

Práctica 6. Diseño e implementación. Tarjetas puntos.

Objetivos

- Practicar la comprensión de un documento de requisitos apoyado por casos de uso.
- Practicar la realización de un diagrama de clases de mediana complejidad.
- Practicar el uso de atributos estáticos.
- Practicar el uso del método `equals()` para comparar strings.
- Practicar el uso de la clase `ArrayList`.

Desarrollo

Tomando como base el documento de requisitos que aparece a continuación:

1. Realizar el diseño (diagrama de clases) de la aplicación.
2. Completar la implementación del diseño realizado.

Documento de requisitos

Descripción general

Se desea realizar una aplicación que permita gestionar las tarjetas de fidelidad de una cadena de tiendas.

La tienda ofrece la posibilidad de utilizar tarjetas de fidelidad que permiten acumular puntos. Cada tarjeta se identifica mediante un código alfanumérico (“TAR1”, “TAR2”, “TAR3”, ...), asignado en el momento de su creación. Para cada tarjeta se almacena además el DNI de su propietario.

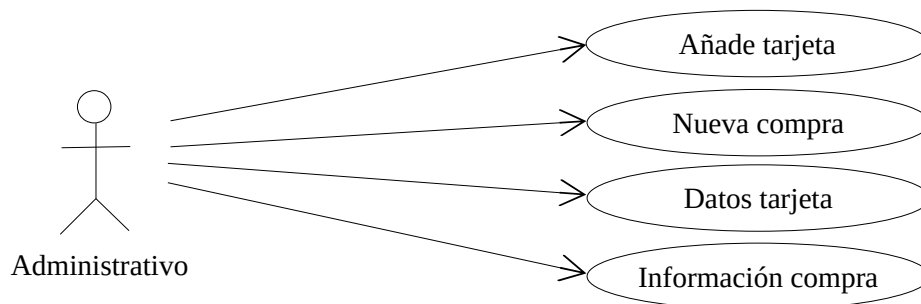
Cada vez que se realiza una compra con una tarjeta, se acumula en ella 0 o más puntos a razón de un punto por cada 10 € (0 puntos para compras inferiores a 10 €, 1 punto para compras iguales o superiores a 10 € pero inferiores a 20 €, etc.).

Las compras realizadas con cada tarjeta quedan almacenadas en un registro histórico en el orden que se realizaron.

Para cada compra se almacena el nombre del producto comprado y su importe.

Casos de uso

Las operaciones que se desea que pueda realizar el administrativo de la cadena de tienda son las descritas en los siguientes casos de uso:



Caso de uso “Añade tarjeta”:

1. El administrativo selecciona la opción “Añade tarjeta”.
2. La aplicación muestra una ventana que permite introducir el DNI del propietario.
3. El administrativo introduce el dato solicitado.
4. La aplicación registra la nueva tarjeta.

Caso de uso “Nueva compra”:

1. El administrativo selecciona la opción “Nueva compra”.
2. La aplicación muestra una ventana que permite introducir el código de la tarjeta, el producto comprado y el importe en euros de la compra.
3. El administrativo introduce los datos solicitados.
4. La aplicación registra la compra en la tarjeta correspondiente.
 - En el caso de que no exista ninguna tarjeta con ese código, se notifica el error.

Caso de uso “Datos tarjeta”:

1. El administrativo selecciona la opción “Datos tarjeta”.
2. La aplicación muestra una ventana que permite introducir el código de la tarjeta.
3. El administrativo introduce el dato solicitado.
4. La aplicación muestra los datos de la tarjeta (código, DNI y puntos acumulados).
 - En el caso de que no exista ninguna tarjeta con ese código, se notifica el error.

Caso de uso “Información compra”:

1. El administrativo selecciona la opción “Información compra”.
2. La aplicación muestra una ventana que permite introducir el código de la tarjeta y la posición de la compra buscada en el histórico de compras de la tarjeta (las posiciones de las compras empiezan en 0).
3. El administrativo introduce los datos solicitados.
4. La aplicación muestra los datos (producto e importe) correspondientes a la compra que ocupa la posición indicada en el registro histórico de la tarjeta.
 - En el caso de que no exista ninguna tarjeta con ese código, o que la posición indicada no sea válida se notifica el error.

Requisitos no funcionales

La aplicación contará con una interfaz gráfica basada en un menú del paquete “fundamentos”.

Se deberá utilizar la **clase ArrayList** para implementar la colección que almacena las tarjetas de la cadena de tiendas así como el histórico con las compras realizadas por cada tarjeta.

Se deberá utilizar un **atributo estático** para generar los códigos de las tarjetas.

Entregar

- Diagrama de clases (en papel).
- Código desarrollado: fichero comprimido con los paquetes (directorios) que contienen las clases Java.
 - Evaluación automática utilizando el servidor **<https://moodle.ctr.unican.es>** (nombre de usuario: igual que el de moodle, contraseña: DNI con letra mayúscula).
 - Entregar también en el **servidor moodle oficial** de la asignatura.