



CENTRO DE ESTUDIOS FINANCIEROS

VIRIATO, 52	28010 MADRID	914 44 49 20
PONZANO, 15	28010 MADRID	914 44 49 20
G. DE GRÀCIA, 171	08012 BARCELONA	934 15 09 88
ALBORAYA, 23	46010 VALENCIA	963 61 41 99

www.cef.es

info@cef.es

Índice Tema 16

Introducción.

1. Concepto de calidad. La calidad del software.
 - 1.1. La calidad aplicada al software.
 - 1.2. Definición de calidad del software.
2. Modelos de calidad del software.
 - 2.1. El modelo de calidad de McCall.
 - 2.1.1. Factores de Calidad del Software.
 - 2.1.2. Criterios de Calidad del Software.
 - 2.1.3. Métricas de Calidad del Software.
 - 2.2. Estrategias de uso de un modelo de calidad.
3. La garantía de calidad del software.
 - 3.1. Implementación de la garantía de calidad.
4. El control de la calidad del software.
 - 4.1. Análisis estáticos.
 - 4.1.1. Auditorías.
 - 4.1.2. Revisiones.
 - 4.1.3. Verificación formal.
 - 4.2. Pruebas dinámicas.
5. El Plan General de Garantía de Calidad.
 - 5.1. Guía metodológica para la elaboración de planes de garantía de calidad.
 - 5.1.1. Agentes que intervienen en un proyecto.
 - 5.1.2. Metodología para la elaboración de Planes específicos de Garantía de Calidad, aplicables a proyectos de desarrollo de aplicaciones informáticas.

- 5.2. Esquema formal para la clasificación de los proyectos informáticos.
 - 5.2.1. Diagrama característico del proyecto.
 - 5.2.2. Modelo de referencia del proceso de desarrollo.
 - 5.2.3. Perfil de Riesgos del proyecto.
 - 5.2.4. Foco de interés en garantía de calidad.
- 5.3. Procedimientos de control de calidad.
 - 5.3.1. Revisiones.
 - 5.3.2. Pruebas.
 - 5.3.3. Procedimientos extraordinarios.
 - 5.3.4. Procedimientos particulares.
- 5.4. Instrumentos de control y elementos auxiliares de control de calidad.
- 6. El tratamiento de la calidad del software en la metodología métrica v.3.
 - 6.1. La interfaz de aseguramiento de la calidad.



CENTRO DE ESTUDIOS FINANCIEROS

VIRIATO, 52	28010 MADRID	914 44 49 20
PONZANO, 15	28010 MADRID	914 44 49 20
G. DE GRÀCIA, 171	08012 BARCELONA	934 15 09 88
ALBORAYA, 23	46010 VALENCIA	963 61 41 99

www.cef.es

info@cef.es

TEMA 16

Control de calidad en el desarrollo del software.

INTRODUCCIÓN.

Dentro del proceso de desarrollo de un Sistema de Información, además de las consabidas actividades de desarrollo y de gestión del proyecto, se pueden distinguir unas actividades de operación, que están relacionadas con el uso que hace una organización del sistema software, y unas actividades de control, que son aquellas que restringen, limitan y dirigen el desarrollo del software.

Dentro de este último grupo de actividades cabe, a su vez, distinguir dos clases: una relacionada con la evolución del software, que incluye actividades de control de cambios y de gestión de la configuración; y otra relacionada con la calidad del software, que incluye la garantía de calidad y la evaluación, validación y verificación del software.

El estudio de estas últimas actividades, orientadas a garantizar la calidad del producto que se entrega, es decir, el software, será el objeto del presente tema.

Así pues, se abordarán los aspectos relacionados con la Calidad del Software buscando un triple objetivo:

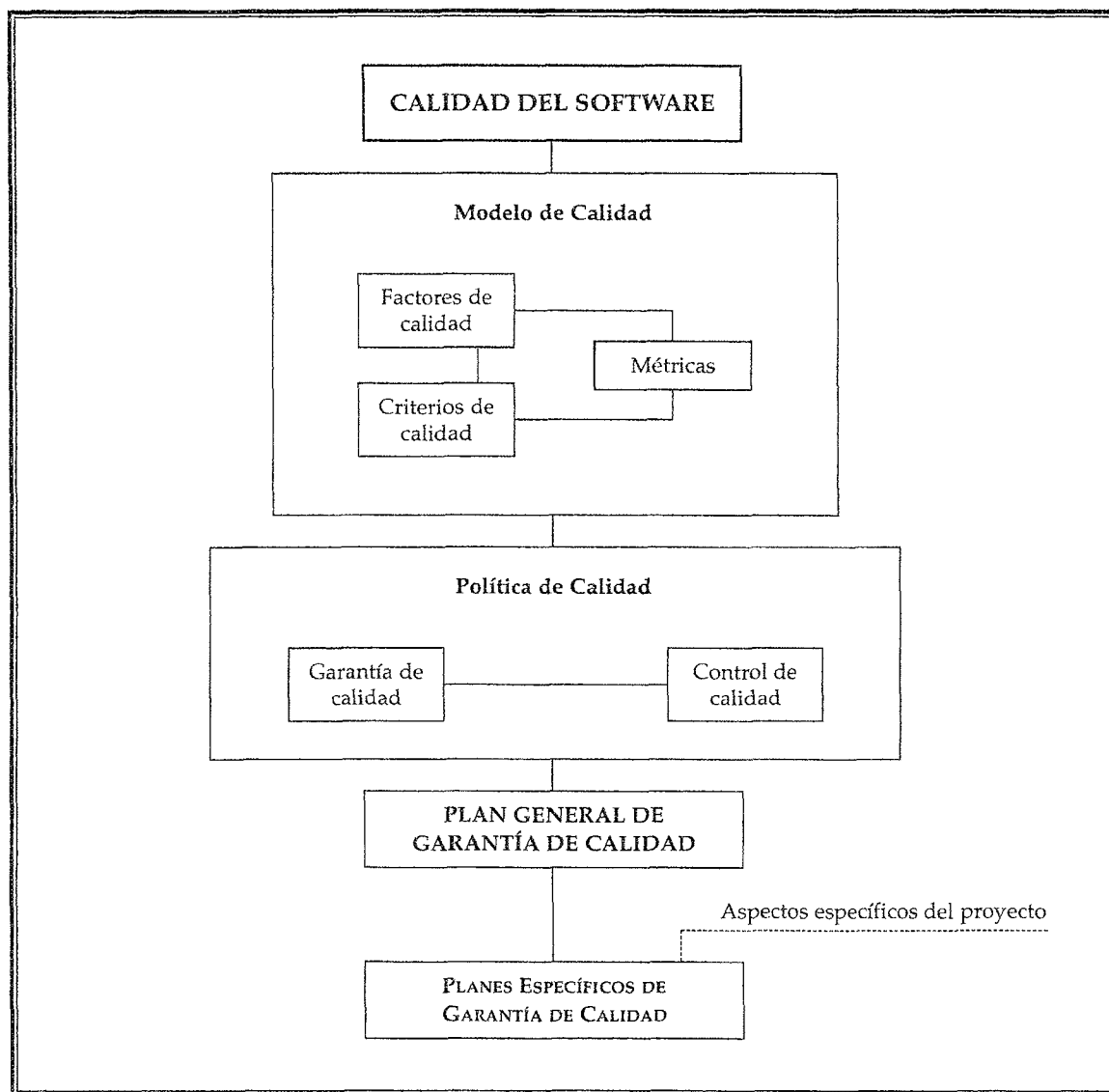
1. Comprender la terminología y los conceptos básicos que se manejan en la garantía de calidad del software.
2. Comprender las actividades de control de calidad, es decir, aquellas cuyo objetivo es evaluar la calidad que posee un producto.
3. Entender el contenido y el proceso de elaboración de un Plan General de Garantía de Calidad.

Para ello, lo primero será definir qué se entiende por calidad y su aplicación al software, para estudiar seguidamente los elementos que definen un Modelo de Calidad, esto es, los factores, criterios y métricas que intervienen en la calidad del software.

A continuación, se analizarán todos los aspectos que componen la Garantía de Calidad del software y el Control de Calidad, para seguidamente referirnos al documento que plasma todos ellos, esto es, al Plan General de Garantía de Calidad.

En este último punto se estudiará detalladamente el Plan General de Garantía de Calidad del Consejo Superior de Informática, aplicable al desarrollo de aplicaciones informáticas en el seno de la Administración Pública. En concreto se analizará todo su contenido y se expondrán algunos ejemplos aclaratorios.

Por último, se hará referencia al tratamiento que da a la Calidad del Software la metodología Métrica v.3 a través de la interface de Aseguramiento de la Calidad, exponiendo los aspectos más relevantes al efecto.



Durante los primeros tiempos de la Informática la calidad del software era una responsabilidad exclusiva del programador, ya que el desafío estaba en desarrollar el hardware de manera que se redujera

el coste del procesamiento y del almacenamiento de los datos. A partir de los años 70, como consecuencia directa del fenómeno conocido como «crisis del software», surge la necesidad de fijar la calidad como objetivo y se acuña el concepto de Garantía de Calidad del Software (SQA, Software Quality Assurance). El desafío es reducir el coste y mejorar la calidad de las soluciones basadas en computadoras, soluciones que se implementan con el software. El concepto de calidad, el control de calidad y la garantía de calidad se convierten en elementos fundamentales de todo producto software, y se hace necesaria la adopción de soluciones que pasan por definir factores cuantificables de la calidad, crear un grupo de calidad del software y elaborar un plan de garantía de calidad del software.

Actualmente los planteamientos sobre la calidad han variado de alcance, prevaleciendo ahora criterios de calidad total que buscan la satisfacción de los clientes con los bienes o servicios suministrados, y que constituyen una estrategia global de la empresa para hacerla más competitiva. Por ello, se incluirá como Anexo al presente tema un breve estudio acerca del concepto de Calidad Total y de los principales aspectos relacionados con la misma, esto es, la familia de normas ISO 9000 (2000) y el Modelo de Excelencia de la EFQM.

1. CONCEPTO DE CALIDAD. LA CALIDAD DEL SOFTWARE.

El Diccionario de la Real Academia de la Lengua Española define la calidad como «la propiedad o conjunto de propiedades inherentes a una cosa, que permiten apreciarla como igual, mejor o peor que las restantes de su misma especie». De esta definición se deduce que la calidad está relacionada con un conjunto de características variables según el producto de que se trate, que se pueden medir y, por tanto, comparar.

Además de la definición expuesta, a la hora de definir la calidad de un producto se pueden adoptar diferentes aproximaciones. Dos de ellas, extraídas de estándares internacionales, son las siguientes:

1. La calidad comprende todos aquellos aspectos o características de un producto o actividad que son significativos en cuanto a la satisfacción de los requisitos.
2. La calidad es el «conjunto de todas aquellas propiedades y características de un producto o servicio que se refieren a su capacidad para satisfacer unas necesidades implícitas o explícitas». (Definición de la ISO).

De estas definiciones se desprende que la calidad es algo relativo. Siempre va a depender de los requisitos o necesidades que se desee satisfacer; por eso, la evaluación de la calidad siempre va a implicar una comparación entre unos requisitos preestablecidos y el producto realmente desarrollado.

1.1. LA CALIDAD APLICADA AL SOFTWARE.

En el caso del software, la calidad del producto software se diferencia de la calidad de otros productos de fabricación industrial debido a las características especiales que tiene el propio software, tales como:

- Es un producto abstracto, no restringido por las leyes de la Física o por los límites de los procesos de fabricación, y su calidad también lo es.
- Se desarrolla, no se fabrica; por tanto, el coste y los errores se generan fundamentalmente en la fase de diseño y no en la de producción.

- No se deteriora con el tiempo. Los problemas que surgen durante el mantenimiento estaban allí desde el principio, es decir, no se generan nuevos errores.
- El software con errores no se rechaza. Se asume que es inevitable que el software presente errores.

También es importante destacar que la calidad de un producto software debe considerarse en todos sus estados de evolución (especificaciones, diseño, código...) y no basta con tener en cuenta sólo la calidad del producto una vez finalizado. Es decir, como primera aproximación al concepto de calidad del software, es importante diferenciar entre la calidad del producto y la calidad del proceso de desarrollo. No obstante, las metas que se fijan para la primera van a determinar las que se establezcan para la segunda, ya que la calidad del producto va a estar en función de la calidad del proceso de desarrollo.

1.2. DEFINICIÓN DE CALIDAD DEL SOFTWARE.

Si definimos el software como el «conjunto de procedimientos, normas y reglas, documentación y programas de ordenador, cuyo funcionamiento conjunto y armonizado permite satisfacer las necesidades del usuario», de esta definición surge la de calidad del software:

Calidad del Software es la «creación de productos software que, tanto eficaz como eficientemente, den completa satisfacción al usuario».

La definición expuesta merece algunos comentarios:

1. La calidad debe supeditarse al grado en el cual un cliente, en sentido amplio, percibe que el software cumple sus expectativas. La palabra «grado» implica una valoración cuantitativa. En este sentido, las métricas constituyen una herramienta que ayuda a cuantificar aspectos de la calidad, de forma que el efecto de las acciones encaminadas a la mejora de la calidad sea medible.
2. En la calidad intervienen los conceptos de eficiencia y de eficacia.

Eficiencia es la capacidad para hacer las cosas bien (do things right). Por ejemplo, los desarrolladores realizan sus actividades correctamente, cometiendo pocos errores.

Eficacia, o efectividad, es la capacidad para hacer las cosas adecuadas (do the right things). Por ejemplo, los desarrolladores llevan a cabo las tareas adecuadas.

3. La calidad tiene un enfoque dirigido a las características del producto.

Como consecuencia de ello, en primer lugar hay que documentar, luego discutir, ya que puede haber distintos puntos de vista en cuanto a las percepciones y las expectativas de cada usuario, y por último, consensuar. Estos pasos dan lugar a un proceso iterativo hasta llegar al consenso, que pasa por determinar los factores y criterios que intervienen en la calidad del software, y por cuantificar esos factores y criterios.

Otros autores han dado definiciones distintas para el concepto de calidad del software. Así, por ejemplo, el IEEE define la calidad del software como el «grado con el cual un cliente o usuario percibe que el software satisface sus expectativas». Jones señala que la calidad del software es la «ausencia de defectos o errores, siendo éstos las desviaciones respecto al comportamiento esperado», etc.

En cualquier caso, los principales problemas a los que nos enfrentamos a la hora de hablar de la calidad de un producto software son:

- La definición misma de la calidad del software. ¿Es realmente posible encontrar un conjunto de propiedades en un producto software que nos dé una indicación de su calidad? Para ello se definen los factores y los criterios de calidad.
- La comprobación de la calidad. ¿Cómo medir el grado de calidad de un producto software en función de sus propiedades? Para ello se definen las métricas de calidad.
- La mejora de la calidad del software. ¿Cómo utilizar la información disponible sobre la calidad del producto software para mejorar su calidad a lo largo del ciclo de vida de desarrollo? Para ello se definen las actividades constructivas de la garantía de calidad del software.

En los siguientes epígrafes se tratarán estos temas.

2. MODELOS DE CALIDAD DEL SOFTWARE.

Los modelos de calidad del software vienen a ayudar a la puesta en práctica del concepto de calidad del software que acabamos de exponer. Según se desprende de dicho concepto, los elementos que determinan o componen dicha calidad, y en definitiva, cualquier Modelo de Calidad, son:

- Los factores de calidad.
- Los criterios de calidad.
- Las métricas de calidad.

En los modelos de calidad, la calidad se define de forma jerárquica.

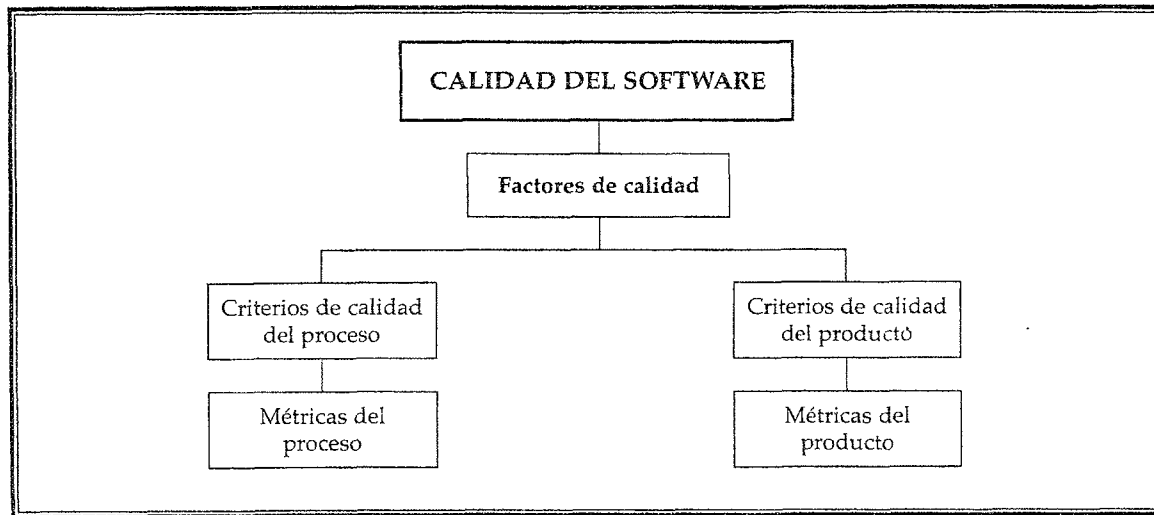
- En el nivel más alto de la jerarquía se encuentran los factores de calidad o atributos de calidad externos, que representan la calidad desde el punto de vista del usuario. Son las características que componen la calidad.
- Cada uno de los factores se descompone en un conjunto de criterios de calidad o atributos de calidad internos, que representan la calidad desde el punto de vista del producto software. Cuando están presentes contribuyen al aspecto de la calidad que representa el factor de calidad asociado.
- Para cada criterio de calidad se definen un conjunto de métricas, que son medidas cuantitativas de ciertas características del producto que, cuando están presentes, dan una indicación del grado en que dicho producto posee un determinado atributo de calidad.

La ventaja de los modelos de calidad es que ésta se convierte en algo concreto que se puede definir, medir y planificar. Por el contrario, las conexiones que se establecen entre factores, criterios y métricas se derivan de la experiencia, de ahí que existan múltiples modelos.

Uno de los modelos de calidad más clásico y extendido es el Modelo de McCall, del que han derivado otros modelos como el de Böhem, o el Modelo SQM de Murine.

2.1. EL MODELO DE CALIDAD DE MCCALL.

McCall identifica en su modelo 11 factores de calidad del software, para los cuales establece 23 criterios y 41 métricas.



2.1.1. Factores de Calidad del Software.

Los factores de calidad representan la calidad desde el punto de vista del usuario. Son, pues, elementos externos, y hacen referencia, según el Modelo de McCall, al comportamiento actual del software (operación del producto), a la facilidad de cambio del software (revisión del producto) y a la facilidad de conversión del software (transición del producto).

PUNTO DE VISTA	FACTORES DE CALIDAD
Operación del producto	<ul style="list-style-type: none">• Facilidad de uso• Integridad• Corrección• Fiabilidad• Eficiencia
Revisión del producto	<ul style="list-style-type: none">• Facilidad de mantenimiento• Facilidad de prueba• Flexibilidad
Transición del producto	<ul style="list-style-type: none">• Facilidad de reutilización• Interoperabilidad• Portabilidad

La siguiente tabla muestra los 11 factores de calidad de McCall y su definición.

MODELO DE CALIDAD DE MCCALL **FACTORES DE CALIDAD DEL SOFTWARE**

FACTORES	DESCRIPCIÓN
1. Corrección	Mide hasta qué punto un producto software cumple sus especificaciones y satisface los objetivos requeridos por el usuario.
2. Fiabilidad	Mide hasta qué punto un producto software realiza las funciones previstas en su diseño, con la precisión necesaria. Es decir, hasta qué punto se puede confiar en el funcionamiento sin errores del software.
3. Eficiencia	Mide el grado de optimización con el que el software utiliza los recursos informáticos para realizar la función que le ha sido asignada.
4. Integridad	Mide hasta qué punto se pueden controlar los accesos ilegales (por personas no autorizadas) a los programas y a los datos.
5. Facilidad de uso (disponibilidad)	Mide el esfuerzo y coste necesario para aprender, operar, preparar la entrada e interpretar la salida de un producto software. Esto es, para aprender a manejar el producto.
6. Facilidad de mantenimiento (mantenibilidad)	Mide el esfuerzo y coste necesario para localizar y corregir un error en un producto software que esté operativo.
7. Facilidad de prueba (verificabilidad)	Mide el esfuerzo y coste necesario para verificar un producto software, con el fin de garantizar que realiza la función prevista. Esto es, el esfuerzo y coste de probar un programa para comprobar que satisface sus requisitos.
8. Flexibilidad	Mide el esfuerzo y coste necesario para modificar un producto software operativo.
9. Portabilidad	Mide el esfuerzo y coste necesario para transferir un producto software de una configuración y/o entorno de hardware a otro.
10. Reutilizabilidad (reusabilidad)	Mide hasta qué punto puede utilizarse un producto software en otras aplicaciones. Es decir, hasta qué punto se puede transferir un módulo o programa a otra aplicación, y con qué esfuerzo.
11. Interoperabilidad	Mide el esfuerzo y coste necesario para acoplar un sistema a otro. Es decir, para conectar dos productos entre sí.

Al planificar la calidad de un producto software hay que seleccionar qué factores de calidad van a ser requisitos de calidad del sistema y organizarlos en orden de importancia. Para ello hay que tener en consideración: la relación con las características peculiares del producto y la fase en que éste se encuentre; el coste del factor calidad frente al beneficio que proporciona; y las interrelaciones entre factores.

2.1.2. Criterios de Calidad del Software.

Como ya se ha dicho, cada uno de los factores de calidad se descompone en un conjunto de criterios o atributos internos de calidad que, cuando están presentes, contribuyen al aspecto de la calidad que el factor asociado representa. Se trata, pues, de una visión de la calidad desde el punto de vista del producto software. Por tanto, los criterios de calidad son elementos internos o de los realizadores del software, y hacen referencia a la forma y estructura de los programas, los datos y los documentos.

La siguiente tabla recoge los 23 criterios de calidad del software, según el modelo de McCall, definiendo cada uno de ellos y mostrando los factores de calidad con que está relacionado.

MODELO DE CALIDAD DE McCALL CRITERIOS DE CALIDAD DEL SOFTWARE

CRITERIOS	DEFINICIÓN	FACTORES RELACIONADOS
1. Trazabilidad (rastreadabilidad)	Atributos del software que permiten un seguimiento desde los requisitos a la implantación.	• Corrección
2. Integridad (completitud)	Atributos del software que proporcionan una implantación completa de las funciones requeridas.	• Corrección
3. Coherencia (consistencia)	Atributos del software que proporcionan uniformidad en las técnicas y notaciones de diseño e implantación utilizadas.	• Corrección • Fiabilidad • Mantenibilidad
4. Precisión	Atributos del software que proporcionan la precisión necesaria en los cálculos y las salidas.	• Fiabilidad
5. Tolerancia al error (a fallos)	Atributos del software que posibilitan la continuidad del funcionamiento bajo condiciones no usuales.	• Fiabilidad
6. Sencillez (simplicidad)	Atributos del software que permiten implantar las funciones de la forma más comprensible posible.	• Fiabilidad • Mantenibilidad • Verificabilidad

.../...

7. Modularidad	Atributos del software que proporcionan una estructura de módulos altamente independientes.	<ul style="list-style-type: none"> • Mantenibilidad • Flexibilidad • Verificabilidad • Portabilidad • Reutilizabilidad • Interoperabilidad
8. Generalidad	Atributos del software que proporcionan amplitud a las funciones implementadas.	<ul style="list-style-type: none"> • Flexibilidad • Reutilizabilidad
9. Expansibilidad	Atributos que posibilitan la expansión del software en cuanto a capacidades funcionales y datos.	<ul style="list-style-type: none"> • Flexibilidad
10. Instrumentación	Atributos que facilitan las mediciones del uso del software y la identificación de errores durante la ejecución.	<ul style="list-style-type: none"> • Verificabilidad
11. Autodescripción	Atributos del software que proporcionan las explicaciones sobre la implantación de las funciones.	<ul style="list-style-type: none"> • Mantenibilidad • Flexibilidad • Verificabilidad • Portabilidad • Reutilizabilidad
12. Eficiencia de ejecución	Atributos del software que minimizan el tiempo de procesamiento.	<ul style="list-style-type: none"> • Eficiencia
13. Eficiencia de almacenamiento	Atributos del software que minimizan el espacio de almacenamiento necesario.	<ul style="list-style-type: none"> • Eficiencia
14. Control de acceso	Atributos del software que proporcionan el control de acceso al software y a los datos.	<ul style="list-style-type: none"> • Integridad
15. Facilidad de auditoría	Atributos del software que proporcionan los medios para auditar el acceso al software y a los datos.	<ul style="list-style-type: none"> • Integridad
16. Operatividad	Atributos del software que determinan la facilidad de los procedimientos de operación del software.	<ul style="list-style-type: none"> • Facilidad de uso
17. Facilidad de aprendizaje	Atributos del software que facilitan la familiarización inicial del usuario y el conocimiento de la operatoria actual.	<ul style="list-style-type: none"> • Facilidad de uso
18. Facilidad de comunicación (comunicatividad)	Atributos del software que proporcionan al usuario entradas y salidas útiles y fácilmente asimilables.	<ul style="list-style-type: none"> • Facilidad de uso

.../...

19. Independencia entre sistema y software	Atributos del software que determinan su independencia del entorno operativo (sistemas operativos, utilidades, etc.).	<ul style="list-style-type: none">• Portabilidad• Reutilizabilidad
20. Independencia del hardware	Atributos del software que determinan su independencia del sistema hardware.	<ul style="list-style-type: none">• Portabilidad• Reutilizabilidad
21. Compatibilidad de comunicaciones	Atributos del software que posibilitan el uso de protocolos de comunicación e interfaces estándar.	<ul style="list-style-type: none">• Interoperabilidad
22. Compatibilidad de datos	Atributos del software que posibilitan el uso y representaciones de datos estándar.	<ul style="list-style-type: none">• Interoperabilidad
23. Concisión	Atributos del software que posibilitan la implantación de una función con la menor cantidad de código posible.	<ul style="list-style-type: none">• Mantenibilidad

2.1.3. Métricas de Calidad del Software.

Al planificar la calidad de un producto hay que especificar para cada criterio el nivel de calidad que se considera aceptable; es decir, el valor mínimo o máximo aceptable. Por tanto, para cada uno de los criterios de calidad es necesario definir un conjunto de métricas, que son medidas cuantitativas de ciertas características del producto que, cuando están presentes, dan una indicación del grado en que dicho producto posee un determinado atributo de calidad.

Las métricas de calidad, en su mayor parte medidas subjetivas, permiten la cuantificación de los factores y los criterios de calidad. Fundamentan su evaluación en la medida de los factores de calidad del software, a través de los criterios de calidad, y se basan en el examen detallado de los productos ya en proceso, si bien, también requieren un análisis estático de las especificaciones, diseño y programación de las aplicaciones.

A título de ejemplo, la medida para el factor de calidad «corrección», en el que intervienen tres criterios de calidad (completitud, trazabilidad y consistencia), se calculará aplicando la fórmula:

$$(x + y + z)/3$$

x es la medida para la completitud.

y es la medida para la trazabilidad.

z es la medida para la consistencia.

McCall, al definir su modelo, da una serie de criterios para evaluar cada atributo que, en general responden a una fórmula de regresión del tipo:

$$C = R_1 \times M_1 + R_2 \times M_2 + \dots + R_n \times M_n$$

donde:

C es la medida para el criterio de calidad considerado.

R_n son los coeficientes de regresión.

M_n son las distintas métricas asociadas al criterio de calidad.

Entre las métricas de calidad son importantes las medidas de la fiabilidad del software, las cuales tratan de evaluar si el software desarrollado cumple con los requerimientos que planteó el usuario al encargarlo, y se basan en la observación de las ejecuciones de los programas que forman la aplicación. Algunos indicadores que se utilizan para evaluar la fiabilidad del software son los siguientes:

- Número de errores en el programa/número de líneas de código.
- Número de errores aparecidos/meses de uso.
- Porcentaje de usuarios con problemas.

Entonces, por ejemplo, se puede medir la fiabilidad como:

$$\text{Fiabilidad} = 1 - (\text{Número de errores en el programa/número de líneas de código})$$

Análogamente se pueden definir otras métricas como:

- Facilidad de mantenimiento = $1 - 0,1 \times (\text{Número medio de días-hombre por corrección})$.
- Portabilidad = $1 - (\text{Esfuerzo para portar/Esfuerzo para implantar})$.
- Flexibilidad = $1 - 0,05 \times (\text{Número medio de días-hombre por cambio})$.

2.2. ESTRATEGIAS DE USO DE UN MODELO DE CALIDAD.

A tenor de todo lo expuesto hasta ahora, podemos resumir el proceso a seguir para aplicar la calidad del software a un proyecto de desarrollo de un Sistema de Información.

1. El primer paso será elegir el Modelo de Calidad del Software que se va a adoptar. Existen dos estrategias al respecto:
 - a) Elegir un modelo de calidad fijo o estándar, como, por ejemplo, el de McCall, el de Boehm, o cualquier otro, en cuyo caso se aceptan los factores, criterios y métricas que define el modelo, así como las relaciones entre factores y criterios y entre criterios y métricas, y sólo es necesario seleccionar el subconjunto de factores de calidad que se van a considerar como requisitos de calidad para el proyecto concreto.

- b) Definir un modelo de calidad particular, en cuyo caso hay que adoptar un conjunto de factores de calidad como requisitos de calidad del proyecto y hay que decidir la descomposición más adecuada en criterios de calidad para los factores seleccionados.
2. En función del modelo elegido, se seleccionarán y ordenarán por orden de importancia los factores de calidad a considerar, consensuadamente entre el ingeniero del software y el usuario.

En la selección de los factores de calidad que van a ser requisitos de calidad del sistema hay que tener en consideración la relación que tienen los factores con las características peculiares del producto o proyecto, el coste del factor de calidad frente al beneficio que proporciona y las implicaciones del factor de calidad sobre el ciclo de vida.

3. Una vez elegidos los factores de calidad, el modelo proporcionará automáticamente los criterios de calidad en que se descompone cada factor y se elegirán un conjunto de métricas apropiadas a cada criterio.

Un enfoque simplista, pero que permite obtener una medida aproximada de la calidad del software, es utilizar métricas del tipo:

$$\text{Métrica} = \text{Número de defectos conocidos} / \text{Tamaño}$$

Los defectos conocidos pueden ser muy diversas cosas relacionadas con las distintas fases del ciclo de vida, por ejemplo, en la codificación, los errores sintácticos del programa.

Por su parte, el tamaño es una medida interna del producto, como, por ejemplo, el número de líneas de código, los puntos de función, el número ciclomático de McCabe, etc.

4. Por último, en función de las métricas elegidas, se establecerán para cada criterio tanto los valores deseables como los valores mínimos aceptables.

3. LA GARANTÍA DE CALIDAD DEL SOFTWARE.

La garantía de calidad del software viene definida por un conjunto de acciones que permiten asegurar que el producto responde a las necesidades expresadas por el usuario.

Reifer define la garantía de calidad del software como un «conjunto de actividades de planificación, estimación y supervisión de las actividades de desarrollo, realizadas de forma independiente al equipo de desarrollo, para asegurar que los productos software resultantes cumplen los requisitos establecidos».

Bersoff la define como un «conjunto de procedimientos, técnicas y herramientas, aplicados durante el ciclo de desarrollo de un producto, para asegurar que dicho producto satisface o excede los estándares o niveles de calidad preestablecidos».

Por último, para el IEEE la garantía de calidad del software (SQA) es un «modelo sistematizado y planificado de todas las acciones necesarias para proporcionar la adecuada confianza que el proyecto precisa para los requerimientos técnicos establecidos».

En resumen, la garantía de calidad abarca todas aquellas actividades que se realizan con el fin de asegurar un cierto nivel de calidad en el producto y supone dos tareas principales:

1. Planificar la calidad, es decir, seleccionar, clasificar y ponderar las propiedades de calidad que se van a establecer como requisitos, tanto del producto como del proceso, y elegir los mecanismos de control de calidad que se van a utilizar para medir y evaluar estas características.
2. Supervisar y construir la calidad, esto es, supervisar y corregir, si es necesario, el trabajo que se está realizando, con el objetivo de satisfacer los requisitos establecidos; y, determinar las actividades constructivas de la calidad, que son aquellas actividades preventivas cuya finalidad es evitar la introducción de errores.

La garantía de calidad de un producto software terminado se obtiene garantizando la calidad de su desarrollo; por tanto, cada actividad se debe materializar en algún documento o producto intermedio sobre el que sea posible efectuar verificaciones y validaciones. Surge así un concepto ampliamente relacionado con el de garantía de calidad, que es el de control de calidad, cuyo objetivo es comprobar si un producto posee, o no, unas determinadas características de calidad. El ámbito del control de calidad es el producto software, mientras que el ámbito de la garantía de calidad es, además, el proceso de desarrollo.

La relación entre el control de calidad y la garantía de calidad se establece a través del conjunto de personas que constituyen el grupo de garantía de calidad, el cual, es el responsable de definir el tipo de control de calidad que se va a realizar, pero no de efectuar dichos controles, que serán realizados por el equipo de desarrollo; y, además, va a utilizar los resultados del control de calidad para evaluar y mejorar el proceso de desarrollo, es decir, para conseguir productos de más calidad.

La garantía de calidad del software necesita un entorno de trabajo que cubra la gestión de proyectos, la metodología de trabajo, la gestión de la configuración, y la propia gestión de la calidad, la cual se sustenta en dos documentos:

- El Plan General de Calidad, o si se quiere, el Plan General de Garantía de Calidad, que recoge las orientaciones específicas de la organización en relación con la calidad del software y en función de la naturaleza general de los proyectos.
- El Plan Específico de Garantía de Calidad de cada proyecto software, que debe redactarse para cada proyecto específico que lo requiera por sus condiciones críticas, basándose en el Plan General de Calidad de la organización.

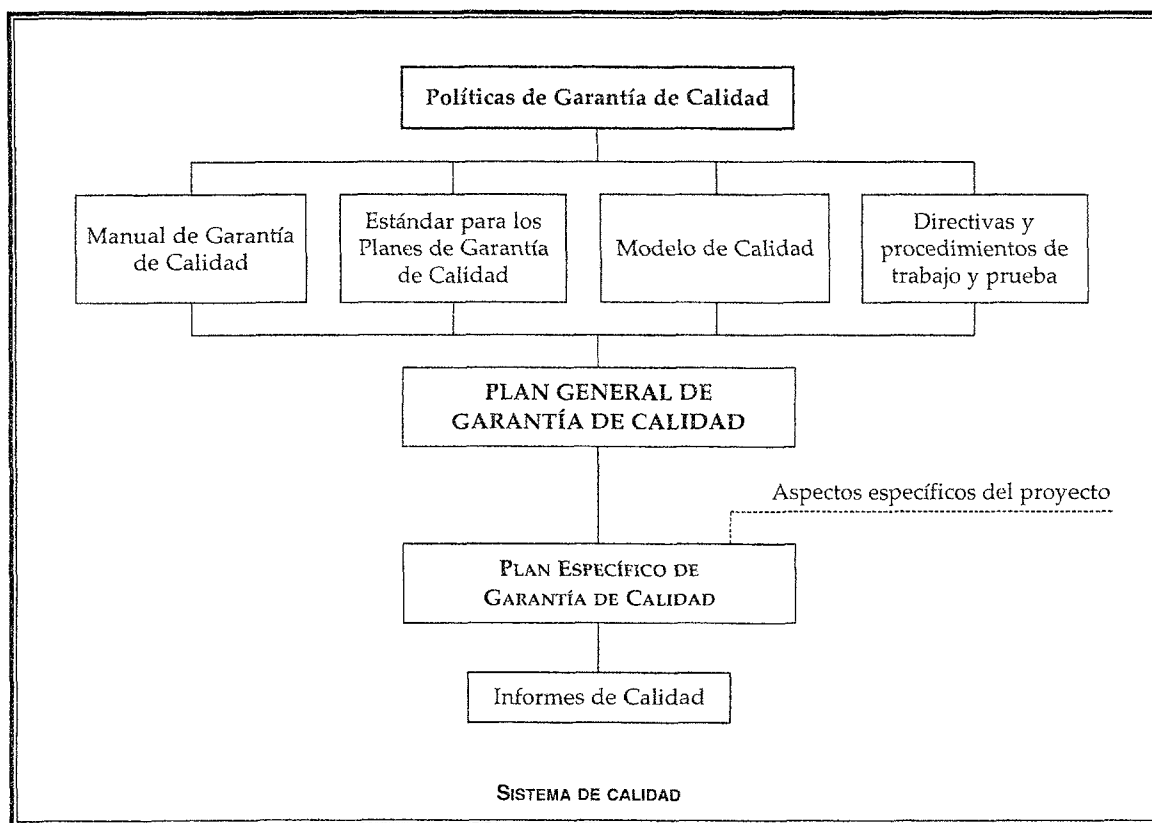
3.1. IMPLEMENTACIÓN DE LA GARANTÍA DE CALIDAD.

El tipo y número de actividades de garantía de calidad que es necesario adoptar en un proyecto depende del tamaño y complejidad de los productos software que se estén desarrollando y de otros factores como la metodología de desarrollo que se vaya a utilizar, la estructura organizativa de la empresa, la motivación del personal, etc.

El instrumento que define cómo implementar la garantía de calidad es el sistema de calidad, y es el marco en el que se establecen las diferentes estrategias, actividades y herramientas de garantía de calidad que se van a utilizar, y la forma en que se reparten las tareas y responsabilidades de garantía de calidad entre las diferentes unidades organizativas de la empresa y el personal. El sistema de calidad deberá especificar:

- La terminología, principios, responsabilidades, procesos y estándares en que se basa el sistema de calidad. Esto se documenta en el «Manual de garantía de calidad».
- El modelo de calidad en que se basa la garantía de calidad.
- Las directivas y los procedimientos de trabajo y prueba, que van a facilitar la puesta en práctica de las actividades de garantía de calidad.
- La manera de integrar las diferentes tareas de garantía de calidad en el modelo de proceso de desarrollo del software que sigue la organización o un proyecto concreto en cuestión.

Todo ello se plasma en un documento que se denomina «Plan General de Garantía de Calidad» (PGGC).



A partir del Plan General de Garantía de Calidad, es necesario desarrollar para cada proyecto su propio Plan Específico de Garantía de Calidad teniendo en cuenta los aspectos específicos que van a influir en la calidad, el cual describirá con detalle qué actividades de garantía de calidad hay que aplicar en cada una de las fases del desarrollo.

Finalmente, los resultados obtenidos con la aplicación de las diferentes actividades de garantía de calidad se reflejarán en los «informes de calidad».

4. EL CONTROL DE LA CALIDAD DEL SOFTWARE.

Como ya se ha indicado, para garantizar la calidad del software es necesario controlar la misma. El ámbito del control de calidad es, consecuentemente, el producto software, y el objetivo de las actividades de control de calidad es comprobar si un producto posee, o no, unas determinadas características de calidad.

Las actividades de control de calidad se pueden clasificar en dos categorías: análisis estáticos y pruebas dinámicas. Mientras los primeros analizan el objeto sin necesidad de ejecutarlo, las segundas requieren la ejecución del objeto que está siendo probado.

4.1. ANÁLISIS ESTÁTICOS.

Los métodos de análisis estáticos son: las auditorías, las revisiones y la verificación formal.

4.1.1. Auditorías.

Una auditoría consiste en realizar una investigación para determinar el grado de cumplimiento y la adecuación de los procedimientos, instrucciones, especificaciones, códigos, estándares u otros requisitos de tipo contractual establecidos, y la efectividad de la implementación realizada.

Se pueden considerar tres tipos de auditorías:

- Auditoría del producto, cuyo objetivo es cuantificar el grado de conformidad del producto con las características requeridas.
- Auditoría del proceso, cuyo objetivo es evaluar el proceso de desarrollo determinando dónde se puede mejorar. Se suelen realizar dos tipos de auditorías de proceso:
 - Auditorías de proyecto, para evaluar la productividad y eficacia del equipo de trabajo, y la efectividad de los métodos y herramientas utilizados.
 - Auditorías de gestión del proyecto, para evaluar la efectividad de las prácticas de gestión realizadas y la organización del proyecto.
- Auditorías del sistema de garantía de calidad, cuyo objetivo es evaluar el propio sistema de garantía de calidad establecido.

4.1.2. Revisiones.

Una revisión es una reunión formal en la que se presenta el estado actual de los resultados de un proyecto a un usuario, cliente u otro tipo de persona interesada, y se realiza un análisis estructurado de los mismos. Las revisiones redundan en una mejora directa de la calidad del objeto que se examina y provocan, indirectamente, una mejora de la calidad del proceso de desarrollo.

Las diferencias más importantes entre las revisiones y las auditorías son las siguientes:

- Las revisiones se llevan a cabo en las primeras fases del desarrollo, mientras que las auditorías se realizan en las fases finales.
- Las revisiones se usan para recoger datos; las auditorías utilizan estos datos.
- El objetivo de las revisiones es detectar defectos; el de las auditorías es monitorizar el proceso.

Hay dos tipos fundamentales de revisiones:

- Inspecciones, en las que los participantes van leyendo paso a paso el documento y comprobando el cumplimiento de los criterios de la lista de comprobación.
- Walkthroughs, en las que se demuestra la funcionalidad del objeto revisado mediante la simulación de su funcionamiento con casos de prueba y ejemplos.

Por otra parte, según la forma de llevar a cabo las revisiones se puede distinguir entre:

- Revisiones formales, si son un evento público, se informa por escrito de los resultados y todos los participantes son responsables de la calidad de la revisión.
- Revisiones informales, en caso contrario.

Por último, según el objeto que se revise, se suele distinguir entre:

- Revisiones técnicas, tales como: la revisión de la especificación de requisitos, la del diseño, la del código, la de las pruebas y la del manual de usuario.
- Revisiones del proyecto, las cuales están orientadas a la gestión y su objetivo es controlar la progresión del proyecto, evaluar los riesgos asociados al mismo y evaluar de forma general el producto.

4.1.3. Verificación formal.

La verificación formal consiste en demostrar matemáticamente la corrección de un programa con respecto a sus especificaciones. Por lo general, esta técnica sólo se utiliza en sistemas críticos, debido al coste que conlleva.

4.2. PRUEBAS DINÁMICAS.

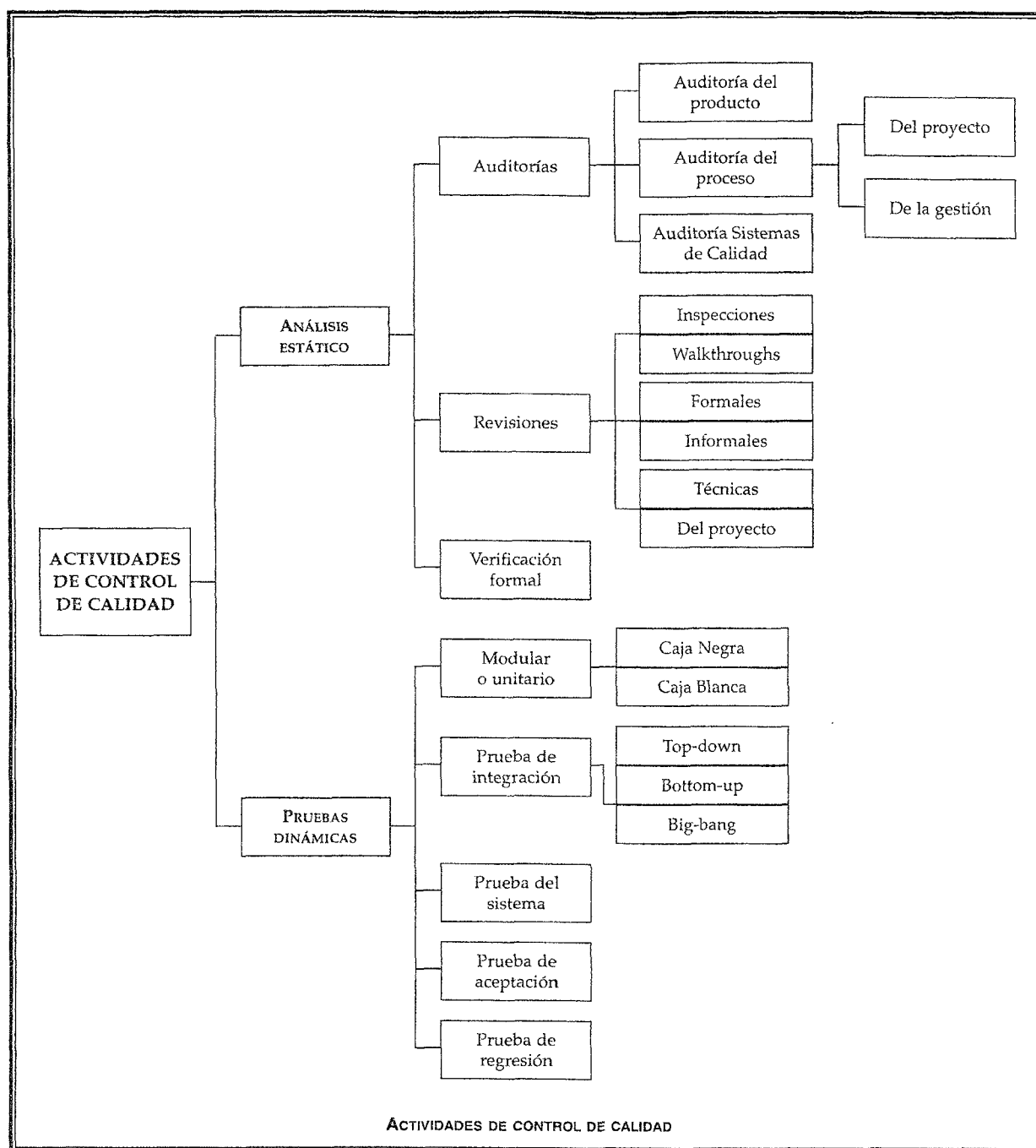
Se llaman pruebas dinámicas a aquellas que requieren la ejecución, interpretación o simulación del objeto que se está probando o de un modelo del mismo. El conjunto mínimo de pruebas que se deben realizar son las siguientes:

1. Prueba modular, prueba unitaria o prueba de componentes. Consiste en la prueba de cada módulo aislado del resto del sistema. Los métodos de prueba modular o unitaria son:
 - a) Los métodos de caja negra, también llamados prueba funcional. Consisten en probar la funcionalidad del módulo, sin tener en cuenta la estructura del mismo.

- b) Los métodos de caja blanca, también llamados prueba estructural. Consisten en probar la estructura del módulo, con independencia de la funcionalidad que tenga que satisfacer.
2. Prueba de integración. Su objetivo es comprobar que las interfaces entre los distintos módulos son correctas. Se realiza a medida que los diferentes módulos del sistema se integran en el mismo. Las posibles estrategias de integración son tres:
- a) Top-down, o de arriba a abajo. Consiste en empezar la integración y la prueba por los módulos que están en los niveles superiores.
 - b) Bottom-up, o de abajo a arriba. Consiste en empezar la integración y prueba por los módulos que están en los niveles inferiores.
 - c) Big-bang. Consiste en integrar y probar todo al mismo tiempo.
3. Prueba del sistema. Se realiza cuando se han integrado todos los módulos, y su objetivo es comprobar que el sistema satisface los requisitos del usuario, tanto funcionales como no funcionales.
4. Prueba de aceptación. Se realiza una vez que el sistema se ha implantado en su entorno real de funcionamiento, y su objetivo es demostrar al usuario que el sistema satisface sus necesidades.
5. Prueba de regresión. Tiene como objeto comprobar que toda nueva versión de un producto software es de no menos calidad que la versión anterior.

Cada uno de los diferentes tipos de pruebas implica realizar un conjunto de actividades y producir un conjunto de salidas, tal como se muestra en el siguiente cuadro.

ACTIVIDADES ESTÁNDAR DE PRUEBA	SALIDAS ESTÁNDAR ASOCIADAS
1. Planificación de la prueba	Plan de pruebas
2. Diseño de la prueba	Documento de diseño de la prueba
3. Determinación de los casos de prueba	Especificación de los casos de prueba
4. Planificación del procedimiento de prueba	Especificación del procedimiento de prueba
5. Ejecución de la prueba	Informe de los casos de prueba
6. Análisis y evaluación de la prueba	Informe de la prueba



5. EL PLAN GENERAL DE GARANTÍA DE CALIDAD.

El Consejo Superior de Informática (CSI), máximo órgano colegiado de la Administración Pública, responsable de la preparación, elaboración, desarrollo y aplicación de la política informática del Gobierno, encargó al Ministerio para las Administraciones Públicas (MAP) la elaboración de un Plan General de Garantía de Calidad (PGGC) aplicable al desarrollo de equipos lógicos, que pudiera ser puesto en práctica en cualquier sistema de información que fuera a desarrollarse dentro de la Administración Pública.

Los objetivos principales del Plan consisten en proporcionar:

- Un marco de referencia y una metodología para la elaboración de planes de garantía de calidad específicos para cada proyecto informático promovido por la Administración del Estado.
- Una metodología para la clasificación de los distintos proyectos a efectos de garantía de calidad. La metodología permite asignar a cada proyecto el modelo de desarrollo más adecuado (modelo de referencia), determinar los posibles riesgos en que se puede incurrir (perfil de riesgos) y fijar el nivel de intensidad que se debe aplicar en el control de cada fase del proyecto (foco de interés).
- Unos procedimientos e instrumentos de control de carácter genérico que puedan ser utilizados en los planes de garantía de calidad específicos de cada proyecto.

El PGGC tiene las siguientes características:

- a) En lo referente a estándares sigue las normas ISO e IEEE en materia de calidad del software.
- b) Asume como definición de calidad del software la dada por R. Pressman: «Calidad es la conformidad con los requisitos que se hayan declarado explícitamente sobre funcionalidad y rendimiento, cumplimiento de estándares o normas documentadas que se hayan establecido, y existencia de otras características implícitas que son de esperar en un producto desarrollado en un contexto de práctica profesional».
- c) El PGGC se estructura en cuatro fascículos:
 1. Guía Metodológica para la elaboración de Planes de Garantía de Calidad.
 2. Esquema formal para la clasificación de Proyectos Informáticos.
 3. Procedimientos de Control de Calidad.
 4. Instrumentos de control y elementos auxiliares de Control de Calidad.

Los dos primeros fascículos constituyen el Bloque Metodológico, el cual hace referencia a una serie de técnicas y métodos de orden práctico diseñados específicamente para ser aplicados en el diseño y definición de los planes específicos de garantía de calidad de los proyectos informáticos promovidos por la Administración del Estado.

Los fascículos tercero y cuarto forman el Bloque Instrumental, el cual contiene un conjunto seleccionado de procedimientos e instrumentos de garantía de calidad, que aplicados de acuerdo con los métodos definidos en el bloque anterior, permitirán su puesta en práctica en proyectos concretos de desarrollo de aplicaciones.

5.1. GUÍA METODOLÓGICA PARA LA ELABORACIÓN DE PLANES DE GARANTÍA DE CALIDAD.

La Guía Metodológica para la elaboración de Planes de Garantía de Calidad, que constituye el fascículo 1 del PGGC para el desarrollo de equipos lógicos, del Consejo Superior de Informática, establece básicamente:

1. Los agentes que deben intervenir en el desarrollo de un Proyecto Informático.

2. Una metodología para la elaboración de planes específicos de control de calidad, aplicables a proyectos de desarrollo de sistemas de información.

5.1.1. Agentes que intervienen en un proyecto.

A los efectos de este Plan General de Garantía de Calidad, se considerará que en un determinado proyecto los agentes que intervienen en relación con la garantía de calidad son los que se indican en la siguiente tabla:

AGENTES	RESPONSABILIDAD
USR: Organismo usuario o promotor del proyecto. Estará representado por las personas que, perteneciendo al mismo, se consideren oportunas.	Es el cliente a todos los efectos, y es quien demanda la calidad del desarrollo. Intervendrá en determinados procedimientos de revisión y en las pruebas de aceptación de la aplicación.
DIR: Director del Proyecto. Lo nombra el organismo usuario como responsable del proyecto por parte de la Administración.	Es el responsable de la definición de las líneas generales y de la aprobación del Plan de Garantía de Calidad correspondiente, e impulsará su ejecución y puesta en práctica.
EDS: Equipo de Desarrollo. Está constituido por las personas responsables del desarrollo del software, ya sean del organismo usuario, de una empresa contratista o de ambos.	Es el responsable de la producción de los distintos componentes del software, y es el sujeto pasivo de todos los procedimientos ordinarios de control (revisiones y pruebas).
EGC: Equipo de Garantía de Calidad. Está constituido por las personas responsables del ejercicio de estas funciones, ya pertenezcan al organismo usuario o a una empresa externa; pero, en cualquier caso, será independiente del EDS en el contexto del proyecto.	Es el encargado del diseño detallado del Plan de Garantía de Calidad del proyecto, y es el sujeto activo principal en los procedimientos ordinarios de control incluidos en el Plan. A su vez, podrá ser sujeto pasivo si se llevan a cabo procedimientos extraordinarios.
AUD: Equipo de Auditoría. Es potestativo. En su caso, estará formado por las personas responsables del ejercicio de estas funciones, ya pertenezcan al organismo usuario o a una empresa externa; pero, en cualquier caso, será independiente del EDS y del EGC en el contexto del proyecto.	Intervendrá exclusivamente en aquellos casos en los que se hayan previsto procedimientos extraordinarios de control (auditorías), razón por la que la presencia de este agente no tendrá lugar en la mayoría de los proyectos.

5.1.2. Metodología para la elaboración de Planes específicos de Garantía de Calidad, aplicables a proyectos de desarrollo de aplicaciones informáticas.

El procedimiento que se seguirá para la elaboración del Plan de Garantía de Calidad de un proyecto de desarrollo de aplicaciones informáticas comprende la realización secuencial de las siguientes actividades:

1. Actuaciones preliminares.
2. Caracterización del proyecto a efectos de garantía de calidad.
3. Selección y adaptación de procedimientos de control de calidad.
4. Selección y adaptación de instrumentos de control y elementos auxiliares de garantía de calidad.
5. Redacción y aprobación del Plan de Garantía de Calidad del proyecto.

Las actuaciones preliminares se llevan a cabo antes de comenzar la elaboración del Plan de Garantía de Calidad de un proyecto determinado y consisten en:

- a) Constitución del grupo de representantes del organismo usuario en el proyecto (USR).
- b) Designación del Director del Proyecto (DIR).
- c) Elaboración de las especificaciones de usuario para el desarrollo de la aplicación y, en su caso, del Pliego de Condiciones para la contratación. Este documento es fundamental para las funciones de garantía de calidad del proyecto, ya que constituye su punto de referencia.

La caracterización del proyecto a efectos de garantía de calidad consiste en aplicar el Esquema Formal de Clasificación (EFC) de los proyectos informáticos, esto es:

- a) Obtener el Diagrama Característico (DC) del proyecto.
- b) Obtener, a partir del DC, el modelo de referencia para el desarrollo.
- c) Obtener, a partir del DC, el perfil de riesgos del proyecto.
- d) Obtener, a partir del DC y del modelo de referencia para el desarrollo, el foco de interés en garantía de calidad.

La selección y adaptación de procedimientos de control de calidad consiste en seleccionar, a partir del modelo de referencia para el desarrollo del proyecto y de su foco de interés en garantía de calidad, los procedimientos genéricos de control a aplicar en cada una de las fases y componentes, y aplicarlos, directamente, como indica el fascículo 3, o mediante algún tipo de adaptación.

La selección y adaptación de instrumentos de control y elementos auxiliares de garantía de calidad supone seleccionar, en función del procedimiento de control que se haya de aplicar, los instrumentos y elementos auxiliares necesarios para llevar a cabo dicho procedimiento, y utilizarlos como indica el fascículo 4 o mediante alguna adaptación.

Finalmente, la redacción y aprobación del Plan de Garantía de Calidad del proyecto supondrá redactar el documento denominado Plan de Garantía de Calidad, que se ajustará a una determinada estructura, y en su aprobación por parte del Director del Proyecto (DIR).

5.2. ESQUEMA FORMAL PARA LA CLASIFICACIÓN DE LOS PROYECTOS INFORMÁTICOS.

El fascículo 2 del PGGC establece un esquema formal para clasificar los proyectos informáticos (EFC), que permitirá determinar los tres elementos esenciales para definir su plan particular de garantía de calidad. Estos elementos son:

- a) El modelo de referencia para el proceso de desarrollo, que permite identificar las distintas fases del desarrollo y los productos o componentes a obtener en cada una.
- b) El perfil de riesgos del proyecto, que permite la detección y evaluación preliminar de los riesgos extraordinarios que pueden comprometer el desarrollo del proyecto.
- c) El foco de interés en garantía de calidad para el control de riesgos ordinarios, que permite detectar las fases y/o productos del desarrollo que son potencialmente más críticos, y en los cuales se deberá ejercer un control más intenso.

Para llevar a cabo la clasificación del proyecto se partirá del documento de especificaciones de usuario o del pliego de condiciones para la contratación y se obtendrá, en primer lugar, el diagrama característico del proyecto, que determina su clasificación tipológica a efectos de garantía de calidad. A partir del diagrama característico se determinará el modelo de referencia más adecuado para el desarrollo, el perfil de riesgos del proyecto y, finalmente el foco de interés en garantía de calidad.

En los apartados que siguen se estudian cada uno de estos pasos.

5.2.1. Diagrama característico del proyecto.

La finalidad del Diagrama característico (DC) es estimar los valores de los atributos o factores característicos del proyecto a desarrollar, los cuales determinarán la clasificación tipológica del proyecto a efectos de garantía de calidad.

Los factores característicos o atributos de un proyecto que considera el PGGC son un total de dieciocho, agrupados en tres grupos homogéneos, tal como se indica a continuación. Para todos los atributos se estimará un valor comprendido entre 1 y 5.

1. Grupo 1: Atributos relativos al producto que se va a desarrollar.

Estos atributos hacen referencia exclusivamente a las características intrínsecas de la aplicación informática. Son los siguientes:

1. Dimensión (DIM): se refiere al tamaño del software, el cual se determinará a través del número de líneas de código fuente (KLDC), o a través del número de puntos de función ajustados (PF). En cualquier caso, para asignar valor al atributo se suele tener en cuenta el esfuerzo de desarrollo correspondiente al tamaño estimado, según el siguiente baremo:

ESFUERZO	DIMENSIÓN	VALOR DEL ATRIBUTO
• Entre 1 y 6 meses-hombre	Pequeño	1
• Entre 6 y 24 meses-hombre	Moderado	2
• Entre 24 y 60 meses-hombre	Medio	3
• Entre 60 y 120 meses-hombre	Grande	4
• Mayor o igual que 120	Muy grande	5

2. Complejidad (COMP): se refiere a la complejidad intrínseca del software a desarrollar. Es un factor muy subjetivo que se debe valorar en función del conocimiento que se tenga de la aplicación o a través de las métricas de complejidad de Halstead o de McCabe.
3. Requisitos de fiabilidad (FIAB): se entiende por fiabilidad la probabilidad de que la aplicación funcione en las condiciones previstas de operación, según los requisitos expresados en las especificaciones de diseño. Es decir, sin ningún tipo de fallos.
4. Requisitos de seguridad (SEC): se entiende por nivel de seguridad la exigencia del cumplimiento de una serie de requisitos como: restricciones de acceso, ante accesos no deseados dificultar la filtración de información, etc.
5. Requisitos de comportamiento externo (CEX): refleja las exigencias que pueden existir en cuanto a funcionalidad y rendimiento de la aplicación desde el punto de vista del usuario final.
6. Requisitos de comportamiento interno (CIN): este atributo refleja las limitaciones y restricciones que pueden existir en relación con factores como: ocupación máxima en memoria del código objeto, tamaño máximo del código fuente, etc.
7. Grado de definición, estructura y modularidad de las especificaciones (DESP): este atributo hace referencia al nivel de detalle, grado de precisión y modularidad del documento de especificaciones para la contratación del desarrollo de que se trate.

II. Grupo 2: Atributos relativos al entorno de implantación previsto.

Dentro de este grupo se incluyen los atributos que hacen referencia a las características principales del entorno en que se va a implantar la aplicación informática. Antes de enumerar estos atributos, veamos el concepto de máquina virtual.

Se entiende por «máquina virtual» el conjunto estructurado de todos los recursos informáticos (hardware + software) necesarios para la operación de una aplicación determinada. Los componentes de la «máquina virtual» se pueden agrupar en tres niveles:

- Nivel 1: hace referencia a la plataforma hardware sobre la que actúa la aplicación.
- Nivel 2: hace referencia al sistema operativo y a los demás productos de software de sistemas que se sitúan directamente sobre la plataforma hardware.

- Nivel 3: se refiere al software de carácter intermedio sobre los que se apoya la aplicación (compiladores, gestores de bases de datos, sistemas de seguridad, etc.).

Los atributos relativos al entorno previsto de implantación son los siguientes:

8. Tipología de la «máquina virtual» (TPMV): se refiere a las características técnicas de los elementos que componen los dos primeros niveles de la máquina virtual, es decir, plataforma hardware y software de sistemas. El valor de este atributo estará en correspondencia con el grado de sofisticación de dichos componentes.
9. Funcionalidad de la «máquina virtual» (FCMV): se refiere al nivel de servicios que proporciona la máquina virtual a la aplicación que actúa sobre la misma. La valoración se realiza en sentido inverso. Es decir, una máquina rica en funcionalidad tendrá valores bajos, mientras que una máquina pobre tendrá valores altos.
10. Grado de distribución y heterogeneidad de la «máquina virtual» de implantación (DHMV): este atributo se refiere al número de plataformas de que consta la máquina virtual y a las diferencias o similitudes entre las mismas en cualquiera de los tres niveles.
11. Carga de trabajo (CTR): se refiere a la carga de trabajo a que se verá sometida la aplicación en su entorno de implantación. Los parámetros que la caracterizan son, entre otros: el número de usuarios concurrentes, el régimen transaccional, etc.
12. Nivel de interacción con otras aplicaciones o datos (INT): este atributo se refiere a la existencia de interfaces entre la aplicación que se va a desarrollar y otras aplicaciones o estructuras de datos ajenas a la misma, existentes en el entorno de implantación.
13. Diferencias entre los entornos de desarrollo y de implantación (DIFE): este atributo refleja el grado de disparidad o semejanza que puede existir, tanto desde el punto de vista técnico como organizativo, entre los entornos de implantación y desarrollo.

III. Grupo 3: Atributos relativos al propio proyecto o proceso de desarrollo.

Dentro de este grupo se incluyen los atributos que hacen referencia a las características propias del proyecto y del proceso de desarrollo del mismo. Son los siguientes:

14. Coste total estimado del proyecto (COST): es el coste presupuestado para el desarrollo, tal como se ha estimado por el usuario. Los valores se dan según la escala:
 - Coste menor o igual que 18.000 euros (3 millones de ptas.).
 - Coste entre 18.000 y 60.000 euros (3 y 10 millones de ptas.)
 - Coste entre 60.000 y 150.000 euros (10 y 25 millones de ptas.)
 - Coste entre 150.000 y 600.000 euros (25 y 100 millones de ptas.)
 - Coste mayor o igual que 600.000 euros (100 millones de ptas.).

15. Plazo estimado de desarrollo (PLZ): es el límite que no debe sobrepasarse en cuanto al tiempo transcurrido entre el inicio del desarrollo y su finalización satisfactoria.

- Plazo menor o igual que 3 meses.
- Plazo entre 3 y 6 meses.
- Plazo entre 6 meses y un año.
- Plazo entre uno y dos años.
- Plazo mayor o igual que 2 años.

16. Estabilidad del proyecto (EPRY): se refiere al grado de estabilidad de las especificaciones a lo largo del desarrollo del proyecto. Entre los factores que se consideran para valorar este atributo se incluyen: el grado de definición y la precisión de las especificaciones, el número de interlocutores y centros afectados por el proyecto, la neutralidad del problema respecto a disposiciones normativas de probable aparición, etc.

17. Evaluación previa del contratista (ECON): hace referencia al nivel de incertidumbre o desconfianza que se pueda tener sobre el contratista, en caso de contratación externa, o sobre el equipo de desarrollo, en caso de desarrollos internos.

18. Disponibilidad de recursos para garantía de calidad (REC): este atributo pretende reflejar la disponibilidad real que existe por parte del usuario o promotor del proyecto para llevar a cabo el plan de calidad que se diseñe en cada caso, sin limitaciones significativas (tiempo, presupuesto, personal, etc.). La valoración de este atributo se realiza en sentido inverso. Esto es, cuantas más limitaciones existan, menor valor.

Una vez valorados los atributos del proyecto, se elaborará el diagrama característico del mismo, el cual no es más que un reflejo gráfico de la evaluación del proyecto en relación con cada uno de los dieciocho atributos señalados.

A un atributo se le puede dar un valor único si se tiene una certeza total en su estimación, o bien, un rango de valores si no se está en condiciones de estimarlo de forma suficientemente exacta.

La disposición gráfica del diagrama característico es la que se muestra a continuación:

ATRIBUTOS			1	2	3	4	5
01	Dimensión	DIM					
02	Complejidad	COMP					
03	Fiabilidad	FIAB					
04	Seguridad	SEC					
05	Comportam. externo	CEX					
06	Comportam. interno	CIN					
07	Definición especificaciones	DESP					

.../...

.../...

08	Tipología máquina virtual	TPMV					
09	Funcionalidad máquina virtual	FCMV					
10	Distribución máquina virtual	DHMOV					
11	Carga de trabajo	CTR					
12	Interacción otras aplicaciones	INT					
13	Diferencia entornos	DIFE					
14	Coste del proyecto	COST					
15	Plazo de desarrollo	PLZ					
16	Estabilidad del proyecto	EPRY					
17	Evaluación contratista	ECON					
18	Disponibilidad recursos	REC					

5.2.2. Modelo de referencia del proceso de desarrollo.

El modelo de referencia para el desarrollo permite identificar y describir:

- Cada una de las fases en que se puede descomponer el proceso de desarrollo del proyecto.
- La naturaleza y el contenido de los trabajos a realizar por el contratista en cada fase.
- Los distintos productos y componentes a obtener y entregar como resultado de cada fase.

El PGGC considera cinco modelos genéricos de referencia aplicables a los desarrollos de aplicaciones promovidas por la Administración del Estado. Estos modelos son:

1. Modelo secuencial básico.
2. Modelo secuencial intermedio.
3. Modelo secuencial detallado.
4. Modelo de evolución de prototipos.
5. Modelo de desarrollo modular.

Las fases de que consta cada uno de los modelos de referencia que contempla el PGGC, son las siguientes:

1. Modelo Secuencial Básico. F1: diseño. F2: programación. F3: implantación y pruebas de aceptación.
2. Modelo Secuencial Intermedio. F1: elaboración de las especificaciones de diseño. F2: diseño funcional. F3: diseño técnico detallado. F4: programación. F5: implantación y pruebas de aceptación.

3. Modelo Secuencial Detallado. F1: planificación del desarrollo. F2: elaboración de las especificaciones de diseño. F3: diseño funcional. F4: diseño técnico detallado. F5: programación. F6: integración. F7: implantación y pruebas de aceptación.

4. Modelo de Desarrollo Evolutivo por Prototipos.

Procesos de experimentación (k ciclos). E1: especificaciones del prototipo. E2: diseño rápido. E3: programación del prototipo. E4: evaluación y refinamiento.

Proceso de producción (secuencial). P1: especificaciones finales del diseño. P2: diseño final. P3: programación. P4: implantación y pruebas de aceptación.

5. Modelo de desarrollo modular.

F1: estudio preliminar y planificación del desarrollo.

F2: desarrollo (k módulos paralelos): F2.1: especificaciones. F2.2: diseño (F2.2.1: diseño funcional. F2.2.2: diseño técnico). F2.3: programación. F2.4: pruebas.

F3: integración de módulos.

F4: pruebas de aceptación.

El siguiente cuadro muestra los productos o «entregables» que se obtienen en cada fase, en función del modelo de referencia para el desarrollo de que se trate.

PRODUCTOS O «ENTREGABLES»	MODELOS DE REFERENCIA PARA EL DESARROLLO				
	SEC. BÁS.	SEC. INT.	SEC. DET.	EV. PROT.	DES. MOD.
DBP: Doc. Base de Planificación			F1		F1
DED: Doc. Especificacs. de Diseño	F1	F1	F2	P1	F2.1
DDD: Doc. Descripción del Diseño	F1			P1 (1)	F2.1 (3)
DDF: Doc. de Diseño Funcional		F2	F3	P2 (2)	F2.2.1 (4)
DDT: Doc. de Diseño Técnico		F3	F4	P2 (2)	F2.2.2 (4)
DTP: Doc. Técnica de Programación	F2	F4	F5	P3	F2.3
DOP: Documentación de Operación	F3	F5	F7	P4	F4
DRU: Doc. de Referenc. para Usuar.	F3	F5	F7	P4	F4
IEP: Informe de Evoluc. del Protot.				E4	
Código Fuente	F2	F4	F5	P3	F2.3
APL: Aplicación (sw. ejecutable)	F3	F5	F6, F7	P4	F2.4, F3, F4

(1) Si el proceso de producción sigue un modelo secuencial básico.

(2) Si el proceso de producción sigue un modelo secuencial intermedio.

(3) Para los módulos que sigan un modelo secuencial básico.

(4) Para los módulos que sigan un modelo secuencial intermedio.

Una vez obtenido el Diagrama Característico (DC) del proyecto, el procedimiento para determinar el modelo de referencia para el desarrollo que más se ajusta a las características de dicho proyecto consiste en evaluar los Factores de Discriminación del Proyecto en relación con cada uno de los cinco modelos de referencia, contrastando el Diagrama Característico del proyecto con la plantilla de comparación correspondiente a cada modelo y eligiendo aquel modelo que presente menor factor de discriminación.

Como ejemplo se adjuntan las plantillas de comparación del Modelo Secuencial Básico y del Modelo Secuencial Intermedio.

MODELOS DE REFERENCIA PARA EL DESARROLLO. PLANTILLAS DE COMPARACIÓN

Modelo Secuencial Básico

ATRIBUTOS		1	2	3	4	5
01	DIM					
02	COMP					
03	FIAB					
04	SEC					
05	CEX					
06	CIN					
07	DESP					
08	TPMV					
09	FCMV					
10	DHMOV					
11	CTR					
12	INT					
13	DIFE					
14	COST					
15	PLZ					
16	EPRY					
17	ECON					
18	REC					

Modelo Secuencial Intermedio

ATRIBUTOS		1	2	3	4	5
01	DIM					
02	COMP					
03	FIAB					
04	SEC					
05	CEX					
06	CIN					
07	DESP					
08	TPMV					
09	FCMV					
10	DHMOV					
11	CTR					
12	INT					
13	DIFE					
14	COST					
15	PLZ					
16	EPRY					
17	ECON					
18	REC					

5.2.3. Perfil de Riesgos del proyecto.

El perfil de riesgos del proyecto permite la detección y evaluación preliminar de los riesgos potenciales, especialmente los de carácter extraordinario, que pueden comprometer el desarrollo del proyecto. A estos efectos, el Plan General de Garantía de Calidad del Consejo Superior de Informática considera los siguientes siete tipos de riesgos potenciales:

1. R1: defectos graves y recurrentes en el comportamiento externo de la aplicación, bien sea por mala adecuación funcional o por problemas de rendimiento, falta de fiabilidad, seguridad, etc.
2. R2: baja calidad de los distintos productos obtenidos en las fases del desarrollo, lo que implica gravemente la mantenibilidad de la aplicación y la gestión de su configuración.

3. R3: dificultades graves de implantación de la aplicación por mala adecuación a su entorno real de operación, debidas a un bajo nivel de funcionalidad de la máquina virtual, falta de homogeneidad entre los entornos de desarrollo e implantación, etc.
4. R4: imposibilidad práctica de mantener los costes de desarrollo en consonancia con los límites establecidos en la contratación, ya sea por la necesidad de realizar ampliaciones o revisiones importantes del proyecto, o por la aparición de costes ocultos.
5. R5: incumplimiento grave de los plazos de ejecución.
6. R6: imposibilidad de gestionar y controlar adecuadamente el desarrollo del proyecto, por razones prácticas de inmanejabilidad del mismo.
7. R7: inconclusión del proyecto, que correspondería a una situación extrema en alguno de los problemas anteriores.

El procedimiento de obtención del perfil de riesgos de un proyecto concreto es el siguiente:

1. A partir del Diagrama Característico del proyecto en cuestión, se obtendrá para cada uno de los siete tipos de riesgos señalados (R_i) el valor del Coeficiente de Divergencia del riesgo (CD_i), el cual se calculará como la diferencia de los valores medios que en el Diagrama Característico del proyecto posean los denominados grupos de atributos antagonistas (grupo de atributos A y grupo de atributos B).

Por ejemplo, para el tipo de riesgo R1, los grupos de atributos antagonistas que considera el PGGC son: grupo A: FIAB, SEC, CEX, CTR, ECON; grupo B: DESP, PLZ, REC.

2. Una vez obtenidos los Coeficientes de Divergencia, se obtendrá el diagrama correspondiente al perfil de riesgos, representándose en abscisas los siete tipos de riesgos, y en ordenadas los valores correspondientes a sus respectivos Coeficientes de Divergencia. Los umbrales de referencia que se establecen son:

- Riesgos potenciales extraordinarios si: $CD_i \geq 3$
- Riesgos potenciales ordinarios si: $-3 < CD_i < 3$
- Ineficiencia potencial de la asignación de recursos si: $CD_i \leq -3$

Según esto, los criterios de interpretación de los valores de CD_i obtenidos es el siguiente:

- Si $0 \leq CD_i < 3$, estamos ante un nivel ordinario de riesgos previos en relación con el factor considerado, que pueden ser controlados.
- Si $3 \leq CD_i \leq 5$, el nivel de riesgos previos es extraordinario y por tanto, muy difícilmente controlable. En este caso habrá que redefinir el proyecto para reducir el desajuste entre los atributos de los grupos A y B hasta valores admisibles.
- Si $0 \geq CD_i > -3$, el planteamiento del proyecto es suficientemente satisfactorio en relación con el tipo de riesgo considerado. Es decir, nivel de riesgo ordinario.
- Si $-3 \geq CD_i \geq 5$, el planteamiento inicial del proyecto resulta desajustado por defecto, en relación con el tipo de riesgo considerado, y aunque esto no es negativo desde el punto de vista de la garantía de calidad, por razones de eficiencia en la asignación de recursos, conviene que se proceda a la redefinición del proyecto mediante el reajuste de los atributos correspondientes.

5.2.4. Foco de interés en garantía de calidad.

El foco de interés en garantía de calidad permite determinar, para cada una de las fases y productos del modelo de desarrollo elegido para un proyecto en cuestión, un nivel de intensidad que sirva para graduar las funciones de garantía de calidad.

Los niveles de intensidad que considera el PGGC son:

- Nivel 1: intensidad nominal.
- Nivel 2: intensidad especial.
- Excepcionalmente, el PGGC también considera un nivel 0: intensidad mínima, que no se obtiene del foco de interés, y que sólo es aplicable en situaciones en las que no se disponga de recursos para abordar la ejecución de las tareas regulares de garantía de calidad de la forma apropiada.

El procedimiento para determinar el foco de interés consiste en aplicar sobre el Diagrama Característico del proyecto de que se trate, una plantilla específica, la cual está en función del modelo de referencia para el desarrollo que corresponda al proyecto en cuestión. La plantilla consta de una serie de zonas encuadradas, que señalan la conveniencia de adoptar un grado de intensidad especial (nivel 2) en el control de calidad de las fases y productos del desarrollo que se indican dentro del recuadro.

A título de ejemplo se adjuntan las plantillas de comparación correspondientes al modelo secuencial básico y al intermedio.

PLANTILLAS DE COMPARACIÓN PARA LA DETERMINACIÓN DEL FOCO DE INTERÉS

Modelo Secuencial Básico

ATRIBUTOS	1	2	3	4	5
DIM					
COMP					
FIAB					
SEC				de F1 a F3	
CEX				F3	
CIN					
DESP					
TPMV					
FCMV					
DHMOV					
CTR					
INT					
DIFE					
COST					
PLZ	F3				
EPRY					
ECON				F3	F1 a F3
REC					

Modelo Secuencial Intermedio

ATRIBUTOS	1	2	3	4	5
DIM				F1, F2	
COMP				F3, F5	
FIAB					
SEC				de F1 a F5	
CEX					
CIN				F3 y F5	
DESP					
TPMV					
FCMV			F3		
DHMOV					
CTR				F5	
INT					
DIFE				F5	
COST				F1 a F4	
PLZ	F5				
EPRY					
ECON				F5	F1 a F5
REC					

5.3. PROCEDIMIENTOS DE CONTROL DE CALIDAD.

El fascículo 3 del PGGC define una serie de procedimientos genéricos para realizar el control de calidad de los productos resultantes en los distintos modelos de referencia. La elección del procedimiento de control para cada producto o «entregable» dependerá del producto en sí y del nivel de intensidad que hubiera resultado en la determinación del foco de interés en garantía de calidad para la fase a la que corresponda el producto.

El Plan General de Garantía de Calidad clasifica los procedimientos genéricos de control de calidad en los siguientes cuatro grupos:

1. Revisiones.
2. Pruebas.
3. Procedimientos extraordinarios.
4. Procedimientos particulares.

5.3.1. Revisiones.

Las revisiones son procedimientos de control aplicables a productos de tipo documental.

Según el nivel de intensidad que hubiera resultado en la determinación del foco de interés en garantía de calidad para la fase a la que corresponda el producto, el PGGC distingue los siguientes tipos de revisiones:

- Revisiones Mínimas (RM). Son procedimientos sencillos de control, consistentes en el examen de las características más visibles de los diferentes productos de tipo documental obtenidos durante el desarrollo. Se aplicarán con carácter excepcional cuando existan limitaciones muy severas en la disponibilidad de recursos para abordar las tareas de garantía de calidad. Corresponden a un nivel de intensidad mínimo (nivel 0) en el control de calidad.
- Revisiones Técnicas Formales (RTF). Se aplicarán a los productos documentales obtenidos en aquellas fases del desarrollo para las que el nivel de intensidad resultante en la determinación del foco de interés sea el nominal, es decir, nivel 1.

El objetivo de una RTF consiste en la evaluación sistemática de los documentos obtenidos durante el desarrollo, con el fin de garantizar que resultan conformes, en cuanto a sus aspectos formales y contenido técnico, con las especificaciones, normas o estándares que se hayan establecido en el Plan de Garantía de Calidad específico del proyecto.

- Inspecciones Detalladas (ID). Se aplicarán a los productos documentales obtenidos en aquellas fases del desarrollo para las que el nivel de intensidad resultante en la determinación del foco de interés sea el especial, es decir, nivel 2.

El objetivo de una Inspección Detallada consiste en la detección e identificación de cualquier tipo de defectos, errores o problemas que puedan afectar a un documento obtenido al final de una fase de desarrollo. La principal diferencia con las RTF radica en que mientras que éstas tienden a comprobar la adecuación del producto documental a las normas o estándares establecidos, las Inspecciones Deta-

lladas se dirigen explícitamente a la búsqueda de fallos y deficiencias de cualquier tipo. Son, por tanto, procedimientos de control muy exhaustivos que obedecen a una fuerte exigencia de garantía de calidad en la fase correspondiente.

5.3.2. Pruebas.

Las Pruebas son procedimientos de control que se aplican a productos o componentes de tipo software ejecutable, es decir, programas en formato y soporte adecuados. Estos programas pueden responder, bien a la aplicación en su totalidad, o bien a partes homogéneas de la misma (módulos).

El PGGC contempla la realización de las siguientes clases de pruebas:

- Pruebas de Validación de Módulos (PVM).

El objetivo de estas pruebas es comprobar que el comportamiento de los programas correspondientes a un módulo determinado de la aplicación, cumple con los requisitos de funcionalidad y rendimiento establecidos en los documentos de especificaciones y de diseño.

- Pruebas de Integración (PI).

El objetivo de estas pruebas es comprobar que el funcionamiento conjunto de los programas correspondientes a los distintos componentes y módulos de la aplicación es correcto, para lo cual deberá investigarse el comportamiento de las interfaces existentes entre los mismos.

- Pruebas de Aceptación de la Aplicación (PAA). La realización de estas pruebas es un procedimiento de control para todos los modelos de referencia para el desarrollo. Las Pruebas de Aceptación deben garantizar el correcto funcionamiento del software final que corresponde a la aplicación completa en su versión definitiva, sobre la máquina virtual que exista en el entorno de implantación.

5.3.3. Procedimientos extraordinarios.

El Plan General de Garantía de Calidad considera procedimientos extraordinarios de control de calidad a las Auditorías (AUD).

El objetivo fundamental de una Auditoría consiste en garantizar que tanto los procesos de desarrollo que lleva a cabo el equipo correspondiente, como, muy especialmente, las funciones de control que está realizando el equipo de garantía de calidad (EGC), se ajustan estrictamente a las normas y procedimientos establecidos en el Plan de Garantía de Calidad específico del proyecto.

Una Auditoría tiene como objeto el proyecto informático completo, revisándose en el transcurso del procedimiento todos los documentos generados a lo largo del proceso de desarrollo y a lo largo de la realización de las actividades de garantía de calidad. Por esa razón, el equipo encargado de llevar a cabo la auditoría debe ser independiente tanto del equipo de desarrollo como del de garantía de calidad.

5.3.4. Procedimientos particulares.

El Plan General de Garantía de Calidad considera procedimientos particulares de control de calidad a la Evaluación de Prototipos (EVP).

Este procedimiento se aplica exclusivamente durante la fase de experimentación del modelo de desarrollo por evolución de prototipos. Su objetivo es proceder al refinamiento sucesivo en la definición de las especificaciones finales del proyecto, por medio de la evaluación que realizará fundamentalmente el usuario sobre los distintos prototipos (o versiones) que vaya realizando el equipo de desarrollo.

5.4. INSTRUMENTOS DE CONTROL Y ELEMENTOS AUXILIARES DE CONTROL DE CALIDAD.

El fascículo 4 del PGGC para el desarrollo de equipos lógicos, del Consejo Superior de Informática, establece los instrumentos de control y elementos auxiliares que, genéricamente, requiere cada uno de los procedimientos de control de calidad descritos anteriormente para poder llevarse a cabo. El Plan define dichos instrumentos de control y elementos auxiliares con carácter general para todos los proyectos que puedan ser objeto de las tareas de garantía de calidad, y distingue tres tipos diferentes:

1. Listas de Control (LC). Son instrumentos de ayuda que facilitan la realización de las tareas de control de calidad sobre los productos de tipo documental.

Las Listas de Control que considera el PGGC son:

- Listas de Control para las Revisiones Mínimas (LC.RM).
 - Listas de Control para las Revisiones Técnicas Formales (LC.RTF).
 - Listas de Control para las Inspecciones Detalladas (LC.ID).
 - Listas de Control para las Revisiones de Usuario (LC.RU).
2. Guiones de Recomendaciones (GR). Son instrumentos que aportan criterios generales, sugerencias y normas de buena práctica para la realización de determinadas tareas de garantía de calidad; por ejemplo, reuniones de revisión conjunta, pruebas y auditorías.
 3. Elementos Auxiliares. Tienen como objeto reflejar las actuaciones llevadas a cabo en la realización de cada tarea de garantía de calidad.

Los principales Elementos Auxiliares que considera el PGGC son:

- Las Hojas de Comentario, tanto de las revisiones como de usuario.
- Las Hojas de aprobación.
- Las Listas de Acciones Correctivas.
- Los Planes de Pruebas.
- Los Informes de Pruebas.
- Etcétera.

Todos los Elementos Auxiliares generados durante la aplicación de un Plan de Garantía de Calidad específico para un proyecto formarán parte, junto con los productos obtenidos en cada fase del desarrollo, del Dossier de Garantía de Calidad del Proyecto.

6. EL TRATAMIENTO DE LA CALIDAD DEL SOFTWARE EN LA METODOLOGÍA MÉTRICA V.3.

La metodología MÉTRICA Versión 3 contempla tanto el desarrollo de Sistemas de Información para las distintas tecnologías que actualmente conviven, como los aspectos de gestión que aseguran que un Proyecto cumple sus objetivos en términos de calidad, coste y plazos. En una única estructura la metodología cubre el desarrollo estructurado y el desarrollo orientado a objetos, facilitando a través de cuatro interfaces (Gestión de Proyectos, Gestión de Configuración, Aseguramiento de Calidad y Seguridad) la realización de los procesos de apoyo u organizativos.

En lo que se refiere a calidad, Métrica v.3 sigue los estándares propuestos por las normas ISO 9000-2000 «Sistemas de Gestión de la Calidad. Fundamentos y Vocabulario», ISO 9001-2000 «Sistemas de Gestión de la Calidad. Requisitos», y por la norma ISO/IEC TR 15.504/SPICE.

Siguiendo la norma ISO 9000-2000, Métrica v.3 define la calidad como el grado en que un conjunto de características inherentes cumple con unos requisitos, y prevé una interfaz de Aseguramiento de la Calidad, cuyo objetivo es proporcionar un marco común de referencia para la definición y puesta en marcha de planes específicos de aseguramiento de calidad aplicables a proyectos concretos.

6.1. LA INTERFAZ DE ASEGURAMIENTO DE LA CALIDAD.

Las actividades de la interfaz de Calidad en Métrica v.3 están orientadas a verificar la calidad de los productos. Son actividades que evalúan la calidad y que son realizadas por un grupo de Asesoramiento de la Calidad independiente del equipo de desarrollo. Estas actividades no entran en contradicción con el Plan General de Garantía de Calidad (PGGC), siendo lo suficientemente abiertas como para soportar una nueva versión del PGGC en el futuro.

Las actividades contempladas en la interfaz de Aseguramiento de la Calidad permiten:

- Reducir, eliminar y prevenir las deficiencias de calidad de los productos a obtener.
- Alcanzar una razonable confianza en que las prestaciones y servicios esperados por el cliente o el usuario queden satisfechas.

Para conseguir estos objetivos, es necesario desarrollar un plan de aseguramiento de calidad específico que se aplicará durante la planificación del proyecto de acuerdo a la estrategia de desarrollo adoptada en la gestión del proyecto. En el plan de aseguramiento de calidad se reflejan las actividades de calidad a realizar (normales o extraordinarias), los estándares a aplicar, los productos a revisar, los procedimientos a seguir en la obtención de los distintos productos durante el desarrollo en Métrica v.3 y la normativa para informar de los defectos detectados a sus responsables y realizar el seguimiento de los mismos hasta su corrección.

El grupo de aseguramiento de calidad participa en la revisión de los productos seleccionados para determinar si son conformes o no a los procedimientos, normas o criterios especificados. Las activida-

des a realizar por el grupo de aseguramiento de calidad vienen gobernadas por el plan. Sus funciones están dirigidas a identificar las posibles desviaciones en los estándares aplicados, así como en los requisitos y procedimientos especificados y a comprobar que se han llevado a cabo las medidas preventivas o correctoras necesarias.

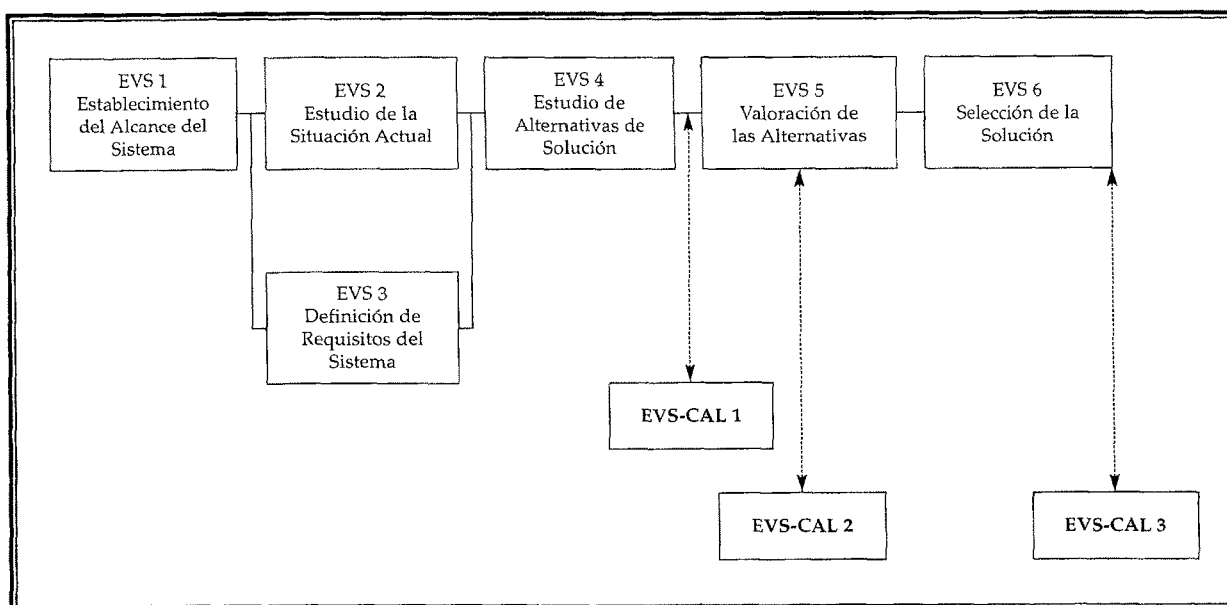
Las revisiones son una de las actividades más importantes del aseguramiento de la calidad. Además existen procedimientos extraordinarios, como las auditorías, aplicables en desarrollos singulares y en el transcurso de las cuales se revisarán tanto las actividades de desarrollo como las propias de aseguramiento de calidad. La detección anticipada de errores evita el que se propaguen a los restantes procesos de desarrollo, reduciendo substancialmente el esfuerzo invertido en los mismos. En este sentido es importante destacar que el establecimiento del plan de aseguramiento de calidad comienza en el Estudio de Viabilidad del Sistema y se aplica a lo largo de todo el desarrollo, en los procesos de Análisis, Diseño, Construcción, Implantación y Aceptación del Sistema y en su posterior Mantenimiento.

A continuación se muestra la correspondencia entre las actividades de los diferentes procesos de Métrica v.3 y las de la interfaz de aseguramiento de la calidad.

• Estudio de viabilidad del sistema.

En este proceso el grupo de aseguramiento de calidad inicia el estudio de los sistemas de información definidos en cada alternativa de solución propuesta, con el fin de identificar las condiciones en que se van a desarrollar y/o a implantar, así como las características que deben reunir en cuanto a operación, mantenibilidad y portabilidad, para satisfacer las necesidades del cliente y los requisitos especificados.

Una vez tomada la decisión de llevar a cabo un plan de aseguramiento de calidad en las alternativas propuestas, se define el contenido de dicho plan de acuerdo a los estándares de calidad. El plan debe cubrir todas las necesidades establecidas de modo que, aquellas normas impuestas por los usuarios o clientes que difieran de las existentes en el sistema de calidad, deben quedar también reflejadas en el plan.



Las actividades y tareas de aseguramiento de la calidad contempladas en el proceso EVS son las siguientes:

EVS-CAL 1: Identificación de las Propiedades de Calidad para el sistema.

- 1.1. Constitución del Equipo de Aseguramiento de la Calidad.
- 1.2. Determinación de los sistemas objeto de Aseguramiento de la Calidad.
- 1.3. Identificación de las Propiedades de Calidad.

EVS-CAL 2: Establecimiento del Plan de Aseguramiento de la Calidad.

- 2.1. Necesidad del Plan para las alternativas propuestas.
- 2.2. Alcance del Plan de Aseguramiento de la Calidad.
- 2.3. Impacto en el coste del Sistema.

EVS-CAL 3: Adecuación del Plan de Aseguramiento de la Calidad a la solución.

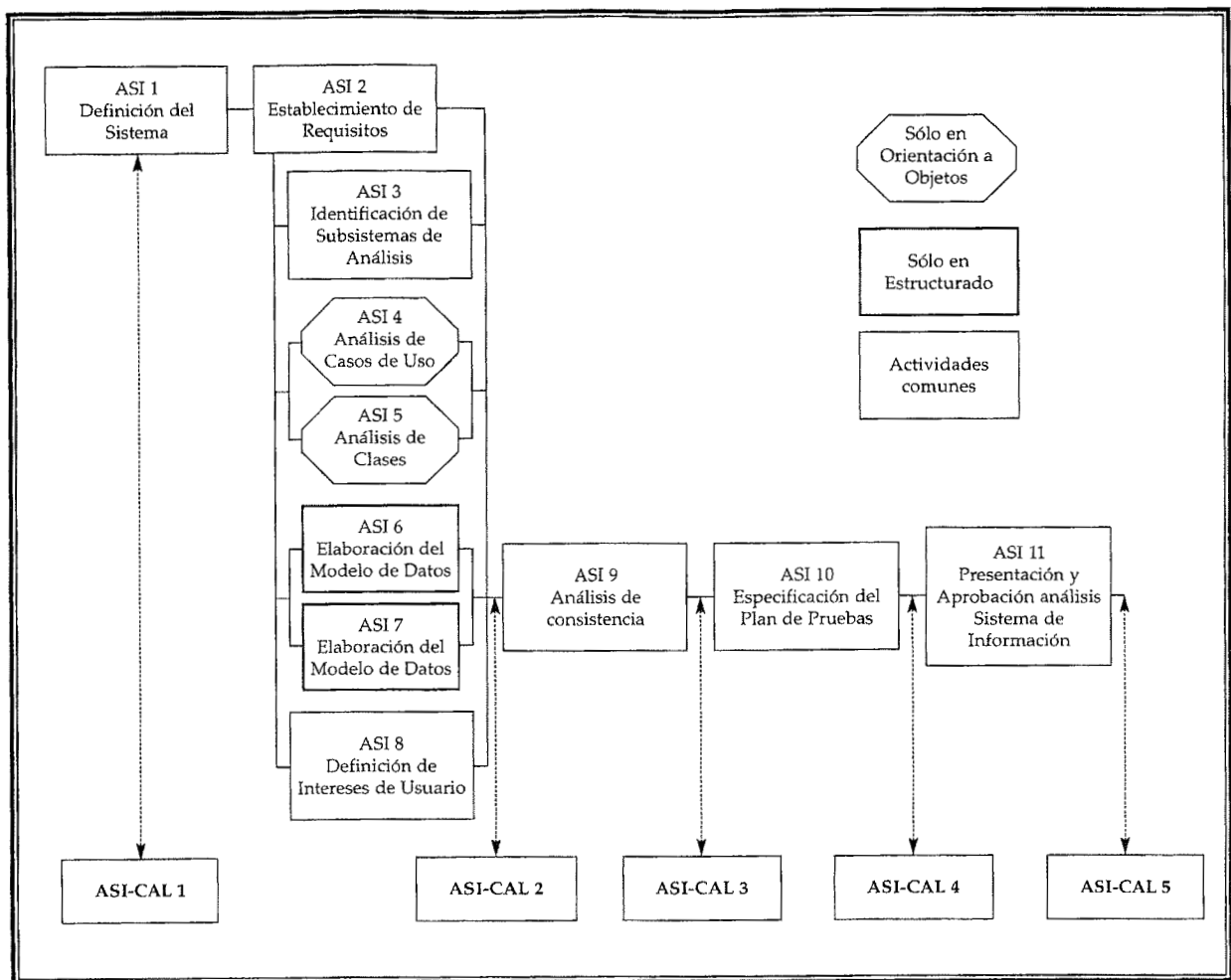
- 3.1. Ajuste del Plan de Aseguramiento de la Calidad.
- 3.2. Aprobación del Plan de Aseguramiento de la Calidad.

• **Análisis del sistema de información.**

En este proceso se define de forma detallada el plan de aseguramiento de calidad para un sistema de información, a partir de la especificación resultante del Estudio de Viabilidad del Sistema, y se detallan los estándares y normas a cumplir, las revisiones a llevar a cabo y sobre qué productos, así como los procedimientos y mecanismos necesarios para resolver los problemas que surjan, definiendo las acciones preventivas o correctoras e identificando quiénes son los responsables en cada caso.

En el Análisis del Sistema de Información el grupo de aseguramiento de calidad se implica directamente en la revisión de los siguientes productos:

- Catálogo de requisitos.
- Modelos resultantes del análisis.
- Plan de pruebas.



Las actividades y tareas de aseguramiento de la calidad contempladas en el proceso ASI son las siguientes:

ASI-CAL 1: Especificación inicial del Plan de Aseguramiento de Calidad.

1.1. Definición del Plan de Aseguramiento de Calidad para el Sistema de Información.

ASI-CAL 2: Especificación detallada del Plan de Aseguramiento de Calidad.

2.1. Contenido del Plan de Aseguramiento de Calidad para el Sistema de Información.

ASI-CAL 3: Revisión del Análisis de Consistencia.

3.1. Revisión del Catálogo de Requisitos.

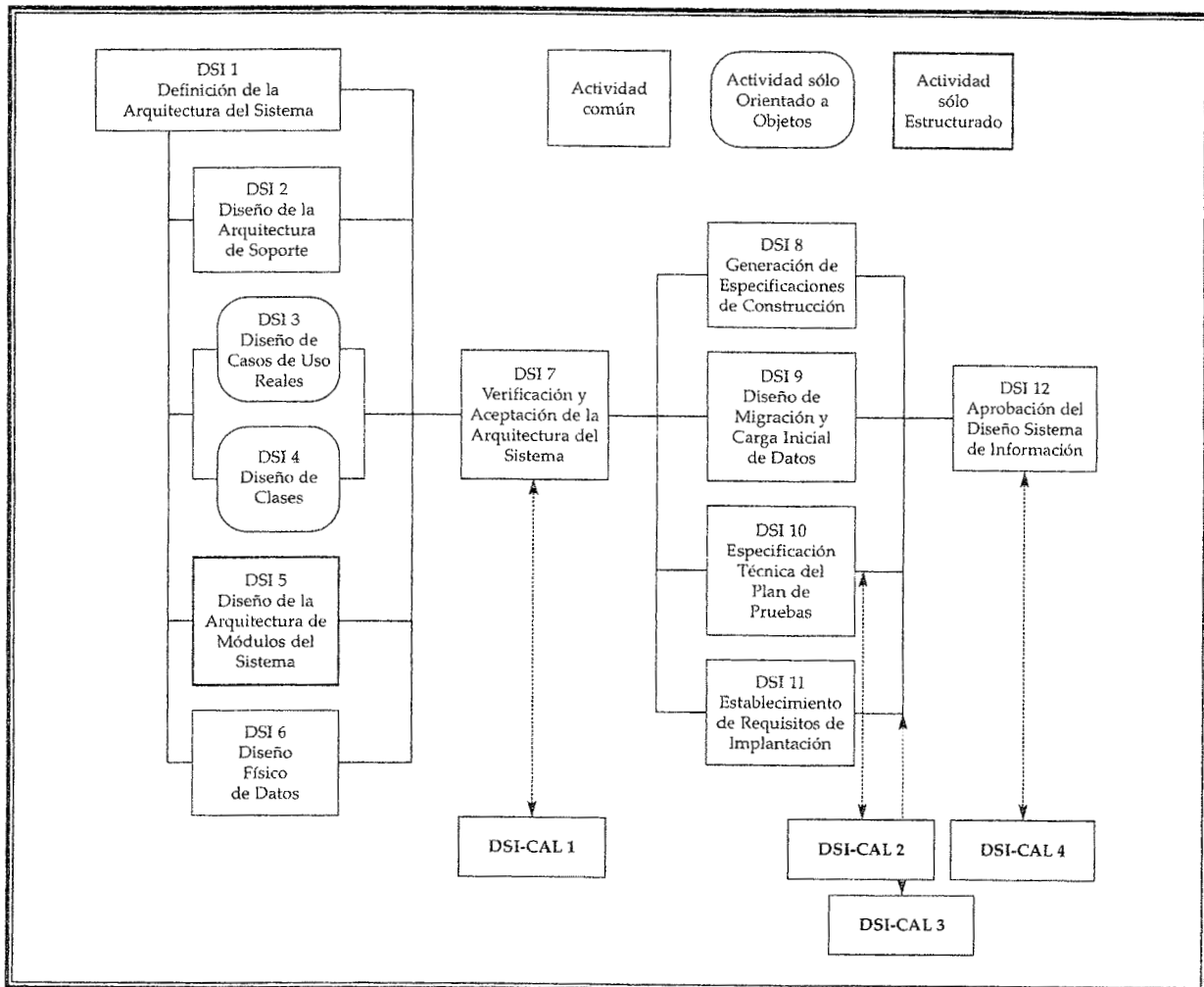
3.2. Revisión de la Consistencia entre productos.

ASI-CAL 4: Revisión del Plan de Pruebas.

ASI-CAL 5: Registro de la Aprobación del Análisis del Sistema de Información.

• Diseño del sistema de información.

Las revisiones del diseño se centran en confirmar que los requisitos especificados en el Análisis del Sistema de Información se han traducido en una arquitectura conforme al entorno tecnológico seleccionado. Asimismo, se revisan los requisitos que deben cumplir los distintos niveles de pruebas especificados en el plan de pruebas, y también se realiza una revisión de la identificación de los requisitos no funcionales relacionados con la documentación de usuario e implantación.



Las actividades y tareas de aseguramiento de la calidad contempladas en el proceso DSI son las siguientes:

DSI-CAL 1: Revisión de la verificación de la Arquitectura del Sistema.

1.1. Revisión de la Consistencia entre productos del Diseño.

1.2. Registro de la Aceptación de la Arquitectura del Sistema.

DSI-CAL 2: Revisión de la especificación técnica del Plan de Pruebas.

2.1. Revisión del diseño de las pruebas unitarias, de integración y del sistema.

2.2. Revisión del Plan de Pruebas.

DSI-CAL 3: Revisión de los Requisitos de Implantación.

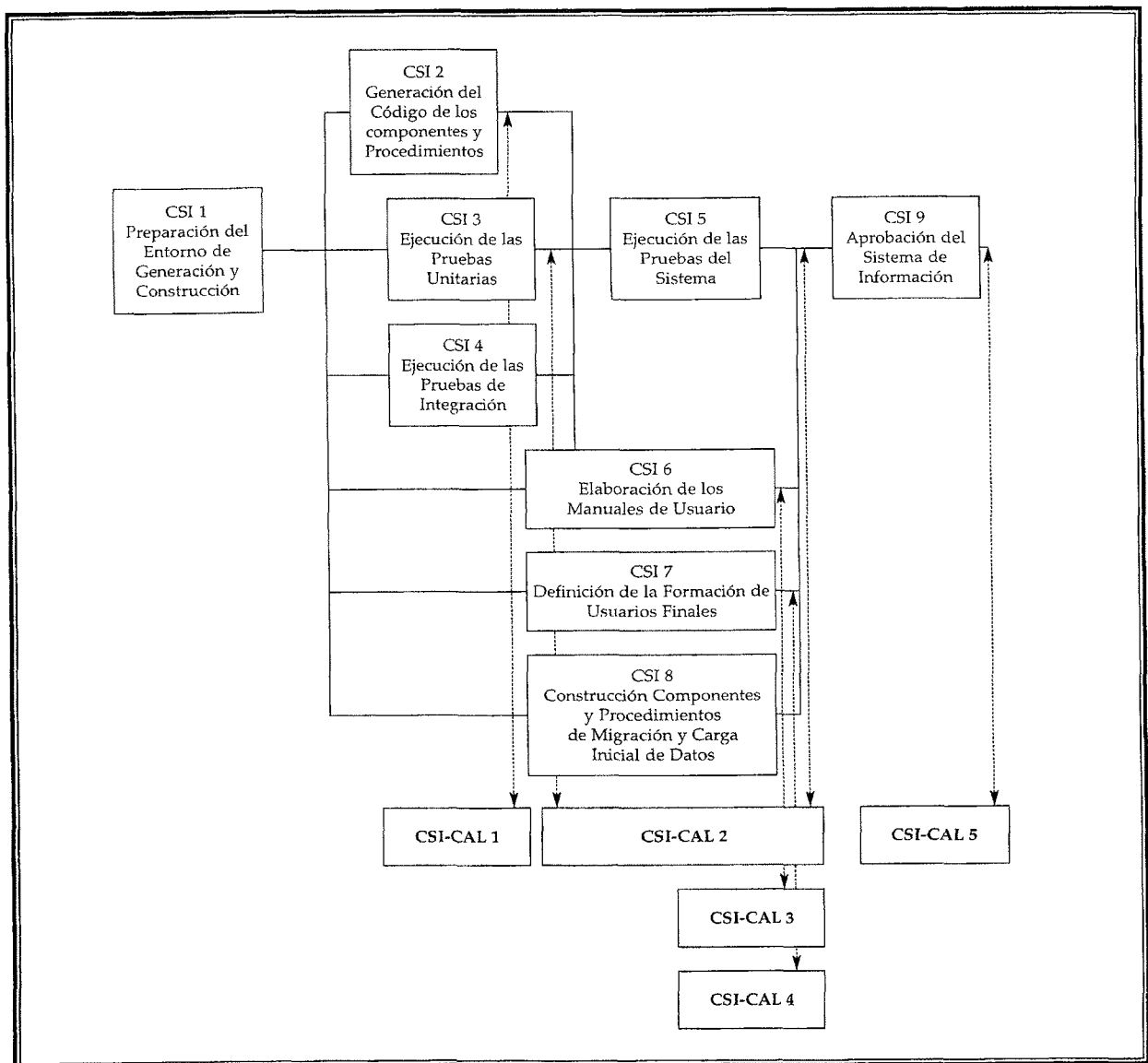
3.1. Revisión de los Requisitos de Documentación de Usuario.

3.2. Revisión de los Requisitos de Implantación.

DSI-CAL 4: Registro de la Aprobación del Diseño del Sistema de Información.

• Construcción del sistema de información.

En este proceso el grupo de aseguramiento de calidad revisa los estándares de nomenclatura y normativa aplicada en la generación del código de componentes, en la evaluación de los resultados de las pruebas, en los manuales de usuario y en el esquema de formación. Asimismo, respecto a las pruebas revisa que se han llevado a cabo según los criterios fijados en el plan de aseguramiento de calidad.



Las actividades y tareas de aseguramiento de la calidad contempladas en el proceso CSI son las siguientes:

CSI-CAL 1: Revisión del Código de Componentes y Procedimientos.

1.1. Revisión de estándares.

CSI-CAL 2: Revisión de las Pruebas Unitarias, de Integración y del Sistema.

2.1. Revisión de la realización de las Pruebas Unitarias.

2.2. Revisión de la realización de las Pruebas de Integración.

2.3. Revisión de la realización de las Pruebas del Sistema.

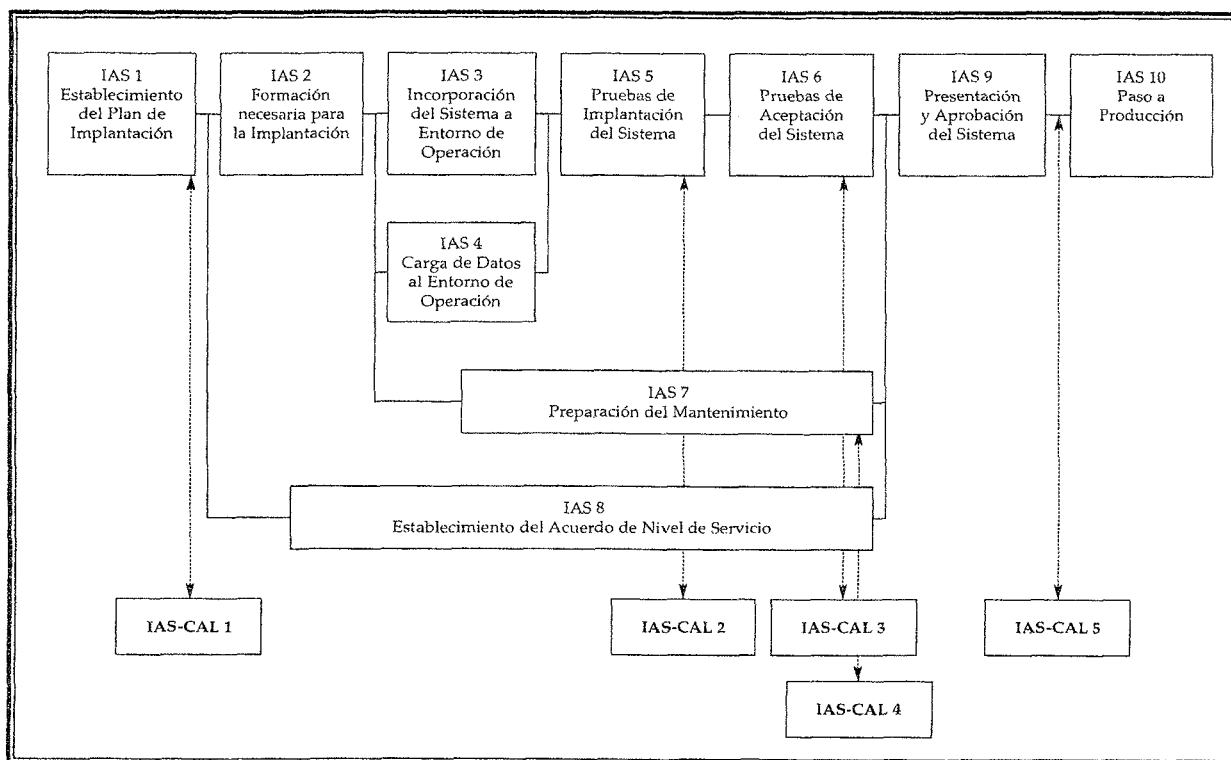
CSI-CAL 3: Revisión de los Manuales de Usuario.

CSI-CAL 4: Revisión de la Formación a Usuarios Finales.

CSI-CAL 5: Registro de la Aprobación del Sistema de Información.

• **Implantación y aceptación del sistema.**

En este proceso el grupo de aseguramiento de calidad en este proceso es responsable de revisar la existencia de un plan de implantación del sistema y de comprobar que se han realizado las pruebas de implantación y de aceptación según el plan de pruebas establecido en Métrica v.3 y en la normativa acordada en el plan de aseguramiento de calidad. Asimismo, el grupo de aseguramiento de calidad debe asegurar que se le entrega el producto software al responsable de mantenimiento para que pueda asumir el servicio, una vez que el sistema se encuentre en producción.



Las actividades y tareas de aseguramiento de la calidad contempladas en el proceso IAS son las siguientes:

IAS-CAL 1: Revisión del Plan de Implantación del Sistema.

IAS-CAL 2: Revisión de las Pruebas de Implantación del Sistema.

2.1. Revisión de la realización de las Pruebas de Implantación.

2.2. Registro de la Aprobación de las Pruebas de Implantación.

IAS-CAL 3: Revisión de las Pruebas de Aceptación del Sistema.

3.1. Revisión de la realización de las Pruebas de Aceptación.

3.2. Registro de la Aprobación de las Pruebas de Aceptación.

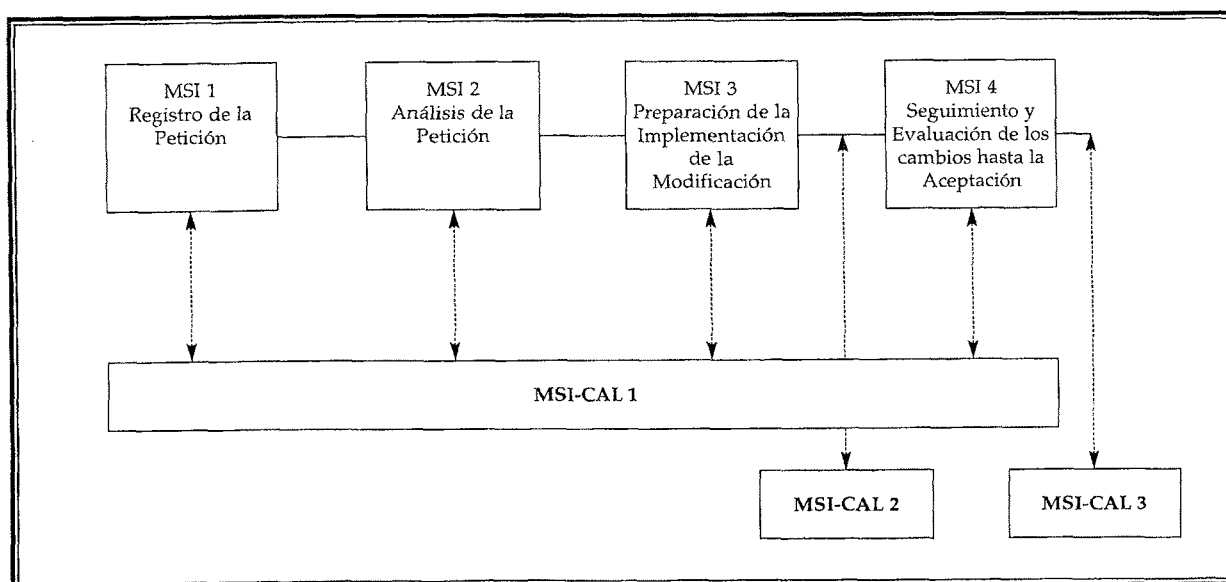
IAS-CAL 4: Revisión del Plan de Mantenimiento del Sistema.

IAS-CAL 5: Registro de la Aprobación de la Implantación del Sistema.

• Mantenimiento del sistema de información.

El grupo de aseguramiento de calidad intervendrá durante el mantenimiento, efectuando revisiones de seguimiento periódicas, que sirvan para constatar que el mantenimiento establecido para el sistema de información se realiza de forma correcta. Según las implicaciones del cambio podrá ser necesario revisar:

- El contenido del plan de pruebas de regresión.
- La ejecución de las pruebas de regresión.
- Las verificaciones y casos de prueba incluidos para los cambios.
- Las incidencias detectadas.



Las actividades y tareas de aseguramiento de la calidad contempladas en el proceso MSI son las siguientes:

MSI-CAL 1: Revisión del Mantenimiento del Sistema de Información.

MSI-CAL 2: Revisión del Plan de Pruebas de Regresión.

MSI-CAL 3: Revisión de la realización de las Pruebas de Regresión.

