



Universidad de Oviedo

Examen práctico: Máquina de venta de entradas

Alumno: _____

Instrucciones

IMPORTANTE: una vez hayáis creado un nuevo espacio de trabajo e importado el proyecto inicial descargado, no olvidéis renombrarlo dentro de Eclipse con el siguiente formato: **apellido1_apellido2_nombre**, en minúsculas y sin acentos ni ningún otro carácter especial. Por ejemplo, una alumna llamada **María Peláez Nuño** tendría que dar a su proyecto en Eclipse el siguiente nombre: **pelaez_nuno_maria**.

Introducción

Estamos desarrollando el software que controla una máquina de venta de entradas para distintos espectáculos, como las que se encuentran en determinados sitios públicos.

La máquina, internamente, cuenta con los siguientes componentes físicos: una pantalla en la que se muestra la información, un teclado para que el usuario pueda introducir los datos que se le pidan durante el proceso de compra, el lector de tarjetas, una ranura para la inserción de billetes y monedas y, por último, una impresora con la que imprime las entradas al finalizar la compra.



Cuando un usuario se acerca a la máquina, ésta muestra en la pantalla de inicio los espectáculos disponibles. El usuario sólo tiene que seleccionar uno para iniciar el proceso de compra. Tras indicar el número de entradas que desea y mostrar el importe total, la máquina le pedirá que seleccione el modo de pago.

Actualmente se permite el pago de dos formas: en efectivo o con tarjeta de crédito.

En el primer caso, el usuario irá introduciendo en la propia máquina el dinero, hasta que se alcance una cantidad igual o mayor que el importe total, devolviendo, en su caso, la cantidad introducida de más. El usuario también puede pulsar el botón de cancelar en cualquier momento, en cuyo caso se le devolverá la cantidad introducida hasta ese momento y se finalizará sin comprar.

Para los pagos con tarjeta, tras introducir la tarjeta o acercarla al lector, la máquina leerá los datos de la misma y se comunicará con la pasarela de pagos del banco, que será la encargada de llevar a cabo la transacción, si es una tarjeta válida. En caso de que la pasarela rechace la tarjeta no se llevará a cabo la venta de las entradas.

Si por el contrario el pago se efectuó correctamente, sea a través del medio que sea, la máquina imprimirá las entradas y finalizará el proceso de compra.

Código entregado

Se proporciona un prototipo inicial donde ya se encuentra implementada toda la operativa de la máquina descrita en el punto anterior. A continuación se ofrecen algunas aclaraciones importantes que os ayudarán a comprender el diseño inicial y, sobre todo, cómo acometer correctamente las ampliaciones que se solicitan en el siguiente apartado.

En el paquete **hardware** se encuentran los controladores de los dispositivos físicos descritos. Como tales, **no se pueden modificar**.

Toda la **interacción con el usuario** deberá llevarse a cabo a través de los objetos **display** y **keyboard**, tal como se está haciendo ahora. Son la única forma que tiene la máquina de mostrar información al usuario y que éste introduzca datos, respectivamente (veréis que en algunas partes del código entregado no se está haciendo así, sino que en vez de eso se está escribiendo directamente, o leyendo, en la salida y entrada del sistema, respectivamente; eso es porque en esos casos se están **simulando** determinados procesos, pero que en una implementación real no se haría así y que no forman parte de la interfaz de usuario).

Observaréis que el código de la interfaz de usuario aparentemente está entremezclado con la lógica de negocio en la propia máquina. Pero pensad que, en este caso, estamos hablando de un dispositivo físico, donde **la propia máquina es la interfaz**. No debéis intentar arreglar esa parte, no se trata de eso (porque no hay nada que arreglar, está perfectamente bien así en este caso).

Por último, se proporcionan, en un par de paquetes aparte, **ruralvia** y **paypal**, el código cliente de la pasarela de pagos necesario para los pagos con tarjeta y de la API de PayPal, respectivamente, para la ampliación que se pedirá a continuación. No es código

nuestro y, por tanto, de nuevo, **no podemos modificarlo**. Únicamente tenéis que comprender cómo funcionan dichas clases para usarlas donde corresponda.

Qué se pide

Una vez analizado el código proporcionado y ejecutado el programa para comprender cómo funciona, deberéis llevar a cabo las siguientes ampliaciones.

Añadir un nuevo medio de pago: PayPal

Ahora se desea ofrecer también la posibilidad de pagar mediante PayPal. Para ello, cuando el usuario escoja dicho medio de pago, se le deberán pedir su nombre de usuario y contraseña de PayPal, y utilizar el cliente proporcionado (**PayPalAPI**) para realizar el pago, de forma similar a como se describe a continuación.

```
// mostrar en pantalla "Iniciando sesión en PayPal..."
// pedir nombre de usuario y contraseña por pantalla
String username = ...
String password = ...
String token = paypal.logIn(username, password);
if (!paypal.isLoggedIn(token))
    // mostrar "Nombre de usuario o contraseña incorrectas" y finalizar
// si el nombre de usuario y contraseña son correctos, PayPal nos
// habrá devuelto un token válido, que usaremos para realizar el pago
boolean isValid = paypal.checkout(token, amount);
```

Una vez realizado el rediseño necesario para permitir los pagos mediante PayPal, deberían poder añadirse en un futuro **nuevos medios de pago** sin necesidad de hacer **ningún cambio en la máquina**.

Añadir descuentos

Más adelante, aparece un nuevo requisito: se quiere permitir que, durante el proceso de compra, tras indicar el número de entradas y mostrar el importe a pagar, y antes de seleccionar el medio de pago, el usuario pueda introducir un código promocional que dé lugar a determinados descuentos en la cantidad a pagar.

Específicamente, se pide poder añadir los siguientes tipos de descuento:

- Una **cantidad fija** (por ejemplo, 15 euros), que se restará del importe total a pagar. En este caso hay que contemplar la posibilidad de que el descuento pudiera ser superior al

importe, en cuyo caso se descontará la cantidad total a pagar (es decir, nunca nos debería quedar un importe negativo, sino en todo caso cero).

- Un **porcentaje** (por ejemplo, el 10 %) sobre el importe total.
- Una **entrada gratis**. En este caso se descontará, del total de la compra, el importe correspondiente al precio de una entrada.

Como se puede observar, todos los descuentos lo que hacen es restar una determinada cantidad del importe total a pagar. La diferencia es cómo se calcula.

Por supuesto, como el diseño de software no consiste en adivinar el futuro, y esto era algo que no se podía prever, para implementar esta nueva funcionalidad no quedará más remedio que modificar el código de la máquina para que durante el proceso de pago se solicite el código promocional (sólo se puede introducir uno). En caso de que no se tenga ningún código o que no quiera aplicarlo, bastará con que el usuario deje dicho campo en blanco. Si, por el contrario, se introduce uno, se aplicará el descuento correspondiente y se mostrará por pantalla el nuevo importe resultante antes de proceder a efectuar el pago (o un mensaje indicando que dicho código no es válido, según corresponda).

Una vez hecho eso, para probar la nueva funcionalidad, se configurará la máquina con los siguientes códigos promocionales y sus respectivos descuentos asociados:

- «**15EURACOS**», un descuento fijo de 15 euros sobre el importe de la compra.
- «**VERANO20**», un descuento del 10 % sobre el importe de la compra.
- «**FREE1**», que descontará el precio de una entrada del total a pagar.

Naturalmente, una vez añadida esta funcionalidad, el diseño resultante deberá ser tal que se puedan añadir y eliminar códigos promocionales y sus respectivos descuentos asociados sin necesidad de tocar el código de la máquina.

Identificar los patrones y cambios realizados

En cada una de las clases e interfaces que se hayan visto cambiadas con respecto a la versión inicial, así como en las que hayan aparecido nuevas, se deberá indicar, mediante comentarios en el propio código fuente, qué cambios se han realizado con respecto al código inicial proporcionado y por qué (si ha sido durante el rediseño inicial, para añadir nuevos medios de pago o para los descuentos), y qué se consigue con ello.

En caso de que dichos cambios supongan algún patrón de diseño, indicar cuál es, en esos mismos comentarios, especificando qué papel desempeña esa clase o interfaz en la estructura del patrón, así como cuáles de sus métodos se corresponden con los equivalentes de la estructura genérica del patrón correspondiente.