

Resumen y 75 Preguntas del capítulo de Diseño de Sistemas

Resumen amplio del capítulo:

Este capítulo profundiza en la fase de diseño en el ciclo de vida del desarrollo de sistemas, posterior al análisis de sistemas. El diseño de sistemas se centra en transformar el "qué debe hacer" el sistema en "cómo debe hacerlo", produciendo especificaciones detalladas que sirven como base para la construcción.

Se destacan dos enfoques principales:

- Desarrollo propio ("construcción"): crear un sistema desde cero dentro de la organización.
- Adquisición ("compra"): adaptar software comercial existente.

El diseño de sistemas abarca varios métodos:

- Diseño estructurado moderno
- Ingeniería de la información (IE)
- Elaboración de prototipos
- Desarrollo conjunto de aplicaciones (JAD)
- Desarrollo rápido de aplicaciones (RAD)
- Diseño orientado a objetos (OOD)

El proceso incluye tareas como:

1. Diseño de la arquitectura de aplicaciones.
2. Diseño de bases de datos.
3. Diseño de la interfaz del sistema.
4. Consolidación de especificaciones.
5. Actualización del plan del proyecto.

En adquisiciones de software comercial, se agregan fases como investigación, solicitud y evaluación de propuestas, validación y selección de proveedores.

Se enfatiza que el sistema diseñado debe integrarse con sistemas existentes. Se describen arquitecturas distribuidas (cliente/servidor, web, etc.) y su impacto en la toma de decisiones.

Preguntas y respuestas (1-75):

1. ¿Qué es el diseño de sistemas de información?

Es la especificación detallada de una solución computarizada basada en el análisis de sistemas.

2. ¿Cuál es la diferencia entre análisis y diseño de sistemas?

El análisis identifica el problema de negocio, mientras que el diseño especifica cómo solucionarlo técnicamente.

3. ¿Qué enfoque se utiliza cuando el sistema se desarrolla internamente?

Desarrollo o 'construcción'.

4. ¿Qué enfoque se utiliza cuando se compra un software existente?

Adquisición o 'compra'.

5. ¿Qué se diseña en la arquitectura de la aplicación?

La distribución de datos, procesos e interfaces en redes.

6. ¿Cuál es el objetivo principal del diseño de sistemas?

Crear una solución técnica detallada que satisfaga los requerimientos del usuario final.

7. ¿Qué elementos forman parte de la arquitectura de aplicaciones?

Datos, procesos, interfaces y tecnología de red.

8. ¿Qué es el diseño estructurado moderno?

Técnica orientada a procesos que organiza el sistema en módulos jerárquicos.

9. ¿Qué se busca con la cohesión de módulos?

Que cada módulo realice una sola función específica.

10. ¿Qué se busca con el acoplamiento bajo entre módulos?

Minimizar la dependencia mutua entre ellos.

11. ¿Qué es la Ingeniería de la Información (IE)?

Método basado en modelos y centrado en datos.

12. ¿Qué es la elaboración de prototipos?

Creación iterativa de versiones funcionales preliminares del sistema.

13. ¿Qué ventaja tienen los prototipos?

Permiten detectar errores y redefinir requerimientos tempranamente.

14. ¿Qué riesgo implica el uso excesivo de prototipos?

La omisión de especificaciones formales y descontrol del alcance.

15. ¿Qué es el desarrollo conjunto de aplicaciones (JAD)?

Método participativo que integra usuarios y diseñadores en talleres colaborativos.

16. ¿Qué es el desarrollo rápido de aplicaciones (RAD)?

Combinación de prototipos, JAD y técnicas estructuradas para acelerar proyectos.

17. ¿Qué es el diseño orientado a objetos (OOD)?

Técnica que integra datos y procesos en entidades llamadas objetos.

18. ¿Qué define la arquitectura de la aplicación?

Las tecnologías a utilizar para implementar el sistema.

19. ¿Qué herramientas documentan la arquitectura de aplicaciones?

Diagramas de flujo físico de datos (DFD físicos).

20. ¿Qué son los DFD físicos?

Diagramas que representan las decisiones técnicas y humanas del diseño.

21. ¿Qué tipos de procesos existen en los DFD físicos?

Procesos automatizados y procesos manuales.

22. ¿Qué representan los flujos de datos físicos?

Transferencias de datos entre procesos, almacenamientos y agentes externos.

23. ¿Qué representan los almacenamientos de datos físicos?

Bases de datos, archivos, medios de respaldo o archivos manuales.

24. ¿Qué diferencia hay entre agente externo lógico y físico?

No hay diferencia; los agentes externos no cambian entre DFD lógicos y físicos.

25. ¿Qué tipo de diseño es necesario cuando se compran paquetes de software?

El diseño arquitectónico, de interfaz y de integración con sistemas existentes.

26. ¿Qué paso sigue después de diseñar la arquitectura de la aplicación?

Diseñar la base de datos.

27. ¿Qué implica el diseño de la base de datos?

Determinar la estructura lógica, modelo de datos y relaciones necesarias.

28. ¿Qué otra tarea sigue después del diseño de la base de datos?

Diseñar la interfaz del sistema.

29. ¿Qué abarca la interfaz del sistema?

Pantallas, entradas, salidas y diálogos con el usuario.

30. ¿Qué es un prototipo de interfaz?

Representación gráfica funcional de la interacción usuario-sistema.

31. ¿Qué documento contiene todas las especificaciones del diseño?

El paquete de diseño del sistema.

32. ¿Qué incluye el paquete de diseño?

Diseños técnicos, reglas de negocio, arquitectura, interfaces y bases de datos.

33. ¿Qué se hace al final del diseño de sistemas?

Actualizar el plan del proyecto.

34. ¿Qué puede implicar actualizar el plan del proyecto?

Ajustes en tiempos, recursos y costos.

35. ¿Qué se debe investigar al comprar software comercial?

Productos disponibles, proveedores y funcionalidades.

36. ¿Qué es una solicitud de propuesta (RFP)?

Documento que pide a los proveedores especificaciones técnicas y precios.

37. ¿Qué debe incluir una RFP?

Requerimientos funcionales, técnicos, financieros y de soporte.

38. ¿Qué diferencia hay entre RFP y RFQ?

La RFP pide propuesta detallada; la RFQ pide solo cotización de precios.

39. ¿Por qué es importante validar las afirmaciones de los proveedores?

Para asegurar que el software cumple con los requerimientos.

40. ¿Qué se debe hacer antes de seleccionar un proveedor?

Jerarquizar las propuestas según criterios establecidos.

41. ¿Qué aspecto legal se formaliza con el proveedor elegido?

El contrato de adquisición o servicio.

42. ¿Qué es un sistema distribuido?

Sistema donde los componentes están ubicados en distintas ubicaciones físicas.

43. ¿Qué es la arquitectura cliente/servidor?

Modelo donde un cliente solicita servicios a un servidor central.

44. ¿Qué ventajas tiene la arquitectura cliente/servidor?

Escalabilidad, distribución de tareas, mejor mantenimiento.

45. ¿Qué desventaja puede tener cliente/servidor?

Mayor complejidad en la administración y seguridad.

46. ¿Qué es una arquitectura de internet?

Uso de tecnologías web para la distribución y acceso a sistemas.

47. ¿Qué ventaja tiene la arquitectura basada en internet?

Accesibilidad universal y menor costo de infraestructura.

48. ¿Qué es una base de datos distribuida?

Conjunto de bases de datos lógicamente integradas pero físicamente distribuidas.

49. ¿Qué beneficio da la distribución de bases de datos?

Mejora la disponibilidad, redundancia y rendimiento.

50. ¿Qué criterio se usa al dividir bases de datos distribuidas?

Por localización, funcionalidad y frecuencia de uso.

51. ¿Qué problemas puede tener una base distribuida?

Sincronización, seguridad y administración compleja.

52. ¿Qué impacto tiene la tecnología en el diseño de sistemas?

Determina posibilidades, limitaciones y costos del sistema.

53. ¿Qué se busca con la normalización de datos?

Eliminar redundancias y mejorar integridad.

54. ¿Qué es la cardinalidad en bases de datos?

La relación cuantitativa entre entidades (uno a uno, muchos a uno, etc.).

55. ¿Por qué es útil una entidad débil?

Porque depende de otra entidad para tener sentido y persistencia.

56. ¿Qué es una llave primaria?

Campo que identifica unívocamente cada registro en una tabla.

57. ¿Qué es una llave foránea?

Campo que enlaza una tabla con otra, refiriendo a su llave primaria.

58. ¿Qué importancia tiene el diseño de formularios?

Facilita la entrada precisa y eficiente de datos al sistema.

59. ¿Qué es un informe del sistema?

Una salida impresa o digital que resume datos del sistema para los usuarios.

60. ¿Qué elemento facilita la navegación en interfaces gráficas?

Los menús, íconos y botones de acción.

61. ¿Qué es un mockup?

Boceto visual que representa el diseño de una interfaz.

62. ¿Qué es una especificación técnica?

Documento que define cómo funcionará internamente un componente del sistema.

63. ¿Qué es una prueba piloto?

Implementación parcial del sistema para evaluar desempeño real.

64. ¿Qué es una retroalimentación del usuario?

Opinión o evaluación que da el usuario sobre el sistema o su diseño.

65. ¿Qué es escalabilidad en diseño de sistemas?

Capacidad del sistema para crecer o adaptarse a nuevas demandas.

66. ¿Qué diferencia a una arquitectura robusta?

Soporta fallos, alta disponibilidad y mantenimiento sencillo.

67. ¿Qué es interoperabilidad?

Capacidad de diferentes sistemas de comunicarse entre sí.

68. ¿Qué rol tiene la documentación técnica?

Guía a desarrolladores, operadores y usuarios del sistema.

69. ¿Qué técnica permite verificar procesos en tiempo real?

Simulación o modelado dinámico.

70. ¿Qué es mantenimiento preventivo?

Acciones periódicas para evitar fallos en el sistema.

71. ¿Qué es mantenimiento correctivo?

Corrección de errores o defectos detectados después de la entrega.

72. ¿Qué es mantenimiento adaptativo?

Modificación del sistema para hacerlo compatible con cambios externos.

73. ¿Qué función tiene el control de versiones?

Registrar y gestionar cambios en archivos del sistema.

74. ¿Por qué es importante la usabilidad?

Determina qué tan fácil y eficiente es el uso del sistema por humanos.

75. ¿Qué es un backlog en diseño ágil?

Lista priorizada de funcionalidades pendientes o por implementar.

76. ¿Qué es un entregable?

Producto concreto que debe completarse como parte de un proyecto.