



Hospitality and Specialty Retail: Iberia, Caribbean and Latin America

LASA Call Center Interface

Información Técnica

Historial de Actualizaciones

Versión	Fecha	Realizada por	Comentarios
1.0.0.0	18-May-12	Anael Brito	
1.0.0.0	22-May-12	Anael Brito	Adición de pasos de configuración de Aloha POS/Aloha Take Out a la documentación de Instalación y Configuración.
1.0.1.0	04-Jul-12	Anael Brito	Corrección de brecha en seguridad de comunicaciones
1.0.2.0	08-Feb-13	Anael Brito	Adición de código de respuesta UnknownException.
1.0.2.0	08-Feb-13	Anael Brito	Optimización de uso de recursos.
1.0.3.0	06-May-13	Anael Brito	Corrección de error en que al solicitar cancelar una orden si esta fue cancelada satisfactoriamente no se devolvía el código de respuesta Success.
2.0.0.0	05-Jul-13	Anael Brito	Adición de funciones para adicionar documentos de texto asociados a una cuenta.

Esquema de Datos

Al describir cada elemento en LASA Call Center Interface este documento presenta una tabla que detalla todos los posibles elementos hijos que están disponibles para un elemento padre en el esquema. Aunque la mayoría de los elementos pueden ser especificados tanto como de entrada como de salida, algunos elementos son utilizados exclusivamente como de entrada, así también otros están solamente disponibles como salida. Refiérase a la siguiente leyenda para una descripción de los símbolos mostrados en las tablas que representan las reglas de uso de cada elemento.

● Para entrada en el XML, Este símbolo indica que el elemento asociado puede ser opcionalmente especificado, pero no es requerido. Para salida en el XML, este símbolo indica que el elemento asociado estará presente si está disponible.

● Para entrada en el XML, Este símbolo indica que el elemento es requerido. De manera similar, para salida en el XML, este símbolo indica que el elemento asociado siempre estará presente.

✕ Para entrada en el XML, Este símbolo indica que el elemento es un elemento de salida únicamente y que no debe ser enviado por la aplicación externa. Para salida del XML, este símbolo indica que el elemento asociado es un elemento de entrada únicamente y que no será retornado por a la aplicación externa.

Definición del elemento Order

Order				
Elemento	Tipo	Descripción	Entrada	Salida
OrderId	Integer (64-bit)	ID de orden de ATO. Este número es generado por ATO para identificar de manera única la orden.	✕	●
ReferenceId	Integer (64-bit)	Un identificador opcional para permitir el seguimiento de la orden por parte de aplicaciones externas.	●	●
OrderNumber	Integer (32-bit)	Check ID en Aloha POS. Este número es generado por Aloha POS.	✕	●

SequenceNumber	Integer (16-bit)	Numero de secuencia generado por ATO que es incrementado y asignado a cada orden nueva. Este número se reinicia durante el Fin de Día.	X	●
Customer	Element	Nodo de un elemento tipo Customer . Consulte la definición del esquema de datos de Customer.	●	●
OrderTime	Date/Time	Fecha y hora en que la orden fue establecida.	●	●
PrepTime	Integer (32-bit)	Cantidad de tiempo en minutos requerida para preparar la orden.	●	●
PromiseTime	Date/Time	Fecha y hora en que el cliente espera que la orden esté lista.	●	●
FirstSentToKitchenTime	Date/Time	Hora en que la orden fue enviada a cocina por primera vez. Este valor no estará presente si la orden aun no ha sido enviada a cocina.	X	●
LastSentToKitchenTime	Date/Time	Hora en que la orden fue enviada a cocina por última vez. Esta difiere de FirstSentToKitchenTime cuando se agregan a la orden nuevos artículos después de que esta fue enviada a cocina. Este valor no estará presente si la orden aun no ha sido enviada a cocina.	X	●
FirstBumpTime	Date/Time	La primera vez que la orden es completada en cocina. Este valor no estará presente si la orden aun no ha salido de cocina.	X	●
LastBumpTime	Date/Time	La última vez que la orden es completada en cocina. Este valor no estará presente si la orden aun no ha salido de cocina.	X	●
FirstPaymentTime	Date/Time	Primera vez que se aplica un pago a la cuenta en el Punto de Venta. Este valor no estará presente si no ha sido aplicado ningún pago a la cuenta.	X	●
LastPaymentTime	Date/Time	Ultima vez que un pago fue aplicado a la cuenta en el Punto de Venta. Este valor no estará presente si ningún pago ha sido aplicado a la cuenta.	X	●
DispatchTime	Date/Time	Para órdenes de despacho, este valor representa la hora en que el motorista asignado a la orden salió de la tienda para entregar la orden. Este valor no estará presente para ordenes que no sean de despacho o que aun no han sido despachadas.	X	●
FulfilledTime	Date/Time	Hora en que la orden fue entregada al cliente. Para ordenes de despacho, esta es la hora en que motorista volvió de su ruta de entregas.	X	●
ClosedTime	Date/Time	Hora en que la orden fue cerrada en ATO.	X	●
Source	String	Fuente de la orden. Puede ser cualquiera de las siguientes: <ul style="list-style-type: none"> • None • Phone • Counter • Fax • Web • DriverThru 	● ¹	●

Mode	String	<p>Modo de pedido. Puede ser uno de los siguientes:</p> <ul style="list-style-type: none"> • None • CallIn • WalkIn • Delivery • EatIn • DriveThru • Curbside • FaxedIn • Web • Catering 	●	●
ModeCharge²	Decimal	<p>Monto de cargos por servicio adicionado a la orden, basado en el modo de pedido especificado (si aplica). Note que los cargos por servicio puede que no sean aplicados a la orden hasta que esta sea enviada a cocina en el restaurante. Cualquier cargo indicado en este elemento ya está incluido en el subtotal.</p>	✕	●
Status	String	<p>Estado de la orden en ATO. Puede ser uno de los siguientes:</p> <ul style="list-style-type: none"> • None Nueva orden. • PreDelay Nueva orden que será orden futura. • InDelay Orden retrasada en ATO (orden futura). • KitchenDelay Orden retrasada en el sistema de cocina. • Cooking Orden está siendo preparada en cocina. • Prepared Preparación de la orden completada. La orden esta lista para el cliente. • InTransit La orden ha sido despachada para ser entregada al cliente. • Fulfilled Orden ha sido entregada al cliente. • Closed Orden ha sido pagada y ha sido entregada al cliente. 	● ³	●
Items	Element	Colección de nodos tipo Item o FailedItem . Consulte la definición de los mismos.	●	●
SubTotal	Decimal	Monto total cargado para todos los artículos de la orden, excluyendo impuestos pero incluyendo los cargos por modo de pedido.	✕	●

DiscountTotal	Decimal	Monto total de descuentos aplicados a la orden. Este valor aun no es soportado por ATO.	X	●
Tax	Decimal	Monto de impuesto cargado a la orden.	X	●
TaxExempt	Boolean	Indica que esta orden debe ser exenta de impuesto. El cliente asociado debe tener un ID de exención de impuestos para que la orden pueda ser exenta. Un ID de exención de impuestos puede ser proveído también por medio del campo TaxId en el elemento Customer descrito posteriormente	○	○
PaymentTotal	Decimal	Monto total de todos los pagos aplicados a la cuenta.	X	○
Total⁴	Decimal	Monto total aplicado a la cuenta incluyendo impuestos.	X	●
EstimatedSubTotal	Decimal	Actualmente en desuso.	○	○
EstimatedTax	Decimal	Usado para indicar el impuesto que la aplicación externa estima que será aplicado. Si este valor es especificado y el impuesto final es mayor se aplicara un crédito impositivo para cubrir la diferencia.	○	○
Tendered	Boolean	Indica si la orden ha sido completamente pagada.	X	●
Tenders	Element	Colección de nodos tipo Tender. Vea la definición del elemento Tender.	○	○
Comps	Element	Colección de nodos tipo Comp. Vea la definición del elemento Tender.	○	○
Promos	Element	Colección de nodos tipo Promo. Vea la definición del elemento Tender.	○	○
DriverId	Integer (32-bit)	ID del motorista asociado a la orden en el sistema de punto de venta para órdenes de despacho.	X	○
OrderNotes	String	Texto que es asignado a nivel de orden para una orden, por ejemplo instrucciones especiales de entrega.	○	○













Notas del elemento Order

1. Si el elemento **Source** no es especificado por la aplicación externa, se establecerá como **Web** de manera predeterminada.
2. El elemento **ModeCharge** solo es soportado Aloha TakeOut 1.2.200 o superior.
3. Para ordenes futuras, el elemento Status debe establecerse como "InDelay" en las ordenes nuevas. Ordenes inmediatas deben omitir este elemento. Además debe establecer el campo PromiseTime con la fecha y hora en que desea que la orden este lista.
4. El elemento **Total** solo es soportado por Aloha TakeOut 1.2.44 o superior, sin embargo este valor puede obtenerse sumando los elementos **SubTotal** y **Tax**.

Definición del elemento Customer (como parte de una orden)

El esquema a continuación describe los detalles de un cliente para un cliente asociado a una orden que está siendo agregada o consultada de ATO. Este elemento Customer es distinto del elemento Customer (independiente) que es usado cuando se actualizan o consultan registros de clientes independientemente de órdenes. La principal diferencia es que los elementos Address y PhoneNumber son contenidos directamente bajo el elemento **Customer** cuando este se basa en una orden. Esto se debe a que cualquier orden solo puede tener una dirección y número de teléfono. Diferente al elemento **Customer** (independiente) descrito más adelante.

Customer (como parte de una orden)				
Elemento	Tipo	Descripción	Entrada	Salida
CustomerId	Integer (64-bit)	ID del cliente generado por ATO.	● ¹	●
ReferenceId	Integer (64-bit)	Identificador opcional del cliente que puede ser especificado por aplicaciones externas. Debe ser único para cada cliente.	● ¹	●
FirstName	String	Nombres del cliente.	● ¹	●
LastName	String	Apellidos del cliente.	● ¹	●
EmailAddress	String	Correo electrónico del cliente.	● ¹	●
PhoneNumber	String	Número de teléfono del cliente. ²	● ¹	●
PhoneType	String	Tipo de número de teléfono. Puede ser uno de los siguientes: <ul style="list-style-type: none"> • Home Teléfono de su casa. • Cell Teléfono móvil. • Business Teléfono del trabajo. 	● ³	●
AddressLine1	String	Primera línea de la dirección del cliente.	●	●
AddressLine2	String	Segunda línea de la dirección del cliente.	●	●
City	String	Ciudad de la dirección del cliente.	●	●
State	String	Estado de la dirección del cliente.	●	●
ZipCode	String	Código Zip de la dirección del cliente.	●	●
DeliveryZone	String	Nombre de la zona de despacho que contiene la dirección especificada. Esta es determinada por ATO basado en la configuración de áreas de despacho en el restaurante.	✕	●
AddressNotes	String	Notas asociadas con la dirección del cliente.	●	●

AddressType	String	Tipo de dirección del cliente. Puede ser una de las siguientes: <ul style="list-style-type: none"> • Home Dirección de la casa. • Business Dirección del trabajo. • Other Direcciones de ninguno de los tipos anteriores. 	 4	
SaveAddress	Boolean	Si es verdadero, la dirección del cliente en la orden será guardada como una de las direcciones del cliente en ATO. Si es falso, solo será usada en esta orden.	 5	
LoyaltyId	String	ID del programa de lealtad asociado al cliente.		
TaxId	String	ID de impuesto para uso en órdenes exentas de impuesto.		
Notes	String	Notas asociadas con el cliente.		
Vehicle	Element	Detalles del vehículo del cliente, usado para ordenes en la acera. Consulte la definición del elemento Vehicle.		

Notas del elemento Customer

1. Cuando se establece un cliente para una nueva orden, el cliente puede ser identificado de una de las siguientes tres formas: Por **CustomerId**, por **ReferenceId**, por **EmailAddress** o una combinación de **FirstName**, **LastName**, y **PhoneNumber**. Sin embargo, debe incluir toda la información del cliente en cada nueva orden de modo que un Nuevo cliente pueda ser creado en caso que este aun no se encuentre registrado.
2. Para ATO, el elemento **PhoneNumber** puede ser especificado son o sin caracteres separadores. Como entrada, solo el primer separador tiene relevancia en como ATO interpretara el número de teléfono. Si no se especifican separadores, ATO intentara aplicar el mejor formato basado en el tamaño del número de teléfono. Si al menos un separador es establecido, el primer separador será usado como el fin del código de área. Otros separadores serán ignorados. Los separadores validos son: '-', '.', '(', ')', o espacio. El elemento **PhoneNumber** retornado por ATO será formateado de acuerdo al diseño de número de teléfono establecido en la configuración del restaurante.
3. Si no se establece el elemento **PhoneType**, el teléfono será establecido como el número de la casa.
4. Si no se establece el elemento **AddressType**, este será establecido como la dirección de la casa.
5. Si no se establece el elemento **SaveAddress**, la dirección será usada para esta orden únicamente.

Definición del elemento Customer (de forma independiente)

El esquema a continuación describe los detalles del cliente como un registro independiente de una orden. Este objeto XML es usado para agregar y modificar clientes en la base de datos de ATO sin necesidad de crear una orden. Este esquema permite que múltiples números de teléfono y direcciones sean asociados con el cliente. El elemento Customer (basado en una orden) establecido anteriormente solo soporta un número de teléfono y dirección por orden.

Customer (independiente)				
Elemento	Tipo	Descripción	Entrada	Salida
CustomerId	Integer (64-bit)	ID del cliente generado por ATO	● ¹	●
ReferenceId	Integer (64-bit)	Identificador opcional del cliente que puede ser especificado por aplicaciones externas. Debe ser único para cada cliente.	● ¹	●
FirstName	String	Nombres del cliente.	● ¹	●
LastName	String	Apellidos del cliente.	● ¹	●
EmailAddress	String	Correo electrónico del cliente.	● ¹	●
PhoneNumbers	Element	Colección de elementos tipo PhoneNumber . Consulte la definición del elemento PhoneNumber.	● ¹	●
Addresses	Element	Colección de elemento tipo Address . Consulte la definición del elemento Address.	●	●
LoyaltyId	String	ID del programa de lealtad asociado al cliente.	●	●
TaxId	String	ID de impuesto para uso en órdenes exentas de impuesto.	●	●
Notes	String	Notas asociadas con el cliente.	●	●
Vehicle	Element	Detalles del vehículo del cliente, usado para ordenes en la acera. Consulte la definición del elemento Vehicle.	●	●
OrderCount	Integer (32-bit)	Numero de ordenes de ATO completadas para este cliente.	✕	●

Notas del elemento Customer

1. Cuando se especifica un cliente existente para actualización, el cliente puede ser identificado por uno de los siguientes métodos: **CustomerId**; **ReferenceId**; **EmailAddress**; o una combinación de **FirstName**, **LastName**, y **PhoneNumber**.























Definición del elemento Phone Number

PhoneNumber				
Elemento	Tipo	Descripción	Entrada	Salida
Type	String	Tipo de número de teléfono, puede ser uno de los siguientes: <ul style="list-style-type: none">• Home Teléfono de la casa.• Cell Teléfono móvil.• Business Teléfono de la oficina.	● ¹	●
Number	String	Número de teléfono incluyendo el código de área. Consulte las notas acerca del formato.	● ²	●
Extension	String	Numero de extensión asociada con el número de teléfono. Este normalmente se usa solo para teléfono del trabajo.	●	●
IsPreferred	Boolean	Si verdadero, indica que el número de teléfono asociado es el número preferido del cliente para contactarlo. Solo un número en la colección debe tener este valor establecido en verdadero.	●	●

Notas del elemento Phone Number

1. Si no se envía el elemento **PhoneType** entonces el teléfono se utilizara como el teléfono de la casa.
2. Para ATO, el elemento **Number** puede ser especificado con o sin caracteres separadores. Como entrada, solo el primer carácter tiene importancia en como ATO interpretara el número de teléfono. Si no se especifican separadores, ATO intentara aplicar el mejor formato basado en el tamaño del número de teléfono suministrado. Si al menos un separador es suministrado, el primer separador será aplicado para denotar el fin de la porción del número de teléfono correspondiente al número de área. Todos los demás separadores son ignorados. Los separadores validos son: '-', ':', '(', ')', o espacio. El elemento **Number** retornado por ATO estará en el formato de número de teléfono de acuerdo al formato establecido en la configuración del restaurante.

Definición del elemento Address

Address				
Elemento	Tipo	Descripción	Entrada	Salida
Type	String	Tipo de dirección asociada con el cliente. Puede ser uno de los siguientes: <ul style="list-style-type: none"> • Home Dirección de la casa • Business Dirección del trabajo • Other Otra dirección 	 ¹	
AddressLine1	String	Primera línea de la dirección del cliente		
AddressLine2	String	Segunda línea de la dirección del cliente		
City	String	Ciudad de la dirección del cliente		
State	String	Estado o región de la dirección del cliente		
ZipCode	String	Código postal de la dirección del cliente		
Country	String	País de la dirección del cliente		
Company	String	Nombre de la compañía asociada a la dirección en caso que se trate de una dirección del trabajo		
DeliveryZone	String	Nombre de la zona de despacho que contiene la dirección especificada. Esta es determinada por la configuración de áreas de despacho de ATO.		
AddressNotes	String	Notas asociadas a la dirección del cliente		
IsPreferred	Boolean	Si es verdadero, indica que la dirección asociada es la dirección preferida para contactar al cliente. Solo una dirección en la colección debe tener este campo verdadero.		

Notas del elemento Address

1. Si no se especifica el elemento **AddressType** entonces la dirección se considerara como de tipo **Home** de manera predeterminada.

Definición del elemento Item/FailedItem

Normalmente, solo el elemento **Item** aparecerá en el XML para una orden. El elemento **FailedItem** es usado para representar artículos que fallaron en ser inyectados en ATO por alguna razón (ej. El artículo no está disponible para el sitio). El elemento **FailedItem** maneja exactamente el mismo esquema que el elemento **Item**. La única razón de introducción del elemento **FailedItem** es para proveer compatibilidad a aplicaciones externas que no esperan que los artículos fallidos sean retornados en el XML de la orden. Con versiones más antiguas de la API, solo los artículos inyectados satisfactoriamente eran retornados en la colección **Items** del elemento **Order**. Este modelo debe permitir a cualquier aplicación que dependa del esquema antiguo continuar su operación de la misma manera.

Item/FailedItem				
Elemento	Tipo	Descripción	Entrada	Salida
ItemId	Integer (32-bit)	ID de artículo en ATO. Este número es generado por ATO para identificar de manera única este artículo con esta orden. Este ID puede no ser constante a lo largo de la vida de la orden.	✗	●
ReferenceId	Integer (32-bit)	Identificador opcional del artículo a ser definido por la aplicación externa. Cada identificador debe ser único. Este ID puede ser referenciado por una promoción o un descuento.	○	○
Level	Integer (8-bit)	Indica la posición de los artículos dentro del árbol de herencia del artículo. Artículos en la raíz de la orden (no modificadores) tienen un nivel de cero.	✗	●
MenuItemId	Integer (32-bit)	Identificador que especifica el artículo basado en la base de datos del punto de venta.	●	●
ModCodeId	Integer (32-bit)	Código modificador asociado con el artículo, tal y como fue definido en el punto de venta	○	○
ModGroupId	Integer	ID de grupo de modificadores asociado con el modificador.	○	○
Name	String	Nombre descriptivo del artículo.	○	○
Quantity	Integer	Número de veces que este artículo es incluido en la orden. El valor predeterminado es 1 a menos que se indique lo contrario.	○	●
Price	Decimal	Precio del artículo. Si se especifica, el sistema utilizara este precio, caso contrario se usara el precio establecido en el punto de venta.	○	○
SubItems	Element	Colección de elementos Item o FailedItem que representan los sub-ítems o modificadores del artículo.	○	○

ResultCode	String	Como respuesta, indica si el artículo especificado fue satisfactoriamente inyectado al punto de venta en el restaurant. El código de resultado Failed solo deben recibirse en los elementos FailedItem. Consulte la definición de los códigos de respuesta para más detalles.	X	●
------------	--------	---	---	---

Definición del elemento SpecialMessage

SpecialMessage				
Elemento	Tipo	Descripción	Entrada	Salida
Message	String	Texto para el mensaje.	●	●

Definición del elemento Tender

Tender				
Elemento	Tipo	Descripción	Entrada	Salida
Type	String	Modo de pago que el elemento Tender representa. Puede ser uno de los siguientes: <ul style="list-style-type: none"> • Cash¹ • Visa • MasterCard • Amex • Discover • Diners • JCB • HouseAccount² • PrePaid • GiftCard 	●	●
CardNumber	String	Número de cuenta asociado con un pago con tarjeta de crédito o gift card.	● ³	X
ExpirationDate	Date	Fecha de expiración para pagos con tarjeta de crédito. La fecha debe ser especificada en el siguiente formato: YYYY-MM-DD.	● ³	X
PayBalance	Boolean	Campo bandera que indica que el monto de pago debe ser igual al balance pendiente.	●	●
Amount	Decimal	Monto de pago aplicado a la orden. Este elemento debe ser ignorado si el campo PayBalance es establecido en true.	● ⁴	●

Tip	Decimal	Monto de propina aplicado a la orden. Este monto es adicional al campo Amount descrito anteriormente.	●	●
AuthCode	String	Código de autorización retornado por el procesador de tarjetas de crédito para pagos con tarjetas de crédito.	✕	●
Status	String	Estatus del pago. Puede ser uno de los siguientes: <ul style="list-style-type: none"> • None • Waiting • Approved • Declined • Failed • PreAuth 	✕	●

Notas del elemento Tender

1. El modo de pago **Cash** es incluido solo en consultas. Este no debe ser especificado al inyectar una orden.
2. Si el modo de pago **HouseAccount** es especificado al inyectar una orden, el monto asociado será cargado a la house account asociada con el cliente en ATO. Si el cliente no tiene una house account asociada al sitio, la inyección de la orden fallara.
3. Los pagos con tarjeta de crédito requieren los elementos **CardNumber** y **ExpirationDate**.
4. El elemento Amount no es requerido cuando el elemento **PayBalance** es true.

Definición del elemento Promo

El elemento Promo representa una promoción aplicada a la orden, como un cupón o un precio especial de combo. Algunas promociones pueden aplicar a toda la cuenta, mientras que otras pueden aplicar solo a ciertos artículos específicos de la cuenta. En caso que la cuenta aplique solo a ciertos artículos específicos de la cuenta puede ser necesario seleccionar a que artículos de la orden aplica la promoción. Dependiendo del tipo de promoción, el punto de venta en el restaurante puede automáticamente seleccionar los artículos a los cuales la promoción aplica.

Promo				
Elemento	Tipo	Descripción	Entrada	Salida
PromoTypeId	Integer (32-bit)	Indica el tipo de promoción tal y como fue definido en el punto de venta	●	●
IdentifierData	String	Algunos tipos de promociones requieren información de identificaciones adicional, como por ejemplo un número de cupón.	●	●
Amount ¹	Decimal	Monto monetario de la promoción.	✕	✕
ItemSelections	Element	Colección de elementos ItemSelection que identifican los artículos a los cuales aplica la promoción	●	●

Notas del elemento Promo

1. El valor de **Amount** no está soportado actualmente por el entorno ATO para el elemento Promo. Este aparece aquí documentado para potencial futuro uso.

Definición del elemento Comp

Comp				
Elemento	Tipo	Descripción	Entrada	Salida
CompTypeid	Integer (32-bit)	Indica el tipo de descuento tal y como aparece en el punto de venta	●	●
Rate	Decimal	Porcentaje de descuento. Si es enviado, el valor debe ser entre 0.01 (1%) y 1.00 (100%).	● ¹	●
Amount	Decimal	Monto monetario de descuento.	● ²	●
ItemSelections	Element	Colección de elementos ItemSelection que identifican los artículos a los que el descuento aplica	●	●

Notas del elemento Comp

1. El valor del campo **Rate** solo debe ser especificado por la aplicación externa cuando el tipo de descuento indicado es configurado en el restaurante para requerir el ingreso de un porcentaje. Cuando la orden es consultada, el campo Rate contendrá el porcentaje de descuento que fue aplicado realmente, sin importar si este fue enviado o ya se encontraba configurado.
2. El valor de **Amount** solo debe ser especificado por la aplicación externa cuando el tipo de descuento indicado es configurado en el restaurante para requerir digitar el monto de descuento. Cuando la orden es consultada, el campo Rate contendrá el porcentaje de descuento que fue aplicado realmente, sin importar si este fue enviado o ya se encontraba configurado.

Definición del elemento ItemSelection

El elemento **ItemSelection** es usado en los elementos **Comp** y **Promo** para especificar a qué artículos aplicar la promoción o descuento. Algunas promociones requieren múltiples grupos de selección. Por ejemplo, para una promoción donde tu compras un artículo y obtienes otro gratis, puede ser necesario identificar ambos artículos, el artículo calificador y el artículo gratis. Esto se haría con dos grupos de selección por separado.

ItemSelection				
Elemento	Tipo	Descripción	Entrada	Salida
Groupid (Attribute)	Integer (32-bit)	Identifica el grupo de selección al que pertenecen los artículos especificados. Si el este elemento es omitido se usara el Group id 0. Esto es normalmente suficiente para la mayoría de las promociones y descuentos que requieren un solo grupo de selección.	●	●
ItemReferenceid	Integer (32-bit)	Identifica a un artículo por su id de referencia que es incluido en el grupo de selección. Este elemento debe ser repetido por cada artículo en el grupo de selección actual.	●	●

Definición del elemento Vehicle

Vehicle				
Elemento	Tipo	Descripción	Entrada	Salida
Id	Integer (64-bit)	ID del vehículo generado por ATO.	✕	●
Make	String	Marca del vehículo (ej. "Ford").	○	○
Model	String	Modelo del vehículo (ej. "Ranger").	○	○
Color	String	Color del vehículo (ej. "Azul").	○	○

Notas del elemento Vehicle

1. Aplicaciones externas no deben especificar el elemento ID para nuevos vehículos. El elemento será generado por ATO durante el procesamiento de la orden.

Definición del elemento Driver

El esquema del elemento Driver es usado cuando se consulta el estatus de un motorista en ATO. Este es un elemento de salida únicamente. La información del motorista no puede ser inyectada a ATO.

Driver				
Elemento	Tipo	Descripción	Entrada	Salida
DriverId	Integer (32-bit)	ID del motorista en el punto de venta	X	●
FirstName	String	Nombre del motorista	X	●
LastName	String	Apellido del motorista	X	●
Status	String	<p>Estado actual del motorista. Puede ser uno de los siguientes:</p> <ul style="list-style-type: none"> • Unknown Estado del motorista actualmente desconocido • Idle Motorista ha fichado entrada, pero no se encuentra despachando ninguna orden • In Motorista a regresado de un despacho • Out Motorista se encuentra entregando una orden de despacho • OffDuty Motorista ha fichado salida 	X	●
LastDispatchTime	Date/Time	Ultima vez en la fecha del negocio actual en que el motorista salió a entregar una orden de despacho	X	●
LastReturnTime	Date/Time	Ultima vez en la fecha del negocio actual en que el motorista regreso de entregar una orden de despacho	X	●
CurrentOrderCount	Integer (32-bit)	Numero de ordenes pendientes actualmente asignadas al motorista	X	●
CompletedOrderCount	Integer (32-bit)	Número total de ordenes entregadas por el motorista durante la fecha del negocio actual	X	●

Definición del elemento PrintDocument

El esquema del elemento PrintDocument es usado cuando se adiciona un texto a una cuenta existente y está formado por varios comandos de impresión que pueden ser especificados en cualquier orden dentro del elemento PrintDocument.

PrintDocument				
Elemento	Tipo	Descripción	Entrada	Salida
PrintDocument	Element	Colección de comandos de impresión.	●	✕

A continuación se menciona la estructura de cada uno de los comandos de impresión. Comenzaremos con los comandos de impresión que solo reciben un parámetro.

Comandos de impresión que reciben un solo parámetro				
Elemento	Tipo	Descripción	Entrada	Salida
PrintLine	String (35)	Imprime una línea de texto justificado a la izquierda	●	✕
PrintCentered	String (35)	Imprime una línea de texto centrado	●	✕
LineFeed	Int (Max 20)	Imprime la cantidad especificada de líneas en blanco	●	✕
PrintFilled	Char (1)	Rellena una línea de texto repitiendo el carácter especificado. Por ejemplo para imprimir una línea de asteriscos.	●	✕

A continuación se menciona la estructura de los comandos de impresión que reciben más de un parámetro.

Comandos de impresión que reciben más de un parámetro				
Elemento	Tipo	Descripción	Entrada	Salida
PrintLeftRight	Elements	Imprime dos textos en la misma línea. Un texto justificado a la izquierda y otro a la derecha. Recibe como parámetros dos elementos, un elemento Left y un elemento Right.	●	✕
PrintLeftCenter	Elements	Imprime dos textos en la misma línea. Un texto justificado a la izquierda y otro centrado. Recibe como parámetros dos elementos, un elemento Left y un elemento Center.	●	✕
PrintRight	Elements	Imprime dos textos en la misma línea. Un texto centrado y otro justificado a la derecha. Recibe como parámetros dos elementos, un elemento Left y un elemento Center.	●	✕
PrintLeftCenterRight	Elements	Imprime tres textos en la misma línea. Un texto justificado a la izquierda, uno al centro y otro a la derecha. Recibe como parámetros tres elementos, un elemento Left, un elemento Center y un elemento Right.	●	✕
PrintLeftRightEx	Elements	Imprime un texto y rellena la línea con un carácter. El texto es recibido con el parámetro Left y el carácter es recibido con el parámetro Right.	●	✕
PrintStyle	Elements	Establece el estilo de impresión utilizado a partir de su declaración. Recibe dos parámetros: un elemento CPI y un elemento Style	●	✕

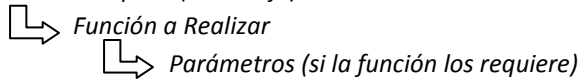
A continuación se menciona la estructura de los parámetros los comandos de impresión que reciben más de un parámetro.

Comandos de impresión que reciben más de un parámetro				
Elemento	Tipo	Descripción	Entrada	Salida
Left	String (35)	Texto justificado a la izquierda	●	✕
Center	String (35)	Texto justificado al centro	●	✕
Right	String (35)	Texto justificado a la derecha	●	✕
CPI	Integer (32 bit)	Este valor siempre debe ser enviado con el valor 1	●	✕
Style	Integer (32 bit)	Establece el estilo de fuente del texto. Debe utilizar uno de los siguientes valores: <ul style="list-style-type: none"> • 0 - Letra Normal • 1 - Letra Mediana • 2 - Normal Doble Altura • 3 - Letra Grande • 4 - Letra Normal Bold • 5 - Letra Mediana Bold 	●	✕

Estructura de Mensajes

A continuación se menciona la estructura de cada una de las funciones soportadas por LASA Call Center Interface. La siguiente es la estructura básica de una llamada a una función:

CallCenterRequest (Valor Fijo)



Por ejemplo GetCustomersByName se llamaría de la siguiente forma:

```
<CallCenterRequest>
  <GetCustomersByName>
    <CustomerFirstName>
      Juan
    </CustomerFirstName>
    <CustomerLastName>
      Perez
    </CustomerLastName>
  </GetCustomersByName>
</CallCenterRequest>
```

La respuesta a cada función, en caso que haya sido ejecutada satisfactoriamente, varía de acuerdo a la función misma, por ejemplo algunas funciones devuelven un registro, otras devuelven un listado de registros, en todos los casos la respuesta se recibe en formato XML. **En caso de error** todas las funciones devuelven un resultado en el siguiente formato donde ResultString es el código de respuesta recibido de acuerdo al listado mencionado en la sección **Códigos de Respuesta** que se encuentra en este mismo documento:

```
<CallCenterResult>
  ResultString
</CallCenterResult>
```

GetAllDrivers

Esta función devuelve un listado de todos los motoristas registrados en Aloha Take Out. No recibe parámetros. La solicitud de la misma se hará de la siguiente forma:

```
<CallCenterRequest>
  GetAllDrivers
</CallCenterRequest>
```

Su respuesta, en caso que la función haya sido ejecutada correctamente, iniciaría con un elemento Drivers que contiene una lista de elementos Driver (antes mencionado en este documento). Un ejemplo de respuesta exitosa de esta función es el siguiente:

```
<? xml version="1.0"?>
<Drivers xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Driver>
    <DriverId>920</DriverId>
    <FirstName>DiCaprio</FirstName>
    <LastName>Leonardo</LastName>
    <Status>OffDuty</Status>
    <CurrentOrderCount>0</CurrentOrderCount>
    <CompletedOrderCount>0</CompletedOrderCount>
  </Driver>
  <Driver>
    <DriverId>921</DriverId>
    <FirstName>Pattinson</FirstName>
    <LastName>Robert</LastName>
    <Status>OffDuty</Status>
    <CurrentOrderCount>0</CurrentOrderCount>
    <CompletedOrderCount>0</CompletedOrderCount>
  </Driver>
</Drivers>
```

GetDriverById

Esta función devuelve un elemento Driver con la información del motorista cuyo ID corresponda al valor enviado en el parámetro DriverId. Un ejemplo de una llamada a la misma, en este caso obteniendo los datos del motorista con ID 920, se haría de la siguiente forma:

```
<CallCenterRequest>
  <GetDriverById>
    <DriverId>
      920
    </DriverId>
  </GetDriverById>
</CallCenterRequest>
```

Su respuesta, en caso que la función haya sido ejecutada correctamente, sería un elemento Driver (antes mencionado en este documento). Un ejemplo de respuesta exitosa de esta función es el siguiente:

```
<?xml version="1.0"?>
<Driver xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <DriverId>920</DriverId>
  <FirstName>DiCaprio</FirstName>
  <LastName>Leonardo</LastName>
  <Status>Idle</Status>
  <CurrentOrderCount>0</CurrentOrderCount>
  <CompletedOrderCount>0</CompletedOrderCount>
</Driver>
```

GetAllCustomers

Esta función devuelve un listado de todos los clientes registrados en Aloha Take Out. No recibe parámetros. La solicitud de la misma se hará de la siguiente forma:

```
<CallCenterRequest>
  GetAllCustomers
</CallCenterRequest>
```

Su respuesta, en caso que la función haya sido ejecutada correctamente, iniciaría con un elemento Customers que contiene una lista de elementos Customer (antes mencionado en este documento). Un ejemplo de respuesta exitosa de esta función es el siguiente:

```
<?xml version="1.0"?>
<Customers xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Customer>
    <CustomerId>44225278255526</CustomerId>
    <Referenceld>0</Referenceld>
    <FirstName>Sofia</FirstName>
    <LastName></LastName>
    <OrderCount>1</OrderCount>
    <Addresses />
    <PhoneNumbers>
      <PhoneNumber>
        <Type>Home</Type>
        <Number>9555555</Number>
        <Extension />
        <IsPreferred>true</IsPreferred>
      </PhoneNumber>
    </PhoneNumbers>
  </Customer>
  <Customer>
    <CustomerId>44225278250205</CustomerId>
    <Referenceld>0</Referenceld>
    <FirstName>Marcos</FirstName>
    <LastName></LastName>
    <OrderCount>4</OrderCount>
    <Addresses>
      <Address>
        <Type>Home</Type>
        <AddressLine1>Camino Gino Girardi 2277</AddressLine1>
        <AddressLine2 />
        <City>Santiago</City>
        <State>CHILE</State>
        <ZipCode />
        <Country />
        <Company />
        <IsPreferred>true</IsPreferred>
      </Address>
    </Addresses>
    <PhoneNumbers>
      <PhoneNumber>
        <Type>Home</Type>
        <Number>7938895</Number>
        <Extension />
        <IsPreferred>true</IsPreferred>
      </PhoneNumber>
    </PhoneNumbers>
  </Customer>
</Customers>
```

GetCustomerById

Esta función devuelve un elemento Customer con la información del cliente cuyo ID corresponda al valor enviado en el parámetro CustomerId. Un ejemplo de una llamada a la misma, en este caso obteniendo los datos del cliente con ID -8589934593, se haría de la siguiente forma:

```
<CallCenterRequest>
  <GetCustomerById>
    <CustomerId>
      -8589934593
    </CustomerId>
  </GetCustomerById>
</CallCenterRequest>
```

Su respuesta, en caso que la función haya sido ejecutada correctamente, sería un elemento Customer (antes mencionado en este documento). Un ejemplo de respuesta exitosa de esta función es el siguiente:

```
<?xml version="1.0"?>
<Customer>
  <CustomerId>44225278255526</CustomerId>
  <ReferenceId>0</ReferenceId>
  <FirstName>Sofia</FirstName>
  <LastName>-</LastName>
  <OrderCount>1</OrderCount>
  <Addresses />
  <PhoneNumbers>
    <PhoneNumber>
      <Type>Home</Type>
      <Number>9555555</Number>
      <Extension />
      <IsPreferred>true</IsPreferred>
    </PhoneNumber>
  </PhoneNumbers>
</Customer>
```

GetCustomerByReference

Esta función devuelve un elemento Customer con la información del cliente cuyo número de referencia corresponda al valor enviado en el parámetro CustomerReference. Un ejemplo de una llamada a la misma, en este caso obteniendo los datos del cliente con referencia 999, se haría de la siguiente forma:

```
<CallCenterRequest>
  <GetCustomerByReference>
    <CustomerReference>
      999
    </CustomerReference>
  </GetCustomerByReference>
</CallCenterRequest>
```

Su respuesta, en caso que la función haya sido ejecutada correctamente, sería un elemento Customer (antes mencionado en este documento).

GetCustomersByName

Esta función devuelve un listado de los elementos Customer que cumplan con los parámetros FirstName y LastName recibidos. Un ejemplo de una llamada a la misma, en este caso obteniendo los clientes con primer nombre Anael y apellido Brito, se haría de la siguiente forma:

```
<CallCenterRequest>
  <GetCustomersByName>
    <CustomerFirstName>
      Anael
    </CustomerFirstName>
    <CustomerLastName>
      Brito
    </CustomerLastName>
  </GetCustomersByName>
</CallCenterRequest>
```

Su respuesta, en caso que la función haya sido ejecutada correctamente, sería un listado de elementos Customer (antes mencionado en este documento).

GetCustomersByPhone

Esta función devuelve un listado de los elementos Customer que cumplan con los parámetros FirstName y LastName recibidos. Un ejemplo de una llamada a la misma, en este caso obteniendo los clientes con número de teléfono 25502821, se haría de la siguiente forma:

```
<CallCenterRequest>
  <GetCustomersByPhone>
    <CustomerPhone>
      25502821
    </CustomerPhone>
  </GetCustomersByPhone>
</CallCenterRequest>
```

Su respuesta, en caso que la función haya sido ejecutada correctamente, sería un listado de elementos Customer (antes mencionado en este documento).

AddCustomer

Agrega el cliente especificado a ATO. Recibe como parámetro un elemento Customer. Un ejemplo de una llamada a la misma sería el siguiente:

```
<CallCenterRequest>
  <AddCustomer>
    <Customer>
      <FirstName>Anael</FirstName>
      <LastName>Brito</LastName>
      <AddressLine1 />
      <AddressLine2 />
      <AddressLine2>123 Main St.</AddressLine2>
      <City>Anytown</City>
      <EmailAddress>anael.brito@ncr.com</EmailAddress>
      <ReferenceId>20120430102049</ReferenceId>
      <Notes>Notas de Anael Brito</Notes>
    </Customer>
  </AddCustomer>
</CallCenterRequest>
```

Si el cliente fue adicionado correctamente se obtendrá como resultado un elemento Customer con la información del mismo tal y como se encuentra en ATO.

UpdateCustomer

Esta función actualiza la información del cliente especificado en ATO. Recibe como parámetro un elemento Customer con toda la nueva información del cliente. Un ejemplo de una llamada a la misma sería el siguiente:

```
<CallCenterRequest>
  <UpdateCustomer>
    <Customer>
      <FirstName>Anael</FirstName>
      <LastName>Brito</LastName>
      <AddressLine1 />
      <AddressLine2 />
      <AddressLine2>123 Main St.</AddressLine2>
      <City>Anytown</City>
      <EmailAddress>anael.brito@ncr.com</EmailAddress>
      <ReferenceId>20120430102049</ReferenceId>
      <Notes>Notas de Anael Brito</Notes>
    </Customer>
  </UpdateCustomer>
</CallCenterRequest>
```

Si el cliente fue actualizado correctamente se obtendrá como resultado un elemento Customer con la información del mismo tal y como se encuentra en ATO.

DeleteCustomer

Esta función elimina el cliente especificado. Recibe como parámetro el ID del cliente a eliminar. No puede eliminar un cliente que tenga ordenes pendientes en ATO. Un ejemplo de una llamada a esta función es el siguiente:

```
<CallCenterRequest>
  <DeleteCustomer>
    <CustomerId>
      10000
    </CustomerId>
  </DeleteCustomer>
</CallCenterRequest>
```

Su resultado es un código de respuesta indicando si la función se ejecuto correctamente, tal y como se describió anteriormente en este documento.

GetAllCurrentOrders

Esta función devuelve un listado de todas las órdenes actuales registradas en Aloha Take Out. No recibe parámetros. La solicitud de la misma se hará de la siguiente forma:

```
<CallCenterRequest>
  GetAllCurrentOrders
</CallCenterRequest>
```

GetAllFutureOrders

Esta función devuelve un listado de todas las órdenes futuras registradas en Aloha Take Out. No recibe parámetros. La solicitud de la misma se hará de la siguiente forma:

```
<CallCenterRequest>
  GetAllFutureOrders
</CallCenterRequest>
```

GetOrderById

Esta función obtiene un elemento Order que corresponde a la orden con el ID especificado en el parámetro OrderId. Un ejemplo de una llamada a la misma sería el siguiente:

```
<CallCenterRequest>
  <GetOrderById>
    <OrderId>506806165290</OrderId>
  </GetOrderById>
</CallCenterRequest>
```

GetOrderByReference

Esta función obtiene un elemento Order que corresponde a la orden con el número de referencia especificado en el parámetro Reference. Un ejemplo de una llamada a la misma sería el siguiente:

```
<CallCenterRequest>
  <GetOrderByReference>
    <Reference>123456</Reference>
  </GetOrderByReference>
</CallCenterRequest>
```

GetOrderByCheckId

Esta función obtiene un elemento Order que corresponde a la orden con el ID de Cuenta en el punto de venta (ID interno de Aloha) especificado en el parámetro CheckId. Un ejemplo de una llamada a la misma sería el siguiente:

```
<CallCenterRequest>
  <GetOrderByCheckId>
    <CheckId>1048578</CheckId >
  </GetOrderByCheckId >
</CallCenterRequest>
```

AddOrder

Esta función inyecta la orden especificada en ATO. Recibe como parámetro un elemento Order. Si esta se ejecuto correctamente devuelve un elemento Order con la información de la orden tal y como fue creada en ATO. Para crear órdenes futuras refiérase a la definición del elemento Order. Un ejemplo de una llamada a esta función es la siguiente:

```
<CallCenterRequest>
  <AddOrder>
    <Order>
      <Customer>
        <FirstName>Daniel</FirstName>
        <LastName>Brito</LastName>
        <AddressLine1>123 Main St.</AddressLine1>
        <City>Anytown</City>
        <State>CA</State>
        <ZipCode>12345</ZipCode>
        <PhoneNumber>555-555-0000</PhoneNumber>
        <Notes>Deliver to deck in back.</Notes>
      </Customer>
      <Mode>Delivery</Mode>
      <Items>
        <Item>
          <MenuItemId>4073</MenuItemId>
          <Quantity>1</Quantity>
          <Price>40.05</Price>
        </Item>
      </Items>
    </Order>
  </AddOrder>
</CallCenterRequest>
```

Para ordenes futuras, el elemento Status debe establecerse como “InDelay” en las ordenes nuevas. Ordenes inmediatas deben omitir este elemento. Además debe establecer el campo PromiseTime con la fecha y hora en que desea que la orden este lista.

CalculateTaxAndTotal

Esta función inyecta la orden especificada en el punto de venta para calcular el total y los impuestos. Este método es similar a AddOrder, la principal diferencia entre ambos es que en CalculateTaxAndTotal la orden es anulada después de la realización de los cálculos. Si esta se ejecuto correctamente devuelve un elemento Order con los datos de la orden.

CancelOrder

Esta función cancela la orden con el ID especificado. Recibe dos parámetros, OrderId que indica el ID en ATO de la orden a cancelar, y CancellIfOrdered que si se especifica en verdadero, la orden será cancelada aunque esta ya haya sido enviada a cocina, si se especifica en falso, la orden solo será cancelada si esta aun no ha sido enviada a la cocina. Su resultado es un código de respuesta indicando si la función se ejecuto correctamente, tal y como se describió anteriormente en este documento. Un ejemplo de una llamada a esta función es la siguiente:

```
<CallCenterRequest>
  <CancelOrder>
    <OrderId>506806165290</OrderId>
    <CancellIfOrdered>False</CancellIfOrdered>
  </CancelOrder>
</CallCenterRequest>
```

CancelOrderByReference

Esta función cancela la orden con el número de referencia especificado. Recibe dos parámetros, Reference que indica el numero de referencia en ATO de la orden a cancelar, y CancellIfOrdered que si se especifica en verdadero, la orden será cancelada aunque esta ya haya sido enviada a cocina, si se especifica en falso, la orden solo será cancelada si esta aun no ha sido enviada a la cocina. Su resultado es un código de respuesta indicando si la función se ejecuto correctamente, tal y como se describió anteriormente en este documento. Un ejemplo de una llamada a la misma es el siguiente:

```
<CallCenterRequest>
  <CancelOrderByReference>
    <Reference>123456</Reference>
    <CancellIfOrdered>False</CancellIfOrdered>
  </CancelOrderByReference>
</CallCenterRequest>
```

AddTextToOrderById

Esta función adiciona un documento de texto a la orden con el ID especificado en el parámetro OrderId. Al estar vinculado el texto a la orden este será impreso en la impresora de Aloha POS al imprimir o reimprimir la cuenta. Un ejemplo de una llamada a la misma sería el siguiente:

```
<CallCenterRequest>
  <AddTextToOrderById>
    <OrderId>506806165290</OrderId>
    <PrintDocument>
      <PrintCentered>Texto Centrado</PrintCentered>
      <PrintLeftRight><Left>Texto a la izquierda</Left><Right>Texto a la derecha</Right></PrintLeftRight>
    </PrintDocument>
  </AddTextToOrderById>
</CallCenterRequest>
```

Su resultado es un código de respuesta indicando si la función se ejecutó correctamente, tal y como se describió anteriormente en este documento.

AddTextToOrderByReference

Esta función adiciona un documento de texto a la orden con la referencia especificada en el parámetro Reference. Al estar vinculado el texto a la orden este será impreso en la impresora de Aloha POS al imprimir o reimprimir la cuenta. Un ejemplo de una llamada a la misma sería el siguiente:

```
<CallCenterRequest>
  <AddTextToOrderByReference>
    <Reference>506806165290</Reference>
    <PrintDocument>
      <PrintCentered>Texto Centrado</PrintCentered>
      <PrintLeftRight><Left>Texto a la izquierda</Left><Right>Texto a la derecha</Right></PrintLeftRight>
    </PrintDocument>
  </AddTextToOrderByReference>
</CallCenterRequest>
```

Su resultado es un código de respuesta indicando si la función se ejecutó correctamente, tal y como se describió anteriormente en este documento.

AddTextToOrderByCheckId

Esta función adiciona un documento de texto a la orden con el Check Id especificado en el parámetro CheckId. Al estar vinculado el texto a la orden este será impreso en la impresora de Aloha POS al imprimir o reimprimir la cuenta. Un ejemplo de una llamada a la misma sería el siguiente:

```
<CallCenterRequest>
  <AddTextToOrderByCheckId>
    <CheckId>506806165290</CheckId>
    <PrintDocument>
      <PrintCentered>Texto Centrado</PrintCentered>
      <PrintLeftRight><Left>Texto a la izquierda</Left><Right>Texto a la derecha</Right></PrintLeftRight>
    </PrintDocument>
  </AddTextToOrderByCheckId>
</CallCenterRequest>
```

Su resultado es un código de respuesta indicando si la función se ejecutó correctamente, tal y como se describió anteriormente en este documento.

Ejemplo de un llamado AddTextToOrder

Si por ejemplo enviamos el siguiente mensaje:

```
<CallCenterRequest>
  <AddTextToOrderById>
    <OrderId>506806165406</OrderId>
    <PrintDocument>
      <PRINTLINE>Print line sample</PRINTLINE>
      <LINEFEED>1</LINEFEED>
      <PRINTCENTERED>Print centered sample</PRINTCENTERED>
      <PRINTLEFTRIGHT><LEFT>Left</LEFT><RIGHT>Right</RIGHT></PRINTLEFTRIGHT>
      <PRINTLEFTCENTER><LEFT>Left</LEFT><CENTER>Center</CENTER></PRINTLEFTCENTER>
      <PRINTRIGHT><LEFT>Left</LEFT><RIGHT>Right</RIGHT></PRINTRIGHT>
      <PRINTLEFTCENTERRIGHT><LEFT>Left</LEFT><CENTER>Center</CENTER><RIGHT>Right</RIGHT></PRINTLEFTCENTERRIGHT>
      <PRINTFILLED>*</PRINTFILLED>
      <PRINTLEFTRIGHTEX><LEFT>Left</LEFT><RIGHT>_</RIGHT></PRINTLEFTRIGHTEX>
    </PrintDocument>
  </AddTextToOrderById>
</CallCenterRequest>
```

En el punto de venta se imprimirá el siguiente texto:

```
Print line sample
Print centered sample
Left                Center                Right
Left                Center                Right
Left                Center                Right
*****
Left                _____
```

Códigos de Respuesta

Código de Respuesta	Descripción
Success	Operación completada satisfactoriamente.
TrialPeriodExpired	El periodo de prueba de LASA CallCenter ha expirado
RequestFormatInvalid	La solicitud recibida no se encuentra en un formato valido
TakeOutServerOffline	La interfaz con el servidor de Aloha Take Out está fuera de línea. Espere un momento y vuelva a intentar.
UnsupportedRequest	La función solicitada no está soportada o esta no pudo ser identificada
ItemVoided	Artículo ha sido anulado
Failure	La operación encontró un error no especificado
ConfigError	Ocurrió un error al tratar de cargar la configuración de Aloha Take Out
NotInitialized	La interfaz de Aloha Take Out aun no ha sido inicializada
ServiceUnavailable	El servicio de TakeOut solicitado no está disponible
ServiceCallFailure	El servicio de llamada fallo al tratar de de completar la operación
ServiceCallException	El servicio de llamada lanzo una excepción. Examine el objeto ResultDetail para más detalle acerca de la excepción
CommunicationException	El servicio de llamada lanzo una excepción de comunicación. Examine el objeto ResultDetail para más detalle acerca de la excepción
InvalidServiceActivator	Service activator es nulo.
XmlParseFailure	Fallo al tratar de interpretar un XML
PartialListFailure	Un método de TakeOut interface que opera en una lista de objetos fallo en uno o más objetos. Examine la lista de objetos ResultDetail para determinar que objeto falló
ProtectionFailure	La operación fallo debido a que no fue posible proteger información sensitiva como por ejemplo un número de tarjeta de crédito
InvalidOrder	La orden especificada no existe
InvalidParameter	Alguno de los parámetros enviados a el método de la API es invalido
GeneralException	Un error general no manejado ocurrió al ejecutar el método. Examine el objeto ResultDetail para mas detalles acerca de la excepción
KeyGenerationFailure	Falla al generar la llave RSA para protección de datos
GuestValidationFailure	Falla al adicionar Nuevo cliente o validar un cliente existente
CheckCreationFailure	Falla al crear una cuenta para la orden en el sistema de punto de venta
OrderRemovalFailure	Falla al remover una orden del sistema de punto de venta. Esta puede ser una orden temporal que fue requerida para crear una orden futura.

FutureOrderCreationFailure	Falla al tratar de crear la orden futura en Aloha Take Out
NullParameterFailure	Un parámetro en nulo fue pasado a un método que no soporta valores en nulo
ObjectSerializeFailure	Falla al serializar un objeto de datos para transmisión
ObjectDeserializeFailure	Falla al des-serializar un objeto de datos
LoginFailure	Falla al iniciar sesión en el sistema de punto de venta
ClockInFailure	Falla al fichar entrada en el sistema de punto de venta
PosInterfaceLockFailure	Falla al tratar de ejecutar un bloqueo en el sistema de punto de venta
NotLicensed	La aplicación no cuenta con licencia para usar esta API
OrderNotFound	La orden especificada no existe
AlreadyOrdered	La orden ya fue enviada a la cocina
InvalidItem	Artículo no existe
InvalidModifier	Artículo modificador no existe
ModifierNotAllowed	Artículo modificador no permitido para este artículo
ModifierRequirementNotMet	No se han especificado todos los modificadores requeridos para el artículo
UnavailableItem	Este artículo no está disponible por el momento
FailedModifier	Indica que al menos un modificador fallo al tratar de aplicárselo al artículo padre
FailedParentItem	Indica que falló el registro del artículo padre asociado con el artículo modificador
NoHouseAccountOnFile	Indica que el cliente no tiene una cuenta de crédito asociada a su perfil
CustomerAlreadyExists	El cliente a ser adicionado ya existe en Aloha Take Out
CustomerDoesNotExist	El cliente especificado no existe en Aloha Take Out
CustomerHasPendingOrders	El cliente especificado tiene ordenes pendientes en Aloha Take Out
PosInterfaceNotFunctioning	La interfaz entre Aloha Take Out y el sistema de punto de venta no está funcionando actualmente
CheckNotFound	Indica que la cuenta asociada no existe en el punto de venta
PaymentVoidFailure	Falla al anular un pago en la cuenta
CheckIsClosed	La cuenta especificada se encuentra cerrada en el sistema de punto de venta
DriverDoesNotExist	El motorista especificado no se encuentra registrado en Aloha Take Out
NoTaxIdOnFile	Indica que el cliente no tiene un identificador fiscal asociado a su perfil
NoLoyaltyIdOnFile	Indica que el cliente no tiene un programa de lealtad asociado a su perfil
OrderCapacityExhausted	La capacidad de ordenes configurada para este periodo se ha agotado

ItemCapacityExhausted	La capacidad de artículos configurada para este periodo se ha agotado
InvalidOrderMode	El Modo de pedido especificado no es valido
UnknownException	Ocurrió un error inesperado que no pudo ser identificado. Consulte el historial de eventos de la aplicación para más detalles.
InvalidPrintXmlSchema	El esquema del XML de impresión tiene un esquema inválido. Consulte el historial de eventos de la aplicación para más detalles.
UnableToCreateXsdSchemaForValidation	Ocurrió un error tratando de crear el archivo XSD para validar el esquema del XML de impresión. Consulte el historial de eventos de la aplicación para más detalles.
UnableToCreateXmlFileForValidation	Ocurrió un error tratando de crear el archivo XML para validarlo contra el esquema XSD de impresión. Consulte el historial de eventos de la aplicación para más detalles.
UnableToAddPrintDocumentToCheck	Ocurrió un error tratando de vincular el XML recibido con la cuenta. Consulte el historial de eventos de la aplicación para más detalles.
UnableToCreateTextDocumentsFolder	Ocurrió un error tratando de crear la carpeta donde se almacenaran los archivos XML para impresión. Consulte el historial de eventos de la aplicación para más detalles.