**FORMATO PARA EL DESARROLLO DE COMPONENTE FORMATIVO**

|  |  |
| --- | --- |
| PROGRAMA DE FORMACIÓN | Análisis y desarrollo de *software* |

|  |  |  |  |
| --- | --- | --- | --- |
| COMPETENCIA | 220501093. Evaluar requisitos de la solución de software de acuerdo con metodologías de análisis y estándares. | RESULTADOS DE APRENDIZAJE | 220501093-04. Verificar los modelos realizados en la fase de análisis de acuerdo con lo establecido en el informe de requisitos. |

|  |  |
| --- | --- |
| NÚMERO DEL COMPONENTE FORMATIVO | 5 |
| NOMBRE DEL COMPONENTE FORMATIVO | Validación de documentos |
| BREVE DESCRIPCIÓN | La validación de documentos son aquellos instrumentos que permiten verificar el estado de un proceso específico de manera detallada. Son tomados como punto de referencia para tener una acertada toma de decisiones dentro del proceso que se ejecutará o el sistema que se está diseñando. |
| PALABRAS CLAVE | Validación, documentos, decisiones, diseño. |

|  |  |
| --- | --- |
| ÁREA OCUPACIONAL | 2 - CIENCIAS NATURALES, APLICADAS Y RELACIONADAS |
| IDIOMA | Español |

1. **TABLA DE CONTENIDOS:** 
   * + 1. **Informe de análisis**

## Proceso de búsqueda de la información evaluación y síntesis de las fuentes

# **Instrumentos de medición o verificación**

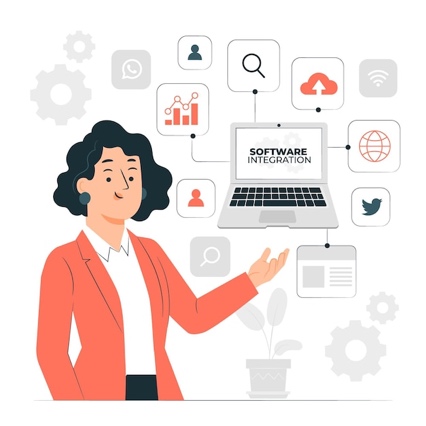
## 2.1 Lista de chequeo

## Cálculo de métricas de *software*

# **Trazabilidad**

1. **INTRODUCCIÓN**

¿Sabías que la falta de calidad de los sistemas que se desarrollan es uno de los mayores contribuyentes a las llamadas crisis del *software*?

Actualmente, el desarrollo del *software* ha tenido un crecimiento muy importante y su implementación en diferentes áreas y productos ha crecido a pasos agigantados; por este motivo, en este componente se profundiza en la creación y correcto uso de los instrumentos que permiten la descripción de las características y las funcionalidades de un sistema, verificando el estado de los procesos e identificando si los requisitos que se esperan obtener son los adecuados, lo cual facilita una correcta trazabilidad y se logra una óptima toma de decisiones.

Para lograrlo, estudiaremos qué es el informe de análisis y qué son las fuentes de información, para continuar con los instrumentos de medición o verificación, donde encontramos la lista de chequeo y el cálculo de métricas de *software*, y finalizaremos con la trazabilidad, uno de los temas trascendentales en el proceso de desarrollo de *software,* porque permite hacer un seguimiento riguroso al proyecto, desde la idea básica hasta los entregable.

Bienvenido.

1. **DESARROLLO DE CONTENIDOS:** 
   * + 1. **Informe de análisis**

**Dorrego (1994), define que el informe de análisis son documentos en los cuales se plasma una serie de procesos, de acuerdo con los requerimientos y objetivos a alcanzar, también conocidos como *reporting* y su fin es realizar un análisis exhaustivo del sistema que se desea desarrollar y el rendimiento de los procesos, logrando influir directamente en la toma de decisiones.**

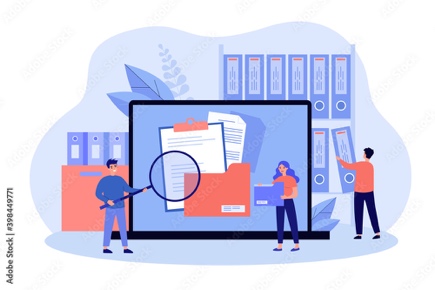
A continuación, en el siguiente video, se presentan algunos aspectos a tener en cuenta para el desarrollo de un informe de análisis.



## **1.1 Proceso de búsqueda de la información evaluación y síntesis de las fuentes**

Al momento de realizar un informe, el principal elemento de trabajo es la información que tenemos o hemos recolectado para tal fin.

Esta información debe ser:



* Confiable.
* Objetiva.
* Precisa
* Actualizada.

Aquí se busca evaluar la calidad de las fuentes como son seguimientos, pruebas, historias de usuario, listas de chequeo, y en este caso específico, el informe de requerimientos de *software*.

Comencemos hablando de las fuentes de información, las cuales son todos aquellos medios de los cuales procede la información, que satisfacen las necesidades de conocimiento de una situación o problema presentado, que posteriormente será utilizado para lograr los objetivos esperados.

De acuerdo a su origen se clasifican en: fuentes primarias y fuentes secundarias.

Según Torres *et al,* (2019):

Las fuentes primarias, son aquellas en las que los datos provienen directamente del objeto de estudio.

Las fuentes secundarias son aquellas que parten de datos preelaborados, como pueden ser datos obtenidos de otros proyectos, de Internet, de medios de comunicación.

### A su vez, las fuentes primarias pueden subdividirse en:

|  |
| --- |
| CF5\_1\_1\_Fuentes\_primarias |

**Tenga en cuenta:** las fuentes primarias se adquieren por medio de una investigación directa al objeto de estudio, por métodos establecidos con anticipación. Para reunir datos primarios, lo ideal es recurrir a un plan.

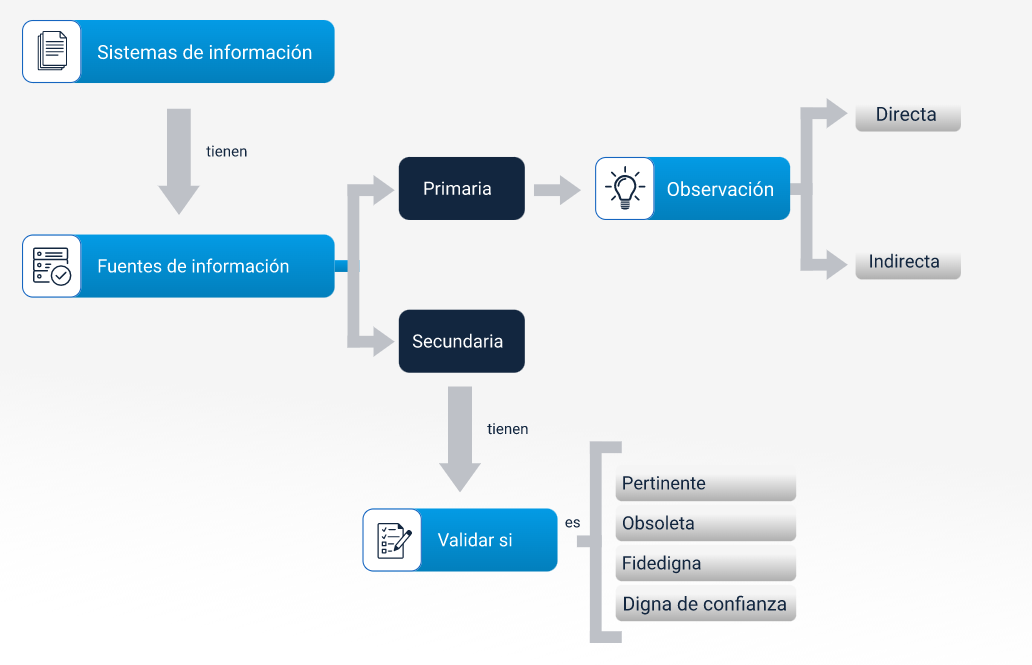
Ahora, las fuentes secundarias, para ser utilizadas, deben ser detalladas con cuatro preguntas básicas:

|  |
| --- |
| CF5\_1\_1\_Fuentes\_secundarias |

En el siguiente esquema, se resume de forma general, la forma en que se estructuran los sistemas de información.

**Figura 1**

Sistemas de información



Teniendo en cuenta la información anterior, debemos tener presente que un informe se debe estructurar de la siguiente manera:

* + - * 1. **Resumen** de no más de 200 palabras describiendo los objetivos y motivación del trabajo.
  1. **Introducción:** una introducción no debe contener más de 2 páginas, se debe detallar el ámbito, objetivos y resultados del trabajo, este debe contener (Acosta Hoyos, 1972):

|  |
| --- |
| CF5\_1\_1\_Estructura\_informe |

* 1. **Contenido:** tantas secciones como hagan falta para explicar los contenidos técnicos del trabajo. Se puede ilustrar con imágenes y esquemas si es considerado necesario.

Uno de los componentes medulares del contenido es la redacción, aquí debemos tener en cuenta:

|  |
| --- |
| CF5\_1\_1\_Contenido |

**Nota**: en las conclusiones debemos argumentar cómo hemos cubierto nuestros objetivos.

* 1. **Metodología:** es la descripción del proceso seguido para la elaboración del trabajo, es decir, se explica el diseño de la investigación, como: hipótesis, variables, tipo de estudio, diseño utilizado, instrumentos de recolección de información, muestreo y procedimientos.

Acosta Hoyos (1972), afirma que la metodología de investigación tecnológica nos brinda las pautas para resolver problemas de la realidad y tiene base empírica, porque aplica los conocimientos teóricos de la ciencia a la práctica, adoptando el método experimental en la solución de los problemas en forma sistémica.

El tipo de investigación es tecnológico, porque busca resolver un problema práctico para satisfacer las necesidades de una solución de *software.*

**Hipótesis**

Según Pájaro (2002), se tienen dos grandes vías a partir de las cuales se descubren hipótesis y por lo tanto, se formulan o redactan:

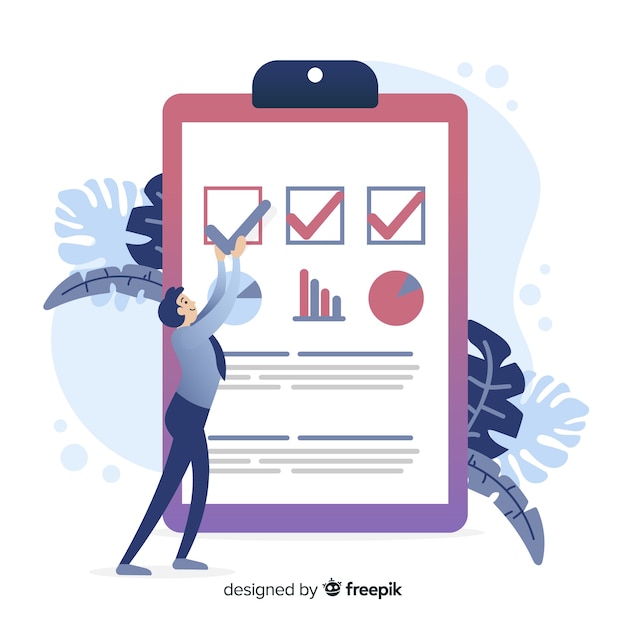
|  |
| --- |
| CF5\_1\_1\_Hipotesis |

En ambos casos, es importante recalcar que el programador ha tenido una etapa previa de observación y acumulación de datos relacionados con el problema que está investigando; por lo que la tradición científica permite desprender al menos tres procedimientos que llevan a la formulación de hipótesis, los cuales son:

1. La información respecto a la observación de un fenómeno es tan clara que conduce fácilmente a una hipótesis.
2. El programador tiene una serie de convicciones o prejuicios filosóficos, producto de una gran experiencia en torno al problema que aborda, lo cual lo conduce fácilmente a una hipótesis,
3. Una analogía, con relación al problema de interés podrá conducir a la hipótesis buscada.

“Las variables son factores que intervienen tanto como causa o como resultado dentro del proceso o fenómeno de la realidad formando parte esencial de la estructura del tema, Las variables intervienen como causa o como efecto en el proceso investigativo, estas quedan identificadas desde el momento en que se define el problema”. (Espinoza Montez 2018)

# **Instrumentos de medición o verificación**



Romero Alvarado (2018), plantea que los instrumentos de medición son herramientas necesarias para garantizar la calidad de los productos, sea por medición, verificación o control; es importante identificar si los instrumentos de verificación cumplen con los parámetros establecidos.

Por otro lado, los instrumentos de verificación son formatos de control que se elaboran con el fin de registrar mediciones o el cumplimiento total o parcial de tareas, acciones, funciones o el desempeño de una actividad en particular, bien sea de un sistema, un funcionario, un proceso o procedimiento.

Existen varios instrumentos de verificación como son: lista de chequeo y cálculo de métricas de *software.*

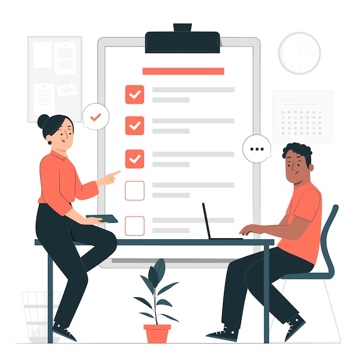
## 

## **2.1 Lista de chequeo**

La lista de chequeo es el instrumento más utilizado para recolectar información en la etapa de requisitos, también conocida como *checklist* u hoja de verificación. Es un formato que puede ser analógico o digital, y su función es comprobar de una forma metódica y sistemática, el cumplimiento o no de los requisitos que contiene la lista, por este motivo debe ser de fácil uso y muy concretas.

Es preferible redactar un cuestionario fácil de responder, que pueda contemplar todos los sitios necesarios y más fundamentales; por lo general, la *checklist* tiene varios destinatarios y usuarios, lo cual sirve para hacer un auto control o control interno, aunque también sirve para supervisar el trabajo de otros, pudiendo verificar el cumplimiento o no de ciertas funciones o parámetros; a esto lo conocemos con el nombre de control externo.

¿Qué debemos tener en cuenta al momento de elaborar una lista de chequeo?



* Se debe contener toda la información que se necesite.
* Debe ser corta, concisa y entendible.
* Debe estar redactada en un lenguaje sencillo y comprensible y con muy buena ortografía.
* El formato debe ser amigable y con buen diseño.

Debemos seleccionar un nombre clave, el cual identifique para qué sirve el instrumento, por ejemplo, verificación de equipos de cómputo; también se deben escribir las categorías o posibles variables de la información que se espere encontrar, establecer la escala de incidencia y diseñar la cuadrícula, asegurándose que todas las partes de la lista de chequeo, estén claramente descritas.

Esta herramienta busca quitar problemas y que se obtenga un formato claro y fácil de usar, entre más básico mejor.

### Los siguientes, son los pasos para elaborar una lista de chequeo adecuada:

|  |
| --- |
| CF5\_2\_1\_Pasos |

Al momento de verificar este proceso, debemos tener en cuenta:

|  |
| --- |
| CF5\_2\_1\_Tener\_en\_cuenta |

La siguiente tabla, presenta un ejemplo de lista de chequeo.

**Tabla 1**

Lista de chequeo Módulo de registro de usuarios

| Ítem | Criterio o Actividad | Cumple | No Cumple | No  Aplica |
| --- | --- | --- | --- | --- |
| 1 | ¿El tiempo de respuesta en los procesos es el esperado? |  |  |  |
| 2 | ¿Se especifican los criterios de seguridad en el sistema? |  |  |  |
| 3 | ¿El proceso está cuantificado en cantidad de usuarios y ancho de banda? |  |  |  |
| 4 | ¿Se especificaron posibles fallas en el sistema? |  |  |  |
| 5 | ¿Se definieron contingencias a posibles fallas? |  |  |  |
| 6 | ¿Hay estrategias de detección de errores en el sistema? |  |  |  |
| 7 | ¿Se especifican requisitos mínimos de hardware para la implementación del sistema? |  |  |  |

También se puede generar una tabla donde se coloque el número del criterio o ítem y una observación.

## **2.2 Cálculo de métricas de *software***

La ingeniería del *software* tiene como objetivo producir un sistema de óptima calidad. Para lograr este objetivo, se deben usar métodos efectivos junto con herramientas y aplicaciones actuales. Al mismo tiempo, un buen desarrollador debe medir si la calidad se va a cumplir en todo aspecto.

La definición estándar de calidad en ISO-8402 es “La totalidad de rasgos y características de un producto, proceso o servicio que sostiene la habilidad de satisfacer estados o necesidades implícitas”. (López Bartolomé, 2014)

Pereira *et al* (2012), refieren que la calidad de una solución de *software* o aplicación, es tan buena como los requisitos que la detallan y las mediciones que evalúan los siguientes aspectos:

1. Requisitos.
2. Análisis.
3. Modelos de diseño.
4. Código fuente.
5. Casos de prueba.

Para obtener estas métricas de calidad, el desarrollador debe usar medidas técnicas, que evalúan la calidad con objetividad, y los errores y defectos.

Los anteriores autores, afirman que las métricas que provienen de estas medidas, proporcionan una indicación de la efectividad de las actividades de control, de calidad, errores detectados, arrojando una visión profunda de la eficacia de cada una de las actividades evaluadas en la métrica. A continuación, se presenta los factores y la categorización de los factores en el desarrollo de *software:*

|  |
| --- |
| CF5\_2\_2\_Factores\_Categorizacion |

Ahora bien, Pérez Quintero (2019), recomienda presentar el esquema de puntuación desarrollado por McCall como una escala donde se emplean unas métricas y criterios. Para conocerlas, lo invitamos a explorar el PDF **Tabla de métricas de *software*.**

# 

# **Trazabilidad**

Es uno de los temas trascendentales en el proceso de desarrollo de *software*, porque permite hacer un seguimiento riguroso al proyecto desde la idea básica hasta los entregables. Los modelos de trazabilidad facilitan que los desarrolladores realicen una correcta evaluación del costo e impacto de las posibles modificaciones en los modelos y código del *software*, ayudando a enlazar o relacionar los requisitos con otros elementos del ciclo de vida, principalmente, casos de prueba y código.

En la opinión de Hernández *et al* (2015), hay que tener en cuenta que el código que se implementa debe ser consecuente con los requisitos, se puede estimar con precisión el esfuerzo que lleva implementar una petición de cambio sobre un requisito. Hay que tener en cuenta que la trazabilidad debe ser bidireccional, de requisitos a código y código a requisitos. Para este fin, se utiliza una matriz de trazabilidad de requerimientos, lo cual asegura un total cumplimiento del proyecto.

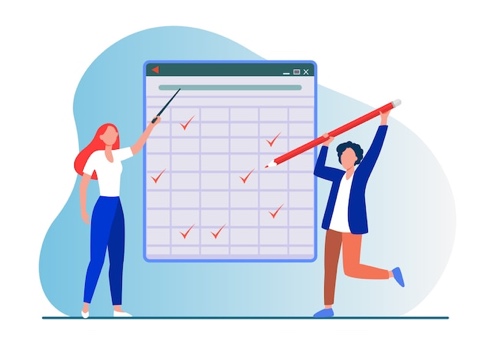
Para elaborar esta matriz, hay que tener como insumo fundamental los requisitos de *software*, entre ellos encontramos:

|  |
| --- |
| CF5\_3\_Requisitos |

Entonces, ¿cuál es la finalidad de elaborar una matriz de trazabilidad?

|  |
| --- |
| CF5\_3\_Finalidad |

¿Y cómo elaboramos una matriz de trazabilidad?

Tradicionalmente, se puede hacer manualmente con lápiz y papel o en una hoja de cálculo, aquí se relacionan los requisitos y los objetivos que se plantearon, adicionalmente, se acompaña de unos campos que aportan información y dan coherencia a esta relación.

Hoy en día existen programas de herramientas de gestión que ayudan para este fin, como Redmine, Trac o similar, aunque hay un gran abanico y de uso libre en el mercado.

A continuación, analicemos un ejemplo de matriz de trazabilidad.

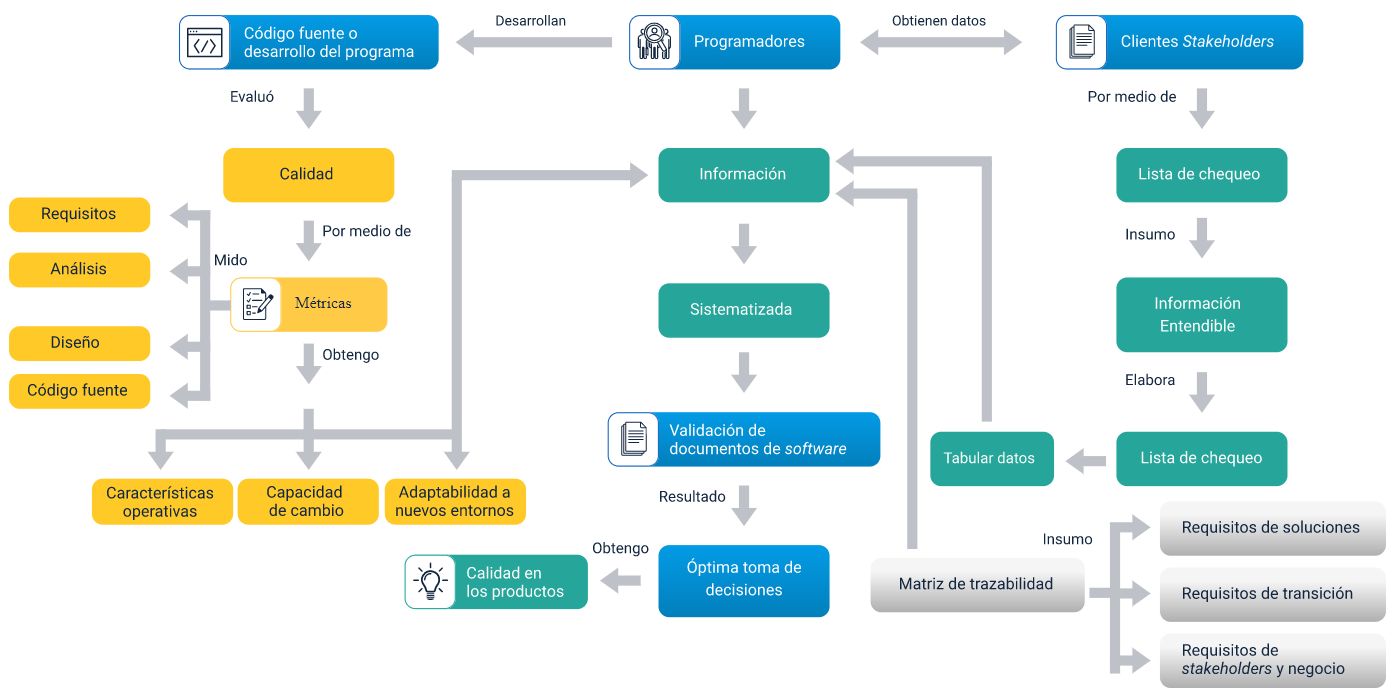
**Tabla 2**

**Ejemplo** Matriz de trazabilidad

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ID | Requisitos | Tipo | Prioridad | Estado | Objetivo | Funcionalidad | Entrega |
| 1 | El sistema recoge los siguientes datos: Nombre  Apellido  Cédula  Correo electrónico Dirección  Teléfono | Venta | Alta | Activo | Obtener datos del usuario. | Registrar  Modificar  Eliminar | Entregado |
| 2 | Los datos serán registrados por un usuario registrado en el sistema. | Venta | Media | Activo | Garantizar la seguridad de la información. | Registrar  Modificar  Eliminar | Pendiente |
| 3 | El sistema recogerá los datos que el cliente va a adquirir con su código de barras. | Venta | Alta | Activo | Mejorar el control de existencias de productos . | Registrar  Modificar  Eliminar | Entregado |
| 4 | El sistema validará que los datos sean correctamente ingresados. | Venta | Baja | Activo | Validaciones | Registrar  Modificar  Eliminar | Pendiente |
| 5 | El sistema será capaz de generar la factura con los datos, ingresado haciendo la búsqueda en la base de datos. | Facturación | Alta | Activo | Agilizar el sistema de facturación. | Registrar  Modificar  Eliminar | Pendiente |
| 6 | El sistema será capaz de atender un número limitado de consultas o peticiones. | Venta | Alta | Activo | Control de sentencias SQL. | Registrar  Modificar  Eliminar | Pendiente |

1. **SÍNTESIS**

A continuación, se presenta una síntesis de la temática estudiada en el componente formativo:



1. **ACTIVIDADES DIDÁCTICAS (Se debe incorporar mínimo 1, máximo 2)**

|  |  |
| --- | --- |
| DESCRIPCIÓN DE ACTIVIDAD DIDÁCTICA | |
| Nombre de la Actividad | Conceptos básicos de validación de documentos |
| Objetivo de la actividad | Afianzar algunos de los conceptos de validación de documentos. |
| Tipo de actividad sugerida | Arrastrar y soltar |
| Archivo de la actividad  (Anexo donde se describe la actividad propuesta) | CF5\_Actividad\_didactica.docx |

1. **MATERIAL COMPLEMENTARIO:**

|  |  |  |  |
| --- | --- | --- | --- |
| Tema | Referencia APA del Material | Tipo de material  (Video, capítulo de libro, artículo, otro) | Enlace del Recurso o  Archivo del documento o material |
| Trazabilidad | Datko, C., & Carllinni, Y. (2014). Derivación de requisitos y construcción de trazabilidad entre artefactos del proceso de desarrollo. In XLIII Jornadas Argentinas de Informática e Investigación Operativa (43JAIIO)-XVII Concurso de Trabajos Estudiantiles (EST). | Otro | <https://webcache.googleusercontent.com/search?q=cache:sfsgC82GPXMJ:https://43jaiio.sadio.org.ar/proceedings/EST/3_759-2539-1-DR.pdf+&cd=1&hl=es&ct=clnk&gl=co> |

1. **GLOSARIO:**

Los siguientes conceptos que se presentan a continuación fueron tomados de Pérez Quintero (2019).

|  |  |
| --- | --- |
| TÉRMINO | SIGNIFICADO |
| Capacidad de expansión: | el grado con que se pueden aumentar el diseño arquitectónico, de datos o procedimental. |
| Complexión: | el grado con que sé a logrado la implementación total de una función. |
| Concisión: | lo compacto que resulta ser el programa en términos de líneas de código. |
| Consistencia: | el uso de un diseño uniforme y de técnicas de documentación, a través del proyecto de desarrollo del *software.* |
| Eficiencia de ejecución: | el rendimiento del funcionamiento de un programa. |
| Estandarización de comunicaciones: | el nivel de empleo de estándares de interfaces, protocolos y anchos de banda. |
| Estandarización de datos: | el empleo de estructuras y tipos de datos estándares a lo largo del programa. |
| Exactitud: | la exactitud de los cálculos y del control. |
| Facilidad de auditoría: | la facilidad con la que se puede justificar el cumplimiento de los estándares. |
| Formación: | el grado en que el *software* ayuda a los nuevos usuarios a manejar el sistema. |
| Generalidad: | la extensión de aplicación potencial de los componentes del programa. |
| Independencia del *hardware:* | el grado con que se desacopla el *software* del *hardware* donde opera. |
| Instrumentación: | el grado con que el programa vigila su propio funcionamiento e identifica los errores que suceden. |
| Modularidad: | la independencia funcional de componentes de programa. |
| Operatividad: | la facilidad de operación de un programa. |
| Tolerancia al error: | el deterioro causado cuando un programa descubre un error. |
| Trazabilidad: | la capacidad de alcanzar una representación del diseño o un componente real del programa, hasta los requisitos. |

1. **REFERENCIAS BIBLIOGRÁFICAS:**

Acosta-Hoyos, L. E. (1972). Guía práctica para la investigación y redacción de informes (Vol. 146). Editorial Paidós.

Ayabaca, L. P., & Bernal, S. M. (2017). Verificación y Validación de Software Software Verification and Validation. Revista Killkana Técnica. Vol, 1(3).

Bichachi, D. S. (2004). El uso de las listas de chequeo (CHECK-LIST) como herramienta para controlar la calidad de las leyes. Recuperado de: [Enlace del recurso](https://claudiabernazza.ar/ssgp/html/pdf/check_list.pdf).

Botta, M., de Roig, N. F., & Izaguirre, I. (2002). Tesis, monografías e informes: nuevas normas y técnicas de investigación y redacción. Biblos.

Dorrego, E. (1994). Modelo para la producción y evaluación formativa de medios instruccionales, aplicado al video y al software. Revista de Tecnología Educativa, 12(3), 313-327.

Espinoza Freire, E. E. (2018). Las variables y su operacionalización en la investigación educativa. Parte I. Conrado, 14, 39-49.

Espinoza Montes, C. (2010). Metodología de investigación tecnológica.

Hernandez, F., Roldán, M. L., Vegetti, M. M., Gonnet, S. M., & Leone, H. P. (2015). TracEDaaS: Captura y Trazabilidad de Artefactos del Proceso de Diseño.

López Bartolomé, C. (2014). El desarrollo de software Open Source analizado desde dentro (Bachelor's thesis).

Pájaro, D. (2002). La formulación de hipótesis. Cinta de Moebio. Revista de Epistemología de Ciencias Sociales, (15).

Pereira, B., Ayaach, F., Quintero, H., Granadillo, I., & Bustamante, J. (2012). Métricas de calidad de software. Calidad Del Software.

Perez Quintero, C. I. (2019). Implementación del modelo Mccall para medir la calidad del software Siplan (Doctoral dissertation).

Romero Alvarado, W. A. (2018). Los instrumentos de verificación y su incidencia en el control de los procesos productivos de la paletera Armijos.

Tabares, M. S., Barrera, A. F., Arroyave, J. D., & Pineda, J. D. (2007). Un método para la trazabilidad de requisitos en el proceso unificado de desarrollo. Revista EIA, (8), 69-82.

Torres, M., Salazar, F. G., & Paz, K. (2019). Métodos de recolección de datos para una investigación.

1. **CONTROL DEL DOCUMENTO**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Nombre | Cargo | Dependencia | Fecha |
| Autor (es) | Mario Fernando Meneses Calvache | Experto Temático | Regional Cauca - Centro de Teleinformática y Producción Industrial | Septiembre 2021 |
| Ana Catalina Córdoba Sus | Evaluadora instruccional | Regional Antioquia - Centro de Servicios de Salud | Abril 2024 |
| Olga Constanza Bermúdez Jaimes | Responsable Línea de Producción Antioquia | Regional Antioquia - Centro de Servicios de Salud | Abril 2024 |

1. **CONTROL DE CAMBIOS**

**(Diligenciar únicamente si realiza ajustes a la Unidad Temática)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Nombre | Cargo | Dependencia | Fecha | Razón del Cambio |
| Autor (es) |  |  |  |  |  |