

Introducción

En este documento se desarrolla las actualizaciones realizadas a la práctica realizada en la primera unidad de trabajo del primer trimestre la cual consistía en crear una interfaz gráfica simulando la comandera que utiliza un camarero, con la que crea comandas que luego mandaría a la cocina, la cocina no está implementada.

Índice

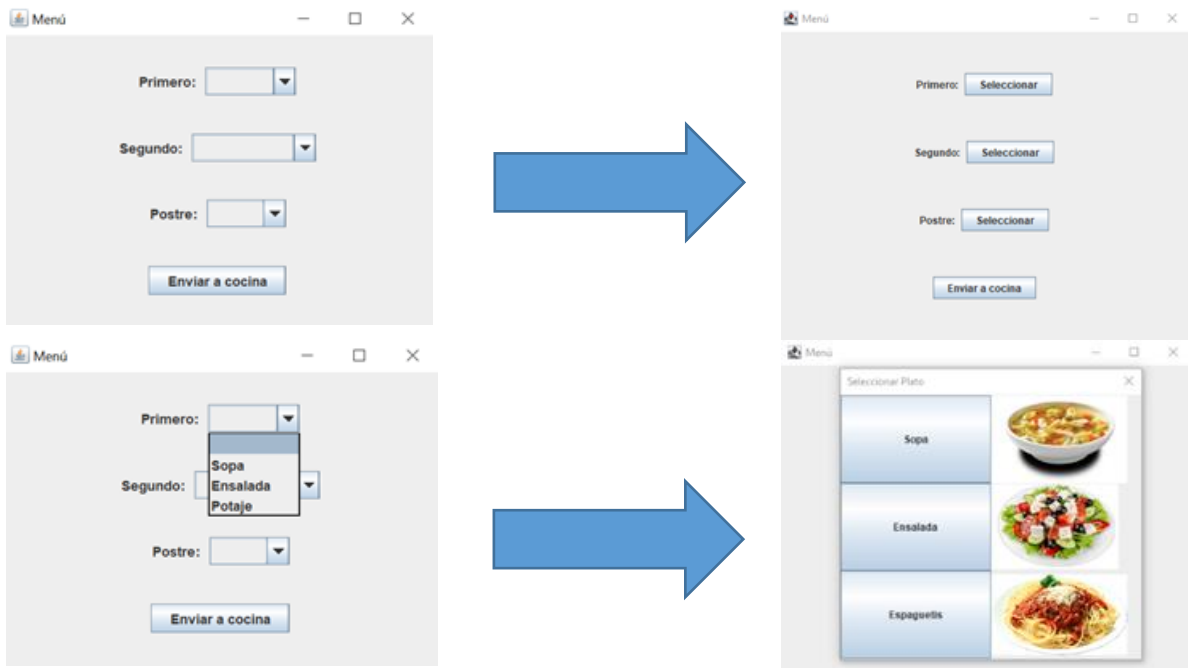
Pautas de diseño.....	2
Pauta 1	2
Pauta 2	2
Pauta 3	2
Almacenamiento de comandas.....	3
Directorio “Comandas”	3
El documento XML	4
Lectura de comandas	4
Corolario	4

Pautas de diseño

En este apartado revisaremos las pautas de diseño que cumple la aplicación.

Pauta 1

La primera pauta de diseño es referente al menú, hemos eliminado las opciones en cascada para sustituirlos por un cuadro de dialogo, el cual dependiendo de la opción que haya sido seleccionada mostrará unos platos u otros.



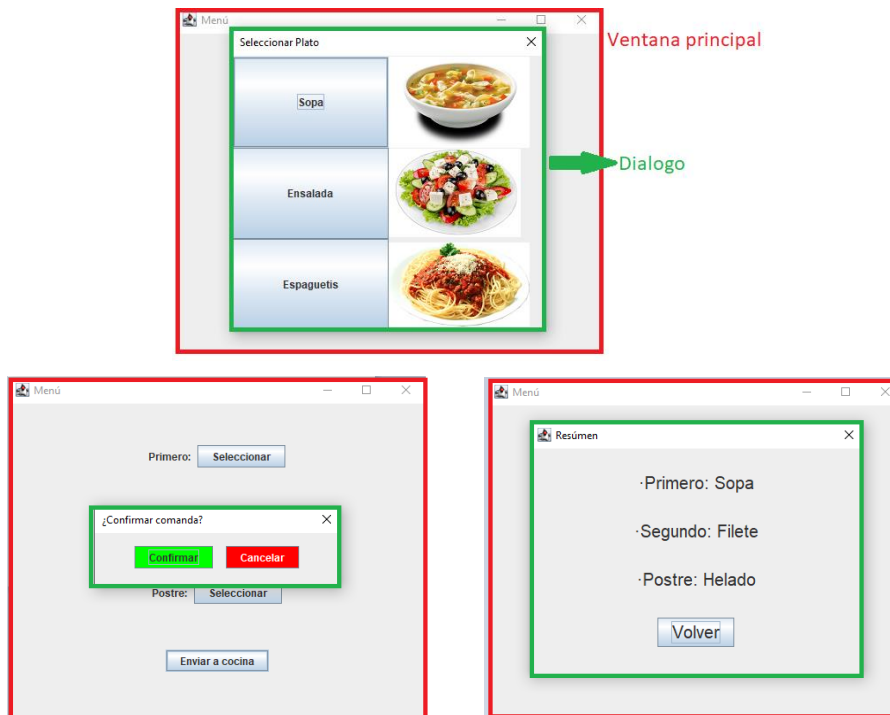
Pauta 2

Se han añadido imágenes para aumentar la claridad de los textos.



Pauta 3

La tercera pauta de diseño es evitar que se superpongan unas ventanas de otras. Para evitar que un diálogo oculten totalmente la ventana principal se ha reducido su tamaño. Su posición siempre será el centro de la ventana principal. También se cumple que no tenemos nunca más de tres ventanas simultáneas, como mucho se tienen dos.

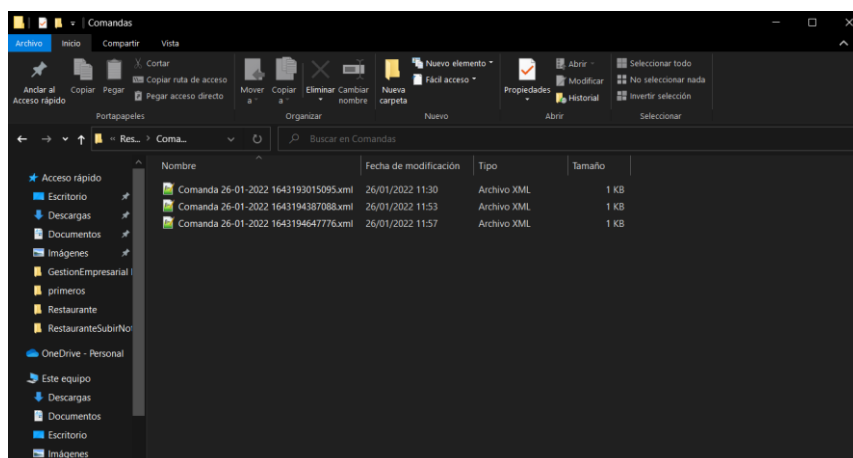


Almacenamiento de comandas

Para dotar de persistencia al desarrollo se ha creado una clase externa, llamada “GuardarComanda”, que se encargara de guardar mediante un FileOutputStream los datos con la estructura xml que se describe a continuación.

Antes de nada comprobará si existe el directorio “Comandas” para que si no es así lo cree. Cada comanda será guardada en ficheros independientes, para acercarnos lo más posible a la realidad, dado que en un restaurante no se tiene una comanda con todas si no que se van guardando cada comanda independientemente.

Directorio “Comandas”



GuardarComanda nombrará al fichero con la siguiente estructura:

Comanda **fecha actual** **identificador**.xml

- **Fecha actual.** Para facilitar la organización posterior.
- **Identificador.** Para evitar que haya dos ficheros con el mismo nombre le añadiremos este identificador el cual es generado con la fecha y hora actual convertida en milisegundos para que así sea imposible que haya dos ficheros con el mismo nombre.

El documento XML

Estructura:

```
<?xml version="1.0" encoding="UTF-8"?>
<comanda fecha="26-01-2022 11:30:15">
  <primero>Sopa</primero>
  <segundo>Filete de pollo</segundo>
  <postre>Natillas</postre>
</comanda>
```

Consta de un elemento raíz "comanda" el cual tendrá un atributo "fecha", con la fecha y hora correspondiente a cuando se ha realizado la comanda, y tres elementos hijos "primero", "segundo" y "postre" que contendrán los platos que el usuario ha seleccionado.

Lectura de comandas

Ahora revisaremos como se realiza la lectura de comandas mediante la clase "Lectura".

La clase "Lectura" está planteada para ejecutarse por comandos dado que recibe la ruta del directorio "Comandas" en el primero de estos. Primero se comprobará que el directorio recibido es correcto para que si no es así se lance una excepción. Si el directorio es correcto se hará un listado de este que será almacenado.

Posteriormente se recorrerá dicha lista para ir leyendo cada fichero. Antes de intentar leer un fichero se comprobará mediante la siguiente expresión regular si cumple el formato con el cual se guardan las comandas para solo cuando este se cumpla se lea el fichero.

Expresión regular \longrightarrow `^Comanda.*xml$`

Solo si el fichero empieza por la palabra "Comanda" y tiene una extensión xml será leído.

Las comandas serán impresas por la consola de la siguiente manera:

```
Comanda 26-01-2022 1643199221286.xml
<?xml version="1.0" encoding="UTF-8"?>
<comanda fecha="26-01-2022 13:13:41">
  <primero>Sopa</primero>
  <segundo>Filete</segundo>
  <postre>Helado</postre>
</comanda>
Comanda 26-01-2022 1643199227898.xml
<?xml version="1.0" encoding="UTF-8"?>
<comanda fecha="26-01-2022 13:13:47">
  <primero>Ensalada</primero>
  <segundo>Paella</segundo>
  <postre>Tarta</postre>
</comanda>
```

Nombre del fichero

Contenido del fichero

Corolario

Este desarrollo ha tenido una dificultad media, dado que si se los pasos a seguir eran bien estructurados y el desarrollo realizado previamente fue realizado para que fuera flexible a los cambios se realizaba con cierta comodidad.

El mayor problema que ha surgido ha sido la forma de guardar los datos en el documento xml, ya que si se quiere guardar todo en un mismo documento hay que ir desplazando el elemento raíz teniendo cuidado también de que nuestro fichero no llegue a una dimensiones demasiado grandes pasado el tiempo.