**Proyecto Route Lite (ROL)**

Especificación del Caso de Uso: Capturar Ventas a Consignación – CUROLMOV44

VERSIÓN 1.13

**HISTÓRICO DE REVISIÓN**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión** | **Descripción** | **Autor** |
| 14/04/2014 | 1.0 | Elaboración de la especificación del caso de uso Capturar Ventas a Consignación para el Proyecto Route Lite de acuerdo al Estándar de Casos de Uso 1.4.  Folio CAI 0003016  (Route Lite, 1.1.0.0) | Belem Lizeth Jiménez Arévalo |
| 30/05/2014 | 1.1 | Agregar validación en la Captura de Producto de las siguientes actividades para que no se permita capturar producto de tipo contenido que no se pueda vender por separado:   * Capturar Pedido * Capturar Movimiento sin Inventario Fuera de Visita * Capturar Ventas a Consignación * Administrar Cambios de Producto * Buscar Productos   Folio CAI 0003106  (Route Lite, 1.1.0.0) | Belem Lizeth Jiménez Arévalo |
| 24/06/2014 | 1.2 | 1. Agregar funcionalidad en la actividad de Capturar Ventas a Consignación para realizar el arrastre del saldo del préstamo de producto cuando se le puede prestar producto al cliente <Cliente.Prestamo = 1>. Realizar el arrastre de envase (Cargo) al agregar, modificar y eliminar un producto en los escenarios de Creación, Modificación y Eliminación de la Venta a Consignación, así como en el escenario de Liquidación a partir del producto devuelto (Abono). 2. Si al cliente se le puede realizar el préstamo de envase <Cliente.Prestamo = 1> y si el parámetro de configuración indica que al cliente se le debe validar el límite de préstamo de envase <Cliente.ValidarLimEnvase = 1>:  * Agregar funcionalidad para la validación del límite de envase en el escenario de creación y modificación de la Venta a Consignación, para asegurar que no se exceda el límite de préstamo de envase <Cliente.LimiteEnvase> con el producto vendido. Lanzar mensaje de advertencia de que se está llegando al límite del préstamo de envase cuando se cumpla el porcentaje de envase configurado para dicho fin <MOTConfiguracion.PorLimEnvase>.   Folio CAI 0003123  (Route Lite, 1.1.0.0) | Belem Lizeth Jiménez Arévalo |
| 30/10/2014 | 1.3 | Se requiere agregar a ProductoUnidad.TipoEstado un valor por referencia que indique que está activo y se suba al móvil, pero que no se pueda vender. Esto para poder obtener las equivalencias en cajas de los productos.  Se modifica en el e-Route Android las consultas que hacen referencia a ProductoUnidad para que no muestren los registros con ProductoUnidad.TipoEstado = 2 y sólo se considere para obtener el factor de equivalencia en cajas en el reporte de inventario de Costeña.  Folio CAI 0003300  (La Costeña, 1.1.0.0) | Belem Lizeth Jiménez Arévalo |
| 22/05/2015 | 1.4 | Folio CAI: 0003568  Realizar el cambio correspondiente para considerar los valores del valor por referencia "TMENSAJE", consideran solo el cero "0 = Ninguno" y el "1 = Impresión de Ticket" e ignorar los demás valores.  **Se realiza en la versión 1.4.0.0 (Route Lite)**  (Ayari, 1.4.0.0) | Nancy Elizabeth Villalobos Plascencia |
| 21/03/2016 | 1.5 | Folio CAI: 0003896  Habilitar mensaje para impresión bajo demanda en aplicación móvil (Android)  (Materias Primas, 1.6.0.0) | Nancy Elizabeth Villalobos Plascencia |
| 12/08/2016 | 1.6 | Mandar llamar la búsqueda simplificada de acuerdo al parámetro <MOTConfiguracion.ManejoDobleUnidad>. (Se documenta funcionalidad de la búsqueda simplificada).  Folio CAI 0003971  (QDL, 1.9.0.0 – 4.14.0.0) | Belem Lizeth Jiménez Arévalo |
| 23/08/2016 | 1.7 | Folio CAI: 0003967  \*Se agrega un nuevo valor a la tabla de ConfigParametro: -OrdenCapturaUnidad \*En el campo Valor se pondrán los números del valor por Referencia ordenados y separados por coma. \*En la pantalla de Captura se mostrará el orden de acuerdo a este valor \*ESTE PARAMETRO SOLO SE VALIDARÁ SI SE ENCUENTRA ACTIVO EL PARAMETRO <MOTConfiguración.ManejoDobleUnidad>  Folio CAI: 0003969  \*Se modificará la captura de Producto en las siguientes actividades: -Carga de Producto -Descarga -Devolución al Almacén -MSIFV -Ajustes -Pedido/MSIV -Cambios -Consignas -Devolución de Cliente \*Se tomaría como referencia si el producto tiene una unidad alterna, para desaparecer la captura de la unidad y agregar un nuevo campo para capturar ambas unidades del producto. \*Por default al capturar las piezas realizar el cálculo de los Kilogramos correspondientes y permitir realizarle modificaciones (Solo cuando es la doble Unidad) \*Nuevo estado que nos indique que tenemos que capturar una unidad Alterna  Folio CAI: 0003968  \*Modificar el ListView para mostrar la información de las dos unidades capturadas.  **Se realiza en la versión 4.14.0.0 (Route) y 1.9.0.0 (Android)**  (QDL, 1.9.0.0) | Nancy Elizabeth Villalobos Plascencia |
| 13/10/2016 | 1.8 | - Se utilizará el parámetro "TipoClaveProducto" que se encuentra dentro de CONHist para saber si se muestra el teclado numérico o el alfanumérico.  - Se utilizará el parámetro "DigitoClaveProd" que se encuentra dentro de CONHist para completar con ceros a la izquierda las claves de Productos capturadas.  NOTA: Al utilizar esta configuración, no se podrá realizar ningún tipo de búsqueda parcial dentro de la pantalla de captura, para la búsqueda se tendrá que utilizar el icono de "Lupa" y la búsqueda la hará con el teclado alfanumérico.  **Se realiza en la versión 1.7.5.0 y se replica a partir de la 1.9.0.0**  Folio CAI 0004028  (Lechera GDL, 1.7.5.0 - 4.12.5.0) | Belem Lizeth Jiménez Arévalo |
| 08/11/2016 | 1.9 | Modificar todas las actividades de TransProd en el móvil para validar sus permisos de usuario configurados, y en caso de que cuenten con el permiso de firma "S" (Sign), soliciten la firma del usuario al dar Continuar antes de imprimir el ticket, mandando llamar el componente de firma creado:  - Pedido y Mov sin Inv. en Visita (es la misma actividad)  - Cambios de Producto  - Devoluciones  - Mov Sin Inv Fuera de Visita  - Ajustes  - Descargas  - Devoluciones al Almacén  - Cargas  - Consignas  Folio CAI 0004044  (eRoute, 1.9.0.0 – 4.14.0.0) | Belem Lizeth Jiménez Arévalo |
| 22/12/2016 | 1.10 | - Realizar los cambios necesarios para permitir realizar el envío del ticket en formato PDF para las actividades que se realizan dentro de la visita en el móvil.  - El envío se realiza al cliente ya sea por medio de correo electrónico o SMS (dependiendo la configuración), a partir de sus datos registrados en el sistema. En caso de que el cliente no cuente con un correo o teléfono registrados, o bien desee actualizarlos, podrán capturarse en el momento.  - La configuración del tipo de Envío será por medio del parámetro <MOTConfiguracion.MensajeImpresion>, agregando un nuevo valor para el envío por SMS (el valor para Envío por Correo ya existe).  Folio CAI 0004073  (eRoute, 1.10.0.0 – 4.15.0.0) | Belem Lizeth Jiménez Arévalo |
| 24/01/2018 | 1.11 | Folio CAI: 0004658  Considerar solicitar un campo como obligatorio para la captura de "Folio goma", la propuesta es utilizar el mismo parámetro de Pedido Adicional con su respectivo Campo  BYDSA (1.14.1.0, 4.19.1.0) | Nancy Elizabeth Villalobos Plascencia |
| 27/04/2018 | 1.12 | Folio CAI: 0004824  Cuando se lee el código de barras de producto, si regresa más de 1 registro, deberá mostrar un listado con la clave y nombre de producto para que el vendedor seleccione cual desea agregar  **Se realiza en la versión 1.16.0.0 (BYD)** | Nancy Villalobos |
| 16/07/2018 | 1.13 | En la actividad de Consignas del móvil, al ingresar a la creación de una nueva consigna, validar que el número de clientes con consignas abiertas guardado en la nueva tabla, no exceda el límite de consignas configurado para el Vendedor. En caso de excederlo o de que el límite de consignas del Vendedor sea igual a cero, mostrar mensaje informativo y no permitir crear la consigna. Si aún no se ha excedido el límite del consignas del Vendedor, validar que el cliente al que se le pretende crear la nueva consigna no corresponda a uno de los clientes guardados en la nueva tabla, de ser así, mostrar mensaje informativo y no permitir crear la consigna.  **Se realiza en la versión 1.15.0.1 y se replica a partir de la 1.16.0.0**  Folio CAI 0004948  (eRoute, 1.17.0.0 – 4.22.0.0) | Belem Lizeth Jiménez Arévalo |

Tabla de Contenido

[Especificación de Casos de Uso: Capturar Ventas a Consignación – CUROLMOV44 9](#_Toc520459570)

[1 Introducción 9](#_Toc520459571)

[2 Caso de uso: Capturar Ventas a Consignación – CUROLMOV44 9](#_Toc520459572)

[2.1 Descripción 9](#_Toc520459573)

[3 Diagrama de Casos de Uso 9](#_Toc520459574)

[4 Precondiciones 9](#_Toc520459575)

[4.1 Actores 9](#_Toc520459576)

[4.2 Generales 10](#_Toc520459577)

[5 Flujo de eventos 10](#_Toc520459578)

[5.1 Flujo básico 10](#_Toc520459579)

[5.2 Flujos alternos 13](#_Toc520459580)

[5.2.1 Opcionales 13](#_Toc520459581)

[5.2.1.1. AO01 Crear Venta a Consignación 13](#_Toc520459590)

[5.2.1.2. AO02 Calcular Generales Venta a Consignación 41](#_Toc520459591)

[5.2.1.3. AO03 Modificar Venta a Consignación 49](#_Toc520459592)

[5.2.1.4. AO04 Eliminar Venta a Consignación 51](#_Toc520459593)

[5.2.1.5. AO05 Liquidar Venta a Consignación 57](#_Toc520459594)

[5.2.1.6. AO06 Liquidar Venta a Consignación por Devolución 62](#_Toc520459595)

[5.2.1.7. AO07 Liquidar Venta a Consignación por Cancelación 69](#_Toc520459596)

[5.2.1.8. AO08 Terminar Proceso 82](#_Toc520459597)

[5.2.2 Generales 88](#_Toc520459598)

[6 Poscondiciones 89](#_Toc520459599)

[6.1 Generales 89](#_Toc520459600)

[7 Firmas de Aceptación 89](#_Toc520459601)

# Capturar Ventas a Consignación – CUROLMOV44

# 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: Capturar Ventas a Consignación – CUROLMOV44

## Descripción

Permite al actor crear, modificar, eliminar o liquidar una venta a consignación. Se valida y modifica el inventario y se aplican promociones al igual que en los pedidos, la diferencia es que no se consideran los descuentos, ni tampoco se considera como venta.

# Diagrama de Casos de Uso

**

# Precondiciones

## Actores

1. Vendedor

## Generales

1. El actor debe de tener un usuario registrado en el sistema.
2. El actor debe de estar activo.
3. El actor debe de tener un perfil asignado para acceder al sistema.
4. El vendedor debe tener asignado el módulo de Autoventa

# Flujo de eventos

## Flujo básico

1. El caso de uso inicia cuando es invocado por el siguiente caso de uso:
   1. [**Seleccionar Actividades para la Visita – CUROLMOV11**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV11_SelecActividadesVisita.docx)
2. El sistema obtiene la siguiente información:
   1. **VARValor**
      1. VARCodigo = ‘TRPMOT’
      2. VAVClave = 12
      3. Grupo = ‘Consignacion’
      4. Estado = 1
   2. Si <no se obtuvo el tipo motivo predefinido para Consignación>
      1. El sistema registra en sesión la siguiente información:
         1. TipoMotivo = Falso
      2. El sistema presenta el mensaje “[I0284] Para llevar a cabo la venta a consignación necesita activar el motivo 12 en el grupo Consignación de TRPMOT”
      3. El sistema continúa en el [**paso 3**](#paso3) del flujo básico
   3. El sistema registra en sesión la siguiente información:
      1. GrupoMotivo = “Consignacion”
      2. TipoMotivo = Verdadero
3. El sistema obtiene de sesión la siguiente información:
   1. RUTClave
   2. DiaClave
   3. ModuloMovDetalle (Objeto)
   4. ModuloTermTipoIndice
   5. UsuarioId
   6. Cliente (Objeto)
4. El sistema obtiene la siguiente información:
   1. **Vendedor**
      1. VendedorID
      2. AlmacenID
      3. USUId = UsuarioId registrado en sesión
   2. El sistema registra en sesión la siguiente información:
      1. VendedorID
      2. AlmacenID
   3. **MOTConfiguracion**
      1. ManejoDobleUnidad
   4. El sistema registra en sesión la siguiente información:
      1. ManejoDobleUnidad (Verdadero/Falso)
   5. **CONHist**
      1. TipoClaveProducto
      2. DigitoClaveProd
   6. El sistema registra en sesión la siguiente información:
      1. TipoClaveProducto (String/Numeric)
      2. DigitoClaveProd
5. El sistema obtiene la siguiente información de la forma de venta default para el cliente actual:
   1. **CLIFormaVenta**
      1. ClienteClave = ClienteClave registrado en sesión
      2. CFVTipo
   2. El sistema registra en sesión la siguiente información:
      1. CFVTipo
6. El sistema obtiene la siguiente información del histórico de configuración más reciente:
   1. **CONHist**
      1. ConsignaContado
      2. CancConsigLiqui
      3. TipoLimiteCredito
      4. TipoCobranza
      5. PagoAutomatico
      6. PreLiquidacion
   2. El sistema registra en sesión la siguiente información:
      1. ConsignaContado (Verdadero/Falso)
      2. CancConsigLiqui (Verdadero/Falso)
      3. TipoLimiteCredito
      4. TipoCobranza (Facturas/Ventas/Ambas)
      5. PagoAutomatico (Verdadero/Falso)
      6. PreLiquidacion (Verdadero/Falso)
7. El sistema presenta el siguiente título de pantalla:
   1. Consignación
   2. Fecha: Se presenta la fecha del día de trabajo actual (DiaClave), a partir de la información registrada en sesión.
8. El sistema registra en sesión la siguiente información:
   1. TotalInicialCredito = 0
9. El sistema obtiene la siguiente información:
   1. **TransProd**
      1. TransProdID
      2. ClienteClave = <Cliente.ClienteClave> registrado en sesión (objeto Cliente)
      3. Folio
      4. Tipo = ‘24’
      5. TipoFase
      6. FechaCaptura
      7. Total
      8. Saldo
      9. Enviado
10. Si <no se obtuvieron Ventas a Consignación para el cliente actual> Y si <el cliente actual tiene configurada la forma de venta a “Crédito” <CLIFormaVenta.CFVTipo = 2 donde CLIFormaVenta.ClienteClave = ClienteClave registrado en sesión>> O si <el parámetro de configuración indica que se le pueden hacer consignaciones a los clientes cuya forma de venta sea únicamente contado (ConsignaContado = 1) de acuerdo a la información registrada en sesión>
    1. El sistema continúa con el flujo alterno opcional [**AO01 Crear Venta a Consignación**](#_AO01_Crear_Movimiento)
11. El sistema presenta los siguientes datos como encabezado de la actividad:
    1. **Ruta:** Se deberá presentar la concatenación de la Clave (Ruta.RUTClave) y nombre de la Ruta <Ruta.Descripcion>, ejemplo: “1001 Preventa 1”. Obtener la información de la ruta actual a partir de la información registrada en sesión.
    2. **Día de Trabajo:** Se deberá presentar el día de trabajo actual (DiaClave), a partir de la información registrada en sesión.
12. Para cada movimiento de Venta a Consignación obtenido
    1. El sistema presenta la siguiente información:
       1. Folio: <TransProd.Folio>. Mostrar de sólo lectura.
       2. **Fase:** <TransProd.TipoFase>. Mostrar de sólo lectura la descripción del valor por referencia correspondiente <VAVDescripcion.Descripcion> donde <VAVDescripcion.VARCodigo = ‘TRPFASE’> y <VAVDescripcion.VAVClave = TransProd.TipoFase>, de acuerdo al tipo de lenguaje configurado para el sistema.
       3. **Saldo:** <TransProd.Saldo>. Mostrar de sólo lectura. Si la consignación se encuentra liquidada <TransProd.TipoFase = 6 ‘Liquidado’ u 8 ‘Cancelada por Liquidación’> y si además tiene promociones aplicadas <TrpPrp.TransProdID = TransProd.TransProdID de la consignación>, mostrar este dato como vacío.
       4. **Total:** <TransProd.Total>. Mostrar de sólo lectura.
       5. **Fecha:** <TransProd.FechaCaptura>. Mostrar de sólo lectura en formato dd/mm/aaaa.
13. El sistema presenta la siguiente información en la barra inferior de la pantalla:
    1. **Saldo Total:** <∑TransProd.Saldo> de las ventas a consignación obtenidas.
14. Si <el cliente actual NO tiene configurada la forma de venta a “Crédito” <CLIFormaVenta.CFVTipo <> 2 donde CLIFormaVenta.ClienteClave = ClienteClave registrado en sesión> Y si <el parámetro de configuración indica que NO se le pueden hacer consignaciones a los clientes cuya forma de venta sea únicamente contado (ConsignaContado = 0) de acuerdo a la información registrada en sesión> **O** si <no se obtuvo el tipo motivo predefinido para Consignación (TipoMotivo = Falso) de acuerdo a la información registrada en sesión>
    1. El sistema presenta la opción **Continuar** de manera deshabilitada
15. Si <el actor selecciona la opción **Continuar**>
    1. El sistema continúa con el flujo alterno opcional **[AO01 Crear Venta a Consignación](#_AO01_Crear_Movimiento)**
16. Si <el actor **selecciona sostenidamente** un movimiento de Venta a Consignación de la lista> Y si <se obtuvo el tipo motivo predefinido para Consignación (TipoMotivo = Verdadero) de acuerdo a la información registrada en sesión>
    1. El sistema registra en sesión la siguiente información del movimiento seleccionado:
       1. TransProdId
    2. El sistema presenta las siguientes opciones:
       1. *Modificar*
       2. *Liquidar*
       3. *Eliminar*
    3. Si <el actor selecciona la opción **Modificar**>
       1. Si <el movimiento de Venta a Consignación se encuentra en fase “Liquidada” o “Cancelada por Liquidación” <TransProd.TipoFase = 6 u 8>>
          1. Si <el parámetro de configuración indica que NO se puede cancelar una venta a consignación por liquidación (CancConsigLiqui = 0) de acuerdo a la información registrada en sesión>
             1. El sistema continúa con el flujo alterno opcional [**AO05 Liquidar Venta a Consignación**](#_AO04_Liquidar_Venta)
          2. El sistema presenta el mensaje “[E0689] No se puede $0$ una consignación en fase Liquidada”, sustituir el valor de $0$ por “Modificar”.
          3. El sistema continúa en el paso donde fue invocado
       2. Si <el movimiento de Venta a Consignación cuenta con abonos asociados, es decir si <TransProd.Total> es diferente de <TransProd.Saldo + ∑TrpTpd.Total donde <TrpTpd.TransProdId1 = TransProd.TransProdID>>
          1. El sistema presenta el mensaje “[E0688] No se puede Modificar ó Eliminar $0$ porque tiene Abonos aplicados”, sustituir el valor de $0$ por “Consignación”.
       3. Si <el movimiento de Venta a Consignación no fue realizado en la visita actual, es decir si <TransProd.VisitaClave> es diferente de la VisitaClave registrada en sesión>
          1. Si <el movimiento de Venta a Consignación se encuentra en fase “Liquidada” o “Cancelada por Liquidación” <TransProd.TipoFase = 6 u 8>>
             1. El sistema presenta el mensaje “[E0747] No puede editarse la consigna, ya que fue $0$ en otra visita”, sustituir el valor de $0$ por “Liquidada”.
          2. El sistema presenta el mensaje “[E0747] No puede editarse la consigna, ya que fue $0$ en otra visita”, sustituir el valor de $0$ por “Creada”.
       4. Si <la información del movimiento de Venta a Consignación ya fue enviado al servidor <TransProd.Enviado = 1>>
          1. El sistema presenta el mensaje “[E0596] No se puede $0$ el registro porque ya se transfirió al servidor”, sustituir el valor de $0$ por “Modificar”.
          2. El sistema continúa en el paso donde fue invocado
       5. Si <la información del movimiento de Venta a Consignación no ha sido enviado al servidor <TransProd.Enviado = 0>>
          1. El sistema continúa con el flujo alterno opcional **[AO03 Modificar Venta a Consignación](#_AO03_Modificar_Venta)**
    4. Si <el actor selecciona la opción **Eliminar**>
       1. Si <el movimiento de Venta a Consignación se encuentra en fase “Liquidada” o “Cancelada por Liquidación” <TransProd.TipoFase = 6 u 8>>
          1. El sistema presenta el mensaje “[E0689] No se puede $0$ una consignación en fase Liquidada”, sustituir el valor de $0$ por “Eliminar”.
          2. El sistema continúa en el paso donde fue invocado
       2. Si <el movimiento de Venta a Consignación cuenta con abonos asociados, es decir si <TransProd.Total> es diferente de <TransProd.Saldo + ∑TrpTpd.Total donde <TrpTpd.TransProdId1 = TransProd.TransProdID>>
          1. El sistema presenta el mensaje “[E0688] No se puede Modificar ó Eliminar $0$ porque tiene Abonos aplicados”, sustituir el valor de $0$ por “Consignación”.
       3. Si <el movimiento de Venta a Consignación no fue realizado en la visita actual, es decir si <TransProd.VisitaClave> es diferente de la VisitaClave registrada en sesión>
          1. El sistema presenta el mensaje “[E0731] No se puede eliminar la $0$ ya que fue creada en otra visita”, sustituir el valor de $0$ por “Consignación”.
       4. Si <la información del movimiento de Venta a Consignación ya fue enviada al servidor <TransProd.Enviado = 1>>
          1. El sistema presenta el mensaje “[E0596] No se puede $0$ el registro porque ya se transfirió al servidor”, sustituir el valor de $0$ por “Eliminar”.
          2. El sistema continúa en el paso donde fue invocado
       5. Si <la información del movimiento de Venta a Consignación no ha sido enviada al servidor <TransProd.Enviado = 0>>
          1. El sistema continúa con el flujo alterno opcional **[AO04 Eliminar Venta a Consignación](#_AO03_Eliminar_Venta)**
    5. Si <el actor selecciona la opción **Liquidar**>
       1. Si <el movimiento de Venta a Consignación se encuentra en fase “Liquidada” o “Cancelada por Liquidación” <TransProd.TipoFase = 6 u 8>>
          1. El sistema presenta el mensaje “[E0689] No se puede $0$ una consignación en fase Liquidada”, sustituir el valor de $0$ por “Liquidar”.
          2. El sistema continúa en el paso donde fue invocado
       2. El sistema continúa con el flujo alterno opcional **[AO05 Liquidar Venta a Consignación](#_AO04_Liquidar_Venta)**
17. Si <el actor selecciona la opción **Regresar**>, el sistema continúa en el [**paso 1**](#FIN)**8** del flujo básico
18. Finaliza el caso de uso

## Flujos alternos

### Opcionales



### [AO01 Crear Venta a Consignación](#AO01_r)

1. El sistema valida si el Vendedor puede realizar nuevas Ventas a Consignación:
   1. El sistema obtiene la siguiente información:
      1. **Vendedor**
         1. ValidaLimiteConsignas
         2. LimiteConsignas
      2. **VendedorConsigAb**
         1. VendedorID
         2. ClienteClave
   2. Si <el Vendedor tiene activo el parámetro para validar el número de clientes distintos que pueden tener consignas abiertas <Vendedor.ValidaLimiteConsignas = 1>>
      1. El sistema registra en sesión la siguiente información:
         1. NumeroClientes = Cantidad de clientes con consignas abiertas realizadas por el Vendedor, a partir de la información obtenida en el paso anterior <VendedorConsigAb.ClienteClave>.
      2. Si <el límite de consignas del vendedor es igual a cero <Vendedor.LimiteConsignas = 0>>
         1. El sistema presenta el mensaje “[I0338] El vendedor no tiene permitido realizar ventas a consignación”
         2. El sistema continúa en el [**paso 7**](#PasoPresentaTitulo) del flujo básico
      3. Si <el cliente actual NO se encuentra en el listado de clientes registrados con consignas abiertas para el vendedor <VendedorConsigAb.ClienteClave *NO* *INCLUYE A* ClienteClave registrado en sesión>>
         1. Si <el límite de consignas del vendedor es igual o menor al número de clientes con consignas abiertas registradas para el vendedor <Vendedor.LimiteConsignas <= *NumeroClientes* registrado en sesión>>
            1. El sistema presenta el mensaje “[I0339] No es posible capturar la Consigna debido a que el número de clientes con consignas abiertas para el vendedor ha llegado a su límite”
            2. El sistema continúa en el [**paso 7**](#PasoPresentaTitulo) del flujo básico
         2. Si <el límite de consignas del vendedor es mayor al número de clientes con consignas abiertas registradas para el vendedor <Vendedor.LimiteConsignas > *NumeroClientes* registrado en sesión >>
            1. El sistema continúa en el [**paso 2**](#PasoObtenerFolioMov) del flujo alterno opcional[**AO01 Crear Venta a Consignación**](file:///C:\Duxstar\Productos\eRoute\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV44_CapturarVentasConsignación.docx#_AO01_Crear_Movimiento)
      4. Si <el cliente actual SI se encuentra en el listado de clientes registrados con consignas abiertas para el vendedor <VendedorConsigAb.ClienteClave *INCLUYE A* ClienteClave registrado en sesión>>
         1. El sistema continúa en el [**paso 2**](#PasoObtenerFolioMov) del flujo alterno opcional[**AO01 Crear Venta a Consignación**](file:///C:\Duxstar\Productos\eRoute\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV44_CapturarVentasConsignación.docx#_AO01_Crear_Movimiento)
   3. Si <el Vendedor NO tiene activo el parámetro para validar el número de clientes distintos que pueden tener consignas abiertas <Vendedor.ValidaLimiteConsignas = 0>>
      1. El sistema continúa en el siguiente paso de este flujo alterno opcional
2. El sistema obtiene la siguiente información:
   1. **Folio**
      1. FolioID
      2. ModuloMovDetalle = ModuloMovDetalle.ModuloMovDetalleClave registrado en sesión (objeto ModuloMovDetalle)
3. Si <la actividad no cuenta con un folio asignado (FolioID = NULL o ‘ ‘)>
   1. El sistema presenta el mensaje “[E0771] No hay un folio configurado para la actividad”.
4. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Transacción – CUROLGEN05**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN05_AdministrarTransaccion.doc)
   1. El sistema envía como parámetro la siguiente información:
      1. GenerarVentaConsignacion
         1. ModuloMovDetalle = ModuloMovDetalle registrado en sesión (objeto)
   2. El sistema recibe como parámetro la siguiente información:
      1. PS\_TransProd
      2. PS\_Null
   3. Si <ocurrió alguna excepción durante la generación del encabezado de la transacción (PS\_Null)>
      1. El sistema continúa en el [**paso 1**](#FIN)**8** del flujo básico
   4. El sistema registra en sesión la información del objeto recibido para el encabezado de la transacción:
      1. TransProd
5. El sistema presenta la siguiente información:
   1. Lista de Precios: Se presenta la información de la lista de precios, a través de la concatenación de la clave <Precio.ProductoClave donde TransProd.PCEPrecioClave (objetoTransProd) = Precio.ProductoClave> y nombre del mismo <Precio.Nombre>, separados por un guion medio, ejemplo: “0002 – Lista Especial”
6. Si <ManejoDobleUnidad = 1 “Verdadero”>
   1. El sistema obtiene la siguiente información:
      1. **ConfigParametro**
         1. Parametro = OrdenCapturaUnidad
         2. Identificador = ModuloMovDetalleClave registrado en sesión
         3. Valor
7. El sistema presenta y solicita la siguiente información:
   1. Producto (\*)
   2. Si <ManejoDobleUnidad = 0 “Falso”>
      1. Unidad (\*) (c)
      2. Cantidad (\*)
   3. Si <ManejoDobleUnidad = 1 “Verdadero”>
      1. Si <Existe una unidad donde <ProductoUnidad.TipoEstado = Alterna> donde <ProductoUnidad.ProductoClave = Producto actual>
         1. Para cada Unidad de acuerdo al orden configurado <ConfigParametro.Valor>, se va a buscar la descripción correspondiente al Valor por Referencia UNIDADV, que coincida con los valores registrados en el parámetro y se muestran de acuerdo al orden de aparición
            1. <NombreUnidad>
            2. Cantidad (\*)
      2. Si <No Existe una unidad donde <ProductoUnidad.TipoEstado = Alterna> donde <ProductoUnidad.ProductoClave = Producto actual>
         1. Solo se presenta la siguiente información de la unidad que se tiene activa
            1. <NombreUnidad>
            2. Cantidad (\*)
8. Para cada producto incluido en el objeto Productos
   1. El sistema obtiene la siguiente información del objeto Productos:
      1. Productos
         1. ProductoClave
         2. Descripcion
         3. SubEmpresaID
         4. DecimalProducto
         5. Unidad
         6. UnidadAlterna (\*\*En caso de aplicar)
         7. Cantidad
         8. CantidadAlterna (\*\*En caso de aplicar)
         9. Precio
         10. Subtotal
         11. Impuesto
         12. Total
   2. El sistema presenta la siguiente información:
      1. Producto: Se presenta la descripción del producto, a través de la concatenación de la clave <Productos.ProductoClave> y nombre del mismo <Productos.Descripcion>, separados por un guion medio, ejemplo: “P0001 - Jugo Durazno”
      2. Cantidad: Se presenta la cantidad de producto <Productos.Cantidad>. Formatear la cantidad a mostrar de acuerdo al número de decimales configurado para el producto <Productos.DecimalProducto>.
      3. Unidad: Se presenta la descripción de la unidad de venta del producto, para ello ir a <VAVDescripcion> y obtener <VAVDescripcion.Descripcion> donde <VAVDescripcion.VARCodigo = ‘UNIDADV’> y <VAVDescripcion.VAVClave = Productos.Unidad>, de acuerdo al tipo de lenguaje configurado para el sistema.
      4. Si <ManejoDobleUnidad = 1 “verdadero”>
         1. Cantidad Alterna: se presenta la cantidad de producto capturada o calculada en la unidad Alterna <Productos.CantidadAlterna>
         2. Unidad Alterna: Se presenta la descripción de la unidad Alterna del producto, para ello ir a <VAVDescripcion> y obtener <VAVDescripcion.Descripcion> donde <VAVDescripcion.VARCodigo = ‘UNIDADV’> y <VAVDescripcion.VAVClave = Productos.UnidadAlterna>, de acuerdo al tipo de lenguaje configurado para el sistema
      5. Precio: Se presenta el precio unitario del producto <Productos.Precio>, precedido del signo “$”.
      6. Subtotal: Se presenta el subtotal del producto <Productos.Subtotal>, precedido del signo “$”.
      7. Impuesto: Se presenta el importe de los impuestos aplicados al producto <Productos.Impuesto>, precedido del signo “$”.
      8. Total: Se presenta el importe total de la partida del producto <Productos.Total>, precedido del signo “$”.
9. El sistema presenta la siguiente información:
   1. **Total Importe:** Se presenta la sumatoria del importe total de las partidas que conforman el detalle de la transacción, <∑TransProdDetalle.Total>, precedido del signo “$”.
   2. **Productos:** Se presenta el número de partidas que conforman el detalle de la transacción.
10. Para cada producto agregado por el usuario
    1. Si <el actor posiciona el cursor dentro del campo de texto para captura de Producto>
       1. Si <el parámetro de configuración indica que el tipo de clave de los productos es alfanumérica (TipoClaveProducto = String) de acuerdo a la información registrada en sesión>
          1. El sistema presenta el teclado alfanumérico
       2. Si <el parámetro de configuración indica que el tipo de clave de los productos es numérica (TipoClaveProducto = Numeric) de acuerdo a la información registrada en sesión>
          1. El sistema presenta el teclado numérico
    2. Si <el parámetro de configuración indica que el tipo de clave de los productos es numérica (TipoClaveProducto = Numeric) de acuerdo a la información registrada en sesión>
       1. Si <el actor proporciona la Clave del Producto>
          1. El sistema formatea la clave proporcionada autocompletando con ceros a la izquierda hasta tener una cifra de acuerdo al número de dígitos configurado para la clave del producto (DigitoClaveProd) registrado en sesión
          2. El sistema obtiene la información relacionada al producto cuya información sea igual a la Clave de Producto proporcionada:
             1. **Producto**

ProductoClave = Clave de Producto proporcionada y formateada

Nombre

Contenido

Venta

DecimalProducto

* + - 1. El sistema continúa en el [**paso 9.3.3.1**](#PasoSiProductoClaveoIDCoincide) del flujo alterno opcional[**AO01 Crear Venta a Consignación**](file:///C:\Duxstar\Productos\eRoute\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV44_CapturarVentasConsignación.docx#_AO01_Crear_Movimiento)
  1. Si <el actor proporciona Clave del Producto / ID del Producto / Código de Barras / Descripción>
     1. El sistema obtiene la información relacionada a los productos cuya información sea igual a la Clave de Producto, Id, Código de Barras o Descripción proporcionados:
        1. **Producto**
           1. ProductoClave
           2. Nombre
           3. NombreLargo
           4. Id
           5. Contenido
           6. Venta
           7. SubEmpresaID
           8. DecimalProducto
           9. **ProductoUnidad**

ProductoClave

PRUTipoUnidad

CodigoBarras

TipoEstado = 1 o 3

* + 1. Si <existen coincidencias parciales en Nombre / NombreLargo / ProductoClave / ID>
       1. El sistema registra en sesión la siguiente información
          1. Cadena =Cadena de búsqueda introducida por el usuario
       2. Si <el parámetro de configuración indica que se tiene activado el Manejo de Doble Unidad (ManejoDobleUnidad = 1) de acuerdo con la información registrada en sesión>
          1. El caso de uso incluye la funcionalidad del caso de uso [**Buscar Productos – CUROLMOV20**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV20_BuscarProductos.docx)

El sistema envía como parámetro la siguiente información:

Cadena

Esquema = Todos

TipoValidarExistencia = 1

ListaPrecios = <TransProd.PCEPrecioClave> registrado en sesión (objeto TransProd)

TransProdID = <TransProd.TransProdID> registrado en sesión (objeto TransProd)

TipoMovimiento = <TransProd.TipoMovimiento> registrado en sesión (objeto TransProd)

TipoTransProd = <TransProd.Tipo> registrado en sesión (objeto TransProd)

BusquedaSimple = 1

El sistema recibe como parámetro la siguiente información:

Productos

Clave del Producto

* + - * 1. El sistema continúa en el [**paso 9.3.3.1**](file:///C:\Duxstar\Productos\eRoute\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV44_CapturarVentasConsignación.docx#PasoSiProductoClaveoIDCoincide) del flujo alterno opcional[**AO01 Crear Venta a Consignación**](file:///C:\Duxstar\Productos\eRoute\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV44_CapturarVentasConsignación.docx#_AO01_Crear_Movimiento)
      1. El caso de uso incluye la funcionalidad del caso de uso [**Buscar Productos – CUROLMOV20**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV20_BuscarProductos.docx)
         1. El sistema envía como parámetro la siguiente información:

Cadena

Esquema = Todos

TipoValidarExistencia = 1

ListaPrecios = <TransProd.PCEPrecioClave> registrado en sesión (objeto TransProd)

TransProdID = <TransProd.TransProdID> registrado en sesión (objeto TransProd)

TipoMovimiento = <TransProd.TipoMovimiento> registrado en sesión (objeto TransProd)

TipoTransProd = <TransProd.Tipo> registrado en sesión (objeto TransProd)

BusquedaSimple = 0

* + - * 1. El sistema recibe como parámetro la siguiente información:

Productos

Clave del Producto

Unidad

Cantidad

Precio

* + - 1. Para cada producto incluido en el objeto Productos
         1. El sistema obtiene la siguiente información del producto actual:

**Producto**

ProductoClave

Nombre

SubEmpresaID

DecimalProducto

* + - * 1. El sistema actualiza el objeto Productos con la siguiente información para el producto actual:

Productos

Descripcion = <Producto.Nombre>

SubEmpresaID

DecimalProducto

* + - * 1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Inventario Móvil – CUROLGEN13**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN13_AdministrarInventario.doc)

El sistema envía como parámetro la siguiente información:

ActualizarInventario

ProductoClave = ProductoClave (objeto Productos)

TipoUnidad = Unidad (objeto Productos)

Cantidad = Cantidad (objeto Productos)

TipoTransProd = <TransProd.Tipo> registrado en sesión (objeto TransProd)

TipoMovimiento = <TransProd.TipoMovimiento> registrado en sesión (objeto TransProd)

AlmacenID = AlmacenID registrado en sesión

El sistema recibe la siguiente información:

OP\_Verdadero

* + - * 1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Detalle de la Transacción – CUROLGEN06**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN06_AdministrarDetalleTransaccion.doc)

El sistema envía como parámetro la siguiente información:

GenerarDetalleVentaConsignacion

TransProdId = <TransProd.TransProdID> (objeto TransProd)

ProductoClave = ProductoClave (objeto Productos)

TipoUnidad = Unidad (objeto Productos)

Cantidad = Cantidad (objeto Productos)

CantidadOriginal = 0

PrecioClave = <TransProd.PCEPrecioClave> (objeto TransProd)

El sistema recibe como parámetro la siguiente información:

PS\_TransProdDetalle

El sistema registra la información del objeto recibido dentro de un arreglo:

**ArregloTransProdDetalle**

TransProdDetalle

* + - * 1. El sistema obtiene la siguiente información del objeto recibido:

**ArregloTransProdDetalle**

TransProdDetalle

Subtotal

Impuesto

Total

* + - * 1. El sistema actualiza el objeto Productos con la siguiente información para el producto actual:

Productos

Subtotal

Impuesto

Total

* + - * 1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Transacción – CUROLGEN05**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN05_AdministrarTransaccion.doc)

El sistema envía como parámetro la siguiente información:

CalcularTotalesTransaccion

TransProd = <TransProd.TransProdID> registrado en sesión (objeto TransProd)

El sistema recibe la siguiente información

TransProd

El sistema actualiza la información del objeto registrado en sesión:

TransProd

* + - * 1. Si <al cliente se le debe realizar el manejo del préstamo de envase <Cliente.Prestamo = 1 donde Cliente.ClienteClave = TransProd.ClienteClave (objeto TransProd registrado en sesión)>

Si <el producto actual se puede prestar <ProductoDetalle.ProductoClave = ProductoClave (objeto Productos)> y <ProductoDetalle.PRUTipoUnidad = Unidad (objeto Productos)> y <ProductoDetalle.Prestamo = 1>>

El sistema obtiene información del Saldo del préstamo del producto, donde <ProductoPrestamoCli.ClienteClave = TransProd.ClienteClave (objeto TransProd registrado en sesión)> y <ProductoPrestamoCli.ProductoClave = ProductoDetalle.ProductoDetClave donde ProductoDetalle.ProductoClave = ProductoClave (objeto Productos) y ProductoDetalle.PRUTipoUnidad = Unidad (objeto Productos)>:

**ProductoPrestamoCli**

Saldo

Si <no se obtuvo un registro de ProductoPrestamoCli para el Cliente y producto actuales>

El caso de uso incluye la funcionalidad del caso de uso [**Administrar Préstamo de Producto – CUROLGEN15**](file:///C:\Users\Belem\AppData\Roaming\Microsoft\ECU_GEN\CUROLGEN15_AdministrarPrestamoDeProducto.docx)y le envía como parámetro la siguiente información:

CrearPrestamo

ClienteClave = <TransProd.ClienteClave> (objeto TransProd registrado en sesión)

ProductoClave = <ProductoDetalle.ProductoDetClave> donde <ProductoDetalle.ProductoClave = ProductoClave (objeto Productos) y ProductoDetalle.PRUTipoUnidad = Unidad (objeto Productos)>

El sistema recibe la siguiente información:

OP\_Verdadero

El caso de uso incluye la funcionalidad del caso de uso [**Administrar Préstamo de Producto – CUROLGEN15**](file:///C:\Users\Belem\AppData\Roaming\Microsoft\ECU_GEN\CUROLGEN15_AdministrarPrestamoDeProducto.docx)y le envía como parámetro la siguiente información:

ActualizarPrestamo

ClienteClave = <TransProd.ClienteClave> (objeto TransProd registrado en sesión)

ProductoClave = <ProductoDetalle.ProductoDetClave> donde <ProductoDetalle.ProductoClave = ProductoClave (objeto Productos) y ProductoDetalle.PRUTipoUnidad = Unidad (objeto Productos)>

TipoTransProd = <TransProd.Tipo> (objeto TransProd registrado en sesión)

Cantidad = Cantidad (objeto Productos)

TipoUnidad = Unidad (objeto Productos)

Vendido = 0

El sistema recibe la siguiente información:

OP\_Verdadero

* + - * 1. Si <el producto cuenta con una SubEmpresa asociada <Productos.SubEmpresaID <> Null o ‘ ‘> (objeto Productos)>

El sistema actualiza la siguiente información para la transacción registrada en sesión:

**TransProd**

SubEmpresaID = SubEmpresaID (objeto Productos)

* + - 1. El sistema continúa en el [**paso 7**](#PasoParaCadaProductoObjetoProductos) del flujo alterno opcional[**AO01 Crear Venta a Consignación**](#_AO01_Crear_Movimiento)
    1. Si <ProductoClave coincide> o Si <ID coincide>
       1. Si <Contenido es **Verdadero** <Producto.Contenido = 1>>
          1. Si <Venta es**Falso** <Producto.Venta = 0>>

El sistema presenta el mensaje “[E0725] El producto no está configurado para entregarse por separado”

El sistema continúa en el[**paso 6.1**](#PasoPresentaSolicitaProducto)del flujo alterno opcional[**AO01 Crear Venta a Consignación**](#_AO01_Crear_Movimiento)

* + - 1. El sistema registra la siguiente información del producto seleccionado dentro del objeto Productos
         1. Productos

ProductoClave = <Producto.ProductoClave>

Descripcion = <Producto.Nombre>

SubEmpresaID

DecimalProducto = <Producto.DecimalProducto>

* + - 1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Inventario Móvil – CUROLGEN13**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN13_AdministrarInventario.doc)
         1. El sistema envía como parámetro la siguiente información:

ObtenerExistencia

ProductoClave = ProductoClave (objeto Productos)

* + - * 1. El sistema recibe la siguiente información:

OP\_Existencia

* + - 1. El sistema presenta la siguiente información del producto:
         1. Descripción = <Producto.Nombre>. Mostrar de sólo lectura
         2. **Exist =** OP\_Existencia (formatear la cantidad de acuerdo al número de decimales configurado para el producto <Producto.DecimalProducto>). Mostrar de sólo lectura.
      2. El sistema continúa en el [**paso 9.3.7**](file:///C:\Duxstar\Productos\eRoute\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV44_CapturarVentasConsignación.docx#PasoObtenerUnidadesProducto) del flujo alterno opcional[**AO01 Crear Venta a Consignación**](#_AO01_Crear_Movimiento)
    1. Si <CodigoBarras coincide>
       1. Si <El sistema encuentra más de un registro con el mismo Código de Barras>
          1. El sistema presenta en una nueva pantalla emergente la siguiente información:

Código de Barras: <Información del código leído>

Para cada Producto encontrado con el mismo CB

<ProductoUnidad.PRUTipoUnidad>

<Producto.ProductoClave>

<Producto.NombreLargo>

* + - * 1. Si <Si el actor selecciona un registro del listado>

El sistema continúa en el siguiente paso desde donde fue invocado

* + - 1. Si <Contenido es **Verdadero** <Producto.Contenido = 1>>
         1. Si <Venta es**Falso** <Producto.Venta = 0>>

El sistema presenta el mensaje “[E0725] El producto no está configurado para entregarse por separado”

El sistema continúa en el[**paso 6.1**](#PasoPresentaSolicitaProducto)del flujo alterno opcional[**AO01 Crear Venta a Consignación**](#_AO01_Crear_Movimiento)

* + - 1. El sistema registra la siguiente información del producto seleccionado dentro del objeto Productos
         1. Productos

ProductoClave = <Producto.ProductoClave>

Descripcion = <Producto.Nombre>

SubEmpresaID

DecimalProducto = <Producto.DecimalProducto>

* + - 1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Inventario Móvil – CUROLGEN13**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN13_AdministrarInventario.doc)
         1. El sistema envía como parámetro la siguiente información:

ObtenerExistencia

ProductoClave = ProductoClave (objeto Productos)

* + - * 1. El sistema recibe la siguiente información:

OP\_Existencia

* + - 1. El sistema presenta la siguiente información del producto:
         1. Descripción = <Producto.Nombre>. Mostrar de sólo lectura.
         2. **Exist =** OP\_Existencia (formatear la cantidad de acuerdo al número de decimales configurado para el producto <Producto.DecimalProducto>). Mostrar de sólo lectura.
      2. El sistema presenta la unidad correspondiente al código de barras proporcionado. Mostrar de sólo lectura y posicionar el cursor dentro del combo de selección de cantidad de producto:
         1. Unidad
      3. El sistema continúa en el [**paso 9.3.10**](#PasoActorProporcionaCantidad) del flujo alterno opcional[**AO01 Crear Venta a Consignación**](#_AO01_Crear_Movimiento)
    1. Si <el código de barras no coincide>
       1. El sistema presenta el mensaje“[E0863] El Código de Barras no corresponde a un producto existente”
       2. El sistema continúa en el[**paso 6.1**](#PasoPresentaSolicitaProducto)del flujo alterno opcional[**AO01 Crear Venta a Consignación**](#_AO01_Crear_Movimiento)
    2. Si <no existen coincidencias parciales o coincidencias idénticas>
       1. El sistema presenta el mensaje “[E0878] No existen coincidencias de producto o ya se encuentran todos capturados”
       2. El sistema continúa en el[**paso 6.1**](#PasoPresentaSolicitaProducto)del flujo alterno opcional[**AO01 Crear Venta a Consignación**](#_AO01_Crear_Movimiento)
    3. El caso de uso incluye la funcionalidad del caso de uso [**Consultar Características del Producto – CUROLGEN10**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN10_Consultar%20Características%20del%20Producto.doc)
       1. El sistema envía como parámetro la siguiente información:
          1. ObtenerUnidadesProducto

ProductoClave

PrecioClave

* + - 1. El sistema recibe como parámetro la siguiente información:
         1. ListaTiposUnidades
    1. El sistema valida las Unidades relacionadas al producto:
       1. Si <el producto sólo tiene una unidad> y Si <ManejoDobleUnidad = Falso>
          1. Si <el producto no ha sido capturado>

El sistema presenta la unidad correspondiente. Mostrar de sólo lectura y posicionar el cursor dentro del combo de selección de cantidad de producto:

Unidad

El sistema continúa en el [**paso 9.3.10**](#PasoActorProporcionaCantidad) del flujo alterno opcional[**AO01 Crear Venta a Consignación**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV31_CapturarMovSinInvFueraVisita.docx#_AO01_Crear_Movimiento)

* + - * 1. Si <el producto ya fue capturado>

El sistema continúa en el [**paso 10.1**](#PasoSeleccionProducto) del flujo alterno opcional[**AO01 Crear Venta a Consignación**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV31_CapturarMovSinInvFueraVisita.docx#_AO01_Crear_Movimiento)

* + - 1. Si <el producto tiene más de una unidad> y Si <ManejoDobleUnidad = Falso>
         1. Si <el producto no ha sido capturado>

El sistema presenta el listado de las unidades relacionadas al producto seleccionado:

Unidad

El sistema continúa en el [**paso 9.3.9**](#PasoActorProporcionaUnidad)del flujo alterno opcional[**AO01 Crear Venta a Consignación**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV31_CapturarMovSinInvFueraVisita.docx#_AO01_Crear_Movimiento)

* + - * 1. Si <el producto ya fue capturado>

El sistema presenta el listado de las unidades relacionadas al producto seleccionado que no se encuentren ya capturadas:

Unidad

El sistema continúa en el [**paso 9.3.9**](#PasoActorProporcionaUnidad)del flujo alterno opcional[**AO01 Crear Venta a Consignación**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV31_CapturarMovSinInvFueraVisita.docx#_AO01_Crear_Movimiento)

* + - 1. Si <ManejoDobleUnidad = 1 “verdadero”>
         1. El sistema obtiene las unidades relacionadas al producto donde <ProductoUnidad.TipoEstado <> 0>

ProductoUnidad

ProductoClave = Producto Actual

PRUTipoUnidad

TipoEstado <> 0

* + - * 1. Para cada Unidad relacionada al Producto

El caso de uso incluye la funcionalidad del caso de uso [Determinar Lista de Precios – CUROLGEN04](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN04_DeterminarListaPrecio.doc)

El sistema envía como parámetro la siguiente información

ObtenerPrecio

ListaPreciosPredeterminada de acuerdo con las reglas de negocio [**RNGEN171 Información en Sesión**](file:///C:\Users\Route\Trunk\Analisis\EspecificacionRequerimientos\General\ERM_Reglas_de_Negocio.docx#RNGEN171) y [**RNROLMOV195 Información Contenida en el Objeto del Encabezado de la Transacción**](file:///C:\Amesol\Productos\Route\Trunk\Analisis\EspecificacionRequerimientos\General\ERM_Reglas_de_Negocio.docx#RNROLMOV195)

ProductoClave

PRUTipoUnidad

El sistema recibe como parámetro la siguiente información:

PrecioProductoVig

Si <PrecioProductoVig = NULL>, entonces registrar la siguiente información en el objeto de Productos

**Productos**

UnidadAlterna = <ProductoUnidad.PRUTipoUnidad> de la Unidad actual.

Si <PrecioProductoVig <> NULL>, entonces registrar la siguiente información en el objeto de Productos

**Productos**

Unidad = <ProductoUnidad.PRUTipoUnidad> de la Unidad actual

* + 1. El actor proporciona la siguiente información, inmediatamente después posicionar el cursor dentro del combo de selección de cantidad de producto:
       1. Unidad (\*) (c)
    2. El actor proporciona la siguiente información:
       1. Cantidad (\*)
    3. Si <ManejoDobleUnidad = 1 “Verdadero”>
       1. Si <existe más de una Unidad relacionada al Producto> y Si <la primera Unidad a mostrar es la Unidad donde <ProductoUnidad.TipoEstado = 1 “Activa>, el sistema obtiene la siguiente información:
          1. **ProductoDetalle**

ProductoClave

PRUTipoUnidad = <ProductoUnidad.PRUTipoUnidad> donde <ProductoUnidad.ProductoClave = ProductoDetalle.ProductoClave> y <ProductoUnidad.TipoEstado = 3 “Alterna”>

Factor

* + - * 1. El sistema realiza el siguiente calculo:

Cantidad Capturada \* <ProductoDetalle.Factor> *\*\*De la Unidad Alterna*

* + - * 1. El sistema presenta la Cantidad obtenida del cálculo en la siguiente Unidad mostrada *\*\*Formatear la cantidad a mostrar de acuerdo al número de decimales configurado para el producto y la unidad <ProductoUnidad.DecimalProducto> donde <ProductoUnidad.ProductoClave = Producto Actual> y <ProductoUnidad.TipoEstado = 3 “Alterna”>*
        2. Si <MOTConfiguracion.ModificarUnidadAlterna = 0>, se muestra este campo con la cantidad calculada de solo Lectura

El sistema actualiza el objeto de Productos con la siguiente información para el producto actual:

**Productos**

CantidadAlterna = Cantidad Calculada

* + - * 1. Si <MOTConfiguracion.ModificarUnidadAlterna = 1>, se habilita el campo para captura con la cantidad calculada y se permite al usuario la modificación *\*\*Formatear la cantidad a mostrar de acuerdo al número de decimales configurado para el producto y la unidad <ProductoUnidad.DecimalProducto> donde <ProductoUnidad.ProductoClave = Producto Actual> y <ProductoUnidad.TipoEstado = 3 “Alterna”>*

Si <el usuario modifica la Cantidad calculada>

El sistema valida la cantidad Modificada de acuerdo con la Validación [VA06 Validar Porcentaje de Variación](#VA06)

El sistema registra y actualiza la siguiente información:

**Productos**

CantidadAlterna = Cantidad Modificada

CantidadAlternaOriginal = Cantidad Calculada

Si <el usuario NO modifica la Cantidad calculada>

El sistema actualiza el objeto de Productos con la siguiente información para el producto actual:

**Productos**

CantidadAlterna

* + - 1. Si <existe más de una Unidad relacionada al Producto> y Si <la primera Unidad a mostrar es la Unidad donde <ProductoUnidad.TipoEstado = 3 “Alterna”>, el sistema obtiene la siguiente información:
         1. **ProductoDetalle**

ProductoClave

PRUTipoUnidad = <ProductoUnidad.PRUTipoUnidad> donde <ProductoUnidad.ProductoClave = ProductoDetalle.ProductoClave> y <ProductoUnidad.TipoEstado = 3 “Alterna”>

Factor

* + - * 1. El sistema realiza el siguiente calculo:

Cantidad Capturada / <ProductoDetalle.Factor>

* + - * 1. El sistema presenta la Cantidad obtenida del cálculo en la siguiente Unidad mostrada *\*\*Formatear la cantidad a mostrar de acuerdo al número de decimales configurado para el producto y la unidad <ProductoUnidad.DecimalProducto> donde <ProductoUnidad.ProductoClave = Producto Actual> y <ProductoUnidad.TipoEstado = 1 “Activo”>*
        2. Si <MOTConfiguracion.ModificarUnidadAlterna = 0>, se muestra este campo de solo Lectura

El sistema actualiza el objeto de Productos con la siguiente información para el producto actual:

**Productos**

CantidadAlterna = Cantidad Calculada

* + - * 1. Si <MOTConfiguracion.ModificarUnidadAlterna = 1>, se habilita el campo para captura y se permite al usuario la modificación de la cantidad calculada *\*\*Formatear la cantidad a mostrar de acuerdo al número de decimales configurado para el producto y la unidad <ProductoUnidad.DecimalProducto> donde <ProductoUnidad.ProductoClave = Producto Actual> y <ProductoUnidad.PRUTipoUnidad =1 “Activo”>*

Si <el usuario modifica la Cantidad calculada>

El sistema valida la cantidad Modificada de acuerdo con la Validación [VA06 Validar Porcentaje de Variación](#VA06)

El sistema registra y actualiza la siguiente información:

**Productos**

CantidadAlterna = Cantidad Modificada

CantidadAlternaOriginal = Cantidad Calculada

Si <el usuario NO modifica la Cantidad calculada>

El sistema actualiza el objeto de Productos con la siguiente información para el producto actual:

**Productos**

CantidadAlterna= Cantidad Calculada

* + 1. Si <el actor proporciona la Cantidad> y si <el actor selecciona la opción **Agregar**>
       1. El sistema valida la Cantidad de producto proporcionada por el actor:
          1. Si <la cantidad proporcionada por el actor es igual o menor a cero>

El sistema presenta el mensaje “[E0012] La cantidad debe ser mayor a cero”

El sistema continúa en el paso anterior de donde fue invocado, posicionando el cursor dentro del combo de selección de cantidad de producto

* + - 1. Si <ManejoDobleUnidad = 0 “Falso”>
         1. El sistema actualiza el objeto Productos con la siguiente información para el producto actual:

Productos

Unidad

Cantidad (formatear la cantidad de acuerdo al número de decimales configurado para el producto <Producto.DecimalProducto>)

* + - 1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Inventario Móvil – CUROLGEN13**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN13_AdministrarInventario.doc)
         1. El sistema envía como parámetro la siguiente información:

ValidarExistencia

ProductoClave = ProductoClave (objeto Productos)

TipoUnidad = Unidad (objeto Productos)

Cantidad = Cantidad (objeto Productos)

TipoMovimiento = <TransProd.TipoMovimiento> registrado en sesión (objeto TransProd)

TipoTransProd = <TransProd.Tipo> registrado en sesión (objeto TransProd)

* + - * 1. El sistema recibe la siguiente información:

OP\_Verdadero/Falso

OP\_Cantidad

* + - 1. Si <no se tiene suficiente existencia disponible para sacar del inventario la cantidad de producto proporcionada (OP\_Falso) >
         1. El sistema presenta la siguiente información:

Cantidad = OP\_Cantidad (formatear la cantidad de acuerdo al número de decimales configurado para el producto <Producto.DecimalProducto>, posicionar el cursor dentro del combo de selección de cantidad de producto)

* + - * 1. El sistema continúa en el [**paso 9.3.12**](#PasoActorModificaCantidadYAgregar) del flujo alterno opcional[**AO01 Crear Venta a Consignación**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV31_CapturarMovSinInvFueraVisita.docx#_AO01_Crear_Movimiento)
      1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Inventario Móvil – CUROLGEN13**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN13_AdministrarInventario.doc)
         1. El sistema envía como parámetro la siguiente información:

ActualizarInventario

ProductoClave = ProductoClave (objeto Productos)

TipoUnidad = Unidad (objeto Productos)

Cantidad = Cantidad (objeto Productos)

TipoTransProd = <TransProd.Tipo> registrado en sesión (objeto TransProd)

TipoMovimiento = <TransProd.TipoMovimiento> registrado en sesión (objeto TransProd)

AlmacenID = AlmacenID registrado en sesión

* + - * 1. El sistema recibe la siguiente información:

OP\_Verdadero

* + - 1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Detalle de la Transacción – CUROLGEN06**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN06_AdministrarDetalleTransaccion.doc)
         1. El sistema envía como parámetro la siguiente información:

GenerarDetalleVentaConsignacion

TransProdId = <TransProd.TransProdID> registrado en sesión (objeto TransProd)

ProductoClave = ProductoClave (objeto Productos)

TipoUnidad = Unidad (objeto Productos)

Cantidad = Cantidad (objeto Productos)

CantidadOriginal = 0

PrecioClave = <TransProd.PCEPrecioClave> (objeto TransProd)

* + - * 1. El sistema recibe como parámetro la siguiente información:

PS\_TransProdDetalle

PS\_Null

* + - * 1. Si <ocurrió alguna excepción durante la generación del encabezado de la transacción (PS\_Null)>

El sistema continúa en el [**paso 6.1**](#PasoPresentaSolicitaProducto) del flujo alterno opcional[**AO01 Crear Venta a Consignación**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV31_CapturarMovSinInvFueraVisita.docx#_AO01_Crear_Movimiento)

* + - * 1. El sistema registra la información del objeto recibido dentro de un arreglo:

**ArregloTransProdDetalle**

TransProdDetalle

* + - 1. El sistema obtiene la siguiente información del objeto recibido:
         1. **ArregloTransProdDetalle**

TransProdDetalle

Precio

Subtotal

Impuesto

Total

* + - 1. El sistema actualiza el objeto Productos con la siguiente información para el producto actual:
         1. Productos

Precio

Subtotal

Impuesto

Total

* + - 1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Transacción – CUROLGEN05**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN05_AdministrarTransaccion.doc)
         1. El sistema envía como parámetro la siguiente información:

CalcularTotalesTransaccion

TransProd = <TransProd.TransProdID> registrado en sesión (objeto TransProd)

* + - * 1. El sistema recibe la siguiente información

TransProd

* + - * 1. El sistema actualiza la información del objeto registrado en sesión:

TransProd

* + - 1. Si <al cliente se le debe realizar el manejo del préstamo de envase <Cliente.Prestamo = 1 donde Cliente.ClienteClave = TransProd.ClienteClave (objeto TransProd registrado en sesión)>
         1. Si <el producto capturado se puede prestar <ProductoDetalle.ProductoClave = ProductoClave (objeto Productos)> y <ProductoDetalle.PRUTipoUnidad = Unidad (objeto Productos)> y <ProductoDetalle.Prestamo = 1>>

El sistema obtiene información del Saldo del préstamo del producto, donde <ProductoPrestamoCli.ClienteClave = TransProd.ClienteClave (objeto TransProd registrado en sesión)> y <ProductoPrestamoCli.ProductoClave = ProductoDetalle.ProductoDetClave donde ProductoDetalle.ProductoClave = ProductoClave (objeto Productos) y ProductoDetalle.PRUTipoUnidad = Unidad (objeto Productos)>:

**ProductoPrestamoCli**

Saldo

Si <no se obtuvo un registro de ProductoPrestamoCli para el Cliente y producto actuales>

El caso de uso incluye la funcionalidad del caso de uso [**Administrar Préstamo de Producto – CUROLGEN15**](file:///C:\Users\Belem\AppData\Roaming\Microsoft\ECU_GEN\CUROLGEN15_AdministrarPrestamoDeProducto.docx)y le envía como parámetro la siguiente información:

CrearPrestamo

ClienteClave = <TransProd.ClienteClave> (objeto TransProd registrado en sesión)

ProductoClave = <ProductoDetalle.ProductoDetClave> donde <ProductoDetalle.ProductoClave = ProductoClave (objeto Productos) y ProductoDetalle.PRUTipoUnidad = Unidad (objeto Productos)>

El sistema recibe la siguiente información:

OP\_Verdadero

El caso de uso incluye la funcionalidad del caso de uso [**Administrar Préstamo de Producto – CUROLGEN15**](file:///C:\Users\Belem\AppData\Roaming\Microsoft\ECU_GEN\CUROLGEN15_AdministrarPrestamoDeProducto.docx)y le envía como parámetro la siguiente información:

ActualizarPrestamo

ClienteClave = <TransProd.ClienteClave> (objeto TransProd registrado en sesión)

ProductoClave = <ProductoDetalle.ProductoDetClave> donde <ProductoDetalle.ProductoClave = ProductoClave (objeto Productos) y ProductoDetalle.PRUTipoUnidad = Unidad (objeto Productos)>

TipoTransProd = <TransProd.Tipo> (objeto TransProd registrado en sesión)

Cantidad = Cantidad (objeto Productos)

TipoUnidad = Unidad (objeto Productos)

Vendido = 0

El sistema recibe la siguiente información:

OP\_Verdadero

* + - 1. Si <ManejoDobleUnidad = 1 “Verdadero”>
         1. El sistema registra la siguiente información:

**TPDDatosExtra**

TransProdId *(Obtenido de ArregloTransProdDetalle)*

TransProdDetalleID (Obtenido de ArregloTransProdDetalle)

UnidadAlterna (Objeto Productos)

Cantidad (Objeto Productos)

CantidadAlterna (Objeto Productos)

CantidadAlternaOriginal (Objeto Productos \*\*En caso de aplicar)

MFechaHora = Fecha y Hora Actual del Sistema

MUsuarioID = USUId registrado en sesión

* + - 1. Si <el producto cuenta con una SubEmpresa asociada <Productos.SubEmpresaID <> Null o ‘ ‘> (objeto Productos)>
         1. El sistema actualiza la siguiente información para la transacción registrada en sesión:

**TransProd**

SubEmpresaID = SubEmpresaID (objeto Productos)

* + - 1. El sistema continúa en el [**paso 7**](file:///C:\Duxstar\Productos\eRoute\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV44_CapturarVentasConsignación.docx#PasoParaCadaProductoObjetoProductos)del flujo alterno opcional[**AO01 Crear Venta a Consignación**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV31_CapturarMovSinInvFueraVisita.docx#_AO01_Crear_Movimiento)

1. Si <el actor **selecciona** un producto>
   1. El sistema obtiene del objeto Productos la siguiente información del producto seleccionado:
      1. ProductoClave
      2. Descripcion
      3. SubEmpresaID
      4. DecimalProducto
      5. Unidad
      6. UnidadAlterna (\*\*En caso de aplicar)
      7. Cantidad
      8. CantidadAlterna (\*\*En caso de aplicar)
      9. Precio
      10. Subtotal
      11. Impuesto
      12. Total
   2. El sistema recupera la siguiente información del producto:
      1. **ArregloTransProdDetalle**
         1. TransProdDetalle
   3. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Inventario Móvil – CUROLGEN13**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN13_AdministrarInventario.doc)
      1. El sistema envía como parámetro la siguiente información:
         1. ObtenerExistencia
            1. ProductoClave = ProductoClave (objeto Productos)
      2. El sistema recibe la siguiente información:
         1. OP\_Existencia
   4. El sistema presenta y solicita la siguiente información en la sección de captura, posicionando el cursor dentro del combo de selección de cantidad de producto:
      1. Producto = <Productos.ProductoClave>. Mostrar de sólo lectura.
      2. Descripción = <Productos.Descripcion>. Mostrar de sólo lectura.
      3. **Exist =** OP\_Existencia (formatear la cantidad de acuerdo al número de decimales configurado para el producto <Producto.DecimalProducto>). Mostrar de sólo lectura.
      4. Si <ManejoDobleUnidad = o “Falso”>
         1. Unidad = <Productos.Unidad>. Mostrar de sólo lectura.
      5. Cantidad = <Productos.Cantidad>. Formatear la cantidad a mostrar de acuerdo al número de decimales configurado para el producto <Productos.DecimalProducto>.
   5. Si <el actor **modifica** la cantidad> y si <el actor selecciona la opción **Agregar**>
      1. El sistema valida la Cantidad de producto proporcionada por el actor:
         1. Si <la cantidad proporcionada por el actor es igual o menor a cero>
            1. El sistema presenta el mensaje “[E0012] La cantidad debe ser mayor a cero”
            2. El sistema continúa en el paso anterior de donde fue invocado, posicionando el cursor dentro del combo de selección de cantidad de producto
      2. Si <ManejoDobleUnidad = 1 “Verdadero”>
         1. Si <existe más de una Unidad relacionada al Producto> y Si <la primera Unidad a mostrar es la Unidad donde <ProductoUnidad.TipoEstado = 1 “Activa>, el sistema obtiene la siguiente información:
            1. **ProductoDetalle**

ProductoClave

PRUTipoUnidad = <ProductoUnidad.PRUTipoUnidad> donde <ProductoUnidad.ProductoClave = ProductoDetalle.ProductoClave> y <ProductoUnidad.TipoEstado = 3 “Alterna”>

Factor

* + - * 1. El sistema realiza el siguiente calculo:

Cantidad Capturada \* <ProductoDetalle.Factor> \*\*De la Unidad Alterna

* + - * 1. El sistema presenta la Cantidad obtenida del cálculo en la siguiente Unidad mostrada *\*\*Formatear la cantidad a mostrar de acuerdo al número de decimales configurado para el producto y la unidad <ProductoUnidad.DecimalProducto> donde <ProductoUnidad.ProductoClave = Producto Actual> y <ProductoUnidad.TipoEstado = 3 “Alterna”>*
        2. Si <MOTConfiguracion.ModificarUnidadAlterna = 0>, se muestra este campo con la cantidad calculada de solo Lectura

El sistema actualiza el objeto de Productos con la siguiente información para el producto actual:

**Productos**

CantidadAlterna = Cantidad Calculada

* + - * 1. Si <MOTConfiguracion.ModificarUnidadAlterna = 1>, se habilita el campo para captura con la cantidad calculada y se permite al usuario la modificación *\*\*Formatear la cantidad a mostrar de acuerdo al número de decimales configurado para el producto y la unidad <ProductoUnidad.DecimalProducto> donde <ProductoUnidad.ProductoClave = Producto Actual> y <ProductoUnidad.TipoEstado = 3 “Alterna”>*

Si <el usuario modifica la Cantidad calculada>

El sistema valida la cantidad Modificada de acuerdo con la Validación [VA06 Validar Porcentaje de Variación](#VA06)

El sistema registra y actualiza la siguiente información:

**Productos**

CantidadAlterna = Cantidad Modificada

CantidadAlternaOriginal = Cantidad Calculada

Si <el usuario NO modifica la Cantidad calculada>

El sistema actualiza el objeto de Productos con la siguiente información para el producto actual:

**Productos**

CantidadAlterna

* + - 1. Si <existe más de una Unidad relacionada al Producto> y Si <la primera Unidad a mostrar es la Unidad donde <ProductoUnidad.TipoEstado = 3 “Alterna”>, el sistema obtiene la siguiente información:
         1. **ProductoDetalle**

ProductoClave

PRUTipoUnidad = <ProductoUnidad.PRUTipoUnidad> donde <ProductoUnidad.ProductoClave = ProductoDetalle.ProductoClave> y <ProductoUnidad.TipoEstado = 3 “Alterna”>

Factor

* + - * 1. El sistema realiza el siguiente calculo:

Cantidad Capturada / <ProductoDetalle.Factor>

* + - * 1. El sistema presenta la Cantidad obtenida del cálculo en la siguiente Unidad mostrada *\*\*Formatear la cantidad a mostrar de acuerdo al número de decimales configurado para el producto y la unidad <ProductoUnidad.DecimalProducto> donde <ProductoUnidad.ProductoClave = Producto Actual> y <ProductoUnidad.TipoEstado = 1 “Activo”>*
        2. Si <MOTConfiguracion.ModificarUnidadAlterna = 0>, se muestra este campo de solo Lectura

El sistema actualiza el objeto de Productos con la siguiente información para el producto actual:

**Productos**

CantidadAlterna = Cantidad Calculada

* + - * 1. Si <MOTConfiguracion.ModificarUnidadAlterna = 1>, se habilita el campo para captura y se permite al usuario la modificación de la cantidad calculada *\*\*Formatear la cantidad a mostrar de acuerdo al número de decimales configurado para el producto y la unidad <ProductoUnidad.DecimalProducto> donde <ProductoUnidad.ProductoClave = Producto Actual> y <ProductoUnidad.PRUTipoUnidad =1 “Activo”>*

Si <el usuario modifica la Cantidad calculada>

El sistema valida la cantidad Modificada de acuerdo con la Validación [VA06 Validar Porcentaje de Variación](#VA06)

El sistema registra y actualiza la siguiente información:

**Productos**

CantidadAlterna = Cantidad Modificada

CantidadAlternaOriginal = Cantidad Calculada

Si <el usuario NO modifica la Cantidad calculada>

El sistema actualiza el objeto de Productos con la siguiente información para el producto actual:

**Productos**

CantidadAlterna= Cantidad Calculada

* + 1. Si <ManejoDobleUnidad = 0 “Falso”>
       1. El sistema actualiza el objeto Productos con la siguiente información para el producto actual:
          1. Productos

Cantidad (formatear la cantidad mostrada de acuerdo al número de decimales configurado para el producto <Producto.DecimalProducto>)

* + 1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Inventario Móvil – CUROLGEN13**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN13_AdministrarInventario.doc)
       1. El sistema envía como parámetro la siguiente información:
          1. ValidarExistencia

ProductoClave = ProductoClave (objeto Productos)

TipoUnidad = Unidad (objeto Productos)

Cantidad = Cantidad (objeto Productos)

CantidadAnterior = <TransProdDetalle.Cantidad>

TipoMovimiento = <TransProd.TipoMovimiento> registrado en sesión (objeto TransProd)

TipoTransProd = <TransProd.Tipo> registrado en sesión (objeto TransProd)

* + - 1. El sistema recibe la siguiente información:
         1. OP\_Verdadero/Falso
         2. OP\_Cantidad
    1. Si <no se tiene suficiente existencia disponible para sacar del inventario la diferencia entre la cantidad anterior y la nueva cantidad, en caso de que ésta última sea mayor (OP\_Falso)>
       1. El sistema presenta la siguiente información:
          1. Cantidad = <TransProdDetalle.Cantidad> + OP\_Cantidad (formatear la cantidad de acuerdo al número de decimales configurado para el producto <Producto.DecimalProducto>, posicionar el cursor dentro del combo de selección de cantidad de producto)
       2. El sistema continúa en el [**paso 10.5**](#PasoActorModificaCantidadYAgregar) del flujo alterno opcional[**AO01 Crear Venta a Consignación**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV31_CapturarMovSinInvFueraVisita.docx#_AO01_Crear_Movimiento)
    2. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Inventario Móvil – CUROLGEN13**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN13_AdministrarInventario.doc)
       1. El sistema envía como parámetro la siguiente información:
          1. ActualizarInventario

ProductoClave = <TransProdDetalle.ProductoClave>

TipoUnidad = <TransProdDetalle.TipoUnidad>

Cantidad = <TransProdDetalle.Cantidad>

TipoTransProd = <TransProd.Tipo> registrado en sesión (objeto TransProd)

TipoMovimiento = <TransProd.TipoMovimiento> registrado en sesión (objeto TransProd)

AlmacenID = AlmacenID registrado en sesión

Cancelar = 1

* + - 1. El sistema recibe la siguiente información:
         1. OP\_Verdadero
    1. Si <al cliente se le debe realizar el manejo del préstamo de envase <Cliente.Prestamo = 1 donde Cliente.ClienteClave = TransProd.ClienteClave (objeto TransProd registrado en sesión)>
       1. Si <el producto modificado se puede prestar <ProductoDetalle.ProductoClave = ProductoClave (objeto Productos)> y <ProductoDetalle.PRUTipoUnidad = Unidad (objeto Productos)> y <ProductoDetalle.Prestamo = 1>>
          1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Préstamo de Producto – CUROLGEN15**](file:///C:\Users\Belem\AppData\Roaming\Microsoft\ECU_GEN\CUROLGEN15_AdministrarPrestamoDeProducto.docx)y le envía como parámetro la siguiente información:

ActualizarPrestamo

ClienteClave = <TransProd.ClienteClave> (objeto TransProd registrado en sesión)

ProductoClave = <ProductoDetalle.ProductoDetClave> donde <ProductoDetalle.ProductoClave = TransProdDetalle.ProductoClave y ProductoDetalle.PRUTipoUnidad = TransProdDetalle.TipoUnidad>

TipoTransProd = <TransProd.Tipo> (objeto TransProd registrado en sesión)

Cantidad = <TransProdDetalle.Cantidad> \* -1

TipoUnidad = <TransProdDetalle.TipoUnidad>

Vendido = 0

El sistema recibe la siguiente información:

OP\_Verdadero

* + 1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Inventario Móvil – CUROLGEN13**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN13_AdministrarInventario.doc)
       1. El sistema envía como parámetro la siguiente información:
          1. ActualizarInventario

ProductoClave = ProductoClave (objeto Productos)

TipoUnidad = Unidad (objeto Productos)

Cantidad = Cantidad (objeto Productos)

TipoTransProd = <TransProd.Tipo> registrado en sesión (objeto TransProd)

TipoMovimiento = <TransProd.TipoMovimiento> registrado en sesión (objeto TransProd)

AlmacenID = AlmacenID registrado en sesión

* + - 1. El sistema recibe la siguiente información:
         1. OP\_Verdadero
    1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Detalle de la Transacción – CUROLGEN06**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN06_AdministrarDetalleTransaccion.doc)
       1. El sistema envía como parámetro la siguiente información:
          1. EliminarDetalleTransacción

TransProdId = <TransProd.TransProdID> registrado en sesión (objeto TransProd)

TransProdDetalleId = <TransProdDetalle.TransProdDetalleID> registrado en sesión (ArregloTransProdDetalle)

* + - 1. El sistema recibe como parámetro la siguiente información:
         1. Verdadero
    1. El sistema elimina del arreglo la información del objeto correspondiente al detalle del producto:
       1. **ArregloTransProdDetalle**
          1. TransProdDetalle
    2. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Detalle de la Transacción – CUROLGEN06**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN06_AdministrarDetalleTransaccion.doc)
       1. El sistema envía como parámetro la siguiente información:
          1. GenerarDetalleVentaConsignacion

TransProdId = <TransProd.TransProdID> registrado en sesión (objeto TransProd)

ProductoClave = ProductoClave (objeto Productos)

TipoUnidad = Unidad (objeto Productos)

Cantidad = Cantidad (objeto Productos)

CantidadOriginal = 0

PrecioClave = <TransProd.PCEPrecioClave> (objeto TransProd)

* + - 1. El sistema recibe como parámetro la siguiente información:
         1. PS\_TransProdDetalle
         2. PS\_Null
      2. Si <ocurrió alguna excepción durante la generación del encabezado de la transacción (PS\_Null)>
         1. El sistema continúa en el [**paso 6.1**](#PasoPresentaSolicitaProducto) del flujo alterno opcional[**AO01 Crear Venta a Consignación**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV31_CapturarMovSinInvFueraVisita.docx#_AO01_Crear_Movimiento)
      3. El sistema registra la información del objeto recibido dentro de un arreglo:
         1. **ArregloTransProdDetalle**

TransProdDetalle

* + 1. El sistema obtiene la siguiente información del objeto recibido:
       1. **ArregloTransProdDetalle**
          1. TransProdDetalle

Subtotal

Impuesto

Total

* + 1. El sistema actualiza el objeto Productos con la siguiente información para el producto actual:
       1. Productos
          1. Subtotal
          2. Impuesto
          3. Total
    2. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Transacción – CUROLGEN05**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN05_AdministrarTransaccion.doc)
       1. El sistema envía como parámetro la siguiente información:
          1. CalcularTotalesTransaccion

TransProd = <TransProd.TransProdID> registrado en sesión (objeto TransProd)

* + - 1. El sistema recibe la siguiente información
         1. TransProd
      2. El sistema actualiza la información del objeto registrado en sesión:
         1. TransProd
    1. Si <ManejoDobleUnidad = 1 “Verdadero”>
       1. El sistema registra la siguiente información:
          1. **TPDDatosExtra**

TransProdId (Obtenido de ArregloTransProdDetalle)

TransProdDetalleID (Obtenido de ArregloTransProdDetalle)

UnidadAlterna (Objeto Productos)

Cantidad (Objeto Productos)

CantidadAlterna (Objeto Productos)

CantidadAlternaOriginal (Objeto Productos \*\*En caso de aplicar)

MFechaHora = Fecha y Hora Actual del Sistema

MUsuarioID = USUId registrado en sesión

* + 1. Si <al cliente se le debe realizar el manejo del préstamo de envase <Cliente.Prestamo = 1 donde Cliente.ClienteClave = TransProd.ClienteClave (objeto TransProd registrado en sesión)>
       1. Si <el producto modificado se puede prestar <ProductoDetalle.ProductoClave = ProductoClave (objeto Productos)> y <ProductoDetalle.PRUTipoUnidad = Unidad (objeto Productos)> y <ProductoDetalle.Prestamo = 1>>
          1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Préstamo de Producto – CUROLGEN15**](file:///C:\Users\Belem\AppData\Roaming\Microsoft\ECU_GEN\CUROLGEN15_AdministrarPrestamoDeProducto.docx)y le envía como parámetro la siguiente información:

ActualizarPrestamo

ClienteClave = <TransProd.ClienteClave> (objeto TransProd registrado en sesión)

ProductoClave = <ProductoDetalle.ProductoDetClave> donde <ProductoDetalle.ProductoClave = ProductoClave (objeto Productos) y ProductoDetalle.PRUTipoUnidad = Unidad (objeto Productos)>

TipoTransProd = <TransProd.Tipo> (objeto TransProd registrado en sesión)

Cantidad = Cantidad (objeto Productos)

TipoUnidad = Unidad (objeto Productos)

Vendido = 0

El sistema recibe la siguiente información:

OP\_Verdadero

* + 1. El sistema continúa en el [**paso 7**](file:///C:\Duxstar\Productos\eRoute\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV44_CapturarVentasConsignación.docx#PasoParaCadaProductoObjetoProductos)del flujo alterno opcional[**AO01 Crear Venta a Consignación**](#_AO01_Crear_Movimiento)

1. Si <el actor **selecciona** **sostenidamente** un producto>
   1. El sistema obtiene del objeto Productos la siguiente información del producto seleccionado:
      1. ProductoClave
      2. Descripcion
      3. SubEmpresaID
      4. DecimalProducto
      5. Unidad
      6. UnidadAlterna (\*\*En caso de aplicar)
      7. Cantidad
      8. CantidadAlterna (\*\*En caso de aplicar)
      9. Precio
      10. Subtotal
      11. Impuesto
      12. Total
   2. El sistema recupera la siguiente información del producto:
      1. **ArregloTransProdDetalle**
         1. TransProdDetalle
   3. El sistema presenta la siguiente opción:
      1. Eliminar
   4. Si <el actor selecciona la opción **Eliminar**>
      1. El sistema presenta el mensaje “[P0233] El producto será eliminado de la lista. ¿Desea Continuar?”
         1. Si <el actor selecciona la opción **No**>
            1. El sistema continúa en el paso donde fue invocado
         2. Si <el actor selecciona la opción **Si**>
            1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Inventario Móvil – CUROLGEN13**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN13_AdministrarInventario.doc)

El sistema envía como parámetro la siguiente información:

ActualizarInventario

ProductoClave = ProductoClave (objeto Productos)

TipoUnidad = Unidad (objeto Productos)

Cantidad = Cantidad (objeto Productos)

TipoTransProd = <TransProd.Tipo> registrado en sesión (objeto TransProd)

TipoMovimiento = <TransProd.TipoMovimiento> registrado en sesión (objeto TransProd)

AlmacenID = AlmacenID registrado en sesión

Cancelar = 1

El sistema recibe la siguiente información:

OP\_Verdadero

* + - * 1. Si <al cliente se le debe realizar el manejo del préstamo de envase <Cliente.Prestamo = 1 donde Cliente.ClienteClave = TransProd.ClienteClave (objeto TransProd registrado en sesión)>

Si <el producto modificado se puede prestar <ProductoDetalle.ProductoClave = ProductoClave (objeto Productos)> y <ProductoDetalle.PRUTipoUnidad = Unidad (objeto Productos)> y <ProductoDetalle.Prestamo = 1>>

El caso de uso incluye la funcionalidad del caso de uso [**Administrar Préstamo de Producto – CUROLGEN15**](file:///C:\Users\Belem\AppData\Roaming\Microsoft\ECU_GEN\CUROLGEN15_AdministrarPrestamoDeProducto.docx)y le envía como parámetro la siguiente información:

ActualizarPrestamo

ClienteClave = <TransProd.ClienteClave> (objeto TransProd registrado en sesión)

ProductoClave = <ProductoDetalle.ProductoDetClave> donde <ProductoDetalle.ProductoClave = TransProdDetalle.ProductoClave y ProductoDetalle.PRUTipoUnidad = TransProdDetalle.TipoUnidad>

TipoTransProd = <TransProd.Tipo> (objeto TransProd registrado en sesión)

Cantidad = <TransProdDetalle.Cantidad> \* -1

TipoUnidad = <TransProdDetalle.TipoUnidad>

Vendido = 0

El sistema recibe la siguiente información:

OP\_Verdadero

* + - * 1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Detalle de la Transacción – CUROLGEN06**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN06_AdministrarDetalleTransaccion.doc)

El sistema envía como parámetro la siguiente información:

EliminarDetalleTransacción

TransProdId = <TransProd.TransProdID> registrado en sesión (objeto TransProd)

TransProdDetalleId = <TransProdDetalle.TransProdDetalleID> registrado en sesión (ArregloTransProdDetalle)

El sistema recibe como parámetro la siguiente información:

Verdadero

* + - * 1. El sistema elimina del arreglo la información del objeto correspondiente al producto:

**ArregloTransProdDetalle**

TransProdDetalle

Si <ManejoDobleUnidad = 1 “Verdadero”>

TPDDatosExtra

TransProdID = <ArregloTransProdDetalle.TransProdDetalleID>

TransProdDetalleID

* + - * 1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Transacción – CUROLGEN05**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN05_AdministrarTransaccion.doc)

El sistema envía como parámetro la siguiente información:

CalcularTotalesTransaccion

TransProd de acuerdo con la regla de negocio [**RNGEN171 Información en Sesión**](file:///C:\Amesol\Productos\Route\Trunk\Analisis\EspecificacionRequerimientos\General\ERM_Reglas_de_Negocio.docx#RNGEN171) y [**RNROLMOV195 Información Contenida en el Objeto del Encabezado de la Transacción**](file:///C:\Amesol\Productos\Route\Trunk\Analisis\EspecificacionRequerimientos\General\ERM_Reglas_de_Negocio.docx#RNROLMOV195)

El sistema recibe la siguiente información

TransProd

El sistema actualiza la información del objeto registrado en sesión:

TransProd

* + - * 1. El sistema elimina del objeto Productos la siguiente información del producto:

Productos

ProductoClave

Descripcion

SubEmpresaID

DecimalProducto

Unidad

UnidadAlterna (\*\*En caso de aplicar)

Cantidad

CantidadAlterna (\*\*En caso de aplicar)

Precio

Subtotal

Impuesto

Total

* + - * 1. El sistema continúa en el [**paso 7**](file:///C:\Duxstar\Productos\eRoute\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV44_CapturarVentasConsignación.docx#PasoParaCadaProductoObjetoProductos) del flujo alterno opcional[**AO01 Crear Venta a Consignación**](#_AO01_Crear_Movimiento)

1. Si <el actor selecciona la opción **Buscar**>
   1. Si <el parámetro de configuración indica que se tiene activado el Manejo de Doble Unidad (ManejoDobleUnidad = 1) de acuerdo con la información registrada en sesión>
      1. El caso de uso incluye la funcionalidad del caso de uso [**Buscar Productos – CUROLMOV20**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV20_BuscarProductos.docx)
         1. El sistema envía como parámetro la siguiente información:
            1. Cadena = (Cadena vacía)
            2. Esquema = Todos
            3. TipoValidarExistencia = 1
            4. ListaPrecios = <TransProd.PCEPrecioClave> registrado en sesión (objeto TransProd)
            5. TransProdID = <TransProd.TransProdID> registrado en sesión (objeto TransProd)
            6. TipoMovimiento = <TransProd.TipoMovimiento> registrado en sesión (objeto TransProd)
            7. TipoTransProd = <TransProd.Tipo> registrado en sesión (objeto TransProd)
            8. BusquedaSimple = 1
         2. El sistema recibe como parámetro la siguiente información:
            1. Productos

Clave del Producto

* + 1. El sistema continúa en el [**paso 9.3.3.1**](file:///C:\Duxstar\Productos\eRoute\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV44_CapturarVentasConsignación.docx#PasoSiProductoClaveoIDCoincide) del flujo alterno opcional[**AO01 Crear Venta a Consignación**](file:///C:\Duxstar\Productos\eRoute\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV44_CapturarVentasConsignación.docx#_AO01_Crear_Movimiento)
  1. El caso de uso incluye la funcionalidad del caso de uso [**Buscar Productos – CUROLMOV20**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV20_BuscarProductos.docx)
     1. El sistema envía como parámetro la siguiente información:
        1. Cadena = (Cadena vacía)
        2. Esquema = Todos
        3. TipoValidarExistencia = 1
        4. ListaPrecios = <TransProd.PCEPrecioClave> registrado en sesión (objeto TransProd)
        5. TransProdID = <TransProd.TransProdID> registrado en sesión (objeto TransProd)
        6. TipoMovimiento = <TransProd.TipoMovimiento> registrado en sesión (objeto TransProd)
        7. TipoTransProd = <TransProd.Tipo> registrado en sesión (objeto TransProd)
        8. BusquedaSimple = 0
     2. El sistema recibe como parámetro la siguiente información:
        1. Productos
           1. Clave del Producto
           2. Unidad
           3. Cantidad
           4. Precio
  2. El sistema continúa en el [**paso 9.3.2.4**](#PasoParaCadaProductoRegresadoBusqueda)del flujo alterno opcional[**AO01 Crear Venta a Consignación**](#_AO01_Crear_Movimiento)

1. Si <el actor selecciona la opción **Continuar**>
   1. El sistema valida la información proporcionada por el usuario:
      1. Si <el actor no proporciona la información requerida>
         1. El sistema presenta el mensaje “[BE0001] El campo $0$ es requerido”, sustituir el valor de $0$ por el nombre del campo
         2. El sistema continúa en el punto correspondiente al campo que no se haya proporcionado
      2. Si <no se proporcionó por lo menos un producto para la transacción>
         1. El sistema presenta el mensaje “[E0044] Se debe asignar por lo menos un Producto a $0$”, sustituir el valor de $0$ por el nombre de la actividad actual
         2. El sistema continúa en el [**paso 6.1**](#PasoPresentaSolicitaProducto) del flujo alterno opcional[**AO01 Crear Venta a Consignación**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV31_CapturarMovSinInvFueraVisita.docx#_AO01_Crear_Movimiento)
   2. Si <el parámetro de configuración indica que se puede cancelar una venta a consignación por liquidación (CancConsigLiqui = 1) de acuerdo a la información registrada en sesión>
      1. El caso de uso incluye la funcionalidad del caso de uso [**Calcular Promociones – CUROLGEN03**](file:///C:\Users\Belem\AppData\Roaming\Microsoft\ECU_GEN\CUROLGEN03_CalcularPromociones.doc)
         1. El sistema envía como parámetro la siguiente información:
            1. Preparar
      2. El caso de uso incluye la funcionalidad del caso de uso [**Aplicar Promociones – CUROLMOV19**](file:///C:\Users\Belem\AppData\Roaming\Microsoft\Word\CUROLMOV19_Aplicar%20Promociones.doc)
         1. El sistema envía como parámetro la siguiente información:
            1. TransProd = TransProd registrado en sesión (objeto TransProd)
      3. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Transacción – CUROLGEN05**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN05_AdministrarTransaccion.doc)
         1. El sistema envía como parámetro la siguiente información:
            1. CalcularTotalesTransaccion

TransProd = <TransProd.TransProdID> registrado en sesión (objeto TransProd)

* + - 1. El sistema recibe la siguiente información
         1. TransProd
      2. El sistema actualiza la información del objeto registrado en sesión:
         1. TransProd
  1. El sistema continúa con el flujo alterno opcional **[AO02 Calcular Generales Venta a Consignación](#_AO02_Calcular_Generales)**

1. Si <el actor selecciona la opción **Regresar**>
   1. Si <se realizaron cambios (se agregaron o modificaron productos)>, el sistema presenta el mensaje “[BP0002] Se perderán los cambios. ¿Está seguro de regresar?”
      1. Si <el actor selecciona la opción **No**>
         1. El sistema continúa en el paso en donde fue invocado
      2. Si <el actor selecciona la opción **Si** >
         1. El sistema elimina la siguiente información de la transacción actual, realizando roll back a la misma:
            1. TransProd
            2. TransProdDetalle
            3. TPDDatosExtra (\*\*En caso de aplicar)
            4. TPDImpuesto
            5. TpdDes
            6. TpdDesVendedor
            7. TrpPrp
            8. Inventario
         2. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Clientes – CUROLGEN14**](../ECU_GEN/CUROLGEN14_AdministrarClientes.doc)
            1. El sistema envía como parámetro la siguiente información:

ActualizarSaldoCteActual

Importe = <TransProd.Total> registrado en sesión (objeto TransProd)

* + - * 1. El sistema recibe la siguiente información:

Verdadero

* + - 1. El sistema continúa en el [**paso 7**](#PasoPresentaTitulo) del flujo básico
  1. Si <no se realizaron cambios (no se agregaron o modificaron productos)>
     1. Si <proviene del flujo alterno opcional[**AO01 Crear Venta a Consignación**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV31_CapturarMovSinInvFueraVisita.docx#_AO01_Crear_Movimiento)>
        1. El sistema elimina la siguiente información de la transacción generada, realizando roll back a la misma:
           1. TransProd
        2. El sistema continúa en el [**paso 7**](#PasoPresentaTitulo) del flujo básico
     2. Si <proviene de los flujos alternos opcionales [**AO03 Modificar Venta a Consignación**](#_AO02_Modificar_Movimiento) o [**AO04 Eliminar Venta a Consignación**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV44_CapturarVentasConsignación.docx#_AO03_Eliminar_Venta)>
        1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Clientes – CUROLGEN14**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN14_AdministrarClientes.doc)
           1. El sistema envía como parámetro la siguiente información:

ActualizarSaldoCteActual

Importe = <TransProd.Total> registrado en sesión (objeto TransProd)

* + - * 1. El sistema recibe la siguiente información:

Verdadero

* + - 1. El sistema continúa en el [**paso**](#paso7) **7** del flujo básico

### [AO02 Calcular Generales Venta a Consignación](#AO02_r)

1. El sistema presenta el siguiente título de pantalla:
   1. Generales
   2. Fecha: Se presenta la fecha del día de trabajo actual (DiaClave), a partir de la información registrada en sesión.
2. El sistema presenta y solicita la siguiente información:
   1. **Folio:** <TransProd.Folio> registrado en sesión (objeto TransProd). Mostrar de sólo lectura.
   2. **Fase:** Se presenta la descripción de la fase de la transacción, para ello ir a <VAVDescripcion> y obtener <VAVDescripcion.Descripcion> donde <VAVDescripcion.VARCodigo = ‘TRPFSE> y <VAVDescripcion.VAVClave = TransProd.TipoFase > (objeto TransProd registrado en sesión), de acuerdo al tipo de lenguaje configurado para el sistema. Mostrar de sólo lectura.
   3. **Fecha:** <TransProd.FechaCaptura> registrado en sesión (objeto TransProd). Mostrar de sólo lectura en formato dd/mm/aaaa.
   4. **Lista Precios:** <TransProd.PCEPrecioClave> registrado en sesión (objeto TransProd). Mostrar de sólo lectura.
   5. **Subtotal:** <TransProd.Subtotal> registrado en sesión (objeto TransProd). Mostrar de sólo lectura, precedido del signo “$”.
   6. **Impuesto:** <TransProd.Impuesto> registrado en sesión (objeto TransProd). Mostrar de sólo lectura, precedido del signo “$”.
   7. **Total:** <TransProd.Total> registrado en sesión (objeto TransProd). Mostrar de sólo lectura, precedido del signo “$”.
   8. **Comentarios:** Permitir capturar notas o comentarios para la venta a consignación.
   9. **Pedido Adicional:** Permite capturar el folio del pedido adicional. Si <Cliente.ExigirPedidoAdicional=1> y además <Cliente.FormatoPedidoAdicional <>”” o Null> verificar que el formato del campo capturado si es correcto (Ver sección: “[Formato Pedido Adicional](#Formato)”), y cumple con la mascara definida en <Cliente.FormatoPedidoAdicional, si no cumple el sistema presenta el mensaje: “[ME0593] El formato para el campo $0$ debe ser $1$“, sustituir $0$ por el nombre del campo Pedido Adicional y $1$ sustituirlo por la mascada contenida en <Cliente.FormatoPedidoAdicional>. Si la información capturada en el campo Pedido Adicional cumple con el formato asignar a <TRPVtaAcreditada.PedidoAdicional> para el registro en el sistema (ESTE CAMPO SOLO APARECERÁ SI SE TIENE ACTIVADO EL PARÁMETRO <Cliente.ExigirPedidoAdicional=1> )

**Formato Pedido Adicional**

Para cada carácter capturado en el campo Pedido Adicional, verificar que cumplan con las restricciones (longitud, letra, número, delimitador) definidas en <Cliente.FormatoPedidoAdicional>, validando los datos de entrada dependiendo del siguiente formato:

A = Puede ser cualquier letra en mayúscula

9 = Puede ser cualquier número

X = Puede ser cualquier letra mayúscula o número

a = Puede ser cualquier letra mayúscula o minúscula

x = Puede ser cualquier letra mayúscula o minúscula o número

z = Puede ser cualquier carácter

Cualquier otro carácter fuera de los aquí definidos se considerará un delimitador

1. Si <el actor selecciona la opción **Terminar**>
   1. Si la configuración del cliente indica que es requerido el campo de pedido adicional <Cliente.ExigirPedidoAdicional=1 “Activo”>, verificar si el campo está vacío < TRPVtaAcreditada.PedidoAdicional =’‘ OR NULL>, si es así, mandar el mensaje de error [BE0001] “El campo $0$ es requerido”, sustituir la variable $0$ por la descripción del campo y posicionarlo en el campo de Pedido Adicional
   2. Si <al cliente se le debe realizar el manejo del préstamo de envase <Cliente.Prestamo = 1> (objeto Cliente registrado en sesión)> y Si <el parámetro de configuración indica que al cliente se le debe validar el límite de préstamo de envase <Cliente.ValidarLimEnvase = 1> (objeto Cliente registrado en sesión)>
      1. ***Si en los detalles de la venta a consignación se incluye por lo menos un producto que se pueda prestar, es decir:*** Si <TransProdDetalle.ProductoClave = ProductoDetalle.ProductoClave> y <TransProdDetalle.TipoUnidad = ProductoDetalle.PRUTipoUnidad> donde <ProductoDetalle.Prestamo = 1>
         1. El sistema obtiene y registra en sesión el límite de envase establecido para el cliente, donde <Cliente.ClienteClave = Cliente Actual>:
            1. **Cliente**

LimiteEnvase

* + - 1. El sistema obtiene y registra en sesión el Saldo General de Envase del Cliente, a partir de la sumatoria del Saldo de todos los productos que el cliente tiene en préstamo:
         1. ***SaldoGeneralEnvase =*** <∑ProductoPrestamoCli.Saldo> donde <ProductoPrestamoCli.ClienteClave = Cliente Actual>
      2. ***Si el Saldo General de Envase del Cliente excede al Límite de Envase Establecido para el Cliente, es decir:*** Si <*SaldoGeneralEnvase* **ES MAYOR QUE** Cliente.LimiteEnvase>
         1. El sistema presenta el mensaje “[E0917] No se permite continuar, debido a que se ha excedido el Límite de Préstamo de envase establecido”
         2. El sistema continúa en el paso donde fue invocado, de manera que se pueda regresar a modificar la venta a consignación
      3. ***Si el Saldo General de Envase del Cliente NO excede al Límite de Envase Establecido para el Cliente, es decir:*** Si <*SaldoGeneralEnvase* **ES MENOR QUE** Cliente.LimiteEnvase>
         1. El sistema obtiene y registra en sesión el porcentaje que detona el mensaje de advertencia para indicar que se está llegando al límite del préstamo de envase:

**MOTConfiguracion**

PorLimEnvase

* + - * 1. El sistema obtiene y registra en sesión el Porcentaje del Límite de Envase en el que se encuentra el Cliente actualmente, a partir de:

***PorcentajeLimiteEnvase*** = (*SaldoGeneralEnvase* \* 100) / <Cliente.LimiteEnvase>

* + - * 1. ***Si el Porcentaje del Límite de Envase en el que se encuentra el Cliente actualmente es igual o mayor al porcentaje configurado para detonar el mensaje de advertencia de que se está llegando al límite del préstamo de envase, es decir:*** Si <*PorcentajeLimiteEnvase* **ES MAYOR O IGUAL A** MOTConfiguracion.PorLimEnvase>

El sistema presenta el mensaje “[I0276] “El Límite de Préstamo de Envase está por cumplirse, se tiene un $0$, con respecto a lo establecido”, sustituir el valor de $0$ por el valor de *PorcentajeLimiteEnvase* + “%”

El sistema continúa en el [**paso 3.2**](#paso3_2_AO02) del flujo alterno opcional [**AO02 Calcular Generales Venta a Consignación**](#_AO02_Calcular_Generales)

* + 1. El sistema continúa en el [**paso 3.2**](#paso3_2_AO02) del flujo alterno opcional [**AO02 Calcular Generales Venta a Consignación**](#_AO02_Calcular_Generales)
  1. Si <el parámetro de configuración indica que se debe realizar la validación del límite de crédito en la actividad de Consignación (TipoLimiteCredito = 2 “Pedido/Consignación” ó 4 “Pedido/Consignación(sólo mensaje)”) de acuerdo a la información registrada en sesión>
     1. El sistema obtiene y registra en sesión la siguiente información, donde <TransProd.ClienteClave (objeto TransProd registrado en sesión) = CLIFormaVenta.ClienteClave y CLIFormaVenta.CFVTipo = 2>:
        1. **CLIFormaVenta**
           1. ValidaLimite
           2. LimiteCredito
           3. DiasCredito
     2. Si <la configuración de la forma de venta del cliente indica que se debe validar el límite de crédito <CLIFormaVenta.ValidaLimite = 1>>
        1. Si <el módulo actual es el de Venta o Reparto (ModuloTermTipoIndice = 1 ó 3) de acuerdo a la información registrada en sesión>
           1. Si <el parámetro de configuración indica que la cobranza se realiza sobre Ventas (TipoCobranza = 1)> o si <el parámetro de configuración indica que la cobranza se realiza sobre Ambas (TipoCobranza = 2) y la configuración del cliente indica que la cobranza se le realiza sobre Ventas <Cliente.TipoFiscal = 1 donde Cliente.ClienteClave = TransProd.ClienteClave (objeto TransProd registrado en sesión)>>

El sistema obtiene el importe total del saldo de las ventas a crédito y ventas a consignación realizadas al cliente actual:

***SaldoCliente*** = <∑TransProd.Saldo> donde <TransProd.VisitaClave = Visita.VisitaClave> y <TransProd.DiaClave = Visita.DiaClave> y <TransProd.ClienteClave = Visita.ClienteClave> y <(TransProd.Tipo = 1 y TransProd.CFVTipo = 2) o (TransProd.Tipo = 24)> y <TransProd.TipoFase <> 0>.

El sistema obtiene el importe total del saldo deudor del cliente, incluyendo el total de la venta a consignación actual:

***SaldoCliente*** = *SaldoCliente* + <TransProd.Total> (objeto TransProd registrado en sesión) – TotalInicialCredito.

* + - * 1. Si <el parámetro de configuración indica que la cobranza se realiza sobre Facturas (TipoCobranza = 0)> o si <el parámetro de configuración indica que la cobranza se realiza sobre Ambas (TipoCobranza = 2) y la configuración del cliente indica que la cobranza se le realiza sobre Facturas <Cliente.TipoFiscal = 2 donde Cliente.ClienteClave = TransProd.ClienteClave (objeto TransProd registrado en sesión)>>

El sistema obtiene el importe total del saldo de las facturas y ventas a crédito, y de las ventas a consignación realizadas al cliente actual:

***SaldoCliente*** = <∑TransProd.Saldo> donde <TransProd.VisitaClave = Visita.VisitaClave> y <TransProd.DiaClave = Visita.DiaClave> y <TransProd.ClienteClave = Visita.ClienteClave> y <(TransProd.Tipo = 8 y TransProd.CFVTipo = 2) o (TransProd.Tipo = 24)> y <TransProd.TipoFase <> 0>.

***SaldoCliente*** = *SaldoCliente* + <∑TransProd.Saldo> donde <TransProd.VisitaClave = Visita.VisitaClave> y <TransProd.DiaClave = Visita.DiaClave> y <TransProd.ClienteClave = Visita.ClienteClave> y <TransProd.Tipo = 1> y <TransProd.CFVTipo = 2> y <TransProd.TipoFase <> 0> y <TransProd.FacturaId = Null>.

El sistema obtiene el importe total del saldo deudor del cliente, incluyendo el total de la venta a consignación actual:

***SaldoCliente*** = *SaldoCliente* + <TransProd.Total> (objeto TransProd registrado en sesión) – TotalInicialCredito.

* + - 1. Si <el módulo actual es el de Preventa (ModuloTermTipoIndice = 2) de acuerdo a la información registrada en sesión>
         1. Si <el parámetro de configuración indica que la cobranza se realiza sobre Ventas (TipoCobranza = 1)> o si <el parámetro de configuración indica que la cobranza se realiza sobre Ambas (TipoCobranza = 2) y la configuración del cliente indica que la cobranza se le realiza sobre Ventas <Cliente.TipoFiscal = 1 donde Cliente.ClienteClave = TransProd.ClienteClave (objeto TransProd registrado en sesión)>>

El sistema obtiene el importe total del saldo de las ventas a crédito y ventas a consignación realizadas al cliente actual, más el saldo del cliente:

***SaldoCliente*** = <∑TransProd.Total> donde <TransProd.VisitaClave = Visita.VisitaClave> y <TransProd.DiaClave = Visita.DiaClave> y <TransProd.ClienteClave = Visita.ClienteClave> y <(TransProd.Tipo = 1 y TransProd.CFVTipo = 2) o (TransProd.Tipo = 24)> y <TransProd.TipoFase = 1>.

***SaldoCliente*** = *SaldoCliente* + <Cliente.SaldoEfectivo donde Cliente.ClienteClave = TransProd.ClienteClave (objeto TransProd registrado en sesión)>

El sistema obtiene el importe total del saldo deudor del cliente, incluyendo el total de la venta a consignación actual:

***SaldoCliente*** = *SaldoCliente* + <TransProd.Total> (objeto TransProd registrado en sesión) – TotalInicialCredito.

* + - 1. Si <la configuración del cliente indica que el pago con cheque NO afecta el saldo <Cliente.ActualizaSaldoCheque = 0 donde Cliente.ClienteClave = TransProd.ClienteClave (objeto TransProd registrado en sesión)>>
         1. El sistema registra en sesión la siguiente información:

TipoTrp = Cadena vacía

* + - * 1. Si <el parámetro de configuración indica que la cobranza se realiza sobre Ventas (TipoCobranza = 1)> o si <el parámetro de configuración indica que la cobranza se realiza sobre Ambas (TipoCobranza = 2) y la configuración del cliente indica que la cobranza se le realiza sobre Ventas <Cliente.TipoFiscal = 1 donde Cliente.ClienteClave = TransProd.ClienteClave (objeto TransProd registrado en sesión)>>

TipoTrp = 1

* + - * 1. Si <el parámetro de configuración indica que la cobranza se realiza sobre Facturas (TipoCobranza = 0)> o si <el parámetro de configuración indica que la cobranza se realiza sobre Ambas (TipoCobranza = 2) y la configuración del cliente indica que la cobranza se le realiza sobre Facturas <Cliente.TipoFiscal = 2 donde Cliente.ClienteClave = TransProd.ClienteClave (objeto TransProd registrado en sesión)>>

TipoTrp = 8

* + - * 1. El sistema obtiene el importe total de los abonos del cliente realizados con cheque sobre las ventas o facturas a crédito (según corresponda al tipo de cobranza) y sobre las ventas a consignación realizadas:

***AbonosCheque*** = <∑ABNDetalle.Importe> donde <ABNDetalle.ABNId = AbnTrp.ABNId> y <AbnTrp.TransProdID = TransProd.TransProdId> y <TransProd.VisitaClave = Visita.VistaClave> y <TransProd.DiaClave = Visita.DiaClave> donde <ABNDetalle.TipoPago = 2 “Cheque”> y <TransProd.ClienteClave = Visita.ClienteClave> y <TransProd.CFVTipo = 2 “Crédito”> y <(TransProd.Tipo = TipoTrp registrado en sesión) o (TransProd.Tipo = 24)> y <TransProd.TipoFase <> 0>.

* + - * 1. El sistema obtiene el importe total del saldo deudor del cliente, incluyendo el importe de los abonos realizados con cheque:

***SaldoCliente*** = *SaldoCliente* +*AbonosCheque.*

* + - * 1. El sistema obtiene el importe total del saldo de los abonos del cliente NO realizados con cheque:

***SaldoAbonos*** = <∑Abono.Saldo> donde <Abono.VisitaClave = Visita.VistaClave> y <Abono.DiaClave = Visita.DiaClave> donde <TransProd.ClienteClave registrado en sesión = Visita.ClienteClave> y <Abono.ABNId <> Abono.ABNId (donde Abono.ABNId = ABNDetalle.ABNId y Abono.VisitaClave = Visita.VisitaClave y Abono.DiaClave = Visita.DiaClave y TransProd.ClienteClave registrado en sesión = Visita.ClienteClave y ABNDetalle.TipoPago = 2 “Cheque”)>.

* + - * 1. El sistema obtiene el importe total del saldo deudor del cliente, excluyendo el importe del saldo de los abonos NO realizados con cheque:

***SaldoCliente*** = *SaldoCliente* – *SaldoAbonos.*

* + - 1. Si <la configuración del cliente indica que el pago con cheque SI afecta el saldo <Cliente.ActualizaSaldoCheque = 1 donde Cliente.ClienteClave = TransProd.ClienteClave (objeto TransProd registrado en sesión)>>
         1. El sistema obtiene el importe total del saldo de los abonos del cliente:

***SaldoAbonos*** = <∑Abono.Saldo> donde <Abono.VisitaClave = Visita.VistaClave> y <Abono.DiaClave = Visita.DiaClave> donde <TransProd.ClienteClave registrado en sesión = Visita.ClienteClave>.

* + - * 1. El sistema obtiene el importe total del saldo deudor del cliente, excluyendo el importe del saldo de los abonos realizados:

***SaldoCliente*** = *SaldoCliente* – *SaldoAbonos.*

* + - 1. El sistema obtiene la siguiente información:
         1. ***SaldoCliente*** = LimiteCredito (registrado en sesión) – *SaldoCliente*
      2. Si <el saldo del cliente excede su límite de crédito (*SaldoCliente* < 0)>
         1. El sistema obtiene y registra en sesión el porcentaje de riesgo de la empresa:

**CONHist**

PorcentajeRiesgo

* + - * 1. El sistema obtiene el importe por el que podría excederse el cliente, de acuerdo a su límite de crédito y al porcentaje de riesgo de la empresa:

***Riesgo***  = [LimiteCredito (registrado en sesión) \* PorcentajeRiesgo (registrado en sesión)] / 100.

* + - * 1. Si <el saldo del cliente excede el riesgo (*SaldoCliente* (valor absoluto)> *Riesgo*)>

Si <el parámetro de configuración indica que se debe realizar la validación del límite de crédito en la actividad de Consignación (TipoLimiteCredito = 2 “Pedido/Consignación”) de acuerdo a la información registrada en sesión>

El sistema presenta el mensaje “[E0737] No se puede Terminar la Consignación, ha Excedido el Límite de Crédito del Cliente en $0$”, sustituir el valor de $0$ por el valor absoluto de *SaldoCliente*.

El sistema continúa en el paso en donde fue invocado

Si <el parámetro de configuración indica que se debe realizar la validación del límite de crédito en la actividad de Consignación (TipoLimiteCredito = 4 “Pedido/Consignación(sólo mensaje)”) de acuerdo a la información registrada en sesión>

El sistema presenta el mensaje “[I0266] Este movimiento está sujeto a aprobación debido a que el cliente ha excedido su límite de crédito en $0$”, sustituir el valor de $0$ por el valor absoluto de *SaldoCliente*.

El sistema continúa en el [**paso 3.2**](#paso3_2_AO02) del flujo alterno opcional[**AO02 Calcular Generales Venta a Consignación**](#_AO02_Calcular_Generales)

* 1. El sistema actualiza la siguiente información para la transacción registrada en sesión <TransProd.TransProdID = TransProd.TransProdID registrado en sesión (objeto TransProd)>:
     1. **TransProd**
        1. Si <el cliente actual SI tiene configurada la forma de venta a “Crédito” <CLIFormaVenta.CFVTipo = 2 donde CLIFormaVenta.ClienteClave = ClienteClave registrado en sesión>
           1. DiasCredito = DiasCredito registrado en sesión
        2. Notas = Contenido del campo “Comentarios”.
        3. Saldo = <TransProd.Total> registrado en sesión (objeto TransProd)
        4. MFechaHora = Fecha y hora actual del sistema
        5. MUsuarioID = UsuarioId registrado en sesión
     2. **TRPVtaAcreditada**
        1. TransProdId = TransProdId del Objeto TransProd
        2. PedidoAdicional = Contenido del campo “Pedido Adicional”.
        3. MFechaHora = Fecha y hora actual del sistema
        4. MUsuarioID = UsuarioId registrado en sesión
  2. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Clientes – CUROLGEN14**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN14_AdministrarClientes.doc)
     1. El sistema envía como parámetro la siguiente información:
        1. ActualizarSaldoCteActual
           1. Importe = <TransProd.Total> registrado en sesión (objeto TransProd)
     2. El sistema recibe la siguiente información:
        1. Verdadero
  3. Si <proviene del flujo alterno opcional[**AO01 Crear Venta a Consignación**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV31_CapturarMovSinInvFueraVisita.docx#_AO01_Crear_Movimiento)>
     1. El sistema realiza la confirmación del folio generado para la transacción
     2. Si <el Vendedor tiene activo el parámetro para validar el número de clientes distintos que pueden tener consignas abiertas <Vendedor.ValidaLimiteConsignas = 1>>
        1. Si <el cliente actual NO se encuentra en el listado de clientes registrados con consignas abiertas para el vendedor <VendedorConsigAb.ClienteClave *NO* *INCLUYE A* ClienteClave registrado en sesión>> y si <el límite de consignas del vendedor es mayor al número de clientes con consignas abiertas registradas para el vendedor <Vendedor.LimiteConsignas > *NumeroClientes* registrado en sesión>>
           1. El sistema registra la siguiente información, donde <VendedorConsigAb.VendedorID = VendedorId registrado en sesión>:

**VendedorConsigAb**

VendedorID = VendedorId registrado en sesión

ClienteClave = Clave del cliente actual (ClienteClave registrado en sesión)

MFechaHora = Fecha y hora actual del sistema

MUsuarioID = UsuarioId registrado en sesión

* 1. El sistema valida los permisos de usuario para firma:
     1. El caso de uso incluye la funcionalidad del caso de uso [**Validar Permisos de Usuario para Firma – CUROLMOV83**](file:///C:\Duxstar\Productos\eRoute\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV83_ValidarPermisosUsuarioFirma.docx)
        1. El sistema envía como parámetro la siguiente información:
           1. TipoIndice = <ModuloMovDetConsignación.TipoIndice> registrado en sesión (objeto ModuloMovDetConsignación)
           2. TransProdId = <TransProd.TransProdId> registrado en sesión (objeto TransProd)
        2. El sistema recibe como parámetro la siguiente información:
           1. Firma = Verdadero/Falso
     2. El sistema continúa en el siguiente paso de donde fue invocado
  2. El sistema realiza la confirmación de la transacción (Commit) correspondiente en el sistema:
     1. TransProd
     2. TransProdDetalle
     3. TPDDatosExtra (\*\*En caso de aplicar)
     4. TPDImpuesto
     5. TpdDes
     6. TpdDesVendedor
     7. TrpPrp
     8. Inventario
     9. TRPVtaAcreditada (\*\*en caso de aplicar)
     10. VendedorConsigAb (\*\*en caso de aplicar)
  3. El sistema obtiene la siguiente información:
     1. **MOTConfiguracion**
        1. MensajeImpresion
     2. **ConfigParametro**
        1. Parametro = NumImpresiones
        2. Identificador = ModuloMovDetalle Registrado en sesión
        3. Valor
  4. Si <el parámetro de configuración indica que se debe presentar el mensaje de impresión (MensajeImpresion = 1)>
     1. El sistema presenta el mensaje “[P0103] ¿Desea imprimir el ticket?”
        1. Si <el actor selecciona la opción **No**>
           1. El sistema continúa en el [**paso 1**](#FIN)**8** del flujo básico
        2. Si <el actor selecciona la opción **Si**>
           1. El sistema obtiene la siguiente información de la transacción registrada en sesión (objeto TransProd registrado en sesión):

**TransProd**

TransProdID

Tipo

Folio

FechaCaptura

Total

TipoFase

ClienteClave

DiaClave

* + - * 1. El sistema envía al dispositivo de impresión el recibo del movimiento, incluyendo la funcionalidad del caso de uso **Imprimir Recibos – RNGENXX** y enviando como parámetros la siguiente información:

ListaDocumentos

TransProdID = <TransProd.TransProdID>

Tipo = <TransProd.Tipo>

ValorReferencia = ‘TRPTIPO’

TipoRecibo = ‘TRP’

Folio = <TransProd.Folio>

DescTipo = (Cadena vacía)

Fecha = <TransProd.FechaCaptura>

TransProdTotal = <TransProd.Total>

TipoFase = <TransProd.TipoFase>

ClienteClave = <TransProd.ClienteClave>

DiaClave = <TransProd.DiaClave>

SubEmpresaId = Null

FactElect = 0

LogoSoloPrimerRecibo = 1

Impresiones = Impresiones + 1

* + - * 1. ~~El sistema continúa en el~~ [**~~paso 1~~**](#FIN)**~~8~~** ~~del flujo básico~~
        2. Si <el parámetro de número de impresiones <ConfigParametro.Parametro = NumImpresiones> se encuentra en 0 <ConfigParametro.Valor = 0> o no se encontró ningún valor para esta actividad>

El sistema continúa en el [paso 3.8.1](#PasoReimpresion) del flujo [AO02 Calcular Generales Venta a Consignación](#AO02)

* + - * 1. Si <Se encontró un parámetro para esta actividad y es mayor a cero “0”<ConfigParametro.Valor > 0>

Si <Impresiones < <ConfigParametro.Valor>>

El sistema continúa en el [paso 3.8.1.2.1](#PasoNumImpresiones) del flujo [AO02 Calcular Generales Venta a Consignación](#AO02)

El sistema continúa en el [**paso 1**](#FIN)**8** del flujo básico

* 1. Si <el parámetro de configuración indica que no se debe presentar el mensaje de impresión (MensajeImpresion = 0)>
     1. El sistema continúa en el [**paso 1**](#FIN)**8** del flujo básico
  2. Si <la configuración indica “Envío Correo” o “Envío SMS” <MOTConfiguracion.MensajeImpresion = 2 ó 3>
     1. El caso de uso incluye la funcionalidad del caso de uso [**Enviar Ticket por Correo o SMS – CUROLMOV85**](file:///C:\Duxstar\Productos\eRoute\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV85_EnviarTicketCorreoSMS.docx)
        1. El sistema envía como parámetro la siguiente información:
           1. TipoEnvio = <MOTConfiguracion.MensajeImpresion>
           2. TransProdId = <TransProd.TransProdID> registrado en sesión (objeto TransProd)
           3. ABNId = Null
        2. El sistema recibe como parámetro la siguiente información:
           1. EnvioTicket = Verdadero/Falso
     2. El sistema continúa en el siguiente paso de donde fue invocado

1. Si <el actor selecciona la opción **Regresar**>
   1. El sistema continúa en el [**paso 4**](#paso4_AO01) del flujo alterno opcional[**AO01 Crear Venta a Consignación**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV31_CapturarMovSinInvFueraVisita.docx#_AO01_Crear_Movimiento)

### [AO03 Modificar Venta a Consignación](#AO03_r)

1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Transacción – CUROLGEN05**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN05_AdministrarTransaccion.doc)
   1. El sistema envía como parámetro la siguiente información:
      1. ObtenerTransaccion
         1. TransProdID = TransProdID registrado en sesión
   2. El sistema recibe como parámetro la siguiente información
      1. TransProd
   3. El sistema registra en sesión la información del objeto recibido para el encabezado de la transacción:
      1. TransProd
2. El sistema actualiza en sesión la siguiente información:
   1. TotalInicialCredito = <TransProd.Total> registrado en sesión (objeto TransProd)
3. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Clientes – CUROLGEN14**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN14_AdministrarClientes.doc)
   1. El sistema envía como parámetro la siguiente información:
      1. ActualizarSaldoCteActual
         1. Importe = [<TransProd.Total> registrado en sesión (objeto TransProd)] \* -1
   2. El sistema recibe la siguiente información:
      1. Verdadero
4. El sistema presenta la siguiente información:
   1. Lista de Precios: Se presenta la información de la lista de precios, a través de la concatenación de la clave <Precio.ProductoClave donde TransProd.PCEPrecioClave (objetoTransProd) = Precio.ProductoClave> y nombre del mismo <Precio.Nombre>, separados por un guion medio, ejemplo: “0002 – Lista Especial”
5. El sistema presenta y solicita la siguiente información:
   1. Producto (\*)
   2. Si <ManejoDobleUnidad = 0 “Falso”>
      1. Unidad (\*) (c)
      2. Cantidad (\*)
   3. Si <ManejoDobleUnidad = 1 “Verdadero”>
      1. Si <Existe una unidad donde <ProductoUnidad.TipoEstado = Alterna> donde <ProductoUnidad.ProductoClave = Producto actual>
         1. Para cada Unidad de acuerdo al orden configurado <ConfigParametro.Valor>, se va a buscar la descripción correspondiente al Valor por Referencia UNIDADV, que coincida con los valores registrados en el parámetro y se muestran de acuerdo al orden de aparición
            1. <NombreUnidad>
            2. Cantidad (\*)
      2. Si <No Existe una unidad donde <ProductoUnidad.TipoEstado = Alterna> donde <ProductoUnidad.ProductoClave = Producto actual>
         1. Solo se presenta la siguiente información de la unidad que se tiene activa
            1. <NombreUnidad>
            2. Cantidad (\*)
6. El sistema obtiene la siguiente información de los detalles de la transacción <TransProdDetalle.TransProdID = TransProd.TransProdID registrado en sesión (objetoTransProd)>:
   1. **TransProdDetalle**
      1. TransProdDetalleID
7. Para cada detalle de la transacción obtenido
   1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Detalle de la Transacción – CUROLGEN06**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN06_AdministrarDetalleTransaccion.doc)
      1. El sistema envía como parámetro la siguiente información:
         1. ObtenerDetalleTransaccion
            1. TransProd = TransProd.TransProdID registrado en sesión (objetoTransProd)
            2. TransProdDetalleID
      2. El sistema recibe como parámetro la siguiente información:
         1. TransProdDetalle
   2. El sistema registra la información del objeto recibido dentro de un arreglo:
      1. **ArregloTransProdDetalle**
         1. TransProdDetalle
   3. El sistema registra en el objeto productos la siguiente información del producto actual de acuerdo a la información contenida en el objeto del detalle de la transacción:
      1. Productos
         1. ProductoClave
         2. Unidad
         3. UnidadAlterna (\*\*En caso de aplicar)
         4. Cantidad
         5. CantidadAlterna (\*\*En caso de aplicar)
         6. Precio
         7. Subtotal
         8. Impuesto
         9. Total
   4. El sistema obtiene la siguiente información del producto actual:
      1. **Producto**
         1. ProductoClave
         2. Nombre
         3. SubEmpresaID
         4. DecimalProducto
   5. El sistema actualiza el objeto Productos con la siguiente información para el producto actual:
      1. Productos
         1. Descripcion = <Producto.Nombre>
         2. SubEmpresaID
         3. DecimalProducto
8. Para cada producto incluido en el objeto Productos
   1. El sistema obtiene la siguiente información del objeto Productos:
      1. Productos
         1. ProductoClave
         2. Descripcion
         3. SubEmpresaID
         4. DecimalProducto
         5. Unidad
         6. UnidadAlterna (\*\*En caso de aplicar)
         7. Cantidad
         8. CantidadAlterna (\*\*En caso de aplicar)
         9. Precio
         10. Subtotal
         11. Impuesto
         12. Total
   2. El sistema presenta la siguiente información:
      1. Producto: Se presenta la descripción del producto, a través de la concatenación de la clave <Productos.ProductoClave> y nombre del mismo <Productos.Descripcion>, separados por un guion medio, ejemplo: “P0001 - Jugo Durazno”
      2. Cantidad: Se presenta la cantidad de producto <Productos.Cantidad>. Formatear la cantidad a mostrar de acuerdo al número de decimales configurado para el producto <Productos.DecimalProducto>.
      3. Unidad: Se presenta la descripción de la unidad de venta del producto, para ello ir a <VAVDescripcion> y obtener <VAVDescripcion.Descripcion> donde <VAVDescripcion.VARCodigo = ‘UNIDADV’> y <VAVDescripcion.VAVClave = Productos.Unidad>, de acuerdo al tipo de lenguaje configurado para el sistema.
      4. Si <ManejoDobleUnidad = 1 “verdadero”>
         1. Cantidad Alterna: se presenta la cantidad de producto capturada o calculada en la unidad Alterna <Productos.CantidadAlterna>
         2. Unidad Alterna: Se presenta la descripción de la unidad Alterna del producto, para ello ir a <VAVDescripcion> y obtener <VAVDescripcion.Descripcion> donde <VAVDescripcion.VARCodigo = ‘UNIDADV’> y <VAVDescripcion.VAVClave = Productos.UnidadAlterna>, de acuerdo al tipo de lenguaje configurado para el sistema
      5. Precio: Se presenta el precio unitario del producto <Productos.Precio>, precedido del signo “$”.
      6. Subtotal: Se presenta el subtotal del producto <Productos.Subtotal>, precedido del signo “$”.
      7. Impuesto: Se presenta el importe de los impuestos aplicados al producto <Productos.Impuesto>, precedido del signo “$”.
      8. Total: Se presenta el importe total de la partida del producto <Productos.Total>, precedido del signo “$”.
9. El sistema presenta la siguiente información:
   1. **Total Importe:** Se presenta la sumatoria del importe total de las partidas que conforman el detalle de la transacción, <∑TransProdDetalle.Total>, precedido del signo “$”.
   2. **Productos:** Se presenta el número de partidas que conforman el detalle de la transacción.
10. Para cada producto agregado por el usuario
    1. El sistema continúa en el [**paso 9.1**](#PasoParaCadaProductoAgregado) del flujo alterno opcional[**AO01 Crear Venta a Consignación**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV31_CapturarMovSinInvFueraVisita.docx#_AO01_Crear_Movimiento)
11. Si <el actor **selecciona** un producto>
    1. El sistema continúa en el [**paso 10.1**](#PasoSeleccionProducto) del flujo alterno opcional[**AO01 Crear Venta a Consignación**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV31_CapturarMovSinInvFueraVisita.docx#_AO01_Crear_Movimiento)
12. Si <el actor **selecciona** **sostenidamente** un producto>
    1. El sistema continúa en el [**paso 11.1**](#PasoSeleccionSostenidaProducto) del flujo alterno opcional[**AO01 Crear Venta a Consignación**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV31_CapturarMovSinInvFueraVisita.docx#_AO01_Crear_Movimiento)
13. Si <el actor selecciona la opción **Buscar**>
    1. El sistema continúa en el [**paso 12.1**](#PasoBuscar) del flujo alterno opcional[**AO01 Crear Venta a Consignación**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV31_CapturarMovSinInvFueraVisita.docx#_AO01_Crear_Movimiento)
14. Si <el actor selecciona la opción **Continuar**>
    1. El sistema continúa en el [**paso 13.1**](#PasoContinuar) del flujo alterno opcional[**AO01 Crear Venta a Consignación**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV31_CapturarMovSinInvFueraVisita.docx#_AO01_Crear_Movimiento)
15. Si <el actor selecciona la opción **Regresar**>
    1. El sistema continúa en el [**paso 14.1**](#PasoRegresar) del flujo alterno opcional[**AO01 Crear Venta a Consignación**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV31_CapturarMovSinInvFueraVisita.docx#_AO01_Crear_Movimiento)

### [AO04 Eliminar Venta a Consignación](#AO04_r)

1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Transacción – CUROLGEN05**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN05_AdministrarTransaccion.doc)
   1. El sistema envía como parámetro la siguiente información:
      1. ObtenerTransaccion
         1. TransProdID = TransProdID registrado en sesión
   2. El sistema recibe como parámetro la siguiente información
      1. TransProd
2. **MOTConfiguracion**
   1. ManejoDobleUnidad
   2. El sistema registra en sesión la información del objeto recibido para el encabezado de la transacción:
      1. TransProd
3. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Clientes – CUROLGEN14**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN14_AdministrarClientes.doc)
   1. El sistema envía como parámetro la siguiente información:
      1. ActualizarSaldoCteActual
         1. Importe = [<TransProd.Total> registrado en sesión (objeto TransProd)] \* -1
   2. El sistema recibe la siguiente información:
      1. Verdadero
4. El sistema presenta la siguiente información de sólo lectura:
   1. Lista de Precios: Se presenta la información de la lista de precios, a través de la concatenación de la clave <Precio.ProductoClave donde TransProd.PCEPrecioClave (objetoTransProd) = Precio.ProductoClave> y nombre del mismo <Precio.Nombre>, separados por un guion medio, ejemplo: “0002 – Lista Especial”
5. El sistema presenta la siguiente información de sólo lectura:
   1. Producto (\*)
   2. Si <ManejoDobleUnidad = 0 “Falso”>
      1. Unidad (\*) (c)
      2. Cantidad (\*)
   3. Si <ManejoDobleUnidad = 1 “Verdadero”>
      1. Si <Existe una unidad donde <ProductoUnidad.TipoEstado = Alterna> donde <ProductoUnidad.ProductoClave = Producto actual>
         1. Para cada Unidad de acuerdo al orden configurado <ConfigParametro.Valor>, se va a buscar la descripción correspondiente al Valor por Referencia UNIDADV, que coincida con los valores registrados en el parámetro y se muestran de acuerdo al orden de aparición
            1. <NombreUnidad>
            2. Cantidad (\*)
      2. Si <No Existe una unidad donde <ProductoUnidad.TipoEstado = Alterna> donde <ProductoUnidad.ProductoClave = Producto actual>
         1. Solo se presenta la siguiente información de la unidad que se tiene activa
            1. <NombreUnidad>
            2. Cantidad (\*)
6. El sistema obtiene la siguiente información de los detalles de la transacción <TransProdDetalle.TransProdID = TransProd.TransProdID registrado en sesión (objetoTransProd)>:
   1. **TransProdDetalle**
      1. TransProdDetalleID
7. Para cada detalle de la transacción obtenido
   1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Detalle de la Transacción – CUROLGEN06**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN06_AdministrarDetalleTransaccion.doc)
      1. El sistema envía como parámetro la siguiente información:
         1. ObtenerDetalleTransaccion
            1. TransProd = TransProd.TransProdID registrado en sesión (objetoTransProd)
            2. TransProdDetalleID
      2. El sistema recibe como parámetro la siguiente información:
         1. TransProdDetalle
   2. El sistema registra la información del objeto recibido dentro de un arreglo:
      1. **ArregloTransProdDetalle**
         1. TransProdDetalle
   3. El sistema registra en el objeto productos la siguiente información del producto actual de acuerdo a la información contenida en el objeto del detalle de la transacción:
      1. Productos
         1. ProductoClave
         2. Unidad
         3. UnidadAlterna(\*\*En caso de aplicar)
         4. Cantidad
         5. CantidadAlterna (\*\*En caso de aplicar)
         6. Precio
         7. Subtotal
         8. Impuesto
         9. Total
   4. El sistema obtiene la siguiente información del producto actual:
      1. **Producto**
         1. ProductoClave
         2. Nombre
         3. DecimalProducto
   5. El sistema actualiza el objeto Productos con la siguiente información para el producto actual:
      1. Productos
         1. Descripcion = <Producto.Nombre>
         2. DecimalProducto
8. Para cada producto incluido en el objeto Productos
   1. El sistema obtiene la siguiente información del objeto Productos:
      1. Productos
         1. ProductoClave
         2. Descripcion
         3. DecimalProducto
         4. Unidad
         5. UnidadAlterna (\*\*En caso de aplicar)
         6. Cantidad
         7. CantidadAlterna (\*\*En caso de aplicar)
         8. Precio
         9. Subtotal
         10. Impuesto
         11. Total
   2. El sistema presenta la siguiente información de sólo lectura:
      1. Producto: Se presenta la descripción del producto, a través de la concatenación de la clave <Productos.ProductoClave> y nombre del mismo <Productos.Descripcion>, separados por un guion medio, ejemplo: “P0001 - Jugo Durazno”
      2. Si <ManejoDobleUnidad = 1 “verdadero”>
         1. Cantidad Alterna: se presenta la cantidad de producto capturada o calculada en la unidad Alterna <Productos.CantidadAlterna>
         2. Unidad Alterna: Se presenta la descripción de la unidad Alterna del producto, para ello ir a <VAVDescripcion> y obtener <VAVDescripcion.Descripcion> donde <VAVDescripcion.VARCodigo = ‘UNIDADV’> y <VAVDescripcion.VAVClave = Productos.UnidadAlterna>, de acuerdo al tipo de lenguaje configurado para el sistema
      3. Cantidad: Se presenta la cantidad de producto <Productos.Cantidad>. Formatear la cantidad a mostrar de acuerdo al número de decimales configurado para el producto <Productos.DecimalProducto>.
      4. Unidad: Se presenta la descripción de la unidad de venta del producto, para ello ir a <VAVDescripcion> y obtener <VAVDescripcion.Descripcion> donde <VAVDescripcion.VARCodigo = ‘UNIDADV’> y <VAVDescripcion.VAVClave = Productos.Unidad>, de acuerdo al tipo de lenguaje configurado para el sistema.
      5. Precio: Se presenta el precio unitario del producto <Productos.Precio>, precedido del signo “$”.
      6. Subtotal: Se presenta el subtotal del producto <Productos.Subtotal>, precedido del signo “$”.
      7. Impuesto: Se presenta el importe de los impuestos aplicados al producto <Productos.Impuesto>, precedido del signo “$”.
      8. Total: Se presenta el importe total de la partida del producto <Productos.Total>, precedido del signo “$”.
9. El sistema presenta la siguiente información:
   1. **Total Importe:** Se presenta la sumatoria del importe total de las partidas que conforman el detalle de la transacción, <∑TransProdDetalle.Total>, precedido del signo “$”.
   2. **Productos:** Se presenta el número de partidas que conforman el detalle de la transacción.
10. Si <el actor selecciona la opción **Continuar**>
    1. El sistema presenta el siguiente título de pantalla:
       1. Generales
       2. Fecha: Se presenta la fecha del día de trabajo actual (DiaClave), a partir de la información registrada en sesión.
    2. El sistema presenta la siguiente información DE SÓLO LECTURA:
       1. **Folio:** <TransProd.Folio> registrado en sesión (objeto TransProd). Mostrar de sólo lectura.
       2. **Fase:** Se presenta la descripción de la fase de la transacción, para ello ir a <VAVDescripcion> y obtener <VAVDescripcion.Descripcion> donde <VAVDescripcion.VARCodigo = ‘TRPFSE> y <VAVDescripcion.VAVClave = TransProd.TipoFase > (objeto TransProd registrado en sesión), de acuerdo al tipo de lenguaje configurado para el sistema. Mostrar de sólo lectura.
       3. **Fecha:** <TransProd.FechaCaptura> registrado en sesión (objeto TransProd). Mostrar de sólo lectura en formato dd/mm/aaaa.
       4. **Lista Precios:** <TransProd.PCEPrecioClave> registrado en sesión (objeto TransProd). Mostrar de sólo lectura.
       5. **Subtotal:** <TransProd.Subtotal> registrado en sesión (objeto TransProd). Mostrar de sólo lectura, precedido del signo “$”.
       6. **Impuesto:** <TransProd.Impuesto> registrado en sesión (objeto TransProd). Mostrar de sólo lectura, precedido del signo “$”.
       7. **Total:** <TransProd.Total> registrado en sesión (objeto TransProd). Mostrar de sólo lectura, precedido del signo “$”.
       8. **Comentarios:** <TransProd.Notas> registrado en sesión (objeto TransProd). Mostrar de sólo lectura.
       9. **Pedido Adicional:** <TRPVtaAcreditada.PedidoAdicional>. Mostrar solo Si la configuración del cliente indica que es requerido el campo de pedido adicional <Cliente.ExigirPedidoAdicional=1 “Activo”>. Mostrar de sólo lectura.
    3. Si <el actor selecciona la opción **Terminar**>
       1. El sistema presenta el mensaje “[P0001] ¿Desea eliminar el registro seleccionado?”
          1. Si <el actor selecciona la opción **No**>
             1. El sistema continúa en el [**paso 7**](#PasoPresentaTitulo) del flujo básico
          2. Si <el actor selecciona la opción **Si**>
             1. El sistema obtiene la siguiente información de los objetos registrados en el arreglo:

**ArregloTransProdDetalle**

TransProdDetalle

* + - * 1. Para cada detalle de la transacción obtenido

El caso de uso incluye la funcionalidad del caso de uso [**Administrar Inventario Móvil – CUROLGEN13**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN13_AdministrarInventario.doc)

El sistema envía como parámetro la siguiente información:

ActualizarInventario

ProductoClave = TransProdDetalle.ProductoClave

TipoUnidad = TransProdDetalle.TipoUnidad

Cantidad = TransProdDetalle.Cantidad

TipoTransProd = <TransProd.Tipo> registrado en sesión (objeto TransProd)

TipoMovimiento = <TransProd.TipoMovimiento> registrado en sesión (objeto TransProd)

AlmacenID = AlmacenID registrado en sesión

Cancelar = 1

El sistema recibe la siguiente información:

OP\_Verdadero

Si <al cliente se le debe realizar el manejo del préstamo de envase <Cliente.Prestamo = 1 donde Cliente.ClienteClave = TransProd.ClienteClave (objeto TransProd registrado en sesión)>

Si <el producto modificado se puede prestar <ProductoDetalle.ProductoClave = ProductoClave (objeto Productos)> y <ProductoDetalle.PRUTipoUnidad = Unidad (objeto Productos)> y <ProductoDetalle.Prestamo = 1>>

El caso de uso incluye la funcionalidad del caso de uso [**Administrar Préstamo de Producto – CUROLGEN15**](file:///C:\Users\Belem\AppData\Roaming\Microsoft\ECU_GEN\CUROLGEN15_AdministrarPrestamoDeProducto.docx)y le envía como parámetro la siguiente información:

ActualizarPrestamo

ClienteClave = <TransProd.ClienteClave> (objeto TransProd registrado en sesión)

ProductoClave = <ProductoDetalle.ProductoDetClave> donde <ProductoDetalle.ProductoClave = TransProdDetalle.ProductoClave y ProductoDetalle.PRUTipoUnidad = TransProdDetalle.TipoUnidad>

TipoTransProd = <TransProd.Tipo> (objeto TransProd registrado en sesión)

Cantidad = <TransProdDetalle.Cantidad> \* -1

TipoUnidad = <TransProdDetalle.TipoUnidad>

Vendido = 0

El sistema recibe la siguiente información:

OP\_Verdadero

El caso de uso incluye la funcionalidad del caso de uso [**Administrar Detalle de la Transacción - CUROLGEN06**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN06_AdministrarDetalleTransaccion.doc)

El sistema envía como parámetro la siguiente información:

EliminarDetalleTransaccion

TransProdID = <TransProd.TransProdID> registrado en sesión (objeto TransProd)

TransProdDetalleID

El sistema recibe como parámetro la siguiente información:

Verdadero

* + - * 1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Transacción – CUROLGEN05**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN05_AdministrarTransaccion.doc)

El sistema envía como parámetro la siguiente información:

EliminarTransaccion

TransProd = <TransProd.TransProdID> registrado en sesión (objeto TransProd)

El sistema recibe como parámetro la siguiente información:

Verdadero

* + - * 1. Si <ManejoDobleUnidad = 1 “Verdadero”>

El sistema elimina la siguiente información

TPDDatosExtra

TransProdID = <ArregloTransProdDetalleEntrada.TransProdDetalleID>

TransProdDetalleID

* + - * 1. Si <el Vendedor tiene activo el parámetro para validar el número de clientes distintos que pueden tener consignas abiertas <Vendedor.ValidaLimiteConsignas = 1>>

El sistema obtiene información de las Ventas a Consignación que se encuentran abiertas (no liquidadas) para el cliente actual

**TransProd**

TransProdID

VisitaClave = <Visita.VisitaClave> donde <Visita.ClienteClave = ClienteClave registrado en sesión>

Tipo = 24

TipoFase = 2

Si <se obtuvo una o más Ventas a Consignación sin liquidar para el cliente actual>

El sistema continúa en el [**paso**](#PasoPresentaTitulo) **7** del flujo básico

Si <NO se obtuvieron Ventas a Consignación sin liquidar para el cliente actual>

Si <el cliente actual se encuentra en el listado de clientes registrados con consignas abiertas para el vendedor <VendedorConsigAb.ClienteClave *INCLUYE A* ClienteClave registrado en sesión>>, el sistema elimina la siguiente información, donde <VendedorConsigAb.VendedorID = VendedorId registrado en sesión> y <VendedorConsigAb.ClienteClave = ClienteClave registrado en sesión>:

**VendedorConsigAb**

VendedorID

ClienteClave

MFechaHora

MUsuarioID

* + - * 1. El sistema continúa en el [**paso**](#PasoPresentaTitulo) **7** del flujo básico
  1. Si <el actor selecciona la opción **Regresar**>
     1. El sistema continúa en el [**paso 3**](#paso3_AO04) del flujo alterno opcional [**AO04 Eliminar Venta a Consignación**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV44_CapturarVentasConsignación.docx#_AO03_Eliminar_Venta)

1. Si <el actor selecciona la opción **Regresar**>
   1. El sistema continúa en el [**paso 13.1**](#paso13_1_AO01) del flujo alterno opcional[**AO01 Crear Venta a Consignación**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV31_CapturarMovSinInvFueraVisita.docx#_AO01_Crear_Movimiento)

### [AO05 Liquidar Venta a Consignación](#AO05_r)

1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Transacción – CUROLGEN05**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN05_AdministrarTransaccion.doc)
   1. El sistema envía como parámetro la siguiente información:
      1. ObtenerTransaccion
         1. TransProdID = TransProdID registrado en sesión
   2. El sistema recibe como parámetro la siguiente información
      1. TransProd
   3. El sistema registra en sesión la información del objeto recibido para el encabezado de la transacción:
      1. TransProd
2. El sistema registra en sesión la siguiente información:
   1. TotalLiquidacion = <TransProd.Total> registrado en sesión (objeto TransProd).
   2. ImporteSaldo = <TransProd.Saldo> registrado en sesión (objeto TransProd).
   3. TotalDevoluciones = 0
3. El sistema presenta la siguiente información:
   1. **Folio:** <TransProd.Folio> registrado en sesión (objeto TransProd). Mostrar de sólo lectura.
   2. **Fecha**: <TransProd.FechaCaptura> registrado en sesión (objeto TransProd). Mostrar de sólo lectura en formato dd/mm/aaaa.
4. El sistema obtiene la siguiente información de los detalles de la transacción <TransProdDetalle.TransProdID = TransProd.TransProdID registrado en sesión (objetoTransProd)>:
   1. **TransProdDetalle**
      1. TransProdDetalleID
5. Para cada detalle de la transacción obtenido
   1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Detalle de la Transacción – CUROLGEN06**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN06_AdministrarDetalleTransaccion.doc)
      1. El sistema envía como parámetro la siguiente información:
         1. ObtenerDetalleTransaccion
            1. TransProd = TransProd.TransProdID registrado en sesión (objetoTransProd)
            2. TransProdDetalleID
      2. El sistema recibe como parámetro la siguiente información:
         1. TransProdDetalle
   2. El sistema registra la información del objeto recibido dentro de un arreglo:
      1. **ArregloTransProdDetalle**
         1. TransProdDetalle
   3. El sistema registra en el objeto productos la siguiente información del producto actual de acuerdo a la información contenida en el objeto del detalle de la transacción:
      1. Productos
         1. ProductoClave
         2. Unidad
         3. UnidadAlterna (\*\*En caso de aplicar)
         4. Cantidad
         5. CantidadAlterna (\*\*En caso de aplicar)
         6. Precio
         7. Total
   4. El sistema obtiene la siguiente información del producto actual:
      1. **Producto**
         1. ProductoClave
         2. Nombre
         3. DecimalProducto
   5. El sistema actualiza el objeto Productos con la siguiente información para el producto actual:
      1. Productos
         1. Descripcion = <Producto.Nombre>
         2. DecimalProducto
         3. Devolucion = 0
         4. Liquidar = <Productos.Cantidad – Productos.Devolucion>
6. Para cada producto incluido en el objeto Productos
   1. El sistema obtiene la siguiente información del objeto Productos:
      1. Productos
         1. ProductoClave
         2. Descripcion
         3. DecimalProducto
         4. Unidad
         5. UnidadAlterna (\*\*En caso de aplicar)
         6. Cantidad
         7. CantidadAlterna (\*\*En caso de aplicar)
         8. Precio
         9. Total
         10. Devolucion
         11. Liquidar
   2. El sistema presenta la siguiente información de sólo lectura:
      1. Producto: Se presenta la descripción del producto, a través de la concatenación de la clave <Productos.ProductoClave> y nombre del mismo <Productos.Descripcion>, separados por un guion medio, ejemplo: “P0001 - Jugo Durazno”
      2. Total: Se presenta el importe total de la partida del producto <Productos.Total>, precedido del signo “$”.
      3. Cantidad: Se presenta la cantidad de producto <Productos.Cantidad>. Formatear la cantidad a mostrar de acuerdo al número de decimales configurado para el producto <Productos.DecimalProducto>.
      4. Unidad: Se presenta la descripción de la unidad de venta del producto, para ello ir a <VAVDescripcion> y obtener <VAVDescripcion.Descripcion> donde <VAVDescripcion.VARCodigo = ‘UNIDADV’> y <VAVDescripcion.VAVClave = Productos.Unidad>, de acuerdo al tipo de lenguaje configurado para el sistema.
      5. Si <ManejoDobleUnidad = 1 “verdadero”>
         1. Cantidad Alterna: se presenta la cantidad de producto capturada o calculada en la unidad Alterna <Productos.CantidadAlterna>
         2. Unidad Alterna: Se presenta la descripción de la unidad Alterna del producto, para ello ir a <VAVDescripcion> y obtener <VAVDescripcion.Descripcion> donde <VAVDescripcion.VARCodigo = ‘UNIDADV’> y <VAVDescripcion.VAVClave = Productos.UnidadAlterna>, de acuerdo al tipo de lenguaje configurado para el sistema
   3. El sistema presenta y solicita la siguiente información:
      1. **Devolución:** Se presenta la cantidad de producto a devolver <Productos.Devolucion>, permitir la captura de una nueva cantidad a devolver. Formatear la cantidad a mostrar de acuerdo al número de decimales configurado para el producto <Productos.DecimalProducto>.
      2. **Liquidar:** Se presenta la diferencia resultante entre la cantidad de producto entregada en la Venta a Consignación y la cantidad de producto a devolver <Productos.Liquidar>. Mostrar de sólo lectura.
7. El sistema presenta la siguiente información de sólo lectura:
   1. **Importe a Liquidar** = TotalLiquidacion registrado en sesión, precedido del signo “$”, (formatear el importe a dos decimales).
   2. **Devoluciones** = TotalDevoluciones registrado en sesión, precedido del signo “$”, (formatear el importe a dos decimales).
   3. Si <si la Venta a Consignación NO tiene promociones aplicadas <TransProd.Promocion = 0> (objeto TransProd registrado en sesión)>
      1. **Saldo =** ImporteSaldo registrado en sesión, precedido del signo “$”, (formatear el importe a dos decimales).
8. Si <el actor proporciona una cantidad para la **Devolución**> y si <el actor selecciona la opción **Agregar**>
   1. El sistema valida la cantidad de producto proporcionada por el actor:
      1. Si <la diferencia resultante de la cantidad de producto entregada en la Venta a Consignación menos la cantidad a devolver proporcionada por el actor es menor a cero>
         1. El sistema presenta el mensaje “[E0216] La Cantidad debe ser Mayor ó Igual a $0$ y Menor ó Igual a $1$”, sustituir el valor de $0$ por ‘0’ y el de $1$ por <Productos.Cantidad>
         2. El sistema continúa en el paso anterior de donde fue invocado, posicionando el cursor dentro del campo de captura para la devolución de producto
      2. Si <la cantidad de producto a devolver proporcionada por el actor es menor a cero>
         1. El sistema presenta el mensaje “[E0216] La Cantidad debe ser Mayor ó Igual a $0$ y Menor ó Igual a $1$”, sustituir el valor de $0$ por ‘0’ y el de $1$ por <Productos.Cantidad>
         2. El sistema continúa en el paso anterior de donde fue invocado, posicionando el cursor dentro del campo de captura para la devolución de producto
   2. El sistema actualiza el objeto Productos con la siguiente información para el producto actual:
      1. Productos
         1. Devolucion = Cantidad de producto a devolver proporcionada por el actor (formatear la cantidad de acuerdo al número de decimales configurado para el producto <Producto.DecimalProducto>).
         2. Liquidar = <Productos.Cantidad – Productos.Devolucion> (formatear la cantidad de acuerdo al número de decimales configurado para el producto <Producto.DecimalProducto>).
   3. El sistema registra en sesión la siguiente información:
      1. ImporteLiqui = 0
      2. ImporteDev = 0
   4. Para cada producto incluido en el objeto productos:
      1. El sistema registra en sesión la siguiente información:
         1. ImporteProducto = 0
         2. ImporteDevProducto = 0
      2. Si <el subtotal del importe a liquidar del producto es mayor a cero [Liquidar \* Precio (objeto Productos)] > 0>
         1. El caso de uso incluye la funcionalidad del caso de uso [**Calcular Impuestos – CUROLGEN02**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN02_CalcularImpuestos.doc)
            1. El sistema envía como parámetro la siguiente información:

BuscarImpuestos

ProductoClave = ProductoClave (objeto Productos)

TipoImpuesto = <Cliente.TipoImpuesto> registrado en sesión (objeto Cliente)

* + - * 1. El sistema recibe como parámetro la siguiente información

PS\_ImpuestosRelacionados

* + - * 1. El sistema registra en sesión la siguiente información:

ListaImpuestos = PS\_ImpuestosRelacionados

* + - 1. El caso de uso incluye la funcionalidad del caso de uso [**Calcular Impuestos – CUROLGEN02**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN02_CalcularImpuestos.doc)
         1. El sistema envía como parámetro la siguiente información:

CalcularImpuestos

ImpuestosRelacionados = ListaImpuestos registrada en sesión

PrecioBase = Liquidar \* Precio (objeto Productos)

PrecioUnitario = Precio (objeto Productos)

* + - * 1. El sistema recibe como parámetro la siguiente información

PS\_TotalImpuesto

* + - * 1. El sistema registra en sesión la siguiente información:

ImpuestosProducto = PS\_TotalImpuesto (formatear el importe a dos decimales).

* + - 1. El sistema actualiza en sesión la siguiente información:
         1. ImporteProducto = [Liquidar \* Precio (objeto Productos)] + ImpuestosProducto registrado en sesión
    1. El sistema actualiza en sesión la siguiente información:
       1. **ImporteLiqui** = *ImporteLiqui* + ImporteProducto
    2. Si <el subtotal del importe a devolver del producto es mayo a cero [Devolucion \* Precio (objeto Productos)] > 0>
       1. El caso de uso incluye la funcionalidad del caso de uso [**Calcular Impuestos – CUROLGEN02**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN02_CalcularImpuestos.doc)
          1. El sistema envía como parámetro la siguiente información:

CalcularImpuestos

ImpuestosRelacionados = ListaImpuestos registrada en sesión

PrecioBase = Devolucion \* Precio (objeto Productos)

PrecioUnitario = Precio (objeto Productos)

* + - * 1. El sistema recibe como parámetro la siguiente información

PS\_TotalImpuesto

* + - * 1. El sistema registra en sesión la siguiente información:

ImpuestosProducto = PS\_TotalImpuesto (formatear el importe a dos decimales).

* + - 1. El sistema actualiza en sesión la siguiente información:
         1. ImporteDevProducto = [Devolucion \* Precio (objeto Productos)] + ImpuestosProducto registrado en sesión
    1. El sistema actualiza en sesión la siguiente información:
       1. **ImporteDev** = *ImporteDev* + ImporteDevProducto
  1. El sistema actualiza en sesión la siguiente información:
     1. TotalLiquidacion = ImporteLiqui
     2. TotalDevoluciones = ImporteDev
     3. ImporteSaldo = <TransProd.Saldo (objeto TransProd)> - ImporteDev
  2. El sistema continúa en el [**paso 6**](#paso6_AO05) del flujo alterno opcional [**AO05 Liquidar Venta a Consignación**](#_AO04_Liquidar_Venta)

1. Si <el actor selecciona la opción **Terminar**>
   1. Si <se muestra el saldo final de la Liquidación, es decir si la Venta a Consignación NO tiene promociones aplicadas <TransProd.Promocion = 0> (objeto TransProd registrado en sesión)>
      1. Si <el saldo final de la Liquidación es menor a Cero (ImporteSaldo < 0)>
         1. El sistema presenta el mensaje “[E0697] La Devolución no puede ser efectuada, queda un saldo a Favor del Cliente”
         2. El sistema continúa en el paso donde fue invocado
   2. Si <el importe final a liquidar es igual a Cero (TotalLiquidacion = 0)> Y si <el Saldo de la Venta a Consignación es diferente al Total de ésta <TransProd.Saldo <> TransProd.Total> (objeto TransProd registrado en sesión)>
      1. El sistema presenta el mensaje “[E0697] La Devolución no puede ser efectuada, queda un saldo a Favor del Cliente”
      2. El sistema continúa en el paso donde fue invocado
   3. Si <el parámetro de configuración indica que NO se puede cancelar una venta a consignación por liquidación (CancConsigLiqui = 0) de acuerdo a la información registrada en sesión>
      1. El sistema continúa con el flujo alterno opcional **[AO06 Liquidar Venta a Consignación por Devolución](#_AO06_Liquidar_Venta)**
      2. El sistema recibe como parámetro la siguiente información:
         1. Fallo
      3. El sistema continúa con el flujo alterno opcional **[AO08 Terminar Proceso](#_AO08_Terminar_Proceso)** y le envía como parámetro la siguiente información:
         1. Fallo
   4. Si <el parámetro de configuración indica que SI se puede cancelar una venta a consignación por liquidación (CancConsigLiqui = 1) de acuerdo a la información registrada en sesión>
      1. Si <por lo menos un producto del objeto Productos cuenta con una cantidad a devolver mayor a Cero <Productos.Devolucion > 0>>
         1. El sistema registra en sesión la siguiente información:
            1. TipoLiquidacion = GeneraDevolucion
         2. El sistema presenta el mensaje “[P0228] ¿Desea enviar como una consigna el producto devuelto?”
            1. Si <el actor selecciona la opción **No**>

TipoLiquidacion = GeneraDevolucion

* + - * 1. Si <el actor selecciona la opción **Si**>

TipoLiquidacion = GeneraConsignacion

* + 1. El sistema continúa con el flujo alterno opcional **[AO07 Liquidar Venta a Consignación por Cancelación](#_AO07_Liquidar_Venta)**
    2. El sistema recibe como parámetro la siguiente información:
       1. Fallo
    3. El sistema continúa con el flujo alterno opcional [**AO08 Terminar Proceso**](#_AO08_Terminar_Proceso)y le envía como parámetro la siguiente información:
       1. Fallo

1. Si <el actor selecciona la opción **Regresar**>
   1. Si <se realizaron cambios (se capturó producto a devolver)>, el sistema presenta el mensaje “[BP0002] Se perderán los cambios. ¿Está seguro de regresar?”
      1. Si <el actor selecciona la opción **No**>
         1. El sistema continúa en el paso en donde fue invocado
      2. Si <el actor selecciona la opción **Si** >
         1. El sistema continúa en el [**paso**](#PasoPresentaTitulo) **7** del flujo básico
   2. Si <no se realizaron cambios (no se capturó producto a devolver)>
      1. El sistema continúa en el [**paso**](#PasoPresentaTitulo) **7** del flujo básico

### [AO06 Liquidar Venta a Consignación por Devolución](#AO06_r)

1. El sistema obtiene información de los movimientos de devolución asociados a la Venta a Consignación, donde <TrpTpd.TransProdID1 = TransProd.TransProdID registrado en sesión (objetoTransProd)>:
   1. **TrpTpd**
      1. TransProdID
2. Si <se obtuvo un movimiento de devolución asociado a la Venta a Consignación>
   1. El sistema registra en sesión la siguiente información:
      1. TransProdIDDevolucion = TransProdID de la devolución obtenida
   2. El sistema elimina la siguiente información:
      1. **TrpTpd**
         1. TransProdID
         2. TransProdID1 = TransProd.TransProdID registrado en sesión (objetoTransProd)
         3. TransProdDetalleID
   3. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Transacción – CUROLGEN05**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN05_AdministrarTransaccion.doc)
      1. El sistema envía como parámetro la siguiente información:
         1. ObtenerTransaccion
            1. TransProdID = TransProdIDDevolucion registrado en sesión
      2. El sistema recibe como parámetro la siguiente información
         1. TransProd
      3. El sistema registra en sesión la información del objeto recibido para el encabezado de la transacción de devolución:
         1. TransProdDev
   4. El sistema obtiene la siguiente información de los detalles de la devolución <TransProdDetalle.TransProdID = TransProdDev.TransProdID registrado en sesión (objetoTransProdDev)>:
      1. **TransProdDetalle**
         1. TransProdDetalleID
   5. Para cada detalle de la devolución obtenido
      1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Detalle de la Transacción – CUROLGEN06**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN06_AdministrarDetalleTransaccion.doc)
         1. El sistema envía como parámetro la siguiente información:
            1. ObtenerDetalleTransaccion

TransProd = <TransProdDev.TransProdID> registrado en sesión (objetoTransProdDev)

TransProdDetalleID

* + 1. El sistema recibe como parámetro la siguiente información:
       1. TransProdDetalle
    2. El sistema registra la información del objeto recibido dentro de un arreglo:
       1. **ArregloTransProdDetalleDev**
          1. TransProdDetalle
  1. Para cada detalle de la devolución incluido en el ArregloTransProdDetalleDev
     1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Inventario Móvil – CUROLGEN13**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN13_AdministrarInventario.doc)
        1. El sistema envía como parámetro la siguiente información:
           1. ActualizarInventario

ProductoClave = <TransProdDetalle.ProductoClave>

TipoUnidad = <TransProdDetalle.TipoUnidad>

Cantidad = <TransProdDetalle.Cantidad>

TipoTransProd = <TransProdDev.Tipo> registrado en sesión (objeto TransProdDev)

TipoMovimiento = <TransProdDev.TipoMovimiento> registrado en sesión (objeto TransProdDev)

AlmacenID = AlmacenID registrado en sesión

GrupoMotivo = GrupoMotivo registrado en sesión

Cancelacion = 1

* + - 1. El sistema recibe la siguiente información:
         1. OP\_Verdadero
    1. Si <al cliente se le debe realizar el manejo del préstamo de envase <Cliente.Prestamo = 1 donde Cliente.ClienteClave = TransProd.ClienteClave (objeto TransProd registrado en sesión)>
       1. Si <el producto actual se puede prestar <ProductoDetalle.ProductoClave = TransProdDetalle.ProductoClave> y <ProductoDetalle.PRUTipoUnidad = TransProdDetalle.TipoUnidad > y <ProductoDetalle.Prestamo = 1>>
          1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Préstamo de Producto – CUROLGEN15**](file:///C:\Users\Belem\AppData\Roaming\Microsoft\ECU_GEN\CUROLGEN15_AdministrarPrestamoDeProducto.docx)y le envía como parámetro la siguiente información:

ActualizarPrestamo

ClienteClave = <TransProd.ClienteClave> (objeto TransProd registrado en sesión)

ProductoClave = <ProductoDetalle.ProductoDetClave> donde <ProductoDetalle.ProductoClave = TransProdDetalle.ProductoClave y ProductoDetalle.PRUTipoUnidad = TransProdDetalle.TipoUnidad>

TipoTransProd = <TransProdDev.Tipo> registrado en sesión (objeto TransProdDev)

Cantidad = <TransProdDetalle.Cantidad> \* -1

TipoUnidad = <TransProdDetalle.TipoUnidad>

Vendido = 2

El sistema recibe la siguiente información:

OP\_Verdadero

* + 1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Clientes – CUROLGEN14**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN14_AdministrarClientes.doc)
       1. El sistema envía como parámetro la siguiente información:
          1. ActualizarSaldoCteActual

Importe = <TransProdDetalle.Total>

* + - 1. El sistema recibe la siguiente información:
         1. Verdadero
    1. El sistema actualiza la siguiente información para la transacción de la Venta a Consignación registrada en sesión <TransProd.TransProdID = TransProd.TransProdID registrado en sesión (objeto TransProd)>:
       1. **TransProd**
          1. Saldo = *Saldo* + <TransProdDetalle.Total>
    2. El sistema obtiene y registra en sesión la siguiente información para el producto actual, donde <ProductoDetalle.ProductoClave = TransProdDetalle.ProductoClave y ProductoDetalle.PRUTipoUnidad = TransProdDetalle.TipoUnidad>:
       1. **ProductoDetalle**
          1. Factor
    3. El caso de uso incluye la funcionalidad del caso de uso [**Calcular Cuotas – CUROLGEN07**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN07_CalcularCuotas.doc)
       1. El sistema envía como parámetro la siguiente información:
          1. VerificarCuotas

VendedorId = VendedorId registrado en sesión

ClienteClave = <Cliente.ClienteClave> registrado en sesión (objeto Cliente)

ProductoClave = <TransProdDetalle.ProductoClave>

Cantidad = [<TransProdDetalle.Cantidad> \* Factor registrado en sesión] \* -1

TipoCuota = 1 (“Producto”)

* + - 1. El sistema recibe la siguiente información
         1. PS\_CuoVen
    1. El caso de uso incluye la funcionalidad del caso de uso [**Calcular Cuotas – CUROLGEN07**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN07_CalcularCuotas.doc)
       1. El sistema envía como parámetro la siguiente información:
          1. VerificarCuotas

VendedorId = VendedorId registrado en sesión

ClienteClave = <Cliente.ClienteClave> registrado en sesión (objeto Cliente)

ProductoClave = <TransProdDetalle.ProductoClave>

Cantidad = [<TransProdDetalle.Cantidad> \* Factor registrado en sesión] \* -1

TipoCuota = 3 (“Kilos”)

* + - 1. El sistema recibe la siguiente información
         1. PS\_CuoVen
  1. El sistema elimina la siguiente información:
     1. **TPDImpuesto**
        1. TransProdID = <TransProdDev.TransProdID> registrado en sesión (objetoTransProdDev)
     2. **TrpPrp**
        1. TransProdID = <TransProdDev.TransProdID> registrado en sesión (objetoTransProdDev)
     3. **TpdDes**
        1. TransProdID = <TransProdDev.TransProdID> registrado en sesión (objetoTransProdDev)
     4. **TPDDesVendedor**
        1. TransProdID = <TransProdDev.TransProdID> registrado en sesión (objetoTransProdDev)
     5. **TransProdDetalle**
        1. TransProdID = <TransProdDev.TransProdID> registrado en sesión (objetoTransProdDev)
  2. El sistema actualiza la siguiente información para la transacción de la devolución registrada en sesión <TransProd.TransProdID = TransProdDev.TransProdID registrado en sesión (objeto TransProdDev)>:
     1. **TransProd**
        1. FechaCaptura = <Dia.FechaCaptura> donde <Dia.DiaClave = DiaClave registrado en sesión>
        2. FechaSurtido = Fecha y hora actual del sistema
        3. FechaHoraAlta = Fecha y hora actual del sistema

1. El sistema obtiene información del siguiente objeto:
   1. **ModuloMovDetalle**
      1. ModuloMovDetalleClave
      2. TipoIndice = 13 (“Devolución de Clientes”)
   2. El sistema registra en sesión la siguiente información del objeto obtenido:
      1. ModuloMovDetDevoluciones
2. El caso de uso incluye la funcionalidad del caso de uso [**CUADMESC13 Generar Folio**](file:///C:\Users\Route\Trunk\Analisis\EspecificacionRequerimientos\ECU_ESC\CUADMESC13_GenerarFolio.docx)
   1. El sistema envía como parámetro la siguiente información:
      1. GenerarFolio
         1. ModuloMovDetalleClave = ModuloMovDetDevoluciones.ModuloMovDetalleClave
   2. El sistema recibe como parámetro y registra en sesión la siguiente información:
      1. PS\_Folio
3. Si <se obtuvo un folio para la devolución de producto a generar (PS\_Folio <> Null o ‘ ‘)>
   1. El sistema registra la siguiente información para el nuevo movimiento de devolución a generar:
      1. **TransProd**
         1. TransProdID = Utilizar <Método KeyGen.GetId>
         2. VisitaClave = VisitaClave registrada en sesión
         3. DiaClave = DiaClave registrado en sesión
         4. PCEPrecioClave = <TransProd.PCEPrecioClave> registrado en sesión (objeto TransProd)
         5. PCEModuloMovDetClave = <TransProd.PCEModuloMovDetClave> registrado en sesión (objeto TransProd)
         6. ClienteClave = ClienteClave registrado en sesión
         7. Folio = PS\_Folio registrado en sesión
         8. Tipo = <ModuloMovDetDevoluciones.TipoTransProd> registrado en sesión
         9. TipoPedido = <ModuloMovDetDevoluciones.TipoPedido> registrado en sesión
         10. TipoFase = 2 (“Surtido”)
         11. TipoMovimiento = <ModuloMovDetDevoluciones.TipoMovimiento> registrado en sesión
         12. FechaHoraAlta = Fecha y hora actual del sistema
         13. FechaCaptura = <Dia.FechaCaptura> donde <Dia.DiaClave = DiaClave registrado en sesión>
         14. FechaEntrega = <Dia.FechaCaptura> donde <Dia.DiaClave = DiaClave registrado en sesión>
         15. FechaFacturacion = ‘1900-01-01 00:00:00.000’
         16. FechaSurtido = Fecha y hora actual del sistema
         17. FechaCancelacion = ‘1900-01-01 00:00:00.000’
         18. TipoMotivo = 12
         19. Enviado = 0
         20. MFechaHora = Fecha y hora actual del sistema
         21. MUsuarioID = UsuarioId registrado en sesión
      2. El sistema registra en sesión la siguiente información:
         1. TransProdIDDevolucion = TransProdID de la devolución generada
   2. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Transacción – CUROLGEN05**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN05_AdministrarTransaccion.doc)
      1. El sistema envía como parámetro la siguiente información:
         1. ObtenerTransaccion
            1. TransProdID = TransProdIDDevolucion registrado en sesión
      2. El sistema recibe como parámetro la siguiente información
         1. TransProd
      3. El sistema registra en sesión la información del objeto recibido para el encabezado de la transacción de devolución:
         1. TransProdDev
4. Para cada producto incluido en el objeto productos
   1. Si <el producto cuenta con una cantidad a devolver mayor a Cero <Productos.Devolucion > 0>>
      1. El sistema registra en sesión la siguiente información:
         1. Partida = 0
      2. El caso de uso incluye la funcionalidad del caso de uso [**Calcular Impuestos – CUROLGEN02**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN02_CalcularImpuestos.doc)
         1. El sistema envía como parámetro la siguiente información:
            1. BuscarImpuestos

ProductoClave = ProductoClave (objeto Productos)

TipoImpuesto = <Cliente.TipoImpuesto> registrado en sesión (objeto Cliente)

* + - 1. El sistema recibe como parámetro la siguiente información
         1. PS\_ImpuestosRelacionados
      2. El sistema registra en sesión la siguiente información:
         1. ListaImpuestos = PS\_ImpuestosRelacionados
    1. El caso de uso incluye la funcionalidad del caso de uso [**Calcular Impuestos – CUROLGEN02**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN02_CalcularImpuestos.doc)
       1. El sistema envía como parámetro la siguiente información:
          1. CalcularImpuestos

ImpuestosRelacionados = ListaImpuestos registrada en sesión

PrecioBase = Devolucion \* Precio (objeto Productos)

PrecioUnitario = Precio (objeto Productos)

* + - 1. El sistema recibe como parámetro la siguiente información
         1. PS\_TotalImpuesto
      2. El sistema registra en sesión la siguiente información:
         1. ImpuestosProducto = PS\_TotalImpuesto (formatear el importe a dos decimales).
    1. El sistema registra la siguiente información:
       1. **TransProdDetalle**
          1. TransProdID = <TransProdDev.TransProdID> registrado en sesión (objeto TransProdDev)
          2. TransProdDetalleID = <TransProdDetalle.TransProdDetalleID> correspondiente a la partida de la Venta a Conisgnación de la cual se realiza la devolución.
          3. ProductoClave = ProductoClave (objeto Productos)
          4. TipoUnidad = Unidad (objeto Productos)
          5. Partida = Partida registrada en sesión + 1
          6. Cantidad = Devolucion (objeto Productos)
          7. Precio = Precio (objeto Productos)
          8. DescuentoPor = 0
          9. DescuentoImp = 0
          10. Subtotal = Devolucion \* Precio (objeto Productos)
          11. Impuesto = ImpuestosProducto registrado en sesión
          12. Total = <TransProdDetalle.Subtotal> + <TransProdDetalle.Impuesto>
          13. Promocion = 0
          14. MFechaHora = Fecha y hora actual del sistema
          15. MUsuarioID = UsuarioId registrado en sesión
          16. Enviado = 0
       2. El sistema registra en sesión la siguiente información:
          1. TotalDetalle = <TransProdDetalle.Total>
    2. El caso de uso incluye la funcionalidad del caso de uso [**Calcular Impuestos – CUROLGEN02**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN02_CalcularImpuestos.doc)
       1. El sistema envía como parámetro la siguiente información:
          1. GuardarDetalle

TransProdId = <TransProdDetalle.TransProdID> actual

TransProdDetalleId = <TransProdDetalle.TransProdDetalleID> actual

ImpuestosRelacionados = ListaImpuestos

* + - 1. El sistema recibe como parámetro la siguiente información
         1. Verdadero
    1. El sistema actualiza la siguiente información para la transacción de la Venta a Consignación registrada en sesión <TransProd.TransProdID = TransProd.TransProdID registrado en sesión (objeto TransProd)>:
       1. **TransProd**
          1. Saldo = *Saldo* – <TransProdDetalle.Total> actual
    2. El sistema registra la siguiente información:
       1. **TrpTpd**
          1. TransProdID = <TransProdDev.TransProdID> registrado en sesión (objeto TransProdDev)
          2. TransProdID1 = <TransProd.TransProdID> registrado en sesión (objeto TransProd)
          3. TransProdDetalleID = <TransProdDetalle.TransProdDetalleID> actual
          4. Cantidad = <TransProdDetalle.Cantidad> actual
          5. Precio = <TransProdDetalle.Precio> actual
          6. Subtotal = <TransProdDetalle.Subtotal> actual
          7. Impuesto = <TransProdDetalle.Impuesto> actual
          8. Total = <TransProdDetalle.Total> actual
          9. MFechaHora = Fecha y hora actual del sistema
          10. MUsuarioID = UsuarioId registrado en sesión
    3. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Inventario Móvil – CUROLGEN13**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN13_AdministrarInventario.doc)
       1. El sistema envía como parámetro la siguiente información:
          1. ActualizarInventario

ProductoClave = ProductoClave (objeto Productos)

TipoUnidad = Unidad (objeto Productos)

Cantidad = Devolucion (objeto Productos)

TipoTransProd = <TransProdDev.Tipo> registrado en sesión (objeto TransProdDev)

TipoMovimiento = <TransProdDev.TipoMovimiento> registrado en sesión (objeto TransProdDev)

AlmacenID = AlmacenID registrado en sesión

GrupoMotivo = GrupoMotivo registrado en sesión

Cancelacion = 0

* + - 1. El sistema recibe la siguiente información:
         1. OP\_Verdadero
    1. Si <al cliente se le debe realizar el manejo del préstamo de envase <Cliente.Prestamo = 1 donde Cliente.ClienteClave = TransProd.ClienteClave (objeto TransProd registrado en sesión)>
       1. Si <el producto actual se puede prestar <ProductoDetalle.ProductoClave = ProductoClave (objeto Productos)> y <ProductoDetalle.PRUTipoUnidad = Unidad (objeto Productos)> y <ProductoDetalle.Prestamo = 1>>
          1. El sistema obtiene información del Saldo del préstamo del producto, donde <ProductoPrestamoCli.ClienteClave = TransProd.ClienteClave (objeto TransProd registrado en sesión)> y <ProductoPrestamoCli.ProductoClave = ProductoDetalle.ProductoDetClave donde ProductoDetalle.ProductoClave = ProductoClave (objeto Productos) y ProductoDetalle.PRUTipoUnidad = Unidad (objeto Productos)>:

**ProductoPrestamoCli**

Saldo

* + - * 1. Si <no se obtuvo un registro de ProductoPrestamoCli para el Cliente y producto actuales>

El caso de uso incluye la funcionalidad del caso de uso [**Administrar Préstamo de Producto – CUROLGEN15**](file:///C:\Users\Belem\AppData\Roaming\Microsoft\ECU_GEN\CUROLGEN15_AdministrarPrestamoDeProducto.docx)y le envía como parámetro la siguiente información:

CrearPrestamo

ClienteClave = <TransProd.ClienteClave> (objeto TransProd registrado en sesión)

ProductoClave = <ProductoDetalle.ProductoDetClave> donde <ProductoDetalle.ProductoClave = ProductoClave (objeto Productos) y ProductoDetalle.PRUTipoUnidad = Unidad (objeto Productos)>

El sistema recibe la siguiente información:

OP\_Verdadero

* + - * 1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Préstamo de Producto – CUROLGEN15**](file:///C:\Users\Belem\AppData\Roaming\Microsoft\ECU_GEN\CUROLGEN15_AdministrarPrestamoDeProducto.docx)y le envía como parámetro la siguiente información:

ActualizarPrestamo

ClienteClave = <TransProd.ClienteClave> (objeto TransProd registrado en sesión)

ProductoClave = <ProductoDetalle.ProductoDetClave> donde <ProductoDetalle.ProductoClave = ProductoClave (objeto Productos) y ProductoDetalle.PRUTipoUnidad = Unidad (objeto Productos)>

TipoTransProd = <TransProdDev.Tipo> registrado en sesión (objeto TransProdDev)

Cantidad = Devolucion (objeto Productos)

TipoUnidad = Unidad (objeto Productos)

Vendido = 2

El sistema recibe la siguiente información:

OP\_Verdadero

* + 1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Clientes – CUROLGEN14**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN14_AdministrarClientes.doc)
       1. El sistema envía como parámetro la siguiente información:
          1. ActualizarSaldoCteActual

Importe = <TransProdDetalle.Total> actual \* -1

* + - 1. El sistema recibe la siguiente información:
         1. Verdadero
  1. Si <el producto cuenta con una cantidad a liquidar mayor a Cero <Productos.Liquidar > 0>>
     1. El sistema obtiene y registra en sesión la siguiente información para el producto actual, donde <ProductoDetalle.ProductoClave = ProductoClave (objeto Productos) y ProductoDetalle.PRUTipoUnidad = Unidad (objeto Productos)>:
        1. **ProductoDetalle**
           1. Factor
     2. El caso de uso incluye la funcionalidad del caso de uso [**Calcular Cuotas – CUROLGEN07**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN07_CalcularCuotas.doc)
        1. El sistema envía como parámetro la siguiente información:
           1. VerificarCuotas

VendedorId = VendedorId registrado en sesión

ClienteClave = <Cliente.ClienteClave> registrado en sesión (objeto Cliente)

ProductoClave = ProductoClave (objeto Productos)

Cantidad = Cantidad (objeto Productos) \* Factor registrado en sesión.

TipoCuota = 1 (“Producto”)

* + - 1. El sistema recibe la siguiente información
         1. PS\_CuoVen
    1. El caso de uso incluye la funcionalidad del caso de uso [**Calcular Cuotas – CUROLGEN07**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN07_CalcularCuotas.doc)
       1. El sistema envía como parámetro la siguiente información:
          1. VerificarCuotas

VendedorId = VendedorId registrado en sesión

ClienteClave = <Cliente.ClienteClave> registrado en sesión (objeto Cliente)

ProductoClave = ProductoClave (objeto Productos)

Cantidad = Cantidad (objeto Productos) \* Factor registrado en sesión.

TipoCuota = 3 (“Kilos”)

* + - 1. El sistema recibe la siguiente información
         1. PS\_CuoVen

1. Si <por lo menos un producto del objeto Productos cuenta con una cantidad a devolver mayor a Cero <Productos.Devolucion > 0>>
   1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Transacción – CUROLGEN05**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN05_AdministrarTransaccion.doc)
      1. El sistema envía como parámetro la siguiente información:
         1. CalcularTotalesTransaccion
            1. TransProd = <TransProdDev.TransProdID> registrado en sesión (objeto TransProdDev)
      2. El sistema recibe la siguiente información
         1. TransProd
      3. El sistema actualiza la información del objeto registrado en sesión:
         1. TransProdDev
   2. Si <no se obtuvo un movimiento de devolución asociado a la Venta a Consignación (TransProdIDDevolucion registrado en sesión = ‘ ‘)>
      1. El sistema realiza la confirmación del folio generado para el movimiento de devolución
   3. El sistema continúa en el [**paso 9**](#paso9_AO06) del flujo alterno opcional [**AO06 Liquidar Venta a Consignación por Devolución**](#_AO06_Liquidar_Venta)
2. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Transacción – CUROLGEN05**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN05_AdministrarTransaccion.doc)
   1. El sistema envía como parámetro la siguiente información:
      1. EliminarTransaccion
         1. TransProd = <TransProdDev.TransProdID> registrado en sesión (objeto TransProdDev)
   2. El sistema recibe como parámetro la siguiente información:
      1. Verdadero
3. El sistema actualiza la siguiente información para la transacción de la Venta a Consignación registrada en sesión <TransProd.TransProdID = TransProd.TransProdID registrado en sesión (objeto TransProd)>:
   1. **TransProd**
      1. TipoFase = 6 (“Liquidado”)
      2. FechaFacturacion = Fecha y hora actual del sistema
      3. VisitaClave1 = VisitaClave registrada en sesión
      4. DiaClave1 = DiaClave registrado en sesión
      5. MFechaHora = Fecha y hora actual del sistema
      6. MUsuarioID = UsuarioId registrado en sesión
4. Si <la información no fue registrada o actualizada correctamente (ocurrió alguna excepción)>
   1. El sistema realiza roll back a la transacción
   2. El sistema presenta el mensaje con el error obtenido
   3. El sistema registra la siguiente información:
      1. Fallo = Verdadero
5. El sistema registra la siguiente información:
   1. Fallo = Falso
6. El sistema continúa en el [**paso 9.3.2**](#paso9_3_2_AO05) del flujo alterno opcional [**AO05 Liquidar Venta a Consignación**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV44_CapturarVentasConsignación.docx#_AO04_Liquidar_Venta)y envía como parámetro la siguiente información:
   1. Fallo

### [AO07 Liquidar Venta a Consignación por Cancelación](#AO07_r)

1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Clientes – CUROLGEN14**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN14_AdministrarClientes.doc)
   1. El sistema envía como parámetro la siguiente información:
      1. ActualizarSaldoCteActual
         1. Importe = [<TransProd.Saldo> registrado en sesión (objeto TransProd)] \* -1
   2. El sistema recibe la siguiente información:
      1. Verdadero
2. Si <por lo menos un producto del objeto Productos cuenta con una cantidad a devolver mayor a Cero <Productos.Devolucion > 0>>
   1. Si <el tipo de liquidación registrado en sesión indica que se generará una nueva consigna a partir del producto devuelto (TipoLiquidacion = GeneraConsignacion)>
      1. El sistema obtiene información del siguiente objeto:
         1. **ModuloMovDetalle**
            1. ModuloMovDetalleClave
            2. TipoIndice = 26 (“Venta Consignación”)
         2. El sistema registra en sesión la siguiente información del objeto obtenido:
            1. ModuloMovDetConsignacion
      2. El caso de uso incluye la funcionalidad del caso de uso [**CUADMESC13 Generar Folio**](file:///C:\Users\Route\Trunk\Analisis\EspecificacionRequerimientos\ECU_ESC\CUADMESC13_GenerarFolio.docx)
         1. El sistema envía como parámetro la siguiente información:
            1. GenerarFolio

ModuloMovDetalleClave = ModuloMovDetConsignacion.ModuloMovDetalleClave

* + - 1. El sistema recibe como parámetro y registra en sesión la siguiente información:
         1. PS\_Folio
    1. Si <se obtuvo un folio para la nueva Venta a Consignación a generar (PS\_Folio <> Null o ‘ ‘)>
       1. El sistema registra la siguiente información para el nuevo movimiento de Venta a Consignación:
          1. **TransProd**

TransProdID = Utilizar <Método KeyGen.GetId>

VisitaClave = VisitaClave registrada en sesión

DiaClave = DiaClave registrado en sesión

PCEPrecioClave = <TransProd.PCEPrecioClave> registrado en sesión (objeto TransProd)

PCEModuloMovDetClave = <TransProd.PCEModuloMovDetClave> registrado en sesión (objeto TransProd)

SubEmpresaID = <TransProd.SubEmpresaID> registrado en sesión (objeto TransProd)

ClienteClave = <TransProd.ClienteClave> registrado en sesión (objeto TransProd)

Folio = PS\_Folio registrado en sesión

Tipo = <TransProd.Tipo> registrado en sesión (objeto TransProd)

TipoPedido = <TransProd.TipoPedido> registrado en sesión (objeto TransProd)

TipoFase = 2 (“Surtido”)

TipoMovimiento = <TransProd.TipoMovimiento> registrado en sesión (objeto TransProd)

FechaHoraAlta = Fecha y hora actual del sistema

FechaCaptura = <Dia.FechaCaptura> donde <Dia.DiaClave = DiaClave registrado en sesión>

FechaEntrega = <Dia.FechaCaptura> donde <Dia.DiaClave = DiaClave registrado en sesión>

FechaFacturacion = ‘1900-01-01 00:00:00.000’

FechaSurtido = Fecha y hora actual del sistema

FechaCancelacion = ‘1900-01-01 00:00:00.000’

TipoMotivo = 12

Enviado = 0

MFechaHora = Fecha y hora actual del sistema

MUsuarioID = UsuarioId registrado en sesión

* + - * 1. El sistema registra en sesión la siguiente información:

TransProdIDNvaConsigna = TransProdID de la venta a consignación generada

* + - 1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Transacción – CUROLGEN05**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN05_AdministrarTransaccion.doc)
         1. El sistema envía como parámetro la siguiente información:

ObtenerTransaccion

TransProdID = TransProdIDNvaConsigna registrado en sesión

* + - * 1. El sistema recibe como parámetro la siguiente información

TransProd

* + - * 1. El sistema registra en sesión la información del objeto recibido para el encabezado de la transacción de la nueva venta a consignación:

TransProdNvaConsigna

* 1. El sistema obtiene información del siguiente objeto:
     1. **ModuloMovDetalle**
        1. ModuloMovDetalleClave
        2. TipoIndice = 13 (“Devolución de Clientes”)
     2. El sistema registra en sesión la siguiente información del objeto obtenido:
        1. ModuloMovDetDevoluciones
  2. El caso de uso incluye la funcionalidad del caso de uso [**CUADMESC13 Generar Folio**](file:///C:\Duxstar\Productos\Route\Trunk\Analisis\EspecificacionRequerimientos\ECU_ESC\CUADMESC13_GenerarFolio.docx)
     1. El sistema envía como parámetro la siguiente información:
        1. GenerarFolio
           1. ModuloMovDetalleClave = ModuloMovDetDevoluciones.ModuloMovDetalleClave
     2. El sistema recibe como parámetro y registra en sesión la siguiente información:
        1. PS\_Folio
  3. Si <se obtuvo un folio para la devolución de producto a generar (PS\_Folio <> Null o ‘ ‘)>
     1. El sistema registra la siguiente información para el nuevo movimiento de devolución a generar:
        1. **TransProd**
           1. TransProdID = Utilizar <Método KeyGen.GetId>
           2. VisitaClave = VisitaClave registrada en sesión
           3. DiaClave = DiaClave registrado en sesión
           4. PCEPrecioClave = <TransProd.PCEPrecioClave> registrado en sesión (objeto TransProd)
           5. PCEModuloMovDetClave = <TransProd.PCEModuloMovDetClave> registrado en sesión (objeto TransProd)
           6. ClienteClave = ClienteClave registrado en sesión
           7. Folio = PS\_Folio registrado en sesión
           8. Tipo = <ModuloMovDetDevoluciones.TipoTransProd> registrado en sesión
           9. TipoPedido = <ModuloMovDetDevoluciones.TipoPedido> registrado en sesión
           10. TipoFase = 2 (“Surtido”)
           11. TipoMovimiento = <ModuloMovDetDevoluciones.TipoMovimiento> registrado en sesión
           12. FechaHoraAlta = Fecha y hora actual del sistema
           13. FechaCaptura = <Dia.FechaCaptura> donde <Dia.DiaClave = DiaClave registrado en sesión>
           14. FechaEntrega = <Dia.FechaCaptura> donde <Dia.DiaClave = DiaClave registrado en sesión>
           15. FechaFacturacion = ‘1900-01-01 00:00:00.000’
           16. FechaSurtido = Fecha y hora actual del sistema
           17. FechaCancelacion = ‘1900-01-01 00:00:00.000’
           18. TipoMotivo = 12
           19. Enviado = 0
           20. MFechaHora = Fecha y hora actual del sistema
           21. MUsuarioID = UsuarioId registrado en sesión
        2. El sistema registra en sesión la siguiente información:
           1. TransProdIDDevolucion = TransProdID de la devolución generada
     2. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Transacción – CUROLGEN05**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN05_AdministrarTransaccion.doc)
        1. El sistema envía como parámetro la siguiente información:
           1. ObtenerTransaccion

TransProdID = TransProdIDDevolucion registrado en sesión

* + - 1. El sistema recibe como parámetro la siguiente información
         1. TransProd
      2. El sistema registra en sesión la información del objeto recibido para el encabezado de la transacción de devolución:
         1. TransProdDev

1. Si <por lo menos un producto del objeto Productos cuenta con una cantidad a liquidar mayor a Cero <Productos.Liquidar > 0>>
   1. El sistema obtiene información del siguiente objeto:
      1. **ModuloMovDetalle**
         1. ModuloMovDetalleClave
         2. TipoIndice = 9 (“Pedido”)
      2. El sistema registra en sesión la siguiente información del objeto obtenido:
         1. ModuloMovDetPedido
   2. El caso de uso incluye la funcionalidad del caso de uso [**CUADMESC13 Generar Folio**](file:///C:\Duxstar\Productos\Route\Trunk\Analisis\EspecificacionRequerimientos\ECU_ESC\CUADMESC13_GenerarFolio.docx)
      1. El sistema envía como parámetro la siguiente información:
         1. GenerarFolio
            1. ModuloMovDetalleClave = ModuloMovDetPedido.ModuloMovDetalleClave
      2. El sistema recibe como parámetro y registra en sesión la siguiente información:
         1. PS\_Folio
   3. Si <se obtuvo un folio para la nueva venta a generar (PS\_Folio <> Null o ‘ ‘)>
      1. El sistema registra la siguiente información para el nuevo movimiento de venta:
         1. **TransProd**
            1. TransProdID = Utilizar <Método KeyGen.GetId>
            2. VisitaClave = VisitaClave registrada en sesión
            3. DiaClave = DiaClave registrado en sesión
            4. PCEPrecioClave = <TransProd.PCEPrecioClave> registrado en sesión (objeto TransProd)
            5. PCEModuloMovDetClave = <TransProd.PCEModuloMovDetClave> registrado en sesión (objeto TransProd)
            6. SubEmpresaID = <TransProd.SubEmpresaID> registrado en sesión (objeto TransProd)
            7. ClienteClave = ClienteClave registrado en sesión
            8. ClientePagoID = 1 (“Efectivo”)
            9. CFVTipo = 1 (“Contado”)
            10. Folio = PS\_Folio registrado en sesión
            11. Tipo = <ModuloMovDetPedido.TipoTransProd> registrado en sesión
            12. TipoPedido = <ModuloMovDetPedido.TipoPedido> registrado en sesión
            13. TipoFase = 2 (“Surtido”)
            14. TipoMovimiento = <ModuloMovDetPedido.TipoMovimiento> registrado en sesión
            15. FechaHoraAlta = Fecha y hora actual del sistema
            16. FechaCaptura = <Dia.FechaCaptura> donde <Dia.DiaClave = DiaClave registrado en sesión>
            17. FechaEntrega = <Dia.FechaCaptura> donde <Dia.DiaClave = DiaClave registrado en sesión>
            18. FechaFacturacion = ‘1900-01-01 00:00:00.000’
            19. FechaSurtido = Fecha y hora actual del sistema
            20. FechaCancelacion = ‘1900-01-01 00:00:00.000’
            21. TipoMotivo = 12
            22. Enviado = 0
            23. MFechaHora = Fecha y hora actual del sistema
            24. MUsuarioID = UsuarioId registrado en sesión
         2. El sistema registra en sesión la siguiente información:
            1. TransProdIDVenta = TransProdID de la nueva venta generada
      2. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Transacción – CUROLGEN05**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN05_AdministrarTransaccion.doc)
         1. El sistema envía como parámetro la siguiente información:
            1. ObtenerTransaccion

TransProdID = TransProdIDVenta registrado en sesión

* + - 1. El sistema recibe como parámetro la siguiente información
         1. TransProd
      2. El sistema registra en sesión la información del objeto recibido para el encabezado de la transacción de venta:
         1. TransProdVta

1. El sistema registra la siguiente información en sesión:
   1. AbonosAnteriores = <TransProd.Total> - <TransProd.Saldo> registrado en sesión (objeto TransProd)
2. Para cada producto incluido en el objeto productos
   1. El caso de uso incluye la funcionalidad del caso de uso [**Calcular Impuestos – CUROLGEN02**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN02_CalcularImpuestos.doc)
      1. El sistema envía como parámetro la siguiente información:
         1. BuscarImpuestos
            1. ProductoClave = ProductoClave (objeto Productos)
            2. TipoImpuesto = <Cliente.TipoImpuesto> registrado en sesión (objeto Cliente)
      2. El sistema recibe como parámetro la siguiente información
         1. PS\_ImpuestosRelacionados
      3. El sistema registra en sesión la siguiente información:
         1. ListaImpuestos = PS\_ImpuestosRelacionados
   2. Si <el producto cuenta con una cantidad a liquidar mayor a Cero <Productos.Liquidar > 0>>
      1. El sistema registra en sesión la siguiente información:
         1. Partida = 0
      2. El caso de uso incluye la funcionalidad del caso de uso [**Calcular Impuestos – CUROLGEN02**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN02_CalcularImpuestos.doc)
         1. El sistema envía como parámetro la siguiente información:
            1. CalcularImpuestos

ImpuestosRelacionados = ListaImpuestos registrada en sesión

PrecioBase = Liquidar \* Precio (objeto Productos)

PrecioUnitario = Precio (objeto Productos)

* + - 1. El sistema recibe como parámetro la siguiente información
         1. PS\_TotalImpuesto
      2. El sistema registra en sesión la siguiente información:
         1. ImpuestosProducto = PS\_TotalImpuesto (formatear el importe a dos decimales).
    1. El sistema registra la siguiente información:
       1. **TransProdDetalle**
          1. TransProdID = <TransProdVta.TransProdID> registrado en sesión (objeto TransProdVta)
          2. TransProdDetalleID = Utilizar <Método KeyGen>
          3. ProductoClave = ProductoClave (objeto Productos)
          4. TipoUnidad = Unidad (objeto Productos)
          5. Partida = Última Partida + 1
          6. Cantidad = Liquidar (objeto Productos)
          7. Precio = Precio (objeto Productos)
          8. DescuentoPor = 0
          9. DescuentoImp = 0
          10. Subtotal = Liquidar \* Precio (objeto Productos)
          11. Impuesto = ImpuestosProducto registrado en sesión
          12. Total = <TransProdDetalle.Subtotal> + <TransProdDetalle.Impuesto>
          13. Promocion = 0
          14. MFechaHora = Fecha y hora actual del sistema
          15. MUsuarioID = UsuarioId registrado en sesión
          16. Enviado = 0
    2. El caso de uso incluye la funcionalidad del caso de uso [**Calcular Impuestos – CUROLGEN02**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN02_CalcularImpuestos.doc)
       1. El sistema envía como parámetro la siguiente información:
          1. GuardarDetalle

TransProdId = <TransProdDetalle.TransProdID> actual

TransProdDetalleId = <TransProdDetalle.TransProdDetalleID> actual

ImpuestosRelacionados = ListaImpuestos

* + - 1. El sistema recibe como parámetro la siguiente información
         1. Verdadero
    1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Transacción – CUROLGEN05**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN05_AdministrarTransaccion.doc)
       1. El sistema envía como parámetro la siguiente información:
          1. CalcularTotalesTransaccion

TransProd = <TransProdVta.TransProdID> registrado en sesión (objeto TransProdVta)

* + - 1. El sistema recibe la siguiente información
         1. TransProd
      2. El sistema actualiza la información del objeto registrado en sesión:
         1. TransProdVta
    1. El sistema obtiene y registra en sesión la siguiente información para el producto actual, donde <ProductoDetalle.ProductoClave = ProductoClave (objeto Productos) y ProductoDetalle.PRUTipoUnidad = Unidad (objeto Productos)>:
       1. **ProductoDetalle**
          1. Factor
    2. El caso de uso incluye la funcionalidad del caso de uso [**Calcular Cuotas – CUROLGEN07**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN07_CalcularCuotas.doc)
       1. El sistema envía como parámetro la siguiente información:
          1. VerificarCuotas

VendedorId = VendedorId registrado en sesión

ClienteClave = <Cliente.ClienteClave> registrado en sesión (objeto Cliente)

ProductoClave = ProductoClave (objeto Productos)

Cantidad = Cantidad (objeto Productos) \* Factor registrado en sesión.

TipoCuota = 1 (“Producto”)

* + - 1. El sistema recibe la siguiente información
         1. PS\_CuoVen
    1. El caso de uso incluye la funcionalidad del caso de uso [**Calcular Cuotas – CUROLGEN07**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN07_CalcularCuotas.doc)
       1. El sistema envía como parámetro la siguiente información:
          1. VerificarCuotas

VendedorId = VendedorId registrado en sesión

ClienteClave = <Cliente.ClienteClave> registrado en sesión (objeto Cliente)

ProductoClave = ProductoClave (objeto Productos)

Cantidad = Cantidad (objeto Productos) \* Factor registrado en sesión.

TipoCuota = 3 (“Kilos”)

* + - 1. El sistema recibe la siguiente información
         1. PS\_CuoVen
  1. Si <el producto cuenta con una cantidad a devolver mayor a Cero <Productos.Devolucion > 0>>
     1. El sistema registra en sesión la siguiente información:
        1. Partida = 0
     2. El caso de uso incluye la funcionalidad del caso de uso [**Calcular Impuestos – CUROLGEN02**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN02_CalcularImpuestos.doc)
        1. El sistema envía como parámetro la siguiente información:
           1. CalcularImpuestos

ImpuestosRelacionados = ListaImpuestos registrada en sesión

PrecioBase = Devolucion \* Precio (objeto Productos)

PrecioUnitario = Precio (objeto Productos)

* + - 1. El sistema recibe como parámetro la siguiente información
         1. PS\_TotalImpuesto
      2. El sistema registra en sesión la siguiente información:
         1. ImpuestosProducto = PS\_TotalImpuesto (formatear el importe a dos decimales).
    1. El sistema registra la siguiente información:
       1. **TransProdDetalle**
          1. TransProdID = <TransProdDev.TransProdID> registrado en sesión (objeto TransProdDev)
          2. TransProdDetalleID = <TransProdDetalle.TransProdDetalleID> correspondiente a la partida de la Venta a Conisgnación de la cual se realiza la devolución
          3. ProductoClave = ProductoClave (objeto Productos)
          4. TipoUnidad = Unidad (objeto Productos)
          5. Partida = Última Partida + 1
          6. Cantidad = Devolucion (objeto Productos)
          7. Precio = Precio (objeto Productos)
          8. DescuentoPor = 0
          9. DescuentoImp = 0
          10. Subtotal = Devolucion \* Precio (objeto Productos)
          11. Impuesto = ImpuestosProducto registrado en sesión
          12. Total = <TransProdDetalle.Subtotal> + <TransProdDetalle.Impuesto>
          13. Promocion = 0
          14. MFechaHora = Fecha y hora actual del sistema
          15. MUsuarioID = UsuarioId registrado en sesión
          16. Enviado = 0
    2. El caso de uso incluye la funcionalidad del caso de uso [**Calcular Impuestos – CUROLGEN02**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN02_CalcularImpuestos.doc)
       1. El sistema envía como parámetro la siguiente información:
          1. GuardarDetalle

TransProdId = <TransProdDetalle.TransProdID> actual

TransProdDetalleId = <TransProdDetalle.TransProdDetalleID> actual

ImpuestosRelacionados = ListaImpuestos

* + - 1. El sistema recibe como parámetro la siguiente información
         1. Verdadero
    1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Transacción – CUROLGEN05**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN05_AdministrarTransaccion.doc)
       1. El sistema envía como parámetro la siguiente información:
          1. CalcularTotalesTransaccion

TransProd = <TransProdDev.TransProdID> registrado en sesión (objeto TransProdDev)

* + - 1. El sistema recibe la siguiente información
         1. TransProd
      2. El sistema actualiza la información del objeto registrado en sesión:
         1. TransProdDev
    1. El sistema actualiza la siguiente información para la transacción de la Venta a Consignación registrada en sesión <TransProd.TransProdID = TransProd.TransProdID registrado en sesión (objeto TransProd)>:
       1. **TransProd**
          1. Saldo = *Saldo* – <TransProdDetalle.Total> actual
    2. El sistema registra la siguiente información:
       1. **TrpTpd**
          1. TransProdID = <TransProdDev.TransProdID> registrado en sesión (objeto TransProdDev)
          2. TransProdID1 = <TransProd.TransProdID> registrado en sesión (objeto TransProd)
          3. TransProdDetalleID = <TransProdDetalle.TransProdDetalleID> actual
          4. Cantidad = <TransProdDetalle.Cantidad> actual
          5. Precio = <TransProdDetalle.Precio> actual
          6. Subtotal = <TransProdDetalle.Subtotal> actual
          7. Impuesto = <TransProdDetalle.Impuesto> actual
          8. Total = <TransProdDetalle.Total> actual
          9. MFechaHora = Fecha y hora actual del sistema
          10. MUsuarioID = UsuarioId registrado en sesión
    3. Si <el tipo de liquidación registrado en sesión indica que se generará una devolución a partir del producto devuelto (TipoLiquidacion = GeneraDevolucion)>
       1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Inventario Móvil – CUROLGEN13**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN13_AdministrarInventario.doc)
          1. El sistema envía como parámetro la siguiente información:

ActualizarInventario

ProductoClave = ProductoClave (objeto Productos)

TipoUnidad = Unidad (objeto Productos)

Cantidad = Devolucion (objeto Productos)

TipoTransProd = <TransProdDev.Tipo> registrado en sesión (objeto TransProdDev)

TipoMovimiento = <TransProdDev.TipoMovimiento> registrado en sesión (objeto TransProdDev)

AlmacenID = AlmacenID registrado en sesión

GrupoMotivo = GrupoMotivo registrado en sesión

Cancelacion = 0

* + - * 1. El sistema recibe la siguiente información:

OP\_Verdadero

* + - 1. Si <al cliente se le debe realizar el manejo del préstamo de envase <Cliente.Prestamo = 1 donde Cliente.ClienteClave = TransProd.ClienteClave (objeto TransProd registrado en sesión)>
         1. Si <el producto actual se puede prestar <ProductoDetalle.ProductoClave = ProductoClave (objeto Productos)> y <ProductoDetalle.PRUTipoUnidad = Unidad (objeto Productos)> y <ProductoDetalle.Prestamo = 1>>

El sistema obtiene información del Saldo del préstamo del producto, donde <ProductoPrestamoCli.ClienteClave = TransProd.ClienteClave (objeto TransProd registrado en sesión)> y <ProductoPrestamoCli.ProductoClave = ProductoDetalle.ProductoDetClave donde ProductoDetalle.ProductoClave = ProductoClave (objeto Productos) y ProductoDetalle.PRUTipoUnidad = Unidad (objeto Productos)>:

**ProductoPrestamoCli**

Saldo

Si <no se obtuvo un registro de ProductoPrestamoCli para el Cliente y producto actuales>

El caso de uso incluye la funcionalidad del caso de uso [**Administrar Préstamo de Producto – CUROLGEN15**](file:///C:\Users\Belem\AppData\Roaming\Microsoft\ECU_GEN\CUROLGEN15_AdministrarPrestamoDeProducto.docx)y le envía como parámetro la siguiente información:

CrearPrestamo

ClienteClave = <TransProd.ClienteClave> (objeto TransProd registrado en sesión)

ProductoClave = <ProductoDetalle.ProductoDetClave> donde <ProductoDetalle.ProductoClave = ProductoClave (objeto Productos) y ProductoDetalle.PRUTipoUnidad = Unidad (objeto Productos)>

El sistema recibe la siguiente información:

OP\_Verdadero

El caso de uso incluye la funcionalidad del caso de uso [**Administrar Préstamo de Producto – CUROLGEN15**](file:///C:\Users\Belem\AppData\Roaming\Microsoft\ECU_GEN\CUROLGEN15_AdministrarPrestamoDeProducto.docx)y le envía como parámetro la siguiente información:

ActualizarPrestamo

ClienteClave = <TransProd.ClienteClave> (objeto TransProd registrado en sesión)

ProductoClave = <ProductoDetalle.ProductoDetClave> donde <ProductoDetalle.ProductoClave = ProductoClave (objeto Productos) y ProductoDetalle.PRUTipoUnidad = Unidad (objeto Productos)>

TipoTransProd = <TransProdDev.Tipo> registrado en sesión (objeto TransProdDev)

Cantidad = Devolucion (objeto Productos)

TipoUnidad = Unidad (objeto Productos)

Vendido = 2

El sistema recibe la siguiente información:

OP\_Verdadero

* + 1. Si <el tipo de liquidación registrado en sesión indica que se generará una nueva consigna a partir del producto devuelto (TipoLiquidacion = GeneraConsignacion)>
       1. El sistema registra la siguiente información:
          1. **TransProdDetalle**

TransProdID = <TransProdNvaConsigna.TransProdID> registrado en sesión (objeto TransProdNvaConsigna)

TransProdDetalleID = Utilizar <Método KeyGen>

ProductoClave = ProductoClave (objeto Productos)

TipoUnidad = Unidad (objeto Productos)

Partida = Última Partida + 1

Cantidad = Devolucion (objeto Productos)

Precio = Precio (objeto Productos)

DescuentoPor = 0

DescuentoImp = 0

Subtotal = Devolucion \* Precio (objeto Productos)

Impuesto = ImpuestosProducto registrado en sesión

Total = <TransProdDetalle.Subtotal> + <TransProdDetalle.Impuesto>

Promocion = 0

MFechaHora = Fecha y hora actual del sistema

MUsuarioID = UsuarioId registrado en sesión

Enviado = 0

* + - 1. El caso de uso incluye la funcionalidad del caso de uso [**Calcular Impuestos – CUROLGEN02**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN02_CalcularImpuestos.doc)
         1. El sistema envía como parámetro la siguiente información:

GuardarDetalle

TransProdId = <TransProdDetalle.TransProdID> actual

TransProdDetalleId = <TransProdDetalle.TransProdDetalleID> actual

ImpuestosRelacionados = ListaImpuestos

* + - * 1. El sistema recibe como parámetro la siguiente información

Verdadero

* + - 1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Transacción – CUROLGEN05**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN05_AdministrarTransaccion.doc)
         1. El sistema envía como parámetro la siguiente información:

CalcularTotalesTransaccion

TransProd = <TransProdNvaConsigna.TransProdID> registrado en sesión (objeto TransProdNvaConsigna

* + - * 1. El sistema recibe la siguiente información

TransProd

* + - * 1. El sistema actualiza la información del objeto registrado en sesión:

TransProdNvaConsigna

1. Si <por lo menos un producto del objeto Productos cuenta con una cantidad a devolver mayor a Cero <Productos.Devolucion > 0>>
   1. Si <el tipo de liquidación registrado en sesión indica que se generará una nueva consigna a partir del producto devuelto (TipoLiquidacion = GeneraConsignacion)>
      1. El caso de uso incluye la funcionalidad del caso de uso [**Aplicar Promociones – CUROLMOV19**](file:///C:\Users\Belem\AppData\Roaming\Microsoft\Word\CUROLMOV19_Aplicar%20Promociones.doc)
         1. El sistema envía como parámetro la siguiente información:
            1. TransProd = TransProdNvaConsigna registrado en sesión (objeto TransProdNvaConsigna)
      2. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Transacción – CUROLGEN05**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN05_AdministrarTransaccion.doc)
         1. El sistema envía como parámetro la siguiente información:
            1. CalcularTotalesTransaccion

TransProd = <TransProdNvaConsigna.TransProdID> registrado en sesión (objeto TransProdNvaConsigna)

* + - 1. El sistema recibe la siguiente información
         1. TransProd
      2. El sistema actualiza la información del objeto registrado en sesión:
         1. TransProdNvaConsigna
    1. El sistema actualiza la siguiente información para la transacción de la nueva Venta a Consignación registrada en sesión <TransProd.TransProdID = TransProdNvaConsigna.TransProdID registrado en sesión (objeto TransProd)>:
       1. **TransProd**
          1. Saldo = <TransProdNvaConsigna.Total> registrado en sesión (objeto TransProdNvaConsigna)
          2. MFechaHora = Fecha y hora actual del sistema
          3. MUsuarioID = UsuarioId registrado en sesión
    2. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Clientes – CUROLGEN14**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN14_AdministrarClientes.doc)
       1. El sistema envía como parámetro la siguiente información:
          1. ActualizarSaldoCteActual

Importe = <TransProdNvaConsigna.Saldo>

* + - 1. El sistema recibe la siguiente información:
         1. Verdadero
  1. El sistema realiza la confirmación del folio generado para el movimiento de devolución
  2. Si <el tipo de liquidación registrado en sesión indica que se generará una nueva consigna a partir del producto devuelto (TipoLiquidacion = GeneraConsignacion)>
     1. El sistema realiza la confirmación del folio generado para el nuevo movimiento de consignación

1. El sistema actualiza la siguiente información para la transacción de la Venta a Consignación registrada en sesión <TransProd.TransProdID = TransProd.TransProdID registrado en sesión (objeto TransProd)>:
   1. **TransProd**
      1. Si <se generó una venta a partir de la venta a consignación, es decir, si existe un objeto TransProdVta registrado en sesión>
         1. FacturaID = <TransProdVta.TransProdID> registrado en sesión (objeto TransProdVta)
      2. TipoFase = 8 (“Cancelada por Liquidación”)
      3. FechaFacturacion = Fecha y hora actual del sistema
      4. VisitaClave1 = VisitaClave registrada en sesión
      5. DiaClave1 = DiaClave registrado en sesión
      6. Saldo = 0
      7. MFechaHora = Fecha y hora actual del sistema
      8. MUsuarioID = UsuarioId registrado en sesión
2. Si <por lo menos un producto del objeto Productos cuenta con una cantidad a liquidar mayor a Cero <Productos.Liquidar > 0>>
   1. El caso de uso incluye la funcionalidad del caso de uso [**Aplicar Promociones – CUROLMOV19**](file:///C:\Users\Belem\AppData\Roaming\Microsoft\Word\CUROLMOV19_Aplicar%20Promociones.doc)
      1. El sistema envía como parámetro la siguiente información:
         1. TransProd = TransProdVta registrado en sesión (objeto TransProdVta)
   2. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Transacción – CUROLGEN05**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN05_AdministrarTransaccion.doc)
      1. El sistema envía como parámetro la siguiente información:
         1. CalcularTotalesTransaccion
            1. TransProd = <TransProdVta.TransProdID> registrado en sesión (objeto TransProdVta)
      2. El sistema recibe la siguiente información
         1. TransProd
      3. El sistema actualiza la información del objeto registrado en sesión:
         1. TransProdVta
   3. El sistema realiza la confirmación del folio generado para el movimiento de venta
   4. El sistema presenta el siguiente título de pantalla:
      1. Venta
      2. Fecha: Se presenta la fecha del día de trabajo actual (DiaClave), a partir de la información registrada en sesión.
   5. El sistema presenta y solicita la siguiente información:
      1. **Folio:** <TransProdVta.Folio> registrado en sesión (objeto TransProdVta). Mostrar de sólo lectura.
      2. **Fase:** Se presenta la descripción de la fase de la transacción, para ello ir a <VAVDescripcion> y obtener <VAVDescripcion.Descripcion> donde <VAVDescripcion.VARCodigo = ‘TRPFSE> y <VAVDescripcion.VAVClave = TransProdVta.TipoFase > (objeto TransProdVta registrado en sesión), de acuerdo al tipo de lenguaje configurado para el sistema. Mostrar de sólo lectura.
      3. **Fecha:** <TransProdVta.FechaCaptura> registrado en sesión (objeto TransProdVta). Mostrar de sólo lectura en formato dd/mm/aaaa.
      4. **Lista Precios:** <TransProdVta.PCEPrecioClave> registrado en sesión (objeto TransProdVta). Mostrar de sólo lectura.
      5. **Subtotal:** <TransProdVta.Subtotal> registrado en sesión (objeto TransProdVta). Mostrar de sólo lectura, precedido del signo “$”.
      6. **Impuesto:** <TransProdVta.Impuesto> registrado en sesión (objeto TransProdVta). Mostrar de sólo lectura, precedido del signo “$”.
      7. **Total:** <TransProdVta.Total> registrado en sesión (objeto TransProdVta). Mostrar de sólo lectura, precedido del signo “$”.
      8. **Comentarios:** Permitir capturar notas o comentarios para la venta generada.
   6. Si <el actor selecciona la opción **Imprimir**>
      1. Si <se creó una nueva venta a partir de la consignación (existe un objeto TransProdVta registrado en sesión)>
         1. El sistema registra en sesión la siguiente información:
            1. SaldoFinal = <TransProdVta.Total> - AbonosAnteriores (registrados en sesión)
         2. Si <el saldo final de la venta generada es menor a Cero, una vez que se le aplican los abonos anteriores realizados a la Venta a Consignación (SaldoFinal < 0)>
            1. El sistema presenta el mensaje “[E0697] La Devolución no puede ser efectuada, queda un saldo a Favor del Cliente”
            2. El sistema continúa en el paso donde fue invocado
         3. El sistema actualiza la siguiente información para la venta generada a partir de la Venta a Consignación, donde <TransProd.TransProdID = TransProdVta.TransProdID registrado en sesión (objeto TransProdVta)>
            1. **TransProd**

Saldo = SaldoFinal registrado en sesión

* + - 1. Si <el parámetro de configuración indica que se tiene activo el Pago Automático (PagoAutomatico = 1) de acuerdo a la información registrada en sesión> Y si <el parámetro de configuración indica que la cobranza se realiza sobre Ventas (TipoCobranza = 1)> O si <el parámetro de configuración indica que la cobranza se realiza sobre Ambas (TipoCobranza = 2) y la configuración del cliente indica que la cobranza se le realiza sobre Ventas <Cliente.TipoFiscal = 1 donde Cliente.ClienteClave = Cliente.ClienteClave (objeto Cliente registrado en sesión)>>
         1. El sistema actualiza la siguiente información para la venta generada a partir de la Venta a Consignación, donde <TransProd.TransProdID = TransProdVta.TransProdID registrado en sesión (objeto TransProdVta)>

**TransProd**

FechaCobranza = Fecha actual

* + - * 1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Abonos – CUROLGEN08**](file:///C:\Users\Belem\AppData\Roaming\Microsoft\ECU_GEN\CUROLGEN08_AdministrarAbonos.doc)

El sistema envía como parámetro la siguiente información:

CrearAbonoPedido

TransProd = <TransProdVta.TransProdID> registrado en sesión (objeto TransProdVta)

El sistema recibe la siguiente información:

Verdadero

* + - * 1. Si <el parámetro de configuración indica que se tiene activa la Preliquidación (PreLiquidacion = 1) de acuerdo a la información registrada en sesión>

El sistema obtiene la siguiente información, donde <PreLiquidacion.Enviado = 0>:

**PreLiquidacion**

PLIId

MontoTotal

Si <la forma de venta de la venta generada a partir de la Venta a Consignación fue de Contado <TransProdVta.CFVTipo = 1> y en Efectivo <TransProdVta.ClientePagoId = 1>>

Si <se obtuvo un registro de PreLiquidacion>, el sistema actualiza la siguiente información para el registro de PreLiquidacion obtenido:

**PreLiquidacion**

MontoTotal = *MontoTotal* + SaldoFinal registrado en sesión

El sistema continúa en el [**paso 8.6.1.4.3.3**](#paso8_6_1_4_3_3_AO01)

El sistema registra la siguiente información:

**PreLiquidacion**

PLIId = Utilizar <Método KeyGen.GetId>

FechaPreLiquidacion = Fecha y hora actual del sistema

MontoTotal = SaldoFinal registrado en sesión

Enviado = o

El sistema actualiza la siguiente información para la venta generada a partir de la Venta a Consignación, donde <TransProd.TransProdID = TransProdVta.TransProdID registrado en sesión (objeto TransProdVta)>:

**TransProd**

PLIId = PLIId del registro de PreLiquidacion obtenido o generado

* + - 1. El sistema actualiza la siguiente información para la venta generada a partir de la Venta a Consignación, donde <TransProd.TransProdID = TransProdVta.TransProdID registrado en sesión (objeto TransProdVta)>
         1. **TransProd**

Notas = Contenido del campo “Comentarios”.

MFechaHora = Fecha y hora actual del sistema

MUsuarioID = UsuarioId registrado en sesión

* + - 1. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Clientes – CUROLGEN14**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN14_AdministrarClientes.doc)
         1. El sistema envía como parámetro la siguiente información:

ActualizarSaldoCteActual

Importe = <TransProdVta.Saldo> registrado en sesión (objeto TransProd)

* + - * 1. El sistema recibe la siguiente información:

Verdadero

// Se transfieren a la venta los abonos realizados en el día de trabajo a la consignación

* + - 1. El sistema actualiza la siguiente información, donde <AbnTrp.TransProdID = TransProd.TransProdID (objeto TransProd registrado en sesión)>
         1. **AbnTrp**

TransProdID = <TransProdVta.TransProdID (objeto TransProdVta registrado en sesión)>

* + 1. El sistema continúa con el flujo alterno opcional [**AO08 Terminar Proceso**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV44_CapturarVentasConsignación.docx#_AO08_Terminar_Proceso)y le envía como parámetro la siguiente información:
       1. Falso
    2. Si <la información no fue registrada o actualizada correctamente (ocurrió alguna excepción)>
       1. El sistema presenta el mensaje con el error obtenido
       2. El sistema continúa con el flujo alterno opcional [**AO08 Terminar Proceso**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV44_CapturarVentasConsignación.docx#_AO08_Terminar_Proceso)y le envía como parámetro la siguiente información:
          1. Verdadero
  1. Si <el actor selecciona la opción **Recalcular**>
     1. El caso de uso incluye la funcionalidad del caso de uso [**Calcular Promociones – CUROLGEN03**](file:///C:\Users\Belem\AppData\Roaming\Microsoft\ECU_GEN\CUROLGEN03_CalcularPromociones.doc)
        1. El sistema envía como parámetro la siguiente información:
           1. Preparar
     2. El caso de uso incluye la funcionalidad del caso de uso [**Aplicar Promociones – CUROLMOV19**](file:///C:\Users\Belem\AppData\Roaming\Microsoft\Word\CUROLMOV19_Aplicar%20Promociones.doc)
        1. El sistema envía como parámetro la siguiente información:
           1. TransProd = TransProdVta registrado en sesión (objeto TransProdVta)
     3. El caso de uso incluye la funcionalidad del caso de uso [**Administrar Transacción – CUROLGEN05**](file:///C:\Amesol\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_GEN\CUROLGEN05_AdministrarTransaccion.doc)
        1. El sistema envía como parámetro la siguiente información:
           1. CalcularTotalesTransaccion

TransProd = <TransProdVta.TransProdID> registrado en sesión (objeto TransProd)

* + - 1. El sistema recibe la siguiente información
         1. TransProd
      2. El sistema actualiza la información del objeto registrado en sesión:
         1. TransProdVta
    1. El sistema continúa en el [**paso 8.5**](#paso8_5_AO07) del flujo alterno opcional [**AO07 Liquidar Venta a Consignación por Cancelación**](#_AO07_Liquidar_Venta)
  1. Si <el actor selecciona la opción **Regresar**>
     1. El sistema realiza roll back a la transacción
     2. El sistema continúa en el [**paso 6**](#paso6_AO05) del flujo alterno opcional [**AO05 Liquidar Venta a Consignación**](#_AO04_Liquidar_Venta)

1. Si <la información no fue registrada o actualizada correctamente (ocurrió alguna excepción)>
   1. El sistema realiza roll back a la transacción
   2. El sistema presenta el mensaje con el error obtenido
   3. El sistema registra la siguiente información:
      1. Fallo = Verdadero
2. El sistema registra la siguiente información:
   1. Fallo = Falso
3. El sistema continúa en el [**paso 9.4.3**](#paso9_4_3_AO05) del flujo alterno opcional [**AO05 Liquidar Venta a Consignación**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV44_CapturarVentasConsignación.docx#_AO04_Liquidar_Venta)y envía como parámetro la siguiente información:
   1. Fallo

### 

### [AO08 Terminar Proceso](#AO08_r)

1. El sistema recibe la siguiente información enviada como parámetro:
   1. Verdadero/Falso
2. Si <la información recibida como parámetro es igual a Falso>:
   1. Si <el Vendedor tiene activo el parámetro para validar el número de clientes distintos que pueden tener consignas abiertas <Vendedor.ValidaLimiteConsignas = 1>>
      1. El sistema obtiene información de las Ventas a Consignación que se encuentran abiertas (no liquidadas) para el cliente actual
         1. **TransProd**
            1. TransProdID
            2. VisitaClave = <Visita.VisitaClave> donde <Visita.ClienteClave = ClienteClave registrado en sesión>
            3. Tipo = 24
            4. TipoFase = 2
      2. Si <se obtuvo una o más Ventas a Consignación sin liquidar para el cliente actual>
         1. El sistema continúa en el [**paso 2.2**](#PasoCommitTerminarProceso) del flujo alterno opcional [**AO08 Terminar Proceso**](file:///C:\Duxstar\Productos\RouteLite\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV44_CapturarVentasConsignación.docx#_AO08_Terminar_Proceso)
      3. Si <NO se obtuvieron Ventas a Consignación sin liquidar para el cliente actual>
         1. Si <el cliente actual se encuentra en el listado de clientes registrados con consignas abiertas para el vendedor <VendedorConsigAb.ClienteClave *INCLUYE A* ClienteClave registrado en sesión>>
            1. El sistema elimina la siguiente información, donde <VendedorConsigAb.VendedorID = VendedorId registrado en sesión> y <VendedorConsigAb.ClienteClave = ClienteClave registrado en sesión>:

**VendedorConsigAb**

VendedorID

ClienteClave

MFechaHora

MUsuarioID

* 1. El sistema realiza la confirmación de la transacción (Commit) correspondiente en el sistema:
     1. TransProd
     2. TransProdDetalle
     3. TPDImpuesto
     4. TpdDes
     5. TpdDesVendedor
     6. TrpPrp
     7. TrpTpd
     8. Inventario
     9. VendedorConsigAb (\*\*en caso de aplicar)
  2. El sistema obtiene la siguiente información:
     1. **MOTConfiguracion**
        1. MensajeImpresion
     2. **ConfigParametro**
        1. Parametro = NumImpresiones
        2. Identificador = ModuloMovDetalle Registrado en sesión
        3. Valor
  3. Si <el parámetro de configuración indica que se debe presentar el mensaje de impresión (MensajeImpresion = 1)>
     1. Si <el parámetro de configuración indica que NO se puede cancelar una venta a consignación por liquidación (CancConsigLiqui = 0) de acuerdo a la información registrada en sesión>
        1. El sistema presenta el mensaje “[P0103] ¿Desea imprimir el ticket?”
           1. Si <el actor selecciona la opción **No**>

El sistema continúa en el [**paso 1**](#FIN)**8** del flujo básico

* + - * 1. Si <el actor selecciona la opción **Si**>

El sistema obtiene la siguiente información de la transacción de Venta a Consignación registrada en sesión (objeto TransProd):

**TransProd**

TransProdID

Tipo

Folio

FechaCaptura

Total

TipoFase

ClienteClave

DiaClave

El sistema envía al dispositivo de impresión el recibo del movimiento, incluyendo la funcionalidad del caso de uso **Imprimir Recibos – RNGENXX** y enviando como parámetros la siguiente información:

ListaDocumentos

TransProdID = <TransProd.TransProdID>

Tipo = <TransProd.Tipo>

ValorReferencia = ‘TRPTIPO’

TipoRecibo = ‘TRP’

Folio = <TransProd.Folio>

DescTipo = (Cadena vacía)

Fecha = <TransProd.FechaCaptura>

TransProdTotal = <TransProd.Total>

TipoFase = <TransProd.TipoFase>

ClienteClave = <TransProd.ClienteClave>

DiaClave = <TransProd.DiaClave>

SubEmpresaId = Null

FactElect = 0

LogoSoloPrimerRecibo = 1

Impresiones = Impresiones + 1

~~El sistema continúa en el~~ [**~~paso 1~~**](#FIN)**~~8~~** ~~del flujo básico~~

Si <el parámetro de número de impresiones <ConfigParametro.Parametro = NumImpresiones> se encuentra en 0 <ConfigParametro.Valor = 0> o no se encontró ningún valor para esta actividad>

El sistema continúa en el [paso 2.3.1.1](#PasoReimpresion2) del flujo [AO08 Terminar Proceso](#AO08)

Si <Se encontró un parámetro para esta actividad y es mayor a cero “0”<ConfigParametro.Valor > 0>

Si <Impresiones < <ConfigParametro.Valor>>

El sistema continúa en el paso 2.3.1.1.2.1 del flujo [AO08 Terminar Proceso](#AO08)

El sistema continúa en el [**paso 1**](#FIN)**8** del flujo básico

* + 1. Si <el parámetro de configuración indica que SI se puede cancelar una venta a consignación por liquidación (CancConsigLiqui = 1) de acuerdo a la información registrada en sesión>
       1. El sistema presenta el mensaje “[P0103] ¿Desea imprimir el ticket?”
          1. Si <el actor selecciona la opción **No**>

El sistema continúa en el [**paso 1**](#FIN)**8** del flujo básico

* + - * 1. Si <el actor selecciona la opción **Si**>

Si <se creó una nueva venta a partir de la consignación (existe un objeto TransProdVta registrado en sesión)>

El sistema obtiene la siguiente información de la transacción de la nueva Venta registrada en sesión (objeto TransProdVta):

**TransProd**

TransProdID

Tipo

Folio

FechaCaptura

Total

TipoFase

ClienteClave

DiaClave

El sistema envía al dispositivo de impresión el recibo del movimiento, incluyendo la funcionalidad del caso de uso **Imprimir Recibos – RNGENXX** y enviando como parámetros la siguiente información:

ListaDocumentos

TransProdID = <TransProd.TransProdID>

Tipo = <TransProd.Tipo>

ValorReferencia = ‘TRPTIPO’

TipoRecibo = ‘TRP’

Folio = <TransProd.Folio>

DescTipo = (Cadena vacía)

Fecha = <TransProd.FechaCaptura>

TransProdTotal = <TransProd.Total>

TipoFase = <TransProd.TipoFase>

ClienteClave = <TransProd.ClienteClave>

DiaClave = <TransProd.DiaClave>

SubEmpresaId = Null

FactElect = 0

LogoSoloPrimerRecibo = 1

Impresiones = Impresiones + 1

Si <se creó una devolución de producto a partir de la consignación (existe un objeto TransProdDev registrado en sesión)>

El sistema obtiene la siguiente información de la transacción de la devolución registrada en sesión (objeto TransProdDev):

**TransProd**

TransProdID

Tipo

Folio

FechaCaptura

Total

TipoFase

ClienteClave

DiaClave

El sistema envía al dispositivo de impresión el recibo del movimiento, incluyendo la funcionalidad del caso de uso **Imprimir Recibos – RNGENXX** y enviando como parámetros la siguiente información:

ListaDocumentos

TransProdID = <TransProd.TransProdID>

Tipo = <TransProd.Tipo>

ValorReferencia = ‘TRPTIPO’

TipoRecibo = ‘TRP’

Folio = <TransProd.Folio>

DescTipo = (Cadena vacía)

Fecha = <TransProd.FechaCaptura>

TransProdTotal = <TransProd.Total>

TipoFase = <TransProd.TipoFase>

ClienteClave = <TransProd.ClienteClave>

DiaClave = <TransProd.DiaClave>

SubEmpresaId = Null

FactElect = 0

LogoSoloPrimerRecibo = 1

Impresiones = Impresiones + 1

Si <se creó una nueva consigna a partir de la consignación anterior (existe un objeto TransProdNvaConsigna registrado en sesión)>

El sistema obtiene la siguiente información de la transacción de la nueva Venta a Consignación registrada en sesión (objeto TransProdNvaConsigna):

**TransProd**

TransProdID

Tipo

Folio

FechaCaptura

Total

TipoFase

ClienteClave

DiaClave

El sistema envía al dispositivo de impresión el recibo del movimiento, incluyendo la funcionalidad del caso de uso **Imprimir Recibos – RNGENXX** y enviando como parámetros la siguiente información:

ListaDocumentos

TransProdID = <TransProd.TransProdID>

Tipo = <TransProd.Tipo>

ValorReferencia = ‘TRPTIPO’

TipoRecibo = ‘TRP’

Folio = <TransProd.Folio>

DescTipo = (Cadena vacía)

Fecha = <TransProd.FechaCaptura>

TransProdTotal = <TransProd.Total>

TipoFase = <TransProd.TipoFase>

ClienteClave = <TransProd.ClienteClave>

DiaClave = <TransProd.DiaClave>

SubEmpresaId = Null

FactElect = 0

LogoSoloPrimerRecibo = 1

Impresiones = Impresiones + 1

~~El sistema continúa en el~~ [**~~paso 1~~**](#FIN)**~~8~~** ~~del flujo básico~~

Si <el parámetro de número de impresiones <ConfigParametro.Parametro = NumImpresiones> se encuentra en 0 <ConfigParametro.Valor = 0> o no se encontró ningún valor para esta actividad>

El sistema continúa en el [paso 2.3.1.1](#PasoReimpresion2) del flujo [AO08 Terminar Proceso](#AO08)

Si <Se encontró un parámetro para esta actividad y es mayor a cero “0”<ConfigParametro.Valor > 0>

Si <Impresiones < <ConfigParametro.Valor>>

El sistema continúa en el paso 2.3.1.1.2.1 del flujo [AO08 Terminar Proceso](#AO08)

El sistema continúa en el [**paso 1**](#FIN)**8** del flujo básico

* 1. Si <el parámetro de configuración indica que no se debe presentar el mensaje de impresión (MensajeImpresion = 0)>
     1. El sistema continúa en el [**paso 1**](#FIN)**8** del flujo básico
  2. Si <la configuración indica “Envío Correo” o “Envío SMS” <MOTConfiguracion.MensajeImpresion = 2 ó 3>
     1. El caso de uso incluye la funcionalidad del caso de uso [**Enviar Ticket por Correo o SMS – CUROLMOV85**](file:///C:\Duxstar\Productos\eRoute\Trunk\Analisis\EspecificacionRequerimientos\ECU_MOV\CUROLMOV85_EnviarTicketCorreoSMS.docx)
        1. El sistema envía como parámetro la siguiente información:
           1. TipoEnvio = <MOTConfiguracion.MensajeImpresion>
           2. TransProdId = <TransProd.TransProdID> registrado en sesión (objeto TransProd)
           3. ABNId = Null
        2. El sistema recibe como parámetro la siguiente información:
           1. EnvioTicket = Verdadero/Falso
     2. El sistema continúa en el siguiente paso de donde fue invocado

1. Si <ocurrió un fallo durante la liquidación de la consignación (Fallo = Verdadero)>:
   1. El sistema elimina la siguiente información de la transacción actual, realizando roll back a la misma:
      1. TransProd
      2. TransProdDetalle
      3. TPDImpuesto
      4. TpdDes
      5. TpdDesVendedor
      6. TrpPrp
      7. TrpTpd
      8. Inventario
2. El sistema continúa en el [**paso**](#PasoPresentaTitulo) **7** del flujo básico

### Generales

N/A

# Poscondiciones

## Generales

1. Registro, actualización y eliminación de TransProd y TransProdDetalle.
2. Registro y eliminación de TrpTpd, TrpPrp, TPDImpuesto, TPDDes, TPDDesVendedor.
3. Registro de Abono, ABNDetalle.
4. Registro y actualización de AbnTrp y PreLiquidacion.
5. Actualización de Inventario.
6. Registro y actualización de TRPVtaAcreditada.
7. Registro y eliminación de VendedorConsigAb.

# Firmas de Aceptación

|  |
| --- |
|  |
| José María Alcalá |
| Gerente de Ingeniería / Departamento de Ingeniería – Duxstar Solutions |
| 10/05/2014 |

|  |
| --- |
|  |
| Belem Lizeth Jiménez Arévalo |
| Analista de Sistemas / Departamento de Análisis y Diseño – Duxstar Solutions |
| 10/05/2014 |