
	Principios y Fundamentos de la Ingeniería del Software	
	Examen convocatoria de Junio 2014	2º Grado en Ing. Informática 25/6/2013

## Maradona, Gestión de supermercado

**Duración: 2:30h**



### Visión del negocio

La empresa **Maradona** nos ha encargado la realización de un software que permita gestionar todos sus procesos de negocio. Su actividad principal es la venta al por menor de productos de consumo doméstico.

Se diferencian dos tipos de clientes, el *cliente personal* y el *cliente corporativo* (Bares, restaurantes, cafeterías, etc.). La diferencia entre los dos tipos de clientes es que el cliente corporativo tiene un contrato con la empresa y un límite de crédito, usando forzosamente la tarjeta de fidelización, mientras que el cliente personal puede pagar en efectivo, con tarjeta bancaria o con tarjeta de fidelización. Los Cajeros de la empresa se encargan de atender las peticiones de los clientes corporativos, de forma que cada Cajero se hace cargo de una cartera de clientes corporativos, y a cada cliente corporativo sólo le atiende un Cajero.

Se desea modelar el software que controle los puestos de gestión de las compras de los clientes de una cadena de supermercados.

Estos puestos permiten al Cajero:

1. Abrir caja. Al comienzo del día, se indica el saldo inicial (la cantidad de dinero metálico de que dispone el terminal).
2. Compra. Cada compra de un cliente está formado por varias líneas de pedido, cada una de las cuales se refiere a un solo producto. El cliente llega con su carro o cesta de compra y deposita los artículos en el mostrador. Se escanea un artículo (su código de barras) o se introduce su código a mano (zona 1 de ilustración 1). A continuación, se introduce el número de unidades del producto. Se actualiza automáticamente el subtotal y el stock del artículo.
3. Fin de la compra. Si no hay más artículos, se totaliza la compra y se introduce la forma de pago, que puede ser:
  - efectivo en cuyo caso se introduce el dinero que se entrega y el sistema devuelve el importe a devolver (se registra como 'cliente anónimo'),
  - tarjeta de fidelización, lo que supone un descuento del 3% en la compra.
  - tarjeta de crédito. Si usamos tarjeta, se hace una petición a un Sistema Externo de Compensación Interbancaria, que nos indicará si la operación ha sido aprobada o rechazada. En caso de rechazo, se le pide nueva tarjeta o efectivo.

Finalmente se imprime un ticket, con todas las líneas de pedido y el total (Ilustración 2).

4. Cambio de cliente. Mientras atendemos a un cliente, podemos aparcas temporalmente su pedido para atender a otro cliente, pudiendo fácilmente retomar el pedido inicial.
5. Consulta de precios y stock. De un artículo, podremos comprobar fácilmente su precio y de cuantas existencias dispone, sin tener que abandonar el pedido actual.

6. Cierre de caja. Al final del día, se totalizan las operaciones de efectivo y se nos informa del dinero que debería haber en el terminal a través de un informe.

Por otro lado, cada tienda tiene un Cajero Jefe, que además de las tareas anteriores, se encarga personalmente de:

7. Alta / Baja / Modificación de cliente. El supermercado dispone de una tarjeta de fidelidad, al que se asocia una cuenta bancaria para los cargos de las compras realizadas en el último mes. Estos clientes y su información puede ser mantenida por el puesto.
8. Informe diario, semanal y mensual de las ventas realizadas.

**Problema 1. Modelado (50% nota).** *Responda a este problema en folios continuos.*

Se le pide que analice, diseñe e implemente los siguientes entregables:

1. **Diagrama de Casos de Uso** (2 puntos)
2. **Diagrama de Clases** (2 puntos)
3. **Diagrama de secuencia del CU Venta** (1 punto)

**Problema 2. Estimación (20% nota).** *Responda a este problema y posteriores en folios continuos*

Una vez identificadas las necesidades del cliente, llega el momento de estimar la duración y esfuerzo necesarios para realizar el proyecto. Se ha especificado la siguiente interfaz de usuario:

Zona 1. Entrada de código de artículo y unidades

Zona 2. Visualización de líneas introducidas previamente

**Ilustración 1. Interfaz de usuario para Compra**

Donde el cajero tiene la opción de escanear un artículo (su código de barras) o introducir su código a mano y luego las unidades (por defecto 1), o bien cambiar a otro cliente, consultar precio/stock de un artículo sin afectar al proceso de venta o de terminar el pedido y pasar al proceso de pago.

Internamente se apoya en el siguiente fichero:

Fichero de almacenamiento de compras					
Código artículo	Unidades	Precio unidad	Subtotal	Id_Compra	Id_Linea

4. Basándose en la propuesta de interfaz, complete la siguiente tabla, para las siguientes transacciones:

- Entrada de artículos (zona 1)
- Visualización de pedido actual (zona 2)
- Fichero de almacenamiento de compra
- Ticket de compra

(1 punto)

```

CENTRO COMERCIAL MARADONA
CIF B21041629
21008 HUELVA

FACTURA SIMPLIFICADA 02040211
FECHA 14/06/2014 TPV: 02 CENTRO:061
=====18:14=====
CASERA 500 ML.      4    0.57    2.28
SPAGHETTI           1    0.91    0.91
...
PANADERIA           5    0.18    0.90
ACTIVIA CREMO.      1    1.80    1.80
MAHOU 5* LATA       6    0.56    3.36

=====IVA=====
BASE   10.03 IVA10%   1.00
BASE    7.98 IVA21%   1.67
BASE    0.87 IVA 4%   0.03
=====IVA INCLUIDO=====
TOTAL FACTURA                                21.58
EFECTIVO                                       22.00
CAMBIO                                       0.42
=====GRACIAS POR SU VISITA=====
  
```

**Ilustración 2. Prototipo de ticket de compra**

Transacción	Tipo de Componente (EE, SE, GLDI, GLDIZ Y CE)	Número de ficheros y datos elementales	Lista de datos elementales

		Tipos de datos elementales		
		1 a 5	6 a 19	20 ó más
Ficheros Referenciados	0 ó 1	S	S	M
	2 ó 3	S	M	C
	4 ó más	M	C	C

**Ilustración 3. Salidas externas**

		Tipos de datos elementales		
		1 a 4	5 a 15	16 ó más
Ficheros Referenciados	0 ó 1	S	S	M
	2	S	M	C
	3 ó más	M	C	C

**Ilustración 4. Entradas externas**

		Tipos de datos elementales		
		1 a 19	20 a 50	51 ó más
Tipos de Registros	1	S	S	M
	2 a 5	S	M	C
	6 ó más	M	C	C

**Ilustración 5. Grupo lógico de datos internos**

5. ¿Calcule el total de puntos de función no ajustados (PFNA) para las 4 transacciones? (0.5 puntos)

DESCRIPCIÓN	SENCILLA	MEDIA	COMPLEJA	TOTAL P.F.
Nº de Entradas Externas	x 3	x 4	x 6	
Nº de Salidas Externas	x 4	x 5	x 7	
Nº Grupos Lógicos de Datos Internos	x 7	x 10	x 15	
Nº de Grupos Lógicos de Datos de Interfaz	x 5	x 7	x 10	
Nº de Consultas Externas	x 3	x 4	x 6	
TOTAL PUNTOS FUNCIÓN NO AJUSTADOS (PFNA)				

Asuma un FA (Factor de ajuste) de 1.0 y si la media general de todos los proyectos está en 11,35 horas-hombre por punto-función (ISBSG).

6. ¿Cuál es el esfuerzo estimado en horas-hombre para las 4 transacciones del programa? (0.5 puntos)

### Problema 3. Construcción (10% nota).

Una vez terminados el análisis y modelado, y realizada la estimación del proyecto, empezamos a codificar. Para construir el software, uno de los primeros pasos será, a partir del **Diagrama de Clases** obtenido, implementar en Java la estructura de las clases.

7. Realice la implementación estructural en Java de las clases **Compra y Cliente** (incluya sólo la declaración de los métodos, no los implemente, ni incluya los *getters* y *setters* de los atributos). (1 punto)

### Problema 4. Pruebas (20% nota).

Uno de los métodos implementados en la clase *Compra* realiza una búsqueda binaria. A la hora de realizar las pruebas de caja blanca,

8. ¿cuántos y cuáles caminos habría que probar en el método *busquedaBinaria*? (2 puntos)

```
class BusquedaBinaria{
    /* Busca un valor numerico dentro de un arreglo numerico...
    * previamente ordenado usando el metodo de busqueda binaria
    *
    * @param arreglo con los elementos; dato a buscar
    * @return posicion del elemento buscado, en caso de no existir retorna -1 */
    public static int busquedaBinaria(int vector[], int dato){
        int n = vector.length;
        int centro,inf=0,sup=n-1;
        while(inf<=sup){
            centro=(sup+inf)/2;
            if(vector[centro]==dato) return centro;
            else if(dato < vector [centro] ){
                sup=centro-1;
            }
            else {
                inf=centro+1;
            }
        }
        return -1; //No encontrado
    }

    public static void main(String []args){
        int[]vector ={1,3,7,8,10,14,23,47,56,60,61,62,65,66,68,69,70,73,76,77,79,80,82};
        int valorBuscado = 47;
        System.out.println(busquedaBinaria(vector,valorBuscado));
    }
}
```