**Proyecto Route Lite (ROL)**

Especificación del Caso de Uso: Reporte de Inventario – CUROLMOV68

VERSIÓN 1.0

**HISTÓRICO DE REVISIÓN**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión** | **Descripción** | **Autor** |
| 05/04/2016 | 1.0 | Elaboración de la especificación del caso de uso Reporte de Inventario para el Proyecto Route Lite de acuerdo al Estándar de Casos de Uso 1.3.  Folio CAI 0003869  (Route Lite, 1.6.0.0) | Belem Lizeth Jiménez Arévalo |
| 05/12/2018 | 1.1 | Folio CAI 0005146  Se agrega la validación correspondiente en base al parámetro "InvImpresionInterlineado" cuyo valor indicara si se imprime el reporte normal de inventario (con valor en 0) o se imprime el nuevo reporte de inventario con interlineado y saltos de línea (con valor != 0).  Se realiza en la versión 1.16.0.0 – 4.21.0.0 y se replica a partir de 1.17.0.0 4.22.0.0.  Requerimiento de PDR. | Erik Alejandro Amador Serrano |
| 14/12/2018 | 1.2 | Folio CAI 0004275  Se agrega la columna “Precio” en reporte general y detallado.  Se realiza en la versión 1.17.0.0 - 4.22.0.0  Se agrega nota en el Folio CAI 0005146 | Erik Alejandro Amador Serrano |
| 23/12/2019 | 1.3 | Folio CAI 0005766  Modificar el Reporte de Inventario Genérico del móvil (REPORTEA = 15), para que ordene los productos en base a la configuración del parámetro “OrdeTickeInveGene”.  **Se realiza en la versión 4.23.1.1 con réplica a partir de la 4.24.0.0**  (PDR, 4.23.1.1) | Erik Alejandro Amador Serrano |

Tabla de Contenido

[1 Introducción 4](#_Toc29393630)

[2 Caso de uso: Reporte de Inventario – CUROLMOV68 4](#_Toc29393631)

[2.1 Descripción 4](#_Toc29393632)

[3 Diagrama de Casos de Uso 4](#_Toc29393633)

[4 Precondiciones 4](#_Toc29393634)

[4.1 Actores 4](#_Toc29393635)

[4.2 Generales 4](#_Toc29393636)

[4.3 Parámetros 5](#_Toc29393637)

[5 Flujo de eventos 5](#_Toc29393638)

[5.1 Flujo básico 5](#_Toc29393639)

[5.2 Flujos alternos 11](#_Toc29393640)

[5.2.1 Opcionales 11](#_Toc29393641)

[5.2.2 Generales 11](#_Toc29393642)

[6 Poscondiciones 11](#_Toc29393643)

[10 Firmas de Aceptación 11](#_Toc29393644)

Especificación del Caso de Uso: Reporte de Inventario – CUROLMOV68

# 1 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.

# 2 Caso de uso: Reporte de Inventario – CUROLMOV68

## 2.1 Descripción

Permite al actor consultar y/o imprimir el detalle del Inventario mediante la relación de los productos y sus saldos en cantidad disponibles a bordo.

# 3 Diagrama de Casos de Uso



# 4 Precondiciones

## 4.1 Actores

1. Vendedor

## 4.2 Generales

1. El actor debe de tener un usuario registrado en el sistema.
2. El actor debe de estar activo.
3. Se debe haber seleccionado un reporte en el sistema.
4. Se debe tener configurada por lo menos una impresora.
5. Se debe haber ingresado a la actividad de Reportes.
6. El reporte debe estar dado de alta en el catálogo de valores por referencia para los reportes del móvil en Android (REPORTEA).

## 4.3 Parámetros

1. ReporteA
2. Detallado
3. General
4. Totalizar

# 5 Flujo de eventos

## 5.1 Flujo básico

1. El caso de uso inicia cuando es invocado por el caso de uso **Generar Reportes – CUROLMOV47**
2. El sistema obtiene el nombre del reporte seleccionado:
   1. **VAVDescripcion**
      1. VARCodigo = ‘REPORTEA’
      2. VAVClave = ReporteA recibido como parámetro
      3. Descripcion
3. El sistema obtiene la siguiente información del vendedor:
   1. **Vendedor**
      1. VendedoId
      2. Nombre
4. El sistema obtiene la siguiente información de las SubEmpresas existentes:
   1. **SubEmpresa**
      1. SubEmpresaId
      2. NombreEmpresa
      3. RFC
      4. Calle
      5. Numero
      6. NumeroInterior
      7. Colonia
      8. Ciudad
      9. Region
5. El sistema obtiene la siguiente información de los productos que se encuentran en inventario:
   1. **Inventario**
      1. ProductoClave
         1. **Producto**
            1. ProductoClave
            2. Nombre
            3. id
            4. SubEmpresaId
         2. **ProductoUnidad**
            1. ProductoClave
            2. PRUTipoUnidad
            3. **ProductoDetalle**

ProductoClave

PRUTipoUnidad

Factor

* + 1. Disponible
    2. NoDisponible
    3. Apartado
    4. Contenido
    5. Pedido

1. Para cada SubEmpresa obtenida en el [**paso 4**](#paso4):
   1. El sistema presenta la siguiente información de la SubEmpresa:
      1. Nombre Empresa: <SubEmpresa.NombreEmpresa>
      2. RFC: <SubEmpresa.RFC>
      3. Domicilio: <SubEmpresa.Calle> + ‘ ‘ + <SubEmpresa.Numero> + ‘-‘ + <SubEmpresa.NumeroInterior> + ‘,’ +

“Col. ” + <SubEmpresa.Colonia> + ‘,’ + <SubEmpresa.Ciudad> + ’,’ + <SubEmpresa.Region>.

* 1. El sistema presenta la siguiente información en el encabezado:
     1. **Impresión:** Se presenta la fecha y hora de impresión del reporte en formato dd/MM/aaaa hh:mm:ss.
     2. **Vendedor:** Se presenta el nombre del vendedor <Vendedor.Nombre>.
     3. **Reporte:** Se presenta el nombre del reporte. Si se seleccionó el filtro de Detallado (Detallado = 1) de acuerdo a la información recibida como parámetro, se presenta el nombre obtenido para el reporte y debajo de éste la leyenda “Detallado”. Si se seleccionó el filtro de General (General = 1) de acuerdo a la información recibida como parámetro, se presenta el nombre obtenido para el reporte y debajo de éste la leyenda “General”.
  2. El sistema presenta las siguientes columnas:
     1. Producto
     2. Exis
     3. Disp
     4. No D
  3. Si <se seleccionó la vista Detallada para el reporte (Detallado = 1) de acuerdo a la información recibida como parámetro>
     1. Para cada producto en inventario asociado a la SubEmpresa actual obtenido en el [**paso 5**](#paso5), donde <Inventario.ProductoClave = Producto.ProductoClave> y <Producto.SubEmpresaId = SubEmpresa.SubEmpresaId actual>:
        1. El sistema presenta la siguiente información ordenándola de manera ascendente, **Si** existe una configuración de parámetros para el ordenamiento de información, es decir, si <ConfigParametro.Parametro = “OrdenTicketInvGen”> entonces se ordena de izquierda a derecha por el o los campos obtenidos en <ConfigParametro.Valor> (si es más de un campo estará separado por comas).
           1. **Producto:** Se presenta la clave del producto <Producto.ProductoClave> seguida del nombre de éste <Producto.Nombre>, separados por un guion medio, ejemplo: “Clave – NombreProducto”.
           2. **Exis:** Se presenta la existencia a bordo del producto, corresponde al inventario total a bordo, ingresado a través de cargas de producto, para ello obtener la información correspondiente al inventario disponible <Inventario.Disponible>.*(\*\*Utilizar dos decimales).*
           3. **Disp:** Se presenta la existencia disponible para venta del producto, ésta corresponde al inventario total a bordo menos lo generado por una devolución o un cambio de producto en mal estado, menos el apartado del producto por una venta, menos el producto contenido (envase) asociado al producto (en caso de existir). Para ello obtener la información correspondiente al inventario de producto disponible, menos el no disponible, menos el apartado, menos el contenido: <Inventario.Disponible – Inventario.NoDisponible – Inventario.Apartado – Inventario.Contenido>. *(\*\*Utilizar dos decimales).*
           4. **No D:** Se presenta la existencia no disponible del producto, ésta corresponde a lo generado por una devolución o un cambio de producto en mal estado, más el producto apartado (producto pedido para reparto en caso de aplicar). Para ello obtener la información correspondiente al inventario de producto no disponible, más el apartado: <Inventario.NoDisponible + Inventario.Apartado>. *(\*\*Utilizar dos decimales).*
           5. Para cada unidad de venta obtenida para el producto <ProductoUnidad.PRUTipoUnidad> el sistema presenta la siguiente información, ordenándola de manera descendente por su factor de conversión <ProductoDetalle.Factor>, donde <ProductoDetalle.PRUTipoUnidad = ProductoUnidad.PRUTipoUnidad>:

**Unidad:** Se presenta la descripción de la unidad de venta obtenida para el producto, con el valor de <ProductoUnidad.PRUTipoUnidad> ir a <VAVDescripcion> y obtener <VAVDescripcion.Descripcion> donde <VAVDescripcion.VARCodigo = ‘UNIDADV’> y <VAVDescripcion.VAVClave = ProductoUnidad.PRUTipoUnidad> de acuerdo al tipo de lenguaje configurado para el sistema.

**Exis:** Se presenta la existencia a bordo del producto en términos de la unidad actual. Para ello:

Si <existe una unidad de venta para el producto cuyo factor sea mayor a la unidad actual>: Obtener de sesión el valor de la variable ExisRestante y dividirla entre el factor de la unidad actual: ExisRestante / <ProductoDetalle.Factor>, obtener la parte entera del resultado y presentarla como *Existencia*. Enseguida, actualizar en sesión la siguiente información: ExisRestante = *ExisRestante* - <ProductoDetalle.Factor \* *Existencia*>. *(\*\*Utilizar dos decimales).*

Si <NO existe una unidad de venta para el producto cuyo factor sea mayor a la unidad actual>: Obtener la información correspondiente al inventario disponible del producto y dividirla entre el factor de conversión de la unidad actual: <Inventario.Disponible / ProductoDetalle.Factor>, obtener la parte entera del resultado y presentarla como *Existencia*. Enseguida, registrar en sesión la siguiente información: ExisRestante = <Inventario.Disponible> - <ProductoDetalle.Factor \* *Existencia*>. *(\*\*Utilizar dos decimales).*

**Disp:** Se presenta la existencia disponible para venta del producto, ésta corresponde al inventario total a bordo menos lo generado por una devolución o un cambio de producto en mal estado, menos el apartado del producto por una venta, menos el producto contenido (envase) asociado al producto (en caso de existir). Para ello:

Si <existe una unidad de venta para el producto cuyo factor sea mayor a la unidad actual>: Obtener de sesión el valor de la variable DispRestante y dividirla entre el factor de la unidad actual: DispRestante / <ProductoDetalle.Factor>, obtener la parte entera del resultado y presentarla como *Disponible*. Enseguida, actualizar en sesión la siguiente información: DispRestante = *DispRestante* - <ProductoDetalle.Factor \* *Disponible*>. *(\*\*Utilizar dos decimales).*

Si <NO existe una unidad de venta para el producto cuyo factor sea mayor a la unidad actual>: Obtener la información correspondiente al inventario de producto disponible, menos el no disponible, menos el apartado, menos el contenido, y dividirla entre el factor de conversión de la unidad actual: (<Inventario.Disponible – Inventario.NoDisponible – Inventario.Apartado – Inventario.Contenido>) / <ProductoDetalle.Factor>, obtener la parte entera del resultado y presentarla como *Disponible*. Enseguida, registrar en sesión la siguiente información: DispRestante = (<Inventario.Disponible – Inventario.NoDisponible – Inventario.Apartado – Inventario.Contenido>) - <ProductoDetalle.Factor \* *Disponible*>. *(\*\*Utilizar dos decimales).*

**No D:** Se presenta la existencia no disponible del producto, ésta corresponde a lo generado por una devolución o un cambio de producto en mal estado, más el producto apartado (producto pedido para reparto en caso de aplicar). Para ello:

Si <existe una unidad de venta para el producto cuyo factor sea mayor a la unidad actual>: Obtener de sesión el valor de la variable NoDRestante y dividirla entre el factor de la unidad actual: NoDRestante / <ProductoDetalle.Factor>, obtener la parte entera del resultado y presentarla como *NoDisponible*. Enseguida, actualizar en sesión la siguiente información: NoDRestante = *NoDRestante* - <ProductoDetalle.Factor \* *NoDisponible*>. *(\*\*Utilizar dos decimales).*

Si <NO existe una unidad de venta para el producto cuyo factor sea mayor a la unidad actual>: obtener la información correspondiente al inventario de producto no disponible, más el apartado, y dividirla entre el factor de conversión de la unidad actual: (<Inventario.NoDisponible + Inventario.Apartado>) / <ProductoDetalle.Factor>, obtener la parte entera del resultado y presentarla como *NoDisponible*. Enseguida, registrar en sesión la siguiente información: NoDRestante = (<Inventario.NoDisponible + Inventario.Apartado>) - <ProductoDetalle.Factor \* *NoDisponible*>. *(\*\*Utilizar dos decimales).*

* + - * 1. **Precio:** Se presenta el precio del producto actual, si <Vendedor.ListaPrecioBase <> null o vacío> entonces presentar <PrecioProductoVig.Precio> donde <Vendedor.VendedorId=id del vendedor actual> y <PrecioProductoVig.PrecioClave = Vendedor.ListaPrecioBase> y vigente al día actual, entre (<PrecioProductoVig.PPVFechaInicio> y <PrecioProductoVig.FechaFin>)), si no presentar un 0.
    1. Si <se seleccionó la Totalización de Productos (Totalizar = 1) de acuerdo a la información recibida como parámetro>
       1. El sistema presenta la siguiente información:
          1. **Total de Productos:** Se presenta como total la sumatoria de cada renglón de producto para las siguientes columnas: Existencia, Disponible y No Disponible.
          2. **Total Unitario:** Se presenta el importe total de todos los productos en su unidad mínima, se consideraran sólo las listas de precios que se llevan a bordo y se tomará el precio más alto, en caso de que el producto no esté contemplado en la listas de precios, este llevará el valor de cero, para ello: <ProductoDetalle.ProductoClave = Inventario.ProductoClave> y <ProductoDetalle.ProductoDetClave = Inventario.ProductoClave>, tomar el registro que tenga el factor menor <ProductoDetalle.Factor>. Con el resultado buscar el precio vigente para el producto <PrecioProductoVig.Precio> donde <PrecioProductoVig.ProductoClave = ProductoDetalle.ProductoClave> y <PrecioProductoVig.PRUTipoUnidad = ProductoDetalle.PRUTipoUnidad>, si se obtiene más de un registro tomar el que tenga el precio más alto y multiplicarlo por la existencia del producto en su unidad mínima: ∑<PrecioProductoVig.Precio> \* (<Inventario.Disponible \* ProductoDetalle.Factor>). En caso de no encontrar ningún registro, presentar el precio en ceros. *(\*\*Utilizar dos decimales).*
          3. **Total Unidad:** Se presenta el importe total de todos los productos en sus cantidades obtenidas para cada una de las unidades de venta calculadas en el [**paso 6.4.1.1.5**](#paso6_4_1_1_5), se consideraran sólo las listas de precios que se llevan a bordo y se tomará el precio más alto, en caso de que el producto no esté contemplado en la listas de precios, este llevará el valor de cero, para ello: buscar el precio vigente para el producto <PrecioProductoVig.Precio> donde <PrecioProductoVig.ProductoClave = ProductoDetalle.ProductoClave> y <PrecioProductoVig.PRUTipoUnidad = TipoUnidad (esto para cada una de las unidades de venta obtenidas y calculadas para el producto), si se obtiene más de un registro tomar el que tenga el precio más alto y multiplicarlo por la cantidad de producto calculada para ese rubro: ∑<PrecioProductoVig.Precio \* CantidadCalculada>. En caso de no encontrar ningún registro, presentar el precio en ceros. *(\*\*Utilizar dos decimales).*
  1. Si <se seleccionó la vista General para el reporte (General = 1) de acuerdo a la información recibida como parámetro>
     1. Para cada producto en inventario asociado a la SubEmpresa actual obtenido en el [**paso 5**](#paso5), donde <Inventario.ProductoClave = Producto.ProductoClave> y <Producto.SubEmpresaId = SubEmpresa.SubEmpresaId actual>:
        1. El sistema presenta la siguiente información ordenándola de manera ascendente, **Si** existe una configuración de parámetros para el ordenamiento de información, es decir, si <ConfigParametro.Parametro = “OrdenTicketInvGen”> entonces se ordena de izquierda a derecha por el o los campos obtenidos en <ConfigParametro.Valor> (si es más de un campo estará separado por comas).
           1. **Producto:** Se presenta la clave del producto <Producto.ProductoClave> seguida del nombre de éste <Producto.Nombre>, separados por un guion medio, ejemplo: “Clave – NombreProducto”.
           2. **Exis:** Se presenta la existencia a bordo del producto, corresponde al inventario total a bordo, ingresado a través de cargas de producto, para ello obtener la información correspondiente al inventario disponible <Inventario.Disponible>.*(\*\*Utilizar dos decimales).*
           3. **Disp:** Se presenta la existencia disponible para venta del producto, ésta corresponde al inventario total a bordo menos lo generado por una devolución o un cambio de producto en mal estado, menos el apartado del producto por una venta, menos el producto contenido (envase) asociado al producto (en caso de existir). Para ello obtener la información correspondiente al inventario de producto disponible, menos el no disponible, menos el apartado, menos el contenido: <Inventario.Disponible – Inventario.NoDisponible – Inventario.Apartado – Inventario.Contenido>. *(\*\*Utilizar dos decimales).*
           4. **No D:** Se presenta la existencia no disponible del producto, ésta corresponde a lo generado por una devolución o un cambio de producto en mal estado, más el producto apartado (producto pedido para reparto en caso de aplicar). Para ello obtener la información correspondiente al inventario de producto no disponible, más el apartado: <Inventario.NoDisponible + Inventario.Apartado>. *(\*\*Utilizar dos decimales).*
           5. **Precio:** Se presenta el precio del producto actual, si <Vendedor.ListaPrecioBase <> null o vacío> entonces presentar <PrecioProductoVig.Precio> donde <Vendedor.VendedorId=id del vendedor actual> y <PrecioProductoVig.PrecioClave = Vendedor.ListaPrecioBase> y vigente al día actual, entre (<PrecioProductoVig.PPVFechaInicio> y <PrecioProductoVig.FechaFin>)), si no presentar un 0.
     2. Si <se seleccionó la Totalización de Productos (Totalizar = 1) de acuerdo a la información recibida como parámetro>
        1. El sistema presenta la siguiente información:
           1. **Total de Productos:** Se presenta como total la sumatoria de cada renglón de producto para las siguientes columnas: Existencia, Disponible y No Disponible.
           2. **Total Unitario:** Se presenta el importe total de todos los productos en su unidad mínima, se consideraran sólo las listas de precios que se llevan a bordo y se tomará el precio más alto, en caso de que el producto no esté contemplado en la listas de precios, este llevará el valor de cero: <ProductoDetalle.ProductoClave = Inventario.ProductoClave> y <ProductoDetalle.ProductoDetClave = Inventario.ProductoClave>, tomar el registro que tenga el factor menor <ProductoDetalle.Factor>. Con el resultado buscar el precio vigente para el producto <PrecioProductoVig.Precio> donde <PrecioProductoVig.ProductoClave = ProductoDetalle.ProductoClave> y <PrecioProductoVig.PRUTipoUnidad = ProductoDetalle.PRUTipoUnidad>, si se obtiene más de un registro tomar el que tenga el precio más alto y multiplicarlo por la existencia del producto en su unidad mínima: <PrecioProductoVig.Precio> \* (<Inventario.Disponible \* ProductoDetalle.Factor>). En caso de no encontrar ningún registro, presentar el precio en ceros. *(\*\*Utilizar dos decimales).*

1. Si <el actor selecciona la opción **Imprimir**>
   1. Si <ConfigParametro.Parametro = “InvImpresionInterlineado” and ConfigParametro.Valor = 0 >.
      1. El sistema envía al dispositivo de impresión el ticket del reporte mostrado, incluyendo la funcionalidad del caso de uso **Imprimir Recibos – RNGENXX** y enviando como parámetros la siguiente información:
         1. ListaDocumentos
            1. ValorReferencia = 1
            2. LogoSoloPrimerRecibo = 1
   2. Si < ConfigParametro.Parametro = “InvImpresionInterlineado” and ConfigParametro.Valor <> 0 >.
      1. El sistema envía al dispositivo de impresión el ticket del reporte mostrado, incluyendo interlineado y saltos de línea, así mismo la funcionalidad del caso de uso **Imprimir Recibos – RNGENXX** y enviando como parámetros la siguiente información:
         1. ListaDocumentos
            1. ValorReferencia = 1
            2. LogoSoloPrimerRecibo = 1
2. Si <el actor selecciona la opción **Regresar**>
   1. El sistema continúa en el [**paso 9**](#Fin) del flujo básico
3. Finaliza el caso de uso

## 5.2 Flujos alternos

### 5.2.1 Opcionales

N/A

### 5.2.2 Generales

N/A

# 6 Poscondiciones

N/A

# 

# 10 Firmas de Aceptación

|  |
| --- |
|  |
| José María Alcalá |
| Gerente de Ingeniería / Departamento de Ingeniería – Duxstar Solutions |
| 08/04/2016 |

|  |
| --- |
|  |
| Jesús Peña |
| Líder de Proyecto / Departamento de Ingeniería – Duxstar Solutions |
| 08/04/2016 |

|  |
| --- |
|  |
| Belem Lizeth Jiménez Arévalo |
| Analista de Sistemas / Departamento de Análisis y Diseño – Duxstar Solutions |
| 08/04/2016 |