

Documentação do Relatório Customizado LCOMR024 - "SC VS PC/AE VS NF"

1. Informações Gerais

- **Programa:** LCOMR024.PRW
- **Autor:** Eder SPVG
- **Data de Criação:** 19/11/2024
- **Descrição:** Este relatório tem como objetivo relacionar Solicitações de Compras (SC), Pedidos de Compra (PC), Autorizações de Entrega (AE) e Notas Fiscais (NF). Ele fornece uma visão consolidada dessas informações, facilitando a análise do processo de compras e recebimento de mercadorias.

2. Funcionalidade e Uso

- **Objetivo Principal:** O relatório permite comparar e cruzar dados de SCs, PCs/AEs e NFs, identificando informações como:
 - Itens da SC e seus correspondentes no PC/AE e NF.
 - Quantidades solicitadas, compradas e recebidas.
 - Valores envolvidos em cada etapa do processo.
 - Status dos Pedidos/AEs (aprovado, não aprovado, etc.).
- **Como Executar o Relatório:**
 - A execução do programa LCOMR024 chama a função `LCOMR024()`.
 - A função `LCOMR024()` verifica a existência da função `U_LUFT_LOG()` e a executa, caso exista (para gerar logs de utilização).
 - Em seguida, a função `AjustaPerg()` é chamada para configurar as perguntas/filtros do relatório.
 - A função `Pergunte()` exibe a tela de perguntas/filtros para o usuário.
 - A função `ReportDef()` é chamada para definir a estrutura do relatório (título, colunas, etc.).
 - O método `oReport:PrintDialog()` exibe a tela de impressão do relatório, onde o usuário pode escolher a impressora, etc.
- **Parâmetros/Filtros do Relatório:** O relatório utiliza os seguintes parâmetros (MV_PARs) para filtrar os dados:
 - MV_PAR01: De Filial
 - MV_PAR02: Até Filial
 - MV_PAR03: De Emissão (SC)
 - MV_PAR04: Até Emissão (SC)
 - MV_PAR05: De SC
 - MV_PAR06: Até SC
 - MV_PAR07: De Produto
 - MV_PAR08: Até Produto
 - MV_PAR09: De Item Cta (SC)
 - MV_PAR10: Até Item Cta (SC)
 - MV_PAR11: De Centro de Custo (SC)
 - MV_PAR12: Até Centro de Custo (SC)
 - MV_PAR13: Tipo de Documento (PC, AE ou Ambos)

- **Informações Apresentadas no Relatório:** O relatório exibe as seguintes informações:
 - Dados da Solicitação de Compra (SC): Filial, Número, Item, Produto, Descrição, Unidade de Medida, Quantidade, Quantidade já Entregue, Solicitante, Observações, Emissão, Aprovação da SC.
 - Dados do Pedido de Compra/Autorização de Entrega (PC/AE): Tipo de Documento (PC ou AE), Número, Item, Quantidade, Preço, Total, Status, Aprovador, Data de Aprovação, Observações, Emissão.
 - Dados da Nota Fiscal (NF): Número, Data de Digitação, Série, Fornecedor, Loja, Rateio, Centro de Custo, Conta Contábil, Percentual de Rateio, Item, Item Conta Contábil, Item NF, Valor Unitário Rateado, Valor Total Rateado, Valor Unitário, Valor Total, Emissão da NF.

3. Detalhes Técnicos

- **Tabelas Utilizadas:**
 - SC1: Solicitação de Compras
 - SB1: Cadastro de Produtos
 - SBM: Grupos de Produto
 - SC7: Pedidos de Compra/Autorizações de Entrega
 - SA2: Cadastro de Fornecedores
 - SD1: Notas Fiscais de Entrada
 - SDE: Rateio de Notas Fiscais
 - SCR: Aprovações
- **Principais Funções:**
 - **LCOMR024()**: Função principal do programa, responsável por controlar a execução do relatório.
 - **ReportDef()**: Função que define a estrutura do relatório (título, cabeçalho, colunas, etc.). Utiliza a classe **TReport** para criar o relatório.
 - **IMPDADOS()**: Função que contém a lógica para buscar os dados no banco de dados e alimentar o relatório. Executa a query SQL.
 - **AjustaPerg()**: Função que configura as perguntas/filtros do relatório, utilizando a função **U_AjuPerg()**.
 - **_Login()**: Função que busca o nome do usuário aprovador.

4. Query SQL do Relatório

A query SQL abaixo é responsável por buscar os dados do relatório. Ela realiza vários joins entre as tabelas SC1, SB1, SBM, SC7, SA2, SD1, SDE e SCR para relacionar as informações de Solicitação de Compra, Pedido de Compra/Autorização de Entrega e Nota Fiscal.

```
SQL
SELECT
  (SELECT CR_USER FROM SCR CRSUB2 WHERE
```

```

CR_NIVEL = (SELECT MAX(CR_NIVEL) FROM SCR CRSUB WHERE
SC7.C7_CONAPRO = 'L' AND CRSUB.CR_NUM = SC7.C7_NUM AND
CRSUB.D_E_L_E_T_ = '' AND CRSUB.CR_STATUS = '03' AND CRSUB.CR_FILIAL =
SC7.C7_FILIAL) AND CRSUB2.CR_STATUS = '03' AND CRSUB2.CR_NUM =
SC7.C7_NUM AND CRSUB2.CR_FILIAL = SC7.C7_FILIAL AND CRSUB2.D_E_L_E_T_ = ''
) AS APROVADOR,
(SELECT CR_DATA LIB FROM SCR CRSUB2 WHERE
CR_NIVEL = (SELECT MAX(CR_NIVEL) FROM SCR CRSUB WHERE
SC7.C7_CONAPRO = 'L' AND CRSUB.CR_NUM = SC7.C7_NUM AND
CRSUB.D_E_L_E_T_ = '' AND CRSUB.CR_STATUS = '03' AND CRSUB.CR_FILIAL =
SC7.C7_FILIAL) AND CRSUB2.CR_STATUS = '03' AND CRSUB2.CR_NUM =
SC7.C7_NUM AND CRSUB2.CR_FILIAL = SC7.C7_FILIAL AND CRSUB2.D_E_L_E_T_ = ''
) AS DATA_APROV,
C1_FILIAL, C7_FILIAL, D1_FILIAL,
C1_NUM, C1_ITEM, C7_NUM, C7_ITEM, C7_NUMSC, C7_ITEMSC,
C7_EMISSAO, C7_FORNECE, C7_LOJA, A2_NOME,
D1_DOC, D1_DTDIGIT, D1_SERIE, D1_FORNECE, D1_LOJA, C7_RESIDUO,
B1_GRUPO,
BM_DESC, C1_PRODUTO, C1_DESCRI, B1_ESPECIF, C1_UM,
C1_SEGUM, C1_EMISSAO, C1_PEDIDO, C1_ITEMPED, C1_SOLICIT,
C1_QUJE, C1_QUANT, C7_QUANT,
C7_PRECO, C7_TOTAL, D1_VUNIT, D1_TOTAL,
C7_CC, C7_ITEMCTA, C7_CONTA,
D1_CC, D1_ITEMCTA, D1_CONTA, C7_TIPO,
CASE
WHEN C7_RATEIO = '1' THEN 'SIM'
WHEN C7_RATEIO = '2' THEN 'NAO'
ELSE C7_RATEIO
END C7_RATEIO,
CASE
WHEN D1_RATEIO = '1' THEN 'SIM'
WHEN D1_RATEIO = '2' THEN 'NAO'
ELSE D1_RATEIO
END D1_RATEIO, D1_VUNIT, D1_TOTAL, DE_CC, DE_CONTA, DE_DOC, DE_FILIAL,
DE_PERC, DE_FORNECE, DE_LOJA,
DE_ITEM, DE_ITEMCTA, DE_ITEMNF, DE_LOJA, DE_SERIE, (D1_VUNIT *
(DE_PERC/100)) AS VAL_UNIT_RAT,
(D1_TOTAL * (DE_PERC/100)) AS VAL_TOT_RAT,
CASE
WHEN C7_CONAPRO = 'B' THEN 'NÃO APROVADO'
WHEN C7_CONAPRO = 'L' THEN 'APROVADO'
WHEN C7_CONAPRO = 'R' THEN 'ELIMINADO POR RESIDUO'
END C7_CONAPRO, C1_SOLICIT, C1_OBS, C7_OBS, C1_EMISSAO, C7_EMISSAO,
D1_EMISSAO, C1_DTAPRO
FROM SC1
LEFT JOIN SB1 ON B1_COD = C1_PRODUTO AND SB1.D_E_L_E_T_ = ''
LEFT JOIN SBM ON B1_GRUPO = BM_GRUPO AND SBM.D_E_L_E_T_ = ''

```

```

LEFT JOIN SC7 ON C7_NUMSC = C1_NUM AND C7_ITEMSC = C1_ITEM AND
C7_FILIAL = C1_FILIAL AND SC7.D_E_L_E_T_ = ''
LEFT JOIN SA2 ON A2_COD = C7_FORNECE AND A2_LOJA = C7_LOJA AND
SA2.D_E_L_E_T_ = ''
LEFT JOIN SD1 ON D1_PEDIDO = C7_NUM AND D1_ITEMPC = C7_ITEM AND
C7_FILIAL = D1_FILIAL AND SD1.D_E_L_E_T_ = ''
LEFT JOIN SDE ON DE_DOC = D1_DOC AND DE_SERIE = D1_SERIE AND DE_FILIAL =
D1_FILIAL AND DE_FORNECE = D1_FORNECE AND DE_LOJA = D1_LOJA AND
DE_ITEMNF = D1_ITEM AND SDE.D_E_L_E_T_ = ''
//LEFT JOIN SCR SCR ON CR_FILIAL = C7_FILIAL AND C7_NUM = CR_NUM AND
SCR.D_E_L_E_T_ = ''
WHERE
    SC1.D_E_L_E_T_ = ''
    AND C1_FILIAL >= MV_PAR01
    AND C1_FILIAL <= MV_PAR02
    AND C1_EMISSAO >= MV_PAR03
    AND C1_EMISSAO <= MV_PAR04
    AND C1_NUM >= MV_PAR05
    AND C1_NUM <= MV_PAR06
    AND C1_PRODUTO >= MV_PAR07
    AND C1_PRODUTO <= MV_PAR08
    AND C1_ITEMCTA >= MV_PAR09
    AND C1_ITEMCTA <= MV_PAR10
    AND C1_CC >= MV_PAR11
    AND C1_CC <= MV_PAR12
    AND C7_TIPO != c_tipo
//AND C7_NUM = '939089'
ORDER BY C1_FILIAL, C1_NUM

```

5. Considerações para Alterações na Query

Esta seção visa orientar sobre as possibilidades e limitações de alterações na query, para auxiliar em eventuais customizações do relatório.

- **Possibilidades de Alteração:**

- **Inclusão de Novos Campos:** É possível incluir novos campos no **SELECT**, desde que as colunas existam nas tabelas utilizadas na query. Se o campo for de uma tabela ainda não incluída, será necessário adicionar um **JOIN** para essa tabela.
- **Alteração de Filtros (WHERE):** Os filtros da cláusula **WHERE** podem ser modificados ou adicionados para restringir ou expandir os dados retornados. É possível adicionar condições com outras colunas, utilizar outros operadores lógicos (OR, IN, etc.) e usar outras funções SQL.
- **Alteração da Ordenação (ORDER BY):** A ordenação dos dados pode ser alterada modificando a cláusula **ORDER BY**. É possível ordenar por uma ou mais colunas, em ordem crescente (ASC) ou decrescente (DESC).

- **Uso de Funções SQL:** Funções SQL como `SUM()`, `AVG()`, `COUNT()`, `MAX()`, `MIN()`, `GROUP BY`, etc., podem ser utilizadas para realizar cálculos e agrupamentos nos dados.
- **Subqueries:** Subqueries (queries dentro de outras queries) podem ser utilizadas para buscar dados relacionados ou aplicar filtros mais complexos. A query já utiliza subqueries para buscar o aprovador e a data de aprovação.
- **Limitações e Cuidados:**
 - **Performance:** Alterações que aumentam a complexidade da query (joins complexos, subqueries excessivas, funções que consomem muitos recursos) podem impactar negativamente a performance do relatório. É importante testar o desempenho após qualquer alteração.
 - **Integridade dos Dados:** É fundamental garantir que as alterações na query mantenham a integridade dos dados, ou seja, que os dados retornados sejam corretos e consistentes com as regras de negócio do sistema.
 - **Impacto em Outras Rotinas:** Alterações nas tabelas e campos utilizados pela query podem impactar outras rotinas do sistema que utilizam as mesmas tabelas. É importante analisar o impacto antes de realizar qualquer alteração.
 - **Complexidade:** Queries muito complexas podem ser difíceis de entender e manter. É recomendável manter a query o mais simples e clara possível, utilizando comentários para explicar a lógica.
 - **Padrões do Protheus:** É importante seguir os padrões de nomenclatura e codificação do Protheus ao realizar alterações na query, para garantir a compatibilidade com o sistema.
 - **Testes:** É fundamental realizar testes completos após qualquer alteração na query, para garantir que o relatório continue funcionando corretamente e que os dados retornados estejam corretos.

6. Exemplos de Consultas RAG

Aqui estão alguns exemplos de perguntas que um sistema RAG poderia responder utilizando esta documentação:

- **Pergunta:** "Quais os parâmetros para filtrar o relatório LCOMR024?"
 - **Resposta:** "Os parâmetros para filtrar o relatório LCOMR024 são: De Filial (MV_PAR01), Até Filial (MV_PAR02), De Emissão (MV_PAR03), Até Emissão (MV_PAR04), De SC (MV_PAR05), Até SC (MV_PAR06), De Produto (MV_PAR07), Até Produto (MV_PAR08), De Item Cta (MV_PAR09), Até Item Cta (MV_PAR10), De Centro de Custo (MV_PAR11), Até Centro de Custo (MV_PAR12) e Tipo de Documento (MV_PAR13)."
- **Pergunta:** "Quais tabelas são utilizadas no relatório LCOMR024?"
 - **Resposta:** "As tabelas utilizadas no relatório LCOMR024 são: SC1 (Solicitação de Compras), SB1 (Cadastro de Produtos), SBM (Grupos de Produto), SC7 (Pedidos de Compra/Autorizações de Entrega), SA2 (Cadastro de Fornecedores), SD1 (Notas Fiscais de Entrada), SDE (Rateio de Notas Fiscais) e SCR (Aprovações)."
- **Pergunta:** "Como posso adicionar o nome do fornecedor na saída do relatório?"

- **Resposta:** "O nome do fornecedor já é exibido no relatório, proveniente do campo A2_NOME da tabela SA2 (Cadastro de Fornecedores). Caso precise de alguma informação adicional sobre o fornecedor, você pode consultar outros campos desta tabela."
- **Pergunta:** "É possível filtrar o relatório por data de aprovação do pedido de compra?"
 - **Resposta:** "Sim, é possível filtrar o relatório por data de aprovação do pedido de compra. A query SQL utiliza o campo CR_DATALIB da tabela SCR (Aprovações) para buscar essa informação. Você pode adicionar condições de filtro na cláusula WHERE da query, utilizando esse campo."
- **Pergunta:** "Quais são os status possíveis para o pedido de compra?"
 - **Resposta:** "Os status possíveis para o pedido de compra são: 'APROVADO' (quando C7_CONAPRO = 'B'), 'APROVADO' (quando C7_CONAPRO = 'L') e 'ELIMINADO POR RESIDUO' (quando C7_CONAPRO = 'R')."
- **Pergunta:** "A query já traz o nome do aprovador do PC/AE?"
 - **Resposta:** "Sim, a query já busca o nome do aprovador do PC/AE. Ela utiliza uma subquery para selecionar o usuário (CR_USER) da tabela SCR (Aprovações) que aprovou o PC/AE."