

Nombre:	Curso:
---------	--------

DEFINICIONES Indicar a qué concepto pertenece lo descrito (3 pts).

Descripción	Indique el concepto
La interacción entre objetos está definida a través de una unidad atómica, pueden ser llamadas o invocaciones a métodos de un objeto en particular o respuestas de un objeto a otro	MENSAJE
Se empaqueta o resguarda dentro de un objeto una pieza de información (atributos) con un comportamiento específico (métodos) que actúa sobre esta información. La ventaja es limitar los efectos de cambios sobre el sistema	ENCAPSULAMIENTO
Esta etapa del Proceso Unificado incluye Funcionalidad, Priorización de los Casos de Uso, un Plan Director del Proyecto, Especificación de los Casos de Uso Análisis, Diseño, Pruebas de certificación dentro de un Proceso iterativo	ELABORACIÓN

RELACIONES Indicar los conceptos que se están relacionando (7 puntos).

Relación	Indique los conceptos relacionados
El diagrama A describe la interacción de objetos que requiere la funcionalidad de los distintos escenarios de un Caso de Uso y los objetos son representados con su ciclo de vida dentro de una serie temporal. El diagrama B muestra lo mismo que el diagrama A, sin embargo, no hay referencia a una serie temporal, su propósito es mostrar la topología del proceso distribuido entre los distintos objetos	DIAGRAMA DE SECUENCIA – DIAGRAMA DE COLABORACIÓN
Todos los agentes usarán los diagramas de UML para un proyecto en particular de desarrollo de software (primer concepto), para garantizar la consistencia entre los requerimientos o la funcionalidad y el código, y viceversa desde el código a la funcionalidad (segundo concepto)	MODELO DE REFERENCIA - TRAZABILIDAD
Se aplican distintas implementaciones a una determinada funcionalidad (primer concepto) en subclases específicas y en la clase general, de modo de transmitir la implementación general del código a las subclases y en cada subclase se agregan las implementaciones específicas o locales.	POLIMORFISMO - HERENCIA

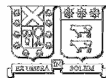
DESARROLLO explique con fundamento (10 puntos).

1. ¿Cuál es la relación entre el Implementador (agente del modelo) con el diagrama de componentes y despliegue?

El implementador es el que comprende mejor la topología de un sistema distribuido, por lo tanto, usará los diagramas de Componentes y los diagramas de despliegue para visualizar los archivos ejecutables, archivos DLL y otros componentes, revisando y visualizando la distribución del despliegue de estos componentes de software en la red (componentes de hardware).

2. Comente la siguiente frase: “Una declaración general de los objetivos es suficiente para comenzar a escribir programas, los detalles se dejan para más adelante”.

Una mala definición inicial de los requerimientos, es la principal causa de pérdida de trabajo en desarrollo de software. Esto repercutirá en todo el desarrollo del proyecto informático, ya que podría obtenerse una solución que solamente cumpla con una parte de los requerimientos y, por ejemplo, el monto de inversión deba incrementarse para cumplir con el resto de las necesidades planteadas, inclusive existirán ocasiones en las que hay que reconstruir lo ya desarrollado y por lo tanto, el impacto será mucho mayor en la etapa de construcción, en cambio si la necesidad hubiera sido detectada en la etapa de análisis o diseño, el impacto será considerablemente menor.



3. UML se integra con una metodología de desarrollo conocida como Proceso Unificado, la cual se divide en 4 fases. ¿A qué fase corresponde cada uno de los siguientes puntos y qué función juegan en cada una de las etapas: a) Definir Umbral de Riesgo, b) Pruebas de Certificación, c) Refactoring?.
- a) Definir Umbral de Riesgo – Etapa de Concepción: En la etapa de Concepción, se comienzan a definir los casos de uso, por lo tanto, se puede llegar a establecer un cronograma de tiempo y esfuerzo, razón por la cual es posible definir un umbral de riesgo primario en esta etapa.
- b) Pruebas de Certificación – Etapa de Elaboración, Construcción, Transición: Las pruebas de certificación se llevan a cabo en varias etapas, ocupando los diagramas de UML desarrollados para el proyecto en particular, como medio de certificar contractualmente el desarrollo del sistema.
- c) Refactoring – Etapa de Construcción: Generalmente, no se utilizan todas las potencialidades o prestaciones de un lenguaje la primera vez que se programa en él, por lo tanto, se requiere hacer refactoring que significa rehacer el código, optimizándolo, pero que finalmente haga la misma tarea, el cambio debiera ser transparente para el usuario.

EJERCICIO (40 puntos)

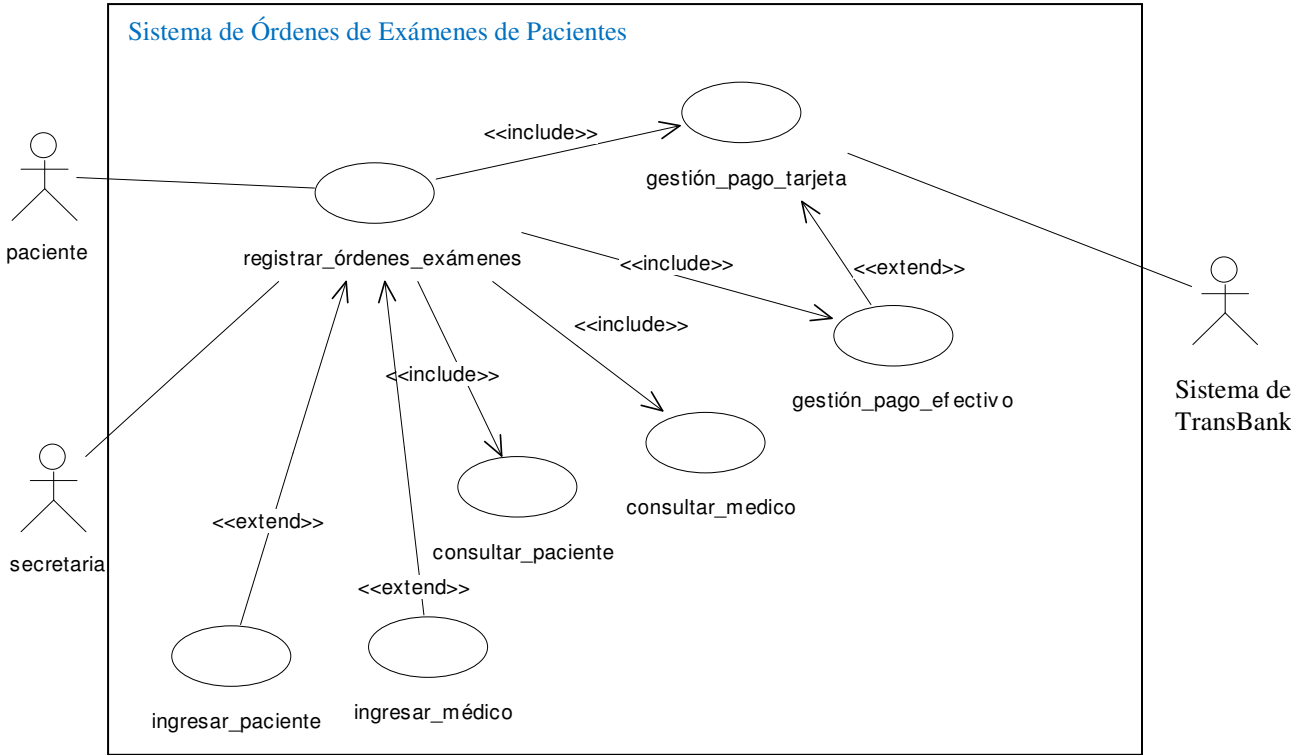
Se tiene el extracto de los siguientes casos de uso:

<p><u>Nombre CU:</u> Registrar orden de exámenes del paciente</p> <p><u>Actores involucrados:</u> paciente, secretaria</p> <p><u>Precondiciones:</u> La secretaria se identifica y autentifica en el sistema</p> <p><u>Escenario de éxito:</u> El paciente llega al laboratorio del centro médico donde se toman los exámenes. El paciente le entrega a la secretaria la orden en papel que le ha sido entregada por el médico. La secretaria inicia una nueva orden de exámenes en el sistema, esta orden se registra con un número. Se le pide el rut al paciente, para lo cual se invoca al cu_consultar_paciente para verificar si el paciente ya existe, y luego se invoca al cu_consultar_médico para verificar si existe en el sistema el médico que solicita los exámenes para esa orden. El nombre del paciente y el nombre del médico aparecen en la orden. También, se solicita la previsión (Fonasa, y todas las Isapres que existen) del paciente para registrarla en la orden. Luego, se requiere registrar el detalle de la orden de exámenes, puesto que un médico puede requerir varios exámenes en una misma orden. Completar el detalle de la orden puede ser iterativo, ya que depende de los distintos tipos de exámenes solicitados. Este detalle contiene información de los exámenes (código, descripción, días en que estará el resultado, valor del examen), se ingresa la cantidad (podrían ser varias muestras de sangre, por ejemplo), la bonificación de la previsión según el Plan del paciente y el copago que es lo que el paciente debe cancelar según la bonificación dada por la isapre. Al finalizar de ingresar todos los exámenes en el detalle, se calcula el total del pago de la orden, el paciente realiza el pago, para lo cual se invoca al cu_gestionar_pago_efectivo ó cu_gestionar_pago_orden_tarjeta. Se registran los datos de la secretaria en la orden de exámenes. Se registra la orden de exámenes con el detalle y se genera un comprobante de atención para el paciente.</p> <p><u>Escenario alternativo:</u> Si un paciente no existe, entonces se invoca a cu_ingresar_paciente. Si un médico no existe, se invoca a cu_ingresar_médico. Si un paciente no puede pagar en efectivo o tarjeta, se cancela la orden.</p> <p><u>Postcondición:</u> Se registra la orden y se tramita la toma de exámenes en el laboratorio.</p>
<p><u>Nombre CU:</u> Gestionar Pago Orden (de exámenes) con tarjeta</p> <p><u>Actores involucrados:</u> paciente, secretaria, Sistema de pago Transbank</p> <p><u>Precondiciones:</u> El paciente ha solicitado una orden de exámenes y pagará vía Transbank.</p> <p><u>Escenario de éxito:</u> La secretaria ingresa el monto, tipo de pago, fecha, paciente, secretaria, orden de examen para el registro en el sistema. El sistema solicita la autorización del pago al Sistema de pago Transbank. Luego, el paciente realiza la transacción con el dispositivo dispuesto para el Sistema de pago Transbank, el cual informa que el pago está aprobado. El sistema recibe la aprobación, la secretaria registra el pago con todos sus datos en el sistema.</p> <p><u>Escenario alternativo:</u> El Sistema de pago Transbank no funciona. Se pide al cliente otra modalidad de pago, lo cual es aceptado por el cliente, entonces se invoca a cu_gestionar_pago_efectivo. Si el cliente no acepta otra modalidad de pago, se cancela el pago y la orden.</p> <p><u>Postcondición:</u> Se registra el pago de la orden de exámenes con tarjeta.</p>



Se pide:

- 1. Hacer el diagrama de casos de uso (12)





2. Haga los diagramas de secuencia para cada caso de uso (16)

Diagrama de secuencia de Registrar Órdenes de Exámenes: (10 puntos)

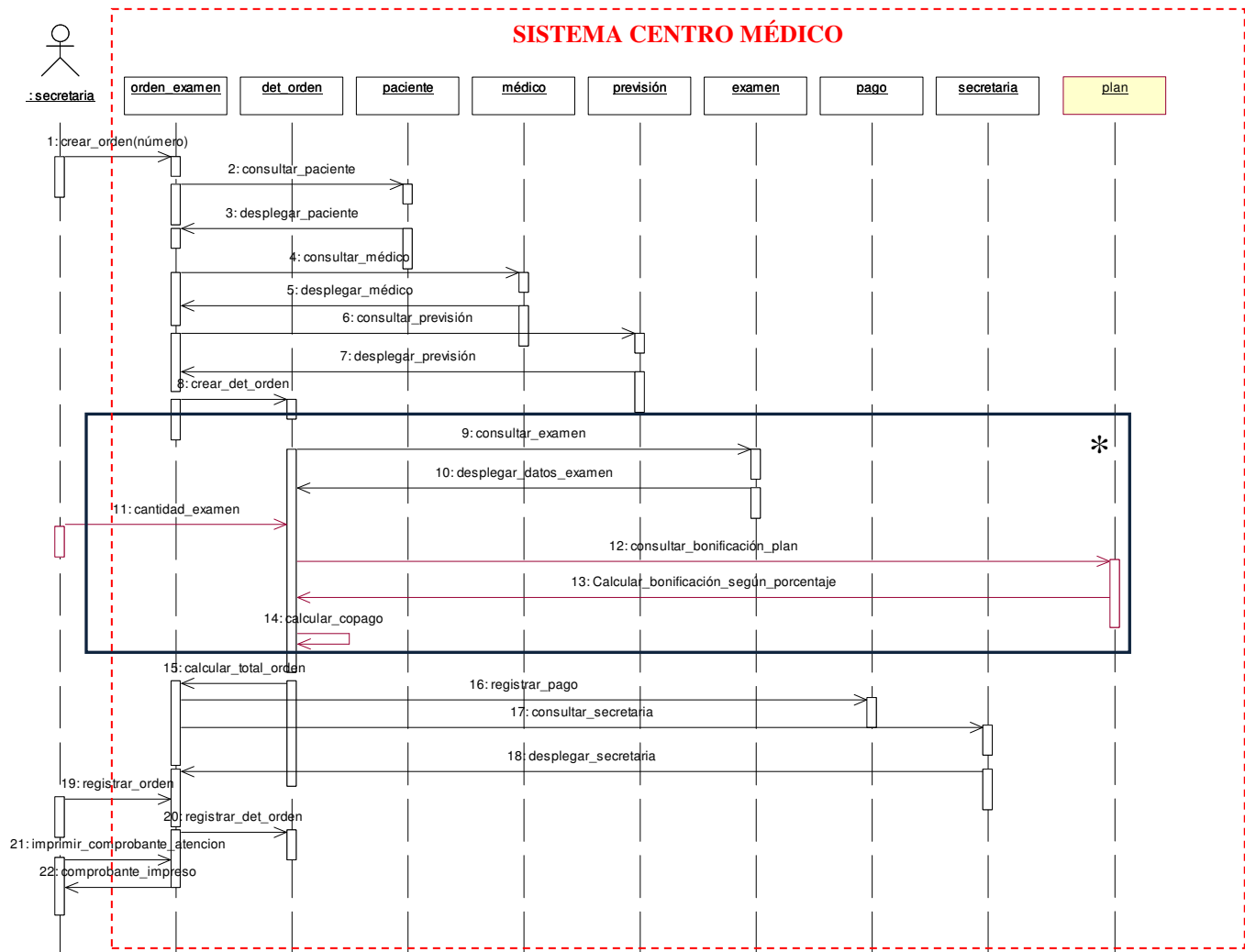
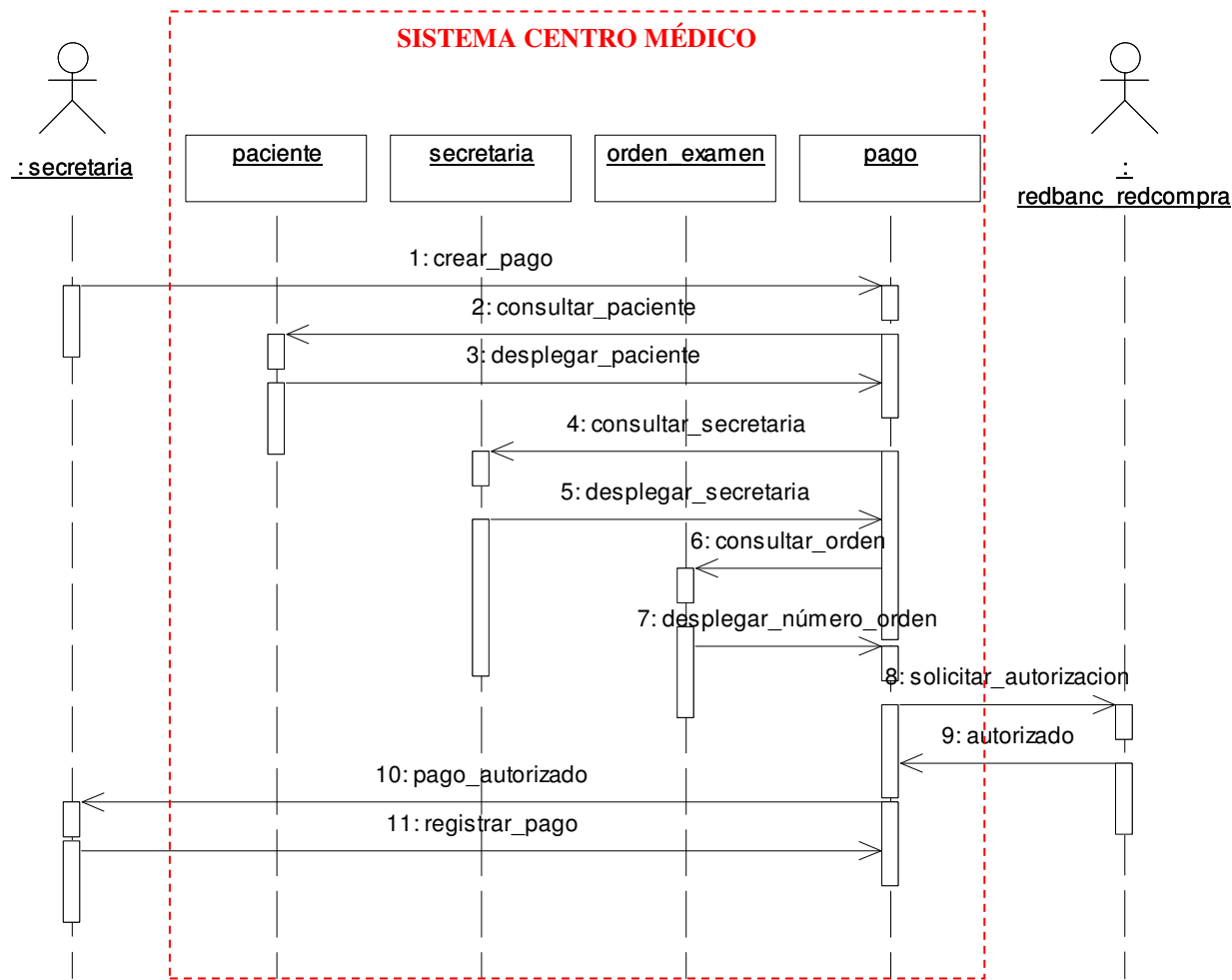


Diagrama de secuencia de Gestionar Pago de Orden (de exámenes) con tarjeta: (6 puntos)



3. Indique en un listado las clases candidatas y los posibles métodos para cada clase (12). Por ejemplo:

Clase	Métodos asociados
PACIENTE	Consultar_paciente
SECRETARIA	Consultar_secretaria
ORDEN_EXAMEN	Crear_orden Calcular_total_orden Registrar_orden (paciente, secretaria, médico, previsión, total) Imprimir_comprobante_atención Consultar_orden_examen
PAGO	Crear_pago Registrar_pago (orden, paciente, secretaria, monto, tipo_pago, fecha)
MÉDICO	Consultar_médico
PREVISIÓN	Consultar_previsión
DET_ORDEN	Crear_det_orden Calcular_bonificación_según_porcentaje Calcular_copago_por_examen Registrar_det_orden(Examen, cantidad,bonificación, copago)
PLAN_PACIENTE	Consultar_bonificacion_por_examen
EXAMEN	Consultar_examen