

[Programa](#) > [Unidad](#) > Evaluación 

Examen Teorico 1

Examen Teorico 1

- Puntaje total: 100.00
- Puntaje de aprobación: 70.00
- Incorrectas restan: No
- Abierta: desde 08/11/2024 17:30 hasta 25/04/2025 19:15

Realización

- Fecha: 21/04/2025 18:06:17
- Tiempo realización: 00:32:14
- Cantidad de veces realizada: 1
- Cantidad de respuestas correctas: 8 / 20

 Liberada - 40.00

La evaluación fue mal realizada y puede rehacerse | [Intentar nuevamente](#)

Unidad 1: Ciclos de vida

Por defecto marque solo una CORRECTA, a menos que se indique: Marque LA FALSA, o Marque 2 o 3..

Pensar como un analista implica (**Multichoice MARQUE 2 CORRECTAS**)

- ☒ Que los programas informáticos se adaptarán al negocio del cliente
- ☐ Que el negocio se deberá alterar para adaptarse a nuestros programas
- ☐ Enfocarse en entender como se construye el código de mi programa
- ☒ Enfocarse en que el software entregado cumpla con los requisitos
- ☐ Nada que ver una cosa con la otra, marco que no hay correctas solo ésta

(5.00 puntos)

¿Cuál es la **relación** de **Ingeniería** del Software con **Code & Fix**?

- ☐ Es una antigua técnica de desarrollo de código sin planos de software, pero NO se relacionan
- ☐ La ingeniería ofrece prototipos para diferentes etapas y tareas de la técnica Code & Fix
- ☐ Code & Fix da soporte al ciclo de vida garantizando sistemas de calidad y libres de errores
- ☐ Se relaciona con UML dados los modelos que el ciclo Code&Fix provee
- ☒ Como consecuencia de que Code &Fix no tiene modelos del sistema, surgió la ingeniería del software

(5.00 puntos)

La **Ingeniería del software** es:

- ☒ Se trata de un área de la informática que ofrece métodos y técnicas de desarrollo de software
- ☐ Es lo mismo que Code&Fix solo que lo estandarizaron como lenguaje
- ☐ Un método que usa al ciclo de vida para expresar cual es el código y corregir
- ☐ Son etapas necesarias para transformar nuestro código con altas probabilidades de éxito
- ☐ Es un marco de referencia para escribir planos o modelos de software
- ☐ Ingeniería, ciclo de vida, UML...Creo que ninguna de estas es correcta

(5.00 puntos)

Iterativo Vs. **Incremental** (Marque una columna o la otra)

Importante: solo marque una casilla de cada columna.

	Iterativo	Incremental
Se puede resentir la calidad del software, de tanto cambiar los requisitos en las iteraciones	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Se usa cuando los equipos de desarrollo están en diferentes lugares, separados geográficamente	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Se utiliza cuando los requerimientos no están claros, ni completos	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Se le entrega al cliente una versión mejorada del producto	<input checked="" type="checkbox"/>	<input type="checkbox"/>

En cada pasada se revisan y completan TODAS las funciones del sistema	<input checked="" type="checkbox"/>	
Permite a cada miembro del equipo desarrollar módulos particulares en incrementos separados		<input checked="" type="checkbox"/>
Se van obteniendo prototipos con más (nuevas) funciones que los anteriores		<input checked="" type="checkbox"/>
Se puede controlar la Calidad del software, al conocerse los requisitos desde inicio se sabe como quedará		<input checked="" type="checkbox"/>
Facilita el desarrollo con la filosofía divide & conquistarás (módulos o incrementos)		<input checked="" type="checkbox"/>
Los entregables NO son operativos, hasta que estén completos (Mejora final)	<input checked="" type="checkbox"/>	

(5.00 puntos)

(MARQUE LA FALSA) La técnica Code & Fix. ¿cual es la Falsa?

- ☐ Finaliza al satisfacer las especificaciones surgidas durante el desarrollo
- ☐ Tiene aumento en costo de trabajo, dinero, y recursos
- ☐ Fue desarrollada por los tres amigos en la década de los '70
- ☐ Tiene un aumento en el tiempo de desarrollo (más horas)
- ☒ Tiene un aumento en la calidad del código que se produce
- ☐ Recomendable en esos casos donde tenemos un microproyecto

(5.00 puntos)

"Ingeniería directa" significa:

- ☐ Me permite generar un ciclo de vida de la aplicación deseada
- ☐ Útil en un sistema que necesito modificar, pero no tengo el diseño
- ☒ Es la generación de código a partir de los modelos ya creados
- ☐ Es la reconstrucción de modelos a partir de código programado
- ☐ Creo que todas las opciones son correctas

(5.00 puntos)

Unidad 2: Captura y CU

Por defecto marque solo una CORRECTA, a menos que se indique: Marque LA FALSA, o Marque 2 o 3..

(Multichoice MARQUE 3 CORRECTAS)

Relaciones de Uso o Inclusión: Siendo A caso base y B caso incluido.

- ☐ El caso incluido B es ejecutado siempre que el caso que lo usa A, es usado por otro caso
- ☐ Aparecen como funcionalidad común, cuando hay exactamente 2 casos que lo usan
- ☒ Sirven para no repetir el texto de la funcionalidad en otros los casos de uso que lo incluyan
- ☐ El caso B es ejecutado solamente si en el caso que lo usa A, se cumple cierta condición (opcional)
- ☒ Aparecen como funcionalidad compartida, cuando 2 o más casos de uso lo invocan
- ☐ Sirven para no modelar comportamientos adicionales u opcionales en una funcionalidad
- ☒ El caso B se ejecuta siempre que un actor invoque el caso A (que también incluye a B)

(5.00 puntos)

Marque: El o los propósitos del modelo de casos de uso.

- ☐ A- Comunicar el comportamiento del sistema al cliente y/o usuario y que este lo valide
- ☐ B- Facilita la comunicación con los desarrolladores sobre cómo funciona el sistema
- ☐ C- Refinar los requisitos funcionales con clases y atributos para que sean más entendibles para el equipo
- ☐ D- Encontrar el conjunto de clases y objetos, la arquitectura para que funcione el sistema
- ☒ E- Creo que las dos son correctas: A y B

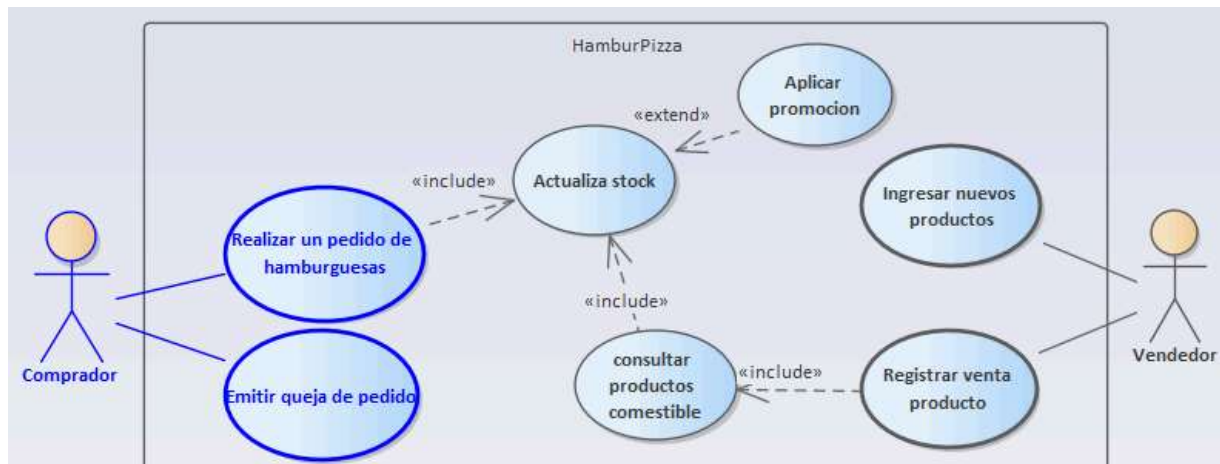
(5.00 puntos)

Actores del modelo de Casos de Uso

- ☐ No son parte del sistema, son terceros que representan posibles roles a jugar por usuarios
- ☐ Tienen necesidades del sistema y lo suplirán de información desde el exterior para conseguirlas
- ☐ Un actor puede ser solamente un receptor pasivo de información en algunos casos
- ☐ Un mismo usuario puede actuar como diferentes actores en distintos momentos
- ☐ Un actor puede representar a otros sistemas que se comunican con mi sistema
- ☐ Es valido decir que un actor es todo esto aquí, y admito que TODAS son correctas

Sin respuesta

(5.00 puntos)



(**MARQUE 3**) ¿Que *TRES* errores vemos en este modelo de casos de uso?

- ☐ Los casos "realizar pedido" y "registra venta producto" son casos duplicados
- ☐ El vendedor "registra venta producto", debo sacar "realizar pedido" del cliente
- ☐ Veo un gusano de <<include>>, un caso incluye a otro que incluye a otro caso
- ☐ Cuando "ingreso nuevos productos", falta un <<include>> a "actualizar stock"
- ☐ El caso "aplicar promoción" no es <<extend>>, debe ser <<associate>> a vendedor
- ☐ En caso "Emitir queja pedido", no es <<associate>>, sino <<extend>> de "realizar pedido"

Sin respuesta

(5.00 puntos)

(Marque 2 correctas) Una relación de asociación actor-caso de uso

- ☐ implica que un actor produce el primer evento y el sistema responde
- ☐ implica que esa funcionalidad se desencadena independiente de otras
- ☐ en uml se traza una línea sólida entre ambos casos de uso asociados
- ☐ implica que ese caso de uso surge, es parte dependiente de otros casos
- ☐ implica que ese CU se ejecuta en tiempo, a continuación de otro CU
- ☐ en uml se asocian con una línea punteada entre ambos casos de uso

Sin respuesta

(5.00 puntos)

(Multichoice x3) El Comportamiento del Sistema

- ☐ Es como el sistema actúa y reacciona a estímulos externos
- ☐ Define los requerimientos funcionales del sistema
- ☐ Se desarrolla y muestra el cliente, quien propone mejoras y clases
- ☐ Es lo que dará soporte y mantenimiento al sistema
- ☐ Para cada mensaje enviado se asume una respuesta
- ☐ Los usuarios aumentan su comprensión del diseño y código
- ☐ Casos de uso describen al sistema y las relaciones con su ambiente

Sin respuesta

(5.00 puntos)

(Marque 2) YouTube +, tiene una sección premium de video pay-per-view:

- Alquila películas a sus clientes por 1 día, de un listado mensual.
- Se inscriben como usuario del sistema web
- Acceden al sistema y envían la solicitud de las películas que requieran.
- Se desbloquea contenido de la película y lo carga a la tarjeta

Precondición: se cumple previamente para habilitar la ejecución del Caso. Algunas cosas pueden ser tratadas como caminos alternativos y no como precondición.

¿Cuáles son las 2(dos) precondiciones para el CU "Alquilar pelis" ?

- ☐ Que YouTube + esté emitiendo la película en el listado del mes
- ☐ Que YouTube + haya desbloqueado el contenido de la peli
- ☐ Que el cliente haya iniciado sesión en el sistema de alquiler

- ☐ Que el cliente no este inscripto a otro sistema de per-per-view
- ☐ Que el cliente tenga asociada una tarjeta de crédito para alquilar
- ☐ Que el cliente no tenga desbloqueado el contenido de otra peli
- ☐ Que el cliente tenga saldo en su tarjeta de crédito para alquilar

Sin respuesta

(5.00 puntos)

Unidad 3: Analisis

Por defecto marque solo una CORRECTA, a menos que se indique: Marque LA FALSA, o Marque 2 o 3..

Relacione conceptos.

Marque 2 casillas o columnas, por cada línea.

	buscarErrores	clases y objetos	codigo	casos de uso	Actor o Usuario	requisitos	Dinamico y Estatico
Captura							
Analisis							
Diseño							
Debugging							
Entorno Externo							
Test Alfa Beta							

Sin respuesta

(5.00 puntos)

Si me quiero inscribir para RENDIR una mesa de examen, el controlador debe:

- ☐ Validar que tenga falta de correlatividades al rendir
- ☐ Crear un nueva correlatividad entre las materias
- ☐ Verificar que actualmente yo este inscripto a la materia
- ☐ Verificar que existe un profesor que dicta la materia
- ☐ Validar que se esté dictando la materia en el programa

- ☐ Validar que no tenga ya una inscripción a la mesa

Sin respuesta

(5.00 puntos)

El encapsulamiento

- ☐ El usuario no se afecta si cambia el funcionamiento interno pero conserva la misma interfaz
- ☐ Es ocultamiento del estado, y sólo se puede cambiar con operaciones de ese objeto
- ☐ Permite esconder la complejidad (de un objeto) detrás de una interfaz con el usuario
- ☐ Cada objeto está aislado del exterior, solo al conocer la interfaz ya se puede utilizar
- ☐ Evita que el usuario cambie el estado (de un objeto) de maneras imprevistas e incontroladas
- ☐ Puedo observar que ninguna opción es falsa, y solo por eso voy a elegir esta opción

Sin respuesta

(5.00 puntos)

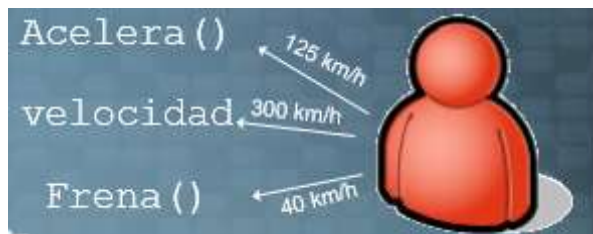
Cuando quiero **transferir** dinero de un cajero entre dos cuentas *c1(origen)*, y *c2(destino)*

- ☐ El control valida que haya saldo en la cuenta c1
- ☐ El control valida que haya saldo en la cuenta c2
- ☐ El control verifica que el PIN sea correcto para c1
- ☐ El control verifica que haya superado el máximo a transferir
- ☐ El control valida la banda magnética de la tarjeta insertada
- ☐ El control valida todo esto, es el camino básico de 'transferir'

Sin respuesta

(5.00 puntos)

*La siguiente imagen nos indica que, los métodos **acelerar()** y **frenar()** trabajan sobre el atributo **velocidad**. ¿De qué concepto hablamos sobre el atributo **velocidad**?*



- ☐ Herencia
- ☐ Polimorfismo
- ☐ Encapsulamiento
- ☐ Abstracción
- ☐ Modularidad
- ☐ Model-View-Control

Sin respuesta

(5.00 puntos)



Pre-condición: Viene con su factura de

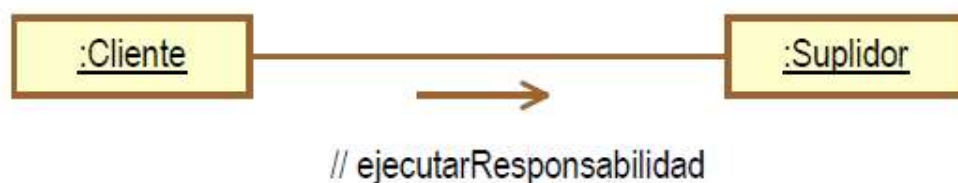
compra, producto a devolver. Se busca la venta, guarda la devolución, y se imprime (en forma de NC). Al analizar éste caso de uso, se identifican **las clases de entidad participantes** son:

- ☐ Nota de credito y Devolucion
- ☐ Producto y Devolución
- ☐ Devolución y Venta
- ☐ Nota de Crédito y Vendedor
- ☐ Nota de Crédito y Producto

Sin respuesta

(5.00 puntos)

Diagrama de Interaccion



- ☐ La responsabilidad va implementada en el objeto cliente
- ☐ La responsabilidad va implementada en el mensaje que se envía
- ☐ La responsabilidad iría implementada en el objeto suplidor
- ☐ La responsabilidad es propia del otro diagrama llamado "diagrama de interaccion"
- ☐ La responsabilidad no se necesita implementarla, solo ejecutarla

Sin respuesta

(5.00 puntos)

[Anterior](#)