

PROGRAMA DE ESTUDIOS

DESARROLLO DE SISTEMAS DE INFORMACIÓN

**ANÁLISIS Y DISEÑO
DE SISTEMAS**

Tema:

**CONCEPTOS FUNDAMENTALES DE
INGENIERÍA DE SOFTWARE**

CONCEPTOS FUNDAMENTALES DE INGENIERÍA DE SOFTWARE

Principios de la ingeniería de software

Una serie de principios, dirige la práctica de la ingeniería de software, que son fundamentales y además contribuyen a la aplicación de los procedimientos del software.

En la práctica, los principios determinan los valores y reglamento que servirán para el momento de investigar o estudiar un problema; luego se buscará y diseñará la solución, se reparará, se probará y al estar listo, se distribuirá entre la comunidad de usuarios.

Existen una serie de principios fundamentales que facilitan los procedimientos y la práctica del software, como por ejemplo:

Agregar valor para los usuarios finales

Mantenerlo sencillo

Fijar la visión (producto-proyecto)

Reconocer que los consumidores deben comprender lo que usted ofrece

Abrirse al futuro

Planear la reutilización

¡pensar!

Sin tomar en cuenta que el modelo sea lineal o iterativo, prescriptivo o ágil, el uso de la estructura general, puede ser aplicada a todos los diseños de proceso.

Principio 1. Ser dinámico.

Las características más importantes del trabajo, debe ser enfático en la economía de acción que consiste en enfocarse en que lo técnico sea lo más sencillo posible, que los productos a elaborar sean precisos y que las decisiones puedan ser tomadas localmente.

Principio 2. Concentrarse en la calidad de cada etapa.

La calidad del producto debe ser el centro focal de la condición de salida para toda la actividad del procedimiento del trabajo que se ha creado.

Principio 3. Estar listo para adaptar.

Ajustar el enfoque a las limitaciones que imponen los problemas, los usuarios y el proyecto en sí.

Principio 4. Conformar un equipo eficaz.

El procedimiento y usos de la ingeniería de software son importantes, pero el objetivo principal son los usuarios. Se debe formar un equipo en donde impere la confianza.

Principio 5. Constituir los mecanismos necesarios para la comunicación y coordinación.

Principios que dirigen el proceso.

Principio 6. Administrar el cambio

El planteamiento puede variar entre formal o informal, pero deben determinarse los mecanismos para aplicar la forma en la que los cambios se solicitan, evalúan, aprueban e implementan.

Principio 7. Calcular el riesgo.

Son muchos los detalles que pueden fallar en el desarrollo de software por lo que se hace imprescindible tener planes de emergencia.

Principio 8. Elaborar productos del trabajo que sumen valor para otros

Todo producto del trabajo que dé como resultado, de parte de la práctica de ingeniería de software pasará a alguien más. La lista de las funciones y características exigidas se dará a la persona que desarrollará un diseño, el diseño pasará a quienes creen un código y así sucesivamente. Asegúrese de que el producto del trabajo imparte la información necesaria y precisa.

Herramientas de Ingeniería de software:

CASE (Ingeniería De Software Asistida Por Computador)

Incorporan un conjunto de programas que permiten mejorar la calidad de un producto, ofreciendo apoyo constantemente a los analistas, ingenieros de software y desarrolladores.

CASE es la adaptación de procesos y tecnología que dan rendimiento a los programas, a través de otros, procesos y su correspondiente documentación.

Recientemente las herramientas para la administración de requisitos de software se condicionaban solo los editores de texto, quienes la realizaban haciendo de ella una labor aburrida y hasta confusa. En la actualidad, contamos con muchísimas opciones, como las que se mencionan a continuación:

IRQA

IRQA es la que posibilita y determina la relación entre el cliente, el distribuidor y los diferentes miembros del equipo que está desarrollando el proyecto.

Facilita la adquisición, constitución y estudio de las condiciones, así como la determinación de la solución mediante el apoyo metodológico adaptable a cada cliente.

CONTROLA

Herramienta de soporte de proceso de ingeniería de software en pequeñas empresas. Fue creada debido al desarrollo del mercado y a la creación grandes y pequeñas empresas, que requerían una herramienta para la evolución de sus proyectos.

Ofrece procedimientos importantes como lo son: Administración de requisitos, gestión de casos de uso, dirección de casos de prueba y error, planificación de liberaciones, administración de implementaciones, control de dependencia entre Implementaciones, matriz de rastreabilidad y rastreabilidad de los requisitos.

OSRMT (Open Source Requirements Management Tool)

Herramienta libre para la administración de requisitos, que tiene unas características propias y las principales son: trabaja en arquitectura cliente/servidor, evolucionada bajo Java; la versión 1.3 trae un módulo para manejar la trazabilidad y lo introduce para el control de cambios; de igual manera, produce la documentación de los requisitos tratados.

FUENTE:

- <https://loopgk.com/2021/09/15/fundamentos-de-la-ingenieria-en-software/>
- <https://togrowagencia.com/conceptos-basicos-del-desarrollo-de-software/>
- <https://institucionbadra.org/fundamentos-de-la-ingenieria-de-software/>
- <https://www.tamps.cinvestav.mx/~ertello/swe/sesion01.pdf>
- BURCH, John; GRUDNISKY, Gary. "Diseño de Sistemas de Información", Grupo Noriega editores.
- SENN, James A. "Análisis y diseño de sistemas de información", 2da. ed., McGraw-Hill.



INSTITUTO
KHIPU