**Proyecto ERM**

**Especificación del Caso de Uso: Generar Interfaces de Salida de Movimientos (Casa García) – CUERMINT148**

**VERSIÓN 1.0**

**HISTÓRICO DE REVISIÓN**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión** | **Descripción** | **Autor** |
| 27/11/2013 | 1.0 | Elaboración de la especificación del caso de uso Generar Interfaces de Salida de Movimientos para el Proyecto ERM de acuerdo al Estándar de Casos de Uso 1.3.  Folio CAI 0002855 | Belem Lizeth Jiménez Arévalo |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Tabla de Contenido

[Especificación de Casos de Uso: Generar Interfaces de Salida de Movimientos (Casa García) – CUERMINT148 4](#_Toc380502592)

[1 Introducción 4](#_Toc380502593)

[2 Caso de uso: Generar Interfaces de Salida de Movimientos (Casa García) – CUERMINT148 4](#_Toc380502594)

[2.1 Descripción 4](#_Toc380502595)

[3 Diagrama de Casos de Uso 4](#_Toc380502596)

[4 Precondiciones 4](#_Toc380502597)

[4.1 Generales 4](#_Toc380502598)

[4.2 Parámetros 5](#_Toc380502599)

[5 Flujo de eventos 5](#_Toc380502600)

[5.1 Flujo básico 5](#_Toc380502601)

[5.2 Flujos alternos 13](#_Toc380502602)

[5.2.1 Opcionales 13](#_Toc380502603)

[5.2.2 Generales 13](#_Toc380502604)

[5.2.3 De excepción 13](#_Toc380502605)

[6 Poscondiciones 13](#_Toc380502606)

[6.1 Generales 13](#_Toc380502607)

[7 Firmas de Aceptación 13](#_Toc380502608)

# Generar Interfaces de Salida de Movimientos (Casa García) – CUERMINT148

# Introducción

Este documento contiene la descripción de una secuencia de interacciones entre un actor y el sistema de tal manera que se puede obtener un resultado con valor particular para el actor dentro del flujo básico. También incluye el listado de las precondiciones y poscondiciones que se deben cumplir al inicio y al final del flujo básico además de los posibles flujos alternos y excepciones.

# Caso de uso: Generar Interfaces de Salida de Movimientos (Casa García) – CUERMINT148

## Descripción

Permite generar un conjunto de 12 archivos de texto de ancho fijo por cada ruta de Autoventa, con la información de las interfaces de salida para los movimientos de Liquidación, Cobranza, Envase y Detalle de Venta.

# Diagrama de Casos de Uso

**

# Precondiciones

## Generales

1. Debe estar configurado el directorio de interfaces para el sistema.
2. Se debe haber descargado la información de agenda para el vendedor en el día de trabajo.
3. Se considera como regla que para delimitar la longitud de los campo a un ancho fijo, se rellenará siempre con espacios vacíos a la izquierda.

## Parámetros

1. IP\_FechaInicio
2. IP\_FechaFin
3. IP\_ClaveUsuario

# Flujo de eventos

## Flujo básico

1. El caso de uso inicia cuando es invocado por el caso de uso [**Generar Interfaces Automáticas de Salida – CUERMINT01**](file:///C:\Amesol\Productos\Route\Trunk\Analisis\EspecificacionRequerimientos\ECU_INT\CI-051-01.doc)
2. El sistema obtiene y registra en sesión el Vendedor correspondiente al usuario para el que se realiza la descarga de información <Vendedor.VendedorID>, donde <Vendedor.USUId = Usuario.USUId> y <Usuario.Clave = IP\_ClaveUsuario>.
3. El sistema obtiene y registra en sesión la Ruta asignada al vendedor para el día de trabajo <VenRut.RUTClave>, donde <VenRut.VendedorID = VendedorID registrado en sesión>
4. El sistema obtiene y registra en sesión el Día de Trabajo correspondiente a la generación de las interfaces de salida <Dia.DiaClave>, donde <Dia.FechaCaptura = IP\_FechaInicio>.
5. El sistema obtiene la ruta del directorio de interfaces <CONHist.DirInterfaz> a partir de la configuración más reciente del histórico de parámetros de configuración, de acuerdo al tipo de lenguaje configurado para el sistema.
6. El sistema valida que exista la ruta obtenida y dentro de ella el directorio donde se depositarán los archivos txt de las interfaces de salida, el cual será llamado “Salida”.
   1. Si <no existe la ruta de los archivos txt de salida>
      1. El sistema crea el archivo BitacoraLog dentro del directorio donde se almacenan los archivos de las bases de datos del vendedor <CONHist.DirectorioSDF>, llevando por nombre el formato: ‘BitacoraLogRuta-yyyyMMddHHmmss.txt’ (de acuerdo a la fecha y hora actual del sistema) donde:
         1. Ruta = RUTClave (registrada en sesión)
         2. yyyy = Año
         3. MM = Mes
         4. dd = Día
         5. HH = Hora (formato 24 horas)
         6. mm= Minuto
         7. ss = Segundo.
      2. El sistema registra en el archivo BitacoraLog la siguiente información:
         1. “No se procesó la información del vendedor <VendedorID>. No existe la ruta del directorio de las interfaces de salida”
            1. El sistema continúa en el [**paso 1**](#FIN)**9** del flujo básico
7. **//Liquidación de la Ruta**
   1. El sistema crea un archivo de texto dentro del directorio de las interfaces de salida, el cual llevará por nombre ‘01R###DD.txt’ donde:
      1. ### = RUTClave (registrada en sesión)
      2. DD = Día enviado como parámetro <IP\_FechaInicio> (únicamente la parte correspondiente al día).
   2. El sistema escribe la siguiente información en el archivo de texto para la Liquidación de la Ruta *(el contenido de cada campo se delimita por la longitud especificada para el mismo)*:
      1. FoliLiqu = ‘L###DDMMAA’ *(Delimitar la longitud del campo a 10 caracteres)*, donde:

### = RUTClave (registrada en sesión)

DD = Día (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

MM = Mes (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

AA = Año (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

* + 1. Fecha = Fecha enviada como parámetro <IP\_FechaInicio>, en formato DD/MM/AAAA. *(Delimitar la longitud del campo a 10 caracteres)*.
    2. Ruta = RUTClave (registrada en sesión). *(Delimitar la longitud del campo a 3 caracteres)*.
  1. Si <ocurrió algún error durante el procesamiento del archivo txt de salida>, el sistema continúa con el flujo alterno de excepción **[AE01 Error al Procesar Interfaz de Salida](#AE01)**

1. **//Bonificaciones Pertenecientes a la Liquidación de la Ruta**
   1. El sistema crea un archivo de texto dentro del directorio de las interfaces de salida, el cual llevará por nombre ‘02R###DD.txt’ donde:
      1. ### = RUTClave (registrada en sesión)
      2. DD = Día enviado como parámetro <IP\_FechaInicio> (únicamente la parte correspondiente al día).
   2. El sistema obtiene el Importe <∑AbnTrp.Importe> de los abonos realizados en el día de trabajo actual <AbnTrp.ABNId = Abono.ABNId y Abono.VisitaClave = Visita.VisitaClave donde Visita.DiaClave = DiaClave registrado en sesión>, por el vendedor actual <AbnTrp.ABNId = Abono.ABNId y Abono.VisitaClave = Visita.VisitaClave donde Visita.VendedorID = VendedorID registrado en sesión>, sobre las ventas realizadas también en el día de trabajo actual <Abono.ABNId = AbnTrp.ABNId y AbnTrp.TransProdID = TransProd.TransProdID y TransProd.FechaCaptura = IP\_FechaInicio>; así como la Referencia de los detalles de dichos abonos <ABNDetalle.Referencia>, cuyo tipo de pago sea igual a Bonificación <ABNDetalle.TipoPago = 5> donde <ABNDetalle.ABNId = AbnTrp.ABNId>.
   3. Para cada Bonificación Perteneciente a la Liquidación de la Ruta obtenida
      1. El sistema escribe la siguiente información en el archivo de texto *(una línea por bonificación obtenida, el contenido de cada campo se delimita por la longitud especificada para el mismo)*:
         1. FoliLiqu = ‘L###DDMMAA’ *(Delimitar la longitud del campo a 10 caracteres)*, donde:

### = RUTClave (registrada en sesión)

DD = Día (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

MM = Mes (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

AA = Año (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

* + - 1. FoliBoni = <ABNDetalle.Referencia> *(Delimitar la longitud del campo a 10 caracteres)*.
      2. Importe = <∑AbnTrp.Importe> *(Convertir el campo a tipo Float (9,2). Delimitar la longitud del campo a 12 caracteres rellenando con espacios a la izquierda o derecha según corresponda).*
  1. Si <ocurrió algún error durante el procesamiento del archivo txt de salida>, el sistema continúa con el flujo alterno de excepción [**AE01 Error al Procesar Interfaz de Salida**](#AE01)

1. **//Créditos Pertenecientes a la Liquidación de la Ruta**
   1. El sistema crea un archivo de texto dentro del directorio de las interfaces de salida, el cual llevará por nombre ‘03R###DD.txt’ donde:
      1. ### = RUTClave (registrada en sesión)
      2. DD = Día enviado como parámetro <IP\_FechaInicio> (únicamente la parte correspondiente al día).
   2. El sistema obtiene el Cliente <TransProd.ClienteClave> y Saldo <TransProd.Saldo> de las ventas a crédito con saldo mayor a cero, no canceladas, realizadas en el día de trabajo actual <TransProd.TransProdID donde TransProd.CFVTipo = 2 y TransProd.Saldo > 0 y TransProd.TipoFase <> 0 y TransProd.FechaCaptura = IP\_FechaInicio>, por el vendedor actual <TransProd.VisitaClave = Visita.VisitaClave donde Visita.VendedorID = VendedorID registrado en sesión>.
   3. Para cada Cliente con Ventas a Crédito con Saldo y Realizadas en el Día de Trabajo
      1. El sistema escribe la siguiente información en el archivo de texto *(una línea por cliente obtenido, el contenido de cada campo se delimita por la longitud especificada para el mismo)*:
         1. FoliLiqu = ‘L###DDMMAA’ *(Delimitar la longitud del campo a 10 caracteres)*, donde:

### = RUTClave (registrada en sesión)

DD = Día (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

MM = Mes (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

AA = Año (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

* + - 1. ClavClie = <Cliente.Clave> donde <Cliente.ClienteClave = TransProd.ClienteClave> *(Delimitar la longitud del campo a 7 caracteres)*.
      2. Importe = Sumatoria del saldo (∑TransProd.Saldo) de las ventas a crédito realizadas al cliente en el día de trabajo actual. *(Convertir el campo a tipo Float (12,2). Delimitar la longitud del campo a 15 caracteres rellenando con espacios a la izquierda o derecha según corresponda).*
  1. Si <ocurrió algún error durante el procesamiento del archivo txt de salida>, el sistema continúa con el flujo alterno de excepción [**AE01 Error al Procesar Interfaz de Salida**](#AE01)

1. **//Cheques Pertenecientes a la Liquidación de la Ruta**
   1. El sistema crea un archivo de texto dentro del directorio de las interfaces de salida, el cual llevará por nombre ‘04R###DD.txt’ donde:
      1. ### = RUTClave (registrada en sesión)
      2. DD = Día enviado como parámetro <IP\_FechaInicio> (únicamente la parte correspondiente al día).
   2. El sistema obtiene el Importe <AbnTrp.Importe> de los abonos realizados en el día de trabajo actual <AbnTrp.ABNId = Abono.ABNId y Abono.VisitaClave = Visita.VisitaClave donde Visita.DiaClave = DiaClave registrado en sesión>, por el vendedor actual <AbnTrp.ABNId = Abono.ABNId y Abono.VisitaClave = Visita.VisitaClave donde Visita.VendedorID = VendedorID registrado en sesión>, sobre las ventas realizadas también en el día de trabajo actual <Abono.ABNId = AbnTrp.ABNId y AbnTrp.TransProdID = TransProd.TransProdID y TransProd.FechaCaptura = IP\_FechaInicio>; así como la Referencia <ABNDetalle.Referencia> y el Banco <ABNDetalle.TipoBanco> de los detalles de dichos abonos, cuyo tipo de pago sea igual a Cheque <ABNDetalle.TipoPago = 2> donde <ABNDetalle.ABNId = AbnTrp.ABNId>.
   3. Para cada Cheque Perteneciente a la Liquidación de la Ruta obtenido
      1. El sistema escribe la siguiente información en el archivo de texto *(una línea por cheque obtenido, el contenido de cada campo se delimita por la longitud especificada para el mismo)*:
         1. FoliLiqu = ‘L###DDMMAA’ *(Delimitar la longitud del campo a 10 caracteres)*, donde:

### = RUTClave (registrada en sesión)

DD = Día (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

MM = Mes (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

AA = Año (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

* + - 1. ClavClie = <Cliente.Clave> donde <Cliente.ClienteClave = Visita.ClienteClave y Visita.VisitaClave = Abono.VisitaClave y Abono.ABNId = AbnTrp.ABNId>. *(Delimitar la longitud del campo a 7 caracteres)*.
      2. NumeCheq = <ABNDetalle.Referencia>, enviar únicamente los cinco últimos dígitos del cheque. *(Delimitar la longitud del campo a 10 caracteres)*.
      3. ClavBanc = <ABNDetalle.TipoBanco>. *(Delimitar la longitud del campo a 2 caracteres)*.
      4. Importe = <∑AbnTrp.Importe> donde <AbnTrp.ABNId = Abono.ABNId y Abono.VisitaClave = Visita.VisitaClave donde Visita.ClienteClave = Cliente.ClienteClave y Cliente.Clave = ‘ClavClie’ actual> *(Convertir el campo a tipo Float (10,2). Delimitar la longitud del campo a 13 caracteres rellenando con espacios a la izquierda o derecha según corresponda)*.
  1. Si <ocurrió algún error durante el procesamiento del archivo txt de salida>, el sistema continúa con el flujo alterno de excepción [**AE01 Error al Procesar Interfaz de Salida**](#AE01)

1. **//Efectivo Perteneciente a la Liquidación de la Ruta**
   1. El sistema crea un archivo de texto dentro del directorio de las interfaces de salida, el cual llevará por nombre ‘05R###DD.txt’ donde:
      1. ### = RUTClave (registrada en sesión)
      2. DD = Día enviado como parámetro <IP\_FechaInicio> (únicamente la parte correspondiente al día).
   2. El sistema obtiene la sumatoria del Importe <∑AbnTrp.Importe> de los abonos cuyo tipo de pago sea igual a Efectivo <ABNDetalle.TipoPago = 1> donde <ABNDetalle.ABNId = AbnTrp.ABNId, realizados en el día de trabajo actual <AbnTrp.ABNId = Abono.ABNId y Abono.VisitaClave = Visita.VisitaClave donde Visita.DiaClave = DiaClave registrado en sesión>, por el vendedor actual <AbnTrp.ABNId = Abono.ABNId y Abono.VisitaClave = Visita.VisitaClave donde Visita.VendedorID = VendedorID registrado en sesión>, sobre las ventas realizadas también en el día de trabajo actual <Abono.ABNId = AbnTrp.ABNId y AbnTrp.TransProdID = TransProd.TransProdID y TransProd.FechaCaptura = IP\_FechaInicio>.
   3. El sistema escribe la siguiente información en el archivo de texto para el Efectivo Perteneciente a la Liquidación de la Ruta *(el contenido de cada campo se delimita por la longitud especificada para el mismo)*:
      1. FoliLiqu = ‘L###DDMMAA’ *(Delimitar la longitud del campo a 10 caracteres)*, donde:

### = RUTClave (registrada en sesión)

DD = Día (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

MM = Mes (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

AA = Año (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

* + 1. Importe = Sumatoria del importe <∑AbnTrp.Importe> de los abonos en efectivo realizados en el día de trabajo sobre las ventas realizadas también en el día de trabajo actual. *(Convertir el campo a tipo Float (12,2). Delimitar la longitud del campo a 15 caracteres rellenando con espacios a la izquierda o derecha según corresponda).*
  1. Si <ocurrió algún error durante el procesamiento del archivo txt de salida>, el sistema continúa con el flujo alterno de excepción [**AE01 Error al Procesar Interfaz de Salida**](#AE01)

1. **//Cobranza de la Ruta**
   1. El sistema crea un archivo de texto dentro del directorio de las interfaces de salida, el cual llevará por nombre ‘06R###DD.txt’ donde:
      1. ### = RUTClave (registrada en sesión)
      2. DD = Día enviado como parámetro <IP\_FechaInicio> (únicamente la parte correspondiente al día).
   2. El sistema obtiene el Cliente <TransProd.ClienteClave> de las ventas a crédito que no se hayan realizado en el día de trabajo, a las que se les hayan realizado abonos en el día de trabajo actual, por el vendedor actual <TransProd.TransProdID = AbnTrp.TransProdID y TransProd.CFVTipo = 2 y TransProd.FechaCaptura <> IP\_FechaInicio, donde AbnTrp.ABNId = Abono.ABNId y Abono.VisitaClave = Visita.VisitaClave donde Visita.DiaClave = DiaClave registrado en sesión y Visita.VendedorID = VendedorID registrado en sesión>.
   3. Para cada Cliente con Ventas a Crédito No Realizadas en el Día de Trabajo Actual obtenidas, a las que se les hayan Realizado Abonos en el Día de Trabajo
      1. El sistema escribe la siguiente información en el archivo de texto *(una línea por cliente obtenido, el contenido de cada campo se delimita por la longitud especificada para el mismo)*:
         1. FoliCobr = ‘CClienteDDMMAA’ *(Delimitar la longitud del campo a 14 caracteres)*, donde:

Cliente = <Cliente.Clave> donde <Cliente.ClienteClave = TransProd.ClienteClave>.

DD = Día (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

MM = Mes (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

AA = Año (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

* + - 1. Fecha = Fecha enviada como parámetro <IP\_FechaInicio>, en formato DD/MM/AAAA. *(Delimitar la longitud del campo a 10 caracteres)*.
      2. Ruta = RUTClave (registrada en sesión). *(Delimitar la longitud del campo a 3 caracteres)*.
      3. ClavClie = <Cliente.Clave> donde <Cliente.ClienteClave = TransProd.ClienteClave> *(Delimitar la longitud del campo a 7 caracteres)*.
  1. Si <ocurrió algún error durante el procesamiento del archivo txt de salida>, el sistema continúa con el flujo alterno de excepción [**AE01 Error al Procesar Interfaz de Salida**](#AE01)

1. **//Bonificaciones Pertenecientes a la Cobranza de la Ruta**
   1. El sistema crea un archivo de texto dentro del directorio de las interfaces de salida, el cual llevará por nombre ‘07R###DD.txt’ donde:
      1. ### = RUTClave (registrada en sesión)
      2. DD = Día enviado como parámetro <IP\_FechaInicio> (únicamente la parte correspondiente al día).
   2. El sistema obtiene el Importe <AbnTrp.Importe> de los abonos realizados en el día de trabajo actual <AbnTrp.ABNId = Abono.ABNId y Abono.VisitaClave = Visita.VisitaClave donde Visita.DiaClave = DiaClave registrado en sesión>, por el vendedor actual <AbnTrp.ABNId = Abono.ABNId y Abono.VisitaClave = Visita.VisitaClave donde Visita.VendedorID = VendedorID registrado en sesión>, sobre las ventas a crédito que no se hayan realizado en el día de trabajo <Abono.ABNId = AbnTrp.ABNId y AbnTrp.TransProdID = TransProd.TransProdID y TransProd.CFVTipo = 2 y TransProd.FechaCaptura <> IP\_FechaInicio>; así como la Referencia de los detalles de dichos abonos <ABNDetalle.Referencia>, cuyo tipo de pago sea igual a Bonificación <ABNDetalle.TipoPago = 5> donde <ABNDetalle.ABNId = AbnTrp.ABNId>.
   3. Para cada Bonificación Perteneciente a la Cobranza de la Ruta obtenida
      1. El sistema escribe la siguiente información en el archivo de texto *(una línea por bonificación obtenida, el contenido de cada campo se delimita por la longitud especificada para el mismo)*:
         1. FoliCobr = ‘CClienteDDMMAA’ *(Delimitar la longitud del campo a 14 caracteres)*, donde:

Cliente = <Cliente.Clave> donde <Cliente.ClienteClave = Visita.ClienteClave y Visita.VisitaClave = Abono.VisitaClave y Abono.ABNId = AbnTrp.ABNId>.

DD = Día (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

MM = Mes (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

AA = Año (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

* + - 1. FoliBoni = <ABNDetalle.Referencia> *(Delimitar la longitud del campo a 10 caracteres)*.
      2. Importe = <∑AbnTrp.Importe> donde <AbnTrp.ABNId = Abono.ABNId y Abono.VisitaClave = Visita.VisitaClave donde Visita.ClienteClave = Cliente.ClienteClave y Cliente.Clave = Cliente incluido en el ‘FoliCobr’> *(Convertir el campo a tipo Float (9,2). Delimitar la longitud del campo a 12 caracteres rellenando con espacios a la izquierda o derecha según corresponda).*
  1. Si <ocurrió algún error durante el procesamiento del archivo txt de salida>, el sistema continúa con el flujo alterno de excepción [**AE01 Error al Procesar Interfaz de Salida**](#AE01)

1. **//Cheques Pertenecientes a la Cobranza de la Ruta**
   1. El sistema crea un archivo de texto dentro del directorio de las interfaces de salida, el cual llevará por nombre ‘08R###DD.txt’ donde:
      1. ### = RUTClave (registrada en sesión)
      2. DD = Día enviado como parámetro <IP\_FechaInicio> (únicamente la parte correspondiente al día).
   2. El sistema obtiene el Importe <AbnTrp.Importe> de los abonos realizados en el día de trabajo actual <AbnTrp.ABNId = Abono.ABNId y Abono.VisitaClave = Visita.VisitaClave donde Visita.DiaClave = DiaClave registrado en sesión>, por el vendedor actual <AbnTrp.ABNId = Abono.ABNId y Abono.VisitaClave = Visita.VisitaClave donde Visita.VendedorID = VendedorID registrado en sesión>, sobre las ventas a crédito que no se hayan realizado en el día de trabajo <Abono.ABNId = AbnTrp.ABNId y AbnTrp.TransProdID = TransProd.TransProdID y TransProd.CFVTipo = 2 y TransProd.FechaCaptura <> IP\_FechaInicio>; así como la Referencia <ABNDetalle.Referencia> y el Banco <ABNDetalle.TipoBanco> de los detalles de dichos abonos, cuyo tipo de pago sea igual a Cheque <ABNDetalle.TipoPago = 2> donde <ABNDetalle.ABNId = AbnTrp.ABNId>.
   3. Para cada Cheque Perteneciente a la Cobranza de la Ruta obtenido
      1. El sistema escribe la siguiente información en el archivo de texto *(una línea por cheque obtenido, el contenido de cada campo se delimita por la longitud especificada para el mismo)*:
         1. FoliCobr = ‘CClienteDDMMAA’ *(Delimitar la longitud del campo a 14 caracteres)*, donde:

Cliente = <Cliente.Clave> donde <Cliente.ClienteClave = Visita.ClienteClave y Visita.VisitaClave = Abono.VisitaClave y Abono.ABNId = AbnTrp.ABNId>.

DD = Día (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

MM = Mes (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

AA = Año (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

* + - 1. NumeCheq = <ABNDetalle.Referencia>, enviar únicamente los cinco últimos dígitos del cheque. *(Delimitar la longitud del campo a 10 caracteres)*.
      2. ClavBanc = <ABNDetalle.TipoBanco>. *(Delimitar la longitud del campo a 2 caracteres)*.
      3. Importe = <∑AbnTrp.Importe> donde <AbnTrp.ABNId = Abono.ABNId y Abono.VisitaClave = Visita.VisitaClave donde Visita.ClienteClave = Cliente.ClienteClave y Cliente.Clave = Cliente incluido en el ‘FoliCobr’> *(Convertir el campo a tipo Float (10,2). Delimitar la longitud del campo a 13 caracteres rellenando con espacios a la izquierda o derecha según corresponda)*.
  1. Si <ocurrió algún error durante el procesamiento del archivo txt de salida>, el sistema continúa con el flujo alterno de excepción [**AE01 Error al Procesar Interfaz de Salida**](#AE01)

1. **//Efectivo Perteneciente a la Cobranza de la Ruta**
   1. El sistema crea un archivo de texto dentro del directorio de las interfaces de salida, el cual llevará por nombre ‘09R###DD.txt’ donde:
      1. ### = RUTClave (registrada en sesión)
      2. DD = Día enviado como parámetro <IP\_FechaInicio> (únicamente la parte correspondiente al día).
   2. El sistema obtiene el Importe <AbnTrp.Importe> de los abonos cuyo tipo de pago sea igual a Efectivo <ABNDetalle.TipoPago = 1> donde <ABNDetalle.ABNId = AbnTrp.ABNId, realizados en el día de trabajo actual <AbnTrp.ABNId = Abono.ABNId y Abono.VisitaClave = Visita.VisitaClave donde Visita.DiaClave = DiaClave registrado en sesión>, por el vendedor actual <AbnTrp.ABNId = Abono.ABNId y Abono.VisitaClave = Visita.VisitaClave donde Visita.VendedorID = VendedorID registrado en sesión>, sobre las ventas a crédito que no se hayan realizado en el día de trabajo <Abono.ABNId = AbnTrp.ABNId y AbnTrp.TransProdID = TransProd.TransProdID y TransProd.CFVTipo = 2 y TransProd.FechaCaptura <> IP\_FechaInicio>.
   3. El sistema escribe la siguiente información en el archivo de texto para el Efectivo Perteneciente a la Cobranza de la Ruta *(el contenido de cada campo se delimita por la longitud especificada para el mismo)*:
      1. FoliCobr = ‘CClienteDDMMAA’ *(Delimitar la longitud del campo a 14 caracteres)*, donde:

Cliente = <Cliente.Clave> donde <Cliente.ClienteClave = Visita.ClienteClave y Visita.VisitaClave = Abono.VisitaClave y Abono.ABNId = AbnTrp.ABNId>.

DD = Día (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

MM = Mes (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

AA = Año (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

* + 1. Importe = Sumatoria del importe <∑AbnTrp.Importe> de los abonos en efectivo realizados en el día de trabajo, sobre las ventas a crédito que no se hayan realizado en el día de trabajo actual, para el cliente actual <AbnTrp.ABNId = Abono.ABNId y Abono.VisitaClave = Visita.VisitaClave donde Visita.ClienteClave = Cliente.ClienteClave y Cliente.Clave = Cliente incluido en el ‘FoliCobr’> *(Convertir el campo a tipo Float (12,2). Delimitar la longitud del campo a 15 caracteres rellenando con espacios a la izquierda o derecha según corresponda).*
  1. Si <ocurrió algún error durante el procesamiento del archivo txt de salida>, el sistema continúa con el flujo alterno de excepción [**AE01 Error al Procesar Interfaz de Salida**](#AE01)

1. **//Envase de la Ruta**
   1. El sistema crea un archivo de texto dentro del directorio de las interfaces de salida, el cual llevará por nombre ‘10R###DD.txt’ donde:
      1. ### = RUTClave (registrada en sesión)
      2. DD = Día enviado como parámetro <IP\_FechaInicio> (únicamente la parte correspondiente al día).
   2. El sistema obtiene el Identificador <TransProd.TransProdID> y el Cliente <TransProd.ClienteClave> de las devoluciones de cliente no canceladas realizadas en el día de trabajo actual, por el vendedor actual <TransProd.Tipo = 3 y TransProd.TipoFase <> 0 y TransProd.FechaCaptura = IP\_FechaInicio donde TransProd.VisitaClave = Visita.VisitaClave y Visita.VendedorID = VendedorID registrado en sesión>.
   3. Para cada devolución de cliente obtenida
      1. El sistema obtiene el Producto <TransProdDetalle.ProductoClave> y la Cantidad <TransProdDetalle.Cantidad> de sus detalles correspondientes <TransProdDetalle.TransProdDetalleID donde TransProdDetalle.TransProdID = TransProd.TransProdID de la devolución>.
   4. Para cada devolución de cliente obtenida
      1. Para cada detalle de la devolución obtenido
         1. El sistema escribe la siguiente información en el archivo de texto *(una línea por detalle obtenido, el contenido de cada campo se delimita por la longitud especificada para el mismo)*:
            1. Si <existe por lo menos una venta no cancelada realizada en la misma visita en que se realizó la devolución de cliente <TransProd.Tipo = 1 y TransProd.TipoFase <> 0 donde TransProd.VisitaClave = TransProd.VisitaClave (de la devolución)>>

FoliTick = <TransProd.Folio (de la primer venta no cancelada obtenida)>. *(Delimitar la longitud del campo a 12 caracteres)*.

* + - * 1. FoliTick = (Cadena vacía). *(Delimitar la longitud del campo a 12 caracteres)*.
        2. Ruta = RUTClave (registrada en sesión). *(Delimitar la longitud del campo a 3 caracteres)*.
        3. Fecha = Fecha enviada como parámetro <IP\_FechaInicio>, en formato DD/MM/AAAA. *(Delimitar la longitud del campo a 10 caracteres)*.
        4. ClavClie = <Cliente.Clave> donde <Cliente.ClienteClave = TransProd.ClienteClave (de la devolución)>. *(Delimitar la longitud del campo a 7 caracteres)*.
        5. ClavPres = <TransProdDetalle.ProductoClave (de la devolución)>. *(Delimitar la longitud del campo a 5 caracteres)*.
        6. Envase = <TransProdDetalle.Cantidad (de la devolución)>. *(Delimitar la longitud del campo a 5 caracteres)*.
  1. Si <ocurrió algún error durante el procesamiento del archivo txt de salida>, el sistema continúa con el flujo alterno de excepción [**AE01 Error al Procesar Interfaz de Salida**](#AE01)

1. **//Venta Clientes Detalle de la Ruta**
   1. El sistema crea un archivo de texto dentro del directorio de las interfaces de salida, el cual llevará por nombre ‘11R###DD.txt’ donde:
      1. ### = RUTClave (registrada en sesión)
      2. DD = Día enviado como parámetro <IP\_FechaInicio> (únicamente la parte correspondiente al día).
   2. El sistema obtiene el Identificador <TransProd.TransProdID> y el Folio <TransProd.Folio> de las ventas no canceladas realizadas en el día de trabajo actual, por el vendedor actual <TransProd.Tipo = 1 y TransProd.TipoFase <> 0 y TransProd.FechaCaptura = IP\_FechaInicio donde TransProd.VisitaClave = Visita.VisitaClave y Visita.VendedorID = VendedorID registrado en sesión>.
   3. Para cada venta obtenida
      1. El sistema obtiene el Producto <TransProdDetalle.ProductoClave>, la Cantidad <TransProdDetalle.Cantidad> y el Precio <TransProdDetalle.Precio> de sus detalles correspondientes <TransProdDetalle.TransProdDetalleID donde TransProdDetalle.TransProdID = TransProd.TransProdID de la venta>.
   4. Para cada venta obtenida
      1. Para cada detalle de la venta obtenido
         1. El sistema escribe la siguiente información en el archivo de texto *(una línea por detalle obtenido, el contenido de cada campo se delimita por la longitud especificada para el mismo)*:
            1. Ruta = RUTClave (registrada en sesión). *(Delimitar la longitud del campo a 3 caracteres)*.
            2. Fecha = Fecha enviada como parámetro <IP\_FechaInicio>, en formato DD/MM/AAAA. *(Delimitar la longitud del campo a 10 caracteres)*.
            3. FoliLiqu = ‘L###DDMMAA’ *(Delimitar la longitud del campo a 10 caracteres)*, donde:

### = RUTClave (registrada en sesión)

DD = Día (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

MM = Mes (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

AA = Año (de acuerdo a la fecha enviada como parámetro <IP\_FechaInicio>).

* + - * 1. FoliTick = <TransProd.Folio (de la venta)>. *(Delimitar la longitud del campo a 12 caracteres)*.
        2. ClavProd = <TransProdDetalle.ProductoClave>. *(Delimitar la longitud del campo a 5 caracteres)*.
        3. Venta = <TransProdDetalle.Cantidad>. *(Delimitar la longitud del campo a 5 caracteres)*.
        4. Importe = <TransProdDetalle.Cantidad \* TransProdDetalle.Precio>. *(Convertir el campo a tipo Float (12,2). Delimitar la longitud del campo a 15 caracteres rellenando con espacios a la izquierda o derecha según corresponda).*
  1. Si <ocurrió algún error durante el procesamiento del archivo txt de salida>, el sistema continúa con el flujo alterno de excepción [**AE01 Error al Procesar Interfaz de Salida**](#AE01)

1. **//Itinerario de la Ruta**
   1. El sistema crea un archivo de texto dentro del directorio de las interfaces de salida, el cual llevará por nombre ‘12R###DD.txt’ donde:
      1. ### = RUTClave (registrada en sesión)
      2. DD = Día enviado como parámetro <IP\_FechaInicio> (únicamente la parte correspondiente al día).
   2. El sistema obtiene el Cliente <Visita.ClienteClave>, la Fecha y Hora de Inicio <Visita.FechaHoraInicial> y la Fecha y Hora de Finalización <Visita.FechaHoraFinal> de las visitas realizadas en el día de trabajo actual, por el vendedor actual <Visita.DiaClave = DiaClave registrado en sesión y Visita.VendedorID = VendedorID registrado en sesión>.
   3. Para cada visita obtenida
      1. El sistema escribe la siguiente información en el archivo de texto *(una línea por visita obtenida, el contenido de cada campo se delimita por la longitud especificada para el mismo)*:
         1. ClavClie = <Cliente.Clave> donde <Cliente.ClienteClave = Visita.ClienteClave>. *(Delimitar la longitud del campo a 7 caracteres)*.
         2. Fecha = Fecha enviada como parámetro <IP\_FechaInicio>, en formato DD/MM/AAAA. *(Delimitar la longitud del campo a 10 caracteres)*.
         3. HoraInic = <Visita.FechaHoraInicial (sólo hora)>. *(Delimitar la longitud del campo a 5 caracteres, ejemplo: ‘99:99’)*.
         4. HoraFina = <Visita.FechaHoraFinal (sólo hora)>. *(Delimitar la longitud del campo a 5 caracteres, ejemplo: ‘99:99’)*.
         5. Duración = <Visita.FechaHoraFinal – Visita.FechaHoraInicial (sólo hora)>. *(Delimitar la longitud del campo a 5 caracteres, ejemplo: ‘99:99’)*.
         6. Si <existe una improductividad de venta asociada a la visita <ImproductividadVenta.VisitaClave = Visita.VisitaClave (visita actual)>>
            1. ClavActiv = <ImproductividadVenta.TipoMotivo>. *(Delimitar la longitud del campo a 3 caracteres)*.
         7. ClavActiv = ‘1’. *(Delimitar la longitud del campo a 3 caracteres)*.
   4. Si <ocurrió algún error durante el procesamiento del archivo txt de salida>, el sistema continúa con el flujo alterno de excepción [**AE01 Error al Procesar Interfaz de Salida**](#AE01)
2. El sistema obtiene las ventas no canceladas con saldo mayor a cero, realizadas por el vendedor actual <TransProd.TransProdID donde TransProd.Tipo = 1 y TransProd.TipoFase <> 0 y TransProd.Saldo > 0 y TransProd.VisitaClave = Visita.VisitaClave donde Visita.VendedorID = VendedorID registrado en sesión>
   1. Para cada venta obtenida
      1. El sistema actualiza el Saldo del documento <TransProd.Saldo = ‘0’>
3. Finaliza el caso de uso

## Flujos alternos

### Opcionales

N/A

### Generales

N/A

### De excepción

**5.2.3.1** **[AE01 Error al Procesar Interfaz de Salida](#AE01_r)**

1. El sistema crea el archivo BitacoraLog dentro del directorio de las interfaces de salida, llevando por nombre el formato: ‘BitacoraLogRuta-yyyyMMddHHmmss.txt’ (de acuerdo a la fecha y hora actual del sistema) donde:
   1. Ruta = RUTClave (registrada en sesión)
   2. yyyy = Año
   3. MM = Mes
   4. dd = Día
   5. HH = Hora (formato 24 horas)
   6. mm= Minuto
   7. ss = Segundo.
2. El sistema registra en el archivo BitacoraLog la siguiente información:
   1. “La información del archivo <NombreArchivoTxt> para el vendedor <VendedorID> no fue procesada. <Mensaje excepción>”
   2. El sistema continúa en el [**paso 1**](#Fin)**9** del flujo básico

# Poscondiciones

## Generales

1. Generación de los siguientes archivos txt con las interfaces de salida para la ruta:
   1. 01R###DD.txt
   2. 02R###DD.txt
   3. 03R###DD.txt
   4. 04R###DD.txt
   5. 05R###DD.txt
   6. 06R###DD.txt
   7. 07R###DD.txt
   8. 08R###DD.txt
   9. 09R###DD.txt
   10. 10R###DD.txt
   11. 11R###DD.txt
   12. 12R###DD.txt

# Firmas de Aceptación

|  |
| --- |
|  |
| José María Alcalá |
| Gerente de Ingeniería / Departamento de Ingeniería – Duxstar Solutions |
| 15/12/2013 |

|  |
| --- |
|  |
| Belem Lizeth Jiménez Arévalo |
| Analista de Sistemas / Departamento de Análisis y Diseño – Duxstar Solutions |
| 15/12/2013 |

|  |
| --- |
|  |
| José Manuel Gallegos |
| Líder de Proyecto / Departamento de Implementación - Duxstar Solutions |
| 15/12/2013 |