

Tarea Programada 1 – Partes de automóviles

Joseph Tenorio Pereira (2019064588)

Jose Pablo Muñoz Montero (2019061904)

1. Introducción

El presente proyecto tiene por objetivo la implementación de una base de datos y aplicación respectiva para un negocio de venta de repuestos. Dicha implementación busca crear una herramienta que permita guardar y administrar la información de varios aspectos del negocio. A grandes rasgos, se requiere almacenar y manipular la información referente a aspectos tales como clientes, partes, proveedores y órdenes. Este negocio tiene almacenadas múltiples partes de repuestos, las cuales son útiles para diversos modelos de automóviles y también son suministradas por distintos proveedores a distintos precios. Estas partes tienen una marca y un fabricante que puede ser de preferencia para los clientes. Dichos clientes pueden ser personas u organizaciones, ellos ponen órdenes en el negocio donde solicitan distintas partes de su proveedor de elección. El negocio también almacena la información de clientes suspendidos e inactivos.

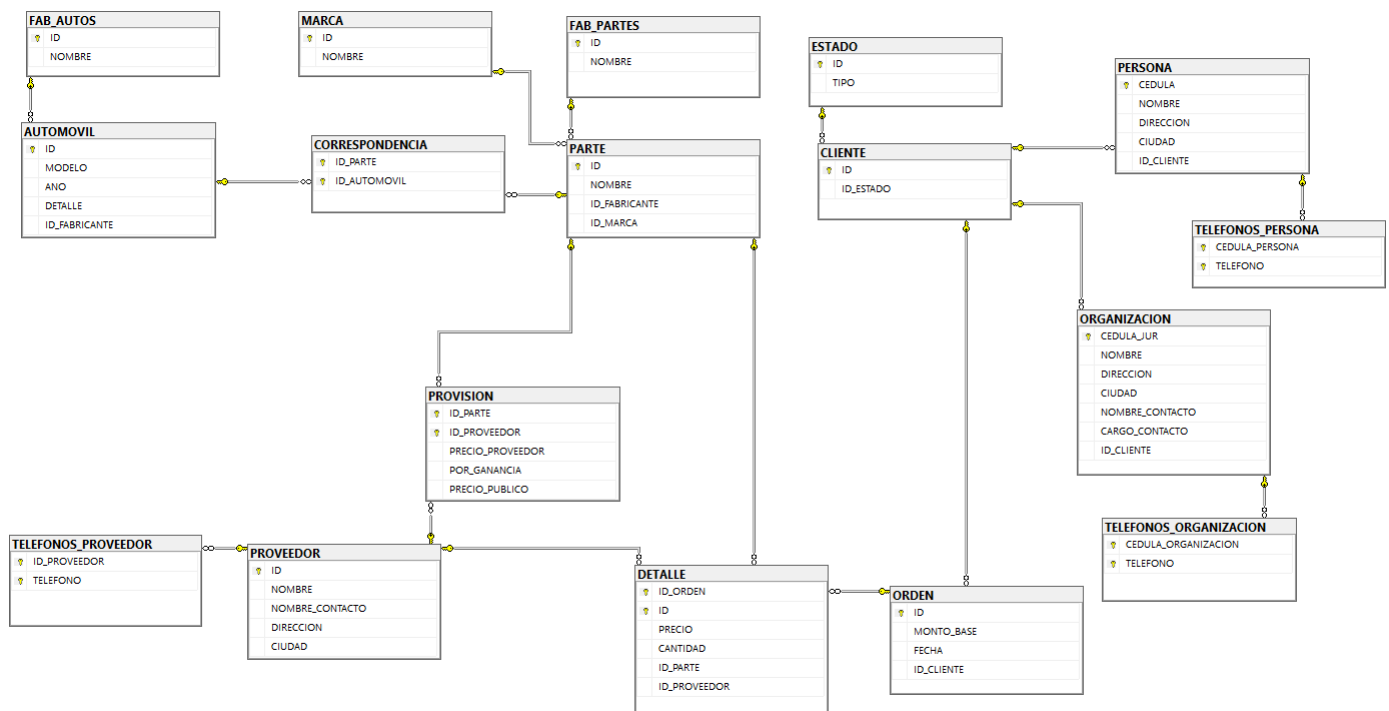
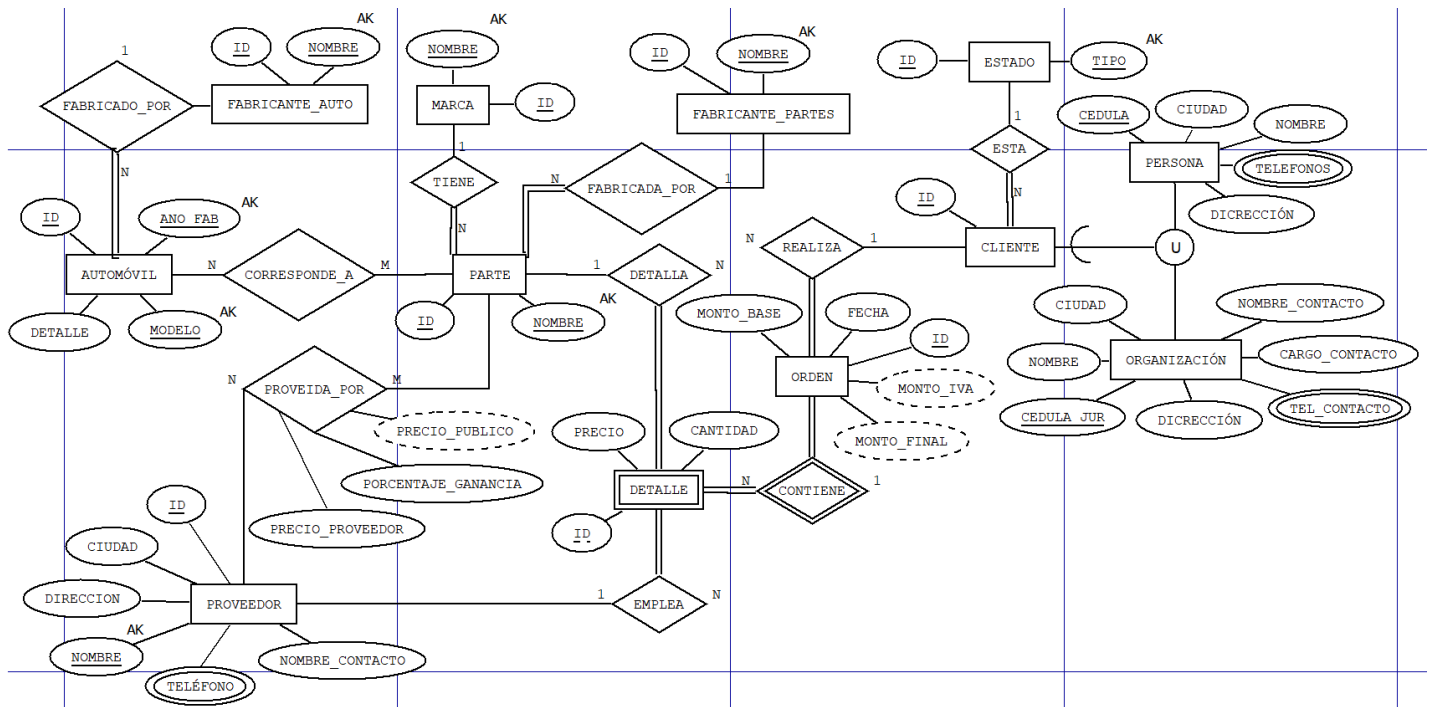
Una vez diseñada una base de datos que se adecue al almacenamiento de la información antes mencionada, se procedió a la construcción de una aplicación para facilitar las funcionalidades del usuario en dicha base de datos. Algunas funcionalidades implementadas son las referentes a aspectos como insertar información de nuevos clientes o partes, asociar las partes con sus respectivos automóviles y proveedores, poner órdenes para los clientes y visualizar ordenadamente la información almacenada. De esta manera los usuarios en el negocio no deben preocuparse por interactuar directamente con la base de datos, y pueden sencillamente manipular su información.

2. Ambiente de Desarrollo

Las herramientas utilizadas para el desarrollo de este proyecto son:

- Microsoft SQL Server Management Studio 2018
- Visual Studio 2019 (incluyendo la carga de trabajo de almacenamiento y procesamiento de datos y los proyectos de integración de servicios de *SQL Server*, obtenibles por medio del paquete *SQL Server Data Tools*)
- Apache NetBeans 11
- Dia Diagram Editor

3. a) Diagramas



Nota: En caso de querer crear la base de datos diseñada a partir del *script* adjuntado a este documento, favor seguir las breves instrucciones dentro del mismo (comentarios al principio del *script*) antes de ejecutarlo.

3. b) Diccionario de Datos

1. Tabla Automóvil: contiene la información respectiva de cada automóvil registrado. Incluye un identificador, el modelo, año, un detalle y una referencia a su fabricante
2. Tabla Cliente: contiene únicamente un identificador único para cada cliente con una referencia a su estado
3. Tabla Correspondencia: contiene la información referente a cuales partes corresponden a cuales automóviles. Almacena una referencia a una parte y un automóvil
4. Tabla Detalle: contiene la información de compra para las partes de ciertos proveedores. Almacena un identificador, una referencia a la parte y al proveedor junto con el precio total y la cantidad deseada además de una referencia a la orden a la cual pertenece
5. Tabla Estado: contiene los diferentes estados disponibles para los clientes
6. Tabla Fab_Autos: contiene los nombres de todos los fabricantes de autos registrados
7. Tabla Fab_Partес: contiene los nombres de todos los fabricantes de partes registrados
8. Tabla Marca: contiene los nombres de todas las marcas de partes registradas.
9. Tabla Orden: contiene la información referente a las órdenes de los clientes a las cuales les pertenecen una serie de detalles. Almacena un identificador, el costo base de la orden, del cual se deriva el costo del IVA y el costo total, la fecha y una referencia al cliente que le pertenece.
10. Tabla Organización: almacena la información de todas las organizaciones registradas, como su nombre, cedula jurídica, dirección, ciudad, el nombre de un contacto y su cargo, además de la referencia al cliente al que pertenece esta información.
11. Tabla Parte: contiene la información de las partes, como un identificador, su nombre y referencias a su fabricante y marca.
12. Tabla Persona: almacena la información de todas las personas registradas, como su nombre, cedula, dirección y ciudad, además de una referencia al cliente al que pertenece dicha información.
13. Tabla Proveedor: almacena la información de todos los proveedores registrados, como un identificador, su nombre, dirección, ciudad y el nombre de un contacto.
14. Tabla Provisión: contiene la información referente a cuales partes son suministradas por cuales proveedores. Contiene una referencia a una parte y un proveedor además del precio que ofrece el proveedor, el porcentaje de ganancia y el precio disponible al publico
15. Tabla Telefonos_Organizacion: contiene los teléfonos de cada organización con una referencia a esa misma
16. Tabla Telefonos_Persona: contiene los teléfonos de cada persona con una referencia a esa misma
17. Tabla Telefonos_Proveedor: contiene los teléfonos de cada proveedor con una referencia a ese mismo.

4. Instrucciones de Uso de la Aplicación

Primero, antes de poder acceder a la aplicación es necesario configurar la función de log in ubicada en la función main del proyecto de aplicación. Esta función toma como parámetros: la dirección ip en que se ubica la base de datos, el nombre de un usuario administrador y la contraseña para acceder a la base de datos, en este mismo orden. Una vez que las credenciales correctas están escritas se puede ejecutar la aplicación.

Al iniciar la aplicación se presenta un menú de 5 opciones, las cuales serán detalladas cada una a continuación. Cabe resaltar que cuando no se esté utilizando la aplicación de la manera correcta esta misma se lo notificara con un mensaje de error, como por ejemplo al dejar algún espacio obligatorio en blanco. Además, cada tabla presente en la aplicación cuenta con una barra de búsqueda que permite filtrar la información mostrada en tiempo real, esta se ubica encima de cada tabla.

Al ingresar a la opción clientes se muestra una tabla con todos los clientes registrados en la base de datos. Las funcionalidades disponibles en esta ventana son:

- Agregar un cliente nuevo con el botón agregar cliente. Aquí se puede ingresar toda la información deseada y seleccionar un estado disponible. Al seleccionar el checkbox de organización se habilitan los espacios exclusivos para organizaciones. El botón aceptar guarda los cambios, y cerrar la ventana los elimina.
- Modificar un cliente existente al hacer click derecho sobre su nombre en la tabla y seleccionar modificar. En el menú modificar se pueden alterar los datos permitidos y además cambiar el estado del cliente. El botón aceptar guarda los cambios, y cerrar la ventana los elimina.

Al ingresar a la opción partes se muestra una tabla con todas las partes registradas en la base de datos. Las funcionalidades disponibles en esta ventana son:

- Agregar una nueva parte con el botón agregar parte. Se ingresan su nombre y selecciona una marca y fabricante existente. El botón aceptar guarda los cambios, y cerrar la ventana los elimina.
- Eliminar una parte al hacer click derecho sobre su nombre en la tabla y seleccionar eliminar. Se eliminara la parte obedeciendo las restricciones del enunciado.

Al ingresar a la opción autos se muestra una tabla con todas las partes que ya están asociadas con uno o varios automóviles. Las funcionalidades disponibles son:

- Relacionar una parte con un automóvil con el botón relacionar, donde se debe seleccionar una parte y automóvil (modelo y año) existente. El botón aceptar guarda los cambios, y cerrar la ventana los elimina.
- Buscar todas las partes que correspondan a un cierto automóvil al ingresar su modelo y año en las respectivas casillas y presionar buscar. Para regresar a una vista de todas las partes se puede presionar refrescar.

Al ingresar a la opción proveedores se muestra una tabla con todas las partes que estén relacionadas con uno o varios proveedores y su información de venta. Las funcionalidades disponibles son:

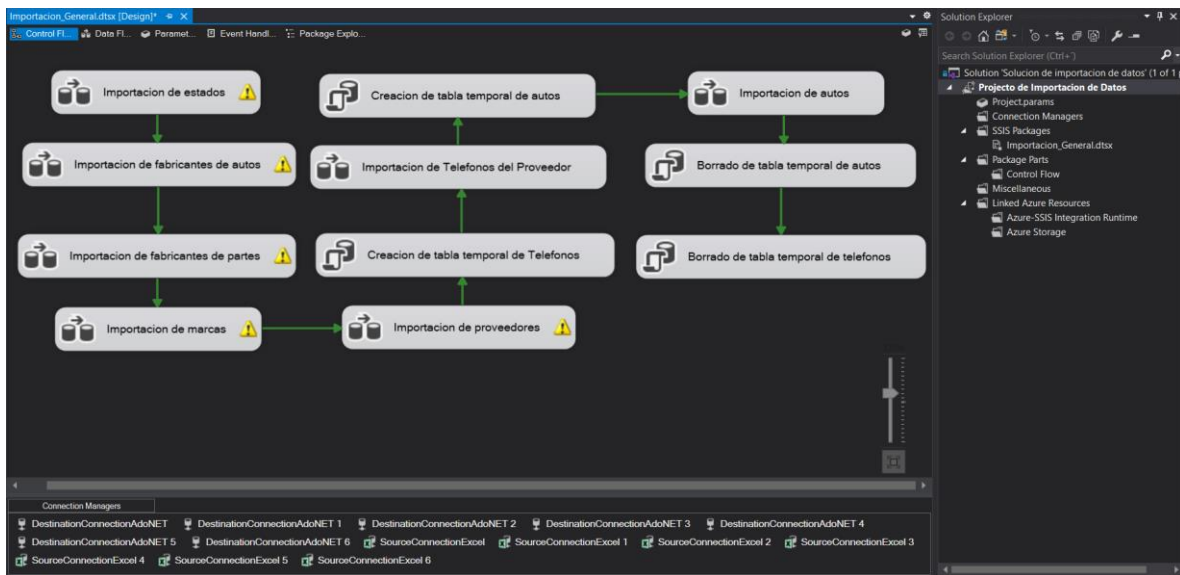
- Relacionar una parte con un proveedor con el botón relacionar. Aquí se selecciona una parte y proveedor existente y se debe ingresar el precio de proveedor y porcentaje de ganancia, el precio público se deriva de esos datos. El botón aceptar guarda los cambios, y cerrar la ventana los elimina.
- Modificar los precios de venta para una parte de cierto proveedor al hacer click derecho en su nombre en la tabla y seleccionar modificar. Aquí se pueden alterar los datos permitidos. El botón aceptar guarda los cambios, y cerrar la ventana los elimina.
- Buscar todos los proveedores que suministren una parte al ingresar su nombre en la casilla marcada y presionar buscar. Para regresar a una vista de todas las partes se puede presionar refrescar.

Al ingresar a la opción órdenes se muestra una tabla con todas las órdenes que se han hecho para los clientes del negocio. Las funcionalidades disponibles son:

- Crear una nueva orden con el botón agregar orden. Se selecciona el cliente y se ingresa la fecha en el formato requerido. El botón aceptar guarda los cambios, y cerrar la ventana los elimina.
- Agregar un detalle que especifique la compra de una cantidad de partes de cierto proveedor a una orden al hacer click derecho sobre su nombre en la tabla y seleccionar modificar. Aquí se pueden observar todos los detalles de la orden seleccionada y además seleccionar una parte y proveedor existentes e ingresar una cantidad para un nuevo detalle. El botón agregar guarda los cambios, y cerrar la ventana los elimina.

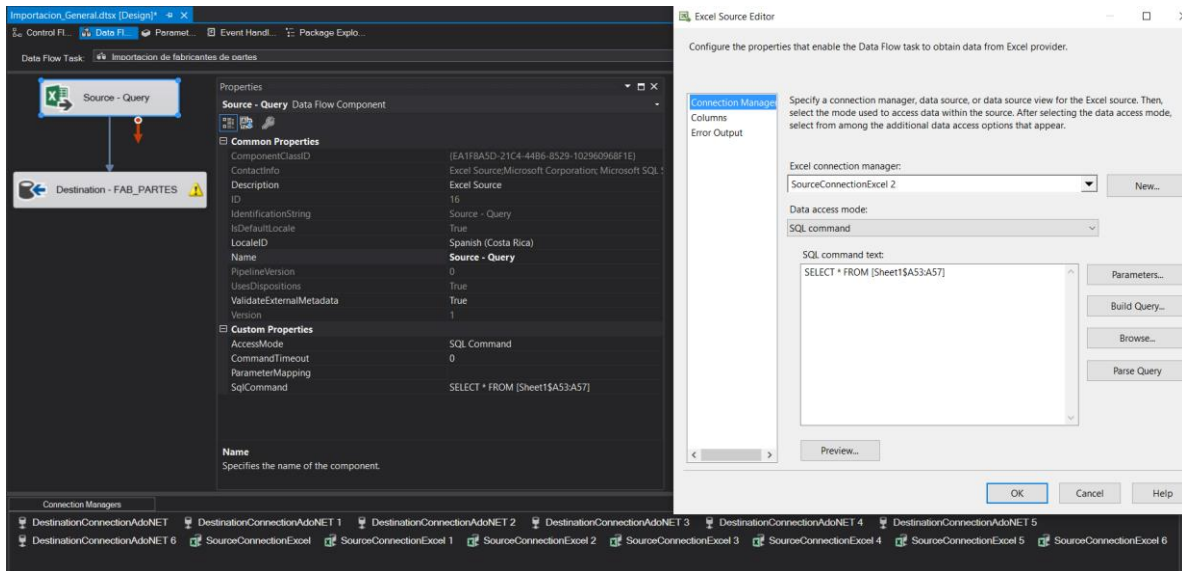
5. Importación de Datos Iniciales

Una vez que se insertó la base de datos diseñada en una instancia de SQL Server, se procedió a la importación de los datos iniciales. Dicha carga inicial se realizó por medio de un proyecto de *integration services* de Visual Studio, el cual se contiene dentro de una solución también de Visual Studio. Los proyectos de *integration services* suelen constituirse de paquetes que, entre muchas otras funciones, son capaces de crear un flujo de datos entre un archivo de Microsoft Excel y una base de datos de SQL Server. En el caso del presente proyecto, se creó un único paquete que establece un flujo de datos entre el archivo de Excel brindado y la base de datos diseñada. En la siguiente imagen se puede apreciar las tareas que ocurren al ejecutar dicho paquete:

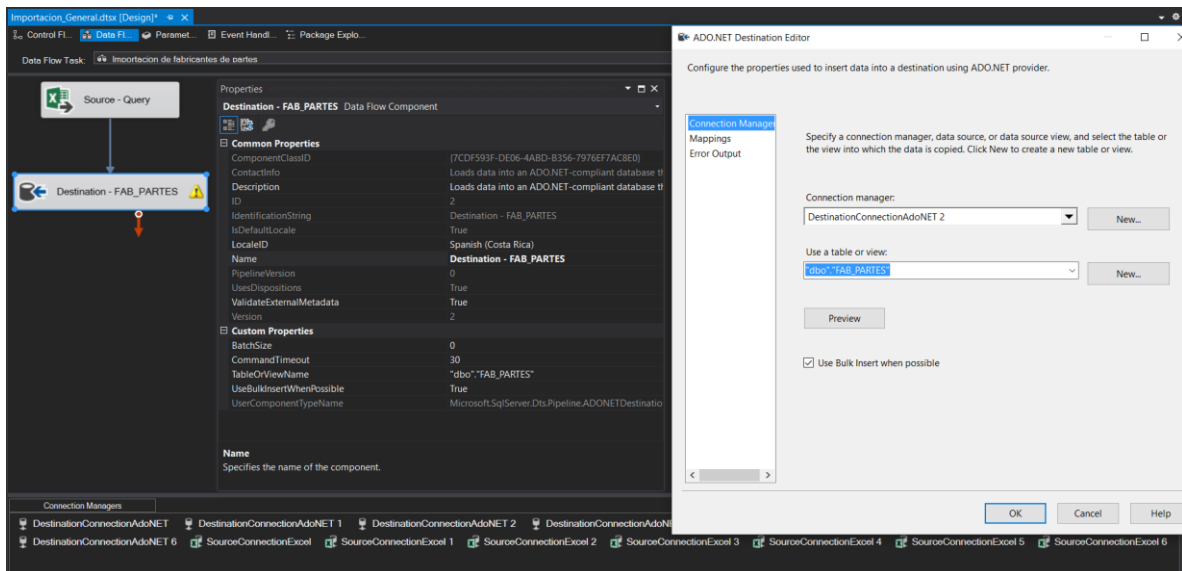


Como se puede observar en la imagen, dichos paquetes son también capaces de ejecutar consultas previamente almacenadas sobre la base de datos en cuestión. En el caso del paquete de importación construido, este crea y manipula dos tablas dentro de la base de datos destino. Asimismo, dichas tablas son eliminadas por el paquete una vez finalizado el movimiento de datos. En cuanto a las tareas de importación dentro del paquete, estas se diseñaron por medio del SQL Server Import and Export Wizard. Como ejemplo de lo anterior, a continuación, se muestran capturas del diseño de la tarea de importación de fabricantes de partes.

A) Propiedades y diseño de la fuente (Archivo de Excel)



B) Propiedades y diseño del destino (Tabla en la base de datos)



Cabe resaltar que este paquete de importación está diseñado para trabajar exclusivamente con el modelo de base de datos del proyecto, además de estar configurado para encontrar dicha base de datos en la instancia de SQL Server usada durante el desarrollo del mismo. Lo anterior ocasiona que el paquete no puede ejecutarse de inmediato en cualquier equipo, no obstante, basta con modificar los *connection managers* del paquete (6 para la base de datos y 6 para el archivo de Excel). Si los *connection managers* apuntan al archivo de Excel o a la instancia de SQL Server respectivamente y dicha instancia posee una base de datos concorde a lo diseñado, la importación se puede realizar ejecutando el único paquete dentro de la solución de Visual Studio adjuntada a esta documentación. Sin embargo, se debe tomar en cuenta que dicha solución solo se puede abrir desde una instalación de Visual

Studio que cumpla lo especificado en la sección de ambiente de desarrollo (2) del presente documento.

6. Resultados de los Casos de Prueba

CLIENTES

Insertar nuevos clientes personas	Éxito
Insertar nuevos clientes organizaciones	Éxito
Modificar un cliente viejo	Éxito
Suspender un cliente	Éxito
Listar clientes	Éxito

PARTES

Insertar nuevas partes	Éxito
Borrar una parte que no participa en una orden	Éxito
Borrar una parte que sí participa en una orden (intentar luego de insertar órdenes)	Éxito
Asociar partes con proveedores	Éxito
Asociar partes con tipos de automóviles	Éxito
Actualizar precios de una parte ofrecida por un proveedor	Éxito
Listar partes por tipos de automóviles	Éxito

ÓRDENES

Localizar proveedores de una parte	Éxito
Insertar una nueva orden y asociar a una orden la compra de una cantidad de partes	Éxito
Caso 1 debe funcionar	Éxito
Caso 2 debe rechazarse (cliente SUSPENDIDA)	Éxito
Cambiar estado a ACTIVO	Éxito
Caso 2 de nuevo, debe funcionar	Éxito

7. Comentarios Finales

El estado final del proyecto cumple satisfactoriamente los objetivos planteados al principio de su desarrollo. La base de datos diseñada se ajusta a la situación planteada y la aplicación desarrollada cumple todas las funcionalidades requeridas por dicho escenario. Además, el aspecto final de la aplicación se encuentra amigable con el usuario gracias al uso de tablas que permiten una visualización sencilla de la información almacenada además de los procesos que se ejecutan. En cuanto a las limitaciones del estado final del proyecto, estas se encuentran casi exclusivamente en la importación de datos iniciales realizada. Lo anterior se debe a que si bien dicha importación se realizó satisfactoriamente por medio de paquetes de proyecto de *integration services*, la naturaleza de dichos paquetes establece varios requisitos de software para su ejecución cada vez que se utilice un equipo distinto, tales como una instalación de Visual Studio y la configuración de los *connection managers* del paquete. La mayor dificultad encontrada durante la realización del proyecto fue la falta de experiencia, ya que fue necesario regresar varias veces al modelo y diseño de la base de datos para realizar cambios ante situaciones que no se habían previsto. Además, a la hora de desarrollar la aplicación el código resultó muy desordenado ya que no se utilizaron procedimientos almacenados, los cuales no se conocían en el momento.