

GET Technologies	INC - 20241115 - Diferente ticket, mismo folio de venta		
Autor	Versión	Fecha de elaboración	Fecha de última actualización
Juan Pérez	1.0	15-11-2024	15-11-2024

Control de versiones

Versión	Fecha	Autor	Descripción
1.0	15-11-2024	Juan Pérez	Se analiza el problema, se encuentra causa raíz y se propone una solución.

	INC - 20241115 - Diferente ticket, mismo folio de venta		
Autor	Versión	Fecha de elaboración	Fecha de última actualización
Juan Pérez	1.0	15-11-2024	15-11-2024

Descripción de incidencia

Información general

Sucursal	1966
Caja	21
Folio de venta	V19662100030613
Fecha de venta	2024-06-24 14:07:43.730
Monedero	Si
Contiene promoción	Si
Versión de software de la incidencia	10.5.60692

Incidencia reportada

Al momento de realizar reimpresiones del ticket de la venta con folio **V19662100030613** estos salen distintos en la información de la transacción.

- El primer ticket fue reimpreso en el momento de la venta 14:07 pm
- El segundo ticket fue reimpreso en la noche 20:40 pm
- En el primer ticket aparecen **2 piezas del producto IMMU-FLO** suplemento con código relacionado: **7500327050267** en el segundo ya solo aparece 1 pieza siendo que es el mismo folio de venta.
- En el primer ticket el total a pagar se muestra por la cantidad de: **\$2,931.25** pero en la parte inferior indica que el cliente pagó **\$2332.25** como si no hubiera pagado completo.
- En el primer ticket indica que **no se abonó al monedero** y en el **segundo si**, adicional el iva no coincide.



INC - 20241115 - Diferente ticket, mismo folio de venta

Autor	Versión	Fecha de elaboración	Fecha de última actualización
Juan Pérez	1.0	15-11-2024	15-11-2024

- La segunda pieza que marca el primer ticket reimpresso **no fue descontada del inventario.**

Primera reimpresión a las **02:07 PM:**



[Ver ticket](#)



INC - 20241115 - Diferente ticket, mismo folio de venta

Autor	Versión	Fecha de elaboración	Fecha de última actualización
Juan Pérez	1.0	15-11-2024	15-11-2024

Segunda reimpresión a las **08:40 PM:**

Farmacias del Ahorro
*** REIMPRESIÓN. ***
COMERCIALIZADORA FARMACEUTICA DE CHIAPAS
REGIMEN FISCAL: GO1-GENERAL DE LEY PERSONAS MORALES
ALFONSO REYES 100 G VALLE PONIENTE SECTOR OLINCA
66196 SANTA CATARINA NL SANTA CATARINA NUEVO LEON
RFC: CFC110121742
LUGAR DE EXPEDICION: 53100
F A MEXICO NORTE, ECONOMISTAS CIRCUITO ECONOMISTAS F
- 5
CIUDAD SATELITE MEXICO NAUCALPAN DE JUAREZ MEXICO
53100
PROGRAMA: Programa de Lealtad Origis
No. Cliente o Cuenta: *****5469
Venta registrada en el programa
V19662100030613 24/06/2024 14:07
Tipo de Venta: CONTADO
Multifuncional: MARIA DEL ROCIO PEÑA FUENTES
Le Atendio: MARIA DEL ROCIO PEÑA FUENTES
Cantidad Descripción Precio Importe
1 IMMU-FLO SUPLEMENTO \$516.38 \$516.38I
Código: 7500327050267
Monedero -5129.09
1 FITOESTIMULINA 30 G \$480.00 \$480.00C
Código: 7501123011704
1 G AHORRO CIPROFLOXAC \$160.00 \$160.00C
Código: 7502276853470
1 MISDAPRE RAC 10/SMG \$1,016.00 \$1,016.00C
Código: 8904091121793
1 NASOGEL SPRAY 30 ML \$195.69 \$195.69I
Código: 705928045309
TOTAL BASE: \$2,368.07
DESCUENTO: \$0.00
IVA: \$93.28
TOTAL A PAGAR: \$2,332.25
PAGO DEL CLIENTE: \$2,332.25
CAMBIO: \$0.00
DOS MIL TRESCIENTOS TREINTA Y DOS PESOS 25/100 M.N.
DESGLOSE:
SUBTOTAL IVA 0%: \$1,656.00
SUBTOTAL IVA 16%: \$582.97
SUBTOTAL VTA IVA EXENTO: \$0.00
I.V.A.%: \$93.28
FORMA DE PAGO:
TARJETA DE CREDITO ACI (AC): \$2,332.25
Autorización : 000707
Tarjeta : XXXXXXXXXXXXXXXX19|0
No. Monedero: *****5469 Tran: 1540785886
Record Promoción: \$81.66 Te faltan \$18.34
Saldo total del monedero: \$69.17
Esta compra acumuló: \$20.43
!!! TU AHORRO \$740.57 !!!
Autorización: 1540785886
Beneficios:
Monedero
25.00% DE DESCUENTO EN IMMU-FLO SUPLEMENTO
ALIMENTICIO 30 CAPS 0267 \$599.00

Cuenta con 30 días naturales para solicitar su
factura electrónica en sucursal donde hizo su
compra o escanee el código QR con su celular o
abra la siguiente dirección
<https://1.fdah.mx/2166da9de43f>

[Ver ticket](#)

	INC - 20241115 - Diferente ticket, mismo folio de venta		
Autor	Versión	Fecha de elaboración	Fecha de última actualización
Juan Pérez	1.0	15-11-2024	15-11-2024

Diagnóstico

Versión de software utilizada	tag: 10.5.60692.12
Se reproduce el error	Si
Se corrige el error	Si

Se revisa el flujo que tuvo la venta en el log [20240624_FARMAX_TILL_SERVER_SALE.XLOG](#), a partir de la **línea 949** se tiene:

```
[2024-06-24 14:07:08:157],FARMAX_TILL_SERVER,SALE,INFO,,Venta :: Metodo:AddSale
FolioVenta:V19662100030613 Estado:N
```

...

```
[2024-06-24 14:07:20:473],FARMAX_TILL_SERVER,SALE,INFO,,Donativo :: Metodo:UpdateSaleHeader
FolioVenta:V19662100030613 Estado:R Monto: IdBeneficiencia: FolioDonativo:
```

...

```
[2024-06-24 14:07:46:262],FARMAX_TILL_SERVER,SALE,INFO,,Donativo :: Metodo:UpdateSaleHeader
FolioVenta:V19662100030613 Estado:V Monto: IdBeneficiencia: FolioDonativo:
```

```
[2024-06-24 14:07:46:279],FARMAX_TILL_SERVER,SALE,INFO,,Venta :: 
Metodo:UpdateSaleAfterWriteSaleDetails FolioVenta:V19662100030613
Estado:V
```

```
[2024-06-24 14:07:57:417],FARMAX_TILL_SERVER,SALE,INFO,,Venta :: Metodo:GetTransaction
FolioVenta:V19662100030613 Estado:R
```

```
[2024-06-24 14:09:24:437],FARMAX_TILL_SERVER,SALE,INFO,,Venta :: Metodo:GetTransaction
FolioVenta:V19662100030613 Estado:V
```

...

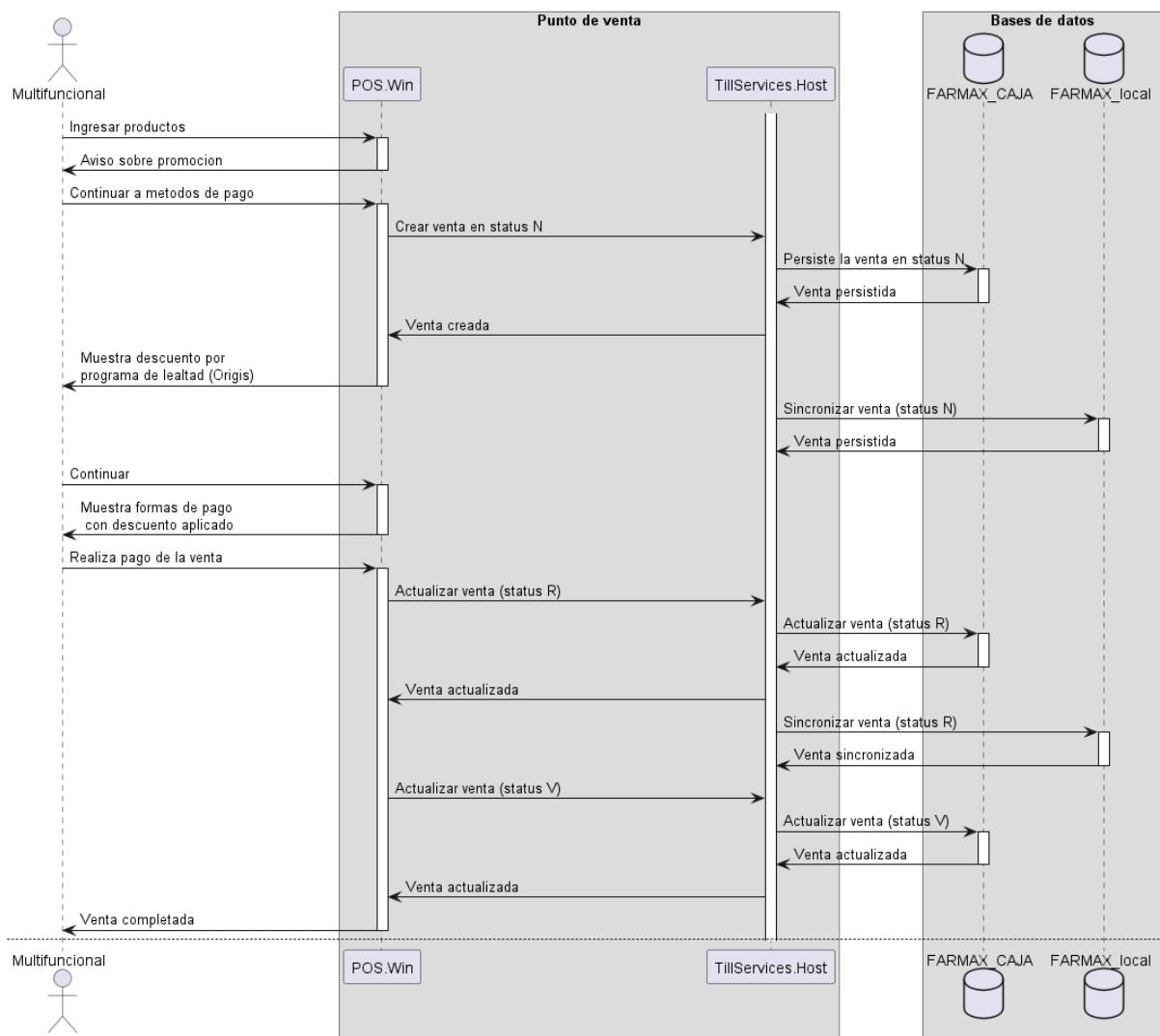
```
[2024-06-24 14:11:43:092],FARMAX_TILL_SERVER,SALE,INFO,,Venta :: Metodo:GetTransaction
FolioVenta:V19662100030613 Estado:V
```

```
[2024-06-24 14:12:36:528],FARMAX_TILL_SERVER,SALE,INFO,,Venta :: Metodo:GetTransaction
FolioVenta:V19662100030613 Estado:F
```

Autor	Versión	Fecha de elaboración	Fecha de última actualización
Juan Pérez	1.0	15-11-2024	15-11-2024

Se observa en la **Línea 968** que la venta fue actualizada a **status V**, se observa que luego, en la **Línea 969**, esta venta es obtenida nuevamente pero con un **status R**, esto no debería ser posible, al menos no para reimprimir la venta.

Se presenta un diagrama de alto nivel de lo ocurrido en la incidencia:

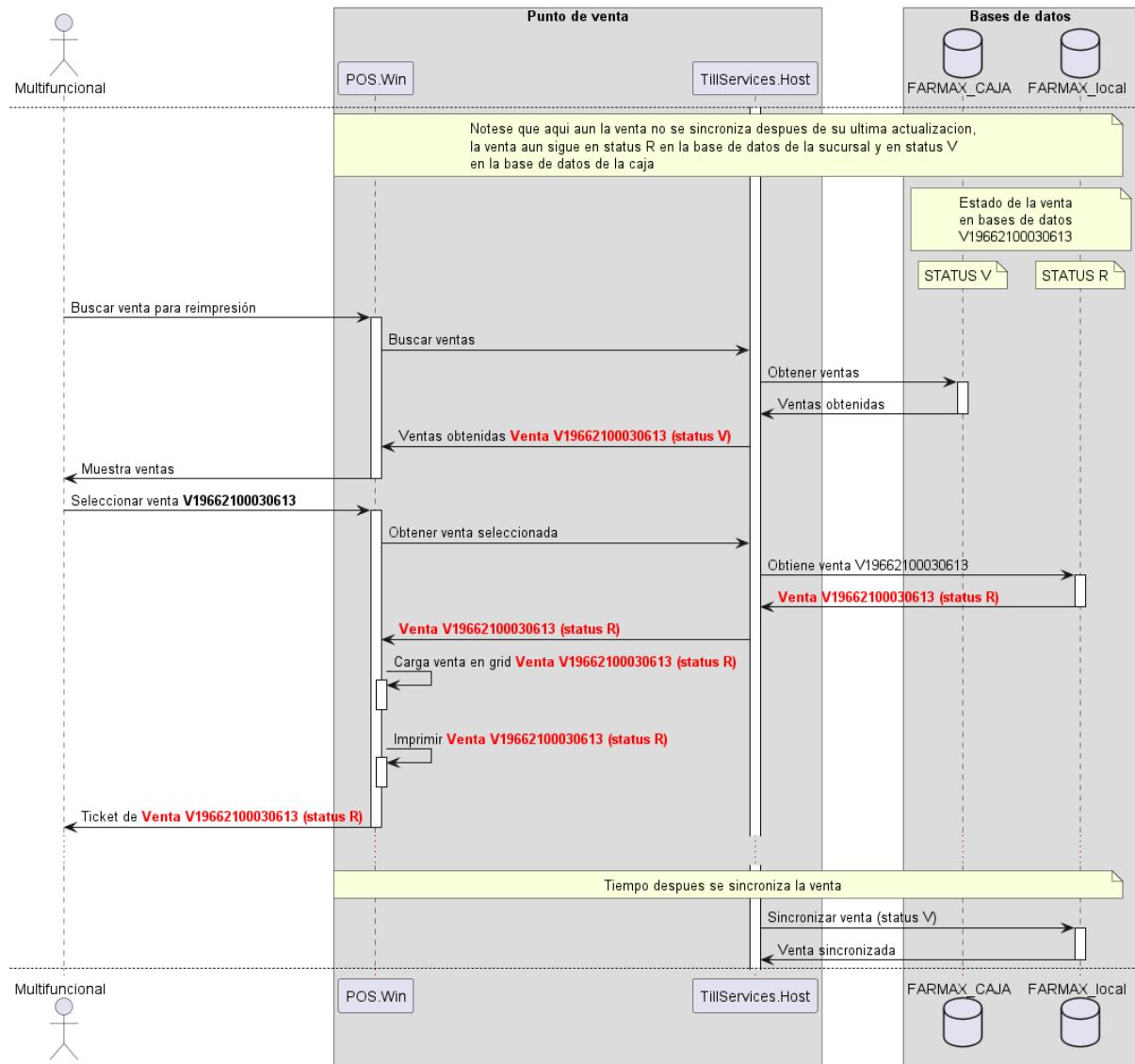


[Ver diagrama de flujo de incidencia \(página 1\)](#)



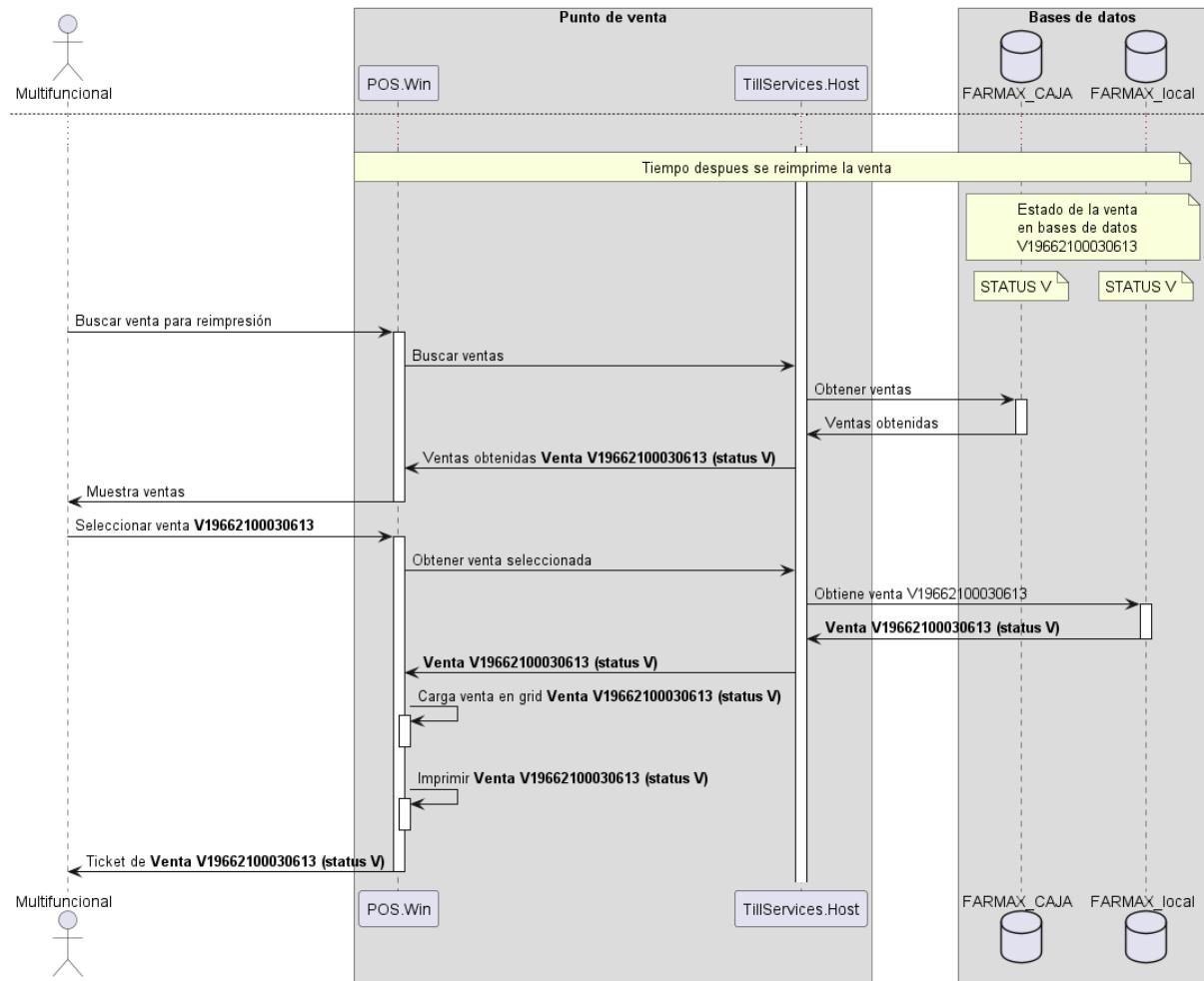
INC - 20241115 - Diferente ticket, mismo folio de venta

Autor	Versión	Fecha de elaboración	Fecha de última actualización
Juan Pérez	1.0	15-11-2024	15-11-2024



[Ver diagrama de flujo de incidencia \(página 2\)](#)

Autor	Versión	Fecha de elaboración	Fecha de última actualización
Juan Pérez	1.0	15-11-2024	15-11-2024



[Ver diagrama de flujo de incidencia \(página 3\)](#)

Las ventas se obtienen desde la base de datos de la caja para ser mostradas como ventas disponibles para reimpresión, una vez se selecciona una venta, esta es cargada desde la base de datos de la sucursal, en este caso se cargó la venta desactualizada desde la base de datos de la sucursal.

Esta carga ocurre en **POS.Win** en **SearchSalesViewModel.SearchByAnyFilter(...)**, en este método se obtienen las ventas desde la base de datos de caja para ser mostradas como disponibles para impresión.



INC - 20241115 - Diferente ticket, mismo folio de venta

Autor	Versión	Fecha de elaboración	Fecha de última actualización
Juan Pérez	1.0	15-11-2024	15-11-2024

Una vez se obtiene la venta, esta se carga desde la base de datos de la sucursal en **POS.Win** en **SaleViewModel.GetSale(...)**, se carga utilizando el **TransactionServiceClient.GetTransactionForReprint(...)**, este método utiliza el servicio de **TransactionService.GetTransaction(...)** ubicado en **TillServices.Host**, esté servicio a su vez utiliza **SaleManager.GetTransaction(...)** para obtener la venta.

GET Technologies	INC - 20241115 - Diferente ticket, mismo folio de venta		
Autor	Versión	Fecha de elaboración	Fecha de última actualización
Juan Pérez	1.0	15-11-2024	15-11-2024

Posibles causas

La información del saldo de monedero no se muestra porque la venta cargada (status R) desde la base de datos de la sucursal no contiene esa información, esta venta aún no ha sido actualizada y no contiene los valores correspondientes al saldo de monedero, por la misma razón no se muestra el método de pago de tarjeta y por la misma razón se muestran 2 piezas para el primer artículo de la venta, todo se debe a que **la transacción obtenida desde la base de datos de la sucursal se encontraba desactualizada**.

La adición de la segunda pieza se da porque en **SaleManager.GetTransaction(...)**, se agrega la cantidad de piezas con descuento como cantidad vendida en el detalle de la venta, antes de ser enviada al cliente, es importante mencionar que **esto solo ocurre con las ventas en status N, R o P**:

```

c# SaleManager.cs x
82     public partial class SaleManager
3343         public SaleTransaction GetTransaction(Guid saleId, string saleFolio, string folioDevolucion, int businessUnitId, string source, bool isTrainingMo
3357             using (var context = new SaleDbContext(contextToUse))
3359             else
3401                 if (saleResult != null && saleResult.Sale != null)
3437
3638
3639
3640         if (saleResult.Sale.status == SaleStatus.New
3641             || saleResult.Sale.status == SaleStatus.PaymentsPending
3642             || saleResult.Sale.status == SaleStatus.Pending)
3643         {
3644             // Si es una venta pendiente de pago con programa de lealtad se verifica si tiene algún beneficio
3645             // para agregarsela a su canasta de productos y se pueda volver a validar nuevamente la venta.
3646             var incompletPromotion = this.GetTemporalPromotionDetailsSale(contextToUse, saleResult.Sale.idTransaccion);
3647             saleResult.Sale.tieneProgramaLealtad = incompletPromotion.Any(p => p.LoyaltyProgramID != EnumLoyaltyProgram.Undefined);
3648             incompletPromotion.Where(p => p.LoyaltyProgramID != EnumLoyaltyProgram.Undefined).ToList().ForEach(p =>
3649             {
3650                 if (saleResult.Sale.trnTransaccionesDet.Any(d => d.renglon == p.RowId))
3651                 {
3652                     // Si es un beneficio de un programa de lealtad de un producto que ya existe en su canasta, se le suma la cantidad
3653                     var productDetail = saleResult.Sale.trnTransaccionesDet.FirstOrDefault(d => d.renglon == p.RowId);
3654                     productDetail.cantidadVendida += p.AppliedPromotionQuantity;
3655                 }
3656             }
3657         }
3658     }
3659 }

```

[Ver figura](#)

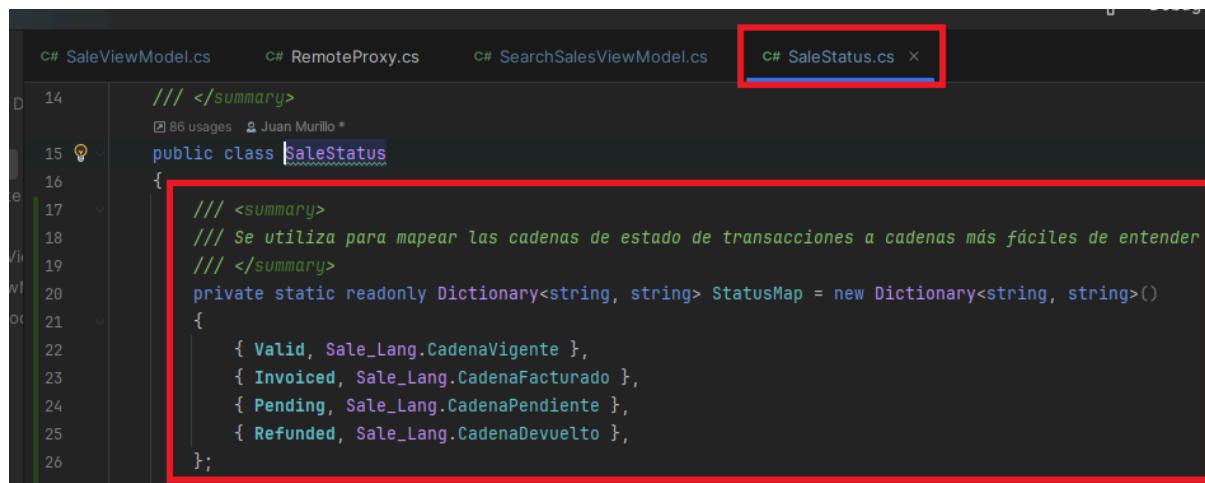
GET Technologies	INC - 20241115 - Diferente ticket, mismo folio de venta		
Autor	Versión	Fecha de elaboración	Fecha de última actualización
Juan Pérez	1.0	15-11-2024	15-11-2024

Propuesta de solución

Componentes impactados por el cambio	
Nombre	Tipo
POS.Win	Frontend

Se debe revisar que la venta en la sucursal se encuentra actualizada con el mismo estado de venta que la venta en la caja.

Se agrega un diccionario para realizar un mapeo entre los estados de las transacciones a una cadena de texto legible por el multifuncional:



```

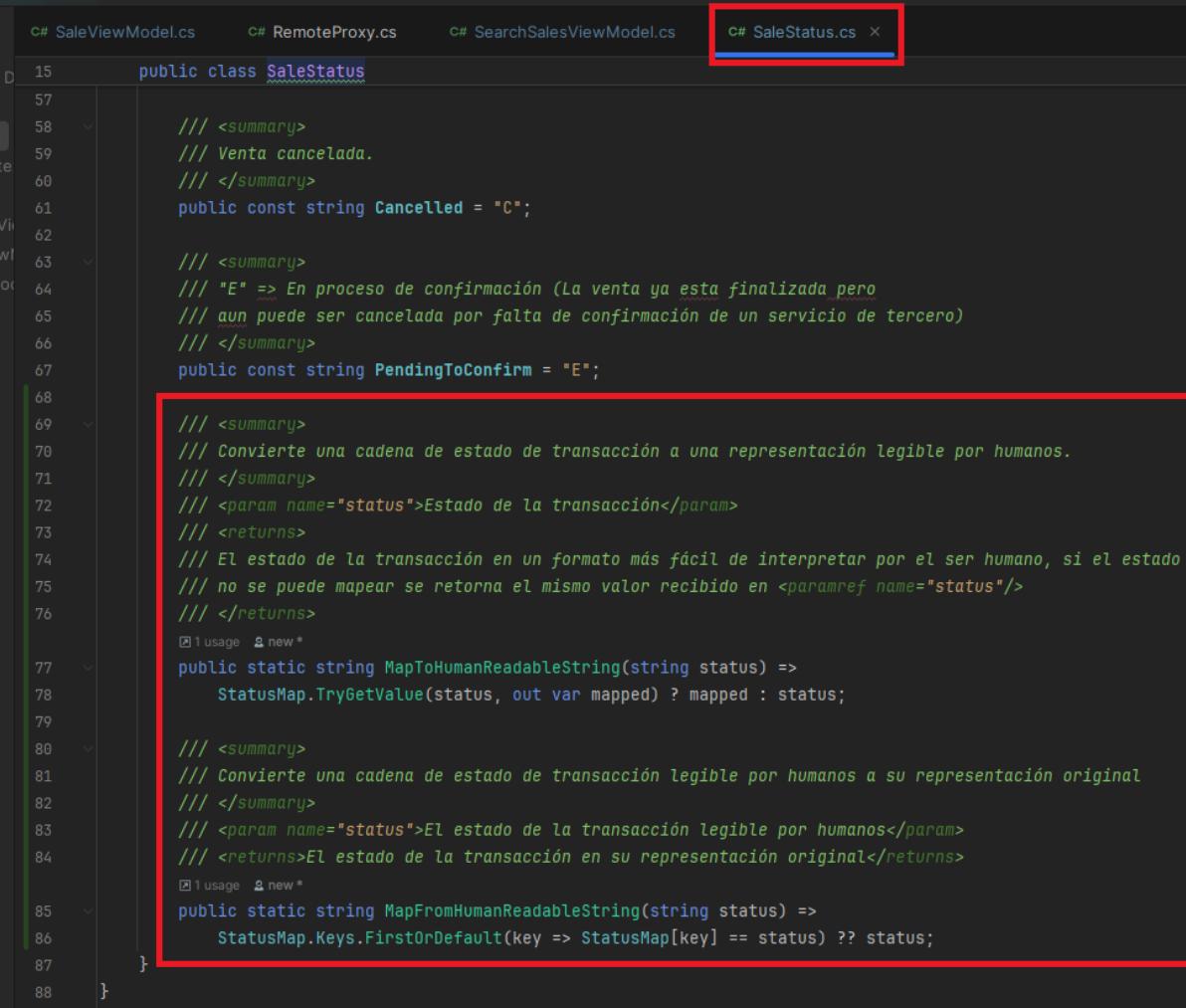
C# SaleViewModel.cs      C# RemoteProxy.cs      C# SearchSalesViewModel.cs      C# SaleStatus.cs ×
D 14      /// </summary>
15  ⚡     [ 86 usages  Juan Murillo *]
16  public class SaleStatus
17  {
18      /// <summary>
19      /// Se utiliza para mapear las cadenas de estado de transacciones a cadenas más fáciles de entender
20      /// </summary>
21      private static readonly Dictionary<string, string> StatusMap = new Dictionary<string, string>()
22      {
23          { Valid, Sale_Lang.CadenaVigente },
24          { Invoiced, Sale_Lang.CadenaFacturado },
25          { Pending, Sale_Lang.CadenaPendiente },
26          { Refunded, Sale_Lang.CadenaDevuelto },
27      };

```

[Ver figura](#)

GET Technologies	INC - 20241115 - Diferente ticket, mismo folio de venta		
Autor	Versión	Fecha de elaboración	Fecha de última actualización
Juan Pérez	1.0	15-11-2024	15-11-2024

Esto se hace con el propósito de implementar dos métodos estáticos que traduzcan los estados de un formato a otro:



```

C# SaleViewModel.cs      C# RemoteProxy.cs      C# SearchSalesViewModel.cs      C# SaleStatus.cs ×
D 15 public class SaleStatus
57
58     /// <summary>
59     /// Venta cancelada.
60     /// </summary>
61     public const string Cancelled = "C";
62
63     /// <summary>
64     /// "E" => En proceso de confirmación (La venta ya esta finalizada pero
65     /// aun puede ser cancelada por falta de confirmación de un servicio de tercero)
66     /// </summary>
67     public const string PendingToConfirm = "E";
68
69     /// <summary>
70     /// Convierte una cadena de estado de transacción a una representación legible por humanos.
71     /// </summary>
72     /// <param name="status">Estado de la transacción</param>
73     /// <returns>
74     /// El estado de la transacción en un formato más fácil de interpretar por el ser humano, si el estado
75     /// no se puede mapear se retorna el mismo valor recibido en <paramref name="status"/>
76     /// </returns>
77     [Usage("new")]
78     public static string MapToHumanReadableString(string status) =>
79         StatusMap.TryGetValue(status, out var mapped) ? mapped : status;
80
81     /// <summary>
82     /// Convierte una cadena de estado de transacción legible por humanos a su representación original
83     /// </summary>
84     /// <param name="status">El estado de la transacción legible por humanos</param>
85     /// <returns>El estado de la transacción en su representación original</returns>
86     [Usage("new")]
87     public static string MapFromHumanReadableString(string status) =>
88         StatusMap.Keys.FirstOrDefault(key => StatusMap[key] == status) ?? status;
89     }
90 }
```

[Ver figura](#)

Esto se debe hacer ya que en la vista de búsqueda de ventas para reimpresión se mapea el status de la venta a estas cadenas, se refactoriza a estos métodos para poder comparar luego con lo obtenido desde la base de datos de la sucursal, originalmente el mapeo se ve así:



INC - 20241115 - Diferente ticket, mismo folio de venta

Autor	Versión	Fecha de elaboración	Fecha de última actualización
Juan Pérez	1.0	15-11-2024	15-11-2024

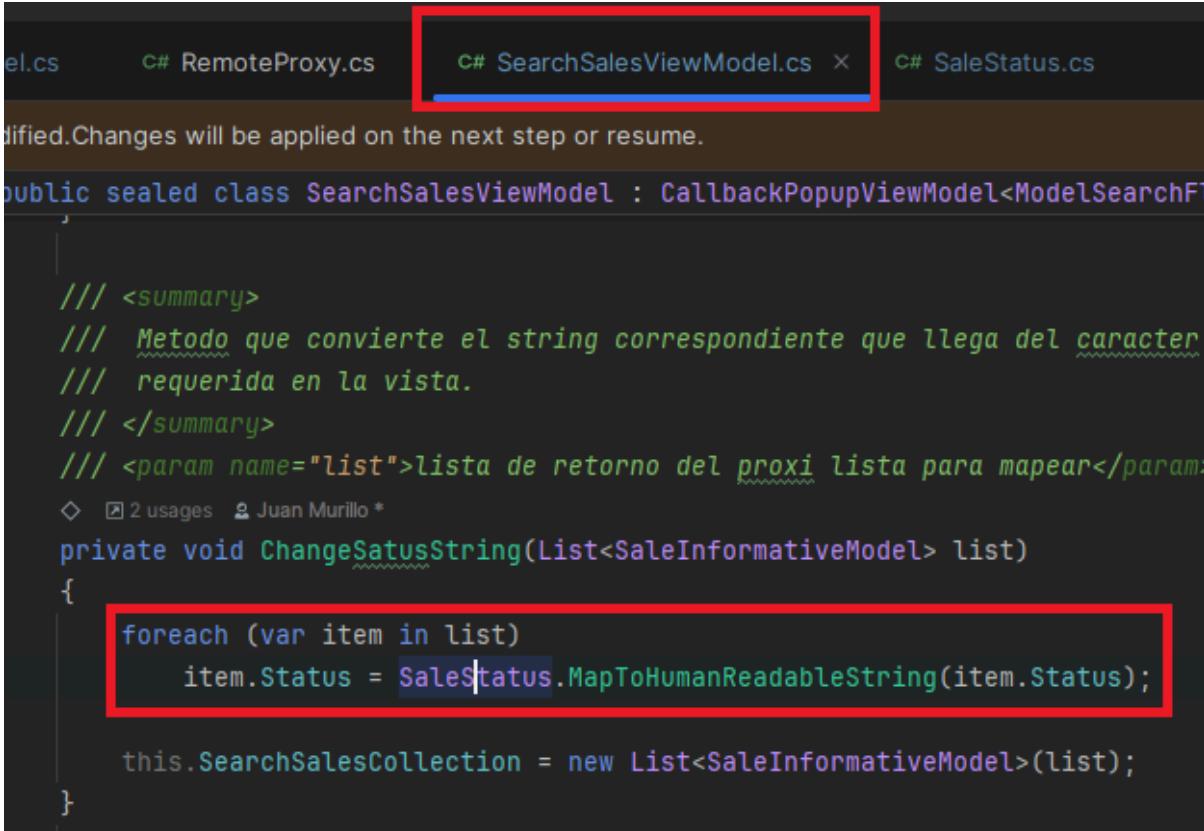
```
public sealed class SearchSalesViewModel : CallbackPopupViewModel<ModelSearchFlag>, IRe
    ///  que convierte el string correspondiente que llega del carácter status a
    /// requerida en la vista.
    /// 
    /// <param name="list">lista de retorno del proxy lista para mapear</param>
    ◇ 2 usages  Juan Murillo
    private void ChangeStatusString(List<SaleInformativeModel> list)
    {
        foreach (var item in list)
        {
            switch (item.Status)
            {
                case SaleStatus.Valid:
                    item.Status = Lang.CadenaVigente;
                    break;
                case SaleStatus.Invoiced:
                    item.Status = Lang.CadenaFacturado;
                    break;
                case SaleStatus.Pending:
                    item.Status = Lang.CadenaPendiente;
                    break;
                case SaleStatus.Refunded:
                    item.Status = Lang.CadenaDevuelto;
                    break;
            }
        }

        this.SearchSalesCollection = new List<SaleInformativeModel>(list);
    }
}
```

[Ver figura](#)

GET Technologies	INC - 20241115 - Diferente ticket, mismo folio de venta		
Autor	Versión	Fecha de elaboración	Fecha de última actualización
Juan Pérez	1.0	15-11-2024	15-11-2024

Se cambia a utilizar el nuevo método:



```

el.cs      C# RemoteProxy.cs      C# SearchSalesViewModel.cs ×      C# SaleStatus.cs

modified.Changes will be applied on the next step or resume.

public sealed class SearchSalesViewModel : CallbackPopupViewModel<ModelSearchF1>
{
    /// <summary>
    /// Metodo que convierte el string correspondiente que llega del caracter requerida en la vista.
    /// </summary>
    /// <param name="list">lista de retorno del proxy lista para mapear</param>
    ◇ 2 usages  Juan Murillo *
    private void ChangeStatusString(List<SaleInformativeModel> list)
    {
        foreach (var item in list)
            item.Status = SaleStatus.MapToHumanReadableString(item.Status);

        this.SearchSalesCollection = new List<SaleInformativeModel>(list);
    }
}

```

[Ver figura](#)

Esto hace que la vista siga teniendo el mismo comportamiento, ahora es más fácil regresar el estado de la venta a su valor original y compararlo con el estado en base de datos de sucursal en **SaleViewModel.GetSale(...)**:



INC - 20241115 - Diferente ticket, mismo folio de venta

Autor	Versión	Fecha de elaboración	Fecha de última actualización
Juan Pérez	1.0	15-11-2024	15-11-2024

```
C# SaleViewModel.cs × Localization Manager Sale.Messages.resx C# SaleTicket.cs C# RemoteProxy.cs C# SearchSalesVI  
Sources are modified. Changes will be applied on the next step or resume.  
122     public sealed class SaleViewModel : ViewModelBase, INavigationAware  
3088         saleModel = client.GetTransactionForReprint(modelInSearch.Guid.Value, modelInSearch.Folio$  
3089     }  
3090  
3091         if (saleModel != null)  
3092     {  
3093         // Validacion de integridad  
3094         // Revisa si la venta en base de datos de sucursal se encuentra en el mismo estado que  
3095         // la venta de la caja.  
3096         var saleStateInStore = saleModel.Status;  
3097         var saleStatusInTill = SaleStatus.MapFromHumanReadableString(modelInSearch.Status);  
3098         if (saleStateInStore != saleStatusInTill)  
3099         {  
3100             this.NotificationManager.ShowMessage(  
3101                 title: Messages.TitleNotification,  
3102                 message: Messages.SaleBadReprintIntegrity,  
3103                 image: MessageIcon.Warning);  
3104  
3105             Log.LogWarning(  
3106                 LogSubModules.SubModule_Sale,  
3107                 string.Format(  
3108                     Messages.SaleBadReprintIntegrityLog,  
3109                     saleModel.SaleIdentifier,  
3110                     saleStatusInTill,  
3111                     saleStateInStore  
3112                 )  
3113             );  
3114         }  
3115     }  
3116  
3117 }
```

[Ver figura](#)

	INC - 20241115 - Diferente ticket, mismo folio de venta		
Autor	Versión	Fecha de elaboración	Fecha de última actualización
Juan Pérez	1.0	15-11-2024	15-11-2024

Los mensajes en cuestión son:

```

var saleStateInStore = saleModel.Status;
var saleStatusInTill = SaleStatus.MapFromHumanReadableString(modelInSearch.Status);
if (saleStateInStore != saleStatusInTill)
{
    this.NotificationManager.ShowMessage(
        title: Messages.TitleNotification,
        message: Messages.SaleBadReprintIntegrity,
        image: MessageIcon.Warning);

    Log.LogWarning(
        LogSubModules.SubModule_Sale
        string.Format(
            Messages.SaleBadReprintIntegrityLog,
            saleModel.SaleIdentifier,
            saleStatusInTill,
            saleStateInStore
        )
    );
}
return;

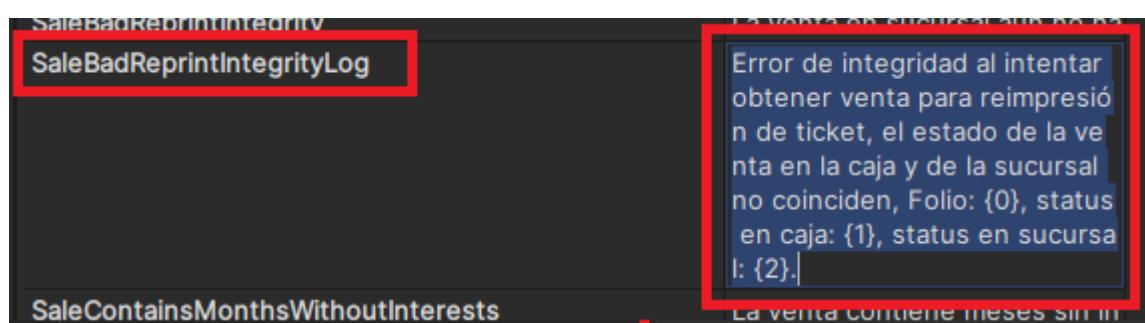
```

resource item SaleBadReprintIntegrity

Value:
Default Culture: "La venta en sucursal aun no ha sido actualizada, intenta de nuevo en un momento."

[Edit with Localization Manager](#)

[Ver figura](#)



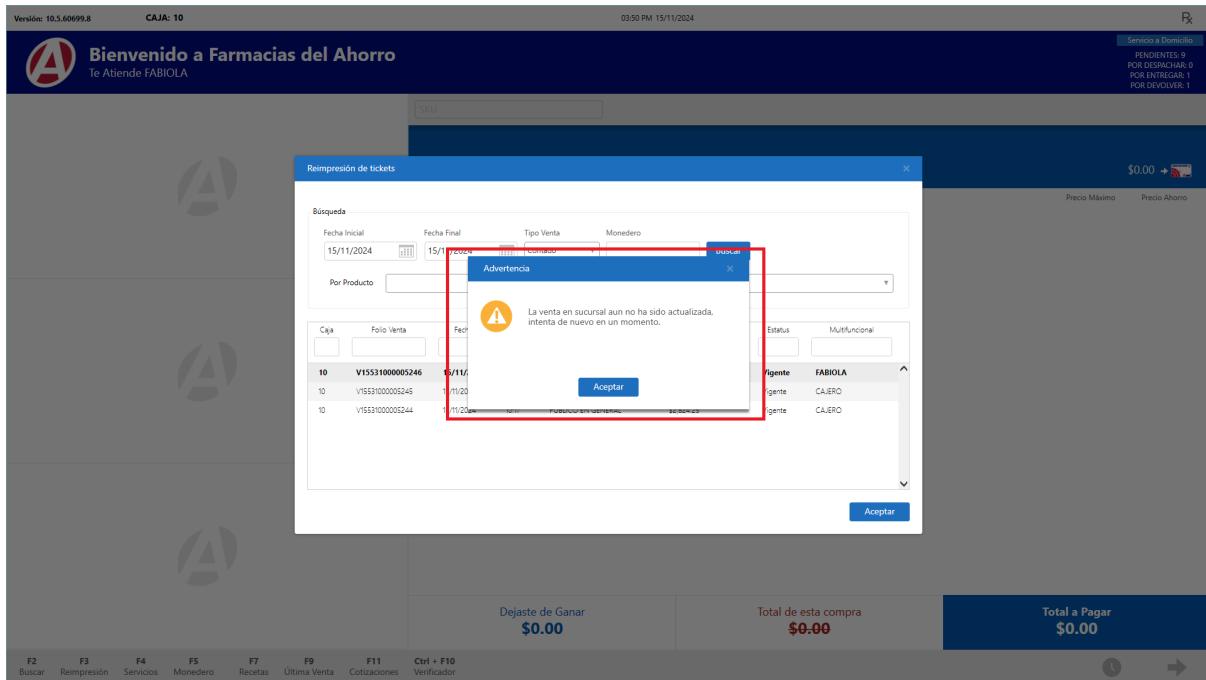
[Ver figura](#)

Se mostrará una notificación al usuario para que intente realizar la impresión nuevamente después de esperar un momento:



INC - 20241115 - Diferente ticket, mismo folio de venta

Autor	Versión	Fecha de elaboración	Fecha de última actualización
Juan Pérez	1.0	15-11-2024	15-11-2024



[Ver figura](#)