

La elaboración de este ejercicio está dirigida a través de una secuencia de **10 preguntas**, agrupadas en **7 secciones**, en las que se solicita realizar determinadas operaciones y tareas de diseño. **La puntuación** es sobre un **total de 11 puntos** (10 más 1 punto de la sección BP). **No todas** las cuestiones **puntúan igual**. Puede utilizar la cantidad de papel que necesite, pero conteste a las preguntas de cada sección en hojas diferentes. Por favor, lea TODO el ejercicio, hasta el final.

Se ruega al Tribunal que facilite, desde el inicio, 3 hojas adicionales para este ejercicio. Sólo se permite el uso del libro de texto original de C. Larman, no se admite ningún otro material.

¡ATENCIÓN!: este examen consiste en el diseño de la implementación de un caso de uso concreto, que se sitúa en un escenario, pero se define en la pregunta número 2. Lo que se presenta inmediatamente es el planteamiento de dicho escenario, por lo que se recomienda leerlo *transversalmente* y acudir a la definición del caso de uso, en la pregunta mencionada.

Planteamiento del escenario de estudio.

El escenario en el que se situará el funcionamiento del caso de uso (pregunta 2), consiste en un **sistema** (que llamaremos Bazınga!) **que automatiza el servicio de la tienda** en una tahona.

Para evitar a sus clientes las incómodas esperas en su tienda, la tahona 'damePantoma' (dPt) ha decidido implantar el sistema Bazınga!, que automatiza el servicio de venta.

El obrador de la tahona produce, de forma artesanal, distintas variedades de pan, principalmente, pero también empanadas y bollería. Su horario de producción es diferente al de la tienda y se desea que, mientras haya **producto** reciente, el sistema Bazınga! esté en funcionamiento; aunque la tienda esté cerrada.

Para dar este servicio, se instala un expendedor en el exterior de la tienda. En el obrador, los **productos** recién elaborados se depositan en recipientes clasificadores, separados por tipo, que abastecen tanto a los mostradores de la tienda (cuando está abierta) como a los pedidos realizados desde el expendedor; que siempre está activo.

Cuando el sistema acepta un pedido, una célula robotizada recoge los productos de cada clasificador, los empaqueta adecuadamente con papel y, mediante un sistema neumático, los coloca en la puerta de suministro del expendedor. Cuando han transcurrido 3 horas desde su producción, el sistema robótico retira los **artículos** antiguos y bien los coloca en los recipientes de tienda (si está abierta), bien los destina al reparto para otros negocios o los desestima para la venta.

El expendedor dispone del hardware necesario para presentar, con claridad y siempre que haya existencias, los **productos** en venta, para seleccionarlos y marcar la cantidad que se desea. Además, admite el **pago** en metálico (con el importe exacto, si se queda sin cambio) y con tarjeta monedero electrónico, de débito y de crédito.

Todo el hardware descrito para el funcionamiento de Bazınga! está controlado por un sistema similar a un autómata programable, que hace las veces de parte de la interfaz entre la lógica del negocio de la aplicación y su capa de presentación. Por consiguiente:

La aplicación Bazınga! es una extensión del software de gestión comercial de dPt. Además de la necesaria capacidad de integración en la aplicación matriz, se desea que tenga sus mismas prestaciones (igual a las de PdV).

Detalles y simplificaciones admitidas:

- Supóngase que todos los precios son '*por unidad*' (que no hay *venta al peso*) y que todos los artículos, de un mismo producto, cuestan y pesan lo mismo.
- Si se dan las circunstancias que motiven la devolución de algún artículo, el proceso se realiza en la tienda y durante el horario comercial.
- Se considera que Bazınga! siempre está activa. Sólo permite seleccionar productos de los que haya existencias (recientes, en su recipiente de alimentación) y, si no, no admite pedidos. Fuera del horario comercial de la tienda, y mientras el obrador esté en funcionamiento, el software de Bazınga! debe ser capaz de funcionar de manera autónoma. Sin embargo, durante el horario de la tienda, el software de gestión de dPt sí está activo, y Bazınga! debe sincronizarse y colaborar con él.
- Como en el resto de la asignatura, la atención del estudio se dirige a la capa de la lógica de la aplicación, a los objetos del dominio del negocio y los mínimos servicios técnicos de apoyo que permitan interactuar con el acceso a los datos u otros sistemas considerados externos. En este caso, la capa de presentación recoge el software de control de los sensores, actuadores y el hardware descrito; que se considerará *transparente* y la interacción entre los actores humanos y la lógica del negocio será directa (como si se tratara de una comunicación mediante lenguaje de invocación de funciones a través de comandos). Igualmente, la interacción entre la lógica del negocio y los sistemas de apoyo externos son los mismos que los de la aplicación matriz de la tienda dPt y se realizará a través de sus mismos adaptadores.

Enunciado de las preguntas.

Sección 1. Evaluación de los **Casos de Uso**

1. (0'5 puntos) Represente, en un diagrama UML de casos de uso, los casos de uso primarios más importantes, sus actores principales y de apoyo y las interacciones correspondientes, para la aplicación Bazınga!
2. (1 punto) Escriba el caso de uso <<ProcesarVenta>> en un formato completo (se recomienda la variante '*en dos columnas*') y estilo esencial. Incluya tanto el escenario principal de éxito como 2 extensiones o flujos alternativos que pudieran ser frecuentes.

Consiste en una compra, y la línea de acciones del flujo básico de éxito discurre desde que el usuario inicia un pedido en el expendedor hasta que se lo sirve, junto con el correspondiente recibo.

Supóngase que el usuario selecciona, por ejemplo, 1 pan payés, 1 broa de maíz y una empanada pequeña de atún, que paga con su tarjeta de crédito. **Aténgase, estrictamente, a esta operación.**

No escriba un encabezamiento demasiado elaborado del caso de uso (es decir, omita *propósito, resumen, antecedentes...*); en su lugar, afronte directamente el transcurso típico de los acontecimientos.

Se recuerda: de aquí en adelante, todas las preguntas se refieren a las especificaciones definidas en esta pregunta y para este caso de uso. El objetivo es que realice el diseño para que también admita las otras funcionalidades y opciones de la aplicación.

Sección 2. Evaluación del **Modelado Conceptual**

3. (2 puntos) En relación con el caso de uso anterior, <<ProcesarVenta>>, construya un Modelo de Dominio y representelo en notación UML. Represente los objetos conceptuales, las relaciones relevantes entre ellos, su cardinalidad y los atributos *candidatos* de los objetos.

Sección 3. (Diseño) Evaluación de los **Eventos del Caso de Uso**

4. **Eventos y Contratos.**

- 4.1. (2 puntos) Circunscrito al caso de uso anterior <<ProcesarVenta>>, construya un Diagrama de Secuencia (diagrama de interacción DS) en UML. Represente los actores y los eventos de los componentes del sistema para este caso de uso.

¡ATENCIÓN!: se pide un diagrama de secuencia en el que represente el paso de mensajes entre los actores y los distintos objetos del modelo, **NO** del Sistema (DSS). Por tanto, **represente las líneas de tiempo de los objetos identificados en el modelo**, **NO** las interacciones entre los actores y una única línea temporal correspondiente al objeto **sistema global**.

- 4.2. (1 punto) A partir de este DS, escriba y desarrolle los contratos de las **2 operaciones principales**, llamémoslas, '*RealizarPedido*' y '*RealizarPago*'. Usted puede llamarlas como le convenga; pero, en adelante, debe mantener esa denominación. Estas operaciones deben ser consecutivas y cubrir todo o la mayor parte del caso de uso. De otra forma **no se calificarán, ni en esta pregunta ni en las siguientes**.

Sección 4. Evaluación de la **Asignación de Responsabilidades y Diseño de Colaboraciones**

5. (1 punto) A partir del contrato de la operación <<RealizarPedido>> que haya indicado en la pregunta 4 (como la haya llamado usted), complete el diagrama de colaboración en UML. Consigne cada mensaje con los patrones GRASP (Experto, Creador, etc.) o cualquier otro que lo justifique. Si añade responsabilidades no explicitadas en el contrato (porque crea que es importante señalarlas), explíquelas brevemente.
6. (1 punto) A partir del contrato de la operación <<RealizarPago>> que haya indicado en la pregunta 4 (con la denominación que haya utilizado allí para esa operación), complete el diagrama de colaboración en UML. Consigne cada mensaje con los patrones GRASP (Experto, Creador, etc.) o cualquier otro que lo justifique. Si añade responsabilidades no explicitadas en el contrato (porque crea que es importante señalarlas), explíquelas brevemente.

Sección 5. Evaluación del **Diagrama de Clases** de diseño

7. (1 punto) Elabore un diagrama de clases para el caso de uso que se está tratando <<ProcesarVenta>> (DCD), centrado en la clase que va a implementar la responsabilidad más característica del caso de uso, la que mejor defina la naturaleza de lo que se hace en él. Represente los nombres de todos los atributos, asociaciones (con la navegabilidad) y métodos de esa clase (excepto '*setters*' y '*getters*' irrelevantes) y de las que estén directamente involucradas con ella en el caso de uso.

Sección 6. Evaluación de la **Transformación del Diseño en Código**

8. (0'5 puntos) A partir de los anteriores diagramas de clases y colaboraciones, elabore y defina la clase que haya establecido, en el desarrollo anterior, como responsable de controlar la correcta secuencia de acciones en el caso de uso <<ProcesarVenta>>. Incluya las definiciones de todas las variables que la componen (miembros), pero escriba solamente la definición completa del cuerpo para el método (o métodos) principal o más significativo: <<se omite el método>>. Ignore los pequeños detalles de sintaxis -el objetivo es evaluar la capacidad fundamental para transformar el diseño en código-. Utilice la sintaxis de Java.

ATENCIÓN: lo que hay entre signos, <<se omite el método>>, es un ejemplo, usted lo debe sustituir por el nombre que haya asignado al método principal que haya elegido.

Sección 7. Preguntas opcionales **BP**. Motivación.

9. (0'5 puntos) Indique qué principios GRASP ha utilizado en el ejercicio y qué responsabilidades ha asignado guiándose por ellos.
10. (0'5 puntos) Indique qué patrones GoF ha utilizado en el ejercicio y qué mejoras ha obtenido, con su uso, en la elaboración o en el comportamiento del desarrollo de la aplicación.

Espacio adicional para borrador o las respuestas

Hoja adicional para borrador o las respuestas

Hoja adicional para borrador o las respuestas

Hoja adicional para borrador o las respuestas

Hoja adicional para borrador o las respuestas