 Proyecto Cupi2	ISIS-1204 Algorítmica y Programación **Descripción**
Ejercicio:	
Autor:	
Fecha:	

Enunciado

Se quiere crear una aplicación (`miDiscoTienda`) para el manejo de la información de una tienda virtual de canciones en formato MP3. Esta aplicación debe permitir visualizar un catálogo de discos y canciones que la tienda tiene a la venta. Cada disco tiene un nombre, un artista, un género y una imagen de la carátula del disco. La tienda se especializa en la venta de canciones en formato MP3 por lo que ofrece toda la información relevante de una canción al usuario. Esta información comprende: el nombre de la canción, su precio individual, la duración en minutos y segundos, el tamaño en *megabytes* (MB) y la calidad de la canción expresada en *kilobytes* por segundo (Kbps).

Se espera que `miDiscoTienda` cuente con la siguiente funcionalidad: (1) presentar al usuario la información de una canción del catálogo (dados el nombre del disco y el nombre de la canción), (2) agregar un disco al catálogo, el cual estará sin canciones inicialmente, (3) agregar una canción a un disco (dados el nombre del disco y toda la información de la canción), (4) vender una canción a un usuario. En este caso se debe teclear la dirección electrónica del usuario para que le sea enviado por correo el respectivo archivo MP3 con la canción. Por cada venta, el programa debe crear un archivo de texto (en el directorio "`data\facturas`"), cuyo nombre debe ser parte de la dirección electrónica del usuario (lo que va antes del símbolo "@") seguido de un número único generado por el programa. Dicho archivo también se envía por correo electrónico al usuario. El programa debe llevar el registro del número de copias vendidas de cada canción. Por último, (5) el programa debe permitir procesar un archivo completo de pedidos de un cliente. En este caso se hace una sola factura.

La información de la tienda debe ser persistente y el proceso debe ser completamente transparente para el usuario. Esto quiere decir que el programa debe ser capaz de guardar la información en un archivo cada vez que el usuario termina la ejecución del mismo y de utilizar dicha información cuando el usuario vuelve a ejecutarlo para reconstruir el estado del modelo del mundo. El programa no debe preguntarle al usuario el nombre del archivo, sino que lo tiene que manejar todo internamente.

En caso de cualquier error en la ejecución del programa, éste debe presentar una caja de diálogo con un mensaje claro que explique la razón del problema. La siguiente es una lista de los errores que se pueden presentar y el tipo de acción que debe realizar el programa:

Errores que se pueden presentar	Acción del Programa
Al intentar agregar un nuevo disco, el programa se da cuenta de que ya existe un disco con ese título.	Mensaje al usuario con el error. El nuevo disco no es agregado.
Al intentar agregar una nueva canción a un disco, el programa se da cuenta que ya existe una canción con ese nombre.	Mensaje al usuario con el error. La nueva canción no es agregada.

Los datos dados por el usuario no son válidos (el precio no es un valor numérico, o el nombre de la canción es vacío).	Mensaje al usuario con el error. El nuevo disco o la nueva canción no son agregadas.
Al intentar escribir el archivo con la factura, hay un error de entrada / salida.	Mensaje al usuario con el error. La venta se hace de todos modos y se envía la canción al usuario sin la factura.
Al intentar leer el archivo de pedidos, no lo encuentra en el sistema de archivos.	Mensaje al usuario con el error.
Durante la lectura del archivo de pedidos, hay un error de entrada / salida.	Mensaje al usuario con el error. La venta se hace hasta donde se haya podido leer el archivo. Se genera una factura parcial con la venta.
Al intentar leer el archivo de pedidos, el programa se da cuenta que éste no tiene el formato pedido.	Mensaje al usuario con el error. Si el error impide la venta global, se anula todo el proceso. Si no, se venden las canciones que tengan el formato pedido, y en la factura se señala cuáles canciones no se pudieron incluir en la venta por un problema de formato.
Al intentar cargar el estado inicial del modelo del mundo, hay un error de entrada / salida.	No se debe comenzar la ejecución del programa y debe aparecer un mensaje de error al usuario.
Al intentar salvar el estado final del modelo del mundo, hay un error de entrada / salida.	No debe dejar que el programa termine su ejecución y debe presentar un mensaje de error al usuario pidiéndole que lo resuelva.
Al hacer una venta, el usuario no suministra una dirección electrónica válida.	Mensaje al usuario con el error. La venta no se hace y no se emite ninguna factura.

Persistencia

1. Factura de venta:

El formato de una factura de venta es el siguiente:

```
miDiscotienda – FACTURA
Fecha:            < día de la semana > < mes > < día del mes > < hora > < año >
Email:            < correo del cliente >
Canción:          < nombre de la cancion > - < nombre del artista >. < nombre del
                  disco >
No de Canciones:  < cantidad de canciones vendidas >
Valor Total:      < valor de la factura >
```

Ejemplo:

Se muestra a continuación como sería una factura válida generada por la aplicación:

```
miDiscotienda – FACTURA
Fecha:            Thu Jul 20 12:02:19 COT 2006-08-06
Email:            mvduque@uniandes.edu.co
Canción:          Rivers of Babylon – Boney M. GOLD
No de Canciones:  1
Valor Total:      2500.00
```

2. Archivo de pedidos:

El formato de un archivo con los pedidos de un cliente es el siguiente:

```
< correo del cliente >
< nombre del disco >#< nombre del artista >#< nombre de la canción >
```

Cada línea del archivo, a excepción de la primera, representa la información de una canción que el usuario desea comprar.

Ejemplo:

Se muestra a continuación como sería el archivo de un pedido válido:

```
mimail@uniandes.edu.co
Confessions on a Dance Floor#Madonna#Sorry
Push the Button#The Chemical Brothers#The Boxer
```

Interfaz

Interfaz de usuario de la tienda