

	GUÍA DE TRABAJO N° 11 - C# + SQL SERVER		
	Educación Media Fortalecida SED/SENA	Programación de Software GRADO 11	
	Ing. Néstor Raúl Suarez Perpiñan Página 1 de 16		

TEMA: DISEÑO Y GENERACIÓN DE REPORTES EN LENGUAJE C# Y SQL SERVER

OBJETIVO:

- ✓ Adquirir los conocimientos necesarios para diseñar y generar reportes utilizando lenguaje C# y bases de datos de SQL Server

I. CONCEPTOS BASICOS

En el ámbito de la informática, los reportes son informes que organizan y exhiben la información contenida en una base de datos, su función es aplicar un formato determinado a los datos para mostrarlos por medio de un diseño atractivo y que sea fácil de interpretar por los usuarios. El reporte, de esta forma, confiere una mayor utilidad a los datos, no es lo mismo trabajar con una planilla de cálculos con cientos de campos que con un dibujo en forma de torta que presenta dichos campos de manera gráfica. En ese orden de ideas los reportes tienen diversos niveles de complejidad, desde una lista o enumeración hasta gráficos mucho más desarrollados. Según el programa informático y la base de datos utilizada, los reportes permiten la creación de etiquetas, la elaboración de facturas, informes detallados, operaciones matemáticas entre otras tareas. Gracias a los reportes los usuarios de las aplicaciones pueden proceder a realizar un resumen de datos o a clasificar estos en grupos determinados. Así, teniendo en cuenta los datos que abordan y la extensión que tienen, estos reportes se pueden luego clasificar en diversos niveles en cuanto a su complejidad dando lugar a catálogos, a listas, a resúmenes, etc

Los informes se utilizan a menudo para presentar una introducción global, en la que se resaltan los principales hechos y tendencias y se tiene la posibilidad de agrupar y ordenar los datos para que el conjunto tenga visualmente más sentido constituye una ventaja clave de los informes. Los informes permiten a los usuarios obtener rápidamente una idea global, por ejemplo, en un informe de ventas se pueden agrupar las ventas por región, vendedor y trimestre, mostrando subtotales así como el total. Si bien los informes permiten embellecer los datos, agrupar y presentarlos de muchas maneras distintas, no alteran los datos subyacentes en las tablas de la base de datos. Los informes combinan datos y diseño que incluye el uso de unos determinados controles. Algunos controles, como los cuadros de texto, muestran datos, otros son elementos decorativos, como las líneas de separación o los logotipos de las empresas o compañías.

	GUÍA DE TRABAJO N° 11 - C# + SQL SERVER		
	Educación Media Fortalecida SED/SENA	Programación de Software GRADO 11	
	Ing. Néstor Raúl Suarez Perpiñan Página 2 de 16		

Un informe puede componerse de páginas y títulos, una página a su vez puede contener varios títulos que agrupan datos, o un título puede incluir datos que ocupan varias páginas. En la plataforma Microsoft .Net se cuenta con varias alternativas o posibilidades para diseñar y generar reportes entre las que se destacan SQL Server Reporting Services (Herramienta de SQL Server para crear, publicar y administrar informes) , reportes locales (que se basan en archivos de definición de informe de cliente (.rdlc)) y el denominado Crystal Reports (componente de Visual Studio enfocado a creación y administración de reportes), esta última es una de las opciones para reportes usada con frecuencia en aplicaciones desarrolladas en plataforma .Net

Crystal Reports ha formado parte de Visual Studio desde principios de los años 90, y ahora es el estándar de elaboración de informes de Visual Studio. Se incluye en todas las copias de Visual Studio Professional y se integra directamente en el entorno de desarrollo. Crystal Reports para Visual Studio incorpora la posibilidad de crear contenido interactivo con calidad de presentación al entorno de Windows. Con Crystal Reports para Visual Studio se puede crear informes complejos y profesionales en un programa basado en GUI, los cuales se pueden conectar a casi diversos tipos de orígenes de base de datos, así como a un conjunto de resultados (por ejemplo, un ADO.NET DataSet).

Los asistentes del diseñador de GUI le permiten establecer fácilmente los criterios de formato, agrupamiento y gráficos, etc. De nuestros reportes. Los informes generados con Crystal Report se puede almacenar tanto en una aplicación Web como en una aplicación para Windows, usando los controles de visualización para Visual Studio. La presentación de informes, tanto en clientes Windows como en HTML es muy interactiva y proporciona funciones como la profundización en gráficos, la exploración de informes y la búsqueda de texto. Crystal Reports para Visual Studio incluye un SDK extenso que puede ser utilizarlo para interactuar con el informe mediante programación en tiempo de ejecución, usando uno de los cuatro modelos de objetos posibles:

- ✓ CrystalReportViewer, modelo de objetos más sencillo.
- ✓ ReportDocument, modelo de objetos con más opciones que el primero.
- ✓ ReportClientDocument, el modelo de objetos más completo. Este modelo de objetos está disponibles con Crystal Reports 2008 o con un servidor especializado (RAS).
- ✓ InfoObject, un modelo de objetos muy eficaz para la programación y configuración de informes en el marco de Crystal Reports Server o BusinessObjects Enterprise.

	GUÍA DE TRABAJO N° 11 - C# + SQL SERVER		
	Educación Media Fortalecida SED/SENA	Programación de Software GRADO 11	
	Ing. Néstor Raúl Suarez Perpiñan Página 3 de 16		

Los modelos de objetos CrystalReportViewer y ReportDocument están disponibles con todas las versiones de Crystal Reports para Visual Studio. Los modelos de objetos ReportClientDocument e InfoObject requieren una licencia actualizada.

Los informes Crystal Report se pueden relacionar con el proyecto de Visual Studio de muchas formas como por ejemplo:

- ✓ Incrustando los informes directamente en el proyecto.
- ✓ Obteniendo acceso a ellos externamente, desde un directorio de archivos.
- ✓ Obteniendo acceso a ellos como servicio Web de informes desde un servidor remoto.
- ✓ Conéctandolos como Crystal Services.
- ✓ Mediante la actualización a Crystal Reports Server puede tener acceso a ellos a través del servidor RAS.
- ✓ Mediante la actualización a BusinessObjects Enterprise, puede tener acceso a ellos a través de Page Server o del servidor RAS incluidos en el marco de BusinessObjects Enterprise.

Crystal Reports se utiliza para diseñar un informe que permita a los usuarios profundizar en un gráfico y filtrar información, en función de las necesidades. Ese informe puede ser incluido en un proyecto web o tipo Windows y después mostrado en un formulario por medio de un control CrystalReportViewer que se enlaza a ese informe. Finalmente se puede interactuar con el informe mediante el uso del modelo de objetos ReportDocument.

CREAR UN REPORTE CRYSTAL REPORTS

Independientemente de cómo se hará la presentación, el reporte se define en un archivo ".RPT", que forma parte de su proyecto. Ese archivo especifica el modo en que se hará la presentación del reporte, pero no incluye los datos propiamente dichos. Los datos se definirán en el tiempo de ejecución. Crystal Reports se basa en un concepto muy común de la definición de reportes: los reportes se componen de "secciones", y cada sección es un espacio horizontal en la página.

Existen diversos tipos de secciones, por ejemplo:

- ✓ Encabezado del reporte, aparece en la primera página.
- ✓ Encabezado de página, aparece en la parte superior de todas las páginas.
- ✓ Detalle: lo más importante. En el tiempo de ejecución, se genera una de estas secciones

	GUÍA DE TRABAJO N° 11 - C# + SQL SERVER		
	Educación Media Fortalecida SED/SENA	Programación de Software GRADO 11	
	Ing. Néstor Raúl Suarez Perpiñan Página 4 de 16		

- ✓ para cada línea (registro) en la base de datos.
- ✓ Pie de página, aparece en la parte inferior de todas las páginas.
- ✓ Pie de reporte, aparece al final del reporte.
- ✓ En un principio, todos los reportes presentan estas cinco secciones. Otras secciones son opcionales, como por ejemplo: Encabezado de grupo, aparece antes de un grupo, Pie de grupo, aparece al final del grupo.

Se podrá omitir una o más secciones, en caso de que no sean necesarias

PRESENTAR UN REPORTE CRYSTAL REPORT EN LA GUI

Una vez definido el reporte, hay que presentarlo en tiempo de ejecución. Para ello, se debe colocar un componente "CrystalReportViewer" y seleccionar la propiedad "Acoplar" para "Llenar". Sólo se necesita un componente "Visualizador", aun cuando se hayan creado diversos reportes en archivos RPT separados. Para presentar un reporte, se ha de agregar un componente "Menú principal" y luego se ha de añadir un elemento de menú para abrir el formulario reporte. Finalmente se debe añadir un código similar al que se muestra a continuación al evento que corresponda según el diseño establecido:

// Ejecuta la consulta a la base datos

```
sqlDataAdapter1.Fill(dataSet11);
```

// Crea el reporte

```
CrystalReport1 Rel = new CrystalReport1();
```

// Asocia el conjunto de datos con el reporte

```
Rel.SetDataSource(dataSet11);
```

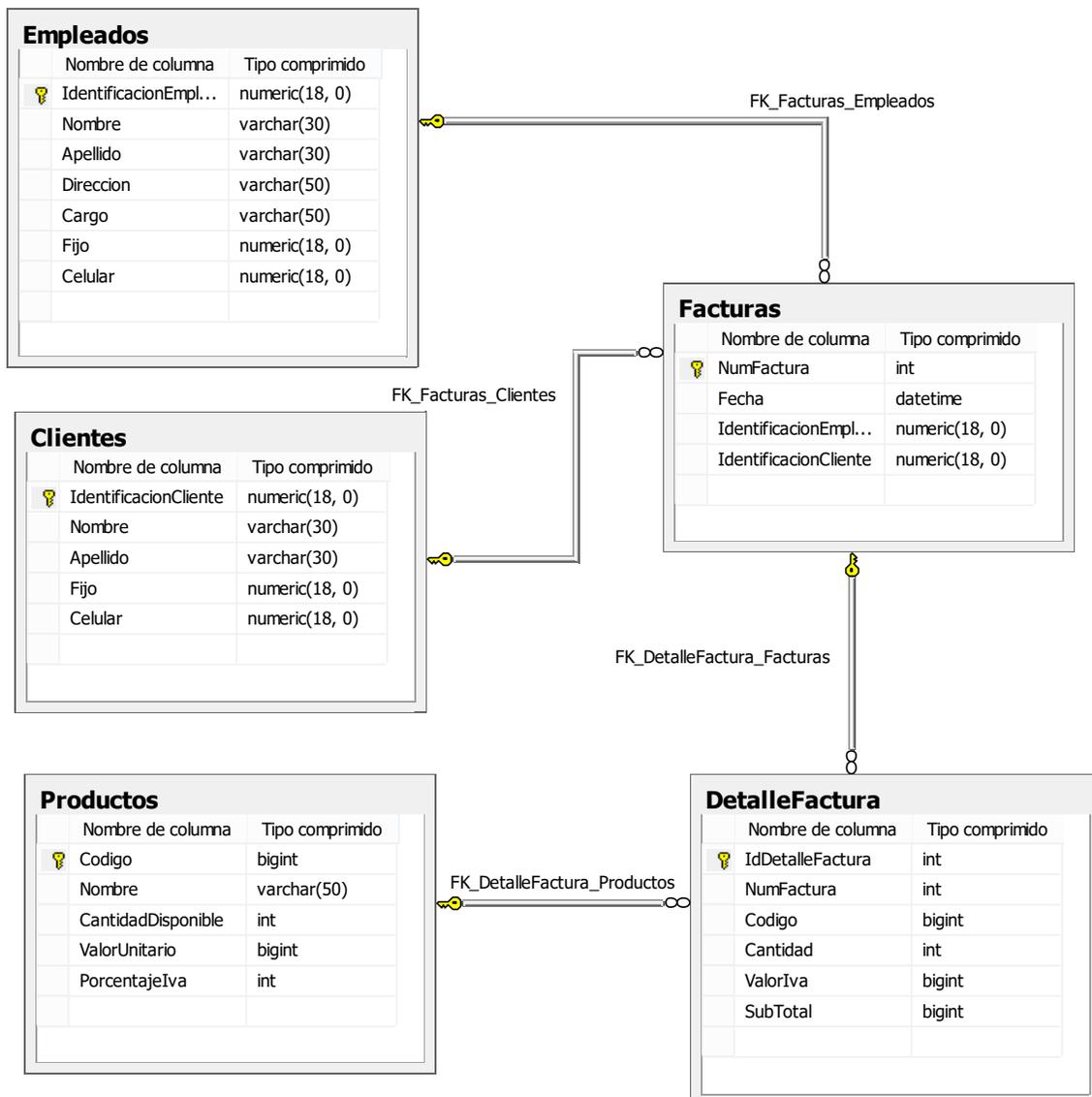
// Asocia el reporte con el Visualizador

```
crystalReportViewer1.ReportSource = Rel;
```

EJERCICIO: REPORTES CON C# Y SQL SERVER

Importante: Este ejercicio sobre diseño y generación de reportes toma como base el ejemplo de facturación trabajado anteriormente en clase. Si su proyecto SENA tiene módulo de facturación puede desarrollar este ejercicio directamente en el.

PASO 1: Cree en SQL SERVER una nueva base de datos llamada “BDDEjemploFactura” con la distribución de tablas y campos que se muestran a continuación:



	GUÍA DE TRABAJO N° 11 - C# + SQL SERVER		
	Educación Media Fortalecida SED/SENA	Programación de Software GRADO 11	
	Ing. Néstor Raúl Suarez Perpiñan Página 6 de 16		

PASO 2: Inserte 5 productos diferentes en la tabla “Productos”, inserte 5 clientes diferentes e inserte 5 empleados diferentes.

PASO 3: Inserte 5 registros en la tabla “Factura” y agréguele 3 productos a cada una en la tabla DetalleFactura.

PASO 4: Crear una Vista llamada “**Vista_DetalleFactura**” con la distribución Tabla.Campo que se muestra a continuación: (DetalleFactura.NumFactura, DetalleFactura.Codigo, Productos.Nombre AS Producto, Productos.ValorUnitario, DetalleFactura.Cantidad, DetalleFactura.ValorIva, DetalleFactura.SubTotal). El resultado de esta vista debe quedar con la distribución de campos similar a como se observa en la siguiente imagen de ejemplo:

Vista_DetalleFactura:

NumFactura	Codigo	Producto	ValorUnitario	Cantidad	ValorIva	SubTotal
1	4565467	Camiseta	20000	2	6400	46400
1	4546777	Jean	50000	5	40000	290000
3	4546777	Jean	50000	1	8000	58000
4	4546777	Jean	50000	1	8000	58000
4	4565467	Camiseta	20000	1	3200	23200
5	4546777	Jean	50000	1	8000	58000
5	4565467	Camiseta	20000	4	12800	92800

PASO 5: Crear una Vista llamada “**Vista_Facturas**” con la distribución Tabla.Campo que se muestra a continuación: (Facturas.NumFactura, Facturas.Fecha, Facturas.IdentificacionEmpleado, Empleados.Nombre + ' ' + Empleados.Apellido AS Empleado, Facturas.IdentificacionCliente, Clientes.Nombre + ' ' + Clientes.Apellido AS Cliente, Clientes.Fijo, Clientes.Celular). El resultado de esta vista debe quedar con la distribución de campos similar a como se observa en la siguiente imagen de ejemplo:

Vista_Facturas:

NumFactura	Fecha	IdentificacionEmpleado	Empleado	IdentificacionCliente	Cliente	Fijo	Celular
1	26/07/2016 0:0...	321	Klrar Ken	3	Lionel Messi	333333	3110000000
2	26/07/2016 0:0...	321	Klrar Ken	3	Lionel Messi	333333	3110000000
3	26/07/2016 0:0...	321	Klrar Ken	3	Lionel Messi	333333	3110000000
4	21/08/2016 0:0...	321	Klrar Ken	1	Pedro Picapiedra	111111	3000000000
5	21/08/2016 0:0...	321	Klrar Ken	4	Luis Suarez	222222	3150000000

PASO 6: Crear un desencadenador(trigger) que permita actualizar automáticamente la Cantidad Disponible en la tabla Productos cada vez que se inserte o actualice un registro en la tabla

	GUÍA DE TRABAJO N° 11 - C# + SQL SERVER		 Servicio Nacional de Aprendizaje
	Educación Media Fortalecida SED/SENA	Programación de Software GRADO 11	
	Ing. Néstor Raúl Suarez Perpiñan Página 7 de 16		

DetalleFactura. Para crear el trigger solicitado abra una ventana de nueva consulta y digite el código que se muestra a continuación:

```
CREATE TRIGGER ActualizarInventario
```

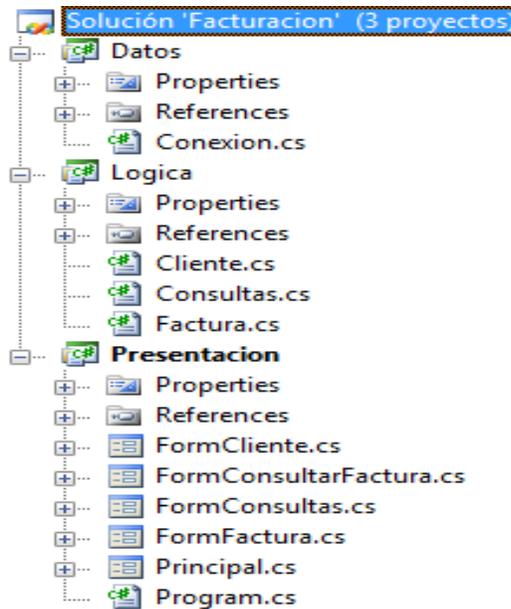
```
ON DetalleFactura
AFTER INSERT, UPDATE
AS
```

```
DECLARE
@Codigo int,
@CantidadVendida int,
@CantidadDisponible int,
@NuevaCantidad int
```

```
SET @Codigo= (Select Codigo From inserted);
SET @CantidadVendida= (Select Cantidad From inserted);
SET @CantidadDisponible = (Select CantidadDisponible From Productos where
Codigo = @Codigo);
```

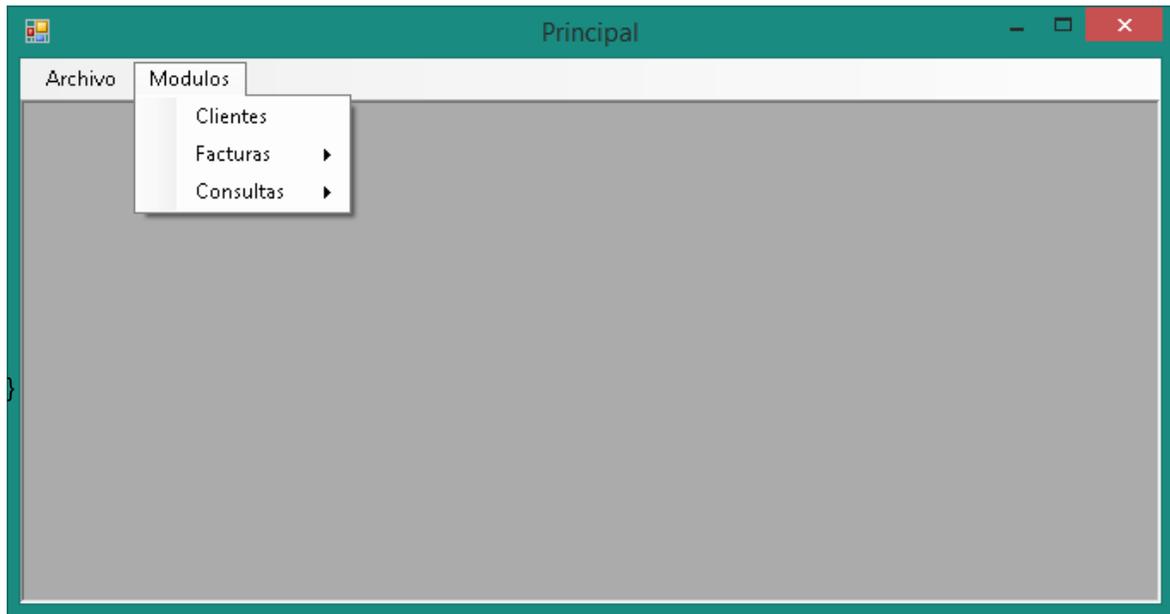
```
SET @NuevaCantidad = @CantidadDisponible - @CantidadVendida
Update Productos Set CantidadDisponible = @NuevaCantidad where Codigo =
@Codigo
```

PASO 7: Cree un proyecto tres capas distribuido con las clases y formularios tal cual como se muestra en la siguiente imagen:

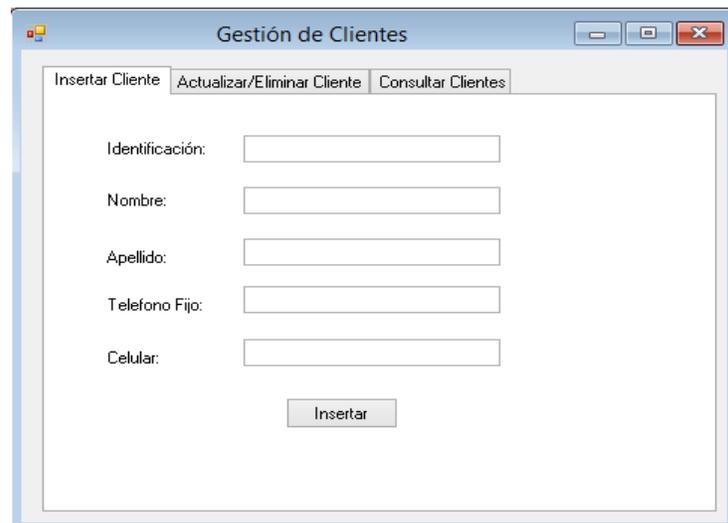


Los formularios deben tener los siguientes diseños:

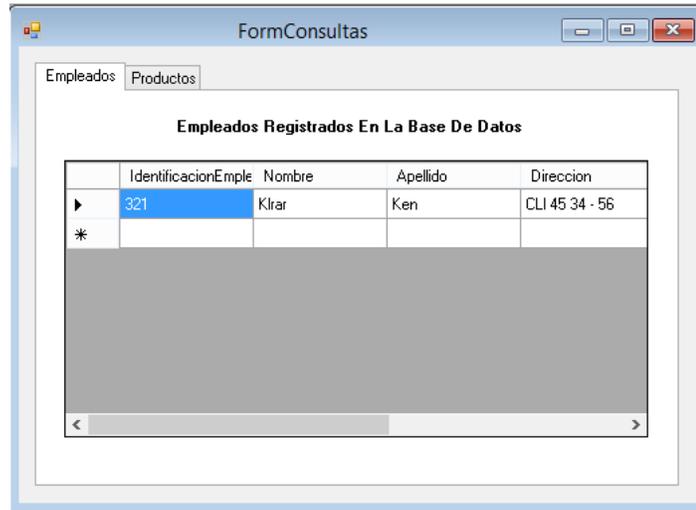
1. Principal



2. FormCliente

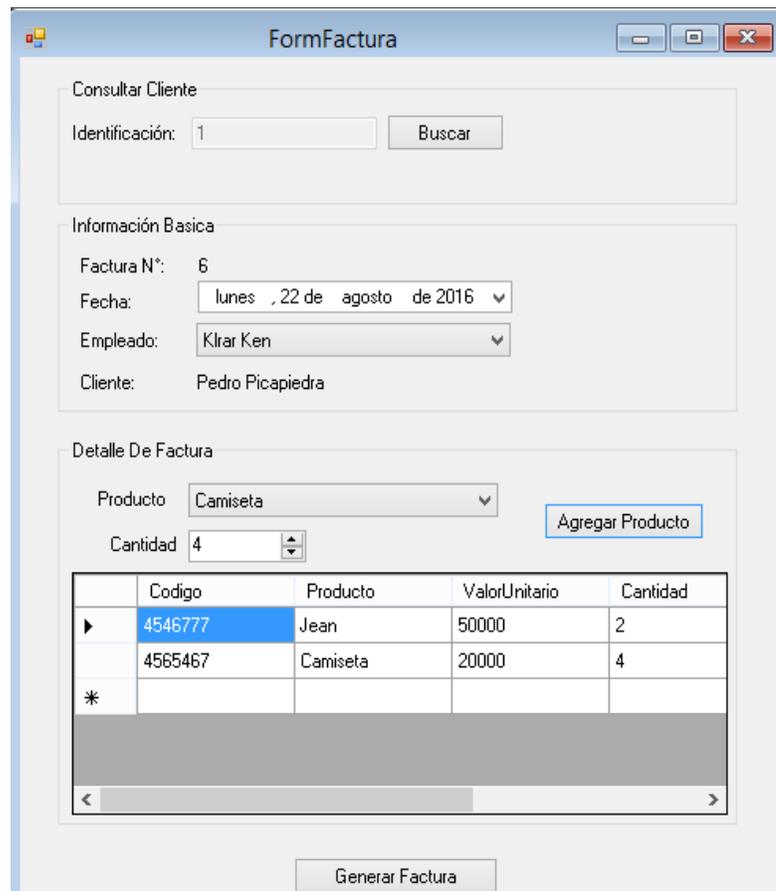


3. FormConsultas



	IdentificacionEmple	Nombre	Apellido	Direccion
▶	321	Klrar	Ken	CLI 45 34 - 56
*				

4. FormFactura



Consultar Cliente

Identificación:

Información Basica

Factura N°:

Fecha: ▼

Empleado: ▼

Cliente:

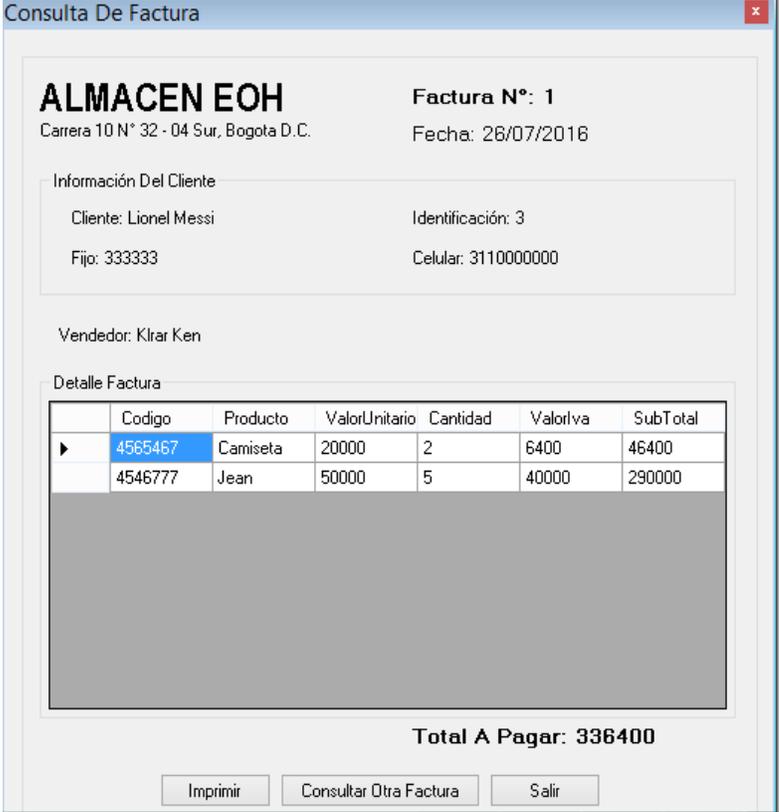
Detalle De Factura

Producto: ▼

Cantidad: ▼

	Codigo	Producto	ValorUnitario	Cantidad
▶	4546777	Jean	50000	2
	4565467	Camiseta	20000	4
*				

5. FormConsultarFactura



ALMACEN EOH Factura N°: 1
Carrera 10 N° 32 - 04 Sur, Bogota D.C. Fecha: 28/07/2016

Información Del Cliente
 Cliente: Lionel Messi Identificación: 3
 Fijo: 333333 Celular: 3110000000

Vendedor: Klar Ken

Detalle Factura

	Codigo	Producto	ValorUnitario	Cantidad	ValorIva	SubTotal
▶	4565467	Camiseta	20000	2	6400	46400
	4546777	Jean	50000	5	40000	290000

Total A Pagar: 336400

Nota: Se debe programar cada formulario para que entreguen las funcionalidades respectivas

A partir del siguiente paso inicia el procedimiento para diseñar y generar un reporte “Crystal Report” con Visual Studio .Net

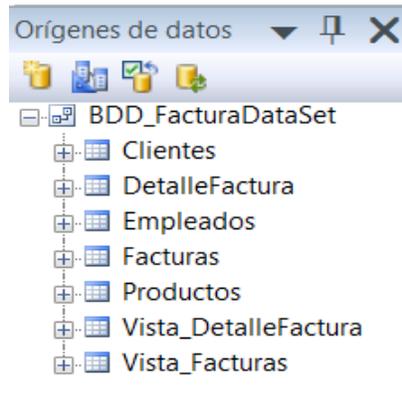
PASO 7: Desde la ventana principal de Visual Studio dar clic en el menú “Datos” y luego en el submenú “Agregar nuevo origen de datos...”. En la ventana que aparece dar clic en el icono “Base de datos” como respuesta a la pregunta “¿De dónde obtendrá la aplicación los datos?” y luego dar clic en lo botón “Nueva conexión”

En la ventana que aparece debe seleccionar Origen de datos (ítem 1), Nombre de servidor (ítem 2) y nombre de bases de datos (Item3) de acuerdo a con la imagen que se muestra a continuación:

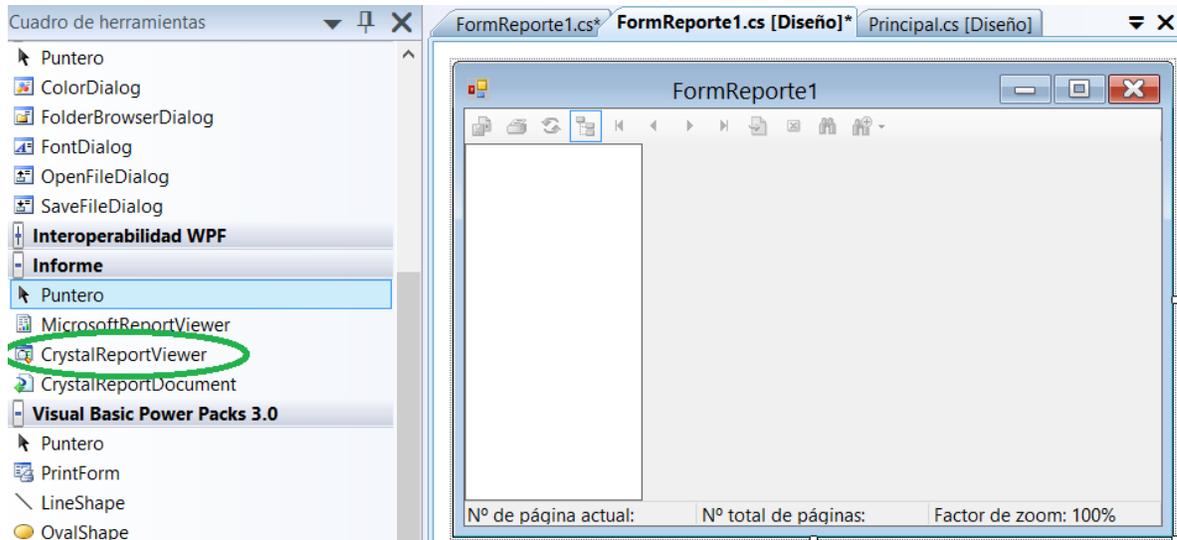
Una vez colocado los parámetros dar clic en el botón “Probar Conexión” para verificar, luego clic en aceptar y clic en el botón siguiente de la ventana. Ya en la siguiente ventana seleccionar los objetos de bases de datos (Primordialmente seleccionar tablas y vistas)

	GUÍA DE TRABAJO N° 11 - C# + SQL SERVER		
	Educación Media Fortalecida SED/SENA	Programación de Software GRADO 11	
	Ing. Néstor Raúl Suarez Perpiñan Página 12 de 16		

Para terminar dar clic en el botón “finalizar” y verificar en la ventana de orígenes de datos que ya se encuentre disponible “BDD_FacturaDataSet” junto con los elementos que fueron seleccionados.



PASO 8: Desde el explorador de soluciones agregue en la capa de presentación un nuevo formulario y colóquelo de nombre “FormReporte1”. En el diseño de este formulario agregue un control “CrystalReportViewer” que encontrará en el cuadro de herramientas en la sección “Informe”

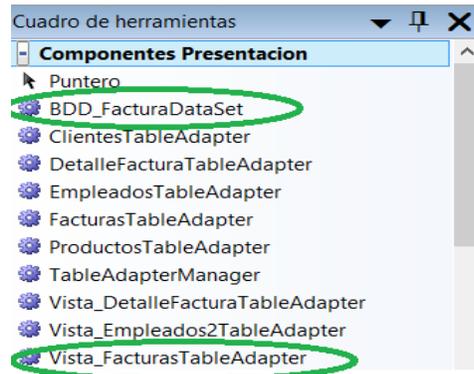


Nota: En este ejercicio se agregará un reporte de las facturas generadas organizadas por fecha, por ende se diseñara el reporte por medio del asistente de Crystal Report usando la vista_facturas.

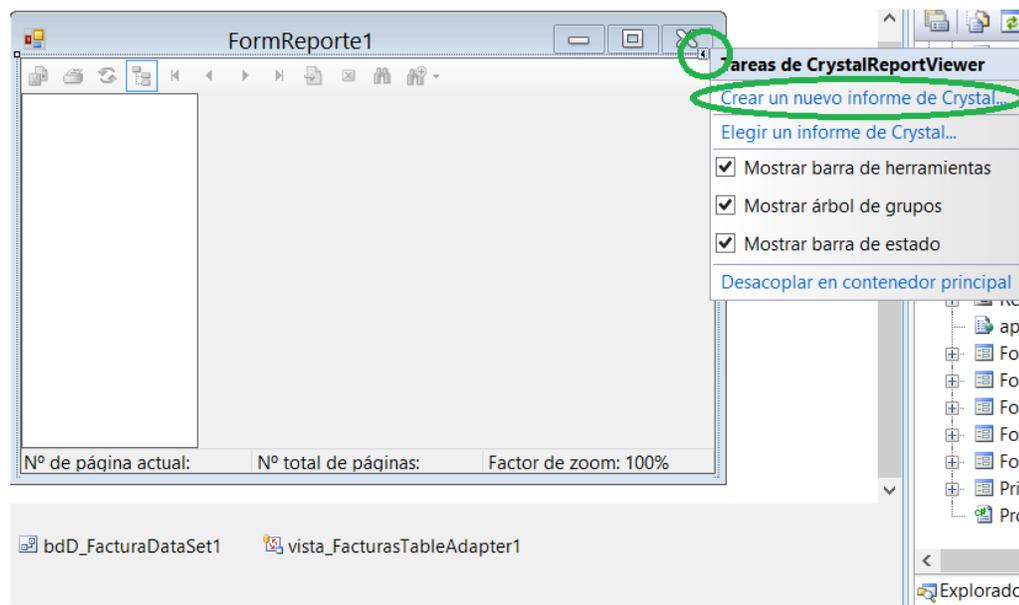
PASO 8: Desde el cuadro de herramientas ubique la sección “Componentes Presentación” y ubique los controles “BDD_FacturaDataSet” y “Vista_FacturasTableAdapter” y agréguelos al formulario. BDD_FacturaDataSet es el dataset que contiene la conexión a los elementos de la

	GUÍA DE TRABAJO N° 11 - C# + SQL SERVER		 SENA Servicio Nacional de Aprendizaje
	Educación Media Fortalecida SED/SENA	Programación de Software GRADO 11	
	Ing. Néstor Raúl Suarez Perpiñan Página 13 de 16		

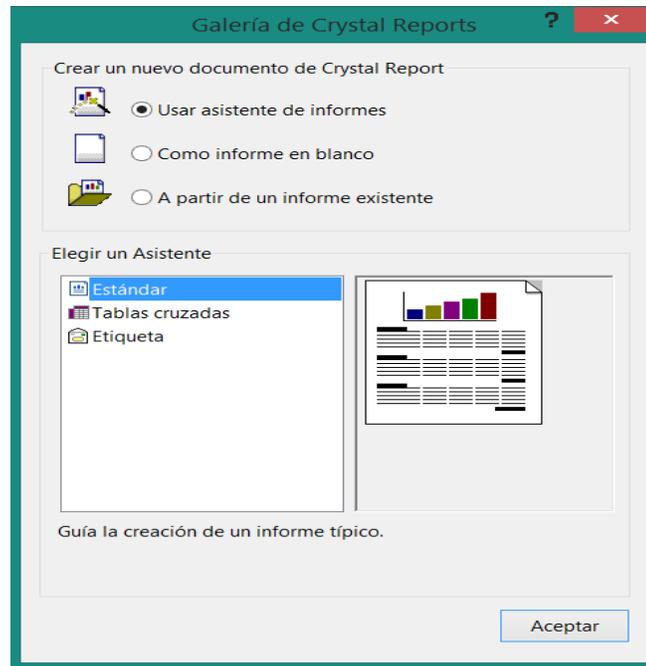
bases de datos que se van a usar y Vista_FacturasTableAdapter este es el adaptador de datos correspondiente a la vista seleccionada para el reporte a diseñar.



PASO 9: Ubicar en la esquina superior del control “CrystalReportViewer” el menu de tareas y escoja allí la opción “Crear un nuevo informe de Crystal Report”

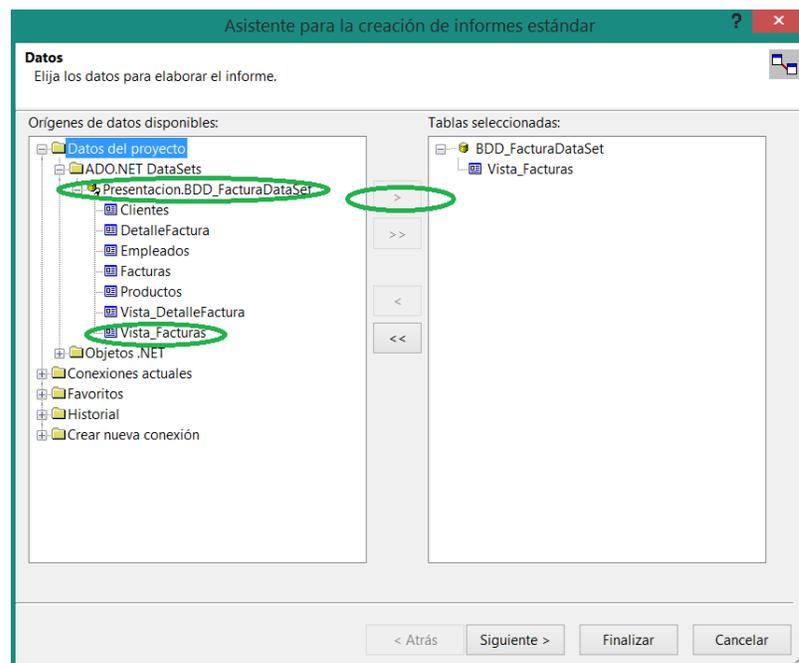


PASO 9: Colóquele un nombre al reporte (o dejar el que aparece por defecto) y siga paso a paso las indicaciones del asistente para diseñar el reporte de Crystal Report respectivo.



Al usar el asistente tenga en cuenta las siguientes indicaciones:

- ✓ En la ventana de orígenes de datos disponibles escoja: Datos del proyecto – ADO.NET DataSets y luego seleccione el DataSet que fue previamente creado en el paso 7 y finalmente ubique la tabla o vista desde donde se generará el reporte, que para este ejercicio es la vista “Vista_Facturas”



	GUÍA DE TRABAJO N° 11 - C# + SQL SERVER		
	Educación Media Fortalecida SED/SENA	Programación de Software GRADO 11	
	Ing. Néstor Raúl Suarez Perpiñan Página 15 de 16		

- ✓ En la ventana de selección de campos escoja los campos que desea que aparezcan en el reporte
- ✓ **Importante:** en la ventana de agrupamiento seleccione el compe bajo el cual se agruparán los datos en el reporte. Para este ejemplo que se requiere un informe de facturas generadas organizadas por fecha se debe escoger el campo "Vista_Facturas.Fecha " en esta ventana
- ✓ Para terminar el asistente de reportes dar clic en el botón "Finalizar"

Observe que el reporte generado tiene siete secciones; las cinco obligatorias y dos más relativas al grupo. Puede editar cuantas veces quiera el reporte creado, al agregar componentes o modificar los que hayan sido colocados. Las secciones contienen componentes que se pueden obtener en dos lugares:

1. Página "Crystal Reports" de la Caja de herramientas, si se trata de textos fijos, líneas y cajas.
2. "Explorador de campos", en el caso de campos de la base de datos, fórmulas, totales, grupos y otros datos específicos del reporte, como fecha, hora y numeración de páginas.

PASO 10: Ubique el evento Load del FormReporte1 y escriba el código que se muestra a continuación, donde `CrystalReport1` es el nombre que le asigno a reporte creado desde el asistente de Crystal Report.

```
private void FormReporte1_Load(object sender, EventArgs e)
{
    this.vista_FacturasTableAdapter1.Fill(this.bdD_FacturaDataSet1
.Vista_Facturas);

    CrystalReport1 reporte = new CrystalReport1();
    reporte.SetDataSource (bdD_FacturaDataSet1);

    crystalReportViewer1.ReportSource = reporte;
}
```

PASO 11: En el formulario principal agregar un menú que permita abrir el FormReporte1 y verifique que el reporte se visualiza correctamente en pantalla, de acuerdo a los datos que fueron seleccionados y agrupados de acuerdo al criterio aplicado desde el asistente de Crystal report.

	GUÍA DE TRABAJO N° 11 - C# + SQL SERVER		
	Educación Media Fortalecida SED/SENA	Programación de Software GRADO 11	
	Ing. Néstor Raúl Suarez Perpiñan Página 16 de 16		

TALLER

Agregar 3 nuevos reportes al ejercicio trabajado en esta guía con las siguientes especificaciones:

1. Un reporte que muestre las facturas generadas agrupadas por cliente (Use la Vista_facturas para este reporte)

2. Un reporte que muestre las facturas generadas agrupadas por empleado (Use la Vista_facturas para este reporte)

3. Un reporte de productos vendidos agrupados por el nombre del producto (Use la Vista_DetalleFactura para este reporte)