para la evaluación del producto del software. Fundamentalmente, en esta parte, se planifican las mediciones y las actividades de evaluación. Específicamente, se incluye:

Preparación de las políticas

- ✓ Definición de objetivos organizacionales y de mejora.
- ✓ Identificación de la tecnología.
- ✓ Asignación de responsabilidades.
- ✓ Identificación e implementación de técnicas de evaluación para software desarrollado y adquirido.
- ✓ Entrenamiento en tecnología, recopilación de datos y herramientas.
- ✓ Comparación y administración de mejoras dentro la organización.

➤ **ISO/IEC-14598 - Parte 3: El proceso para desarrolladores**

En esta parte, se provee de los requerimientos y las recomendaciones para la evaluación del producto software cuando la evaluación es conducida en paralelo con el desarrollo y llevada a cabo por el desarrollador.

Se enfoca en el uso de indicadores que pueden predecir la calidad final del producto midiendo los productos intermedios que se desarrollan durante el ciclo de vida.

Esta parte cubre el planeamiento y evaluación de mediciones internas y externas con el fin de asegurar de que la calidad del producto sea incorporada en la fase de desarrollo. Entonces, una vez identificadas las características fundamentales de calidad y el marco de trabajo de mediciones, deben ser definidas las etapas siguientes:

Organización

Los aspectos organizacionales de desarrollo y de soporte deben formar parte de todo el sistema de calidad y del plan de mediciones. Véase la ISO/IEC 14598 - Parte 2.

Planeamiento del proyecto y requerimientos de calidad

El desarrollo y el ciclo de vida de soporte deben ser establecidos y documentados durante el plan de calidad o en otros documentos. Es de vital importancia verificar que el productor y las medidas de control requeridas sean técnicamente factibles, razonables y alcanzables (dentro de los límites de tiempo).

Especificaciones

En esta fase, el desarrollador realiza un mapeo de los requerimientos internos y externos de calidad, con relación a las especificaciones. Los requerimientos de mediciones resultantes de esta fase deben ser un tipo de mapeo entre las especificaciones de requerimientos, requerimientos externos de calidad, requerimientos internos correspondientes de calidad y atributos especificados junto a sus escalas de medición y valores objetivos que contribuyan a la cuantificación de la calidad del software. Todo esto puede enfocarse por proyecto o por producto.

Diseño y planeamiento

Los procedimientos requeridos para el análisis y recopilación de datos necesitan ser definidos. De esta manera, el plan incluirá: cronogramas, designación de responsabilidades, uso de herramientas, bases de datos y entrenamiento especializado requerido. La precisión de las mediciones y técnicas estadísticas deben ser especificadas (véase la ISO/IEC 14598 - Parte 6). En esta fase, también deberá considerarse

cómo los resultados de las mediciones impactarán en el desarrollo; por lo tanto, acciones de contingencia y de mejora, deben ser consideradas.

Montaje (Build) y pruebas

Durante la etapa de montaje y pruebas, las mediciones actuales son recolectadas, se realizan análisis apropiados y se toman acciones necesarias. En cada fase del desarrollo, debe procurarse lograr un montaje primeramente enfocado a las características internas y externas de calidad que definan la calidad global del producto y que puedan ser validadas por los resultados de las pruebas y la experiencia del usuario. Y como etapa final del proyecto, deberá ser conducida una revisión general para determinar la efectividad global del ejercicio de recolección, para identificar costos versus costos, establecer la validez de las métricas usadas e identificar puntos en los cuales podrían obtenerse beneficios para proyectos futuros. El resultado de esta revisión podría retroalimentar directamente el lanzamiento de futuros productos.

➤ **ISO/IEC-14598 - Parte 4: El proceso para adquisidores**

Esta parte provee los requerimientos y las recomendaciones para que la evaluación del producto software sea conducida en función de los compradores que planean adquirir o reusar un producto de software existente o pre desarrollado. Los que adquieren el producto pueden comprar paquetes completos ya sea desarrollado según ciertas especificaciones o predesarrollados para un mercado más general. Los compradores también podrían ser desarrolladores que desean integrar productos estándar en sus propios diseños de software, o tratarse de desarrolladores buscando herramientas específicas de software. Al respecto, cuatro etapas son necesarias:

Establecimiento de los requerimientos

El alcance de la evaluación necesita ser establecido. Los requerimientos para la calidad del software definidos en la ISO/IEC 9126 pueden ser usados como punto de partida pero otros aspectos como el costo y el de cumplimiento a regulaciones deberán ser también considerados. El tiempo de la evaluación necesita ser consistente con los objetivos; enfoques muy tempranos podrían no proporcionar una figura adecuada de la situación mientras que enfoques muy tardíos podrían ser muy limitados en su uso.

Especificación de la evaluación

Durante la redacción de las especificaciones, deben considerarse:

Los requerimientos de calidad a ser evaluados correlacionados con la calidad en uso y métricas externas con prioridades, además de un umbral de aceptación definido.

El alcance y lo que cubren los casos de prueba donde sean aplicables referencias a módulos de evaluación.

Métodos de recolección de mediciones, información requerida y métodos de análisis.

Diseño de la evaluación

El tipo de evaluación depende del tipo de software que está siendo evaluado. Software bajo desarrollo puede ser abordado en puntos discretos durante el desarrollo o cuando esté completo. Un plan de evaluación necesita considerar:

- Necesidades de acceso a la documentación del producto, herramientas de desarrollo y personal.
- Requerimientos en costos y conocimientos.
- Cronograma de evaluación y arreglos de contingencia, hitos claves y criterio para decisiones de evaluación.

- Métodos y herramientas de reporte, procedimientos para la validación y estandarización sobre proyectos futuros.

Ejecución de la evaluación

Aunque esta etapa podría ser simplemente un registro en un libro de seguimiento, podría tenerse la necesidad de incluir:

- Los resultados mismos y la trazabilidad del producto así como información de configuración.
- Registros de análisis, resultados y decisiones.
- Problemas, limitaciones en las mediciones y cualquier compromiso con relación a los objetivos originales.
- Conclusiones sobre los resultados de la evaluación, pero también sobre los métodos empleados.

➤ **ISO/IEC-14598 - Parte 5: El proceso para evaluadores**

Esta parte provee los requerimientos y recomendaciones para la evaluación del producto software cuando la evaluación es conducida por evaluadores independientes. En esta parte, tienen un rol importante los requerimientos de evaluación, las especificaciones de evaluación, el diseño de la evaluación, las actividades de evaluación y el reporte de evaluación. Estas etapas son resumidas a continuación:

Requerimientos de evaluación

Los requerimientos deberían, adicionalmente, definir:

- La extensión de la cobertura (o el alcance).
- Los objetivos de evaluación y métodos de reporte.
- Las calificaciones e independencia requeridas de un evaluador.

Especificación de la evaluación

Las especificaciones, adicionalmente, deberían cubrir:

Definición del alcance y formato en las métricas empleadas identificando como deberán ser derivadas a partir de los requerimientos del producto.

- La identificación de mediciones no determinísticas para asegurar que ciertos niveles de frecuencia y objetividad requeridos sean obtenidos.
- La identificación de métodos de correlación con relación a los resultados de las mediciones. Se tienen identificadas tres subactividades con relación a la especificación de la evaluación:
 - El análisis de la descripción del producto.
 - La especificación de las mediciones a ser realizadas.
 - La verificación de la especificación resultante frente a los requerimientos de evaluación.

➤ **ISO/IEC-14598 - Parte 6: Documentación de los módulos de evaluación**

En esta parte, se provee de las guías para la documentación del módulo de evaluación. Estos módulos representan la especificación del modelo de calidad y las correspondientes métricas internas y externas que serán aplicadas a una evaluación en particular. Incluye métodos y técnicas de evaluación más las mediciones actuales resultantes de su aplicación. En esta parte, también se considera la administración efectiva de complejidades inherentes a las cuestiones de medición. Las actividades de medición coordinadas son una característica para una evaluación efectiva y un plan necesita proveer un cronograma de evaluación que provea al mismo tiempo información óptima cuando la evaluación sea conducida durante el desarrollo. Los módulos de la evaluación son componentes claves de la ISO/IEC 14598-6 y son usados para proveer un formato consistente y repetible de reporte. Dichos módulos proveen de :

- Visibilidad de la información necesitada para cuadrar con requerimientos específicos de calidad.

- Documentación de las interfaces necesarias con herramientas de medición.

La ISO/IEC 14598-6 trata también sobre los requerimientos de la documentación y divide los módulos de evaluación en los seis componentes siguientes:

- a. Introducción. Cubre el control del documento, las relaciones con otros documentos, los requerimientos técnicos y una razón para el módulo.
- b. Alcance. Se relaciona con la características de calidad o subcaracterísticas que deberán ser alcanzadas, el nivel de la evaluación (tomando en cuenta la importancia de la característica, la técnica de evaluación usada incluyendo cualquier teoría necesaria) y la aplicabilidad del módulo.
- c. Referencias.
- d. Definiciones requeridas.
- e. Entradas requeridas. Comprende datos a ser recopilados y métricas a ser calculadas.
- f. Información sobre la interpretación de los resultados.

2.2.2 Establecer el propósito de la evaluación

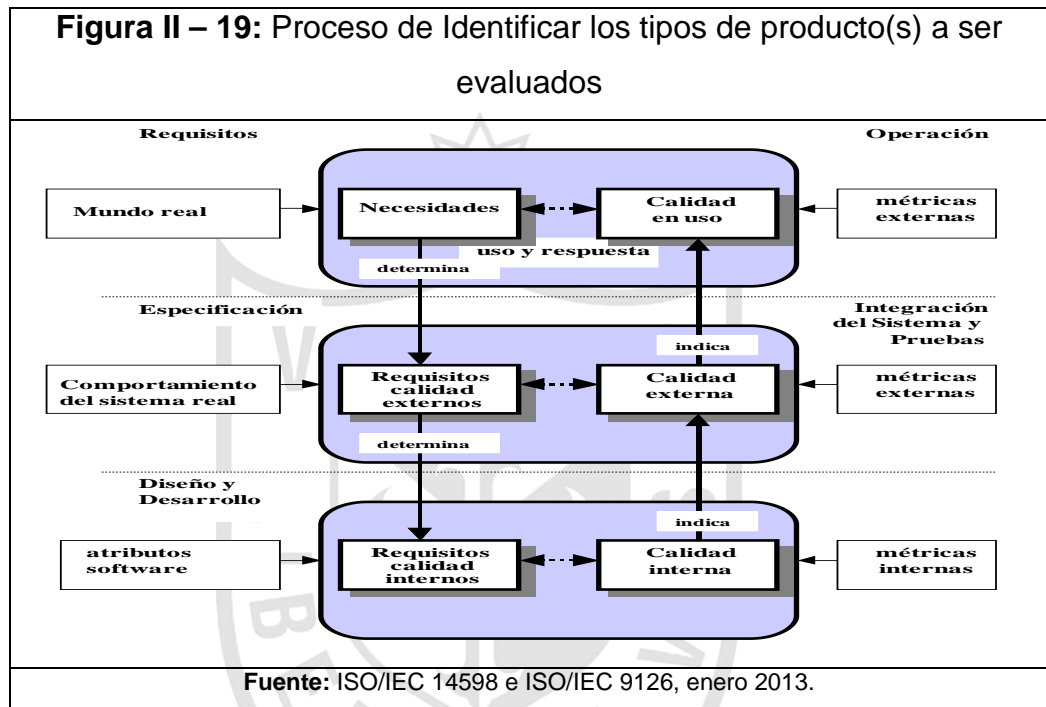
Productos intermedios:

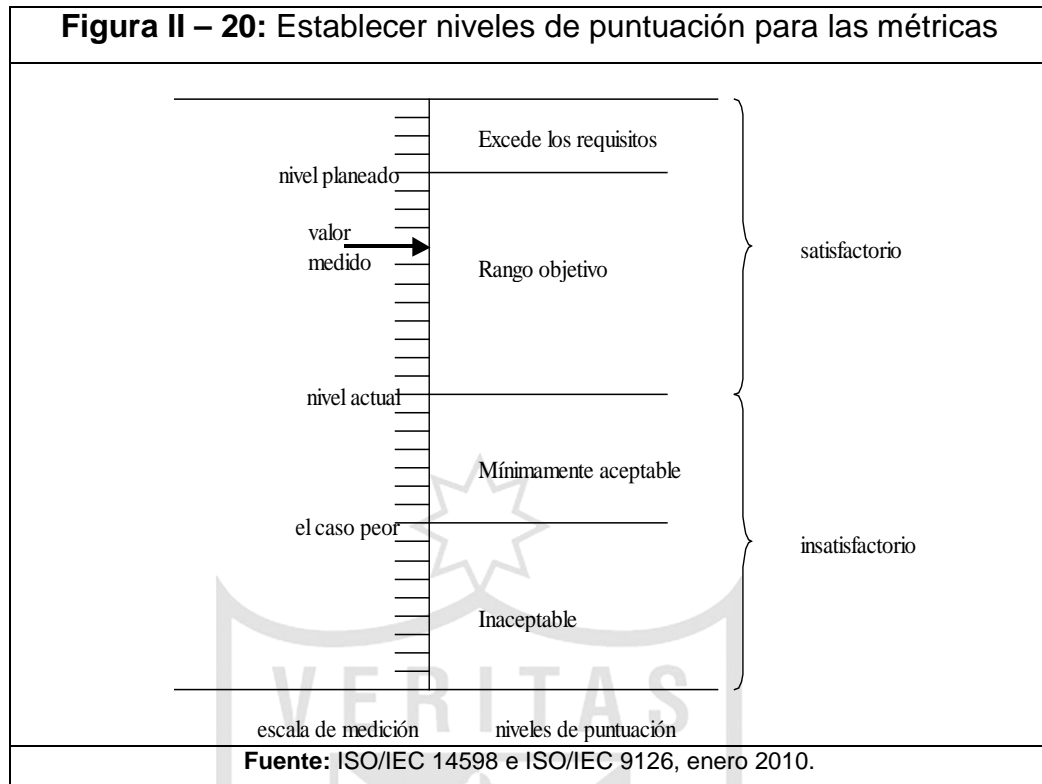
- Decidir sobre la aceptación de un producto intermedio de un subcontratista.
- Decidir cuándo un proceso está completo y cuando remitir los productos al siguiente proceso.
- Predecir o estimar la calidad del producto final.
- Recoger información con objeto de controlar y gestionar el proceso.

Producto final:

- Decidir sobre la aceptación del producto.

- Decidir cuándo publicar el producto.
- Comparar el producto con otros productos competitivos.
- Seleccionar un producto entre productos alternativos.
- Valorar tanto el aspecto positivo como negativo cuando está en uso.
- Decidir cuándo mejorar o reemplazar un producto.





2.3 Modelo teórico o conceptual

La tecnología de información está transformando la manera en que nos relacionamos y trabajamos. Ya no es necesario un centro de reunión donde se concentren las personas y la información de las empresas.

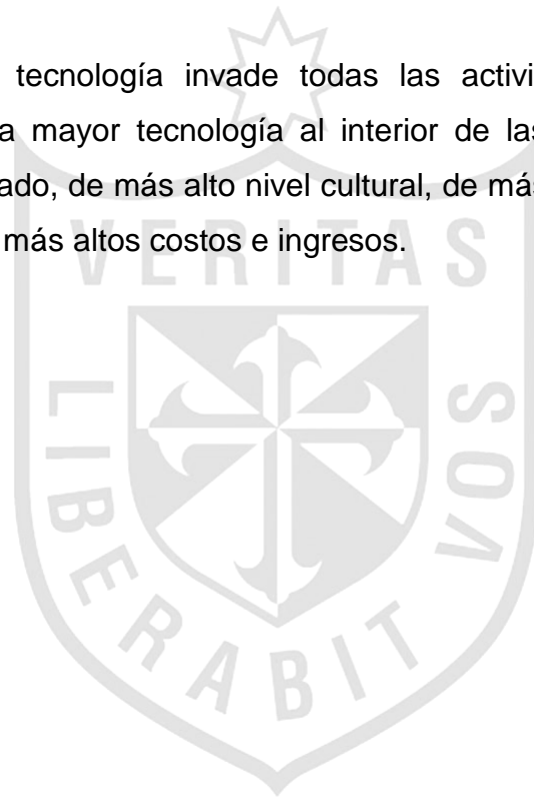
Las personas y organizaciones pueden colaborar y compartir información desde puntos distantes. El reto para las empresas es transformar esta nueva herramienta en algo que agregue valor a su producto o servicio y maximizar los beneficios de esta nueva posibilidad. Actualmente vivimos en un mundo donde los cambios son constantes y donde la creación de alianzas estratégicas surge como una necesidad para lograr ser competitivo.

Las empresas empezarán a intensificar sus alianzas con otras empresas para poder ofrecer los mejores productos y servicios, con la mejor calidad, menor precio y con tiempos de entrega exactos. Las organizaciones deberán enfocarse en la creación y desarrollo continuo de competencias claves

específicas para poder realizar alianzas que sean exitosas. El Diseño de una Metodología de Certificación de Productos de Software Orientado al Sector Público es clave para otorgar un producto de calidad.

En la medida que la sociedad incorpora mayores conocimientos y nuevas tecnologías, va aumentando su eficiencia y productividad y en consecuencia los niveles de todas las empresas que operan dentro de este grupo social.

La tecnología invade todas las actividades. Esto conlleva mayores desafíos, a mayor tecnología al interior de las empresas, se exige personal más calificado, de más alto nivel cultural, de más alta escolaridad y en consecuencia exige más altos costos e ingresos.



CAPÍTULO III

DISEÑO DE UNA METODOLOGÍA DE CERTIFICACIÓN DE PRODUCTOS SOFTWARE ORIENTADO AL SECTOR PÚBLICO

3.1 Descripción general

Del capítulo anterior, podemos mencionar que de acuerdo con los estándares, metodologías y modelos descritos, para poder establecer la Metodología de Certificación de Productos de Software aplicada al sector público, se ha considerado una combinación de tareas y actividades establecidas en dicha información, lo que nos ha permitido que con los recursos existentes tanto en presupuesto como en personas, se pueda definir actividades, responsables, entregables, que si bien van a exigir un mayor consumo de horas, el resultado esperado es tener unos productos con la menor incidencia de errores en el despliegue en producción.

Es necesario indicar que a la actualidad ninguna institución del estado, tiene definido establecer dentro de su área de sistemas el concepto de fábrica de software, y se menciona esto porque muchas de las metodologías y/o modelos están orientados a este tipo de concepto, y de acuerdo con la información se puede considerar la existencia de n roles que deberían establecerse para poder concretar esta implementación, lo escaso en casi la totalidad de las instituciones públicas es el recurso humano.

3.2 Método de elaboración de la metodología de certificación

Se entiende por metodología a la colección de documentación formal referente a los procesos, políticas y procedimientos que intervienen en las diferentes etapas de la ejecución de un proceso.¹³

Diseñar una metodología para la Certificación de Productos de Software, y aplicada al sector público, la cual ha sido elaborada con un enfoque de procesos enmarcado en las fases del Proceso RUP, CMMI y MOPROSOFT. En esta metodología, las actividades de mantenimiento y desarrollo se muestran a través de un flujo de trabajo donde se detalla la interrelación entre el personal de sistemas y el usuario. Nuestra metodología es, además de un proceso, un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas de software como son las aplicaciones Web o aplicaciones cliente / servidor, las de Mainframe, las de código oculto.

Esta metodología está orientada al desarrollo de Nuevos Proyectos de TI sean estos de Servicio de Mantenimiento y Desarrollo de Sistemas de Información.

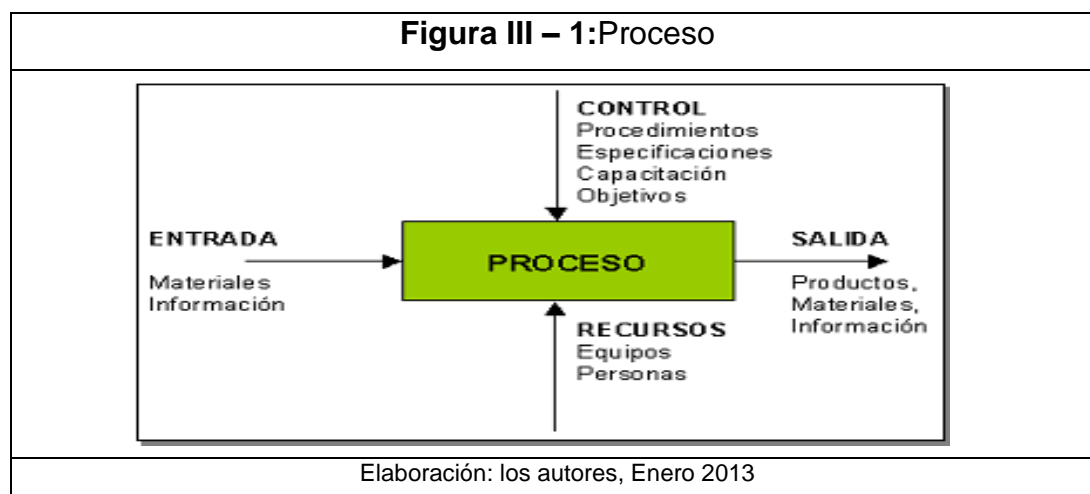
Para el diseño de la investigación se identificaron los siguientes puntos, sobre los cuales se desarrollaría el proyecto:

3.2.1 Definir Metodología de Certificación de Productos Software Orientado al Sector Público, de acuerdo al procedimiento de inspección, alineado a la NTP 12207

La metodología de mantenimiento ha sido definida con un enfoque de procesos enmarcado en las fases de Desarrollo de Software. Definimos proceso como el conjunto de actividades mutuamente

¹³Blanco, S. (s.f.) Marble Station. Recuperado enero 2013. <http://www.marblestation.com/?p=644>

relacionadas o que interactúan, las cuales transforman elementos de entrada en resultados.

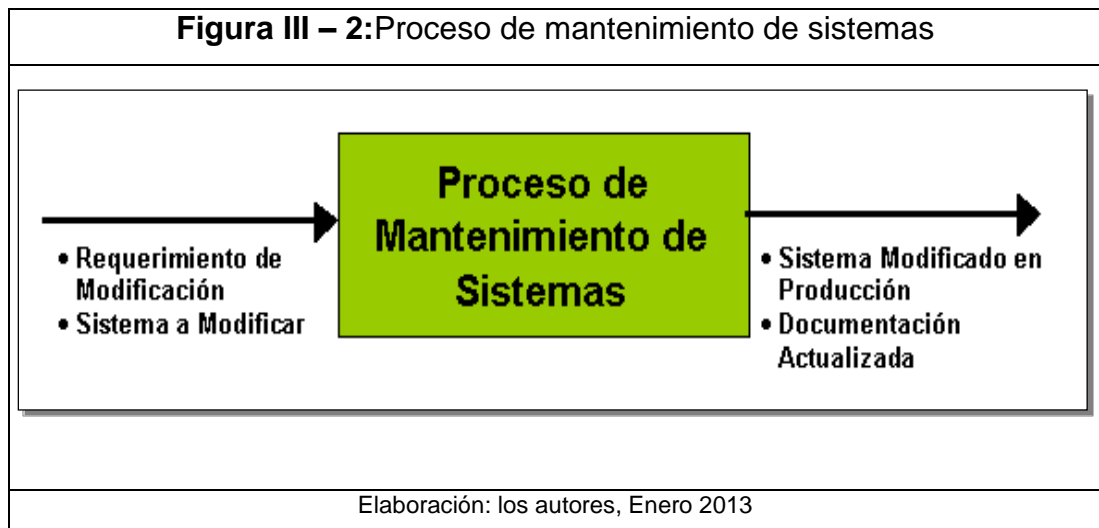


Hemos clasificado los requerimientos en desarrollo y mantenimiento:

Cuadro III - 1: Tipo de requerimientos	
Tipo de requerimiento	Descripción
Desarrollo	Son aquellos requerimientos nuevos, para el desarrollo de un producto software, sean estos desarrollados de forma interna o por algún proveedor externo a la institución.
Mantenimiento	Correctivo. Son aquellos cambios precisos para corregir errores del producto software.
	Evolutivo. Son las incorporaciones, modificaciones y eliminaciones necesarias en un producto software para cubrir la expansión o cambio en las necesidades del usuario.
	Adaptativo. Son las modificaciones que afectan a los entornos en los que el sistema opera, por ejemplo, cambios de configuración del hardware, software de base, gestores de base de datos, comunicaciones, etc.

	Perfectivo.	Son las acciones llevadas a cabo para mejorar la calidad interna de los sistemas en cualquiera de sus aspectos: reestructuración del código, definición más clara del sistema y optimización del rendimiento y eficiencia.
Elaboración: los autores, Enero 2013		

Un proceso define quién está haciendo qué, cuándo y cómo alcanzar un determinado objetivo. En el mantenimiento de sistemas, el objetivo es mejorar un software existente o corregirlo.



Los requerimientos se clasifican en:

Cuadro III - 2: Complejidad de requerimientos

Requerimiento	Evaluar	Complejidad	Descripción
Desarrollo	<ul style="list-style-type: none">☞ Mayores a 500 horas.☞ Costo mayor a 30,000 dólares	Alta	<p>Involucran la implementación de un nuevo módulo o muchas funcionalidades nuevas a un sistema.</p> <p>Es necesario indicar que todos los mantenimientos cuya elaboración incurran en los márgenes de ser mayores a 500 horas o mayores a \$30,000 de inversión, de acuerdo a la política interna de desarrollos históricos estos serán considerados como proyectos de desarrollo.</p>
Mantenimiento	<ul style="list-style-type: none">☞ Menores a 500 horas☞ Costo menos a 30,000 dólares	Baja	<p>Corresponden a requerimientos que involucran modificaciones sencillas a programas ya existentes. Por ejemplo:</p> <ul style="list-style-type: none">☞ Agregar un campo a una pantalla.☞ Modificar la ubicación de campos en un reporte.

Cuadro III - 2: Complejidad de requerimientos			
Requerimiento	Evaluar	Complejidad	Descripción
		Medio	<p>Corresponden a requerimientos que involucran la implementación de funcionalidades puntuales a un módulo o sistema. Por</p> <p>Ejemplo:</p> <ul style="list-style-type: none"> ☞ Implementar el mantenimiento de datos de una tabla. ☞ Implementar un proceso que actualice múltiples tablas.
		Alta	<p>Corresponden a requerimientos que involucran la implementación de un nuevo módulo o muchas funcionalidades nuevas a un sistema. Por ejemplo:</p> <ul style="list-style-type: none"> ☞ Desarrollo de varias opciones para un proceso determinado.
Elaboración: los autores, Enero 2013			

Nota: los proyectos tercerizados, son considerado como nuevo desarrollo

Para poder definir esta metodología, es necesario tomar en cuenta todas las actividades que actualmente se realizan y optimizar aquellas que por falta de definición y/o estrategia no se estén utilizando, para ello se menciona seguidamente parte del proceso de atención de requerimientos de software.

En este proceso la primera actividad de la solicitud es la evaluación técnica económica que establece el usuario de negocio, el cual servirá para la estimación y/o priorización del comité de proyectos, es necesario indicar que el requerimiento puede ser denegado.

En este caso, se notifica al usuario y acaba el proceso, de no ser así, se realiza el registro de los requerimientos de mantenimiento y desarrollo recibidos, con el fin de llevar el control de las mismas y de proporcionar, si fuera necesario, datos estadísticos de los requerimientos recibidos o atendidas en un determinado periodo, sistemas que se han visto afectados por los cambios, en qué medida y el tiempo empleado en la resolución de dichos cambios. Se lleva un catálogo o listado de requerimientos de mantenimiento y desarrollo sobre los sistemas de información, en él se registran una serie de datos que nos permiten disponer de la información antes mencionada.

Una vez registrado el requerimiento e identificado el tipo: Desarrollo o mantenimiento y su origen, se determina de quién es la responsabilidad de atender el requerimiento. Cuando el requerimiento sea remitido, se registra en el catálogo "listado" de requerimientos de desarrollo, mantenimiento y continúa el proceso.

Fases / Proceso			Proceso General ¹⁴	Proceso Básico ¹⁵	Observaciones
Inicio	Pre Inicio	Plantilla de acta reunión	X		
		Lista maestra de requerimientos	X		
		Cartilla mantenimiento	X		
		Checklist mantenimiento	X		
	Inicio	Plantilla acta de reunión	X	X	
		Lista maestra de requerimientos	X		Para el caso de Proceso básico lo que existe es un documento de pre análisis.
		Plantilla de documento de Preanálisis	X	X	Para el proceso básico incluyen los requerimientos solicitados.
		Lista de Incidencias	X	X	Los documentos serán en base al tipo de proceso efectuado.

¹⁴Proceso General aplica para todo nuevo desarrollo de proyecto

¹⁵Proceso Básico: aplica para mantenimiento, o cuando decida ampliar o cambiar algunos módulos del sistema desarrollado

Fases / Proceso		Proceso General ¹⁴	Proceso Básico ¹⁵	Observaciones
	Cartilla mantenimiento	X	X	Basado en el tipo de proceso efectuado.
	Checklist Mantenimiento	X	X	Basado en el tipo de proceso efectuado.
	Matriz de trazabilidad de requerimientos a documentos	X	X	Basado en el tipo de proceso efectuado.
Elaboración	Documento de análisis	X	X	Basado en el tipo de proceso efectuado.
	Lista de Incidencias	X	X	
	Cartilla mantenimiento	X	X	
	Checklist Mantenimiento	X	X	
	Lista maestra de requerimientos	X		Para el caso de Proceso básico lo que existe es un documento de pre análisis.
	Plantilla de matriz de trazabilidad requerimientos a	X	X	Basado en el tipo de proceso efectuado.

Fases / Proceso		Proceso General ¹⁴	Proceso Básico ¹⁵	Observaciones	
		documentos mantenimiento.			
Construcción	Construcción	Documento de análisis actualizado.	X	X	
		Documentos de Aceptación de Pruebas Funcionales (para pruebas internas, de calidad y de aceptación).	X	X	
		Producto de software generado.	X	X	
		Plantilla lista incidencias.	X	X	Basado en el tipo de proceso efectuado.
		Cartilla mantenimiento.	X	X	
		Checklist Mantenimiento.	X	X	
		Lista maestra de requerimientos para mantenimiento.	X		Para el caso de Proceso básico lo que existe es un documento de pre análisis.
		Plantilla de matriz de	X	X	Basado en el tipo de proceso

Fases / Proceso			Proceso General ¹⁴	Proceso Básico ¹⁵	Observaciones
		trazabilidad requerimientos a documentos mantenimiento.			efectuado.
		Producto de software generado.	X	X	
	Pruebas internas	Documento de análisis .actualizado.	X	X	
		Checklist DBA.	X	X	
		Checklist de analista.	X	X	
		Checklist de programador.	X	X	
		Plantilla lista incidencias.	X	X	
		Cartilla mantenimiento.	X	X	
		Plantilla de matriz de trazabilidad requerimientos a documentos mantenimiento.	X	X	Basado en el tipo de proceso efectuado.
		Pruebas de calidad	Documento de análisis.	X	X
	Documento de aceptación de pruebas funcionales.		X	X	

Fases / Proceso			Proceso General ¹⁴	Proceso Básico ¹⁵	Observaciones
		Documento de pase a QA / Producción.	X	X	
		Plantilla lista incidencias.	X	X	
		Cartilla mantenimiento.	X	X	
	Pruebas de aceptación	Documento de análisis.	X	X	
		Documento de aceptación de pruebas funcionales.	X	X	
		Plantilla lista incidencias.	X	X	
		Cartilla mantenimiento.	X	X	
Transición	Manual de Usuario.	Desarrollo	Mantenimiento		
	Manual Sistema.	Desarrollo	Mantenimiento		
	Manual de administración e Instalación.	Desarrollo	Mantenimiento		
	Instalación y configuración	Ejecutable / Código fuente	Ejecutable / Código fuente		

Posteriormente, según se trate de un mantenimiento correctivo o evolutivo, se verifica y reproduce el problema, o se estudia la viabilidad del cambio propuesto por el usuario. En ambos casos, se estudia el alcance de la modificación. Hay que analizar las alternativas de solución identificando, según el tipo de mantenimiento de que se trate, cuál es la más adecuada. El plazo y urgencia de la solución del requerimiento se establece de acuerdo con el estudio anterior.

La definición de la solución incluye el estudio del impacto de la solución propuesta para el requerimiento en los sistemas de información afectados. Mediante el análisis de dicho estudio, la persona responsable del Proceso de Mantenimiento o Desarrollo valora el esfuerzo y coste necesario para la implementación de dicho desarrollo o la modificación.

Cuando el requerimiento lo amerita se hace uso de los diagrama de caso de uso, diagramas de colaboración y diagramas de actividades (que son parte de la notación UML) para explicar el análisis de los requerimientos que incorporan una nueva funcionalidad al sistema.

Las tareas de los procesos de desarrollo que va a ser necesario realizar son determinadas en función de los componentes del sistema actual afectados por la modificación. Estas tareas pertenecen a actividades de los procesos Análisis, Diseño, Construcción e Implantación.

Por último, y antes de la aceptación del usuario, es preciso establecer un plan de pruebas de regresión que asegure la integridad del sistema de información afectado.

La mejor forma de mantener el coste de mantenimiento bajo control es una gestión del Proceso de Mantenimiento y Desarrollo efectiva y comprometida. Por lo tanto, es necesario registrar de forma disciplinada los cambios realizados en los sistemas de información y

en su documentación. Esto repercutirá directamente en la mayor calidad de los sistemas resultantes.

3.2.2 Diseñar una Metodología de Certificación de Productos Software Orientado al Sector Público

Proporcionando un flujo de trabajo que permita cumplir los requisitos que contempla un requerimiento de desarrollo de nuevo sistema de información o de modificación y/o actualización, hasta la satisfacción de los usuarios.

3.2.3 Descripción de las Fases de la Metodología de Certificación de Productos de Software

A continuación, se describen las actividades a realizar en cada una de las fases utilizadas según la metodología propuesta.

- Inicio
- Elaboración
- Construcción
- Transición

Cada Desarrollo de Sistemas de Información será considerado en la metodología como un proyecto.

3.2.4 Documentos de entrada y salidas de entregables por cada subproceso

Seguidamente se presenta un cuadro resumen referente a las fases, proceso, sub proceso, entregables de acuerdo con el tipo de proceso establecido.

Fases / Proceso			Proceso General ¹⁶	Proceso Básico ¹⁷	Observaciones
Inicio	Pre Inicio	Plantilla de acta reunión	X		
		Lista maestra de requerimientos	X		
		Cartilla mantenimiento	X		
		Checklist mantenimiento	X		
	Inicio	Plantilla acta de reunión	X	X	
		Lista maestra de requerimientos	X		Para el caso de Proceso básico lo que existe es un documento de pre análisis.
		Plantilla de documento de preanálisis	X	X	Para el proceso básico incluyen los requerimientos solicitados.
		Lista de Incidencias	X	X	Los documentos serán en base al tipo de proceso efectuado.

¹⁶Proceso General aplica para todo nuevo desarrollo de proyecto

¹⁷Proceso Básico: aplica para mantenimiento, o cuando decida ampliar o cambiar algunos módulos del sistema desarrollado

Fases / Proceso		Proceso General ¹⁶	Proceso Básico ¹⁷	Observaciones
	Cartilla mantenimiento	X	X	Basado en el tipo de proceso efectuado.
	Checklist Mantenimiento	X	X	Basado en el tipo de proceso efectuado.
	Matriz de trazabilidad de requerimientos a documentos	X	X	Basado en el tipo de proceso efectuado.
Elaboración	Documento de análisis	X	X	Basado en el tipo de proceso efectuado.
	Lista de Incidencias	X	X	
	Cartilla mantenimiento	X	X	
	Checklist Mantenimiento	X	X	
	Lista maestra de requerimientos	X		Para el caso de Proceso básico lo que existe es un documento de pre análisis.
	Plantilla de matriz de trazabilidad requerimientos a	X	X	Basado en el tipo de proceso efectuado.

Fases / Proceso		Proceso General ¹⁶	Proceso Básico ¹⁷	Observaciones	
		documentos mantenimiento.			
Construcción	Construcción	Documento de análisis actualizado.	X	X	
		Documentos de Aceptación de Pruebas Funcionales (para pruebas internas, de calidad y de aceptación).	X	X	
		Producto de software generado.	X	X	
		Plantilla lista incidencias.	X	X	Basado en el tipo de proceso efectuado.
		Cartilla mantenimiento.	X	X	
		Checklist Mantenimiento.	X	X	
		Lista maestra de requerimientos para mantenimiento.	X		Para el caso de Proceso básico lo que existe es un documento de pre análisis.
		Plantilla de matriz de	X	X	Basado en el tipo de proceso

Fases / Proceso			Proceso General ¹⁶	Proceso Básico ¹⁷	Observaciones
		trazabilidad requerimientos a documentos mantenimiento.			efectuado.
		Producto de software generado.	X	X	
	Pruebas internas	Documento de análisis .actualizado.	X	X	
		Checklist DBA.	X	X	
		Checklist de analista.	X	X	
		Checklist de programador.	X	X	
		Plantilla lista incidencias.	X	X	
		Cartilla mantenimiento.	X	X	
		Plantilla de matriz de trazabilidad requerimientos a documentos mantenimiento.	X	X	Basado en el tipo de proceso efectuado.
	Pruebas de calidad	Documento de análisis.	X	X	
		Documento de aceptación de pruebas funcionales.	X	X	

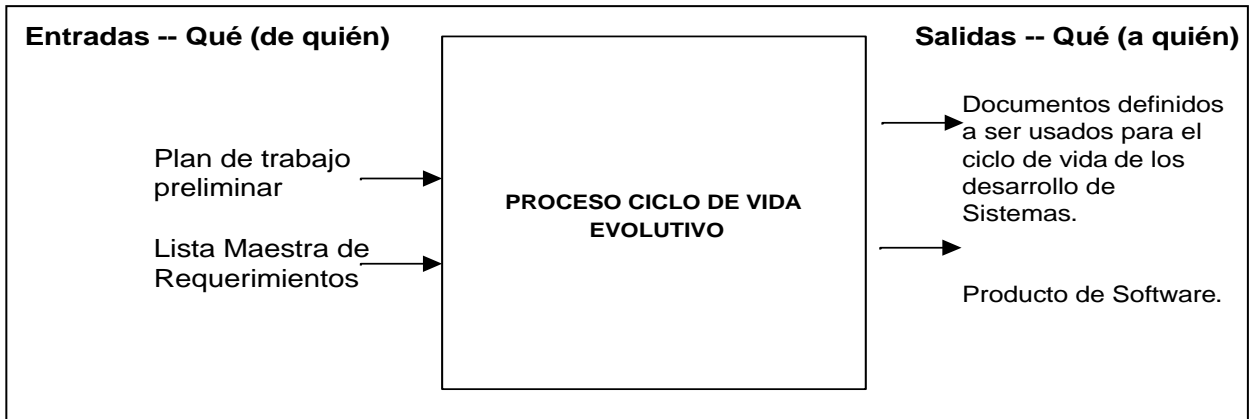
Fases / Proceso			Proceso General ¹⁶	Proceso Básico ¹⁷	Observaciones
		Documento de pase a QA / Producción.	X	X	
		Plantilla lista incidencias.	X	X	
		Cartilla mantenimiento.	X	X	
	Pruebas de aceptación	Documento de análisis.	X	X	
		Documento de aceptación de pruebas funcionales.	X	X	
		Plantilla lista incidencias.	X	X	
		Cartilla mantenimiento.	X	X	
	Transición	Manual de Usuario.	Desarrollo	Mantenimiento	
Manual Sistema.		Desarrollo	Mantenimiento		
Manual de administración e Instalación.		Desarrollo	Mantenimiento		
Instalación y configuración		Ejecutable / Código fuente	Ejecutable / Código fuente		

Diagrama de Contexto Del Proceso

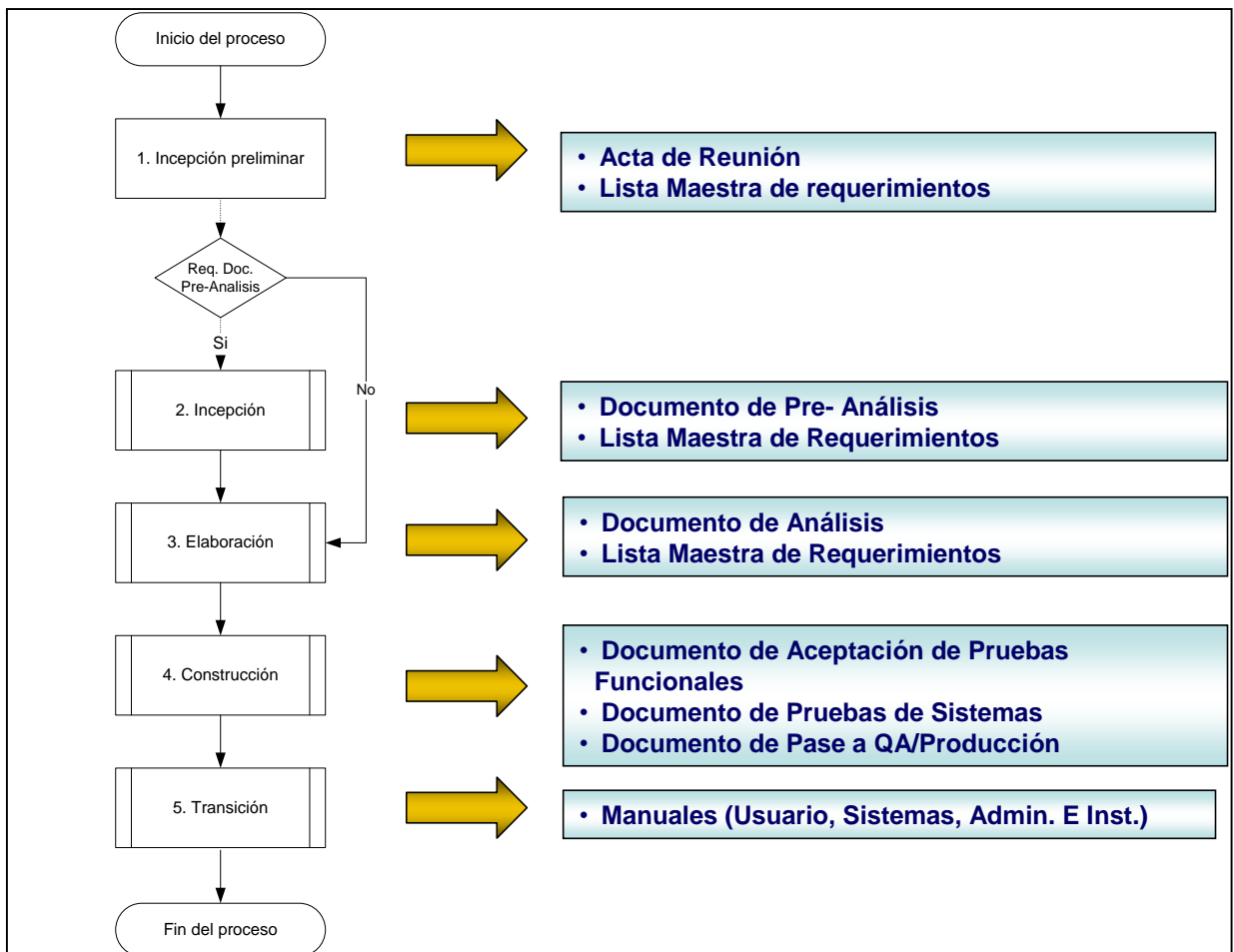
Este gráfico representa la lógica general de los proceso.

Los procesos son:

- General
- Básico



Desarrollo del Proceso General



Desarrollo de los entregables

- MCPS.1. Plantilla de acta de reunión
- MCPS.2. Lista maestra de requerimientos
- MCPS.3 Cartilla mantenimiento
- MCPS.4 Plantilla de documento de preanálisis
- MCPS.5 Plantilla documento de análisis
- MCPS.6 Lista de Incidencias
- MCPS.7 Matriz de trazabilidad de requerimientos a documentos para mantenimiento
- MCPS.8 Matriz de trazabilidad de requerimientos a objetos
- MCPS.9 Plantilla de documentos de aceptación de pruebas funcionales
- MCPS.10 Plantilla prueba de sistemas
- MCPS.11 Plantilla de documento de pase a QA/producción
- MCPS.12 Plantilla de documento casos de pruebas
- MCPS.13 Control de calidad del pase a producción. (Checklist)
- MCPS.14 Checklist mantenimiento
- MCPS.15. Checklist analista / programador
- MCPS.16 Plantilla de informe de pruebas
- MCPS.17 Plantilla de documento de especificación de ambientes
- MCPS.18 Plantilla de manual de usuario
- MCPS.19 Plantilla de manual sistemas
- MCPS.20 Plantilla de manual de administración e instalaciones

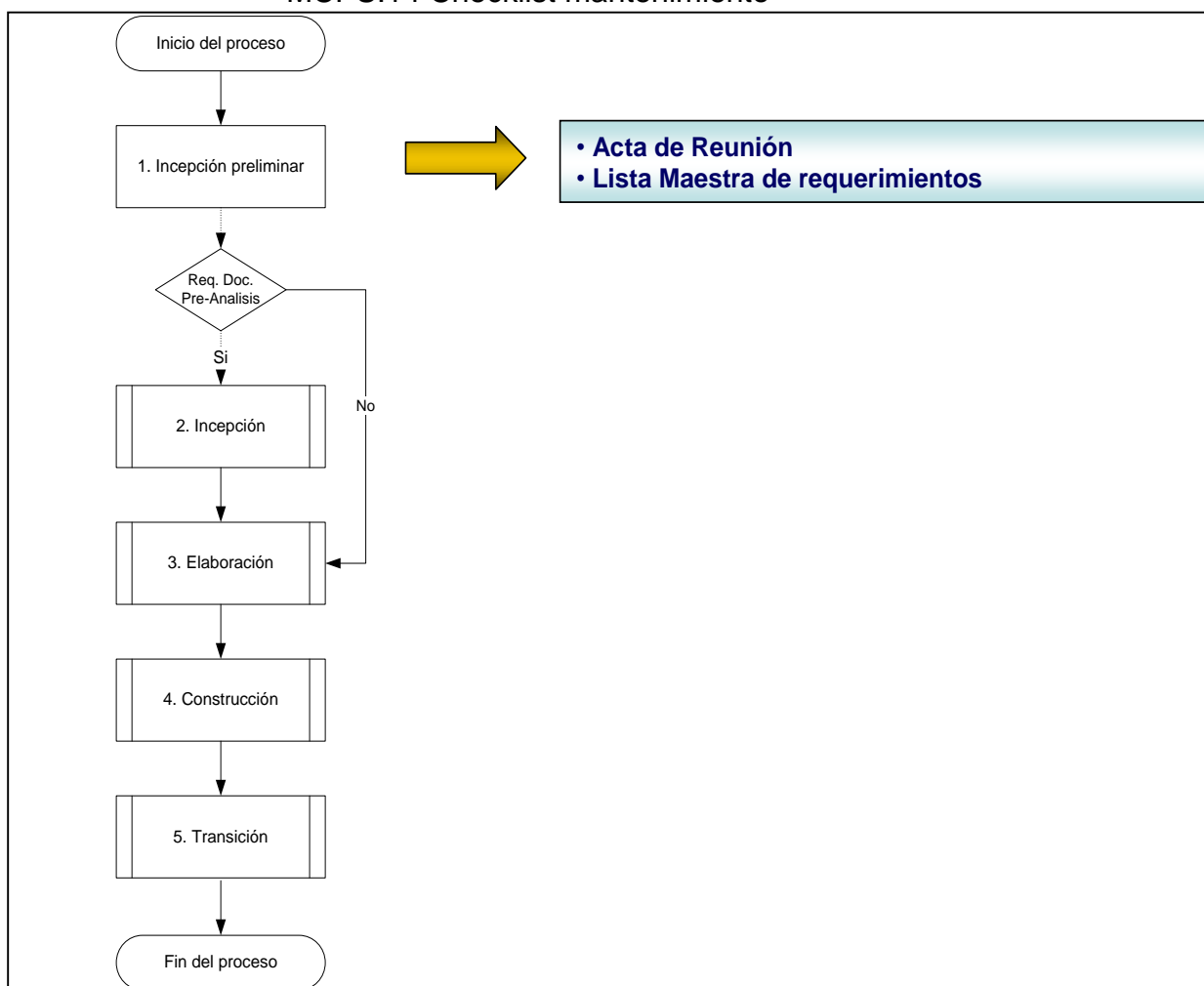
Las fases de los sub proceso general

- Sub proceso de Preinicio
- Sub proceso de Inicio
- Sub proceso de elaboración
- Sub proceso de construcción
- Sub proceso de construcción – pruebas internas
- Sub proceso de construcción – pruebas de calidad
- Sub proceso de construcción – pruebas de aceptación
- Sub proceso de transición

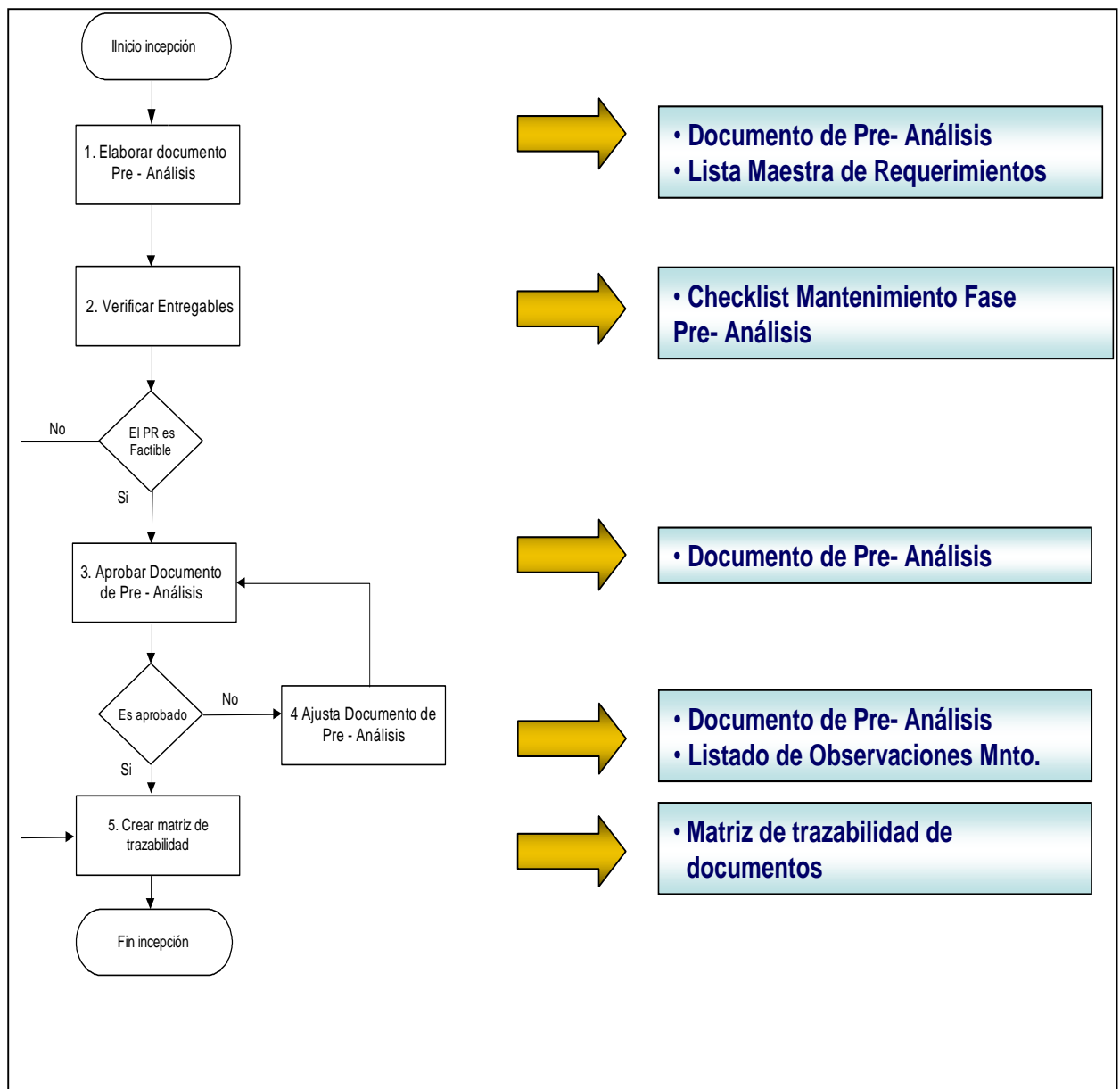
Subproceso de Pre Inicio

Desarrollo de los artefactos

- MCPS.1. Plantilla de acta de reunión
- MCPS.2. Lista maestra de requerimientos
- MCPS.3 Cartilla mantenimiento
- MCPS.14 Checklist mantenimiento



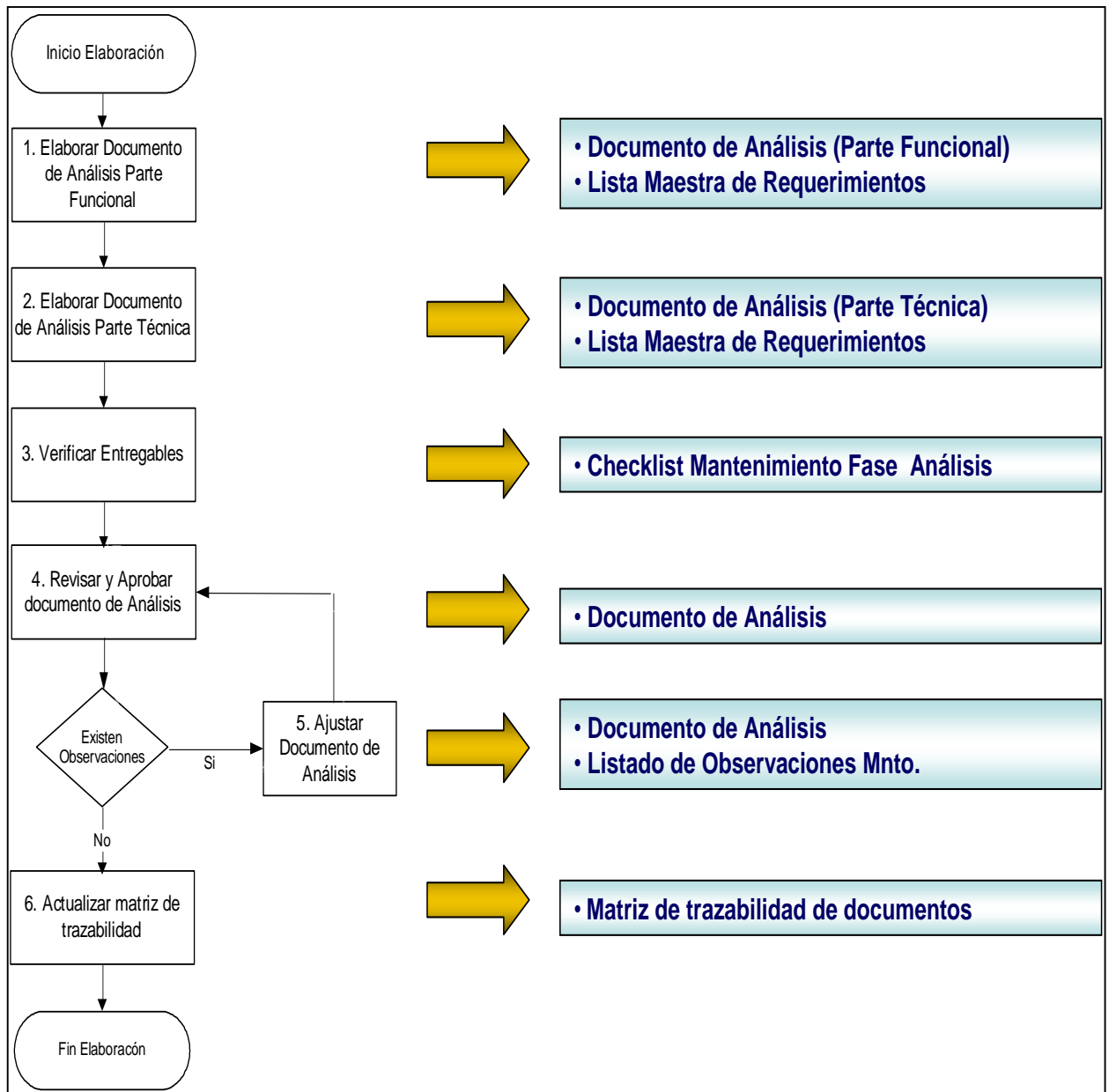
Subproceso de inicio



Desarrollo de los entregables

- MCPS.1. Plantilla de acta de reunión
- MCPS.2. Lista maestra de requerimientos
- MCPS.4 Plantilla de documento de preanálisis
- MCPS.6 Lista de Incidencias
- MCPS.3 Cartilla mantenimiento
- MCPS.14 Checklist mantenimiento
- MCPS.7 Matriz de trazabilidad de requerimientos a documentos para mantenimiento

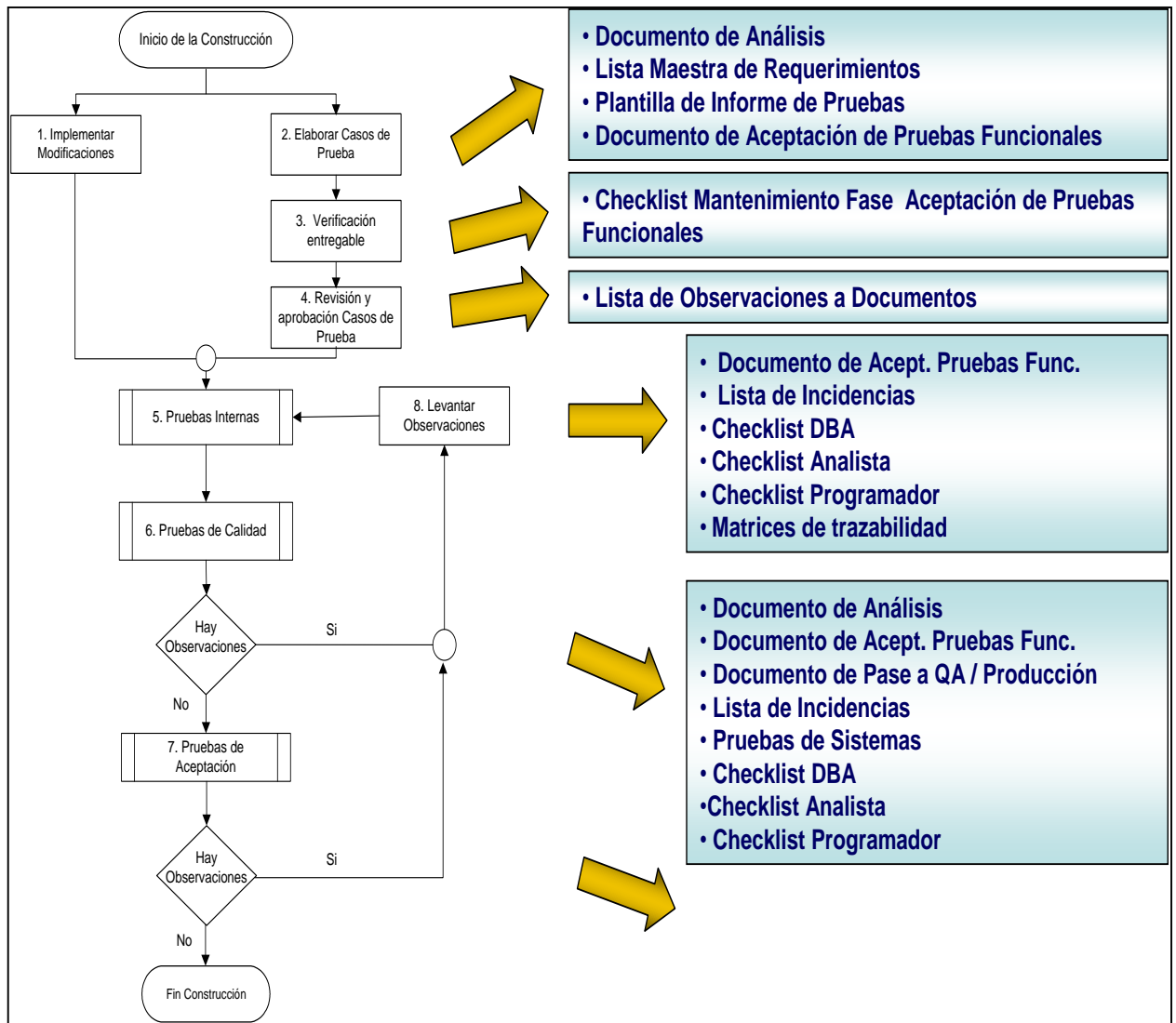
Subproceso de elaboración



Desarrollo de los entregables

- MCPS.5 Plantilla documento de análisis
- MCPS.6 Lista de Incidencias
- MCPS.3 Cartilla mantenimiento
- MCPS.14 Checklist mantenimiento
- MCPS.2. Lista maestra de requerimientos
- MCPS.7 Matriz de trazabilidad de requerimientos a documentos para mantenimiento

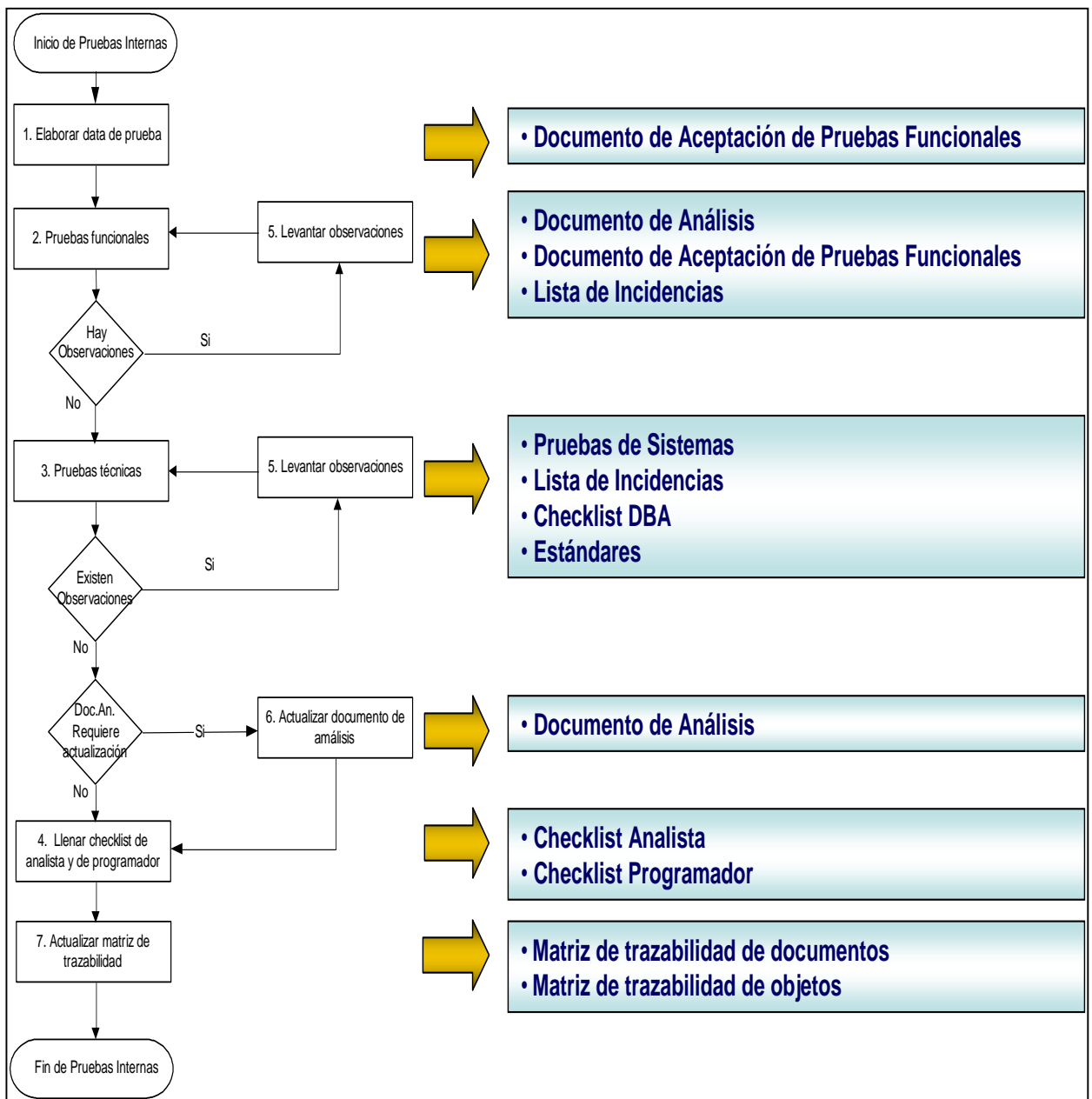
Subproceso de construcción



Desarrollo de los artefactos

- MCPS.5 Plantilla documento de análisis actualizado
- MCPS.9 Plantilla de documentos de aceptación de pruebas funcionales
(para pruebas internas, de calidad y de aceptación)
- Producto de software generado.
- MCPS.6 Lista de Incidencias
- MCPS.3 Cartilla mantenimiento
- MCPS.14 Checklist mantenimiento
- MCPS.2. Lista maestra de requerimientos
- MCPS.7 Matriz de trazabilidad de requerimientos a documentos para mantenimiento

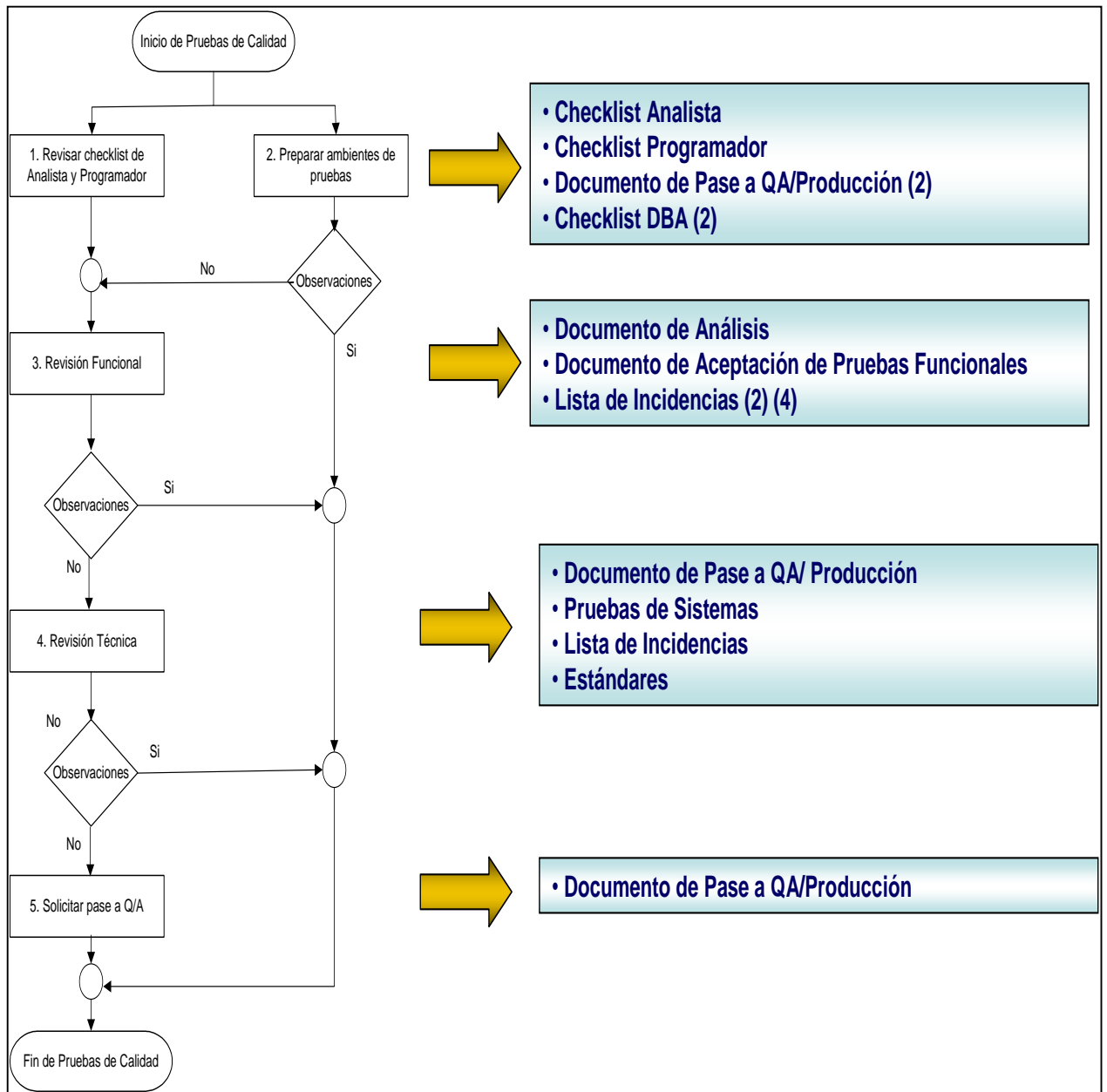
Subproceso de construcción: Pruebas internas



Desarrollo de los entregables

- MCPS.5 Plantilla documento de análisis actualizado
- MCPS.15. Checklist analista / programador / DBA
- MCPS.6 Lista de Incidencias
- MCPS.3 Cartilla mantenimiento
- MCPS.7 Matriz de trazabilidad de requerimientos a documentos para mantenimiento

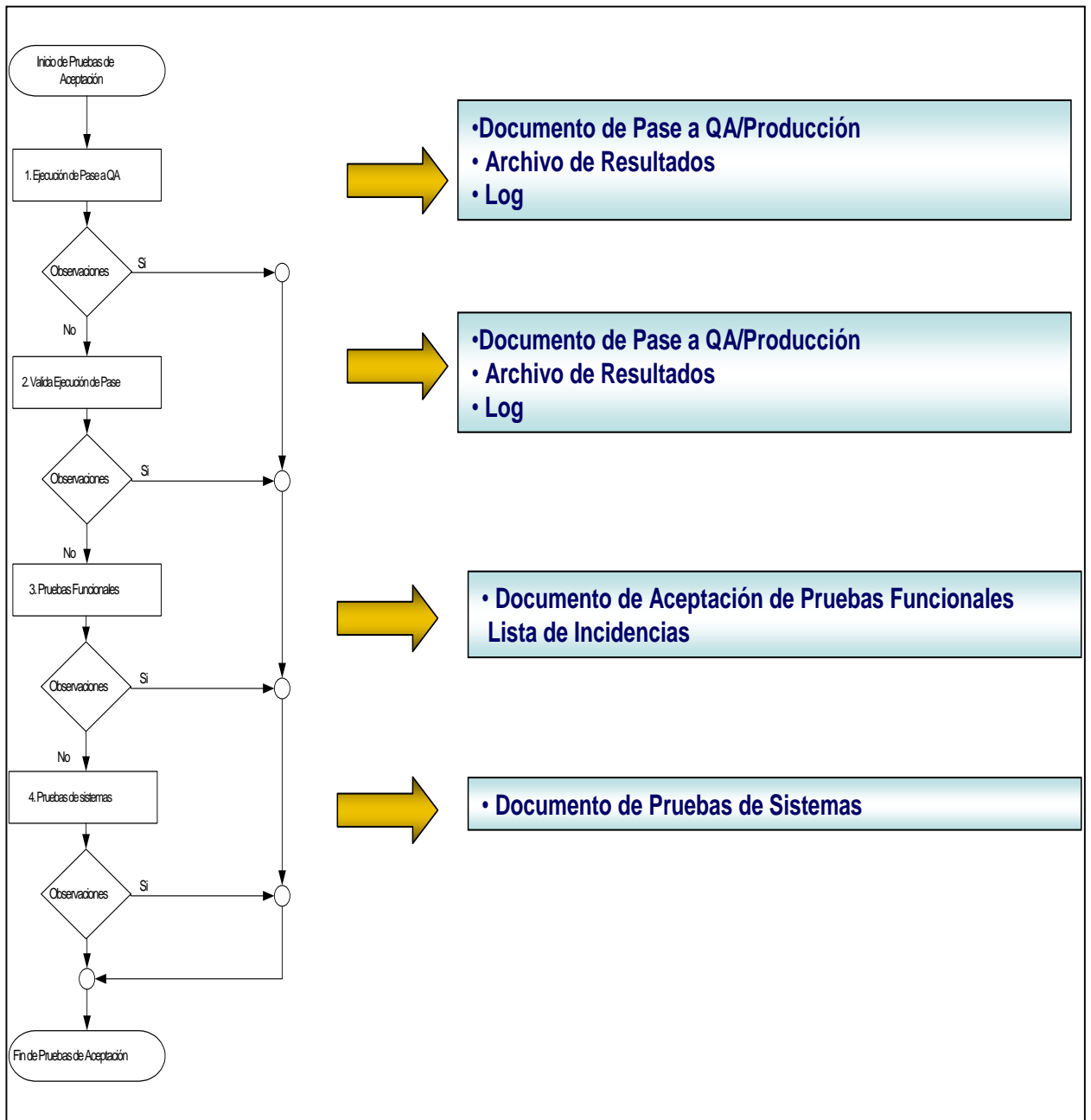
Subproceso de construcción: Pruebas de calidad



Desarrollo de los entregables

- MCPS.5 Plantilla documento de análisis actualizado
- MCPS.9 Plantilla de documentos de aceptación de pruebas funcionales
- MCPS.11 Plantilla de documento de pase a QA/producción
- MCPS.6 Lista de Incidencias
- MCPS.3 Cartilla mantenimiento

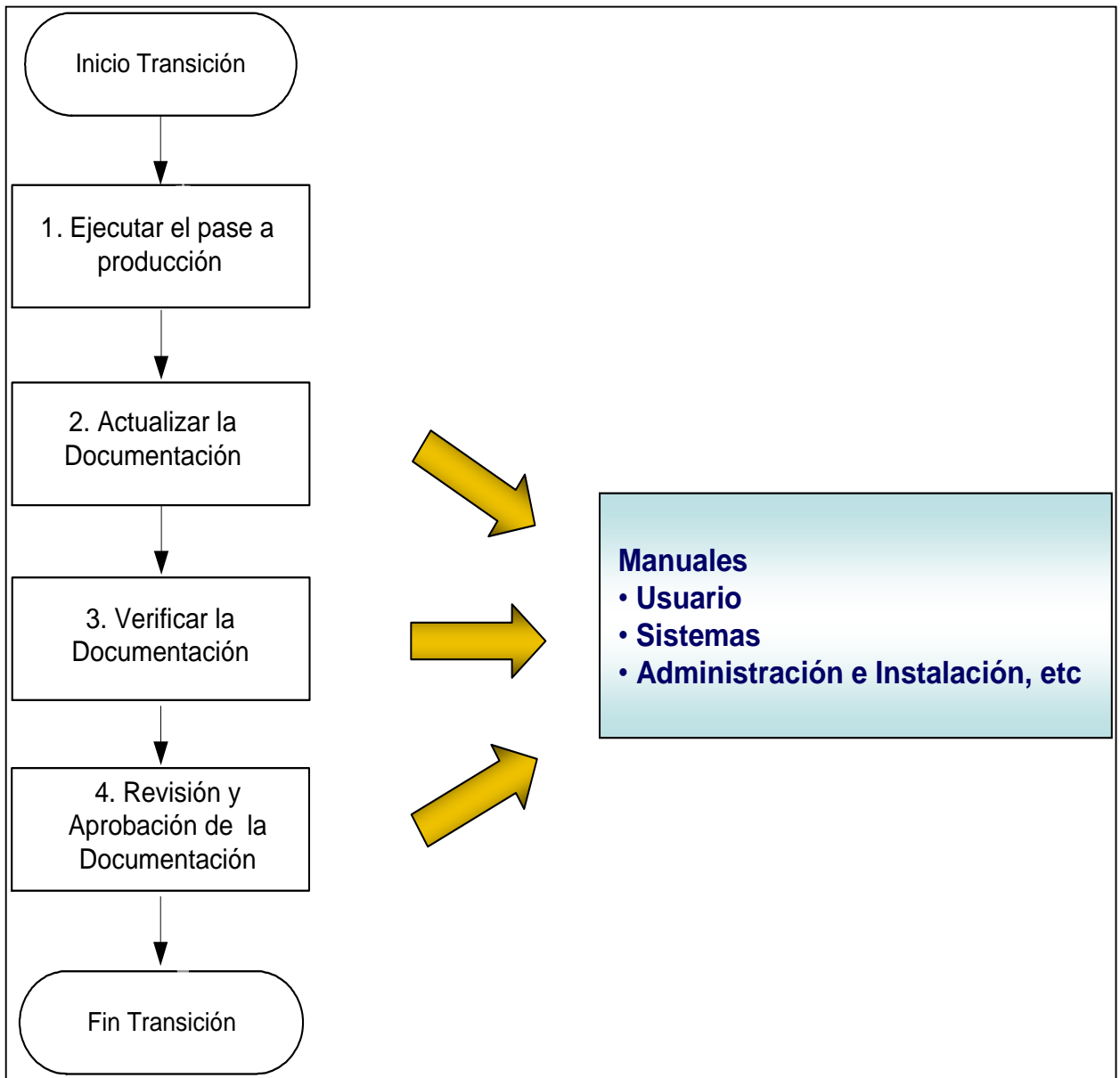
Subproceso de construcción Pruebas de aceptación



Desarrollo de los entregables

- MCPS.5 Plantilla documento de análisis actualizado
- MCPS.9 Plantilla de documentos de aceptación de pruebas funcionales
- MCPS.6 Lista de Incidencias
- MCPS.3 Cartilla mantenimiento

Subproceso de transición



Desarrollo de los entregable

- MCPS.18 Plantilla de manual de usuario
- MCPS.19 Plantilla de manual sistemas
- MCPS.20 Plantilla de manual de administración e instalaciones

Documentos de entrada y salida de entregables
Desarrollo del proceso general

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
1. Inicio preliminar	<ul style="list-style-type: none"> ▪ Cuadro de Priorización de Requerimientos (PR). 	<ul style="list-style-type: none"> ▪ MCPS.2 Lista maestra de requerimientos para mantenimiento. ▪ MCPS.1 Plantilla de Acta de Reunión. 	<ul style="list-style-type: none"> ▪ MCPS.2 Lista maestra de requerimientos para mantenimiento. ▪ MCPS.1 Plantilla de Acta de Reunión. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 	<ul style="list-style-type: none"> ▪ Analistas Programadores 	<p>Se evalúan los Requerimientos, y se llena la MCPS.2 Lista maestra de requerimientos para mantenimiento a fin de estimar los tiempos y complejidad del PR.</p> <p>En esta actividad se define si se elabora el documento de pre-análisis o no, en base a los criterios indicados en la cartilla.</p> <p>El analista de sistemas al que fue asignada la atención del PR es el responsable de convocar a la reunión de Pre-análisis. El Usuario y el Analistas de Sistemas mediante la reunión de Pre-análisis establecen el alcance detallado del requerimiento, evalúan el impacto en el aplicativo y procesos involucrados y el Analista de Sistemas estima de manera preliminar el tiempo de atención.</p> <p>El analista de sistemas elabora el Acta y lo envía el mismo día de la reunión al resto de participantes.</p>

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
2. Inicio	<ul style="list-style-type: none"> ▪ Priorización de Requerimientos . PR. ▪ MCPS.2 Lista Maestra de Requerimientos . 	<ul style="list-style-type: none"> ▪ MCPS.4 Plantilla de documento de Preanálisis. 	<ul style="list-style-type: none"> ▪ MCPS.4 Documento de Preanálisis aprobado. ▪ MCPS.2 Lista maestra de requerimientos para mantenimiento actualizada. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 	<ul style="list-style-type: none"> ▪ Analistas. ▪ Usuario. 	Se elabora el documento de Preanálisis (ver Sub-Proceso Inicio).
3. Elaboración	<ul style="list-style-type: none"> ▪ MCPS.4 Documento de Preanálisis. ▪ MCPS.2 Lista maestra de requerimientos para mantenimiento actualizada. 	<ul style="list-style-type: none"> ▪ MCPS.5 Plantilla Documento de Análisis. 	<ul style="list-style-type: none"> ▪ MCPS.5 Documento de análisis aprobado. ▪ MCPS.2 Lista maestra de requerimientos para mantenimiento actualizada. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 	<ul style="list-style-type: none"> ▪ Analistas. ▪ Usuario. ▪ Coordinadores de sistemas. 	Se elabora el documento de análisis (ver Sub-Proceso Elaboración).

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
4. Construcción	<ul style="list-style-type: none"> ▪ MCPS.5 Documento de análisis aprobado. ▪ MCPS.2 Lista maestra de requerimientos para mantenimiento actualizada. 	<ul style="list-style-type: none"> ▪ MCPS.9 Plantilla de Documentos de Aceptación de Pruebas Funcionales. ▪ MCPS.11 Plantilla de documento de Pase a QA/Producción. 	<ul style="list-style-type: none"> ▪ MCPS.9 Documento de aceptación de Pruebas Funcionales aprobado. ▪ MCPS.11 Documento de Pase a QA/Producción. 	Analista programador.	Analista de Sistemas.	Se implementa la solución y se desarrollan las pruebas funcionales y de sistemas. Se elabora el documento de Pase a QA/Producción (ver Sub-Proceso Construcción).
5. Transición	<ul style="list-style-type: none"> ▪ Solución implementada y probada. ▪ MCPS.5 Documento de análisis aprobado. ▪ MCPS.11 Documento de Pase a QA/Producción. 	<ul style="list-style-type: none"> ▪ MCPS.18 Plantilla de manual de usuario. ▪ MCPS.19 Plantilla de manual de sistemas. ▪ MCPS.20 Plantilla de manual de administración de sistemas. 	<ul style="list-style-type: none"> ▪ MCPS.17 Manual de usuario actualizado. ▪ MCPS.18 Manual de sistemas actualizado. ▪ MCPS.19 Manual de administración de sistemas actualizado. 	Documentador.	<ul style="list-style-type: none"> ▪ Analista de Sistemas. ▪ Usuario. ▪ Coordinadores de sistemas. 	<p>Llevar la solución implementada al ambiente de producción y actualizar los manuales de usuario, de sistemas y de administración de sistemas.</p> <p>Se suben los artefactos de software al repositorio oficial de componentes.</p>

Subproceso de inicio

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
1. Elaborar documento de Preanálisis (Definición del PR)	<ul style="list-style-type: none"> ▪ PR asignada ▪ MCPS.2 Plantilla de lista maestra de requerimientos para mantenimiento ▪ MCPS.1 Plantilla de Acta de Reunión 	<ul style="list-style-type: none"> ▪ MCPS.4 Plantilla de documento de Preanálisis. ▪ MCPS.1 Plantilla de Acta de Reunión. 	<ul style="list-style-type: none"> ▪ MCPS.4 Documento de Preanálisis. ▪ MCPS.2 Plantilla de lista maestra de requerimientos para mantenimiento actualizada. 	Analista de Sistemas.	<ul style="list-style-type: none"> ▪ Analista programador. ▪ Usuario. ▪ Coordinadores de sistemas. 	EL Documento de Pre-Análisis es elaborado en base a los Acuerdos de la Reunión de Preanálisis. En caso de PRs de complejidad alta, se evalúan varias alternativas de solución, sugiriéndose una, en base al resultado de una toma de decisiones estructurada.
2. Verificar Entregables	<ul style="list-style-type: none"> ▪ MCPS.4 Documento de Preanálisis. 	<ul style="list-style-type: none"> ▪ MCPS.14 Checklist Mantenimiento. 	<ul style="list-style-type: none"> ▪ MCPS.14 Checklist Mantenimiento. 	Analista de Sistemas.		El analista de sistemas registra el checklist de Mantenimiento.

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
3. Aprobar documento de Preanálisis	<ul style="list-style-type: none"> ▪ MCPS.4 Documento de Preanálisis. 		<ul style="list-style-type: none"> ▪ MCPS.4 Documento de Preanálisis aprobado. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. ▪ Usuario. ▪ Coordinadores de sistemas. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 	El analista de sistemas envía a los involucrados el documento de pre-análisis para su aprobación u observación. Si no procede se anula el PR y se comunica formalmente a los involucrados.
4. Ajustar documento de Preanálisis	<ul style="list-style-type: none"> ▪ MCPS.4 Documento de Preanálisis observado. ▪ MCPS.6 Plantilla lista incidencias. 	<ul style="list-style-type: none"> ▪ MCPS.6 Plantilla lista incidencias. ▪ MCPS.14 Checklist Mantenimiento. 	<ul style="list-style-type: none"> ▪ MCPS.4 Documento de Preanálisis ajustado. ▪ MCPS.6 Plantilla lista incidencias. ▪ MCPS.14 Checklist Mantenimiento. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 	<ul style="list-style-type: none"> ▪ Analista programador. 	El analista de sistemas realiza los ajustes necesarios y nuevamente lo envía para su revisión y aprobación (Actividad 3). El analista de sistemas actualiza la lista de observaciones a documentos y el checklist de mantenimiento.

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
5. Crear matriz de trazabilidad	<ul style="list-style-type: none"> ▪ MCPS.2 Plantilla de lista maestra de requerimientos para mantenimiento. 	<ul style="list-style-type: none"> ▪ MCPS.7 Plantilla matriz trazabilidad requerimientos a documentos mantenimiento. 	<ul style="list-style-type: none"> ▪ MCPS.7 Plantilla matriz trazabilidad requerimientos a documentos mantenimiento. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 	<ul style="list-style-type: none"> ▪ Analista programador. 	El analista de sistema llena la matriz de trazabilidad de documentos, relacionando los requerimientos de sistema con lo definido en el documento de Preanálisis.

Subproceso de elaboración

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
1. Elaborar la parte funcional documento de Análisis	<ul style="list-style-type: none"> ▪ MCPS.4 Documento de Preanálisis aprobado (si se realizó). ▪ MCPS.2 Lista Maestra de Requerimiento. ▪ MCPS.1 Plantilla de Acta de Reunión. 	<ul style="list-style-type: none"> ▪ MCPS.5 Plantilla Documento de Análisis. 	<ul style="list-style-type: none"> ▪ MCPS.5 Documento de Análisis Parte Funcional. ▪ MCPS.2 Lista maestra de requerimientos para mantenimiento actualizada. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 	<ul style="list-style-type: none"> ▪ Analista programador. ▪ Usuario. ▪ Coordinadores de sistemas. 	El analista de sistemas describe la solución planteada (el qué y el cómo) en términos entendibles para el usuario operativo.

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
2. Elaborar la parte técnica del documento de análisis	<ul style="list-style-type: none"> ▪ MCPS.4 Documento de Preanálisis aprobado (si se realizó). ▪ MCPS.2 Lista Maestra de Requerimientos. ▪ MCPS.1 Acta de Reunión de Preanálisis. 	<ul style="list-style-type: none"> ▪ MCPS.5 Plantilla Documento de Análisis. 	<ul style="list-style-type: none"> ▪ MCPS.5 Documento de Análisis Parte Técnica. ▪ MCPS.2 Lista maestra de requerimientos para mantenimiento actualizada. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 	<ul style="list-style-type: none"> ▪ Analista programador. ▪ Usuario. ▪ Coordinadores de sistemas. 	<p>El analista de sistemas detalla la solución planteada (lógica creada o modificada), utilizando notaciones UML y diagramas que se requieren (Casos de Uso, Actividades, Estado, Clases, Secuencia, Colaboración, Componentes y Despliegue).</p> <p>Elabora el documento de análisis y asigna el grado de complejidad del PR (Alto, medio, bajo), en base a la Lista Maestra de Requerimientos.</p>
3. Verificar Entregables	<ul style="list-style-type: none"> ▪ MCPS.5 Documento de Análisis. 	<ul style="list-style-type: none"> ▪ MCPS.14 Checklist Mantenimiento. 	<ul style="list-style-type: none"> ▪ MCPS.14 Checklist Mantenimiento. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 		<p>El analista de sistemas registra el checklist de Mantenimiento.</p>

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
4. Revisar y aprobar el documento de Análisis	<ul style="list-style-type: none"> ▪ MCPS.5 Documento de análisis. 	<ul style="list-style-type: none"> ▪ MCPS.5 Plantilla Documento de Análisis. 	<ul style="list-style-type: none"> ▪ MCPS.5 Documento de análisis aprobado. 	<ul style="list-style-type: none"> ▪ Analista programador. ▪ Usuario ▪ Coordinadores de sistemas. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 	<p>El Usuario revisa el documento de análisis. El resultado de la revisión es comunicado analista de sistemas.</p> <p>Si existen observaciones, éstas son comunicadas al analista de sistemas, con copia a los usuarios implicados, para que reformule o ajuste el documento (actividad 6).</p> <p>Si no hay observación continua la fase de construcción.</p>

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
5. Ajustar documento de Análisis	<ul style="list-style-type: none"> ▪ MCPS.5 Documento de análisis observado. ▪ MCPS.14 Checklist de mantenimiento. ▪ MCPS.6 Plantilla lista incidencias. 	<ul style="list-style-type: none"> ▪ MCPS.6 Plantilla lista incidencias. ▪ MCPS.14 Checklist Mantenimiento 	<ul style="list-style-type: none"> ▪ MCPS.5 Documento de análisis ajustado. ▪ MCPS.6 Plantilla lista incidencias. ▪ MCPS.14 Checklist Mantenimiento. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 	<ul style="list-style-type: none"> ▪ Analista programador. 	<p>El analista de sistemas realiza los ajustes necesarios y nuevamente lo envía para su revisión y aprobación (Actividad 5).</p> <p>El analista de sistemas actualiza la lista de observaciones y el checklist de mantenimiento.</p>
6. Crear matriz de trazabilidad	<ul style="list-style-type: none"> ▪ MCPS.2 Plantilla de lista maestra de requerimientos para mantenimiento. 	<ul style="list-style-type: none"> ▪ MCPS.7 Plantilla matriz trazabilidad requerimientos a documentos mantenimiento. 	<ul style="list-style-type: none"> ▪ MCPS.2 Plantilla matriz trazabilidad requerimientos a documentos mantenimiento. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 	<ul style="list-style-type: none"> ▪ Analista programador. 	<p>El analista de sistema llena la matriz de trazabilidad de documentos, relacionando los requerimientos de sistema con lo definido en el documento de análisis.</p>

Subproceso de construcción

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
7. Implementar modificación	<ul style="list-style-type: none"> ▪ MCPS.5 Documento de análisis. ▪ MCPS.2 Lista de maestra de requerimientos. 	<ul style="list-style-type: none"> ▪ MCPS.16 Plantilla Informe de pruebas estándares. 	<ul style="list-style-type: none"> ▪ Solución implementada. ▪ MCPS.5 Documento de análisis. ▪ MCPS.2 Lista de maestra de requerimientos para mantenimiento. ▪ MCPS.16 Plantilla Informe de Pruebas. 	Analista programador.	Analista de sistemas.	<p>El analista programador codifica según las especificaciones indicadas en el documento de análisis y bajo los estándares de desarrollo de la Institución.</p> <ul style="list-style-type: none"> ▪ Luego de la codificación, el analista programador realiza las pruebas unitarias, quedando evidencia de las mismas en el documento. “MCPS.16 Plantilla Informe de Pruebas”.
8. Elaborar Casos de Prueba	<ul style="list-style-type: none"> ▪ MCPS.5 Documento de análisis. ▪ MCPS.2 Lista de maestra de requerimientos. 	<ul style="list-style-type: none"> ▪ MCPS.9 Plantilla de Documentos de Aceptación de Pruebas Funcionales. 	<ul style="list-style-type: none"> ▪ MCPS.9 Documentos de Aceptación de Pruebas Funcionales. 	Analista de Sistemas.	Analista programador.	El Analista de Sistemas elabora (define) los casos de prueba.

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
9. Verificación de Entrega	▪ MCPS.9 Documento de Aceptación de Pruebas Funcionales.	▪ MCPS.14 Plantilla de Checklist de Mantenimiento.	▪ Checklist de Mantenimiento.	▪ Analista de Sistemas.	▪ Analista programador.	El Analista de Sistemas registra el Checklist de mantenimiento.
10. Revisión y aprobación de casos de prueba	▪ MCPS.F9 Documento de Aceptación de Pruebas Funcionales.		▪ MCPS.6 Plantilla lista incidencias.	▪ Contraparte.	▪ Analista de Sistemas.	El analista coordina con la contraparte la aprobación de los casos de prueba.
11. Pruebas Internas	▪ MCPS.9 Documentos de Aceptación de Pruebas Funcionales.	▪ MCPS.6 Plantilla lista incidencias. ▪ Plantilla de Documento de pruebas de sistemas.	▪ MCPS.9 Documentos de Aceptación de Pruebas Funcionales	▪ Analista de Sistemas.	▪ Analista programador.	El analista, apoyado por el analista programador realiza las pruebas internas necesarias a fin de verificar el buen funcionamiento de la solución desarrollada.

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
12. Pruebas de Calidad	<ul style="list-style-type: none"> ▪ MCPS.5 Documento de Análisis. ▪ MCPS.9 Documentos de Aceptación de Pruebas Funcionales. 		<ul style="list-style-type: none"> ▪ MCPS.11 Plantilla de Revisión QA-Mantenimiento para la prueba de QA Técnica ▪ MCPS.6 Lista de incidencias 	<ul style="list-style-type: none"> ▪ Analista de Calidad. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. ▪ Analista Programador. 	<p>El analista de Calidad, realiza las pruebas funcionales y técnicas en el ambiente de pruebas del servicio.</p> <p>Durante las pruebas de Calidad se revisa la aprobación de los documentos comprometidos en la atención del PR y se validan que la solución este acorde con el Análisis Funcional del documento de análisis.</p>

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
13. Pruebas de Aceptación	<ul style="list-style-type: none"> ▪ MCPS.9 Documento de aceptación de Pruebas Funcionales. 		<ul style="list-style-type: none"> ▪ MCPS.9 Documento de aceptación de Pruebas Funcionales actualizado. 	<ul style="list-style-type: none"> ▪ Analista programador. ▪ Usuario. ▪ Coordinadores de sistemas. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. ▪ Analista Programador. 	<p>El analista de sistemas convoca al Usuario (los que participaron en la etapa de análisis de requerimiento) para la realización de las pruebas funcionales.</p> <p>Durante las pruebas funcionales validan que la solución este acorde con el Análisis Funcional del documento de análisis, asimismo verifica que la solución satisfaga su necesidad operativa, realizando los casos de pruebas que sugirió el analista de sistemas.</p> <p>Al término de las pruebas se actualiza el MCPS.9 documento "Pruebas Funcionales".</p>

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
14. Levantar Observaciones	▪ MCPS.6 Plantilla lista incidencias.	▪ MCPS.6 Plantilla lista incidencias.	▪ MCPS.6 Plantilla lista incidencias actualizada.	▪ Analista programador.	▪ Analista de Sistemas.	Se actualiza la lista de Incidencias.

Subproceso de pruebas internas

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
1. Elaborar Data de Prueba	<ul style="list-style-type: none"> ▪ MCPS.9 Documentos de Aceptación de Pruebas Funcionales. ▪ Casuística. 		<ul style="list-style-type: none"> ▪ MCPS.9 Documentos de Aceptación de Pruebas Funcionales. 	<ul style="list-style-type: none"> ▪ Analista programador. 		El analista programador elabora la data de prueba en base a lo definido en los casos de prueba y a la casuística entregada (opcionalmente) por el usuario.
2. Integración del Producto	<ul style="list-style-type: none"> ▪ MCPS.11 Documento de Pase a QA/Producción. • Solución implementada. 	<ul style="list-style-type: none"> ▪ MCPS.6 Plantilla lista incidencias. 	<ul style="list-style-type: none"> ▪ MCPS.6 Plantilla lista incidencias. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 	<ul style="list-style-type: none"> ▪ Analista programador. ▪ DBA. 	El analista de sistemas será el responsable de la integración del producto final debiendo enviar los objetos modificados y/o creados al Ambiente de Desarrollo del servicio para el inicio de las pruebas. El DBA ejecuta los archivos del pase en el ambiente de pruebas según lo indicado en el documento de Pase a QA/Producción. Al término remite el Archivo de Resultados (log).

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
3. Pruebas Funcionales	<ul style="list-style-type: none"> • MCPS.5 Documento de análisis. • MCPS.9 Documentos de Aceptación de Pruebas Funcionales. 	<ul style="list-style-type: none"> ▪ MCPS.6 Plantilla lista incidencias. 	<ul style="list-style-type: none"> ▪ MCPS.6 Plantilla lista incidencias. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 	<ul style="list-style-type: none"> ▪ Analista programador. 	El analista programador realiza las pruebas funcionales necesarias a fin de verificar el buen funcionamiento de la solución desarrollada.
4. Pruebas Técnicas	<ul style="list-style-type: none"> • MCPS.5 Documento de análisis. 	<ul style="list-style-type: none"> ▪ Checklist de DBA. ▪ Estándares. 	<ul style="list-style-type: none"> ▪ MCPS.6 Plantilla lista incidencias. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 	<ul style="list-style-type: none"> ▪ Analista programador. 	El analista programador realiza las pruebas técnicas a fin de verificar el cumplimiento de los estándares definidos por la institución.
5. Llenar Checklist de Analista y Programador		<ul style="list-style-type: none"> ▪ MCPS.15 Plantilla Checklist de analista / programador. 	<ul style="list-style-type: none"> ▪ MCPS.15 Plantilla Checklist de analista / programador. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. ▪ Analista Programador. 		<p>MCPS.14 El analista de Sistemas registra el checklist de Analista.</p> <p>El analista programador registra el checklist de programador.</p>

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
6. Levantar Observaciones	<ul style="list-style-type: none"> • Informe de Pruebas revisado. • Observaciones. • MCPS.6 Plantilla lista incidencias. 	<ul style="list-style-type: none"> • MCPS.6 Plantilla lista incidencias. 	<ul style="list-style-type: none"> • MCPS.6 Plantilla lista incidencias. 	<ul style="list-style-type: none"> ▪ Analista programador. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 	<p>El analista programador levanta las observaciones, asimismo actualiza la lista de observaciones.</p> <p>Levantadas las observaciones se realiza nuevamente la actividad 1.</p>
7. Actualizar matriz de trazabilidad	<ul style="list-style-type: none"> • MCPS.7 Matriz de trazabilidad. 	<ul style="list-style-type: none"> • 	<ul style="list-style-type: none"> • MCPS.7 Matriz de trazabilidad actualizada. 	<ul style="list-style-type: none"> ▪ Analista de sistemas. 	<ul style="list-style-type: none"> ▪ Analista programador. 	<p>Se actualiza las matrices de trazabilidad de requerimientos y de componentes.</p>

Subproceso de Pruebas de Calidad

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
<p>1. Revisar Checklist de Analista y Programador</p>	<ul style="list-style-type: none"> ▪ MCPS.15 Plantilla Checklist de analista / programador 	<ul style="list-style-type: none"> ▪ MCPS.8 Plantilla de matriz de trazabilidad requerimientos a objetos. ▪ MCPS.12 Documento de Casos de Pruebas. 	<ul style="list-style-type: none"> ▪ MCPS.15 Plantilla Checklist de analista / programador 	<ul style="list-style-type: none"> ▪ Analista de calidad. 	<ul style="list-style-type: none"> ▪ Analista de Base de Datos. ▪ Analista de Sistemas. ▪ Analista programador. 	<p>El analista de calidad revisa los checklist de analista y programador donde se describe el PR desarrollado.</p> <p>En esta actividad, el Analista de base de datos valida que los objetos indicados en el Documento de Pase a QA coincidan con los definido en la matriz de objetos para los requerimientos de sistema a implementar según el documento de casos de pruebas.</p> <p>Adicionalmente, valida que los requerimientos de sistema a pasar se hallan probado a través del documento de casos de pruebas.</p>

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
2. Preparar Ambiente de Pruebas	<ul style="list-style-type: none"> ▪ MCPS.11 Documento de Pase a QA/Producción. ▪ Solución implementada. 	<ul style="list-style-type: none"> ▪ Plantilla de checklist del DBA. ▪ Estándares. 	<ul style="list-style-type: none"> ▪ Checklist de DBA lleno. 	Analista de Base de Datos.	<ul style="list-style-type: none"> ▪ Analista de Sistemas. ▪ Analista programador. 	El DBA ejecuta los archivos del pase en el ambiente de pruebas según lo indicado en el documento de Pase a QA/Producción. Al término remite el Archivo de Resultados (log).
3. Revisión Funcional	<ul style="list-style-type: none"> ▪ MCPS.5 Documento de Análisis. ▪ MCPS.9 Documento de aceptación de pruebas funcionales. 	<ul style="list-style-type: none"> ▪ MCPS.6 Plantilla lista incidencias. 	<ul style="list-style-type: none"> ▪ MCPS.6 Plantilla lista incidencias. ▪ Aprobación (correo). 	Analista de calidad.	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 	El analista de calidad realiza las pruebas funcionales en base a los casos de prueba elaborados por el analista programador.
4. Revisión Técnica	<ul style="list-style-type: none"> ▪ MCPS.11 Documento de Pase a QA/Producción. ▪ Solución implementada. 	<ul style="list-style-type: none"> ▪ MCPS.6 Plantilla lista incidencias. ▪ MCPS.10 Plantilla de Pruebas de Sistemas. ▪ Estándares. 	<ul style="list-style-type: none"> ▪ MCPS.6 Plantilla lista incidencias. 	Analista de calidad.	<ul style="list-style-type: none"> ▪ Analista Programador. 	La revisión técnica, consiste en verificar el cumplimiento de estándares impuestos por la institución.

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
5. Solicitar Pase a QA	<ul style="list-style-type: none"> ▪ MCPS.11 Documento de Pase a QA/Producción.		<ul style="list-style-type: none"> ▪ MCPS.11 Documento de Pase a QA/Producción.	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 		El analista de Sistemas solicita la ejecución del pase a QA, enviando el documento de pase a QA/Producción.

Subproceso de pruebas de aceptación

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
1. Ejecutar pase a QA/Producción	<ul style="list-style-type: none"> ▪ MCPS.11 Documento de Pase a QA/Producción. ▪ Solución implementada. 		<ul style="list-style-type: none"> ▪ Archivo de resultados. ▪ Log. 	<ul style="list-style-type: none"> ▪ DBA. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 	DBA ejecuta los archivos del pase en el ambiente de pruebas según lo indicado en el documento de Pase a QA/Producción. Al término remite el Archivo de Resultados (log).

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
2. Verificar ejecución de pase a QA/Producción	<ul style="list-style-type: none"> ▪ Archivo de resultados. ▪ Log. 	<ul style="list-style-type: none"> ▪ MCPS.11 Documento de Pase a QA/Producción. 	<ul style="list-style-type: none"> ▪ Certificación de verificación de ambiente (correo). 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 		<p>El analista de sistemas revisa el archivo de resultados y de estar conforme convocará a la realización de las pruebas funcionales y de sistemas.</p> <p>De no estar conforme con los resultados y de ser un problema en la solución desarrollada se volverá a la actividad 1, si el problema es debido a la preparación de los archivos del pase entonces se volverá a la actividad 6. De lo contrario se realizará el pase a QA/Producción nuevamente (actividad 7).</p>

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
3. Pruebas Funcionales	<ul style="list-style-type: none"> ▪ MCPS.9 Documento de aceptación de Pruebas Funcionales. 	<ul style="list-style-type: none"> ▪ MCPS.6 Plantilla lista incidencias. 	<ul style="list-style-type: none"> ▪ MCPS.6 Documento de aceptación de Pruebas Funcionales actualizado. ▪ MCPS.6 Plantilla lista incidencias actualizadas. 	<ul style="list-style-type: none"> ▪ Usuarios. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 	<p>El analista de sistemas convoca al DBA y/o Usuario (los que participaron en la etapa de análisis de requerimiento) para la realización de las pruebas funcionales.</p> <p>Durante las pruebas funcionales validan que la solución este acorde con el Análisis Funcional del documento de análisis, asimismo verifica que la solución satisfaga su necesidad operativa, realizando los casos de pruebas que sugirió el analista de sistemas.</p> <p>Al término de las pruebas se actualiza el documento "Pruebas Funcionales".</p>

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
4. Pruebas de sistemas	<ul style="list-style-type: none"> ▪ MCPS.10 Documento de aceptación de Pruebas de Sistema. 		<ul style="list-style-type: none"> ▪ MCPS.10 Documento de aceptación de Pruebas de Sistema actualizado. ▪ MCPS.6 Plantilla lista incidencias actualizadas. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 	<p>El analista de sistemas convoca a la realización de las pruebas de sistema a los usuarios (si participaron en la etapa de Análisis).</p> <p>El Analista de Sistemas realiza las pruebas técnicas de acuerdo al Documento de aceptación de pruebas de sistema, así mismo verifica que cumplan con los estándares de desarrollo de Institución.</p>

Subproceso de transición

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
15. Ejecutar el pase a producción	<ul style="list-style-type: none"> ▪ Solución implementada y probada. 		<ul style="list-style-type: none"> ▪ Solución implementada y probada. ▪ Autorización del pase a producción. 	<ul style="list-style-type: none"> ▪ Usuario. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 	<p>El personal del Usuario realiza un control de calidad de acuerdo al "Procedimiento de Control de Calidad".</p> <p>De estar conforme da la orden de la ejecución del pase a producción. De lo contrario observa el pase y solicita al analista de sistemas su revisión, volviendo a la actividad 4 del subproceso de Construcción.</p>

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
16. Actualizar la documentación	<ul style="list-style-type: none"> ▪ Documentación del PR. ▪ MCPS.18Manual de usuario. ▪ MCPS.19Manual de sistemas. ▪ MCPS.20Manual de administración de sistemas. ▪ MCPS.17Documento de especificación de ambientes. 	<ul style="list-style-type: none"> ▪ MCPS.18Plantilla de manual de usuario. ▪ MCPS.19Plantilla de manual de sistemas. ▪ MCPS.20Plantilla de manual de administración de sistemas. ▪ MCPS.17Documento de especificación de ambientes. 	<ul style="list-style-type: none"> ▪ MCPS.18Manual de usuario actualizado. ▪ MCPS.19Manual de sistemas actualizado. ▪ MCPS.20Manual de administración de sistemas actualizado. ▪ MCPS.17Documento de especificación de ambientes. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 	<ul style="list-style-type: none"> ▪ DOC. 	<p>El analista de sistemas recaba la documentación relativa al PR y se la entrega al usuario, el DOC actualiza los manuales según los cambios producidos por la tención de PR, se informa de las actualizaciones al analista de sistemas.</p> <p>En caso existir el documento de especificación de ambientes, se actualiza.</p>

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
17. Verificar la documentación	<ul style="list-style-type: none"> ▪ Doc. del PR. ▪ MCPS.18Manual de usuario. ▪ MCPS.19Manual de sistemas. ▪ MCPS.20Manual de administración de sistemas. ▪ MCPS.17Documento de especificación de ambientes. 	<ul style="list-style-type: none"> ▪ Documento de Análisis. 	<ul style="list-style-type: none"> ▪ MCPS.18Manual de usuario actualizado. ▪ MCPS.19Manual de sistemas actualizado. ▪ MCPS.20Manual de administración de sistemas actualizado. ▪ MCPS.17Documento de especificación de ambientes. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 		<p>El analista de sistemas verifica que las modificaciones a los manuales, contemplen las especificaciones definidas en el documento de análisis, se informa de las actualizaciones al usuario en caso amerite su aprobación.</p> <p>En caso de actualización del documento de especificación de ambientes, se verifica.</p>

Descripción de la Tarea	Entrada	Documentos de Soporte	Salida	Responsable	Rol involucrado	Explicación adicional
18. Revisión y Aprobación de la Documentación	<ul style="list-style-type: none"> ▪ Documentación del PR. ▪ MCPS.18 Manual de usuario. ▪ MCPS.19 Manual de sistemas. ▪ MCPS.20 Manual de administración de sistemas. ▪ MCPS.17 Documento de especificación de ambientes. 	<ul style="list-style-type: none"> ▪ MCPS.18Manual de usuario. ▪ MCPS.19Manual de sistemas. ▪ MCPS.20Manual de administración de sistemas. ▪ MCPS.17Documen to de especificación de ambientes. 	<ul style="list-style-type: none"> ▪ MCPS.18Manual de usuario actualizado. ▪ MCPS.19Manual de sistemas actualizado. ▪ MCPS.20Manual de administración de sistemas actualizado. ▪ MCPS.17Docume nto de especificación de ambientes. 	<ul style="list-style-type: none"> ▪ UA. ▪ CS: Sistemas. 	<ul style="list-style-type: none"> ▪ Analista de Sistemas. 	<p>Usuario revisan el manual de usuario, el CS: Sistemas revisa el manual de sistemas y manual de administración de sistemas, de estar actualizados todos los manuales comunican su conformidad al analista de sistemas, de lo contrario comunican sus observaciones.</p>

CAPÍTULO IV

APLICACIÓN DE LA METODOLOGÍA

4.1 Instituciones públicas

Las instituciones públicas del estado peruano son controladas por el Fondo Nacional de Financiamiento de la Actividad Empresarial del Estado – FONAFE, así como de la Presidencia del Consejo de Ministros – PCM, ambas instituciones en la actualidad, para todo tipo de tipo de desarrollo e implementación de proyectos, toman como referencia las principales instituciones que lo conforman, una de las principales es el Banco de la Nación, entidad financiera que por la complejidad y diversidad en el tipo de aplicaciones de software que posee, es la institución que hemos elegido para poder implementar nuestra área de estudio, es necesario indicar que mucho de las aplicaciones del Banco son parte importante con la interacción que tiene con RENIEC, SUNAT, MEF, MTC, BCR, SAT, para el desarrollo de atención a los clientes.

El Banco de la Nación en la actualidad está enmarcada en un proceso de modernización, para lo cual hará suyo este estudio para su implementación, y luego lo pondrá a disposición de las entidades que lo requieran.

4.2 Identificación y análisis de los factores críticos

- ✓ Presupuesto del proyecto adecuado, que le dé soporte al logro de los objetivos planteados de manera oportuna y adecuada.
- ✓ Estabilidad en la administración del Proyecto. Buen manejo y control del plan de trabajo.
- ✓ Renovar procedimientos formales, ágiles y eficientes, que permitan soportar una operación de calidad hacia el cliente externo.
- ✓ Renovar las herramientas tecnológicas necesarias, así como la disposición presupuestaria correspondiente.
- ✓ Elaborar una estructura orgánica del equipo de trabajo del proyecto.
- ✓ Mejorar conocimiento, en todo el equipo del proyecto. Debe tener habilidades y experiencia en las diversas etapas del proyecto.

4.3 Análisis interno

El 27 de enero de 1966, el Congreso de la República aprobó la Ley 16000 por la cual creaba el Banco de la Nación días después el Poder Ejecutivo, bajo la firma del Presidente de la República, Fernando Belaúnde Terry la pone en vigencia, culminando así un largo proceso cuyos antecedentes históricos datan del siglo XIX, pero que recién a partir de 1914, surge verdaderamente la preocupación de crear un Banco que centralice las actividades operativas, económicas y financieras.

El Banco de la Nación encuentra sus antecedentes inmediatos en el año 1905, en el que se crea la Caja de Depósitos y Consignaciones. Esta institución amplió sus actividades en 1927 cuando se le encargó a través la administración del Estanco del Tabaco y Opio, así como la recaudación de las rentas del país, derechos e impuestos del alcohol, defensa nacional y otros.

Finalmente, en diciembre del mismo año se le encarga la recaudación de la totalidad de las rentas de toda la República.

En 1963 se estatiza la Caja de Depósitos y Consignaciones, declarándose de necesidad y utilidad pública. Se recupera para el Estado las funciones de recaudación de las rentas fiscales y la custodia de los depósitos administrativos y judiciales. Tal estatización se realizó cuando la Caja contaba entre sus accionistas con diez bancos: Crédito, Popular, Internacional, Wiese, Comercial, Continental, Gibson, De Lima, Unión y Progreso.

En 1994, durante el gobierno de Alberto Fujimori Fujimori, se modificaron las funciones, las mismas que a la fecha se encuentran vigentes:

- ✓ Participar en las operaciones de comercio exterior del Estado.
- ✓ Otorgar facilidades financieras al Gobierno Central, y a los Gobiernos Regionales y Locales, en los casos en que éstos no sean atendidos por el Sistema Financiero Nacional.
- ✓ Las facilidades financieras que otorga el Banco no están sujetas a los límites que establece la Ley General de Instituciones Bancarias, Financieras y de Seguros.
- ✓ Brindar Servicios de Corresponsalía.
- ✓ Brindar Servicios de Cuentas Corrientes a las Entidades del Sector Público Nacional y a Proveedores del Estado.
- ✓ Otorgar Créditos al Sector Público.
- ✓ Recibir depósitos de ahorros en lugares donde la banca privada no tiene oficinas.

El Banco de la Nación tiene como misión brindar servicios financieros eficientes al sector público y clientes en general, preocupándose en satisfacer las necesidades de interconexión financiera en distritos donde la banca privada no presta sus servicios, coadyuvando a profundizar el sistema financiero peruano y participando activamente en el desarrollo nacional.

Visión y misión

La visión es un concepto que pretende dar orientación a la organización, indicando donde la Alta Dirección considera que debe estar la institución dentro de 5 o 10 años. Esta visión es la que tiene que responder a preguntas como por ejemplo: ¿A dónde queremos ir? ¿En qué tipo de negocios nos encontramos, qué necesidades de nuestros clientes queremos satisfacer, qué capacidades debemos desarrollar? Adicionalmente, toda visión de negocios debe apoyarse en una ventaja competitiva. Esto aplica también para el caso del Banco de la Nación, dado que cumple un rol importante respecto a la “bancarización” nacional, un aspecto que podría promover también la banca comercial regular.

La **Visión del Banco de la Nación** cumple con las tres principales características:

1. Define en qué negocio está la institución y lo ubica como un Banco moderno y eficiente que apoya sustancialmente a los Organismos del Estado.
2. Define la estrategia que debe perseguir la institución a largo plazo, mencionando la calidad en la atención a sus clientes, la proactividad de su personal y los valores éticos del mismo.
3. Se comunica de una forma clara para el entendimiento de todo el personal.

Visión del Banco de la Nación

“Ser el banco reconocido por la excelencia en la calidad de sus servicios, la integridad de su gente y por su contribución al desarrollo nacional”

Algunos comentarios específicos respecto al contenido de la visión del Banco de la Nación se mencionan a continuación:

- ✓ El hecho que el Banco deba ser moderno y eficiente, con capacidad innovadora, es un aspecto que debe tomarse como un punto de partida, es casi una “obligación” de la institución contar con estas características. Por tanto, consideramos que no le brinda mucho valor

agregado a la visión como el “sueño a largo plazo”. Es más, estas características no se logran a largo plazo, sino aspectos que se debe lograr en el corto plazo.

- ✓ El mismo comentario podemos ampliarlo también a contar con personal proactivo y de altos valores éticos. Es cierto que ello implica en el caso del Banco de la Nación un cambio organizacional, cambio que no es fácil de lograr; sin embargo, no debe verse esto como algo a obtener en el largo plazo.

Misión del Banco de la Nación

"Brindamos soluciones financieras con calidad de atención, agregando valor, contribuyendo con la descentralización, ampliando nuestra cobertura de servicios, promoviendo la bancarización y la inclusión."

Objetivos generales:

- Satisfacer la demanda de nuestros clientes, brindándoles servicios de calidad.
- Apoyar al Estado en el proceso de descentralización y desarrollo del país, abordando preferentemente las necesidades de interconexión de las comunidades sin acceso a servicios bancarios.
- Fortalecer en la cultura organizacional de la institución la creatividad, el cambio de actitud y valores, generando una organización con mayor valor.

Estrategias:

- Ampliar la red de oficinas y la presencia del Banco de la Nación a nivel Nacional.
- Implantar dos turnos en las agencias de mayor carga operativa con la finalidad de mejorar la imagen y calidad de servicio.
- Innovar canales de distribución de servicios, acercándolos al cliente y ampliando la capacidad de atención, mejorando la calidad y costo de los servicios que presta el Banco de la Nación.

- Implementar un sistema de comunicaciones amplio y confiable a través de una conexión directa que permita integrarnos con los sistemas de los clientes.
- Mantener un soporte informático integrado y oportuno a las necesidades del cliente, basado en la aplicación de tecnologías de información de vanguardia.
- Aplicar mecanismos que conlleven a perfeccionar integralmente al personal, incrementar la productividad, mejorar la calidad de atención, logrando una organización competitiva, proactiva, moderna y eficiente.

Sin embargo, la visión de la Gerencia General es muy clara respecto a los objetivos institucionales, y estos se enumeran a continuación, siendo aquellos dentro de los cuales se enmarca la operación de la institución:

- Reducción de colas y mejoramiento de servicios
- Aumentar la rentabilidad y reducir los costos
- Reducción de todo riesgo operacional, financiero y de seguridad
- Mejora de los procesos operativos del banco
- Adquisición de sede principal

Para el Diseño de una Metodología de Certificación de Productos Software, la mayor parte del éxito está relacionada con los elementos que conforman el Área de Informática, teniendo en cuenta que es ella la que va a administrar el proceso y definir los lineamientos tecnológicos que se seguirán.

Es importante el compromiso de todos los especialistas que participaron en el desarrollo de este proyecto, ya que se pudo plasmar los conocimientos usando herramientas de TI.

La organización interna ha dado respuesta a planes de acción, preconcebidos de una manera lógica y coherente para el logro de

objetivos, considerando las capacidades internas del Banco de la Nación, así como también las capacidades que pueden tener las empresas especializadas a contratar para el desarrollo de proyectos específicos.

4.4 Análisis del contexto

Actualmente en Banco de la Nación es la única oferta bancaria que llega a todo el Perú. Por lo cual la operatividad de sus servicios no puede paralizar las 24 horas, la experiencia de los especialistas han logrado muchos avances y han podido enfrentar muchas fallas en forma concurrente.

De acuerdo a las estadísticas que lleva el Departamento de Informática se ha podido detectar que un tipo de falla se está repitiendo por lo menos una vez al mes, si bien es cierto la solución se ha dado en tiempo record, no se cuenta con los procedimientos al alcance de todos los especialistas que conforman el proceso integral, motivo por el cual estas fallas ponen en riesgo las actividades del Banco generando tensión, ya que si el Banco no opera se perderán muchos valores intangibles.

4.5 Discusión de alternativas.

Actualmente la calidad desempeña un interesante papel diferenciador en la industria del software. Y este punto de vista no está solo relacionado con mercados que se abren en el exterior, sino también con necesidades internas de nuestra industria, en cuanto a optimización de procesos.

La realidad nos muestra a diario proyectos de software que fracasan y ya no sorprende que los motivos de estos fracasos sean comunes a la mayoría de estos proyectos: desvíos provocados por la escasa visibilidad de los mismos, los tiempos y costos impredecibles, alto grado de dependencia de personas claves, falta de aplicación de prácticas básicas de gestión de proyectos, entre otros. En general, esto sucede en

organizaciones que cuentan con procesos informales que hacen que el desarrollo sea poco predecible y repetible.

La premisa que indica: “La calidad de un producto está determinada por la calidad del proceso que se utiliza para desarrollarlo y mantenerlo”, es hoy la que marca la diferencia.

Las empresas orientadas a la calidad en sus productos, trabajan fuertemente en la mejora de sus prácticas, definiendo e implementando procesos disciplinados que facilitan el cumplimiento de los objetivos del proyecto, estableciendo un marco de trabajo.

La definición misma de “modelo” indica que es una “idealización de la realidad utilizada para plantear un problema”. A partir de allí, nunca podría ser bien implementado si el mismo no es interpretado y adaptado, inteligentemente, a las necesidades de la organización. En particular, CMMI, plantea prácticas para la organización y para los proyectos pero no especifica cómo implementarlas, por lo que brinda a los profesionales de la mejora la posibilidad de aplicar su inteligencia y capacidad para definir procesos acordes con el negocio, que generen el suficiente valor agregado como para lograr los objetivos de la organización y que las personas los cumplan.

Por otra parte, existen beneficios concretos que pueden mostrar las empresas que han implementado el modelo:

- ✓ La improvisación queda a un lado. El proceso de desarrollo y mantenimiento del software está definido e implementado, por lo que se actúa inteligente y proactivamente. Lo mismo sucede con el Gerenciamiento del Proyecto. Esta “inteligencia” se logra con las “personas”, quienes, apoyándose en los “procesos” aportan su “criterio” y “creatividad” para el logro de los objetivos.
- ✓ Mejora el “conocimiento sobre la organización”, los procesos se retroalimentan y se nutren con las experiencias de los proyectos implementados.

Además de estos beneficios que claramente responden a algunos de los cuestionamientos planteados, se suman los siguientes:

- ✓ Mejora en el cumplimiento de los plazos establecidos y compromisos asumidos.
- ✓ Estimaciones de costos y tiempos más certeras por haber sido realizadas sobre la base de experiencias reales y cuantificadas con métodos definidos.
- ✓ Aumento en la satisfacción del cliente por el soporte dado al proyecto con los mecanismos de aseguramiento de la calidad.
- ✓ Los roles y responsabilidades de grupos y miembros del proyecto están claramente definidos, permitiendo un seguimiento y control del proyecto que asegura el logro de los objetivos.
- ✓ Se implementa la reusabilidad en un sentido amplio, pues alcanza no solo al código, sino a toda pieza involucrada en un proyecto de software.

Por otra parte, para implementar el modelo, es imprescindible que la Dirección de la empresa esté convencida de los beneficios que se obtendrán y esté dispuesto a priorizar el Proyecto de Mejora como si fuese un proyecto del negocio, garantizando un camino hacia la “Mejora Continua”. Si esto no ocurre, difícilmente el modelo pueda ser implementado, porque requiere de un importante esfuerzo inicial de los recursos, inversión, y un cambio cultural fuerte en la gente. Por estos motivos, los resultados no se ven inmediatamente sino con el tiempo, y si la Dirección no está dispuesta a dar este tiempo y el soporte necesario, seguramente el proyecto sea abandonado en el camino.

¿Qué solución buscamos?

- ✓ Que sea aplicable a cualquier entidad pública.
- ✓ Que sirva:
 - Como modelo de referencia.
 - Para la mejora del proceso de desarrollo.
 - Para la mejora de la calidad de los productos desarrollados.
- ✓ Que no sea costoso de aplicar (fácil de entender y de aplicar).

- ✓ Que sea base o complemento de otros modelos, como por ejemplo CMMI, ISO 12207, ISO 15504.

Para el diseño de una Metodología de Certificación de Productos Software, Se han aplicado los siguientes criterios para la elaboración de este modelo:

- ✓ La estructura de procesos resultante debe ser acorde con la estructura generalmente empleada por las organizaciones de la industria del software (alta dirección, gestión y operación)
- ✓ La alta dirección tiene un papel importante a través de la planificación estratégica. Debe actuar como promotor del buen funcionamiento de la organización a través de su implicación en la revisión y mejora continua del modelo.
- ✓ El modelo considera a la gestión como proveedora de recursos, procesos y proyectos; así como es responsable de la vigilancia del cumplimiento de los objetivos estratégicos de la organización.
- ✓ El modelo considera a la operación como ejecutora de los proyectos de desarrollo y mantenimiento de software.
- ✓ El modelo integra con claridad y consistencia los elementos indispensables para la definición de los procesos y las relaciones entre ellos.
- ✓ El modelo integra los elementos para realizar la administración de proyectos desde un solo proceso.
- ✓ El modelo integra los elementos para realizar la ingeniería de productos de software en un único marco que incluya los procesos precisos de soporte (verificación, validación, documentación y control de la documentación).
- ✓ Moprosoft se basa en los modelos de procesos ISO 9001:2000, en las áreas de procesos de los niveles 2 y 3 de CMM-SW: CMM-SW v.1.2., en el marco general ISO/IEC12207, ISO/IEC15504 y en prácticas y conceptos de PMBOK y SWEBOOK.¹

¹ Se basa en el PMBOK que se llama SWEBOOK y que trata de definir lo que es la profesión de la ingeniería de software y su contenido.

MOPROSOFT

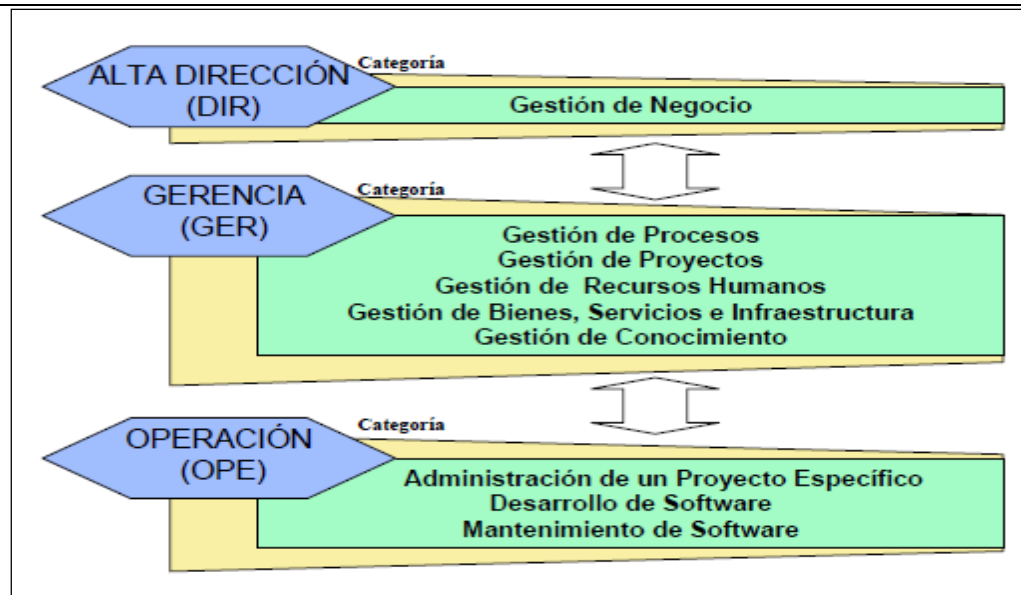
✓ **Características**

- Es específico para el desarrollo y mantenimiento de software.
- Es sencillo de entender y adoptar.
- Facilita el cumplimiento de los requisitos de otros modelos como ISO 9000 y CMMI.
- Se enfoca a procesos.
- Es práctico en su aplicación, principalmente en organizaciones pequeñas, con bajos niveles de madurez.
- Además de ser un marco de referencia o certificación, está orientado a mejorar los procesos y así contribuir a los objetivos de negocio.
- Tiene un bajo costo, tanto para su adopción como para su evaluación.

✓ **Utilidades**

- Mejora la calidad del software producido por la empresa que adopta el modelo.
- Eleva la capacidad de las organizaciones para ofrecer servicios con calidad y alcanzar niveles internacionales de competitividad.
- Integra todos los procesos de la organización y mantiene la alineación con los objetivos estratégicos.
- Inicia el camino a la adopción de los modelos ISO 9000 o CMMI.
- Sirve para implantar un programa de mejora continua.
- Permite reconocer a las organizaciones mexicanas por su nivel de madurez de procesos.
- Facilita la selección de proveedores.
- Permite obtener acceso a las prácticas de ingeniería de software de clase mundial.

Figura IV – 1: Moprosoft



Fuente: link de Comunidad Moprosoft, enero 2013. <http://www.comunidadmoprosoft.org.mx>

CMMI

✓ **Madurez Vs Inmadurez**

Figura IV – 2: Madurez vs Inmadurez - CMMI

Organizaciones con procesos inmaduros	Organizaciones con procesos maduros
Procesos improvisados por los gerentes y desarrolladores	Procesos documentados
Cada uno posee sus propios procesos	Procesos seguidos consistentemente
Procesos comprometidos en orden a cumplir los costos y las fechas acordadas	El rendimiento de los procesos es medido, seguido y entendido
Calidad difícil de predecir	La calidad es predecible porque los procesos están bajo control
Los procesos “viven” mientras viven los desarrolladores	Los procesos “viven” por si solos y son mejorados continuamente
Las nuevas tecnologías corren riesgo de caer en desuso	Las nuevas tecnologías son incorporadas de una manera disciplinada
Tiene pocos recursos propios	Incrementa la productividad
Tiene altibajos en la productividad por rotación del recursos	Entrega con la calidad esperada
Los empleados están descontentos	Satisface a los clientes

Elaboración: los autores, Enero 2013

✓ **Necesita su Organización CMMI**

- Los planes se hacen, pero no necesariamente se siguen.
- No se hace el seguimiento al trabajo real vs el plan. Los planes no son revisados.
- Los requerimientos no son consistentes, los cambios no son manejados.
- Los estimados son muy irreales, su incumplimiento es común.
- Cuando no se puede cumplir con los plazos, surge una atmósfera de crisis.
- Los defectos se encuentran en la fase de pruebas, o peor aún los encuentra el cliente.
- El éxito depende de acciones heroicas de individuos competentes.
- La consistencia en la ejecución es cuestionable

✓ **Se utilizará CMMI para:**

- Ayudará a establecer objetivos y prioridades en mejoras de procesos.
- Ayudará a asegurar procesos estables maduros y con la capacidad requerida.
- Como guía para mejoras de procesos a nivel de proyecto y de organización.
- Como una metodología de evaluación para diagnosticar el estado de los esfuerzos de mejora.
- Se usará la metodología Identificación, Transformación e Implantación - ITI, para transformar la Unidad de Desarrollo de Sistemas a los estándares requeridos por el modelo CMMI de una manera simple y sencilla.

✓ **Implementación del Modelo CMMI**

- La implementación del modelo de CMMI consta de dos partes:
- Consultoría especializada: consiste en realizar el acompañamiento dirigido por un consultor senior, durante un tiempo estimado de un año por nivel de madurez.
- Valoración SCAMPI: consiste en un proceso mediante el cual durante un tiempo estimado de tres meses se recoge evidencias para

comprobar si la organización ha alcanzado el nivel de madurez deseado. La valoración es realizada por una empresa autorizada por el SEI (Software Engineering Institute).

✓ **Niveles y categoría del Modelo CMMI**

- Verificaciones y validaciones del producto y documentar.

Figura IV – 3: Niveles y Categoría del Modelo CMMI

Categoría Nivel de Madurez	Procesos	Gestión de Proyectos	Ingeniería	Soporte
5	OID			CAR
4	OPP	QPM		
3	OPF, OPD, OT	IPM, RSKM, IT, ISM	RD, TS, PI, VER, VAL	DAR, OEI
2		PP, PMC, SAM	REQM	MA, PPQA, CM

Elaboración: los autores, Enero 2013

CAPITULO V

SIMULACION Y RESULTADOS ESPERADOS

5.1 Introducción

En este capítulo, se muestran los formatos llenados para nuestro modelo de Diseño de una Metodología de Certificación de Productos de Software orientado al Sector Público, aplicado al Banco de Nación.

5.2 Listado de formatos y documentos llenados

Se aplicarán a requerimientos (PR) nro. 000912 que tienen prioridad media.

Tiene como requerimiento de sistema:



- ✓ RSIS1: Creación de transacción de asignación de recurso.
- ✓ RSIS2: Creación de transacción de consulta de recursos asignados a los proyectos de desarrollo.

- ✓ RSIS3: Creación de transacción de consulta de recursos asignados a proyecto de mantenimiento.
- ✓ RSIS4: Generación de notificaciones de asignación y/o próximo término de asignación.
- ✓ RSIS5: Reporte de nuevas asignaciones.

Descripción de requerimiento

Creación de transacción de asignación y consulta de recursos asignados a los proyectos de desarrollo y/o mantenimiento, generando notificaciones de asignación y/o próximo término de asignación.



		MCPS.001 Acta de Reunión [SAR – PR - 000912]			
Número de Acta: AR SAR._001					
Sistema	Sistema Administración de Requerimiento			Fecha	18- 09-2013
Usuario	BN			Hora Inicio	9:00
Lugar	BN: Sala de reuniones Departamento de Informática			Hora Término	10:00



Asistentes	
Sistemas	Usuario
Angel Gil Paredes –Jefe Proyecto	Enrique Tello – Stakeholder
Gerson Gonzales – Analista	Orlando Yauri
Alan Perez – Programador	Raul Garcia

Agenda
1.Complejidad del PR 000912
2.Tiempo y Recurso

Acuerdos	
Descripcion	Responsable
Prioridad Media PR 000912	Gerson Gonzales
Duracion 150 horas	

Por sistemas



Por Usuario

	MCPS.2 Lista Maestra de Requerimientos [SAR – PR - 000912]	
LMR-SAR-{18-09-2013}		

Id. Requerimiento Sistema	RSIS1: Creación de transacción de asignación de recurso. RSIS2: Creación de transacción de consulta de recursos asignados a los proyectos de desarrollo. RSIS3: Creación de transacción de consulta de recursos asignados a proyecto de mantenimiento. RSIS4: Generación de notificaciones de asignación y/o próximo termino de asignación. RSIS5: Reporte de nuevas asignaciones
Id. Usuario	RUSU1
Descripción de requerimiento	Creación de transacción de asignación y consulta de recursos proyectos de desarrollo y/o mantenimiento, generando notificaciones de asignación y/o próximo termino de asignación
Tipo requerimiento de sistema	<input type="radio"/> Interactivo <input type="radio"/> Reporte
Es interfaz	<input type="radio"/> Interfaz interna
Tipo de desarrollo	<input type="radio"/> Programa modificado / Ampliación de funcionalidad actual
Afecta arquitectura	<input type="radio"/> No
Dificultad	<input type="radio"/> Media
Prioridad	<input type="radio"/> Crítica
Estado	<input type="radio"/> Propuesto
Juicio Experto - Tiempo en días de construcción	150 Horas

Por sistemas

Por usuario

	MCPS.5 Documento de Análisis [SAR – PR - 000912]	
DA-SAR 21-09-2009		

PR.	000912
Tipo de PR	1) Creación (C) Son los requerimientos que adicionan una nueva funcionalidad usuaria al Sistema.
Fecha de Versión	21-09-2013
Versión	1.0

1. Descripción detallada de Requerimiento	RSIS1: Creación de transacción de asignación de recurso. RSIS2: Creación de transacción de consulta de recursos asignados a los proyectos de desarrollo. RSIS3: Creación de transacción de consulta de recursos asignados a proyecto de mantenimiento. RSIS4: Generación de notificaciones de asignación y/o próximo termino de asignación. RSIS5: Reporte de nuevas asignaciones.
2. Situación Actual	Existe una sobrecarga de asignación a recursos, los cuales son debidos a la no existencia de reporte de control por tipo de proyecto y estadística de disponibilidad de recurso, esto no permite una adecuada asignación de recurso, a los diferentes proyectos que maneja el Departamento de Informática.
3. Alcance	3.1. <u>Detalle de Alcance</u> De acuerdo al requerimiento de usuario se define los siguientes RSIS como alcance de departamento de informática, para la administración de sus recursos técnicos. 3.2. <u>Otros sistemas impactados</u> Ninguno. 3.3. <u>Requerimientos relacionados</u>

	<p>El PR se deberá ejecutar en forma transaccional de manera tal que permita asignar, reasignar, eliminar recursos a los diferentes proyectos en revisión, la emisión de los reportes se deberá efectuar los fines de semana a fin detener los días lunes el reporte para su análisis y trabajo a efectuar.</p> <p>3.4. <u>Exclusiones</u> Ninguna.</p>
<p>4. Especificaciones funcionales: (a detalle)</p>	<p>4.1. <u>Detalle de la solución</u></p> <p>RSIS1: Creación de transacción de asignación de recurso. Creación de una pantalla que permita visualizar los proyectos y/o mantenimientos con la lista de recursos asignados y/o por asignar.</p> <p>RSIS2: Creación de transacción de consulta de recursos asignados a los proyectos de desarrollo. Creación de una pantalla que permita visualizar la consulta de los proyectos con la lista de recursos asignados y/o por asignar.</p> <p>RSIS3: Creación de transacción de consulta de recursos asignados a proyecto de mantenimiento. Creación de una pantalla que permita visualizar la consulta de los proyecto mantenimiento con la lista de recursos asignados y/o por asignar.</p> <p>RSIS4: Generación de notificaciones de asignación y/o próximo termino de asignación. Creación de una pantalla que permita visualizar las notificaciones de asignación y/o próximo término de asignación.</p> <p>RSIS5: Reporte de nuevas asignaciones Generación de reporte online y/o generación periódica de las nuevas asignaciones</p> <p>RSIS6: Reporte termino de asignación Generación de reporte online y/o generación periódica de termino de asignaciones</p> <p>RSIS7: Reporte disponibilidad recurso Generación de reporte online y/o generación periódica de la disponibilidad de recurso.</p> <p>4.2. <u>Impacto en la operativa</u> Con la implementación del PR, permitirá llevar un control efectivo de la asignación real de los recursos</p>

	<p>evitando la sobrecarga y la inoperatividad de algunos recursos.</p> <p>4.3. <u>Accesos</u></p> <p>Para ingreso, actualizaciones y consultas; Jefe de Departamento, Jefe de proyecto.</p> <p>Para consultas, Analistas, programadores y/o personal técnico asociado.</p>
<p>5. Especificaciones técnicas (a detalle):</p>	<p>5.1. <u>Especificaciones técnicas detalladas: [Según el PR, asumir algunas o todas las partes indicadas a continuación]</u></p> <p>5.1.1. <u>Descripción del desarrollo del requerimiento:</u></p> <p>RSIS1: Creación de transacción de asignación de recurso. Definir los recursos CICS, elaborar el programa Cobol Cics que lea el archivo de Recursos (Analistas, Programadores, Revisores, Otros), y el archivo de Proyectos y/o Mantenimientos Aprobados, que valide la disponibilidad de los recursos a asignar y cree un registro log de las actualizaciones efectuadas, así como el registro de recurso por proyectos.</p> <p>RSIS2: Creación de transacción de consulta de recursos asignados a los proyectos de desarrollo. Definir los recursos CICS, elaborar el programa Cobol Cics que de acuerdo al proyecto (de la lista o código ingresado) lea el archivo de recursos por proyecto y muestre la información de recursos asignados así como su status a la fecha de consulta.</p> <p>RSIS3: Creación de transacción de consulta de recursos asignados a proyecto de mantenimiento. Definir los recursos CICS, elaborar el programa Cobol Cics que de acuerdo al mantenimiento (de la lista o código ingresado) lea el archivo de recursos por proyecto y muestre la información de recursos asignados así como su status a la fecha de consulta.</p> <p>RSIS4: Generación de notificaciones de asignación y/o próximo termino de asignación.</p>

De acuerdo con la información registrada en el Archivo de Recursos por Proyecto, para las nuevas asignaciones genera correos de notificaciones y para aquellos que estén por terminar se notifique para la elaboración de cierre de sus actividades.

RSIS5: Reporte de nuevas asignaciones

De acuerdo con la información registrada en el Archivo de Recursos por Proyecto, para las nuevas asignaciones genere una lista de información que puede ser impresa.

RSIS6: Reporte termino de asignación

De acuerdo con la información registrada en el Archivo de Recursos por Proyecto, para las asignaciones cuyo término se dé en el transcurso de la próxima semana, genere una lista de información que puede ser impresa.

RSIS7: Reporte disponibilidad recurso.

De acuerdo con la información registrada en el Archivo de Recursos y la información del archivo de Recursos por Proyecto, muestre la información sobre la disponibilidad del recurso que se esté dando en el periodo de 15 días después de la consulta efectuada esta información puede ser impresa.

5.1.2.Objetos de Aplicación

a) Para aplicativo WEB

No aplica

b) Para aplicativo Cliente/Servidor

No aplica

c) Para aplicativo Host

Definición de Recursos CICS (Transacciones, archivos, programas, pantallas, mapas, etc.)

5.1.3.Objetos de base de datos, Tablas, Archivos

Creación y/o Actualización de Archivos Cics que servirán para la atención de la Asignación y/o Consulta de Recurso asignados a los Proyectos y/o Mantenimientos.

5.2. Tiempos estimados para la atención del requerimiento.



Actividades	Total Horas
INICIO	
* Elaboración del Documento de Preanálisis	0.00 horas
ELABORACIÓN	
* Análisis y Elaboración del Documento de Análisis - Sección Funcional	15.00 horas
* Análisis y Elaboración del Documento de Análisis - Sección Técnica	15.00 horas
* Verificación del Documento de Análisis	2.00 horas
CONSTRUCCIÓN	
* Desarrollo de la Solución	40.00 horas
* Pruebas Internas	10.00 horas
* Control de Calidad Interno	20.00 horas
* Autoriza Pase a QA	8.00 horas
* Pruebas Funcionales y de Sistemas	33.00 horas

	TRANSICIÓN	
	* Pase a Producción	10.00 horas
	Total	150.00 horas
6. Capacitación (SI) / (NO)	Si	
7. Documentación	Ninguna	



Por sistemas

Por usuario





		MCPS.14 Checklist Mantenimiento [SAR – PR - 000912]					
Checklist M-SAR 7-10-2013							
REF.	PUNTOS DE CONTROL	SI	NO	NA	NC	OBSERVACIONES / EVIDENCIAS / JUSTIFICACION	
1	DOCUMENTO DE PREANALISIS						
1.1	¿Se elaboró el documento de Preanálisis?			x			
1.2	¿El PR que implica: Creación, Modificación, Optimización o Procesamiento de datos?			x			
1.3	¿Se ha detallado la sección: Situación actual?			x			
1.4	Dentro de la sección Alcance se han detallado los siguientes ítems:			x			
1.5	1.5.1 Detalle del alcance			x			
	1.5.2 Exclusiones			x			
1.6	¿Se ha detallado la sección: Marco Conceptual?			x			
1.7	¿Se ha detallado la sección: Anexos?			x			
1.8	¿Se ha detallado la sección: Tiempos estimados para la atención del requerimiento?			x			
1.9	¿Se firmó última versión de documento de Preanálisis? Indicar N° versión firmada.			x			
2	DOCUMENTO DE ANÁLISIS						
2.1	¿Se ha detallado la sección: Situación actual?	x					
2.2	Dentro de la sección Alcance se han especificado los siguientes ítems:	x					
2.2.1	Detalle del alcance	x					
2.2.2	Otros sistemas impactados			x			

2.2.3	Requerimientos relacionados			X		
2.2.4	Exclusiones			X		
2.3	¿Se han descrito todos los conceptos necesarios dentro de la sección: Marco Conceptual?			X		
2.4	¿Se ha detallado la sección: Especificaciones Funcionales?	X				
2.5	¿Se han especificado los Anexos de Referencia Relacionados con el PR?	X				
2.6	Dentro de la sección Análisis de la Solución se han descrito los siguientes ítems:	X				
2.6.1	Detalle de la solución.	X				
2.6.2	Impacto en la operativa.	X				
2.6.3	Accesos.	X				
2.7	Indique si se han detallado los siguientes ítems de la sección: Especificaciones Técnicas.	X				
2.7.1	Especificaciones técnicas detalladas.	X				
2.7.1.1	Descripción del desarrollo del requerimiento.	X				
2.7.1.2	Objetos de Aplicación.	X				
2.7.1.3	Objetos de base de datos y/o archivos.	X				
2.7.2	Tiempos estimados para la atención del requerimiento.	X				
2.7.2	Complejidad del PR.	X				
2.8	¿Se ha especificado si se requerirá o no Capacitación al usuario?		X			
2.9	¿Se ha indicado si habrá que documentar algún manual del sistema?			X		
2.10	¿Se aprobó la última versión del documento de Análisis? Indicar N° versión aprobada.	X				

		MCPS.14 Checklist Analista / Programador [SAR – PR - 000912]					
Checklist A/ P -SAR 07- 10 -2013							
REF.	PUNTOS DE CONTROL	SI	NO	NA	NC	Observaciones / Evidencias / Justificación	
REUNIÓN DE PREANÁLISIS							
	¿Se tiene evidencias de la reunión de Preanálisis (Documento de Levantamiento de la Información, Existe el Acta o E-mail que aprueba esta reunión)?			X			
1	FASE DE INICIO: PREANALISIS						
1.1	¿Se elaboró el documento de Preanálisis?			X			
1.2	¿Se firmó última versión de documento de Preanálisis? Indicar N° versión firmada.			X			
2	FASE DE ELABORACIÓN: ANALISIS DE REQUERIMIENTO						
2.1	¿El PR que implica: Crear o Modificar u Optimizar o Error de una funcionalidad, o Procesamiento de datos?	X					
2.11	¿Se han incluido los casos de uso, así como las definiciones y diagramas de entidades y clases nuevas/modificadas en el documento de análisis?			X			
2.12	¿Se modificarán manuales?		X				
2.13	¿Se requerirá capacitación?		X				
2.2	¿Se actualizó el documento de Análisis? Indicar N° última versión.	X				VER 1.0	
2.3	¿Se firmó la última versión de documento de	X				VER 1.0	

	Análisis? Indicar N° última versión firmada.					
3	FASE DE CONSTRUCCIÓN: Checklist del Programador					
3.1	¿Se revisó el cumplimiento de los estándares de BN para las fuentes?			X		
3.2	¿Se revisó el cumplimiento de los estándares de BN para las sentencias de manejo de datos (SQL)?	X				
3.3	¿Se revisó el cumplimiento de los estándares de BN en Seguridad de Información?	X				
3.4	¿Se ejecutaron y validaron las pruebas unitarias necesarias para el PR? ¿Se almaceno en el VSS?	X				
3.5	¿Se cotejó la correspondencia que el código construido o modificado fue revisado con lo especificado en documento de Análisis?	X				
3.6	¿Se cargaron correctamente los objetos construidos en el HARVEST para Cliente Servidor; ENDEVOR para elemento Host?	X				
3.1	FASE DE CONSTRUCCIÓN: Preparación del Ambiente de Pruebas (QA).					
3.7	¿Se verificó que los objetos del doc. De Pase a Producción concuerdan con los del HARVEST para Cliente Servidor; ENDEVOR para elemento Host?	X				
3.2	FASE DE CONSTRUCCIÓN: Pruebas Funcionales y de Sistemas					
3.8	¿Se aprobaron las pruebas funcionales con el Usuario responsable?		X			
3.9	¿Se aprobaron las pruebas de sistemas con el responsable de los usuarios?		X			

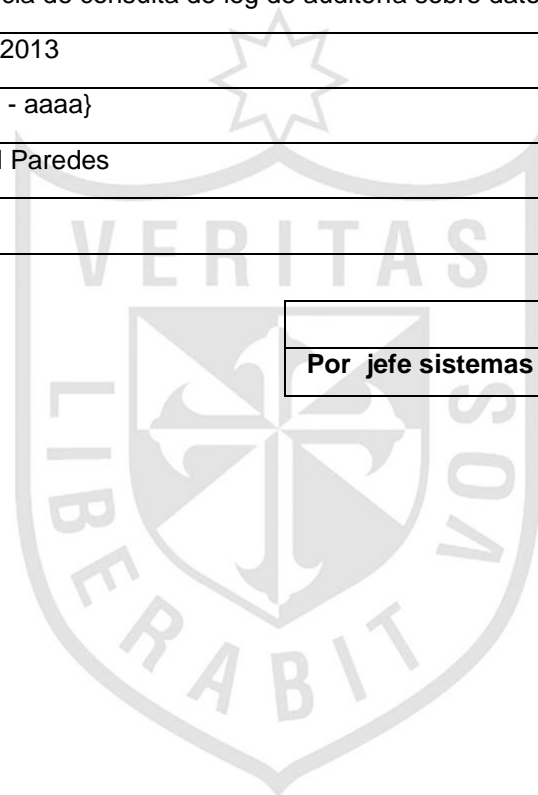
	MCPS.6 Lista de Incidencias [SAR – PR - 000912]	
LI- SAR -{15-10-2013}		



PR.	RSIS1: Creación de transacción de asignación de recurso. RSIS2: Creación de transacción de consulta de recursos asignados a los proyectos de desarrollo. RSIS3: Creación de transacción de consulta de recursos asignados a proyecto de mantenimiento. RSIS4: Generación de notificaciones de asignación y/o próximo termino de asignación. RSIS5: Reporte de nuevas asignaciones.
Ciclo / Proyecto	Mantenimiento.
Tipo Incidencia	<input type="radio"/> Software
Fase	<input type="radio"/> Inicio <input type="radio"/> Construcción
Origen	<input type="radio"/> Calidad <input type="radio"/> Internas
Sistema	SAR
Ubicación	Inexistencia de Funcionalidad
Fecha registro	14 –10 – 2013
Clasificación	<input type="radio"/> Código <input type="radio"/> Forma <input type="radio"/> Programa <input type="radio"/> Funcional

Tipo	○ Error / Aclaración
Estado	○ Pendiente
Caso de prueba / Entregable	Reporte de Mantenimiento Efectuados
Descripción	Inexistencia de consulta de log de auditoría sobre datos de quien efectuó la actualización
Fecha comprometida	20 -10 -2013
Fecha real	{dd -mm - aaaa}
Responsable	Ángel Gil Paredes
# Veces	1

Por analista

Por jefe sistemas



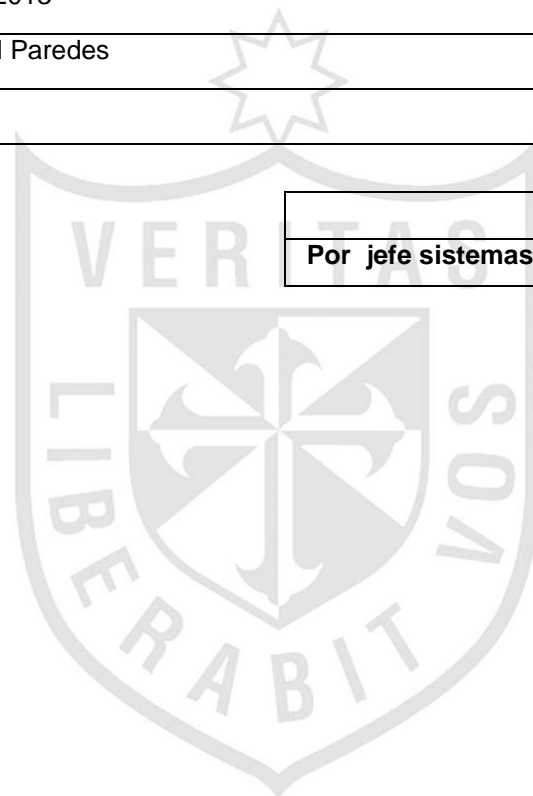
	MCPS.6 Lista de Incidencias [SAR – PR - 000912]	
LI- SAR -{22-10-2013}		



PR.	RSIS1: Creación de transacción de asignación de recurso. RSIS2: Creación de transacción de consulta de recursos asignados a los proyectos de desarrollo. RSIS3: Creación de transacción de consulta de recursos asignados a proyecto de mantenimiento. RSIS4: Generación de notificaciones de asignación y/o próximo termino de asignación. RSIS5: Reporte de nuevas asignaciones.
Ciclo / Proyecto	Mantenimiento
Tipo Incidencia	<input type="radio"/> Software
Fase	<input type="radio"/> Inicio <input type="radio"/> Construcción
Origen	<input type="radio"/> Calidad <input type="radio"/> Internas
Sistema	SAR
Ubicación	Inexistencia de Funcionalidad
Fecha registro	22 –10 – 2013
Clasificación	<input type="radio"/> Código <input type="radio"/> Forma <input type="radio"/> Programa <input type="radio"/> Funcional
Tipo	<input type="radio"/> Error / Aclaración
Estado	<input type="radio"/> Aceptada

Caso de prueba / Entregable	Reporte de Mantenimiento Efectuados
Descripción	Inexistencia de consulta de log de auditoría sobre datos de quien efectuó la actualización
Fecha comprometida	20 -10 -2013
Fecha real	22 -10 -2013
Responsable	Ángel Gil Paredes
# Veces	1

Por analista

Por jefe sistemas





	<p align="center">MCPS.10 Documento Plantilla de Pase a producción [SAR – PR - 000912]</p>	
PPP- SAR 24-10-2013		

PR	000912
1. Descripción	RSIS1: Creación de transacción de asignación de recurso. RSIS2: Creación de transacción de consulta de recursos asignados a los proyectos de desarrollo. RSIS3: Creación de transacción de consulta de recursos asignados a proyecto de mantenimiento. RSIS4: Generación de notificaciones de asignación y/o próximo termino de asignación. RSIS5: Reporte de nuevas asignaciones. RSIS6: Reporte termino de asignación. RSIS7: Reporte disponibilidad recurso.
2. Módulo – Opción	Sistema Administración Requerimiento.
3. Descripción	Creación de transacción de asignación y consulta de recursos asignados a los proyectos de desarrollo y/o mantenimiento, generando notificaciones de asignación y/o próximo término de asignación.
4. Librerías y Objetos	4.1. Objetos para el pase a ambiente QA Iteración 1 – Fecha de solicitud QA [12/10/2009] 4.1.1.Requisitos Definición de transacciones, mapas (pantallas), programas, archivos, en el CICS Administrativo. 4.1.2.Objetos compilados Programas, mapas. 4.1.3.Objetos fuentes Programas, mapas. 4.1.4. Creación y/o modificación de archivo

	<p>Archivo de recursos</p> <p>Archivo de mantenimiento y/o proyecto</p> <p>Archivo de recurso - proyecto.</p> <p>4.1.5. Registro de programas en el sistema. De acuerdo a lo instruido</p> <p>4.1.6. Observaciones De acuerdo a la confiabilidad no ponen los nombres de los programas, mapas, transacciones y archivos usados.</p>
5. Programa	<p><u>Pase a Producción:</u></p> <p>5.1. Requisitos Que se encuentren definido en el CICS</p> <p>5.2. Procedimiento a seguir</p> <p>1. Luego del pase darle new copy a los programas y mapas, definidos.</p> <p>5.3. Configuración en cliente No aplica.</p> <p>5.4. Iteración a ejecutar. Iteración 1 – Fecha de solicitud QA [24/10/2013]</p>
6. Directorio	{Indicar las rutas donde se encuentran las librerías}.
7. Usuarios que tienen acceso	{ [Nombres Y Apellidos] – [Cargo] – [Nivel de Acceso] (cuando el requerimiento es nuevo) }.

Por sistemas

Por usuario

	MCPS.13 Control De Calidad Del Pase A Producción. (Checklist) [SAR – PR - 000912]	
Checklist SAR 23-10-2013		



PR.	000912
1. Descripción	RSIS1: Creación de transacción de asignación de recurso. RSIS2: Creación de transacción de consulta de recursos asignados a los proyectos de desarrollo. RSIS3: Creación de transacción de consulta de recursos asignados a proyecto de mantenimiento. RSIS4: Generación de notificaciones de asignación y/o próximo termino de asignación. RSIS5: Reporte de nuevas asignaciones.

Nº	Tema	ROL	Acción	BN	
				Estado	
				¿Se efectúa la acción?	
				SI	NO
1	Definición de transacciones, mapas (pantallas), programas, archivos, en el CICS Administrativo.	Ing. Sistemas.	Revisar solicitud de definiciones solicitadas.	X	
			Efectuar las definiciones solicitadas.	X	
			Verificar las definiciones efectuadas.	Envía correo de confirmación de acción efectuada.	

Nº	Tema	ROL	Acción	BN	
				Estado	
				¿Se efectúa la acción?	
				SI	NO
2	Creación de archivos: ✓ Archivo de recursos. ✓ Archivo de mantenimiento y/o proyecto. ✓ Archivo de recurso - proyecto.	Producción sistemas.	Efectuar la Solicitud de Creación o Modificación de archivos solicitados.	Envía correo de confirmación de acción efectuada.	
3	Definición de accesos a usuario autorizados.	Oficial de Seguridad.	Creación o modificación de usuario en la aplicación.	Otorga conformidad de creación o modificación de usuario.	

Por sistemas



Por usuario

		MCPS.14 Checklist Mantenimiento [SAR – PR - 000912]					
Checklist M-SAR 23-10-2013							
REF.	PUNTOS DE CONTROL	SI	NO	NA	NC	OBSERVACIONES / EVIDENCIAS / JUSTIFICACION	
1	DOCUMENTO DE PREANÁLISIS						
1.1	¿Se elaboró el documento de Preanálisis?			x			
1.2	¿El PR que implica: Creación, Modificación, Optimización o Procesamiento de datos?			x			
1.3	¿Se ha detallado la sección: Situación actual?			x			
1.4	Dentro de la sección alcance se han detallado los siguientes ítems:			x			
1.5	1.5.1 Detalle del alcance.			x			
	1.5.2 Exclusiones.			x			
1.6	¿Se ha detallado la sección: Marco conceptual?			x			
1.7	¿Se ha detallado la sección: Anexos?			x			
1.8	¿Se ha detallado la sección: Tiempos Estimados para la atención del requerimiento?			x			
1.9	¿Se firmó última versión de documento de Preanálisis? Indicar N° versión firmada.			x			
2	DOCUMENTO DE ANÁLISIS						
2.1	¿Se ha detallado la sección: Situación actual?	x					
2.2	Dentro de la sección alcance se han especificado los siguientes ítems:	x					
2.2.1	Detalle del alcance.	x					

REF.	PUNTOS DE CONTROL	SI	NO	NA	NC	OBSERVACIONES / EVIDENCIAS / JUSTIFICACION
2.2.2	Otros sistemas impactados.			x		
2.2.3	Requerimientos relacionados.			x		
2.2.4	Exclusiones.			x		
2.3	¿Se han descrito todos los conceptos necesarios dentro de la sección: Marco conceptual?			x		
2.4	¿Se ha detallado la sección: Especificaciones Funcionales?	x				
2.5	¿Se han especificado los Anexos de Referencia Relacionados con el PR?	x				
2.6	Dentro de la sección análisis de la solución se han descrito los siguientes ítems:	x				
2.6.1	Detalle de la solución.	x				
2.6.2	Impacto en la operativa.	x				
2.6.3	Accesos.	x				
2.7	Indique si se han detallado los siguientes ítems de la sección Especificaciones Técnicas:	x				
2.7.1	Especificaciones técnicas detalladas	x				
2.7.1.1	Descripción del desarrollo del requerimiento.	x				
2.7.1.2	Objetos de Aplicación.	x				
2.7.1.3	Objetos de base de datos y/o archivos.	x				
2.7.2	Tiempos estimados para la atención del requerimiento.	x				
2.7.2	Complejidad del PR.	x				
2.8	¿Se ha especificado si se requerirá o no Capacitación al usuario?		x			
2.9	¿Se ha indicado si habrá que documentar algún manual del			x		

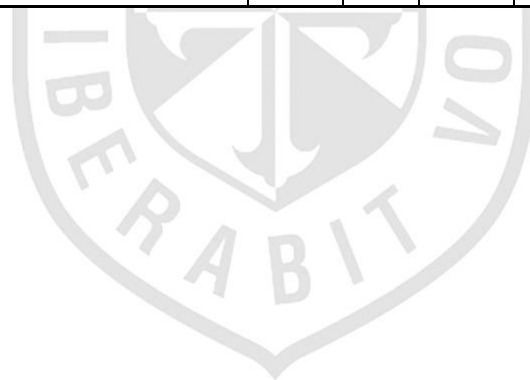
REF.	PUNTOS DE CONTROL	SI	NO	NA	NC	OBSERVACIONES / EVIDENCIAS / JUSTIFICACION
	sistema?					
2.10	¿Se aprobó la última versión del documento de Análisis? Indicar N° versión aprobada.	x				
3	DOCUMENTO DE PRUEBAS FUNCIONALES					
	Para la elaboración y definición de los casos de pruebas:					
3.1	¿Ha utilizado el mismo formato del documento de aceptación de las pruebas funcionales para elaborar los casos de pruebas?	x				
3.2	¿Se elaboraron los casos de prueba, por cada escenario presentado en el documento de Análisis, durante la etapa de implementación?	x				
3.3	¿Se elaboraron los casos de prueba necesarios?	x				
3.4	¿Se solicitó la aprobación de los casos de prueba al UA y/o GRS, antes de las pruebas internas?	x				
3.5	¿El usuario aprobó los casos de prueba, antes de las pruebas internas?		x			
	Para la preparación y realización de las pruebas Funcionales:					
3.6	¿Se programaron con anticipación las pruebas funcionales?	x				
3.7	¿Las pruebas funcionales se llevaron a cabo en la hora programada? Indicar motivo de retraso		x			Corrección de errores que derivó en retraso por falta de indefinición de reporte de cambios.
3.8	¿Las pruebas funcionales se llevaron a cabo de forma correcta?	x				
3.9	¿Se generaron comentarios / ocurrencias como resultado de	x				

REF.	PUNTOS DE CONTROL	SI	NO	NA	NC	OBSERVACIONES / EVIDENCIAS / JUSTIFICACION
	las pruebas funcionales?					
3.10	¿Se confirmó la capacitación?		x			
3.11	¿Se requiere cartilla de usuarios?		x			
3.12	¿Se generaron documentos anexos? Indicar.		x			
3.13	¿Son necesarios requerimientos complementarios?		x			
4	DOCUMENTO DE PASE A QA/PRODUCCIÓN					
	Respecto de los Objetos para el Pase a Producción:					
4.1	¿Se actualizo la lista de iteraciones?	x				
4.2	¿Se especifican los requisitos necesarios?	x				
4.3	¿Se indicaron los objetos compilados?	x				
4.4	¿Se indicaron los objetos fuentes?	x				
4.5	¿Se indicaron los scripts de base de datos?			x		
4.6	¿Es necesario realizar algún registro de programas en el sistema?	x				
	Respecto del Pase a Producción:					
4.7	¿Se especifican los requisitos necesarios?	x				
4.8	¿Se indicó el procedimiento a seguir?	x				
4.9	¿Se especificó si es necesaria alguna configuración en el cliente?			x		
4.10	¿Se indicó la iteración a ejecutar?	x				

		MCPS.14 Checklist Analista / Programador [SAR – PR - 000912]					
Checklist A/ P - SAR 23-10-2013							
REF.	PUNTOS DE CONTROL	SI	NO	NA	NC	Observaciones / Evidencias / Justificación	
REUNIÓN DE PREANÁLISIS							
	¿Se tiene evidencias de la reunión de Preanálisis (Documento de Levantamiento de la Información, Existe el Acta o E-mail que aprueba esta reunión)?			X			
1	FASE DE INICIO: PREANÁLISIS						
1.1	¿Se elaboró el documento de Preanálisis?			X			
1.2	¿Se firmó última versión de documento de Preanálisis? Indicar N° versión firmada.			X			
2	FASE DE ELABORACIÓN: ANÁLISIS DE REQUERIMIENTO						
2.1	¿El PR que implica: Crear o Modificar u Optimizar o Error de una funcionalidad, o Procesamiento de datos?	X					
2.11	¿Se han incluido los casos de uso, así como las definiciones y diagramas de entidades y clases nuevas/modificadas en el documento de análisis?			X			
2.12	¿Se modificarán manuales?		X				
2.13	¿Se requerirá capacitación?		X				
2.2	¿Se actualizó el documento de Análisis? Indicar N° última versión.	X				VER 1.0	
2.3	¿Se firmó la última versión de documento de análisis? Indicar N° última versión firmada.	X				VER 1.0	
3	FASE DE CONSTRUCCIÓN: Checklist del Programador						

3.1	¿Se revisó el cumplimiento de los estándares de BN para las fuentes?			X		
3.2	¿Se revisó el cumplimiento de los estándares de BN para las sentencias de manejo de datos (SQL)?	X				
3.3	¿Se revisó el cumplimiento de los estándares de BN en Seguridad de Información?	X				
3.4	¿Se ejecutaron y validaron las pruebas unitarias necesarias para el PR? ¿Se almaceno en el SS?	X				
3.5	¿Se cotejó la correspondencia que el código construido o modificado fue revisado con lo especificado en documento de Análisis?	X				
3.6	¿Se cargaron correctamente los objetos construidos en el HARVEST para Cliente Servidor; ENDEVOR para elemento Host?	X				
3.1	FASE DE CONSTRUCCIÓN: Preparación del Ambiente de Pruebas (QA).					
3.7	¿Se verificó que los objetos del doc. de Pase a Producción concuerdan con los del HARVEST para Cliente Servidor; ENDEVOR para elemento Host?	X				
3.2	FASE DE CONSTRUCCIÓN: Pruebas Funcionales y de Sistemas					
3.8	¿Se aprobaron las pruebas funcionales con el Usuario responsable?		X			
3.9	¿Se aprobaron las pruebas de sistemas con el responsable de los usuarios?		X			
4	FASE DE TRANSICIÓN: PASE A PRODUCCIÓN					
4.1	¿Los archivos mencionados en el documento de Pase a Producción se encuentran todas en la vista "VERSIONES"?	X				

	del HARVEST para Cliente Servidor; ENDEVOR para elemento Host del respectivo paquete en la vista "APROBACIÓN_DESARROLLO?"					
4.2	¿Los objetos y ejecutables definidos en doc. De Análisis son iguales a los del Doc. Pase Producción?	X				
5	FASE DE TRANSICIÓN: CAPACITACIÓN					
5.1	¿Se preparó material para capacitación a los usuarios?			X		
5.2	¿Se realizó capacitación a los usuarios?			X		
6	FASE DE TRANSICIÓN: DOCUMENTACIÓN					
6.1	¿Se actualizaron los manuales?			X		
7	ENCUESTA DE SATISFACCIÓN					
7.1	¿Se actualizó la información para el envío de encuestas de satisfacción?			X		



		MCPS.001 Acta de Cierre [SAR – PR - 000912]			
Número de Acta: AC SAR_ 001			28- 10 - 2013		
Sistema	Sistema Administración de Requerimiento.	Fecha:	28- 10-2013		
Usuario	BN.	Hora Inicio:	9:00		
Lugar	BN: Sala de reuniones Departamento de Informática.	Hora Término:	10:00		

Asistentes	
Sistemas	Usuario
Angel Gil Paredes –Jefe Proyecto	Enrique Tello – Stateholder
Gerson Gonzales – Analista	Orlando Yauri
Alan Perez – Programador	Raul Garcia
Agenda	
1.Entrega del proyecto PR 000912	
cuerdos	
Descripcion	Responsable
Aceptacion del PR 000912	Gerson Gonzales

Por sistemas

Por usuario

CAPÍTULO VI

CONSTRATE DE HIPÓTESIS

6.1 Prueba Chi Cuadrado

Las pruebas chi-cuadrado son un grupo de contrastes de hipótesis que sirven para comprobar afirmaciones acerca de las funciones de probabilidad (o densidad) de una o dos variables aleatorias.

Estas pruebas no pertenecen propiamente a la estadística paramétrica pues no establecen suposiciones restrictivas en cuanto al tipo de variables que admiten, ni en lo que refiere a su distribución de probabilidad ni en los valores y/o el conocimiento de sus parámetros.

Se aplican en dos situaciones básicas:

- ☞ Cuando queremos comprobar si una variable, cuya descripción parece adecuada, tiene una determinada función de probabilidad. La prueba correspondiente se llama chi-cuadrado de ajuste.
- ☞ Cuando queremos averiguar si dos variables (o dos vías de clasificación) son independientes estadísticamente.

6.1. Resultados

Esta hoja es el Excel de la Pestaña: **Resultados** (dos hojas que se pegan en una sola hoja)



6.3. Valores Esperados

Esta hoja es el Excel de la Pestaña: **Valores esperados.**(son 3 hojas que se pegan y son una sola)



6.4 Decisiones

Decisiones, está en Excel

Son dos pestañas

Primera pestaña de Excel **decisión 1, es una página** (son dos hojas que se pegan)

Para conteo de hoja serian 221



Segunda pestaña de Excel decisión 2, **es una página** (son tres hojas que se pegan)

Para conteo de hoja serian 222



CONCLUSIONES

1. La aplicación de la Metodología de Certificación de Productos de Software orientado al Sector Público ha permitido tener un control adecuado sobre las planificaciones que se deben de efectuar a nivel de tareas, actividades, recursos, costos entre otros, que permiten medir el desempeño, competencia y calidad aplicada.
2. El correcto seguimiento de las actividades definidas a lo largo del desarrollo de la implementación de los requerimientos de software, ha permitido una verificación, validación y captura de información necesaria para dar cumplimiento a las normas de calidad mencionadas en la Metodología sugerida.
3. La metodología propuesta para la Certificación de Productos de Software orientado al Sector Público, es una metodología general, que se ha desarrollado teniendo en cuenta tanto las metodologías existentes en el mercado así como las principales normas y políticas establecidas por las instituciones de control a la que son sometidas las instituciones públicas.
4. Las organizaciones podrán seleccionar las fases, actividades y entregables, de acuerdo a su entorno organizativo, así como a sus necesidades de implementación, que puede ser de forma gradual.

5. Esta metodología propuesta integra el uso de herramientas de TI, existentes en el mercado y los cuales dependiendo de su plataforma de desarrollo deberán de aplicarlas, para el presente desarrollo se hace mención de las herramientas que debería usar el Banco de la Nación.

6. Las nuevas metodologías de certificación de productos de software, surgen como consecuencia de la búsqueda de la flexibilidad para responder rápidamente a las demandas de aplicaciones de TI de la organización en su conjunto (entornos dinámicos, complejos e inciertos). Estas metodologías, lo que permiten, es lograr minimizar los errores de definición, construcción e implementación de productos de Software.



RECOMENDACIONES

1. **Establecer un compromiso formal con la alta dirección.** Esto es sumamente importante y constituye el punto de partida para lograr el desarrollo y consolidación de la implementación de la Metodología de Certificación de Productos de Software.
2. **Desarrollar una identidad propia y única.** Es importante que cada institución de acuerdo a sus posibilidades logre la implementación de forma gradual de la metodología desarrollada, independientemente de poder contar o no con las herramientas de TI para un mejor desarrollo.
3. **Desarrollar una “cultura de confianza”.** Resulta trascendente el desarrollo de una cultura de confianza sobre las aplicaciones solicitadas, esto será en la manera de integrar al usuario cada vez más en el control y desarrollo del proyecto, que permitirá mantener la comunicación sobre el desarrollo efectuado así como de los resultados obtenidos.
4. **Explotar competencias claves.** Desarrollar, permanentemente, actividades de capacitación que permitan conocer a los usuarios el flujo de la información y control que se realiza en las actividades del desarrollo de sus requerimientos solicitados.

5. **Mejorar las comunicaciones.** El uso de herramientas colaborativas, así como de herramientas de gestión, permitirán un conocimiento adecuado de los avances de los proyectos en desarrollo.

6. **Establecer y difundir los beneficios que otorga el aplicar la Metodología de Certificación de Productos de Software.** Se debe gestionar y difundir diversos beneficios que se logra con la implementación y correcta aplicación de la metodología desarrollada.



FUENTES DE INFORMACION

Bibliográficas

1. Lamarca, I., Rodríguez, J., García, J
(2007). Gestión de proyectos informáticos: métodos, herramientas y casos. España, Editorial UOC, S.L
2. Caballero, O.
(2006) **“Tecnologías de Información y Herramientas para la Administración de Proyectos de Software”**. Revista Digital Universitaria. Vol. 7, No. 6. Recuperado de <http://www.revista.unam.mx/vol.7/num6/art47/int47.htm>
3. Cervantes, O.
(2006) Tecnologías de Información y Herramientas para la Administración de Proyectos de Software. Revista Digital Universitaria, 7 (6), pp. 3-5, doi: 1067-6079
4. Gabriel, U.
(2005) Evaluación De Proyectos. 5ta Edición. Mexico, Editorial: Mcgraw-hill.
5. Instituto Nacional de Tecnologías de la Comunicación, S.A. (INTECO)
(2009), Guía de mejores prácticas de calidad de producto. España

6. ISO/IEC 9126 2001
Information Technology – Software quality characteristics and metrics –
Canada. Part 1 Quality characteristics

7. ISO/IEC 20000.
(2010). Guía completa de aplicación para la gestión de los servicios de
tecnologías de la información. España, Edita AENOR

8. Jesús Dextre Tuya.
(2009). Information Technology Infrastructure Library - ITIL.

9. Jacobson, I. Booch, G. y Rumbaugh, J.
(2004) “El Proceso de Desarrollo de Software”. Madrid,
Editorial Addison Wesley,

10. Jacobson, I. Booch, G. y Rumbaugh, J.
(2006), Lenguaje Unificado de modelado UML 2.0. segunda edición. Addison
Wesley

11. MaryBeth Chrissis.
(2009). CMMI: Guía para la Integración de procesos y la mejora de
productos.

12. NTP-ISO/IEC 27001:2008
(2008) EDI. Tecnología de la información Técnicas de seguridad. Sistemas
de gestión de seguridad de la información. Perú. 1ª Edición

13. NTP-ISO/IEC 12207:2006
(2006). Tecnología de la Información, Procesos del ciclo de vida del
software. Perú. 2ª Edición.

14. Oktaba, H., Piattini, M., Pino, F., Orozco, M., Alquicira, C.
(2008) "COMPETISOFT: Mejora de Procesos Software para Pequeñas y Medianas Empresas y Proyectos". Editorial Ra Ma, España.
15. Oktaba H; Su, A., Martinez, A., Quintanilla, G., Ruvalcaba, M., Lopez, F., Alquicira C.
(2005).Modelo de Procesos para la Industria de Software Moprosoft Por Niveles de Capacidad de Procesos editorial Ra Ma, España
16. Palacios, Juan
(2007). Flexibilidad con Scrum. 1ª Edición
17. Pantaleo, G.
(2011) "Calidad en el Desarrollo de Software". 1ra Edición. Buenos Aires, Editorial Alfaomega.
Puede revisar Capitulo I: presenta el concepto de calidad asociados al desarrollo de software. En base a los estándares de calidad.
18. Pressman, Roger
(2012). Ingeniería del Software; un enfoque práctico. México. Mc Graw Hill, Interamericana de España Editores. 5ª Edición.
19. Project Management Institute
(2012). Guía de los Fundamentos de la Dirección de Proyectos (Guía del PMBOK®). 5ta edición. EEUU, Editorial Newtown Square, Pennsylvania. 5ª Edición.
20. Romero M., Gesvin
(2004). UML con Rational Rose. Perú Editorial Megabyte SAC.

21. Software Engineering Institute

(2010). CMMI® for Acquisition, Version 1.3 CMMI-ACQ, V1.3 (CMU/SEI-2010-TR-032). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University.

22. Software Engineering Institute

(2010). CMMI® for Development, Version 1.3 CMMI-DEV, V1.33 (CMU/SEI-2010-TR-033). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University.

23. Software Engineering Institute

(2006). CMMI® for Development, Version 1.2 CMMI-DEV, V1.22 (CMU/SEI-2006-TR-020). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University.

24. Software Engineering Institute

(2010). CMMI® for Services, Version 1.3 CMMI-SVC, V1.3 (CMU/SEI-2010-TR-034). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University.

25. Sommerville, I.

(2005), Ingeniería de Software. 6ta edición. Mexico D.F. Editorial Pearson.

26. Softex.

(2009). MPS.BR – Mejora de Procesos de Software Brasileiro. Guía General

27. Vega, C., Rivera, L., García, A.

(2008). Mejores prácticas para el Establecimiento y aseguramiento de La calidad de software. 1ra Edición electrónica. España.

Electrónicas

1. Altern Digital. Introducción a CMMI [Administración de Proyectos].

Octubre 2009.

En: <http://alterndigital.net/esp/index.php?p=61&more=1&c=1>

2. Appraisal Wizard, 2003. Formal or informal appraisal tool, Integrated System Diagnostics Incorporated. Octubre 2009.
En: <http://www.isd-inc.com>
3. CMMI for Services, Version 1.2. The Software Engineering Institute. Febrero 2009
En: www.sei.cmu.edu/publications/.
4. HM&S IT-Consulting GmbH. CMM Quest v1.2. Self assessment tool. Octubre 2009.
En: <http://www.cmm-quest.com/>
5. IBM. Rational Unified Process Rational Web Sites. Octubre 2009.
En: www.rational.com
6. Proceedings of the Workshop on Software Engineering Foundations for End-User Programming (SEEUP 2009). The Software Engineering Institute. Noviembre 2009
En: www.sei.cmu.edu/publications/.
7. The United States Air Force Software Technology Support Center (STCS). Mapping from CMM-SW to CMMI-SE/SW/IPPD/SS. Octubre 2009.
En: <http://www.stsc.hill.af.mil/consulting/cmmi/documents.html>
8. El portal de la ISO 27001 en español. En: <http://www.iso27000.es>
9. Entidad Nacional de Acreditación. En <http://www.enac.es>
10. Sitio oficial de IBM Rational Unified Process, que ofrece información y

recursos sobre la plataforma de proceso de desarrollo de software configurable En: <http://www-01.ibm.com/software/pe/rational/rup.shtml>

11. Sitio oficial de IBM. Tutorial Rational Unified Processguidance. (2007)

En:

<http://publib.boulder.ibm.com/infocenter/rsarhlp/v8/index.jsp?topic=/com.ibm.xtools.tutorial.rupguidance/topics/rationalunifiedprocessguidance.htm>
!



GLOSARIO

Acción correctiva

Acción tomada para eliminar las causas de no conformidades, defectos u otras situaciones indeseables existentes, para prevenir su ocurrencia.

Las acciones correctivas pueden comprender cambios, tales como en procedimientos y sistemas, para alcanzar el mejoramiento de la calidad en cualquier etapa del ciclo de la calidad. Hay una diferencia entre corrección y acción correctiva: Corrección se refiere a reparación, reproceso o ajuste y se refiere a la disposición de una no conformidad existente. Acción correctiva se refiere a la eliminación de las causas de una no conformidad.

Acción preventiva

Acción tomada para eliminar las causas de potenciales no conformidades, efectos u otras situaciones indeseables para prevenir su ocurrencia.

Las acciones preventivas pueden significar cambios como en procedimientos y sistemas, para alcanzar el mejoramiento de la calidad en cualquier etapa del ciclo de la calidad.

Acreditación

Certificación realizada por un organismo reconocido de la capacidad, objetividad, competencia e integridad de una agencia, servicio, o individuo para certificar el cumplimiento de la Norma ISO 9000.

Actividad

Conjunto de tareas para realizar un determinado proceso. Se pueden tener actividades de entrada para realizar otras actividades.

Actividades de verificación

Una investigación, prueba, inspección, demostración, análisis o comparación especial de datos para verificar que un producto o servicio o proceso cumple con los requerimientos prescritos

Actividades que afectan a la calidad

Cualquier actividad que afecta a la determinación de las características y funciones del producto o servicio, sus especificaciones, realización o verificación, o los medios para planificarlas, organizarlas, controlarlas, asegurarlas o mejorarlas

Adaptación

Adecuar el sistema de información al entorno externo.

Adecuado

Apropiado para el propósito. El término "adecuado" aparece varias veces en el estándar permitiendo al evaluador variar los criterios de adecuación y por tanto, no usar un proceso finito para verificar que los requerimientos han sido cumplidos

Administración de la calidad

Un enfoque de administración de una organización, centrado en la calidad, basado en la participación de todos sus miembros y buscando el éxito a plazo a través de la satisfacción del cliente, y los beneficios para los miembros de la organización y para la sociedad.

Administración de Requerimientos

Se define como el control en todos aquellos requerimientos y sus correspondientes atributos, que son identificados y almacenados en una base de conocimiento para poder identificar el impacto de los cambios que forman parte del proyecto.

Agentes

Conjunto de criterios englobados en el modelo de la EFQM cuyo enfoque realizado por la organización es relevante para la consecución de la excelencia de los resultados empresariales.

Ambiente de uso

Espacio físico y condiciones en que se utiliza el software.

Análisis

Fase en la que se definen las razones y justificaciones de los sistemas de información.

Análisis de requerimientos

Define el momento del “qué de los sistemas de información”.

Aprendizaje

La adquisición y comprensión de información que puede conducir a la mejora o cambio. Ejemplos de actividades de aprendizaje de las organizaciones son el benchmarking, las evaluaciones y/o auditorías internas y externas, y los estudios de mejores prácticas. Ejemplos de aprendizaje individual serían la formación y la cualificación personal.

Aplicación

Aplicación sobre la cual el software va dirigido.

Aprobado

Confirmado como que cumple los requerimientos

Apoyo de los desarrolladores

Asesoría o ayuda por parte de los desarrolladores.

Artefacto

Entregable de una actividad de una metodología. Se pueden tener artefactos de entrada que se utilizan para fabricar otros artefactos.

Aseguramiento

Prueba (verbal o escrita) que asegura que algo ocurrirá o no, o que ha ocurrido o no

Aseguramiento de la calidad

Todas las actividades planificadas y sistemáticas implementadas dentro de un Sistema de la Calidad que permiten demostrar confianza en que un producto o servicio cumplirá con los requisitos de la calidad.

Aspectos humanos

Formación de personal, creación y coordinación de equipos.

Auditoría

Es una herramienta de gestión que comprende una evaluación sistemática, documentada, periódica y objetiva del funcionamiento de la organización en su conjunto o de alguna de las unidades que la integran, usualmente realizada por alguien distinto de la persona responsable de ello.

Auditoría de calidad

Examen sistemático e independiente con el fin de determinar si las actividades y los resultados relativos a la Calidad satisfacen las disposiciones preestablecidas, y si estas disposiciones son aplicadas en forma efectiva y son apropiadas para alcanzar los objetivos.

Auditoría informática

Proceso de recoger, agrupar y evaluar evidencias para determinar si un sistema informatizado salvaguarda los activos, manteniendo la integridad de los datos. Lleva a cabo eficazmente los fines de la organización, administra eficientemente los recursos

Autoevaluación

Examen global, sistemático y regular de las formas de hacer y los resultados de una organización comparados con un Modelo de Excelencia Empresarial. Ofrece una imagen del estado de la organización "en un momento preciso" que suele expresarse en puntos fuertes, áreas de mejora y una puntuación.

Base de testeo

La información y/o documentación que se utilice para diseñar los casos de test.

Calidad

Conjunto de propiedades y de características de un producto o servicio, que le confieren su aptitud para satisfacer unas necesidades explícitas e implícitas.

Calidad externa

La extensión para la cual un producto satisface necesidades explícitas e implícitas cuando es usado bajo condiciones específicas.

Calidad interna

Es la totalidad de atributos del producto que determinan su habilidad para satisfacer las necesidades propias e implícitas bajo condiciones específicas.

Calificación

La acción de evaluar el valor medido al nivel de calificación adecuado. Utilizado para determinar el nivel de calificación asociado con el software para una característica específica de calidad.

Cambiabilidad

Subcaracterística de mantenimiento, que indica la cantidad de esfuerzo requerido para una modificación o borrado de un defecto.

Capacidad de proceso

El rango de resultados esperados que pueden lograrse siguiendo un proceso.

Capacidad de recuperación

Subcaracterística de fiabilidad, que indica la capacidad del sistema para restablecer su nivel de respuesta después de un fallo crítico o error hardware.

Catálogo de servicios

Una lista o repositorio de definiciones de servicios estandarizados. Los catálogos de servicios pueden incluir niveles de detalle variados acerca de los niveles de servicio disponibles, calidad, precios, elementos negociables adaptables, y términos y condiciones.

Casos de test

Conjunto de entradas, precondiciones para la ejecución y salidas esperadas desarrolladas con el objetivo de testear un aspecto concreto del software (ejecutar un camino del programa en particular, verificar la conformidad de un requisito concreto, detectar tipos de errores específicos).

CMMI (Capability Maturity Model Integration)

Modelo para la mejora o evaluación de los procesos de desarrollo y mantenimiento de sistemas y productos de software.

Complejidad del software

Grado en el que se van involucrando muchos elementos físicos (periféricos), que de alguna forma contribuyen con ejecución del software.

Componente

Entregable de una metodología.

Cobertura de decisión

Número de decisiones ejecutadas durante los tests dividido entre número total de decisiones en programa.

Cobertura de instrucción

Número de instrucciones ejecutadas durante los tests dividido entre número total de instrucciones en programa.

Coexistencia

Subcaracterística de portabilidad, que indica la capacidad del software de coexistir con otro software independiente en un entorno común compartiendo recursos.

Comportamiento temporal

Subcaracterística de eficiencia, que indica las características del software que influyen en el tiempo de respuesta y procesado y productividad cuando se ejecuta su función.

Comprensión

Subcaracterística de facilidad de uso, que indica las características del software que influyen en el esfuerzo del usuario para reconocer el concepto lógico y su aplicación.

Control interno

Actividades operativas claves destinadas a prevenir los riesgos efectivos y potenciales a los que se enfrentan las organizaciones.

Controles preventivos

Controles que evitan el hecho, como un software de seguridad que impida los accesos no autorizados al sistema.

Controles detectivos

Controles que manifiestan cuando fallan los controles preventivos para tratar de conocer cuanto antes el evento.

Controles correctivos

Controles que facilitan la vuelta a la normalidad cuando se han producido incidencias.

Defecto

Una manifestación de un error.

Driver

Programa que invoca un componente bajo testeo, por ejemplo para simular un componente cuyo código todavía no está disponible (está todavía en desarrollo) o un componente externo.

Eficiencia

Conjunto de características que determinan la relación entre el nivel de rendimiento del software y el número de recursos usados, bajo ciertas condiciones dadas. Se divide en las subcaracterísticas comportamiento temporal, utilización de recursos.

Error

Una acción humana que puede producir resultados incorrectos.

Extreme programming:

Metodología de desarrollo perteneciente a las metodologías ágiles.

Estabilidad

Subcaracterística de mantenimiento, que indica volumen de riesgos de efectos inesperados tras una modificación.

Facilidad de aprendizaje

Subcaracterística de facilidad de uso, que indica las características software que influyen en el esfuerzo del usuario para aprender su aplicación (control, entrada, salida).

Facilidad de instalación

Subcaracterística de portabilidad, que indica las características del software que influyen en el esfuerzo requerido para instalar el software en un entorno especificado.

Facilidad de prueba

Subcaracterística de mantenimiento, que indica la capacidad del software para permitir que sea validado tras ser modificado.

Facilidad de uso

Conjunto de características que influyen en el esfuerzo requerido para el uso y la evaluación individual de cada uso por parte de un conjunto de usuarios dados. Se divide en las subcaracterísticas comprensión, facilidad de aprendizaje, operatividad, atractivo.

Fallo

Una desviación del funcionamiento esperado.

Fase

Conjunto de actividades agrupadas que tienen como punto final un hito.

Fiabilidad

Grado en que el sistema responde bajo las condiciones definidas durante un intervalo de tiempo dado. Se divide en las subcaracterísticas madurez, tolerancia a fallos, capacidad de recuperación.

Funcionalidad

Grado en que las necesidades asumidas o descritas se satisfacen. Se dividen en las subcaracterísticas idoneidad, precisión, interoperabilidad, seguridad.

Idoneidad

Subcaracterística de funcionalidad, que indica el grado en que las funciones que soportan las tareas especificadas están presentes.

IEEE 829

Estándar para elaborar la documentación de testeado de software.

Ingeniería de software

Disciplina tecnológica y administrativa orientada a la producción sistemática de productos de programación, que son desarrollados y modificados a tiempo dentro de un presupuesto definido.

Integridad

Grado con que puede controlarse el acceso al software o a los datos a personal no autorizado. Proceso que permite eliminar errores que se presenten en la etapa de prueba.

Interacción

Interacción con el usuario final, donde se establece la comunicación entre el usuario y los desarrolladores.

Inspección

Una revisión en que el líder prepara un “checklist” que sirve como guía de la reunión y contiene los puntos en que los revisores se tienen que fijar. El líder distribuye el checklist, el artefacto bajo testeo y otros materiales a los participantes antes de la reunión. Los revisores tienen que estudiar el checklist y el artefacto bajo testeo antes de la reunión.

Interoperabilidad

Subcaracterística de funcionalidad, que indica el grado en que el sistema puede interactuar con otros sistemas.

ISO 9000

Normativa de calidad en la gestión y aseguramiento de calidad de software. Define los conceptos y directrices de la calidad de software.

ISO/IEC 9126

Estándar que define un modelo de calidad de producto software.

ITIL (InformationTechnology Infrastructure Library)

Es un compendio de publicaciones, o librería, que describen de manera sistemática un conjunto de “buenas prácticas” para la gestión de los servicios de Tecnología Informática (en adelante TI).

Madurez

Subcaracterística de fiabilidad, que indica la frecuencia con la que ocurren los fallos.

Mantenimiento

Esfuerzo requerido para implementar cambios. Se divide en las subcaracterísticas capacidad de ser analizado, confiabilidad, estabilidad, facilidad de prueba.

Meta

Nivel de resultados (producto, efecto, calidad, eficiencia, etc.) que debe ser alcanzado.

Metodología de desarrollo

Conjunto de procesos y actividades delimitadas para cumplir el objetivo del desarrollo de software.

Métricas de software

Aquella aplicación continúa de técnicas basadas en la medida de los procesos de desarrollo del software, para producir una información de gestión significativa.

Mejores prácticas

Conjunto de acciones cuya principal disciplina en los equipos de desarrollo de software es garantizar la calidad mediante la reducción de fallas al liberar el sistema.

Moprosoft

El propósito del Modelo de Procesos para la Industria de Software (Moprosoft) en México es fomentar la estandarización de las operaciones en las organizaciones, a través de la incorporación de las mejores prácticas en gestión e ingeniería de software.

Niveles de madurez

Procesos de desarrollo de software en una escala de cierta cantidad de niveles en donde se tienen en cuenta aspectos muy variados de los procesos de desarrollo como el grado de ambigüedad de las especificaciones, la verificación independiente de la fiabilidad de los programas, etc.

NTP

Norma Técnica Peruana.

OMG

Object Management Group

Operatividad

Subcaracterística de facilidad de uso, que indica las características del software que influyen en el esfuerzo del usuario para operar y control operacional.

Outsourcing

Subcontrata de las partes de procesos relacionados con las TICs para que sean realizados por empresas externas.

Portabilidad

Conjunto de características que determinan la capacidad del software para ser transferido de un entorno de operación a otro. Se divide en las subcaracterísticas adaptabilidad, facilidad de instalación, coexistencia, reemplazo.

PR

Priorización de Requerimientos

Precisión

Subcaracterística de funcionalidad, que indica el grado de exactitud de los efectos del sistema.

Procesos

Son una secuencia de actividades orientadas a generar un valor añadido sobre una ENTRADA para conseguir un resultado, y una SALIDA que a su vez satisfaga los requerimientos del cliente.

Reemplazo

Subcaracterística de portabilidad, que indica las características del software que influyen en la posibilidad y esfuerzo requerido para usarlo en lugar de otro software en el mismo entorno.

Revisión

Reuniones de un grupo definido de personas cuyo objetivo es encontrar errores en un artefacto de software; sirven para testear requisitos, diseño, planes, manuales y software. Participantes de las revisiones son: los autores que han escrito el artefacto; los revisores que tienen que detectar errores; el secretario que documenta los errores encontrados; el presentador que expone/explica el artefacto bajo testeo; el líder que dirige la reunión, elige la fecha para la reunión y invita a los participantes. Generalmente se distingue 2 tipos de revisiones: inspecciones (formal) walkthroughs (más informal).

Rol

Función realizada por un usuario de una metodología.

RUP (Rational Unified Process)

Metodología de desarrollo perteneciente a las metodologías pesadas.

SEI

Software Engineering Institute.

Seguridad

Subcaracterística de funcionalidad, que indica el grado en que un acceso no autorizado (accidental o deliberado) se prevenga y se permita un acceso autorizado.

Stakeholder

Cualquier persona interesada en, afectada por y/o implicada con el funcionamiento del sistema software. Por ejemplo, el usuario, el cliente, nuestra empresa, etc.

Test de aceptación

Dirigido a los criterios de aceptación previamente establecidos (por ejemplo con el cliente).

Testeo de caminos

Técnica que permite derivar una estructura de flujo de un diseño procedural o código y usar esta estructura como una guía para definir un conjunto básico de casos de test (caminos de ejecución).

Testeo de comportamiento

El desarrollo de los casos de test se basa en las funcionalidades y/o el comportamiento que el software debe tener (p.ej. requisitos, especificaciones, conocimiento del dominio, repositorio de defectos, etc.).

Testeo de configuración

Testeo de un sistema bajo diferentes configuraciones de:

- Hardware: discos duros, impresoras, CPU, sensores, tarjetas gráficas, tarjetas de sonido
- Sistemas operativos y/o versiones de un sistema operativo
- Sistemas GUI (p.ej. MS Windows, X-Windows) y sus diferentes versiones
- Bases de datos, etc.

Testeo de estrés

Testeo del comportamiento del sistema bajo cargas muy altas con el objetivo de romper el sistema y encontrar los límites del sistema.

Testeo de integración

Testeo de los interfaces de y la interacción entre las unidades previamente testeadas mientras se ensambla el sistema entero. Hay varias estrategias para determinar el orden en qué vamos a testear las interfaces: top-down, bottom-up, big-bang, etc.

Testeo de localización

Testear en un sistema de software la capacidad de estar configurado con parámetros de localidad, por ejemplo: diferentes lenguajes, diferentes conjuntos de caracteres (p.ej. ñ), diferencias en zona de hora, diferencias en formato de hora y fechas (p.ej. 2pm, 9-30-2004), diferentes teclados, tamaño de papel (A4, etc.).

Testeo de regresión

Testeo que se necesita después de hacer cambios en el software para asegurar que no se ha introducido defectos después de corregir un error, después de añadir más funcionalidades o durante el desarrollo iterativo.

Testeo de rendimiento y carga

Testeo basado en los requisitos de rendimiento; por ejemplo utilización de memoria, tiempo de respuesta (lapso de tiempo que transcurre entre que un usuario hace una petición y que la respuesta es recibida por este) o throughput (cantidad de transacciones procesadas por periodo de tiempo).

Testeo de seguridad

Testear de un sistema de software la capacidad de prevenir acceso no autorizado.

Testeo de sistema

Testeo del sistema entero con el objetivo de encontrar defectos que no tienen su origen en la interacción de componentes. Puede incluir testeo de funcionalidad, rendimiento y carga, estrés, configuración, seguridad, instalabilidad, localización, usabilidad.

Testeo de software

Proceso, la acción y el efecto de testear software.

Testeo unitario

Testeo de unidades o componentes de software individuales.

Tipo de componente:

Tipo de entregable de una metodología

Tolerancia a fallos

Subcaracterística de fiabilidad, que indica el grado en que el sistema mantiene un nivel de respuesta ante fallos del sistema o interfaces.

TPI

Modelo que proporciona una idea general de la madurez del proceso de testeo en una organización, a partir de ahí se establecen unos pasos de mejora graduales y controlados. TPI® y TMAP® son marcas registradas por y pertenecientes a SOGETI en España y otros países.

UML

Unified Modeling Language. Es el diseño de aplicaciones basada en componentes.

Usabilidad

Capacidad de un software de ser comprendido, aprendido, usado y ser atractivo para el usuario, en condiciones específicas de uso.

Efectividad, eficiencia y satisfacción con la que un producto permite alcanzar objetivos específicos a usuarios específicos en un contexto de uso específico.

Utilización de recursos

Subcaracterística de eficiencia, que indica las características del software que influyen en el número de recursos usados, y la duración de su uso, cuando se lleva a cabo su función.

Validación

La validación es realizada normalmente sobre el producto final bajo condiciones operacionales definidas. Puede ser necesaria en las fases iniciales. “Validado” es utilizado para designar el estado correspondiente.

Verificación

Comprobación de que se está construyendo el producto correctamente.



ANEXOS

ANEXO 1 Plantillas

ANEXO 2 Marco lógico



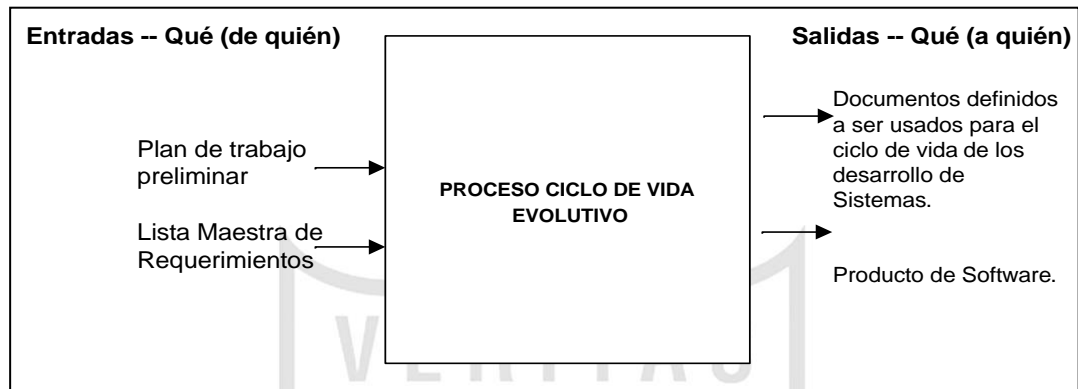
ANEXO 1. Plantillas

Diagrama de contexto y flujogramas
Desarrollo de los artefactos
Acta de reunión
Lista maestra de requerimiento
Cartilla proceso mantenimiento
Documento de preanálisis
Documento de análisis
Lista de incidencias
Matriz de trazabilidad de requerimiento a documentar para mantenimiento
Matriz de trazabilidad de requerimiento a objetos
Documento plantilla de documentos de aceptación de pruebas funcionales
Documento plantilla de pase a producción
Documento plantilla de prueba de sistemas
Documento casos de pruebas
Control de calidad del pase a producción
Checklist de mantenimiento
Checklist analista / programador
Informe de pruebas
Plantilla de documento de especificación de ambiente
Manual de usuario
Manual de sistemas
Manual de administración e instalación

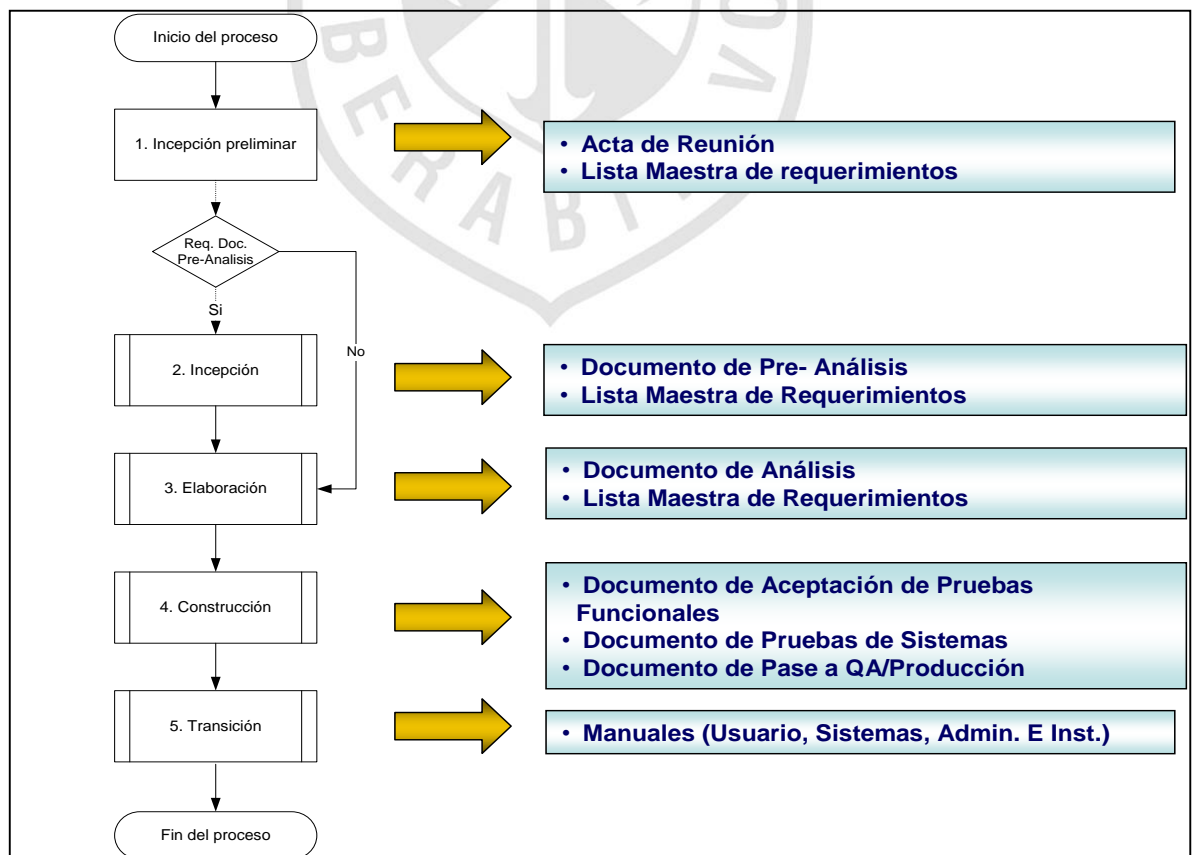
DISEÑO DE UNA METODOLOGÍA DE CERTIFICACIÓN DE PRODUCTOS SOFTWARE ORIENTADO AL SECTOR PÚBLICO

Proceso de ingeniería: Entregables

Diagrama de Contexto Del Proceso





Desarrollo del Proceso – Flujoograma de Proceso



Desarrollo de los artefactos

- BN.F.1. Plantilla de acta de reunión
- BN.F.2. Lista maestra de requerimientos
- BN.F.3 Cartilla mantenimiento
- BN.F.4 Plantilla de documento de pre-análisis
- BN.F.5 Plantilla documento de análisis
- BN.F.6 Lista de Incidencias
- BN.F.7 Matriz de trazabilidad de requerimientos a documentos para mantenimiento
- BN.F.8 Matriz de trazabilidad de requerimientos a objetos
- BN.F.9 Plantilla de documentos de aceptación de pruebas funcionales
- BN.F.10 Plantilla prueba de sistemas
- BN.F.11 Plantilla de documento de pase a QA/producción
- BN.F.12 Plantilla de documento casos de pruebas
- BN.F.13 Control de calidad del pase a producción. (Checklist)
- BN.F.14 Checklist mantenimiento
- BN.F.15. Checklist analista / programador
- BN.F.16 Plantilla de informe de pruebas
- BN.F.17 Plantilla de documento de especificacion de ambientes
- BN.F.18 Plantilla de manual de usuario
- BN.F.19 Plantilla de manual sistemas
- BN.F.20 Plantilla de manual de administración e instalaciones

	BN.F.1 Acta de Reunión [Nombre del Sistema]	
---	--	---

Número De Acta: **AR-{Iniciales del Sistema}-{dd-mm-aa}**

Sistema	<Nombre del Sistema>	Fecha	
Usuario	BN	Hora Inicio	
Lugar	BN – <Lugar de la Reunión>	Hora Término	



Asistentes	
Sistemas	Usuario

Agenda
1.
2.
3.
4.
5.
6.

Acuerdos	
Descripcion	Responsable

Por sistemas



Por usuario

	BN.F.2 Lista Maestra de Requerimientos [Nombre del Sistema]	
LMR-{Iniciales del Sistema}-{dd-mm-aa}		

Id. Requerimiento Sistema	{XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX}
Usuario	{XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX}
Descripción de requerimiento	{XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX}
Tipo requerimiento de sistema	<input type="radio"/> Proceso <input type="radio"/> Interactivo <input type="radio"/> Reporte
Es interfaz	<input type="radio"/> No es interfaz <input type="radio"/> Interfaz externa <input type="radio"/> Interfaz interna
Tipo de desarrollo	<input type="radio"/> Programa nuevo <input type="radio"/> Programa modificado
Afecta arquitectura	<input type="radio"/> Si <input type="radio"/> No
Dificultad	<input type="radio"/> Alta <input type="radio"/> Media <input type="radio"/> Baja
Prioridad	<input type="radio"/> Crítica <input type="radio"/> Ventajosa <input type="radio"/> Posible <input type="radio"/> Baja
Estado	<input type="radio"/> Propuesto <input type="radio"/> Autorizado <input type="radio"/> En implementación <input type="radio"/> En pruebas <input type="radio"/> En producción
Juicio Experto - Tiempo en días de construcción	{XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX}

Por sistemas

Por usuario

	BN.F.3 Cartilla Proceso Mantenimiento [Nombre del sistema]	
CPM-{Iniciales del Sistema}- {dd-mm-aa}		
1. Historial de las Revisiones	{XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX}	
2. Objetivo y Alcance de la Cartilla	{XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX}	
3. PR's Urgentes: Criterios para La Atención de PR's Urgentes	<ul style="list-style-type: none"> ○ Todos los PR's urgentes NO REQUIEREN de una tipificación especial para ser atendidos (no existen PR's "Muy Urgentes" ni PR's "Urgentes Normales"). ○ La aprobación de entregables del PR urgente será efectuada por el medio de aprobación que sea el más inmediato y necesario dada la urgencia del PR. ○ Los siguientes son los artefactos que deberán trabajarse como mínimo para la atención del PR urgente: <ul style="list-style-type: none"> ◆ Documento de análisis ◆ Documento de Pase a QA/Producción. ◆ Checklist AP ◆ Checklist AS ◆ Checklist QA ○ El plazo de regularización de actividades y artefactos del PR urgente tendrá un tiempo máximo de 48 horas. Contándose el tiempo luego del pase a producción del PR. La regularización del PR incluye todas las actividades y artefactos definidos, dejando el PR en el estado "TERMINADO". ○ Una vez culminada la regularización de los artefactos, el AS deberá comunicar al AC para la revisión respectiva. Será responsabilidad del AC efectuar el seguimiento respectivo. ○ Todas las actividades del PR urgente deberán agregarse en el cronograma del actual ciclo de Producción, especificándose para ello el avance real. 	

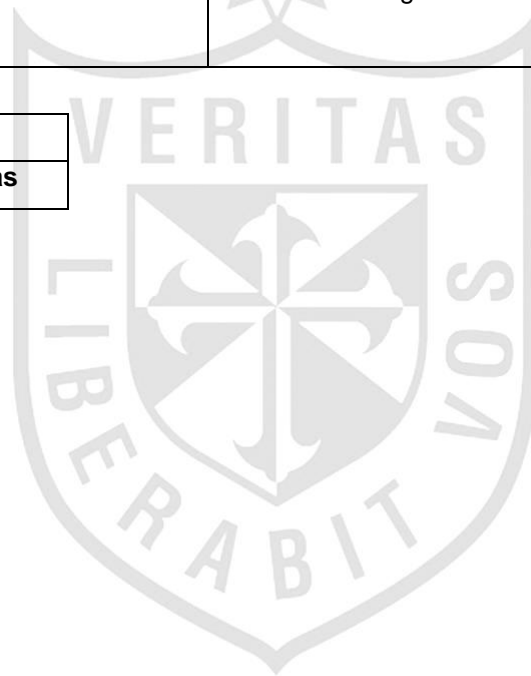
<p>4. Cronograma Criterios Para La Elaboración Del Cronograma De Mantenimiento</p>	<ul style="list-style-type: none"> ○ Las actividades de las Secciones “Seguimiento del Mantenimiento” e “Inicio del mantenimiento Ciclo Siguiente”, deberán programarse en base a una mayor prioridad con el objetivo de no dividir las actividades del proceso de Ingeniería. ○ En caso sea necesario efectuar un ajuste al Documento de Análisis, estando el PR en la etapa de Construcción, en esta etapa deberá considerarse las siguientes actividades: <ul style="list-style-type: none"> ◆ Ajuste del Documento de Análisis ◆ Revisión y Aprobación de Documento de Análisis – CS <p>Estas actividades deberán considerarse después de la actividad “Pruebas Unitarias”.</p> ○ Si durante el ciclo de producción ingresa un PR Urgente, se deberá considerar la actividad Reprogramación del Cronograma y las actividades que son parte de la estimación de tiempos. Estas actividades deberán considerarse como parte del Seguimiento de Mantenimiento, dentro de la semana correspondiente, de la siguiente manera: <ul style="list-style-type: none"> ◆ Seguimiento del mantenimiento (8) <ul style="list-style-type: none"> ▪ Semana 1 ▪ Semana 2 ▪ Semana 3 ▪ Semana 4 <ul style="list-style-type: none"> ➤ Asignación de Trabajo ➤ Reuniones ➤ Reprogramación de Cronograma <ul style="list-style-type: none"> ❖ Elaborar lista maestra de requerimiento ❖ Estimar requerimientos ➤ Envío del Cronograma
--	--



<p>5. Cronograma Criterios Para La Elaboración Del Cronograma De Mantenimiento</p>	<ul style="list-style-type: none"> ○ Las actividades de las Secciones “Seguimiento del Mantenimiento” e “Inicio del mantenimiento Ciclo Siguiete”, deberán programarse en base a una mayor prioridad con el objetivo de no dividir las actividades del proceso de Ingeniería. ○ En caso sea necesario efectuar un ajuste al Documento de Análisis, estando el PR en la etapa de Construcción, en esta etapa deberá considerarse las siguientes actividades: <ul style="list-style-type: none"> ◆ Ajuste del Documento de Análisis ◆ Revisión y Aprobación de Documento de Análisis – CS <p>Estas actividades deberán considerarse después de la actividad “Pruebas Unitarias”.</p> ○ Si durante el ciclo de producción ingresa un PR Urgente, se deberá considerar la actividad Reprogramación del Cronograma y las actividades que son parte de la estimación de tiempos. Estas actividades deberán considerarse como parte del Seguimiento de Mantenimiento, dentro de la semana correspondiente, de la siguiente manera: <ul style="list-style-type: none"> ◆ Seguimiento del mantenimiento (8) <ul style="list-style-type: none"> ▪ Semana 1 ▪ Semana 2 ▪ Semana 3 ▪ Semana 4 <ul style="list-style-type: none"> ➤ Asignación de Trabajo ➤ Reuniones ➤ Reprogramación de Cronograma <ul style="list-style-type: none"> ❖ Elaborar lista maestra de requerimiento ❖ Estimar requerimientos ➤ Envío del Cronograma
<p>6. Pre-Análisis</p> <p>Criterios para la Elaboración del Documento de Pre-Análisis</p>	<p>Se elaborará un documento de Pre - Análisis sólo en los siguientes casos:</p> <ul style="list-style-type: none"> ○ La definición del requerimiento no es clara. ○ La complejidad del Requerimiento es Alta ○ El Tipo de Requerimiento es: <ul style="list-style-type: none"> ◆ Creación (Complejidad Alta)

	<ul style="list-style-type: none"> ◆ Modificación (Complejidad Alta) ◆ Optimización (Complejidad Alta) ◆ Procesamiento de Datos (Complejidad Alta) <p>No aplica efectuar el documento de Pre-Análisis en los siguientes casos:</p> <ul style="list-style-type: none"> ○ El tipo de PR es de complejidad Mediana/Baja ○ El Tipo de Requerimiento es: <ul style="list-style-type: none"> ◆ Creación (Complejidad Mediana/Baja) ◆ Modificación(Complejidad Mediana/Baja) ◆ Optimización(Complejidad Mediana/Baja) ◆ Procesamiento de Data (Complejidad Baja/ Mediana) ◆ Error ○ El PR es urgente
--	---

Por sistemas

Por usuario





	BN.F.4 Documento de Pre-Análisis [Nombre del Sistema]	
DPreA-{Iniciales del Sistema}-{dd-mm-aa}		

PR	{Nro. De PR}
Tipo de PR	<p>1) Creación (C) Son los requerimientos que adicionan una nueva funcionalidad usuaria al Sistema.</p> <p>2) Modificación (M) Son requerimientos que buscan cambiar una funcionalidad que posee el Sistema.</p> <p>3) Optimización (O) Son los requerimientos de carácter técnico que buscan mejorar el rendimiento del Sistema.</p> <p>4) Errores (E) Son los requerimientos que buscan corregir un error de programación o un error que impide la correcta operación del Sistema en ambiente de producción.</p> <p>5) Procesamiento de Datos (P) Son los requerimientos que buscan adicionar, actualizar o eliminar registros contenidos en la Base de Datos y que no pueden realizarse por el Sistema o cuyo procesamiento por el Sistema implicaría invertir demasiado tiempo.</p>
Fecha de Versión	{DD-MM-AA}
Versión	{XX.X}
1. Descripción detallada de Requerimiento	{Esta etapa consiste en establecer el alcance del requerimiento, detallar el requerimiento, realizar las definiciones iniciales, y evaluar el impacto en el(os) aplicativo(s) y proceso(s) involucrado(s). Es la versión primaria del análisis funcional.}
2. Situación Actual	{Describir la situación actual bajo cuyo marco se origina el requerimiento}
3. Alcance	{Acotar claramente al alcance del PR} 3.1. Producto(s) final(es) del requerimiento 3.2. Otros sistemas Impactados {Indicar que otros módulos y/o sistemas se podrían ver

	impactados con el desarrollo del PR }	
4. Marco Conceptual	{XX}	
5. Especificaciones Funcionales	{XX}	
5.1. Alternativas De Solución	{XX}	
5.2. Evaluación De Alternativas	{XX}	
6. Anexos de Referencia Relacionados con el PR	{XX}	
7. Tiempos Estimados para la Atención del Requerimiento	Actividad	Horas
	INCEPCIÓN	
	• Reunión de Pre-Análisis	
	• Elaboración y Aprobación del Documento de Análisis	
	ELABORACIÓN	
	• Elaboración y Aprobación del Documento de Análisis	
	CONSTRUCCIÓN	
	• Desarrollo de la Solución	
	• Pruebas Internas	
	• Control de Calidad Interno	
	• Pase a QA	
	• Pruebas funcionales y sistemas.	
	TRANSICIÓN	
• Pase a Producción		
Total		

Por sistemas

Por usuario

	BN.F.5 Documento de Análisis [Nombre del Sistema]	
DA-{Iniciales del Sistema}-{dd-mm-aa}		

PR	{Nro. De PR}
Tipo de PR	<ol style="list-style-type: none"> 1) Creación (C) Son los requerimientos que adicionan una nueva funcionalidad usuaria al Sistema. 2) Modificación (M) Son requerimientos que buscan cambiar una funcionalidad que posee el Sistema. 3) Optimización (O) Son los requerimientos de carácter técnico que buscan mejorar el rendimiento del Sistema. 4) Errores (E) Son los requerimientos que buscan corregir un error de programación o un error que impide la correcta operación del Sistema en ambiente de producción. 5) Procesamiento de Datos (P) Son los requerimientos que buscan adicionar, actualizar o eliminar registros contenidos en la Base de Datos y que no pueden realizarse por el Sistema o cuyo procesamiento por el Sistema implicaría invertir demasiado tiempo.
Fecha de Versión	{DD-MM-AA}
Versión	{XX.X}

1. Descripción detallada de Requerimiento	{Esta etapa consiste en determinar la Solución al requerimiento (satisfacción de la necesidad usuaria). Contiene el Análisis Funcional enriquecido (basado en el Preanálisis) y detalla el Análisis Técnico de la Solución. Asimismo implica la revisión de ambos aspectos del Análisis}
2. Situación Actual	{Describir la situación actual bajo cuyo marco se origina el requerimiento}
3. Alcance	{Acotar claramente al alcance del PR}



	<p>3.1. Producto(s) final(es) del requerimiento</p> <p>3.2. Otros sistemas Impactados</p> <p>{Indicar que otros módulos y/o sistemas se podrían ver impactados con el desarrollo del PR }</p>
<p>4. Especificaciones funcionales (a detalle)</p>	<p>{Describir con detalle el(os) producto(s) que se obtendrá(n). Deberá contener por lo menos: }</p> <p>4.1. <u>Definiciones detalladas</u></p> <ul style="list-style-type: none"> • Definiciones • Conceptos • Criterios • Restricciones • Consistencias operativas, etc. <p>4.2. <u>Características detalladas del Producto resultante de la atención del requerimiento</u></p> <ul style="list-style-type: none"> • Pantallas • Reportes • Accesos y perfiles de usuario, etc. <p>4.3. <u>Casos de Análisis</u></p> <p>{Casuística obtenida y/o creada para efectos del PR}</p> <p>4.4. <u>Impacto en la Operativa</u></p> <p>{Describir cuál va ser el impacto en la operativa (en procesos y/o procedimientos), de haberlo}</p> <p>4.5. <u>Documentos de Referencia relacionados con el PR</u></p> <p>{Información escrita proporcionada por los interesados, o referencia a información necesaria para el adecuado desarrollo del PR (normas legales, procedimientos operativos, etc.) }</p> <p>4.6. <u>Análisis Funcional</u></p> <p>{Es esta parte se deberá describir la solución planteada (el qué y el cómo) en términos entendibles para el usuario Operativo}</p>
<p>5. Especificaciones Técnicas (a detalle):</p>	<p>5.1. <u>Requerimientos relacionados</u></p> <p>{Identificar similitudes y/o relaciones con otros PR's registrados y establecer riesgos y/o ventajas de desarrollarlos en secuencia, en paralelo o integrados en un mismo desarrollo }</p> <p>5.2. <u>Análisis Técnico [cuerpo del análisis]</u></p> <p>5.2.1. <u>Especificaciones técnicas detalladas</u></p>

	<p>{Según el PR, asumir algunas o todas las partes indicadas a continuación}</p> <p>5.2.2. <u>Descripción del Desarrollo de Requerimiento:</u></p> <ul style="list-style-type: none"> • Detalle de la solución planteada (Lógica creada o modificada) • Alcance del Desarrollo (qué objetos, parámetros, procesos, programas afecta) • Dependencias <p>5.2.3. <u>Objetos de Base de Datos (Tablas, Vistas, Funciones, Procedimientos, Índices, etc.)</u></p> <p>A. <u>Tablas / vistas:</u></p> <ul style="list-style-type: none"> • Definición de la(s) tabla / vista(s) a crear / modificar con su respectiva estructura. • Detallar todas las tablas a crear con su respectiva estructura y llave. • Si es necesario añadir algunas observaciones. <p>B. <u>Funciones y Procedimientos</u></p> <ul style="list-style-type: none"> • Definición de Argumentos y valor(es) de retorno • Tablas afectadas. <p>C. <u>Índices</u></p> <ul style="list-style-type: none"> • Definición del Índice, • Tablas asociadas • Orden de columnas <p>5.2.4. <u>Creación y/o Modificación de Objetos en el Aplicativo</u></p> <ul style="list-style-type: none"> • Definición de Objetos: [Formularios, Reportes, Parámetros, clases, librerías, conexiones, etc.] • Módulo: [Nombre del Módulo] • Opciones final:[Árbol de Opciones como quedarán finalmente tras la
--	--

	<p>creación modificación]</p> <p>5.2.5. <u>Otros</u> Cualquier otro aspecto técnico no contenido en los puntos anteriores.</p> <p>5.3. <u>Impacto en otros Sistemas</u> 5.3.1. Sistemas Afectados 5.3.2. Acciones a tomar</p> <p>5.4. <u>Tiempos estimados para la atención de requerimiento</u> 5.4.1. Complejidad del PR</p>
6. Capacitación (SI) / (NO)	{Indicar si se requiera capacitación sobre el uso del producto resultante del PR}
7. Documentación	{Indicar que manuales se tendrán que actualizar como resultado del desarrollo del PR}

Por sistemas



Por usuario

	BN.F.6 Lista de Incidencias [Nombre del Sistema]	
LI-{Iniciales del Sistema}-{dd-mm-aa}		

PR	{Nro. De PR}
Ciclo / Proyecto	Mantenimiento / Desarrollo
Tipo Incidencia	Puede ser: <input type="radio"/> Software <input type="radio"/> Documental
Fase	<input type="radio"/> Inicio <input type="radio"/> Elaboración <input type="radio"/> Construcción <input type="radio"/> Transición
Origen	<input type="radio"/> Calidad <input type="radio"/> Internas <input type="radio"/> Aceptación - Funcionales
Línea	{XXXXXXXXXXXXXXXXXXXXXXXXXXXX}
Sistema	{XXXXXXXXXXXXXXXXXXXXXXXXXXXX}
Ubicación	{XXXXXXXXXXXXXXXXXXXXXXXXXXXX}
Fecha registro	{dd –mm - aaaa}
Clasificación	<input type="radio"/> Código, Forma, Programa, Funcional, Redacción, Contenido
Tipo	<input type="radio"/> Error <input type="radio"/> Aclaración
Estado	<input type="radio"/> Pendiente <input type="radio"/> Proceso <input type="radio"/> Levantada
Caso de prueba / Entregable	{XXXXXXXXXXXXXXXXXXXXXXXXXXXX}
Descripción	{XXXXXXXXXXXXXXXXXXXXXXXXXXXX}
Fecha comprometida	{dd –mm - aaaa}
Fecha real	{dd –mm - aaaa}
Responsable	{XXXXXXXXXXXXXXXXXXXXXXXXXXXX}
# Veces	{XXXXXXXXXXXXXXXXXXXXXXXXXXXX}

Por analista

Por jefe sistemas

	BN.F.7 Matriz de trazabilidad de requerimientos a documentos para mantenimiento [Nombre del Sistema]	
MTDOC-{Iniciales del Sistema}-{dd-mm-aa}		



PR	{Nro. De PR}
Id. Requerimiento de usuario	{XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX}
Descripción de requerimiento de usuario	{XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX}
Id. Requerimiento de Sistema	{XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX}
Descripción de requerimiento de sistema o use case	{XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX}
Id. escenario	{XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX}
Inicio preliminar	
¿Se trató en reunión de pre-análisis?	{XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX}
Nombre del documento de resultado de reunión de pre-análisis	{XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX}
Registrado por	{XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX}
Inicio	
Nombre del documento de pre-análisis	{XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX}
Registrado por	{XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX}
Elaboracion	
Nombre del documento de análisis	{XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX}
Registrado por	{XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX}
Contruccion	
Implementado (Si, No)	{XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX}
Documento con el que se realizó Pruebas Internas	
Registrado por	
Documento con el que se realizó Pruebas de Calidad	

Registrado por	
Documento con el que se realizó Pruebas de Aceptación	
Registrado por	

Por analista

Por jefe de sistema





	BN.F.8 Matriz de de trazabilidad de requerimientos a objetos [Nombre del Sistema]	
MTRO-{Iniciales del Sistema}-{dd-mm-aa}		

PR	{Nro. De PR}	REQM	
Módulo	[Representa el módulo del sistema. Usado para organizar el sistema en módulo]		
Paquete de objeto	[Nombre del Package el cual contendra el objeto mencionado en la matriz de trazabilidad]		
Nombre de objeto	[Nombre del objeto considerado para el inventario, dicho nombre sera el que posea en el repositorio de fuentes]		
Complejidad	[Nivel de complejidad del objeto. Podra ser Alto , Medio o Bajo segun el objeto a crear o modificar].		
Lenguaje	[Ver cuadro Lenguaje de Programación de la "Hoja Parámetros" del presente artefacto].		
Tipo de objeto	[Tipo de Objeto, según cuadro referenciado en la "Hoja Parámetros" del presente artefacto, podrás ser Objeto Java u Objeto PowerBuilder].		
Capa	[Nivel en la cual se emplea el objeto del Inventario, podra ser Base de datos, Servidor, etc.].		
Observaciones	[Observación a considerar, si se cree conveniente, con respecto al objeto listado].		
Ruta en repositorio	[Ruta desde la cual se obtuvo el objeto mencionado en el repositorio de fuente]		

Por analista



Por jefe de sistema

	BN.F.9 Documento Plantilla de documentos de aceptación de pruebas funcionales	
[Nombre del Sistema]		
PDAPF-{Iniciales del Sistema}-{dd-mm-aa}		

PR		{NRO. DE PR}			
1. Descripción		{Requerimiento descrito en el PR}			
2. Calendario de pruebas					
Fecha	Hora inicio programada	Hora fin programada	Hora inicio real	Hora de fin real	Motivo de retraso
3. Calendario de pruebas					
Fecha	Casos de Prueba	Culminado Satisfactoriamente	Culminado con Problemas	No Culminado	
4. Comentarios / Ocurrencias					
{Se anotan las observaciones: errores, aspectos a ajustar, pequeños añadidos al alcance, etc.}				[Nombre de la persona que observa]	
5. Documentos Anexos					
{Se anotarán los reportes o pantallas resultantes de las pruebas y se adjuntaran los impresos resultantes}					
6. Requerimientos Complementarios					
{cuando como consecuencia del PR se deriva otra necesidad de usuario que debe atenderse con un nuevo PR o cuando existen PR's ya registrados que complementen el probado se deberá indicar en este recuadro}					
7. Nombres y Cargos de las personas que realizaron las Pruebas Funcionales					

Por sistemas

Por usuario

	BN.F.10 Documento Plantilla de Pase a producción [Nombre del Sistema]	
PPP-{Iniciales del Sistema}-{dd-mm-aa}		

PR	{NRO. DE PR}
1. Descripción	{Requerimiento descrito en el PR}

2. Módulo – Opción	[Nombre de Modulo – Nombre de Opción o del Régimen según sea el caso]
3. Descripción	{Requerimiento descrito en el PR}

4. Librerías y Objetos	{Nombre de las Librerías y Objetos afectados, indicando el Tipo de Programa (Java, Cobol, Fox Pro, etc.), Bases de Datos y/o archivos (que también pueden ser sujetos o modificaciones)}
------------------------	---

Nombre	Tipo	Origen	Destino	Fecha
[Nombre Objeto, Librería, Programa, Base de Datos y/o Archivo]	[Referencia: Objeto. Librería, Programa, Base de Datos y/o Archivo]	[Ruta de Destino del Aplicativo]	[Ruta del Destino del Aplicativo]	[Manejo de la versión]



5. Programa	{Nombre y ruta de los programas que lo invocan}
-------------	--

6. Directorio	{Indicar las rutas donde se encuentran las librerías}
---------------	--

7. Usuarios que tiene acceso	{ [Nombres Y Apellidos] – [Cargo] – [Nivel de Acceso] (cuando el requerimiento es nuevo) }
------------------------------	---

Por sistemas



Por usuario

	BN.F.11 Documento Plantilla de Prueba de sistemas [Nombre del Sistema]	
PPS-{Iniciales del Sistema}-{dd-mm-aa}		

PR	{NRO. DE PR}
1. Introducción	[La introducción brinda una vista rápida de todo el documento. Incluye el objetivo, alcance, definiciones, abreviaturas utilizadas en el documento]
2. Objetivo	[Indicar el objetivo del documento]
3. Alcance	[Una breve descripción del alcance de este documento, qué otro(s) sistema(s) están asociados o se ven afectados por este documento]
4. Definiciones y abreviaturas	[Esta sección brinda la definición de aquellos términos y abreviaciones requeridas para interpretar adecuadamente el contenido de este documento. Esta información debería ser provista en un glosario del documento y aquí únicamente hacer referencia a dicho documento]
5. Procesos de prueba	{Pasos a seguir para el seguimiento evaluación y otras operaciones que sean necesarias}
6. Material y equipo	{Material e información necesaria para realización de las pruebas}

Por sistemas

Por usuario

	BN.F.12 Documento Casos de Pruebas [Nombre del Sistema]	
PPS-{Iniciales del Sistema}-{dd-mm-aa}		
PR	{NRO. DE PR}	
1. Introducción	[La introducción brinda una vista rápida de todo el documento. Incluye el objetivo, alcance, definiciones, abreviaturas utilizadas en el documento]	
2. Objetivo	[Indicar el objetivo del documento]	
3. Alcance	[Una breve descripción del alcance de este documento, qué otro(s) sistema(s) están asociados o se ven afectados por este documento]	
4. Definiciones y abreviaturas	[Esta sección brinda la definición de aquellos términos y abreviaciones requeridas para interpretar adecuadamente el contenido de este documento. Esta información debería ser provista en un glosario del documento y aquí únicamente hacer referencia a dicho documento]	
5. Referencia	[pasos los casos de uso a probar se encuentran definidos en el plan de iteraciones, y la definición de los mismos en los documentos de análisis]	
6. Resumen	[Este documento contiene las siguientes secciones: <ul style="list-style-type: none"> ▪ Casos de Prueba ▪ Pruebas de ciclo completo (Flujos de trabajo)] 	
7. Casos de Pruebas		
Código de caso de prueba	Caso de Uso	Escenario
{Código del Caso de Prueba} [Compuesto por la concatenación del código del caso de uso y código del escenario Ejemplo: CUS001E01]	{Código del Caso de uso: Nombre del Caso de Uso} [Ejemplo: CUS001: Buscar Documento.]	{Código del Escenario: Nombre del Escenario} [Ejemplo: E01: Búsqueda de Documentos pertenecientes a la UO del Usuario.]

8. Detalle de los casos de pruebas

8.1. {Código del caso de Prueba - Código del caso de Uso: Nombre del caso de Uso. Código de Escenario: Nombre del Escenario}

[Ejemplo:

CUS001E01 – CUS001: Buscar Documento. E01: Búsqueda de Documentos pertenecientes a la UO del Usuario.]

8.1.1. Criterios

[Criterios que definen el escenario]

1. Criterio de definición de escenario 1.
2. Criterio de definición de escenario 2.
3. ...

8.1.2. Flujo de actividades

[Flujo de actividades, indicando a que flujo (básico o alternativo) pertenece]

9. Detalle de los casos de pruebas

9.1. {Código del caso de Prueba - Código del caso de Uso: Nombre del caso de Uso. Código de Escenario: Nombre del Escenario}

[Ejemplo:

CUS001E01 – CUS001: Buscar Documento. E01: Búsqueda de Documentos pertenecientes a la UO del Usuario.]

9.1.1. Criterios

[Criterios que definen el escenario]

4. Criterio de definición de escenario 1.
5. Criterio de definición de escenario 2.
6. ...

9.1.2. Flujo de actividades

[Flujo de actividades, indicando a que flujo (básico o alternativo) pertenece]

Paso	Instrucción	Resultados Esperados
1	Instrucción 1	Resultado 1
2	Instrucción 2	Resultado 2
3	Instrucción 3	Resultado 3

9.1.3.Puntos de control

[Puntos de control del conjunto de pruebas, es decir, donde el flujo puede variar]

1. {Punto de control 1}.
2. {Punto de control 1}.
3. ...



9.1.4.Puntos de revisión

[Puntos de revisión del conjunto de pruebas, donde se debe validar lo indicado]

Id	Punto de Control	Validaciones a realizar.
1	Punto de revisión 1	- Validación 1
2	Punto de revisión 2	- Validación 1 - Validación 2
3	Punto de revisión 3	- Validación 3

Por sistemas

Por usuario

		BN.F.13 Control De Calidad Del Pase A Producción. (Checklist)			
[Nombre del Sistema]					
Checklist -{Iniciales del Sistema}-{dd-mm-aa}					
PR			{NRO. DE PR}		
1. Descripción			{Requerimiento descrito en el PR}		
Nº	Tema	ROL	Acción	BN	
				Estado	
				¿Se efectúa la acción?	
				SI	NO
1	Base de Datos	DBA	DML: cambio en información de Base de Datos (select, insert, update, delete)	Otorga Conformidad de Cambio DML	Otorga Conformidad de que no existe un cambio DML
			DDL: cambio en estructura de Base de Datos (create, revoke, grant, alter)	Otorga Conformidad de Cambio DDL	Otorga Conformidad de que no existe un cambio DDL
			DCL: control de privilegios en Base de Datos	Otorga Conformidad de Cambio DCL	Otorga Conformidad de que no existe un cambio DCL
2	Seguridad de Base de Datos	DBA	Solicitud de Creación o Modificación de Usuarios en Base de Datos	Otorga Conformidad de Solicitud de Creación o Modificación de Usuario	Otorga Conformidad que no se da Creación o Modificación de Usuario
3	Seguridad de la Aplicación	Oficial de Seguridad	Creación o modificación de usuario en la aplicación	Otorga Conformidad de Creación o Modificación de Usuario	Otorga Conformidad de No Creación o No Modificación de Usuario

Por sistemas

Por usuario



**BN.F.14 Checklist Mantenimiento****[Nombre del Sistema]**

Checklist M -{Iniciales del Sistema}-{dd-mm-aa}



REF.	PUNTOS DE CONTROL	SI	N O	NA	NC	OBSERVACIONES / EVIDENCIAS / JUSTIFICACION
1	DOCUMENTO DE PRE-ANALISIS					
1.1	¿Se elaboró el documento de Pre-Análisis?					
1.2	¿El PR que implica: Creación, Modificación, Optimización o Procesamiento de datos?					
1.3	¿Se ha detallado la sección: Situación Actual?					
1.4	Dentro de la sección Alcance se han detallado los siguientes ítems:					
1.5	1.5.1 Detalle del Alcance					
	1.5.2 Exclusiones					
1.6	¿Se ha detallado la sección: Marco Conceptual?					
1.7	¿Se ha detallado la sección: Anexos?					
1.8	¿Se ha detallado la sección: Tiempos Estimados para la atención del requerimiento?					
1.9	¿Se firmó última versión de documento de Pre-Análisis? Indicar N° versión firmada					
2	DOCUMENTO DE ANALISIS					
2.1	¿Se ha detallado la sección: Situación Actual?					
2.2	Dentro de la sección Alcance se han especificado los siguientes ítems:					
2.2.1	Detalle del alcance					
2.2.2	Otros sistemas impactados					
2.2.3	Requerimientos relacionados					
2.2.4	Exclusiones					
2.3	¿Se han descrito todos los conceptos necesarios dentro de la sección: Marco					

	Conceptual?					
2.4	¿Se ha detallado la sección: Especificaciones Funcionales?					
2.5	¿Se han especificado los Anexos de Referencia Relacionados con el PR?					
2.6	Dentro de la sección Análisis de la Solución se han descrito los siguientes ítems:					
2.6.1	Detalle de la solución					
2.6.2	Impacto en la operativa					
2.6.3	Accesos					
2.7	Indique si se han detallado los siguientes ítems de la sección: Especificaciones Técnicas					
2.7.1	Especificaciones técnicas detalladas					
2.7.1.1	Descripción del desarrollo del requerimiento					
2.7.1.2	Objetos de Aplicación					
2.7.1.3	Objetos de base de datos					
2.7.2	Tiempos estimados para la atención del requerimiento					
2.7.2	Complejidad del PR					
2.8	¿Se ha especificado si se requerirá o no Capacitación al usuario?					
2.9	¿Se ha indicado si habrá que documentar algún manual del sistema?					
2.1	¿Se aprobó la última versión del documento de Análisis? Indicar N° versión aprobada					
3	DOCUMENTO DE PRUEBAS FUNCIONALES					
	Para la elaboración y definición de los casos de pruebas:					
3.1	¿Ha utilizado el mismo formato del documento de aceptación de las pruebas funcionales para elaborar los casos de pruebas?					
3.2	¿Se elaboraron los casos de prueba, por cada escenario presentado en el documento de Análisis, durante la					

	etapa de implementación?					
3.3	¿Se elaboraron los casos de prueba necesarios?					
3.4	¿Se solicitó la aprobación de los casos de prueba al UA y/o GRS, antes de las pruebas internas?					
3.5	¿El Usuario aprobó los casos de prueba, antes de las pruebas internas?					
	Para la preparación y realización de las pruebas Funcionales:					
3.6	¿Se programaron con anticipación las pruebas funcionales?					
3.7	¿Las pruebas funcionales se llevaron a cabo en la hora programada? Indicar motivo de retraso					
3.8	¿Las pruebas funcionales se llevaron a cabo de forma correcta?					
3.9	¿Se generaron comentarios / ocurrencias como resultado de las pruebas funcionales?					
3.10	¿Se confirmó la capacitación?					
3.11	¿Se requiere cartilla de usuarios?					
3.12	¿Se generaron documentos anexos? Indicar					
3.13	¿Son necesarios requerimientos complementarios?					
4	DOCUMENTO DE PASE A QA/PRODUCCION					
	Respecto de los Objetos para el Pase a Producción:					
4.1	¿Se actualizó la lista de iteraciones?					
4.2	¿Se especifican los requisitos necesarios?					
4.3	¿Se indicaron los objetos compilados?					
4.4	¿Se indicaron los objetos fuentes?					
4.5	¿Se indicaron los scripts de base de datos?					
4.6	¿Es necesario realizar algún registro de programas en el sistema?					

	Respecto del Pase a Producción:					
4.7	¿Se especifican los requisitos necesarios?					
4.8	¿Se indico el procedimiento a seguir?					
4.9	¿Se especifico si es necesaria alguna configuración en el cliente?					
4.10	¿Se indicó la iteración a ejecutar?					



	BN.F.16 Informe de Pruebas [Nombre del Sistema]	
Informe de pruebas -{Iniciales del Sistema}-{dd-mm-aa}		

PR	{NRO. DE PR}
Resumen Ejecutivo	{Explicar en una hoja todo el plan en resumen}

1. Introducción	[La introducción brinda una vista rápida de todo el documento. Incluye el objetivo, alcance, definiciones, abreviaturas utilizadas en el documento]
2. Objetivo	[Indicar el objetivo del documento]
3. Alcance	[Una breve descripción del alcance de este documento, qué otro(s) sistema(s) están asociados o se ven afectados por este documento.]
4. Definiciones Y Abreviaciones	[Una breve descripción del alcance de este documento, qué otro(s) sistema(s) están asociados o se ven afectados por este documento.]
5. Pruebas Ejecutadas	{Detalle de las pruebas realizadas, detallando las ocurrencias de cada sesión}

5.1. Usuarios Expertos				
Nº	Usuario Experto	Área	Evaluación	Pruebas
1.	{Nombre del usuario experto}	{Área o División a la que pertenece}	{Proceso o módulo en el que participará}	{Tipo de pruebas a realizar}
2.	{Nombre del usuario experto}	{Área o División a la que pertenece}	{Proceso o módulo en el que participará}	{Tipo de pruebas a realizar}
3.	{Nombre del usuario experto}	{Área o División a la que pertenece}	{Proceso o módulo en el que participará}	{Tipo de pruebas a realizar}

5.2. Estado de las Pruebas				
Modulo o Sistema	Prueba	Fecha de la Prueba	Estado	Observaciones



{Nombre del Módulo o Sistema Probado}	{Tipo de pruebas a realizar}	{Fecha en que se realizo la prueba}	{Si se ejecuto o se encuentra pendiente o fue cancelada}	{Comentarios u observaciones relevantes}
	{Tipo de pruebas a realizar}	{Fecha en que se realizo la prueba}	{Si se ejecuto o se encuentra pendiente o fue cancelada}	{Comentarios u observaciones relevantes}

5.3. Pruebas Con Observaciones Pendientes

Modulo o Sistema	Prueba	Observaciones	Nueva Fecha de Prueba	Usuario Experto
{Nombre del Módulo o Sistema Probado}	{Tipo de pruebas a realizar}	{observación ocurrida durante la prueba}	{Fecha en que se realizara la prueba}	{Nombre del usuario experto}
	{Tipo de pruebas a realizar}	{observación ocurrida durante la prueba}	{Fecha en que se realizara la prueba}	{Nombre del usuario experto}

Por sistemas

Por usuario

	<p align="center">BN.F17 Plantilla de documento de especificación de ambientes</p> <p align="center">[Nombre del Sistema]</p>	
<p>DEA -{Iniciales del Sistema}-{dd-mm-aa}</p>		

<p>1. Introducción</p>	<p>[La introducción brinda una vista rápida de todo el documento. Incluye el objetivo, alcance, definiciones, abreviaturas utilizadas en el documento]</p>
<p>2. Arquitectura</p>	<p>[Coloque aquí la descripción y las características de la arquitectura empleada por el sistema, explique cuales son los nodos necesarios para garantizar la correcta operatividad del sistema. Asimismo, detalla el objetivo y el software empleado por los diferentes nodos que componen el sistema].</p>

<p>2.1. {Nombre de Nodo 1}</p>	
<p>OBJETIVO</p>	<p>[Esta sección indica el objetivo del nodo para la correcta ejecución del sistema.] [Ejemplo: Servidor de Aplicaciones, Servidor de Base de Datos, etc.]</p>
<p>SOFTWARE</p>	<p>[Esta sección indica el componente de software que ejecuta el nodo.] [Ejemplo: Oracle IAS, Oracle 9i, etc.]</p>

<p>2.2. {nombre de Nodo 2}</p>	
<p>OBJETIVO</p>	<p>[Esta sección indica el objetivo del nodo para la correcta ejecución del sistema.] [Ejemplo: Servidor de Aplicaciones, Servidor de Base de Datos, etc.]</p>
<p>SOFTWARE</p>	<p>[Esta sección indica el componente de software que ejecuta el nodo.] [Ejemplo: Oracle IAS, Oracle 9i, etc.]</p>



<p>3. Ambientes</p> <p>[Esta sección indica la relación de ambientes que deben componer el sistema.]</p> <p>3.1. Ambiente de desarrollo</p> <p>[Coloque aquí la descripción del ambiente.] [Coloque a continuación la lista de nodos que componen el ambiente.]</p>

<p>{Nombre de nodo 1}</p> <p>[Esta sección indica las características del nodo.]</p>	
<p>{Característica 1}</p> <p>[Coloque aquí el nombre de la característica 1 del nodo, por ejemplo: Nombre, IP, Memoria, Disco duro, Otros]</p>	<p>[Coloque aquí la descripción de la característica 1 del nodo. Por ejemplo en el caso de Otros: conexión HTTP.]</p>
<p>{Característica 2}</p> <p>[Coloque aquí el nombre de la característica 2 del nodo, por ejemplo: Nombre, IP, Memoria, Disco duro, Otros]</p>	<p>[Coloque aquí la descripción de la característica 2 del nodo. Por ejemplo en el caso de Otros: conexión HTTP.]</p>
<p>3.2. Ambiente de pruebas internas</p> <p>[Coloque aquí la descripción del ambiente.]</p> <p>[Coloque a continuación la lista de nodos que componen el ambiente.]</p> <p>{NOMBRE DE NODO 1}</p> <p>[Esta sección indica las características del nodo.]</p>	
<p>{Característica 1}</p> <p>[Coloque aquí el nombre de la característica 1 del nodo, por ejemplo: Nombre, IP, Memoria, Disco duro, Otros]</p>	<p>[Coloque aquí la descripción de la característica 1 del nodo. Por ejemplo en el caso de Otros: conexión HTTP.]</p>
<p>{Característica 2}</p> <p>[Coloque aquí el nombre de la característica 2 del nodo, por ejemplo: Nombre, IP, Memoria, Disco duro, Otros]</p>	<p>[Coloque aquí la descripción de la característica 2 del nodo. Por ejemplo en el caso de Otros: conexión HTTP.]</p>
<p>3.3. Ambiente de pruebas de aceptación</p> <p>[Coloque aquí la descripción del ambiente]</p> <p>[Coloque a continuación la lista de nodos que componen el ambiente]</p> <p>{nombre de nodo 1}</p>	

[Esta sección indica las características del nodo]	
{Característica 1} [Coloque aquí el nombre de la característica 1 del nodo, por ejemplo: Nombre, IP, Memoria, Disco duro, Otros]	[Coloque aquí la descripción de la característica 1 del nodo. Por ejemplo en el caso de Otros: conexión HTTP.]
{Característica 2} [Coloque aquí el nombre de la característica 2 del nodo, por ejemplo: Nombre, IP, Memoria, Disco duro, Otros]	[Coloque aquí la descripción de la característica 2 del nodo. Por ejemplo en el caso de Otros: conexión HTTP.]
3.4. Ambiente de producción [Coloque aquí la descripción del ambiente.] [Coloque a continuación la lista de nodos que componen el ambiente.] {nombre de nodo 1} [Esta sección indica las características del nodo]	
{Característica 1} [Coloque aquí el nombre de la característica 1 del nodo, por ejemplo: Nombre, IP, Memoria, Disco duro, Otros]	[Coloque aquí la descripción de la característica 1 del nodo. Por ejemplo en el caso de Otros: conexión HTTP.]
{Característica 2} [Coloque aquí el nombre de la característica 2 del nodo, por ejemplo: Nombre, IP, Memoria, Disco duro, Otros]	[Coloque aquí la descripción de la característica 2 del nodo. Por ejemplo en el caso de Otros: conexión HTTP.]



Por sistemas

Por usuario

	BN.F18 Manual de Usuario [Nombre del Sistema]	
MU -{Iniciales del Sistema}-{dd-mm-aa}		
1. Introducción	[La introducción brinda una vista rápida de todo el documento. Incluye el objetivo, alcance, definiciones, abreviaturas utilizadas en el documento]	
2. Objetivo	[Indicar el objetivo del documento]	
3. Alcance	[Una breve descripción del alcance de este documento, qué otro(s) sistema(s) están asociados o se ven afectados por este documento]	
4. Definiciones y Abreviaciones	[Esta sección brinda la definición de aquellos términos y abreviaciones requeridas para interpretar adecuadamente el contenido de este documento. Esta información debería ser provista en un glosario del documento y aquí únicamente hacer referencia a dicho documento]	
5. Funciones de Sistemas	[Se debe colocar un listado de las funciones que soporta el sistema. Es decir que cosas permite realizar el sistema]	
6. Módulos de Sistemas	[En caso el sistema se encuentre compuesto por módulos claramente diferenciados, en caso contrario se debe indicar únicamente el módulo principal del sistema]	
El sistema se encuentra compuesto por módulos los cuales se describen a continuación		
Módulo	Descripción	
{ Nombre del módulo }	{ Descripción breve del módulo }	
Opción {Nombre de la opción}	{Coloque aquí la descripción correspondiente a la opción del módulo} [A continuación debe indicar la secuencia de pasos a seguir para completar la ejecución de la opción que corresponda y la pantalla asociada. Coloque tantos pasos como sean necesarios]	
Acciones	Pantallas	
1. {Indique que acción debe realizarse en el sistema}	{Coloque la imagen correspondiente aquí. Utilice flechas y elipses para resaltar los elementos más importantes}	
Anexos	[Colocar los anexos que se consideren necesarios.	

Por sistemas

Por usuario



	BN.F.19 Manual de Sistemas [Nombre del Sistema]	
MS -{Iniciales del Sistema}-{dd-mm-aa}		

1. Introducción	[La introducción brinda una vista rápida de todo el documento. Incluye el objetivo, alcance, definiciones, abreviaturas utilizadas en el documento]
2. Objetivo	[Indicar el objetivo del documento]
3. Alcance	[Una breve descripción del alcance de este documento, qué otro(s) sistema(s) están asociados o se ven afectados por este documento]
4. Definiciones y Abreviaciones	[Esta sección brinda la definición de aquellos términos y abreviaciones requeridas para interpretar adecuadamente el contenido de este documento. Esta información debería ser provista en un glosario del documento y aquí únicamente hacer referencia a dicho documento]
5. Modelo de Negocio	<p>[Se debe incluir el documento de Modelo de Negocio]</p> <p>Contiene los siguientes diagramas:</p> <ul style="list-style-type: none"> • Diagrama de Actividad de Negocio • Diagrama de Casos de Uso • Diagrama de Clases de Negocio.
6. Alcance del Sistema	<p>[Se debe incluir el documento de Alcance del Sistema]</p> <p>Contiene los siguientes:</p> <ul style="list-style-type: none"> • Lista de Requerimientos • Análisis de Restricciones • Catalogo de Normas
7. Glosario de Términos	[Se debe incluir el documento de Glosario de términos]
8. Análisis y Diseño	<p>[Se debe incluir el documento de Análisis y Diseño.]</p> <p>Contiene los siguientes:</p> <ul style="list-style-type: none"> • Modelo de Casos de Uso • Modelo de Clases • Modelo de Comportamiento • Modelo de Interacción • Modelo de Datos • Diseño de Interfaz de Usuario

	<ul style="list-style-type: none"> • Descripción de Interfaz con otros sistemas
9. Implementación	<p>[Se debe incluir el documento de Implementación.]</p> <p>Contiene los siguientes diagramas:</p> <ul style="list-style-type: none"> • Diagrama de Componentes • Diagrama de Distribución
10. Prototipo	[Se debe incluir el documento de revisión de Prototipo]
11. Estándares	[Se debe incluir el documento de Estándares]
12. Plan de Migración	[Se debe incluir el documento de Plan de Migración]
13. Plan de Riesgos	[Se debe incluir el documento de Plan de Riesgos]
14. Plan de Iteración	[Se debe incluir el documento de Plan de Interacción.]
15. Plan de Implantación	[Se debe incluir el documento de Plan de Implantación]
16. Anexo	[Colocar los anexos que se consideren necesarios. Puede omitirse en caso no se considere necesario]

Por sistemas

Por usuario

	BN.F.20 Manual de Administración e Instalación [Nombre del Sistema]	
MADMI -{Iniciales del Sistema}-{dd-mm-aa}		

17.Introducción	[La introducción brinda una vista rápida de todo el documento. Incluye el objetivo, alcance, definiciones, abreviaturas utilizadas en el documento]
18.Objetivo	[Indicar el objetivo del documento]
19.Alcance	[Una breve descripción del alcance de este documento, qué otro(s) sistema(s) están asociados o se ven afectados por este documento]
20.Definiciones y Abreviaciones	[Esta sección brinda la definición de aquellos términos y abreviaciones requeridas para interpretar adecuadamente el contenido de este documento. Esta información debería ser provista en un glosario del documento y aquí únicamente hacer referencia a dicho documento]
21.Configuración de Sistemas	Esta sección contiene todo lo necesario para verificar la integridad de la aplicación y de la base de datos. Además se dan las pautas para el despliegue y puesta en producción del sistema. También se lista los requerimientos mínimos necesarios en las estaciones cliente para ejecutar adecuadamente el sistema.
5.1. Base de datos	
5.1.1. Inventario de Objetos de la Base de Datos	
<p>En esta sección se muestra un listado de todos los archivos utilizados en el sistema y que son necesarios para la correcta ejecución del sistema.</p> <p>{Coloque aquí cualquier comentario adicional que se considere necesario}</p> <p>El detalle de todos los archivos se muestra en el anexo 3 el cual debería ser contrastado contra la estructura de la aplicación en producción.</p>	
5.1.2. Valores en los Archivos de Configuración	
<p>En esta sección se muestra un listado de todos los archivos de configuración utilizados en el sistema y que son necesarios para la correcta ejecución del sistema.</p> <p>{Coloque aquí cualquier comentario adicional que se considere necesario}</p> <p>El detalle de todos los valores y de los archivos de configuración se encuentra en el anexo 4 el cual debería ser contrastado contra los archivos de configuración de la aplicación en producción.</p>	

5.2. Aplicación
5.2.1. Inventario de la Aplicación
<p>En esta sección se muestra un listado de todos los archivos utilizados en el sistema y que son necesarios para la correcta ejecución del sistema.</p> <p>{Coloque aquí cualquier comentario adicional que se considere necesario}</p> <p>El detalle de todos los archivos se muestra en el anexo 3 el cual debería ser contrastado contra la estructura de la aplicación en producción.</p>
5.2.2. Valores en los Archivos de Configuración
<p>En esta sección se muestra un listado de todos los archivos de configuración utilizados en el sistema y que son necesarios para la correcta ejecución del sistema.</p> <p>{Coloque aquí cualquier comentario adicional que se considere necesario}</p> <p>El detalle de todos los valores y de los archivos de configuración se encuentra en el anexo 4 el cual debería ser contrastado contra los archivos de configuración de la aplicación en producción.</p>
5.2.3. Despliegue de la Aplicación
{Coloque aquí una explicación detallada de los pasos necesarios ó consideraciones a tener en cuenta para realizar el correcto despliegue ó pase a producción de la aplicación}
5.2.4. Puesta en Marcha de la Aplicación
{Coloque aquí una explicación detallada de los pasos necesarios ó consideraciones a tener en cuenta para poner en funcionamiento de la aplicación. En caso se trate de una aplicación Web se puede indicar a que URL se debe acceder. En caso se trate de una aplicación de escritorio indicar que archivo se debe ejecutar y la forma de ingresar al sistema.}
5.2.5. Procedimiento de Verificación
{Coloque aquí una explicación detallada de los pasos necesarios ó consideraciones a tener en cuenta para probar el correcto funcionamiento del sistema.}
5.3. Requerimientos para las Estaciones de Trabajo
<p>Las estaciones de trabajo deben cumplir los siguientes requerimientos como mínimo para una correcta ejecución de la aplicación:</p> <ul style="list-style-type: none"> • { Requerimiento1 } • { Requerimiento2 } • { Requerimiento3 } • { Requerimiento4 }
6. Recomendaciones
{Coloque aquí las recomendaciones que se consideren necesarias}

Anexos 1	Inventario de Objetos de Base De Datos
[Las tablas que se listan a continuación sirven para determinar el inventario de objetos de la base de datos, sin embargo dependiendo del motor de base de datos a utilizar alguno de ellos podrían no aplicarse y podrían no ser considerados]	

Sinónimos Públicos	
Nombre del Sinónimo	Apuntando a
{Nombre del objeto}	{ Ubicación exacta del objeto al cual referencia }
Cantidad Total	{ Coloque aquí la cantidad total de objetos }

Roles	
Nombre del Rol	Privilegios
{Nombre del objeto}	Privilegios de Sistema {Coloque aquí los privilegios que correspondan} Privilegios de Objeto {Coloque aquí los privilegios que correspondan}
Cantidad Total	{ Coloque aquí la cantidad total de objetos }

Privilegios de los Usuarios Propietarios de los Esquemas en Base de Datos	
Nombre del Usuario	Privilegios
{Nombre del objeto}	Roles asignados {Coloque aquí los roles que correspondan} Privilegios de Sistema {Coloque aquí los privilegios que correspondan} Privilegios sobre Objetos {Coloque aquí los privilegios que correspondan}
Cantidad Total	{ Coloque aquí la cantidad total de objetos }

Paquetes	
Nombre del Paquete	Propietario
{Nombre del objeto}	{ Nombre del propietario }
Cantidad Total	{ Coloque aquí la cantidad total de objetos }

Tablas	
Nombre de la tabla	Propietario
{Nombre del objeto}	{ Nombre del propietario }
Cantidad Total	{ Coloque aquí la cantidad total de objetos }

Vistas	
Nombre de la vista	Propietario
{Nombre del objeto}	{ Nombre del propietario }
Cantidad Total	{ Coloque aquí la cantidad total de objetos }

Índices		
Nombre del índice	Nombre de la tabla	Propietario
{Nombre del objeto}		{ Nombre del propietario }
Cantidad Total	{ Coloque aquí la cantidad total de objetos }	

Triggers	
Nombre del trigger	Nombre de la tabla
{Nombre del objeto}	{ Nombre de la tabla }
Cantidad Total	{ Coloque aquí la cantidad total de objetos }

[NOTA: en general usar una tabla similar a la siguiente para listar cualquier otro objeto que falte incluir. Esta tabla puede ser modificada si se considera necesario para lograr una mejor comprensión]

{Nombre del Tipo de Objeto}	
Nombre del objeto	{Alguna tipo de descripción del objeto }
{Nombre del objeto}	{ Descripción }
Cantidad Total	{ Coloque aquí la cantidad total de objetos }

Anexos 2	Valores en las Tablas de Configuración
[Repita la siguiente tabla por cada una de las tablas de configuración del sistema. Puede colocar la cantidad de columnas que se consideren necesarias].	

Tabla : { Nombre de la tabla }		
Nombre de la columna	Valor	Usado para

Anexos 3	Valores en las Tablas de Configuración
<p>[Los archivos que se listan a continuación sirven para determinar el inventario de archivos de la aplicación.</p> <p>Repita la siguiente tabla por cada uno de los tipos de archivos que se utilizan en el sistema]</p>	

{Coloque aquí el nombre del tipo de archivo que corresponda (Librerías, Clases, Páginas Jsp, etc.)}	
Nombre del archivo	Ubicación
{Nombre del archivo}	{ Ruta ó ubicación exacta del archivo }
Cantidad Total	{ Coloque aquí la cantidad total de objetos }

Anexos 4	Archivos de Configuración de la Aplicación
[Repita el siguiente bloque por cada uno de los archivos de configuración del sistema]	

Archivo : { Nombre del archivo }
{Coloque el contenido del archivo de configuración aquí.}
{A continuación indique el procedimiento a seguir cuando se necesite modificar un valor en el archivo de configuración}

Procedimiento para modificar el archivo
1.
2.

Por sistemas

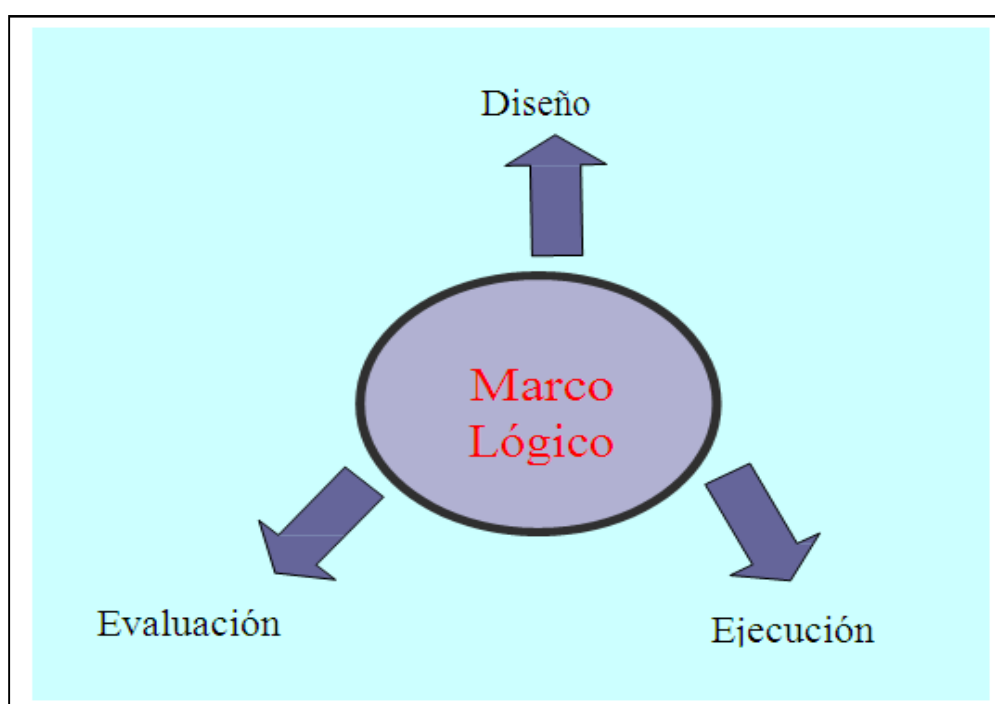
Por usuario

ANEXO 2

1. MARCO LÓGICO

La Metodología de Marco Lógico es una herramienta para facilitar el proceso de conceptualización, diseño, ejecución y evaluación de proyectos. Su énfasis está centrado en la orientación por objetivos, la orientación hacia grupos beneficiarios y el facilitar la participación y la comunicación entre las partes interesadas.¹

Figura A2 – 1: Marco Lógico y Ciclo de Vida del Proyecto



Fuente: Material docente curso del ILPES sobre Marco Lógico, Seguimiento y Evaluación” (Plinio Montalbán).

¹TheLogicalFrameworkApproach.AusGUIDELines,AusAID,página2

El método fue elaborado originalmente como respuesta a tres problemas comunes a proyectos:

- Planificación de proyectos carentes de precisión, con objetivos múltiples que no estaban claramente relacionados con las actividades del proyecto.
- Proyectos que no se ejecutaban exitosamente, y el alcance de la responsabilidad del gerente del proyecto no estaba claramente definida.
- No existía una imagen clara de cómo luciría el proyecto si tuviese éxito, y los evaluadores no tenían una base objetiva para comparar lo que se planeaba con lo que sucedía en la realidad.

El método del marco lógico encara estos problemas, y provee además una cantidad de ventajas sobre enfoques menos estructurados:

- Aporta una terminología uniforme que facilita la comunicación y que sirve para reducir ambigüedades;
- Aporta un formato para llegar a acuerdos precisos acerca de los objetivos, metas y riesgos del proyecto que comparten los diferentes actores relacionados con el proyecto;
- Suministra un temario analítico común que pueden utilizar los involucrados, los consultores y el equipo de proyecto para elaborar tanto el proyecto como el informe de proyecto, como también para la interpretación de éste;
- Enfoca el trabajo técnico en los aspectos críticos y puede acortar documentos de proyecto en forma considerable;
- Suministra información para organizar y preparar en forma lógica el plan de ejecución del proyecto;
- Suministra información necesaria para la ejecución, monitoreo y evaluación del proyecto;
- Proporciona una estructura para expresar, en un solo cuadro, la información más importante sobre un proyecto.

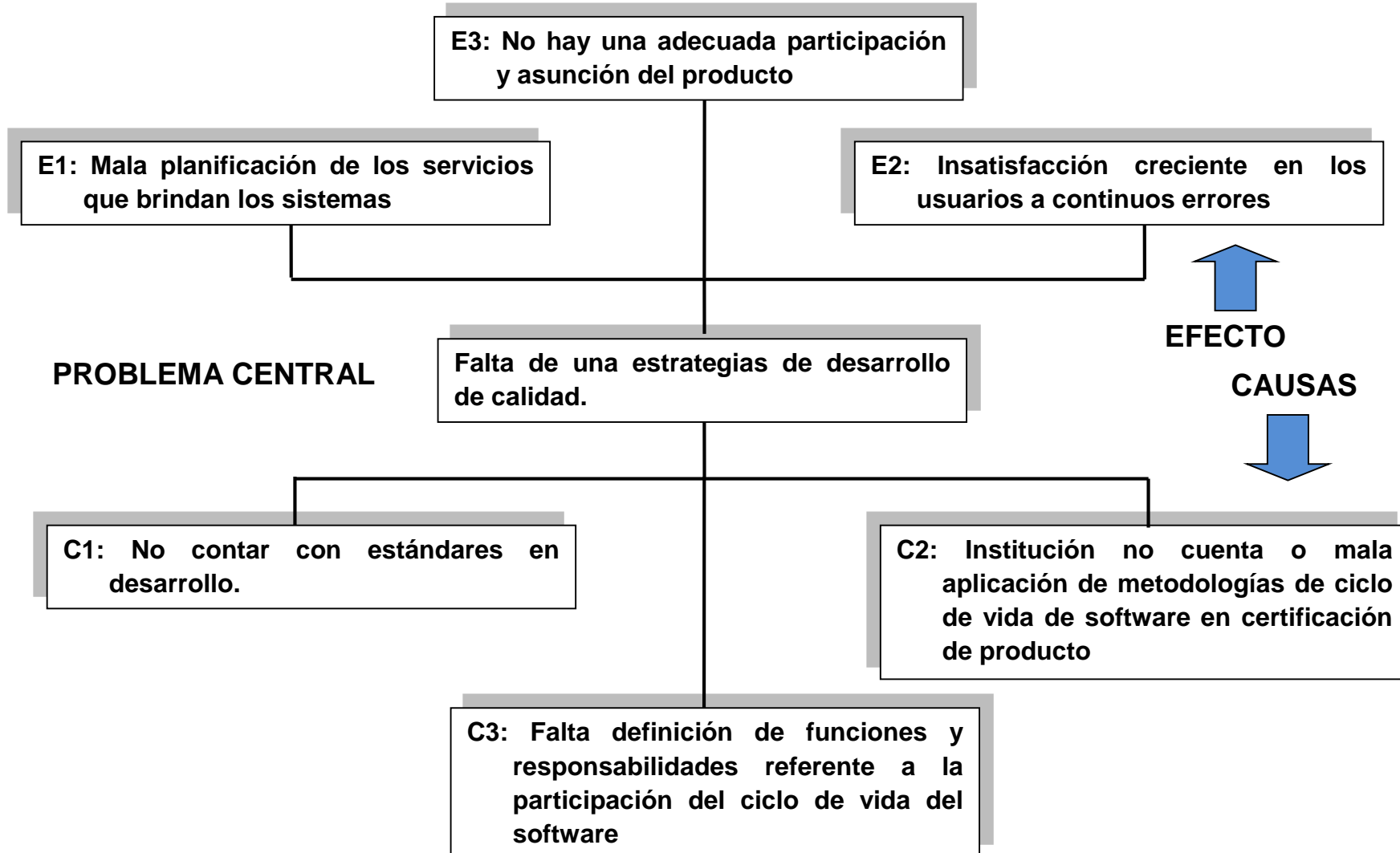
Es importante hacer una distinción entre lo que es conocido como Metodología de Marco Lógico y la Matriz de Marco Lógico. La Metodología contempla análisis del problema, análisis de los involucrados, jerarquía de objetivos y selección de una estrategia de implementación óptima. El producto de esta metodología analítica es la Matriz (el marco lógico), la cual resume lo que el proyecto pretende hacer y cómo, cuáles son los supuestos claves y cómo los insumos y productos del proyecto serán monitoreados y evaluados.²

²Manual de Gestión del Ciclo de Proyecto. Comisión Europea. Marzo de 2001. Página 9.

1.1. INVOLUCRADOS

GRUPO	INTERES	PROBLEMAS PERCIBIDO	RECURSO Y MANDATO
Usuario	Tener un sistema confiable y de bajo costo.	<ul style="list-style-type: none"> ▪ El sistema no es muy confiable. ▪ Errores en la parte funcional. ▪ No cumple en muchos casos con sus necesidades 	R. Disponibilidad en el ciclo de vida de los sistemas para que sean más confiables. Desarrollo
Cliente	Información entrega en línea.	<ul style="list-style-type: none"> ▪ Las colas por la caída de los sistemas. 	R. Alguna disponibilidad de usar el sistema si es confiable.
Proveedor	Mejorar los desarrollos y mantenimientos de los sistemas.	<ul style="list-style-type: none"> ▪ Convocatorias y licitaciones en periodo muy largos. ▪ La falta de atención, ante los problemas presentado a los clientes. 	<p>R. ver las atenciones de los clientes.</p> <p>R. agilizar los procesos de licencia.</p>
Gerencia General	Mejorará los canales de comunicación	<ul style="list-style-type: none"> ▪ Presupuesto insuficiente ▪ Muchos reclamos por parte del cliente 	<p>R. Presupuesto anual operativo asignado por el Gobierno central.</p> <p>M. Servir con los mejores intereses del banco.</p>

1.2. ARBOL DE PROBLEMAS



1.3. MATRIZ DE MARCO LOGICO

MARCO LÓGICO				
	RESUMEN NARRATIVO	INDICADOR VERIFICABLE OBJETIVAMENTE	MEDIOS OPERATIVOS	SUPUESTOS
FIN	<ul style="list-style-type: none"> Contribuir a mejorar la calidad de servicio de software que se implemente en producción 	<ul style="list-style-type: none"> Encuesta satisfacción de usuario 	<ul style="list-style-type: none"> Resultados de la encuesta 	-----
PROPÓSITO	<ul style="list-style-type: none"> Disminuir la insatisfacción del usuario de producción errores en el software para este fin. Se implementara una metodología certificación de producto de software 	<ul style="list-style-type: none"> Cuadro estadístico de producción de errores 	<ul style="list-style-type: none"> Los resultado de los últimos cinco semestre 	<ul style="list-style-type: none"> Mejores en la definición de los requerimientos funcionales. Mejores de los requerimientos técnicos.
COMPONENTE	<ul style="list-style-type: none"> Proceso de Desarrollo <ul style="list-style-type: none"> Plan de proyecto Requerimientos del software Implementación y pruebas unitarias Integración y pruebas de software Aceptación de software Transición del software 	<ul style="list-style-type: none"> Plan desarrollo de software "Proyecto" 	<ul style="list-style-type: none"> Cronograma 	<ul style="list-style-type: none"> Conformidad de los entregables
	<ul style="list-style-type: none"> Proceso de Mantenimiento <ul style="list-style-type: none"> Análisis de problema o Planteamiento de la modificación Implementación de la modificación Revisión y aceptación del mantenimiento 	<ul style="list-style-type: none"> Plan mantenimiento de software "Proyecto" 	<ul style="list-style-type: none"> Cronograma 	<ul style="list-style-type: none"> Conformidad de los entregables
	<ul style="list-style-type: none"> Proceso de Certificación <ul style="list-style-type: none"> Planificación de pruebas Despliegue de producto software Pruebas previas Pruebas de acreditación Pase a producción Seguimiento de producto certificado 	<ul style="list-style-type: none"> Plan de pruebas 	<ul style="list-style-type: none"> Cronograma 	
ACTIVIDADES	<ul style="list-style-type: none"> Proceso de Desarrollo <ul style="list-style-type: none"> ✓ Actividades de Requerimientos de Software. 	<ul style="list-style-type: none"> Retrasos de fechas 	<ul style="list-style-type: none"> Lista de incidencias 	

MARCO LÓGICO				
	RESUMEN NARRATIVO	INDICADOR VERIFICABLE OBJETIVAMENTE	MEDIOS OPERATIVOS	SUPUESTOS
	<ul style="list-style-type: none"> ✓ Actividades del Diseño de Software. ✓ Actividades de Implementación (codificación) y Pruebas Unitarias. ✓ Actividades de Integración. ✓ Actividades de apoyo a la Aceptación del Software. ✓ Actividades de Transición del Software a Certificación. 	programados		<ul style="list-style-type: none"> ▪ Capacitaciones
	<ul style="list-style-type: none"> ▪ Proceso de Mantenimiento <ul style="list-style-type: none"> ✓ Análisis del problema o Planteamiento de la modificación. ✓ Implementación de las Modificaciones. ✓ Revisión y Aceptación del Mantenimiento 	<ul style="list-style-type: none"> ▪ Retrasos de fechas programados 	<ul style="list-style-type: none"> ▪ Lista de incidencias 	<ul style="list-style-type: none"> ▪ Capacitaciones
	<ul style="list-style-type: none"> ▪ Proceso de Certificación <ul style="list-style-type: none"> ✓ Actividades de Participación en Requerimientos de Software. ✓ Actividades de Planificación de las Pruebas. ✓ Actividades de Despliegue del producto software en ambiente de Certificación. ✓ Actividades de Pruebas Previas. ✓ Actividades de Pruebas de Acreditación. ✓ Actividades de Pase a Producción. ✓ Actividades de Seguimiento y Control. 	<ul style="list-style-type: none"> ▪ Retrasos de fechas programados 	<ul style="list-style-type: none"> ▪ Resultados de las pruebas 	<ul style="list-style-type: none"> ▪ Capacitaciones