ESCUELA SUPERIOR DE INGENIERÍA INFORMÁTICA

INGENIERÍA INFORMÁTICA

Curso Académico 2009/2010

Proyecto de Fin de Carrera

UNA REVISIÓN Y COMPARATIVA DE MODELOS DE PROCESOS DE PRUEBA

Autora: Sandra Sánchez Melchor

Tutor: Javier Garzás Parra

ÍNDICE

ÍNDICE **2**

CAPÍTULO 1 INTRODUCCIÓN **5**

1.1 JUSTIFICACIÓN Y PLANTEAMIENTO DE ESTE TRABAJO 5

1.1.1 Problemática *5*

1.1.2 Propuesta de este trabajo *6*

1.1.3 Ventajas *6*

1.2 OBJETIVO 7

1.3 EL CONTEXTO DE TRABAJO 8

1.3.1 La calidad del Software *8*

1.3.2 Modelo de Mejora *9*

1.3.3 Las pruebas del Software *10*

1.3.4 El proceso de Pruebas del Software o Testing *11*

1.4 MÉTODO DE TRABAJO 12

1.5 ESTRUCTURA DE ESTE TRABAJO 13

CAPÍTULO 2 REVISIÓN DE TRABAJOS **14**

2.1 CMMI 14

2.1.1 Áreas de Proceso que cubren las Pruebas *15*

2.1.2 Proceso de Validación *15*

2.1.3 Proceso de Verificación *17*

2.2 ISO/IEC 12207 22

2.2.1 Procesos que cubren las pruebas *23*

2.2.2 Proceso de Validación del Software *23*

2.2.3 Proceso de Verificación del Software *25*

2.3 TMM Y TMMI 28

2.3.1 Niveles TMMI *31*

2.4 TPI 34

2.5 TESTPAI 36

2.6 TMAP 39

2.6.1 Los 4 elementos esenciales de TMap *40*

2.7 IT MARK 43
2.7.1 Perspectivas desde las que trabaja 44
2.7.2 Niveles IT Mark 44
2.7.3 Certificación IT Mark 46
2.8 COMPETISOFT 47
2.8.1 Modelo de Procesos 49
2.8.2 El Modelo de Evaluación. 52
2.8.3 El Modelo de Mejora 53
2.9 ISO/IEC 29110 56
2.9.1 Visión general (TR 29110-1) 58
2.9.2 Perfiles (ISP) 58
2.9.3 Guías 59
2.10 ISO/IEC 29119 61
2.10.1 Parte 1 - Conceptos y Vocabulario 62
2.10.2 Parte 2 - Proceso de pruebas 63
2.10.3 Parte 3 - Documentación de pruebas 67
2.10.4 Parte 4 - Técnicas de pruebas 67

CAPÍTULO 3 "UNA REVISION Y COMPARATIVA DE MODELOS DE PROCESOS DE PRUEBA" 68

3.1 COMPARATIVA DE MODELOS 68
3.2 PUNTOS FUERTES Y PUNTOS DÉBILES DE LOS MODELOS ESTUDIADOS 72
3.2.1 CMMI 72
3.2.2 ISO/IEC 12207 73
3.2.3 TMM/TMMI 74
3.2.4 TPI 74
3.2.5 TESTPAI 75
3.2.6 TMAP 75
3.2.7 IT MARK 76
3.2.8 COMPETISOFT 77
3.2.9 ISO/IEC 29110 78
3.2.10 ISO/IEC 29119 78
3.3 RECOMENDACIÓN FINAL 79

CAPÍTULO 4 CONCLUSIONES 81

4.1 ANÁLISIS DE LA CONSECUCIÓN DE OBJETIVOS 81

4.2 LOGROS Y APORTACIONES 82

4.3 LÍNEAS FUTURAS 82

ANEXO I. ÍNDICE DE FIGURAS Y TABLAS 84

REFERENCIAS 85

Capítulo 1 INTRODUCCIÓN

1.1 JUSTIFICACIÓN Y PLANTEAMIENTO DE ESTE TRABAJO

Las aplicaciones de software son cada vez más importantes en las labores cotidianas. Potenciado con las mejoras tecnológicas en los equipos de computación y el auge de Internet, soportan en muchos casos la mayor parte de las operaciones de negocio.

El software se ha convertido en una actividad programada, planificada y con un ciclo de vida definido que le otorga un carácter formal.

Las deficiencias en el proceso de pruebas de software son consecuencia de los problemas que se puedan dar a lo largo del proceso en que las entidades intentan conseguir sus objetivos.

Uno de los aspectos más descuidado en el proceso de desarrollo del software es el de las pruebas, a pesar de ser el medio por el cual se puede asegurar la calidad del producto software. Cumplir con los requisitos del usuario en la mayoría de los casos no es suficiente, para ello hace falta definir un proceso de pruebas serio, uniforme para todos los proyectos, es decir, estandarizado y que cumpla las mejores prácticas de pruebas siempre bajo un modelos que asegure la mejora continua.

1.1.1 Problemática

El software no sólo está cada vez más presente en nuestras vidas cotidianas (casi sin que nos demos cuenta de ello), si no también se vuelve cada vez más complejo y extenso. Prácticamente todos los sistemas críticos, como aparatos médicos, centrales nucleares o sistemas de guiado de trenes y aviones, hacen un uso intensivo del software. Además, el diseño e implementación de software es un proceso cada vez más complejo, en el cual, cualquier fallo puede tener gravísimas consecuencias

Aún cuando no faltan ejemplos de problemas ocasionados por los defectos del software, muchas organizaciones siguen subestimando sus riesgos y posibles consecuencias.

En el caso de instituciones financieras o empresas basadas en Internet, los daños producidos por errores del software pueden llegar a ser mucho mayores, por ejemplo cuando impiden las transacciones o limitan el acceso de los clientes a sus servicios.

Cada año se pierden miles de millones de euros a causa de los defectos del software. Sin embargo, muchas empresas no son conscientes de los riesgos que conllevan el desarrollo

y la modificación de aplicaciones informáticas. Para solucionar este problema, primero se han de identificar y medir estos riesgos, para después pasar a neutralizarlos mediante un proceso de pruebas estructurado.

1.1.2 Propuesta de este trabajo

Las actividades de pruebas del software pasan casi desapercibidas, pues normalmente no se emplea el tiempo necesario en crear casos de prueba que permitan garantizar la calidad del software. La ausencia de una organización clara en la planificación del proyecto y de políticas organizacionales que apoyen este proceso debido al desconocimiento o inaplicabilidad de algunos modelos de calidad aumentan el riesgo de producir software que inmediatamente reporta continuos errores o fallos graves. No se trata sólo de invertir más tiempo o contratar más personal, lo que se necesita es una metodología que defina actividades y responsabilidades.

Existen diferentes modelos de referencia que definen el conjunto de buenas prácticas a realizar dentro de una organización con el objetivo de llevar a cabo una mejora de procesos.

Estos modelos ofrecen el marco de referencia necesario para determinar el conjunto de fortalezas y debilidades en el área de pruebas del software.

Un mismo modelo puede resultar ineficiente para una organización mientras que para otra puede ser de gran ayuda en su proceso de desarrollo de software. Por este motivo, cada organización debería estudiar los distintos estándares y modelos que existen para poder seleccionar el que más se adecue a su situación. Con tal fin, este proyecto trata de facilitar dicha labor a las organizaciones, estudiando los modelos más destacados en el área de las pruebas del software, realizando a su vez una tabla comparativa y una guía de selección de modelo.

1.1.3 Ventajas

Existen importantes beneficios para las organizaciones software asociados a la implantación de modelos de mejora de procesos de mejora, tales como:

- Se produce un incremento de la satisfacción del cliente al utilizar un software con una cantidad de errores inferior.
- Se incrementa la eficiencia del proceso de desarrollo.
- Se facilita la definición y cumplimiento de los objetivos de calidad.
- Se incrementa la satisfacción de los trabajadores debido a que se proporcionan herramientas y recursos apropiados para la realización eficiente del trabajo.

- Reducción de un 20% en los errores en el software entregados al cliente
- El esfuerzo en pruebas de software se reduce en un importante porcentaje del esfuerzo total del proyecto.
- El número de errores detectados en las pruebas de aceptación se reduce respecto de los errores detectados en las pruebas de integración.
- Los errores informados por el cliente/usuarios se reducen en un importante porcentaje.
- Se detecta una importante reducción de los costes de ejecución.
- Reducción en el tiempo de entrega e incremento en la eficiencia de las pruebas.

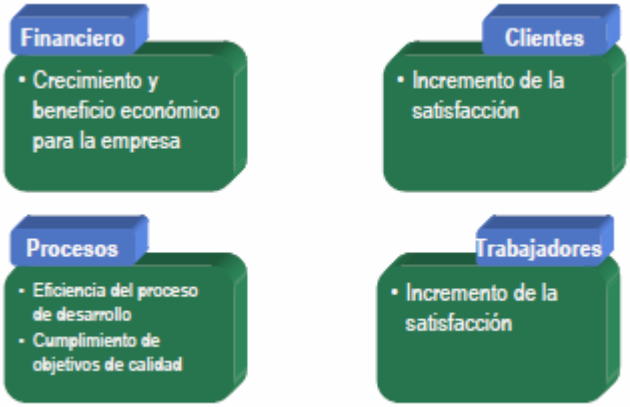

Figura 1. Beneficios en la mejora de procesos de pruebas del Software

Además de los beneficios asociados a la implantación de un modelo de mejora de procesos de pruebas del software se deben añadir los beneficios relativos al objetivo de este proyecto.

1.2 OBJETIVO

Este proyecto se centra en 4 objetivos principales:

- Estudiar las mejores prácticas, modelos y normas de pruebas software que actualmente existen.
- Realizar un estudio comparativo de todos los modelos estudiados, relacionándolos en una tabla, con el objetivo de identificar similitudes y diferencias entre dichos modelos.
- Obtener fortalezas y debilidades de cada propuesta para facilitar así la selección del modelo que más se adecue a cada organización.

- Elaboración de una recomendación final a la hora de seleccionar el modelo que mejor se adapte a la organización que desea implantarlo teniendo en cuenta diversos aspectos, tales como tipo de organización, coste de implantación, madurez del modelos, etc.

1.3 EL CONTEXTO DE TRABAJO

1.3.1 La calidad del Software

El término "calidad", como tal, tiene una definición ambigua. No obstante, las normas y modelos internacionales ofrecen varias acepciones, basadas sobre todo en la adaptabilidad de uso (mediante la calidad de diseño y la calidad de conformidad del usuario). Según estas normas, comúnmente aceptadas, la calidad debe ser mensurable y predecible, y debe incorporar varios factores básicos:

- Ausencia de defectos
- Satisfacción del usuario
- Conformidad con los requerimientos

Según la norma ISO 9126 sobre Tecnologías de la Información y calidad de los productos software, será sustituida por la ISO 25000, de la que se hablará más adelante, las características de la calidad son:

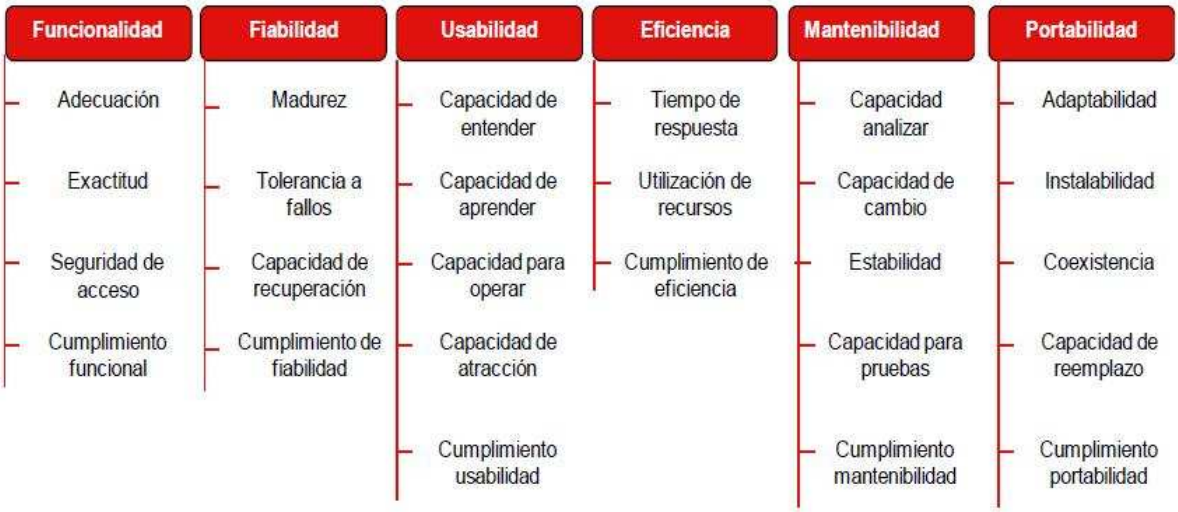

Figura 2. Características de calidad (ISO 9126)

1.3.2 Modelo de Mejora

Un modelo de mejora es un patrón a seguir, mediante el cual, una empresa intenta modificar sus procesos para ser más eficaces y reducir costes. Mediante el seguimiento de este patrón, que en la mayor parte de los casos conlleva reorganizaciones internas, se persigue implícitamente la mejora en la calidad de los productos desarrollados.

La Ingeniería del Software abarca un amplio espectro de disciplinas, entre las cuales se encuentra el dominio en la aplicación de modelos y estándares de calidad. Estos permiten que las empresas puedan implementar la calidad en una doble vertiente: a nivel proceso y a nivel producto. Cada modelo o estándar puede, pues, tener una aplicación concreta o limitada orientada a lograr mejorar determinados objetivos.

1.3.2.1 La mejora orientada hacia los procesos de una organización

El proceso software objeto de mejora podría definirse como el "conjunto de actividades, métodos, prácticas y transformaciones que la gente usa para desarrollar y mantener software y los productos de trabajo asociados, como planes de proyecto, diseño de documentos, código, pruebas y manuales de usuario" (según el *Software Engineering Institute*). La definición de ISO, por otra parte, define este proceso como "el proceso o conjunto de procesos usados por una organización o proyecto, para planificar, gestionar, ejecutar, monitorizar, controlar y mejorar sus actividades software relacionadas".

El objetivo final de un modelo de procesos es lograr una representación clara de los procesos reales de desarrollo que se implementan en una organización, con la cual poder trabajar para planificar las mejoras a incluir en cada uno de esos procesos. Si la representación conceptual del proceso es buena, los análisis de estos procesos sobre el papel permitirán a la empresa la posibilidad de automatizarlos, controlar su eficiencia, comprobar las interacciones con otros procesos, y ofrecer a la Dirección una nueva fuente de información, como puede ser la información actual del estado de cada proceso en cualquier momento, el significado que debe darse a cada uno de los puntos de decisión, etc.

La mejora del producto final pasa, según estos modelos, por la mejora de los procesos que llevan a su creación. La adopción del modelo o metodología adecuados podrá realizar esta mejora con una correcta implantación, dotando implícitamente al producto final de una calidad manifiesta.

La calidad a nivel de procesos puede ser evaluada de manera genérica o específica, según el modelo o estándar seleccionado. Todo modelo o estándar a nivel de proceso tiene que tener un ámbito de aplicación específico y tiene como finalidad la mejora continua.

1.3.2.2 La mejora orientada hacia un producto

Dentro del conjunto de modelos y metodologías de mejora de la calidad, existen un conjunto de modelos que sitúan el foco de la mejora no en el proceso, sino en el propio producto final. Para ello, se aplica un modelo de pruebas en el ciclo de vida del software.

La estrategia en este caso se basa en el establecimiento de varios puntos de testeo y verificación sobre todas las etapas del ciclo de vida (requisitos, análisis, diseño y construcción), además de la última etapa de pruebas finales. Las técnicas a emplear suelen ser las siguientes:

- Detección temprana de errores mediante la anticipación de los bancos de pruebas.
- Utilización de técnicas especializadas, como pueden ser las pruebas estáticas, dinámicas, pruebas unitarias, pruebas de caja negra y caja blanca, etc.
- Incorporación a los equipos de trabajo de grupos especializados en *testing*.

No obstante, y a pesar de que los pilares básicos del desarrollo de software son comunes para todas las organizaciones, por regla general la aplicación de este tipo de aproximaciones a la calidad mediante los modelos orientados a producto dependen de cada organización, por lo que las soluciones no son fácilmente generalizables.

1.3.3 Las pruebas del Software

El software no sólo está cada vez más presente en nuestras vidas cotidianas (casi sin que nos demos cuenta de ello), sino también se vuelve cada vez más complejo y extenso. Prácticamente todos los sistemas críticos, como aparatos míticos, centrales nucleares o sistemas de guiado de trenes y aviones, hacen un uso intensivo del software. Además, el diseño e implementación de software es un proceso cada vez más complejo, en el cual cualquier fallo puede tener gravísimas consecuencias.

Las pruebas representan una actividad fundamental en el desarrollo de software y, en muchos casos, suponen prácticamente el único medio empleado en los proyectos para la verificación y validación del software.

Las pruebas de software son seguramente la actividad más común de control de calidad realizada en los proyectos de desarrollo o mantenimiento de aplicaciones y sistemas.

Aunque un aseguramiento de calidad de software más eficaz debería incluir otras técnicas como, por ejemplo, inspecciones y revisiones (automatizadas o no) de modelos y documentos no ejecutables de las primeras fases de desarrollo, no existe proyecto de desarrollo que no realice de manera más o menos exhaustiva y formal pruebas de software. Las pruebas de software se definen como “una actividad en la cual un sistema o uno de sus

componentes se ejecuta en circunstancias previamente especificadas, los resultados se observan y registran y se realiza una evaluación de algún aspecto".

1.3.4 El proceso de Pruebas del Software o Testing

Testing es el proceso de planificación, preparación y medición, destinado a establecer las características de un sistema de información y a demostrar la diferencia entre el estado real y el estado requerido.

Los objetivos del Testing son:

- Encontrar los fallos más importantes lo antes posible y al menor coste.
- Mejorar la calidad de los productos.
- Informar sobre la calidad y los riesgos.

En el mercado actual, las organizaciones se centran cada vez más en obtener el máximo rendimiento de sus sistemas. Una entrega rápida y la calidad del software son cada vez más importantes, y los riesgos de una insuficiente calidad son cada vez mayores.

El testing es, por tanto, una de las acciones más importantes que una organización debe llevar a cabo para controlar todos estos riesgos ya que nos permite conocer la calidad del software y los riesgos asociados a una insuficiente calidad de éste. De este modo, gracias a él, las organizaciones disponen del conocimiento necesario para tomar decisiones sobre la implantación o no de un sistema desarrollado.

El Software testing o como se conoce en español las pruebas de software se aplican como una etapa más del proceso de desarrollo de software, su objetivo es asegurar que el software cumpla con las especificaciones requeridas y eliminar los posibles defectos que este pudiera tener. En un principio la mayoría de empresas de desarrollo contaban con una etapa de pruebas demasiado informal, en la actualidad el software testing se ha convertido en una de las etapas más críticas del ciclo de vida del desarrollo de software y esto ha causado el origen de diversas metodologías.

En la actualidad el software testing se hace más complicado ya que debe hacer frente a una gran cantidad de metodologías de desarrollo, lenguajes de programación, sistemas operativos, hardware etc.

Es por esto que el testing debe apoyarse en metodologías generales que revisan los aspectos más fundamentales que debe considerar todo proceso de pruebas. Debido a esta complejidad actualmente se cuentan con una gran cantidad de software diseñado exclusivamente para la etapa de pruebas, incluyendo la gestión del proceso de software

testing, la administración y seguimiento de errores, la administración de los casos de prueba, automatización de pruebas etc.

Una vez culminadas las etapas de análisis, diseño y desarrollo se inicia la etapa de pruebas, en esta etapa lo recomendable es que el software se mantenga en un ambiente aislado o separado del ambiente de desarrollo o de producción, lo ideal es preparar un ambiente de pruebas lo más parecido a los ambientes que existen en producción para asegurar su correcto funcionamiento en esa futura etapa, se debe considerar adquirir un equipo de pruebas especializado "software tester" o analista de pruebas, con experiencia, estas personas tienen una formación que les permite detectar una gran cantidad de errores en tiempos mínimos, así como una metodología especifica que les permite hacer el trabajo de manera correcta, algunas empresas más informales utilizan a los futuros usuarios del sistema como testers situación que puede traer una serie de problemas debido a la poca experiencia que pueden tener los usuarios en la detección de errores, además se obvian pruebas importantes como las pruebas de Esfuerzo o "Stress testing", también se dejan de lado las pruebas unitarias o pruebas modulares, las que deberían asegurar que cada modulo del sistema trabaje correctamente de manera independiente, otro error muy conocido en empresas de software es el uso de los mismos desarrolladores como analistas de pruebas, es muy difícil probar con objetividad un software si nosotros mismos lo hemos desarrollado, un técnico o analista programador empezara a probar con la idea preconcebida de que su hijito trabaja a la perfección e inconscientemente evitara realizar pruebas más exhaustivas considerando que las mismas podrían ser absurdas o innecesarias, lo bueno es que poco a poco estas ideas van quedando descartadas y se van alineando conceptos hacia un software testing profesional.

1.4 MÉTODO DE TRABAJO

El desarrollo de este trabajo se divido en varias fases.

En primer lugar, se ha realizado un estudio del problema. Para ello se ha reunido información sobre términos relacionados con la Calidad del Software, Pruebas del Software, Testing o Procesos de Pruebas del Software, Modelos de Mejora, Modelos de Procesos del Software, etc. Una vez comprendidos dichos términos, se procedió a reunir información acerca de los modelos de procesos que existen actualmente y a realizar resúmenes de dichos modelos centrándonos en los procesos de pruebas del software (ya que este proyecto trata dichos procesos).

Con toda la información reunida se elaboró un estudio comparativo de los modelos de procesos de pruebas del software estudiados, el cual incluye una tabla comparativa de los

modelos referenciados, las debilidades y fortalezas de cada modelo y por último, una recomendación final acerca de cómo elegir el modelo que más se adecue a la organización que desea implantar el modelo de procesos.

A la vez que se realizaban las tareas descritas anteriormente, se iba elaborando esta memoria.

1.5 ESTRUCTURA DE ESTE TRABAJO

La memoria de este proyecto está compuesta por 4 apartados o capítulos:

1. **Introducción**: este apartado está dedicado a la presentación la justificación y planteamiento de este proyecto del problema, incluyendo la situación actual, problemática, solución y ventajas; a la exposición de los objetivos, al contexto de trabajo, a la metodología de trabajo empleada y la estructura de esta memoria.
2. **Revisión de trabajos**: en este apartado se expone una breve introducción a cada uno de los modelos de procesos de pruebas de software estudiados.
3. **Una revisión y comparativa de modelos de procesos de prueba**: este apartado incluye la tabla comparativa de los modelos estudiados en el apartado anterior y una guía para seleccionar un modelo.
4. **Conclusiones y trabajos futuros**: en el que se detallan las conclusiones obtenidas del proyecto y las futuras líneas de investigación en el campo de los modelos de procesos de pruebas del software.

Al final del documento se adjuntan las referencias de la bibliografía consultada para la realización de este proyecto.

Capítulo 2 REVISIÓN DE TRABAJOS

A continuación se revisarán los modelos de procesos más destacados que existen actualmente en el campo de Tecnologías de la Información.

2.1 CMMI

Capability Maturity Model Integration (CMMI, Integración de Modelos de Capacidad de Madurez) es un modelo para la mejora y evaluación de procesos para el desarrollo, mantenimiento y operación de sistemas de software.

CMMI es la evolución de CMM. CMM Fue desarrollado desde 1987 hasta 1997. El objetivo del proyecto CMMI es mejorar la usabilidad de modelos de madurez integrando varios modelos diferentes en un solo marco (framework). Fue creado por miembros de la industria, el gobierno y el SEI[1] (Software Engineering Institute). Entre los principales patrocinadores se incluyen la Oficina del Secretario de Defensa (OSD) y la National Defense Industrial Association.

Se desarrolla sobre el *principio de calidad de Jurán* de solvencia contrastada en los 80 en la producción industrial: "**la calidad del resultado depende principalmente de la calidad de los procesos empleados en su desarrollo**".

En la actualidad hay tres áreas cubiertas por los modelos de CMMI: *Desarrollo, Adquisición y Servicios*.

La versión actual de CMMI es la versión 1.2. Hay tres constelaciones de la versión 1.2 disponible:

- **CMMI para el Desarrollo** (CMMI-DEV o CMMI for Development), en él se tratan procesos de desarrollo de productos y servicios.
- **CMMI para la adquisición** (CMMI-ACQ o CMMI for Acquisition), en él se tratan la gestión de la cadena de suministro, adquisición y contratación externa en los procesos del gobierno y la industria.

[1] ***Software* Engineering Institute.** El Instituto de Ingeniería de *Software* de los Estados Unidos es un organismo financiado por el Gobierno Federal de Estados Unidos y operado por la universidad Carnegie Mellon de este mismo país. Pretende ayudar a las organizaciones a mejorar el estado de las prácticas de ingeniería, su meta es proporcionar a las organizaciones las pautas de actuación necesarias para obtener mejoras observables en su proceso de desarrollo del *software*, de manera que desarrollen productos sin defectos respetando requerimientos, fechas y costos.

- **CMMI para servicios** (CMMI-SVC o CMMI for Services), está diseñado para cubrir todas las actividades que requieren gestionar, establecer y entregar Servicios.

Independientemente de la constelación\modelo que opta una organización, las prácticas CMMI deben adaptarse a cada organización en función de sus objetivos de negocio.

Una organización es evaluada (por ejemplo, usando un método de evaluación como SCAMPI) y recibe una calificación de nivel 1-5 si sigue los niveles de Madurez (si bien se comienza con el nivel 2). En caso de que quiera la organización, puede coger áreas de proceso y en vez de por niveles de madurez puede obtener los niveles de capacidad en cada una de las Áreas de Proceso, obteniendo el "Perfil de Capacidad" de la Organización.

2.1.1 Áreas de Proceso que cubren las Pruebas

El modelo CMMI v1.2 (CMMI-DEV) contiene las siguientes 22 áreas de proceso, de las cuales 2 cubren los procesos de pruebas:

- Validación (VAL)
- Verificación (VER)

2.1.2 Proceso de Validación

El propósito de Validación (VAL) es demostrar que un producto o componente de producto se ajusta a su uso previsto cuando se sitúa en su entorno previsto.

2.1.2.1 Resumen de Metas y prácticas específicas

SG1 Preparar la validación.

SP 1.1 Seleccionar los productos a validar.

Las actividades de preparación incluyen seleccionar los productos y los componentes de producto a validar, y establecer y mantener el entorno, los procedimientos y los criterios de validación.

Se prepara el entorno requerido para validar el producto o el componente de producto. El entorno puede comprarse o puede especificarse, diseñarse y construirse.

SP 1.2 Establecer el entorno de validación.

Productos de trabajo típicos

- Entorno de validación.

Subprácticas

- Identificar los requerimientos del entorno de validación.

- Identificar los productos suministrados por el cliente.
- Identificar los elementos de reutilización.
- Identificar el equipamiento y las herramientas
- Identificar los recursos de validación que están disponibles para reutilización y modificación.
- Planificar en detalle la disponibilidad de los recursos.

SP 1.3 Establecer los procedimientos y los criterios de validación.

Los procedimientos y criterios de validación incluyen la prueba y evaluación de los servicios de mantenimiento, de formación y de soporte.

Productos de trabajo típicos

- Procedimientos de validación.
- Criterios de validación.
- Procedimientos de prueba y evaluación para mantenimiento, formación y soporte.

Subprácticas

- Revisar los requerimientos del producto para asegurar que se identifican y se resuelven los problemas que afectan a la validación del producto o del componente de producto.
- Documentar el entorno, escenario operacional, procedimientos, entradas, salidas y criterios para la validación del producto o del componente de producto seleccionado.
- Evaluar el diseño a medida que madura en el contexto del entorno de validación, para identificar problemas de validación.

SG 2 Validar el producto o los componentes de producto.

El producto o los componentes de producto son validados para asegurar que sean adecuados para usar en su entorno operacional previsto.

SP 2.1 Realizar la validación.

Se realizan las actividades de validación y se recogen los datos resultantes de acuerdo a los métodos, procedimientos y criterios establecidos.

Los procedimientos de validación, tal como se ejecutaron, deberían documentarse, y las desviaciones que ocurren durante la ejecución deberían ser anotadas, según sea apropiado.

Productos de trabajo típicos

- Informes de validación.
- Resultados de validación.
- Matriz de referencias cruzadas de validación.
- Registro de procedimientos tal como se ejecutaron.
- Demostraciones operacionales.

SP 2.2 Analizar los resultados de la validación.

Los datos resultantes de las pruebas de validación, inspecciones, demostraciones o evaluaciones de validación, se analizan frente a los criterios de validación definidos.

Productos de trabajo típicos

- Informes de deficiencias en la validación.
- Problemas de validación.
- Petición de cambio de procedimiento.

Subprácticas

- Comparar los resultados reales con los resultados esperados.
- En base a los criterios de validación establecidos, identificar los productos y los componentes de producto que no funcionan adecuadamente en sus entornos operacionales previstos, o identificar los problemas con los métodos, con los criterios y/o con el entorno.
- Analizar los datos de validación en cuanto a defectos.
- Registrar los resultados del análisis e identificar los problemas.
- Usar los resultados de la validación para comparar las mediciones y el rendimiento reales para el uso previsto o la necesidad operacional.

2.1.3 Proceso de Verificación

El propósito de la Verificación (VER) es asegurar que los productos de trabajo seleccionados cumplen sus requerimientos especificados.

2.1.3.1 Resumen de Metas y prácticas específicas

SG1 Preparar la verificación.

SP 1.1 Seleccionar los productos a verificar.

Los métodos de verificación incluyen, pero no están limitados a, inspecciones, revisiones entre pares, auditorias, reuniones de revisión (walkthroughs), análisis,

simulaciones, pruebas y demostraciones. Las prácticas relativas a revisiones entre pares, como un método específico de verificación, se incluyen en la meta específica 2.

La preparación también implica la definición de herramientas de soporte, equipamiento y software de prueba, simulaciones, prototipos e instalaciones.

Productos de trabajo típicos

- Lista de productos de trabajo seleccionados para la verificación.
- Métodos de verificación para cada producto de trabajo seleccionado.

Subprácticas

- Identificar los productos de trabajo a verificar.
- Identificar los requerimientos a satisfacer por cada producto de trabajo seleccionado.
- Identificar los métodos de verificación que están disponibles para usar.
- Definir los métodos de verificación a usar para cada producto de trabajo seleccionado.
- Enviar la identificación de los productos de trabajo a verificar, los requerimientos a satisfacer y los métodos a usar para su integración con el plan del proyecto.

SP 1.2 Establecer el entorno de verificación.

Seleccionar los productos de trabajo a verificar y los métodos de verificación que serán usados para cada uno.

Productos de trabajo típicos

- Entorno de verificación.

Subprácticas

- Identificar los requerimientos del entorno de verificación.
- Identificar los recursos de verificación que están disponibles para su reutilización y modificación.
- Identificar el equipamiento y las herramientas de verificación.
- Adquirir el equipamiento de soporte y un entorno de verificación, tales como equipamiento y software de prueba.

SP 1.3 Establecer los procedimientos y los criterios de verificación.

Productos de trabajo típicos

- Procedimientos de verificación.
- Criterios de verificación.

Subprácticas

- Generar el conjunto completo e integrado de procedimientos de verificación para productos de trabajo y para cualquier producto COTS[2], según sea necesario.
- Desarrollar y refinar los criterios de verificación cuando sea necesario.
- Identificar los resultados esperados, cualquier tolerancia permitida en la observación, y otros criterios para satisfacer los requerimientos.
- Identificar cualquier equipamiento y componentes ambientales necesarios para dar soporte a la verificación.

SG2 Realizar las revisiones entre pares

Las revisiones entre pares implican un examen metódico de los productos de trabajo por los pares de los productores, para identificar defectos a eliminar y para recomendar otros cambios que son necesarios.

SP 2.1 Preparar las revisiones entre pares.

Las actividades de preparación para las revisiones entre pares normalmente incluyen identificar al personal que se invitará a participar en la revisión entre pares de cada uno de los productos de trabajo; identificar los revisores claves que deben participar en la revisión entre pares; preparar y actualizar cualquier material que será usado durante las revisiones entre pares, como listas de comprobación y criterios de revisión, y el calendario de las revisiones entre pares.

Productos de trabajo típicos

- Calendario de la revisión entre pares.
- Listas de comprobación de la revisión entre pares.
- Criterios de entrada y de salida para los productos de trabajo.
- Criterios para solicitar otra revisión entre pares.
- Material de formación de la revisión entre pares.
- Productos de trabajo seleccionados para revisar.

Subprácticas

- Determinar qué tipo de revisión entre pares será llevada a cabo.
- Definir los requerimientos para recoger datos durante la revisión entre pares.

[2] **COTS** Elementos que pueden comprarse a un vendedor comercial (COTS equivale a "Commercial off the shelf", es decir producto comercial).

- Establecer y mantener criterios de entrada y de salida para la revisión entre pares.
- Establecer y mantener criterios para solicitar otra revisión entre pares.
- Establecer y mantener listas de comprobación para asegurar que los productos de trabajo se revisan consistentemente. Las listas de comprobación se modifican según sea necesario, para tratar el tipo de producto de trabajo específico y la revisión entre pares. Los pares de los desarrolladores de la lista de comprobación y de los usuarios potenciales revisan las listas de comprobación.
- Desarrollar un calendario detallado de la revisión entre pares, incluyendo las fechas para la formación de la revisión entre pares y cuándo estarán disponibles los materiales para la revisión entre pares.
- Asegurar que el producto de trabajo satisface los criterios de entrada de la revisión entre pares antes de su distribución.
- Distribuir a los participantes el producto de trabajo a revisar y su información relativa, con la suficiente antelación para permitir a los participantes prepararse adecuadamente para la revisión entre pares.
- Asignar los roles para la revisión entre pares según sea apropiado.
- Preparar la revisión entre pares revisando el producto de trabajo antes de llevar a cabo la propia revisión entre pares.

SP 2.2 Llevar a cabo revisiones entre pares.

Productos de trabajo típicos

- Resultados de la revisión entre pares.
- Problemas de la revisión entre pares.
- Datos de la revisión entre pares.

Subprácticas

- Llevar a cabo los roles asignados en la revisión entre pares.
- Identificar y documentar los defectos y otros problemas en el producto de trabajo.
- Registrar los resultados de la revisión entre pares, incluyendo los elementos de acción.
- Recoger los datos de la revisión entre pares.
- Identificar los elementos de acción y comunicar los problemas a las partes interesadas relevantes.

- Llevar a cabo una revisión entre pares adicional si el criterio definido indica esta necesidad.
- Asegurar que se satisfacen los criterios de salida para la revisión entre pares.

SP 2.3 Analizar los datos de las revisiones entre pares.

Productos de trabajo típicos

- Datos de la revisión entre pares.
- Elementos de acción de la revisión entre pares.

Subprácticas

- Registrar los datos relativos a la preparación, realización y resultados de la revisión entre pares.
- Almacenar los datos para futura referencia y análisis.
- Proteger los datos para asegurar que los datos de la revisión entre pares no se usan de forma inapropiada.
- Analizar los datos de la revisión entre pares.

SG 3 Verificar los productos de trabajo seleccionados.

SP 3.1 Realizar la verificación.

Productos de trabajo típicos

- Resultados de la verificación.
- Informes de la verificación.
- Demostraciones.
- Registro de los procedimientos tal como se ejecutan.

Subprácticas

- Realizar la verificación de los productos de trabajo seleccionados frente a sus requerimientos.
- Registrar los resultados de las actividades de verificación.
- Identificar los elementos de acción resultantes de la verificación de los productos de trabajo.
- Documentar el método de verificación "tal como se ejecuta" y las desviaciones de los métodos y de los procedimientos disponibles descubiertas durante su realización.

SP 3.2 Analizar los resultados de la verificación.

Productos de trabajo típicos

- Informe de análisis (p. ej., estadísticas de rendimiento, análisis causal de no conformidades, comparación del comportamiento entre el producto real y los modelos, y tendencias).
- Informes de problemas.
- Peticiones de cambio para métodos, criterios y entorno de verificación.

Subprácticas

- Comparar los resultados reales con los resultados esperados.
- En base a los criterios de verificación establecidos, identificar los productos que no han cumplido sus requerimientos o identificar problemas con los métodos, con los procedimientos, con los criterios y con el entorno de verificación.
- Analizar los datos de verificación sobre defectos.
- Registrar todos los resultados del análisis en un informe.
- Usar los resultados de verificación para comparar las mediciones y el rendimiento reales con los parámetros técnicos de rendimiento.
- Proporcionar información sobre cómo pueden resolverse los defectos (incluyendo los métodos de verificación, los criterios y el entorno de verificación) e iniciar las acciones correctivas.

2.2 ISO/IEC 12207

La norma internacional **ISO[3]/IEC[4] 12207** fue publicada el 1 de agosto de 1995. Esta norma internacional estableció 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. Contiene procesos, actividades y tareas que pueden aplicarse durante la

[3] **Organización Internacional para la Estandarización.** Organismo encargado de promover el desarrollo de normas internacionales de fabricación, comercio y comunicación para todas las ramas industriales a excepción de la eléctrica y la electrónica. Su función principal es la de buscar la estandarización de normas de productos y seguridad para las empresas u organizaciones a nivel internacional. La ISO es una red de los institutos de normas nacionales de 163 países, sobre la base de un miembro por país, con una Secretaría Central en Ginebra (Suiza) que coordina el sistema.

[4] **Comisión Electrotécnica Internacional**. Organización encargada de elaborar y publicar las Normas Internacionales para todas las tecnologías eléctricas, electrónicas y relacionadas - conocidos colectivamente como "Electrotecnología". Numerosas normas se desarrollan conjuntamente con la ISO (normas ISO/IEC).

adquisición de un producto o servicio software y durante el suministro, desarrollo, operación, mantenimiento y evolución de productos software.

Esta Norma Internacional también proporciona un proceso que puede ser empleado para definir, controlar y mejorar los procesos del ciclo de vida del software.

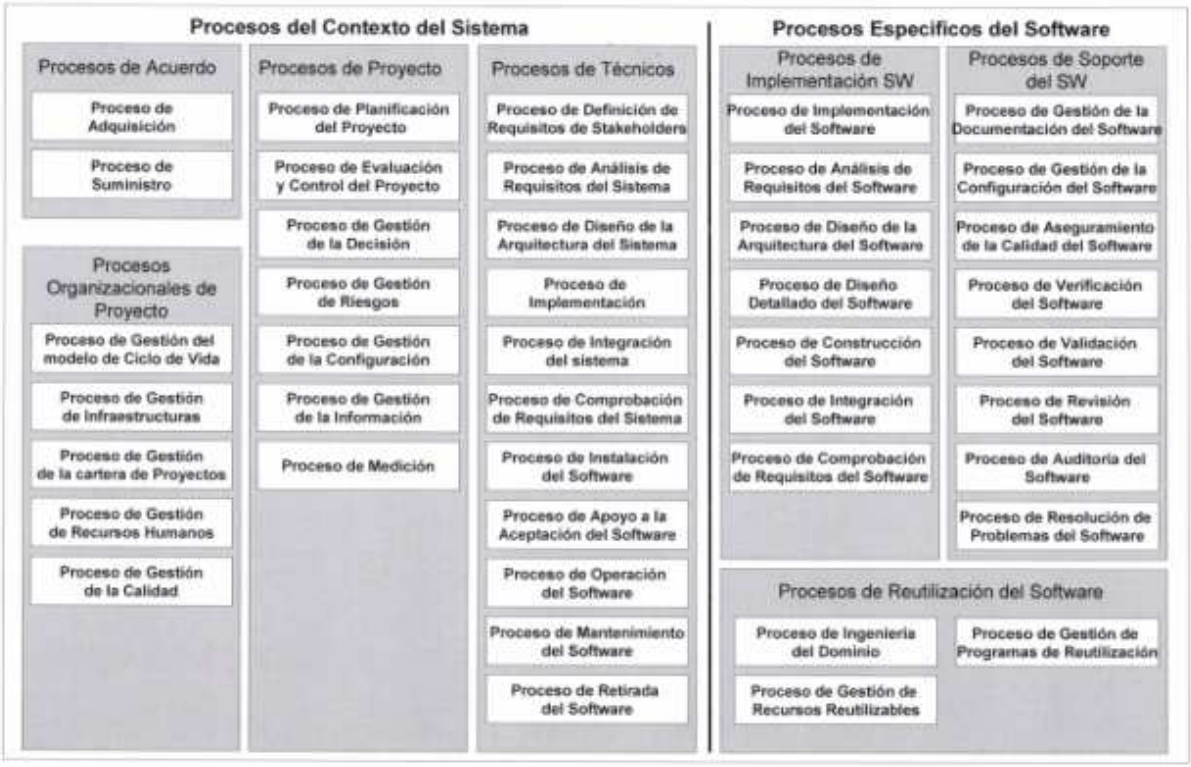

Figura 3. Modelo de procesos ISO/IEC 12207:2008

2.2.1 Procesos que cubren las pruebas

Dentro de los procesos definidos en el Modelo de Procesos de la norma internacional ISO/IEC 12207:2008, 2 cubren el proceso de prueba:

- Proceso de Validación del Software
- Proceso de Verificación del Software

2.2.2 Proceso de Validación del Software

El propósito del Proceso de Validación del Software es confirmar que se cumplen los requisitos para un uso específico previsto del producto trabajo software

2.2.2.1 Resultados

Como resultado de una implementación satisfactoria del Proceso de Validación del Software:

a. Se desarrolla e implementa una estrategia de validación;

b. Se identifican los criterios para la validación de todos los productos de trabajo de software requeridos;
c. Se realizan las actividades de validación requeridas;
d. Se identifican y registran los problemas; y
e. Se demuestra que los productos de trabajo software desarrollados son apropiados para el uso previsto, y
f. Se ponen a disposición del cliente, y otras partes implicadas, los resultados de las actividades definidas.

2.2.2.2 Actividades y tareas

El proyecto llevará a cabo las siguientes actividades y tareas conforme a la política de organización y los procedimientos aplicables en lo que concierne al Proceso de Validación de Software.

i. **Implementación del proceso.** Esta actividad consta de las siguientes tareas:
   a. Se deberá determinar si el proyecto merece un esfuerzo de validación y el grado de independencia organizativa necesaria para dicho esfuerzo.
   b. Si el proyecto merece un esfuerzo de validación, se deberá establecer un proceso de validación para validar el sistema o el producto software. Se deberán seleccionar las tareas de validación definidas más adelante, incluyendo los métodos, técnicas y herramientas asociadas
   c. Si el proyecto merece un esfuerzo independiente, se deberá seleccionar una organización calificada responsable de llevar a cabo este esfuerzo. Se deberá garantizar a esta organización la independencia y autoridad para llevar a cabo las actividades de validación.
   d. Se deberá preparar y documentar un plan de validación. El plan deberá incluir (sin estar limitado a ello) lo siguiente:
      - Elementos sujetos a validación.
      - Tareas de validación a llevar a cabo.
      - Recursos, responsabilidades y plazos para la validación.
      - Procedimientos para hacer llegar los informes de validación al adquiriente y a otras partes
   e. Se deberá implementar el plan de validación. Los problemas y las no conformidades detectadas por el esfuerzo de validación se deberán pasar al proceso de solución de problemas. Se deberán resolver todos los problemas y

no conformidades. Se deberá poner a disposición del adquiriente y otras organizaciones involucradas los resultados de las actividades de validación

ii. **Validación**. Esta actividad consta de las siguientes tareas:

NOTA Para la validación se pueden emplear otros significados (tales como, análisis, modelización, simulación, etc.), además de las pruebas.

a. Preparar los requerimientos de prueba, casos de prueba y especificaciones de prueba seleccionados para analizar los resultados de las pruebas.
b. Asegurar que estos requerimientos de prueba, casos de prueba y especificaciones de prueba reflejan los requerimientos particulares para el uso específico previsto.
c. Llevar a cabo las pruebas de los apartados anteriores, incluyendo:
   - Pruebas con sobrecarga, límites y entradas excepcionales.
   - Pruebas del producto software respecto a su habilidad para aislar y minimizar el efecto de errores; esto es, degradación elegante por fallos, petición de asistencia del operador ante sobrecargas y situaciones límite y excepcionales.
   - Pruebas de usuarios representativos que pueden llevar a cabo con éxito sus tareas previstas usando el producto software.
d. Validar que el producto software satisface su uso previsto.
e. Probar el producto de software, cuando sea apropiado, en áreas seleccionadas del entorno de destino.

2.2.3 Proceso de Verificación del Software

El propósito del proceso de Verificación del Software es confirmar que cada producto de trabajo software y/o servicio de un proceso o proyecto refleja correctamente los requisitos especificados.

2.2.3.1 Resultados

Como resultado de una implementación satisfactoria del Proceso de Verificación del Software:

a. Se desarrollará e implementará una estrategia de verificación;
b. Se identificaran los criterios para la verificación de todos los productos de trabajo de software requeridos;
c. Se realizaran las actividades de verificación requeridas;

d. Se identificaran y registraran los defectos; y
e. Se pondrá a disposición del cliente, y otras partes implicadas, los resultados de las actividades de verificación.

2.2.3.2 Actividades y tareas

El proyecto llevará a cabo las siguientes actividades y tareas conforme a la política de la organización y los procedimientos aplicables en lo que concierne al Proceso de Verificación de Software.

i. **Implementación del proceso.** Esta actividad consta de las siguientes tareas:

a. Se deberá determinar si el proyecto requiere un esfuerzo de verificación y el grado de independencia organizativa necesaria para dicho esfuerzo. Se deberá analizar los aspectos críticos de los requerimientos del proyecto. Los aspectos críticos se deberán evaluar en términos de:
   - La probabilidad de que un error no detectado en los requerimientos del sistema o del software cause muerte o daños personales, fracaso del proyecto, pérdida financiera o pérdida catastrófica o daño a equipos.
   - Madurez y riesgos asociados con la tecnología software usada.
   - Disponibilidad de fondos y recursos.
b. Si el proyecto merece un esfuerzo de verificación, se deberá establecer un proceso de verificación para verificar el producto software.
c. Si el proyecto merece un esfuerzo de verificación independiente, se deberá seleccionar una organización calificada responsable de llevar a cabo la verificación. Se deberá garantizar a esta organización la independencia y autoridad para llevar a cabo las actividades de verificación.
d. Basándose en el análisis anterior sobre el alcance, magnitud, complejidad y aspectos críticos, se deberán determinar las actividades del ciclo de vida y los productos software que requieren verificación. Para estas actividades del ciclo de vida y productos software se deberá seleccionar las actividades y tareas de verificación definidas en el apartado 2.2.3.1, incluyendo los métodos, técnicas y herramientas asociadas para llevarlas a cabo.
e. Basándose en las tareas de verificación seleccionadas, se deberá preparar y documentar un plan de verificación. El plan deberá tener en cuenta las actividades del ciclo de vida y productos software sujetos a verificación, las tareas de verificación requeridas para cada actividad del ciclo de vida y producto software y

los recursos, responsabilidades y plazos asociados. El plan deberá tener en cuenta procedimientos para hacer llegar los informes de la verificación al adquiriente y a otras organizaciones involucradas.

f. Se deberá implementar el plan de verificación. Los problemas y no conformidades detectadas por el esfuerzo de verificación se deberán pasar al proceso de solución de problemas. Se deberán resolver todos los problemas y no conformidades. Se deberá poner a disposición del adquiriente y otras organizaciones involucradas los resultados de las actividades de verificación.

ii. **Verificación**. Esta actividad consta de las siguientes tareas:

- **Verificación de los requisitos**. Los requisitos deberán ser verificados teniendo en cuenta los criterios enumerados a continuación:
  a. Los requerimientos del sistema son consistentes, viables y se pueden probar.
  b. Los requerimientos del sistema han sido adecuadamente asignados a elementos hardware, elementos software y operaciones manuales de acuerdo con los criterios de diseño.
  c. Los requerimientos software son consistentes, viables, se pueden probar y reflejan fielmente los requerimientos del sistema.
  d. Los requerimientos software relacionados con seguridad física y de acceso y otros requerimientos críticos son correctos, según demuestran métodos rigurosos v adecuados.
- **Verificación del diseño**. El diseño deberá ser verificado teniendo en cuenta los criterios enumerados a continuación:
  a. El diseño es correcto, consistente con los requerimientos y trazable hacia ellos.
  b. El diseño implementa la secuencia correcta de eventos, entradas, salidas, interfaces, flujo lógico, asignación de sincronizaciones y tamaños y definición, aislamiento y recuperación ante errores.
  c. El diseño seleccionado se puede derivar de los requerimientos.
  d. El diseño implementa correctamente los requerimientos de seguridad física y de acceso y otros requerimientos críticos, según demuestran métodos rigurosos y adecuados.

- **Verificación del código**. El código deberá ser verificado teniendo en cuenta los criterios enumerados a continuación:
  a. El código es trazable hacia el diseño y los requerimientos, se puede probar, es correcto y cumple con los requerimientos y normas de codificación.
  b. El código implementa la secuencia correcta de eventos, interfaces consistentes, flujo correcto de datos y control, completitud, una adecuada asignación de sincronizaciones y tamaños y definición, aislamiento y recuperación ante errores.
  c. El código seleccionado se puede derivar del diseño o de los requerimientos.
  d. El código implementa correctamente los requerimientos de seguridad física y de acceso y otros requerimientos críticos, según demuestran métodos rigurosos y adecuados.
- **Verificación de la integración.** La integración deberá ser verificada teniendo en cuenta los criterios enumerados a continuación:
  a. Los componentes y unidades software de cada elemento software han sido integrados correcta y completamente en el elemento software.
  b. Los elementos hardware, elementos software y operaciones manuales del sistema han sido completa y correctamente integrados en el sistema.
  c. Las tareas de integración se han llevado a cabo de acuerdo con un plan de integración.
- **Verificación de la documentación**. Se deberá verificar la documentación teniendo en cuenta los criterios enumerados a continuación:
  a. La documentación es adecuada, completa y consistente.
  b. La preparación de la documentación se hace a su debido tiempo.
  c. La gestión de la configuración de los documentos sigue procedimientos especificados.

2.3 TMM y TMMI

TMM (*Test Maturity Model*) es un modelo de referencia centrado en el Proceso de Pruebas creado por el Instituto de Tecnología de Illinois en 1996. La mejora del proceso de pruebas es soportada por una única área de proceso que define un conjunto de niveles y objetivos de madurez en el TMM. Propone 5 niveles de madurez y describe una estructura interna implementada en cada uno de ellos, tal y como se ve en la figura 4.

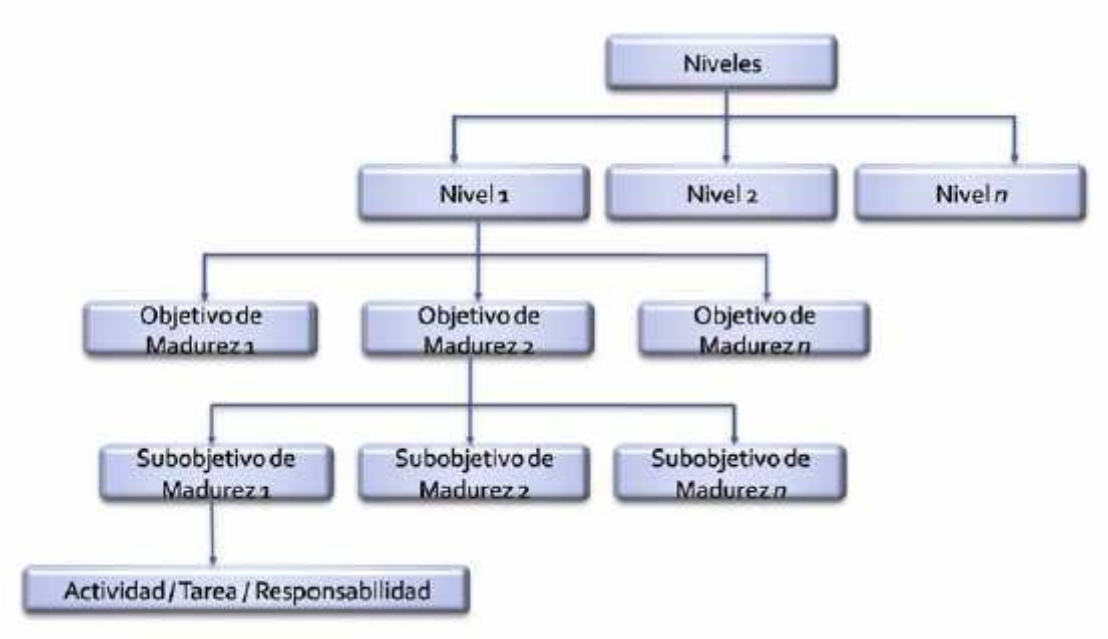

Figura 4. Estructura de TMM

TMMI (Test Maturity Model Integration) es un modelo detallado para mejorar el proceso de prueba, desarrollado por el TMMI Foundation, es la evolución de TMM (Test Maturity Model), ambos muy en la línea CMMI definen cinco niveles de madurez específicos para pruebas.

El desarrollo de TMMI ha utilizado el marco TMM desarrollado por el Instituto de Tecnología de Illinois como una de sus principales fuentes. Además de TMM, el desarrollo fue, en gran parte, guiado por el trabajo realizado sobre CMMI, un modelo de mejora de procesos que tiene amplio apoyo en la industria de TI.

TMMI proporciona un enfoque más amplio y estructurado al proceso de prueba, existe un gran número de experiencias positivas con el modelo TMMI y las organizaciones que la utilizan van en aumento, las experiencias muestran que aplicar el modelo de madurez TMMI tiene un impacto positivo sobre la calidad del producto.

De manera similar a CMMI, TMMI nos presenta una mejora en los procesos relativos a la calidad del software y define también 5 niveles de madurez o jerarquía evolutiva.

Cada nivel de madurez contiene un conjunto de áreas de proceso:

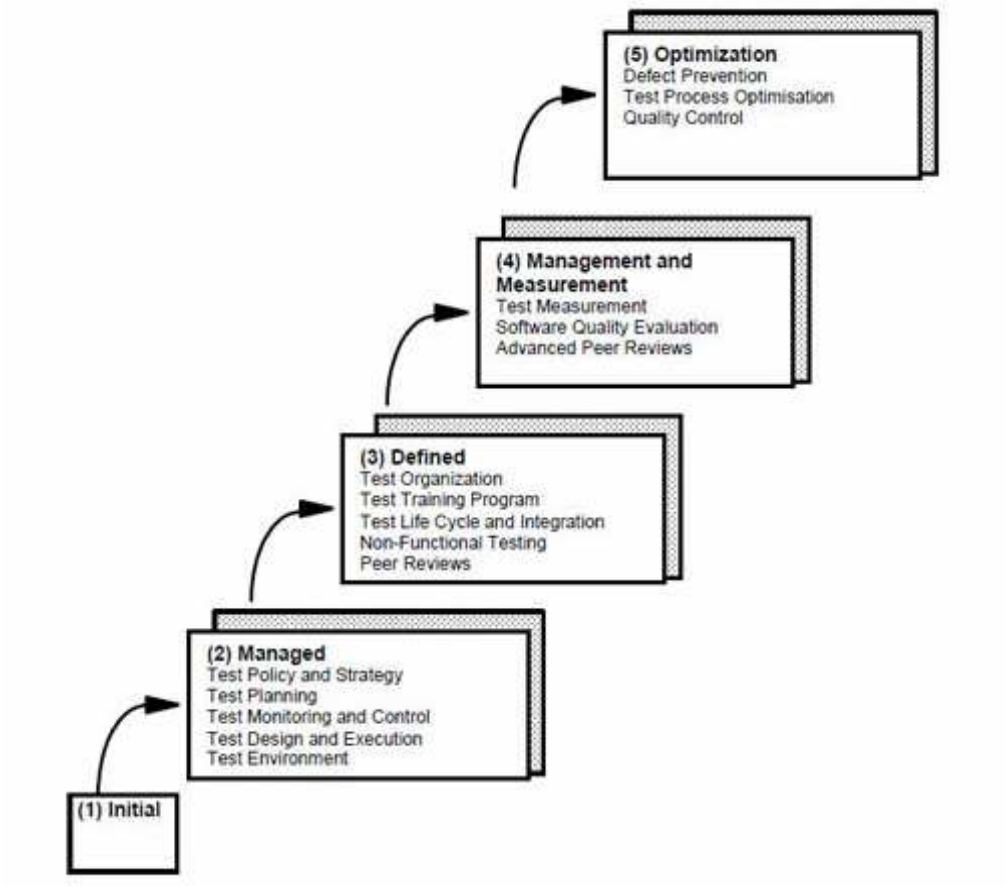

Figura 5. Los 5 niveles de TMMI

La estructura interna del TMMI contiene prácticas de pruebas que pueden ser aprendidas y aplicadas sistemáticamente para apoyar la mejora de calidad gradualmente. Hay 5 niveles en el TMMI que definen la jerarquía de madurez y el camino de evolución a la mejora del proceso de pruebas.

La figura 6 muestra su estructura:

Figura 6. Estructura de TMMI

TMMI no es sólo un modelo teórico, sino un conjunto bien definido de áreas proceso y objetivos basados sobre casos de estudio prácticos. Aunque el modelo es bastante fácil de entender, su aplicación en una organización no siempre es una tarea simple.

2.3.1 Niveles TMMI

Nivel 1. Iniciación

En el nivel de TMMI 1, la prueba es un caótico proceso indefinido y se considera a menudo una parte de la depuración. La organización no suele proporcionar un entorno estable para apoyar los procesos. El éxito en estas organizaciones depende de la competencia y el heroísmo de la gente en la organización y no del uso de procesos prueba. Las pruebas se desarrollan una vez que la codificación se ha completado, intercalándolas con la depuración para detectar errores del sistema. El objetivo de las pruebas a este nivel es mostrar que el software se ejecuta sin errores importantes. Los productos se entregan sin la visibilidad adecuada en materia de calidad y riesgos. Además, no suelen cumplir con sus necesidades, no son estables, o son demasiado lentos para trabajar. En el nivel de madurez 1 no hay áreas de proceso definidas y las organizaciones se caracterizan por una tendencia a cometer, cada vez más, el abandono de los procesos en un momento de crisis, y la incapacidad para repetir sus éxitos. Además, los productos tienden a no ser entregados a tiempo, los presupuestos están invalidados y la calidad no es acorde con las expectativas.

Nivel 2. Planificación

En el nivel de TMMI 2, las pruebas se convierten en un proceso gestionado y claramente separado de la depuración. Sin embargo, aún se perciben como una fase del proyecto que sigue a la codificación. En el contexto de la mejora del proceso de pruebas, se establece una estrategia y un plan de pruebas. Dentro del plan de pruebas se define un enfoque de pruebas, el cual se basa en el resultado de una evaluación del riesgo del producto. Se utilizan técnicas de gestión de riesgos para identificar los riesgos, basados en las necesidades documentadas, de los productos. El plan de pruebas define que se va a probar, cuándo, cómo y por quién se hará. Las pruebas se supervisan y controlan para asegurar que van de acuerdo al plan y qué acciones se pueden tomar si se producen desviaciones. Existen pruebas de múltiples niveles: de unidad, de integración, de sistema y de aceptación. Para cada nivel de prueba identificado hay objetivos específicos definidos en la estrategia de prueba. El principal objetivo de las pruebas en el nivel TMMI 2 es verificar que el producto cumple los requisitos especificados. Además de la necesidad de diferenciar los procesos de prueba y depuración.

Las áreas de proceso del nivel TMMI 2 son:

2.1 Política y Estrategia de Pruebas

2.2 Planificación de Pruebas

2.3 Seguimiento y Control de Pruebas

2.4 Diseño y Ejecución de Pruebas

2.5 Entorno de Pruebas

Nivel 3. Definición

En el nivel de TMMI 3, las pruebas ya no son una fase posterior a la codificación. Están totalmente integradas en el ciclo de vida del desarrollo. La planificación de las pruebas se realiza en una etapa temprana del proyecto, a través de un plan de pruebas principal. Se establece y mejora el conjunto de la organización de los procesos de prueba estándar, que son la base para el nivel de madurez 3. Se crea una organización de pruebas y un programa específico de formación para las pruebas y se lleva a cabo un programa formal de revisión, aunque aún no está totalmente vinculado al proceso de pruebas dinámico. El diseño de las pruebas del nivel 2 TMMI se centran principalmente en pruebas de funcionalidad, diseño de pruebas y técnicas de pruebas se amplían, en función de los objetivos de negocios, para incluir también las pruebas no funcionales, por ejemplo, de usabilidad y / o fiabilidad. Una distinción fundamental entre el nivel de madurez TMMI 2 y 3 es el alcance de las normas, descripciones de procesos y procedimientos. En el nivel de madurez 2, estas pueden

ser muy diferentes en cada caso en particular. En este nivel las normas, descripciones de procesos y procedimientos están adaptadas con el conjunto de la organización de procesos estándar para ajustarse a un determinado proyecto o unidad organizativa y por lo tanto son más consistentes. Además, los procesos se describen con más detalle que en el nivel de madurez 2. Como consecuencia de ello en el nivel de madurez 3, la organización debe revisar las áreas de proceso del nivel de madurez 2.

Las áreas de proceso del nivel TMMI 3 son:

3.1 Organización de Pruebas

3.2 Programa de Formación para las Pruebas

3.3 Ciclo de vida de las Pruebas e Integración

3.4 Pruebas No Funcionales

3.5 Revisión de pares

Nivel 4. Gestión y Medición

En el nivel TMMI 4 la organización de las pruebas es un proceso bien definido, bien fundado y medible. La organización y los proyectos establecen los objetivos cuantitativos para la calidad del producto y el funcionamiento del proceso y los utilizan como criterios para dirigirlos. La calidad del producto y el funcionamiento del proceso se entienden en términos estadísticos y se gestionan a lo largo de todo el ciclo de vida. Estas medidas se incluyen en el repositorio de medidas de la organización para apoyar la toma de decisiones. Las revisiones e inspecciones se consideran parte del Proceso de pruebas y se utilizan como medio para realizar el control de calidad. Las pruebas estáticas y dinámicas se integran como una sola. Los productos se evalúan según los criterios cuantitativos de los atributos de calidad tales como fiabilidad, facilidad de uso y mantenimiento. Las pruebas se perciben como una evaluación, que consta de todas las actividades del ciclo de vida relacionadas con los productos probados y los productos de trabajo correspondientes.

Las áreas de proceso del nivel TMMI 4 son:

4.1 Medidas de Pruebas

4.2 Evaluación de la Calidad de productos

4.3 Revisión avanzada de pares

Nivel 5 Optimización

Sobre la base de todos los resultados que se han logrado mediante el cumplimiento de todos los objetivos de mejora de los niveles de madurez anteriores, las pruebas son ahora un proceso completamente definido y se puede controlar los costes y la eficacia asociados a estas. En el nivel de madurez TMMI 5, la prueba es un proceso con el objetivo de prevenir los defectos. La optimización del funcionamiento del proceso de pruebas se lleva a cabo mediante un proceso gradual e innovadoras y mejoras tecnológicas. Los métodos y técnicas se han optimizado y hay una atención constante a la puesta a punto y mejora de procesos de prueba. Se realiza prevención de defectos y control de calidad. El muestreo estadístico, las medidas de niveles de confianza, unidad, y la fiabilidad conducen el proceso de prueba. Existe un procedimiento detallado para la selección y evaluación de herramientas de prueba. Estas herramientas apoyan al proceso de prueba tanto como sea posible durante el diseño de las pruebas, la ejecución de pruebas, pruebas de regresión, gestión de casos de prueba, etc. El proceso de reutilización también se practica en el nivel 5 con el apoyo de una biblioteca de activos del proceso.

Las áreas de proceso en el nivel 5 son:

5.1 Prevención de defectos

5.2 Optimización del Proceso de Pruebas

5.3 Control de Calidad

2.4 TPI

El modelo **Test Process Improvement (TPI)** se desarrolló basándose en el conocimiento y la experiencia de SOGETI. Este modelo asiste en la mejora de los procesos de pruebas dentro de la organización. Basándose en esto, el modelo permite definir pasos de mejora controlables y graduales.

El esquema de dicho modelo es el siguiente:

Figura 7. Esquema del modelo TPI

TPI tiene en cuenta los diversos aspectos del proceso de pruebas, tales como el uso de herramientas de test, las técnicas de diseño de pruebas o la realización de informes. En su análisis, se hacen visibles los puntos fuertes y débiles del proceso de pruebas. Estos aspectos se denominan áreas clave. El modelo TPI cuenta con 20 áreas clave.

Para clasificar el estado utiliza niveles (que van de A a B a C, etc.), identificando la madurez de cada uno de ellos. Un nivel más alto implica más eficiencia desde el punto de vista de tiempo, de coste y/o de calidad.

Por ejemplo, para el área clave de *Reporting*, los niveles son:

a. Se realizan informes de los defectos encontrados,
b. Se realizan informes del progreso de las pruebas y de la calidad del sistema,
c. Se definen los riesgos para el progreso y para el sistema, y se proporcionan recomendaciones, sostenidas por métricas,
d. Las recomendaciones de las pruebas se extienden hacia la mejora del proceso de desarrollo.

Todos los niveles y áreas clave están interrelacionados en la **Matriz de Madurez** del proceso de pruebas. Esta proporciona una manera adecuada para definir las prioridades internas, teniendo en cuenta las dependencias entre las áreas clave y los niveles correspondientes. En la matriz, cada nivel está relacionado con una determinada escala de madurez del proceso de pruebas. En total, hay 13 niveles de madurez para el proceso de pruebas total. Las áreas clave, los niveles y la matriz se utilizan para identificar los puntos fuertes y débiles del proceso de pruebas actual, para definir el nivel de madurez que queremos

alcanzar, y para establecer las acciones para llevar a cabo la mejora. El uso de la matriz permite seleccionar y evaluar propuestas de mejora con mayor facilidad.

	Áreas clave / Niveles	0	1	2	3	4	5	6	7	8	9	10	11	12	13
1	Estrategia de test		A					B				C		D	
2	Uso ciclo de vida		A			B									
3	Momento de participación			A				B				C		D	
4	Estimación y planificación				A							B			
5	Técnicas de especificación de test		A		B										
6	Técnicas de test estáticas					A		B							
7	Métricas														D
8	Automatización de testing														
9	Entorno de test														C
10	Entorno de oficina				A										
11	Dedicación y motivación		A				B						C		
12	Funciones de test y formación				A			B			C				
13	Campo de aplicación de la metodología					A						B			C
14	Comunicación			A		B							C		
15	Reporting		A			B		C					D		
16	Gestión de defectos		A												
17	Gestión de Testware			A											D
18	Gestión del proceso de test		A		B								C		
19	Evaluación							A			B				
20	White Box testing					A		B		C					

Figura 8. Matriz de Madurez

Los requisitos de cada nivel están definidos en los puntos de control, que se deben cumplir para acceder al nivel correspondiente. Por tanto, los puntos de control proporcionan una mejora objetiva de clasificación. El modelo ofrece sugerencias (consejos y mejoras prácticas) que se puede aplicar para optimizar el proceso de pruebas.

El modelo se debe de ver como una herramienta para la mejora del proceso de pruebas, y es una buena vía de comunicación en la gestión del cambio.

La mejora de los procesos requiere un alto grado de conocimiento y experiencia. Se requieren especialistas en diferentes áreas de testing, organización y procesos de cambio en sí mismo.

2.5 TestPAI

TestPAI es un área de proceso de pruebas integrado con CMMI. Se encuentra situada en el nivel 3 con los procesos de ingeniería. TestPAI ha sido desarrollado para proporcionar el marco de trabajo necesario para que la mejora del proceso de pruebas se desarrolle de forma paralela a la mejora de otros procesos implementados en la organización.

Este trabajo está soportado por el proyecto *Buen Gobierno de las Tecnologías de la Información* financiado por ATOS-ORIGIN S.A.

Cada vez hay más pequeñas y medianas empresas que están adquiriendo conciencia a cerca de la importancia del proceso de prueba y la mejora del mismo. Muchas de ellas están demandando un marco de trabajo que les proporcione las herramientas necesarias para realizar una mejora del proceso de pruebas de un modo poco costoso y sencillo. Además, muchas de esas organizaciones se encuentran trabajando con CMMI para mejorar otros procesos implementados, por lo que necesitan que dicho marco de trabajo sea totalmente compatible e integrable con CMMI. Así, TestPAI surge para proporcionar una solución a este problema.

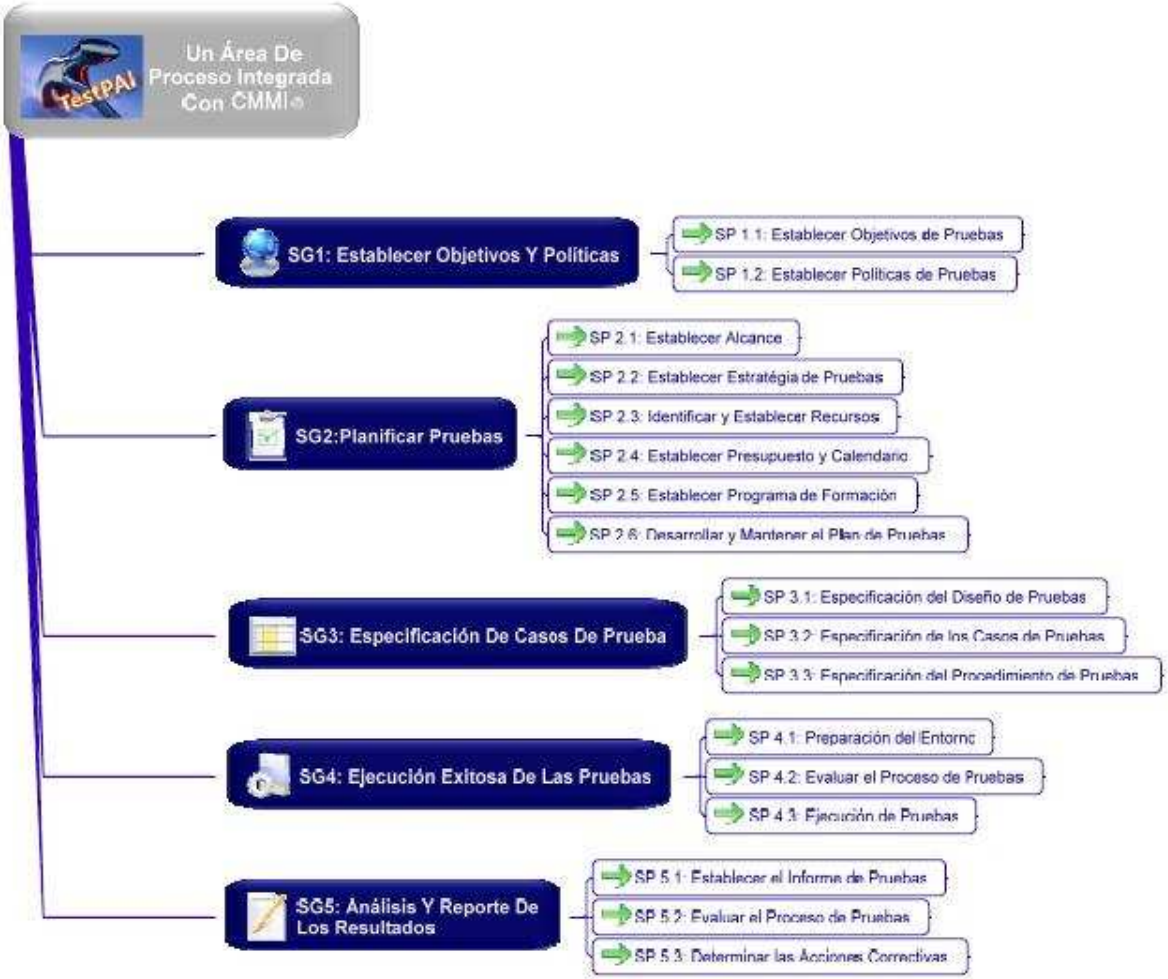

Figura 9. Objetivos y Prácticas Específicas de TestPAI

TestPAI incluye y define todas las prácticas relativas a las pruebas. Además, tiene la misma estructura que CMMI y contempla tanto la representación continua como la representación por etapas. Todo esto permite una sencilla y completa integración entre TestPAI y CMMI. TestPAI define cinco objetivos específicos (SG) y sus correspondientes prácticas específicas (SP), las cuales se muestran en la Figura 9.

El propósito del primer objetivo específico es definir los objetivos y políticas asociados a las pruebas. Se establecerán los objetivos de pruebas y la política de pruebas que permita alcanzar los objetivos definidos. La figura 10 ilustra este objetivo.

Figura 10. Prácticas Específicas del SG1

El propósito del segundo objetivo específico es desarrollar y mantener el Plan de Pruebas. La planificación implica establecer y analizar los diferentes elementos que afectan a la gestión y desarrollo de las pruebas tales como definir alcance, analizar riegos, definir la estrategia o asignar recursos entre otros. La Figura 11 muestra las seis prácticas específicas que contiene éste objetivo.

Figura 11. Prácticas Específicas del SG2

El propósito del tercer objetivo específico es establecer la información necesaria para realizar una ejecución efectiva de las pruebas. Un proceso de pruebas bien definido implica determinar características que no se encuentran descritas en el Plan de Pruebas, como pueden ser las entradas, salidas o resultados esperados de una prueba. La figura 12 muestra las prácticas específicas de este objetivo específico.

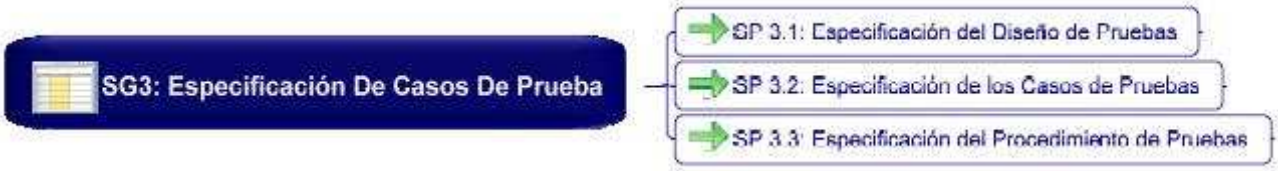

Figura 12. Prácticas Específicas del SG3

El propósito del cuarto objetivo es proporcionar la infraestructura adecuada para realizar las pruebas. Una ejecución de pruebas satisfactoria es un elemento importante en la mejora de la calidad del software, y preparar la infraestructura correctamente contribuye a que la ejecución se desarrolle de forma disciplinada. La figura 13 muestra las prácticas específicas de este objetivo específico.

Figura 13. Prácticas Específicas del SG4

El propósito del quinto objetivo es analizar, evaluar y comunicar los resultados obtenidos, con el objetivo de poder evaluar el proceso de pruebas y la calidad del producto. Contiene tres prácticas específicas que se muestran en la Figura 14.

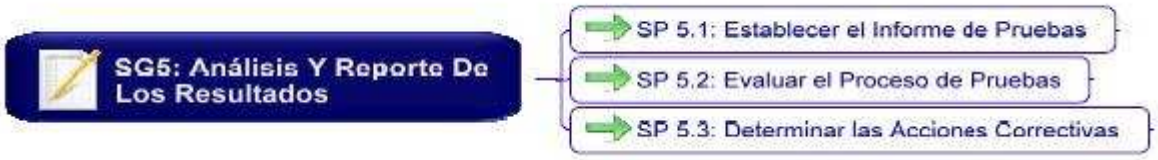

Figura 14. Prácticas Específicas del SG5

2.6 TMAP

TMap (Test Managment Aproach) es un marco de trabajo publicado por Sogeti en 1995 y actualizado como TMAP Next en 2006 y se encuentra difundido por toda la comunidad europea en países como Alemania, Francia, Suecia, Suiza, España, Bélgica, Reino Unido e inclusive Estados Unidos. TMap está basado en Metodologías Estructuradas de Testing.

Al tratarse TMap de un método estructurado, se deben reconocer cinco fases bien definidas:

- **Planificación y Control:** Especificación de "Quién", "Qué", "Cuándo", "Cómo" y "Dónde" probar.
- **Preparación:** Determinación de las especificaciones de calidad del software para obtener una exitosa ejecución del testing.
- **Especificación:** Indica los Test Case involucrados y la infraestructura requerida, incluidas las configuraciones necesarias, lo cual permitirá que la ejecución del Testing comience en cuanto el "Objeto de Testing".

En esta fase se analiza cualquier diferencia entre los resultados esperados y los obtenidos y finalmente se reportan los defectos.

- **Ejecución:** La fase de ejecución inicia cuando el "Objeto de Testing" es liberado.
- **Completitud:** Esta fase consiste en el *reporting* de los resultados evaluación de los procesos de prueba para la mejora del mismo proceso, como también la observación de los "Materiales de Test" que pueden ser reutilizados para proyectos futuros o inclusive para fases de mantenimiento y/o soporte de los productos generados por el proyecto.

TMap ofrece una manera de emprender de forma efectiva y eficiente los procesos de pruebas de software, permitiendo a las organizaciones enfocarse en los objetivos claves para sus negocios.

Lo hace de manera efectiva para que se enfoque en encontrar a tiempo los defectos importante y con una relación directa con el riesgo del producto.

Lo hace de manera eficiente porque es un método aplicable universalmente que enfatiza el re-uso, con una estrategia basada en los riesgos de los objetivos de negocios y de los productos, permitiendo hacer estrategias inteligentes sobre qué testear, cómo testear y cuándo testear, en lugar de testear todo en todo momento.

2.6.1 Los 4 elementos esenciales de TMap

TMap es una metodología de Testing en torno a 4 elementos esenciales:

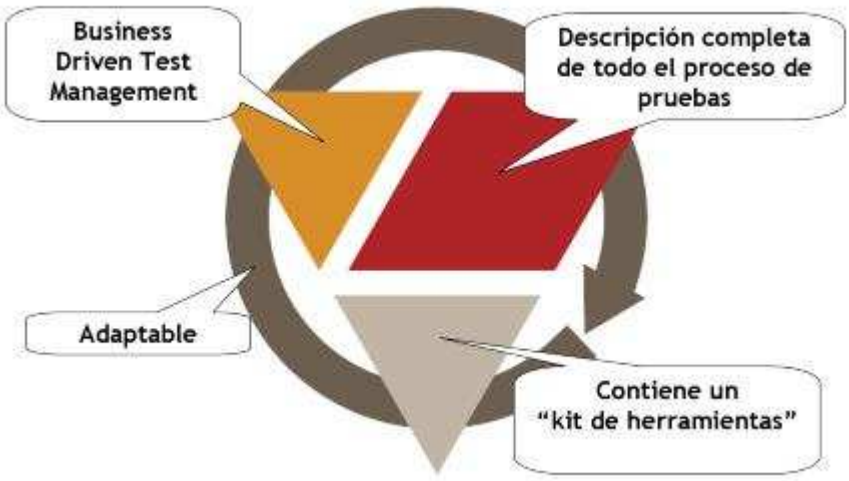

Figura 15. Elementos esenciales de TMap

1. **Business-driven Test Management.** Sus principales características son:
   - el cliente debe poder influenciar los 4 aspectos de control: resultados, riesgos, tiempo y costes;
   - el seguimiento y control del testing se comunica en el lenguaje del cliente en forma de objetivos;
   - los test se basan en los riesgos del producto; y
   - los resultados del testing se hacen visibles.

   El *Business-driven Test Management* consta de los siguientes pasos:
   - Formular la misión y recolectar objetivos de las pruebas.
   - Determinar las categorías de riesgo.
   - Determinar cobertura alta/baja.
   - Se realiza una estimación global para las pruebas y la planificación creadas.
   - Asignar técnicas de pruebas.
   - Proporcionar al cliente información sobre el proceso de test.

2. **Proceso de Test Estructurado.** Consiste en:

- **Master Test Plan**. Se compone de 2 fases: planificación de la totalidad del proceso de test; y control de la totalidad del proceso de test. Su principal objetivo es detectar los defectos más importantes con el menor coste y tiempo posibles.
- **Test de Aceptación y Sistema**: el modelo de Ciclo de Vida (ver figura 16). Este modelo es aplicable a todos los niveles de test, y permite coordinar y sincronizar estos niveles. Transmite responsabilidades y tareas entre grupos involucrados, además de dividir el proceso de test en diferentes fases, actividades y productos.
- **Tests de Desarrollo**. Son parte integral del trabajo realizado por los programadores. El modelo de ciclo de vida (ver figura 16) de TMap Next puede ayudar a identificar y describir las diferentes actividades para los test de desarrollo.
- **Procesos de soporte**. Muchas veces resulta más eficiente centralizar los procesos de soporte de test que implementarlos a nivel de proyecto. El soporte de test puede ponerse en práctica de dos formas diferentes, *Test Expertise Centre* (TEC), o bien *Test Factory* (TF).

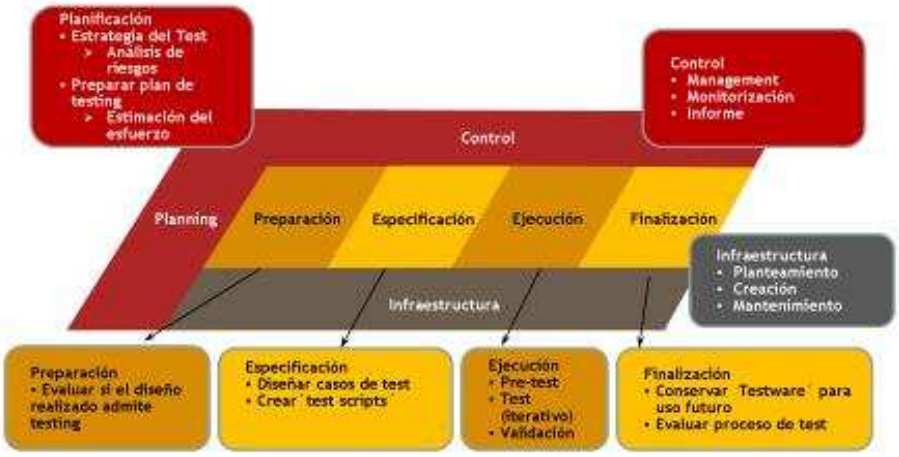

Figura 16. Ciclo de vida de TMap

3. **Kit de herramientas.** Se centra en las técnicas (*¿cómo?*), la infraestructura (*¿dónde? ¿con qué?*) y la organización (*¿quién?*).

- Las **técnicas** incluyen:
  - Estimación de Test
  - Gestión de Defectos
  - Creación de Métricas
  - Análisis de Riesgos del Producto
  - Diseño de Test
  - Evaluación del Producto
- La **infraestructura**. Se definen los entornos de test: DTAP (desarrollo, test, aceptación y producción). Se utilizan herramientas de test que incrementan la productividad, mejoran la calidad del test, y ofrecen más opciones. Y, por último, se definen las condiciones del lugar de trabajo.
- La **organización**. Se establecen los roles y responsabilidades, las tareas y las autorizaciones. El tester debe reunir las siguientes características: ser buen comunicador, poseer capacidad de análisis, ser objetivo, poseer una actitud crítica positiva y ser creativo.

4. **Adaptable.** La adaptabilidad de TMap Next se resume en:

- **Respuesta a los cambios**. En TMap Next, la respuesta a los cambios sucede en las primeras actividades del Master Test Plan. La obtención de información sobre el entorno en el que se ejecuta el test y el establecimiento de posibles cambios desempeñan un papel importante. Si la estrategia de test no es aceptada por el cliente, el plan se adapta.

- **(Re) utilización de productos y procesos**. TMap Next ofrece la posibilidad de utilizar productos y procesos rápidamente, gracias a sus técnicas de diseño, checklists, plantillas, etc. En la fase final, se definen las actividades para identificar la reutilización y preservación de los objetos de test (*testware*).
- **Aprendizaje desde la experiencia**. Al ser un método, TMap Next ofrece el marco necesario para aprender y aplicar lo que ya se ha utilizado con anterioridad. La actividad de evaluación del proceso de test forma parte del Modelo de Ciclo de Vida. Las métricas también son importantes para mejorar el proceso de test de forma continua.
- **Probar antes de usar**. El mejor instrumento que TMap Next ofrece para probar antes de usar son los ensayos de test (*intake*). Otra forma de probar es *customizar* TMap Next para adaptarse a una situación específica.

2.7 IT Mark

IT Mark diseñado por el European Software Institute (ESI) y sus Socios de la Alianza de los ESI Centers, es el primer modelo de calidad internacional diseñado específicamente para las pequeñas y medianas empresas. Es un modelo escalable y muy adecuado para las Pequeñas y Micro Empresas del sector TIC.

Tiene como objetivo brindar un sello de calidad para PyMEs[5] de Tecnologías de la Información, que acredita su madurez y capacidad. También tiene como objetivo mejorar la efectividad organizacional y el éxito en el mercado mediante la mejora de sus procesos. El servicio proporciona conocimiento sobre las capacidades técnicas y gerenciales de la organización a través de la identificación de fortalezas y debilidades, así como del descubrimiento áreas a mejorar de acuerdo con sus metas de negocio.

La comercialización de I.T. Mark se realiza, de momento, en el área de actividad de los ESI Centers hasta ahora constituidos (Brasil, México, Bulgaria, Australia y China), y en los 50 países del mundo en los que tiene cobertura la Red ESI@net.

[5] **Acrónimo de Pequeña y Mediana Empresa**. Conjunto de empresas que, dentro de la CEE, se considera no superan dos de los tres requisitos siguientes: número de empleados (50 para pequeña, 250 para mediana), ventas anuales en millones de UCE (dos para pequeña, 8 para mediana), y activos totales en millones de UCE (uno para pequeña empresa y cuatro para mediana).

2.7.1 Perspectivas desde las que trabaja

Trabaja desde tres perspectivas de la empresa:

a. La **Gestión General** de la empresa, de acuerdo con el modelo 10-squared, y que estudia diez categorías de procesos como son estratégica, comercial, financiera, definición de productos y servicios, conocimiento del mercado, marketing, etc., hasta obtener una visión exhaustiva de la empresa. Cada una de estas categorías tiene en cuenta diez elementos, entre los cuales existen algunos elementos críticos, en función del estado de desarrollo de la empresa: Semilla, Start-up, Desarrollo o Expansión.

b. La **Seguridad de la Información**, basada en la norma ISO/IEC-17799:2005, en la que IT Mark define varios niveles:

   - Centrado en la organización de la seguridad, responsabilidades, requisitos legales (Protección de Datos Personales, etc.), y controles de seguridad
   - Que exige que la gestión de la seguridad se haya convertido en un proceso estandarizado en la organización
   - Mejora Continua del Sistema de Gestión de la Seguridad de la Información

c. Los **Procesos de desarrollo de Software y Sistemas** y núcleo del modelo, basado en CMMI.

2.7.2 Niveles IT Mark

El esquema I.T. Mark distingue tres niveles posibles, y progresivamente más exigentes, en función de la Madurez demostrada en los procesos de cada PYME. Así:

- **I.T. Mark** acredita a una empresa que es consciente de los problemas relacionados con la gestión técnica, de seguridad y del negocio, y que los mantiene habitualmente bajo control. Para ello se admite que algunas de las áreas de proceso puedan no estar suficientemente elaboradas, trabajándose en el caso particular de los Procesos Técnicos sobre CMMI Nivel 2 y por medio de evaluaciones rápidas orientadas fundamentalmente a la identificación de debilidades (de Clase C).
- **I.T. Mark Premium** acredita a una empresa que ha conseguido una Buena Madurez en sus procesos de trabajo técnico, seguridad y del negocio. En este caso los niveles necesarios son considerablemente superiores a los descritos

anteriormente, exigiéndose que todos los procesos evaluados desde los tres puntos de vista están razonablemente desarrollados. Por ejemplo, en el caso de CMMI se utiliza una evaluación bastante detallada de Clase B sobre CMMI Nivel 2, que una vez superada nos facilita una confianza grande de poder superar con un trabajo relativamente escaso la Evaluación Oficial CMMI de Clase A.

- **I.T. Mark Elite** acredita a una empresa que ha conseguido un nivel Superior en la Definición e Institucionalización de sus procesos de trabajo técnico, de seguridad y de negocio, por lo que se confía en que la calidad de sus productos sea buena, debido a la madurez de sus procesos y a la mejora continua.

  En el caso de Elite, los niveles de exigencia son proporcionalmente superiores, y en caso de CMMI se trabaja sobre el Nivel 3.

Nivel	Evaluación de los procesos de negocio	Evaluación de los procesos de seguridad de la información	Evaluación de los proceso de gestión y desarrollo de software	
			Clase de la evaluación	Hallazgos
it mark ELITE	No hay categoría en rojo y > 75%	Nivel 3	Clase B, Nivel de madurez 3	Ningún área de proceso en rojo & Nº áreas de proceso en verde >=11
it mark PREMIUM	No hay categoría en rojo y > 60%	Nivel 2	Clase B, Nivel de madurez 2	Ningún área de proceso en rojo & Nº áreas de proceso en verde >=3
it mark	No más de una categoría en rojo y > 50%	Nivel 1	Clase C, Nivel de madurez 2	Clase C: No más de 2 AP alcanzando menos de 50%. Estas AP no pueden ser ni PP, ni PMC.

Figura 17. Niveles IT Mark

IT Mark está diseñado de forma que define un Camino de Mejora Continua incluso para Micro Empresas, que es totalmente compatible y alineado con el modelo CMMI (como se ve en la figura 18), y que ayuda a las empresas en su permanente búsqueda de competitividad en el mercado.

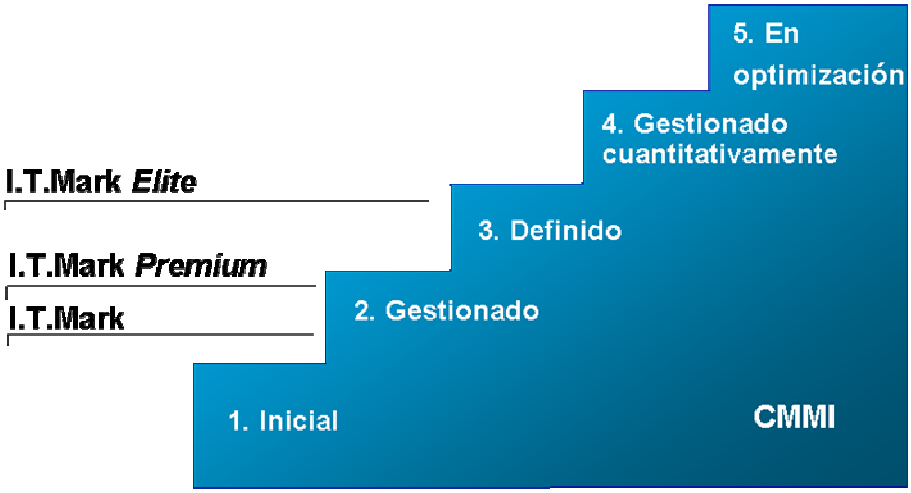

Figura 18. Grados IT Mark y Niveles CMMI

2.7.3 Certificación IT Mark

IT Mark está compuesto por 5 partes:

1. Sesiones de Conocimiento
2. Proceso de Evaluación del Negocio
3. Valoración en Administración de Seguridad de la Información
4. Valoración en procesos de software y sistemas
5. Presentación de resultados y oportunidades de mejora.

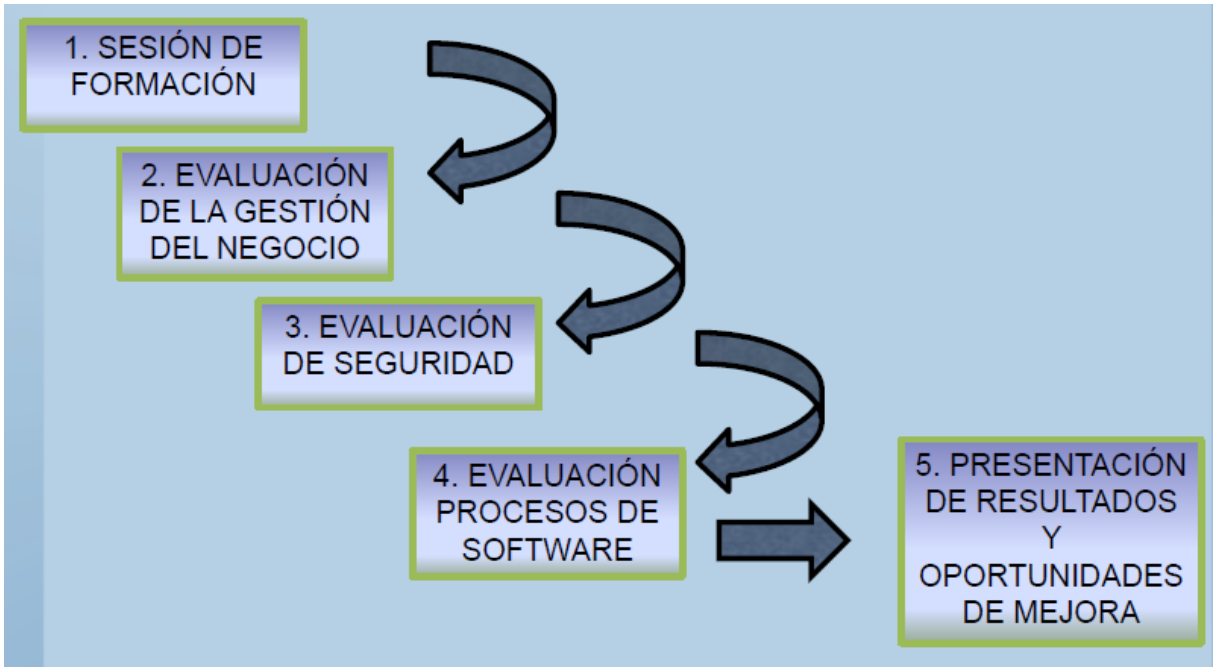

Figura 19. Metodología de Certificación IT Mark

El servicio que presta IT Mark se realiza por medio de evaluadores experimentados y se basa en modelos mundialmente reconocidos y se completa con la explicación de los resultados preliminares a la organización afectada:

- Puntos fuertes y puntos débiles de los Procesos de la organización, de acuerdo con la Buena Práctica y los diferentes modelos descritos.
- Certificado Provisional, en caso en que la evaluación resultara exitosa.

Los puntos débiles constituyen la base para la elaboración del Plan de Mejora, que la empresa deberá desarrollar de acuerdo con sus objetivos estratégicos y su proceso de Mejora Continua.

Una vez implantadas las mejoras recomendadas, se llevará a cabo una evaluación final que resolverá si la PyME cumple los requisitos necesarios que acreditan la calidad de sus procesos. La PyME que alcance el nivel establecido por el modelo IT Mark Standard será acreedora de la certificación IT Mark.

Las empresas que en el momento de la evaluación cumplan con los requerimientos de los niveles de IT Mark obtendrán su certificación. La misma se podrá realizar en una segunda ocasión en las empresas que así lo soliciten al término de la implementación de mejora de procesos.

La certificación propiamente dicha no es el objetivo del Modelo, pero sí constituye un hito alcanzable a corto/medio plazo que proporciona a la empresa un reconocimiento del mercado. Los niveles Premium y Elite son ya más ambiciosos, ya que proporcionan un gran nivel de confianza en alcanzar niveles equivalentes del modelo CMMI por un coste muy inferior.

2.8 COMPETISOFT

El proyecto CYTED COMPETISOFT: “Mejora de procesos para fomentar la competitividad de la pequeña y mediana industria del software de Iberoamérica”, es un proyecto de investigación, financiado por CYTED, programa internacional de cooperación científica y tecnológica multilateral, de ámbito Iberoamericano., en el que participan universidades, empresas, centros públicos y organismos de estandarización de trece países de la región, entre los que se encuentran: Argentina, Brasil, Chile, Colombia, Costa Rica, Cuba, Ecuador, España, México, Perú, Portugal, Uruguay y Venezuela.

El principal objetivo es incrementar el nivel de competitividad de las PyMEs iberoamericanas productoras de software mediante la creación y difusión de un marco metodológico común que, ajustado a sus necesidades específicas, pueda llegar a ser la base

sobre la que establecer un mecanismo de evaluación y certificación de la industria del software reconocido en toda Iberoamérica.

Para lograr ese objetivo general se plantean los siguientes objetivos específicos:

- Desarrollar un Marco Metodológico común ajustado a la realidad socio-económica de las PyMEs iberoamericanas, orientado a la mejora continua de sus procesos. Este Marco Metodológico, que estará compuesto por un Modelo de Procesos, un Modelo de Capacidades y un Método de Evaluación, será validado, en el marco de este proyecto mediante su aplicación controlada, en empresas y organizaciones de diferentes países de la región CYTED.
- Difundir la cultura de la mejora de procesos en el sector informático iberoamericano y más específicamente formar, tanto a investigadores y/o docentes universitarios (formación de formadores) como a profesionales de un buen número de PyMES productoras de software, mediante los cursos que se organizaran en este proyecto CYTED y mediante la difusión -a través de la Web del proyecto- de los materiales de formación que se elaborarán; así como mediante la supervisión y desarrollo de tesis de postgrado para estudiantes y docentes de la región.
- Incidir en los diferentes organismos de normalización y certificación de los países iberoamericanos, para que asuman que los principios metodológicos objeto de este proyecto CYTED pueden ser la base para establecer un mecanismo común y mutuamente reconocido de evaluación y certificación de la industria del software Iberoamericana.

Este proyecto ofrece una visión sobre los componentes de su marco metodológico: el Modelo de Referencia de Procesos, el Modelo de Mejora de Procesos y el Modelo de Evaluación, presenta su aplicación práctica en la industria del software y otros aspectos de interés para la puesta en marcha de la mejora de procesos software en pequeñas y medianas empresas y proyectos software, como son: estrategias para empezar un programa de mejora, gestión de proyectos, verificación y validación, mantenimiento del software, medición de procesos, etc.

2.8.1 Modelo de Procesos

El modelo de procesos de COMPETISOFT está basado en el definido por MoProSoft[6].Tiene tres categorías de procesos: Alta Dirección, Gerencia y Operación que reflejan la estructura de una organización.

En cada proceso están definidos los roles responsables por la ejecución de las prácticas. Los roles se asignan al personal de la organización de acuerdo a sus habilidades y capacitación para desempeñarlos.

En COMPETISOFT se clasifican los roles en Grupo Directivo, Responsable de Proceso y otros roles involucrados. Además se considera al Cliente y al Usuario como roles externos a la organización.

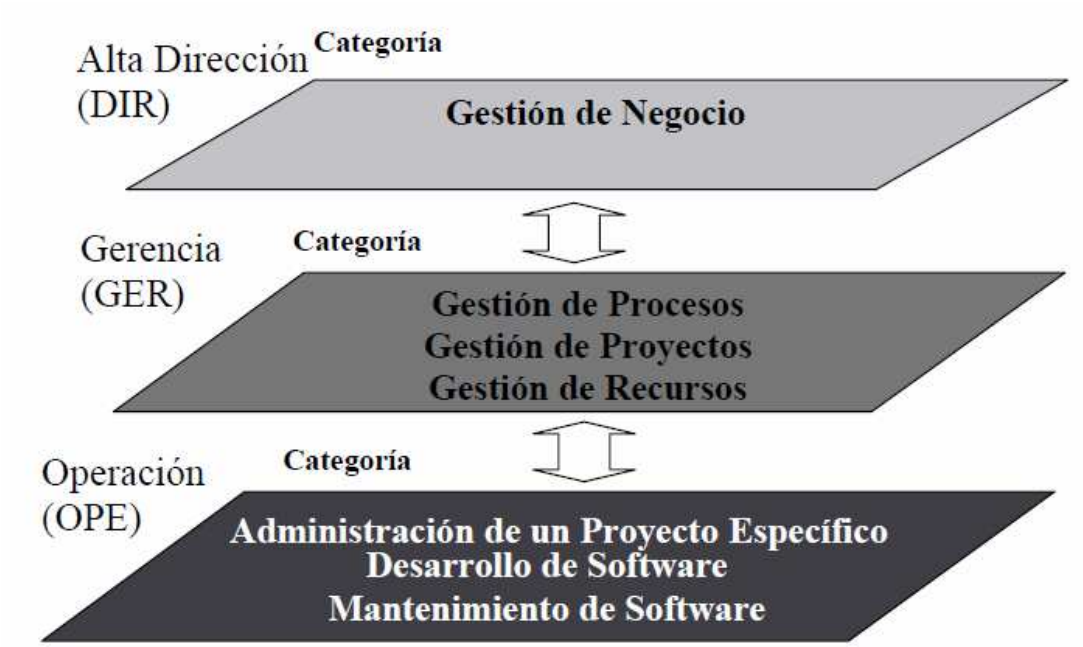

Figura 20. Categorías y Procesos de COMPETISOFT

[6] **Modelo de Procesos para la Industria del Software**. 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. MoProSoft es el nombre del modelo en la comunidad universitaria y profesional, y la norma técnica a la que da contenido es la NMX-059/01-NYCE-2005 que fue declarada Norma Mexicana el 15 de agosto de 2005 con la publicación de su declaratoria en el Diario de la Federación.

2.8.1.1 Categorías de Procesos

Los diez procesos definidos en el modelo COMPETISOFT se incluyen dentro de tres categorías:

1. **Categoría de Alta Dirección (DIR)**

Dentro de la categoría de Alta dirección se incluye únicamente el proceso de Gestión de Negocio que engloba todas las prácticas relacionadas con la gestión de la empresa. Así establece los objetivos y las condiciones necesarias para lograrlos en función de las necesidades de los clientes y de los cambios que, propuestos a raíz de la evaluación de los resultados obtenidos, permitan alcanzar la mejora continua. También debe capacitar a la organización para enfrentarse a un ambiente de cambio, a sus miembros para trabajar en función de los objetivos establecidos, e incluir la gestión de la empresa virtual y la conectividad inter-compañía como una oportunidad de negocio en la industria del software actual. Éstos últimos son puntos a tener en cuenta por compañías que participan en clusters o redes virtuales y son requisitos clave para garantizar la supervivencia de las PyMEs en el mercado actual. Este proceso desarrolla traes actividades fundamentales: Planificación Estratégica, Preparación para la Realización de una Estrategia, Valoración y Mejora Continua de la Organización.

2. **Categoría de Gerencia (GER)**

Esta categoría de procesos Está compuesta por cinco procesos, en los que se incluyen las prácticas necesarias para la gestión de procesos, proyectos y recursos en función de las directrices establecidas desde la Categoría de Alta Dirección. Así mimo, se configura como un punto de unión entre ésta y los procesos de la Categoría de Operación ya que proporciona los elementos para el funcionamiento de los procesos de esta categoría, recibe y evalúa la información generada por éstos y comunica los resultados a la Categoría de Alta Dirección. Dentro de esta categoría se incluyen los procesos: Gestión de Procesos, Gestión de Proyectos, Gestión de Recursos Humanos, Gestión de Bienes, Servicios e Infraestructura y Gestión de Conocimiento.

3. **Categoría de Operación (OPE)**

Categoría de procesos que aborda las prácticas de los proyectos de desarrollo y de mantenimiento de software agrupadas en tres procesos. La Categoría de Operación se sitúa por debajo de la Categoría de Gerencia, por lo que realiza sus actividades de acuerdo a las directrices marcadas por ésta (en consonancia con los objetivos de la organización) y a la que entrega toda la información y productos generados. Dentro de esta categoría se incluyen los

procesos: Administración de un Proyecto Especifico, Desarrollo de Software y Mantenimiento de Software.

2.8.1.2 Procesos

A continuación se listan los procesos definidos por el modelo de procesos:

1. **DIR.1 Gestión de Negocio**

El propósito de Gestión de Negocio es establecer la razón de ser de la organización, sus objetivos y las condiciones para lograrlos, para lo cual es necesario considerar las necesidades de los clientes, así como evaluar los resultados para poder proponer cambios que permitan la mejora continua.

Adicionalmente habilita a la organización para responder a un ambiente de cambio y a sus miembros para trabajar en función de los objetivos establecidos.

2. **GES.1 Gestión de Procesos**

El propósito de Gestión de Procesos es establecer los procesos de la organización, en función de los procesos requeridos identificados en el plan estratégico. Así como definir, planificar, e implantar las actividades de mejora en los mismos.

3. **GES.2 Gestión de Proyectos**

El propósito de la Gestión de Proyectos es asegurar que los proyectos contribuyan al cumplimiento de los objetivos y estrategias de la organización.

4. **GES.3 Gestión de Recursos**

El propósito de Gestión de Recursos es conseguir y dotar a la organización de los recursos humanos, infraestructura, ambiente de trabajo y proveedores, así como crear y mantener la base de conocimiento de la organización. La finalidad es apoyar el cumplimiento de los objetivos del plan estratégico de la organización.

5. **GES.3.1 Gestión de Recursos Humanos**

El propósito de Gestión de Recursos Humanos es proporcionar los recursos humanos adecuados para cumplir las responsabilidades asignadas a los roles dentro de la organización, así como la evaluación del ambiente de trabajo.

6. **GES.3.2 Gestión de Bienes, Servicios e Infraestructura**

El propósito de Gestión de Bienes, Servicios e Infraestructura es proporcionar proveedores de bienes, servicios e infraestructura que satisfagan los requisitos de adquisición de los procesos y proyectos.

7. **GES.3.3 Gestión de Conocimiento**

El propósito de Gestión de Conocimiento es mantener disponible y administrar la base de conocimiento que contiene la información y los productos generados por la organización.

8. **OPE.1 Administración de un Proyecto Específico**

El propósito de la Administración de un Proyecto Específico es establecer y llevar a cabo sistemáticamente las actividades que permitan cumplir con los objetivos de un proyecto en tiempo y costo esperados.

9. **OPE.2 Desarrollo de Software**

El propósito de Desarrollo de Software es la realización sistemática de las actividades de análisis, diseño, construcción, integración y pruebas de productos de software nuevos cumpliendo con los requerimientos especificados.

10. **OPE.3 Mantenimiento de Software**

El propósito de Mantenimiento de Software es la realización sistemática de las actividades necesarias para modificar productos software y adaptarlos a los nuevos requisitos y necesidades del producto.

2.8.2 El Modelo de Evaluación.

El modelo de evaluación se basa en el método de evaluación EvalProSoft[7] y ha sido diseñado en consonancia con la norma internacional ISO/IEC 15504. Define cinco niveles de madurez para la organización: Realizado, Gestionado, Establecido, Predecible y Optimizado.

El modelo de capacidad de COMPETISOFT evalúa la capacidad de los procesos en una escala de 0 a 5, donde 0 especifica que no hay procesos definidos y cinco que la organización ha alcanzado un nivel de mejora continua de sus procesos. Cada uno de estos niveles de madurez se divide en uno o dos atributos de capacidad de procesos y cada atributo mide un aspecto particular del proceso.

Cada uno de los elementos descritos anteriormente tiene una escala específica para su medición. En función de la calificación de los atributos del proceso, se obtiene el nivel de capacidad de un proceso, que están dentro del alcance de la evaluación, y a partir estos, el

[7] **Método de Evaluación de procesos para la industria de Software,** otorga a la organización solicitante un perfil del nivel de capacidad de los procesos implantados y un nivel de madurez de capacidades de la organización. Utiliza el Modelo de Procesos para la Industria de Software, MoProSoft V1.1, como modelo de procesos de referencia, el Modelo de Capacidades de Proceso de la *ISO/IEC 15504-2 Performing an assesment,* cumple los requisitos de la norma *ISO/IEC 15504-2 Performing an assesment* en el Método de Evaluación y usa el estándar *ISO/IEC TR 15504-4 Guidance on performing an assesment* como guía.

perfil del nivel de capacidad de los procesos de la empresa. Cuando el alcance de evaluación incluye a todos los procesos de la organización, se le asigna a ésta un nivel de madurez que corresponde con el máximo nivel de capacidad alcanzado por todos los procesos.

El modelo de evaluación define un proceso de evaluación conforme al patrón de procesos del modelo de referencia de COMPETISOFT. Esto permite que el proceso de evaluación se integre como un proceso más dentro de la organización. El modelo de valuación soporta tres usos:

- Evaluación para la acreditación de capacidades.
- Evaluación de capacidades de un proveedor.
- Autoevaluación de capacidades de un proceso.

En el proyecto COMPETISOFT se ha desarrollado un conjunto de medidas para medir el rendimiento y la capacidad de los procesos software basados en el modelo de evaluación descrito. El primer tipo de medidas está relacionado con la dimensión de la capacidad mientras que el segundo se relaciona con la dimensión del proceso.

Tras la ejecución de la evaluación, se obtienen el Informe de resultados y el Informe Estadístico. El primero contiene el perfil del nivel de capacidad de procesos y el nivel de madurez de la organización, además de un resumen de los hallazgos significativos encontrados. El segundo de los informes contiene datos relativos a la evaluación realizada.

2.8.3 El Modelo de Mejora

El modelo de mejora de COMPETISOFT está basado en Agile SPI[8]. Se caracteriza por ser un modelo ligero con el fin de facilitar su aplicabilidad en las PyMEs desarrolladores de software. El modelo de mejora presenta un proceso, iterativo e incremental, organizado a través de pequeños ciclos de mejora que abarcan diferentes casos de mejora dentro del proyecto global de mejora de la organización. Un caso de mejora es una oportunidad de mejora en las áreas de procesos que la empresa ha seleccionado dependiendo de sus objetivos

[8] **Agile SPI** es un framework de SPI (Software Process Improvement) que se caracteriza por guiar la mejora de los procesos de desarrollo de software, manteniendo el nivel de agilidad que la empresa desee, ser un framework basado en modelos livianos para el soporte de un programa de mejoramiento continuo, a través de un proceso de mejora ágil, estar acorde con una industria dinámica, creativa, innovadora e incierta como lo es la industria del software. Una industria donde el conocimiento y el talento humano son elementos fundamentales para garantizar su éxito.
Básicamente se ha formado su estructura a partir de los componentes primarios de un programa de mejora: una guía de mejora y unos modelos de soporte. Los modelos son: modelo de calidad: Agile SPI – Light Quality Model; el modelo de evaluación: Agile SPI – Light Evaluation Model; y el modelo de métricas: Agile SPI – Light Metrics Model.

de negocio y el Plan Estratégico. El objetivo de esta estructuración es obtener resultados rápidos de mejora, con el fin de mantener la motivación sobre el proyecto de mejora en la organización.

Dentro del modelo de mejora se plantea satisfacer los siguientes propósitos:

- Entrega temprana y continua de mejoras.
- Diagnostico continuo
- Colaboración entre grupos.
- Individuos motivados.

Además, el modelo de mejora de COMPETISOFT para el éxito del proyecto SPI promueve: el desarrollo sostenido del proyecto de mejora, a través del trabajo continuo; una infraestructura técnica y de gestión, adecuada para soportar la mejora de procesos; y el aprendizaje continuo como una disciplina clave para el éxito.

El modelo de mejora de COMPETISOFT describe un proceso que guía la mejora de procesos software en cinco fases:

- INSTALACION: Se crea una propuesta de mejora basada en las necesidades del negocio, que guíe a la organización a través de cada una de las actividades siguientes, esta propuesta debe ser aprobada por la alta dirección para garantizar la asignación de los recursos necesarios involucrados en el proyecto SPI. Se definen los objetivos de mejora, los cuales son establecidos a partir de las necesidades de la empresa. También se debe estructurar una infraestructura de gestión, la cual describe la manera en la cual se organizan las personas comprometidas dentro del proyecto SPI. Esta infraestructura organiza el proyecto SPI teniendo en cuenta un equipo de gestión -EG-, un equipo de tecnología de procesos -ETP- y equipos de mejora -EM-.
- DIAGNÓSTICO: Los grupos EG y ETP realizan actividades de valoración para saber cuál es el estado general de los procesos de la empresa, además de un análisis de los resultados que permitan establecer los casos de mejora y sus prioridades, permitiendo así estructurar el plan general de mejora.
- FORMULACIÓN: El EG toman los casos de mejora más prioritarios (1 ó 2 procesos) a mejorar según los resultados arrojados por la valoración hecha en la fase anterior; con base en los resultados el ETP y el EM realiza una primera iteración de mejora con el fin de realizar una medida del esfuerzo que sirva de base para la estimación del esfuerzo, costo y tiempo que tomarán para llevar a

cabo las demás iteraciones de mejora. Con la información y el aprendizaje ganado se realiza una planificación de la(s) siguiente(s) iteración(es) de mejora.

- MEJORA: se gestiona, todo el esfuerzo de los casos de mejora basándose en la estimación hecha en el plan de ejecución de mejora creado en la fase anterior y se ejecutan las mejoras correspondientes a las diferentes iteraciones de acuerdo a los planes establecidos. Debe existir un documento donde se registre la ejecución de los pilotos de prueba, la evaluación de lo nuevo o la nueva mejora que se ha realizado o creado. Si los planes piloto se han desarrollado satisfactoriamente hay que crear planes de aceptación e institucionalización de los nuevos procesos en la empresa.
- REVISIÓN DEL PROGRAMA: el EG hace una retroalimentación antes de volver a comenzar la fase de inicio. En esta fase todas las lecciones aprendidas y las métricas desarrolladas para medir el cumplimiento de los objetivos sirven como base de conocimiento o fuente de información para las personas involucradas en el siguiente ciclo de mejora. Con toda la información recolectada se debe evaluar el trabajo realizado y se deben corregir o ajustar todos los elementos relacionados con la ejecución de un programa SPI, como la infraestructura establecida, los métodos utilizados, los canales de comunicación y si las soluciones a los problemas identificados fueron las adecuadas.

Figura 21. Modelado bajo SPEM de las fases del Modelo de Mejora de CompetiSoft

El modelo de mejora incluye un conjunto de disciplinas que pueden ser aplicadas en menor o mayor medida dependiendo de la fase en la que se aplique y de las prioridades del proyecto de mejora. Una disciplina de mejora es un cuerpo de conocimiento altamente cohesivo asociado a un objetivo dentro del programa de mejora, tal como evaluar procesos, diseñar procesos, analizar procesos, aprender, entre otros. Las disciplinas definidas para el modelo de mejora son: Gestión del programa, Evaluación, Análisis de resultados, Diseño, Implementación. Gestión de la configuración, Aprendizaje y Entrenamiento.

2.9 ISO/IEC 29110

El objetivo de ISO/IEC 29110 es establecer un marco común para describir perfiles evaluables del ciclo de vida software en VSEs (Very Small Enterprises, organización de menos de 25 empleados) y su publicación está prevista para 2010.

Los informes técnicos ISO/IEC 29110 se denominan "Ingeniería de Software. Perfiles de Ciclo de Vida para Empresas Muy Pequeñas" (Software Engineering. Lifecycle Profiles for Very Small Enterprises, VSE). Un informe técnico es un documento publicado por ISO/IEC que ayuda a la comprensión y al uso de la parte normativa de un estándar. En el ámbito de ISO/IEC 29110, los informes técnicos se utilizan para presentar las guías sobre la implementación de perfiles en VSE. Estas guías proporcionan información práctica para facilitar la implementación de los perfiles definidos. Para ISO/IEC 29110, las guías (directrices) serán promulgadas como informes técnicos.

Un **perfil** es un subconjunto de uno o más estándares necesarios para llevar a cabo una función particular, por ejemplo, identificar un ciclo de vida del software adaptado y adecuado a las necesidades del negocio de una VSE. Un perfil incluye típicamente elementos extraídos de los estándares y diseñados para proporcionar una implementación coherente de funcionalidades específicas. Los perfiles están diseñados para proporcionar:

- Conceptos de diferentes estándares.
- Un sistema de referencia que es significativo para clientes, usuarios y proveedores.
- Una base para el desarrollo de la evaluación de conformidad de forma objetiva, uniforme y reconocida internacionalmente.

Un **Perfil Internacional Estandarizado** (International Standardized Profile, ISP) es un documento acordado internacionalmente que incluye las especificaciones de uno o más perfiles.

La conformidad con los perfiles es la forma mediante la que las VSE muestran y documentan el uso y la comprensión de los estándares internacionales.

Los requisitos clave que el WG 24[9] planea satisfacer con este nuevo estándar son:

- Proporcionar a las VSE un reconocimiento como productoras de sistemas de software de calidad con unos costes muy reducidos en cuanto a la

[9] El **grupo de trabajo SC7-WG24** es un grupo de trabajo constituido por ISO, cuyo objetivo es que sus estándares de procesos de software (o adaptaciones de estos) se puedan aplicar a pequeñas empresas desarrolladoras de software. Este grupo nació en la sesión plenaria del SC7 en Finlandia en el año 2005 y está estableciendo un marco común para describir perfiles evaluables del ciclo de vida de software para su uso en Very Small Enterprises.

implementación y el mantenimiento de toda una colección de estándares de ingeniería de software y sistemas, o de la realización de evaluaciones exhaustivas.

- Elaborar guías que sean fáciles de entender, asequibles y utilizables por las VSE.
- Producir un conjunto de perfiles, los cuales construyan o mejoren procesos existentes de la VSE, u ofrezcan orientación en el establecimiento de esos procesos.
- Atender las necesidades del mercado de las VSE, permitiendo perfiles y niveles para dominios específicos.
- Dar ejemplos con el fin de fomentar que las VSE adopten y sigan los procesos que conducen a un software de calidad, y que consideren las necesidades, los problemas y los riesgos de sus dominios.
- Proporcionar una línea de base de cómo múltiples VSE pueden trabajar de forma conjunta o ser evaluadas como un equipo de proyecto, sobre proyectos que pueden ser más complejos de los que cualquier VSE puede realizar individualmente.
- Elaborar perfiles y guías escalables que están en conformidad con ISO/IEC 12207, 15504 y/o ISO 9001:2000, y permitir que la evaluación sea posible con un mínimo de rediseño de los procesos de la VSE.

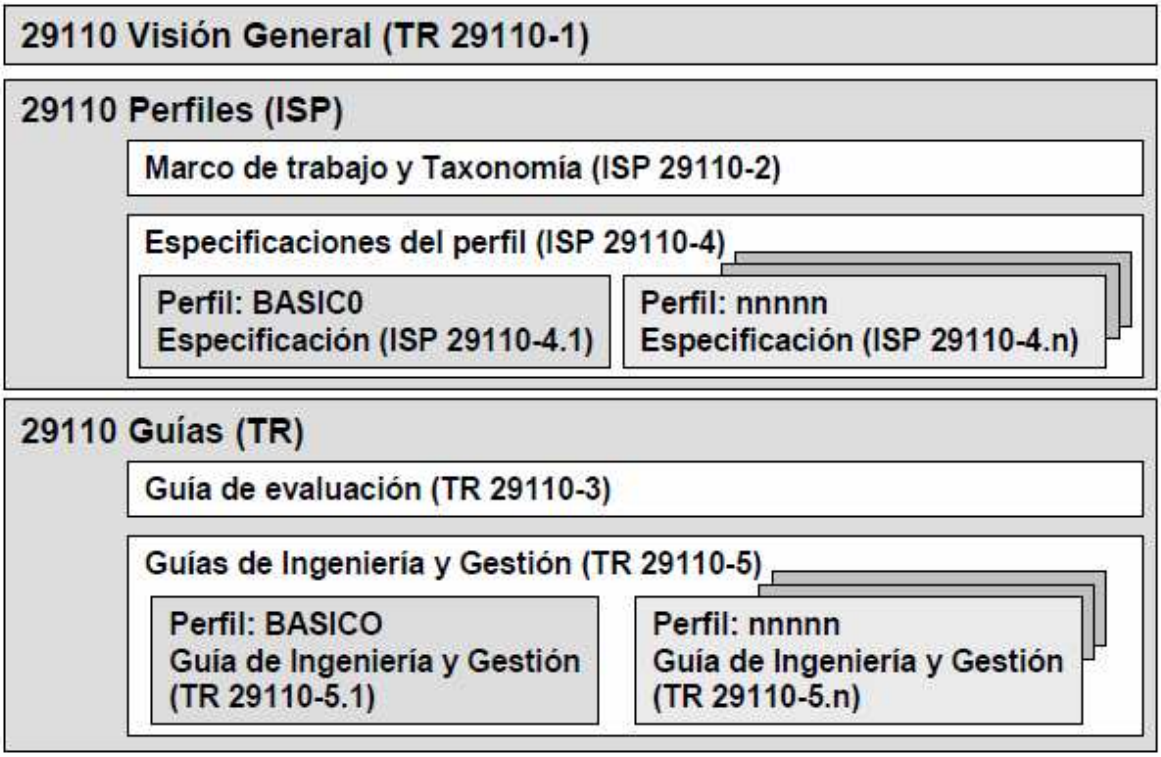

Figura 22. Familia de documentos ISO/IEC 29110

2.9.1 Visión general (TR 29110-1)

La visión general ofrece los conceptos principales necesarios para comprender y utilizar los documentos de ISO/IEC 29110. Introduce los aspectos de negocio, características y requisitos de VSE, y aclara la razón de ser de los perfiles específicos, documentos, estándares y guías de VSE. Asimismo, presenta los conceptos de proceso básico, ciclo de vida y estandarización, y la familia de documentos ISO/IEC 29110.

2.9.2 Perfiles (ISP)

Los perfiles se definen con el objetivo de empaquetar referencias a y/o partes de otros documentos de manera formal para adaptarlos a las necesidades y características de las VSE. Preparar perfiles es un proceso definido de ISO/IEC JTC1, lo cual implica producir dos tipos de documentos:

- Marco de trabajo y taxonomía (TR29110-2). Este documento establece la lógica detrás de la definición y aplicación de los perfiles. Especifica los elementos comunes a todos los perfiles (estructura, conformidad, evaluación) e introduce la taxonomía (catálogo) de los perfiles ISO/IEC 29110. Este documento está dirigido a autores y revisores de ISP (*International Standardized Profiles*), autores de otras partes y de otros perfiles orientados a

VSE. El marco de trabajo y la taxonomía se pueden aplicar a los perfiles identificados según TR 29110-2.

- Especificaciones de perfil (TR29110-4). Por cada perfil hay un documento de especificación de perfil, identificado como 29110-4.X, donde la X es el número asignado al perfil. Su propósito es proporcionar la composición definitiva de un perfil, proporcionar enlaces normativos al subconjunto normativo de estándares (por ejemplo, ISO 12207) usados en el perfil y proporcionar enlaces informativos (referencias) a documentos de "entrada" (por ejemplo, 90003, SWEBOK, de herramientas y otro material de apoyo.

El documento 29110-4.1 "Especificación. Perfil Básico PMI" está dirigido a autores/proveedores de guías. Es un ejemplo de una especificación de perfil. Su propósito es definir una guía de gestión de proyectos y desarrollo de software para un subconjunto de procesos de ISO/IEC 12207, apropiada para las características y necesidades de una VSE. La principal razón para incluir la gestión de proyectos es que el corazón del negocio de la VSE es el desarrollo de software, y su éxito financiero depende de los beneficios del proyecto.

2.9.3 Guías

Las guías contienen directrices de aplicación (de dominio específico) sobre cómo realizar los procesos para alcanzar los niveles de madurez (por ejemplo, actividades recomendadas, medidas, técnicas, plantillas, modelos y métodos, entre otros). Las guías se desarrollan para la implementación del proceso y para la evaluación basada en aspectos de dominio, prácticas y riesgos del negocio. Las guías están dirigidas a VSE y deberían ser accesibles por la VSE en términos de estilo y costo. Hay dos tipos de guías:

- Guía de evaluación (TR29110-3), que describe el proceso que se ha de seguir para realizar una evaluación que determine las capacidades de proceso y la madurez organizativa. Es decir, sirve cuando una organización desea una ejecución de evaluación con el fin de obtener un perfil de capacidad de los procesos implementados y/o un nivel de madurez organizativa. También es aplicable a una situación en la que el cliente solicita a un tercero la ejecución de la evaluación para obtener un perfil del nivel de capacidad del proceso implementado por el proveedor del desarrollo y mantenimiento del software. Igualmente, es adecuada para la autoevaluación y es aplicable a todos los perfiles identificados conforme a TR 29110-3. La metodología de la

evaluación es consistente y acorde con los principios establecidos en ISO/IEC 15504-2.

- Guía de ingeniería y gestión (TR29110-5), que proporciona orientación sobre su implementación y uso o sobre un perfil. Para cada perfil, existe un documento de guía de ingeniería y gestión identificado como 29110-5.x, siendo x el número asignado al perfil (este número coincide con el número que se asigna a la especificación del perfil). Está dirigido a la VSE (personal directivo y técnico), a las organizaciones relacionadas con la VSE (centros de transferencia de tecnología, ministerios de industria del gobierno, estándares nacionales, consorcios y asociaciones, uso académico para la capacitación, autores de productos de software liberados, programas de computación destinados a uso educacional, el comprador y los proveedores, entre otros). Se debe destacar que, por ir dirigido a la VSE, se elabora en un lenguaje más sencillo y de más fácil comprensión que el que se utiliza habitualmente en los estándares.

El documento 29110-5.1 "Guía de ingeniería y gestión. Perfil básico" es un ejemplo de guía de ingeniería y gestión. Este perfil se compone de dos procesos: Gestión de Proyectos e Implementación de Software. El cliente proporciona una Descripción del Producto como una entrada y recibe una Configuración de Software como un resultado de la ejecución del proceso de Implementación de Software, el cual es controlado por el proceso de Gestión de Proyectos.

El proceso de Gestión de Proyectos recibe la Descripción del Producto para elaborar el Plan del Proyecto que sirve para orientar la ejecución del proceso de Implementación del Software. El subproceso de Evaluación y Control de Proyectos compara el Registro del Estado del Progreso del proyecto frente al Plan del Proyecto para eliminar las desviaciones a través de Acciones Correctivas o cambios al Plan del Proyecto. Por último, el subproceso del Cierre del Proyecto entrega la Configuración del Software y obtiene el Documento de Aceptación del cliente para formalizar el cierre del proyecto.

El proceso de Implementación de Software comienza con el subproceso de Iniciación, que recibe el Plan del Proyecto que guía la ejecución de los subprocesos Análisis de Requisitos de Software, Diseño de Arquitectura de Software y Diseño Detallado, Construcción de Software, Integración y Pruebas de Software, y Entrega del Producto. Las tareas de verificación, validación y prueba se incluyen en el trabajo de los subprocesos para

eliminar defectos de los productos. Se establece un Repositorio de Proyecto que protege los productos de trabajo y controla sus versiones y estado.

2.10 ISO/IEC 29119

Son varias las normas que las principales organizaciones de normalización (ISO, IEEE, BSI, etc.) han publicado a lo largo de los últimos años relacionadas directa o indirectamente con la verificación y validación del software. Sin embargo, es en mayo de 2007 cuando ISO creó un grupo de trabajo (WG26[10]) para desarrollar un nuevo "Estándar de Pruebas Software". El objetivo principal de esta norma, actualmente en desarrollo, es que sirva de referente internacional en el ámbito de las pruebas software y que permita tanto eliminar las inconsistencias existentes entre las actuales normas, así como cubrir aquellas áreas del "testing" que simplemente no habían sido tratadas hasta ahora en el resto de normas publicadas. Esta futura norma se conoce como "ISO/IEC 29119 Software Testing", cuya publicación en versión FIS (Final International Standard) está prevista para mediados del año 2012.

La ISO/IEC 29119 cubrirá el ciclo de vida completo, a través del análisis, diseño, implementación y mantenimiento de las pruebas software. Y se basa en las principales normas que en la actualidad son referentes en pruebas del software, modelos de procesos y ciclos de vida, principalmente:

- BSI (British Standards Institution): BS 7925-1, Software Testing: Part 1-Vocabulary y BS 7925-2, Software Testing: Part 2-Software Component Testing.
- IEEE: IEEE Std. 829, Software Test Documentation y IEEE STD 1008, Software Unit Testing, IEEE STD 1012-1998 Software Verification and Validation y IEEE STD 1028-1997 Software Reviews.
- ISO/IEC: ISO/IEC 12207, Software Life Cycle Processes, ISO/IEC 15289, System and Software Life Cycle Process Information Products y ISO/IEC TR 19759, Guide to the Software Engineering Body of Knowledge

[10] **El grupo de trabajo AEN/CTN 71/SC7/GT26** - Pruebas de Software (WG26) fue constituido por AENOR, Organismo de Normalización en España, para participar en la elaboración del futuro estándar ISO sobre pruebas de software, denominado ISO/IEC 29119 Software Testing. La misión principal del GT26 es la colaboración con el grupo de trabajo internacional ISO/IEC JTC1/SC7 Working Group 26 (WG26) en su elaboración, seguimiento de los trabajos y canalización de la postura Española. El grupo de trabajo GT26 está formado por 25 miembros, procedentes de la Industria y de Universidades.

Este estándar, cuya elaboración comenzó en el año 2007 y cuyo primer borrador fue publicado en 2008, incluye los procesos de prueba de software, documentación y técnicas. Este estándar tiene como característica que puede ser aplicado de forma independiente por diferentes organizaciones y/o proyectos. Presenta una estructura de componentes definida, amplia y abarcadora de los elementos requeridos para el desarrollo del entorno de prueba y consta de cuatro partes:

1. Conceptos y Vocabulario
2. Proceso de Pruebas
3. Documentación de Pruebas
4. Técnicas de Prueba

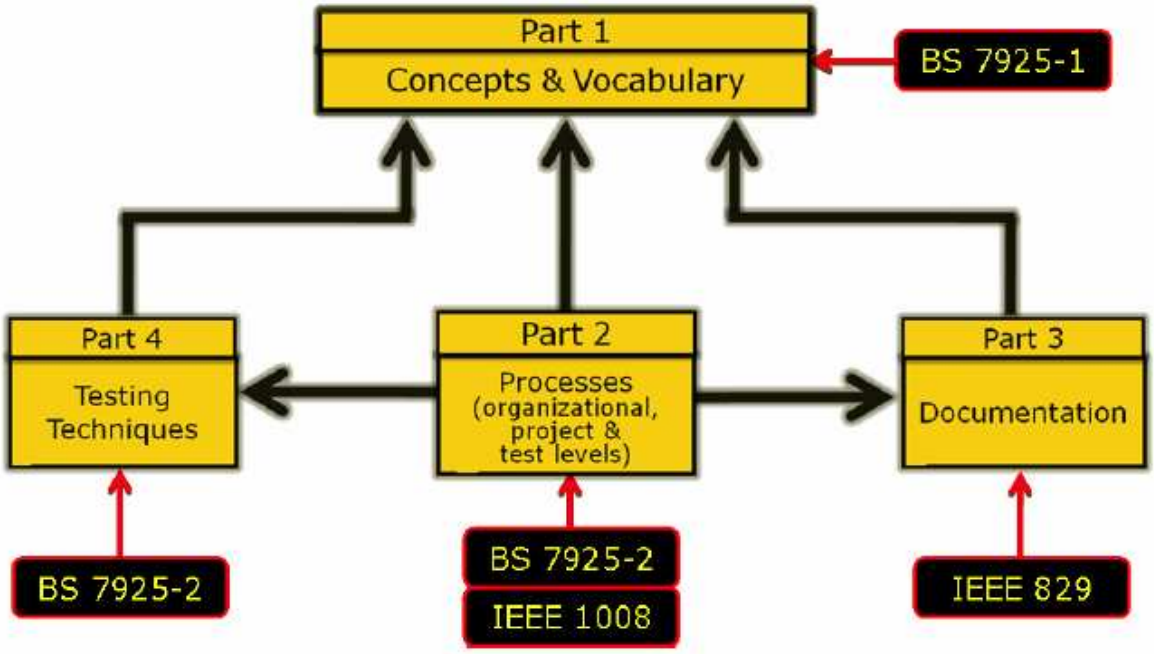

Figura 23. Estructura ISO/IEC 29119

Las partes 2 (procesos) y 3 (documentación) han sido planificadas en una primera secuencia, que tras sucesivas revisiones por expertos externos e internos a los diferentes comités producirán los primeros borradores del comité (*Committee Draft*: CD) para continuar con los procesos de tramitación y revisión habituales en la elaboración de los estándares ISO.

2.10.1 Parte 1 - Conceptos y Vocabulario

El objetivo de la parte 1 es dar una visión general de la norma y de los conceptos generales de pruebas de software y proporcionar un vocabulario de términos de pruebas de software que cubre el ciclo de vida del software el examen. El borrador de la parte 1 se

encuentra actualmente en fase de desarrollo. Se prevé que esta parte se incluyan los siguientes temas:

- Introducción a las pruebas de software
- Pruebas de software en un contexto organizativo
- El proceso de prueba
- La relación entre las pruebas y el desarrollo
- Implicaciones de los modelos de ciclo de vida de desarrollo software (por ejemplo, cascada, espiral, ágil)
- Tipos de pruebas, pruebas técnicas y fases/niveles de prueba
- Pruebas basadas en el Riesgo
- Elementos de Pruebas
- Requisitos de verificación del sistema
- Requisitos de validación del sistema

2.10.2 Parte 2 - Proceso de pruebas

La parte 2 de la norma cubrirá un modelo de proceso de pruebas genérico que pueda ser utilizado en cualquier desarrollo de software y ciclo de vida de pruebas. En particular, el modelo de procesos tal y como figura en el último borrador de trabajo está formado por tres niveles:

- Procesos de la organización
- Procesos de gestión
- Procesos "fundamentales"

En el nivel superior se encuentran los procesos de la organización, que no son específicos de un determinado proyecto de pruebas. Permiten definir las políticas y estrategias aplicables a toda la organización o a una línea de proyectos.

Para un proyecto de pruebas se tienen los niveles de procesos de gestión y procesos "fundamentales". Se define un conjunto de procesos de gestión genéricos para permitir flexibilidad y adaptación a diferentes contextos. Estos procesos son la planificación, monitorización y control, y finalización de las pruebas. Los diferentes procesos de gestión se podrán instanciar en uno o en varios dependiendo de la situación. Por ejemplo, en un proyecto simple puede existir solamente un plan de pruebas global (una sola instancia). En un proyecto más complejo puede existir un plan maestro y otros planes subordinados a éste para cubrir

diferentes niveles de prueba (por ejemplo, pruebas de integración, sistema o aceptación) o diferentes tipos de prueba. Cada uno de ellos sería una instancia del proceso de gestión.

Los procesos fundamentales abarcan los aspectos técnicos de las pruebas: diseño e implementación, puesta a punto del entorno, ejecución de pruebas y la notificación de resultados de las pruebas. También se incluirán variantes de los procesos para contemplar tanto pruebas dinámicas como estáticas.

Los siguientes diagramas ilustran las diferentes capas del proceso que se describen en la Parte 2 (borrador de septiembre 2009).

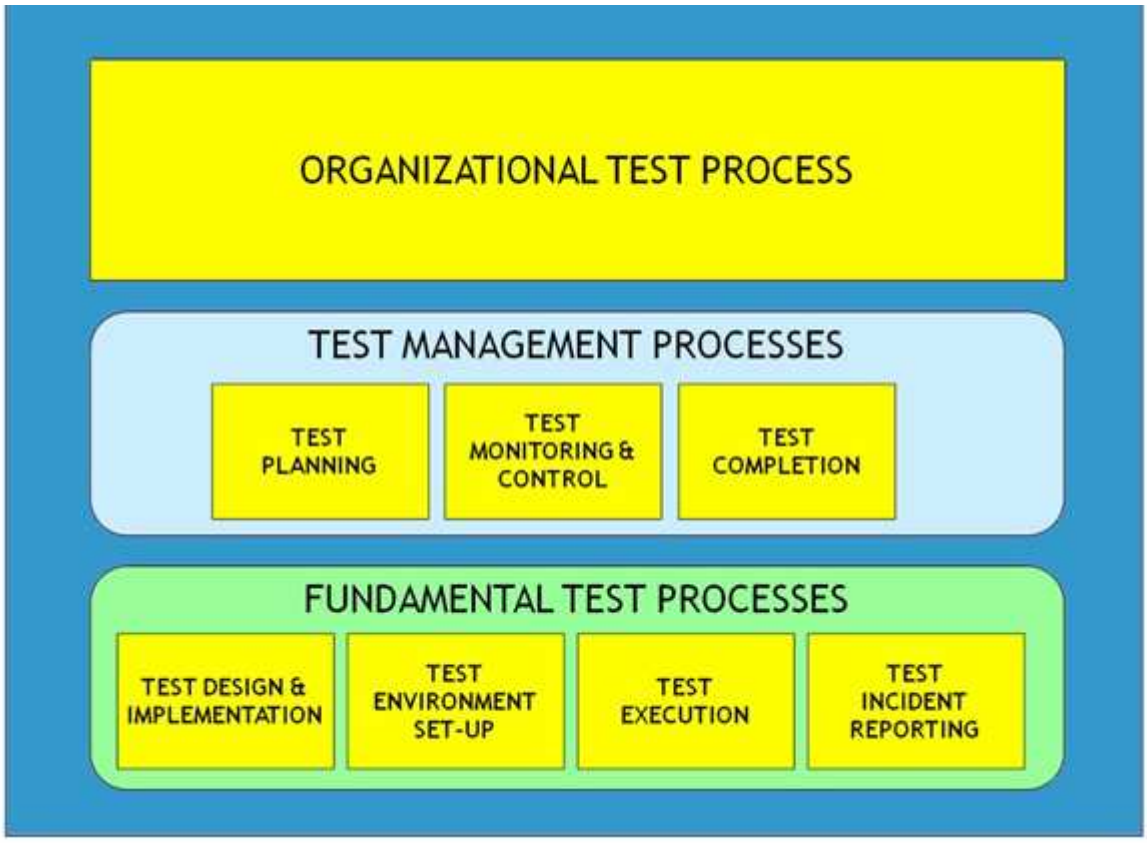

Figura 24. Vista general del Modelo de Proceso de Pruebas (Borrador, Sep. 2009)

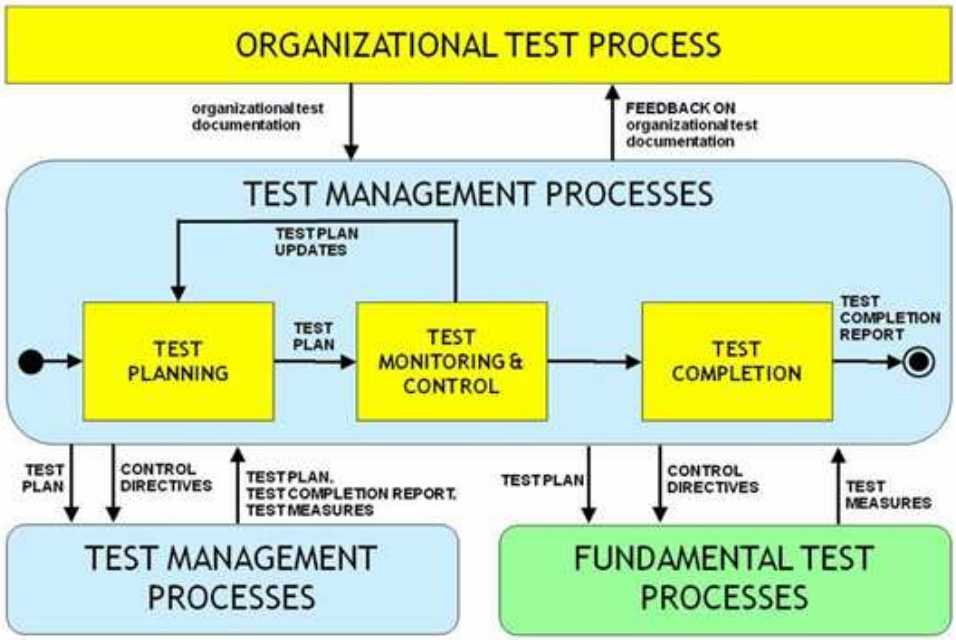

Figura 25. Vista alternativa del Modelo de Proceso de Pruebas

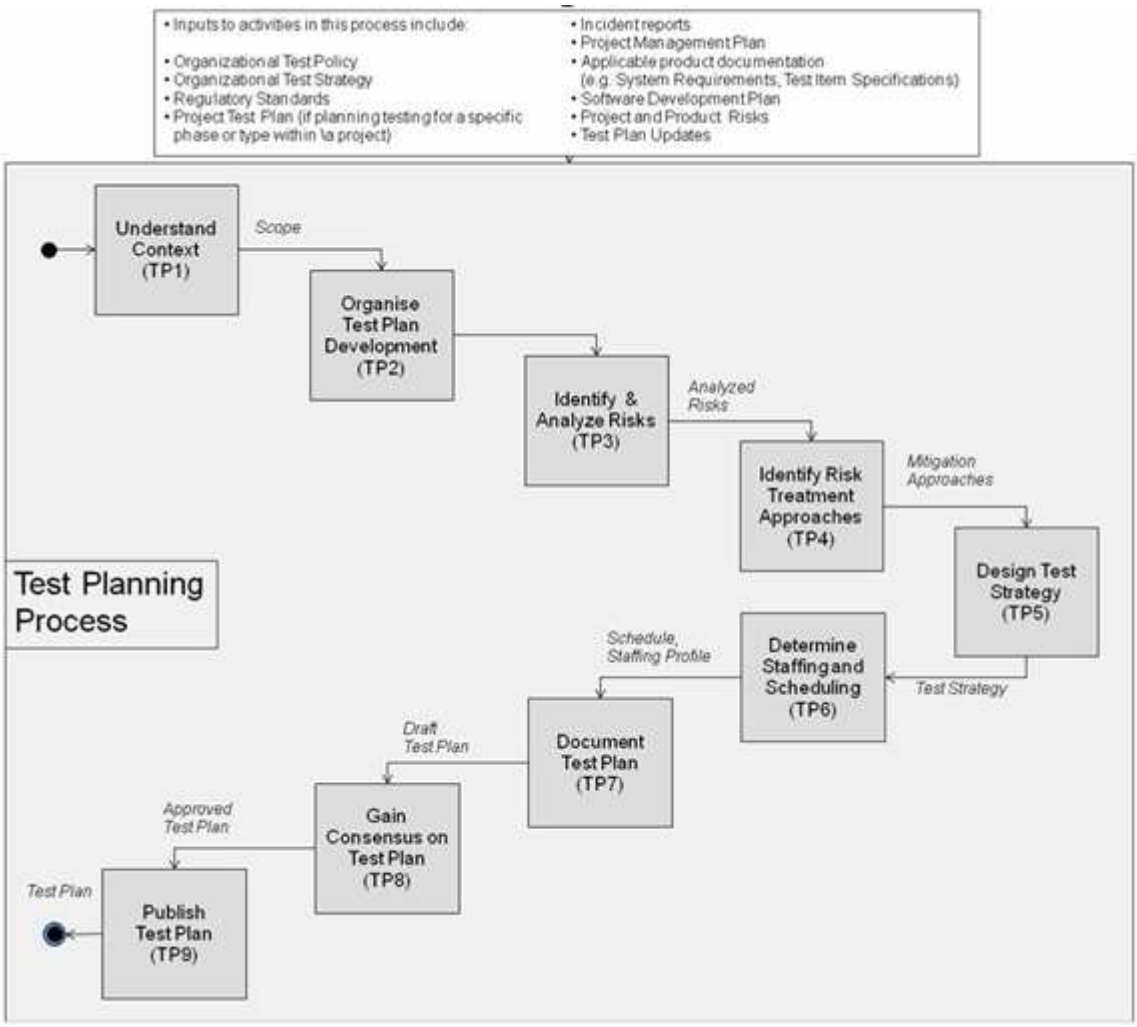

Figura 26. Planificación del Proceso de Pruebas

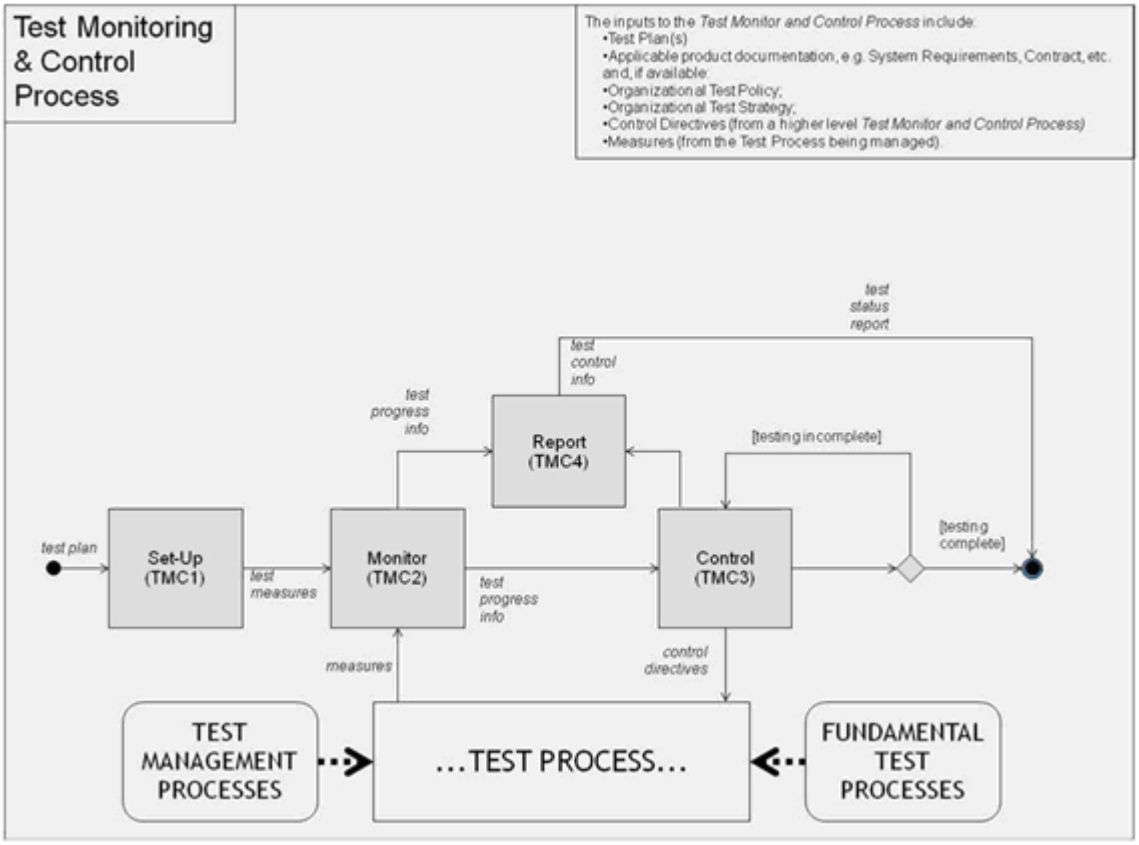

Figura 27. Seguimiento y Control del Proceso de Pruebas

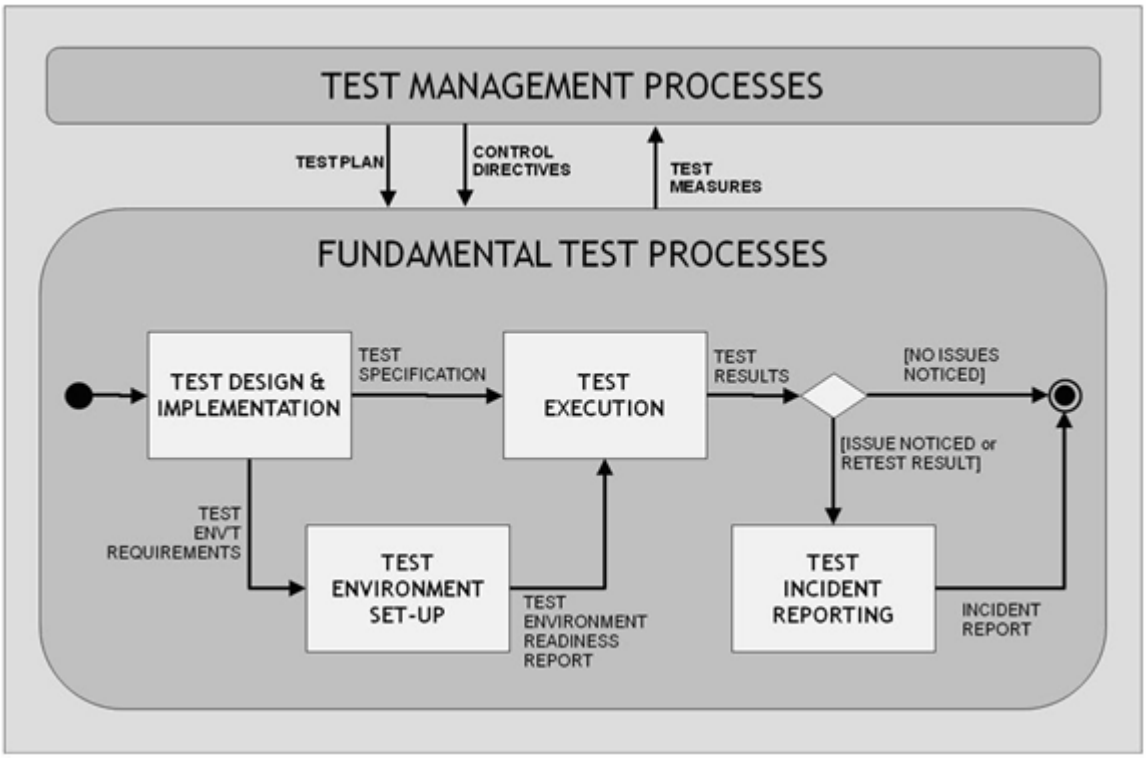

Figura 28. Proceso de pruebas fundamentales

2.10.3 Parte 3 - Documentación de pruebas

Parte 3 de la norma se centra en la documentación de las pruebas a lo largo de todo ciclo de vida del proceso de pruebas del software. Esto contendrá plantillas de todas las capas del modelo de proceso 29119, incluyendo:

- Política organizacional del Proceso de prueba
- Estrategia organizacional del Proceso de prueba
- Proyecto de Gestión del Proceso de prueba
- Proceso de pruebas fundamentales

IEEE ha dado el permiso de ISO para utilizar el conocido Estándar de Documentación de Pruebas IEEE 829 como base para esta parte de la norma. ISO 29119 eventualmente reemplazará IEEE 829.

2.10.4 Parte 4 - Técnicas de pruebas

La parte 4 del estándar cubrirá las Técnicas de pruebas del software a través de todos los tipos de pruebas, incluyendo estáticas (por ejemplo: revisiones, inspecciones, tutoriales), funcionales (por ejemplo: caja negra, caja blanca), no funcionales (por ejemplo: funcionamiento, seguridad, utilidad) y basadas en experiencia (por ejemplo: cálculo de error, experimental).

La Sociedad de Informática británica ha dado el permiso de ISO para utilizar el Estándar de Componentes de Pruebas del Software BS-7925-1/2 como base para esta parte del estándar. La ISO 29119 tarde o temprano reemplazará al BS-7925-1/2.

Capítulo 3 "UNA REVISION Y COMPARATIVA DE MODELOS DE PROCESOS DE PRUEBA"

3.1 COMPARATIVA DE MODELOS

Tras reunir suficiente información (la cual se puede consultar en el apartado de Referencias, al final de este trabajo) para poder estudiar los modelos de procesos de pruebas más destacados del mercado, se ha procedido al análisis de estos datos. Los resultados de tal análisis pueden verse claramente en la tabla comparativa, Tabla 2.

En esta tabla, se ha intentado relacionar los modelos revisados tratando los aspectos más relevantes de cada uno de ellos, de tal manera que sea posible la comparación entre cada uno de ellos de forma clara y sencilla, con el fin de que una organización sea capaz de elegir el modelo de procesos que más se adecue a sus objetivos.

Para realizar el análisis de los modelos citados en el capítulo anterior se han definido seis criterios: Tipo, Cobertura completa, Completamente definido, Representación por etapas, Representación continua, y Tipo de organización. Estos criterios se encuentran definidos en la Tabla 1.

Criterio	**Definición**
Madurez del modelo	Antigüedad del modelo. Año de publicación.
Propietario o Abierto	Un modelo es propietario si pertenece a una organización privada, de lo contrario el modelo es abierto.
Tipo	Tipo de modelo de mejora. Puede ser un modelo de mejora de procesos si cubre los diferentes procesos existentes en la organización, o un modelo de mejora del proceso de pruebas si solamente cubre el proceso de pruebas.
Específico de pruebas	Un modelo de referencia es Específico de pruebas si incluye todas las prácticas relativas al proceso de pruebas.
Completamente Definido	Un modelo de referencia está Completamente Definido si todas las prácticas definidas en el modelo están completamente especificadas.
Representación por etapas	Un modelo de referencia proporciona la Representación por etapas si describe todos los objetivos y prácticas necesarias para establecer el

	nivel de madurez de la organización.
Representación continua	Un modelo de referencia proporciona la Representación continua si describe todos los objetivos y prácticas necesarias para establecer el nivel de capacidad de procesos.
Adecuado para PyMEs	El modelo de referencia es Adecuado para PyMEs si su implantación en este tipo de organización es relativamente sencilla.

Tabla 1. Criterios definidos para realizar el análisis de los modelos de referencia

Criterio/Modelo	**CMM/CMMI**	**ISO 12207**	**TMM**	**TMMI**	**TPI**	**TestPAI**
Madurez del modelo	1991	1995	1996	2008	1998	2008
Propietario o Abierto	Marca registrada de la Universidad Cornegie Mellon. Propietario	Estándar Internacional registrado por ISO/IEC. Abierto	Marca de servicio registrada del Instituto Tecnológico de Illinois. Propietario	Marca registrada de la Fundación TMMI. Propietario	Marca registrada de Sogeti. Propietario	Trabajo soportado por el proyecto *Buen Gobierno de las Tecnologías de la Información* financiado por ATOS-ORIGIN S.A. Propietario
Tipo	Modelo de mejora de procesos	Modelo de mejora de procesos	Modelo de mejora del proceso de pruebas	Modelo de mejora del proceso de pruebas	Modelo de mejora del proceso de pruebas	Modelo de mejora del proceso de pruebas
Específico de Pruebas	No	No	Si	Si	Si	Si
Completamente Definido	Si	Si	Si	Si	Si	Si
Representación por etapas	Si	Si	Si	Si	Si	Si
Representación continua	Si	Si	No	No	No	No
Adecuado para PyMEs	No	No	No	No	No	Si

Criterio/Modelo	**TMap**	**IT MARK**	**COMPETISOFT**	**ISO/IEC 29110**	**ISO/IEC 29119**
Madurez del modelo	1995	Desconocido	2008	En 2008 (Primer borrador)	Su elaboración comenzó en 2007
Propietario o Abierto	Marca registrada de Sogeti. Propietario	Marca registrada por European Software Institute y sus Socios de la Alianza de los ESI Centers. Propietario	Proyecto de investigación financiado por CYTED, Ciencia y Tecnología para el Desarrollo. Propietario	Estándar Internacional registrado por ISO/IEC. Abierto	Estándar Internacional registrado por ISO/IEC. Abierto
Tipo	Modelo de mejora del proceso de pruebas	Servicio internacional de certificación que evalúa los procesos técnicos y de negocio, diseñado específicamente para PyMEs del sector TI	Modelo de mejora de Procesos para Fomentar la Competitividad de la Pequeña y Mediana Industria del Software de Iberoamérica	Modelo de mejora de procesos para PyMEs	Modelo de mejora del proceso de pruebas
Específico de pruebas	Si	No	No	No	Si
Completamente Definido	Si	Si	Si	No	No
Representación por etapas	No	Si	Si	Si	Si
Representación continua	No	Si	Si	Si	Si
Adecuado para PyMEs	No	Si	Si	Si	Si

Tabla 2. Tabla comparativa de modelos de mejora del proceso de pruebas

3.2 Puntos fuertes y Puntos débiles de los modelos estudiados

Tras analizar en detalle los modelos de procesos comentados en apartados anteriores, es ahora el turno de obtener los puntos fuertes y puntos débiles que presenta cada uno de ellos. De esta forma, se facilitará la elección de un modelo u otro dependiendo de las características de nuestra empresa y de los que objetivos que se pretendan alcanzar.

3.2.1 CMMI

No cabe duda de que CMMI, tanto por su difusión y resultados reportados, ha sido un modelo exitoso. Por ello, a priori, dadas las similitudes, debemos esperar que CMMI ofrezca también brillantes resultados en incremento de la productividad, más rápida respuesta al mercado, reducción de defectos, disminución de costes, planificaciones fiables,…

Entre sus fortalezas podríamos destacar:

- Inclusión de las prácticas de institucionalización, que permiten asegurar que los procesos asociados con cada área de proceso serán efectivos, repetibles y duraderos, [33].
- Guía paso a paso para la mejora, a través de niveles de madurez y capacidad (frente a ISO), [33].
- Transición del 'aprendizaje individual' al 'aprendizaje de la organización' por mejora continua, lecciones aprendidas y uso de bibliotecas y bases de datos de proyectos mejorados, [33].
- Más conocido internacionalmente.
- Más implantado en grandes corporaciones.
- Valoración de Niveles externizable, [34].
- Definidos como un conjunto de áreas de procesos clave.
- Posee un modelo de evaluación (SCAMPI).
- Existen organizaciones evaluadas.
- La documentación es gratuita.

Algunas de sus debilidades son:

- El CMMI puede llegar a ser excesivamente detallado para algunas organizaciones, [33].
- Puede ser considerado rígido, [33].
- Requiere mayor inversión para ser completamente implementado.

- Los servicios de capacitación y consultoría son caros.
- Puede ser difícil de entender ya que es un modelo bastante extenso.
- Son necesarios Auditores acreditados por el SEI en CMMI.
- No existe un esquema de certificación internacional.
- Las evaluaciones son difíciles y no tienen un periodo de vigencia.
- Gran complejidad para las organizaciones pequeñas, no está orientado a PyMEs.
- Es un modelo propietario.
- Menos específico en ingeniería del software.
- No se encuentran definidas todas las mejores prácticas relacionadas con las pruebas software ni el conjunto de los productos de trabajo que son necesarios, [27].

3.2.2 ISO/IEC 12207

La norma internacional ISO/IEC 12207:2008 establece un conjunto de buenas prácticas para guiar a las organizaciones en la mejora de sus procesos de desarrollo y mantenimiento software. Entre sus puntos fuertes podríamos destacar:

- Es una Norma ISO, internacional y abierta.
- Evolucionada a partir de CMM y otros modelos.
- Es más específico en ingeniería del software que el modelo CMMI-DEV.
- Tiene un modelo de evaluación avalado por una organización internacional como ISO, ISO/EIC 15504.
- Fácil integración con otras normas ISO del sector TIC.
- Tiene un menor coste de certificación que otros modelos similares.
- Existen certificaciones de prestigio, como AENOR.

De sus debilidades podríamos comentar:

- Es menos difundido que CMMI.
- Existe menos documentación, además esta está en ingles y es de pago.
- Está orientada a grandes organizaciones y no aborda las necesidades de las pequeñas empresas, donde la aplicación de estos resulta costosa en términos económicos y de esfuerzo, ya que requieren una gran inversión en dinero, tiempo y recursos, sus recomendaciones son complejas de aplicar y el retorno de la inversión se produce a muy largo plazo.

3.2.3 TMM/TMMI

TMMI (Test Maturity Model Integration) es un modelo detallado para mejorar el proceso de prueba que cuenta con las siguientes ventajas:

- Ofrece un modelo con un nivel de pruebas más riguroso que el CMMI, pero muy en su linea.
- Está teniendo un crecimiento rápido por Europa, Asia y USA.
- Es el único método de medida del proceso de pruebas independiente.
- Proporciona certificación y procesos de formación/examen, procedimientos y estándares para acreditación formal y pública de Evaluadores y Evaluadores líder.

Además de ventajas, estos modelos también poseen algunos inconvenientes:

- Está diseñado para organizaciones grandes, no es fácilmente adaptable a estructuras mucho más pequeñas, como las PyMEs.
- El coste de un proceso de mejora es desproporcionado respecto a los recursos disponibles de una PYME según [32].
- No se encuentra totalmente definido; únicamente se ha desarrollado para los procesos situados en el nivel 2 de madurez.
- En el caso de que los desarrolladores sean los testers o equipos integrados de pruebas, este modelo de referencia resulta inapropiado, debido a la dimensión reducida de estos equipos la carga de trabajo sería excesiva.
- Aunque el modelo es bastante fácil de entender, su aplicación en una organización no siempre es una tarea simple.

3.2.4 TPI

TPI (Test Process Improvement) es un modelo de referencia dirigido al proceso de pruebas que dispone de las siguientes ventajas:

- Cuenta con más de 1.100 profesionales de Software Control & Testing (SCT) repartidos entre Estados Unidos, Europa y la India.
- Aporta el conocimiento y la experiencia necesarios para la mejora de los procesos de prueba.
- Es específico de pruebas.

Sin embargo, también presenta inconvenientes como:

- Está diseñado para organizaciones grandes, y no es fácilmente adaptable a estructuras más pequeñas, como las PyMEs.
- Al igual que en el caso de TMMI, el coste de un proceso de mejora es desproporcionado respecto a los recursos disponibles de una PyME.
- Es menos difundido que otros modelos.
- No existe mucha documentación.

3.2.5 TESTPAI

TestPAI es un área de proceso de pruebas totalmente integrado con CMMI. Ha sido desarrollado para proporcionar el marco de trabajo necesario para que la mejora del proceso de pruebas se desarrolle de forma paralela a la mejora de otros procesos implementados en la organización.

Como ventajas de este modelo cabe destacar:

- Es específico de pruebas.
- Es fácilmente adaptable a estructuras pequeñas, como las PyMEs.
- Totalmente integrada con CMMI.
- Podría ser utilizado en organizaciones que utilizan una estructura organizativa de pruebas dónde los desarrolladores son los testers, equipos de pruebas integrados o grupos de SQA[11].

Inconvenientes:

- No es muy conocida.
- No es internacional.
- En el caso de equipos de pruebas independientes o factorías de pruebas este modelo resulta demasiado simple y además no contempla procesos de gestión de servicios, [27].

3.2.6 TMAP

TMAP (Test Management Aproach) es una metodología para el desarrollo de pruebas software que posee las siguientes ventajas:

- Es específico de pruebas.

[11] **SQA**, Software Quality Assurance (aseguramiento de la calidad de SW): es el conjunto de actividades planificadas y sistemáticas necesarias para aportar la confianza adecuada en que el producto logrará satisfacer los requisitos dados de calidad. El aseguramiento de la calidad se enfoca en identificar y evaluar los defectos que pueden afectar al SW.

- Se encuentra fundamentado en la Gestión del Proceso de Pruebas orientado hacia el Negocio de la Organización.
- Está basado en el análisis de riesgos, resultados, tiempo y coste para la Organización.
- Describe un proceso de Pruebas como un ciclo de gestión de un proyecto.
- Posee un conjunto de herramientas completo para el proceso de Pruebas: análisis de riesgos, estrategia de pruebas y priorización de éstas con una clara orientación al Negocio, técnicas de especificación de pruebas, etc. que facilitan que las pruebas sean reutilizables en el futuro.
- Posee reconocimiento internacional.

Como inconvenientes cabe destacar:

- Está diseñado para organizaciones grandes, y no es fácilmente adaptable a estructuras más pequeñas, como las PyMEs.
- Al igual que en el caso de TPI, el coste de un proceso de mejora es desproporcionado respecto a los recursos disponibles de una PYME.
- Es menos difundido que otros modelos.
- No existe mucha documentación.
- No es un estándar internacional.

3.2.7 IT MARK

IT Mark es el primer servicio internacional de certificación que evalúa los procesos técnicos y de negocio, diseñado específicamente para PyMEs del sector TI. Cuenta con una serie de beneficios que se enumeran a continuación:

- Está diseñado para Pequeñas y Micro Empresas (menos de 10 empleados), aunque también es aplicable para grandes Organizaciones.
- Mejora el desempeño del negocio (desde el punto de vista técnico y administrativo).
- Está basado en modelos internacionalmente reconocidos como CMMI e ISO-17799:2005.
- Se ha aplicado con éxito en Europa, Asia e Iberoamérica.
- Ayuda a ganar reconocimiento en el mercado en cuanto a capacidades de TI.
- Económico, rápido y rentable.

- El programa incluye valoraciones en Gestión del Negocio y en Seguridad Informática, ignoradas por otros modelos, por encima de los procesos de Software y Sistemas.
- Está basado en modelos reconocidos mundialmente y sólidas metodologías probadas, repetibles, confiables y útiles para compararse competitivamente.
- Un buen mecanismo para avanzar luego hacia una valoración integral de CMMI.
- Incluye un plan de mejora a través de los grados sucesivos de madurez.
- Los niveles Premium y Elite proporcionan un gran nivel de confianza en alcanzar niveles equivalentes del modelo CMMI por un coste muy inferior.
- La comercialización de I.T. Mark se realiza, de momento, en el área de actividad de los ESI Centers[12] hasta ahora constituidos (Brasil, México, Bulgaria, Australia y China), y en los 50 países del mundo en los que tiene cobertura la Red ESI@net.
- Es un modelo escalable.

Como inconvenientes podríamos destacar:

- Es menos difundido que otros modelos.
- No existe mucha documentación.
- No es un estándar internacional.

3.2.8 COMPETISOFT

El proyecto CompetiSoft (Mejora de procesos para fomentar la competitividad de la pequeña y mediana industria del software de Iberoamérica", es un proyecto de investigación, en el que participan universidades, empresas, centros públicos y organismos de estandarización. Entre sus ventajas podemos incluir las siguientes:

- Es aplicable a la pequeña y mediana empresa.
- Sirve como modelo de referencia para la mejora de los procesos y de la calidad de los productos.
- Es fácil de entender y de aplicar.

[12] **ESI** (Instituto Europeo del Software) es una Fundación sin ánimo de lucro que trabaja en la mejora de los procesos de software desde su fundación en 1993 por la Comisión Europea y un grupo de grandes empresas europeas intensivas en software, con el apoyo del Gobierno Vasco. ESI es Partner del SEI para CMM/CMMI desde 1996 y proporciona servicios oficiales CMMI en los cinco continentes.

- Es base o complemento de otros modelos, como por ejemplo CMMI, ISO 12207, ISO 15504.
- No costoso en su adopción.

3.2.9 ISO/IEC 29110

La norma ISO/IEC 29110 Ingeniería de Software – Modelo de ciclo de vida para pequeñas empresas, pretende establecer un marco común para describir perfiles evaluables del ciclo de vida software en VSEs. Los beneficios de esta norma serán:

- Es una Norma ISO, internacional y abierta.
- Es específica para PyMEs.
- Hará las normas actuales de ingeniería del software más accesibles a este tipo de empresas, adaptándolas a su tamaño y necesidades de negocio, lo cual es importante para su supervivencia en los mercados y para lograr competitividad, enfrentándose a problemas como la globalización creciente de los mercados y los cambios tecnológicos que se están produciendo.

Como desventaja principal de esta norma podemos destacar que aún no ha sido publicada, ya que está en fase de desarrollo, y por lo tanto, de momento, no podemos analizar más en detalle este estándar internacional.

3.2.10 ISO/IEC 29119

El Estándar de Pruebas Software tiene como objetivo principal servir de referente internacional en el ámbito de las pruebas software y permitir tanto eliminar las inconsistencias existentes entre las actuales normas, así como cubrir aquellas áreas del "testing" que simplemente no habían sido tratadas hasta ahora en el resto de normas publicadas. De las ventajas que puede aportarnos esta norma debemos poner hincapié en que:

- Es una Norma ISO, internacional y abierta.
- Cubrirá el ciclo de vida completo, a través del análisis, diseño, implementación y mantenimiento de las pruebas software.
- Se basará en las principales normas que en la actualidad son referentes en pruebas del software, modelos de procesos y ciclos de vida.
- Servirá para guiar la tendencia que existe en torno a la mejora de las pruebas del software, permitiendo normalizar el modo en que se planifican, diseñan, ejecutan y mantienen las pruebas, así como la manera en que se gestiona dentro de una organización el conocido, pero normalmente olvidado, proceso de "testing".

La desventaja más significativa de esta norma es la misma que en el caso de la norma descrita anteriormente, es decir, todavía está en fase de desarrollo por lo que de momento debemos esperar para tener mayor información acerca de los beneficios e inconvenientes de estas normas.

3.3 RECOMENDACIÓN FINAL

A la hora de hacer una recomendación sobre qué modelo implantar en una determinada organización hay que tener en cuenta diversos aspectos; tales como Tipo de organización (PyMEs o grandes empresas), Modelo Específico de Pruebas del software, Madurez del Modelo, Coste de implantación, Difusión del Modelo, etc.

Si consideramos en primer lugar el Tipo de organización, los modelos más adecuados para PyMEs son TESTPAI, COMPETISOFT, IT MARK, ISO/IEC 29110, ISO/IEC 29119 ya que los modelos restantes (CMMI, ISO/IEC 12207, TMM, TMMI, TPI y TMAP) son modelos orientados a organizaciones de mayor envergadura.

Si lo que la organización está buscando es mejorar sus procesos de pruebas del software los modelos TMM, TMMI, TPI, TMAP, TESTPAI, IT MARK y la ISO 29119 son Modelos de Mejora de Procesos de Pruebas; mientras que CMMI, ISO 12207, COMPETISOFT y la ISO 29110 abarcan todo el ciclo de vida del Software.

El modelo de mayor antigüedad es CMMI que data de 1991, aunque le siguen de cerca la ISO 12207 y TMAP de 1995, TMM de 1996, TPI de 1998 siendo los menos maduros COMPETISOFT y TESTPAI de 2008 y las todavía en desarrollo ISO 29110 e ISO 29119 cuyos primeros borradores se publicaron en 2007 y 2008 respectivamente.

CMMI es el modelo que mayor coste de implantación requiere. En este sentido, la clasificación de modelos sería la misma que con el primero de los criterios tratados, ya que los modelos orientados a PyMEs poseen un menor coste de implantación.

En cuanto a la difusión del Modelo cabe destacar que CMMI es un modelo con un amplio reconocimiento internacional, en cambio, en EEUU la ISO 12207 es prácticamente una norma desconocida siendo más conocida en la Comunidad Europea, al igual que las normas ISO 29110 e ISO 29119 (todavía en desarrollo). No hay que descuidar que tanto IT MARK como TMAP poseen también un reconocimiento internacional aunque en menor medida que CMMI. IT MARK se comercializa en el área de actividad de los ESI Centers hasta ahora constituidos (Brasil, México, Bulgaria, Australia y China), y en los 50 países del mundo en los que tiene cobertura la Red ESI@net. Por otra parte, TMAP, se encuentra difundido por toda la comunidad europea en países como Alemania, Francia, Suecia, Suiza, España, Bélgica, Reino

Unido e inclusive Estados Unidos. Además, tanto TMM/TMMI como TPI han tenido un rápido crecimiento por Europa, Asia y EEUU. Por otro lado, COMPETISOFT posee una gran difusión por toda Iberoamérica, por países como Argentina, Brasil, Chile, Colombia, Costa Rica, Cuba, Ecuador, España, México, Perú, Portugal, Uruguay y Venezuela. En cambio, teniendo en cuenta este aspecto, quizá el modelo que menor difusión tiene es TESTPAI.

Cabe destacar que se podrían considerar muchos otros aspectos a la hora de elegir qué modelo para implantar en nuestra organización.

Capítulo 4 CONCLUSIONES

4.1 ANÁLISIS DE LA CONSECUCIÓN DE OBJETIVOS

Uno de los objetivos de este proyecto era el estudio de las mejores prácticas, modelos y normas de pruebas software que actualmente existen en el mercado. Para ello se ha procedido a la búsqueda de información acerca de los modelos más utilizados en el área de Tecnologías de la Información (TI). Tras observar los modelos disponibles, se procedió a la elección de los más destacados: CMM/CMMI, ISO/IEC 12207, TMM/TMMI, TPI, TMAP, TESTPAI, IT MARK, COMPETISOFT, ISO/IEC 29110, ISO/IEC 29119.

Una dificultad encontrada para la consecución de este objetivo fue el no encontrar la misma cantidad de información de todos los modelos. Algunos de ellos, como es el caso del muy difundido modelo de mejora de procesos CMMI dispone de una gran cantidad de información desde publicaciones, manuales, casos de estudio de aplicación en empresas, encuestas, etc. En cambio, modelos como TESTPAI presentan mayor dificultad para encontrar documentación acerca de este modelo. Otros modelos de los que no se dispone prácticamente información son los estándares internacionales ISO/IEC 29110 e ISO/IEC 29119. Cabe destacar que estos modelos se encuentran todavía en fase de desarrollo, por lo que resulta comprensible la falta de documentación sobre estas normas.

El segundo objetivo que se planteó en este proyecto era la realización de un estudio comparativo de todos los modelos estudiados, relacionándolos en una tabla, con el objetivo de identificar similitudes y diferencias entre dichos modelos. Para llevar a cabo esta tarea, primero se estudió qué criterios se debía seguir para comparar los modelos. Una vez seleccionados dichos criterios se realizó un estudio más a fondo de los modelos con el fin de obtener características que los relacionaran. Los resultados del estudio comparativo se plasmaron en la tabla 2 del Capítulo 3 facilitando así su comparación.

Además de la tabla comparativa, se identificaron las fortalezas y debilidades más destacadas de cada propuesta para facilitar así la selección del modelo que más se adecue a cada organización, cumpliendo así el tercero de los objetivos planteados para el proyecto.

Por último, para fortalecer este objetivo, se procedió a la elaboración de una recomendación final a la hora de seleccionar el modelo que mejor se adapte a la organización que desea implantarlo. Esta recomendación tiene en cuenta diversos aspectos tales como tipo de organización, madurez del modelo, difusión del modelo, coste de implantación, si el modelo buscado es específico de pruebas, etc.

Nuevamente, una de las dificultades encontradas para conseguir este objetivo ha sido la falta de información de algunos modelos.

4.2 LOGROS Y APORTACIONES

Tras la consecución de los objetivos planteados en este proyecto, se ha obtenido gran cantidad de información acerca del estado de la calidad del software en el mercado de las Tecnologías de la información y de los modelos de mejora de procesos.

Gracias a la información disponible en este proyecto, cualquier organización puede seleccionar el modelo más que se adecue a las características de esta. Para ello, dispone tanto de resúmenes de los modelos más destacados del mercado TI como de una tabla comparativa y de un resumen de las fortalezas y debilidades que presentan estos modelos. Además, se incluye una recomendación final a la hora de seleccionar el modelo que más se adapte a la organización en función de diversos parámetros.

Si el objetivo del lector no es a nivel profesional, es decir, no está buscando un modelo que se adapte a las necesidades de su organización, si no que su interés es meramente informativo, este proyecto muestra claramente el estado de la industria del software.

4.3 LÍNEAS FUTURAS

En el tema de calidad a nivel de procesos se han desarrollado diversas propuestas para la industria en general.

Además de los modelos elegidos en este proyecto, existen muchos otros que cubren diversos aspectos y han sido desarrollados con distintos propósitos. Entre los modelos relacionados de manera directa o indirecta con los procesos de software se pueden mencionar que en este proyecto no se han tratado de pueden mencionar:

- RUP (Rational Unified Processes),
- ISO/IEC 20000 (gestión de servicios de TI),
- ITIL (biblioteca de infraestructura de tecnologías de información),
- IDEAL (mejora de procesos recomendado para CMMI),
- PSP (proceso de software para persona),
- TSP (proceso de software para equipos de trabajo),
- SCAMPI (método de evaluación de procesos usado para CMMI),
- Quick Locus (método ligero brasileño de evaluación de CMMI),
- PMBOK (Cuerpo de conocimiento de gestión de proyectos de PMI),

- ISO 10006 (directrices para la calidad en la gestión de proyectos),
- MoProSoft (modelo de procesos de referencia mexicano),
- EvalProSoft (método de evaluación basado en 15504 para MoProSoft),
- SIMEP-Sw (conjunto de modelos ligeros para mejor de procesos, colombiano),
- MPS.BR (modelos de mejor y evaluación de procesos brasileños),
- SPIRE (modelo de mejora de la región Europea),
- TOPS (mejora de procesos para PyMEs),
- PROCESSUS, IMPACT y RAPID entre otros.

Sin duda una posible continuación de este proyecto podría ser continuar con la investigación de modelos relacionados con procesos de software, como los mencionados anteriormente y de muchos otras ya que en el mundo de la industria del software existes muchos modelos que no han sido tratados en este proyecto.

Otra posible línea futura del proyecto podría ser la selección de uno de los modelos tratados, su aplicación en una organización de desarrollo software y el posterior estudio de las dificultades y beneficios encontrados a la hora de llevar a cabo esta tarea.

ANEXO I. Índice de figuras y tablas

Figura 1. Beneficios en la mejora de procesos de pruebas del Software 7
Figura 2. Características de calidad (ISO 9126) 8
Figura 3. Modelo de procesos ISO/IEC 12207:2008 23
Figura 4. Estructura de TMM 29
Figura 5. Los 5 niveles de TMMI 30
Figura 6. Estructura de TMMI 31
Figura 7. Esquema del modelo TPI 35
Figura 8. Matriz de Madurez 36
Figura 9. Objetivos y Prácticas Específicas de TestPAI 37
Figura 10. Prácticas Específicas del SG1 38
Figura 11. Prácticas Específicas del SG2 38
Figura 12. Prácticas Específicas del SG3 38
Figura 13. Prácticas Específicas del SG4 39
Figura 14. Prácticas Específicas del SG5 39
Figura 15. Elementos esenciales de TMap 40
Figura 16. Ciclo de vida de TMap 42
Figura 17. Niveles IT Mark 45
Figura 18. Grados IT Mark y Niveles CMMI 46
Figura 19. Metodología de Certificación IT Mark 46
Figura 20. Categorías y Procesos de COMPETISOFT 49
Figura 21. Modelado bajo SPEM de las fases del Modelo de Mejora de CompetiSoft 55
Figura 22. Familia de documentos ISO/IEC 29110 58
Figura 23. Estructura ISO/IEC 29119 62
Figura 24. Vista general del Modelo de Proceso de Pruebas (Borrador, Sep. 2009) 64
Figura 25. Vista alternativa del Modelo de Proceso de Pruebas 65
Figura 26. Planificación del Proceso de Pruebas 65
Figura 27. Seguimiento y Control del Proceso de Pruebas 66
Figura 28. Proceso de pruebas fundamentales 66

Tabla 1. Criterios definidos para realizar el análisis de los modelos de referencia 69
Tabla 2. Tabla comparativa de modelos de mejora del proceso de pruebas 71

REFERENCIAS

[1]. Systems and software engineering — Software life cycle processes. International Standard ISO/IEC FDIS 12207 IEEE Std 12207-2007

[2]. CMMI. Guía para la integración de procesos. Disponible en: http://www.sei.cmu.edu/library/assets/cmmi-dev-v12-spanish.pdf

[3]. Sinopsis de los modelos SW-CMM y CMMI. Juan Palacio, Abril 2006. Disponible en: http://www.navegapolis.net/files/articulos/sinopsis_cmm.pdf

[4]. CMMI. ESI European Software Institute Tecnalia. Disponible en: http://www.esi.es/index.php?op=15.1.1

[5]. Otros modelos de calidad de software. Disponible en: http://www.javiergarzas.com/2009/01/otros-modelos-de-calidad-software.html

[6]. TMMI (van Veenendaal, E., Test Maturity Model Integration (TMMi) Versión 1.0. TMMI Foundation (www.tmmifoundation.org), 2008)

[7]. TPI Test Process Improvement. Mejora del Proceso de Testing. Disponible en: http://www.es.sogeti.com/Global/Fichas%20comerciales/Ficha_TPI.pdf

[8]. TPI – a model for Test Process Improvement. Jari Andersin. Helsinki, 2004. Seminar on Quality Models for Software Engineering Department of Computer Science UNIVERSITY OF HELSINKI. Disponible en: http://www.cs.helsinki.fi/u/paakki/Andersin.pdf

[9]. TestPAI. Un Área de Proceso de Ingeniería de Nivel de Madurez 3. Disponible en: http://sel.inf.uc3m.es/asanz/testpai/testpai_pa_fullversion_esp.pdf

[10]. Versión completa de TestPAI. Disponible en: http://sel.inf.uc3m.es/asanz/testpai/testpai_pa_fullversion_esp.pdf

[11]. TMap Test Management Approach. Disponible en: http://eng.tmap.net/Home/TMap

[12]. TMap/TPI Newsletter. Disponible en: http://www.sogeti.com/upload/Looking%20for%20Solutions/Documents/TMap-TPI%20Newsletter%20Edition%203%202009.pdf

[13]. Los costos de subestimar los defectos de Software. Disponible en: http://www.es.sogeti.com/Global/ART%C3%8DCULOS/ART_MARTIN_CASPER.pdf

[14]. Un sondeo sobre la práctica actual de pruebas de SW en España. Disponible en: Revista Española de Innovación, Calidad e Ingeniería del Software Volumen 1, No. 2, diciembre, 2005.

[15]. Factores que afectan negativamente a la aplicación práctica de las pruebas de software. Actas de los Talleres de las Jornadas de Ingeniería del Software y Bases de Datos, Vol. 2, No. 4, 2008.

[16]. IT Mark. ESI European Software Institute Tecnalia. Disponible en: http://www.esi.es/index.php?op=15.1.2

[17]. El modelo IT Mark. Disponible en: http://www.fedesoft.org/downloads/Sinertic/RESUMEN_MODELO_IT_MARK.pdf

[18]. IT Mark. Paula Andrea Villa Sánchez. Laboratorio de Calidad del Software, Universidad Tecnológica de Perreira. Disponible en: http://isc.utp.edu.co/archivos_seminarios/IT-MARK.pdf

[19]. La Mejora del Proceso Software. Javier de Ramón Ferreiro, ESI Marzo de 2010. Disponible en: http://www.aetic.es/CLI_AETIC/ftpportalweb/documentos/02Mejora%20de%20los%20Procesos%20del%20SW%20%5BModo%20de%20compatibilidad%5D.pdf

[20]. ISO/IEC 29119, hacia una nueva norma para pruebas software. Disponible en: http://kybeleconsulting.blogspot.com/2009/03/isoiec-29119-hacia-un-nueva-norma-para.html

[21]. ISO/IEC 29119 Software Testing. Disponible en: http://softwaretestingstandard.org/

[22]. Grupo de Trabajo AEN/CTN71/SC7/GT26 Pruebas de Software ISO/IEC 29119 Software Testing Standard. Disponible en: http://in2test.lsi.uniovi.es/gt26/

[23]. Perfiles del ciclo de vida del software para pequeñas empresas: los informes técnicos ISO/IEC 29110, José A. Calvo-Manzano, Javier Garzás, Mario Piattini, Francisco J. Pino, Jesús Salillas, José Luís Sánchez. Disponible en: Revista Española de Innovación, Calidad e Ingeniería del Software, Vol.4, No. 2, 2008

[24]. La mejora de procesos en pequeñas empresas y la ISO/IEC 29110, Mª Carmen García, Javier Garzás, Mario Piattini. Disponible en: www.kybeleconsulting.com

[25]. Conferencia sobre CMMI, ISO 12207/ISO 15504 y COMPETISOFT /ISO 29110, Cornelio Hopmann. Disponible en: http://www.scribd.com/doc/19223012/Conferencia-sobre-CMMI-ISO-12207ISO-15504-y-COMPETISOFT-ISO-29110

[26]. La mejora del proceso Software. Javier de Ramón Ferreiro, Marzo 2010. Disponible en: http://www.aetic.es/CLI_AETIC/ftpportalweb/documentos/02Mejora%20de%20los%20Procesos%20del%20SW%20%5BModo%20de%20compatibilidad%5D.pdf

[27]. Definición de un Modelo Organizativo para una Factoría de Pruebas de Software. Ana Sanz, Javier García, Javier Saldaña, Antonio Amescua1, Manuel Velasco. Disponible en: Actas de los Talleres de las Jornadas de Ingeniería del Software y Bases de Datos, Vol. 3, No. 4, 2009.

[28]. COMPETISOFT, Mejora de Procesos para Fomentar la Competitividad de la Pequeña y Mediana Industria del Software de Iberoamérica Versión 0.2, Diciembre 2006. Disponible en: http://afrodita.unicauca.edu.co/~ecaldon/docs/spi/COMPETISOFT_v02_27-11_2315.pdf

[29]. Evaluación del nivel de calidad de una empresa basada en COMPETISOFT; Laura Alonso, Marina Iglesias, Patricia Pesado, Silvia Esponda. Disponible en: http://revista.info.unlp.edu.ar/tesinas/tesis12.pdf

[30]. Fabricas del software: experiencias, tecnologías y organización; Garzás Parra, Javier; Piattini Velthuis, Mario G. Editorial Ra-Ma, 2007.

[31]. Adaptación de las Normas ISO/IEC 12207:2002 e ISO/IEC 15504:2003 para la Evaluación de la Madurez de Procesos Software en Países en Desarrollo; F. J. Pino, F. García, F. Ruiz, M. Piattini; IEEE Latin America Transactions, Vol. 4, No. 2, Abril 2006

[32]. Experiencias de una PYME en la mejora de procesos de pruebas; Antonio de Rojas, Tanja E. J. Vos y Beatriz Marín. Disponible en: Revista Española de Innovación, Calidad e Ingeniería del Software, Vol.5, No. 2, 2009

[33]. Modelos de Evaluación y Mejora de Procesos: Análisis Comparativo; Manuel de la Villa, Mercedes Ruiz e Isabel Ramos. Disponible en: http://ftp.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-120/paper4.pdf

[34]. Sistemas de Calidad para Tecnologías de la Información; Carlos Manuel Fdez. CISA, CISM, Product Manager Certificación TICs, Septiembre-07. Disponible en: www.compite.org.mx/.../CY07-AENOR-Compite-%20TICs-VF-Sep-%202007.ppt