|  |
| --- |
| **FORMULARIO DE PRODUCCIÓN DE CONTENIDOS** |
| **Ingeniería en Ciencias de la Computación** |
| **Gestión de la calidad del software** |
| **Evelyn Lissette Hernández de Jiménez** |

En este documento, el experto temático deberá redactar el contenido de la asignatura. Iniciando con la introducción de la asignatura y siguiendo con los contenidos de cada Unidad de Aprendizaje (5 Unidades).

La introducción debe estar estructurada entre 5 y 10 párrafos y de manera dialogada, introduciendo y contextualizando al alumno sobre los principales asuntos abordados en la asignatura, se deben incorporar las competencias (unidad de competencia) e indicadores de logro (elementos de competencia) que serán desarrolladas en las Unidades de Aprendizaje.

Cada Unidad de Aprendizaje deberá ser dividida en 4 (cuatro) clases, conteniendo:

* Nombre de las Unidades y de las clases. Cada clase deberá tener entre 7 y 10 páginas de contenido escrito, de autoría del desarrollador de contenido contratado. Deben estar de acuerdo con el programa de la asignatura y la distribución de los contenidos determinados en el Formulario de Estructuración de la Asignatura (FEA).
* En cada clase, deberán ser presentados, como mínimo, 2 (dos) enlaces relacionados al contenido presentado en esa clase: para cada enlace deberá colocar una breve descripción de lo que está siendo abordado, presentada de forma dialogada en el contenido; tales descripciones y enlaces deben estar destacados dentro del recuadro “Enlaces a recursos externos”
  + Videos [Se incluyen documentales, investigaciones, segmentos de películas, entrevistas u otros ítems que puedan ser utilizados como material didáctico complementario en la disciplina].



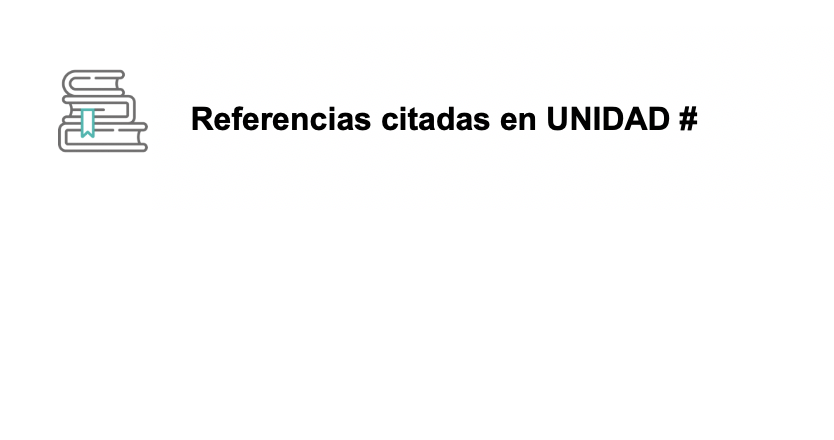
* + Sites [Links de sites que contengan artículos, entrevistas, investigaciones o cualquier otro material que pueda ser utilizado como material didáctico complementario en la asignatura].



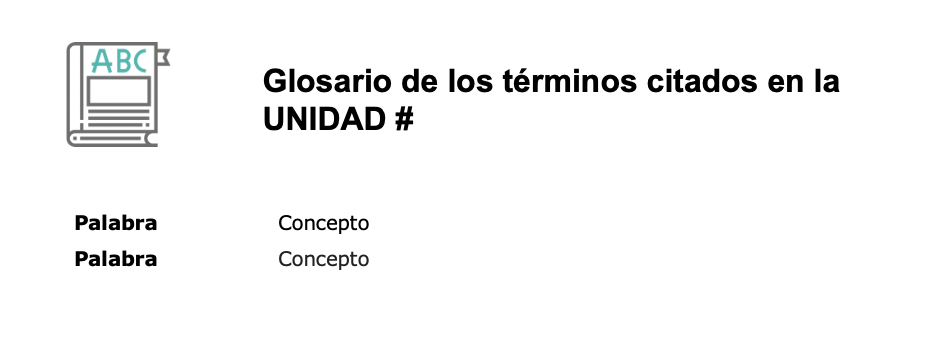
* + Otros [Pueden ser archivos anexados por el desarrollador de contenido y enviados con el contenido de la asignatura].



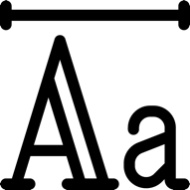
* Al final de cada Unidad deberán ser presentadas, obligatoriamente, todas las referencias citadas en el texto del contenido, de acuerdo con el Sistema de normas APA (séptima edición) con información mínima para que puedan ser adecuadas a las normas. Las frases datos de autor, utilizadas a lo largo del contenido.



* Al final de cada Unidad deberá, también, ser presentado el glosario de la unidad, debe incluir los términos técnicos utilizados en el texto y sus respectivas definiciones.



**Indicaciones para cuerpos de texto**

**Titulo 1:** Arial **24 pts**

**Titulo 2:** Arial **18 pts**

**Titulo 3:** Arial **14 pts**

**Cuerpo:** Arial **11 pts**

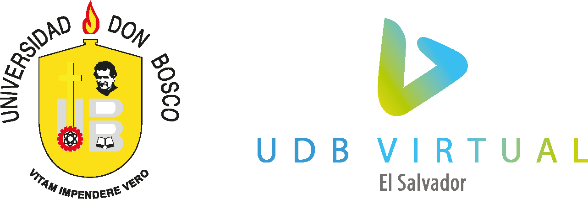
**Otros:**

**Códigos de**

**Programación:** Consolas **11 pts**

**Fórmulas**

**Mátematicas:** Cambria Math **12 pts**



**Gestión de la Calidad de Software**



UNIDAD

**1**

Calidad del software

**Se prohíbe la reproducción total o parcial de esta obra, por cualquier medio, electrónico o mecánico sin la autorización de la Dirección de Educación a Distancia (UDB Virtual).**

Dirección de Educación a Distancia

Apartado Postal, 1874, San Salvador, El Salvador

Tel: 2251-8200 ext: 1743

**Sumario**

**Introducción……………………………………………………………………………. 5**

**Generalidades de gestión de la calidad de software ………………………. 6**

**Calidad del software …………………………………………………………………. 8**

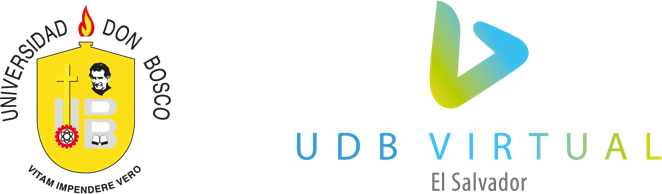
**Aseguramiento de la calidad del software ………………………………………. 9**

**Estrategias de prueba de software ………………………………………………... 14**

**Referencias citadas Unidad 1…………………………………………………………16**

**Glosario**

**Unidad1…………………………………………………………………1**

Introducción de 

la Asignatura

Imagen que contiene computadora, tabla

Descripción generada automáticamente El término calidad, no es nuevo, sino que durante años ha sido una preocupación para los dueños de negocio. Actualmente también los ingenieros de software han tomado más en serio este tema, para lograr ser más competitivos y que su producto, en este caso software, sea de calidad y satisfaga las expectativas de los usuarios.

Para ello es necesario que los profesionales en el área de desarrollo de software, conozcan de herramientas, técnicas y estándares, que le permitan construir todo un andamiaje, sobre lo que es la gestión de la calidad del software y así poder construir instrumentos que permitan la evaluación de cada una de las etapas que comprende el desarrollo del software desde su concepción, el proceso de desarrollo, hasta su finalización, que es el entregable de un producto de software de calidad, que satisfaga las necesidades de todos los usuarios que harán uso de él.

Teniendo en consideración lo planteado en el párrafo anterior, se vislumbra que existe una necesidad latente, en proveer a los profesionales en el área de desarrollo de proyectos de software, una formación que les permita fortalecer las competencias sobre las mejores prácticas que se deben tener en cuenta y a la vez identificar cuáles de ellas son las más pertinentes aplicar, dependiendo del proyecto de software que se vaya a realizar y el tamaño o el contexto de la empresa o cliente para el cuál se está trabajando.

Bajo ese sentido, la asignatura de gestión de la calidad del software tiene como objetivo desarrollar en los estudiantes, las competencias que le permitan establecer todo el marco de selección en cuanto a métodos, técnicas, procedimientos y estándares necesarios para garantizar el proceso de gestión de calidad y que estos sean consistentes con las metas del proyecto y de la organización. A continuación, se presentan las competencias que el estudiante debe de adquirir durante el desarrollo de los contenidos y actividades en las que aplicará cada uno de los temas abordados en cada una de las unidades de aprendizaje de la asignatura.

* Desarrolla sistemas de información informáticos mediante la integración de principios matemáticos, ciencia computacional y prácticas de ingeniería, considerando estándares de calidad y mejores prácticas validadas por la industria del software.
* Mantiene sistemas de información informáticos y servicios generando soluciones propias o recomendadas como resultado del análisis de requerimientos del usuario, en función de estrategias, estándares y tecnologías disponibles.

Para el logro de esas competencias la asignatura de gestión de la calidad de software se ha estructurado en cinco unidades, las cuales comprenden el desarrollo de contenidos que ayudarán a los estudiantes a ir de manera gradual en el desarrollo del andamiaje necesario, para poder aplicar métodos, técnicas, procedimientos y estándares relacionados a la gestión de la calidad de software para garantizar en cada una de las etapas de desarrollo de este: la calidad, la satisfacción de los clientes y la mejora continua. A continuación se lista las temáticas a cubrirse en cada una de las unidades de aprendizaje por las que se encuentra compuesta la asignatura: a) Unidad I: Calidad del software, b) Unidad II: Calidad de productos, c) Unidad III: Calidad de proyectos IT con enfoque (software), d) Unidad IV: Calidad en los procesos y e) Unidad V: Calidad en los datos.

**Clase 1|** Generalidades de gestión de la calidad de software

1. Introducción a la gestión de la calidad del software

La gestión de la calidad de un producto, bien o servicio, debe ser relevante para cualquier organización, ya que esto permite tener una buena aceptación de sus clientes, tanto internos o externos a la organización.

Para el caso del Software como tal, el producto que se entrega es intangible, dado que no es algo que se fabrica, sino que se desarrolla, por lo cuál es de vital importancia que la gestión de la calidad debe prevalecer tanto en el proceso, como en el desarrollo de este.

Bajo ese sentido, es importante que todas las personas involucradas en el proceso del desarrollo de un software en particular tengan claro el concepto de calidad, la evolución histórica de la calidad, la gestión y documentación de la calidad. En los siguientes apartados se desarrolla cada uno de los puntos mencionados anteriormente, con el fin de que las personas cuenten con un basamento teórico relacionados con la gestión de la calidad y así de esta manera pueda implementarla en las empresas desarrolladoras de software.

* 1. Conceptos relacionados con calidad

La palabra calidad no es nueva, y en este siglo XXI, se ha convertido en una de las estrategias dentro de la gestión que tienen las diferentes instituciones, empresas u organizaciones para ser competitivos y líderes en los rubros a los cuales pertenecen.

Pero definir calidad no es fácil, dado que esta palabra engloba una serie de desafíos, en el transcurrir del tiempo, actualmente vivimos en una cuarta revolución industrial, la cual ha tenido un alto impacto en el que hacer de la vida cotidiana de una sociedad, ya que gracias a los descubrimientos de la ciencia y la construcción de todo un ecosistema que cada vez se vuelve más digital, pero esto no implica que la comprensión de la calidad en la actualidad y que es ampliamente aceptada, haya variado radicalmente y que los estándares que norman la calidad, hayan desaparecido, simplemente habrá que hacer ajustes, para que lo nuevo sea normado y pueda ser medido.

En esta ocasión el concepto de calidad que veremos es el que se practica en la actualidad y está enfocado en el cliente, sea este externos o internos, dado que satisfacer las necesidades y expectativas básicas del cliente es una de las cosas que aún son una constante, por tanto, la calidad no se centra solamente en el resultado final de un producto o servicios, sino en todas las etapas que conlleva lograr esos resultados. A continuación, trataremos algunas definiciones, ya que es importante crear un marco conceptual, con el fin de poder orientar de una mejor manera y mejorar lo que es la gestión de la calidad.

Según (Griful & Canela, 2002), las definiciones de calidad pueden varias según el contexto:



* Aplicada al producto, se refiere a una serie de atributos deseables.
* Aplicada al uso del producto, a lo adecuado que es para la aplicación prevista.
* Aplicado a la producción, a que los parámetros del proceso tomen unos determinados valores.
* Aplicada al valor del producto, a que el comprador quede satisfecho con lo que obtiene por el precio que paga. En el lenguaje coloquial, esto es relación calidad-precio.
* En un contexto más ideológico, se puede referir a la excelencia empresarial.

También existen algunos teóricos que se han dado a la tarea de tratar de dar una definición de calidad, según la historia y evolución de esta, se hace referencia a “cinco grandes de la calidad” (Arias Coello, Alicia, s. f., p.12), de los cuales mencionaremos dos de los conceptos planteados por los autores antes mencionados: (Hoyer & Broke, 2001, pp. 2-6)

William Deming

Este autor menciona en su libro Deming: Fuera de la crisis, en el capítulo que lleva por título “Calidad y cliente”, algunos argumentos principales sobre calidad, a continuación, se describen:

* La calidad tiene que estar definida en términos de satisfacción del cliente.
* La calidad es multidimensional. Es virtualmente imposible definir la calidad de un producto o un servicio en términos de una simple característica o agente.
* Hay definitivamente diferentes grados de calidad. Como calidad es esencialmente equitativo a la satisfacción del cliente, la calidad del producto A es mayor a la calidad del producto B, para un cliente en específico, es decir, si A satisface las necesidades del cliente en un mayor grado que lo hace B. Philib B. Crosby

Según Boyer y Broke (2001), Crosby realiza una definición de calidad de nivel uno, en que la calidad de un producto o servicio es equivalente a estar seguro de medir todas las características de un productos o servicio que satisfagan los criterios de especificación.



* 1. Evolución histórica de la calidad

Como hemos mencionado ya en la introducción y en el apartado 1.1, el concepto y la aplicación de calidad no es nuevo, es más como menciona Boyer et al. (2001, p. 2):

Henry Ford, no tuvo un entrenamiento especial en aseguramiento de la calidad, creció en una cultura en cual ya se trabajaba específicamente para obtener excelente desempeño en productos y servicios. De hecho, fue hace casi un siglo que él descubrió que podría obtener ventaja competitiva enfocando su atención en la calidad.

A continuación, se presenta la Tabla 1 en donde se puede ver de manera resumida como ha sido la evolución histórica de la calidad:

Tabla 1

Evolución Histórica de la calidad

|  |  |
| --- | --- |
| Período | Descripción |
| Entorno a los Años veinte | Se establecen conceptos modernos relacionados con la calidad, impulsada, por grandes compañías como la Ford Motor Company, American Telephone & Telegraph, etc. Que comienzan a implantar el criterio de calidad de diversas formas. Fisher, inicia con la aplicación del Diseño estadístico de experimentos. |
| Segunda Guerra Mundial | Durante esta época, Walter Deming o Joseph Juran, participan y desarrollan el programa de gestión de la calidad. También en esta época se desarrolla un nuevo concepto: “control de la calidad toral”, el precursor fue Feigenbaum. Actualmente este concepto se maneja como “Gestión de la calidad total (TQM)”. |
| Años ochenta | En esta época la calidad experimenta una evolución en la industria americana, pasando a considerarse un elemento estratégico fundamental. Crosby introdujo el programa de mejora tratando de concienciar a las empresas para que centraran sus esfuerzos en la necesidad de obtener mayor calidad. La idea era reducir las inspecciones haciendo las cosas bien desde el principio, es decir, “a la primera” (First Time Quality FTQ).  El concepto evoluciona hasta el concepto de Gestión de la calidad total, como nueva filosofía que engloba e integra técnicas que se venían practicando (SPC, DEE) con otras herramientas de más reciente incorporación (QDF, AMFE). |

*Nota:* Elaboración propia, a partir de (Arbós, 2012)

Te invitó a accedas al siguiente enlace: Pineda (2020, enero-junio). Aproximación teórica al concepto de calidad y los sistemas de gestión. Recuperado en noviembre 2022 de <https://aunarcali.edu.co/revistas/index.php/RDCES/article/view/110>, en donde se encuentra más información acerca de la evolución que ha tenido el concepto de calidad, hasta llegar ser reconocido ya no solamente como calidad, sino como “Calidad total o en inglés como Total Quality (TQ)”.



* 1. Conceptos relacionados con la gestión de la calidad

Como hemos podido apreciar hasta el momento, la importancia que tiene la calidad como una estrategia, que permiten a las empresas, organizaciones e instituciones, tanto público, privadas, sector educación, comercio, etc. hacer más competitivos y éxitos en lo que hacen.

El concepto de calidad a evolucionado tanto, que no podemos hablar de manera individual de este, sino que se ha llegado a establecer un concepto nuevo compuesto, por el término gestión y calidad, teniendo como resultado el término “gestión de la calidad”.

(Durán, 1991), habla de un nuevo enfoque de la calidad en la gestión y visualiza tres características “que sustentan y facilitan la buena marchade una empresa: plazo, coste y calidad” (p. 4). Además, argumenta que existe un interés de parte de la dirección de la empresa sobre la calidad, dado que han logrado dilucidar que su aplicación puede “favorecer definitivamente la productividad, eficacia y la imagen de los productos o servicios suministrados” (p. 4).

En ese sentido, (Durán, 1991), nos comenta que las gerencias de las empresas, siendo más consciente de la importancia de la calidad, han realizado esfuerzos por reorientar, reconducirla y proveerla de mecanismos que permitan socializarla, entenderla, aplicarla, medirla y exigirla. Es por ello que hoy en día la calidad se ha vuelto un elemento fundamental como ya mencionamos anteriormente en el nuevo estilo de gestión de las empresas.

Por tanto, es muy importante que tengamos claridad en el significado de este nuevo concepto de “Gestión de la calidad”, que se desprende del concepto de calidad. A continuación se muestra una definición de este nuevo concepto, que es planteado por (Durán, 1991):



Una posible definición de gestión de calidad sería el modo en que la dirección planifica el futuro, implanta los programas y controla los resultados de la función calidad con vistas a su mejora permanente. (p. 5)

* 1. Conceptos relacionados con la documentación de la gestión de la calidad

La documentación de todo proceso es de vital importancia, ya que nos ayuda a mantener el trabajo sistematizado y organizado.

En el caso de la gestión de la calidad, la documentación ofrece las siguientes ventajas:(Meskovska, s. f., párrafo 2)

* Proporciona un marco claro de trabajo de las operaciones de la organización.
* Permite una consistencia de los procesos y una mejor comprensión del Sistema de Gestión de la calidad.
* Brinda evidencias para el logro de los objetivos.

Según el estándar internacional ISO 10013:2001, ofrece una guía que orienta sobre cómo se debe estructurar la documentación de un sistema de gestión de la calidad, esta incluye: manual de calidad, política de calidad, procedimiento de calidad e Instrucciones técnicas (Meskovska, s. f.).

Visita el siguiente enlace: (Meskovska, s. f.). Cómo estructurar la documentación del sistema de gestión de la calidad. Recuperado en octubre de

<https://advisera.com/9001academy/es/knowledgebase/como-estructurar-la-documentacion-del-sistema-de-gestion-de-calidad/#:~:text=El%20prop%C3%B3sito%20y%20los%20beneficios,evidencias%20para%20el%20logro%20de>

Ahí encontrarás información, sobre en qué consiste cada uno de los documentos, que hemos mencionado en el apartado 1.4 párrafo 3, Según la ISO 10013.



* 1. Gestión de la calidad del software

Actualmente nos encontramos inmersos en un mundo globalizado, en donde el acelerado crecimiento de la tecnología hace que el tratar los riesgos que pueden sufrir los sistemas informáticos se vuelvan más complejos, es por ello que se necesita que los desarrolladores y directores de proyectos de software, adopten un método sistemático que les permita asegurar la calidad del producto.

* + 1. Definición

La definición que se ha tomado sobre gestión de la calidad de software es la que nos presenta (Ramos, 2021):



* + 1. Calidad funcional del software

“Refleja qué tan bien satisface un diseño determinado, según los requisitos o especificaciones funcionales” (Ramos, 2021, p. 22).

* + 1. Calidad estructural del software

“Se ocupa del manejo de los requisitos no funcionales que respaldan la entrega de los requisitos funcionales, como la solidez o la capacidad de mantenimiento, y el grado en que el software se produjo correctamente ” (Ramos, 2021, p. 22).

* + 1. Desafío de la calidad del software

En este siglo XXI, el desarrollo de la tecnología se ha dado de una manera veloz, y sobre todo con la pandemia del COVID 19, se aceleraron muchos procesos de digitalización en las diferentes compañías o instituciones, que pertenecen a diferentes sectores de la sociedad.

La demanda de la tecnología sobre todo software, de parte de los usuarios externo e internos, aumentó considerablemente, para unos significa entretenimiento, estudio, negocios, entre otros y para los desarrolladores de proyectos de software, se ha puesto en el ojo del huracán, el tema de calidad, ya que los diferentes software que utilizan los usuarios, para entretenerse, tomar una clase, hacer sus transacciones bancarias, consultar el tráfico, consultar sus notas, etc. se han vuelto actividades del día a día, por lo tanto es necesarios que los productos del software que llegan a los usuarios sean los mejores, en cuánto desempeño, satisfacción en cumplir con las expectativas del cliente que cada vez se vuelve más exigente.

**Clase 2|** Calidad del software

1. Calidad del software

Hablar de la calidad de un de producto de software, es bastante extenso, a continuación, presentamos la Figura 1, en donde se representa en una línea de tiempo, de cómo ha evolucionado el concepto en la historia:

**Figura 1**

*Esquema de la evolución histórica del desarrollo de software y del concepto de calidad asociado.*

Diagrama

Descripción generada automáticamente

*Nota.* Por Panteleo (2016 apartir de Cusumano 2004, p. 20)

* 1. Atributos de la calidad de la ISO9126

Como dice Abud (2004), “hablar de calidad del software implica la necesidad de contar con parámetros que permitan establecer los niveles mínimos que un producto de este tipo debe alcanzar para que se considere de calidad” (p.1). Pero eso no es tarea fácil, ya que muchas veces estos indicadores son más cualitativos, que cuantitativos, es decir que no hay una medición precisa, por lo que ha llevado a realizar una serie de indagaciones y proposiciones de trabajo que permitan identificar factores cualitativos que afectan la calidad del software.

Además, también se ha buscado estandarizar a través de las normas de la familia ISO 9000, el mecanismo de evaluación de la calidad del software (Abud, 2004). Una de las primeras normas de esta familia relacionada para realizar la especificación y evaluación de los atributos de la calidad del software es la ISO 9126, desarrolladas por “ISO/IEC JTC (Subcommittee SC7 – Software and Systems Engineering of International Organization for Standardization)” Djouab y Bari (2016), en ellas se presenta un modelo de calidad,

la primera versión fue presentada en el año de 1991 y posteriormente, en 2001 se publicó una nueva versión de esta primera parte (ISO 2001B) y se completó con tres informes técnicos que proponían diferentes métricas ISO/IEC TR 9126-2: Métricas externas (ISO 2003a), ISO/IEC TR 9126-3: Métricas internas (ISO 2003b) e ISO/IEC TR 9126-4: Métricas para la calidad en uso (ISO 2004f). Panteleo (2016, p. 199)

En la Figura 2, se muestra a manera de esquema, las métricas de calidad en uso, para medir las características o atributos de calidad de software que son aplicables a un producto de software después de ingresar al proceso de operación: Djouab y Bari (2016, p. 371)

**Figura 2**

*ISO/IEC 9126 quality model – external and internal quality*

Diagrama, Texto

Descripción generada automáticamente

*Nota.* Por Djouab y Bari (2016 a partir de Quality of Experience (QoE), p. 371)

Visita el siguiente enlace: Djouab y Bari (2016, mayo) An ISO 9126 Based quality model for the e-learning systems. Recuperado en octubre 2022 de

<https://www.researchgate.net/profile/Rachida-Djouab/publication/276384541_An_ISO_9126_Based_Quality_Model_for_the_e-Learning_Systems/links/55e1de1b08aecb1a7cc828c0/An-ISO-9126-Based-Quality-Model-for-the-e-Learning-Systems.pdf>

Ahí encontrarás información detallada, de cada una de las cuatro partes en que se encuentra divida la ISO 9126.



* 1. Test de comprobación de atributos

En el punto anterior hemos listado los atributos que el estándar ISO 9126 toma en consideración para evaluar la calidad del producto de software que se produce, sin embargo, es importante que cada una de las empresas que desarrollan software, se den a la tarea de establecer un modelo de calidad en el cual incluya, los factores cuantitativos necesarios que les permita valorar el nivel de excelencia del producto dependiendo del objetivo a cumplir y el tipo de software a realizar, esto requerirá que se hagan métricas adhoc, teniendo como base las normas de estandarización.

Dentro de todo el proceso que se realiza para desarrollar un sistema o aplicación, se encuentra la etapa de testeo o prueba, que debe ser transversal en todo el proceso, desde su inicio hasta su finalización y no solo hasta el final. Dado que el objetivo es ir identificando en el proceso, posibles ajustes o fallas para ir solventándolas en el camino, esto elevaría la calidad del producto, caso contrario, el impacto negativo en el resultado del producto estaría latente.

También, es importante señalar que el test de comprobación de los atributos, no solo es de vital importancia que se lleve a cabo en cada una de las etapas del desarrollo del software, sino que es clave incorporar en esas sesiones de revisión a “pares y otros miembros de la organización con experiencia y conocimientos” Panteleo (2016, p. 140)

Otro punto a consideración a tener en cuenta en este proceso es el tener un “criterio de selección que nos permita, por ejemplo, revisar las partes críticas, pero también las partes con potenciales de falencia” Panteleo (2016, p. 140).

A continuación, se presenta en la Figura 3, una forma de trabajo tradicional en relación con las pruebas y en la figura 4 un ambiente de desarrollo con pruebas automatizadas;

**Figura 3**

*Trabajo con tests manuales*

Diagrama

Descripción generada automáticamente

*Nota.* Por Panteleo (2016 apartir de Myers 2004, p. 141)

**Figura 4**

*Trabajo con tests automatizados*

Diagrama

Descripción generada automáticamente

*Nota.* Por Panteleo (2016 apartir de Myers 2004, p. 144)

Para profundizar aún más sobre las pruebas a los sistemas y lograr una mejor comprensión de este, te invito a que visites el siguiente enlace: Pantaleo G. (2016). Calidad en el desarrollo de software. Recuperado en octubre de 2022 de https://cbues.bibliotecasdigitales.com/read/9786076227688/index, aquí encontrarás un libro digital que contiene dicha información, este se encuentra dentro de recursos electrónicos, en la biblioteca de Biblioteca Digital. Capítulo 6



* 1. Garantía de la calidad del software

La garantía de calidad de software es un área clave dentro de la ingeniería de software, ya que el garantizar la calidad del software ayuda a los negocios en la disminución de pérdidas de millones de cantidades de dinero, a causa del mal funcionamiento o el no cumplimiento de los objetivos de los sistemas o aplicaciones.

Teniendo en cuenta lo anterior, es importante que nos cuestionemos entonces, ¿qué es la garantía de la calidad?, a continuación, se presentan dos postulaciones que dan autores, referente a la pregunta planteada: de Antonio A. (s. f., p. 47)

Reifer (1988) define la Garantía de Calidad como “un conjunto de actividades de planificación, estimación y supervisión de las actividades de desarrollo, que se realizan de forma independiente al equipo de desarrollo, de tal forma que los productos de software resultantes cumplen los requisitos establecidos”.

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

Teniendo en cuenta las dos definiciones anteriores, podemos decir entonces, que la garantía de la calidad del software debe entenderse como la planificación de llevar a cabo una seriede actividades o métodos que se ejecutan para asegurar, la calidad del producto de software desarrollado.

De Antonio (s.f), menciona que existen tareas principales, que el grupo de garantía de calidad, deben realizar:

* Planificación de la calidad.
* Supervisión de la calidad.
* Construcción de la calidad.

Para profundizar aún más sobre Actividades de garantía de calidad del software y lograr una mejor comprensión de este, te invito a que visites el siguiente enlace: Gallardo E. (2010). Contribuciones del Instituto Nacional de investigaciones nucleares al avance de la ciencia y la tecnología en México. Recuperado en octubre de 2022 de

<https://www.inin.gob.mx/documentos/publicaciones/contridelinin/Capítulo%2034.pdf>

Acá encontrarán una experiencia de una institución en cuanto al tema de “Garantía de la calidad del Software en el ININ.



* 1. Control de la calidad del software

El estar seguros de que se brindan productos y servicios de calidad es imprescindible para los diferentes tipos de negocios. En las empresas que se dedican al desarrollo de software, están teniendo en este siglo XXI mayor demanda para prestar sus servicios en el desarrollo de producto software a terceros, es decir son parte de un sector económico importante a escala mundial, sobre todo después de la pandemia del COVID 19, en donde todo el mundo se vio forzado de alguna manera a entrar al mundo de la digitalización, en donde las empresas a través de la tecnología encontraron una estrategia, para dar continuidad de sus negocios, todo eso y los nuevos descubrimientos y avances en la ciencia han puesto a la ingeniería del software en un lugar privilegiado.

En relación con eso, es importante que las empresas de software busquen implementar métodos estandarizados bajo sistemas internacionales como lo son las normas ISO, para medir la calidad del software que realizan y de esta manera puedan garantizar a los usuarios o clientes, confiabilidad, mantenibilidad, competitividad y calidad en sus productos.

Para lograr lo antes mencionado, es necesario tener control de la calidad en cada una de las diferentes etapas que se realiza en el desarrollo de software, esto con el fin de “comprobar si un producto posee una determinada característica de calidad en el grado requerido” (de Antonio, s. f. , p.22) , de lo contrario se dice que el producto es defectuoso.

De Antonio (s.f.) refiere, que las actividades de control de calidad se pueden clasificar en dos categorías:

* Controles estáticos, estos permiten analizar el objeto sin necesidad de ejecutarlo.
* Controles dinámicos, estos requieren la ejecución del objeto que se encuentra siendo probado.



*Control de calidad de Software*:

El control de la calidad del software (SQC) es un conjunto de actividades para garantizar la calidad de los productos de software. Estas actividades se centran en determinar los defectos en los productos reales producidos. Implica una acción centra en el producto.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* 1. Costos de la calidad

El tema de costos de calidad, y para este caso del software, implica no solo invertir dinero, sino tiempo, porque lo antes era muy recurrente, que las empresas de desarrollo de software no tomaran muy en serio este tema, esto implicaba dejar toda actividad que se realizara para garantizar la calidad del producto de software de lado, ya que lo primordial, era la entrega del producto en los tiempos establecidos y dentro del presupuesto que se había acordado.

Sin embargo, en la actualidad, es más competitivo quien ofrece productos de calidad, y es por ello, que las empresas dedicadas a este rubro han incorporado el tema de la gestión de la calidad aplicada a la fabricación de software, ya que entregar un producto de software defectuoso, produce inconformidad de parte del cliente y se necesita tiempo y más dinero para poder enmendar lo ya realizado.

De modo que los costos de la calidad del software, deben incluirse dentro la planificación de la realización de esta clase de proyectos, ya que las empresas de software tienen que entregar productos de calidad, en los tiempos y presupuesto establecido, (Slaugher et al., 1998).

También (Slaugher et al., 1998,p. 68) menciona, “que los costos de calidad dividen en dos tipos principales: Conformidad y no conformidad”.

El costo de conformidad es la cantidad gastada para lograr los productos de calidad. Se divide, además, en costos de prevención y evaluación.

Los costos de prevención son aquellos asociados con la prevención de defectos antes de que sucedan.

Los costos de evaluación incluyen medir, evaluar o auditar productos para asegurar la conformidad con los estándares de calidad y desempeño.

El costo de la no conformidad incluye todos los gastos en los que incurre cuando las cosas salen mal. (Slaugher et al., 1998, p. 68)

* 1. Gestión de la calidad total

De acuerdo a (Li et al., 2000), el concepto de la calidad total (TQM, sus siglas en inglés) representa un cambio fundamental, en cuanto a la definición y el tratamiento de la calidad del desarrollo del producto. Esto no es que haya salido de repente, sino fue todo un proceso para llegar a este concepto de TQM, esto viene trabajándose, desde que dio inicio la cuarta revolución industrial, en donde los empresarios estadounidenses, su foco central era el producto.

La definición, como menciona (Li et al., 2000) refiriéndose a la que se encuentra en el diccionario Webster:

“Calidad”, es un grado de excelencia; un distintivo, un atributo.”, Es decir, la calidad es un grado en que un producto está a la altura de su rendimiento, resistencia, mantenibilidad y otros atributos que un cliente espera recibir al comprar un determinado producto.

Para lograr tal propósito, es necesario crear una cultura de concepto de “Calidad Total (TQM)” en el proceso de desarrollo de un producto. Según la Sociedad Americana para Control de la Calidad (ASQC, sus siglas en inglés), la gestión de la calidad total (TQM), es: (Li et al., 2000, pp. 2-3)

Un enfoque de gestión para el éxito a largo plazo a través de la satisfacción del cliente. La gestión de la calidad total se basa en la participación de todos los miembros de una organización para mejorar los procesos, productos, servicios y la cultura en la que trabajan.

La gestión de la calidad total beneficia a todos los miembros de la organización y la sociedad. Los métodos para la implementación de este enfoque se encuentran en las enseñanzas de líderes de la calidad como Philib B., Crosby, W. Edwards Demign, Armand V. Feigenbaum, Kaoru Ishakawa y J.M. Juran.

* + 1. Premios

La calidad, ha tenido una evolución significativa en el tiempo, y muchas empresas han adoptado un modelo de calidad total, la cual ha reemplazado sus métodos tradicionales de chequeo. Esto ha permitido que las empresas logren: un éxito en sus modelos de gestión de la calidad, ya que es una herramienta de apoyo para ir evolucionando y teniendo una mejora continua, a continuación se hace mención de algunos de los primeros modelos y premios de gestión de la calidad que se han desarrollado: (Carro y González, s. f., pp. 49-52)

* Primer modelo fue el de la Unión Japonesa de Científicos e ingenieros (JUSE), cuya organización concede anualmente el Premio Deming a la calidad desde 1951.
* En los Estados Unidos, desde 1987 se concede el Premio Malcolm Baldbrige.
* En 1991, se convoca por primera vez el Premio Europeo de la Calidad.
* En la actualidad está en vías de convocarse el Premio Iberoamericano de la calidad, aunque diversos países de esta región del mundo ya cuentan con premios nacionales de la calidad, como es el caso de la República de Argentina.

**Clase 3|** Aseguramiento de la Calidad del software

1. Aseguramiento de la calidad del software

En la clase 1 y 2 se ha tratado, el tema relacionado con calidad y gestión de la calidad aplicado al desarrollo de software, dado que es importante implementar la calidad en este tipo de producto que no es fabricado sino desarrollado. Ahora en esta clase se tratan temas relacionados con el aseguramiento de la calidad del software, con el fin de conocer todos aquellas actividades o puntos a los cuales hay que darle seguimiento a cada una de las etapas que se ejecutan y son parte del proceso de desarrollo del software.

* 1. Concepto del aseguramiento de la calidad del software

Como ya hemos visto en las dos clases anteriores, sobre la importancia de aplicar gestión de la calidad a los productos de software, ya que ayudan a verificar en cada una de las etapas del desarrollo de este, la calidad de realización y entrega. En la Tabla 2 se presenta una lista de factores que pueden afectar el cumplimiento de los verdaderos requisitos del cliente:

Tabla 2

Factors that can Affect Meeting the True Requirements of the Client

|  |  |  |
| --- | --- | --- |
| Type of requirement | Origin of the expresión | Main cuases of difference |
| True | Mind of the stakeholders | * Unfamiliarity with true requeriments * Instability of requeriments * Different viewpoints of ordering party and users |
| Expressed | User requirements | * Incomplete specification * Lack of standards * Inadequate or difficult communication with the oredering party * Insufficient quality control |
| Specified | Software Specification Document | * Inappropiate use of management and production methods, techniques, and tools. |
| Achieved  Logrado | Documents and Product Code | * Insufficient tests * Insufficient quality control techniques |

*Nota.* Laporte y April (2018 a partir de CEGELEG 1990, p. 21)

Pero ¿cómo aseguramos esa calidad del software?, para dar respuesta a esta pregunta, lo que se necesita, es conocer primero, sobre la definición, del tema, a continuación, se presenta una de las muchas definiciones que se tiene, sobre el concepto de aseguramiento de la calidad.



Aseguramiento de la calidad del software

Es el conjunto de actividades que se realizan para asegurar que el sistema tiene la calidad suficiente. Axelsson y Skoglund (2016)

Según la norma ISO 9000:2000, el aseguramiento de la calidad es la parte de la gestión de la calidad orientada a proporcionar confianza en que se cumplirán los requisitos de calidad. Calidad del software (2012)

La gestión de la calidad del software es un proceso que asegura que se logre el nivel requerido de la calidad del software cuando llega a los usuarios, para que estén satisfecho con su desempeño (p. 1).

* 1. Administración y actividades

Hay que tener claridad en que el aseguramiento de la calidad del software no es sinónimo de hacer pruebas si no que engloba: Calidad del software (2012)

* Un enfoque de gestión de la calidad.
* Métodos y herramientas de Ingeniería del software.
* Revisiones técnicas formales en el proceso de software.
* 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 de generación de informes.

Serrano (2021, posición.164), menciona, que “el aseguramiento de la calidad es un proceso continuo dentro del ciclo vida del desarrollo del software (SDLC) que verifica de forma rutinaria el software desarrollado para garantizar que cumpla con las medidas de calidad deseadas”.

Además, Serrano (2021, posición. 164,174,185), lista actividades, procesos y responsabilidades que deben estar presentes en el aseguramiento de la calidad (ASC):

Actividades

* Definición e implementación de procesos.
* Revisión de cuentas.
* Capacitación.

Procesos

* Metodología de desarrollo del software.
* Gestión de proyectos.
* Gestión de la configuración.
* Desarrollo / gestión de requisitos.
* Estimación.
* Diseño de software.
* Pruebas, etc.

Responsabilidades

Una vez que se han definido e implementado los procesos, el Control de Calidad tiene las siguientes responsabilidades:

* Identificar las debilidades en los procesos.
* Corrija esas debilidades para mejorar continuamente el proceso.

En el siguiente enlace, nos lleva a un artículo, en donde se ha desarrollado un tema relacionado con el aseguramiento de la calidad del software. Carrizo y Alfaro (2018, marzo). Método de aseguramiento de la calidad en una metodología de desarrollo de software con un enfoque práctico. Recuperado en octubre 2022 de <https://www.scielo.cl/scielo.php?script=sci_arttext&pid=S0718-33052018000100114>

Acá encontrarán un artículo en donde se presenta un enfoque práctico como guía estratégica, para administrar la calidad en el desarrollo de un proyecto de software. Para esto, se presenta un método de ACS que consta de tres componentes. La Esencia, que busca que todo el equipo de trabajo entienda el concepto de calidad; que no solo se ve reflejado en actividades o tareas, sino también en la forma cómo trabaja el equipo. Herramientas, que tienen como finalidad controlar la calidad en el proyecto de software y por último, las Métricas utilizadas no solo para medir los resultados obtenidos, sino también para poder mejorar los procesos internos.



* 1. Elementos del aseguramiento de la calidad del software

A continuación, se describen cada uno de los elementos que forman parte del aseguramiento de la calidad del software.

* + 1. Etapas

De acuerdo con el modelo CMMI, el aseguramiento de calidad de un proceso y un producto de software debe considerar. (SEI (2006) referenciado por Hernández (2008), p.14).

* Realización de evaluación objetiva de procesos, productos y servicios generados, tomando como referencia la descripción de los procesos y estándares aplicados.
* Identificación y documentación de comentarios de no conformidad o problemas detectados tanto por el cliente como por el equipo de desarrollo.
* Proporcionar retroalimentación al equipo de desarrollo sobre los resultados de las actividades de control de calidad realizadas.
* Asegurarse de dar seguimiento a los reportes de no conformidades o problemas

En el siguiente enlace, nos lleva a una tesis de maestría en ingeniería de software, y desarrollan un tema de plan de aseguramiento de la calidad. Hernández M. (2008). Definición e implementación de un plan de aseguramiento de calidad en el área de desarrollo de software de LANIA (Maestría en ingeniería en software). Universidad Veracruzana. Recuperado en octubre de 2022 de

https://cdigital.uv.mx/bitstream/handle/123456789/47653/HernandezRodriguezMariadeL.pdf?sequence=1&isAllowed=y

Para profundizar aún más sobre el concepto de calidad y lograr una mejor comprensión de este, te invito a que visites el siguiente enlace: Camisón, Cruz y González (2006). Gestión de la Calidad: enfoques, modelos y sistemas. Recuperado en noviembre 2022 de https://clea.edu.mx/biblioteca/files/original/64db843c11c52aaf913a5322feafd3d8.pdf, aquí encontrarás un libro digital que contiene dicha información. En caso que tuvieses inconveniente de acceder directamente del enlace proporcionado de invitó a que visites la página web de la UDB, en el apartado de biblioteca y luego dirigirse a recursos electrónicos, en donde debes seleccionar la biblioteca virtual Pearson, luego de eso debes loguearte con tus credenciales, con las cuáles accedes al portal web y después en el buscador digitar la palabra “Gestión de la calidad” y te aparecerá la imagen del libro titulado: “Gestión de la calidad: conceptos, enfoques, modelos y sistemas” .



* + 1. Estándares

(Serrano, 2021), en su obra, menciona que, en el aseguramiento de la calidad del software, pueden clasificarse en dos clases principales: (pos. 1619)

* Estándares de gestión de aseguramiento de la calidad del software, incluyendo las metodologías de certificación y evaluación (estándares de gestión de la calidad).
* Estándares del proceso de desarrollo de proyectos de software (estándares del proceso del proyecto).

Estándares de gestión de la calidad

Estos se centran en el sistema SQA de la organización, la infraestructura y los requisitos, dejando la elección de métodos y herramientas a la organización. Con estos estándares las organizaciones pueden asegurar de manera constante que sus productos de software alcancen un nivel aceptable de calidad. Ejemplos de estándares: ISO 9000-3 y el modelo de madurez de capacidad (CMM). (Serrano, 2021, pos. 1619, 16.30)

Estándares del proceso del proyecto

Estos se enfocan en las metodologías para implementar los proyectos de desarrollo y mantenimiento de software. Esto incluye lo siguiente:

* Los pasos para seguir.
* Requerimientos de documentación de diseño.
* Contenido de los documentos de diseño y problemas de revisión.
* Pruebas de software a realizar.
* Temas de prueba.

Muchos estándares de SQA, de esta clase pueden servir como estándares de ingeniería de software y viceversa. (Serrano, 2021, pos. 16.30)

En la Tabla 3, se muestran un resumen de las características de las dos clases principales de estándares mencionados anteriormente:

Tabla 3

Características de estándares de gestión de la calidad y de procesos del proyecto

|  |  |  |
| --- | --- | --- |
| Características | Estándares de gestión de la calidad | Estándares del proceso del proyecto |
| La unidad objetivo | Gestión del desarrollo de software, mantenimiento y las unidades específicas de SQA. | Un equipo de proyecto de desarrollo y mantenimiento de software. |
| El foco principal | Organización de sistemas, infraestructura y requisitos de SQA. | Metodología para la realización de proyectos de desarrollo y mantenimiento de software. |
| El objetivo del estándar | “Que” lograr | “Cómo” realizar |
| El objetivo del estándar | Asegurar la calidad del software del proveedor y evaluar su capacidad de proceso de software. | Asegurar la calidad del software del proveedor y evaluar su capacidad de proceso de software. Asegurar la calidad de un proyecto de software específico. |
| Ejemplos | CMM de ISO 9000-3 SEI | ISO/IEC 12207 IEEEStd 1012-1998 |

*Nota.* PorSerrano (2021, pos. 1648)

* + 1. Revisiones y auditoría

Pressman (2010, p. 354), describe que las revisiones del software son un filtro, para el proceso de ingeniería de software. Esto significa que las revisiones se aplican en varios momentos del desarrollo del software y sirven para la detección de errores y defectos que puedan ser eliminados.

Para Hernández (2008, p. 45), el “aseguramiento de la calidad hace uso de revisiones tanto a nivel de proceso como de producto”, además según el estándar IEEE 730-2002, las revisiones dentro del plan de SQA, deben incluir como mínimo: Hernández (2005, p.25)

* Revisiones de especificación.
* Revisión del diseño de la arquitectura.
* Revisión del diseño detallado.
* Plan de verificación y validación.
* Auditoría funcional (cumplimiento con requerimientos).
* Auditoría Física (consistencia interna y progreso alcanzado contra modelos).
* Revisiones de administración (cumplimiento con el plan).
* Revisión del plan de gestión de la configuración.
* Revisión post-implementación.

Serrano (2021), referencia al modelo de McCall, el cual incluye tres factores de calidad del software en la categoría de revisión del producto, estos son los siguientes: (pos. 132, 142)

* Mantenibilidad

Acá se consideran los esfuerzos que necesitarán los usuarios y el personal de mantenimiento, para identificar las razones de las faltas del software, corregir las fallas y verificar el éxito de las correcciones.

* Flexibilidad

Se ocupa de las capacidades y los esfuerzos necesarios para respaldar las actividades de mantenimiento adaptativo del software.

* Probabilidad

Los requisitos de testabilidad se refieren a la prueba del sistema de software, así como a su funcionamiento. Incluye resultados intermedios predefinidos, archivos de registro y también los diagnósticos automáticos realizados por el sistema de software antes de iniciar el tema.

* + 1. Pruebas

Las pruebas en cualquier proyecto de software son un factor clave. Pressman (2010, p.384), define prueba como:

Un conjunto de actividades que pueden planearse por adelantado y realizarse de manera sistemática. Por esta razón, durante el proceso de software, debe definirse una plantilla para la prueba del software: un conjunto de pasos que incluyen métodos de prueba y técnicas de diseño de casos de prueba específicos.

A continuación, se listan características genéricas, que tiene una plantilla para la prueba: Pressman (2010, p.384)

* + Para realizar una prueba efectiva, debe realizarse revisiones técnicas efectivas. Al hacerlo, eliminará muchos errores antes de comenzar la prueba.
  + La prueba comienza en los componentes y opera “hacia afuera”, hacia la integración de todo el sistema de cómputo.
  + Diferentes técnicas de prueba son adecuadas para distintos enfoques de ingeniería de software y en diferentes momentos en el tiempo.
  + Las pruebas las realiza el desarrollador del software y (para proyectos grandes) un grupo de prueba independiente.
  + Prueba y depuración son actividades diferentes, pero la depuración debe incluirse en cualquier estrategia de prueba.
    1. Colección y análisis de errores

En el apartado anterior, se ha mostrado la importancia que tiene el realizar pruebas, esto con el fin de garantizar que el producto de software cumpla con todo lo requerido por el usuario. Otra de las tareas que es importante administrar es la de relacionada a los errores que pueda encontrarse en cada una de las fases en que se desarrolla el software y no solamente en su fase de finalización, ya que, si eso pasará los costos se elevarían y el tiempo que se debe invertir en corregir es considerable, afectando los tiempos de entrega previstos.

Velez (s. f.),

expresa que si bien es cierto que tenemos la idea generalizada que el software es infalible, no es verídico, ya que los errores y fallas son muchos más frecuentes de lo que pensamos. En ese sentido, el mal diseño de un sistema, la falta de estándares adecuados en el desarrollo de programas o las malas prácticas de programación con llevan no solo falla en los resultados de uno o más proceso sino millonarias pérdidas económicas, tecnológicas y hasta humanas (párrafo 1).

Como es bien sabido en el ámbito de la ingeniería de software, los errores en un sistema de información pueden aparecer en distintas fases de su desarrollo: toma de requerimientos, análisis, diseño, programación, pruebas o implementación; así pues, entre más tarde se detecte la falla, más costoso será corregirla. (párrafo 2)

Por tanto, es importante que se trate de descubrir los errores en etapas tempranas y de esta manera lograr reducirlos y así obtener software de buena calidad y confiable.

La única manera de mejorar es medir cómo se está haciendo algo. El ACS reúne y analiza errores y datos acerca de los defectos para entender mejor cómo se cometen los errores y qué actividades de la ingeniería de software son más apropiadas para eliminarlos. Pressman (2010, p. 270)

* + 1. Administración del cambio

En el desarrollo de un software, puede ocurrir que se den cambios que no estén planificados, ya sea porque el usuario los solicita, porque surgen interpretaciones no adecuadas, ha variado o actualizado el entorno del contexto para el cual se está desarrollando el producto, etc.

Lo descrito anteriormente, trae consecuencias negativas para el desarrollo del producto de software, ya que eso implica hacer ajustes en el flujo normal de trabajo, que se planificó Por ello es importante que esos cambios sean analizados muy cuidadosamente y de manera conjunta y valorar el impacto que se tendrá si son incluidos y si en verdad no podemos dejarlos de lado, porque también causarían afectación al final de la entrega del producto.

En ese sentido, es importante que exista una administración en la solicitud de realización de cambios, dado que todas las personas involucradas en las diferentes fases del proyecto deben estar informadas, para así evitar inconsistencia en la aplicabilidad de estos cambios y así disminuir la afectación negativa que se pueda tener, si estos son llevados a cambio.

Pressman (2010), plantea que “una meta principal de la ingeniería de software es mejorar la facilidad con la que los cambios pueden acomodarse y reducir la cantidad de esfuerzo empleado cuando deban realizarse cambios” (p. 502).

* + 1. Formación de personas

El tema de formación de las personas en el tema de aseguramiento de la calidad en el desarrollo de software es de vital importancia para asegurar la continuidad del negocio y brindar productos de calidad y fiables.

Toda organización de software quiere mejorar sus prácticas de ingeniería de software. Un contribuyente clave de la mejora es la educación de los ingenieros de software, de sus gerentes y de otros participantes. Las organizaciones de ACS llevan el liderazgo en la mejora del proceso de software y es clave para proponer y patrocinar programas educativos. Pressman (2010, p. 370)

**Clase 4|** Estrategias de pruebas de software

Como ya se había conversado sobre las pruebas en la clase 3, y referenciamos a,Pressman (2010, p.384), quien define prueba como:

Un conjunto de actividades que pueden planearse por adelantado y realizarse de manera sistemática. Por esta razón, durante el proceso de software, debe definirse una plantilla para la prueba del software: un conjunto de pasos que incluyen métodos de prueba y técnicas de diseño de casos de prueba específicos.

Teniendo en consideración esa definición a continuación se desarrolla el tema que esta relacionado con las pruebas y es el de estrategias de pruebas de software.

Según la definición encontrada en Wikipedia (*Test strategy*, 2022), se tiene:

Una estrategia de prueba es un esquema que describe el enfoque de prueba del ciclo de desarrollo de software. El propósito de esta es brindar una deducción racional de los objetivos organizacionales de alto nivel a las actividades de prueba reales para cumplir esos objetivos desde una perspectiva de garantía de calidad.

La creación y documentación de una estrategia de prueba debe hacerse de manera sistemática para garantizar que todos los objetivos estén completamente cubiertos y entendidos por todas las partes interesadas. También debe revisarse, cuestionarse y actualizarse con frecuencia a medida que la organización y el producto evolucionan con el tiempo. Además, una estrategia de prueba también debe apuntar a alinear a las diferentes partes interesadas del aseguramiento de la calidad en términos de terminología, niveles de prueba e integración, roles y responsabilidades, trazabilidad, planificación de recursos, etc.

Las estrategias de prueba describen cómo se mitigan los riesgos del producto de las partes interesadas a nivel de prueba, qué tipos de prueba se realizarán y qué criterios de entrada y salida se aplican. Se crean en base a documentos de diseño de desarrollo. Los documentos de diseño del sistema se utilizan principalmente y, en ocasiones, se puede hacer referencia a los documentos de diseño conceptual. Los documentos de diseño describen la funcionalidad del software que se habilitará en la próxima versión. Para cada etapa del diseño de desarrollo, se debe crear una estrategia de prueba correspondiente para probar los nuevos conjuntos de características.

Una estrategia para la prueba de software debe incluir pruebas de bajo nivel, que son necesarias para verificar que un pequeño segmento de código fuente se implementó correctamente, así como pruebas de alto nivel, que validan las principales funciones del sistema a partir de los requerimientos del cliente. Una estrategia debe proporcionar una guía para el profesional y un conjunto de guías para el jefe de proyecto. Puesto que los pasos de la estrategia de prueba ocurren cuando comienza a aumentar la presión por las fechas límite, el avance debe ser medible y los problemas deben salir a la superficie tan pronto como sea posible. Pressman (2010, p.384).

* 1. Diseño, codificación y pruebas

Luego de levantar los requerimientos de lo que el usuario necesita se automatice o actualice, se pasa a la fase de análisis y diseño del software, es aquí en donde hay un mayor coste, luego de tener claro el diseño del sistema, se procede con la codificación y pruebas de este en cada uno de los subproductos entregables. Las pruebas en el desarrollo del producto son muy importantes, por los siguientes apartados se tratan temas relacionados con el testeo de estos, para garantizar entregar un producto funcional.

* + 1. Problemas

Existe la necesidad que se realicen pruebas a los productos de software, ya que uno de los objetivos principales de esto es poder detectar errores o problemas. Para realizar esto se debe en primer lugar tener claro qué se probará.

Actualmente existe una gran variedad de herramientas en las que los probadores de software pueden utilizar para realizar esta actividad, pero aún la mayor carga de trabajo la realiza el ser humano. Pero los profesionales de software de hoy en día y futuros, consideran que las pruebas son un trabajo poco atractivo, lo que genera que sean pocos los profesionales interesados en estar en esta área, sin embargo, puede haber una baja motivación para aquellos que ya laboran realizando testeos al software, lo que puede producir, a realizar pruebas deficientes y pasar por alto los errores del software. Deak et al.(2016)

* + 1. Proceso de diseño

Desde que se inicia un proyecto de software, los encargados deben incluir en la planificación de este, la realización de pruebas, las cuales son un conjunto de actividades que ayudan en la depuración de errores o defectos del producto de software que se está trabajando. Para ello es necesario que los ingenieros de software realicen plantillas, que permitan sistematizar el proceso de la realización de las pruebas. En la Figura 5, se presenta una visión general de las estrategias de prueba de software:

**Figura 5**

*Estrategias de prueba*

Diagrama

Descripción generada automáticamente

*Nota.* Por Pressman (2010, p. 386)

Dentro de la realización de la estrategia de prueba implementar se encuentra el diseño de los casos de pruebas, “estos métodos proporcionan un enfoque sistemático a la prueba”(*Tema 14: Estrategias de prueba del software*, 2022).

Para el enfoque de los casos de pruebas se necesita: requisitos de usuario, requisitos del sistema, diseño del sistema, código. Ver Figura 6.

**Figura 6**

*Modelo de diseño de los casos de prueba*

Diagrama

Descripción generada automáticamente

*Nota.* Por *Tema 14: Estrategias de prueba del software (2022, p. 2)*

* + 1. Coordinación de la construcción

Como se ha dicho anteriormente, el aseguramiento de la calidad de un producto de software es relevante y para lograr es necesario aplicar una serie de métodos, estrategias y técnicas que pueden ayudarnos a cumplir con el objetivo final de entregar un producto confiable y de calidad. Sin embargo, esta no es una tarea fácil.

En esta ocasión, trataremos el tema como organizar las actividades relacionadas con el proceso de pruebas de software, tomando de referencia lo expuesto por Burnstein (2003, referenciado por Rodríguez, 2012, diapositiva 18):

* Planeación, acá se establecen las metas y selecciona una estrategia general de pruebas.
* Preparación, se da introducción del procedimiento general de pruebas a realizar y se construyen los casos de pruebas específicos.
* Ejecución, en esta parte se observa y realiza la medición del comportamiento del producto.
* Análisis, incorpora la verificación y análisis de resultados para identificar si se observaron fallas.
* Seguimiento, si se detectaron fallas, se inicia un monitoreo para asegurar que se elimine el origen de éstas.

En el siguiente enlace, nos lleva una presentación, en donde han desarrollado temáticas relacionadas con las estrategias y técnicas de prueba del software. Rodríguez (2012). Estrategias y técnicas de prueba del software [ Presentación pdf]. Recuperado en octubre de

<https://www.tamps.cinvestav.mx/~ertello/swe/sesion15.pdf>

Acá encontrarán temas como: Conceptos, niveles de prueba, depuración, métodos de prueba, pruebas de caja blanca y pruebas de caja negra.



* + 1. Pruebas

“La prueba de software es un elemento de un tema más amplio que usualmente se conoce como verificación y validación.” Pressman (2010, p. 384)

La verificación se refiere al conjunto de tareas que garantizan que el software implementa correctamente una función específica. La validación es un conjunto diferente de tareas que aseguran que el software que se construye sigue los requerimientos del cliente. Boehm referenciado por Pressman (2010) afirma esto de esta forma:

Verificación: “¿Construimos el producto correctamente?”

Validación: “¿Construimos el producto correcto?”

La verificación y la validación incluyen un amplio arreglo de actividades SQA: revisiones técnicas, auditorías de calidad y configuración, monitoreo de rendimiento, simulación, estudio de factibilidad, revisión de documentación, revisión de base de datos, análisis de algoritmos, pruebas de desarrollo, pruebas de usabilidad, pruebas de calificación, pruebas de aceptación y pruebas de instalación. Aunque las pruebas juegan un papel extremadamente importante en la verificación y validación, también son necesarias muchas otras actividades. Las pruebas representan el último bastión desde donde puede valorarse la calidad y, de manera más pragmática, descubrirse errores. Pero las pruebas no deben verse como una red de seguridad. Como se dice: “no se puede probar la calidad. Si no está ahí antes de comenzar las pruebas, no estará cuando termine de probar”. La calidad se incorpora en el software a lo largo de todo el proceso de ingeniería del software. La adecuada aplicación de métodos y herramientas, revisiones técnicas efectivas, y gestión y medición sólidas conducen a la calidad que se confirma durante las pruebas. Pressman (2010, pp. 384 – 385). A continuación, en la Figura 7, se muestra los pasos de la prueba del software.

**Figura 7**

*Pasos de la prueba del software*

Diagrama

Descripción generada automáticamente

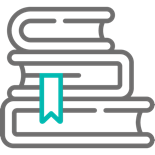
***Nota*.** Por Pressman (2010, p. 387)

En el siguiente enlace, nos lleva a la versión digital en formato pdf del libro Ingeniería de Software. Un enfoque práctico. Pressman (2010). Ingeniería del software. Un enfoque práctico. Recuperado en octubre 2022 de

<http://cotana.informatica.edu.bo/downloads/ld-Ingenieria.de.software.enfoque.practico.7ed.Pressman.PDF>

Revisar los capítulos del 17 al 20, ahí se desarrolla a detalle el tema acerca de las pruebas del software.



**Referencias citadas en UNIDAD I**

Abud, M. (2004). Calidad en la industria del software. La norma ISO-9126. *enero-abril 2004*, *34*. https://www.nacionmulticultural.unam.mx/empresasindigenas/docs/2094.pdf

Arbós, L. C. (2012). *Gestión de la calidad total: Organización de la producción y dirección de operaciones*. Ediciones Díaz de Santos.

Arias Coello, Alicia. (s. f.). *Unidad didáctica: 1. La gestión de la calidad: Conceptos básicos*. https://www.academia.edu/27055056/UNIDAD\_DIDACTICA\_1\_LA\_GESTI%C3%93N\_DE\_LA\_CALIDAD\_CONCEPTOS\_B%C3%81SICOS\_Parte\_1\_Desarrollo\_hist%C3%B3rico\_filosof%C3%ADa\_y\_definici%C3%B3n\_de\_la\_Calidad\_Parte\_2\_El\_enfoque\_a\_los\_clientes\_Calidad\_de\_los\_Servicios\_la\_filosof%C3%ADa\_de\_Parasuraman

*Calidad del software*. (2012). dankocs2012.blogspot.com/2012/aseguramiento-de-la-calidad-de-software.html

Carro, R., & González, D. (s. f.). *Administración de la Calidad Total*.

de Antonio, A. (s. f.). *Gestión, control y garantía de la calidad del software*. https://d1wqtxts1xzle7.cloudfront.net/63957966/LIBRO\_DE\_LA\_MATERIA20200718-79208-cm5ob6-with-cover-page-v2.pdf?Expires=1663915793&Signature=ZwCHNgJpf9DzbvUXufjQDZFY38dr9kRbRRSQyxJ1Ho~WuBBqv6WtuoY5Iqa7saJUvRepk65uVrEPeJhjTdeAhmp7HID-6KgO-TyzQ87zd0ETq9E6Pd3KZVGNd20JU6yvGMsCsKhTQ~HY9D2~z9McCSlkGLMJCvjc1RaozZarDkW38vEjXIej5kInik0JxwUBZC0VjjX0htH2jjhHN8bkoMmZc7ZbwOYl1sfbfsv6goORFcZg75vpBONshqS9i6vefiXKfJjs3EAIpaNm2UfvbMVB-1SmpDBQzVsyrVYj36Ck7N3Mefvq~KhHKh2HB~V7vuH6vCo0ETnxJaftScKCiA\_\_&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA

Deak, A., Stålhane, T., & Sindre, G. (2016). Challenges and strategies for motivating software testing personnel. *Information and Software Technology*, *73*, 1-15. https://doi.org/10.1016/j.infsof.2016.01.002

Djouab, R., & Bari, M. (2016). An ISO 9126 Based Quality Model for the e-Learning Systems. *Mayo 2016*, *6*(5), 370-375.

Durán, M. U. (1991). *Gestión de calidad*. Ediciones Díaz de Santos.

Griful, E., & Canela, M. (2002). *Gestión de la calidad* (1 °). UPC. https://books.google.com.sv/books?hl=es&lr=&id=jpiQg0LLtJ4C&oi=fnd&pg=PA4&dq=que+es+calidad&ots=cGeG1kfVFa&sig=tUu346osasDq7ENqS14-GJ3PXw4#v=onepage&q=que%20es%20calidad&f=false

Hernández Rodríguez, M. de L. (2008). *Definición e implantación de un plan de aseguramiento de calidad en el área de desarrollo de software de LANIA* [Universidad Veracruzana]. https://cdigital.uv.mx/bitstream/handle/123456789/47653/HernandezRodriguezMariadeL.pdf?sequence=1&isAllowed=y

Hoyer, R., & Broke, B. (2001). *Qué es calidad*. https://porquenotecallas19.files.wordpress.com/2015/08/que-es-calidad.pdf

Laporte, C. Y., & April, A. (2018). *Software Quality Assurance*. John Wiley & Sons. https://books.google.com.sv/books?hl=en&lr=&id=pKVFDwAAQBAJ&oi=fnd&pg=PA3&dq=software+quality+assurance&ots=GdawCcB977&sig=Bx1zpfWRfnPvtp2Y3ZyDqcOaULQ&redir\_esc=y#v=onepage&q=software%20quality%20assurance&f=true

Li, E., Chen, H., & Cheung, W. (2000). *Total Quality Managament in Software Development Process*. *14*(1), 4-6 y 35-41.

Meskovska, A. (s. f.). *Documentación del Sistema de Gestión de Calidad de la ISO 9001 – Cómo estructurarla*. Recuperado 20 de septiembre de 2022, de https://advisera.com/9001academy/es/knowledgebase/como-estructurar-la-documentacion-del-sistema-de-gestion-de-calidad/

Panteleo, G. (2016). *Calidad en el desarrollo de software* (2.a ed.). Alfaomega. https://cbues.bibliotecasdigitales.com/read/9786076227688/index

Pressman, R. (2010). *Ingeniería del software. Un enfoque práctico* (Séptima edición). Mc Graw Hill.

Ramos, C. (2021). *Aprende gestión de calidad de software* (Kindle).

Rodríguez, E. (2012, octubre 24). *Estrategias y técnicas de prueba del software*. https://www.tamps.cinvestav.mx/~ertello/swe/sesion15.pdf

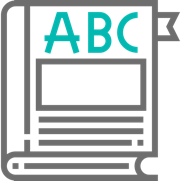
Serrano, C. (2021). *Aprende gestión de la calidad de software en Español: : Construye software y aplicaciones de calidad para tus clientes, cumpliendo los requerimientos necesarios del mismo.*

Slaugher, S., Harte, D., & Krisbnan, M. (1998). *Evaluating the cost of software Quality. The time has come to financially justify investments in software quality improvements, just like we justify other software projects.* *41*(8). https://dl.acm.org/doi/pdf/10.1145/280324.280335

*Tema 14: Estrategias de prueba del software*. (2022). https://kesquivel.files.wordpress.com/2009/08/estrategiaspruebasw.pdf

*Test strategy*. (2022). https://en.wikipedia.org/wiki/Test\_strategy#:~:text=A%20test%20strategy%20is%20an,from%20a%20quality%20assurance%20perspective.

Velez, C. (s. f.). *Errores de Software*. Recuperado 2 de octubre de 2022, de http://www.ii.unam.mx/es-mx/AlmacenDigital/CapsulasTI/Paginas/erroresdesoftware.aspx

**Glosario de los términos citados en la UNIDAD 1**

|  |  |
| --- | --- |
| **ASC** | Aseguramiento de la calidad |
| **Aseguramiento de la calidad del software** | Según la norma ISO 9000:2000, el aseguramiento de la calidad es la parte de la gestión de la calidad orientada a proporcionar confianza en que se cumplirán los requisitos de calidad. (Calidad del software, 2012) |
| **Atributos de la calidad del software** | Son “parámetros que permiten establecer los niveles mínimos que un producto de este tipo debe alcanzar para que se considere de calidad”. Abud (2004, p.1) |
| **Calidad** | la calidad de un producto o servicio es equivalente a estar seguro de medir todas las características de un productos o servicio que satisfagan los criterios de especificación. Boyer y Broke (2001) |
| **Controles estáticos** | Permiten analizar el objeto sin necesidad de ejecutarlo. |
| **Controles dinámicos** | Requieren la ejecución del objeto que se encuentra siendo probado. |
| **Estándares de gestión** | Se centran en el sistema SQA de la organización, la infraestructura y los requisitos, dejando la elección de métodos y herramientas a la organización. Serrano (2021) |
| **Estrategia de prueba** | Es un esquema que describe el enfoque de prueba del ciclo de desarrollo de software. (Wikipedia) |
| **Estándares de proceso** | Se enfocan en las metodologías para implementar los proyectos de desarrollo y mantenimiento de software. Serrano (2021) |
| **Garantía de la calidad del software** | Debe entenderse como la planificación de llevar a cabo una serie de actividades o métodos que se ejecutan para asegurar, la calidad del producto de software desarrollado. |
| **Gestión de la calidad** | Una posible definición de gestión de calidad sería el modo en que la dirección planifica el futuro, implanta los programas y controla los resultados de la función calidad con vistas a su mejora permanente. Duran (191, p. 5) |
| **Gestión de la calidad total** | Un enfoque de gestión para el éxito a largo plazo a través de la satisfacción del cliente. (Li et al., 2000) |
| **Pruebas** | Es un conjunto de actividades que pueden planearse por adelantado y realizarse de manera sistemática. Pressman (2010). |
| **Revisiones** | Las revisiones del software son un filtro, para el proceso de ingeniería de software. Pressman (2010) |
| **SQA** | Security Quality Assurance |
| **TQM** | Total Quality Management |