INTEGRADOR SEGUNDO PARCIAL

**IMPORTANTE:** Es condición excluyente para rendir este examen el **cumplimiento total de las condiciones** de la segunda entrega del Trabajo Práctico.

Desarrollar **individualmente** los temas no aprobados, respetando las consignas específicas si las hubiera.

Se considera no aprobado un tema que, en las instancias anteriores:

* tiene una calificación *regular* (*R*);
* tiene una calificación *mala* (*X*);
* o no tiene calificación (no se desarrolló).

Puede consultar **únicamente** la planilla de devoluciones [haciendo clic en este enlace](https://docs.google.com/spreadsheets/d/187OoMFd708zBQrca4_lAo316sCrQaEf2K7uuMhhcXP8/edit?usp=sharing).

Las consignas de los temas aprobados deberán eliminarse. Al momento de la entrega, este documento sólo deberá contener las consignas y desarrollos correspondientes a **los temas no aprobados**.

Cada tema deberá ser desarrollado **a continuación de su consigna**, sin utilizar **negrita** ni subrayado. En caso de necesitar destacar un concepto o comentario, puede utilizarse el resaltado.

Se valorarán la **claridad de las explicaciones**, la **brevedad y precisión** (sin omisión de información), la **aplicación en ejemplos** diferentes a los vistos en clase, y el **uso de recursos visuales** (cuadros, diagramas, mapas conceptuales, etc).

Suba su parcial en el siguiente Google Form:

<https://forms.gle/5fZQNJMczU4H3yxu8>

# 

# 

# TEMAS

## CURSADA

### Modelo de implantación del usuario

1. Explique las definiciones del modelo de implantación del usuario para la interfaz humano-máquina. ¿Por qué esta interfaz es importante? ¿Qué características debe tener una buena interfaz? Justifique.

El modelo de implantación del usuario es la extensión del modelo esencial

con los requisitos no funcionales impuestos por el usuario.

La interfaz humano-máquina es la forma en que el usuario interactuará con el sistema, esta es muy importante ya que con una buena, el usuario va a poder guiarse de manera más natural y no va a necesitar tanta ayuda para utilizar el sistema como la necesitaría con una mala interfaz.

Para esto, la interfaz debe ser fácil de usar, intuitiva, deben estar detalladas las tareas más complejas en caso de que algo no sea lo suficientemente intuitivo para que el usuario se pueda guiar

### Calidad

**2)** ¿Qué son los atributos de calidad? ¿Para qué sirven? ¿Por qué son importantes?

### Diseño / Diseño estructurado

1. Explique qué son y cómo se relacionan la cohesión y el acoplamiento.

El diseño es la creación del modelo físico es decir, la combinación entre modelo esencial y el de implantación del usuario, es donde se introduce calidad: traduciendo los requerimientos de los clientes

El diseño estructurado básicamente lo que busca es el equilibrio entre el rendimiento, es decir, la velocidad en la que se realiza un trabajo, el control osea que se proteja contra errores humanos y problemas de hardware y la cambiabilidad, que es la capacidad de modificación y/o adaptación del sistema.

Si el diseño no tiene cierta cohesión, el modelo físico no va a ser el óptimo y se va a tener que rehacer ya que quizá no es lo que el cliente espera o no esta armado para que pueda seguir expandiéndose en un futuro, o no cumple con el rendimiento esperado

### Modelo físico

1. Indicar y explicar las partes del Modelo Físico.

El modelo físico es la combinación entre el modelo esencial y el modelo de implantación del usuario, en este se muestra como se utilizaran los recursos físicos para satisfacer estos modelos.

Este modelo se separa en:

El modelo de implantación de programas que se encarga de organizar módulos de forma jerárquica para cada tarea, y el modelo de implantación de sistemas que a su vez se separa en:

El modelo del procesador encargado de asignar almacenes y procesos a procesadores y el modelo de tareas que cumple la función de asignar procesos y almacenes a tareas

### Programación

**5)** *Matar un mosquito con una bazuca* podría ser un ejemplo de:

* 1. control
  2. corrección
  3. eficacia
  4. eficiencia
  5. insensatez
  6. productividad

Justifique la/s opciones escogidas, y enuncie un ejemplo de nuestra área profesional en el que considere que aplica esta analogía.

### Prueba

**6)** Explique cómo se desarrolla el proceso de prueba del software en las dos actividades relevantes del Ciclo de Vida Estructurado.

### Actividades finales del Ciclo de Vida Estructurado

**7)** ¿Bajo qué condiciones se puede considerar que la actividad de Instalación ha sido completada de forma exitosa?

### Mantenimiento de la especificación

**8)** ¿Es necesario mantener la especificación luego de completar el proyecto? Justifique su respuesta.

### Modelado orientado a objetos

**9)** Enuncie los objetivos y principios del modelado según Booch, Rumbaugh y Jacobson. Compárelos con los expresados por Yourdon.

El modelado, tiene como objetivo el poder crear una vista de cómo sería el objeto en cuestión cuando este se haya construido. Esta se crea para mejorar el entendimiento de todos los participantes en la construcción de este objeto determinado. Además se crea el modelo para evitar lo más posible los cambios en el objeto ya construido, pudiendo modificarlos en el modelo a un precio menor y sin tanto esfuerzo.

### Conceptos básicos del paradigma orientado a objetos

**10)** Explique y ejemplifique las siguientes divisiones comunes:

* 1. Abstracción vs. Manifestación concreta
  2. Interfaz vs. Implementación
  3. Tipo vs. Rol

1. Las abstracciones son las características esenciales de una entidad que la distinguen de otras entidades mientras que la manifestación concreta es el cómo estas características toman valores determinados para su utilización

Ejemplo: Una abstracción sería un auto, este tendría ciertas características como por ejemplo color, marca, precio y demás y la manifestación concreta es un auto específico como un Ferrari rojo que vale $500.000 dólares.

1. Una interfaz es un tipo de abstracción en la que no se especifican los métodos a utilizar pero si se sabe que atributos usa, y que devuelven estos métodos. La implementación es la utilización de estas interfaces en las clases

Ejemplo: Una interfaz suma con el método sumar que sé que devuelve un número y tiene otros dos como atributos. Una clase calculadora los puede implementar y va a tener que especificar como se hace la suma, pero si o si va a tener que devolver un número y utilizar estos otros dos números como atributos

### Clasificadores

**11)** Enumere los diferentes clasificadores estudiados, e indique sus instancias. En los casos en los que sea posible, ejemplifique con sintaxis de UML.

Clasificador: Mecanismo que describe características (propiedades encapsuladas) estructurales y de comportamiento

### Relaciones

**12)** Enumere las diferentes relaciones estudiadas, e indique entre qué elementos se pueden dar. En los casos en los que sea posible, ejemplifique con sintaxis de UML.

La generalización es la relación de especialización en la cual las instancias del elemento especializado pueden sustituir a las del general. Por lo general se da entre las clases. 

La asociación se le dice a la relación semántica entre dos clasificadores que implica la conexión entre sus instancias. 

Y la dependencia es una relación semántica entre elementos en la cual un cambio a uno puede afectar a la semántica del otro. Se da más que nada cuando las clases implementan las interfaces.

### Herencia

**13)** ¿En qué contexto se da la herencia? Indique los diferentes elementos estudiados que pueden intervenir en este mecanismo, y explique de qué manera lo hacen.

La herencia se produce cuando un elemento más específico incorpora la estructura y el comportamiento de elementos más generales. Diferentes elementos pueden intervenir en esta. Los atributos, los métodos y las clases son algunos ya que cuando una clase hereda de otra, si o si hereda los atributos y los métodos de la clase padre

### Polimorfismo

**14)** ¿Qué relación tiene el polimorfismo con la herencia?

### Encapsulamiento

**15)** ¿Qué es la visibilidad de un atributo u operación? ¿Cuáles son sus niveles posibles? Ejemplifique con sintaxis de UML.

### Diagrama de clases

**16)** Realizar un diagrama de clases (con clases abreviadas, es decir, sin atributos, operaciones ni responsabilidad) en donde se puedan observar las distintas relaciones entre clasificadores. El ejemplo deberá ser distinto a cualquiera de los presentados en clase o en la bibliografía.

### Conceptos básicos UML

**17)** ¿Qué relación tiene UML con las metodologías de desarrollo de software?

Uml es un lenguaje unificado de modelado, sirve para visualizar y construir un modelo explícito y preciso.

Se relaciona con las metodologías de software porque puede conectarse directamente con una gran variedad de lenguajes de programación y crear modelos de un sistema con gran cantidad de software

### Diagramas UML

**18)** Identifique los diagramas que ofrece UML, clasifíquelos, y explique brevemente para qué puede utilizarse cada uno.

Los diagramas estructurales: Muestran los aspectos estáticos del sistema. En estos están:

Los diagramas de clases los cuales muestra un conjunto de clases, interfaces y colaboraciones, y las relaciones que estos tienen.

Los diagramas de objetos que muestran un conjunto de objetos y sus relaciones.

Los diagramas de componentes que resaltan las distintas interacciones entre los componentes con el sistema y entre sí.

Los diagramas de artefactos muestran el hardware que usa el sistema.

Los diagramas de despliegue los cuales resaltan la configuración en tiempo de ejecución de los nodos de procesamiento, y los componentes que residen en ellos.

Y los diagramas de comportamiento: Muestran los aspectos dinámicos del sistema. En estos se encuentran

Los diagramas de casos de uso muestran los casos de uso y actores, y sus relaciones.

Los diagramas de estados muestran acciones, estados, eventos y transiciones.

Los diagramas de comunicación muestran la organización estructural de los objetos que envían y reciben señales.

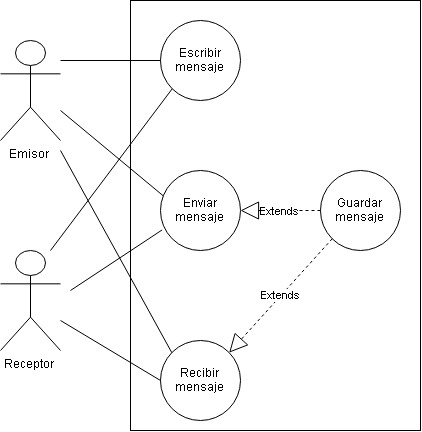
Los diagramas de interacción que muestran un conjunto de objetos y sus relaciones, incluyendo los mensajes que pueden enviarse entre estos.

Los diagramas de actividades resaltan el flujo de control y datos entre actividades.

### Casos de uso

**19)** ¿Qué es un caso de uso? ¿Para qué sirve? ¿Qué información debe contener su especificación?

### Diagrama de casos de uso

**20)** Realizar un diagrama de casos de uso en donde se puedan observar las distintas relaciones entre clasificadores. El ejemplo deberá ser distinto a cualquiera de los presentados en clase o en la bibliografía.

### Historias de usuario

**21)** Identifique y explique los componentes de una historia de usuario.

## PROMOCIÓN

### Relación

**22)** Establezca y fundamente 3 (tres) relaciones, cada una entre 2 (dos) o más conceptos estudiados en la materia, por cada tema de relación no aprobado, indicando explícitamente a cuál de ellos corresponde.

*Ninguna fundamentación puede exceder las 6 (seis) líneas, considerando una tipografía Arial de tamaño 10 sin formato adicional.*