

TRABAJO PRACTICIO 3 – Laboratorio II

Nombre: Anzor, Maia María Luz - División n° 2



NOODLES

Delivery de Comida Asiática

Introducción de la aplicación

Este programa representa la simulación de una aplicación con el fin de gestionar la venta y el envío de comida asiática.

Al iniciarse el programa se podrá observar su menú, el cual posee las opciones de realizar un pedido, o realizar algún cambio sobre los pedidos ya realizados anteriormente.

Al realizar un pedido, esta opción llevara al ingreso de datos del cliente, donde podrá elegir el menú de comida que desee.



(Posee lista con datos precargada, la misma se podrá ver al ingresar a las “opciones de pedidos “esta misma se encuentra en:

tps_laboratorio_ii\TP3\Anzor.Maia.2A.TP3\TP3\bin\Debug\net 5.0-windows)

Registrar Pedido

1 Datos del cliente

Nombre:

Apellido:

Dirección:

Número Teléfono:

Email:

2 Tipo de comida

China	Coreana	Japonesa
<input type="radio"/> Pollo Kung Pao \$1200	<input type="radio"/> Bibimbap \$1200	<input type="radio"/> Udon \$1500
<input type="radio"/> Cerdo agridulce \$1600	<input type="radio"/> Tteokbokki \$1450	<input type="radio"/> Gyudon \$1350
<input type="radio"/> Rollito primavera \$400	<input type="radio"/> Bulgogi \$1200	<input type="radio"/> Tempura \$1600

Total a pagar: \$ 0

Finalizando el ingreso de datos, a continuación, se podrá elegir el tipo de envío que desee el usuario; donde también se podrá ver el detalle de cada tipo de envío.

Al finalizar el ingreso de datos

elegidos por el usuario, se mostrará nueva ventana donde se podrá ver la facturación total realizada hasta el momento.

The image shows a software interface for shipping services. The main window is titled "Despachar" and contains a "Servicio de Envio" section with three radio button options: "Envio Basico", "Envio Estandar", and "Envio Premium". To the right, there is a "Cliente nro: # 1000" label and a "Informacion del Plan" section with a large empty box. A "Finalizar" button is located at the bottom right of the main window. Overlaid on the right side is a smaller window titled "Listado completo de facturacion" with a close button (X). This window displays two customer records with their details and a total price. At the bottom of this window, it shows "PLAN SELECCIONADO: Estandar" and an "OK" button.

Servicio de Envio Cliente nro: # 1000

Plan de Envio

- ☐ Envio Basico
- ☐ Envio Estandar
- ☐ Envio Premium

Informacion del Plan

Finalizar

Listado completo de facturacion X

CLIENTE NUMERO: #1257
NOMBRE: Aylén
APELLIDO: Zaragoza
DIRECCION: San Juan 1030
NUMERO DE TELEFONO: 1169557420
EMAIL: aylen@gmail.com
TIPO Y NOMBRE DE COMIDA/S:
Coreana-Tteokbokki
China-Pollo Kung Pao
PRECIO TOTAL: \$4250

CLIENTE NUMERO: #1784
NOMBRE: Jose
APELLIDO: Martinez
DIRECCION: Callao 1010
NUMERO DE TELEFONO: 1152003412
EMAIL: jose@gmail.com
TIPO Y NOMBRE DE COMIDA/S:
Coreana-Tteokbokki
PRECIO TOTAL: \$1450

PLAN SELECCIONADO: Estandar

OK

En las opciones de pedido es donde se podrá modificar un pedido que de la lista o cancelar; Donde se podrá hacer también un backup de los datos con tres tipos de extensión distintos.

Lista de Clientes

Listado de pedidos

CLIENTE NUMERO: #1257
 NOMBRE: Aylen
 APELLIDO: Zaragoza
 DIRECCION: San Juan 1030
 NUMERO DE TELEFONO: 1169557420
 EMAIL: aylen@gmail.com
 TIPO Y NOMBRE DE COMIDA/S:
 Coreana-Tteokbokki
 China-Pollo Kung Pao
 PRECIO TOTAL: \$4250

CLIENTE NUMERO: #1784
 NOMBRE: Jose
 APELLIDO: Martinez
 DIRECCION: Callao 1010
 NUMERO DE TELEFONO: 1152003412
 EMAIL: jose@gmail.com
 TIPO Y NOMBRE DE COMIDA/S:
 Coreana-Tteokbokki
 PRECIO TOTAL: \$1450

Menu de opciones

Modificar Pedido

Cancelar Pedido

Tipos de Backup

Extension TXT

Extension XML

Extension JSON

Salir

Al intentar modificar algún pedido de la lista, se debe ingresar el numero de cliente a cambiar con los nuevos datos para actualizar.

Modificacion del cliente

Ingrese numero de cliente

(Ejemplo: 2014)

Nuevos datos del cliente

Nombre: Apellido:

Direccion: Numero Telefono:

Email:

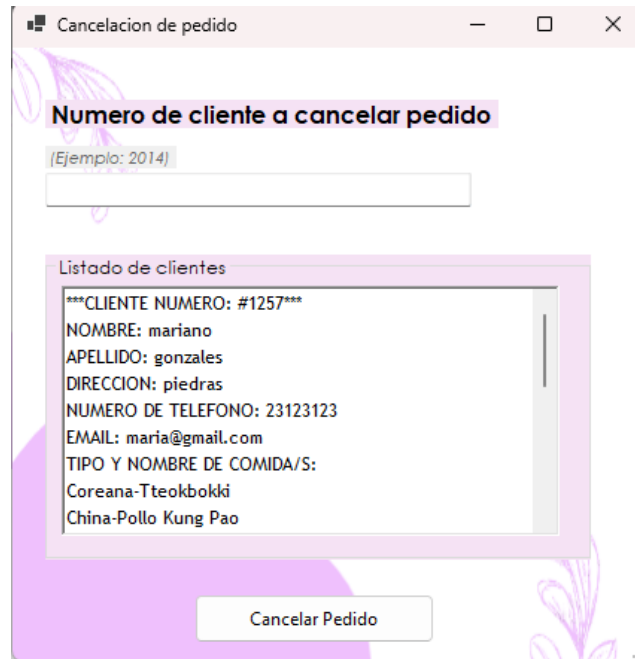
Modificar

Listado de clientes

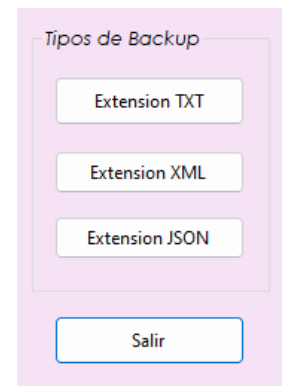
CLIENTE NUMERO: #1257
 NOMBRE: Aylen
 APELLIDO: Zaragoza
 DIRECCION: San Juan 1030
 NUMERO DE TELEFONO: 1169557420
 EMAIL: aylen@gmail.com
 TIPO Y NOMBRE DE COMIDA/S:
 Coreana-Tteokbokki
 China-Pollo Kung Pao
 PRECIO TOTAL: \$4250

CLIENTE NUMERO: #1784
 NOMBRE: Jose
 APELLIDO: Martinez
 DIRECCION: Callao 1010
 NUMERO DE TELEFONO: 1152003412
 EMAIL: iose@gmail.com

En la opción de cancelado, se abrirá una nueva ventana donde se cancelara un pedido según el número de cliente.



Las opciones de guardado se podrán realizar con tres tipos de extensiones: .json - .txt - .xml.
(Los archivos guardados se encuentra en la carpeta:



tps_laboratorio_ii\TP3\Anzor.Maia.2A.TP3\TP3\bin\Debug\net 5.0-windows)

Temas utilizados en este programa:

- Archivos
- Serialización

- Tipos Genéricos
- Interfaces
- Pruebas unitarias
- Excepciones

Archivos:

Se usa este tema para hacer un tipo de guardado de datos(.txt).

Implementa también una interfaz, a partir la cual brindara los métodos para realizar la lectura de guardado de datos en archivos txt.

(Ejemplo del evento encargado de escribir en un archivo txt)

```
private void btnBackUp_Click(object sender, EventArgs e)
{
    //Escritura de archivo .txt
    string ruta = $"{AppDomain.CurrentDomain.BaseDirectory}" + @"BackupListaPedidos.txt";

    if (this.unRegistro.HacerBackUp(this.Contenido, ruta))
    {
        MessageBox.Show("Backup con extension TXT realizado con exito", "Backup finalizado");
    }
}
```

Serialización:

Hará la lectura y escritura con extensión Json y Xml al momento de elegir el tipo de guardado deseado.

Estos métodos se encuentran en la clase encargada de esto.

```
//serializacion de archivo json
1 referencia
public bool SerializarJson(string ruta, List<Cliente> lista)
{
    bool retorno = true;
    string serializado = String.Empty;
    try
    {
        using (StreamWriter sw = new StreamWriter(ruta))
        {
            serializado = JsonSerializer.Serialize(lista);
            sw.WriteLine(serializado);
            retorno = true;
        }
    }
    catch (ExceptionControllers e)
    {
        throw new ExceptionControllers("Excepcion controlada...", e);
    }
    return retorno;
}
```

```
//Serializacion de archivo xml
4 referencias
public bool SerializarXml(string ruta, List<Cliente> lista)
{
    bool retorno = true;
    try
    {
        using (StreamWriter sw = new StreamWriter(ruta))
        {
            XmlSerializer xmlSerializador = new XmlSerializer(typeof(List<Cliente>));
            xmlSerializador.Serialize(sw, lista);
        }
        retorno = true;
    }
    catch (ExceptionControllers e)
    {
        throw new ExceptionControllers("Excepcion controlada...", e);
    }
    return retorno;
}
```

Tipos Genéricos e Interfaces:

Aplicado en la interfaz, la cual brinda los métodos para leer y escribir archivos de extensión txt.

Este mismo se implementa en la clase.

```
3 referencias
public class Registro : IBackup<string>
{
    private List<Cliente> listaClientes;
    private Cliente cliente;
    private string path = null;
}
3 referencias | 1/2 pasando
```

```
public interface IBackup<T>
{
    1 referencia
    string Ruta { get; set; }

    2 referencias
    bool HacerBackup(string contenido, string ruta);
    1 referencia
    T LeerArchivo(string ruta);
}
```

Excepciones:

Aplicado en el proyecto para captura posibles errores que puedan ocurrir en el programa.

Las excepciones mostraran un mensaje al aparecer.

Ejemplo de una excepción aplicada en caso de haber error.

```
public bool ClientesIguales()
{
    bool retorno = false;
    foreach (Cliente cliente in this.unRegistro.Lista)
    {
        if (cliente.NroCliente == this.txtBoxNroClienteBuscado.Text)
        {
            this.unRegistro.Lista.Remove(cliente);
            retorno = true;
            break;
        }
        else
        {
            throw new ExcepcionRetornoFalse("Excepcion controlada");
        }
    }

    return retorno;
}
```