

**MINISTERIO DE DEFENSA**

**SECRETARÍA DE ESTADO DE DEFENSA**

**DIRECCIÓN GENERAL DE ARMAMENTO Y MATERIAL**



---

# **PUBLICACIÓN ESPAÑOLA DE CALIDAD**

## **PECAL - 2210 (Edición A Versión 2 )**

---

**REQUISITOS OTAN DE ASEGURAMIENTO DE  
LA CALIDAD DEL SOFTWARE, SUPLEMENTA-  
RIOS A LA PECAL 2110 O A LA PECAL 2310**

---

**VERSIÓN ESPAÑOLA DE LA AQAP - 2210 (Edition A, Version 2, September 2015)**

---

**MINISTERIO DE DEFENSA**

SECRETARÍA DE ESTADO DE DEFENSA

DIRECCIÓN GENERAL DE ARMAMENTO Y MATERIAL



---

**PUBLICACIÓN  
ESPAÑOLA  
DE CALIDAD**

**PECAL - 2210 (EDICIÓN A VERSIÓN 2 )**

---

**REQUISITOS OTAN DE ASEGURAMIENTO DE LA  
CALIDAD DEL SOFTWARE, SUPLEMENTARIOS A LA  
PECAL 2110 O A LA PECAL 2310**

---

**VERSIÓN ESPAÑOLA DE LA AQAP - 2210 (Edition A, Version 2, September 2015)**

---

**CATÁLOGO GENERAL DE PUBLICACIONES OFICIALES**  
<http://publicacionesoficiales.boe.es/>

Edita:



<http://publicaciones.defensa.gob.es/>

© Autor y editor, 2017

NIPO: 083-17-031-5 (edición en línea)

NIPO: 083-17-032-0 (impresión bajo demanda)

NIPO: 083-17-033-6 (libro electrónico)

Fecha de edición: marzo 2016

Las opiniones emitidas en esta publicación son exclusiva responsabilidad del autor de la misma. Los derechos de explotación de esta obra están amparados por la Ley de Propiedad Intelectual. Ninguna de las partes de la misma puede ser reproducida, almacenada ni transmitida en ninguna forma ni por medio alguno, electrónico, mecánico o de grabación, incluido fotocopias, o por cualquier otra forma, sin permiso previo, expreso y por escrito de los titulares del © Copyright.

PUBLICACIÓN ESPAÑOLA DE CALIDAD

MINISTERIO DE DEFENSA

SECRETARÍA DE ESTADO DE DEFENSA

DIRECCIÓN GENERAL DE ARMAMENTO Y MATERIAL

**REQUISITOS OTAN DE ASEGURAMIENTO DE LA CALIDAD  
DEL SOFTWARE, SUPLEMENTARIOS A LA PECAL 2110  
O A LA PECAL 2310**

**PECAL - 2210**

**Edición A  
Versión 2**

Enero, 2017

Las observaciones relativas a esta publicación deben ser dirigidas a:  
DIRECCIÓN GENERAL DE ARMAMENTO Y MATERIAL  
Subdirección General de Inspección, Regulación  
y Estrategia Industrial de Defensa.  
Área de Inspecciones Industriales  
c/ Donoso Cortés, 92 - 28015 MADRID

Versión española de la

**AQAP-2210**

Edition A

Version 2

September, 2015

Página en blanco intencionadamente

## **NOTAS PRELIMINARES**

1. El documento titulado «Requisitos OTAN de Aseguramiento de la Calidad del *software*, suplementarios a la PECAL 2110 o a la PECAL 2310», título abreviado PECAL 2210 Edición A versión 2, es una publicación no clasificada de la serie «Publicaciones Españolas de Calidad», que se corresponden con las publicaciones análogas de la OTAN (*Allied Quality Assurance Publications, AQAP*). El acuerdo de las naciones interesadas para utilizar las *AQAP* está registrado en el STANAG 4107.
2. La PECAL 2210 Edición A, versión 2 entra en vigor a partir de la fecha de su publicación y sustituye a la PECAL 2210 Edición 1, que debe ser destruida de acuerdo con los procedimientos establecidos.
3. Ninguna parte de esta publicación podrá reproducirse, almacenarse, utilizarse con fines comerciales, adaptarse o transmitirse en modo alguno por medios electrónicos, mecánicos, fotocopias, grabación o de cualquier otra forma sin el previo consentimiento del propietario de la publicación. Con la única excepción de los fines comerciales, lo anteriormente indicado no es aplicable a los mandos y organismos de las naciones de la OTAN y de países asociados.
4. Esta publicación debe ser manejada de acuerdo con lo dispuesto en el documento C-M (2002)60.

Página en blanco intencionadamente





Página en blanco intencionadamente



Página en blanco intencionadamente

## PRÓLOGO

Los requisitos de aseguramiento de calidad del comprador establecidos en esta publicación están basados en la experiencia de que la gestión de la calidad en todo el proceso de desarrollo del *software* es la clave para lograr la calidad del *software* en sistemas informáticos complejos y de misión crítica, tales como los sistemas de armas, sistemas de comunicación y sistemas de mando y control. Para asegurar la calidad del proceso de desarrollo del *software* debe planificarse tal proceso, así como controlarse y mejorarse, con el objetivo de reducir, eliminar y, lo más importante, prevenir las deficiencias de calidad del *software*.

De acuerdo con la normalización internacional para la gestión de la calidad del *software*, se usan las definiciones funcionales más que las organizativas para evitar los problemas introducidos por los conceptos de calidad tradicionales y sus límites organizativos. Por consiguiente, esta publicación no se dirige específicamente a las organizaciones de la calidad del *software*, sino al conjunto de la estructura organizativa y a los diferentes niveles de dirección involucrados en un proyecto *software*.

Esta publicación está concebida para usarse en contratos y define los requisitos para las Actividades de Gestión de la Calidad del *Software* relacionadas con el proyecto a documentar en un Plan de Calidad del Proyecto *Software* (*SPQP*). Estas actividades están basadas en el Sistema de Calidad del *Software* del suministrador. La presente publicación también requiere la evaluación de las Actividades de Gestión de la Calidad del *Software* para asegurar su eficacia.

La aplicación de esta publicación no está restringida a ninguna forma o tipo particular de *software*. Esta publicación no especifica ningún modelo particular de desarrollo del *software*, ni estipula qué métodos de desarrollo del *software* deben usarse. Esta publicación permite flexibilidad para adaptar la documentación y los procedimientos requeridos a los procesos específicos de adquisición y desarrollo del proyecto.

Esta publicación sustituye a la PECAL 2210 Edición 1 y está concebida para su uso junto con la PECAL 2110 o la PECAL 2310, como suplemento específico del *software* orientado al proyecto.

Página en blanco intencionadamente

# 1 INTRODUCCIÓN

## 1.1 Propósito

Esta publicación especifica los requisitos del proyecto orientados a la gestión de la calidad del proceso de desarrollo del *software*. Tanto los procesos técnicos como los de gestión deben ser tenidos en cuenta con el fin de:

- a) Establecer visibilidad en el proceso de desarrollo del *software*.
- b) Detectar problemas de la calidad del *software*, tan pronto como sea posible, en el ciclo de vida del *software*.
- c) Suministrar datos de control de la calidad para la implantación a tiempo de una acción correctiva eficaz.
- d) Confirmar que la calidad se integra en el proceso de desarrollo del *software*.
- e) Proporcionar confianza de que el *software* producido es conforme a los requisitos contractuales.
- f) Asegurar que se proporciona el soporte apropiado de *software* a las actividades que sean requeridas por el contrato a nivel de ingeniería de sistema.
- g) Asegurar que las condiciones de seguridad de intrusión (*security*) y seguridad de personas/producto (*safety*) del proyecto son tenidas en cuenta.

## 1.2 Aplicabilidad

1. Cuando esta publicación se cite en un contrato o pedido, se aplicará a:
  - a) Todos los casos donde se realice desarrollo *software*.
  - b) Todos los casos donde el *software* no entregable sea desarrollado o utilizado bajo el contrato (según se especifica en el párrafo 2.2.4.8).
  - c) Todos los casos donde el mantenimiento del *software* forme parte del contrato, para evitar actividades de desarrollo incontroladas u ocultas que podrían tener consecuencias imprevistas y perjudiciales sobre la calidad del producto *software*.
  - d) Todos los casos donde el *software* preexistente vaya a ser entregado (según se especifica en el párrafo 2.2.4.7).
  - e) Todos los casos en los que el desarrollo del elemento *software* forme parte del *firmware*.
2. En caso de que el contrato contemple solo «parcialmente» actividades de desarrollo o de mantenimiento del *software* deberán, también, aplicarse los requisitos relacionados con esta publicación (por ejemplo, actividades de replicación de *software*, actividades de *software* durante la integración del sistema, definición de requisitos del *software*, servicios de archivo y almacenamiento del *software*, actividades de gestión de subministradores, etcétera).
3. Esta publicación está concebida para su uso junto con la PECAL 2110 o la PECAL 2310 como un suplemento específico del *software*, orientado al proyecto. Si existiese cualquier conflicto entre los requisitos de la PECAL 2110 (o de la PECAL 2310) y esta publicación para el *software* prevalecerán los requisitos de esta publicación.

4. Si existiese alguna inconsistencia entre los requisitos del contrato y esta publicación prevalecerán los requisitos del contrato.
5. Esta publicación también puede utilizarse para la especificación de los requisitos de las peticiones de oferta y para la evaluación de ofertas en la adquisición del *software* mediante concurso. Los requisitos de esta publicación también pueden aplicarse a organismos oficiales que realicen desarrollo o mantenimiento del *software*.

### 1.3 Documentación de referencia

1. PECAL 2110, «Requisitos OTAN de Aseguramiento de la Calidad para el Diseño, el Desarrollo y la Producción».
2. PECAL 2310, «Requisitos OTAN para los Sistemas de Gestión de Calidad de suministradores de aviación, espaciales y de Defensa».
3. ISO 9000, «Sistemas de Gestión de la Calidad- Fundamentos y Vocabulario».
4. ISO/IEC 25010: 2011, «Systems and software engineering-Systems and software Quality Requirements and Evaluation (SQuaRE)-System and software quality models».

### 1.4 Definiciones y acrónimos

#### 1.4.1 Definiciones

Las definiciones aplicables son las de la ISO 9000 o la PECAL 2110 (o la PECAL 2310). Donde las definiciones de la ISO 9000 o la PECAL 2110 (o la PECAL 2310) y esta publicación difieran se aplicarán las definiciones de esta publicación.

#### 1. Control

Actividad para detectar las diferencias entre un proceso/resultado planificado y el real, y para producir los cambios en el proceso o producto con objeto de reducir las diferencias detectadas hasta un nivel definido.

#### 2. Evaluación

Determinación sistemática del grado en el que una entidad cumple con los criterios especificados para ella.

#### Notas:

- a) El término «entidad» incluye al producto actividad, proceso, organización o persona.
- b) *La evaluación de la actividad o del proceso puede realizarse en paralelo con el desarrollo o puede deducirse como el resultado de la verificación del producto del software.*
- c) *La evaluación de la actividad o del proceso puede realizarse supervisando, auditando, calificando el proceso o estableciendo y documentando si es conforme o no al criterio especificado.*

### 3. Firmware

Combinación de un dispositivo *hardware* y de las instrucciones o datos de ordenador que residen como *software* de solo lectura (*read-only software*) en estos dispositivos *hardware*.

### 4. Método

Conjunto de reglas para resolver un problema.

### 5. *Software* no entregable

*Software* que no es necesario entregar según contrato, pero que puede usarse en el desarrollo del *software*.

### 6. *Software* preexistente (*off-the-shelf*)

*Software* entregable que ya está desarrollado y que puede usarse tal como es o con modificaciones. Según la fuente, puede ser: *software* reutilizable, *software* que entrega el cliente o *software* comercial.

### 7. Proceso

Interacción del personal, equipos, material y procedimientos con el propósito de proporcionar un servicio o de producir un producto especificado.

Cada proceso se define como un conjunto de una o más actividades o tareas que pueden realizarse en un período limitado de tiempo. Cada proceso puede descomponerse en actividades, las cuales están caracterizadas por entradas y salidas cuantificables que pueden medirse, controlarse y mejorarse.

### 8. Modelo de desarrollo del *software*

Representación simplificada y abstracta del proceso de desarrollo del *software* (funcionamiento del proceso y resultados) usado con el propósito de planificar y controlar.

### 9. Proceso de desarrollo del *software*

Proceso por el que las necesidades o requisitos del usuario se transforman en un producto *software*.

### 10. Ciclo de vida del *software*

Marco de referencia (*framework*) que contiene los procesos, actividades y tareas involucradas en el desarrollo, operación y mantenimiento de un producto *software*, abarcando toda la vida del sistema desde la definición de sus requisitos hasta el fin de su uso.

### 11. Características de la calidad del *software*

Conjunto de atributos de un producto *software* mediante los cuales se describe, verifica y valida su calidad. Una característica de la calidad *software* puede descomponerse a su vez en múltiples niveles de subcaracterísticas.



**Nota:** De acuerdo con la Norma ISO/IEC 25010: 2011, la calidad del software puede evaluarse usando las ocho características siguientes: Funcionalidad, fiabilidad, facilidad de uso (usabilidad), eficiencia, mantenibilidad, portabilidad, compatibilidad y seguridad.

#### 12. Software/producto *software*

Programas de ordenador, procedimientos, reglas, así como documentación y datos asociados referentes al funcionamiento de un sistema informático.

#### 13. Herramienta *software*

Programa de ordenador utilizado como ayuda al desarrollo, análisis, evaluación, verificación, validación o mantenimiento de otro programa de ordenador o de su documentación.

#### 14. Validación

Confirmación por medio de examen y disposición de evidencia objetiva de que se cumplen los requisitos particulares para un uso previsto específico.

#### **Notas:**

- a) La validación se realiza normalmente en el producto final bajo las condiciones de operación definidas.
- b) Pueden llevarse a cabo múltiples validaciones si hay diferentes usos previstos.

#### 15. Verificación

Proceso para determinar y obtener la evidencia objetiva de si los productos de una determinada fase del proceso de desarrollo del *software* cumplen con los requisitos establecidos durante las fases anteriores.

#### **Notas:**

- a) *La verificación puede realizarse revisando, inspeccionando, probando, verificando, auditando o estableciendo y documentando si los productos son conformes a los requisitos especificados.*
- b) *Una fase en este contexto no implica un período de tiempo en el desarrollo de un producto software.*

### **1.4.2 Acrónimos**

Los siguientes acrónimos aparecen en esta publicación:

EC:	Elemento de configuración.
ECS:	Elemento de configuración <i>software</i> .
EVV:	Evaluación, verificación y validación.
GCS:	Gestión de configuración <i>software</i>
PCPS:	Plan de la calidad del proyecto <i>software</i> .
SCS:	Sistema de la calidad del <i>software</i> .

## 2 REQUISITOS

### 2.1 Sistema de la calidad del *software* (SCS)

1. El suministrador debe aplicar, en el proyecto objeto del contrato, un SCS documentado, eficaz y eficiente. El SCS puede ser una parte integrante de un sistema general de la calidad, pero deberá contener un proceso de gestión de la calidad exhaustivo e integral que será aplicado a lo largo de todo el contrato, asegurando que la calidad se va incorporando al diseño conforme progresa el desarrollo.
2. A través de la correlación de las desviaciones en plazo y en coste con la información de la calidad, el SCS deberá también prever medidas para la detección y corrección a tiempo de cualquier influencia negativa en la calidad y, por tanto, minimizar los riesgos técnicos.
3. Deberán efectuarse disposiciones para las revisiones sistemáticas y periódicas del SCS por la alta dirección del suministrador, o en su nombre, para asegurar su eficacia.

### 2.2 Actividades de gestión de la calidad del proyecto *software*

#### 2.2.1 Generalidades

1. Para lograr la visibilidad y control del proyecto de desarrollo del *software*, el suministrador debe planificar e implantar actividades eficaces de gestión de calidad del *software*.
2. El suministrador debe realizar una revisión formal del contrato para asegurar que se definen todos los requisitos contractuales y para determinar los procesos técnicos y de gestión que necesitan ser planificados y llevados a cabo.
3. Basándose en los requisitos del contrato, los procedimientos e instrucciones del SCS, y en los requisitos específicos del proyecto, las actividades de gestión de la calidad del *software* deben:
  - a) Establecer/identificar, precisar y asignar los requisitos a los productos *software* y elementos de configuración (EC) (véase el apartado 2.2.3).
  - b) Establecer e implantar los procesos técnicos y de gestión necesarios para desarrollar la calidad e incorporarla al *software* (véanse los apartados 2.2.4/2.2.5).
  - c) Establecer e implantar procedimientos para verificar y validar la calidad de los productos *software* y para evaluar los procesos y actividades, incluyendo el *software* no entregable, que impacten en la calidad de los productos *software* (véase el apartado 2.2.6).
  - d) Establecer e implantar los procedimientos necesarios para la gestión de riesgos. El suministrador debe identificar, analizar, priorizar y supervisar las áreas del proyecto que supongan riesgos potenciales técnicos, de costes o de programa. El objetivo de la gestión de riesgos será eliminar o minimizar los mismos.
4. Las actividades de gestión de la calidad del *software* se deben realizar sobre nor-

mas existentes y procedimientos en el SCS de la organización. Cuando este no sea el caso el suministrador debe proporcionar justificación al comprador.

5. Las actividades de gestión de la calidad del *software* se deben documentar en el Plan de Calidad del Proyecto *Software* (PCPS) (véase el apartado 2.2.2).
6. Se deben tomar medidas para posibilitar al comprador la evaluación de las actividades de gestión de la calidad del *software*, quien podrá rechazar su aprobación.

### **2.2.2 Plan de Calidad del Proyecto *Software* (PCPS)**

1. El suministrador debe documentar las actividades de gestión de la calidad del *software* relacionadas con el proyecto en un PCPS. El PCPS puede ser un documento único o parte de otro plan realizado para el contrato. El PCPS debe ser aprobado por las personas de la organización que tienen responsabilidades identificadas en el PCPS y debe estar sujeto a la gestión de la configuración.
2. Si está estipulado en el contrato, el PCPS se debe presentar al comprador para su aprobación. Una vez aprobado por el comprador, el PCPS formará parte del contrato. Cualquier modificación posterior al plan aprobado se debe someter a los procedimientos de gestión de cambios definidos de acuerdo con el comprador y detallados en el PCPS.
3. El PCPS debe contemplar todos los requisitos y debe incluir o referenciar todos los procedimientos necesarios para el cumplimiento de todos los requisitos de esta norma. Si no se requiere específicamente, la información puede presentarse en el Plan en cualquier secuencia y formato.
4. El suministrador debe usar el PCPS como una línea de referencia actual para definir las actividades para supervisar y controlar la calidad del proyecto del *software*. El PCPS debe ser revisado y actualizado según los hitos predefinidos durante el proyecto, así como cuando empiecen a conocerse nuevas definiciones y detalles del desarrollo.

### **2.2.3 Identificación y revisión de los requisitos del *software***

1. El suministrador debe identificar los requisitos del *software* y restricciones del desarrollo.
2. El paso inicial en el proceso de desarrollo del *software* será una revisión de requisitos del *software* y se detallará en el PCPS, si esta no se ha realizado como un hito de desarrollo del sistema.
3. La revisión debe verificar que los requisitos del *software* son completos, consistentes, inequívocos, trazables, viables y pueden validarse.
4. Después de realizar con éxito una revisión de los requisitos del *software*, las especificaciones de requisitos *software* deben ser formalmente aprobadas por las autoridades responsables y están sujetas a la gestión de la configuración.
5. Si las especificaciones de requisitos del *software* son desarrolladas por el suministrador como parte del contrato de un sistema, los requisitos del *software* se deben presentar al comprador, quien podrá desaprobarnos de acuerdo a las condiciones del contrato.

6. Las especificaciones de requisitos del *software* deben incluir una definición clara y precisa de las restricciones del diseño y de las características esenciales de calidad del *software*.
7. El PCPS debe identificar qué normas o guías se aplican para el formato y contenido de las especificaciones de requisitos del *software*.
8. Cualquier incertidumbre en la interpretación de los requisitos contractuales del *software* debe comunicarse al comprador tan pronto como sea posible.

## **2.2.4 Gestión**

### **2.2.4.1 Proceso de desarrollo del *software***

1. El suministrador debe aplicar un modelo de desarrollo que descomponga el proceso de desarrollo en procesos parciales y que satisfaga los siguientes criterios, en relación con la calidad:
  - a) Reducir la complejidad del proceso de desarrollo para asegurar visibilidad y control.
  - b) Describir la integración del *software* y del sistema.
  - c) Describir la arquitectura del *software* del sistema.
  - d) Hacer uso de prácticas aceptadas de ingeniería del *software*.
  - e) Utilizar realimentación de datos (*feedback*) de diseños anteriores.
  - f) Describir claramente las actividades y los resultados esperados.
  - g) Identificar las tareas que son críticas para la calidad y el éxito del proyecto.
  - h) Definir y asignar cronológicamente puntos de control en los cuales pueda verificarse el correcto desarrollo del proceso y la correcta transferencia de sus resultados.
  - i) Describir cómo se controlarán las actividades no planificadas.
  - j) Ofrecer unos criterios inequívocos de comienzo y fin de todos los procesos.
  - k) Proporcionar una identificación y ubicación claras de todas las funciones de calidad dentro de las estructuras organizativas específicas del proyecto.
  - l) Utilizar medidas de la calidad, analíticas y constructivas, probadas y calificadas.
  - m) Proporcionar datos de calidad para la gestión eficaz del proceso de desarrollo.
  - n) Relacionar las actividades de planificación, seguimiento y puesta en servicio con las actividades de la ingeniería del *software*.
  - o) Reducir los riesgos utilizando recursos informáticos que liberen al personal que interviene en el proceso de desarrollo del *software* de actividades repetitivas propensas a error.
2. Cualquier cambio del modelo de desarrollo, adoptado durante el proyecto, debe registrarse en el plan del proyecto.

#### 2.2.4.2 Organización

1. El suministrador debe definir e implantar la estructura organizativa, responsabilidades, autoridades y la interrelación entre los elementos y grupos de la organización que planifiquen, dirijan, realicen y controlen actividades que afecten a la calidad del *software*.
2. El personal que realice evaluaciones, verificaciones y validaciones de la calidad del *software* debe disponer de recursos, responsabilidades, autoridad y experiencia técnica. También debe tener un apropiado nivel de independencia orgánica de la(s) persona(s) que desarrolle(n) el producto *software* o realicen la actividad objeto de la evaluación/verificación/validación para actuar con objetividad y promover el inicio de las acciones correctivas.
3. Se designará un representante con la autoridad necesaria para asegurar que todos los requisitos de esta publicación se cumplen.

#### 2.2.4.3 *Software* no conforme

El suministrador debe:

- a) Establecer y mantener el control de cualquier *software* que no sea conforme a los requisitos especificados, a fin de asegurar la imposibilidad de una utilización o entrega intempestiva.
- b) Notificar al comprador cualquier producto no-conforme recibido de subsuministradores que hayan estado sujetos al Aseguramiento Oficial de Calidad (véase el apartado 2.2.4.5).
- c) Proporcionar, de acuerdo con el comprador, medidas de control que permitan la identificación y la segregación del *software* no conforme.
- d) Documentar de manera exhaustiva la naturaleza de las no conformidades y las funciones afectadas.
- e) Documentar procedimientos para la disposición de los productos no-conformes.
- f) Notificar al comprador cada vez que se disponga la entrega de *software* no conforme.

#### 2.2.4.4 Acciones correctivas

1. El suministrador debe definir e implantar un proceso de acciones correctivas para asegurar que:
  - a) Todos los problemas detectados en procesos y productos son documentados y evaluados para comprobar su validez y analizados a fin de identificar tendencias.
  - b) Se informa de los problemas al nivel de dirección que tiene la autoridad necesaria para asegurar que se toma la acción correctiva oportuna.
  - c) Se toman acciones inmediatas y eficaces para resolver los problemas y corregir las tendencias adversas y se hace un seguimiento e informe de su estado.

- d) Se proporciona al comprador la realimentación, si se requiere en el contrato o en el PCPS.
  - e) Se proporcionan datos para medir y predecir la calidad del proceso de desarrollo del *software*.
  - f) Los registros se mantienen y ponen a disposición del comprador durante la vida del contrato o, en su caso, como en él se especifique.
2. El proceso de acciones correctivas debe tener en cuenta tanto los problemas técnicos como los problemas de gestión encontrados con objeto de prevenir su repetición.

#### **2.2.4.5 Gestión de subsuministradores**

1. Para el *software* subcontratado (entregable o no) desarrollado específicamente para el contrato el suministrador debe:
- a) Aplicar procedimientos eficaces de selección de subsuministradores.
  - b) Definir los requisitos de gestión de la calidad y del producto/servicio *software* incluyendo los requisitos para el PCPS del subsuministrador.
  - c) Llevar a cabo verificaciones/validaciones/evaluaciones de los artículos/procesos subcontratados, incluyendo el PCPS del subsuministrador.
  - d) Definir cómo han de ser tratados los cambios, en particular cuál será la participación del subsuministrador.
  - e) Definir las acciones que puede adoptar el suministrador en caso de que el subsuministrador no actúe conforme al contrato o al PCPS.
2. Cuando lo solicite el comprador, se deben tomar las disposiciones para el aseguramiento oficial de la calidad en las instalaciones del subsuministrador. Cuando el comprador considere que es necesario realizar por su parte una verificación/validación/evaluación de los artículos/procesos del subsuministrador, el suministrador debe incluirlo en los documentos de compra. A petición del comprador, se le suministrarán copias de los documentos de compra junto con los datos técnicos relevantes.

#### **2.2.4.6 Gestión de la Configuración del Software (GCS)**

1. El suministrador debe definir e implantar un proceso de GCS que permita mantener la integridad y la trazabilidad del (de los) producto(s) *software* durante el desarrollo. Las actividades y procedimientos de la GCS deben asegurar que no se produzcan cambios no controlados y deben proporcionar líneas de referencia planificadas y formalmente aceptadas como condición previa para la verificación, seguimiento y control de la calidad del *software*.

El suministrador, específicamente, debe definir e implantar:

- a) Procedimientos para identificar, nombrar y registrar las características físicas, funcionales y de la calidad de los elementos intermedios y finales que serán

controlados (por ejemplo, documentación, código ejecutable, código fuente, listado de programas, bases de datos, especificaciones, casos de prueba, planes) y sus estructuras en cada punto de control del proyecto. Los elementos del entorno de desarrollo y soporte (compiladores, herramientas de desarrollo, sistemas operativos, condiciones de pruebas) también deben formar parte de la estructura del elemento de configuración del *software* (ECS).

- b) Procedimientos para solicitar, evaluar, aprobar/rechazar e implantar los cambios (corrección de errores y mejora) de los ECS de referencia (la práctica de parchear el *software* debe restringirse a situaciones muy excepcionales y temporales. No se podrá hacer sin el conocimiento y acuerdo del comprador. El control de la configuración de parches debe estar establecido en un procedimiento específico).
  - c) Procedimientos para registrar e informar del estado de los ECS del proyecto.
  - d) Auditorías y revisiones para determinar hasta qué punto los ECS reflejan los requisitos físicos, funcionales y las características de calidad (véase el apartado 2.2.6), y para establecer una referencia.
  - e) Procedimientos para controlar las interfaces de los ECS del proyecto con los elementos que no forman parte del alcance directo del desarrollo del *software* (sistema, *hardware*, factor humano, *software* de apoyo).
  - f) Procedimientos para coordinar los cambios realizados en los elementos de *software* desarrollados externamente (véase el apartado 2.2.4.5) e incorporar estos al proyecto.
2. Los cambios en las especificaciones de requisitos del *software* deben evaluarse por el impacto en coste, técnico y en el plazo de entrega, y deben ser comunicados a todas las partes afectadas. Los cambios que afecten a la funcionalidad solo deben implementarse con la aprobación del comprador.
  3. El suministrador también debe identificar las herramientas del *software*, técnicas y equipos necesarios para implantar las actividades de la GCS (véase el apartado 2.2.5) y asignar las responsabilidades y autoridades para las actividades de la GCS a los organismos y personas que forman parte de la estructura del proyecto.

#### **2.2.4.7 Software preexistente (*off-the-shelf*)**

1. Si el suministrador utiliza *software* preexistente, debe asegurar que:
  - a) Su uso no está afectado por ningún derecho de protección de datos (propiedad intelectual).
  - b) Existe evidencia objetiva, antes de su uso, de que el *software* cumplirá las funciones requeridas.
  - c) El *software* está sujeto a gestión de la configuración.
  - d) El *software* está documentado de acuerdo con los requisitos del contrato y de esta publicación.

2. Si el *software* preexistente es entregable y se modifica durante el proceso de desarrollo debe, en ese caso, tratarse como *software* de desarrollo y está sujeto a los requisitos de esta publicación.
3. Si el suministrador considera que el *software* preexistente proporcionado por el comprador no es aceptable para el uso, debe informar al comprador lo más pronto posible de las razones de su rechazo y debe negociar con él las acciones a tomar para remediarlo.
4. El suministrador debe informar al comprador cuando se prevea incorporar *software* preexistente en un producto *software*.

#### **2.2.4.8 Software no entregable**

Si el suministrador emplea *software* no entregable para el desarrollo de *software* entregable, debe asegurar que:

- a) Existe evidencia objetiva, antes de su utilización, de que el *software* realizará las funciones requeridas.
- b) El *software* está sujeto a la gestión de la configuración.

#### **2.2.4.9 Registros de la calidad**

Todos los registros que demuestren que la calidad requerida se ha alcanzado deben estar a disposición del comprador.

Los registros de calidad deben:

- a) Proporcionar evidencia objetiva de que el proceso de desarrollo del *software* se ha llevado a cabo de acuerdo con los requisitos establecidos por el comprador y de acuerdo a prácticas reconocidas de ingeniería *software* detalladas en el PCPS.
- b) Proporcionar datos históricos o de referencia que puedan usarse para detectar tendencias a largo plazo y deficiencias de calidad en el proceso de desarrollo.
- c) Tener trazabilidad con los procedimientos que los rigen.

#### **2.2.4.10 Documentación**

1. El suministrador debe identificar la documentación que debe conservarse relativa al *software*, incluyendo los registros de calidad, junto con una recomendación del tiempo de conservación. El suministrador debe indicar los métodos e instalaciones usados para archivar, preservar y mantener esta documentación.
2. Las licencias del *software* aplicables deben cubrir el uso previsto del producto de *software*.

#### **2.2.4.11 Manipulación y almacenamiento de los soportes del software**

El suministrador debe asegurar que:



- a) El *software* está almacenado de forma que su recuperación esté garantizada.
- b) Tiene un sistema que solo permite acceso al *software* a través de un proceso de autorización, que hace el *software* accesible solo a aquellos que demuestran su necesidad de conocerlo o usarlo.
- c) El entorno está controlado de modo que los soportes físicos en los que está almacenado el *software* no se degraden.
- d) Se proporciona un lugar secundario y seguro para el almacenamiento y recuperación del *software* crítico y las copias de las versiones aprobadas del *software*.

#### **2.2.4.12 Replicación y entrega**

El suministrador deber asegurar que:

- a) El proceso de replicación para generar múltiples copias personalizadas del *software* está bajo control.
- b) El proceso de liberación (versionado) del *software*, incluyendo los métodos para generar múltiples copias personalizadas del *software*, está documentado, es reproducible y permanece bajo control.
- c) Se llevan a cabo los procedimientos para marcar, manipular, almacenar, conservar y empaquetar el *software*, de tal forma que su integridad esté segura hasta que se entregue al destino especificado en el contrato.
- d) Se llevan a cabo los procedimientos para la certificación de la conformidad del *software* con los requisitos del contrato.
- e) Se llevan a cabo los procedimientos para la conservación de los registros relacionados con la distribución de artículos entregables.

#### **2.2.5 Ingeniería del *software***

1. Para las actividades de desarrollo y/o mantenimiento del *software* el suministrador debe emplear métodos, herramientas, recursos y procedimientos reconocidos de ingeniería del *software*. El suministrador también debe identificar y normalizar los acuerdos formales específicos de cualquier notación lingüística o gráfica. Los métodos, herramientas, normas y procedimientos usados deben constituir un soporte del ciclo de vida del *software* para:
  - a) Indicar los requisitos del *software* incluyendo las características de calidad.
  - b) Transformar los requisitos de calidad del *software* orientados al comprador/usuario en características orientadas a ingeniería del *software* y asignarlas al nivel adecuado del diseño.
  - c) Asegurar la trazabilidad en todos los niveles del diseño y de realización.
  - d) Minimizar errores.
  - e) Apoyar a la evaluación/verificación/validación durante el desarrollo y/o el mantenimiento del *software*.

2. Los métodos y procedimientos utilizados deben ser evaluados y documentados y deben permitir la puesta en servicio de los principios y conceptos reconocidos de ingeniería del *software* que tengan influencia sobre la calidad del *software*. Las herramientas *software* deben ser validadas para confirmar su integridad y prestaciones por medio de un método definido.

### 2.2.6 Evaluación, verificación y validación (EVV)

1. El suministrador debe planificar, definir e implantar:
  - a) Un proceso para la evaluación de los métodos, técnicas, procedimientos, herramientas y actividades relacionadas con el *software*.
  - b) Un proceso para la verificación y validación de los elementos y productos *software*.
  - c) Un proceso que permita efectuar las acciones de seguimiento que aseguren que se realizan los cambios necesarios.
  - d) Un proceso para determinar el nivel requerido de reevaluación en caso de corrección de errores o cambios incorporados, tanto a los requisitos como al diseño.
2. El proceso de EVV debe definir:
  - e) Las actividades de EVV y su secuencia en relación con las fases, hitos y programación en el tiempo.
  - f) Los perfiles organizativos, responsabilidades y autoridad para la ejecución de las actividades de EVV (véase el apartado 2.2.4.2).
  - g) Los objetos de la EVV (por ejemplo, documentos de requisitos/desarrollos, productos *software*, procesos de desarrollo, métodos, procedimientos, código fuente, código ejecutable).
  - h) Los criterios para realizar la EVV.
  - i) Los métodos, normas, técnicas, herramientas e instalaciones específicos de EVV.
  - j) Los tipos de métodos de EVV a utilizar (por ejemplo, prueba, revisión, auditoría).
  - k) La documentación de EVV a generar (planes y procedimientos específicos, registros e informes de EVV).
3. Como una parte integrada en el proceso de EVV, el suministrador debe desarrollar/seleccionar e implantar medidas cuantitativas y/o cualitativas para evaluar/verificar/validar las características de calidad del *software* descritas en las especificaciones de requisitos.
4. También deben aplicarse medidas cuantitativas/cualitativas (métricas) para la gestión y control del proceso de desarrollo *software* para el producto *software* objeto del contrato. Tales medidas deben permitir la identificación del nivel real de prestaciones obtenido, la toma de acciones correctivas y el establecimiento de objetivos de mejora.

### 2.2.6.1 Pruebas

1. Como una parte integrada del proceso de EVV, el suministrador debe planificar, definir e implantar un programa de pruebas. Se deben tener en cuenta:
  - a) Las pruebas de los elementos de *software*, de integración, de sistema y de aceptación del *software*.
  - b) El entorno, las herramientas y el *software* de pruebas.
  - c) La documentación del usuario.
  - d) El personal requerido y la formación asociada.
2. El suministrador debe llevar a cabo una revisión de los requisitos y de los criterios de prueba para verificar la idoneidad, viabilidad, trazabilidad y la ausencia de ambigüedad. Las especificaciones de prueba deben incluir los escenarios de prueba, los datos requeridos para las pruebas y los resultados esperados.
3. El suministrador debe definir e implantar medidas para controlar actividades de las pruebas, que incluyan:
  - a) Cuando sea necesario, el establecimiento, documentación y verificación de la configuración del *software* a probar, junto con cualquier *hardware* asociado.
  - b) La actualización de la documentación relativa a las pruebas que permita su repetitividad.
  - c) La confirmación de que las pruebas se realizan de acuerdo con los planes, las especificaciones y los procedimientos aprobados.
  - d) Disposiciones que permitan certificar que los resultados de la prueba son reales y válidos.
  - e) Disposiciones para la revisión y certificación de los informes de prueba.
4. El suministrador debe informar al comprador de las dificultades inusuales encontradas durante las pruebas.

### 2.2.6.2 Revisiones

1. El suministrador debe definir e implantar procedimientos de revisión para verificar que se cumplen los requisitos contractuales del *software*.
2. Las revisiones deben ser identificadas en el proceso general de desarrollo del *software*, del que forman parte integrante. Las revisiones deben ser planificadas y realizadas sistemáticamente, y deben ser críticas con el elemento objeto de revisión.
3. Los procedimientos de revisión deben incluir disposiciones para:
  - a) Describir los objetivos de cada revisión.
  - b) Identificar las funciones, autoridades y responsabilidades de personal involucrado en las revisiones.
  - c) Registrar los resultados de las revisiones.

- d) Asegurar que las acciones resultantes de las revisiones se supervisan a fin de garantizar su finalización en plazo.
4. Toda la documentación del *software* generada bajo el contrato debe, antes de su publicación, ser revisada y aprobada en cuanto a su adecuación por el personal autorizado.

### **2.2.7 Mantenimiento**

1. Si después de la entrega inicial e instalación el mantenimiento del *software* es un requisito especificado, el suministrador debe definir e implantar los procedimientos para realizar esta actividad. Estos procedimientos deben incluir las disposiciones para verificar e informar que el mantenimiento realizado satisface los requisitos especificados.
2. Se deben tener en cuenta:
  - a) El trabajo a realizar.
  - b) Los procedimientos a emplear.
  - c) Los registros e informes a generar.
  - d) Las responsabilidades del suministrador y su interfaz con el comprador.
  - e) Las actividades de gestión de configuración, incluida la identificación del estado inicial del producto a mantener.
  - f) Los métodos para el tratamiento de problemas, su análisis y resolución.
  - g) Las pruebas y aceptación de los cambios.

### **2.3 Recursos humanos**

1. El personal que realiza las tareas específicas asignadas (subcontratados o empleados de la compañía) debe estar cualificado en base a un nivel de estudios, una formación apropiada y/o experiencia requerida.
2. Deben mantenerse los registros apropiados (véase el apartado 2.2.4.10).

### **2.4 Participación y acceso del comprador**

1. El suministrador debe proporcionar al comprador la acomodación y recursos necesarios para el cumplimiento adecuado de su trabajo y toda la ayuda necesaria para la evaluación del programa de calidad de *software* y para la verificación y validación de los productos.
2. El comprador tiene derecho de acceso a cualquiera de las instalaciones del suministrador o subsuministrador donde se estén realizando actividades de cualquier parte del trabajo contratado. No se le deben poner limitaciones al comprador ante cualquier oportunidad para verificar la conformidad de los suministros con los requisitos del contrato. Las herramientas necesarias para la evaluación, verificación y validación deben ponerse a disposición del comprador para un uso razonable.
3. El suministrador debe ser consciente de que las actividades de evaluación, verifica-

ción y validación por el comprador no implican la aceptación, ni sustituyen de forma alguna a las actividades de EVV del suministrador, ni tampoco eximen al suministrador de sus responsabilidades contractuales.

## ÍNDICE

El índice mostrado a continuación está destinado a facilitar la búsqueda de un asunto específico en la PECAL 2210. Se ha escogido un número limitado de términos, lo cual no debe interpretarse como una lista de prioridad. Los términos y expresiones referencian los apartados en que aparecen. Estos pueden aparecer en más de uno. Los «apartados principales» están subrayados.

El apartado 1.4 está dedicado a definiciones y acrónimos.

<b><u>TÉRMINOS O EXPRESIONES</u></b>	<b><u>PÁRRAFO</u></b>
Acción correctiva	1.1, 2.2.4.2, <u>2.2.4.4</u>
Evaluación (véase también EVV)	1.2.5, 1.3, 1.4.1.2, 1.4.2, 2.2.1, 2.2.4.2, 2.2.4.5, 2.2.5, <u>2.2.6</u> , 2.4
EVV	1.4.2, <u>2.2.6</u> , 2.4
<i>Firmware</i>	1.2.1, 1.4.1.3
Gestión de la calidad	2.1, <u>2.2</u> , <u>2.2.1</u> , <u>2.2.2</u> , 2.2.4.5
Gestión de riesgos	2.1, <u>2.2.1</u> , 2.2.4.1
Gestión de la configuración de <i>software</i> o GCS	1.4.2, 2.2.1, 2.2.2, 2.2.3, <u>2.2.4.6</u> , 2.2.4.7, 2.2.4.8, 2.2.6.1, 2.2.7
Herramienta de <i>software</i>	1.4.1.13, 2.2.4.6, <u>2.2.5</u> , 2.2.6, 2.2.6.1, 2.4
Ingeniería del <i>software</i>	2.2.4.1, 2.2.4.9, <u>2.2.5</u>
Manipulación y almacenamiento	1.2.2, <u>2.2.4.11</u> , 2.2.4.12
Mantenimiento del <i>software</i>	1.2.1, 1.2.2, 1.2.5, 1.4.1.11, 2.2.5, 2.2.6.1, <u>2.2.7</u>
PCPS	1.4.2, 2.2.1, <u>2.2.2</u> , 2.2.3, 2.2.4.4, 2.2.4.5, 2.2.4.9
Proceso de desarrollo del <i>software</i>	1.1, 1.4.1.9, 1.4.1.10, 2.2.3, <u>2.2.4.1</u> , 2.2.4.4, 2.2.4.7, 2.2.4.9, 2.2.6, 2.2.6.2
Pruebas	2.2.4.6, 2.2.6, <u>2.2.6.1</u> , 2.2.7
Registros	2.2.4.4, <u>2.2.4.9</u> , 2.2.4.10, 2.2.4.12, 2.2.6, 2.2.7, 2.3
<i>Software</i> no conforme	<u>2.2.4.3</u>
<i>Software</i> no entregable	1.2.1, 1.4.1.5, 2.2.1, 2.2.4.5, <u>2.2.4.8</u>

Software preexistente	1.2.1, 1.4.1.6, <u>2.2.4.7</u>
Subministrador	1.2.2, 2.2.4.3, <u>2.2.4.5</u> , 2.4
Trazabilidad	<u>2.2.4.6</u> , 2.2.5, 2.2.6.1
Validación (véase también EVV)	1.4.1.14, 1.4.2, 2.2.4.2, 2.2.4.5, 2.2.5, <u>2.2.6</u> , 2.4
Verificación (véase también EVV)	1.4.1.2, 1.4.1.15, 1.4.2, 2.2.4.2, 2.2.4.5, 2.2.4.6, 2.2.5, <u>2.2.6</u> , 2.2.6.1, 2.4

