

Facultad de Matemática, Astronomía,
Física y Computación
Universidad Nacional de Córdoba
Asignatura: Ingeniería del Software I
Segundo Cuatrimestre de 2024

1 | 9,5 Apellido: ACHURUE BENZENO
2 | 10 Nombre: TOMAS
3 | 10 Nro. hojas entregadas: 3
4 | 10
5 | 7

93% = (8,9)

Examen Parcial No. 1

Contestar con lapicera no verde

(U)

Ej. 1b. Describa los atributos principales asociados a la calidad de software según el estándar internacional de calidad de productos de software.

Ej. 2b. Describa la Metodología de diseño estructurado.

Ej. 3b. Explique los principales criterios de evaluación del diseño.

Ej. 4b. Describa cohesión. Explique los tipos de cohesión que aparecen en el diseño orientado a objetos.

Ej. 5b. Considere la siguiente descripción de un sistema y describa el caso de uso: *Cancelación de una Reserva*.

Una empresa de coworking ha solicitado un sistema para gestionar las reservas de salas de reuniones y espacios de trabajo. Los usuarios deben poder reservar espacios a través de una plataforma en línea, y el sistema debe garantizar que no haya conflictos de horarios. Además, debe permitir realizar cancelaciones, notificar a los usuarios sobre sus próximas reservas y generar reportes mensuales de uso para los administradores del coworking.

Los usuarios podrán ver la disponibilidad de las salas de reuniones y espacios de trabajo, y hacer reservas indicando la fecha, la hora de inicio y la duración. Los usuarios podrán cancelar sus reservas con un aviso previo de al menos 24 horas. El sistema debe enviar correos electrónicos a los usuarios confirmando sus reservas, recordatorios 1 hora antes del uso y notificaciones en caso de cancelaciones. El sistema no debe permitir la superposición de reservas para la misma sala o espacio. El administrador del coworking debe poder generar reportes mensuales que muestren las horas de uso de cada sala y las tasas de ocupación, como archivo PDF o Excel.

Ejercicio N° 16:

LOS ATRIBUTOS DE CALIDAD DEFINIDOS POR EL ESTÁNDAR INTERNACIONAL DE CALIDAD DE PRODUCTOS DE SOFTWARE SON:

- ✓ • **FUNCIONALIDAD:** EL SOFTWARE DEBE CUMPLIR CON TODAS LAS FUNCIONALIDADES REQUERIDAS.
- ✓ • **USABILIDAD:** EL SOFTWARE DEBE SER COMPENDIBLE, APRENDIBLE Y USABLE.
- ✓ • **CONFIABILIDAD:** REALIZACIÓN DE LOS REQUERIMIENTOS BASO LAS CONDICIONES ESTABLECIDAS EN UN TIEMPO DETERMINADO.
- ✓ • **MANTENIBILIDAD:** EL SOFTWARE DEBE ESTAR PREPARADO PARA AFRONTAR CAMBIOS Y A SEA POR METODOS O MANTENIMIENTO.
- ✓ • **EFICIENCIA:** EL SOFTWARE DEBE LOGAR UN BUEN DESEMPEÑO DADAS LAS RESTRICCIONES.
- ✓ • **PORTABILIDAD:** EL SOFTWARE DEBE PODER ADAPTARSE A DISTINTOS ENTORNOS SIN NECESIDAD DE REALIZAR MÁS ACCIONES QUE LAS QUE ESTÁN DEFINIDAS PARA ELLO.

Ejercicio N° 26:

LA METODOLOGÍA DE DISEÑO ESTRUCTURADO FORMA PARTE DEL PROCESO DE DISEÑO FUNCIONAL DE UN SISTEMA Y CONSISTE EN UNA SERIE DE PASOS A SEGUIR QUE FINALIZAN CON LA CONSTRUCCIÓN DE UN DIAGRAMA DE ESTRUCTURA DEL SISTEMA. LOS PASOS SON:

1) CREAR UN DIAGRAMA DE FLUJO DE DATOS DEL PROBLEMA. ESTE TENDRÁ ENTIDADES, TRANSFORMACIONES Y DATOS.

2) IDENTIFICAR LAS MAI Y MAO DEL DIAGRAMA.

• **MAI:** ENTRADAS MÁS ABSTRACTAS, SON LOS DATOS QUE INGRESAN AL SISTEMA LUEGO DE PASAR POR TRANSFORMACIONES DE VALIDACIÓN, FORMATEO, ETC.

• **MAO:** SON LAS SALIDAS MÁS ABSTRACTAS, ES DECIR, LOS DATOS QUE SALEN DEL SISTEMA PRINCIPAL PREVIO A TRANSFORMACIONES DE FORMATEO, ORDENAMIENTO, ETC.

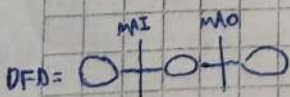
3) REALIZAR EL PRIMER NIVEL DE FACTORIZACIÓN: AQUÍ COMIENZA LA CONSTRUCCIÓN DEL DIAGRAMA DE ESTRUCTURA, COMPUESTO POR:

• **MÓDULO COORDINADOR:** CONTROLA LAS INTERACCIONES DE LOS SIGUIENTES 3 MÓDULOS.

• **MÓDULO DE ENTRADA:** ES SUBORDINADO POR LAS TRANSFORMACIONES PREVIO A LAS MAI. PROCESA LOS DATOS EXTERNOS HASTA CONVERTIRLOS EN MAI.

• **MÓDULO DE TRANSFORMACIÓN:** ES SUBORDINADO POR LAS TRANSFORMACIONES CENTRALES DEL SISTEMA. REALIZA EL PROCESAMIENTO MÁS IMPORTANTE DE LOS DATOS.

• **MÓDULO DE SALIDA:** SIMILAR AL MÓDULO DE ENTRADA, PERO PARA LAS TRANSFORMACIONES POSTERIORES A LAS MAO.



4) REFACTORIZAR LOS MÓDULOS DE ENTRADA, SALIDA Y TRANSFORMACIÓN. L

ESTE PASO CONSISTE EN APLICAR EL PASO 3 A LOS MÓDULOS OBTENIDOS, DANDO CADA UNO UNA COORDINADA DE SUS SUB-MÓDULOS.

SE REPITE HASTA OBTENER MÓDULOS "ATÓMICOS" I.E. QUE REALICEN UNA TAREA SIMPLE DE PROCESAMIENTO DE DATOS.

5) MEJORAR EL DISEÑO OBTENIDO

Aquí SE PUEDEN APLICAR HEURÍSTICAS DE DISEÑO QUE LO ANALIZAN EN TÉRMINOS DE:

- TAMAÑO: POR EJEMPLO, LA CANTIDAD DE MÓDULOS Y SUS TAMAÑOS.
- CANTIDAD DE ENTRADAS Y SALIDAS DE CADA MÓDULO, IDEALMENTE MUCHAS DE ENTRADA (MUY UTILIZADO) Y POCAS DE SALIDA.
- ALCANCE DEL EFECTO DE UN MÓDULO: CUÁNTOS MÓDULOS SE VEN AFECTADOS POR UNA DECISIÓN EN UN MÓDULO.
- ALCANCE DE CONTROL DE UN MÓDULO: SU CANTIDAD DE MÓDULOS SUBORDINADOS.
- EXISTEN MÁS HEURÍSTICAS QUE PUEDEN SER UTILIZADAS.

OBS: ALCANCE DE EFECTO & ALCANCE DE CONTROL.

EJERCICIO Nº 3b: CRITERIOS DE EVALUACIÓN DEL DISEÑO

- ✓ CONEXIÓN: EL DISEÑO DEBE CUMPLIR CON LOS REQUERIMIENTOS ESTABLECIDOS Y DEBE SER FACTIBLE DADAS LAS RESTRICCIONES DEFINIDAS.
- ✓ EFICIENCIA: EL DISEÑO DEBE HACER BUEN USO DE LOS RECURSOS DISPONIBLES. ESTO PUEDE AYUDAR CON LA ESCALABILIDAD DEL SISTEMA Y CON LA REDUCCIÓN DE COSTOS DE OPERACIÓN.
- ✓ SIMPLICIDAD: UN DISEÑO SIMPLE FACILITA LA COMPRESIÓN, EL TESTING Y LA MANTENIBILIDAD DE UN SISTEMA.

Ejercicio N° 4b:

LA COHESIÓN ES UN TÉRMINO QUE CAPTURA EL CONCEPTO DE CONEXIÓN INTRA-MODULAR DE UN SISTEMA. EN OTRAS PALABRAS, DEFINE QUE TAN FUERTEMENTE SE VINCULAN LOS ELEMENTOS QUE COMPONEN UN MÓDULO ESPECÍFICO. EN UN BUEN DISEÑO SE ESPERAN ALTOS NIVELES DE COHESIÓN, LOS CUALES TAMBIÉN AYUDAN A REDUCIR LA DEPENDENCIA INTER-MODULAR DEL SISTEMA.

EN EL DISEÑO ORIENTADO A OBJETOS, DONDE LOS MÓDULOS SUELEN ESTAR DEFINIDOS POR CLASES, EXISTEN TRES TIPOS DE COHESIÓN:

- ✓ DE MÉTODO: ^{todos} Si los elementos que conforman un método contribuyen a la ejecución de una única función, HAY UNA MAYOR COHESIÓN.
- ✓ DE CLASE: UNA CLASE QUE ENCAPSULA UN ÚNICO CONCEPTO TIENE MÁS COHESIÓN QUE UNA EN DONDE NO OCURRE ESO.
- ✓ DE HERENCIA: POR ESPECIALIZACIÓN/GENERALIZACIÓN (MÁS COHESIÓN)
POR RENOVO (MENOS COHESIÓN)

Ejercicio N° 5b:

CASO DE USO 1: CANCELACIÓN DE UNA RESERVA

ACTOR PRIMARIO: USUARIO

PRECONDICIÓN: EL USUARIO DEBE TENER UNA RESERVA EXISTENTE.

ESCENARIO EXITOSO PRINCIPAL:

cuando se logre?

1- EL USUARIO REALIZA UN AVISO DE CANCELACIÓN DE RESERVA.

2- EL SISTEMA VERIFICA QUE FALTEN AL MENOS 24 HORAS PARA LA FECHA DE INICIO DE LA RESERVA, EN UNA LA RESERVA DEL SISTEMA Y ENVÍA UNA NOTIFICACIÓN AL CORREO ELECTRÓNICO DEL USUARIO INFORMÁNDOLE LA CANCELACIÓN.

ESCENARIOS EXCEPCIONALES:

2.a)

1- EN EL PUNTO 2 DEL ESCENARIO EXITOSO PRINCIPAL, EL SISTEMA DETECTA QUE FALTAN MENOS DE 24 HORAS PARA EL INICIO DE LA RESERVA

• SE LE INFORMARÁ AL USUARIO QUE NO ES POSIBLE REALIZAR LA CANCELACIÓN DE RESERVA DESIADA.