

Mismo corte para 2 multifuncionales

1. Resumen del Problema

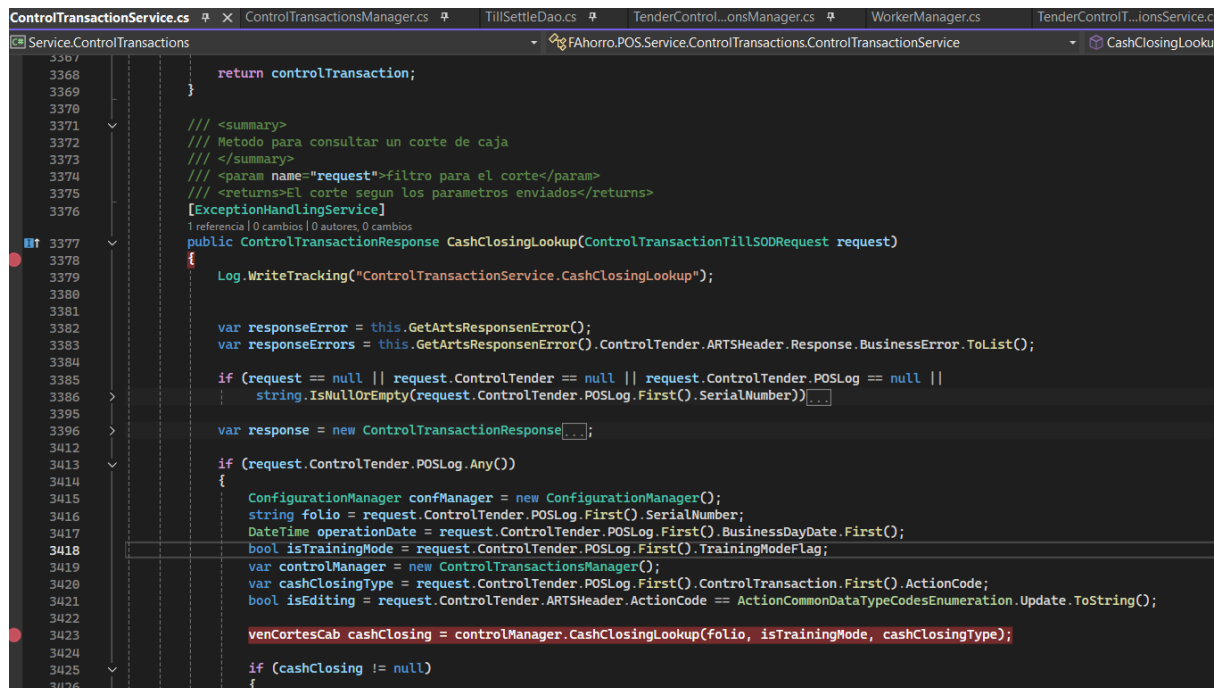
El incidente identificado con el número INC 2024-191414 fue reportado vía Proactivanet, detallando un error relacionado con el corte de turno CT0326220000359. El problema reportado consiste en que, al imprimir el corte por primera vez, se mostraba correctamente el nombre del multifuncional **Olivia Palma Muñoz**. Sin embargo, al realizar la reimpresión del mismo corte, el nombre que apareció fue el de otro multifuncional, **María de Lourdes Lecona Cruz**. Este comportamiento inconsistente genera confusión y afecta el correcto seguimiento de los turnos en el sistema.

2. Análisis Inicial

Primer paso: Identificación de métodos involucrados en los contratos WCF para la reimpresión del corte de caja

Como primer paso, se identifican los métodos responsables o involucrados en los contratos WCF relacionados con la reimpresión del corte de caja. En este caso, se trata del método **CashClosingLookup**, el cual realiza una consulta en el repositorio de **ControlTransaction**. Dicho método, a su vez, realiza una consulta en la base de datos del sistema "Store".

A continuación, se muestra el flujo de los métodos relacionados y la consulta LINQ convertida a SQL, tal como fue arrojada por el servicio.



```
ControlTransactionService.cs | ControlTransactionsManager.cs | TillSettleDao.cs | TenderControl...ionsManager.cs | WorkerManager.cs | TenderControlT...ionsService.c
Service.ControlTransactions
3367
3368         return controlTransaction;
3369     }
3370
3371     /// <summary>
3372     /// Metodo para consultar un corte de caja
3373     /// </summary>
3374     /// <param name="request">filtro para el corte</param>
3375     /// <returns>El corte segun los parametros enviados</returns>
3376     [ExceptionHandlerService]
3377     public ControlTransactionResponse CashClosingLookup(ControlTransactionTillSODRequest request)
3378     {
3379         Log.WriteTracking("ControlTransactionService.CashClosingLookup");
3380
3381         var responseError = this.GetArtsResponseError();
3382         var responseErrors = this.GetArtsResponseError().ControlTender.ARTSHeader.Response.BusinessError.ToList();
3383
3384         if (request == null || request.ControlTender == null || request.ControlTender.POSLog == null ||
3385             string.IsNullOrEmpty(request.ControlTender.POSLog.First().SerialNumber))
3386         {
3387             var response = new ControlTransactionResponse();
3388
3389             if (request.ControlTender.POSLog.Any())
3390             {
3391                 ConfigurationManager confManager = new ConfigurationManager();
3392                 string folio = request.ControlTender.POSLog.First().SerialNumber;
3393                 DateTime operationDate = request.ControlTender.POSLog.First().BusinessDayDate.First();
3394                 bool isTrainingMode = request.ControlTender.POSLog.First().TrainingModeFlag;
3395                 var controlManager = new ControlTransactionsManager();
3396                 var cashClosingType = request.ControlTender.POSLog.First().ControlTransaction.First().ActionCode;
3397                 bool isEditing = request.ControlTender.ARTSHeader.ActionCode == ActionCommonDataTypeCodesEnumeration.Update.ToString();
3398
3399                 venCortesCab cashClosing = controlManager.CashClosingLookup(folio, isTrainingMode, cashClosingType);
3400
3401                 if (cashClosing != null)
3402                 {
3403                     // ...
3404                 }
3405             }
3406         }
3407     }
```

Img. Se muestra el servicio responsable de la comunicación del WCF (pos service) con el aplicativo de Caja (win).

```

/// <summary>
/// Busca un corte de caja segun el filtro seleccionado
/// </summary>
/// <param name="folio">numero de folio para hacer la consulta</param>
/// <param name="isTrainingMode">indica si es un corte en modo entrenamiento</param>
/// <param name="cashClosingType">Tipo de corte</param>
/// <returns>Retorna un corte de caja</returns>
1 referencia | 0 cambios | 0 autores, 0 cambios
public venCortesCab CashClosingLookup(string folio, bool isTrainingMode, string cashClosingType)
{
    venCortesCab cashClosing;
    using (var context = new ControlTransactionsDBContext())
    {
        var tillSettle = new TillSettleDao(context);
        cashClosing = tillSettle.GetCashClosingByFolio(folio, isTrainingMode, cashClosingType);
    }

    return cashClosing;
}

```

```

/// <summary>
/// Obtiene un corte de caja con su detalle
/// </summary>
/// <param name="folio">folio a consultar</param>
/// <param name="isTrainingMode">indica si esta en modo entrenamiento</param>
/// <param name="cashCloingType">Tipo de corte</param>
/// <returns>Retoruna un corte de caja</returns>
3 referencias | 0 cambios | 0 autores, 0 cambios
public venCortesCab GetCashClosingByFolio(string folio, bool isTrainingMode, string cashCloingType)
{
    var cashClosing = this.DataContext.venCortesCab.Include("venTurnosCajaReg.genCajasCat").Include("venCortesDet").Include("venCortesCabHist")
        .FirstOrDefault(a => a.folioCorte == folio &&
            a.estaModoEntrenamiento == isTrainingMode &&
            a.tipoCorte == cashCloingType);
    return cashClosing;
}

```

Img. Se obtienen los registrados de la bd del Store asociados a los filtros por folio, modoEntrenamiento y tipo de corte.

A continuación se muestra el query convertida de linq a Sql server:

```

-----
DECLARE @folioCorte NVARCHAR(50) = 'CT0326220000359',
        @tipoCorte nchar(1) = 'P',
        @codigoSucursal int    = 0326;

```

```

SELECT
    [UnionAll1].[codigoSucursal] AS [C1],
    [UnionAll1].[codigoCorte] AS [C2],
    [UnionAll1].[codigoRetiro] AS [C3],
    [UnionAll1].[codigoTurno] AS [C4],
    [UnionAll1].[codigoSucursal1] AS [C5],
    [UnionAll1].[tipoCorte] AS [C6],
    [UnionAll1].[folioCorte] AS [C7],
    [UnionAll1].[estaModoEntrenamiento] AS [C8],
    [UnionAll1].[fechaOperacion] AS [C9],
    [UnionAll1].[codigoUsuario] AS [C10],
    [UnionAll1].[fechaCreacion] AS [C11],
    [UnionAll1].[fechaActualizacion] AS [C12],
    [UnionAll1].[versionSoftware] AS [C13],
    [UnionAll1].[idFechaOperacion] AS [C14],
    [UnionAll1].[codigoSucursal2] AS [C15],
    [UnionAll1].[codigoTurno1] AS [C16],
    [UnionAll1].[codigoSucursal3] AS [C17],
    [UnionAll1].[numeroCaja] AS [C18],
    [UnionAll1].[codigoCajero] AS [C19],
    [UnionAll1].[estaModoEntrenamiento1] AS [C20],
    [UnionAll1].[fechaOperacion1] AS [C21],
    [UnionAll1].[dotacionInicial] AS [C22],

```

```

[UnionAll1].[fechaHoraInicioTurno] AS [C23],
[UnionAll1].[fechaHoraFinalTurno] AS [C24],
[UnionAll1].[cajaDisponible] AS [C25],
[UnionAll1].[codigoUsuario1] AS [C26],
[UnionAll1].[fechaActualizacion1] AS [C27],
[UnionAll1].[versionSoftware1] AS [C28],
[UnionAll1].[idFechaOperacion1] AS [C29],
[UnionAll1].[codigoGrupoCierreDia] AS [C30],
[UnionAll1].[noTurno] AS [C31],
[UnionAll1].[ultimoHeartBeat] AS [C32],
[UnionAll1].[reutilizarTurno] AS [C33],
[UnionAll1].[cierreAutomaticoTurno] AS [C34],
[UnionAll1].[estadoActual] AS [C35],
[UnionAll1].[codigoSucursal4] AS [C36],
[UnionAll1].[numeroCaja1] AS [C37],
[UnionAll1].[estaActivo] AS [C38],
[UnionAll1].[codigoCaja] AS [C39],
[UnionAll1].[C1] AS [C40],
[UnionAll1].[codigoCorte1] AS [C41],
[UnionAll1].[codigoFormaPago] AS [C42],
[UnionAll1].[renglon] AS [C43],
[UnionAll1].[importeFisico] AS [C44],
[UnionAll1].[importeTeorico] AS [C45],
[UnionAll1].[importeRetiro] AS [C46],
[UnionAll1].[cambio] AS [C47],
[UnionAll1].[tipoDeCambio] AS [C48],
[UnionAll1].[importeFaltanteRecuperado] AS [C49],
[UnionAll1].[codigoCorte2] AS [C50],
[UnionAll1].[C2] AS [C51],
[UnionAll1].[C3] AS [C52],
[UnionAll1].[C4] AS [C53],
[UnionAll1].[C5] AS [C54],
[UnionAll1].[C6] AS [C55],
[UnionAll1].[C7] AS [C56],
[UnionAll1].[C8] AS [C57],
[UnionAll1].[C9] AS [C58],
[UnionAll1].[C10] AS [C59],
[UnionAll1].[C11] AS [C60],
[UnionAll1].[C12] AS [C61],
[UnionAll1].[C13] AS [C62],
[UnionAll1].[C14] AS [C63]
FROM (SELECT
    CASE WHEN ([Extent4].[codigoFormaPago] IS NULL) THEN CAST(NULL AS int) ELSE 1 END AS [C1],
    [Extent1].[codigoSucursal] AS [codigoSucursal],
    [Extent1].[codigoCorte] AS [codigoCorte],
    [Extent1].[codigoRetiro] AS [codigoRetiro],
    [Extent1].[codigoTurno] AS [codigoTurno],
    [Extent1].[codigoSucursal] AS [codigoSucursal1],
    [Extent1].[tipoCorte] AS [tipoCorte],
    [Extent1].[folioCorte] AS [folioCorte],
    [Extent1].[estaModoEntrenamiento] AS [estaModoEntrenamiento],
    [Extent1].[fechaOperacion] AS [fechaOperacion],
    [Extent1].[codigoUsuario] AS [codigoUsuario],
    [Extent1].[fechaCreacion] AS [fechaCreacion],
    [Extent1].[fechaActualizacion] AS [fechaActualizacion],
    [Extent1].[versionSoftware] AS [versionSoftware],
    [Extent1].[idFechaOperacion] AS [idFechaOperacion],
    [Extent1].[codigoSucursal] AS [codigoSucursal2],
    [Extent2].[codigoTurno] AS [codigoTurno1],
    [Extent2].[codigoSucursal] AS [codigoSucursal3],
    [Extent2].[numeroCaja] AS [numeroCaja],
    [Extent2].[codigoCajero] AS [codigoCajero],
    [Extent2].[estaModoEntrenamiento] AS [estaModoEntrenamiento1],
    [Extent2].[fechaOperacion] AS [fechaOperacion1],
    [Extent2].[dotacionInicial] AS [dotacionInicial],
    [Extent2].[fechaHoraInicioTurno] AS [fechaHoraInicioTurno],

```

```

[Extent2].[fechaHoraFinalTurno] AS [fechaHoraFinalTurno],
[Extent2].[cajaDisponible] AS [cajaDisponible],
[Extent2].[codigoUsuario] AS [codigoUsuario1],
[Extent2].[fechaActualizacion] AS [fechaActualizacion1],
[Extent2].[versionSoftware] AS [versionSoftware1],
[Extent2].[idFechaOperacion] AS [idFechaOperacion1],
[Extent2].[codigoGrupoCierreDia] AS [codigoGrupoCierreDia],
[Extent2].[noTurno] AS [noTurno],
[Extent2].[ultimoHeartBeat] AS [ultimoHeartBeat],
[Extent2].[reutilizarTurno] AS [reutilizarTurno],
[Extent2].[cierreAutomaticoTurno] AS [cierreAutomaticoTurno],
[Extent2].[estadoActual] AS [estadoActual],
[Extent3].[codigoSucursal] AS [codigoSucursal4],
[Extent3].[numeroCaja] AS [numeroCaja1],
[Extent3].[estaActivo] AS [estaActivo],
[Extent3].[codigoCaja] AS [codigoCaja],
[Extent4].[codigoCorte] AS [codigoCorte1],
[Extent4].[codigoFormaPago] AS [codigoFormaPago],
[Extent4].[renglon] AS [renglon],
[Extent4].[importeFisico] AS [importeFisico],
[Extent4].[importeTeorico] AS [importeTeorico],
[Extent4].[importeRetiro] AS [importeRetiro],
[Extent4].[cambio] AS [cambio],
[Extent4].[tipoDeCambio] AS [tipoDeCambio],
[Extent4].[importeFaltanteRecuperado] AS [importeFaltanteRecuperado],
[Extent4].[codigoCorte] AS [codigoCorte2],
CAST(NULL AS uniqueidentifier) AS [C2],
CAST(NULL AS uniqueidentifier) AS [C3],
CAST(NULL AS uniqueidentifier) AS [C4],
CAST(NULL AS uniqueidentifier) AS [C5],
CAST(NULL AS int) AS [C6],
CAST(NULL AS varchar(1)) AS [C7],
CAST(NULL AS varchar(1)) AS [C8],
CAST(NULL AS bit) AS [C9],
CAST(NULL AS datetime2) AS [C10],
CAST(NULL AS int) AS [C11],
CAST(NULL AS datetime2) AS [C12],
CAST(NULL AS int) AS [C13],
CAST(NULL AS bigint) AS [C14]
FROM [dbo].[venCortesCab] AS [Extent1]
INNER JOIN [dbo].[venTurnosCajaReg] AS [Extent2] ON [Extent1].[codigoTurno] = [Extent2].[codigoTurno]

```

```

-----
INNER JOIN [dbo].[genCajasCat] AS [Extent3] ON ([Extent2].[codigoSucursal] = [Extent3].[codigoSucursal]) AND
([Extent2].[numeroCaja] = [Extent3].[numeroCaja])
LEFT OUTER JOIN [dbo].[venCortesDet] AS [Extent4] ON [Extent1].[codigoCorte] = [Extent4].[codigoCorte]
WHERE ([Extent1].[folioCorte] = @folioCorte) AND ([Extent1].[estaModoEntrenamiento] = 0) AND ([Extent1].[tipoCorte] =
@tipoCorte)
UNION ALL
SELECT
2 AS [C1],
[Extent5].[codigoSucursal] AS [codigoSucursal],
[Extent5].[codigoCorte] AS [codigoCorte],
[Extent5].[codigoRetiro] AS [codigoRetiro],
[Extent5].[codigoTurno] AS [codigoTurno],
[Extent5].[codigoSucursal] AS [codigoSucursal1],
[Extent5].[tipoCorte] AS [tipoCorte],
[Extent5].[folioCorte] AS [folioCorte],
[Extent5].[estaModoEntrenamiento] AS [estaModoEntrenamiento],
[Extent5].[fechaOperacion] AS [fechaOperacion],
[Extent5].[codigoUsuario] AS [codigoUsuario],
[Extent5].[fechaCreacion] AS [fechaCreacion],
[Extent5].[fechaActualizacion] AS [fechaActualizacion],
[Extent5].[versionSoftware] AS [versionSoftware],
[Extent5].[idFechaOperacion] AS [idFechaOperacion],
[Extent5].[codigoSucursal] AS [codigoSucursal2],
[Extent6].[codigoTurno] AS [codigoTurno1],

```

```

[Extent6].[codigoSucursal] AS [codigoSucursal3],
[Extent6].[numeroCaja] AS [numeroCaja],
[Extent6].[codigoCajero] AS [codigoCajero],
[Extent6].[estaModoEntrenamiento] AS [estaModoEntrenamiento1],
[Extent6].[fechaOperacion] AS [fechaOperacion1],
[Extent6].[dotacionInicial] AS [dotacionInicial],
[Extent6].[fechaHoraInicioTurno] AS [fechaHoraInicioTurno],
[Extent6].[fechaHoraFinalTurno] AS [fechaHoraFinalTurno],
[Extent6].[cajaDisponible] AS [cajaDisponible],
[Extent6].[codigoUsuario] AS [codigoUsuario1],
[Extent6].[fechaActualizacion] AS [fechaActualizacion1],
[Extent6].[versionSoftware] AS [versionSoftware1],
[Extent6].[idFechaOperacion] AS [idFechaOperacion1],
[Extent6].[codigoGrupoCierreDia] AS [codigoGrupoCierreDia],
[Extent6].[noTurno] AS [noTurno],
[Extent6].[ultimoHeartBeat] AS [ultimoHeartBeat],
[Extent6].[reutilizarTurno] AS [reutilizarTurno],
[Extent6].[cierreAutomaticoTurno] AS [cierreAutomaticoTurno],
[Extent6].[estadoActual] AS [estadoActual],
[Extent7].[codigoSucursal] AS [codigoSucursal4],
[Extent7].[numeroCaja] AS [numeroCaja1],
[Extent7].[estaActivo] AS [estaActivo],
[Extent7].[codigoCaja] AS [codigoCaja],
CAST(NULL AS uniqueidentifier) AS [C2],
CAST(NULL AS int) AS [C3],
CAST(NULL AS int) AS [C4],
CAST(NULL AS decimal(19,4)) AS [C5],
CAST(NULL AS decimal(19,4)) AS [C6],
CAST(NULL AS decimal(19,4)) AS [C7],
CAST(NULL AS decimal(19,4)) AS [C8],
CAST(NULL AS decimal(19,4)) AS [C9],
CAST(NULL AS decimal(19,4)) AS [C10],
CAST(NULL AS uniqueidentifier) AS [C11],
[Extent8].[codigoCorteHist] AS [codigoCorteHist],
[Extent8].[codigoCorte] AS [codigoCorte1],
[Extent8].[codigoRetiro] AS [codigoRetiro1],
[Extent8].[codigoTurno] AS [codigoTurno2],
[Extent8].[codigoSucursal] AS [codigoSucursal5],
[Extent8].[tipoCorte] AS [tipoCorte1],
[Extent8].[folioCorte] AS [folioCorte1],
[Extent8].[estaModoEntrenamiento] AS [estaModoEntrenamiento2],
[Extent8].[fechaOperacion] AS [fechaOperacion2],
[Extent8].[codigoUsuario] AS [codigoUsuario2],
[Extent8].[fechaActualizacion] AS [fechaActualizacion2],
[Extent8].[versionSoftware] AS [versionSoftware2],
[Extent8].[idFechaOperacion] AS [idFechaOperacion2]
FROM [dbo].[venCortesCab] AS [Extent5]
INNER JOIN [dbo].[venTurnosCajaReg] AS [Extent6] ON [Extent5].[codigoTurno] = [Extent6].[codigoTurno]
-----
INNER JOIN [dbo].[genCajasCat] AS [Extent7] ON ([Extent6].[codigoSucursal] = [Extent7].[codigoSucursal]) AND
([Extent6].[numeroCaja] = [Extent7].[numeroCaja])
INNER JOIN [dbo].[venCortesCabHist] AS [Extent8] ON [Extent5].[codigoCorte] = [Extent8].[codigoCorte]
-----
WHERE ([Extent5].[folioCorte] = @folioCorte) AND ([Extent5].[estaModoEntrenamiento] = 0) AND ([Extent5].[tipoCorte] =
@tipoCorte)) AS [UnionAll1]
ORDER BY [UnionAll1].[codigoCorte] ASC, [UnionAll1].[codigoTurno1] ASC, [UnionAll1].[codigoSucursal4] ASC,
[UnionAll1].[numeroCaja1] ASC, [UnionAll1].[C1] ASC
-----

```

Ejecución de la consulta

Al ejecutar la consulta anteriormente mencionada, se obtuvieron los siguientes datos:

Como punto de referencia, las columnas C10 (codigoUsuario), C19 (codigoCajero) y C26(codigoUsuario1) son las de mayor relevancia ya que nos muestran los identificadores del Usuario dentro de las tablas de AdmEmpleados y SegUsuarios.

	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	
	Codigousuario	C11	C12	C13	C14	C15	C16	C17	C18	codigoCajero	C20	C21	C22	C23	C24	C25	codigoUsuario1	fechaActu
	22542	2024-04-08 21:53:08.2	2024-04-08 21:53:08.243	100050682	2024040700	326	72D1F78A-1326	22	110799	0	2024-04-07	2000.0000	2024-04-08 16:4	2024-04-08 21:53:08.1			22537	2024-04-08 21:53:08.2
	22542	2024-04-08 21:53:08.2	2024-04-08 21:53:08.243	100050682	2024040700	326	72D1F78A-1326	22	110799	0	2024-04-07	2000.0000	2024-04-08 16:4	2024-04-08 21:53:08.1			22537	2024-04-08 21:53:08.2
	22542	2024-04-08 21:53:08.2	2024-04-08 21:53:08.243	100050682	2024040700	326	72D1F78A-1326	22	110799	0	2024-04-07	2000.0000	2024-04-08 16:4	2024-04-08 21:53:08.1			22537	2024-04-08 21:53:08.2
	22542	2024-04-08 21:53:08.2	2024-04-08 21:53:08.243	100050682	2024040700	326	72D1F78A-1326	22	110799	0	2024-04-07	2000.0000	2024-04-08 16:4	2024-04-08 21:53:08.1			22537	2024-04-08 21:53:08.2
	22542	2024-04-08 21:53:08.2	2024-04-08 21:53:08.243	100050682	2024040700	326	72D1F78A-1326	22	110799	0	2024-04-07	2000.0000	2024-04-08 16:4	2024-04-08 21:53:08.1			22537	2024-04-08 21:53:08.2
	22542	2024-04-08 21:53:08.2	2024-04-08 21:53:08.243	100050682	2024040700	326	72D1F78A-1326	22	110799	0	2024-04-07	2000.0000	2024-04-08 16:4	2024-04-08 21:53:08.1			22537	2024-04-08 21:53:08.2

Al realizar la búsqueda de los identificadores correspondientes a las columnas mencionadas previamente, se determinó que pertenecen a las usuarias **María de Lourdes** y **Olivia**.

A	B	C	D	E	F	G
idUsuario	idEmpleado	idNomina	idPuesto	nombre	apellidoPatern	apellidoMaterno
	20853	20853	953	MARCELA ERANDI	RODRIGUEZ	CALDERON
	45275	45275	953	SILVIA	LEYVA	SALMERON
22542	110799	110799	951	OLIVIA	PALMA	MUÑOZ
22537	118332	118332	57	MARIA DE LOURDES	LECONA	CRUZ MANJARREZ
	167663	167663	951	MARIA ELENA	MEDINA	JAIMES

Al investigar el query ejecutado en el servicio, se encontró que los identificadores de los empleados se obtienen de la tabla venTurnosCajaReg. En esta tabla, se pueden observar dos columnas clave: codigoUsuario y codigoCajero.

- La columna codigoCajero hace referencia a la tabla admEmpleados, donde se almacenan los datos del cajero.
- La columna codigoUsuario está vinculada a la tabla SegUsuariosCat, que contiene la información de los usuarios del sistema.

A continuación, se muestra una imagen con los registros de la tabla **venTurnosCajaReg**.

- El registro subrayado en **amarillo** corresponde al corte de turno asociado al **issue actual**.
- El registro resaltado en **gris** corresponde al turno del empleado **Olivia**, utilizado para realizar una comparativa entre ambos.

A		B		C		D		E		F		G		H		I		J		K		L	
codigoTurno		codigoSucur		numeroCaja		codigoCajero		estaModoEntrena		idFechaOperacion		fechaOperacion		dotacioninici		fechaHorainicioTurno		fechaHoraFinalTurno		cajaDisponible		codigoUsuario	
72D1F78A-1516-4DB3-8F74-D17F99D31819		326		22		110799		0		20240407000326		2024-04-07		2000.0000		2024-04-08 16:46:16.480		2024-04-08 21:53:08.46.1		22537		2	
D548F9EC-56F4-EE11-B2E6-DC4A3E7B05FD		326		22		110799		0		20240406000326		2024-04-06		2000.0000		2024-04-06 14:47:48.510		2024-04-06 22:53:46.02.1		22542		2	

- En el registro resaltado en amarillo, se puede observar que están relacionados dos multifuncionales asociados al turno. La responsable de realizar el corte de turno en dicha caja es **Olivia**. Sin embargo, al realizar la reimpresión del ticket de corte, aparece como responsable **María de Lourdes**.
- En el registro resaltado en gris, se puede observar que el flujo es correcto, ya que los dos identificadores, tanto del empleado como del usuario, corresponden al mismo multifuncional, en este caso, **Olivia**.

Análisis de la tabla venCortesCab

Otro punto a resaltar es que, al analizar la tabla **venCortesCab**, donde se almacenan los encabezados de los cortes de turno, se identificó una relación con la tabla **venTurnosCajaReg** a través de la columna **codigoTurno**. Sin embargo, se encontró una discrepancia entre las columnas **idFechaOperacion** y **fechaOperacion**, ya que las fechas están desfasadas y no coinciden como corresponde.

En el registro **resaltado en negritas y subrayado en amarillo**, se muestra el **folio de corte** asociado al **issue mencionado**.

A	B	C	D	E	F	G	H	I
codigoCorte	codigoRetiro	codigoTurno	codigoSucursal	idFechaOperacion	fechaOperacion	tipoCorte	folioCorte	
A831E2DC-F7F6-EE11-B2F0-DC4A3E7B05FD	A731E2DC-F7F6-EE11-B2F0-DC4A3E7B05FD	6E97DC82-83F6-EE11-B2F0-DC4A3E7B05FD	326	20240409000326	2024-04-06	P	CT0326220000364	0
88065848-B3F6-EE11-B2F0-DC4A3E7B05FD	87095848-B3F6-EE11-B2F0-DC4A3E7B05FD	D92B5C96-8DF6-EE11-B2F0-DC4A3E7B05FD	326	20240409000326	2024-04-06	P	CT0326220000363	0
5EE880CD-8CF6-EE11-B2F0-DC4A3E7B05FD	5DE880CD-8CF6-EE11-B2F0-DC4A3E7B05FD	7BA4C0FD-99D1-40A3-8E27-A2F45119FF7	326	20240407000326	2024-04-06	P	CT0326220000362	0
17855713-88F6-EE11-B2F0-DC4A3E7B05FD	16855713-88F6-EE11-B2F0-DC4A3E7B05FD	37059F08-72F6-EE11-B2F0-DC4A3E7B05FD	326	20240409000326	2024-04-06	P	CT0326220000361	0
8A024493-2AF6-EE11-B2EE-DC4A3E7B05FD	89024493-2AF6-EE11-B2EE-DC4A3E7B05FD	7A70B80A-25F6-EE11-B2EE-DC4A3E7B05FD	326	20240409000326	2024-04-06	P	CT0326220000360	0
3FA378AC-24F6-EE11-B2EE-DC4A3E7B05FD	3EA378AC-24F6-EE11-B2EE-DC4A3E7B05FD	72D1F78A-1616-4DB3-8F74-D17F99D31819	326	20240407000326	2024-04-06	P	CT0326220000359	0
78DA688D-F9F5-EE11-B2ED-DC4A3E7B05FD	77DA688D-F9F5-EE11-B2ED-DC4A3E7B05FD	EC9FB219-7D0D-4227-B4B4-E80414231448	326	20240407000326	2024-04-06	P	CT0326220000358	0

3. Diagnóstico

Con base en el problema identificado en el reporte del ticket, donde aparecen dos multifuncionales diferentes, y tras revisar las capturas de los registros de las tablas mencionadas, se concluye que el incidente fue causado por **manipulación manual en la base de datos**, específicamente en la tabla **venTurnosCajaReg**. Aunque no se cuentan con los **logs del día en que ocurrió el issue**, el análisis del funcionamiento del servicio y de la aplicación sugiere que esta es la causa probable del problema. Esto se debe a que tanto el servicio como la aplicación asigna automáticamente el **mismo usuario** en ambas columnas (**codigoUsuario** y **codigoCajero**) al momento de **abrir un turno**. La discrepancia observada indica que hubo una modificación directa en la base de datos que alteró estos valores.

Pasos para solucionar el issue

Para resolver este problema, se deben seguir los siguientes pasos:

- 1. Verificar registros afectados:** Revisar en la tabla **venTurnosCajaReg** los registros donde los identificadores de las columnas **codigoUsuario** y **codigoCajero** no coincidan, identificando aquellos que estén incorrectos.
- 2. Corregir inconsistencias manuales:** Actualizar manualmente los registros afectados, asegurando que ambos identificadores correspondan al usuario que abrió el turno.

*El script **ActualizarUsuario_Turno** tiene como función actualizar el registro de la tabla **VenTurnosCajaReg** y solo asignar un usuario al turno en curso en dicha caja.

*El script **insertarTurno**, cumple con la función de crear el turno para dicha caja asociando el usuario, cumpliendo una serie de requisitos (fecha Operación creada, usuario registrado, etc)

- 3. Implementar controles adicionales:** Agregar validaciones o restricciones en la base de datos y en la aplicación para evitar la modificación manual de los campos **codigoUsuario** y **codigoCajero**.
- 4. Probar la funcionalidad:** Ejecutar nuevamente el proceso de reimpresión de tickets para asegurar que el problema ha sido resuelto.
- 5. Monitoreo de incidentes:** Implementar un monitoreo temporal para asegurar que el flujo se mantenga correcto y que no ocurran nuevas discrepancias.