**HISTÓRICO DE REVISIÓN**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión** | **Descripción** | **Autor** |
| 18/02/2019 | 1.0 | Folio CAI 0005285  - En caso de estar activo el parámetro CapturaCadLote, al capturar manual o por código de barras, mostrar pantalla emergente con listado de lotes y su existencia (sólo si existe más de un lote), en el caso de Búsqueda de Producto se utilizará la Búsqueda con Lotes.  - Una vez capturado el producto y lote, si está activo el nuevo parámetro para manejo de inventario a nivel de lote, se deberá validar existencia en inventario para el lote seleccionado, así como descontar el producto del inventario y lote correspondientes en caso de haber suficiente existencia.  \*\*NOTA: Se considera manejar un detalle del movimiento por cada producto, unidad y lote capturados.  (Deliprime, 1.18.0.0 - 4.23.0.0) | Belem Lizeth Jiménez Arévalo |
|  |  |  |  |

Se deberán realizar las siguientes modificaciones en la captura de producto para el Pedido dentro del módulo de **AUTOVENTA** <ModuloTerm.TipoIndice = 1>.

1. Esta nueva funcionalidad aplicará dentro del pedido ÚNICAMENTE si existe una configuración de parámetros Activa para Captura de Caducidad y Lote <ConfigParametro.Parametro = ‘CapturaCadLote’> donde <ConfigParametro.Valor = 1> y si existe una configuración de parámetros Activa para Manejo de Inventario por Lotes <ConfigParametro.Parametro = ‘InventarioPorLotes’> donde <ConfigParametro.Valor = 1>.
   1. 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. Cantidad
            6. Lote
            7. Caducidad
            8. Precio
            9. Subtotal
            10. Impuesto
            11. Total
      2. El caso de uso incluye la funcionalidad del caso de uso **Administrar Inventario por Lotes – CUROLGENXX**
         1. El sistema envía como parámetro la siguiente información:
            1. ObtenerExistencia

ProductoClave = ProductoClave (objeto Productos)

TipoUnidad = Unidad (objeto Productos)

Lote = Lote (objeto Productos)

* + - 1. El sistema recibe la siguiente información:
         1. OP\_Existencia
    1. El sistema actualiza la siguiente información en el objeto Productos para el producto actual:
       1. Productos
          1. Existencia = OP\_Existencia (formatear la cantidad de acuerdo al número de decimales configurado para el producto <Producto.DecimalProducto>).
    2. El sistema presenta la siguiente información:
       1. 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>.
       2. 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.
       3. 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”
       4. Lote: Se presenta el lote al que pertenece el producto <Productos.Lote>.
       5. Caducidad: Se presenta la fecha de caducidad del lote al que pertenece el producto <Productos.Caducidad>, en formato dd/mm/aa.
       6. Existencia: Se presenta la existencia restante del producto, unidad y lote en el inventario <Productos.Existencia>.
       7. Precio: Se presenta el precio unitario del producto <Productos.Precio>, precedido del signo “$”.
       8. Subtotal: Se presenta el subtotal del producto <Productos.Subtotal>, precedido del signo “$”.
  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

* + 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. 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 = Cadena registrada en sesión

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

Lote

Caducidad

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

**Producto**

ProductoClave

Nombre

DecimalProducto

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

Productos

Descripcion = <Producto.Nombre>

DecimalProducto

* + - * 1. El caso de uso incluye la funcionalidad del caso de uso **Administrar Inventario por Lotes – CUROLGENXX**

El sistema envía como parámetro la siguiente información:

ActualizarInventario

ProductoClave = ProductoClave (objeto Productos)

TipoUnidad = Unidad (objeto Productos)

Lote = Lote (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:

GenerarDetallePedido

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. El sistema registra la siguiente información:

**TpdDatosExtra**

TransProdID = <TransProdDetalle.TransProdId>

TransProdDetalleID = <TransProdDetalle.TransProdDetalleID>

Lote = Lote (objeto Productos)

MFechaHora = Fecha y hora actual del sistema

MUsuarioID = UsuarioId registrado en sesión

* + - 1. El sistema continúa en el [**paso 1.1**](#PasoParaCadaProductoObjetoProductos)del flujo actual
    1. Si <ProductoClave coincide> o Si <ID coincide>
       1. El sistema continúa en el [**paso 1**](#paso1_CapturarLotes) del flujo **[Búsqueda con Lotes](#CapturarLotesAEntregar)** y le envía como parámetro la siguiente información:
          1. ProductoClave
    2. Si <CodigoBarras coincide>
       1. El sistema continúa en el [**paso 1**](#paso1_CapturarLotes) del flujo [**Búsqueda con Lotes**](#CapturarLotesAEntregar)y le envía como parámetro la siguiente información:
          1. ProductoClave
          2. TipoUnidad
    3. 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 elcampo correspondiente a la captura del Producto
    4. 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 elcampo correspondiente a la captura del Producto
  1. Si <el actor selecciona un producto>
     1. El sistema continúa en el [**paso 1**](#paso1_CapturarLotes) del flujo [**Búsqueda con Lotes**](#CapturarLotesAEntregar)y le envía como parámetro la siguiente información:
        1. ProductoClave
        2. TipoUnidad
  2. 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. DecimalProducto
        4. Unidad
        5. Cantidad
        6. Lote
        7. Caducidad
        8. Precio
        9. Subtotal
        10. Impuesto
        11. 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**>

El sistema continúa en el paso donde fue invocado

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

El caso de uso incluye la funcionalidad del caso de uso **Administrar Inventario por Lotes – CUROLGENXX y le** envía como parámetro la siguiente información:

ActualizarInventario

ProductoClave = ProductoClave (objeto Productos)

TipoUnidad = Unidad (objeto Productos)

Lote = Lote (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

El sistema elimina la siguiente información:

**TpdDatosExtra**

TransProdID = <TransProdDetalle.TransProdId>

TransProdDetalleID = <TransProdDetalle.TransProdDetalleID>

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

El sistema elimina del arreglo la información del objeto correspondiente al producto:

**ArregloTransProdDetalle**

TransProdDetalle

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

El sistema elimina del objeto Productos la siguiente información del producto:

Productos

ProductoClave

Descripcion

DecimalProducto

Unidad

Cantidad

Lote

Caducidad

Precio

Subtotal

Impuesto

Total

* + - * 1. El sistema continúa en el [**paso 1.1**](#PasoParaCadaProductoObjetoProductos) del flujo actual
  1. Si <el actor selecciona la opción **Buscar**>
     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

Clave del Producto

Unidad

Cantidad

Precio

Lote

Caducidad

* + 1. El sistema continúa en el [**paso 1.2.2.3**](#PasoParaCadaProductoRegresadoBusqueda) del flujo actual
  1. Si <el actor selecciona la opción **Continuar**>
     1. El sistema continúa con el flujo habitual del pedido en Autoventa (continuar la aplicación de promociones en caso de aplicar, con el Tab de Totales del Pedido y la impresión del ticket)

# [Búsqueda con Lotes](#CapturarLotesAEntregar_r)

1. El sistema obtiene la siguiente información de los parámetros de entrada:
   1. ProductoClave
   2. TipoUnidad (en caso de aplicar)
2. El sistema obtiene la siguiente información:
   1. **InventarioLotes**
      1. ProductoClave = ProductoClave recibido como parámetro
      2. TipoUnidad = TipoUnidad recibida como parámetro (en caso de aplicar)
      3. Lote
      4. Disponible
      5. NoDisponible
      6. Apartado
      7. Contenido
      8. **Existencia**: Se deberá obtener la existencia del producto en inventario a partir del siguiente cálculo: *Existencia* = Disponible – Apartado – No Disponible – Contenido.
3. Si <el producto no ha sido capturado, es decir, si no existe un registro donde <TransProdDetalle.ProductoClave = Producto actual>>
   1. El sistema presenta la pantalla de Buscar Producto con la siguiente información:
      1. Producto: Permite capturar una cadena de búsqueda.
      2. Para cada lote obtenido relacionado al Producto actual que cuente con existencia en inventario (*Existencia > 0*)
         1. Unidad: Se presenta la descripción de la unidad de venta del producto y lote actuales, para ello ir a <VAVDescripcion> y obtener <VAVDescripcion.Descripcion> donde <VAVDescripcion.VARCodigo = ‘UNIDADV’> y <VAVDescripcion.VAVClave = InventarioLotes.TipoUnidad>, de acuerdo al tipo de lenguaje configurado para el sistema.
         2. Producto: Se presenta la concatenación de la clave del producto <Productos.ProductoClave> más su nombre <Productos.Descripcion>.
         3. Lote: Se presenta el lote del producto <InventarioLotes.Lote>.
         4. Caducidad: Se presenta la caducidad del producto <LoteCaducidad.Caducidad> donde <LoteCaducidad.Lote = InventarioLotes.Lote>.
         5. Existencia: Se presenta la existencia del lote actual en el inventario a partir del siguiente cálculo: *Existencia* = Disponible – Apartado – No Disponible – Contenido.
         6. Precio: Se presenta el precio vigente para el producto y unidad actual <PrecioProductoVig.Precio>.
         7. El sistema presenta y solicita la siguiente información:
            1. Cantidad: Presentar por defecto la cantidad igual a cero. Formatear la cantidad a mostrar de acuerdo al número de decimales configurado para el producto <Productos.DecimalProducto>.
      3. Si <el actor proporciona una **cantidad mayor a Cero** para un lote y unidad> y si <el actor abandona el campo o selecciona la opción **Enter**>
         1. Si <la *Existencia* del producto en Inventario es menor a la cantidad proporcionada por el actor>
            1. El sistema presenta la siguiente información:

Cantidad: Presentar como cantidad la *Existencia* del lote en el inventario. Formatear la cantidad a mostrar de acuerdo al número de decimales configurado para el producto <Productos.DecimalProducto>.

1. Si <el producto ya fue capturado, es decir, si existe por lo menos un registro donde <TransProdDetalle.ProductoClave = Producto actual>>
   1. Para cada registro del producto actual incluido en el objeto Productos
      1. El sistema obtiene la siguiente información del objeto Productos:
         1. Productos
            1. ProductoClave = Producto actual
            2. Unidad
            3. Cantidad
            4. Lote
            5. Caducidad
            6. Precio
            7. Subtotal
            8. Impuesto
            9. Total
   2. El sistema presenta la pantalla de Buscar Producto con la siguiente información:
      1. Producto: Permite capturar una cadena de búsqueda.
      2. Para cada registro del producto actual incluido en el objeto Productos
         1. Unidad: Se presenta la descripción de la unidad de venta del producto y lote actuales, 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.
         2. Producto: Se presenta la concatenación de la clave del producto <Productos.ProductoClave> más su nombre <Productos.Descripcion>.
         3. Lote: Se presenta el lote del producto <Productos.Lote>.
         4. Caducidad: Se presenta la caducidad del producto <Productos.Caducidad>.
         5. Existencia: Se presenta la existencia del lote actual en el inventario a partir del siguiente cálculo: *Existencia* = Disponible – Apartado – No Disponible – Contenido.
         6. Precio: Se presenta el precio vigente para el producto y unidad actual <PrecioProductoVig.Precio>.
         7. El sistema presenta y solicita la siguiente información:
            1. Cantidad: Se presenta la cantidad de producto capturada previamente para el lote <Productos.Cantidad>.
      3. Para cada lote obtenido relacionado al Producto actual que cuente con existencia en inventario (*Existencia > 0*) y que no se encuentre dentro del objeto Productos
         1. Unidad: Se presenta la descripción de la unidad de venta del producto y lote actuales, para ello ir a <VAVDescripcion> y obtener <VAVDescripcion.Descripcion> donde <VAVDescripcion.VARCodigo = ‘UNIDADV’> y <VAVDescripcion.VAVClave = InventarioLotes.TipoUnidad>, de acuerdo al tipo de lenguaje configurado para el sistema.
         2. Producto: Se presenta la concatenación de la clave del producto <Productos.ProductoClave> más su nombre <Productos.Descripcion>.
         3. Lote: Se presenta el lote del producto <InventarioLotes.Lote>.
         4. Caducidad: Se presenta la caducidad del producto <LoteCaducidad.Caducidad> donde <LoteCaducidad.Lote = InventarioLotes.Lote>.
         5. Existencia: Se presenta la existencia del lote actual en el inventario a partir del siguiente cálculo: *Existencia* = Disponible – Apartado – No Disponible – Contenido.
         6. Precio: Se presenta el precio vigente para el producto y unidad actual <PrecioProductoVig.Precio>.
         7. El sistema presenta y solicita la siguiente información:
            1. Cantidad: Presentar por defecto la cantidad igual a cero. Formatear la cantidad a mostrar de acuerdo al número de decimales configurado para el producto <Productos.DecimalProducto>.
   3. Si <el actor proporciona una **cantidad mayor a Cero** para un lote y unidad que no se encuentren dentro del objeto Productos> y si <el actor abandona el campo o selecciona la opción **Enter**>
      1. El sistema continúa en el [**paso 3.1.3.1**](#PasoAjustarExistencia) del flujo actual
   4. Si <el actor **modifica** la cantidad de un lote y unidad que ya se encuentren dentro del objeto Productos> y si <el actor abandona el campo o selecciona la opción **Enter**>
      1. El sistema actualiza el objeto Productos con la siguiente información para el producto actual:
         1. Productos
            1. Cantidad (formatear la cantidad mostrada de acuerdo al número de decimales configurado para el producto <Producto.DecimalProducto>)
      2. El caso de uso incluye la funcionalidad del caso de uso **Administrar Inventario por Lotes – CUROLGENXX**
         1. El sistema envía como parámetro la siguiente información:
            1. ValidarExistencia

ProductoClave = ProductoClave (objeto Productos)

TipoUnidad = Unidad (objeto Productos)

Lote = Lote (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 4.4**](#PasoModificarCantidadLote) del flujo actual

1. Si <el actor selecciona la opción **Agregar**>
   1. Para cada lote y unidad con Cantidad capturada mayor a Cero QUE NO SE ENCUENTRE DENTRO DEL OBJETO PRODUCTOS
      1. El sistema registra en el objeto Productos la siguiente información del producto actual:
         1. Productos
            1. ProductoClave = <Producto.ProductoClave>
            2. Descripcion = <Producto.Nombre>
            3. DecimalProducto = <Producto.DecimalProducto>
            4. Unidad = <InventarioLotes.TipoUnidad>
            5. Cantidad = Cantidad (formatear la cantidad de acuerdo al número de decimales configurado para el producto <Producto.DecimalProducto>)
            6. Lote = <InventarioLotes.Lote>
            7. Caducidad = <LoteCaducidad.Caducidad> donde <LoteCaducidad.Lote = InventarioLotes.Lote>
      2. El caso de uso incluye la funcionalidad del caso de uso **Administrar Inventario por Lotes – CUROLGENXX**
         1. El sistema envía como parámetro la siguiente información:
            1. ActualizarInventario

ProductoClave = ProductoClave (objeto Productos)

TipoUnidad = Unidad (objeto Productos)

Lote = Lote (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. GenerarDetallePedido

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. 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**

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:
          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 registra la siguiente información:
       1. **TpdDatosExtra**
          1. TransProdID = <TransProdDetalle.TransProdId>
          2. TransProdDetalleID = <TransProdDetalle.TransProdDetalleID>
          3. Lote = Lote (objeto Productos)
          4. MFechaHora = Fecha y hora actual del sistema
          5. MUsuarioID = UsuarioId registrado en sesión
  1. Para cada lote y unidad cuya Cantidad haya sido modificada, es decir, si <TransProdDetalle.Cantidad <> Productos.Cantidad>
     1. El caso de uso incluye la funcionalidad del caso de uso **Administrar Inventario por Lotes – CUROLGENXX**
        1. El sistema envía como parámetro la siguiente información:
           1. ActualizarInventario

ProductoClave = <TransProdDetalle.ProductoClave>

TipoUnidad = <TransProdDetalle.TipoUnidad>

Lote = Lote (objeto Productos)

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. El caso de uso incluye la funcionalidad del caso de uso **Administrar Inventario por Lotes – CUROLGENXX**
       1. El sistema envía como parámetro la siguiente información:
          1. ActualizarInventario

ProductoClave = ProductoClave (objeto Productos)

TipoUnidad = Unidad (objeto Productos)

Lote = Lote (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 sistema elimina la siguiente información:
       1. **TpdDatosExtra**
          1. TransProdID = <TransProdDetalle.TransProdId>
          2. TransProdDetalleID = <TransProdDetalle.TransProdDetalleID>
    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. 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
      2. El sistema elimina del arreglo la información del objeto correspondiente al detalle del producto:
         1. **ArregloTransProdDetalle**

TransProdDetalle

* + 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. GenerarDetallePedido

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. 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**

TransProdDetalle

Subtotal

Impuesto

Total

* + - 1. El sistema actualiza el objeto Productos con la siguiente información para el producto actual:
         1. 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)
       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 registra la siguiente información:
       1. **TpdDatosExtra**
          1. TransProdID = <TransProdDetalle.TransProdId>
          2. TransProdDetalleID = <TransProdDetalle.TransProdDetalleID>
          3. Lote = Lote (objeto Productos)
          4. MFechaHora = Fecha y hora actual del sistema
          5. MUsuarioID = UsuarioId registrado en sesión
  1. El sistema continúa en el [**paso 1.1**](#PasoParaCadaProductoObjetoProductos) del flujo principal

1. Si <el actor selecciona la opción **Regresar**>
   1. Para cada lote y unidad cuya Cantidad haya sido modificada, es decir, si <TransProdDetalle.Cantidad <> Productos.Cantidad>
      * 1. El sistema actualiza el objeto Productos con la siguiente información para el producto actual:
           1. Productos

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

* + 1. El sistema continúa en el paso donde fue invocado

1. Finaliza el escenario actual