

# Aseguramiento de la calidad del software



## Tabla de contenido

1. Introducción
2. Mapa conceptual
3. Aseguramiento de la calidad de software
  - 3.1. Objetivos de SQA
4. Grupos de SQA
  - 4.1. Roles del grupo SQA
    - 4.1.1. Líder del proyecto
    - 4.1.2. Administrador de desarrollo
    - 4.1.3. Administrador de planificación
    - 4.1.4. Administrador de apoyo
    - 4.1.5. Administrador de calidad y apoyo
    - 4.1.6. Desarrolladores
  - 4.2. Actividades del grupo SQA
    - 4.2.1. Plan de SQA
    - 4.2.2. Revisión de software
    - 4.2.3. Análisis cuantitativo - métricas
5. Fases del ciclo de vida de un proyecto de software para el SQA
  - 5.1. Fase de planificación
  - 5.2. Fase de análisis de requerimientos

5.3. Fase de diseño y desarrollo

Control del documento

5.4. Fase de implementación

Control de cambios

5.5. Fase de integración y prueba

Créditos

5.6. Fase de aceptación y entrega

Creative Common

## 6. Modelos y estándares que garantizan el SQA

6.1. Modelos CMM-CMMI

6.2. Estándar ISO

6.2.1. ISO 9001

6.2.2. ISO 9000-3

6.2.3. ISO 9004-2

6.2.4. ISO/IEC 12207

6.2.5. ISO / IEC 15504 (SPICE)

6.3. ESTANDAR IEEE

6.3.1. IEEE 730- 2002

6.3.2. IEEE 829-1998

6.3.3. IEEE 830 – 1998

6.3.4. IEEE 1012 – 2004

6.3.5. IEEE 1061 - 1998

## 7. Glosario

## 8. Referencias bibliográficas

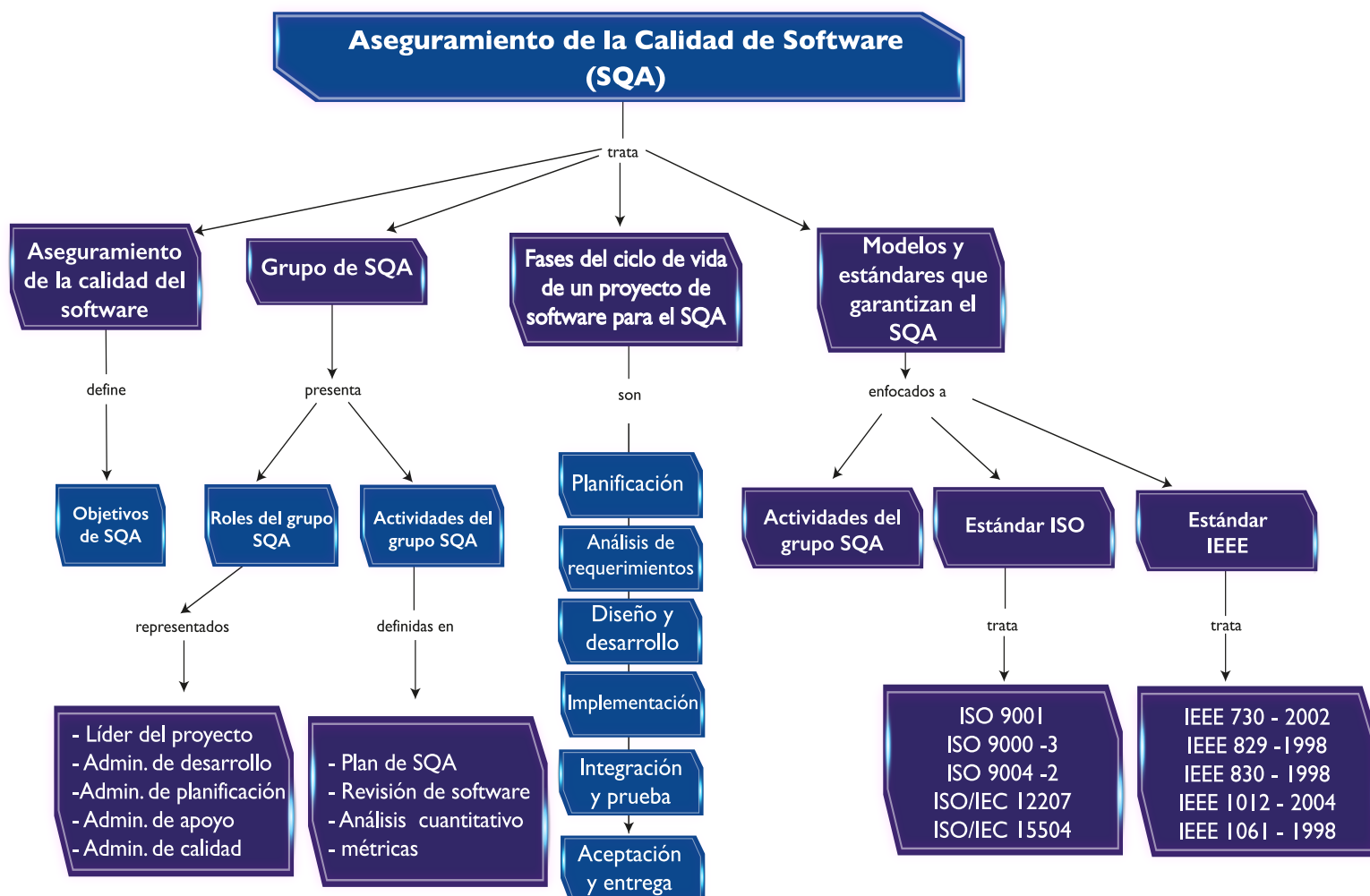
# Calidad en el desarrollo de Software

## 1. Introducción

En la presente actividad de aprendizaje se abordarán conceptos, modelos, estándares, procesos y políticas necesarias a seguir para el aseguramiento de la calidad de software, proceso que se reconoce con la sigla SQA, del inglés: Software Quality Assurance.

SQA reúne un conjunto de procesos a seguir por parte del personal directivo, administrativo y operativo de una compañía. Las evaluaciones, auditorías, implementación de procedimientos para información y seguimiento de errores, entre otros, son temas que se tratarán con el fin de desarrollar competencias en el Aprendiz, de cara al aseguramiento de la calidad de software.

## 2. Mapa conceptual



Fuente: SENA

## 3. Aseguramiento de la calidad de software

El Aseguramiento de la Calidad de Software (SQA), es el conjunto de actividades planificadas y sistemáticas necesarias para aportar la confianza que el software buscará satisfacer, dados los requisitos de calidad. Este aseguramiento se diseña para cada aplicación antes de comenzar a desarrollarla y no después Ortiz (2011).

El SQA pretende establecer:

- Un enfoque de gestión de calidad.
- Métodos y herramientas de Ingeniería del software.
- Revisiones técnicas formales en el proceso del software.
- Una estrategia de prueba multi-escala.
- El control de la documentación del software y de los cambios realizados.
- Procedimientos para ajustarse a los estándares de desarrollo del software.
- Mecanismos de medición y generación de informes.

### 3.1. Objetivos de SQA

El Aseguramiento de la Calidad de Software (SQA) reposa sobre cuatro objetivos principales:

- Planificar las actividades de SQA.
- Verificar la adherencia de los productos de trabajo y de las actividades a los estándares, procedimientos y requerimientos establecidos.
- Informar a los grupos e individuos afectados sobre las actividades de SQA y sus resultados.
- Comunicar a la administración superior sobre desviaciones no resueltas dentro del proyecto.

Para lograr estos objetivos es necesario conformar un grupo responsable de SQA (Software Quality Group), que gestione actividades y tareas a lo largo del ciclo de vida de un proyecto de desarrollo de software y que evalúe su relación con otras áreas de práctica.





## 4. Grupos de SQA

La ejecución del Aseguramiento de la Calidad de Software (SQA), comprende un conjunto metodologías aplicables por un equipo de profesionales especializados, de preferencia ingenieros con perfil de liderazgo.

### 4.1. Roles del grupo SQA

El rol del grupo de SQA consiste en guiar al equipo de desarrollo para alcanzar un producto de alta calidad.

#### 4.1.1. Líder del proyecto

Persona con perfil de ingeniero o afín, con cierto grado de experiencia en la administración y coordinación de proyectos en desarrollo de software, encargado de coordinar al equipo, asegurar que todos cumplan con su trabajo (reporte de datos).

#### 4.1.2. Administrador de desarrollo

Persona con perfil de ingeniero o afín, encargado de controlar el avance del proyecto en su fase de diseño y desarrollo.

#### 4.1.3. Administrador de planificación

Persona con perfil de ingeniero o afín, encargado de establecer el plan de trabajo del grupo SQA, y a su vez, verificar el cumplimiento del mismo.

#### 4.1.4. Administrador de apoyo

Persona con perfil de ingeniero o afín, encargado de ayudar al equipo a conseguir las herramientas necesarias para que pueda realizar el trabajo.

#### 4.1.5. Administrador de calidad y apoyo

Persona con perfil de ingeniero o afín, encargado de proponer un plan de calidad, proceso y resultado.

#### 4.1.6. Desarrolladores

Personal con perfil de ingeniero o afín, con cierto grado de experiencia en análisis, diseño y desarrollo de software, en un lenguaje y ambiente de desarrollo específico.





Figura 2. Grupo SQA  
Fuente: SENA

## 4.2. Actividades del grupo SQA

Toda empresa, aparte de tener una certificación, debe implementar un grupo SQA conformado por recurso humano que permita ofrecer productos de mayor calidad frente a la competencia (López, s.f., p.1). Una vez conformado, el grupo debe desarrollar las siguientes actividades:

- Preparar el Plan de SQA para cada proyecto.
- Participar en el desarrollo de la descripción del proceso de software para un proyecto.
- Revisar las actividades de ingeniería en acuerdo con el proceso definido.
- Auditar los productos de trabajo designados, para verificar su adherencia con aquellos definidos en el modelo de proceso.
- Asegurar que las desviaciones en el desarrollo y en los productos de trabajo sean documentadas y apoyadas por el procedimiento de documentación.
- Registrar cualquier inconformidad e informar a la administración superior.
- Coordinar la gestión de configuración.
- Apoyar la recolección y análisis de métricas de software.

## 4.2.1. Plan de SQA

Un plan de Aseguramiento de la Calidad de Software (SQA), describe las actividades necesarias para garantizar que el software cumple con los requisitos del cliente y se encuentra acorde con los documentos contractuales y técnicos que se hayan suscrito. “El plan de SQA, también describe la manera como el proyecto será auditado, para asegurar las políticas de cumplimiento por parte de la organización” (Mayorga & Arce, 2013).

En virtud de las actividades de aprendizaje estipuladas, se recomienda observar el material complementario de estudio titulado *Plantilla para elaborar un plan de SQA*, el cual contiene dos ejemplos que guían la elaboración práctica de un plan para el aseguramiento de la calidad de software.

## 4.2.2. Revisión de software

Es una práctica que se realiza después del diseño y codificación del software, con el fin de detectar fallos o errores y como resultado se obtiene un conjunto de actividades a realizar, según el diagnóstico producido. Objetivos de la revisión:

- Mejorar las necesidades del producto (software).
- Decidir qué partes del producto (software) no necesitan o no se deben mejorar.
- Obtener calidad uniforme o más predecible, facilitando el trabajo técnico.
- Usar como modelo para detectar errores.

Los factores que permiten ejecutar el monitoreo de procesos, la evaluación del producto y las auditorías al interior de una organización son:

• **Estándares:** son la base de cualquier sistema de calidad de software. Por tal motivo, al final de este material, se hará mención a los estándares orientados al Aseguramiento de la Calidad de Software (SQA). Los estándares, aportan los referentes para la evaluación de actividades y productos; además establecen el marco de trabajo para el desarrollo de software, constituyéndose en un factor crítico (Mayorga & Arce, 2013).

Según Soto (2012), el actuar de los estándares está orientado en los siguientes campos de acción:

- Ciclo de vida del software
- Documentación
- Código fuente





- Criterios para denominar los ítems de configuración
- Procedimientos y protocolos
- **Revisiones de software:** proveen una metodología para detectar defectos en los productos durante el ciclo de vida del software. “Cuenta con seis etapas: Planificación, orientación, preparación, inspección, rework y seguimiento, las cuales son llevadas a cabo por un equipo con tareas y responsabilidades definidas, con documentación específica y por un período determinado” (Soto, 2012).
- **Prueba:** por medio de esta se podrá obtener un reporte en diferentes niveles del proyecto, incluyendo el diseño, planificación y ejecución. Por lo anterior el SQA debe garantizar que:
  - Los procedimientos de prueba verifican los requerimientos según el plan.
  - La versión del software evaluada sea la actual.
  - Los procedimientos sean utilizados.
  - Cualquier problema detectado durante esta actividad, sea registrado e informado oportunamente.
  - Los informes entregados correspondan a la realidad y sean completos.
  - La corrección de los errores sea realizada antes de la entrega del producto final.
- **Análisis de defectos:** ocurren a lo largo de todo el ciclo de vida del software. Por eso será necesario unir esfuerzos para su detección y corrección. Esta actividad es responsable de corregir las deficiencias actuales en el proceso y de disminuir los defectos en futuros proyectos.
- **Gestión de configuración:** (Software Configuration Management, SCM) es el proceso mediante el cual se identifican los elementos del sistema,

verificando que estén correctos y se definen sus necesidades de cambio a lo largo del ciclo de vida. Su objetivo es establecer y mantener la integridad de los productos de software a través del ciclo de vida del proceso y para ello SCM se compone de cuatro actividades principales (Soto, 2012).

  - **La identificación de la configuración:** proporciona un método único y específico para identificar cada instancia (*release*, *versión*, *etc.*) de un producto de software.
  - **El control de cambios:** asegura que cada modificación sobre alguna instancia del producto sea conocida, autorizada y documentada.

- **La contabilidad de la configuración:** permite establecer un seguimiento e informar sobre el estado de la configuración en un tiempo dado.

- **Las auditorías:** establecen si el producto ha sido construido de acuerdo a los requerimientos y que el software esté realmente representado por la documentación que le acompaña.

## 4.2.3. Análisis Cuantitativo - métricas

Lo que no se mide, no se puede mejorar; por ende, la medición entrega herramientas para que el usuario sea exacto a la hora de tomar una decisión. “Los procesos de ingeniería de software, logran medirse en cada una de sus fases (análisis, diseño, implementación o implantación) mediante la definición de métricas que permitan evaluar, predecir o mejorarse continuamente” (Soto, 2012).

**Clasificación de las métricas de software.** Se definen métricas con el fin de estimar costos y esfuerzos en la realización de un proyecto, además, para saber exactamente el grado de avance y de evaluación de riesgos. A través de estos cálculos se pueden establecer planes de mejoras y ajustar flujos de trabajo o tareas. Las métricas permiten la medición de software desde diferentes ambientes y estas se clasifican según el tipo de indicadores, como se observa en la siguiente imagen.

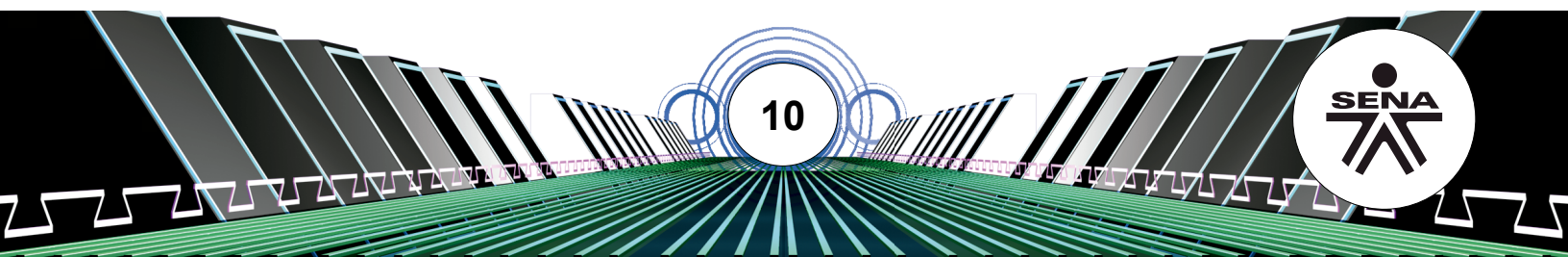
de complejidad	Métricas que definen la medición de la complejidad: volumen, tamaño, anidaciones, y configuración.
de calidad	Métricas que definen la calidad del software: exactitud, estructuración o modularidad, pruebas, mantenimiento.
de competencia	Métricas que intentan valorar o medir las actividades de productividad de los programadores con respecto a su certeza, rapidez, eficiencia y competencia
de desempeño	Métricas que miden la conducta de módulos y sistemas de un software, bajo la supervisión del SO o hardware.
estilizadas	Métricas de experimentación y de preferencia: estilo de código, convenciones, limitaciones, etc.

Figura 3. Clasificación de las métricas de software

Fuente: Verónica Cruz (2016)

## 5. Fases del ciclo de vida de un proyecto de software para el SQA

Durante el ciclo de vida de un proyecto, el Aseguramiento de la Calidad de Software (SQA) es un proceso permanente y continuo que se desarrolla a través de fases.



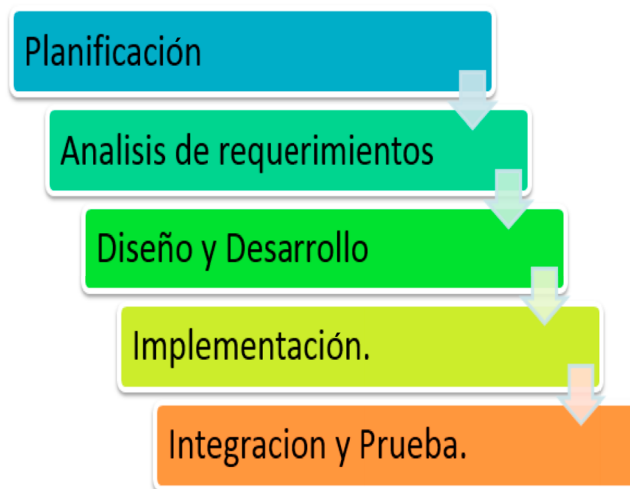


Figura 4. Fases del ciclo de vida del software

Fuente: SENA

## 5.1. Fase de planificación

Soto (2012) indica que “Durante la etapa de planificación, el grupo del SQA debe participar en la elaboración del plan de proyecto”. En esta fase, el grupo SQA tiene la responsabilidad de:

- Producir el plan de SQA y verificar que los procesos, procedimientos y estándares identificados en el plan del proyecto sean apropiados, claros, específicos y auditables.
- Elaborar el contenido del plan de SQA, el cual debe identificar evaluaciones, auditorías y revisiones, estándares, procedimientos de seguimiento, reporte de errores y documentación.

## 5.2. Fase de análisis de requerimientos

Mediante informe de análisis, el grupo SQA debe corroborar que queden expresados todos los requerimientos funcionales, técnicos, operacionales y de interfaz, de manera tal que puedan verificarse en el producto final.

## 5.3. Fase de diseño y desarrollo

En esta fase, dentro de las actividades del grupo de SQA se requiere asegurar:

- La adherencia del diseño y su documentación a los estándares definidos en el plan del proyecto.
- La presencia de todo módulo en el diseño.
- La incorporación de los resultados de las inspecciones en el diseño.
- El ingreso del diseño a la configuración del software tras su aprobación.

## 5.4. Fase de implementación

En esta fase, al grupo de SQA le corresponde auditar:

- Los resultados de las actividades de diseño y codificación.
- El estado de todos los entregables.
- Las actividades de gestión de configuración y de la biblioteca del software.
- Los informes sobre desviaciones y las acciones correctivas.

## 5.5. Fase de integración y prueba

En esta fase al grupo de SQA le corresponde garantizar la concordancia de las pruebas con el plan y los procedimientos definidos, así como también analizar toda desviación que haya sido informada y corregida.

Además, deberá certificar que las actividades de prueba se han completado satisfactoriamente y que el software y su documentación se encuentran listos para la entrega del producto final.

## 5.6. Fase de aceptación y entrega

En esta fase, el grupo de SQA es responsable de realizar la última auditoría de configuración del software desarrollado, con el propósito de controlar y determinar su entrega final.

## 6. Modelos y estándares que garantizan el SQA

A continuación, se realizará un recorrido por los métodos y estándares específicos para el aseguramiento de la calidad de software.

### 6.1. Modelos CMM – CMMI

Es un enfoque de mejoramiento de procesos lanzado en el año 2000 por el Instituto de Ingeniería de Software (SEI) de la Universidad de Carnegie Mellon, bajo la dirección de Watts Humphrey.

Enseña pautas de mejoramiento a las empresas, teniendo en cuenta niveles de madurez en los procesos o áreas claves. De acuerdo con los criterios de CMM-CMMI, una empresa desarrolladora se clasifica según su progreso en cinco niveles de madurez. Las organizaciones que emplean este método, son clasificadas y certificadas según el estadio en el que se encuentran.



## 6.2. Estándar ISO

La sigla ISO representa a la Organización Internacional de Estandarización, en inglés International Organization for Standardization. Según se reporta en el Blog calidad ISO (2013), los orígenes de esta denominación,

Se remontan a 1946 cuando delegados de 25 países decidieron...que, debido a la globalización, nacía la necesidad de establecer normas de reconocimiento internacional, con el fin de homogeneizar aspectos como la seguridad, la salud, el entorno, etc. La organización ISO desarrolla normas internacionales en materia de productos, servicios, procesos, materiales y sistemas, tanto para la evaluación como la gestión y puesta en práctica de procedimientos (Blog calidad ISO, 2013).

A continuación, se orientará la normatividad ISO respecto del Aseguramiento de la Calidad de Software (SQA).

### 6.2.1. ISO 9001

Es una norma internacional que toma en cuenta las actividades de una organización, sin distinción de sector o actividad. El propósito de esta norma es la satisfacción del cliente y la capacidad de proveer productos y servicios que cumplan con las exigencias internas y externas de la organización. Según el blog normas9000.com (2017), “Es la norma de calidad con mayor reconocimiento en el mundo, pues se estima que más de un millón de organizaciones se encuentran certificadas bajo sus estándares”.

### 6.2.2. ISO 9000-3

Esta norma está orientada al control de calidad que debe ser aplicado a todas las fases de la producción de software, incluido el mantenimiento y tareas posteriores a su implantación. Dentro de sus características se destacan:

- El control de calidad debe ser aplicado a todas las fases de la producción de software, incluido el mantenimiento y tareas posteriores a su implantación.
- Debe existir una estricta colaboración entre la organización que adquiere el software y el proveedor del mismo.
- Los proveedores de software deben establecer su sistema de calidad y asegurarse que la organización en pleno lo ponga en práctica.

La Norma ISO 9003, era comúnmente confundida con la norma ISO 9001. Sin embargo, para evitar esta confusión se unieron las normas IEEE 730 y la ISO /IEC 12207 por la ISO 9000-3.





## 6.2.3. ISO 9004-2

Esta norma internacional proporciona un enfoque más amplio sobre la gestión de la calidad que la norma ISO 9001; trata las necesidades y las expectativas de todas las partes interesadas y proporciona orientación para la mejora sistemática y continua del desempeño global de la organización.

El objetivo de esta norma consiste en proporcionar directrices para la aplicación y uso de un sistema de gestión de la calidad, conducente a mejorar el desempeño total de la organización. Esta orientación cubre el establecimiento, operación (mantenimiento) y mejora continua de la eficacia y la eficiencia del sistema de gestión de la calidad.

## 6.2.4. ISO / IEC 12207

Norma orientada a los procesos de ciclo de vida del software de la organización ISO, establece un proceso de ciclo de vida para el software que incluye procesos y actividades que se aplican desde la definición de requisitos, pasando por la adquisición y configuración de los servicios del sistema, hasta la finalización de su uso.

## 6.2.5. ISO / IEC 15504 (SPICE)

Es un modelo de mejora y evaluación de procesos de desarrollo y mantenimiento de sistemas o productos de software, comúnmente utilizado en Europa y basado en SPICE (Software Process Improvement and Capability) (SEI – CMMI, s.f.). Presenta las siguientes características:

- Enfocado en la evaluación, establece un marco para evaluar, más que ser un método propiamente dicho.
- Comprende las áreas de:
  - a. Mejora de procesos.
  - b. Evaluación de procesos.
  - c. Determinación de capacidad.
- Está alineado con el estándar ISO 12207 que define los procesos del ciclo de vida de desarrollo y mantenimiento de Software.

## 6.3. Estándar IEEE

El Instituto de Ingeniería Eléctrica y Electrónica, conocido por sus siglas IEEE, es una asociación mundial de ingenieros dedicada a la estandarización y el desarrollo en áreas técnicas. Según el portal oficial de IEEE, su trabajo es promover la creatividad, el desarrollo y la integración, compartir y aplicar los avances en las tecnologías de la información, electrónica y ciencias en general para beneficio de la humanidad y de los mismos profesionales.

### 6.3.1. IEEE 730-2002

El estándar IEEE 730 es una recomendación para elaborar un SQA para los proyectos de desarrollo de software. Proporciona requisitos mínimos para la preparación y el contenido de los planes de aseguramiento de la calidad de software. Fue escrito para ser utilizado en las fases de desarrollo y mantenimiento del software.

### 6.3.2. IEEE 829-1998

Define la documentación generada en cada una de las fases del proyecto de pruebas.

### 6.3.3. IEEE 830-1998

El propósito fundamental de esta guía es proporcionar orientaciones para redactar informes y documentos útiles. Plantea pautas de contenido y componentes a tener en cuenta en la descripción del software.

### 6.3.4. IEEE 1012-2004

Detalla los procesos de verificación y validación del software y su organización.

### 6.3.5. IEEE 1061-1998

Define el establecimiento, la implementación, el análisis y la validación de métricas de calidad de software.



## 7. Glosario

**CMM:** sigla que refiere al modelo de madurez de capacidades, en inglés Capability Maturity Model. Este modelo establece un conjunto de prácticas o procesos claves agrupados en áreas de la organización.

**CMMI:** sigla del término en inglés Capability Maturity Model Integration. Modelo utilizado para la mejora y evaluación de procesos para el desarrollo, mantenimiento y operación de sistemas de software.

**IEEE:** sigla formada por el nombre del Institute of Electrical and Electronics Engineers (Instituto de Ingenieros Eléctricos y Electrónicos).

**IEEE 1012 – 2004:** estándar que detalla los procesos de verificación y validación del software y su organización.

**IEEE 1061 – 1998:** estándar que define el establecimiento, la implementación, el análisis y la validación de métricas de calidad de software.

**IEEE 730 – 2002:** estándar que define la información que debe contener un plan de aseguramiento de la calidad de software, y su relación con otros procesos implicados (gestión de incidencias, gestión de la configuración).

**IEEE 829 – 1998:** estándar que define la documentación generada en cada una de las fases del proyecto de pruebas.

**IEEE 830 – 1998:** estándar que proporciona una guía de buenas prácticas para la elaboración de una especificación de requisitos.

**ISO:** sigla de la Organización Internacional de Normalización (International Organization for Standardization, en inglés). Se encarga de dictar los lineamientos para el diseño o desarrollo, producción, instalación y servicio; con el objetivo de conseguir la satisfacción del cliente.

**ISO / IEC 12207:** estándar para los procesos del ciclo de vida del Software.

**ISO / IEC 29119:** estándar que proporciona una norma definitiva para las pruebas de software que define el vocabulario, procesos, documentación, técnicas y un modelo de evaluación del proceso de pruebas de software que se puede utilizar dentro de cualquier ciclo de vida de desarrollo.

**ISO / IES 15504 (SPICE):** modelo orientado a la mejora y evaluación de los procesos de desarrollo y mantenimiento de sistemas de información y productos de software.

**ISO 9000-3:** norma que está basada en el control de la calidad que debe ser aplicada a todas las fases de la producción de software, incluido el mantenimiento e implementación.

**ISO 9001:** norma que especifica los requerimientos para un Sistema de Gestión de la Calidad (SGC) organizacional.

**SQA:** siglas del término en inglés: Software Quality Assurance; utilizadas para indicar Aseguramiento de la Calidad en el Software.

**SQUARE:** nombre con el que se conoce a la norma ISO 25000:2005.

## 8. Referencias bibliográficas

Abellán, J. (2014, agosto 1). *Calidad del Software*. [Archivo de video] Recuperado de <https://www.youtube.com/watch?v=Hf-47kSvkHc>

BSI GROUP. (2015). *Norma ISO 9001 - Gestión de la calidad*. Recuperado de <https://www.bsigroup.com/es-ES/Gestion-de-Calidad-ISO-9001/>

Club responsable de calidad (Abril 15 de 2013). *¿Qué significan las siglas ISO?* [web log post]. Recuperado de <http://blogdecalidadiso.es/que-significan-las-siglas-iso/>

Cruz, Verónica (2016, Marzo 3). *Métricas de calidad de software*. [web log post]. Recuperado de <http://calidad-desarrollo-soft-vfcruzhdz.blogspot.com.co/2016/03/>

Estayno, M., Dapozo, G., Cuenca Plethc, L., & Greiner, C. (2009). *Modelos y métricas para evaluar la calidad de software*. Recuperado de <http://sedici.unlp.edu.ar/handle/10915/19762>

Fillottrani, P. (2007). *Calidad en el desarrollo de Software - Modelos de calidad de software*. Recuperado de <http://www.cs.uns.edu.ar/~prf/teaching/SQ07/clase6.pdf>

García, P. (2003). *Calidad en el desarrollo y mantenimiento de software*. Madrid - España: RA-MA Editorial.

Gómez, D. (07 de 05 de 2009). *¿Qué significa calidad de software?* Recuperado de <https://dosideas.com/noticias/reflexiones/564-ique-significa-calidad-de-software>

Mayorga Pabón, J., & Arce Arias, Y. (2013). *Material de formación actividad de aprendizaje 3: Pruebas de Software*. Armenia - Quindío: Centro de Comercio y turismo - Regional Quindío SENA.

Noriega Quintana, D. (s.f.). *Calidad del Software*. Recuperado, de <https://www.monografias.com/trabajos59/calidad-software/calidad-software.shtml>

Presidencia del Consejo de Ministros. (2013). *Oficina Nacional de Gobierno Electrónico e Informática*. Recuperado de: <http://www.pcm.gob.pe/normaslegales/2013/DS-081-2013-PCM.pdf>



# Calidad en el desarrollo de Software

Pressman R. (2010). *Ingeniería del Software - Un enfoque practico* (Vol. Calidad de Software). México: Mac Graw Hill.

Soto, R. Ernesto (2012). *Aseguramiento de la calidad*. Recuperado de [https://www.mindmeister.com/generic\\_files/get\\_file/1221052?filetype=attachment\\_file](https://www.mindmeister.com/generic_files/get_file/1221052?filetype=attachment_file).

Scalone, F. (2006). *Estudio comparativo de los modelos y estándares de calidad del software*. Buenos Aires - Argentina: Universidad Tecnológica Nacional.

Vega Lebrun, C., Rivera Prieto, L., & García Santillán, A. (2008). *Mejores prácticas para el establecimientos y aseguramiento de la calidad de software*. Ciudad de México: Universidad Cristóbal Colón.

## Control del documento

	Nombre	Cargo	Dependencia	Fecha
Autor (es)	Humberto Amaya Alvear	Experto Técnico	Centro de Gestión de Mercados, Logística y Tecnologías de la Información Distrito Capital - Bogotá – SENA.	Octubre de 2017.
	Jorge Bustos Gómez	Asesor Pedagógico	Centro de Comercio y Servicios Regional Tolima.	Octubre de 2017.

## Control de cambios

	Nombre	Cargo	Dependencia	Fecha	Razón del Cambio
Autor (es)	Jorge Bustos Gómez	Asesor pedagógico	Centro de Comercio y Servicios Regional Tolima.	Noviembre de 2017	Correcciones y actualización del formato.



## Créditos



Equipo de Adecuación  
Centro de Comercio y Servicios SENA Regional Tolima  
Línea de Producción

**Director regional:** Félix Ramón Triana Gaitán

**Subdirector de centro:** Álvaro Fredy Bermúdez Salazar

**Coordinadora de formación profesional integral:** Gloria Inés Urueña Montes

**Senior equipo de adecuación:** Jorge Bustos Gómez

**Experto temático:** Humberto Amaya Alvear

**Asesora pedagógica:** Gissela del Carmen Alvis Ladino

**Guionista:** Ofelia Victoria Torres Gómez

**Diseño y diagramación:** Pedro Nel Cabrera Vanegas  
Wilson Alfonso Mahecha Zocadagui  
Jefferson Alejandro Fuertes González

**Programadores:** Davison Gaitán Escobar  
Daniel Santiago Bonilla Posada  
Oscar Daniel Espitia Marín  
Ivan Dario Rivera Guzmán



**creative  
commons**



**Atribución - No Comercial - Sin Derivar.  
(BY-NC-ND)**

No se permite un uso comercial de la obra original ni la generación de obras derivadas.