

- Cada pregunta de la **parte múltiple opción** contestada correctamente tiene un valor de 2 puntos. Cada pregunta incorrecta de la múltiple opción resta 0,66 puntos.
- **El puntaje total del parcial es 50 puntos.**

Múltiple Opción

- 1) Entre los atributos de calidad más relevantes de un producto de software para la construcción de modelos UML se encuentran:
 - a) **Adecuación al uso, facilidad de uso, amigabilidad de la interfaz de usuario, tiempo de respuesta.**
 - b) Consumo de recursos, tiempo de respuesta, capacidad de procesamiento, facilidad de verificación.
 - c) Facilidad de mantenimiento, modularidad, correctitud, confiabilidad.
 - d) Facilidad de aprendizaje, facilidad de verificación, facilidad de mantenimiento, legibilidad del código.

- 2) Acerca de la técnica para obtener requisitos mediante entrevistas individuales:
 - a) **Es interactiva y flexible, y es muy rica la información que se obtiene.**
 - b) No es costosa y depende de las habilidades interpersonales.
 - c) Deben estar en todo proyecto y se deben hacer con todos los usuarios que van a utilizar el sistema.
 - d) Requiere seguir estrictamente la agenda de preguntas prevista.

- 3) Respecto a modelos de proceso de software:
 - a) El proceso en cascada al día de hoy prácticamente ha dejado de ser utilizado en la industria, habiendo sido sustituido por otros modelos como el de Especificación Operacional.
 - b) **Una de las principales ventajas de un proceso en fases con liberaciones parciales en producción es que permite acortar el plazo para que cliente y usuarios puedan contar con resultados que les sean de utilidad.**
 - c) Una de las principales ventajas de un proceso en fases con evaluaciones internas parciales es que permite reducir el plazo de salida al mercado.
 - d) Todas las anteriores son correctas.

- 4) Usted es responsable de la arquitectura de software de un sistema:
 - a) Deberá incluir los subsistemas que componen el sistema, las interfaces e interacción entre ellos
 - b) (a) y la definición de la arquitectura la realiza una única persona para evitar costos innecesarios
 - c) **(a) y realizando la arquitectura de software se toman decisiones de diseño de forma temprana**
 - d) (a) y no es necesario tener en cuenta los requerimientos no funcionales ya que no dependen del estilo arquitectónico elegido

- 5) Supongamos que tenemos un sistema del cual se requiere poner en producción una parte de las funcionalidades a mitad del proyecto y otra parte al final, y se ha decidido utilizar el modelo de proceso en fases con liberaciones parciales en producción.
Indique cuál de las afirmaciones es correcta:
 - a) La arquitectura del producto no podrá estar totalmente definida hasta luego de la segunda mitad de la segunda fase del proyecto (luego de haber puesto en producción la primer liberación).
 - b) **Que la arquitectura definida en la primera fase no soporte o no pueda ser extendida para soportar los requisitos del sistema final debe ser un riesgo a identificar, el cual hay que gestionar durante la primer y segunda fase del proyecto.**
 - c) Los requisitos de las funcionalidades de la segunda fase se deben relevar luego de haber puesto en producción la primera. Debido a esto, la arquitectura definida en la primer fase debe ser escalable.
 - d) Validar la arquitectura del sistema es una actividad que debe realizarse únicamente en la segunda fase del proyecto por temas de costo/beneficio.

- 6) Cambiar la Arquitectura de Software de un sistema ya construido:
 - a) Es relativamente sencillo si se usó el estilo en capas para la misma.
 - b) Podría requerir mucho esfuerzo.
 - c) **(b) por lo que es conveniente realizar evaluaciones de la arquitectura de un sistema durante su definición (antes de implementarlo completamente).**
 - d) Es algo que se debe realizar periódicamente. Aproximadamente cada 8 meses se producen cambios tecnológicos que obligan a cambiar la arquitectura del sistema.

- 7) Dadas las siguientes afirmaciones sobre el diseño de la interfaz con el usuario indique cuál opción es correcta.
1. Se deben utilizar términos que le son familiares al usuario.
 2. Por cuestiones de costo/beneficio no se deben considerar la diversidad de "tipos de usuarios" que puede tener el sistema. Esto sería incluso imposible de realizar.
 3. Existen 2 aspectos que son clave al diseñar la interfaz de usuario: la forma de presentar la información al usuario y la forma en la cual el usuario interactúa con el sistema.
 4. Los mensajes de error no es algo que deba preocupar al desarrollador de interfaces de usuario ya que un sistema no depende de los mismos para funcionar correctamente.
- a) Todas son correctas.
 - b) 1, 2 y 3 son correctas. 4 no lo es.
 - c) 1, 3 y 4 son correctas. 2 no lo es
 - d) 1 y 3 son correctas. 2 y 4 no lo son.

- 8) Un sistema modular permite:
- a) Que los distintos módulos sean desarrollados por distintos equipos de desarrollo (o personas)
 - b) (a) y en general hace más fácil su verificación
 - c) (b) y que se encuentren más rápidamente programadores con baja productividad.
 - d) (c) y finalizar en plazo el proyecto

- 9) El cliente le hace llegar a su equipo de desarrollo un conjunto de estándares y normas técnicas que se utilizan en la empresa, y que necesariamente deben cumplirse durante la construcción del sistema. ¿Cómo clasificaría esta solicitud del cliente?
- a) Como una restricción del proyecto.
 - b) Como un requerimiento funcional.
 - c) Como un requerimiento no funcional.
 - d) Se podrían cumplir a) y c)

- 10) Dados los siguientes proyectos y modelos de proceso.

Proyecto:

A - De alto riesgo, muy complejo, alcance no del todo definido, varios subconjuntos de funcionalidad identificados como útiles para los usuarios, larga duración (dos años o más).

B – Riesgo medio, alcance definido, subconjunto de funcionalidad definido y útil para los usuarios, duración media (seis meses).

C – Riesgo bajo, simple, alcance definido, corta duración (seis semanas).

Proceso:

1 – Cascada

2 – En Fases con Evaluaciones Parciales Internas

3 – En Fases con Liberaciones Parciales en Producción

4 – De Prototipación

Marque la asignación que le parezca más adecuada. Si un mismo proyecto aparece asignado a más de un proceso, debe entenderse que cualquiera de los dos resultaría adecuado.

Para los procesos 2 y 3 se puede especificar además entre paréntesis el modelo de proceso a utilizar en cada fase.

Por ejemplo 2(1) significa Proceso en Fases con Evaluaciones Parciales Internas y en cada fase se aplica Cascada.

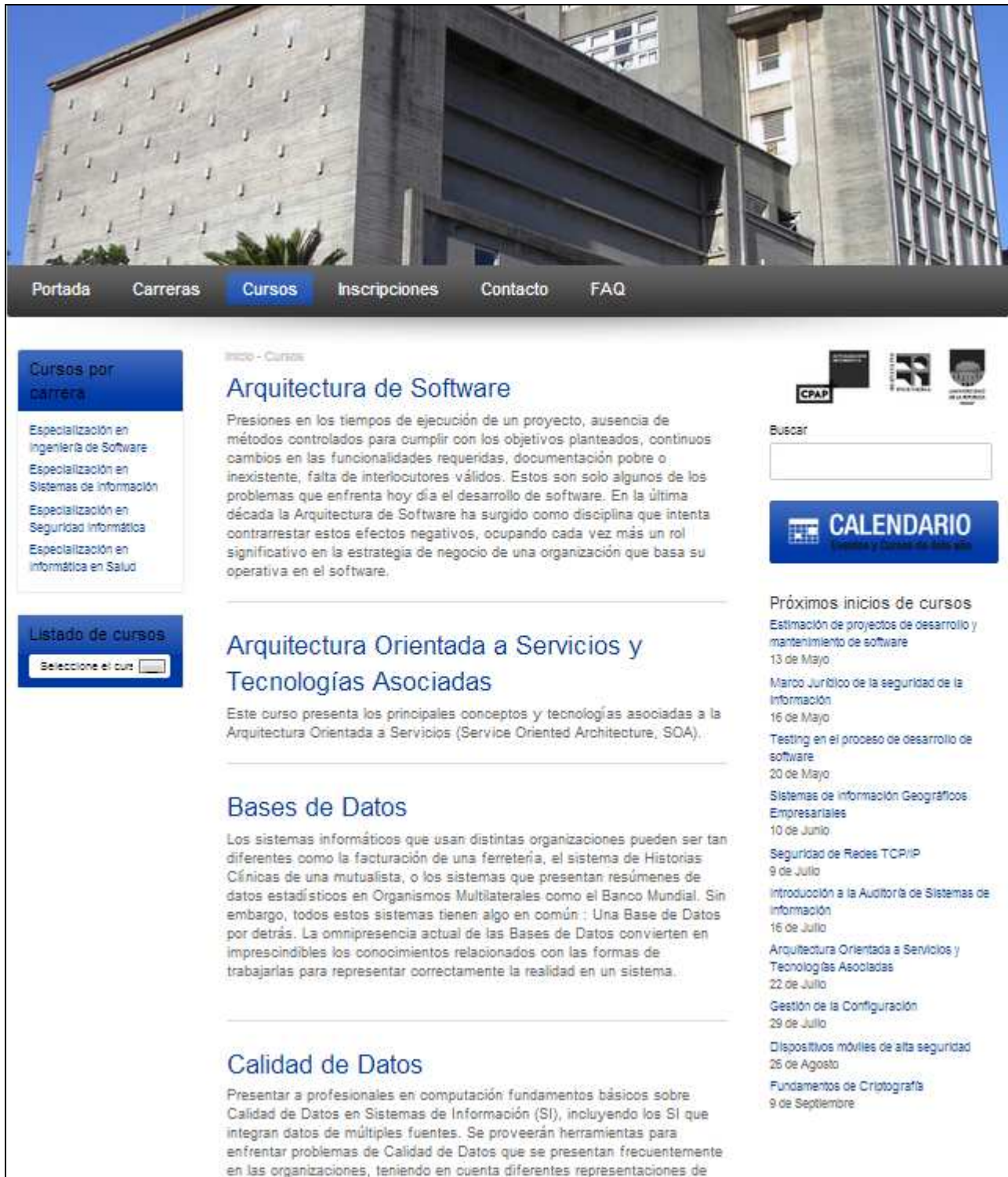
3(2(1)) significa que en este caso en cada fase de 3 se aplica En Fases con Evaluaciones Parciales Internas y en cada fase de 2 se aplica Cascada.

- a) A3(2(1)), B3(2(1)), C2(1)
- b) A4, B4, C4
- c) A1, B1, C1
- d) A2(1), B2(1), C3(2(1))

-
- 11) Los requisitos del sistema y los requisitos del software:
- Son lo mismo.
 - Los requisitos del sistema incluyen los del software.
 - Los requisitos del software incluyen las personas, los procesos manuales y otros elementos de soporte.
 - Los requisitos del sistema incluyen las personas, los procesos normales, pero no los elementos de soporte.
-
- 12) El Proceso de Ingeniería de Requisitos:
- Consiste en la obtención, análisis, especificación, verificación y validación de los requisitos del sistema y termina cuando se los tiene descriptos y validados.
 - Termina cuando se tiene una versión estable del Documento de Especificación de Requisitos.
 - Termina cuando el SRS no sólo es estable sino que está validado por el cliente y en línea base.
 - Continúa hasta el final del proyecto, porque incluye la gestión del cambio.
-
- 13) Sobre la gestión de riesgos:
- Comprende actividades que se llevan a cabo solamente durante la planificación del proyecto, ya que es en etapas tempranas del proyecto cuando se tiene el mayor conocimiento para identificar los principales riesgos del proyecto, categorizarlos, analizarlos y planificar sus respuestas.
 - La planificación de la gestión de riesgos, su identificación y categorización se realiza en etapas tempranas del proyecto. Sin embargo, tanto el análisis (cualitativo y cuantitativo) como la planificación de la respuesta a los riesgos se debe realizar una vez que el riesgo se produce (se hace real), ya que es en ese momento cuando se obtiene dicha información.
 - Durante la planificación del proyecto la mejor alternativa es planificar con el enfoque «si todo sale bien». Dado que los riesgos son eventos inciertos que por lo general no se manifiestan, el esfuerzo asociado a las actividades de gestión de riesgos resulta despreciable y por lo tanto no se incluyen en el cronograma del proyecto.
 - Durante el transcurso del proyecto pueden existir cambios sobre los riesgos que fueron identificados en etapas tempranas. Estos cambios incluyen desde una recategorización del riesgo, cambios en el impacto o probabilidad asignados, la elaboración de un nuevo plan de mitigación o contingencia, la eliminación de un riesgo, o la identificación de nuevos riesgos.
-
- 14) El objetivo de la validación de requisitos es asegurarse que:
- El sistema que se está desarrollando, efectivamente realiza lo que especifican los requisitos
 - Se está construyendo el sistema correctamente, ya que se han dado todos los pasos descriptos en el modelo de proceso.
 - Los requisitos no funcionales relevados son razonables.
 - Los requisitos relevados se corresponden con las necesidades del cliente.
-
- 15) Respecto a la Ingeniería de Software:
- Se ha visto influenciada entre otros factores, por el proceso de globalización con los consiguientes aumentos de la competencia y de la presión por acortar los plazos para salir al mercado (time to market).
 - Uno de los aspectos que ha sobrevivido a la prueba del tiempo es la utilidad de construcción de prototipos, esto es artefactos de software con un comportamiento que por razones fundadas suponemos análogo en algún aspecto al que va a tener el producto final, lo que permite evaluar indirectamente ese aspecto del comportamiento del producto final.
 - Está relacionada con la ingeniería de sistemas, en la medida que al construir software en realidad estamos construyendo un sistema compuesto por software, hardware y personas y es este sistema el que tiene que funcionar de forma adecuada.
 - Todas las anteriores son correctas.
-
- 16) Los requisitos deben ser trazables para permitir:
- Determinar su origen.
 - Calcular rápidamente el impacto de los cambios pedidos.
 - Rastrearlos en futuros desarrollos o mejora de la documentación.
 - Todas las anteriores.

-
- 17) Al definir la arquitectura de una solución:
- Es ideal para el arquitecto de software que el proyecto se realice siguiendo un proceso de desarrollo de software en cascada, de esta forma los requerimientos estarán bien definidos al inicio de la etapa de Diseño y esto garantizará la buena definición de la arquitectura.
 - Es importante involucrar distintos interesados en su definición y/o validación. Entre otros beneficios, esto incrementará el compromiso de los interesados en las definiciones alcanzadas y aportará a la detección temprana de defectos.
 - Es necesario contar con los casos de uso definidos.
 - (b) y (c) son correctas.
-
- 18) El enfoque de valor ganado permite:
- Controlar el avance del proyecto y saber si va atrasado o adelantado, sin necesidad de analizar el avance en el camino crítico.
 - Conocer cuándo el proyecto está atrasado o adelantado, en base a la diferencia entre el costo real y el planificado.
 - Predecir cuánto habrá costado el proyecto una vez terminado, si se sigue gastando y avanzando al mismo ritmo.
 - Ninguna de las anteriores.
-
- 19) Dadas las siguientes afirmaciones sobre la arquitectura de software indique cuál opción es correcta
- Las arquitecturas distribuidas tienen como desventaja que son complejas y que son difíciles de gestionar.
 - La elección del estilo y la estructura de la arquitectura más adecuados en un caso concreto, depende fuertemente de los requerimientos no funcionales
 - Una desventaja de una arquitectura SOA es que el enlace a los Servicios no se puede realizar en tiempo de ejecución.
 - Cambiar la arquitectura de software de un sistema ya construido es relativamente sencillo si se usa el estilo en capas o de tubos y filtros ya que cada componente (capa o filtro) es independiente del resto.
- Se cumplen 1 y 2. No se cumplen 3 y 4.
 - Se cumplen 1 y 4. No se cumplen 2 y 3.
 - Se cumplen todas.
 - Se cumplen 2 y 4. No se cumplen 1 y 3.
-
- 20) Construir un prototipo
- Sirve para relevar requisitos, porque puede usarse para definir la interfaz de usuario de forma que quienes la vayan a utilizar puedan experimentar su utilización
 - resulta adecuado para relevar y validar requisitos del software, no así para evaluar aspectos del diseño del mismo
 - no sirve para relevar requisitos, porque consiste solamente en desarrollar una parte o un aspecto del producto para evaluar la factibilidad de implementar el enfoque elegido con la tecnología a utilizar.
 - exige un esfuerzo adicional, que sólo se justifica si el prototipo puede evolucionar al producto final

Imagine que la nueva página del Centro de Posgrados y Actualización Profesional en Informática (CPAP) aún no se encuentra construida y que su equipo se deberá encargar de su desarrollo. El siguiente diseño de la página de "Cursos" es lo que el diseñador gráfico acaba de traerle a su escritorio.



Portada Carreras Cursos Inscripciones Contacto FAQ

Cursos por carrera

- Especialización en Ingeniería de Software
- Especialización en Sistemas de Información
- Especialización en Seguridad Informática
- Especialización en Informática en Salud

Listado de cursos

Seleccione el curso

Inicio - Cursos

Arquitectura de Software

Presiones en los tiempos de ejecución de un proyecto, ausencia de métodos controlados para cumplir con los objetivos planteados, continuos cambios en las funcionalidades requeridas, documentación pobre o inexistente, falta de interlocutores válidos. Estos son solo algunos de los problemas que enfrenta hoy día el desarrollo de software. En la última década la Arquitectura de Software ha surgido como disciplina que intenta contrarrestar estos efectos negativos, ocupando cada vez más un rol significativo en la estrategia de negocio de una organización que basa su operativa en el software.

Arquitectura Orientada a Servicios y Tecnologías Asociadas

Este curso presenta los principales conceptos y tecnologías asociadas a la Arquitectura Orientada a Servicios (Service Oriented Architecture, SOA).

Bases de Datos

Los sistemas informáticos que usan distintas organizaciones pueden ser tan diferentes como la facturación de una ferretería, el sistema de Historias Clínicas de una mutualista, o los sistemas que presentan resúmenes de datos estadísticos en Organismos Multilaterales como el Banco Mundial. Sin embargo, todos estos sistemas tienen algo en común: Una Base de Datos por detrás. La omnipresencia actual de las Bases de Datos convierten en imprescindibles los conocimientos relacionados con las formas de trabajarlas para representar correctamente la realidad en un sistema.

Calidad de Datos

Presentar a profesionales en computación fundamentos básicos sobre Calidad de Datos en Sistemas de Información (SI), incluyendo los SI que integran datos de múltiples fuentes. Se proveerán herramientas para enfrentar problemas de Calidad de Datos que se presentan frecuentemente en las organizaciones, teniendo en cuenta diferentes representaciones de

CPAP **UNIVERSIDAD DE BUENOS AIRES**

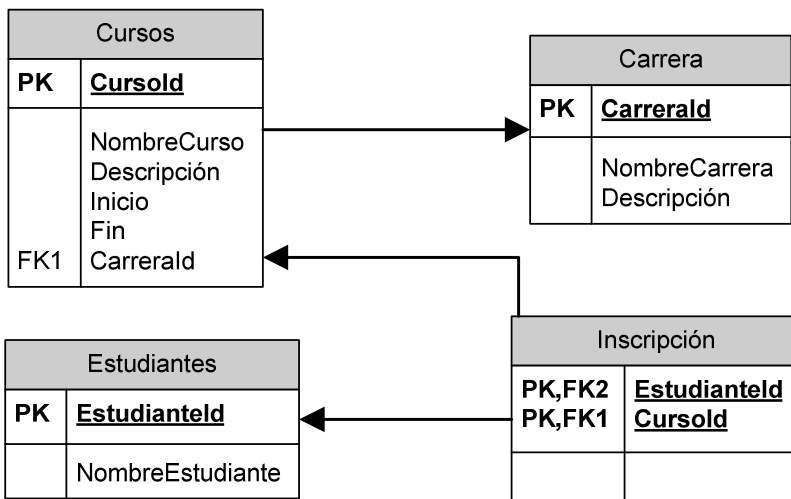
Buscar

CALENDARIO
Cursos y fechas de inicio

Próximos inicios de cursos

- Estimación de proyectos de desarrollo y mantenimiento de software
13 de Mayo
- Marco Jurídico de la seguridad de la información
16 de Mayo
- Testing en el proceso de desarrollo de software
20 de Mayo
- Sistemas de Información Geográficos Empresariales
10 de Junio
- Seguridad de Redes TCP/IP
9 de Julio
- Introducción a la Auditoría de Sistemas de Información
16 de Julio
- Arquitectura Orientada a Servicios y Tecnologías Asociadas
22 de Julio
- Gestión de la Configuración
29 de Julio
- Dispositivos móviles de alta seguridad
26 de Agosto
- Fundamentos de Criptografía
9 de Septiembre

Usted además cuenta con el siguiente modelo de datos.



Conteste entonces las siguientes preguntas en relación a la página Cursos:

- 21) Le piden una estimación de esfuerzo, ha decidido utilizar la técnica de puntos de función. Las transacciones que podemos identificar en la página desplegada son:
- Consulta de carreras (listado esquina superior izquierda), Consulta de cursos por carrera (seleccionando una carrera del listado esquina superior izquierda y desplegado en panel central), Consulta de todos los cursos (contenido central inicial de la página), Listado de todos los cursos (combo-box esquina central izquierda), Consulta de datos de un curso dado (esquina central izquierda o selección de un curso en panel central o en listado de cursos de inicio próximo), Lista de cursos con inicio próximo (listado en la columna más a la derecha).
 - Consulta de carreras (listado esquina superior izquierda), Consulta de cursos por carreras (listado esquina superior izquierda), Consulta de cursos (contenido central de la página y listado en la columna más a la derecha).
 - a) y Búsqueda de cursos por palabra (campo buscar, encima de listado de cursos con inicio próximo).**
 - a), pero cabe aclarar que la Búsqueda de palabra en cursos no se cuenta porque no se muestra el resultado en esta página, solo se pide la palabra a buscar.
- 22) Sobre los FTRs involucrados en las transacciones, indique cuál opción es correcta:
- La Consulta de carreras tiene un FTR sólo (Carreras), la Consulta de todos los cursos tiene dos FTRs (Cursos y Carreras)
 - La Consulta de carreras tiene un FTR (Carreras), la Consulta de todos los cursos tiene un FTR (Cursos)
 - b) y la Lista de cursos con inicio próximo tiene un FTR (Cursos).**
 - a) y los ILFs de Estudiantes e Inscripciones no se utilizan en las transacciones de la página web a desarrollar que se muestra en pantalla.
- 23) En cuanto a los DETs de las transacciones, indique cuál opción es correcta. Nota: Se asume que la fecha de hoy se saca del sistema.
- La Consulta de todos los cursos tiene dos DETs (Nombre y Descripción)
 - La Consulta de Carreras tiene un DET (Nombre)
 - La Lista de cursos con inicio próximo tiene dos DETs (Nombre y Fecha de Inicio)
 - La Consulta de cursos por carrera tiene 4 DETs (Carrerald, Cursold, NombreCurso y Descripción (de curso)).
 - La Consulta de todos los cursos tiene tres DETs (Cursold, Nombre y Descripción)
 - La Consulta de Carreras tiene dos DETs (Carrerald y Nombre)
 - Los botones del paginado no se cuentan como DETs.
- I, II, y III son correctas.
 - IV, V y VI son correctas.
 - I, II, III y VII son correctas.**

d) IV, V, VI y VII son correctas.

24) Usted ya realizó las siguientes actividades:

- Determinar la frontera de la aplicación.
- Identificación de transacciones.
- Identificación de archivos internos y externos.
- Conteo de DETs y FTRs para cada transacción.

¿Qué le está faltando para poder contar con una estimación de esfuerzo utilizando la técnica de puntos de función?

Sobre transacciones

- I) Determinar el tipo de cada transacción.
- II) Contar los DETs de cada transacción.
- III) Caracterización de la complejidad de las transacciones.
- IV) Contribución de Transacciones.

Sobre datos

- V) Contar los DETs y los RETs de los ILF y EIF.
- VI) Contar los DETs y los FTRs de los ILF y EIF.
- VII) Caracterización de la complejidad de los datos.
- VIII) Contribución de los Datos.
- IX) Utilizar los coeficientes de ajuste.
- X) Estimar esfuerzo a partir de los PFs calculados y el desempeño histórico del equipo.

- a) I, III, IV, VII, VIII, IX. La actividad IX ya da como resultado una estimación de esfuerzo.
- b) I, III, IV, V, VII, VIII, IX y X.
- c) I, II, III, IV, VI, VII, VIII, IX y X.
- d) I, III, IV, IX y X. El aporte de los datos ya fue contado al contar las transacciones.

25) Una empresa distribuidora mayorista desea implantar el paquete de software XYZ para soportar la gestión de inventarios de mercadería, pedidos a proveedores y entregas a clientes. Esta empresa cuenta actualmente con un sistema que funciona con tecnología obsoleta que soporta parcialmente estas funciones. Sus oficinas y depósito están ubicados en un mismo edificio. La empresa no cuenta con equipos informáticos con capacidad disponible como para implantar el paquete. El paquete XYZ tiene previsto un conjunto grande de parámetros para adaptarlo a las necesidades específicas de cada organización. Sin embargo, existe cierto riesgo de que resulte necesario incorporar modificaciones al software. Se pide construir un WBS para el proyecto, considerando que el resultado esperado es contar con el paquete XYZ implantado y funcionando en la empresa, de forma de satisfacer sus necesidades.

Se podría considerar incluir:

- I) Adquisición de nuevos equipos informáticos.
- II) La adaptación de los parámetros del paquete.
- III) El desarrollo de las modificaciones y el testeado del software.
- IV) La implantación del paquete.
- V) Las pruebas de aceptación del paquete.
- VI) El mantenimiento del paquete implantado.

Indique qué opciones deberían ser incluidas:

- a) I, II, III, IV, V. El WBS no debe incluir el mantenimiento, ya que no es parte del proyecto.
- b) II, III, IV, V. El WBS no debe incluir las adquisiciones, ya que no son parte del desarrollo del producto. Estos deberán ser provistos por otros servicios o secciones.
- c) I, II, III. El WBS no debe incluir la implantación ni las pruebas de aceptación, ya que ese no son un aspecto del desarrollo del producto, ni el mantenimiento, ya que tampoco es parte del proyecto.
- d) I, II, IV, V. El WBS no debe incluir las modificaciones al software, ya que no es seguro que se deban hacer.