



**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**

**FACULTAD DE INGENIERÍA**

**Procesamiento, descarga y  
estandarización automática de  
información de portales  
comerciales del área de retail**

**INFORME DE ACTIVIDADES PROFESIONALES**

Que para obtener el título de  
**Ingeniero en computación**

**P R E S E N T A**

Salvador Benítez López

**ASESOR DE INFORME**

M.I. Vicente Flores Olvera



Ciudad Universitaria, Cd. Mx., 2017



## Índice de figuras

Ilustración 1. Retail (Compañía).....	7
Ilustración 2. Movilidad (Compañía).....	8
Ilustración 3. Comercial (Compañía).....	8
Ilustración 4. Cloud (Compañía).....	9
Ilustración 5. Soluciones avanzadas (Compañía).....	9
Ilustración 6. Business Intelligence.....	11
Ilustración 7. Esquema del flujo de información de Retail.....	12
Ilustración 8. De Access a SQL Server.....	15
Ilustración 9. Diseño de etapas del proyecto.....	16
Ilustración 10. Ejemplo AutoIT (código).....	21
Ilustración 11. Información de componentes AutoIT.....	22
Ilustración 12. Programador de tareas de Windows.....	23
Ilustración 13. Microsoft SQL Server.....	24
Ilustración 14. DER de la aplicación.....	29
Ilustración 15. Tablas creadas en SQL Server 2008 R2.....	30
Ilustración 16. Proceso de estandarización de la información descargada.....	32
Ilustración 17. Proceso de la carga de información a la base de datos (ETL).....	33
Ilustración 18. SQL Server Integration Services.....	34
Ilustración 19. Interfaz gráfica de SSIS.....	35
Ilustración 20. Diseño de la herramienta ForEach Loop.....	36
Ilustración 21. Ventana de extracción de datos (Flat File Source).....	37
Ilustración 22. Código del movimientos del archivos mediante Script Task.....	38
Ilustración 23. Creación de Job en SQL Server.....	38
Ilustración 24. Diseño completo del DTS en SSIS.....	38
Ilustración 25. Creación de Job en SQL Server (2).....	39
Ilustración 26. Proceso de trabajo del proyecto.....	39
Ilustración 27. Metodología de trabajo RAD.....	41
Ilustración 28. Creación de reporte en Excel con conexión a SQL Server.....	45
Ilustración 29. Estadísticas generadas con la información.....	46
Ilustración 30. SQL Server Reporting Services.....	47
Ilustración 31. Generación de reportes multiplataforma SSRS.....	48

## Índice de tablas

Tabla 1 Software para la automatización de tareas.....	17
Tabla 2. Base de datos de Access .....	25
Tabla 3. Comparativa de trabajo en tiempo. ....	44

## Introducción

En la actualidad, la información es uno de los activos más importantes de toda empresa, ya que es la base para la toma de decisiones estratégicas que se realizan diariamente. Para poder tomar una decisión adecuada es necesario analizar de forma correcta dicha información ayudándose de herramientas actuales para este fin.

Estas herramientas día a día van actualizándose haciendo más sencillo, rápido y eficiente la forma que se procesan los datos, y la manera en la que estos se pueden visualizar. Por ello es necesario que las empresas deban ir de la mano con estas herramientas para poder agilizar procesos y mejorar los tiempos de respuesta de cada una de las aplicaciones que se necesitan.

En este documento se presenta una solución, a través de un proceso automatizado, para que una empresa “mayorista de tecnología” cuente con información confiable que le permita establecer estrategias para cumplir sus cuotas de mercado.

En el primer capítulo describo a la empresa donde estuve laborando, detallando las áreas que la conforman, así como las actividades que realicé en un área en específico.

En el segundo capítulo menciono la forma en que se trabajaba en el proceso que se me encargó debido a que se realizaban de forma manual.

En el tercer capítulo explico de forma concisa cómo llegué a la solución que actualmente está en funcionamiento; su desarrollo, donde denoto el proceso de descarga automática de datos de cada uno de los portales y cómo fue que elegí el software de cada uno de los módulos. Posteriormente menciono cómo se realizó la creación de reportes automáticos que deben enviarse a las personas correspondientes con la información requerida.

Por último muestro los resultados en el desarrollo del proyecto para verificar si se cumplieron las expectativas, así como los ajustes que se realizaron al generar los reportes para aprovechar la información y presentar mejor los datos, ya sea en

reportes geográficos o en la creación de portales web dentro de la intranet de la empresa para la agilización del conocimiento de los datos.

## CAPÍTULO 1. Descripción de la empresa

Antes de comenzar esta sección me referiré a la compañía como el lugar en el que me encuentro laborando actualmente.

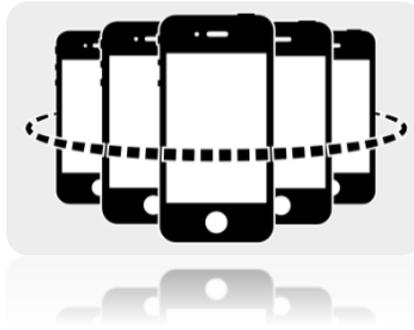
El principal giro de la compañía es ser un mayorista de tecnología, siendo uno de los más grandes que existen en México y que tiene presencia global. Esta se divide en más de 5 divisiones principales que ofrecen diferentes soluciones y productos que varían con el cliente que las solicite. A continuación haré una breve explicación de lo que realiza cada una de estas:

- Consumo
  - Es el área en el que me encuentro actualmente. Esta área tiene como finalidad ofrecer a los *retailers* productos que venderán a sus clientes. Estos productos son principalmente categorizados para entretenimiento y de uso cotidiano.



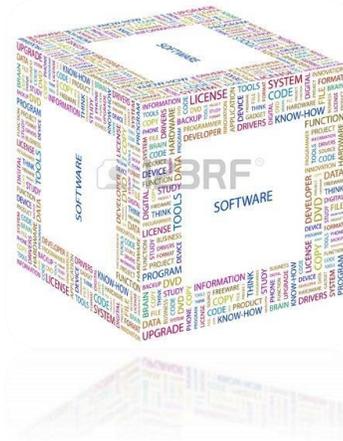
*Ilustración 1. Retail (Compañía)*

- Movilidad
  - Esta área tiene como finalidad vender cada uno de los artículos en telefonía celular a las empresas enfocadas en telecomunicaciones e instituciones gubernamentales.



*Ilustración 2. Movilidad (Compañía)*

- Comercial
  - Está área está dedicada a productos componentes para la creación de soluciones tecnológicas implementadas en otras empresas. Un ejemplo pueden ser servidores, licencias para equipos, impresoras entre otros.



*Ilustración 3. Comercial (Compañía)*

- Cloud
  - Está área ofrece servicios en la nube tal como su nombre lo describe. Ofrece cada una de las diferentes oportunidades para las PyMES como:
    - SaaS (Software as a Service)
    - IaaS (Infrastructure as a Service)
    - PaaS (Product as a Service)



*Ilustración 4. Cloud (Compañía)*

- Soluciones avanzadas
  - Esta área es muy reciente ya que debido a que las empresas ahora no desean invertir en la investigación de nuevas soluciones, la compañía las ofrece diseñándolas con base a lo que el cliente necesita.



*Ilustración 5. Soluciones avanzadas (Compañía)*

La compañía tiene muchas áreas de oportunidad y así mismo ha tenido un impacto muy grande en México por lo que tiene valores únicos denotados a continuación:

- Innovación: La compañía siempre está en un proceso de mejora continua ya que lo que hoy servirá no necesariamente lo hará el día de mañana. Siempre

hay que buscar formas más eficientes de realizar lo que hoy se hace. Parte del proceso que realicé está dentro de este rubro.

- Responsabilidad: Siempre hay que tomar responsabilidad en los actos que están y estarán por realizarse. Si existe algún error debe asumirse la responsabilidad y corregirlo cuanto antes. Este rubro también se refiere a tomar partido en las próximas actividades a realizarse, ya que con el apoyo de todos es posible realizar lo que se planea.
- Integridad: Todas las personas que laboren deberán ser integras, siendo entonces que todo lo que decimos ser sea lo que hagamos. Solo podremos salir adelante si nuestros actos concuerdan con lo que prometemos.
- Trabajo en equipo y respeto: En una compañía no existe un trabajo que no afecte únicamente a quien lo realiza, por lo que debemos realizar las actividades en equipo y aportando lo mejor de nosotros mismos para dar los mejores resultados, esto con respeto a nuestros colegas y nuestros clientes.
- Aprendizaje: Así como se mencionó que la compañía debe estar en un mejoramiento continuo, así lo debemos estar también nosotros. Así mismo debemos aprender de las experiencias diarias para poder tener bases de referencia en nuestras acciones.
- Responsabilidad Social: Una compañía debe ser recíproca con lo que ha podido lograr con la sociedad. Es por esto que siempre deben realizarse actividades que apoyen a nuestra comunidad, ya sea con apoyos de cualquier índole.

Ya que he descrito lo que realiza la compañía, procederé a explicar que funciones que he realizado y cual me encuentro realizando en este momento.

Mi puesto inicial era el de Business Analyst Jr reportando directamente al área de Business Intelligence. Ahí me encargaba principalmente al análisis y proceso de la información de ventas, compras e inventarios que se obtenían de los retailers (tiendas comerciales) para posteriormente generar los reportes necesarios tanto para la compañía como para cada uno de los proveedores. La información recabada era

mucha y el tiempo de proceso manual era demasiado, siendo estas dos razones muy fuertes para el inicio de un nuevo proceso que ayudase a la automatización de las tareas ya realizadas, el cual se detalla más adelante.

Después de un año y medio se me asignó a una nueva área con el puesto de Retail Business Analyst. En este puesto ya no se me encargaba la mejora de procesos en su totalidad sino la de analizar la información y la de asistir a las juntas en donde se me requería para el correcto pago de diferentes cuentas por pagar.

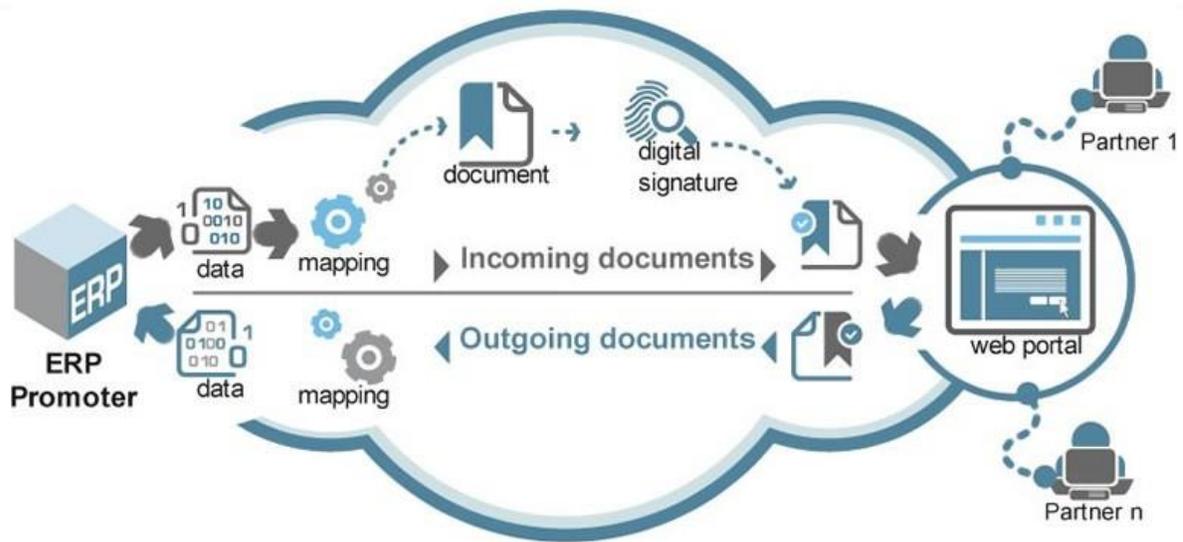


*Ilustración 6. Business Intelligence*

## CAPÍTULO 2. Antecedentes

En el tiempo en que he estado laborando dentro de la empresa, he estado realizando diferentes proyectos, pero a continuación denotaré qué actividades se realizaban antes de la culminación de este para poder tener una visión general del proceso y así mismo el porqué de la solución propuesta y realizada.

Todo se basa en la información que proveen los retailers a través de sus portales comerciales, puesto que muchos de los cálculos se realizan con base en las ventas que han tenido de nuestros productos y el inventario que tienen en su momento.



*Ilustración 7. Esquema del flujo de información de Retail.*

Existen en total más de 30 retailers a los cuales facturamos productos del área de consumo. Cerca de 15 retailers cuentan con un portal comercial al cual podemos acceder a su información; el resto nos brinda estos datos por medio de correo electrónico debido a que su información es muy poca o no ven viable la implementación de un portal.

Debido a que los retailers sin portal nos compartían su información por medio de correo electrónico semanalmente, un compañero descargaba la información de forma manual

que nos compartían por medio de un archivo adjunto. Después de que se tenían todos los datos, se comenzaban a procesar todos los archivos con macros que se realizaron en Excel para poder estandarizar la información y almacenarla en Access.

Access era una solución rápida para la problemática que se tenía en ese entonces, pero para el 2014 se tenían cientos de millones de registros en una tabla sin normalizar, por lo que paulatinamente la consulta de los datos se fue haciendo cada vez más lenta.

Por último los reportes en Excel únicamente se actualizaban y entregaban a quien se debiera por medio de correo electrónico.

Fue a partir del 2015 que se empezó a trabajar en una solución que se ajustara a las necesidades de los usuarios y proveedores.

## CAPÍTULO 3. Definición del problema

He descrito ya como se trabajaba anteriormente para la descarga y procesamiento de información de los portales comerciales y para aquellos que nos brindaban el correo.

Todo este proceso era funcional, sin embargo presentaba varias desventajas:

- Si se deseaba conocer la información de un día en particular no era posible, debido a que solo se conocía el consolidado semanal. Para el caso de los inventarios todo hacía referencia al día domingo, que es el fin de la semana fiscal.
- El tiempo para poder descargar la información y procesar los datos era de dos días completos, por lo que alguien debía realizar esa tarea repetitiva todas las semanas.
- Muchos de los reportes que se generan no son únicamente para uso interno, también hay algunos que son compartidos para nuestros proveedores por lo que ellos conocían la información hasta 3 días después de que finalizará la semana.
- El escalar la funcionalidad de esta forma de trabajo era complicada, ya que Access si bien es una herramienta diseñada para el almacenamiento de información como bases de datos, carece de muchas funcionalidades que implementan los manejadores de bases de datos (SQL Server, Oracle, MySQL etc.).
- No podíamos hacer uso de DTS ni de herramientas que permitían la rápida inserción de datos a la base de datos.



*Ilustración 8. De Access a SQL Server*

Fue con base a estos puntos que se empezó a diseñar la solución teniéndose que resolverse varios temas:

- El tiempo destinado a la descarga de información era muy alto. Se tenía que encontrar una forma en que esta se realizará de forma automática.
- Para los retailers que no contaban con un portal comercial debía al menos encontrarse una forma en disminuir el tiempo en que se pudiera procesar la información.
- Se tenía que ocupar un Manejador de bases de datos. Access ya no era una solución viable.

Teniendo claro lo objetivos que se tenían que cumplir al término del proyecto se comenzó a segmentar las etapas del trabajo, quedando de la siguiente forma:

1. Descarga de información automática
2. Elección del manejador de base de datos.
3. Creación de una base de datos y las tablas en donde se almacenarían los datos.
4. Extracción y estandarización de la información descargada de los portales y correos electrónicos.
5. Creación de un DTS para permitir la carga rápida de datos.



*Ilustración 9. Diseño de etapas del proyecto.*

A continuación se explicará a detalle que realice para cada uno de los puntos señalados:

### **Descarga de información automática.**

Cuando se me comentó que la información que necesitábamos se encontraba dentro de los portales comerciales mi primer pensamiento fue el intentar descargar un archivo por medio de soluciones rápidas, ya sea por medio de FTP o algún otro medio, sin embargo no estaba ni un poco cerca de la solución.

Los portales comerciales son completamente diferentes uno del otro, por lo que no podía realizar una solución para que funcionase con todos al mismo tiempo, debía encontrar una forma de acceder a la información de cada uno por separado. Otro punto importante es que el archivo que ofrece la información no se encuentra a simple vista de un botón para descargar, uno debe seleccionar diferentes opciones en la página para poder tener acceso a los datos.

Fueron varias las opciones que empecé a tomar en consideración para poder atacar el problema, sin embargo todas se venían abajo porque no era un simple archivo en la red que pudiese descargar, sino que el mismo sitio autogeneraba con base a las

opciones seleccionadas. Con esto en mente tenía entonces que encontrar algo que me ayude a simular a seleccionar en automático dichas opciones en el sitio para poder entonces descargar la información.

Encontré varias opciones las cuales describiré a continuación así mismo sus ventajas y desventajas que encontré:

*Tabla 1 Software para la automatización de tareas.*

 <p><b>AutoHotKey</b> Automation. Hotkeys. Scripting.</p>	<p><i>“AutoHotKey es un software libre que permite la creación de macros y de automatización de tareas repetitivas en Windows” (Sitio de AutoHotKey)</i></p> <p>Como lo menciona el mismo sitio, AutoHotKey es un software que permite realizar tareas repetitivas dentro de Windows.</p> <p><b>Ventajas</b></p> <ul style="list-style-type: none"><li>• Es un software libre, por lo que no habría que realizarse un gasto. Esto es un buen punto porque podríamos hacer pruebas desde un inicio sin ningún costo.</li><li>• Existe documentación amplia.</li><li>• Permite acceso por medio de librerías a información contenida en archivos desde TXT hasta XLSX, así como bases de datos.</li></ul> <p><b>Desventajas</b></p> <ul style="list-style-type: none"><li>• No tenía forma de acceder a controles creados por Windows a excepción de Windows Explorer. Es sumamente importante acceder a los controles de algún navegador al menos.</li></ul>
--	---



*“AutoIT es un software libre basado en el lenguaje para la creación de scripts en Basic. Está diseñado para automatizar procesos de las interfaces gráficas de Windows.” (Sitio de AutoIT)*

AutoIT es un software bastante integro que cumple con muchas de las funciones que necesitábamos.

### **Ventajas**

- Es un software libre, por lo que no existe un costo por usarlo y realizar pruebas con él.
- Existen muchos desarrolladores que crearon diversas librerías aparte de las que ya estaban implementadas en él.
- Hay librerías para poder tener acceso a todos los controles de Internet Explorer.
- Permite acceso a información por de ADO Objects.
- Su sintaxis es muy similar a VBA, por lo que programar sería sencillo debido a la familiaridad del lenguaje visto en la facultad.

### **Desventajas**

- Debido que los servidores no tienen permisos de administrador, AutoIT no puede acceder a todos los controles de Windows.
- Muchos de las librerías que vi que eran funcionales están creadas por desarrolladores externos. Es muy improbable que no exista un soporte posterior.

*“Foxtrot RPA software funciona para automatizar los procesos entre Windows, tu navegador y aplicaciones de negocio.” (Sitio de Foxtrot)*

Foxtrot es un software muy amigable que permite que el usuario no ingrese ninguna línea de código. En mi caso no es muy relevante eso sino que cumpla con las funciones que se requerían.

### **Ventajas**

- Es demasiado amigable, permitiendo que cualquier usuario pueda automatizar sus tareas.
- Tiene acceso de lectura y escritura de forma sencilla a cualquier tipo de archivo ya que tiene librerías que trabajan con la paquetería de Office.
- Tiene mucho soporte y documentación
- Tiene acceso a los controles de Internet Explorer

### **Desventajas**

- No es un software libre, por lo que supondría un gasto a la compañía.

**Foxtrot**



<p><b>FastKeys</b></p> 	<p>Es un software muy similar a los descritos anteriormente.</p> <p><b>Ventajas</b></p> <ul style="list-style-type: none"><li>• Ofrece acceso a movimientos de mouse y teclado de forma sencilla.</li><li>• Permite acceso para escribir directamente en cualquier control generado por Windows.</li></ul> <p><b>Desventajas</b></p> <ul style="list-style-type: none"><li>• No es un software libre, por lo que al igual que el caso anterior, supondría un gasto para la compañía.</li><li>• No tiene acceso a los controles de Internet Explorer.</li></ul>
--	--

Revise todas las posibles opciones para poder tomar la decisión de cual software ocupar en el desarrollo. Tome como puntos primordiales lo siguiente:

1. Deberá ser una herramienta que tuviera tener accesos a los controles de algún navegador, puesto que necesito acceder a uno para obtener la información.
2. Preferentemente deberá ser un software libre, ya que de lo contrario implicaría un costo a la compañía.
3. Las interfaces gráficas son buenas, pero no es lo que busco en este momento. A lo largo de la carrera tuve materias dedicadas a la programación que me ayudaron a aprender varios lenguajes de programación.

Dicho esto, opte por diseñar e implementar lo que se tenía pensado con **AutoIT**. El software al ser libre no presenta algún gasto por parte de la compañía, además de que permite el acceso a los controles de Windows junto con los de Internet Explorer, por lo que no debería presentar algún error.

Foxtrot es una herramienta muy poderosa y lo mencionó porque estuve a punto de elegirlo, lamentablemente la licencia cuesta alrededor de 500 dólares únicamente la versión personal. Para obtener un estimado de la versión empresarial se debe contactar al sitio.

Auto IT es un lenguaje que el manejo de la mayoría de los controladores de Windows para realizar tareas automáticas y está basado en el lenguaje Basic, por lo que no es muy complicado el aprenderlo.

```
Run("notepad.exe")
WinWaitActive("Untitled - Notepad")
Send("This is some text.")
WinClose("Untitled - Notepad")
WinWaitActive("Notepad", "Save")
;WinWaitActive("Notepad", "Do you war
Send("!n")
```

*Ilustración 10. Ejemplo AutoIT (código)*

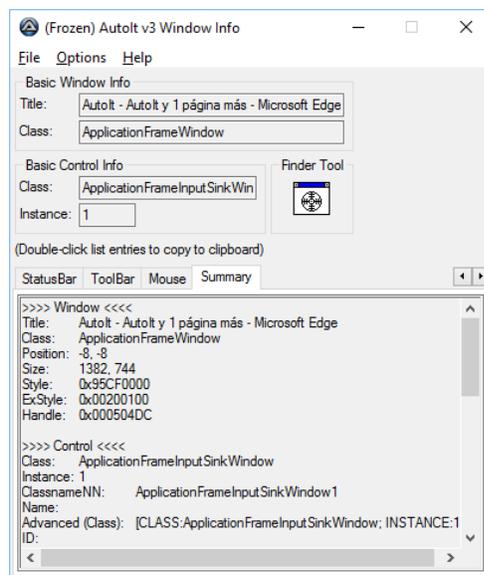
Empecé a desarrollar cada uno de los scripts con los que podría entrar a cada uno de los portales y obtener la información, que serían en total 15. Estos no podrían tener alguna relación entre sí, ya que cada sitio era completamente diferente.

Al ir desarrollando cada uno de los scripts me di cuenta que la contraseña iba cambiando con el paso del tiempo, ya que regularmente expira cada 3 meses, por lo que dejar este dato fijo en cada uno de los códigos fuente de los scripts no era una opción. Decidí ocupar la librería ADO para poder realizar una conexión una tabla en SQL SERVER para poder tomar de un catálogo el dato.

Teniendo este dato proseguí a imitar los pasos que se tienen que hacer por medio del script, pero encontrar los controles internos de Internet Explorer no es tan sencillo pero afortunadamente AutoIT ofrece una herramienta al momento de instalar que permite

por medio de una herramienta al estilo “drag & drop” obtener todos los datos de un punto en específico, tales como:

- Nombre del proceso
- Posición del puntero ( Absoluta por parte de Windows e interna del control)
- Numero de instancia
- ID



*Ilustración 11. Información de componentes AutoIT*

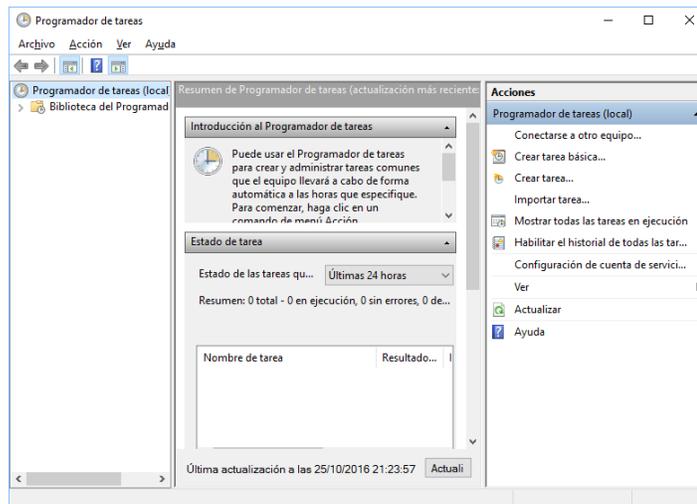
Gracias a esto pude obtener todos los controladores y manipularlos para que me dieran el resultado esperado.

El resultado por cada retailer fue la creación de un script en formato .au3, pero la AutoIT tiene un convertidor de archivos .au3 a.exe, por lo que se generó cada uno de estos.

El siguiente paso fue realizar un ejecutable que llamé a todos estos scripts para que se ejecutasen en un equipo remoto para que este no interrumpa los procesos

que se realizan diariamente. Este ejecutable se mandó a llamar todos los días a las 7 de la mañana para que en cuanto llegue pueda yo procesar esta información.

El último punto que me gustaría resaltar es que existen ocasiones en que los portales no traen la información correcta, por lo que no realicé un proceso para que esta información se insertará en la base de datos de forma automática porque debo revisar que los datos sean coherentes tanto en ventas e inventarios.



*Ilustración 12. Programador de tareas de Windows*

## **Elección del manejador de bases de datos**

El manejador de bases de datos es un rubro muy importante ya que es aquí en donde se almacenará la información que descarguemos de los portales, así como cualquier otra cosa que necesitemos realizar.

Cuando me toco decidir opté por empezar a observar como manejaban la información otras áreas, ya que así mi división tiene información esencial, así mismo la tienen otras divisiones como finanzas, legal entre otras.

Me di cuenta que todos ellos manejaban SQL Server, por lo que conectarme a otras bases de datos por medio de otro manejador sería muy complicado.

En ese punto me di cuenta que hay veces en las que se debe trabajar con los recursos que la compañía tiene por el momento; uno puede optar por proponer nuevas ideas e inclusive una nueva estructura para hacer más eficiente el proceso pero si la compañía no lo ve redituable, tendrás que adecuarte a lo que hay.

Yo no conocía mucho SQL Server como software, pero conocía las herramientas que ofrecía y las asimilaba con otros manejadores que si dominaba, tales como MySQL, PostgreSQL y Oracle ya que estuve trabajando con ellos en gran parte de la carrera.

Muchos de los cálculos que se pensaban realizar serían muy complicados realizarlos con una sola consulta, por lo que me percate que SQL Server tiene una interfaz gráfica muy amigable con el usuario para el desarrollo de estas y no solo para esto, sino para cualquier tipo de mantenimiento creación de Jobs o inclusive para monitorear el rendimiento del manejador.



*Ilustración 13. Microsoft SQL Server*

Daré a continuación algunos puntos que destacan a SQL Server como manejador:

- Tiene integrado SSIS, la cual permite realizar procesos ETL para la carga de información. Este fue un punto clave que explicaré con detalle más adelante.
- Tiene integrador SSRS, el cual permite por medio de Reporting Services la generación de reportes con parámetros dinámicos y los cuales pueden ser publicados en cualquier sitio interno para la consulta de información.

- Actualmente a la fecha de la elaboración el documento tiene acceso a Power BI, el cual es una herramienta de Microsoft que ayuda la generación de reportes empresariales, integración de herramientas de Business Intelligence
- La compañía tiene licencias para el desarrollo de software por medio de Visual Studio, por lo que la integración con este a SQL Server se hace de forma muy sencilla. También la paquetería de Office tiene una integración muy rápida y práctica con datos extraídos de alguna tabla de SQL Server.
- La seguridad es muy amplia debido a que la asignación de permisos se realiza con el usuario de Windows con el que se ingresa al equipo que se esté ocupando. No hay necesidad de crear usuarios extras ni contraseñas (aunque es posible). Así mismo la asignación de permisos es directa.

### Creación de una base de datos

Una vez que elegí el manejador de base de datos que ocuparía proseguí a diseñar la base de datos y que datos se almacenarían.

Tome como base la base de datos que se tenía de Access para darme una idea de que datos se almacenarían. Los datos que se almacenaban son los siguientes:

*Tabla 2. Base de datos de Access*

Nombre del campo	Descripción	Tipo de dato en SQL Server
<b>Retailer</b>	Este campo contenía el nombre del retailer al cual se hacía referencia en el registro	Varchar(50)
<b>Store Name</b>	Este campo contenía el nombre de la tienda. Hay clientes que no cuentan	Varchar(50)

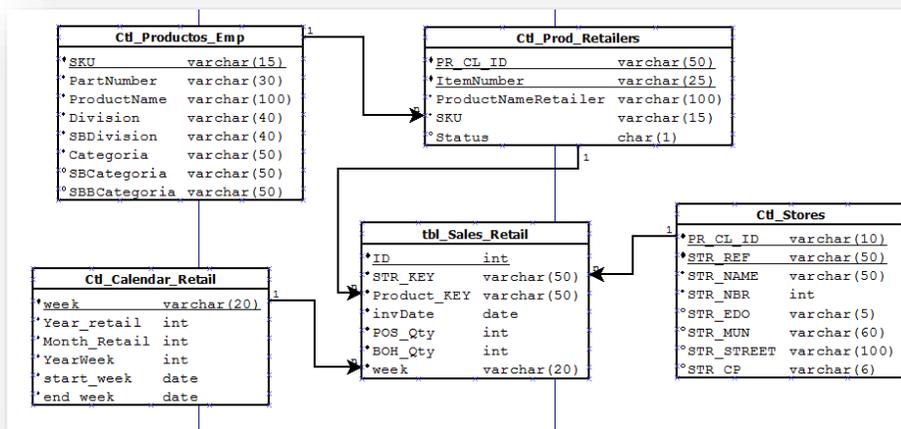
	con este campo por lo que podía ser nulo.	
<b>Store Number</b>	Este campo contenía el número de la tienda a la cual hacemos referencia en el registro. Este campo puede ser nulo ya que hay retailers que no nos muestran este campo.	Int
<b>Division</b>	Este campo sirve para categorizar el producto al cual hace referencia el producto.	Varchar(30)
<b>SBDivision</b>	Este campo sirve para categorizar el producto al cual hace referencia el producto.	Varchar(30)
<b>Category</b>	Este campo sirve para categorizar el producto al cual hace referencia el producto.	Varchar(30)
<b>SBCategory</b>	Este campo sirve para categorizar el producto al cual hace referencia el producto.	Varchar(30)
<b>SBBCategory</b>	Este campo sirve para categorizar el producto al cual hace referencia el producto.	Varchar(30)

<b>Item Number</b>	Con esta clave el retailer identifica su producto.	Varchar(25)
<b>SKU</b>	Con esta clave nosotros identificamos el producto para realizar un cruce entre su producto y el nuestro	Varchar(10)
<b>Part Number</b>	Con esta clave podemos hacer un cruce de nuestro producto con el que ofrece el proveedor de la marca.	Varchar(15)
<b>Sum of POS Qty</b>	Este valor es la cantidad de venta que se tuvo en este registro tomando en cuenta el día, el retailer, el artículo y la tienda principalmente.	Int
<b>Sum of On Hand Qty</b>	Este valor es la cantidad de inventario que el retailer tiene en este registro tomando en cuenta el día, el retailer, el artículo y la tienda principalmente.	Int
<b>InvDate</b>	Es la fecha a la que hace referencia el registro.	Date
<b>Week</b>	Es la semana fiscal en la que se encuentra la fecha del registro (El campo InvDate).	Varchar(30)

Muchos de estos datos no se obtienen directamente de lo que nos ofrece el portal comercial o que nos llega por correo electrónico, por lo que anteriormente se tenía un archivo en Excel con cada uno de los catálogos posibles para poder completar esta tabla. Esto a un nivel muy bajo era en sí una pequeña base de datos, por lo que al momento de crear la base de datos comencé a almacenar dichos catálogos. Estos se agrupaban en 4 principalmente:

- Catálogo de productos de los retailers
  - Aquí se almacenarán la relación entre los productos de los clientes con los nuestros. Muchas veces no ocupan la misma clave que la de nosotros, por lo que aquí encontraremos dicha relación.
- Catálogo de productos de la compañía.
  - Aquí se encontrará la categorización completa de los productos que tenemos. Esto servirá mucho para poder diferenciar los productos desde cualquier apartado mencionado en la tabla.
- Catálogo de tiendas.
  - Aquí se encontrarán los completos de las tiendas que tengamos disponibles para cada uno de los retailers. Entre estos se encuentran el nombre de la tienda, su código postal, su dirección, etc.
- Catálogo de semanas fiscales.
  - Este nos servirá para poder ingresar en la tabla el número de semana fiscal en la que se encuentra el registro consultado.

Teniendo la idea clara de cómo sería la base de datos que almacenaría la información, me di a la tarea de crear el DER, el cual me serviría para como base y documentación del trabajo que estaría realizando.

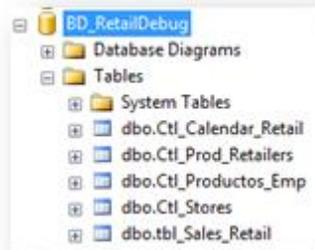


*Ilustración 14. DER de la aplicación*

Una vez hecho esto proseguí en crear la base de datos en SQL SERVER. El manejador ofrece varias herramientas bastantes amigables y sencillas para la creación de cualquier objeto, ya sea una tabla o un índice, inclusive el mantenimiento de la base es sencillo mediante este. La versión que se está ocupando hasta la fecha es la 2008 R2 debido a que es con la que se trabaja actualmente en la compañía.

Column Name	Data Type	Allow Nulls
SKU	varchar(15)	<input type="checkbox"/>
PartNumber	varchar(30)	<input checked="" type="checkbox"/>
ProductName	varchar(100)	<input checked="" type="checkbox"/>
Division	varchar(40)	<input checked="" type="checkbox"/>
SBDivision	varchar(40)	<input checked="" type="checkbox"/>
Categoria	varchar(40)	<input checked="" type="checkbox"/>
SBCategoria	varchar(40)	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

Column Name	Data Type	Allow Nulls
PR_CL_ID	varchar(50)	<input type="checkbox"/>
ItemNumber	varchar(25)	<input type="checkbox"/>
ProductNameRetailer	varchar(100)	<input type="checkbox"/>
SKU	varchar(15)	<input type="checkbox"/>
Status	char(1)	<input type="checkbox"/>
		<input type="checkbox"/>



*Ilustración 15. Tablas creadas en SQL Server 2008 R2*

Como podemos ver, esta es la estructura de la base de datos para el trabajo que estaremos realizando. Actualmente existen más tablas dentro de la base de datos que sirven como catálogos principalmente para la creación de reportes adicionales a los requeridos en ese entonces pero la tabla en donde se almacenan los datos es exactamente la misma.

En el área de consumo existen actualmente más de 1500 productos por lo que estos fueron ingresados en la tabla de **Ctl\_Productos\_Emp**. En el caso de los Retailers como lo expliqué hace un momento, cada uno de ellos lo categoriza de cierto modo, por lo que cada uno de los productos que factuemos deberemos registrarlos en la tabla de **Ctl\_Prod\_Retailers** como el Retailer lo identifica. Para el caso del calendario no habrá ningún cambio ya que este se encuentra ya categorizado por la empresa, solo deberemos poblar la tabla de **Ctl\_calendar\_retail**.

En el caso de las tiendas tenemos dos casos diferentes:

- Existen tiendas que son fácilmente identificables por el número de tiendas que se proveen en los reportes descargados en los portales de los Retailers así como su nombre, por lo que la relación entre este y la tabla en donde se almacenarán las ventas es realmente sencilla.
- Por otro lado tenemos Retailers que únicamente nos ofrecen el nombre de la tienda, haciendo que la regla que denotamos en el punto pasado no sea viable ya que no contamos con un número de tienda.

Para poder resolver este problema se cree un campo dentro de la tabla denominado **Str\_Ref** que servirá para poder almacenar la clave de Retailer y el número de tienda para aquellos que la tengan o en caso de que no lo tengan el nombre de la tienda.

### **Extracción y estandarización de la información descargada de los portales y correos electrónicos.**

Una vez diseñada e implementada la base de datos para el almacenamiento de la información descargada de los distintos portales y del correo electrónico me dispuse a crear una forma de estandarizar dichos datos para que se pudiesen ingresar directamente a la tabla de **tbl\_Sales\_Retail**.

Cada uno de los portales ofrecen en sus reportes distinta información que se complementa con las ventas e inventarios de los productos reportados; sin embargo como no todos ofrecen lo mismo es necesario discernir y extraer únicamente lo necesario, que para nosotros será lo siguiente:

- Nombre del producto
- Clave del producto como lo identifica el Retailer.
- Nombre de la tienda ( Si está disponible, en caso contrario se colocará como “No Disponible”)
- Número de la tienda ( Si está disponible, en caso contrario se colocará como 0)
- Venta del producto

- Inventario del producto
- Fecha de dicha transacción
- Clave del producto con el que lo conoce la empresa.

Cada portal ofrece esto en distintos formatos; habrá quienes permitan descargar un archivo en formato XLS o XLSX que permitan la fácil manipulación de los datos pero hay quienes entregan los reportes en TXT, XML o inclusive no ofrecen una forma de descargar la información. Para este último caso se trabajó para poder mediante AutoIT extraer los datos directos del sitio y almacenarlos en un archivo en formato XLSX.



*Ilustración 16. Proceso de estandarización de la información descargada.*

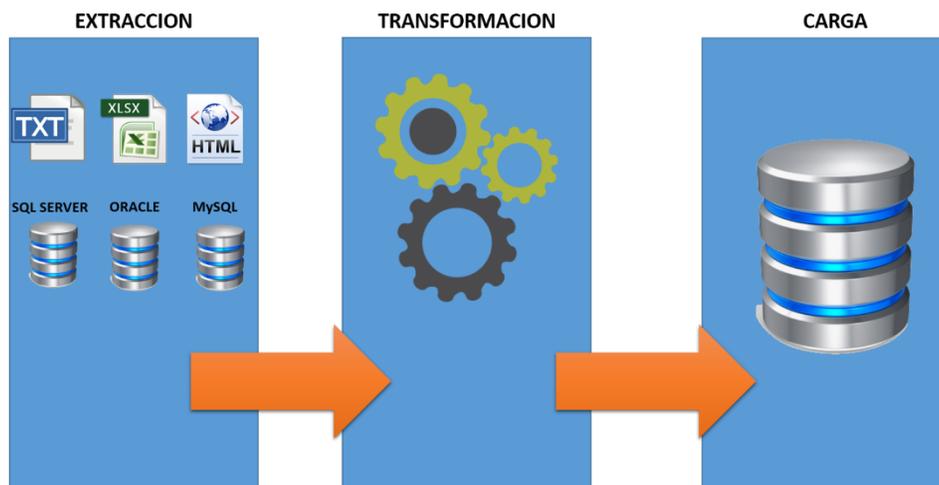
Como mucha información era procesada mediante Excel intenté buscar varias formas de trabajar la información y poderla procesar para poderla ingresar mediante un DTS a la base de datos. Hubo muchas herramientas que vinieron a mi cuando estaba realizando dicho proceso, sin embargo la más viable parecía ser VBA mediante la creación de macros en Excel, las razones fueron las siguientes:

- Ofrece soporte a las extensiones con las que se descargan los archivos de los portales, entre ellos: XLS, XLSX, XML, TXT, etc.
- Permite la rápida manipulación de los datos.
- El lenguaje es demasiado intuitivo, inclusive ofrece la opción para poder grabar las acciones que uno realiza para ejecutarlas una y otra vez.
- Muchas de las personas que trabajan dentro de la compañía conocen principalmente Excel, por lo que sería fácil para ellos el trabajar con la herramienta diseñada y el futuro mantenimiento. Además, si se requiriese una nueva actualización sería rápido implementarla.

Existen otras herramientas que podrían considerarse pero se necesitarían librerías o paquetes especiales que ayuden a la conexión con los formatos mencionados.

Hecho esto, comencé a programar las macros necesarias para cada uno de los Retailers.

Se programaron en ese entonces más de 30 macros. El resultado de cada una de ellas era el almacenamiento de la información necesaria en un archivo en formato TXT que sería tomado por el DTS para la inserción de datos a la base de datos. Este archivo tendría como nombre la clave del Retailer y la fecha de los registros contenidos en el formato yyyyymmdd.



*Ilustración 17. Proceso de la carga de información a la base de datos (ETL)*

El proceso que se está describiendo hasta el momento es comúnmente llamado también ETL (Extraction, Transformation, Load) el cual permite extraer la información de diferentes fuentes, transformarlas en los formatos requeridos y cargarla en la base de datos central.

Ahora solo nos falta un último paso, el cual es la creación del DTS que nos permitirá conectarnos a nuestra tabla e insertar los datos.

### **Creación de un DTS para permitir la carga rápida de datos.**

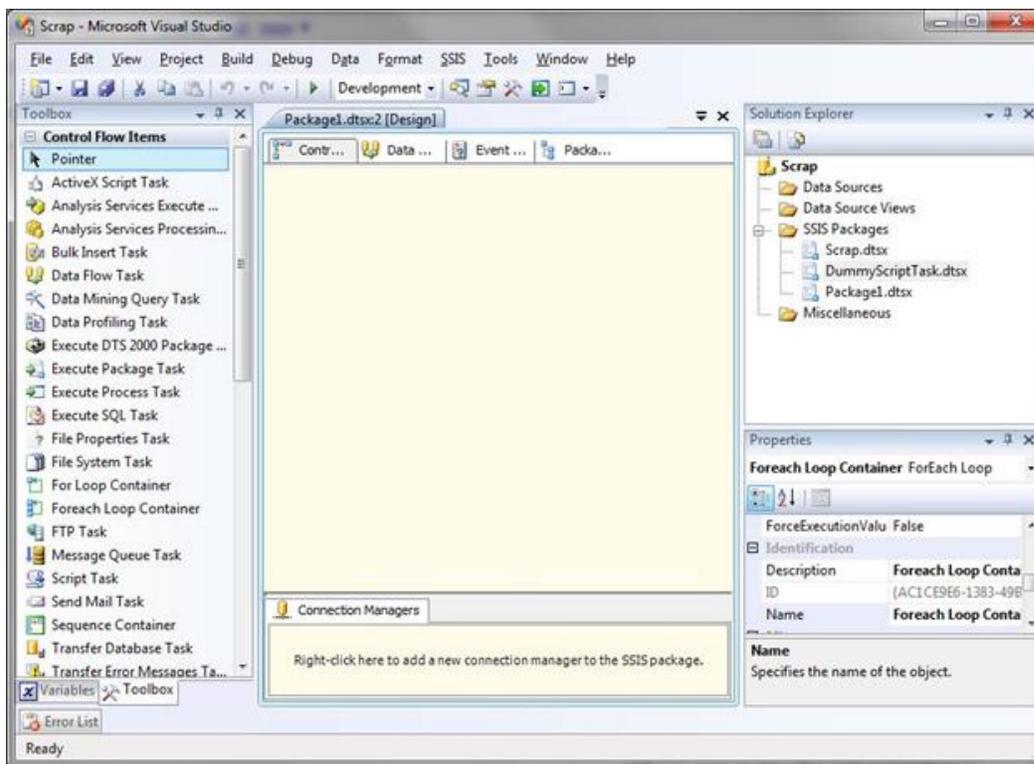
Los Servicios de Transformación de Datos, o DTS, es un conjunto de objetos y utilidades para permitir la automatización de extracción, transformación y carga a o desde una base de datos. La herramienta en SQL Server con la que se desarrolla se denomina SSIS, la cual significa SQL Server Integration Services.



*Ilustración 18. SQL Server Integration Services*

SQL Server también ofrece herramientas como SSAS (SQL Server Analysis Services) y SSRS (SQL Server Reporting Services) con las cuales podremos desarrollar herramientas de análisis de datos y de generación de reportes automáticos y dinámicos empresariales.

La creación de estos reportes es un tanto distinta a un lenguaje de programación ya que en realidad lo que haremos con la herramienta será indicar de donde tomaremos los datos, que haremos con ellos durante la transición de estos y cuál será su destino. Para esto tendremos la siguiente interfaz gráfica.



*Ilustración 19. Interfaz gráfica de SSIS.*

Del lado derecho encontraremos las herramientas que nos permitirán el seleccionar nuestros datos y la manipulación de estos. En el centro encontraremos nuestra área de trabajo en donde podremos arrastrar las herramientas seleccionadas e ir las conectando una con otra. A la derecha encontraremos todos los archivos de nuestro proyecto; encontraremos nuestro proyecto rápidamente con la extensión dtsx. Por último en la parte inferior se encontrarán todas las conexiones que establezcamos a cualquier fuente o destino de información.

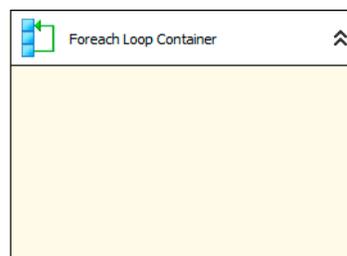
Una vez que en ese entonces conocí el cómo funcionaba la herramienta me dediqué a ver cuál sería la forma más rápida y eficaz de cargar la información.

Había varios factores a tomar en cuenta antes de empezar a diseñarlo:

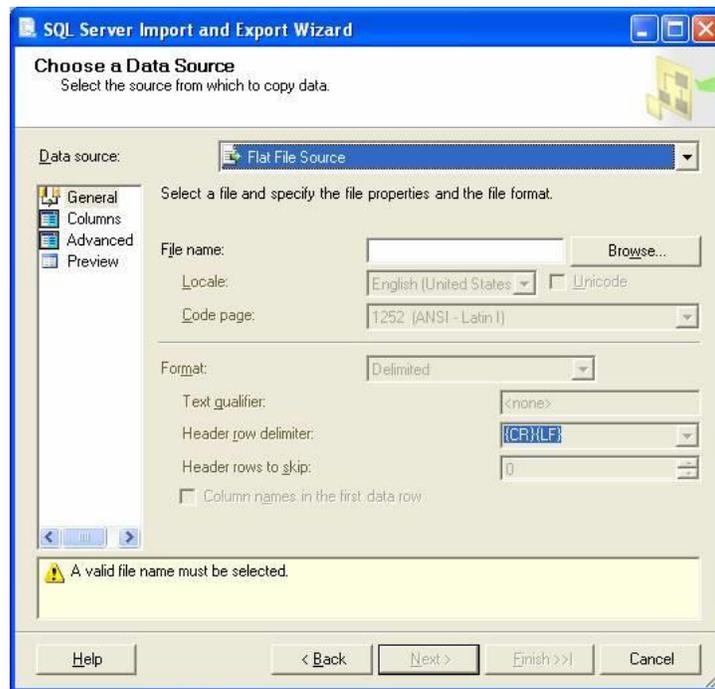
1. La información en los archivos TXT que se procesarían contenían la información tal cual como se diseñó la base de datos, por lo que no sería necesario la transformación de algún campo.
2. Si uno elige un archivo como fuente de información toma ese exclusivamente. Se necesitaría algo que tomará todos los archivos de dicha carpeta.
3. Es necesario guardar la evidencia de cada uno de los archivos cargados por si ocurriese algún problema.

Con la estandarización de la información el primer punto fue sencillo de aplicarse, debido a que las macros que manejarían los datos diarios generarían los archivos con los datos en el orden correcto para que no se tuviese que mover algo extra, sin embargo para los dos puntos restantes se tuvo que generar algo más.

Como mencioné anteriormente, SSIS nos permite la creación de DTS y de manipular los datos inclusive antes de la inserción o recolección de datos, pero también nos permite “iterar” o crear ciclos como si estuviésemos programando mediante herramientas **For Each Loop** agregadas de forma gráfica.



*Ilustración 20. Diseño de la herramienta ForEach Loop.*



*Ilustración 21. Ventana de extracción de datos (Flat File Source)*

Mediante estas dos herramientas lo que se realizó es ir iterando el archivo seleccionado mediante el ciclo For Each, haciendo que pudiesen insertarse todos los datos de la carpeta seleccionada.

Sin embargo aún faltaba un último punto. Necesitábamos guardar una evidencia del archivo insertado a la base de datos ya sea por motivos de respaldo de información así como de evidencia para alguna auditoría. Para poder solucionar esto se recurrió a la ayuda de otra herramienta llamada **Script Task** la cual nos permite realizar acciones con cualquier dato que este en transición. Para esto tomaremos el archivo que esté trabajándose, haciendo que una vez que terminé de insertar los datos se mueva a una carpeta designada.

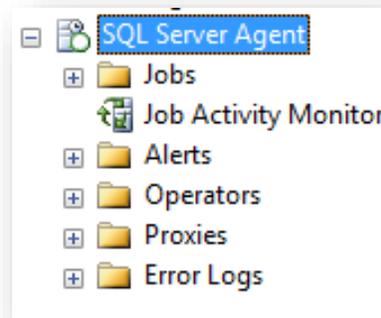
```

public void Main()
{
    // ALL: Add your code here
    File.Move(Dts.Variables["FilePath"].Value.ToString(), Dts.Variables["FilePath"].Value.ToString().Replace("
Dts.TaskResult = (int)ScriptResults.Success;
}

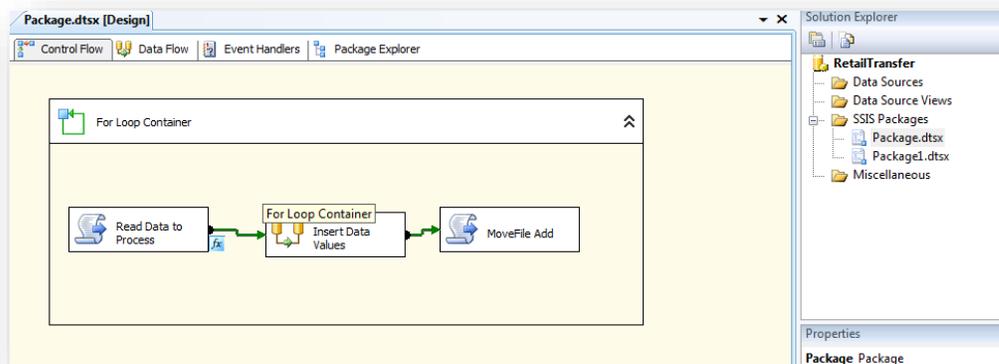
```

*Ilustración 22. Código del movimientos del archivos mediante Script Task.*

Con esto hemos se creó el paquete que contenía el DTS, sin embargo solo faltaba un punto en cuestión, el cual era crear un Job a partir de este. Para poderlo realizar solo debemos realizarlo a partir del menú **SQL Server Agent** que se encuentra a la izquierda de Sql Server.

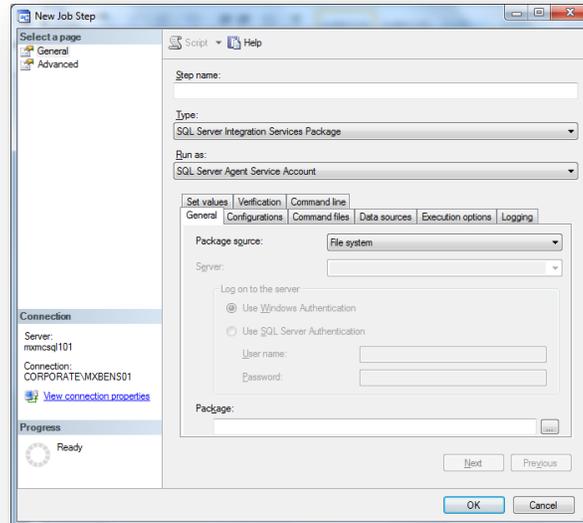


*Ilustración 23. Creación de Job en SQL Server*



*Ilustración 24. Diseño completo del DTS en SSIS.*

Deberemos darle un nuevo nombre al Job y seleccionar Steps, dando clic en el botón New Step. De ahí solo seleccionaremos la ruta de nuestro paquete.



*Ilustración 25. Creación de Job en SQL Server (2).*

Con esto se creará nuestro Job. Como comentario adicional, es posible que este se ejecute de forma periódicamente asignándolo como tarea de SQL Server. Es importante recalcar que el equipo debe permanecer encendido si desea realizarse esto.

### Detalles finales

Al haber realizado cada uno de las secciones necesarias para la elaboración de la solución, lo próximo que realicé fue conjuntar todo para que funcionase en bloques.



*Ilustración 26. Proceso de trabajo del proyecto.*

Para esto, se creó un proceso en el cual se abriría el archivo en Excel que contiene las macros que estandarizan la información que verificaría si existe algún archivo nuevo. Si esto llegase a ocurrir se ejecutaría la macro del Retailer en cuestión para poder generar el archivo.

Por último coloque que el Job que llamaría el DTS se ejecutaría cada 15 min, por lo que estaría en línea insertando la información requerida.

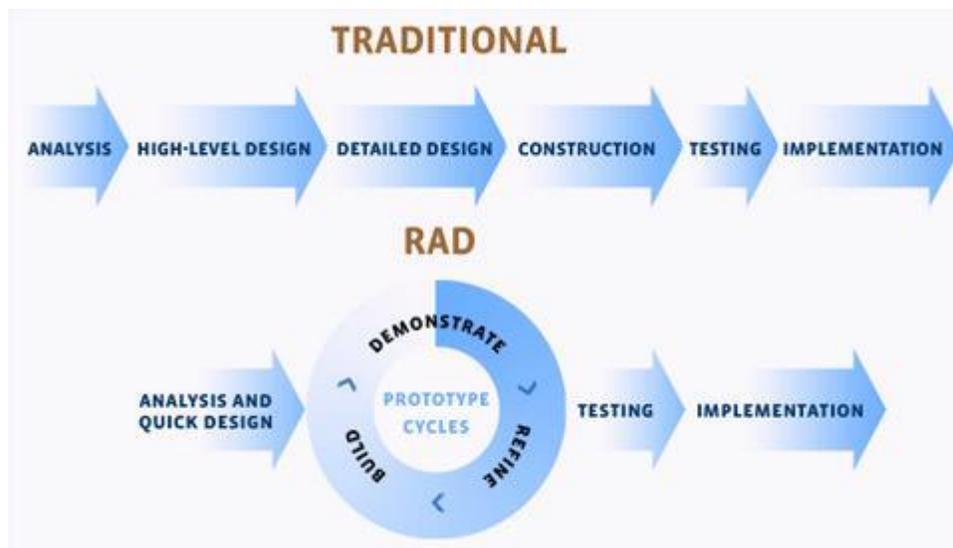
## CAPÍTULO 4. Metodología utilizada

La metodología de trabajo es tan importante como el proyecto que se está realizando, ya que sin esta es complicado llevar un orden durante su realización, así como en el futuro mantenimiento o actualización que se requiera.

En la actualidad existen diversas metodologías las cuales la facultad imparte en la asignatura de Ingeniería de Software en plan de la carrera de Ingeniería en Computación a la realización de este documento sin embargo describiré la que ocupe para el desarrollo del proyecto: RAD.

### **RAD (Rapid Application Development)**

RAD, siglas del desarrollo rápido de aplicaciones es una metodología de desarrollo de software, que implica el desarrollo iterativo y la construcción de prototipos. Ocupe esta debido a que la solución requerida era urgente y a la vez fiable debido a que es un proceso que maneja datos sensibles que son requeridos para otros procesos importantes en la compañía, tales como pagos de financiamientos y promociones, así como para la creación de reportes que son necesarios para la generación de estrategias de ventas y sugeridos para los Retailers.



*Ilustración 27. Metodología de trabajo RAD.*

A continuación describo la metodología utilizada:

1. Análisis y diseño rápido: El análisis de los requerimientos del proyecto fueron realizándose observando la situación ya que el único método que se tenía en aquel momento era cuando la tarea se realizaba de forma manual. Los requerimientos descritos anteriormente se dieron a partir de juntas con la persona que realizaba esto, puesto que se buscaba mejorar y que el tiempo designado a estos pasos se dedicará al análisis de información.

En el caso del diseño también se tomó como base los pasos realizados, ya que como tal no era una herramienta gráfica que utilizaría, sino una que se ejecutaría de forma automática.

2. Desarrollo y fase de pruebas: Durante la realización del proyecto se fueron realizando pruebas bajo una base de datos de pruebas mientras las actividades se realizaban aún de forma continua en la base de datos en Access que se tenía en ese entonces. Esto se hizo de esta forma debido a que la información es muy sensible. Las ejecuciones de los scripts, la estandarización y el proceso de inserción se revisó paso por paso para garantizar que los datos no se vieses afectados.

3. Implementación e interacción: Cuando todo funcionaba de forma fiable me dispuse a crear una copia de la base de datos que tenía en Access a su respectiva tabla de SQL Server para partir de ahí y seguir anexando la información para que el proceso fuese transparente. Se verificó esto durante dos semanas por si ocurría algo. Durante esas dos semanas se arreglaron detalles e implementaron mejoras en los scripts para que se eliminaran posibles fallas.

Como dato adicional, una vez terminada la implementación me dispuse a documentar todo el proyecto y así mismo archivar todas las minutas que se tuvieron para poder conocer la forma de trabajo. Teniendo esta información sería más sencillo el poder realizar algún cambio en el futuro ya que es posible que algún día se necesite y se ajuste la herramienta a las necesidades que se requieran en un futuro.

## Resultados

Después de observar, analizar, desarrollar, implementar y probar la herramienta que ayudaría a la solución de la problemática me dispuse a medir los resultados obtenidos contra la solución que se tenía anteriormente, ya que es de suma importancia el conocer este dato porque solo así podremos ver si en verdad lo que se realizó cumpliría los objetivos establecidos al inicio del documento así como las necesidades de la compañía.

Los resultados se medirían de la siguiente manera:

- Tiempo de descarga.
- Compatibilidad con las herramientas anteriormente creadas para la generación de reportes.
- Minimizar en medida de lo posible en que el usuario realice el proceso.
- Escalabilidad del sistema desarrollado

### **Tiempos de descarga**

Los tiempos de descarga dependen mucho de los Retailers ya que hay portales que nos brindan la información de forma sencilla, colocando solo algunos parámetros, sin embargo hay otros que requieren entrar a bastantes ligas hasta encontrar la información o inclusive en los peores casos hay otros que uno debe ir copiando y pegando dicha información en un Excel de forma manual debido a que no ofrece posibilidad de exportar los datos.

En el caso de los Retailers que solo requieren algunos parámetros el tiempo es similar que cuando se hacía manualmente pero la ventaja es que ahora ese tiempo puede destinarse a otras actividades. En donde podremos ver la diferencia son aquellos que mencioné al último, en donde uno debía realizar el copiado y pegado de los datos por

tienda y artículo que se mostraba dentro del sitio web, evitando la posibilidad de exportar la información; la diferencia de tiempo fue de 2 horas aproximadamente.

*Tabla 3. Comparativa de trabajo en tiempo.*

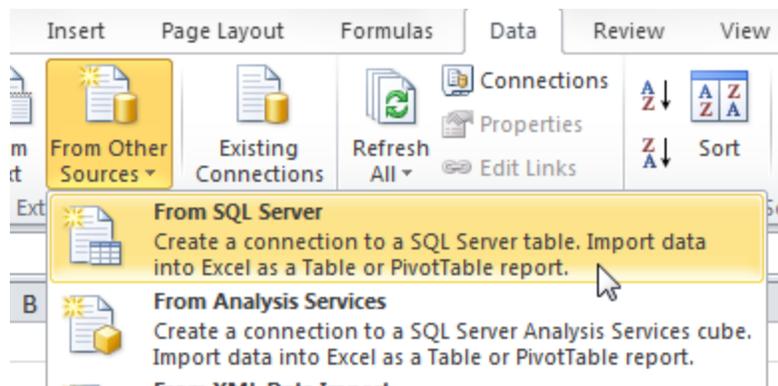
Tipo de Retailer	Tiempo de forma manual	Tiempo de forma automatizada
<b>Retailers con sencilla posibilidad de exportación de datos.</b>	190 segundos aproximadamente	180 segundos aproximadamente
<b>Retailers que requerían insertar valores dentro del sitio.</b>	260 segundos aproximadamente	220 segundos aproximadamente
<b>Retailers sin posibilidad de exportación de datos</b>	3 horas aproximadamente	1 hora aproximadamente

Como mencioné anteriormente, quizás en los primeros dos ejemplos no se ve tanto la diferencia, pero si tomamos a cuenta que son cerca de 26 cuentas las que se encuentran en estas dos categorías veremos que el tiempo total reducido es bastante. En el último caso fue en donde se vio la mejora en su totalidad, ya que las personas tenían bastante problemática con 4 cuentas; tal era el caso que aquella persona que le tocará ver esa información era el día que tendría que salir tarde de la oficina.

### **Compatibilidad con las herramientas anteriormente creadas para la generación de reportes.**

La solución ayudaba mucho a aquellas personas que se encargaban de ver la información, pero ahora teníamos que ver otro tema, los datos no eran únicamente conocidos por el personal del departamento en el que laboraba sino que también lo visualizaban al menos otras dos áreas.

Muchos de los reportes que se trabajan en la compañía son en Microsoft Excel, por lo que todos se mandaban a llamar con la fuente de datos en Access. Para poder dar solución a esto creé vistas en SQL Server que mostraban la información requerida en los reportes y haciéndolos disponibles dentro de la intranet de la empresa apoyándome del área de sistemas en cuestión de los permisos. Cada archivo lo que haría de aquí en adelante sería mandar a llamar los datos de dicha vista.



*Ilustración 28. Creación de reporte en Excel con conexión a SQL Server.*

En caso de que se requiriese algún cambio en el reporte lo único necesario de aquí en adelante sería modificar la vista, haciendo el proceso transparente para el personal.

Los resultados en este rubro se midieron con base a dos factores:

1. La velocidad en la que la información se actualizaba

Este tema no estaba considerado dentro de los rubros iniciales pero sin embargo es importante tomarlo debido a que con la ayuda de SQL Server ahora podíamos crear índices con base a las consultas que se realizaban con mayor frecuencia. El tiempo era diferente con cada consulta pero el tiempo en general se redujo aproximadamente en un 45%, haciendo que los datos fueran consultados de forma más rápida. Estos índices deberían rehacerse cada cierto tiempo debido a que la tabla principal de donde se obtenían los datos crece día con día.

2. La transparencia en los movimientos para que los reportes sirvieran de la misma forma que antes.

Este punto ya lo mencioné antes, puesto que los reportes gracias a la ayuda de las vistas en SQL Server se pudieron realizar de forma normal. Lo único que realicé fue cambiar la fuente de datos de los reportes principales.



*Ilustración 29. Estadísticas generadas con la información.*

### **Minimizar en medida de lo posible en que el usuario realice el proceso.**

La principal tarea de este proyecto era reducir el tiempo invertido por parte de las partes del personal que realizaba esta tarea. Puedo mencionar en este punto que el objetivo se cumplió debido a que ahora el tiempo invertido en esto ahora se ocupa para analizar la información, verificando que no exista una incongruencia ya que existen ocasiones en que el sitio consultado no ofrece la información correcta.

Existen a la fecha actualizaciones que requieren hacerse a la herramienta ya que ha ocurrido varias veces en las que el sitio de algún portal cambie, colocando los datos ya sea en otro sitio o en otro formato pero gracias a la documentación hecha es fácil realizar dicho cambio en la transición del día, haciendo que no se vea afectado las funciones del negocio en la compañía.

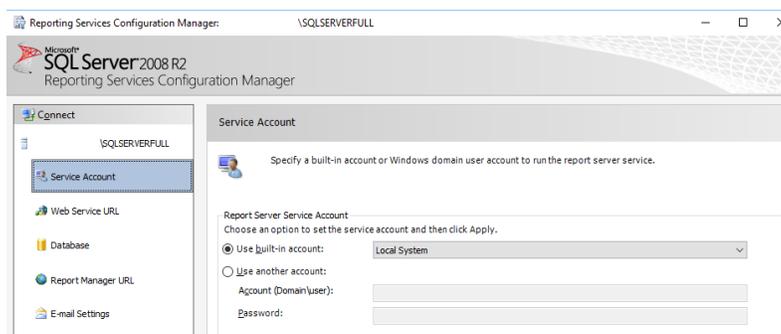
## Escalabilidad del sistema desarrollado

Un punto muy importante en la realización de un sistema son los siguientes pasos puesto que siempre debe obtenerse algún beneficio de este al realizarse no solo en su momento sino en las siguientes actualizaciones que se quieran hacer.

Este proyecto se realizó en el año 2016 y desde entonces han existido muchos módulos extras que sirven como apoyo para esto. Los denotó a continuación:

### 1. Envío de reportes automatizados

La compañía está muy ligada a las herramientas desarrolladas por Microsoft (Siendo SQL Server el Manejador de Bases de datos) por lo que se aprovechó para ocupar la herramienta SSRS (SQL Server Reporting Services). La herramienta



*Ilustración 30. SQL Server Reporting Services*

La ventaja de trabajar con esta herramienta es que pudo instalarse en el servidor en donde se alojaba el manejador y la base de datos. Esta herramienta ofrece varias opciones para la generación de reportes, ya sea la creación de estos de forma local o exportándose de forma web (inclusive móvil) en diferentes formatos.



*Ilustración 31. Generación de reportes multiplataforma SSRS*

Por último, uno puede crear las consultas que se necesitan para la creación del reporte con variables, permitiendo modificar algún parámetro de la consulta dentro de la palabra reservada WHERE, haciendo que el usuario pueda obtener la información deseada únicamente.

En la empresa esto lo ocupamos para aquellas personas que necesitaban la información un poco más simplificada que un archivo de Excel con los datos en una hoja. Una gráfica o tabla dinámica es lo que necesitan, por lo que SSRS hacía ese trabajo por ellos, exportándolo a un archivo PDF. En el caso de los proveedores se trabajó para que el servidor para que una vez que se creará dicho reporte se enviará a una lista de distribución de Outlook con los correos de los destinatarios que necesitaban los datos

## 2. Creación de portales web

Muchas de las personas que trabajan dentro de la compañía regularmente tienen tareas que son muy recurrentes, ya sea en el área en la que me encontraba o en otras. Para solucionar esto hace tiempo muchas personas del área de sistemas desarrollaron una herramienta que fuera sencilla para los usuarios poder ejecutar y que estuviera familiarizada con lo que ellos trabajaban diariamente: Excel. El uso de macros tiene muchas ventajas pero a su vez muchas desventajas que denoto a continuación:

- Ventajas
  - Es muy sencillo de trabajar
  - Microsoft Excel tiene un grabador de macros el cual te permite crear una sin la necesidad de conocer algo de programación.
  - Aun conociéndose el código y el programar es muy sencillo el realizar un cambio.
- Desventajas
  - El grabador de macros es muy limitado, muchas de las funciones de no se permiten ocupar durante este modo.
  - Deben tenerse los datos de la misma forma siempre para que esta funcione.
  - Si la macro llega a presentar un error de ejecución se detiene por completo el proceso.
  - Al ser muchos los usuarios que ocupaban las macros, si se requería un cambio era tardado realizar el cambio para todos.

Debido a todo esto se empezaron a desarrollar distintas aplicaciones web instaladas en la intranet que permitían a los usuarios acceder a una ruta designada en un servidor (IIS), la cual ejecutaba las tareas que ellos realizaban.

Algunas de las aplicaciones son las siguientes:

➤ Cálculo automático de promociones:

El cálculo de promociones es una actividad que ocurre entre la compañía con cada una de las tiendas comerciales permitiéndoles ofrecer precios más baratos de los productos que ya se vendieron a los usuarios finales. Se realiza tomando el descuento base de cada producto vendido por la cantidad final vendida del periodo de la promoción. Este monto es discutido entre el vendedor representante de la compañía con el comprador representante de la tienda comercial para poder expedir finalmente una nota de cargo.

La aplicación desarrollada evitaba al vendedor tener que realizar el cálculo por sí mismo, lo único que hacían era descargar un archivo de la aplicación web y revisarlo en una junta con el comprador.

➤ Cálculo automático de financiamientos mensuales.

Existen productos que son de difícil desplazamiento, ya sea por su costo base o porque son productos de introducción en el mercado. Para esto la compañía realiza un apoyo adicional por cada unidad vendida que se paga de forma mensual pactada con el comprador representante de la tienda departamental. Este apoyo es un porcentaje del monto de compra del producto de la compañía a la tienda el cual no supera al 7%.

Igual que en el caso anterior, la aplicación realiza este cálculo de forma automática, haciendo que el vendedor pueda tener la información a la mano para la junta de pago.

Esto que acabo de explicar son algunos de los usos que se les ha dado a la información descargada, aplicaciones que se han podido desarrollar debido al tiempo que se ha ahorrado con el proceso ETL automático, sin embargo se tienen otros planes para mejorar tanto este proceso como a las aplicaciones desarrolladas.

A continuación explico estos:

- ❖ Mejora de herramienta desarrollada

La herramienta desarrollada tiene un punto débil: siempre depende de la disponibilidad del portal consultado. Además de esto, hay muchas tiendas comerciales que aún no implementan un sitio web en donde pueda consultarse esta información teniendo que recurrir a recibirla por correo electrónico.

Para poder solucionar esto se requiere el apoyo tanto de la tienda comercial como de la compañía, ya que sin la información de esta no puede trabajar el proceso.

Se tiene pensado el crear un portal en el cual puedan acceder de forma sencilla cada uno de los retailers y que mediante almacenamiento en la nube puedan guardar el archivo de las ventas, compras e inventarios en el formato establecido para que puedan desde aquí tomarse los datos para el proceso.

Lo mismo ocurre con aquellos que no cuentan con un portal.

Esta solución unificaría el proceso para cada uno de los Retailers evitando el proceso de transformación de datos y de posiblemente la carga directa a la base de datos mediante SSIS.

- ❖ Mejora de reportes geográficos y temporales

Hoy en día muchas de las ventas registradas no tienen asignada la tienda en la cual fue realizada la transacción debido a que esta no está catalogada (refiriéndome a los datos geográficos de la tienda). Esto puede mejorarse

obteniendo dicha información por parte de las tiendas comerciales o haciendo un estudio de localización de tiendas. Ya sea por uno u otro medio puede registrarse cuales son las localidades en donde se desplaza más producto haciendo que cada uno de los centros de distribución tengan la mercancía que se requiere únicamente para evitar el Aging o la falta de producto. Además de realizar esto pueden mapas geográficos en donde se vea más detallado esto con la ayuda de Power BI, herramienta que se está empezando a ocupar dentro de la compañía.

Otro punto importante además es cuando se realiza la venta que, si bien ya se tiene este dato almacenado en la base de datos hay que entender que existen semanas en el año que las ventas regularmente serán altas y aquellas en las que serán bajas. Tener un análisis de esto también puede ser muy benéfico ya que no es posible comparar 2 semanas un año contra otro si ocurrió un lanzamiento de un producto importante en alguno de los dos o si ocurre un problema económico, solo por citar algunos ejemplos.

## Conclusiones

La realización de este proyecto me ayudo bastante a entender por qué está diseñado el plan de estudios de la facultad de ingeniería cuando la curso de esa manera. Gracias a varias asignaturas (tales como Ingeniería de Software, Administración de proyectos, bases de datos, lenguajes de programación entre otras) fue factible el que desarrollará el proyecto ya que en el mundo laboral no existen las actividades claramente solicitadas pues siempre están en un proceso de cambio.

Aprendí además de primera mano que los proyectos realizados deben ser acorde a lo que el cliente solicita, puesto que las aplicaciones tienen siempre como fin facilitar o evitar algunas tareas; de nada servirá la aplicación si no cumple con la expectativa esperada.

En el ambiente laboral he aprendido que es algo complicado cambiar algunas formas de trabajo en algunas empresas, pero siempre se debe buscar la forma de ayudar en los procesos puesto que existieron veces en donde tuve que sentarme y ver paso a paso cada uno de los pasos para poder llegar a una solución en común.

Entendí que es completamente normal no comprender algunos conceptos cuando entras a una empresa que tiene un giro completamente diferente al de tu carrera pero aprendí a complementar mis conocimientos y mi experiencia con los objetivos de la compañía, haciendo que a la fecha en la que escribo este documento tenga conocimiento de otra área completamente distinta al de la carrera que estudié en un inicio en la facultad de ingeniería (Economía).

Por último puedo remarcar por experiencia que un ingeniero siempre debe estar informado de las nuevas tecnologías ya que una vez que nosotros brindemos una solución es muy probable que con el paso del tiempo exista siempre una forma mejor y más sencilla de realizar lo que hicimos.

## Glosario

**Servidor.** Es un equipo de cómputo que está al servicio de otros, a los cuales se les denomina clientes. Este proporciona información requerida a cada uno de los clientes; estos pueden ser computadoras, celulares, impresoras, entre otros.

**Red de computadoras.** También se le conoce como red únicamente. La red se puede detallar como un conjunto de dispositivos (equipos de cómputo, celulares, impresoras, etc.) conectados por medio de cableado, señales, ondas o cualquier otro método de transporte de datos los cuales sirven para transmitir y compartir información, recursos e inclusive servicios.

**Sistema / Aplicación.** Es aquel software que tiene como propósito facilitar la realización de una o varias tareas específicas. Estas son diseñadas regularmente para el usuario final.

**Sistema Web.** La finalidad es la misma que la de un sistema cotidiano, lo único que varía aquí es la manera en la que se accede a este. Se efectúa regularmente por un navegador web, el cual realiza una petición a un servidor para desplegar este sistema sobre el navegador.

**Página web.** Es un documento electrónico el cual contiene información textual, visual y/o sonora que se encuentra alojada en un servidor y que puede ser accedida mediante un navegador web.

**Requerimiento.** Es la descripción del comportamiento que debe tener el sistema que se vaya a realizar. En este se incluyen casos de uso en donde se describen todas las interacciones que tendrá el usuario con este sistema. Está enfocada tanto para el grupo de desarrolladores como para el cliente.

**Base de datos.** Es una colección de datos organizados y estructurados según un determinado modelo de información que refleja no solo los datos en sí mismos, sino

también las relaciones que existen entre ellos. Cada una de estas se diseñó con un propósito en específico.

**Servidor dedicado.** Como tal la acción que realiza es la misma que la de uno normal, solo que a este solo se le asigna solo una tarea, la cual quizás requiera más recursos del equipo.

**Alcance (Software).** Es la suma total de los requisitos y características que tendrá el sistema. Se utiliza para saber cuáles serán las tareas que ejecutará el software y hasta donde deberá trabajarse.

**Versión (Software).** Es el proceso de asignación de un nombre o número único que sirve para indicar el nivel de desarrollo que se tienen sobre este.

**Petición (Software).** Es la acción que realiza un equipo de cómputo a otro para solicitar información de cualquier tipo.

**Mantenimiento (Software).** Es la modificación de un producto de software después de la entrega para corregir errores, mejorar el rendimiento u otros detalles que hayan surgido tras la instalación del mismo. Es una de las actividades más habituales de la ingeniería del software. También se les conoce como actualizaciones.

**Sistema operativo.** Es un conjunto de programas relacionados entre sí, que permiten administrar los recursos de la computadora para obtener un mejor rendimiento y aprovechamiento de los mismos. Además de proporcionar herramientas que permitan asegurar y proteger la información.

**Lenguaje de programación.** Se diseñaron estos lenguajes para desarrollar software. Existen diferentes niveles que varían según en donde se trabajó; estos son de bajo, medio y alto nivel.

**Manejador de base de datos.** Es un conjunto de programas que permiten el almacenamiento, modificación y extracción de información en una base de datos. Además permite añadir, borrar, editar y analizar los datos que contiene.

**Cuentas de usuario.** En su estricta definición es la indicación al equipo de cómputo que una persona lo está ocupando. Se utiliza para poder proteger información del usuario.

**.NET.** Es un framework de Microsoft que hace un énfasis en la transferencia de redes, con independencia de la plataforma de hardware y que permita un rápido desarrollo de aplicaciones.

**DBMS.** Sus siglas significan DataBase Management System. Es el software que se encarga de realizar diferentes tareas con la base de datos. Existen diversos manejadores, como MySQL, SQL Server, Oracle entre otros.

**Entidad (Base de datos).** Objeto que es distinguible de otros objetos por medio de un conjunto específico de atributos.

**Atributo (Base de datos).** Es aquel que representa una propiedad o característica de una entidad.

**Visual Basic .NET.** Es un lenguaje de programación orientado a objetos que puede considerarse como la evolución de Visual Basic implementada sobre su framework .NET.

**SQL.** Structured Query Language es un lenguaje declarativo que permite realizar diferentes tipos de operaciones sobre esta.

**DER.** Diagrama entidad-relación. Es una herramienta para el modelado de datos que permite representar las entidades relevantes de un sistema de información así como sus interrelaciones y sus propiedades.

**Relación (Bases de datos).** Es la asociación entre 2 o más entidades que puede ser clasificadas en 3 tipos:

- Relación uno a uno.
- Relación uno a muchos.
- Relación muchos a muchos.

**Tabla (Bases de datos).** Es en donde se guardarán la información referente a una entidad. Su estructura en una base de datos relacional se compone de campos y registros, en donde los campos son los atributos o características de la tabla y los registros cada uno de los objetos que se almacenan que cumplen con todos los campos.

**Normalización (Bases de datos).** Es el proceso en el que se aplican una serie de reglas a las relaciones obtenidas para evitar la redundancia de los datos, disminuir problemas de actualización de datos en las tablas e inclusive proteger la integridad de los datos.

**Forma normal (Bases de datos).** Proporcionan los criterios para determinar el grado de vulnerabilidad de una tabla a inconsistencias y anomalías lógicas. Mientras más alta, menos sufrirá estos percances. Se iniciaron con 3 formas normales, pero en la actualidad existen 6 incluyendo además la Boyce Codd así como la forma normal de dominio clave.

**Llave primaria (Bases de datos).** Es un conjunto de uno o más atributos de una tabla que, tomados colectivamente nos permiten identificar a un registro como único.

**Llave foránea (Bases de datos).** Es uno o más campos de una tabla que hacen referencia la llave primaria de otra tabla. Una de estas indica que existe una relación con otra tabla.

**Diccionario de datos.** Es un conjunto de metadatos que contiene las características lógicas y puntuales de los datos que se van a utilizar. En este se encuentra la lista de todos los elementos que forman parte del flujo de datos de todo el sistema.

**Integridad de datos.** Se menciona este término cuando la información que se encuentra dentro de la base de datos está completa y los tipos de datos son coherentes con respecto a lo que hay.

## Bibliografía

- **Peter Rob.** (2010). *Database Systems. Design, Implementation, and Management*. EU: CENGAGE Learning. Ninth Edition.
  - Capítulo 3 y 4. Páginas: 30-121
- **Clare Chucher.** (2012). *Beginning Database Design: From Novice to Profesional*. EU: Apress. Second Edition.
  - Capítulo 3. Páginas: 25-41
- **Linda V. Gillman.** (2008). *Guía de los fundamentos para la dirección de proyecto*. EE.UU: PMI.
  - Capítulo 1 y 2.
- **Que es un servidor**  
Fecha de consulta: 3 de Mayo del 2017
  - [http://www.aprenderaprogramar.com/index.php?option=com\\_attachments&task=download&id=487](http://www.aprenderaprogramar.com/index.php?option=com_attachments&task=download&id=487)
- **Software de aplicación**  
Fecha de consulta: 3 de Mayo del 2017
  - [http://proyectoova.webcindario.com/software\\_de\\_aplicacin.html](http://proyectoova.webcindario.com/software_de_aplicacin.html)
- **Especificación de requisito de software**  
Fecha de consulta: 3 de Mayo del 2017
  - [http://es.wikipedia.org/wiki/Especificaci%C3%B3n\\_de\\_requisitos\\_de\\_software](http://es.wikipedia.org/wiki/Especificaci%C3%B3n_de_requisitos_de_software)

- **Bases de datos**

Fecha de consulta: 3 de Mayo del 2017

- [http://www.hipertexto.info/documentos/b\\_datos.htm](http://www.hipertexto.info/documentos/b_datos.htm)

- **SSRS**

Fecha de consulta: 3 de Mayo del 2017

- <https://msdn.microsoft.com/es-MX/library/ms159106.aspx>

- **SSIS**

Fecha de consulta: 3 de Mayo del 2017

- <https://docs.microsoft.com/en-us/sql/integration-services/sql-server-integration-services>

- **Retailer**

Fecha de consulta: 3 de Mayo del 2017

- <https://www.thebalance.com/what-is-a-retailer-2890386>

- **WOS**

Fecha de consulta: 3 de Mayo del 2017

- [http://www.parkeravery.com/pov\\_InventoryPlanningMethods.html](http://www.parkeravery.com/pov_InventoryPlanningMethods.html)

- **DTS (Migración de datos)**

Fecha de consulta: 3 de Mayo del 2017

- [https://msdn.microsoft.com/es-es/library/cc707786\(v=sql.105\).aspx](https://msdn.microsoft.com/es-es/library/cc707786(v=sql.105).aspx)