

Finales Ingeniería de Software II

Una empresa de fletes de La Plata ha hecho convenios con otras empresas más pequeñas en el resto del país para optimizar los viajes y lograr mejores beneficios. Para eso necesita gestionar la información de los viajes de manera óptima. En una primera etapa piensan trabajar con 20 empresas con proyección a ir creciendo cada semestre. Los requerimientos se presentan muy variantes (cambiantes) tanto funcionales como no funcionales.

Se solicita que la interfaz sea intuitiva y pensada para evitar errores y poder informar claramente cómo actuar en cada caso.

[(1) Mencione y describa dos principios de Nielsen que apunten al feedback y la prevención de errores que son dos de los problemas detectados en la interface]. Tiene que soportar una fuerte demanda de

acceso y con un rendimiento óptimo las 24 horas del día. Para ello deciden hacer un sitio donde presentar sus servicios y que la gente pueda solicitarlos sin importar donde se encuentren.

El software por desarrollar parece ser de tamaño medio, pero se torna complejo por la organización de rutas de reparto y geolocalización de cada entrega que solicitaron para ser vistas en el sistema, esto va a hacer que el equipo esté mucho tiempo trabajando junto abordando nuevas tareas y capacitaciones para desarrollar el producto.

Deciden solicitar el desarrollo a SoftIT una empresa con amplia experiencia en Ingeniería de Software.

En la empresa SoftIT aceptaron el trabajo, y, dada las características del desarrollo decidieron Identificar y definir los elementos en el sistema, para controlar sus cambios a lo largo de todo el desarrollo.

[(2) a) ¿A qué estaría haciendo referencia? b) ¿Cuáles serían los elementos que controlar? Dé al menos 2 ejemplos.]

Leandro, líder del proyecto piensa en la mejor constitución del equipo que lo desarrollará. **[(3) Enumere al menos 3 factores a considerar cuando se planifica una estructura de equipo].**

Bruno (socio de Leandro), identifica los riesgos a cubrir. Le preocupa la decisión y luego de ordenarlos trazó una línea de corte.

[(4) a) Defina qué es la línea de corte. b) Teniendo en cuenta la decisión del inciso anterior, enumere 3 riesgos de este proyecto, cada uno con su estrategia].

Leandro sabe de la importancia de tener todo debidamente probado con lo cual sugiere diferentes técnicas de caja blanca y caja negra, y va realizando las pruebas de integración **[(6) Indique qué integración utilizaría, porqué y describa brevemente la misma].**

La empresa de fletes decide comprar un nuevo vehículo para sumar a la flota:

	Tarea	Precedida por	Duración
A	Hacer un estudio de factibilidad	-	3
B	Lista de los modelos posibles	A	1
C	Investigar todos los modelos posibles	B	3
D	Entrevista con el mecanismo	C	1
E	Obtener precios en las agencias	C	2
F	Compilar los datos adecuados	B,E	1
G	Escoger los tres modelos mejores	F	2
H	Conducción de prueba con los 3 modelos	G	3
I	Pedir datos de garantía y de financiamiento	G	3
J	Escoger un vehículo	H,I	1
K	Comprar en automóvil	F,G,J	2

[(7)Calcular las fechas más tempranas, más tarifas, realizar el grafico y encontrar el camino critico]

Video donde lo explica: [8_Clase 5 Planificación Temporal - 2022](#)

1. Describe 4 criterios de diseño de software.
 2. Menciona los paradigmas de equipo y describe 1.
 3. Menciona 4 principios de Nielsen y describe 1.
 4. 2 ventajas y 2 desventajas del patrón repositorio.
 5. Cuándo usar juicio experto.
-
1. ¿Qué es y para qué sirve un indicador?
 2. ¿Cuáles son los requerimientos no funcionales que se ven afectados por la arquitectura?
 3. ¿Qué es un riesgo? ¿Cuáles son los ítems de más alto riesgo según Bohem?
 4. ¿Cuáles son las consideraciones que se deben tener en cuenta a la hora de diseñar una interfaz?
 5. ¿Qué es la Gestión de Configuración? Definir línea base. Ejemplifique
 6. Rejuvenecimiento.
 7. Calcule el camino crítico.
-
- 1)Tipos de planificación organizacional vistos en la materia.
 - 2)Características de todos los riesgos
 - 3)Definir Recuperabilidad (principio de diseño)
 - 4)Diferencias entre modelo repositorio y modelo cliente servidor
 - 5)Qué es GCS. Línea Base, ejemplifique
 - 6)Describa rejuvenecimiento
- 1) Describa el problema de las “4 p”.
 - 2)¿Cuáles son los elementos claves de la gestión de proyectos?
 - 3)Describa el modelo MOI.
 - 4)¿Qué define el diseño arquitectónico? Describa los tipos de Organización del Sistema.
 - 5)Enumere y describa los principios de Diseño.
 - 6)Defina y describa GCS.
- 7)Ejercicio de PERT: Hallar tiempos tempranos - tardíos - caminos críticos y duración total del proyecto
-
1. ¿Qué es un proyecto? Describa el problema de las 4 “P”.
 2. En Planificación temporal, ¿Qué tareas se deben realizar para hacer el seguimiento y control del proyecto?
 3. ¿Cuáles son los tipos (áreas) de Diseño de Software? Describa.
 4. ¿Qué es Verificación? ¿Qué es Validación? De todos los tipos de pruebas vistos, ¿Cuál dirías que se está verificando y cual que se está validando al usar la prueba?
 5. Describa el Proceso de GCS.
 6. ¿Qué es el rejuvenecimiento del software? Describa los tipos.
 7. La empresa de pastas HOGAREÑAS S.A. desea hacer un estudio del tiempo que demora en hacer una lasaña, para esto ha identificado las siguientes actividades (en minutos). Encontrar el camino crítico, calculando tiempos tempranos y tiempos tardíos. Calcular el tiempo final.
-
1. ¿Qué es un riesgo? ¿Cómo se clasifican?
 2. ¿Cómo puede ser la organización del sistema? (diseño arquitectónico).
 3. ¿En qué se diferencian la verificación a la validación? Enumere las pruebas de integración que conozca.
 4. ¿Qué tipos de diseño de software conoce?
 5. Como es el proceso de GCS.
 6. Dada la siguiente tabla cuál o cuáles son el/los camino/s críticos? ¿Cuándo terminará el proyecto? Si llueve durante el periodo de excavación, ¿cuánto tiempo se retrasa el proyecto?

7. Explique el P-CMM (modelo de capacitación y motivación del personal)

- 1. Enumere y describa las técnicas de estimación que conozca.**
 - 2. Que es la gestión de configuración. Defina línea de base. Ejemplifique.**
 - 3. Describa el rejuvenecimiento.**
 - 4. Una empresa de software (bla bla bla) lleva un proyecto con requisitos muy estrictos (bla bla) pero no tiene el personal necesario para cierto proyecto. Analice la siguiente situación: Es imposible encontrar el personal con las habilidades adecuadas.**
 - a) Analice el riesgo.**
 - b) Planifique (Estrategias de tratamientos de riesgos).**
 - 5. Qué consideraciones se deberían tener en cuenta a la hora de desarrollar una interfaz de usuario.**
 - 6. Diferencie Modelo de Repositorio y Modelo Cliente/Servidor.**
 - 7. Un diagrama de Pert (Fácil).**
-
- 1. ¿Qué es la planificación? ¿Cuáles son los objetivos de la planificación temporal y organizacional?.**
 - 2. ¿Qué es el diseño? Defina acoplamiento y cohesión y sus diferentes grados.**
 - 3. ¿Qué es la GCS?**
 - 4. ¿Qué son los riesgos? ¿Cómo se clasifican?. Ejemplifique.**
 - 5. Defina estimación y los tipos de estimación que conoce.**
 - 6. ¿Qué es el mantenimiento? Defina el rejuvenecimiento.**
 - 7. Calcular camino crítico (PERT + CMP)**

Nº	Afirmación	V/F
1	Los elementos de la configuración del software son solo los programas y los datos.	
2	Una línea base es un concepto de GCS que nos ayuda a controlar los cambios.	
3	El análisis del riesgo no implica dejar de lado la gestión de riesgo, ya que podría llevar a obtener una deuda técnica.	
4	El concepto de evaluación de un riesgo no implica que siguiendo esta estrategia, la probabilidad de que el riesgo se materialice sea cero.	
5	Una tarea en la planificación temporal se describe por 3 elementos: precursor, duración y fecha de entrega.	
6	La estrategia de desarrollo en cascada es un proceso secuencial que se hace en etapas.	
7	La investigación de usuario es una de las fases del desarrollo de la experiencia de usuario (UX).	
8	La mantenibilidad y el rendimiento impactan en el diseño de la arquitectura de software.	
9	Las métricas GQM son base de datos que se usan en el ejemplo de diseño de la arquitectura de software en capas.	
10	Las métricas estándar se relacionan con las características de calidad del software.	
11	No es recomendable usar una métrica que “mida” cierto objetivo.	
12	Si se realiza una prueba y el software ha fallado significa que no hace lo que especifican los requerimientos.	
13	Un defecto por omisión en el software resulta cuando algún aspecto del código es incorrecto.	
14	La prueba de caja negra también denominada prueba de comportamiento, se centra en los requisitos funcionales del software analizando las entradas y salidas.	
15	La complejidad del software es una métrica que proporciona una medición cuantitativa del comportamiento de la calidad del software.	
16	Las técnicas de estrategias de pruebas son parte de la Verificación y Validación incluidas en el aseguramiento de la calidad del software.	
17	Las pruebas de regresión de software solo se pueden hacer en forma manual.	