# MODELOS DE CALIDAD

Los modelos de calidad, son herramientas que brindan guía a las diferentes organizaciones para una mejora continua, orientándolas a ser mas competitivas mediante especificaciones de que requisitos deben cumplir para poder ofrecer servicios y productos de alto nivel. Así, un modelo de calidad de software es un conjunto de buenas prácticas para el ciclo de vida del software, enfocado en los procesos de gestión y desarrollo del mismo.

La calidad en el software, se refiere al grado de satisfacción que tiene el usuario final, teniendo como base el cumplimiento de los diferentes requisitos establecidos en la fase de levantamiento de información de un proyecto, entre estos se tienen en cuenta la funcionalidad, portabilidad, confiabilidad, mantenibilidad, usabilidad y eficiencia.

## CMMI

Las siglas de CMMI hacen referencia a Capability Maturity Model Integration (Integración de Modelos de Madurez de las Capacidades). Este es un conjunto de modelos que se basan en las mejores prácticas en la gestión de los procesos, desarrollados a través de un proyecto conjunto en el que participaron el SEI (Software Engineering Institute), el gobierno estadounidense y algunos miembros de la industria. En estos modelos se establecen cinco niveles de ‘madurez’ de una organización en función de si tiene o no una serie de características que detalla cada modelo. Las organizaciones pueden ser evaluadas y, en función de dicha evaluación, se las puede otorgar un nivel de madurez del 1 al 5 [1]

### Objetivos del CMMI:

* + Ganar reconocimiento de las empresas en la industria
  + Crear valor accionario, de las empresas que hacen uso de estos modelos.
  + Mejorar la y definir el satisfacción del cliente cuando se enfrenta a una empresa con determinado nivel de madurez.
  + Incrementar la participación en el mercado.
  + Producir servicios y Productos de alta calidad.

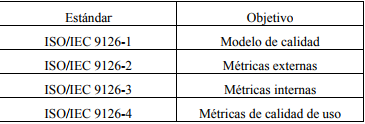
### Niveles de Madurez (por Etapas):

* **Nivel 1 (Inicial):** El proceso es impredecible, es reactivo y pobremente controlado.
* **Nivel 2 (Administrado):** El proceso es reactivo y se caracteriza por su aplicación a proyectos.
* **Nivel 3 (Definido):** El proceso es proactivo y se ve a nivel de la organización.
* **Nivel 4 (Administrado Cuantitativamente):** El proceso es medido y controlado.
* **Nivel 5 (Optimizado):** El proceso se enfoca en la mejora continua.

## ISO/IEC 9126

Es una familia de estándares que regulan la calidad de los productos de software en los que se tiene en cuenta: las características intrínsecas y externas, los modelos que le componen, la forma en la que se miden este tipo de características y la funcionalidad del modelo propuesto.

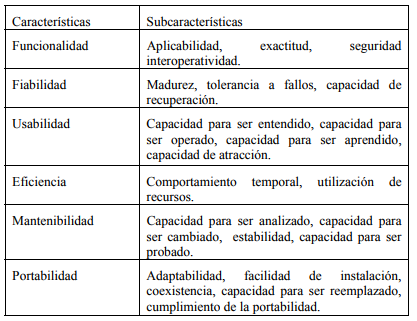
Esta norma se compone de cuatro partes que comparten el mismo título general: Tecnologías de la Información – Calidad de los productos software. [3]



Tomada de [3].

ISO/IEC 9126-1:

Define una serie de características que sustentan el modelo de calidad que se propone. Según ISO, a cada característica se le otorga una métrica que le permite ser cuantificada dentro de un contexto. Por lo que los parámetros de evaluación de cada métrica diferirán si se trata la calidad interna o externa del software.



Tomada de [3]

Las medidas que se utilizan para las métricas varían de acuerdo a la característica que se evalúa. La norma tiene en cuenta 4 tipos:

* + Presencia: 0 o 1
  + Tiempo: Segundos/minutos/horas/días
  + Nivel: 0-4
  + Ratio: %

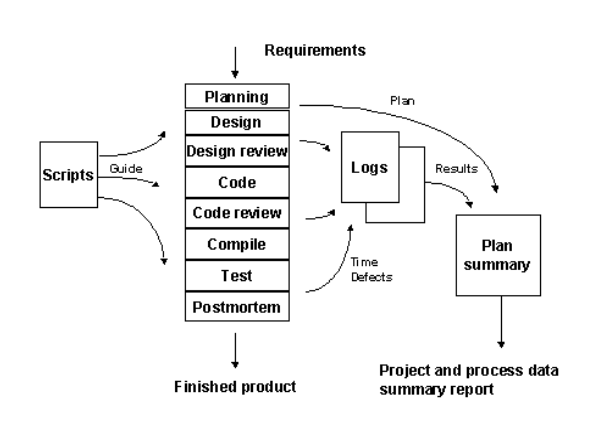
Estas medidas se pueden combinar con el fin de obtener resultados más refinados y concisos, por ejemplo, la medida de nivel puede ser combinada con medidas de tiempo, así, se presume que el resultado será más claro facilitando asi su entendimiento.

Luego de que se definiesen las métricas, al sumar cada uno de sus valores multiplicado por el peso de cada característica (definido a consideración) da como el resultado un dato único que representa el indicador de calidad del software. Ese proceso se puede extender para subcaracterísticas hijas que se sumaran para posteriormente obtener el resultado equivalente de una característica padre. Estos resultados se presentarán ya sea en forma de tabla con sus respectivos valores numéricos o mediante un gráfico de 6 puntas donde cada punta representa una característica del modelo de calidad.

## PSP (Process Software Personal)

Marco de trabajo diseñado para enseñar a los programadores a hacer mejor su trabajo. Muestra cómo estimar y planificar el trabajo y como mejorar la calidad de los programas. Los métodos de calidad llevan consigo tiempo para ser aprendidos, practicados y dominados, sin embargo, estos son de gran ayuda a la hora de mejorar consistentemente la calidad del desarrollo.

PSP es la disciplina del trabajo con alta calidad. El trabajo del ingeniero de software según PSP, se puede resumir en planificar el trabajo, hacer el trabajo de acuerdo al plan y producir productos de máxima calidad. El costo del personal en el desarrollo de un producto de software representa más del 70%, por esta razón, la productividad y los hábitos de los ingenieros de software determinan la mayoría de las veces la conclusión o resultado del proceso de desarrollo de software. [4]



Tomada de [5]

Los principios de PSP son:

1. Cada ingeniero es diferente, para ser más eficiente, debe planificar su trabajo basándose en su experiencia personal.
2. Usar procesos bien definidos y cuantificados
3. Los ingenieros deben asumir la responsabilidad personal de la calidad de sus productos.
4. Cuanto antes se detecten y corrijan los errores menos esfuerzo será necesario
5. Es más efectivo y económico evitar los defectos que detectarlos y corregirlos.
6. Trabajar bien es siempre la forma más rápida y económica de trabajar.

De forma similar a CMMI, PSP maneja una estructura por niveles de acuerdo al cumplimiento de una serie de KPAs (Key Process Area)

### PSP puede ser aplicado en:

• Desarrollo de programas.

• Definición de requerimientos.

• Documentación.

• Pruebas de sistemas.

• Mantenimiento de sistemas. [6]

### **Desventajas de Aplicar PSP**

* + El tiempo requerido para conocerlo puede resultar costoso.
  + El costo emocional por mantener una disciplina.
  + El ego del cambio en las costumbres.

### **Ventajas de Aplicar PSP**

* + La estimulación por nuevas ideas
  + Una estructura de trabajo de mejoramiento personal
  + Tomar control del propio trabajo
  + La sensación de logro
  + Una base mejorada para el trabajo en grupo (TSP)
  + La convicción de que es lo mejor que se puede hacer [7]

## TSP (Team Software Process)

Conjunto de prácticas y estrategias que debe seguir un administrador para poder aprovechar el valor que le ofrece a una empresa o grupo de trabajo [8]. El TSP es un marco de trabajo de procesos diseñado específicamente para equipos de software. Para lograr los beneficios, TSP requiere que los miembros del equipo hayan sido entrenados en el Proceso Personal de Software PSP.

El TSP es un proceso diseñado para equipos de software auto-dirigidos y de alto desempeño, ayudándolos a planear su trabajo, negociar compromisos con la gerencia, dar seguimiento a sus compromisos y producir productos de calidad mientras, de forma paralela, se mejora su rendimiento. El marco de trabajo de TSP incluye roles, plantillas, procesos, guías, especificaciones y listas de chequeo [9]

### Principios del TSP:

* Es necesario incorporar a los técnicos en las actividades de planeación, lo cual ayuda a obtener planes más detallados e incentiva su compromiso con el mismo.
* La carga de trabajo del equipo debe estar equilibrada para mejorar los tiempos requeridos en el proyecto.
* Debe primar el interés por mantener los atributos de calidad definidos para el producto y así aumentar la productividad. [6]



Tomado de [10]

### Los Roles en los equipos en TSP son:

* Líder del Equipo: Dirige al equipo, se asegura que todos reporten sus datos de los procesos y completen su trabajo tal y como se planeó. Realiza los reportes semanales del avance del equipo.
* Gestor de desarrollo: Guía al equipo en el diseño y desarrollo del producto.
* Gestor de Planificación: Apoya y guía al equipo en la planificación y seguimiento del trabajo.
* Gestor de Calidad/Proceso: Apoya al equipo en definir sus necesidades acerca del proceso. Genera estándares para obtener un trabajo uniforme. Modera las inspecciones y revisa cada artefacto generado.
* Administrador de Requerimientos/Soporte: Dirige al equipo en el desarrollo de requerimientos de software y ayuda a dar a conocer la tecnología y en las necesidades de apoyo administrativo [10]

## Modelo ISO/ IEC 15504 (SPICE)

También denominada SPICE (Software Process Improvement and Capability Etermination), proporciona una base para realizar evaluaciones de la capacidad de los procesos de software y permite reflejar los resultados obtenidos sobre una escala común.

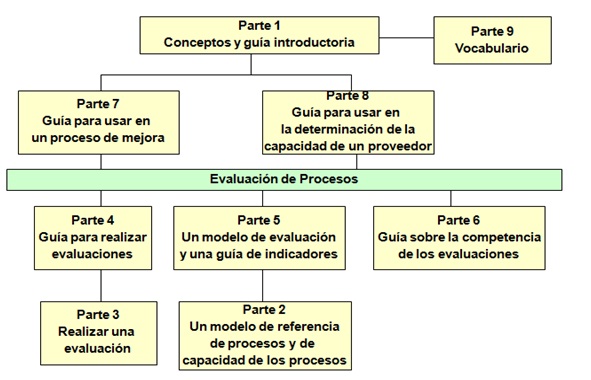
### Sirve para:

* Esta es una norma de mejora continua, puesto que una vez identificado el nivel de la empresa, se proporciona diferentes caminos para poder alcanzar el siguiente nivel.
* Para comprobar la evolución de una organización en el tiempo o para observar su situación respecto a la competencia.
* Para la definición de estrategias de mejora

### Características:

* Esta permite ser aplicada a cualquier o empresa.
* Esta es independiente de la organización, el modelo del ciclo de vida, la metodología y la tecnología.
* Se encarga de cubrir diferentes objetivos para la evaluación de procesos:
* Determinación de la capacidad (niveles de capacidad o de madurez);
* Evaluar el cumplimiento de determinados requisitos del ciclo de vida de desarrollo de software.

### Componentes de la norma SPICE



Tomado de [12]

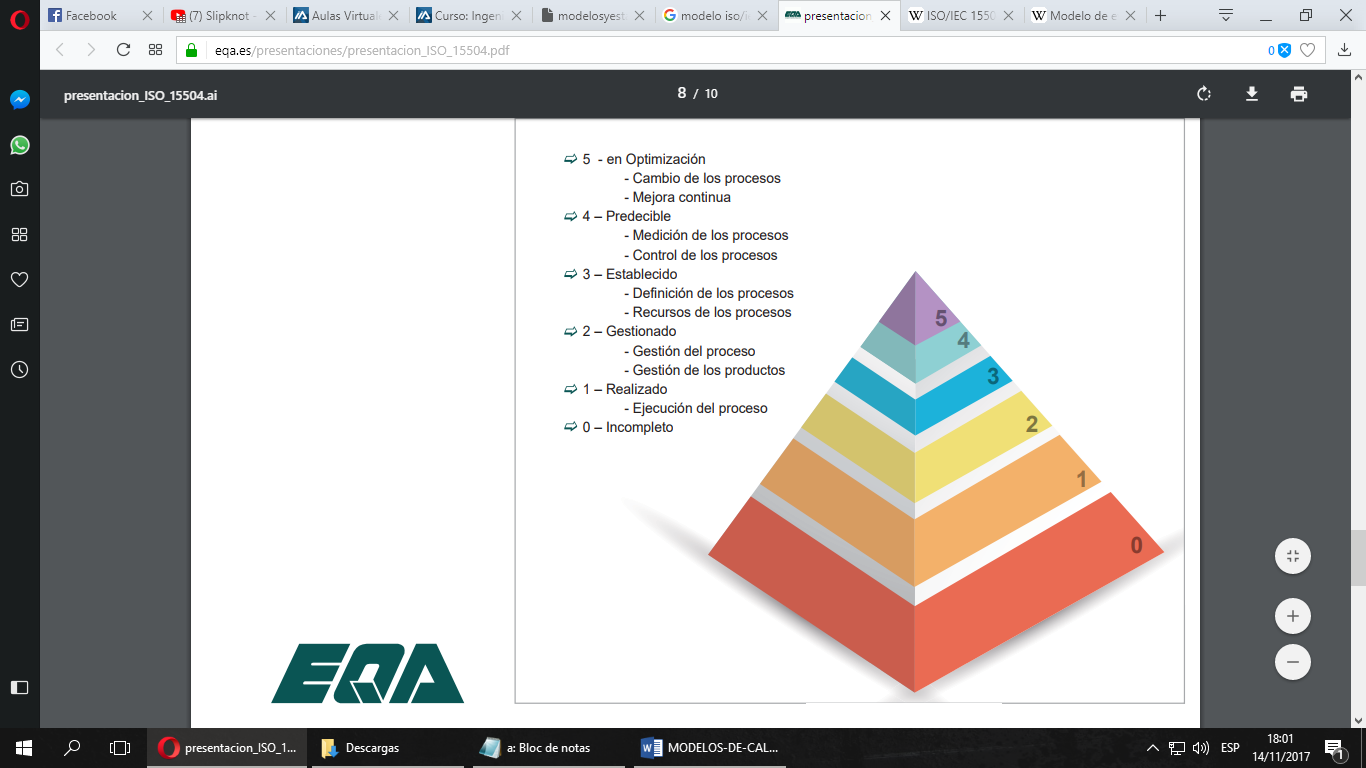
Los atributos de proceso están organizados en niveles de capacidad, permitiendo una puntuación única generada para el proceso. En estos niveles de capacidad se presentan incrementos en la capacidad del proceso, en una escala de 6 puntos desde:

0 – Incompletos: el proceso no es capaz de conseguir sus objetivos

Hasta:

5 – En optimización: el proceso es capaz de alcanzar sus objetivos y está continuamente mejorando.

Esta escala define un camino para la mejora individual de cada proceso



Tomado de [11]

### Ventajas:

* Utiliza un modelo de buenas prácticas actualizado y específico de desarrollo software (ISO 12207:2008).
* Evalúa por niveles de madurez, la evaluación más común entre los modelos de mejora ya es fácil de entender.
* Norma específica para el desarrollo y mantenimiento de software.
* Orientado a mejorar los procesos para contribuir a los objetivos del negocio.

### Desventajas:

* Ya que el dominio de procesos es tan amplio para abarcar todos los posibles ciclos de vida, puede resultar difícil que todos los atributos de proceso sean universales, lo que facilita confusión durante el proceso de desarrollo

**Referencias**

[1] <http://www.cantabriatic.com/que-es-cmmi-y-para-que-sirve/>

[2] <https://msdn.microsoft.com/es-es/library/ee461556.aspx>

[3] <http://redgeomatica.rediris.es/redlatingeo/2008/11993618.pdf>

[4] [http://www.unipamplona.edu.co/unipamplona/portalIG/home\_40/recursos/03\_v13 \_18/revista\_16/27102011/01.pdf](http://www.unipamplona.edu.co/unipamplona/portalIG/home_40/recursos/03_v13%20_18/revista_16/27102011/01.pdf)

[5] <https://jjegonzalezf.files.wordpress.com/2010/12/sshot-5.png>

[6] [https://senaintro.blackboard.com/bbcswebdav/institution/semillas/228106\_2\_VIRT UAL-2015/contenido/oaaps/oaap10/aa2/oa\_calidad/oa.pdf](https://senaintro.blackboard.com/bbcswebdav/institution/semillas/228106_2_VIRT%20UAL-2015/contenido/oaaps/oaap10/aa2/oa_calidad/oa.pdf)

[7] <http://desarrollosoftware.webnode.es/unidad-iii-proceso-personal-de-desarrollo-de-software-psp-/>

[8] <https://www.ibm.com/developerworks/ssa/podcast/13/tsp-psp.pdf>

[9] [https://sg.com.mx/revista/45/integrando-tsp-y-cmmi-lo-mejor dosmundos#.WgkEnFvWzIU](https://sg.com.mx/revista/45/integrando-tsp-y-cmmi-lo-mejor%20dosmundos#.WgkEnFvWzIU)

[10] <http://fernandoarciniega.com/tsp-team-software-process/>

[11] <https://eqa.es/presentaciones/presentacion_ISO_15504.pdf>

[12] <http://software-ufps.blogspot.com.co/2011/07/modelo-de-calidad-spice.html>