

TEMA: RESOLUCIÓN DE PREGUNTAS INGENIERIA DE SOFTWARE

- 1. Haciendo referencia a la distribución de costos del software indicados en la Sección 1.1.6, explique por qué es apropiado considerar que el software es más que programas que son ejecutados por los usuarios finales de un sistema.**

De acuerdo a la distribución de costos del software es que nos damos cuenta como es el funcionamiento interno del software, ya que éste es más que un conjunto de programas que son ejecutados por el usuario ya que el software incluye una serie tanto de programas como de procesos, los cuales son usados por usuarios. Estos procesos son divididos en diferentes modelos que nos ayudarán a comprender de una forma descriptiva y simplificada las actividades que se ejecutarán en dicho proceso.

- 2. ¿Cuáles son las diferencias entre el desarrollo de un producto de software genérico y el desarrollo de un software personalizado?**

Los ingenieros de software desarrollan productos de software que se venden a un cliente. Existen 2 tipos de software:

Productos genéricos. Son sistemas aislados que se venden al mercado abierto a cualquier cliente que le sea posible comprarlos. La organización que desarrolla el software controla su especificación

Productos personalizados. Son sistemas requeridos por un cliente en particular. El contratista de software desarrolla el software especialmente para ese cliente. Las especificaciones son desarrolladas y controladas por la organización que compra el software. Los desarrolladores de software deben trabajar con esas especificaciones.

- 3. ¿Cuáles son los cuatro atributos importantes que todos los productos de software deben tener? Sugiera otros cuatro atributos que pueden ser significativos.**

- **Mantenibilidad.**- el software debe cumplir las necesidades de cambio.
- **Eficiente.**- utilice los recursos necesarios para su funcionamiento adecuado
- **Amigable.**- tenga una interfaz amigable y de fácil acceso para el cliente.
- **Seguro.**- sea un software que cumpla con las políticas de seguridad que necesite el cliente
- **Otros:** confiabilidad y usabilidad.
- **Atributos sugeridos:** económico, reusable, adaptable, portable

- 4. ¿Cuál es la diferencia entre un modelo del proceso del software y un proceso del software? Sugiera dos formas en las que un modelo del proceso del software ayuda en la identificación de posibles mejoras del proceso.**

Un proceso del software es un conjunto de actividades y resultados asociados que producen un producto de software, como especificaciones, desarrollo, validación, y evolución del software, mientras que el modelo de proceso de software es una descripción simplificada de un proceso del software que presenta una visión de ese proceso.

Modelo de flujo de trabajo: Muestra la secuencia de actividades en el proceso junto con sus entradas, salidas y dependencias.

Modelo de rolación: Representa los roles de las personas involucrada en el proceso del software y las actividades de las que son responsables

5. Explique por qué los costos de pruebas de software son particularmente altos para productos de software genéricos que se venden a un mercado amplio.

Ya que los productos genéricos se desarrollan para un mercado general por lo que deben ser sistemas robustos y por esta razón se deben realizar pruebas minuciosas antes de sacarlo al mercado por lo que el costo de pruebas y desarrollo del software son altos en comparación con los costos de las especificaciones.

6. Los métodos de la ingeniería del software se empezaron a utilizar cuando la tecnología CASE estuvo disponible para apoyarlos. Mencione cinco tipos de métodos de ayuda que proporcionen las herramientas CASE.

- a. Generador de códigos que automáticamente genera código fuente a partir del modelo del sistema.
- b. Módulos de análisis que verifican el modelo del sistema.
- c. Generadores de informes que ayudan a crear la documentación del sistema.
- d. Editor de diagramas de diseño
- e. Guías de procesos para los ingenieros de software.

7. Además de los retos de la heterogeneidad, la rápida entrega y la confianza, identifique otros problemas y retos que la ingeniería del software afrontará en el siglo xxi.

- El software no sea compatible con el hardware, sistemas operativos u otros programas.
- Interfaz y manejo simple en programas de funciones complejas.
- El hardware no fuera lo suficientemente potente y funcional para soportar el software.

8. Comente si los ingenieros profesionales deben atestiguar de la misma forma que los doctores o los abogados.

Claro que sí, ya que los ingenieros de software tienen responsabilidades profesionales y sociales por lo que la ingeniería de software se debe llevar a cabo dentro de un marco legal y social, por lo que los ingenieros de software deben comportarse de una forma ética y moral responsable

9. Para cada una de las cláusulas del Código de Ética de la ACM/IEEE que se muestra en la Figura 1.6, sugiera un ejemplo apropiado que ilustre esa cláusula.

Público: Software amigable para que sea de fácil utilización por parte de los usuarios por ejemplo un motor de búsqueda de internet como GOOGLE.

Cliente y Empleador: El software debe cumplir con los requerimientos del cliente, por ejemplo un software que maneje la nómina y pago a los empleados de una empresa.

Producto: Un claro ejemplo de ello es el sistema operativo Windows, que después de haber suministrado su software permite la actualización y descarga de paquetes que mejoran la funcionalidad y corrigen errores presentes en el producto.

Juicio: Se debe asumir su responsabilidad tanto de manera profesional como ética y moral.

Gestión: Administrar los tiempos en el desarrollo del producto.

Profesión: Se debe ser una persona responsable en el desarrollo y mantenimiento del producto.

Colegas: Sociedades y grupos de apoyo para el desarrollo de software.

Personal: La aparición de nuevas tecnologías de desarrollo de software hace que los profesionales de desarrollo se mantengan en continua actualización

10. Para contrarrestar al terrorismo, muchos países están planeando el desarrollo de sistemas informáticos que sigan la pista de un gran número de sus ciudadanos y de sus acciones. Desde luego, esto tiene implicaciones sobre la privacidad. Comente la ética de desarrollar este tipo de sistema.

Los ingenieros deben ser muy cuidadosos con la manipulación de la información porque esta es muy delicada por lo que debe ser manejada con responsabilidad y total discreción. Por lo tanto las partes involucradas en el desarrollo del software deben tener un acuerdo de confidencialidad en todos los aspectos de una empresa.

1. ¿Qué es la ingeniería de software?

Es un enfoque sistemático de principios y métodos de la ingeniería con el fin de diseñar o construir aplicaciones o programas que sean rentables, fiables y que funcionen de manera eficiente para el usuario.

2. ¿Cuál es la diferencia entre un programador y un Ingeniero de software?

Un programador está encargado de los detalles de codificación y modificación de los algoritmos o líneas de códigos, es decir, sobre un conjunto de instrucciones codificadas en un lenguaje de programación que ejecuta el hardware de un computador para realizar una determinada tarea.

Un ingeniero de software es el que está encargado del ciclo de vida de un software, detalles, documentación y la administración de un proyecto.

3. ¿Cuál es la diferencia entre un Ingeniero de Software y un Ingeniero de Sistemas?

El ingeniero de sistema está encargado de todo el sistema de cómputo y el ingeniero de software es el encargado en la creación de aplicaciones o programas, es decir el ciclo de vida del software.

4. ¿Cuál es la diferencia entre un Ingeniería de Software y la Computación?

Ingeniería de Software utiliza los principios de la ingeniería para el diseño e implementación del software

Computación es la disciplina se ocupa del estudio de sistemas de cómputo incluyendo procesos algorítmicos y principios tanto en el diseño de software y hardware.

5. ¿Cuáles son los productos obtenidos durante el proceso de la ingeniería de software?

Programas y documentos y datos que se producen como consecuencia de las actividades y tareas definidas por el proceso.

- Líneas de Códigos
- Documentación(Manual de Usuarios – Tutoriales de Instalación – Manuales Técnicos)
- Programas o Aplicaciones
- Diseños
- Diccionario de Datos

6. ¿Qué es el software?

El software son aplicaciones o programas creados para empresas, clientes con el fin de desarrollarse para la venta en mercados.

El software puede ser genérico y específico.

7. ¿Qué es un proceso de software?

También conocido como ciclo de vida de desarrollo de software es aplicada para el desarrollo de un producto de software. Un proceso de software es un conjunto de actividades, acciones y tareas que se ejecutan cuando va a crearse algún producto del trabajo.

8. ¿Por qué es importante el software?

El software es importante para cualquier sistema informático o basado en informática, puesto que sin él, este no funcionaría. El software es quien da las órdenes, quien indica que debe hacer a cada máquina con sus elementos, cuando y como. Un computador sin software sería simplemente un conjunto de chips, cables, periféricos e interruptores totalmente inerte y sin función alguna.

9. ¿Cuáles son los objetivos de la ingeniería de software?

- Diseñar aplicaciones informáticas que se ajusten a las necesidades del usuario.
- Organizar la realización de pruebas que verifiquen el correcto funcionamiento de los programas y que se ajustan a los requisitos de análisis y diseño.
- Introducir procedimientos de calidad en los sistemas, evaluando métricas e indicadores y controlando la calidad del software producido.
- Estimar los costes de un proyecto y determinar los tiempos de desarrollo
- Organizar y supervisar el trabajo de su equipo de los técnicos de mantenimiento y los ingenieros de sistemas y redes.
- Intervenir en todas las fases del ciclo de vida de un productodirigiendo equipos de trabajo de desarrollo software.

10. ¿Cuál es la clasificación del software, y dé un ejemplo de cada uno?

El software se clasifica en:

1) **Software de sistema**

Un software de aplicación es un conjunto de programas imprescindibles para el funcionamiento del sistema.

También conocido como software de base, y su función consiste en control e interacción con el sistema operativo, además de dar control del hardware y soporte a programas diversos

Ejemplo: bibliotecas (OpenGL para la aceleración gráfica), PNG para el sistema gráfico, la velocidad del disco duro, sistemas operativos (Windows 8)

2) **Software de Programación**

El software de programación es utilizado para crear el software de sistema y el de aplicación. Esto se basa según el nivel de complejidad que presenta y el objetivo para el cual fue diseñado, se distinguen dos tipos de lenguajes de programación: de bajo nivel y de alto nivel. Programación de bajo nivel se orienta a una determinada máquina o clase de máquina, el alto nivel es destinado a la resolución de problemas específicos.

Ejemplo: Editores de texto (Pascal, Edit de MS-DOS) Compiladores (C, C++, Visual Basic, Fortran, Cobol) Intérpretes (Ensamblador, Java) Enlazadores (Oracle, Circle) Depuradores (GNU Debugger (gdb), SoftICE) Entornos de Desarrollo Integrados ó IDE (NetBean, Visual Basic

3) **Software de Aplicación**

El software de aplicación es el que se utiliza en una computadora para realizar las tareas que son útiles para los usuarios. Este incluye programas que se ejecutan para realizar trabajos relacionados con áreas específicas, o aplicaciones.

En el software de aplicación se encuentra una característica importante para poder utilizarlo, es fundamental que exista un software de sistema que permita su ejecución.

Dentro del software de aplicación podemos distinguir tres tipos:

- Juegos
- Utilitarios
- De gestión

Ejemplo: procesadores de texto (Microsoft Word), hojas de cálculo (Microsoft Excel), gestores de base de datos (Microsoft Access), programas graficadores (Adobe PhotoShop), gestores de comunicaciones (Microsoft mail), etc.

11. Describa cada una de las actividades del método clásico del ciclo de vida del software

1) Investigación Preliminar

La solicitud para recibir ayuda de un sistema de información puede originarse por varias razones: sin importar cuales sean estas, el proceso se inicia siempre con la petición de una persona(administrador, empleado o especialista en sistemas).

2) Determinación de los requisitos del sistema

El aspecto fundamental del análisis de sistemas es comprender todas las facetas importantes de la parte de la empresa que se encuentra bajo estudio. Los analistas, al trabajar con los empleados y administradores, deben estudiar los procesos de una empresa para dar respuesta a las preguntas clave

¿Qué es lo que hace?

¿Cómo se hace?

¿Con que frecuencia se presenta?

3) Diseño del sistema (diseño lógico)

El diseño de un sistema de información produce los detalles que establece la forma en la que el sistema cumplirá con los requerimientos identificados durante la fase del análisis. Esta fase se denomina como diseño lógico en contraste con la de desarrollo del software, a la que se denomina diseño físico

4) Desarrollo de software (Diseño físico)

Los encargados de desarrollar software son aquellos que pueden instalar el software comprados a terceros o realizar los programas diseñados a la satisfacción del usuario. El costo de un software depende a la factibilidad, disponibilidad de programadores y las líneas de código.

5) Prueba de sistemas

El sistema se lo ejecuta de manera elemental es decir en modo prueba, para asegurarse que el software no tenga fallas y este de acuerdo a las especificaciones del usuario. Se realizan conjunto de pruebas para obtener resultados favorables.

6) Implementación y Evaluación

En este proceso lo que se realiza es llevar los estudios de los pro y contras de establecer un sistema nuevo, generalmente se manejan los dos sistemas a la par (el antiguo y el a implementar), nunca se desligan, la idea de trabajar simultáneamente es que se pueda ver los problemas que puedan surgir, se crea una especie de ambiente para que el sistema nuevo sea aprobado por los usuarios, en este momento se verá si es amigable, interfaz, y eficacia.

Los desarrolladores dejan siempre abierta la posibilidad de mejorar lo ya planteado, todo varía de acuerdo a las necesidades de la empresa y el diseño del software. La evolución de un

sistema se lleva a cabo para identificar puntos débiles y fuertes, la evaluación ocurre a lo largo de cualquiera de las siguientes:

- Evaluación operacional
- Impacto organizacional
- Opinión de los administradores
- Desempeño del desarrollo

Mientras existan mejoras de software los sistemas debe evolucionar, de esta manera se actualizan las herramientas y no se hacen obsoletas las aplicaciones.