

Parcial 1 2021-11-16

Pregunta 1

Finalizado

Puntúa como
1,00

🚩 Marcar
pregunta

Describe exhaustivamente tres de los desafíos de la Ingeniería del Software mencionados en la materia.

A) Calidad: aunque no hay una definición fija, la calidad se basa en distintos aspectos:

- * Funcionalidad: cumplimiento de las necesidades establecidas.
- * Eficiencia: desempeño apropiado relativo a la cantidad de recursos usados.
- * Mantenibilidad: capacidad de ser modificado para mejoras y correcciones.
- * Portabilidad: adaptable a distintos entornos sin aplicar acciones de más.
- * Usabilidad: capacidad de ser comprendido, aprendido y usado.
- * Confiabilidad: realización de lo requerido dentro de las condiciones y los tiempos establecidos.

2) Productividad: Tanto una solución que demora mucho tiempo como una que entrega un software barato y de baja calidad son inaceptables. El costo del software es la mano de obra, medido en persona/mes (PM). Como no hay una productividad estándar, es una medida razonable suponer una productividad de 1000 LOC por persona por mes.

C) Consistencia: sucesiva producción de sistemas de alta calidad y productividad.

Comentario:

10

de dónde sale lo de 1000LOC?

Pregunta 2

Finalizado

Puntúa como
1,00

🚩 Marcar
pregunta

¿Cuáles son las actividades básicas del proceso de requerimientos?

Explíquelas.

El proceso de requerimientos típicamente consiste de tres tareas básicas: el problema o análisis de los requerimientos, la especificación de los requerimientos y la validación de los requerimientos.

El **análisis del problema** consiste en lograr una buena comprensión de las necesidades, requerimientos, y restricciones del software, mediante entrevistas con el cliente y usuarios, lectura de manuales, estudio del sistema actual, etc. Durante el análisis se busca asegurar la completitud y consistencia del diseño, sin entrar en detalles internos del mismo. Existen distintos enfoques sobre como analizar a los problemas y las relaciones entre ellos.

El objetivo de la **especificación de los requerimientos** es la producción de la SRS. Se diferencia de la etapa de análisis en que se enfoca en el comportamiento externo del sistema, mientras que la anterior se concentra en la estructura del problema. La SRS no es una formalización del modelo, porque en ella se especifican aspectos como el tratamiento de errores, requerimientos en el desempeño, restricciones en el diseño, etc. Una SRS debe ser comprensible por todas las partes involucradas. Para esto, los lenguajes de especificación deben facilitar su escritura y aprendizaje. Además, se debe seguir una estructura ordenada, organizando el documento en secciones y subsecciones bien definidas.

En la etapa de **validación** se realiza la corrección de errores. Para esto, el autor de la SRS, el cliente, usuarios y desarrolladores revisan la SRS en busca de defectos a corregir. Se pueden usar herramientas como listas de control (preguntas generales que pueden descubrir fallas). Además, existen otras que son útiles para evaluar modelados y análisis de especificaciones (autómatas, Alloy, etc.). En esta fase también se utilizan métricas para poder estimar costos, tiempos y el esfuerzo que demandara. El principal factor es el tamaño, el cual solo puede ser estimado.

Comentario:

A) ¿Qué es un estilo arquitectónico?

B) Describa exhaustivamente uno de los estilos arquitectónicos vistos en la asignatura.

Un estilo arquitectónico define una familia de arquitecturas que son útiles para solucionar una determinada clase de problemas. Los estilos pueden proveer ideas para crear una vista de arquitectura para el problema que tengamos. Los estilos también pueden combinarse para formar vistas más ricas o mejores.

Cliente-servidor:

las componentes son clientes o servidores. Los clientes inician la comunicación, y solo se relacionan con el servidor (no con otros clientes).

Los conectores son las solicitudes y las respuestas. Es un estilo multi-nivel:

- * Nivel de cliente: contiene a los clientes.
- * Nivel intermedio: contiene las reglas del servicio.
- * Nivel de base de datos: donde reside la información.

Pregunta 4

Correcta

Puntúa 1,00
sobre 1,00

🚩 Marcar
pregunta

Marque la opción verdadera, según lo enseñado en clases.

Seleccione una:

- A. El principio abierto-cerrado implica el principio de Liskov.
- B. El principio de Liskov se cumple si o si el principio abierto-cerrado se cumple.
- C. El principio de Liskov implica el principio abierto-cerrado.

Respuesta correcta

La respuesta correcta es: El principio de Liskov implica el principio abierto-cerrado.

Comentario:

10

Pregunta 5

Finalizado

Puntúa como
1,00

🚩 Marcar
pregunta

A) ¿Qué son y para qué sirven las métricas en el diseño orientado a objetos?

B) Elija y describa dos de ellas.

A) Las métricas son distintas formas con las que se puede evaluar los diseños, sirven para proveer una evaluación cuantitativa del diseño.

B) Cantidad de Hijos (NOC):

* Cantidad de subclases inmediatas a C, mientras más NOC tenga C entonces mayor será el reúso.

* Nos da una idea del reúso de una clase además de cuanto influencia tiene C sobre otros elementos del diseño: mientras más grande la influencia de C mayor será la importancia de la corrección de diseño de esta clase.

Métodos pesados por clases (WMC):

La complejidad de una clase está ligada a la cantidad de métodos que esta posea y la complejidad de dichos métodos.

Sean M_i los métodos de la clase C con $1 \leq i \leq n$ y sea $C(M_i)$ la complejidad del método i entonces

$WMC =$ a la sumatoria de la complejidad de cada método. Si el resultado es alto entonces la clase será propensa a errores.

La comparación de si es alto o no debe realizarse comparando los WMC de otras clases.