

TALLER FUNDAMENTOS 14538

1.- La ingeniería de software combina

- A. Diseño, codificación y pruebas de aceptación del usuario
- B. Codificación y pruebas
- C. Técnicas de ingeniería con prácticas de desarrollo de software
- D. Ninguna de las anteriores es cierta

Respuesta

La ingeniería de software no se limita únicamente a la codificación y las pruebas, sino que abarca un enfoque más amplio que incluye técnicas de ingeniería como la gestión de requerimientos, el diseño arquitectónico, la planificación del proyecto, entre otros aspectos fundamentales para el desarrollo de software de calidad. Por lo tanto, la respuesta correcta es aquella que reconoce la combinación de técnicas de ingeniería con prácticas específicas de desarrollo de software.

3.- Que es un proceso de software

- A. Representa los roles de las persona involucradas en este proceso y las actividades de las que son responsables
- B. Representa el proceso como un conjunto de actividades cada una de las cuales realiza alguna transformación en los datos
- C. Conjunto de actividades cuya meta es el desarrollo u optimización de software
- D. Muestra la secuencia de actividades en el proceso junto con las entradas, salidas y dependencias.

Respuesta

Un proceso de software no solo describe las actividades que se realizan durante el desarrollo de software, sino que también especifica cómo estas actividades se relacionan entre sí, qué entradas y salidas tienen, y cuáles son las dependencias entre las distintas etapas. Es una representación detallada y estructurada de cómo se lleva a cabo el desarrollo de software desde el inicio hasta la entrega.

4. Las notaciones que permiten representar la vía estática o la estructura del sistema son

- A. Diagramas de secuencia
- B. Diagramas de actividad
- C. Diagramas de casos de uso
- D. Diagramas de clases y objetos

Respuesta

Los diagramas de clases y objetos son utilizados específicamente para representar la estructura estática de un sistema, mostrando las clases, sus atributos, métodos y las relaciones entre las clases. Esto proporciona una vista detallada de la estructura del sistema sin entrar en la dinámica del comportamiento como lo hacen otros tipos de diagramas como los de secuencia o actividad.

5. Se utilizan para representar modelos conceptuales de datos almacenados en repositorios de información esta información corresponde a Fundamentos de Ingeniería de Software un enfoque práctico

A. Diagramas entidad relación

B. Diagramas clases y objetos

C. Diagramas de estados

D. Diagramas de secuencia

Respuesta

Los diagramas entidad-relación son herramientas fundamentales en el modelado de datos, utilizados para representar las entidades dentro de un sistema, sus atributos y las relaciones entre ellas. Este tipo de diagrama es crucial para el diseño de bases de datos y la gestión de la información dentro de un sistema de software.

6. La aplicación de estándares de desarrollo externos o internos durante la construcción ayuda a lograr los objetivos del proyecto, en lo referente a: eficiencia, calidad y costo.

¿Cuáles son los estándares de construcción que ayudan a lograr los objetivos de un proyecto

A. Estándares para formatos y contenidos de documentos (nombres, diseño y sangría).

B. Estándares para lenguajes como Java y C++ (lenguajes/ides programación).

C. Estándares de buenas prácticas de codificación (métodos de comunicación).

D. Todas las anteriores son ciertas

Respuesta

La aplicación de estándares para el formato y contenido de documentos, para lenguajes de programación, y de buenas prácticas de codificación son todos cruciales para asegurar la eficiencia, calidad y control de costos en un proyecto de desarrollo de software. Estos estándares promueven la consistencia, facilitan la colaboración entre equipos, y ayudan a evitar errores comunes que podrían afectar negativamente al proyecto.

Une y selecciona las respuestas correctas

PROGRAMACIÓN	Ing Software
a. Solo un desarrollador	1. Equipos de desarrollo
b. Aplicaciones de juguete	2. Sistema roles
c. Uno o pocos involucrados	3. Sistema complejos
d. Mantenimiento mínimo	4. Mantenimiento es + 60% costos de desarrollo

Opc. A a1, b2, c3, d4 Respuesta 1

OK(X) NOK()

Opc. B. a4, b2, c3, d1 Respuesta 2

OK() NOK(X)

Opc D. a3, b2, c1, d4 Respuesta 3

OK() NOK(X)

Opc. C. a2, b1, c3, d4 Respuesta 4

OK(X) NOK()

- a. Solo un desarrollador** va con **2. Sistema de roles** porque un solo desarrollador no necesita un sistema de roles complejo.
- b. Aplicaciones de juguete** va con **1. Equipos de desarrollo** porque las aplicaciones simples no requieren un equipo de desarrollo grande.
- c. Uno o pocos involucrados** va con **3. Sistemas complejos** porque los sistemas complejos generalmente necesitan la colaboración de varias personas.
- d. Mantenimiento mínimo** va con **4. Mantenimiento es + 60% costos de desarrollo** porque un sistema con mantenimiento mínimo no debería tener costos de mantenimiento tan altos.

2. Seleccione las definiciones de diagramas de UML, con el término correspondiente.

1. Muestran el flujo de control de una actividad. Se puede usar para representar actividades concurrentes.	Diagramas de actividad.
2. Muestran las interacciones entre un grupo de objetos, con énfasis en el orden de tiempo de los mensajes que se pasan entre los objetos	Diagramas de secuencia.
3. Muestran el flujo de control de estado a estado y cómo el comportamiento de un componente cambia en función de su estado actual en una máquina de estados	Diagramas de transición de estados.
4. Muestran las interacciones que ocurren entre un grupo de objetos; se hace hincapié en los objetos, sus enlaces, y los mensajes que intercambian en estos enlaces	Diagramas de comunicación.

4. Cómo se construye un buen sistema de información (SI) considerando que el punto de partida es:

- I. Utilizar herramientas de desarrollo como medio para alcanzar un producto de calidad
- II. La definición de requisitos claros es una parte del proceso, pero no es relevante
- III. Las pruebas y validaciones no son indispensables par la construcción del producto
- IV. Utilizar un proceso definido con fases claras, donde cada una de estas genera un producto final.

Se utiliza esta ya que utilizar un proceso definido con fases claras y productos finales en cada una de ellas asegura que el desarrollo del sistema de información sea eficiente, controlado y de alta calidad, lo que es esencial para el éxito de cualquier proyecto de SI.

defensa de perfil 15 mins max

tema de proyecto

objetivos

alcance

qué resultados pretende

2 historias de usuario

matriz de historias de usuario(última versión) basada en el mapa de procesos