

Dirección de Comercio Exterior y  
Administración de Tratados  
Comerciales  
Ministerio de Industria y Comercio

# SISTEMA DE ADMINISTRACIÓN DE CONTINGENTES ARANCELARIOS DE LA REPÚBLICA DOMINICANA



## MANUAL TÉCNICO

Mayo 2010

Este informe fue escrito por **Iván Javier Mendoza Neira** y reproducido por Chemonics International Inc. bajo el Proyecto para la Implementación del DR-CAFTA de la República Dominicana, Contrato Núm. IQC #EEM-I-00-07-00008-00.

# **MANUAL TÉCNICO DEL SISTEMA DE ADMINISTRACIÓN DE CONTINGENTES ARANCELARIOS DE LA REPÚBLICA DOMINICANA**

## **RENUNCIA**

Las perspectivas del autor expresadas en esta publicación no reflejan necesariamente las opiniones ni de la Agencia de los Estados Unidos para el Desarrollo Internacional ni del Gobierno de los Estados Unidos.

## TABLA DE CONTENIDOS

---

SIGLAS	ii
SECCIÓN I INTRODUCCIÓN	I-1
SECCIÓN II MANUAL TÉCNICO	II-1
A. Estrategia de Desarrollo	II-2
B. Capa de Datos	II-2
C. Capa de Reglas de Negocio	II-2
D. Capa de Presentación	II-2
E. Diagrama de Diálogo de Capas	II-4
F. Detalle de las Reglas de Negocio	II-5
G. Plataforma de Desarrollo	II-7

## **SIGLAS**

---

CAC	Certificados de Asignación de Contingentes
CD	Disco compacto
DICOEX	Dirección de Comercio Exterior y Administración de Tratados Comerciales
DGA	Dirección General de Aduanas
DR-CAFTA	Acuerdo de Libre Comercio en RD – Estados Unidos y Centroamérica
MA	Ministerio de Agricultura
MIC	Ministerio de Industria y Comercio
OTCA	Oficina de Tratados Comerciales Agrícolas
USAID	Agencia de los Estados Unidos para el Desarrollo Internacional

**SECCIÓN I**  
**INTRODUCCIÓN**

---

## **SECCIÓN I**

### **INTRODUCCIÓN**

---

El Manual Técnico pretende informar a nivel técnico la manera en que está construida la aplicación, y la estrategia o ingeniería de software que se utilizó. Adicionalmente, pretende detallar ciertos aspectos como las Funciones o Procedimientos utilizados, la estructura de datos o tablas, y el nombre de la base de datos.

Igualmente se detallará la herramienta de desarrollo, así como su versión, la base de datos, y la tecnología de conectividad de la base de datos.

**SECCIÓN II**  
**MANUAL TÉCNICO**

---

## SECCIÓN II

### MANUAL TÉCNICO

---

#### A. Estrategia de Desarrollo

Se utilizó como estrategia de desarrollo n-capas con el objetivo de separar la aplicación en componentes y hacer más fácil su mantenimiento y escalabilidad.

Las capas que se utilizaron fueron:

1. Datos
2. Reglas de Negocio
3. Presentación

#### B. Capa de Datos

La capa de datos es donde creamos nuestras clases con sus atributos y sus métodos para el acceso a los datos. La misma se modeló creando DataSets, en los que podemos hacer nuestras clases, atributos y métodos que nos permitirán almacenar, recuperar, eliminar y actualizar los datos.

Se utilizaron dos DataSets con las declaraciones de las clases y sus métodos, para posteriormente ser usados por la capa de presentación. Estos fueron nombrados:

1. SIACDataSet
2. SIACDataSetPrd

Modelar la capa de datos con el DataSet tiene como ventaja que muchas utilidades que deberíamos programar como desarrolladores ya vienen incluidas en él, lo que nos permite solamente dedicarnos a crear nuestros métodos, ahorrándonos parte del trabajo. Ejemplo de estas utilidades podemos mencionar los DataAdapter que se encargan de llenar las tablas con los datos necesarios, y BindingSource que nos hacen el enlace entre nuestros objetos y los datos.

#### C. Capa de Reglas de Negocio

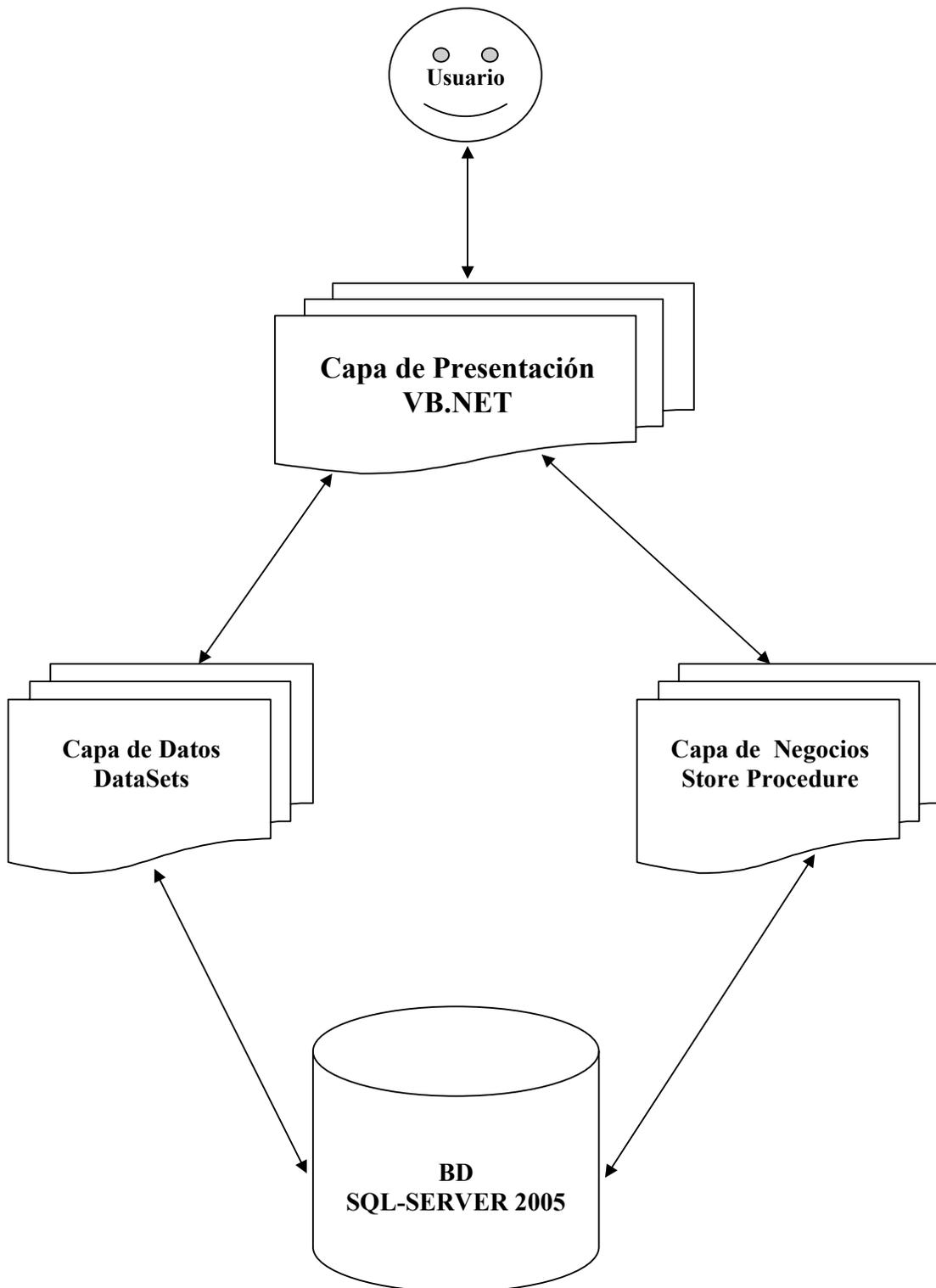
Esta capa es la que se encarga de procesar los datos según las reglas de negocio. La misma se modeló utilizando Transac-SQL, lenguaje para crear los Store Procedure (Procedimientos Almacenados) y Funciones. Todas las reglas de negocio están en la base de datos, en su mayoría en Store Procedure, y algunas en Funciones de la Base de Datos.

#### D. Capa de Presentación

Esta es la capa que se encarga de interactuar con el usuario final de la aplicación. La misma está compuesta por pantallas e informes. Ésta fue modelada en Visual Basic.Net. Ésta, además de hacer la presentación de los datos al usuario, hace la validación al momento de introducción de datos. Las validaciones locales o las que hace la capa de presentación son validaciones típicas, por ejemplo un tipo de datos,

que no existan campos vacíos, etc. y la validación que tiene que ver con las reglas de negocio se le pasan a la capa de reglas de negocio, devolviendo ésta un valor interpretado por la de presentación para su debida ínter actuación con el usuario.

## E. Diagrama de Diálogo de Capas



## F. Detalle de las Reglas del Negocio

Explica cada una de las reglas de negocio, procedimientos almacenados o funciones, documentando cual es el objetivo de cada una de ellas. Cada uno de los SP o Funciones se encuentran documentados, por lo que aquí no entraremos a ese detalle.

- 1. Genera\_orden\_de\_llegada:** Al momento de ingresar una Solicitud de Cuota es necesario generar un orden según su llegada. Esto se logra ejecutando este procedimiento Almacenado.
- 2. Genera\_orden\_de\_llegada\_por\_actualización:** Cuando una Solicitud de cuota se actualiza, se vuelve a generar el orden de llegada. Esto es útil cuando se completan los datos de una Solicitud que había sido rechazada. Al complementarse se toma como fecha de llegada la fecha que está completando los datos, por lo tanto hay que generar un nuevo orden de llegada según la Fecha en que se completa.
- 3. Ms\_cierra\_contingente:** Cuando un contingente ha cerrado su ciclo de vida útil, éste debe ser cerrado. Esto se logra mandando a ejecutar este procedimiento almacenado.
- 4. Ms\_datos\_cac:** Al momento de hacer un reporte o presentar por pantalla los datos relacionados de un Certificado de Asignación de Contingentes (CAC), lo mandamos a traer mediante el procedimiento almacenado.
- 5. Ms\_distribuir\_contingente\_en\_cuotas:** Una vez que se halla hecho la recepción de las solicitudes de cuota de un determinado Contingente, es necesario hacer la distribución según el reglamento. Esto se logra ejecutando este procedimiento almacenado.
- 6. Ms\_distribuir\_contingente\_en\_cuotas\_remanente:** Posterior a la distribución de cuotas es posible que quede un remanente o sobrante. Una vez que se ha hecho la publicación en los medios de dicho sobrante y se recibieron las solicitudes, se hace la distribución ejecutando este procedimiento.
- 7. Ms\_distribuir\_contingente\_en\_cuotas\_devolución:** En los primeros días del mes 9 de cada año los importadores deben de confirmar si utilizarán la cuota asignada o devolverán parte de ésta. Al finalizar este proceso, si hay devoluciones, éstas deben de asignarse, siempre siguiendo los pasos de publicación y recepción de solicitudes, ejecutando este procedimiento almacenado.
- 8. Ms\_genera\_consecutivo\_cac:** Al momento de crear una solicitud de CAC, si ésta es del Tipo CAFTA-DR se genera un consecutivo automático. Esto se logra ejecutando este procedimiento almacenado.

9. **Ms\_genera\_nueva\_solicitud:** Cuando se hace una solicitud de cuota, el ID de ésta es generado automáticamente por el software. Esto se logra ejecutando este procedimiento almacenado.
10. **Ms\_gradua\_importadores:** Los Importadores pueden graduarse en un determinado producto de un determinado TLC, y de esta manera tendrá mayores posibilidades de obtener cuotas. Para poder lograr la graduación de los Importadores ejecutamos este procedimiento almacenado.
11. **Ms\_preasigna\_cuotas:** En ciertos contingentes, cuando los importadores graduados solicitan más de lo que les corresponde, se procede a prorratear de acuerdo al porcentaje de importaciones históricas. Para calcular este porcentaje se ejecuta este procedimiento almacenado.
12. **Ms\_reindexar\_orden\_de\_llegada:** En las solicitudes de cuota es importantísimo el orden de llegada. Éste debe ser consecutivo y por lo tanto no debe de tener baches cuando se elimina una solicitud. Para lograr esto se ejecuta este procedimiento almacenado.
13. **Ms\_requiere\_certificado\_de\_libre\_venta:** Al solicitar un CAC el software revisa si el contingente requiere que se especifique el Certificado de Libre Venta. Esto se logra saber ejecutando este procedimiento almacenado.
14. **Ms\_revisar\_si\_tiene\_sanciones:** Al momento de registrar una Solicitud de Cuota es necesario verificar si el Importador posee sanciones en ese producto de ese TLC. Esto se logra ejecutando este procedimiento almacenado.
15. **Ms\_saldo\_de\_cuota\_para\_cac:** Al momento de solicitar un CAC se debe consultar el saldo sobre la Cuota de Asignación que se está solicitando. Esto se logra ejecutando este procedimiento almacenado.
16. **Ms\_toneladas\_metricas\_disponibles:** Al momento de hacer una solicitud de cuota es necesario saber cuantas toneladas métricas están disponibles, según sea el tipo de solicitud: Primera Asignación, Por Remanente ó Por Devolución.
17. **Ms\_totales\_consumos\_de\_cuotas:** Este procedimiento almacenado calcula el saldo según CAC y según importaciones; útil para hacer reportes.
18. **Ms\_valida\_reglas\_de\_asignacion:** Al momento de estar adjuntando los Productos a un Contingente el software valida que estos posean las mismas reglas de asignación. Esto se logra ejecutando este procedimiento almacenado.

- 19. Ms\_valida\_requisitos:** Al momento de Ingresar una Solicitud de Cuota se validan los requisitos que exige el Contingente. Esto se logra ejecutando este procedimiento almacenado.
- 20. Ms\_valida\_si\_se\_preasigna:** Al momento de ejecutar el proceso Preasignar, éste valida si el Contingente posee productos que se preasignan.
- 21. Año\_perpetuo:** Esta función devuelve el año en que una regla de asignación se vuelve perpetua.
- 22. Importaciones\_por\_importador:** Esta función devuelve el valor importado en un año de un TLC de un producto, por importador.
- 23. Importaciones\_promedio:** Esta función devuelve las importaciones promedio por importador.
- 24. Sac\_delos\_cac:** Devuelve los códigos Arancelarios de un CAC, estos para ser impresos en el CAC oficial.

## **G. Plataforma de Desarrollo**

La plataforma de desarrollo es Microsoft.Net, se utilizaron las siguientes herramientas:

1. Microsoft Framework 2.0
2. Microsoft SQL Server 2005
3. Microsoft Visual Basic.Net
4. Microsoft Windows 2003 Server.