

El programa creado se usa para la gestión de cortes de una fabrica textil, la cual puede fabricar solo las marcas implementadas.

(1 corte de ropa son muchas prendas, generalmente salen en curvas parejas)

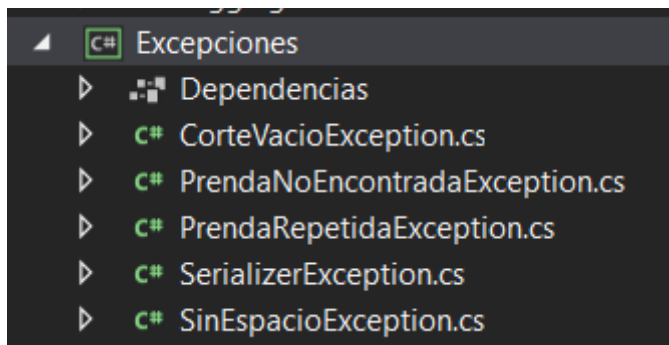
En las funcionalidades que podemos encontrar tenemos el ABM y también exportar e importar en un formato serializado, el archivo es creado y tomado desde el escritorio.

La fabrica tiene una capacidad de fabricar la cantidad de metros de tela disponible.

***ACLARACION:** Una vez creado el corte si se elimina el corte no afectara a la tela disponible ya que no se puede volver a usar la tela que ya se confecciono.

Temas de clases implementados:

Clase 15 – Excepciones: Posee 5 excepciones las cuales se detallan a continuación, también atrapa excepciones no controladas por si ocurre algún error no esperado.



Clase 16 – Test Unitarios: Se realizaron 3 pruebas diferentes y corrieron con éxito

- 1) Que se instancien las listas de la fabrica
- 2) Al agregar un corte nuevo superar el limite y que lance una excepción
- 3) Que agregue bien un corte

Clase 17 – Tipos genéricos: En la clase para guardar el archivo se implemento ya que puede recibir un tipo de dato genérico para luego generalizar el que sea.

```
4 referencias
public class ImpExpXML<T> : ISerializer<T>
{
    // ...
}
```

Clase 18 – Interfaces: En la clase indumentaria y en el proyecto de serilizer se implementó este tema, indicándole la estructura que debe usar

```

13 referencias
interface IPrenda
{
    int Id { get; set; }
    9 referencias
    int S { get; set; }
    9 referencias
    int M { get; set; }
    9 referencias
    int L { get; set; }
    9 referencias
    int XL { get; set; }
    9 referencias
    int XXL { get; set; }

    3 referencias
    enum Prenda Tipo { get; set; }

    4 referencias
    enum Marca Marca { get; set; }

    4 referencias
    string Modelo { get; set; }

    8 referencias
    string Variante { get; set; }
}

1 referencia
interface ISerializer<T>
{
    2 referencias
    bool Guardar(string path, T datos);

    2 referencias
    bool Leer(string path, out T datos);
}

```

Clase 19 – Archivos y serialización: En el proyecto Serializer esta implementado todo en formato XML

```

4 referencias
public bool Guardar(string path, T data)
{
    XmlTextWriter writer = null;
    XmlSerializer serializer = null;
    try
    {
        writer = new XmlTextWriter(path, Encoding.UTF8);
        writer.Formatting = Formatting.Indented;
        serializer = new XmlSerializer(typeof(T));
        serializer.Serialize(writer, data);
        return true;
    }
    catch (Exception ex)
    {
        throw new SerializerException(ex);
    }
    finally
    {
        if (writer != null)
        {
            writer.Close();
        }
    }
}

3 referencias
public bool Leer(string path, out T data)
{
    XmlTextReader reader = null;
    XmlSerializer serializer = null;
    try
    {
        reader = new XmlTextReader(path);
        serializer = new XmlSerializer(typeof(T));
        data = (T)serializer.Deserialize(reader);
        return true;
    }
    catch (Exception ex)
    {
        throw new SerializerException(ex);
    }
    finally
    {
        if (reader != null)
        {
            reader.Close();
        }
    }
}

```

Clase 21/22 – Bases de datos: En la clase acceso a datos.

```

public List<Indumentaria> LoadFromBD()
{
    List<Indumentaria> lista = new List<Indumentaria>();

    try
    {
        this.comando = new SqlCommand();
        this.comando.CommandType = CommandType.Text;
        this.comando.Connection = this.conexion;
        comando.CommandText = "SELECT * FROM Cortes ORDER BY Id";
        this.conexion.Open();
        SqlDataReader odr = comando.ExecuteReader();

        while (odr.Read())
        {
            if (odr["Tipo"].ToString() == "Joggings")
            {
                int Id = odr.GetInt32(0);
                int S = odr.GetInt32(1);

                Joggings corte1 = new Joggings(odr.GetInt32(0), odr.GetInt32(1), odr.GetInt32(2), odr.GetInt32(3), odr.GetInt32(4));
                if (odr["Variante"].ToString() == "Con punio")
            }
        }
    }
}

```

Clase 23 – Hilos: En la clase del frmPrincipal se aplica un hilo para refrescar cada 500 ms los datos.

```

public void ActualizarGrilla()
{
    try
    {
        while (this.pauseThread == false)
        {
            Thread.Sleep(500);

            if (this.dtGdViewCortes.SelectedRows.Count > 0)
            {
                rowSelected = this.dtGdViewCortes.SelectedRows[0].Index;
            }

            if (this.dtGdViewCortes.InvokeRequired)
            {
                this.dtGdViewCortes.BeginInvoke((MethodInvoker)delegate ()
                {
                    dtGdViewCortes.DataSource = null;
                    this.dtInd = _acceso.LoadDTFromBD();

                    dtGdViewCortes.DataSource = this.dtInd;
                    configurarGrilla();

                });
            }
        }
    }
    catch (Exception)
    {
    }
}

```

Clase 24 – Eventos y delegados: En la clase AccesoDatos tiene un evento para corroborar la tela disponible, en caso de que no sea suficiente lanza una excepcion

```

private void SinTela(object sender, EventArgs e)
{
    if (sender is double)
    {
        if (_acceso.TelaDisponibleBD() < (double)sender)
        {
            throw new TelaInsuficienteException();
        }
    }
}

```

```

21 private DataTable dtInd;
22 private AccesoDatos _acceso;
23 Thread hiloActualizador;
24 private bool pauseThread = false;
25 private int rowSelected = -1;
26 /// <summary>
27 /// Constructor por defecto
28 /// </summary>
29 1 referencia
30 public frmPrincipal()
31 {
32     InitializeComponent();
33     this.FB = new Fabrica();
34     this._acceso = new AccesoDatos();
35     this.dtInd = new DataTable();
36     this.dtInd = _acceso.LoadDTFromBD();
37     this.lstInd = new List<Indumentaria>();
38     this.btnImport.Click += new System.EventHandler(this.Importar);
39     this.FormClosing += new System.Windows.Forms.FormClosingEventHandler(this.Cerrar);
40
41     //asocio el evento
42     this._acceso.SinTelaEvent += SinTela;
43
44     this.hiloActualizador = new Thread(this.ActualizarGrilla);

```

```

public delegate void Delegado(object sender, EventArgs e);
public event Delegado SinTelaEvent;

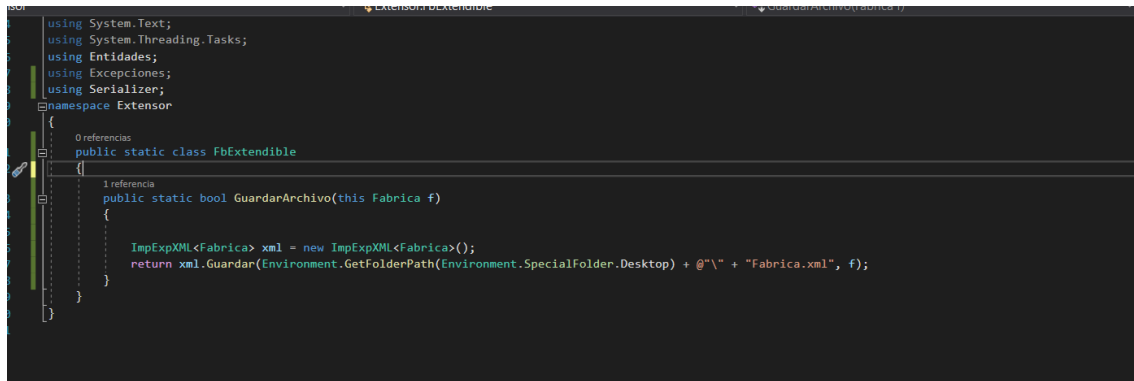
```

```

if (this.SinTelaEvent != null)
{
    this.SinTelaEvent(metrosTela, EventArgs.Empty);
}

```

Clase 25 – Métodos de extensión: En el proyecto extensor, se extiende la clase fabrica para guardar el archivo .



USE [master]

GO

/***** Object: Database [Santoliquido_TP4_BD] Script Date: 18/7/2021 22:22:32 *****/

CREATE DATABASE [Santoliquido_TP4_BD]

CONTAINMENT = NONE

ON PRIMARY

(NAME = N'Santoliquido_TP4_BD', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL15.SQLEXPRESS\MSSQL\DATA\Santoliquido_TP4_BD.mdf' , SIZE = 8192KB , MAXSIZE = UNLIMITED, FILEGROWTH = 65536KB)

LOG ON

(NAME = N'Santoliquido_TP4_BD_log', FILENAME = N'C:\Program Files\Microsoft SQL Server\MSSQL15.SQLEXPRESS\MSSQL\DATA\Santoliquido_TP4_BD_log.ldf' , SIZE = 8192KB , MAXSIZE = 2048GB , FILEGROWTH = 65536KB)

WITH CATALOG_COLLATION = DATABASE_DEFAULT

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET COMPATIBILITY_LEVEL = 150

GO

IF (1 = FULLTEXTSERVICEPROPERTY('IsFullTextInstalled'))

begin

EXEC [Santoliquido_TP4_BD].[dbo].[sp_fulltext_database] @action = 'enable'

end

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET ANSI_NULL_DEFAULT OFF

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET ANSI_NULLS OFF

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET ANSI_PADDING OFF

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET ANSI_WARNINGS OFF

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET ARITHABORT OFF

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET AUTO_CLOSE OFF

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET AUTO_SHRINK OFF

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET AUTO_UPDATE_STATISTICS ON

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET CURSOR_CLOSE_ON_COMMIT OFF

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET CURSOR_DEFAULT GLOBAL

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET CONCAT_NULL_YIELDS_NULL OFF

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET NUMERIC_ROUNDABORT OFF

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET QUOTED_IDENTIFIER OFF

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET RECURSIVE_TRIGGERS OFF

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET DISABLE_BROKER

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET AUTO_UPDATE_STATISTICS_ASYNC OFF

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET DATE_CORRELATION_OPTIMIZATION OFF

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET TRUSTWORTHY OFF

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET ALLOW_SNAPSHOT_ISOLATION OFF

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET PARAMETERIZATION SIMPLE

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET READ_COMMITTED_SNAPSHOT OFF

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET HONOR_BROKER_PRIORITY OFF

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET RECOVERY SIMPLE

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET MULTI_USER

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET PAGE_VERIFY CHECKSUM

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET DB_CHAINING OFF

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET FILESTREAM(NON_TRANSACTED_ACCESS = OFF)

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET TARGET_RECOVERY_TIME = 60 SECONDS

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET DELAYED_DURABILITY = DISABLED

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET ACCELERATED_DATABASE_RECOVERY = OFF

GO

ALTER DATABASE [Santoliquido_TP4_BD] SET QUERY_STORE = OFF

GO

USE [Santoliquido_TP4_BD]

GO

/***** Object: Table [dbo].[Cortes] Script Date: 18/7/2021 22:22:33 *****/

SET ANSI_NULLS ON

GO

SET QUOTED_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Cortes](

[Id] [bigint] IDENTITY(1,1) NOT NULL,

[S] [int] NULL,

[M] [int] NULL,

[L] [int] NULL,

[XL] [int] NULL,

[XXL] [int] NULL,

[Tipo] [varchar](50) NULL,

[Marca] [varchar](50) NULL,

[Modelo] [varchar](50) NULL,

[Variante] [varchar](50) NULL,

CONSTRAINT [PK_Cortes] PRIMARY KEY CLUSTERED

(

[Id] ASC

)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]

) ON [PRIMARY]

GO

/***** Object: Table [dbo].[MetrosTela] Script Date: 18/7/2021 22:22:33 *****/

SET ANSI_NULLS ON

GO

SET QUOTED_IDENTIFIER ON

GO

CREATE TABLE [dbo].[MetrosTela](

[Id] [bigint] IDENTITY(1,1) NOT NULL,

[MetrosTela] [float] NULL,

```
CONSTRAINT [PK_MetrosTela] PRIMARY KEY CLUSTERED
(
    [Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
USE [master]
GO
ALTER DATABASE [Santoliquido_TP4_BD] SET READ_WRITE
GO
```