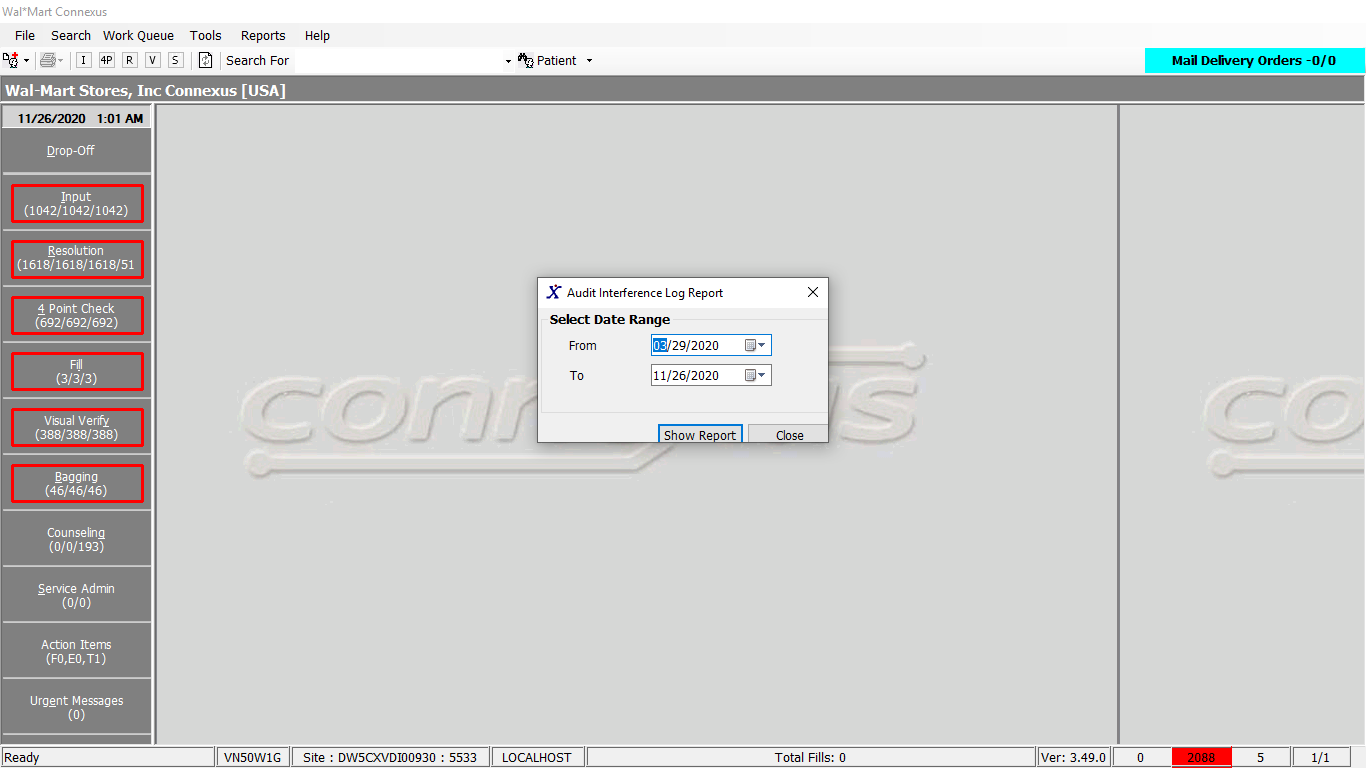
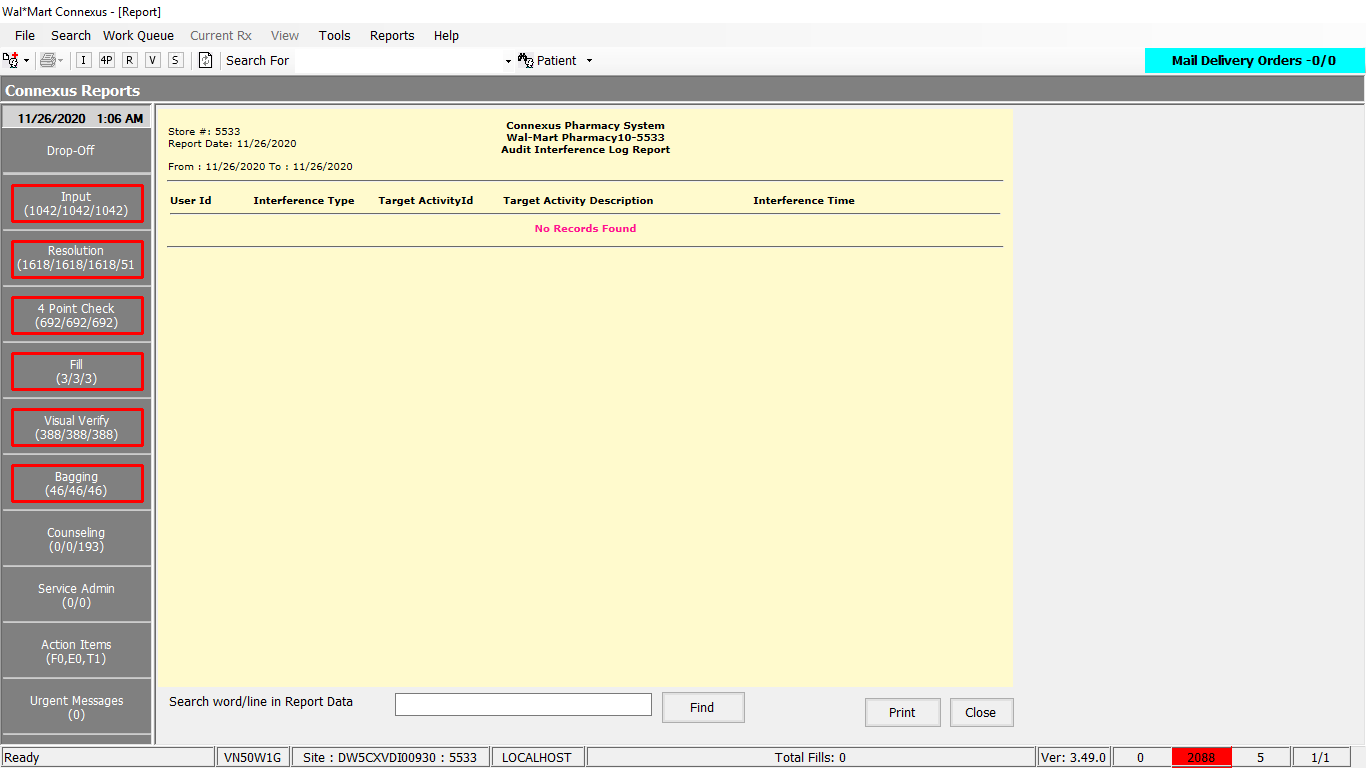
## Reports/ Audit / Audit Interference Log Report:

Request:



Response:



Queries

1. Public Const GetAuditInterferenceInfo As String = " SELECT userid,CASE WHEN task\_id = 2101 THEN 'Delete' WHEN task\_id = 2102 " &

" THEN 'Update' END modification\_type, activity\_comment, activity\_ts " &

" FROM error\_and\_audit:appl\_activity WHERE DATE(activity\_ts) BETWEEN ? AND ? AND activity\_type\_code = ? "

DAL:

Public Function GetAuditInterferenceData(ByVal siteId As Integer, ByVal startDate As Date, ByVal endDate As Date) \_

As List(Of GetAuditInterferenceInfoBE) Implements IReportDAL.GetAuditInterferenceData

Using (New Tracer(ConnexusCommonConstants.ConnexusTracer))

Dim info As List(Of GetAuditInterferenceInfoBE) = Nothing

Dim parameterlist As New List(Of DbParameter)

Dim auditSerializer As Xml.Serialization.XmlSerializer = CacheWrapperDal.AuditSerializer

Dim auditInfo As List(Of Audit.AuditEntry) = Nothing

Dim reader As IO.TextReader = Nothing

parameterlist.Add(ParameterHelper.BuildParameter("StartDate", DbType.Date, startDate))

parameterlist.Add(ParameterHelper.BuildParameter("EndDate", DbType.Date, endDate))

parameterlist.Add(ParameterHelper.BuildParameter("ActivityTypeCode", DbType.Int32, Audit.AuditActivityType.AuditInterferenceLog))

Using dr As IDataReader = Daab.ExecuteReader(DatabaseEnum.Error\_And\_Audit, siteId, ReportQueries.GetAuditInterferenceInfo, parameterlist)

While dr.Read

Dim item As New GetAuditInterferenceInfoBE

If Not dr.IsDBNull(dr.GetOrdinal("userid")) Then

item.UserId = dr.GetString(dr.GetOrdinal("userid"))

End If

If Not dr.IsDBNull(dr.GetOrdinal("modification\_type")) Then

item.InterferenceType = dr.GetString(dr.GetOrdinal("modification\_type"))

End If

'Fetching target activity id and target activity description

If Not dr.IsDBNull(dr.GetOrdinal("activity\_comment")) Then

reader = New IO.StringReader(dr.GetString(dr.GetOrdinal("activity\_comment")))

auditInfo = CType(auditSerializer.Deserialize(reader), List(Of Audit.AuditEntry))

Dim commentInfo As Audit.AuditEntry = auditInfo.Find(Function(data As Audit.AuditEntry) \_

data.Name.Equals(ServicesCommonConstants.TargetActivityId,

StringComparison.InvariantCultureIgnoreCase))

item.TargetActivityID = If(Not String.IsNullOrEmpty(commentInfo.Value), Convert.ToInt32(commentInfo.Value), 0)

commentInfo = New Audit.AuditEntry

commentInfo = auditInfo.Find(Function(data As Audit.AuditEntry) \_

data.Name.Equals(ServicesCommonConstants.TargetActivityTypeDesc,

StringComparison.InvariantCultureIgnoreCase))

item.TargetActivityDesc = If(Not String.IsNullOrEmpty(commentInfo.Value), Convert.ToString(commentInfo.Value), String.Empty)

End If

If Not dr.IsDBNull(dr.GetOrdinal("activity\_ts")) Then

item.InterferenceTS = dr.GetDateTime(dr.GetOrdinal("activity\_ts"))

End If

If info Is Nothing Then

info = New List(Of GetAuditInterferenceInfoBE)

End If

info.Add(item)

End While

End Using

Return info

End Using

End Function