Manual Técnico

Tarjeta Inteligente de Pago

Cómo utilizar el software, conocer su estructura y funcionamiento

2013

Equipo 4

Facultad de telemática

03/06/2013

Equipo 4:

Barbosa Gonzales Edsel

Garza Gallegos José Luis

Gómez Peralta Kevin Roberto

Silva Contreras Amayelli Itzel

Contenido

[Introducción 1](#_Toc373480597)

[Requerimientos del sistema 2](#_Toc373480598)

[Documentos de referencia 3](#_Toc373480599)

[DIAGRAMA DE CONTEXTO NIVEL 0: 4](#_Toc373480600)

[DIAGRAMA DE CONTEXTO NVEL 1: 4](#_Toc373480601)

[Conceptos generales 5](#_Toc373480602)

[Diccionario de Datos: 6](#_Toc373480603)

[Código 9](#_Toc373480604)

[Soporte Técnico 26](#_Toc373480605)

Introducción  
  
Actualmente el sistema de cobro del transporte en el estado de Colima, principalmente en el área conurbada Colima – Villa de Alvares es un poco monótona, recurriendo al tradicional pago del trasporte cuando el usuario usa este medio.

Por lo tanto en la universidad de colima se ha llevado a cabo un proyecto en el cual se propone la utilización de una Tarjeta Inteligente de pago, en la cual se registraran los datos más importantes de la persona poseedora de esta, además de servir como identificación ante la universidad y la FEC, será un monedero electrónico en el cual el usuario podrá hacer recargas y posteriormente cuando se suba a una ruta el pasara la tarjeta por un sensor y automáticamente se le descontara el precio acordado.

En Colima la venta de sistemas se reduce casi exclusivamente a sistemas genéricos provistos por tiendas de autoservicio, habiendo pocos grupos de desarrolladores. El aumento del uso de sistemas de pago en línea la venta de software de manera directa entre las empresas y los clientes es el otro medio qué más se utiliza en el Estado. Nuestro principal mercado se encontrará en los negocios, pequeñas y medianas empresas que se abren continuamente en Colima que necesitan sistemas de punto de venta, sitios web y sistemas para la administración de sus servicios y manejo de datos.

Por tal motivo lo que nosotros proponemos es un prototipo, después de que obtengamos el presupuesto y apoyo necesario, este proyecto se concluirá.

# Requerimientos del sistema

El sistema Di-Bus se entregará en un paquete que incluye el cd de instalación, y un manual para el usuario en un libro pequeño y en un cd con el documento en PDF. El sistema es ligero y eficaz, no desperdicia recursos e incluye un sistema de ayuda inteligente en línea. Para su correcto funcionamiento se recomiendan equipos con las siguientes características:

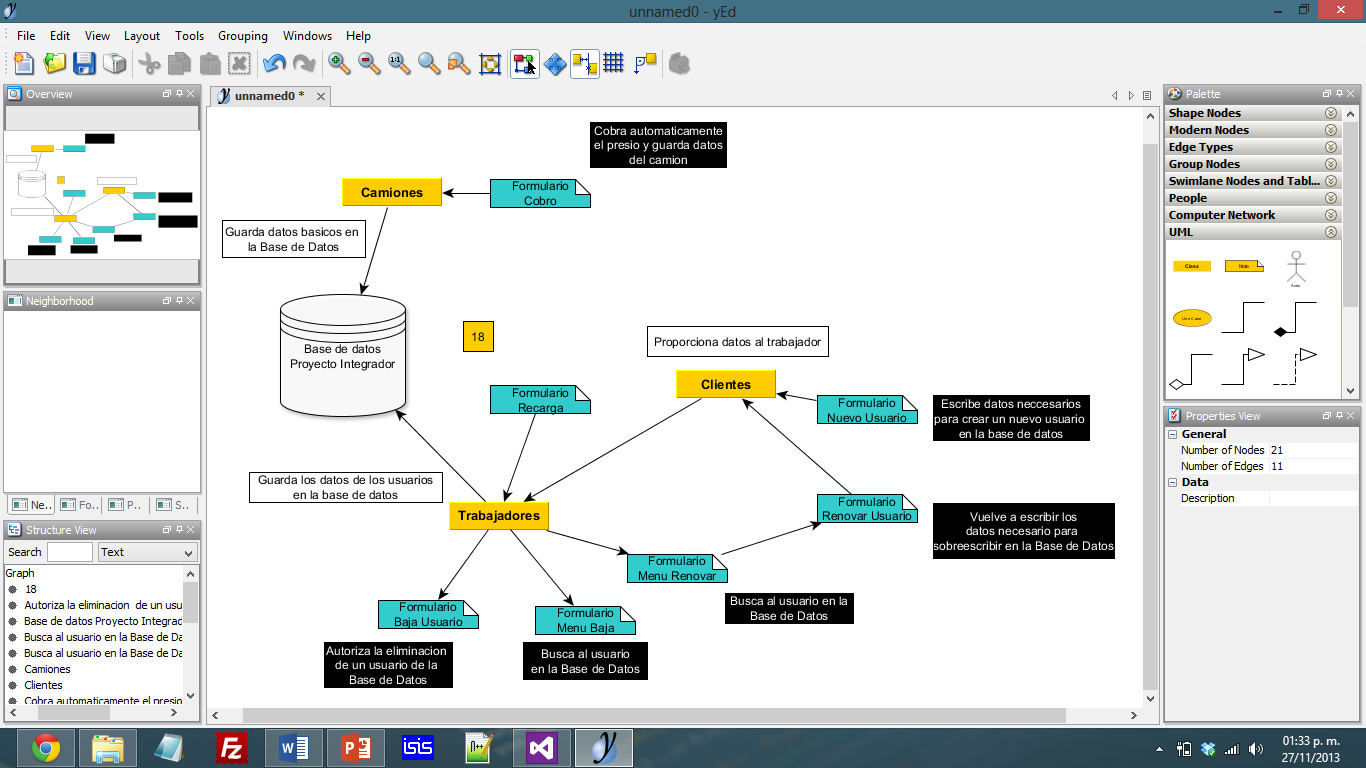
|  |  |
| --- | --- |
| Componente | Requisitos de Di-Bus 1.0 |
| Equipo y procesador | Procesador de x86- o de x64 bits de 1 gigahercio (GHz) o más rápido con conjunto de instrucciones SSE2 |
| Memoria (RAM) | RAM (32 bits) de 1 gigabyte (GB); RAM (64 bits) de 2 gigabytes (GB) |
| Disco duro | 3,0 gigabytes (GB) de espacio disponible en disco |
| Pantalla | La aceleración de hardware de gráficos requiere una tarjeta gráfica DirectX 10 y una resolución de 1024 x 576 |
| Sistema operativo | Windows XP, Windows 7 u 8, Windows Server 2008 R2 o Windows Server 2012 |
| Explorador | Microsoft Internet Explorer 8, 9 o 10; Mozilla Firefox 10.x o una versión posterior; Apple Safari 5; o Google Chrome 17.x. |
| Versión .NET | 3.5, 4.0 o 4.5 |
| Requisitos y consideraciones adicionales | Algunas funciones y gráficos pueden variar, en función de la configuración de su sistema. Puede que algunas características necesiten hardware o conectividad de servidores adicionales o avanzados. |

# Documentos de referencia

* Manual de Usuario.
* James Kovacs (7 de septiembre de 2007). «C#/.NET History Lesson
* Designing Components with the C# STL
* Large-Scale Software Architecture A Practical Guide using UM
* Visual C# Developer Center, Frequently Asked Questions About Visual C# .NET 2003
* Programmer's Heaven C# School Book
* Torgersen, Mads (October 27, 2008). "New features in C# 4.0". Microsoft. Retrieved October 28, 2008.
* Don Box and Anders Hejlsberg (February 2007). "LINQ: .NET Language-Integrated Query". Microsoft. Retrieved March 21, 2011.
* *C# Language Specification* (4th ed.). Ecma International. June 2006. Retrieved January 26, 2012.
* *Dan Fernandez's Blog*. Blogs.msdn.com. January 29, 2004. Retrieved October 4, 2012.
* Foundations.Of.Algorithms.Using.Cpp.Pseudocode.3rd.Edition.e
* Effcient C# Programming Techniques Ebook – fly
* Petzold, Charles (2002). *Programming Microsoft Windows with C#*. Microsoft Press. ISBN 0-7356-1370-2.
* Drayton, Peter; Albahari, Ben; Neward, Ted (2002). *C# Language Pocket Reference*. O'Reilly. ISBN 0-596-00429-X.

DIAGRAMA DE CONTEXTO NIVEL 0:

# DIAGRAMA DE CONTEXTO NVEL 1:



Conceptos generales  
El programa está compuesto por una interfaz general o principal, la cual nos permite movernos entre un menú en la parte superior para poder realizar diferentes actividades, entre las cuales se encuentra registrar un nuevo usuario a una base de datos (pidiendo datos como lo son: CURP, nombre, apellido paterno, apellido materno, sexo, escuela, grado, grupo, tipo de escolaridad, dirección, colonia, teléfono fijo, teléfono celular, correo, edad, lugar de nacimiento, fecha de nacimiento, estado civil, alergia y tipo de sangre), borrar usuarios inactivos de la base de datos y actualizar la información de cualquier usuario. Automáticamente se sabrá si el nuevo usuario registrado será estudiante, público en general o persona de la tercera edad.

De igual manera en este programa se pueden hacer recargas de dinero electrónico a un usuario cualquiera previamente registrado; las cantidades son $12, $24, $36, $48, $60 y $72.

También centrara con una simulación que representa el cobro que realiza cada camión, al deslizar el usuario su credencial (en nuestro casa ingresar su CURP) el sistema identificara automáticamente a la persona y que tipo de usuario es, de esta manera se le cobrara el pasaje a la persona ($3 o $6) dependiendo y se le restara al saldo de esta misma.

Cuenta con un apartado llamado consulta, en el cual solo sirve para consultar la información de los usuarios registrados y de los camiones también registrador (no se pueden agregar datos).

Otra cosa que pusimos fue un cuadro estadístico, en el cual se obtienen el saldo de cada una de los usuarios y posteriormente se obtiene las medidas de tendencia central.

En el apartado de Informe, se muestran las ganancias de cada uno de los camiones por día, y en otro apartado la cantidad total de usuarios que se subieron a cada camión.

# Diccionario de Datos:

|  |  |  |  |
| --- | --- | --- | --- |
| ***VARIABLE*** | ***TIPO*** | ***DESCRIPCIÓN*** | ***Localización/Formularios*** |
| cmd | OleDbCommand | Establece cualquier tipo de instrucción que se le dé a la Base de Datos | Recarga, Nuevo Usuario,  Renovar Usuario, Cobrar,  Estadisticas, Baja Usuario |
| cn | OleDbConnection | Establece una conexión con la Base de Datos | Recarga, Nuevo Usuario,  Renovar Usuario, Cobrar,  Estadisticas, Baja Usuario |
| dr | OleDbDataReader | Establece un Lector con la Base de Datos | Recarga, Nuevo Usuario,  Renovar Usuario, Cobrar,  Estadisticas, Baja Usuario |
| q | string | Guarda la cadena con la cual le daremos las instrucciones a la Base de Datos | Recarga, Nuevo Usuario,  Renovar Usuario, Cobrar,  Estadisticas, Baja Usuario |
| Registros | String[ ] | Guarda los registros a actualizarse en la Base de Datos | Renovar Usuario |
| Existe | bool | Ayuda a verificar si ya existe una CURP en la Base de Datos | Recarga, Nuevo Usuario, Cobrar |
| Total | int | Sirve para saber la cantidad a descontar o recargar | Recarga, Cobrar |
| Saldo | int | Muestra el Saldo Actual de la persona | Recarga, Cobrar |
| Recarga | int | Es la cantidad a recargar | Recarga |
| tipo | string | Establece el tipo de usuario en momento | Nuevo Usuario, Cobrar |
| coll | AutoCompleteStringCollection | Son los datos del TextBox en momento para la función autocompletar | Menu Renovar, Menu Baja, Cobrar, Recargar, Consulta Usuario |
| Conexion | OleDbConnection | Establece la conexión con la Base de Datos | Menu Renovar, Menu Baja, Cobrar, Consulta Camiones, Consulta Usuarios |
| Instruccion | OleDbCommand | Establece cualquier instrucción con la Base de Datos | Menu Renovar, Menu Baja, Cobrar, Consulta Camiones, Consulta Usuarios |
| Lector | OleDataReader | Establece un Lector para la Base de Datos | Menu Renovar, Menu Baja, Cobrar |
| cobro | int | Es lo que se le cobrara al usuario en turno | Cobrar |
| constring | string | Indica de donde estará la Base de Datos | Consulta Camiones, Consulta Usuarios |
| i | int | Inicializador para un ciclo Foreach | Estadisticas |
| w | int | Inicializador para un ciclo Foreach | Estadisticas |
| Adaptador | OleDbDataAdapter | Se adapta automáticamente a lo que tenga la Base de Datos | Consulta Camiones, Consulta Usuarios |
| Table | DataTable | Se guargan campos de una Base de Datos | Consulta Camiones, Consulta Usuarios |
| bSource | BindingSource | Sirve para hacer el autocompletado de un DataGrid | Consulta Camiones, Consulta Usuarios |
| DV | DataView | De aquí se muestran los datos al DataGrid | Consulta Camiones, Consulta Usuarios |
| Cuadro\_Estadistico | DataTable | Se guardan campos de una Base de Datos (temporal) | Estadisticas |
| Lista | Int[ ] | Aquí se guardan los saldos de los usuarios | Estadisticas |
| Moda | Int[ ] | Se vuelven a guardar los saldos de los usuarios | Estadisticas |
| Indicador | bool | Sirve para no repetir Marcas de Clase | Estadisticas |
| Elementos | int | Indica la cantidad de saldos | Estadisticas |
| Inicializador1 | int | Necesario para un ciclo Foreach | Estadisticas |
| Inicializador2 | int | Necesario para un ciclo for | Estadisticas |
| Inicializador3 | int | Necesario para un ciclo for | Estadisticas |
| Inicializador4 | int | Necesario para un ciclo for | Estadisticas |
| Inicializador5 | Int | Necesario para un ciclo for | Estadisticas |
| Frecuencia | Int | Sirve para saber la cantidad de elementos dentro de un rango | Estadisticas |
| Frecuencia2 | Int | Necesario para un ciclo for | Estadisticas |
| Mediana | Int | La mediana de todos los saldos | Estadisticas |
| NumeroClase | Int | Determina el numero de clase | Estadisticas |
| MarcaClase | int | Determinar las marcas de clases | Estadisticas |
| Sumatoria | Int | El total de los saldos | Estadisticas |
| Rango | Int | Indica cuantos números hay desde el saldo mas bajo hasta el mas alto | Estadisticas |
| ValorMaximo | Int | Establece el saldo mas alto | Estadisticas |
| UltimoValor | Int | Establece el saldo mas alto | Estadisticas |
| Intervalo | Int | Indica el intervalo de cada Marca de Clase | Estadisticas |
| IntervaloMenor | Int | Indica el Intervalo Menor de cada Marca de Clase | Estadisticas |
| IntervaloMayor | Int | Indica el Intervalo Mayor de cada Marca de Clase | Estadisticas |
| FrecuenciaAcumulada | Int | Indica la Frecuencia Acumulada de cada Marca de Clase | Estadisticas |
| Contador | Int | Necesario para un ciclo for | Estadisticas |
| Media | Doublé | Establece la media de los saldos | Estadisticas |
| FrecuenciaRelativa | Doublé | Indica la Frecuencia Relativa de cada Marca de Clase | Estadisticas |
| FrecuenciaRelativaAcumulada | Doublé | Indica la Frecuencia Relativa Acumulada de cada Marca de Clase | Estadisticas |
| Grados | Doublé | Indica los grados de cada Marca de Clase | Estadisticas |
| Mediana2 | Doublé | Necesaria para calcular la Mediana | Estadisticas |
| Varianza | Doublé | Establece la varianza de los saldo | Estadisticas |
| Desviacion | Doublé | Establece la Desviacion Estandar de los Saldos | Estadisticas |
| Intervalo2 | Doublé | Sirve para calcular el intervalo | Estadisticas |
| CoeficientedeVariacion | Doublé | Sirve para calcular el Coficiente de Variacion de los Saldos | Estadisticas |
| DesviacionMedia | Doublé | Sirve para calcular la Desviacion Media de los Saldos | Estadisticas |

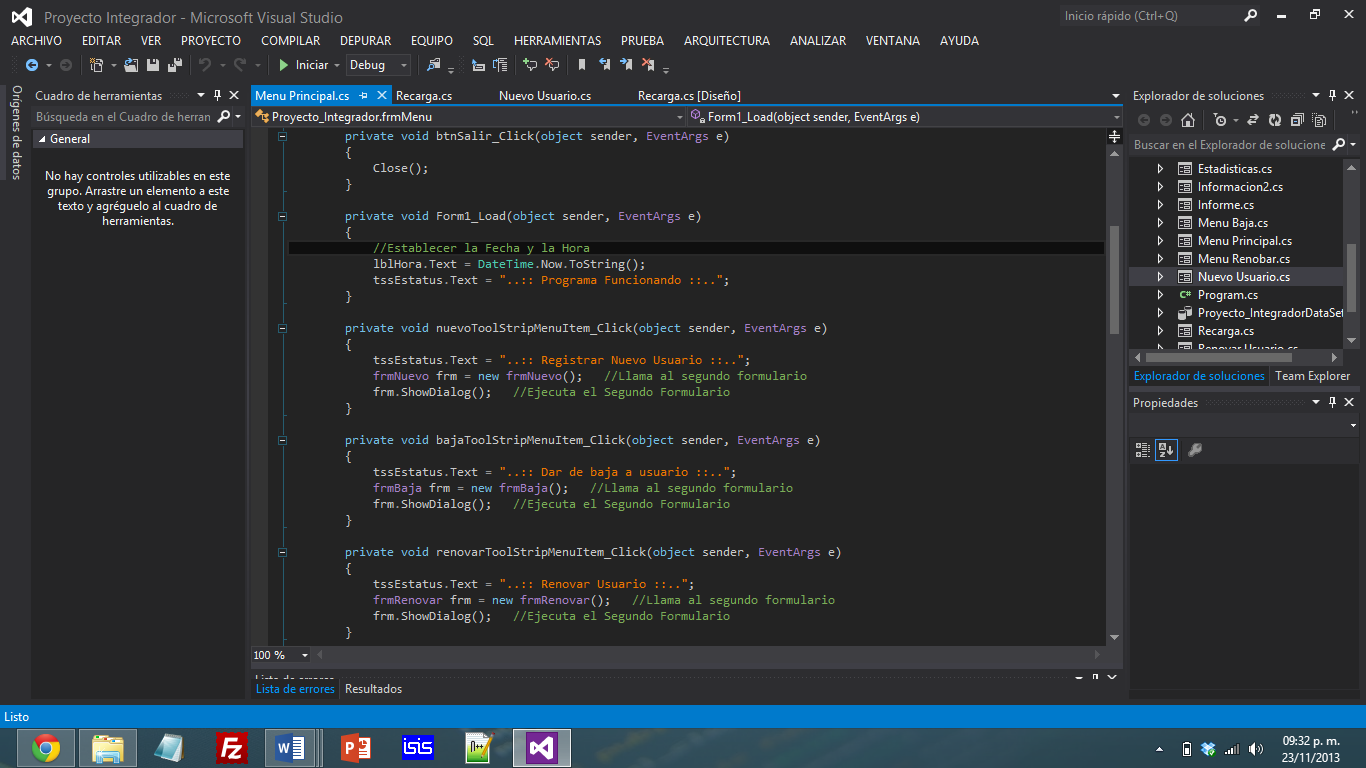
# TABLA DE CONVERSIONES:

|  |  |  |
| --- | --- | --- |
| NOMBRE | FORMULARIO | Conversión Necesaria |
| txtNombre | Renovar, Baja y Nuevo | “Convert.ToString” o “.ToString()” |
| txtEscuela | Renovar, Baja y Nuevo | “Convert.ToString” o “.ToString()” |
| txtGrado | Renovar, Baja y Nuevo | “Int32.Parse()” o “Convert.ToInt32” |
| txtGrupo | Renovar, Baja y Nuevo | “Convert.ToChar” o “.ToChar()” |
| txtDireccion | Renovar, Baja y Nuevo | “Convert.ToString” o “.ToString()” |
| txtFijo | Renovar, Baja y Nuevo | “Int32.Parse()” o “Convert.ToInt32” |
| txtCelular | Renovar, Baja y Nuevo | “Int32.Parse()” o “Convert.ToInt32” |
| txtCorreo | Renovar, Baja y Nuevo | “Convert.ToString” o “.ToString()” |
| txtEdad | Renovar, Baja y Nuevo | “Int32.Parse()” o “Convert.ToInt32” |
| txtLugar | Renovar, Baja y Nuevo | “Convert.ToString” o “.ToString()” |
| txtFecha | Renovar, Baja y Nuevo | “Convert.ToString” o “.ToString()” |
| txtCURP | Renovar, Baja y Nuevo | “Convert.ToString” o “.ToString()” |
| txtEstadoCivil | Renovar, Baja y Nuevo | “Convert.ToString” o “.ToString()” |
| txtAlergia | Renovar, Baja y Nuevo | “Convert.ToString” o “.ToString()” |
| txtTipo | Renovar, Baja y Nuevo | “Convert.ToString” o “.ToString()” |
| txtCurpCobrar | Cobrar | “Convert.ToString” o “.ToString()” |
| txtCurpRecargar | Cobrar | “Convert.ToString” o “.ToString()” |
| txtCurpBaja | Baja | “Convert.ToString” o “.ToString()” |
| txtCurpRenovar | Renovar | “Convert.ToString” o “.ToString()” |
| DataGridConsultaCamiones | Consulta | No necesaria |
| DataGridConsultaUsuarios | Consulta | No necesaria |
| DataGridConsultaInformes | Informes | No necesaria |

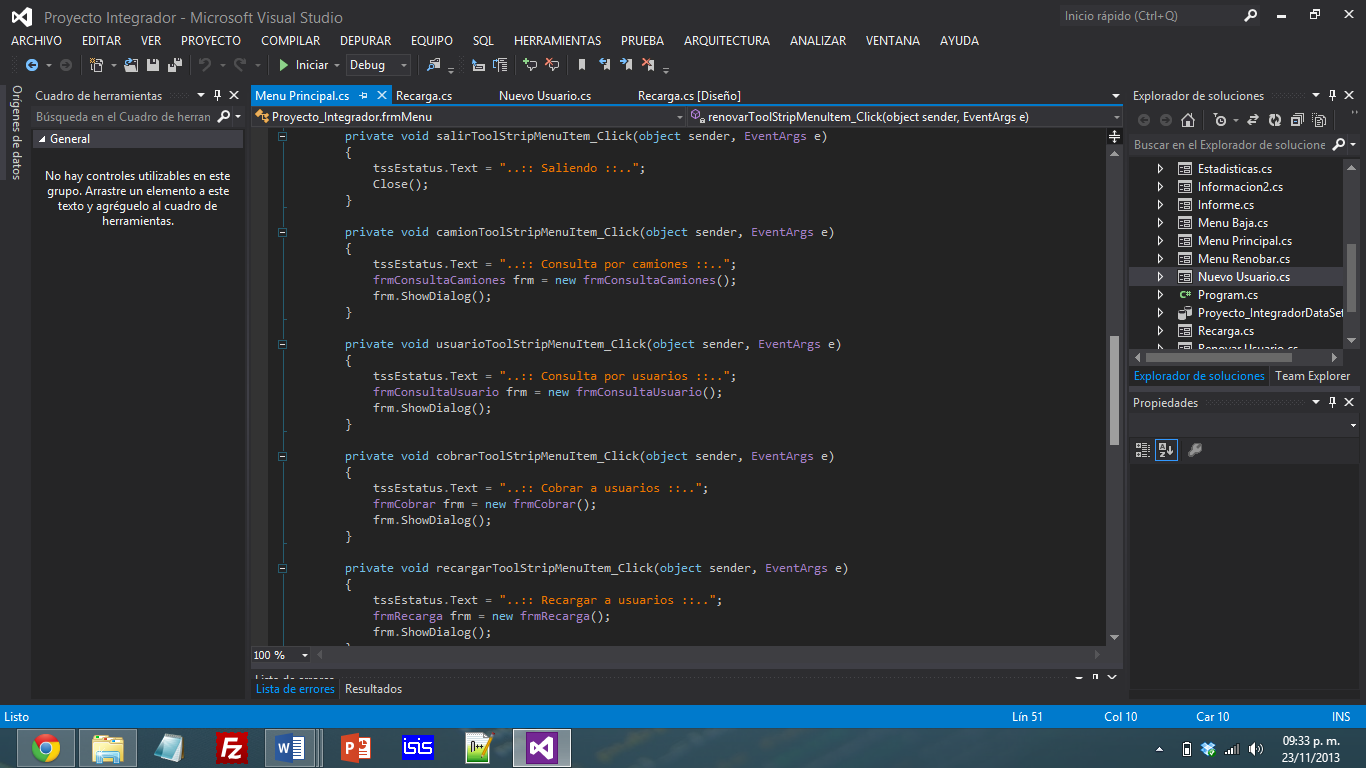
# Código

A continuación se muestra el código necesario para la realización de este programa

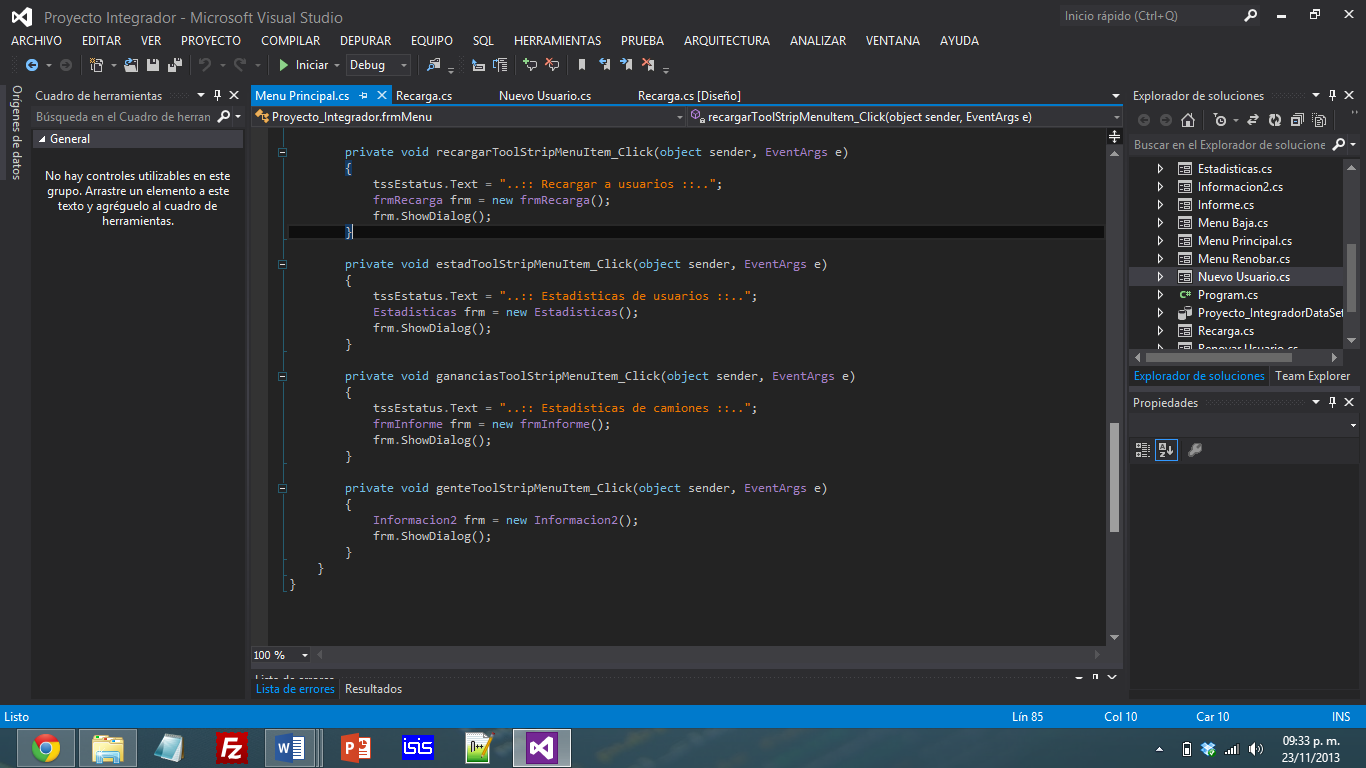
**MENU PRINCIPAL**



En esta sección se indica que al cargarse el formulario principal se establezca la hora y en la barra de estado se escriba “Programa Funcionando. Después se indique que acción realizara cada botón del Menú

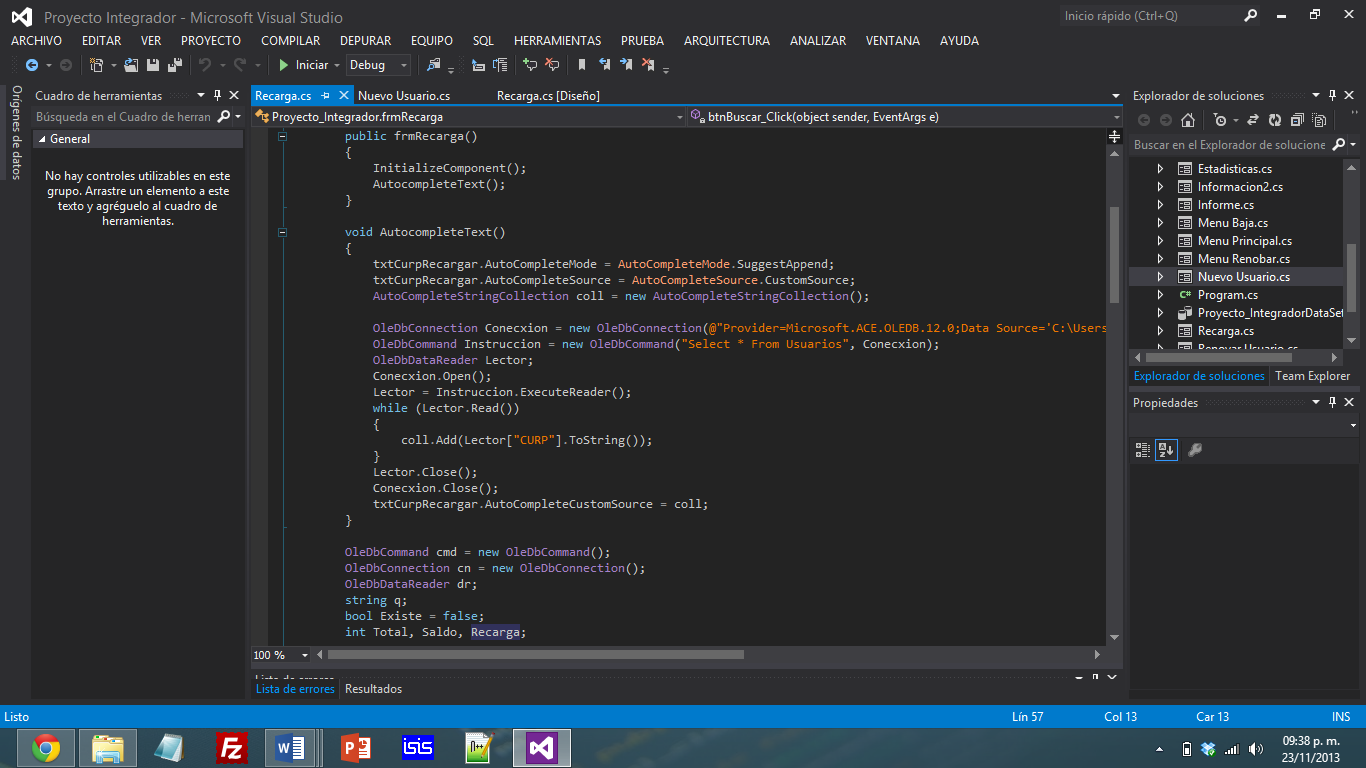


Continuamos declarando que hace cada botón del menú.



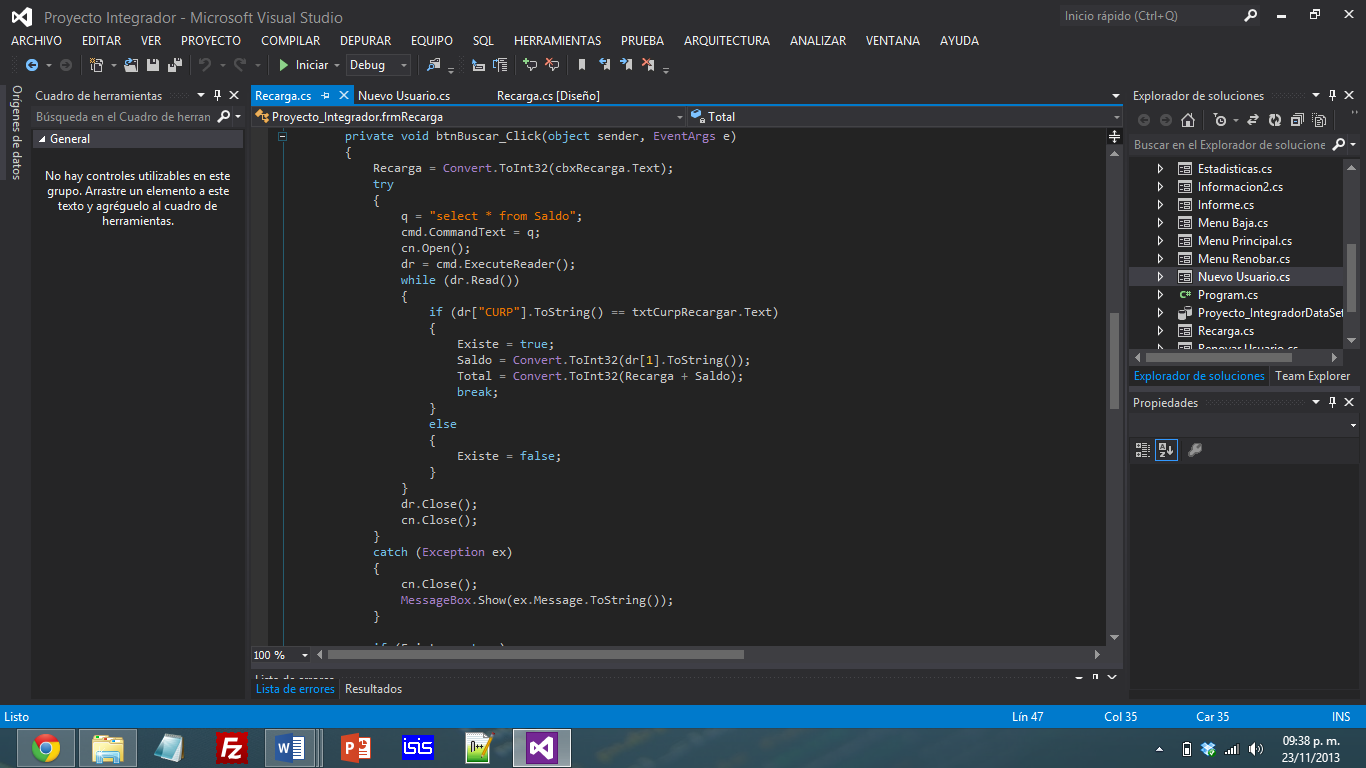
Continuamos declarando que hace cada botón del menú.

**RECARGAS**

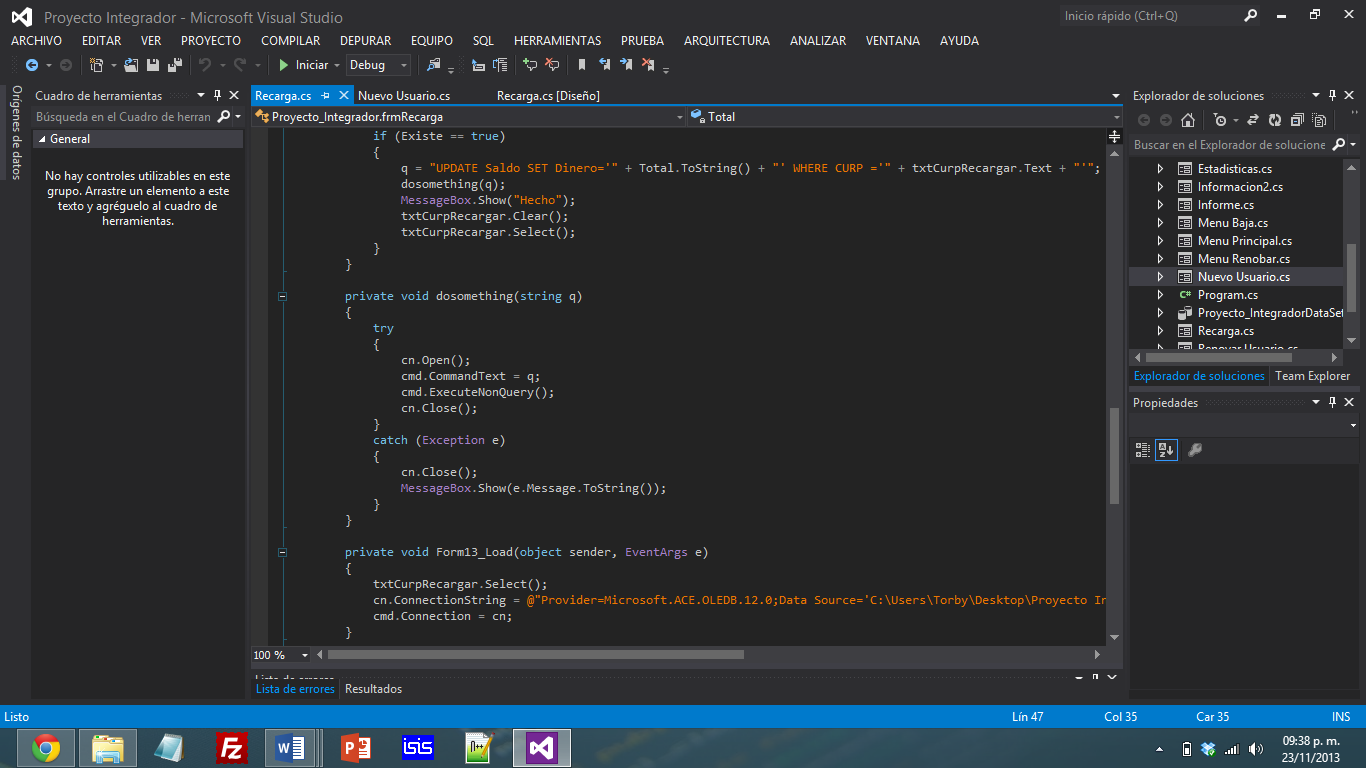


En esta parte del código para la sección de recarga, lo que hacemos es que el TextBox sea autocompletadle, solo con los datos que se existen en la columna CURP de la base de datos, de esta forma es más fácil la ubicación de los usuarios y al mismo tiempo se hace la validación de este o no registrado un usuario.

Posteriormente se establece una nueva conexión de con la base de datos para poder hacer cambios o lectura en esta.

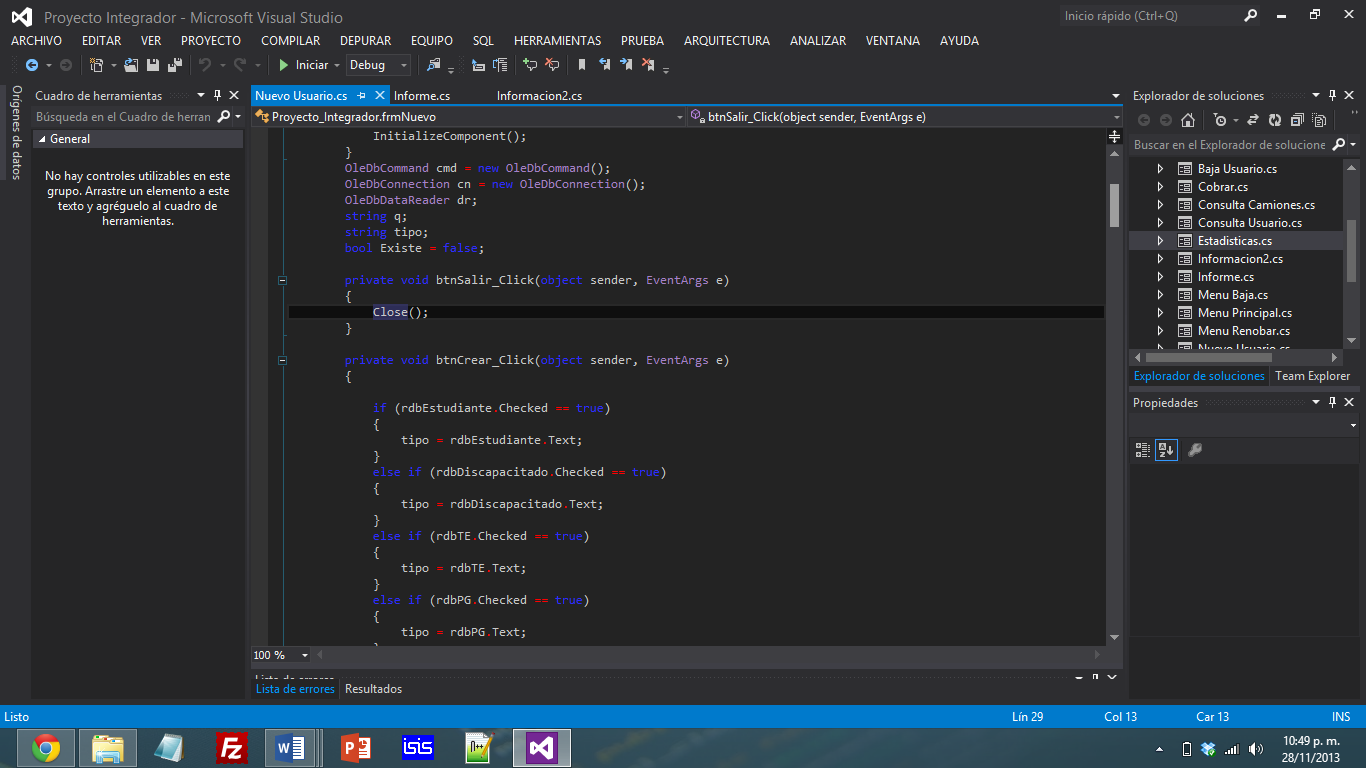


Aquí se obtiene el saldo anterior del usuario (al que previamente se le solicito su CURP) y se convierte a valor entero para poder trabajar con el, asi mismo el valor que solicito que le recargaran se convierte en entero y se le suma al anterior.

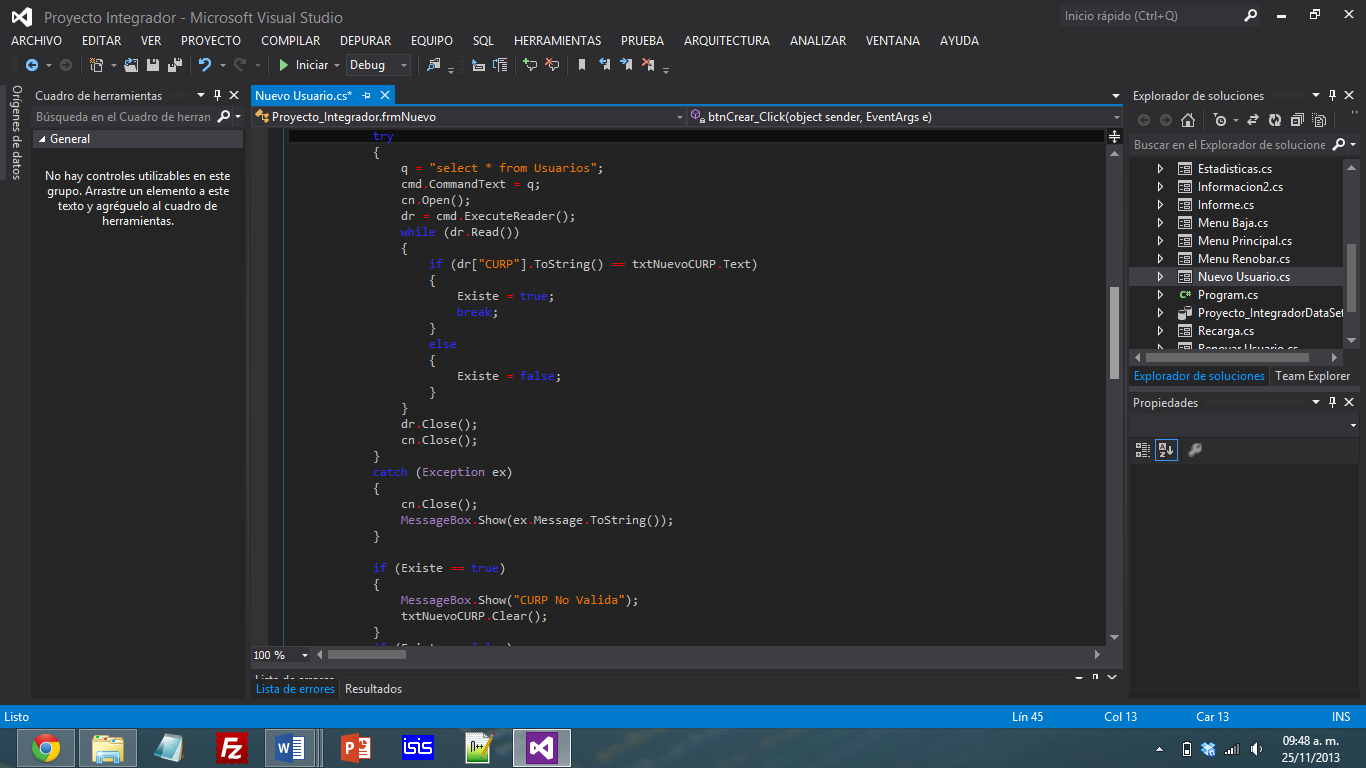


Aquí en primera instancia se hace un UPDATE con el saldo anteriormente calculado, y este se pone en la columna Dinero fila CURP del usuario de la Base de datos Saldo. Después se muestra el método necesario para poder insertar, borrar o hacer update dentro de la base de datos (lo ocuparemos más adelante) y por ultimo al cargar el formulario se carga la base de datos.

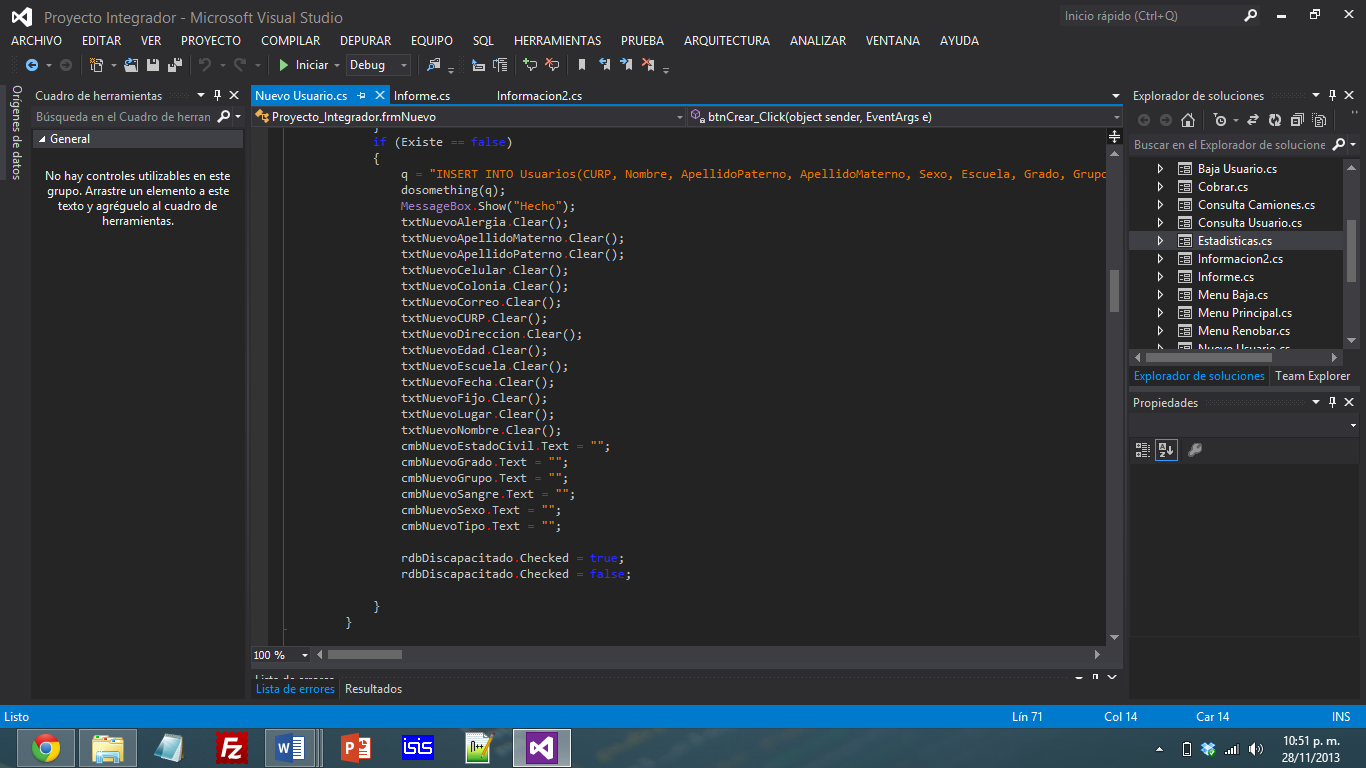
**NUEVO USUARIO**



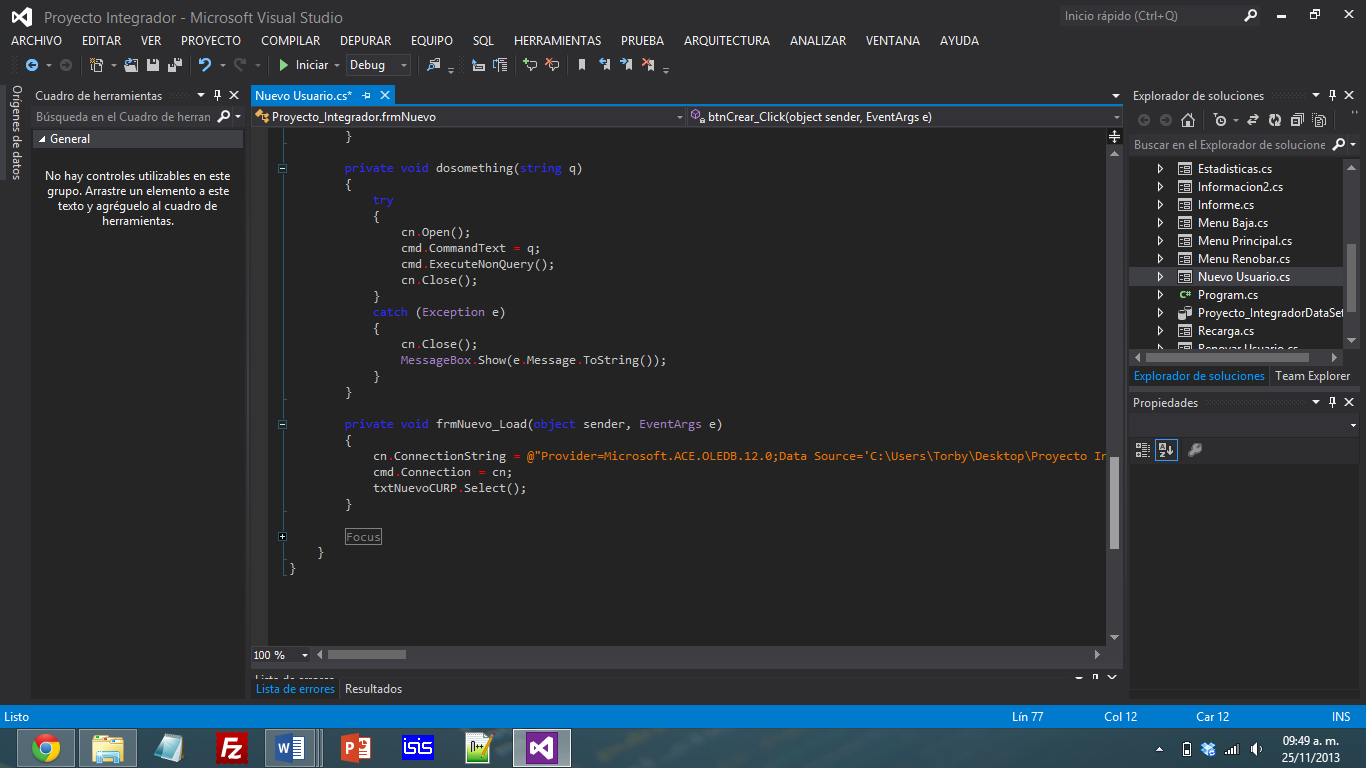
Ahora en la sección para crear un Nuevo usuario, se vuelve a establecer una conexión con la base de datos, se muestra el código del botón “Salir”, además se establece que tipo de usuario es según la selección de los radiobuttons puestos en el menú.



Aquí se verifica que la CURP que el usuario ingrese para crear un nuevo usuario no exista en la base de datos, de lo contrario muestra un mensaje de error.



Aquí se muestran los datos que se ingresaran en la tabla usuario y se ponen en blanco todos los datos



Aquí muestra la conexión con la base de datos.

**MENU BAJA**

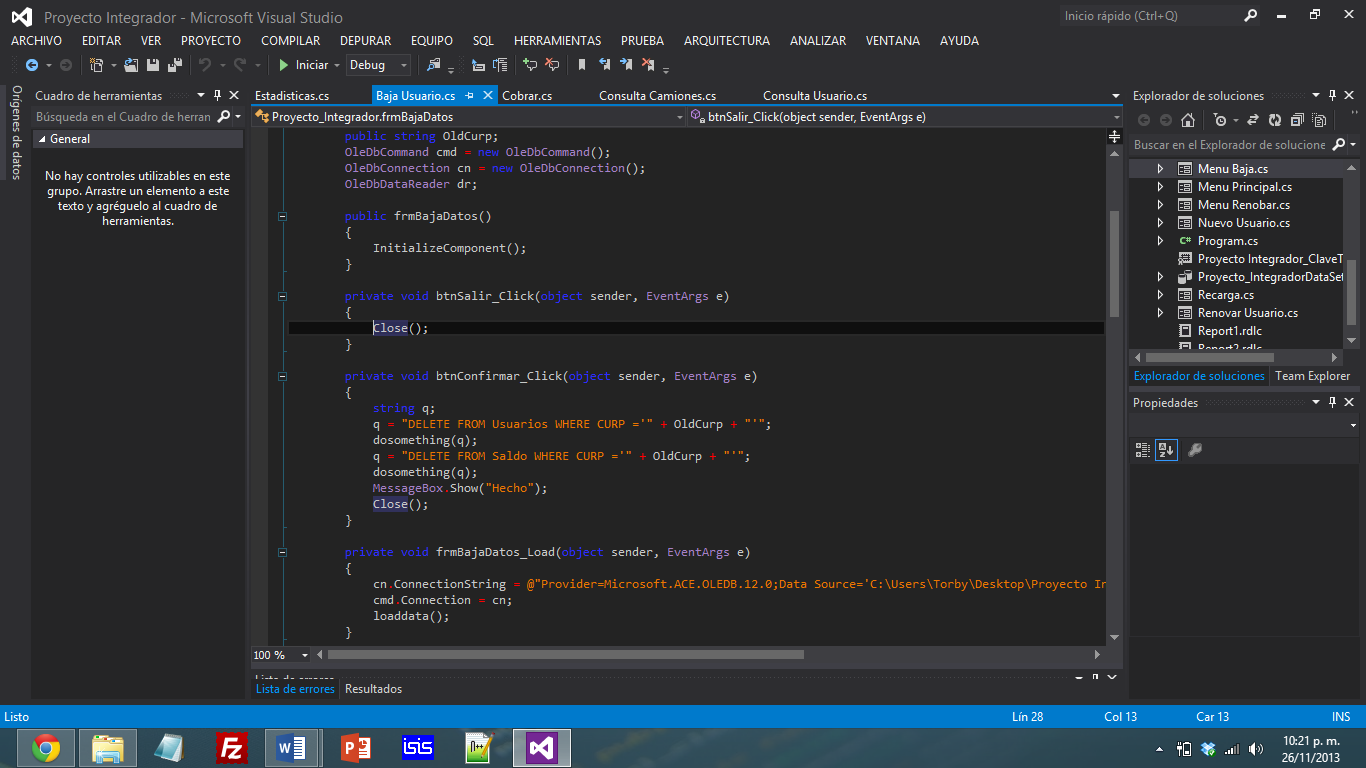


Esta es la función de autocompletar el textbox en el formulario de Menú Baja, en este se observa que se establece una conexión con la Base de Datos y todos sus valores de una cierta columna se agregan al textbox.

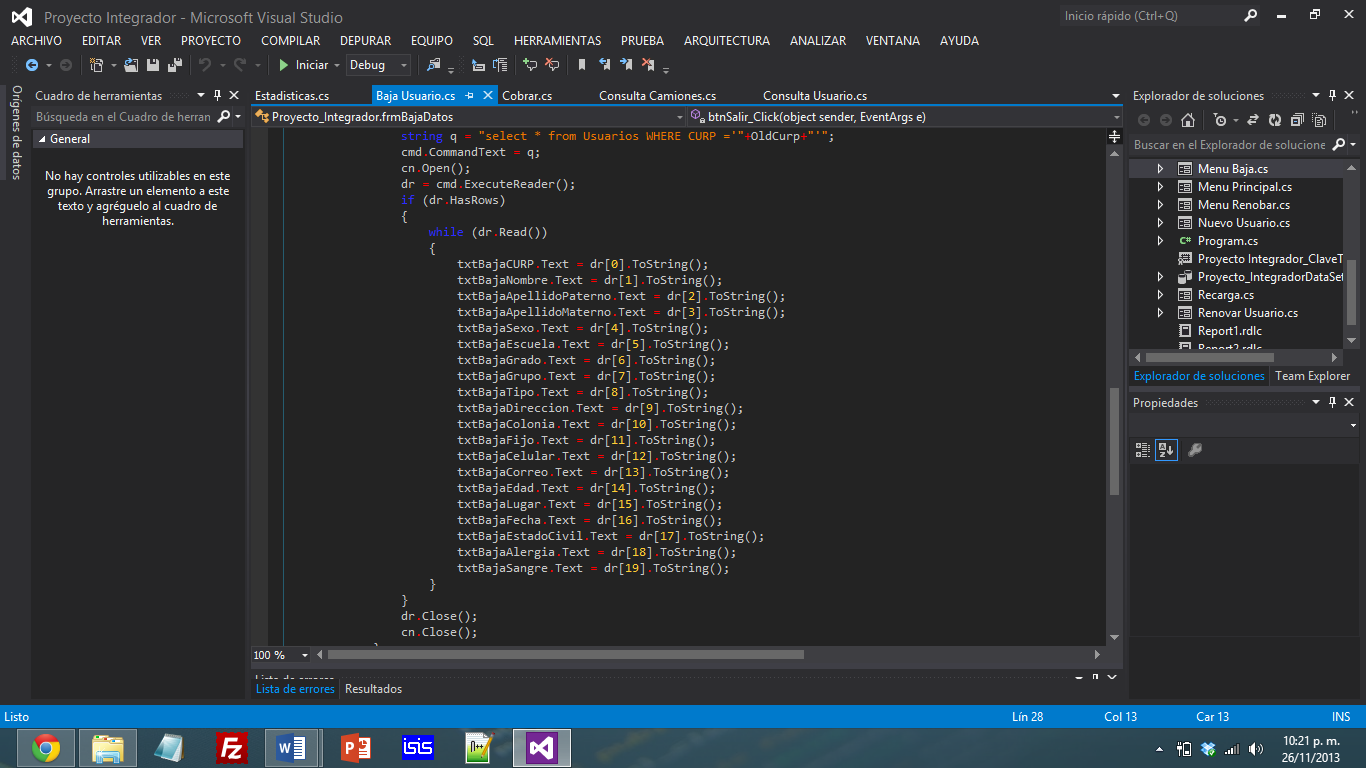


Aquí se muestra la acción el botón salir, la acción a realizar al presionar el botón enter, así como el método buscar y lo que se hacer al cargar el formulario.

**BAJA USUARIO**

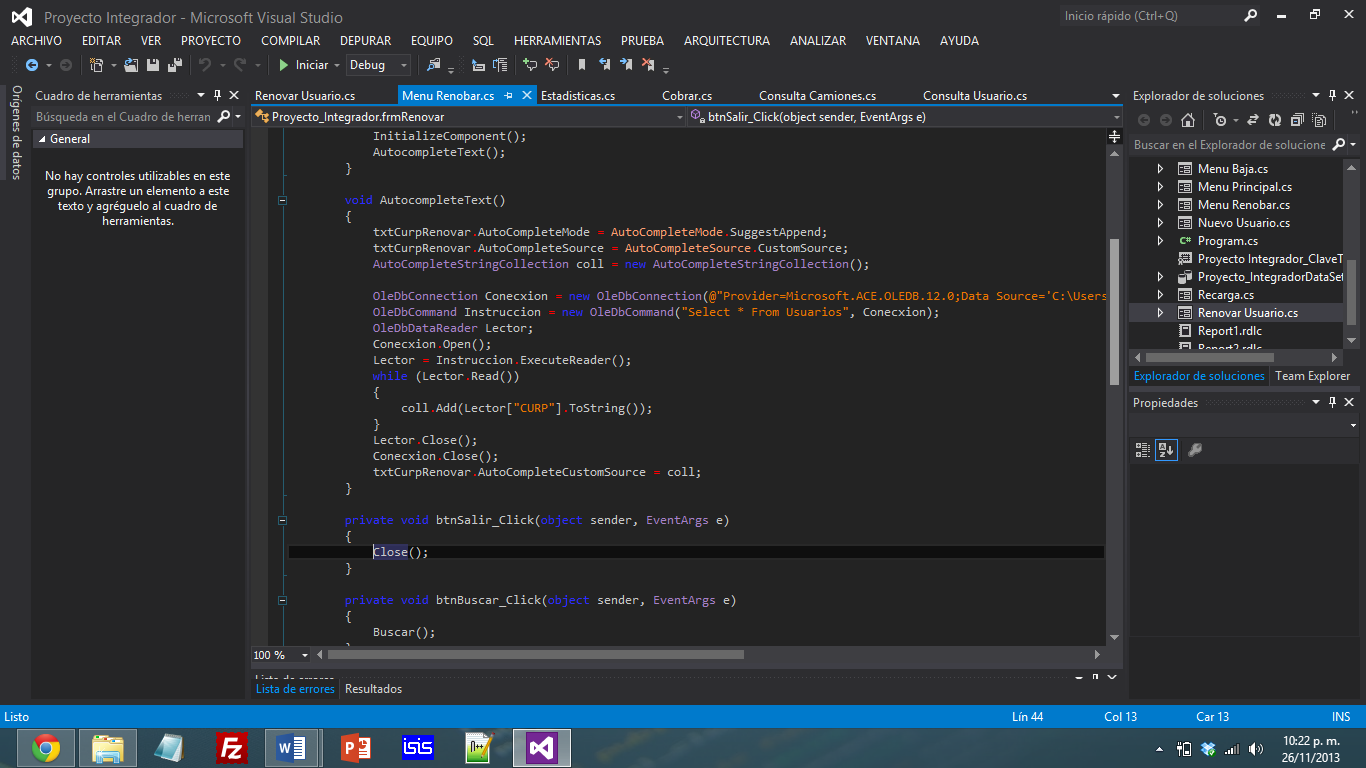


Aquí se muestra la conexión con la base de datos, lo que hace el botón Salir así como lo que hace el botón borrar.

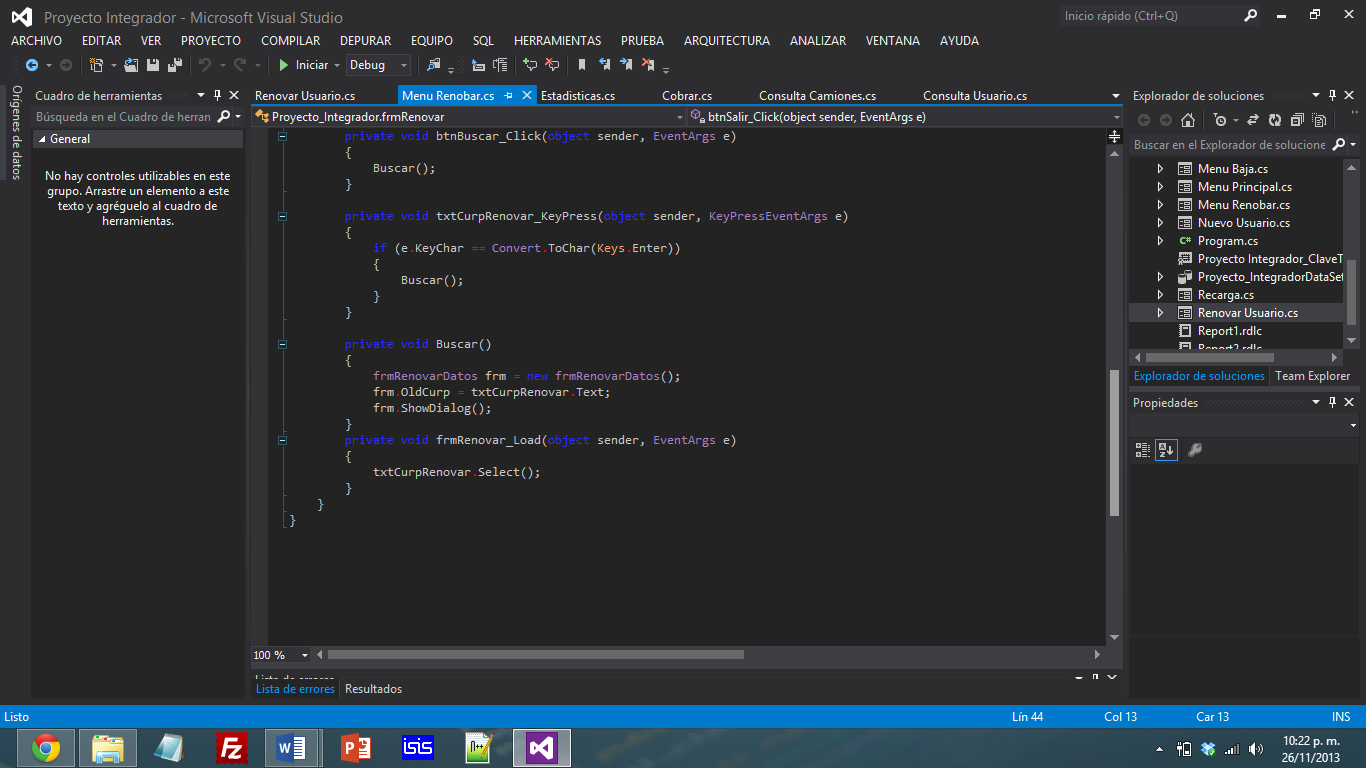


Aquí se lee lo que existe en la base de datos y cada columna de la fila necesaria se pone en los textbox para que el usuario conozca los datos anteriores y vea lo que se borrara.

**MENU RENOMBRAR**



Esta es la función de autocompletar el textbox en el formulario de Menú Baja, en este se observa que se establece una conexión con la Base de Datos y todos sus valores de una cierta columna se agregan al textbox, además de lo que hace el botón Salir

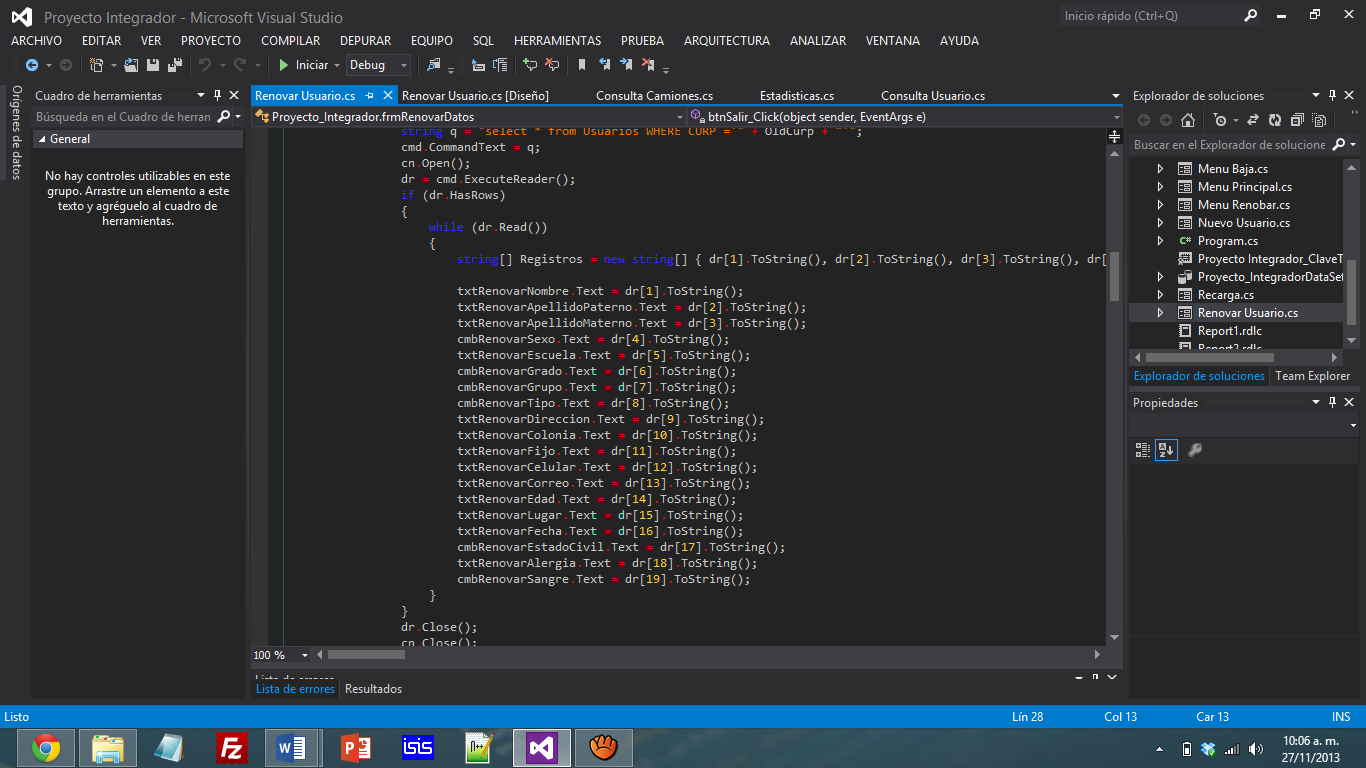


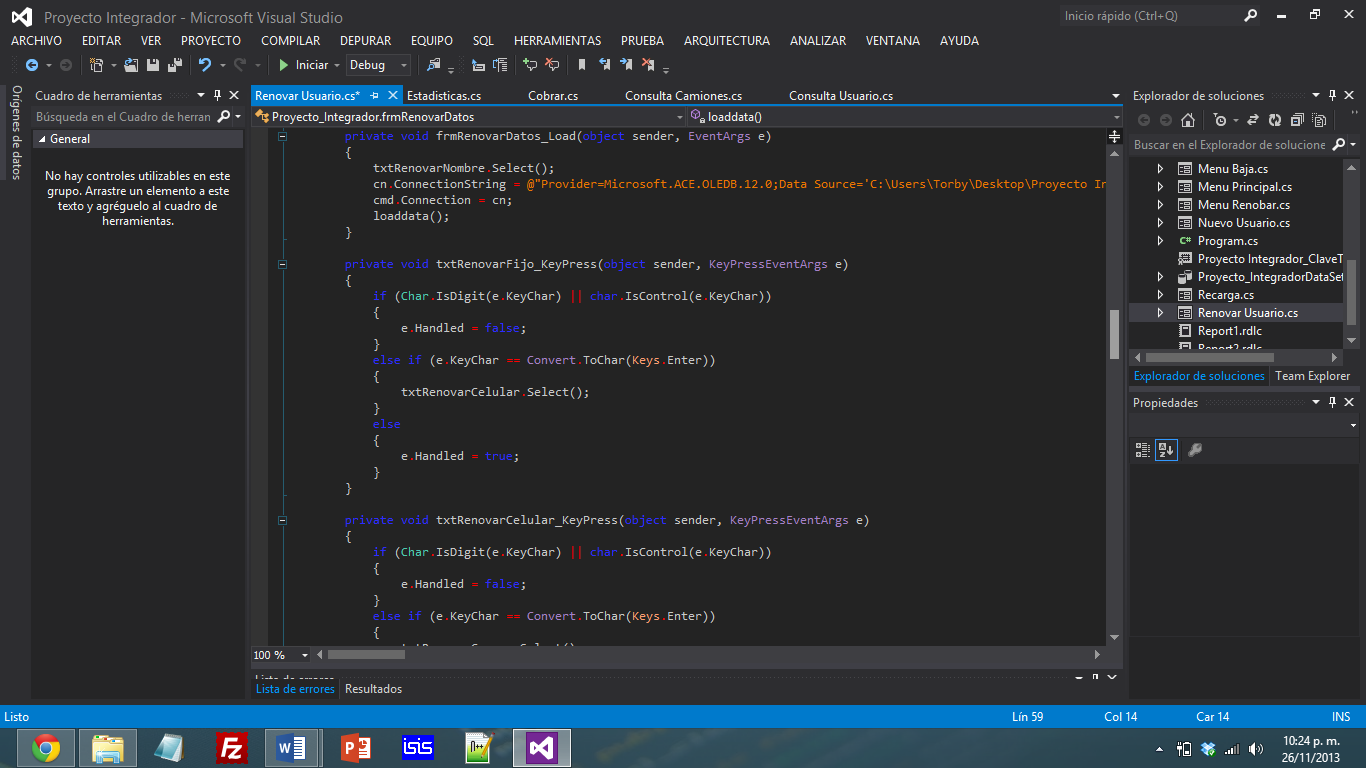
Aquí se aprecia lo que hace el botón Renovar, el botón salir, así como lo que se hace al cargar el formulario y lo que realiza la función buscar.

**RENOVAR USUARIO**



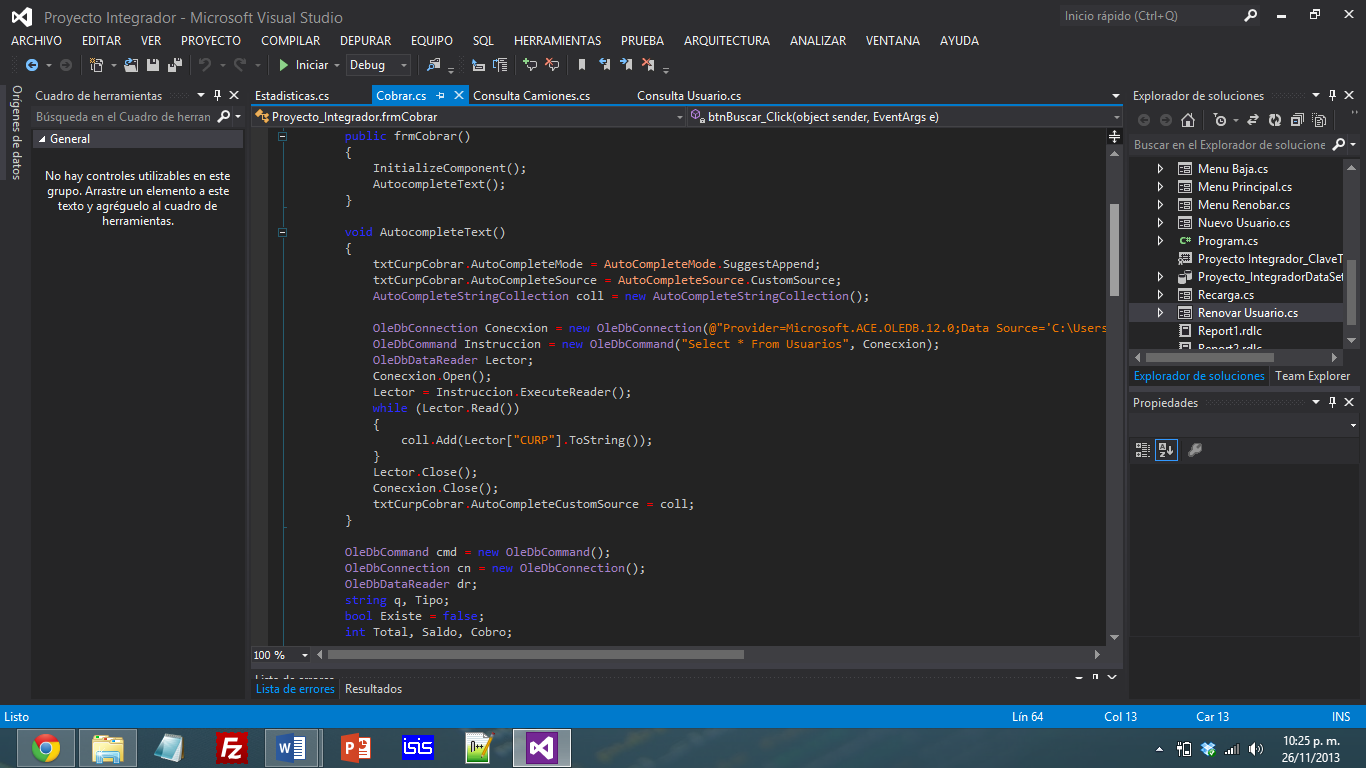
Aquí se observa la conexión con la base de datos y la manera de como modificar los valores de la base datos por los escritos por el usuario



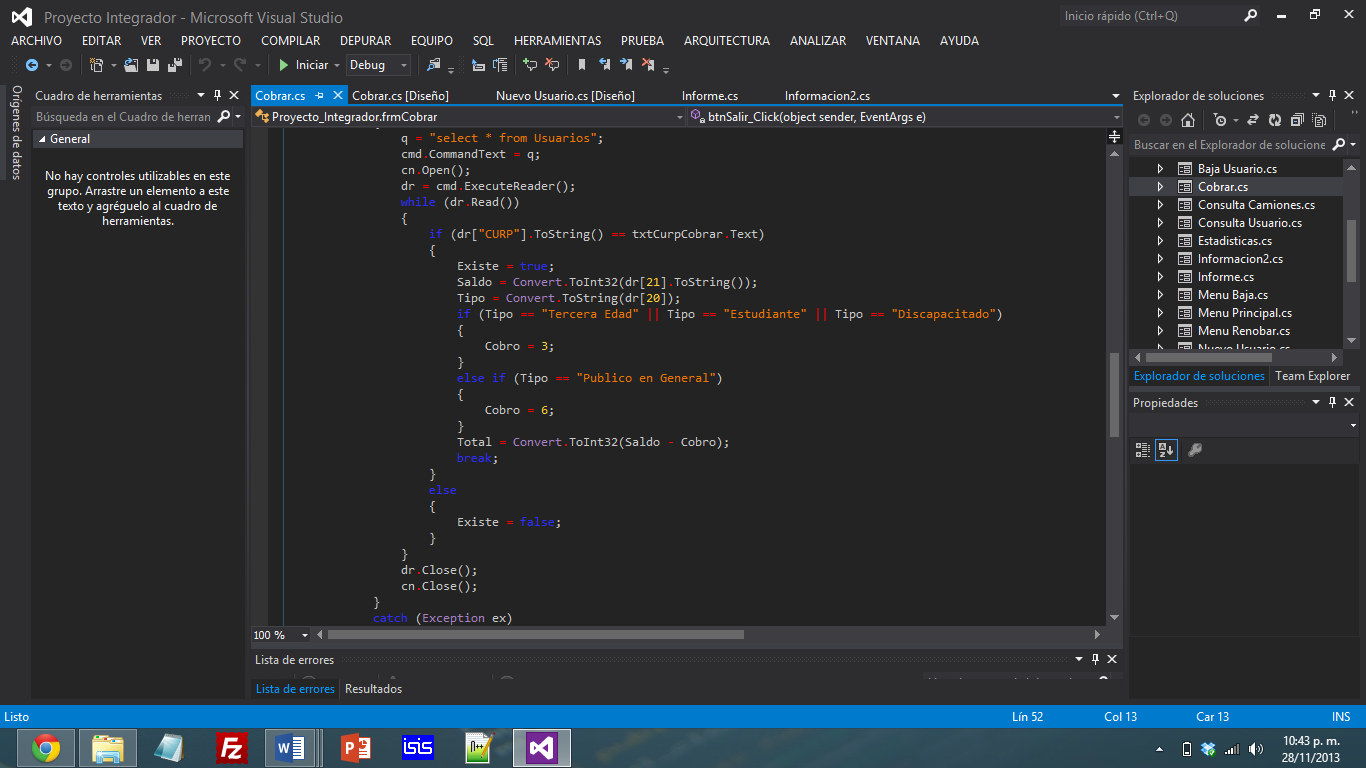


Aquí se muestra donde está ubicada la base de datos. También las reglas necesarias para que solo se acepten números en el campo teléfono fijo.

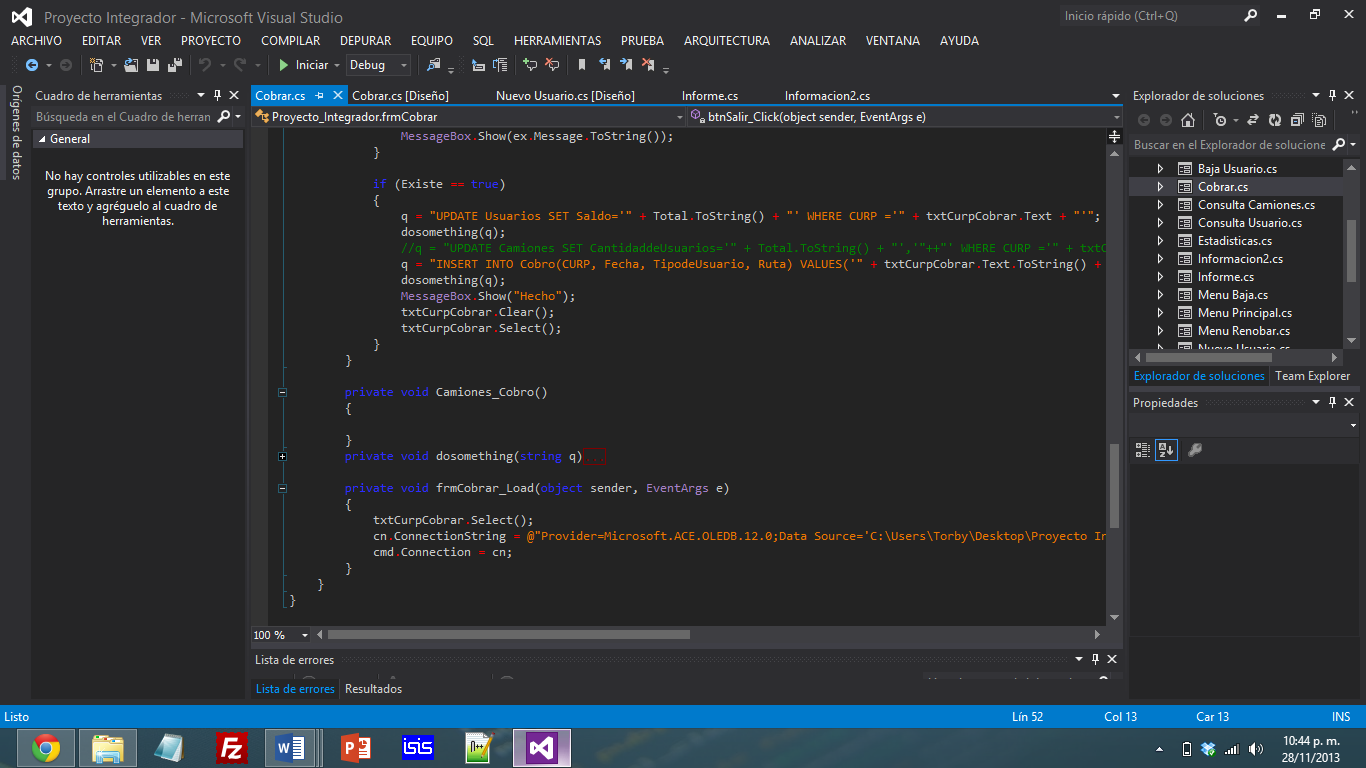
**COBRO**



Esta es la función de autocompletar el textbox en el formulario de Menú Baja, en este se observa que se establece una conexión con la Base de Datos y todos sus valores de una cierta columna se agregan al textbox, además de otras variables necesarias.

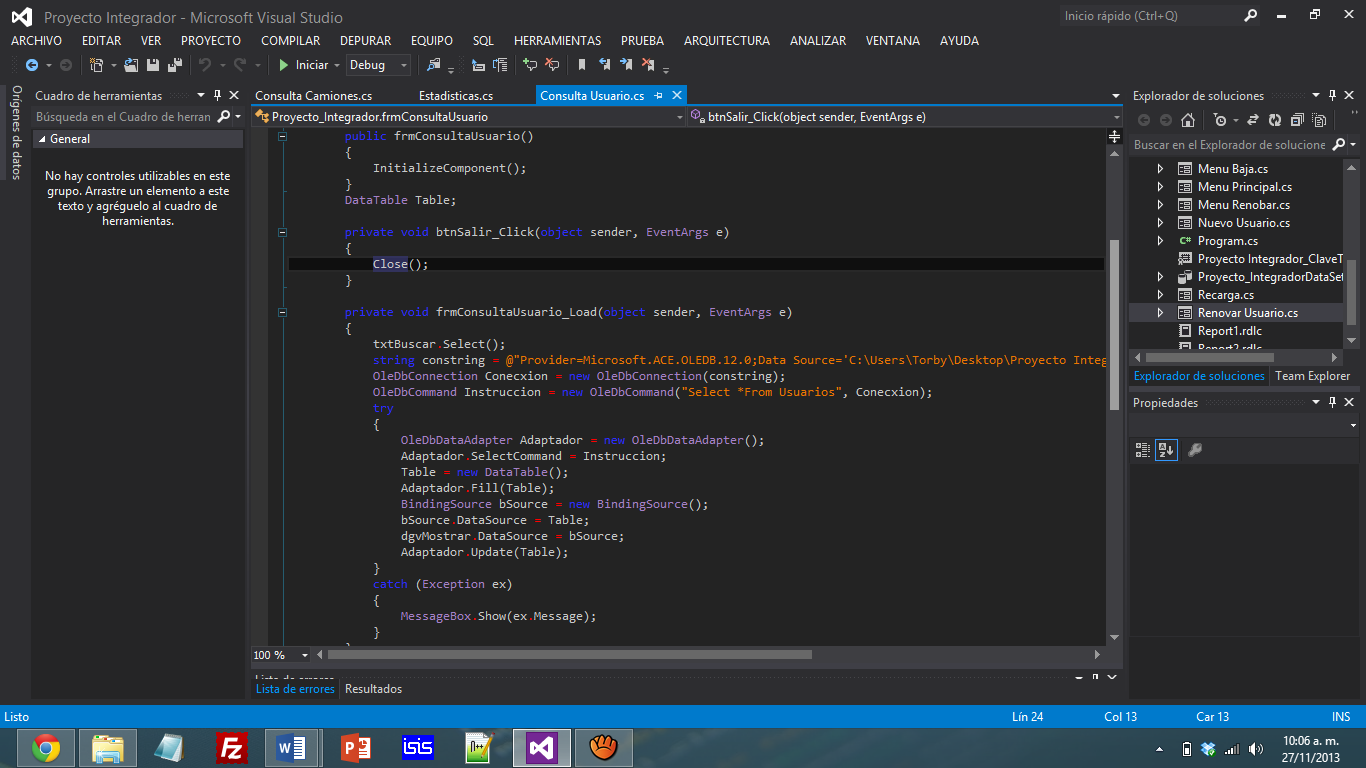


Aquí se observa es que se leen los datos de un usuario en específico para así determinar la cantidad que se cobrara.

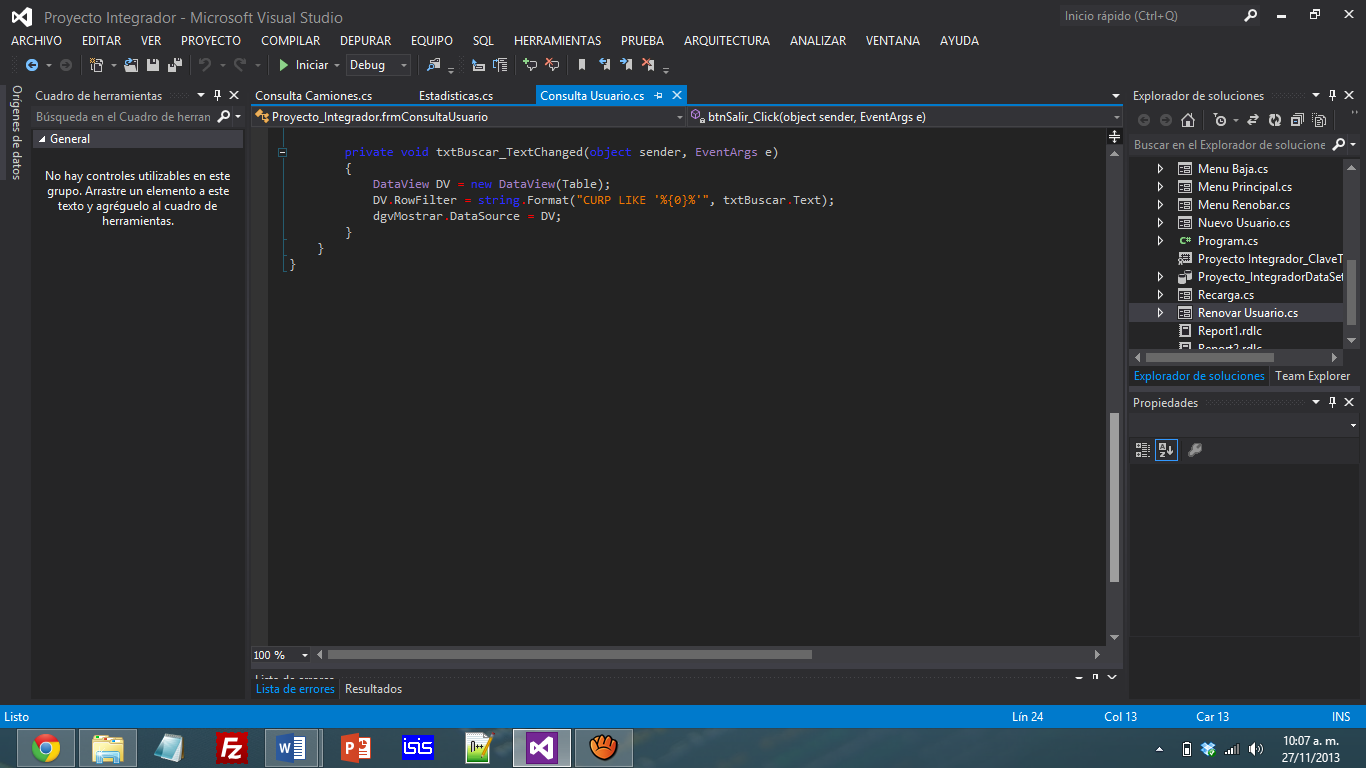


Aquí se muestra la ubicación de la base de datos, asimismo la cantidad que ahora tiene el usuario y se inserta un nuevo campo donde se mencione la ruta, la fecha y el usuario que se subió a esa ruta.

**CONSULTA USUARIOS**

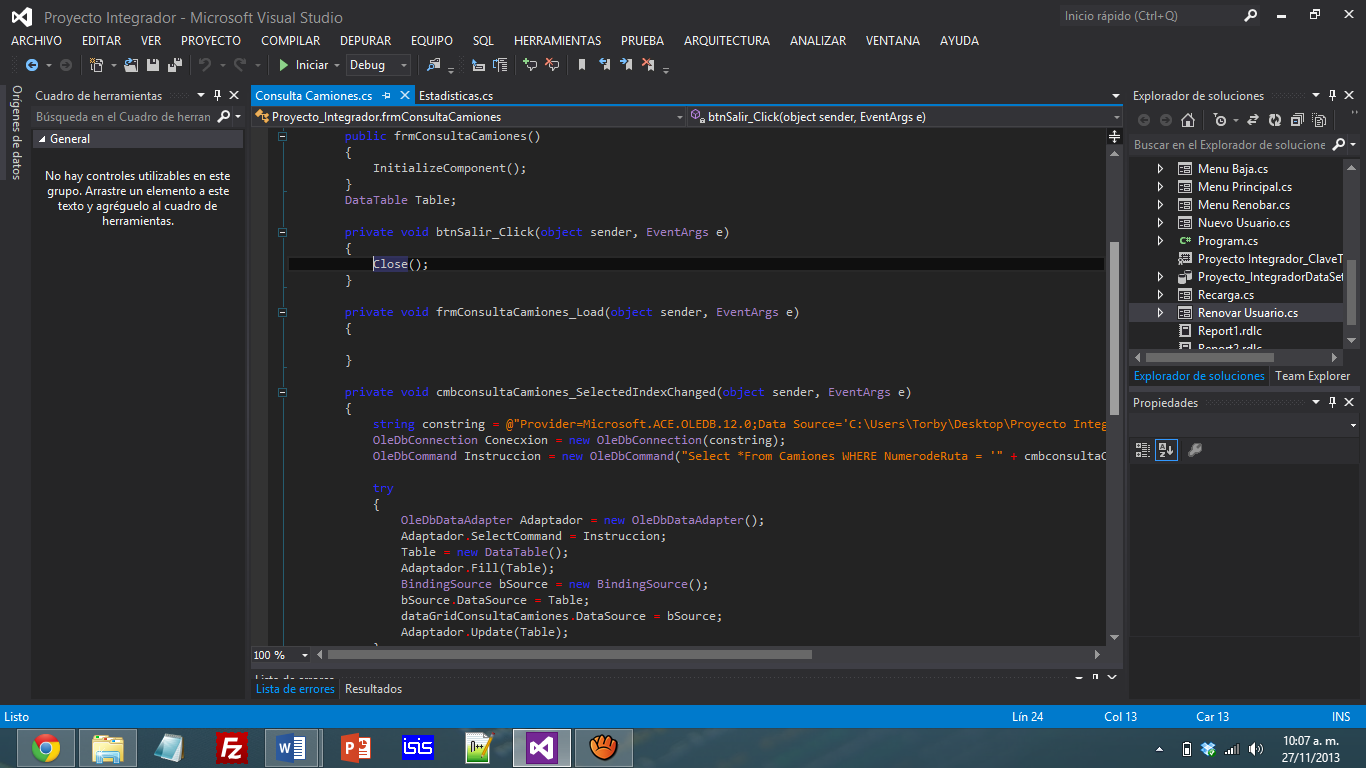


En esta imagen se alcanza a apreciar la ubicación de la base de datos, así como la conexión y la manera de cómo se llena el DataGrid



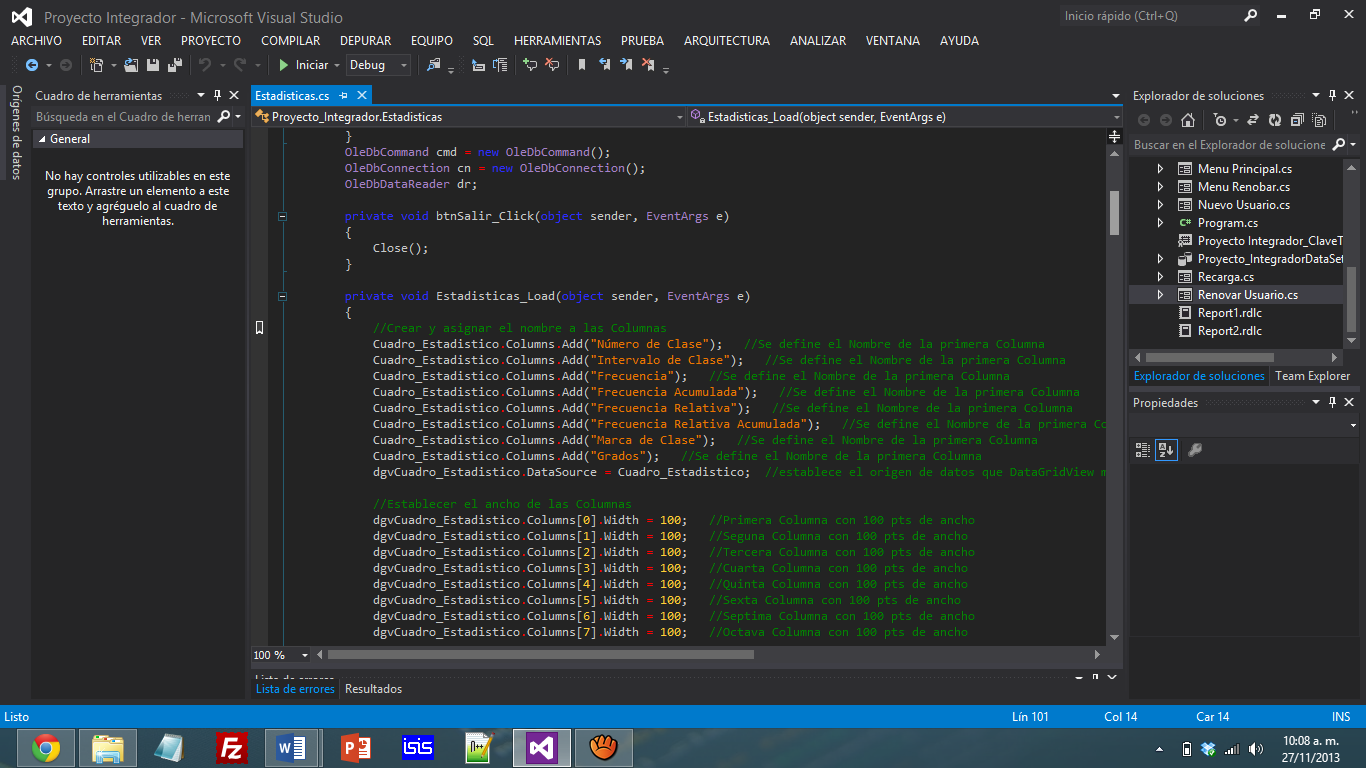
Aquí se muestra lo necesario para que cuando el usuario este escribiendo algo en el texbox, se busque automáticamente en el DataGrid.

**CONSULTA CAMIONES**

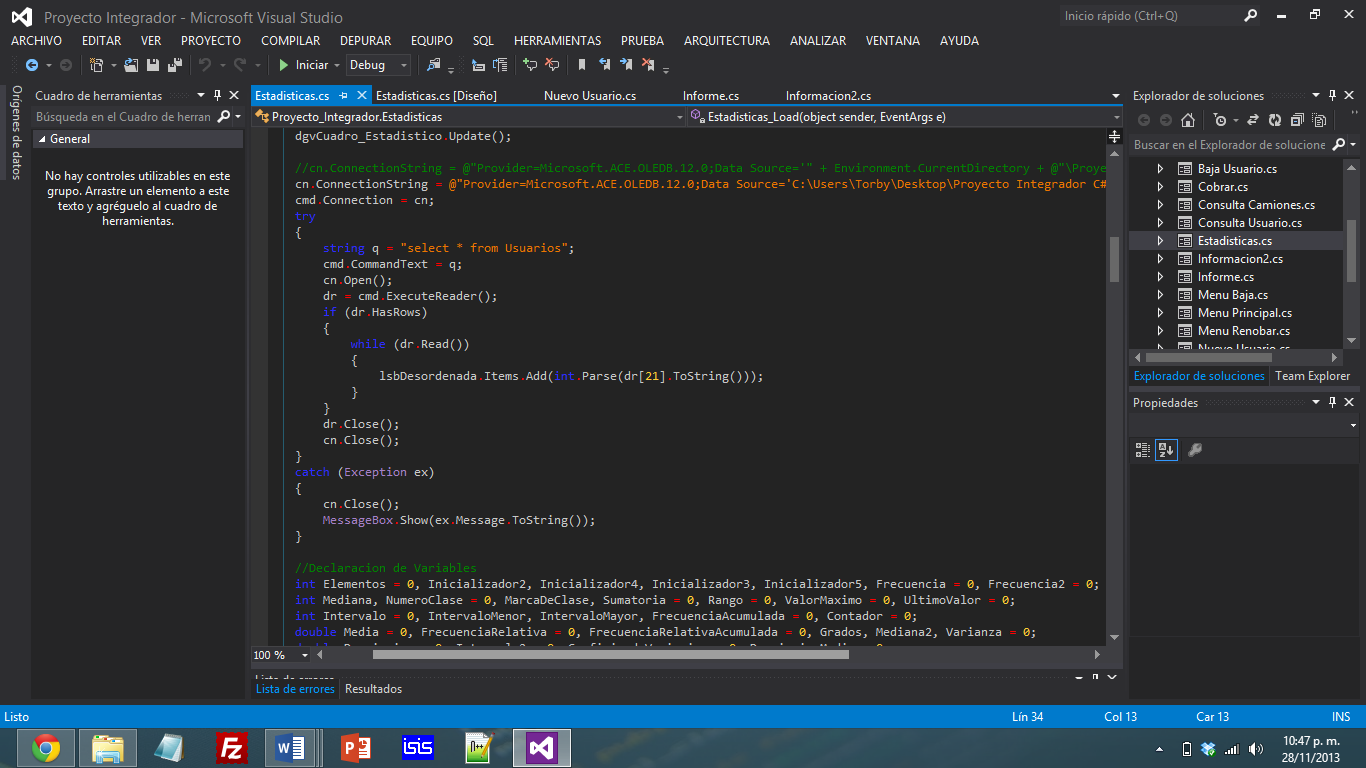


Aquí se muestra lo que hace el botón Salir, además del código necesario para la ubicación de la base de datos, también la manera de cómo llenar el DataGrid de este formulario.

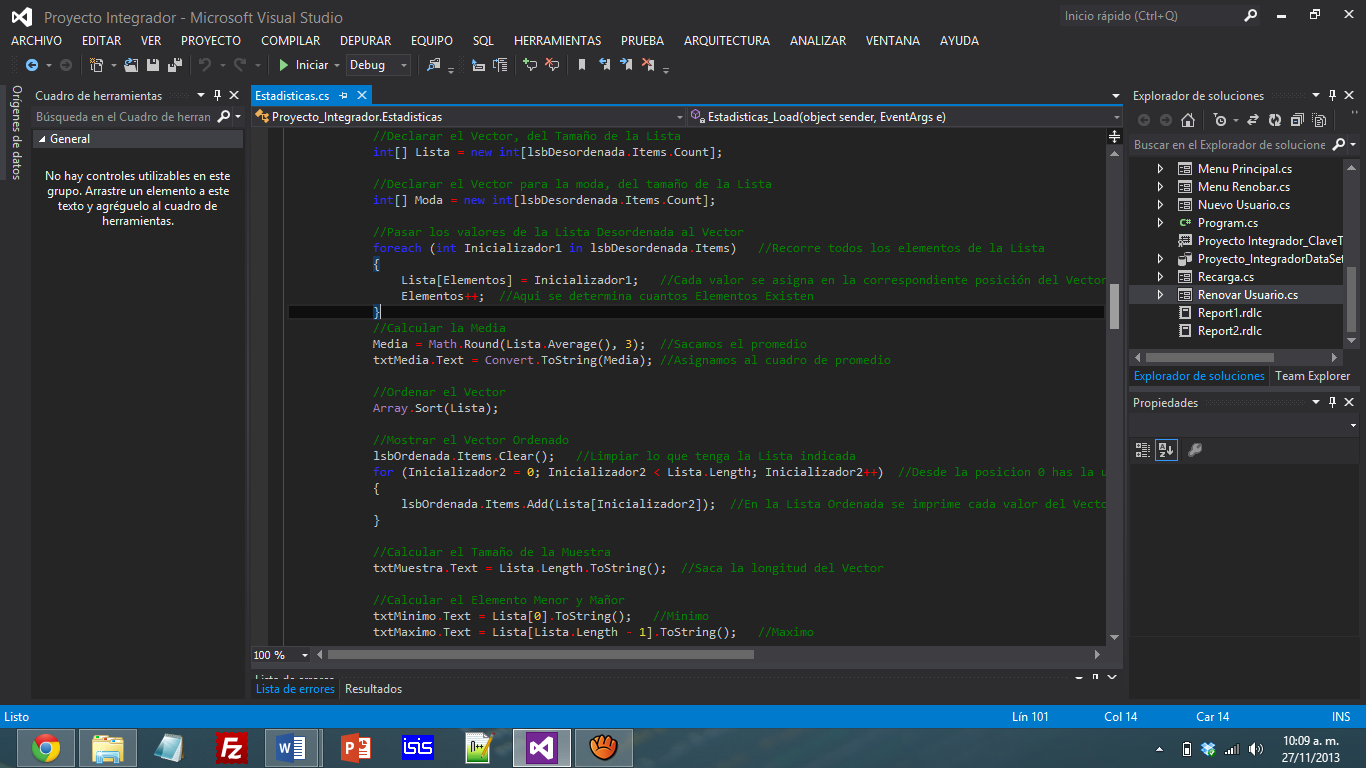
**ESTADISTICAS**



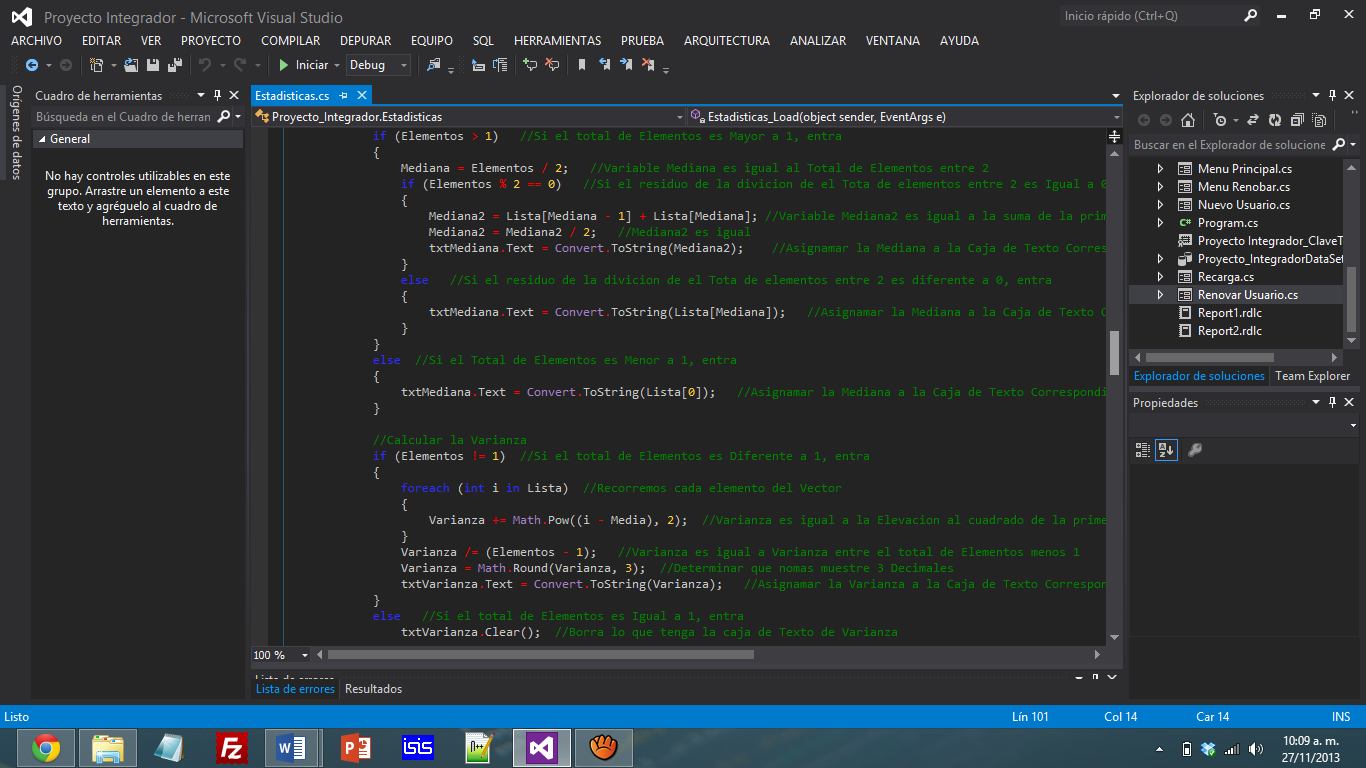
Aquí se muestra la manera de cómo está estructurada la base de datos temporal, así como el tamaño y los nombres de las columnas.



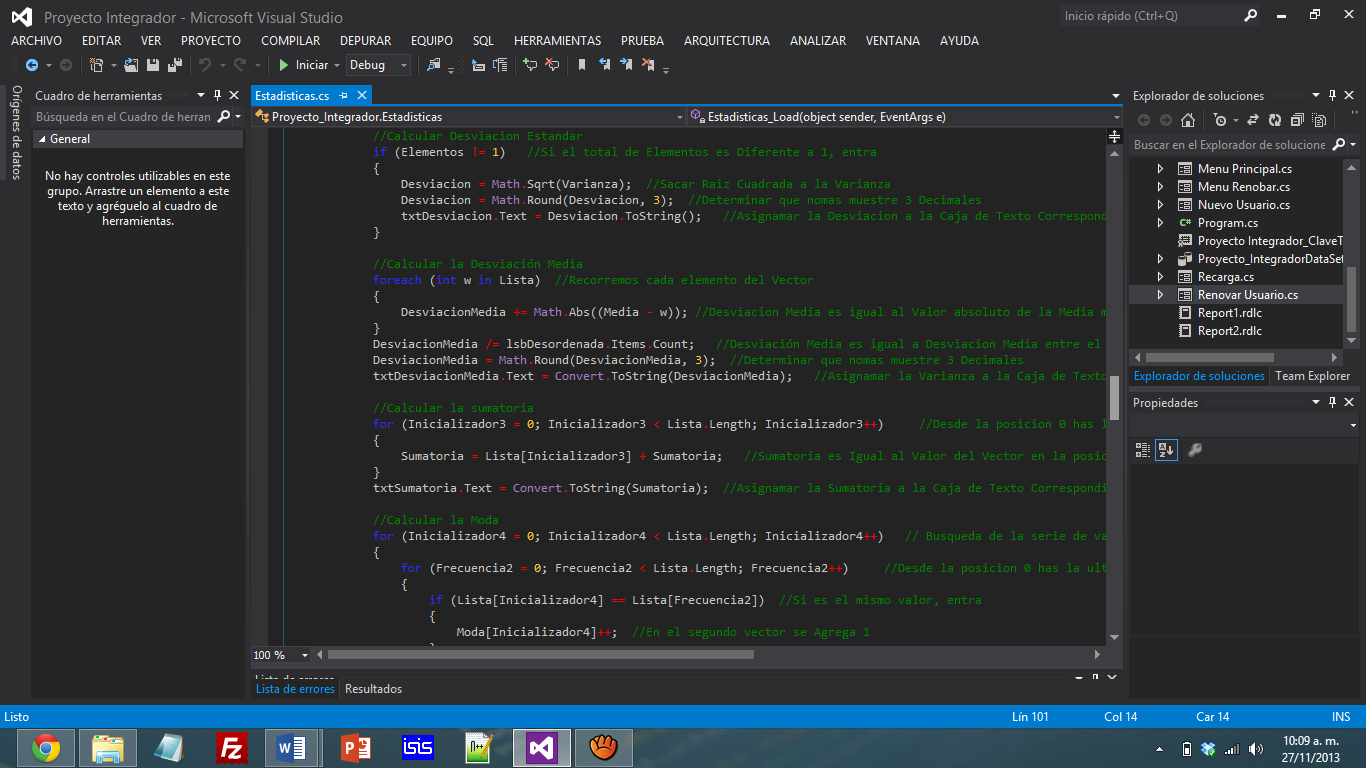
En esta imagen se ven algunas de las variables necesarias para el programa de Estadísticas. También la conexión con lavase de datos y la obtención de los saldos de cada una de las personas.



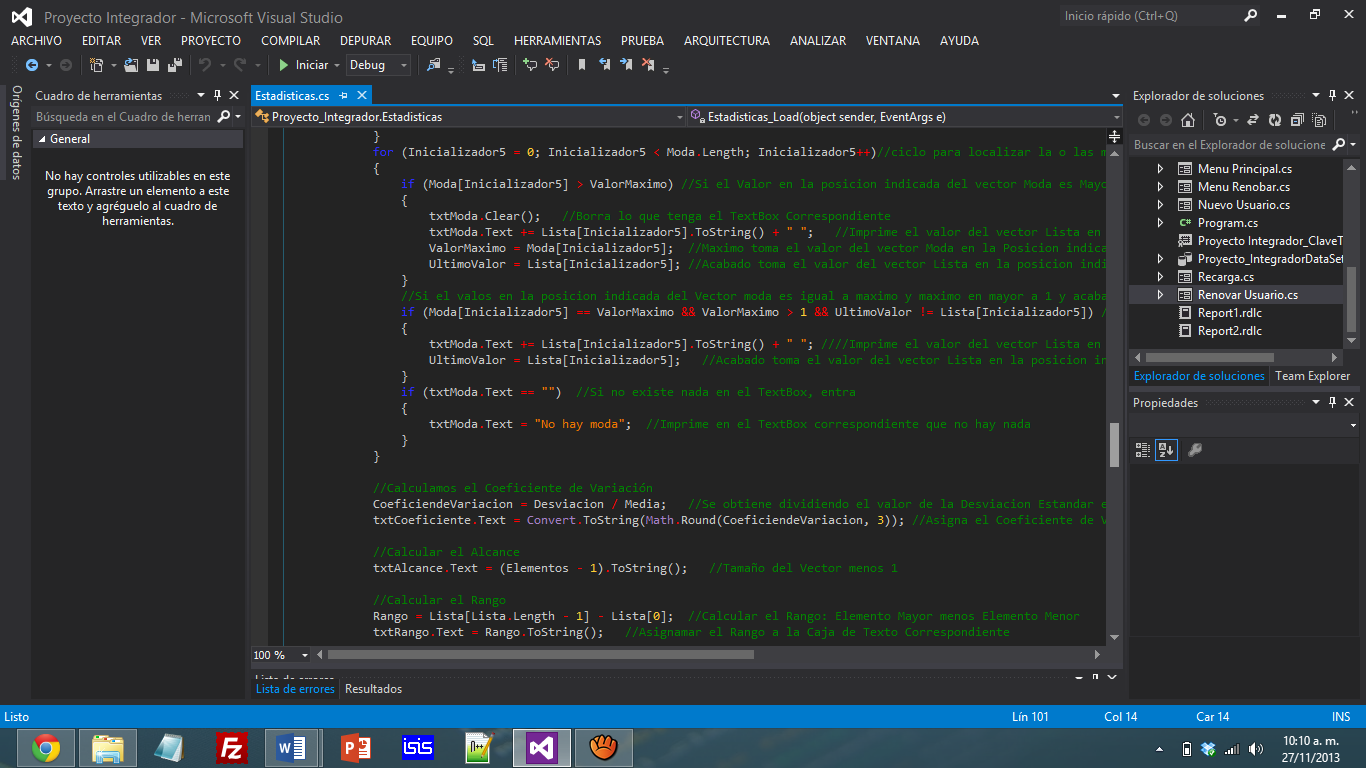
Aquí son algunas de las operaciones necesarias para empezar con los cálculos de las medidas de tendencia central, por ejemplo para calcular el tamaño de la muestra.



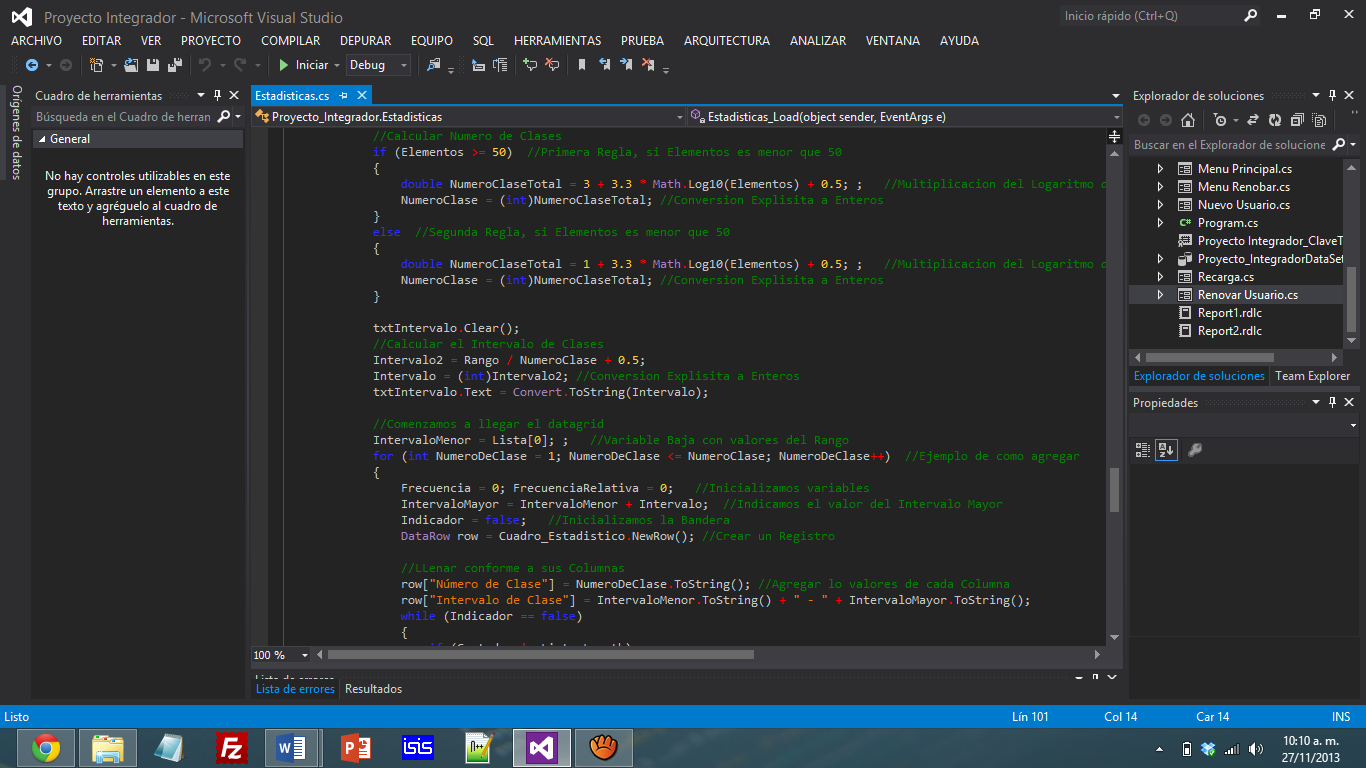
De igual manera se muestran más operaciones, en este caso para obtener la varianza y la moda.



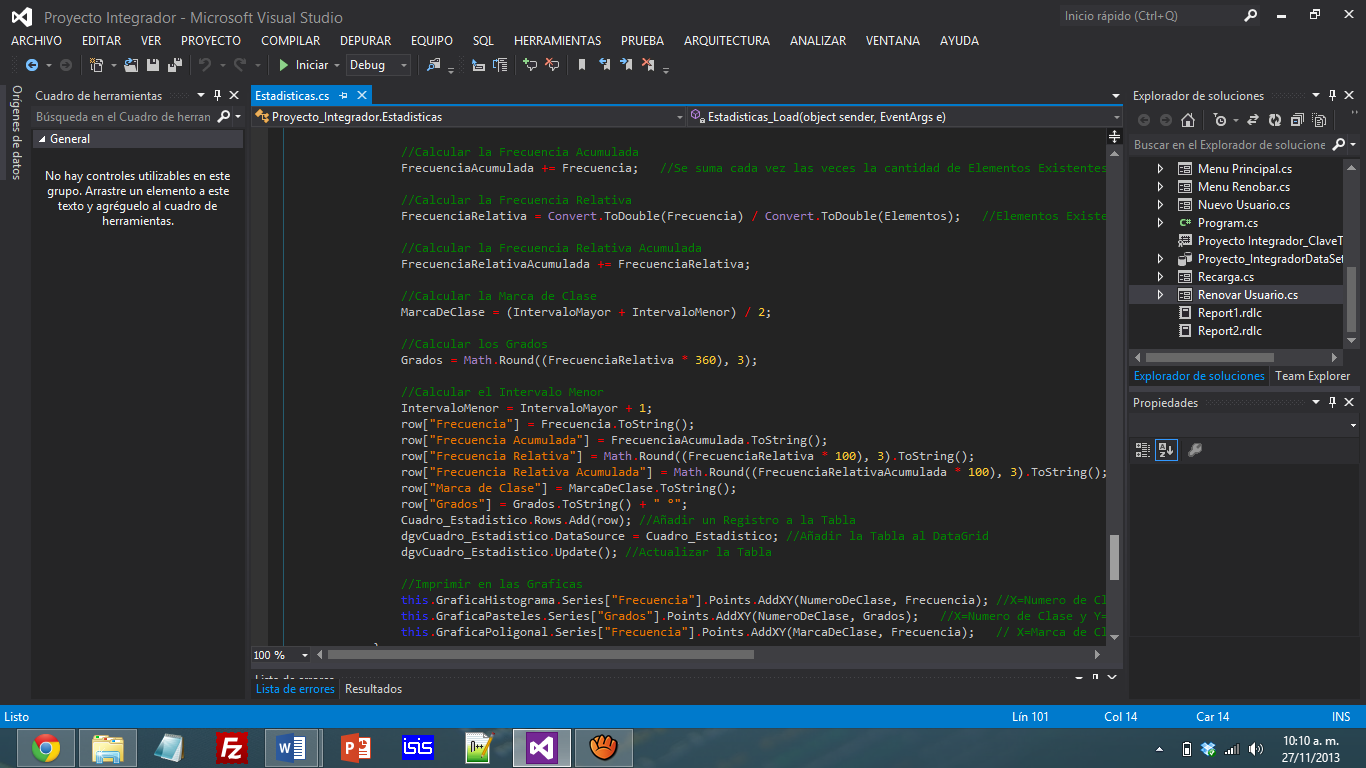
Más operaciones, aquí para obtener la Desviación Estándar, la desviación media, la sumatoria, etc.



Más operaciones para obtener el coeficiente de variación, el alcance y el rango

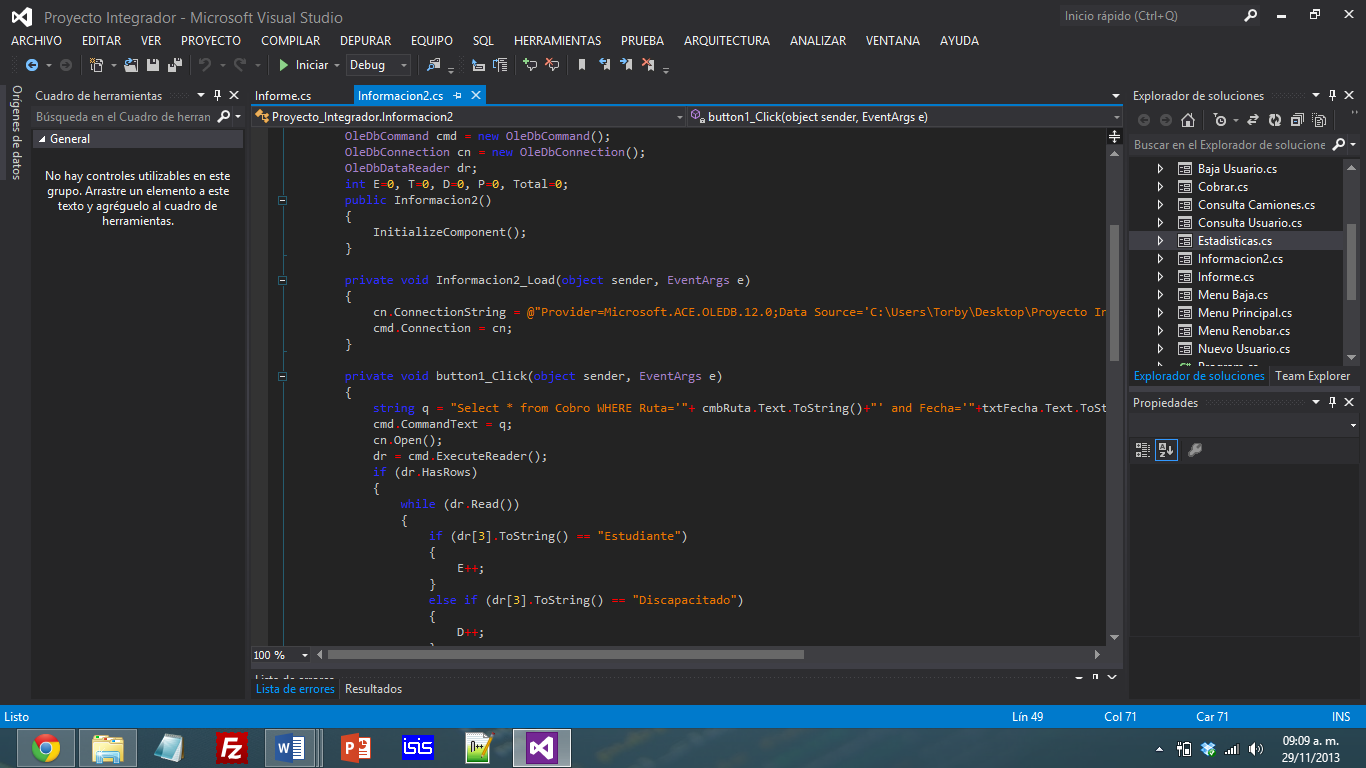


Más operaciones.

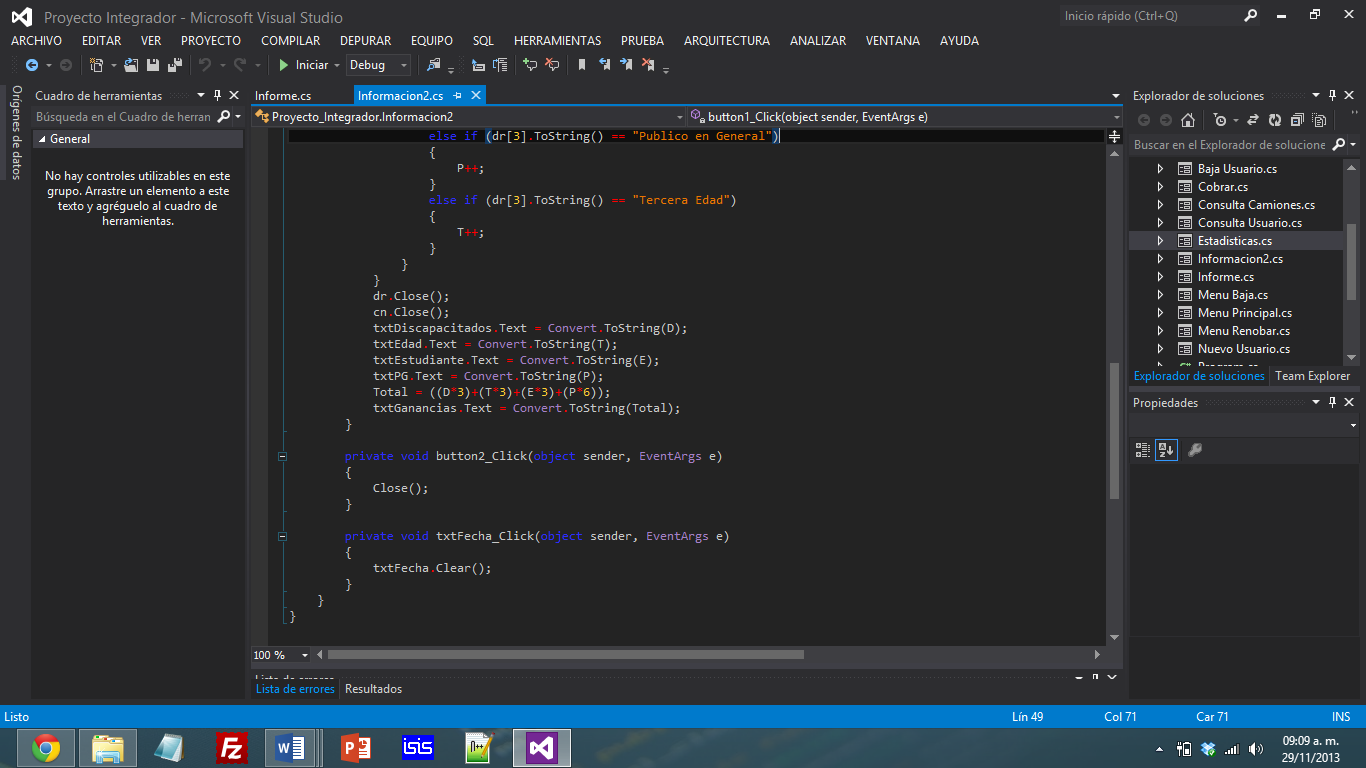


Ya por último se muestran los resultados en el DataGrid y en las gráficas necesarias.

**INFORME POR CAMIONES.**

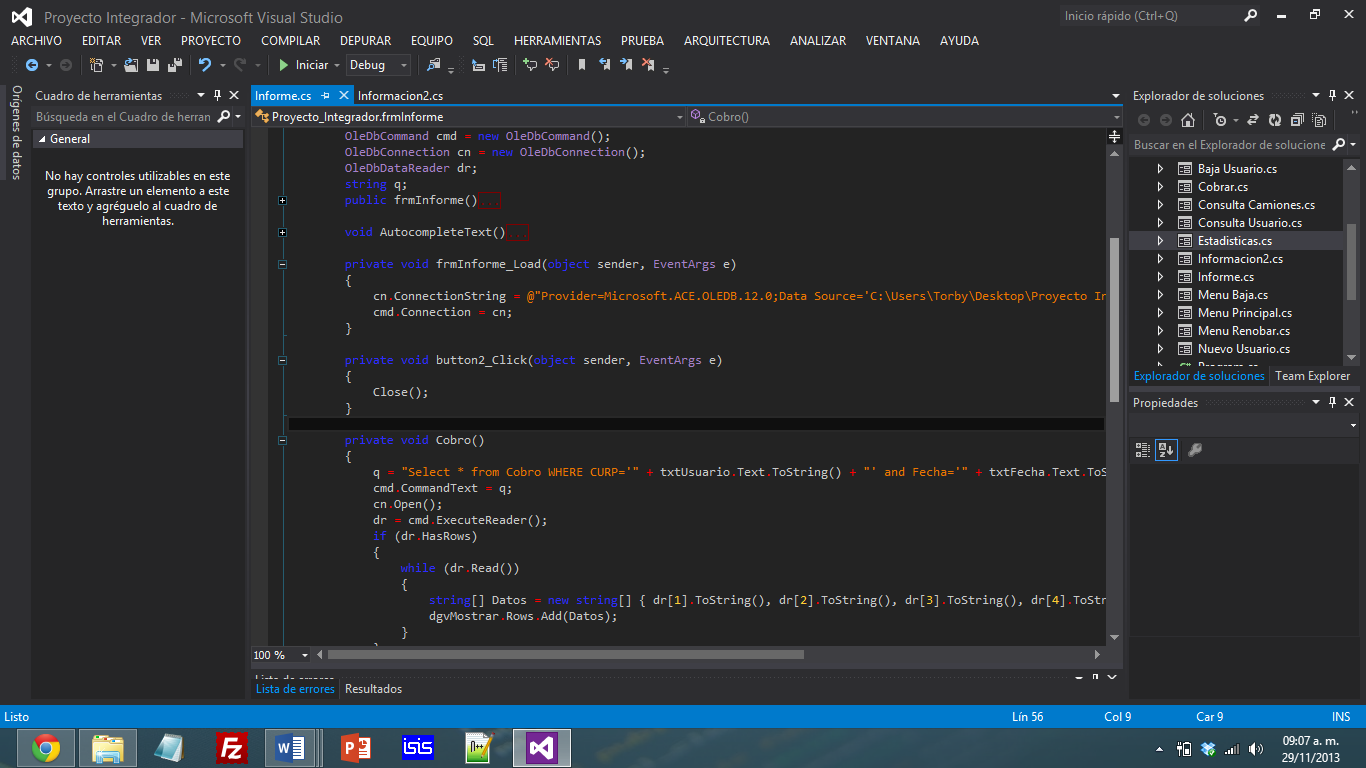


Aquí se establece una conexión con la Base de Datos, de igual manera se establecen variables necesarias. En el botón Buscar, lo que se hace es leer los datos de una de las tablas en un rango de filas específicas, después determinamos los tipos de usuarios de cada campo antes mencionado.

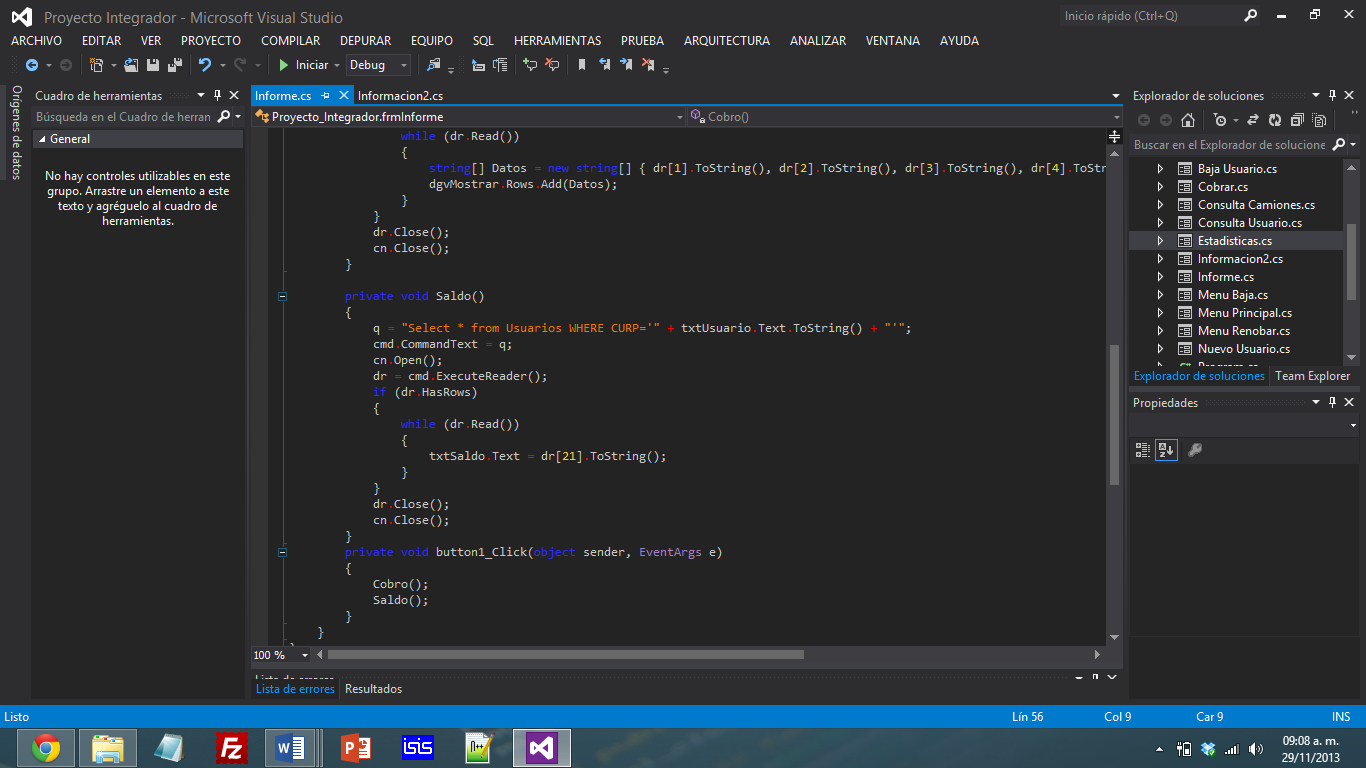


Aquí continuamos con la identificación de usuarios, además mostramos los datos en textbox y calculamos que debería tener un camión específico ese día.

**INFORME POR USUARIO.**

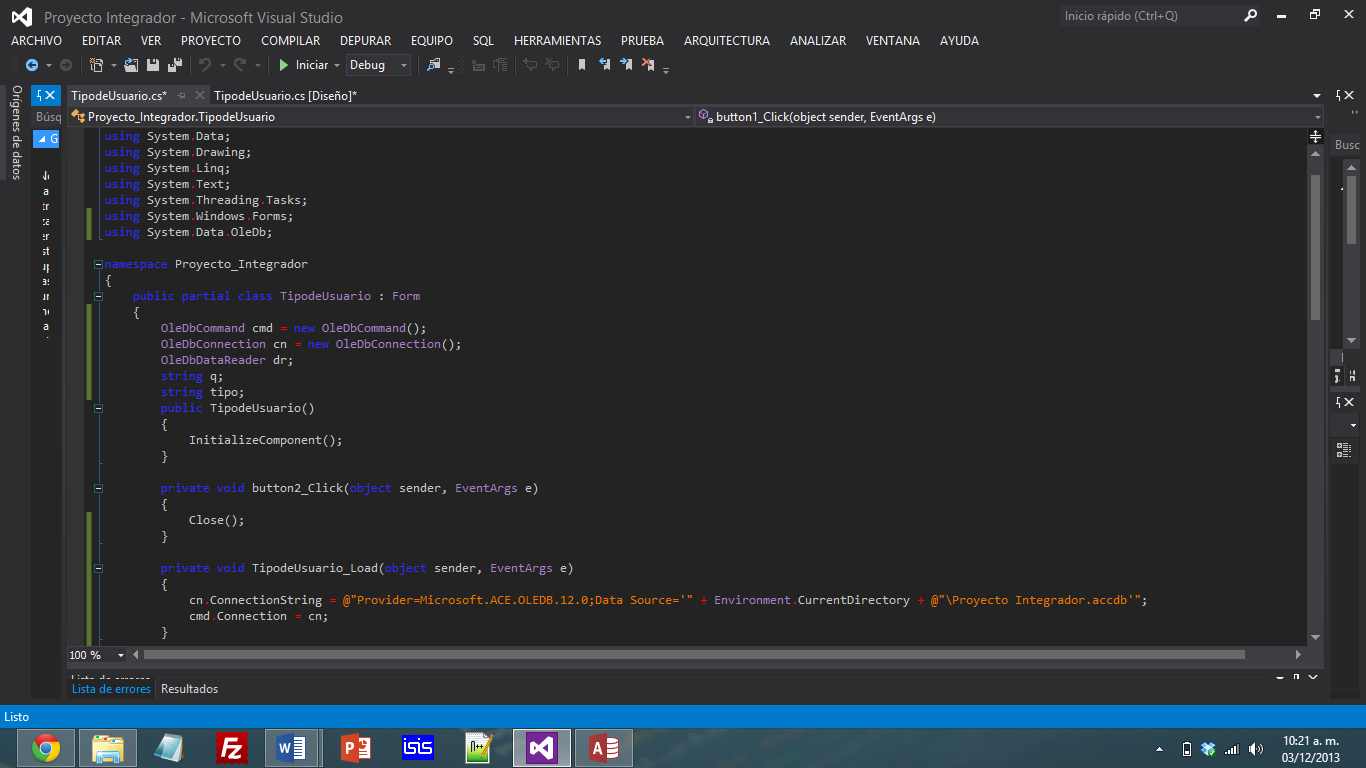


Aquí se establece una conexión con la base de datos, y se declaran variables necesarias, posteriormente se leen datos de esta y se muestran en un DataGrid.

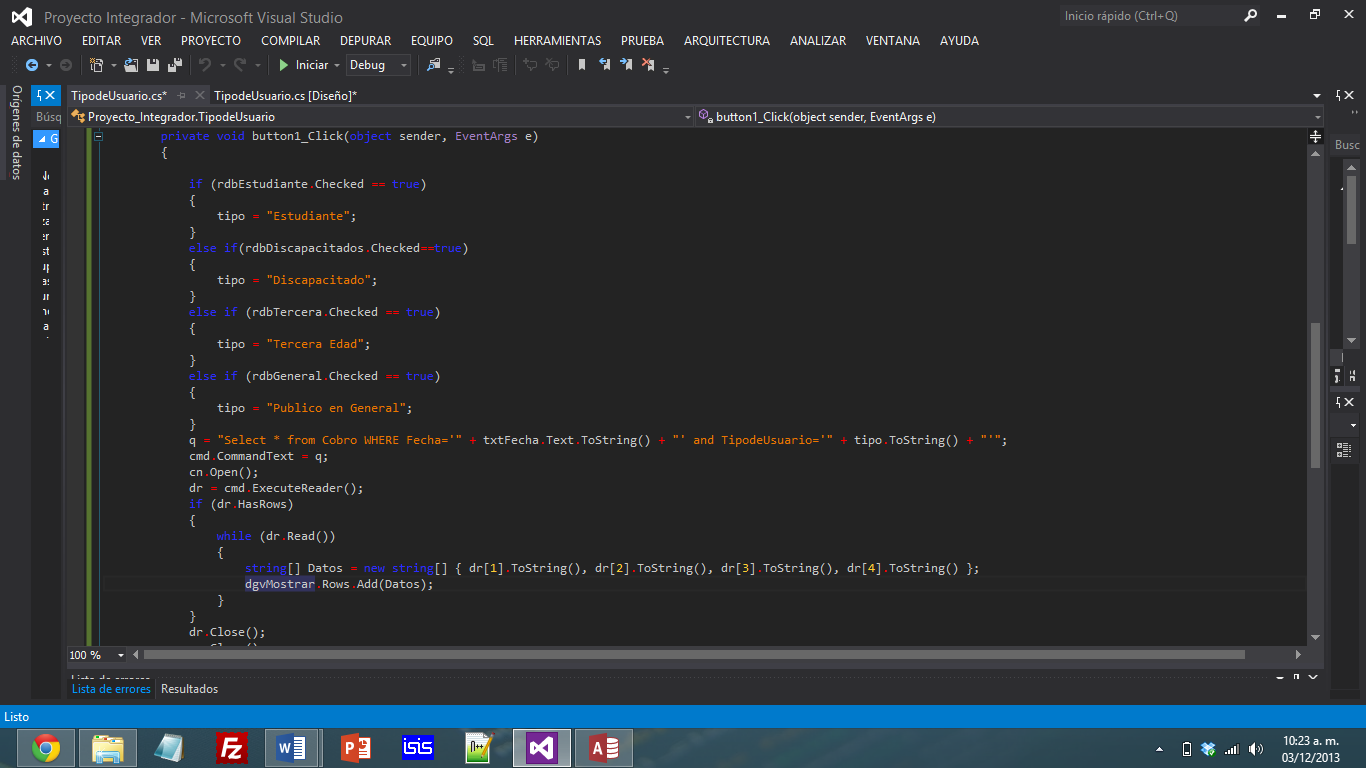


Aquí se leen los datos de otra tabla, pero a diferencia de lo anterior, sirve para obtener el saldo de la persona en curso.

**INFORME POR TIPO DE USUARIOS:**



En esta ventana se observa que se crean variables para poder conectarnos con la Bas de Datos, de igual manera la dirección de esta y el comportamiento del botón Salir.



En esta ventana se observa las condiciones para leer de la base datos, tomando en cuenta la fecha y el RadioButton seleccionado, después los valores se basan en un vector y posteriormente se muestra en un DataGridView.

# Soporte Técnico

Problemas y soporte técnico a los siguientes correos electrónicos y números telefónicos:

|  |  |  |
| --- | --- | --- |
| NOMBRE | E-MAIL | TELEFONOS |
| Amayelli Itzel Silva Contreras | Xx\_liliana\_xx@hotmail.com | (044)3123202658 |
| Edsel Barbosa González | torby@outlook.com | (044)3121282869 |
| José Luis Garza Gallegos | jgarza@ucol.mx | (044)3123100517 |
| Kevin Roberto Gómez Peralta | robertokev@hotmail.com | (044)3121524394 |