

Implementação ABAP da ferramenta Master Contract para gestão especializada de contractos/ordens de venda na GFI France

Mateus Henrique Soares Fonseca

Relatório de projeto apresentado à Escola Superior de Tecnologia e Gestão para obtenção do
grau de mestre em Informática

Orientadores:

Prof. João Paulo Ribeiro Pereira

Bragança

Outubro 2024

Resumo

O seguinte relatório documenta o projeto desenvolvido no âmbito da dissertação do mestrado em Informática.

No contexto existente, apenas era permitido criar contratos independentes via transações standard, sem que estes partilhassem qualquer ligação para além de herança de dados e/ou na criação por cópia. Permitir que estes pudessem ser agregados num contrato mestre (com as devidas restrições relativas a códigos de empresa, centros de custos, parceiros/vendedores, etc...) facilitaria e agilizaria o monitoramento e gestão dos projetos em termos de recursos, calendarização, eixos de afetação...

O trabalho desenvolvido teve como objetivo desenvolver uma implementação de funcionalidades extra ao standard da SAP, nomeadamente a nova ferramenta custom da GFI France, o Master Contract. Esta permite criar, modificar e visualizar Master Contracts, bem como assigná-los a contractos via transações standard (VA01, VA02, VA03, VA41, VA42 e VA43). Foi também possibilitada a criação e assinatura em massa via ficheiro Excel e criar Master Contracts via webservice, baseado em SOAP.

Numa primeira fase, foi necessário construir todo *backbone*, isto é, domínios, elementos de dados, estruturas e tabelas bem como implementar os novos dados nas tabelas standard já existentes, VBAK (relativa ao cabeçalho das ordens de venda/contratos) e VBAP (relativa aos itens das ordens de venda/contratos). Na segunda fase, construir e desenvolver o programas custom para a criação, manutenção e visualização de Master Contracts, o programa para a criação em massa bem como outro para a assinatura em massa, criar o webservice para a criação dum Master Contract, e por fim implementar as transações standard para permitir a assinatura manual do Master Contract. Na terceira fase foi necessário aplicar o uso de autorizações através de roles assignadas a utilizadores e implementar rotinas standard da VOFM, no caso específico da transferência de dados na criação de contratos por cópia.

Finalmente, na última fase, foi necessário aplicar uma extensa sessão de teste, para verificar a funcionalidade do novo desenvolvimento e assegurar a consistência dos dados.

Palavras-chave: SAP, Master Contract, Webservice, VOFM, Contractos, Ordens de Venda, VBAP, VBAK

Abstract

The following report documents the project developed as part of the master's dissertation in Computer Science.

In the existing context, it was only possible to create independent contracts via standard transactions, without them sharing any connection other than data inheritance and/or creation by copy. Allowing these to be aggregated into a master contract (with the appropriate restrictions regarding company codes, cost centers, partners/vendors, etc...) would facilitate and speed up the monitoring and management of projects in terms of resources, scheduling, allocation axes ...

The work carried out aimed to develop an implementation of extra functionalities to the SAP standard, in particular the new custom tool from GFI France, the Master Contract. This allows us to create, modify and view Master Contracts, as well as assign them to contracts via standard transactions (VA01, VA02, VA03, VA41, VA42 and VA43). It was also possible to create and assign in bulk via Excel file and create Master Contracts via webservice, based on SOAP.

In the first phase, it was necessary to build the entire backbone, that is, domains, data elements, structures and tables, as well as implement the new data in the existing standard tables, VBAK (related to the header of sales orders/contracts) and VBAP (related to sales order/contract items). In the second phase, build and develop the custom programs for the creation, maintenance and visualization of Master Contracts, the program for mass creation as well as another for mass assigning, create the webservice for creating a Master Contract, and finally implement standard transactions to allow manual assigning of the Master Contract. In the third phase, it was necessary to apply the use of authorizations through roles assigned to users and implement standard VOFM routines, in the specific case of data transfer when creating contracts by copy.

Finally, in the last phase, it was necessary to carry out an extensive test session to verify the functionality of the new development and ensure data consistency.

Keywords: SAP, Master Contract, Webservice, VOFM, Contracts, Sales Orders, VBAP, VBAK

Índice Geral

Resumo	iii
Abstract.....	v
Índice Geral	ix
Lista de Siglas/Abreviaturas.....	xiii
Índice de Figuras	xv
Índice de Tabelas	xix
Índice de Listagens	xxi
1. Introdução.....	1
1.1. Enquadramento.....	2
1.2. Objetivos.....	2
1.3. Estrutura do Documento	3
2. Problema/Contextualização.....	5
2.1. Introdução.....	5
2.2. Estado da Arte	5
2.2.1. Transações VA01/02/03/41/42/43.....	6
2.2.2. Transação SE38	7
2.2.3. Transação SE37	8
2.2.4. Transação SE11	8
2.2.5. Transação VOFM	9
3. Revisão Bibliográfica	11
3.1. SAP.....	11
3.1.1. Arquitetura e Módulos do SAP	11
3.1.2. Benefícios da Implementação do SAP	11
3.1.3. Desafios da Implementação do SAP	12
3.1.4. O Futuro do SAP e ERP	12
3.2. ERP SAP	12
3.2.1. ERP SAP: Benefícios e Desafios	12
3.2.2. O Futuro do ERP SAP	13
3.3. SAP GUI.....	13
3.3.1. Conceito e Importância.....	13
3.3.2. Benefícios e Desafios	14
3.3.3. Futuro do SAP GUI.....	14
3.3.4. Transações de SAP GUI utilizadas no âmbito do projeto	15
4. Ferramentas/Tecnologias.....	17

4.1.	SAP ABAP	17
5.	Arquitetura e Modelação do Sistema	21
5.1.	Requisitos Funcionais.....	21
5.1.1.	Criar, modificar e visualizar um ou mais Master Contract.....	21
5.1.2.	Manter as tabelas de Eixos de Afetação	22
5.1.3.	Criar um Master Contract via Webservice baseado em SOAP	22
5.1.4.	Criar Master Contracts em massa através dum ficheiro EXCEL	22
5.1.5.	Integrar a nova ferramenta do Master Contract nas Transações Standard SAP para as Ordens de Venda/Contratos	23
5.1.6.	Assinação em massa de Master Contracts a Ordens de Venda/Contratos Standard	23
5.2.	Requisitos Não-Funcionais.....	23
5.2.1.	Requisitos Gerais	23
5.2.2.	Tecnologias.....	24
5.3.	Diagrama de Classes.....	25
5.4.	Diagrama de Atividades	27
5.5.	Diagrama de Casos de Uso.....	31
6.	Desenvolvimento	33
6.1.	Implementação dos Objetos do Dicionário de Dados	33
6.1.1.	Tabela	33
6.1.2.	Estrutura	34
6.2.	Desenvolvimento programa principal Master Contract.....	37
6.2.1.	Report ZS4FR_MASTER_CONTRACT	37
6.2.2.	Include ZS4FR_MASTER_CONTRACT_DAT.....	40
6.2.3.	Inlcude ZS4FR_MASTER_CONTRACT_SCR	40
6.2.4.	Restantes Includes	43
6.3.	Adaptação das Transações Standard.....	51
6.4.	Implementação de Rotinas Standard VOFM.....	53
6.5.	Programa de criação em massa de Master Contracts	54
6.6.	Programa de assinação em massa de Master Contracts.....	60
6.7.	Desenvolvimento do Módulo de Função ZS4FR_WS_MASTER_CONTRACT que alimentará o Webservice	63
7.	Análise e Discussão de Resultados.....	67
7.1.	Introdução	67
7.2.	Teste do Programa principal do Master Contract ZS4FR_MASTER_CONTRACT	67
7.2.1.	Criação manual dum Master Contract	68
7.2.2.	Funcionalidade de Visualizar	68

7.2.3.	Funcionalidade de Modificar.....	70
7.2.4.	Teste das tabelas dos eixos.....	72
7.3.	Teste dos novos campos num contrato standard	74
7.3.1.	A nível de cabeçalho.....	74
7.3.2.	A nível de item	75
7.4.	Teste das rotinas VOFM para criação por referência.....	76
7.5.	Testo do Módulo de Função para criação.....	77
7.6.	Teste da Criação em massa.....	79
7.7.	Teste da Assinação em Massa	82
7.8.	Proposta de trabalho futuro.....	84
8.	Conclusões.....	87
Anexos	91

Lista de Siglas/Abreviaturas

SAP *Systems, Applications, and Products*

ABAP *Advanced Business Application Programming*

MC *Master Contract*

ERP *Enterprise Resource Planning*

GUI *Graphical User Interface*

SOAP *Simple Object Access Protocol*

XLSX *eXcel Spreadsheet eXtensible Markup Language*

ALV *ABAP List Viewer*

BAPI *Business Application Programming Interface*

Índice de Figuras

Figura 1 – Ecrã inicial transação VA41 (transversal ás restantes transações acima mencionadas).....	7
Figura 2 – Ecrã inicial transação SE38.....	7
Figura 3 - Ecrã inicial transação SE37	8
Figura 4 - Ecrã inicial transação SE11	9
Figura 5 - Ecrã inicial transação VOFM	10
Figura 6 – Módulos de SAP	13
Figura 7 – SAP GUI Easy Access	14
Figura 8 – Logotipo SAP.....	17
Figura 9 - Diagrama de Classes.....	26
Figura 10 - Diagrama de Atividades: Visualizar um Master Contract.....	27
Figura 11 - Diagrama de Atividades: Criar um Master Contract	28
Figura 12 - Diagrama de Atividades: Modificar um Master Contract	29
Figura 13 - Diagrama de Atividades: Assignar um Master Contract a um contrato standard.....	30
Figura 14 - Diagrama de Casos de Uso	31
Figura 15 - Tabela ZS4FR_MAST_CONTR	34
Figura 16 - Tabela VBAK do Cabeçalho da Ordem de Venda	35
Figura 17 - Tabela VBAP do Item da Ordem de Venda	35
Figura 18 - Estrutura Master Contract na tabela VBAK	36
Figura 19 - Estrutura Master Contract na tabela VBAB	36
Figura 20 - Report ZS4FR_MASTER_CONTRACT	38
Figura 21 - Transação ZS4_MASTER_CONTRACT	39
Figura 22 - Role ZS4FR_MASTER_CONTRACT_ROLE para o acesso á transação ZS4_MASTER_CONTRACT.....	39
Figura 23 - Include ZS4FR_MASTER_CONTRACT_DAT	40
Figura 24 - Objeto de Autorização Z_ADMIN_MC	41
Figura 25 - Role com autorização para alterar as tabelas de eixo	41
Figura 26 - Ecrã de Seleção da transação ZS4_MASTER_CONTRACT.....	42
Figura 27 - Tab do ecrã de seleção para manutenção das tabelas de eixos	42
Figura 28 - Método trata_fieldcatalog	44
Figura 29 - Método trata_layout.....	44
Figura 30 - Chamada da função para visualização do ALV	46
Figura 31 - Visualização do ALV	46
Figura 32 - Ecrã 0300 para visualização do Master Contract.....	47
Figura 33 - Objeto de Autorização Z_MAINT_MC	48
Figura 34 - Role ZMCS4_MC_AMOUNTS	48
Figura 35 - Ecrã 0500 para modificação dum Master Contract	49
Figura 36 - Ecrã 0400 para criação dum Master Contract.....	50
Figura 37 - Ecrã 8309 para o Master Contract a nível de cabeçalho.....	51
Figura 38 - Ecrã 8459 para o Master Contract a nível de item.....	51
Figura 39 - Atualização da flag para subemter alterações á tabela standard.....	52
Figura 40 - Alinhamento das tabelas standard	52
Figura 41 - Exemplo de implementação de rotina VOFM (051)	54
Figura 42 - Melhoria de Implementação que guarda as alterações às rotinas da VOFM54	
Figura 43 - Uso do módulo de função 'ALSM_EXCEL_TO_INTERNAL_TABLE' ...	55

Figura 44 - Objetos de Log criados na SLG0.....	56
Figura 45 - Inicialização do objeto de Log.....	56
Figura 46 - Geração do ficheiro Excel com entradas de erro	57
Figura 47 - Módulo de função 'BAL_LOG_MSG_ADD'	57
Figura 48 - Módulo de função 'BAL_DB_SAVE'.....	58
Figura 49 - Layout do ecrã 0100	58
Figura 50 - Módulo de função 'BAL_DSP_LOG_DISPLAY'	59
Figura 51 - ALV de sucesso	59
Figura 52 - ALV Hierárquico em Árvore de Erros	59
Figura 53 - Objeto de log para o programa de assinatura em massa.....	60
Figura 54 - Exemplo de preenchimento do cabeçalho da ordem de venda/contrato	61
Figura 55 - Exemplo de preenchimento do/s item/s da ordem de venda/contrato	62
Figura 56 - Uso do Módulo de Função 'BAPI_SALESORDER_CHANGE'	62
Figura 57 - Módulo de Função ZS4FR_WS_MASTER_CONTRACT	64
Figura 58 - Criação do Webservice	64
Figura 59 - Webservice ZS4FR_WS_MASTER_CONTRACT	64
Figura 60 - Definições da binding do webservice via SOAMANAGER	65
Figura 61 - Teste da função de criar um Master Contract	68
Figura 62 - Teste da função de visualizar vários Master Contract	69
Figura 63 - Teste da função de visualizar um Master Contract.....	69
Figura 64 - Continuação do teste da função de visualizar um Master Contract.....	70
Figura 65 - Teste da função de modificar um Master Contract.....	70
Figura 66 - Continuação do teste da função de modificar um Master Contract.....	71
Figura 67 - Registo Modificado	72
Figura 68 - Registo Modificado	72
Figura 69 - Registos da tabela "Offer Family" antes da nova adição.....	73
Figura 70 - Visão de inserção na tabela.....	73
Figura 71 - Registos do Eixo Offer Family atualizados	73
Figura 72 - Contrato 41107084 antes da assinatura do MC 1000002434.....	74
Figura 73 - Contrato 41107084 após a assinatura do MC 1000002434.....	74
Figura 74 - Tabela VBAK após assignação do contrato mestre	75
Figura 75 - Campo Offer alterado	75
Figura 76 - Tabela do Item VBAP com o campo Offer modificado e mesmo número de MC.....	76
Figura 77 - Criação dum contrato por referência do 41107084 com MC já atribuído ...	76
Figura 78 - Informações do MC transferidas com sucesso	76
Figura 79 - Payload usado no exemplo de teste para o Webservice.....	77
Figura 80 - Retorno de Erro.....	77
Figura 81 - Mensagem explicativa do erro	77
Figura 82 - Execução bem-sucedida do webservice.....	78
Figura 83 - Visualização do Master Contract criado via webservice	78
Figura 84 - Visualização do Master Contract criado via webservice	78
Figura 85 - Seleção do ficheiro a usar na criação em massa	79
Figura 86 - Resultado da Criação em massa	79
Figura 87 - Log geral após pressionar " Show Log Results".....	80
Figura 88 - Log de sucesso obtido após pressionar o botão "Show Success Results" ...	80
Figura 89 - Log de erros obtido após pressionar o botão "Show Error Results"	80
Figura 90 - Ficheiro de erro corretamente criado	80
Figura 91 - Log disponível na SLG1	81
Figura 92 - Pesquisa de apenas 1 Master Contract para ser visualizado	81

Figura 93 - Visualização do Master Contract criado pelo programa de criação em massa	81
Figura 94 - Visualização do Master Contract criado pelo programa de criação em massa	82
Figura 95 - Contrato 41107084 antes de executar o programa de assinatura em massa.	82
Figura 96 - Contrato 41107083 antes de executar o programa de assinatura em massa.	83
Figura 97 - Seleção do ficheiro a ser usado para a Assinação em Massa.....	83
Figura 98 - Log do resultado da execução do programa de assinatura em massa.....	83
Figura 99 - Master Contract 1000002434 assignado ao contrato 41107083.....	84
Figura 100 - Master Contract 1000002435 assignado ao contrato 41107084.....	84

Índice de Tabelas

Tabela 1 - Transações usadas no âmbito do projeto.....	15
--	----

Índice de Listagens

Listagem 1 - Código Authority-Check	42
---	----

1. Introdução

No contexto de gestão empresarial de empresas, o SAP (Systems, Applications, and Products) é uma das ferramentas mais robustas e amplamente utilizadas para a integração de processos de negócio. Entre as diversas funcionalidades, destacam-se para o contexto do projeto que me proponho a desenvolver, as transações VA01/02/03 e VA41/42/43, voltadas para a criação, modificação e visualização de ordens de venda e contratos, respetivamente.

Normalmente, um utilizador/cliente pode proceder á criação, gestão ou visualização duma ordem de venda/contrato pelas transações acima descritas, fornecendo para isso todas as informações necessárias com, por exemplo, vírgula, Empresa, Cliente, Fornecedor, Divisão, Data de Criação, Data de Entrega, Referência Descritiva, Item, Quantidade de Item, Preço Unitário, Unidade de Medida, Moeda... Estas informações e muitas outras permitem uma melhor gestão de stock, cumprimentos de prazos, eficiência operacional, entre outras.

Embora o processo *standard* seja bastante robusto e eficaz, a empresa da GFI França entendeu que este ainda carece de maior controlo e monitorização para entregar maior eficácia, rapidez e robustez nos produtos vendidos aos seus clientes. A empresa percebeu que precisava de poder agregar vários contratos *standard* num contrato mestre (obedecendo a critérios essenciais como manter a coerência entre dados, nomeadamente códigos de empresa, centros de lucro, fornecedores...) e assim ter uma visão global do fluxo de vendas, orçamentos, prazos de entrega.

Para tal, for requerido a criação de uma ferramenta *custom* que permita a criação de tais contratos mestre, Master Contracts, assim como a sua manutenção e visualização. Foi também pedido que essa mesma informação esteja disponível nos contratos standard, tanto ao nível de cabeçalho como a nível de item.

1.1. Enquadramento

Na ferramenta SAP, sempre que um cliente deseja criar uma ordem de venda/contrato, este recorre às funcionalidades standard já disponíveis para o efeito. Para tal preenche os dados necessários do cabeçalho e do item e depois procede á criação.

Tendo em conto o processo descrito, podemos concluir que embora diferentes contratos possam partilhar o mesmo cliente, código de empresa, fornecedor, centro de custo, ainda assim não possuem uma ligação direta que lhes permita triangular dados, como por exemplo, orçamentos, prazos, eixos de afetação do negócio, custos totais revistos e outros parâmetros que o negócio considera crucial.

Por esse motivo foi necessário desenvolver uma funcionalidade custom para agregar diferentes contratos existentes num contrato mestre, o Master Contract. Este pode ser assignado á vários contratos na ferramenta standard (VA01/02/41/42), bem como via programas custom desenvolvidos para o efeito, tornando este processo mais robusto, célere e eficaz.

1.2. Objetivos

O objetivo deste projeto teve como principal foco o desenvolvimento duma ferramenta que pudesse agregar vários contratos standard num contrato custom, bem como assegurar a consistência da base de dados standards com a introdução de novos dados na mesma. De uma forma mais sintetizada, foi necessário atingir os seguintes pontos:

- Desenvolver uma aplicação no ERP de SAP para a criação, modificação e visualização de contratos mestre, bem como customização de dados das tabelas de eixos de afetação;
- Desenvolver uma aplicação em ERP de SAP para a criação de Master Contracts em massa através dum ficheiro EXCEL;
- Desenvolver uma aplicação em ERP de SAP para a assinatura de Master Contracts a contratos standard em massa através dum ficheiro EXCEL;
- Criação dum Webservice em SOAP para ser consumido pelo sistema de modo a criar um contrato mestre (a ser utilizado pela equipa de CRM);
- Implementar a ferramenta e ecrãs standard de criação, gestão e visualização de ordens de venda/contratos standard para poderem modificar e/ou visualizar o contrato mestre assignado.
- Implementar rotinas VOFM standard de modo a garantir a coesão dos dados na base de dados;
- Criar os elementos do dicionário de dados necessários bem como modificar as estruturas e tabelas standard já existentes;

1.3. Estrutura do Documento

De forma a facilitar a leitura deste relatório e de demonstrar o trabalho realizado, este documento foi dividido em oito capítulos.

- Capítulo 1 – Introdução

No primeiro capítulo, é feita uma introdução e apresentado um enquadramento ao tema abordado, descrevendo os objetivos definidos para a sua implementação.

- Capítulo 2 – Problema/Contextualização

Neste capítulo, faz-se uma breve contextualização do problema apresentado, juntamente com a revisão do estado da arte.

- Capítulo 3 – Revisão Bibliográfica

No terceiro capítulo, são discutidos os conceitos teóricos relevantes para este projeto.

- Capítulo 4 – Ferramentas/Tecnologias

Neste abordamos as ferramentas optadas para serem usadas e as tecnologias usadas no desenvolvimento deste projeto.

- Capítulo 5 – Arquitetura e Modelação do Sistema

O quinto capítulo trata da arquitetura e modelação. Assim sendo, são apresentados os diagramas Unified Modeling Language (UML), exemplificando com mais clareza e detalhe o funcionamento da funcionalidade.

- Capítulo 6 – Desenvolvimento/Implementação

Neste capítulo abordamos a solução desenvolvida, relatando os programas, webservice, elementos de dados relevantes, alterações/implementações feitas no standard, etc...

- Capítulo 7 - Análise e Discussão de Resultados

No sétimo capítulo, faz-se uma análise de todo o trabalho realizado, destacando os resultados obtidos e as dificuldades que foram ultrapassadas.

- Capítulo 8 – Conclusão

No último capítulo, é apresentada a conclusão, que oferece uma visão geral do trabalho desenvolvido e discute os resultados alcançado.

2. Problema/Contextualização

2.1. Introdução

Como já referido no capítulo anterior, sempre que é pretendido criar uma ordem de venda/contrato é necessária a utilização das transações standard V01 ou VA41 para a criação. Após preencher os dados obrigatórios tanto a nível de cabeçalho como a nível de item, procede-se á gravação. Este processo grava então a nova/o ordem de venda/contrato na base de dados standard.

Infelizmente, mesmo partilhando dados chave entre si não é possível agregar as mesmas num contrato geral, num contrato mestre.

Dada esta necessidade de agrupar vários contratos/ordens de venda num contrato mestre, este projeto surge na ambição de suprimir essa necessidade, promovendo uma maior rigor e controlo de orçamentos e prazos, promovendo por isso mais eficiência e robustez nos processos do negócio

2.2. Estado da Arte

De forma a implementar uma solução robusta, eficiente e eficaz realizou-se uma pesquisa sobre como a informação era processada na criação de ordens de venda/contratos, quais os programas standard usados em background e quais transações são cruciais para tal.

Como resultado dessa investigação foi possível averiguar que para a criação, modificações e visualização das ordens de venda/contratos são usadas, respectivamente as:

- VA01 e/ou VA41 para criação de ordens de venda e contratos;
- VA02 e/ou VA42 para modificação de ordens de venda e contratos;
- VA03 e/ou VA43 para visualização de ordens de venda e contratos;
- SE38 para criar, modificar e visualizar os programas específicos;
- SE37 para criar, modificar e visualizar os módulos de função específicos, e posteriormente os webservice;
- SE11 para criar, modificar e visualizar elementos do dicionário de dados;
- VOFM para criação, modificação e visualização de rotinas standard de processamento;

2.2.1. Transações VA01/02/03/41/42/43

As transações VA01/02/03 são utilizadas para a criação, modificação e visualização de uma ordem de venda. São transações fundamentais no módulo SAP SD (Sales and Distribution), sendo utilizadas para registrar pedidos de clientes, incluindo os detalhes sobre os produtos ou serviços, quantidades, preços e prazos de entrega. A criação de uma ordem de venda envolve várias etapas, incluindo a verificação da disponibilidade do stock e o cálculo do preço final com base nas condições definidas no sistema.

As transações VA41/42/43 estão relacionadas com a VA01/02/03, porém são utilizadas para a criação, modificação e visualização de acordos-quadro de vendas. Um acordo-quadro é um contrato a longo prazo entre a empresa e o cliente, que estabelece os termos para futuras ordens de venda.

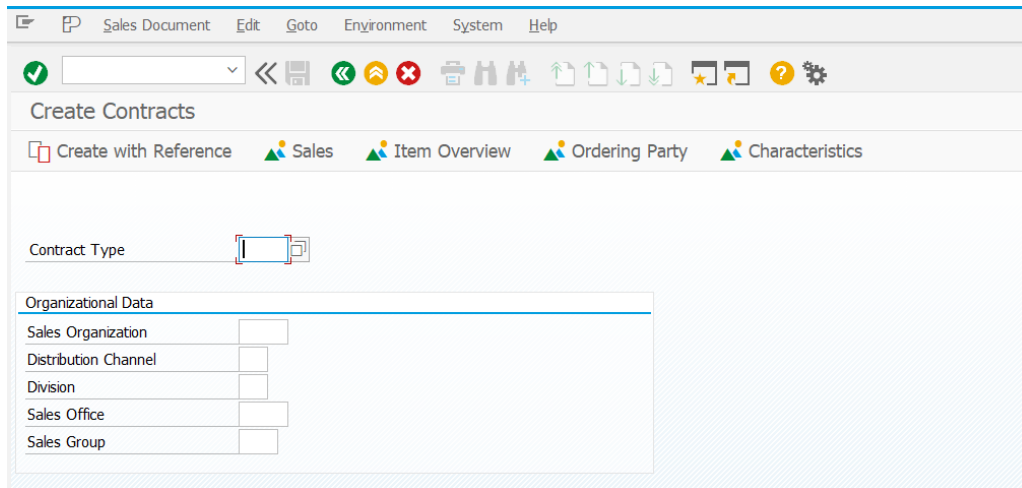


Figura 1 – Ecrã inicial transação VA41 (transversal ás restantes transações acima mencionadas)

2.2.2. Transação SE38

A transação SE38 permite a gestão de programas ABAP no SAP. A linguagem ABAP (Advanced Business Application Programming) é utilizada para o desenvolvimento de aplicações dentro do sistema SAP. Com a transação SE38, os utilizadores podem criar, modificar, visualizar e executar programas ABAP. Esta transação é crucial para desenvolvedores que trabalham com personalizações e automatizações no sistema, permitindo a criação de relatórios, interfaces e outros desenvolvimentos à medida das necessidades do negócio.

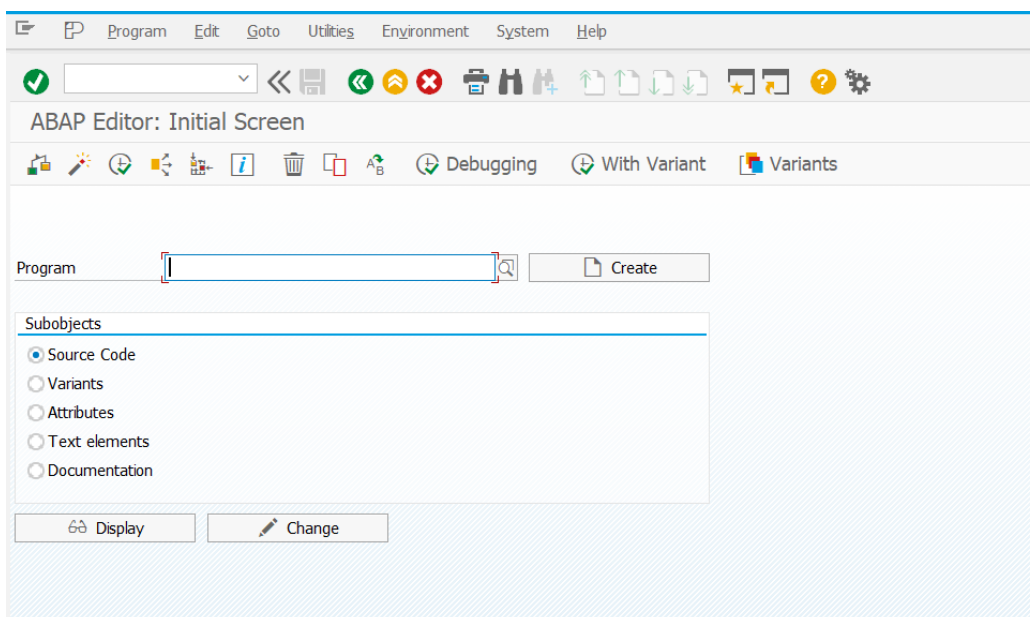


Figura 2 – Ecrã inicial transação SE38

2.2.3. Transação SE37

A transação SE37 é usada para a gestão de módulos de função ABAP. Um módulo de função é um conjunto de instruções que podem ser reutilizadas em diferentes programas ou aplicações dentro do sistema SAP. A SE37 permite criar, modificar, visualizar e testar esses módulos de função. Esta transação é essencial para a reutilização de código e para o desenvolvimento de funcionalidades que possam ser partilhadas entre diferentes programas ABAP, aumentando a eficiência no desenvolvimento. Posteriormente, podemos criar webservices baseado em SOAP para fornecer essas funcionalidades online.

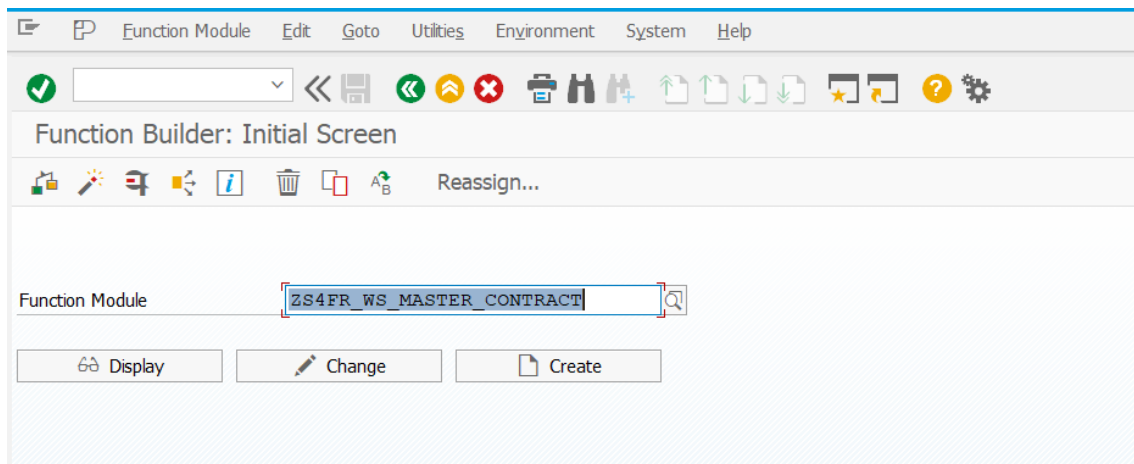


Figura 3 - Ecrã inicial transação SE37

2.2.4. Transação SE11

A transação SE11 é utilizada para aceder ao Dicionário de Dados ABAP (ABAP Dictionary). O Dicionário de Dados contém a estrutura dos dados utilizados no sistema SAP, como tabelas, elementos de dados, domínios e vistas. A SE11 é utilizada para criar e modificar tabelas e os seus componentes, bem como para gerir os relacionamentos entre essas tabelas. Esta transação é essencial para a definição da base de dados lógica do sistema SAP e para garantir a integridade dos dados.

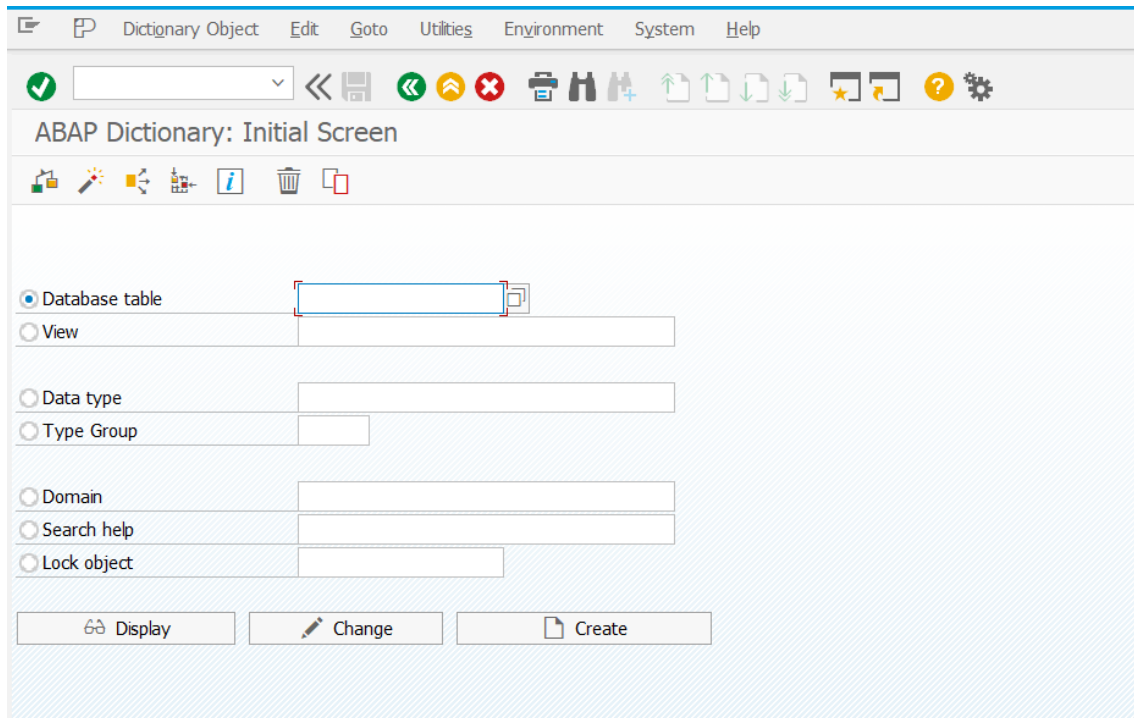


Figura 4 - Ecrã inicial transação SE11

2.2.5. Transação VOFM

A transação VOFM permite a criação e gestão de rotinas para o cálculo e processamento de documentos no módulo SAP SD. Estas rotinas são usadas para definir a lógica personalizada que determina como os preços são calculados, como as ofertas e pedidos de vendas são processados e como os impostos são aplicados. Através da VOFM, é possível ajustar o comportamento padrão do sistema para se alinhar com as necessidades específicas de negócio de uma empresa, garantindo maior flexibilidade na gestão de vendas.

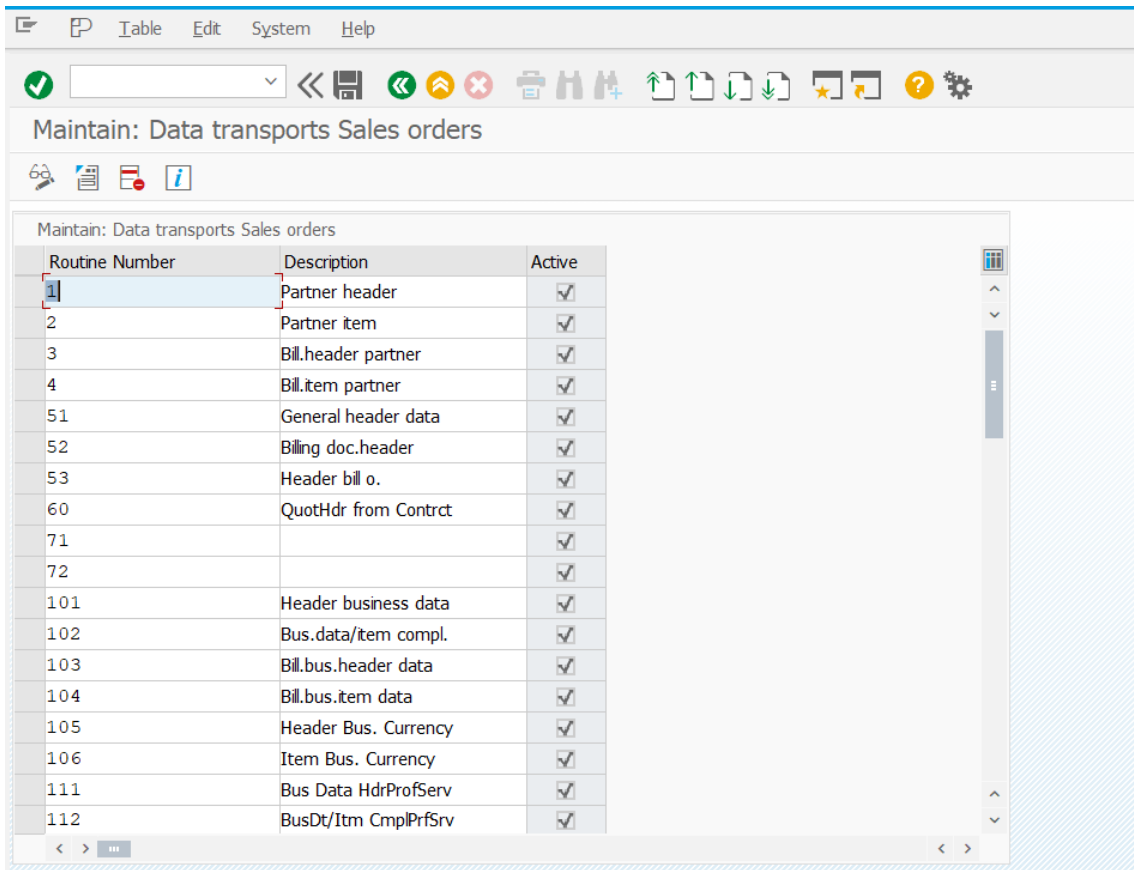


Figura 5 - Ecrã inicial transação VOFM

3. Revisão Bibliográfica

3.1. SAP

O SAP (Systems, Applications, and Products in Data Processing) é um dos sistemas ERP mais amplamente utilizados globalmente, fundado em 1972. Inicialmente focado em contabilidade financeira, evoluiu para um sistema abrangente, integrando processos empresariais como finanças, vendas, logística, e recursos humanos [1]. A sua principal vantagem é a capacidade de integrar dados e processos em tempo real, facilitando a coordenação entre departamentos [2].

3.1.1. Arquitetura e Módulos do SAP

O SAP opera através de uma arquitetura modular, sendo os principais módulos o FI (Financial Accounting), responsável pela contabilidade financeira [3], o SD (Sales and Distribution), que automatiza processos de vendas [4], o MM (Materials Management), utilizado na gestão de stock e compras [5], e o HR (Human Resources), que gere todos os processos de recursos humanos [6]. Estes módulos interagem para promover a eficiência e eliminação de redundâncias nos dados.

3.1.2. Benefícios da Implementação do SAP

A implementação do SAP oferece benefícios como a automação de processos, a integração de dados e a facilitação da tomada de decisões [7]. O sistema centraliza as informações e fornece análises em tempo real, essenciais para decisões estratégicas [1]. No entanto, a resistência à mudança e o custo elevado são desafios frequentes, exigindo uma gestão cuidadosa [8].

3.1.3. Desafios da Implementação do SAP

A implementação de um sistema SAP é um processo complexo, com desafios como o elevado custo e a resistência dos colaboradores à mudança [9]. Muitas empresas subestimam a complexidade do alinhamento do SAP com os processos de negócio existentes [10], e a ausência de um planejamento adequado pode resultar em falhas operacionais [9].

3.1.4. O Futuro do SAP e ERP

Com o lançamento do SAP S/4HANA, que utiliza a base de dados in-memory HANA, o SAP proporciona processamento em tempo real, permitindo o acesso a grandes volumes de dados de forma rápida [11]. A integração de novas tecnologias como a inteligência artificial (IA), IoT e blockchain, bem como a migração para soluções em nuvem, continuarão a transformar o SAP e a mantê-lo como uma ferramenta essencial para as empresas [12].

3.2. ERP SAP

O ERP (Enterprise Resource Planning) é uma solução que integra diversas áreas funcionais numa plataforma única, centralizando dados e otimizando processos empresariais [2]. Sistemas ERP, como o SAP, são essenciais para melhorar a eficiência operacional e facilitar a partilha de informações em tempo real [1].

3.2.1. ERP SAP: Benefícios e Desafios

O SAP S/4HANA é a versão mais recente do SAP, oferecendo processamento de grandes volumes de dados em tempo real com a tecnologia in-memory HANA [11]. A flexibilidade modular do SAP permite personalizações para diferentes setores e tamanhos de empresas [1]. Contudo, a implementação de SAP enfrenta desafios significativos, como custos elevados, complexidade e resistência dos colaboradores, exigindo uma gestão de mudança eficaz [9][8].

3.2.2. O Futuro do ERP SAP

O futuro do SAP está ligado à integração de inteligência artificial (IA), IoT e blockchain, além da migração para soluções em nuvem como o SAP Cloud Platform [12]. Estas inovações permitem automatizar processos e proporcionar maior flexibilidade às empresas.



Figura 6 – Módulos de SAP

3.3. SAP GUI

3.3.1. Conceito e Importância

O SAP GUI (Graphical User Interface) é a interface padrão que permite aos utilizadores interagirem com o sistema SAP, sendo fundamental para o acesso e execução de transações. Através da GUI, os utilizadores podem aceder aos diversos módulos do SAP, realizar operações empresariais e visualizar dados em tempo real [1]. O SAP GUI desempenha um papel vital na facilitação do uso do sistema, tornando o ERP mais acessível a diferentes níveis de utilizadores [2].

3.3.2. Benefícios e Desafios

O principal benefício do SAP GUI é a sua interface amigável que facilita o acesso ao vasto conjunto de transações e funcionalidades do sistema SAP. Melhora a eficiência dos utilizadores ao fornecer um ponto de acesso direto às operações empresariais. No entanto, a curva de aprendizagem do SAP GUI pode ser íngreme para novos utilizadores, e a interface, embora funcional, pode parecer datada quando comparada com soluções mais modernas de ERP [9].

3.3.3. Futuro do SAP GUI

Com a evolução das interfaces, o SAP GUI tem sido complementado por interfaces baseadas em browser, como o SAP Fiori, que oferece uma experiência de utilizador mais moderna e responsiva [12]. No entanto, o SAP GUI continua a ser amplamente utilizado, especialmente em implementações mais antigas e em processos empresariais críticos.

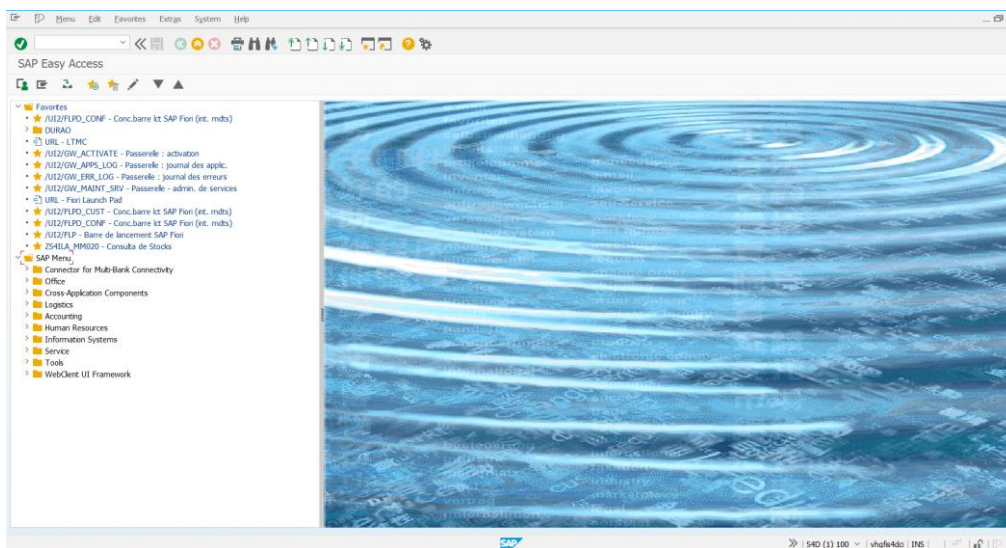


Figura 7 – SAP GUI Easy Access

3.3.4. Transações de SAP GUI utilizadas no âmbito do projeto

Várias transações no SAP GUI são amplamente utilizadas para diferentes funções empresariais, mas para o desenvolvimento da solução proposta incidiu-se principalmente nas seguintes:

Tabela 1 - Transações usadas no âmbito do projeto

Transação	Descrição	Módulo
VA01/VA02/VA03	A transação VA01 é usada para a criação de ordens de vendas, a VA02 para a modificação de ordens de vendas, e a VA03 para a visualização.	SD
VA41/VA42/VA43	A transação VA41 é utilizada para criar contratos de vendas, a VA42 para a modificação, e a VA43 para a visualização de contratos	SD
SE38	Permite a execução e gestão de programas ABAP. Essencial para o desenvolvimento e manutenção de scripts personalizados no sistema SAP.	ABAP
SE37	Utilizada para a gestão de funções ABAP, permitindo a criação e teste de módulos de funções. Ferramenta importante para personalização e integração.	ABAP
SE11	Usada para o dicionário de dados, onde se definem tabelas, vistas e elementos de dados. Vital para a organização e estruturação de dados no SAP.	ABAP/SD
SE16N	Transação que permite visualizar e analisar dados nas tabelas do SAP de forma mais eficiente, substituindo a antiga SE16.	ABAP
VOFM	Permite a criação de rotinas de cópia e regras de cálculo no módulo SD, facilitando a	SD

	personalização dos processos de vendas e distribuição.	
--	--	--

Ao abordar a estrutura organizacional interna do SAP, é imprescindível compreender o conceito de Dados Mestre. Este conceito abrange todos os tipos de informação necessários para a gestão de uma empresa, incluindo dados relativos a clientes, fornecedores, contas, funcionários e materiais, entre outros. O acesso a estas informações específicas é mediado pelo Mandante.

O Mandante representa uma unidade organizacional dentro do sistema SAP, refletindo o grupo corporativo e constituindo o nível mais elevado da hierarquia do sistema. A sua função principal é a configuração, desenvolvimento e testes. Dentro de um Mandante, é possível criar um número ilimitado de empresas, cada uma com as suas próprias tabelas e configurações. Este arranjo evita a duplicação de dados e informações, tornando o Mandante um espaço exclusivo e singular para o armazenamento de dados.

4. Ferramentas/Tecnologias

4.1. SAP ABAP

Como já mencionado no capítulo anterior, a linguagem de programação usada no desenvolvimento deste projeto é ABAP.



Figura 8 – Logotipo SAP

ABAP (Advanced Business Application Programming) é uma linguagem de programação de alto nível desenvolvida pela SAP, concebida para o desenvolvimento de aplicações empresariais no sistema SAP ERP. Esta linguagem é amplamente utilizada para criar programas que interagem com bases de dados relacionais e suportam processos de negócio de grande escala dentro da plataforma SAP [13].

O ambiente de desenvolvimento mais comum é o SAP GUI, através do ABAP Workbench, que inclui ferramentas para a criação, depuração e gestão de programas [13]. ABAP desempenha um papel central na customização de sistemas SAP, permitindo a criação de relatórios, interfaces e funcionalidades adicionais de acordo com as necessidades específicas de cada empresa. A sua capacidade de integração com outros sistemas empresariais torna-o uma escolha robusta para a gestão de processos empresariais críticos [14].

O ABAP List Viewer (ALV) é uma das ferramentas mais versáteis e amplamente utilizadas no desenvolvimento de relatórios dentro do ambiente SAP, sendo fundamental para a apresentação estruturada e interativa de dados. O ALV permite uma fácil manipulação dos dados exibidos, como ordenação, filtragem e agregação, facilitando a análise de grandes volumes de informações empresariais. Existem três formas principais de visualização oferecidas pelo ALV: ALV de lista simples, ALV em árvore (tree) e ALV sequencial hierárquica.

- **A ALV de lista simples:** apresenta os dados em formato tabular, oferecendo uma visualização direta e personalizável, sendo amplamente utilizada em cenários onde é necessário exibir listagens ordenadas e facilmente navegáveis.
- **O ALV tree:** fornece uma estrutura hierárquica que permite representar dados com relações entre diferentes níveis, sendo útil para a visualização de dependências ou agrupamentos dentro de um conjunto de dados.
- **A ALV sequencial hierárquica:** combina a visualização em lista com uma estrutura hierárquica, permitindo a apresentação de dados em múltiplos níveis de detalhe, o que é ideal para contextos onde as informações possuem uma relação de pai e filho.

No contexto deste projeto, foram implementadas duas abordagens de visualização utilizando ALV. Primeiramente, foi utilizada a ALV de lista simples, recorrendo ao módulo de função REUSE_ALV_GRID_DISPLAY_LVC, para apresentar as informações dos Master Contract, nomeadamente no programa principal do Master Contract ou no programa de criação em massa do mesmo.

Adicionalmente, foi utilizada a função REUSE_ALV_HIERSEQ_LIST_DISPLAY para a implementação de uma ALV sequencial hierárquica, permitindo a visualização de dados em diferentes níveis de hierarquia. Esta abordagem foi especialmente útil para representar

as informações dos erros detetados na criação em massa de master contracts, mais especificamente na relação “Master Contract” para “Registo de Erros”.

5. Arquitetura e Modelação do Sistema

Neste capítulo, é apresentada toda a arquitetura e modelação desenvolvidas no processo de implementação do projeto. São expostos os requisitos funcionais e não funcionais, o diagrama de classes, o diagrama de atividades e o diagrama de casos de uso, com o objetivo de ilustrar de forma mais clara e detalhada o funcionamento da aplicação.

Além disso, são descritas as atividades que podem ser realizadas pelo utilizador, tanto na gestão da ferramenta do master contract como na sua aplicação a contratos standard.

5.1. Requisitos Funcionais

5.1.1. Criar, modificar e visualizar um ou mais Master Contract

- **Ator:** Utilizador da aplicação com acesso á transação (via atribuição de role);
- **Tecnologia:** Programa Report SAP ABAP;
- **Descrição:** O programa desenvolvido deve permitir criar Master Contracts, modificar os já existentes e visualizar todos, ou uma lista restrita, baseado num

ecrã de seleção; Este acesso é controlado através de autorizações estabelecidas na transação PFCG;

5.1.2. Manter as tabelas de Eixos de Afetação

- **Ator:** Utilizador com acesso á janela de manutenção das tabelas (via atribuição de role);
- **Tecnologia:** Programa Report SAP ABAP;
- **Descrição:** O programa desenvolvido deve permitir aos utilizadores aceder á janela do ecrã de seleção onde pode aceder às visões de manutenção das tabelas dos Eixos. Este acesso é controlado através de autorizações estabelecidas na transação PFCG;

5.1.3. Criar um Master Contract via Webservice baseado em SOAP

- **Ator:** Equipa de Sales Force CRM;
- **Tecnologia:** SAP ABAP Webservices;
- **Descrição:** O webservice deve ser disponibilizado á equipa de CRM e consumido de tal maneira que seja possível criar um Master Contract com o paylaod de input recebido. No fim retorna o número do Master Contract criado;

5.1.4. Criar Master Contracts em massa através dum ficheiro EXCEL

- **Ator:** Utilizador geral;
- **Tecnologia:** Programa Report SAP ABAP;
- **Descrição:** O report deve conseguir ler ficheiros EXCEL com informações para novos Master Contract, extrair essa informação e criar na base de dados esses mesmos novos contratos. No fim deverá gerar um ficheiro com as linhas de excel que retornaram erro e a sua mensagem de erro respetivamente. Terá também de gerar um objeto de log, que será salvo no sistema para futura consulta; A criação

em massa de Master Contracts terá de respeitar as regras de negócio, nomeadamente relação entre empresas, centros de lucro, clientes, vendedores, etc;

5.1.5. Integrar a nova ferramenta do Master Contract nas Transações Standard SAP para as Ordens de Venda/Contratos

- Ator: Utilizador geral;
- Tecnologia: Programa Report SAP ABAP/ Transação Standard SAP;
- Descrição: As transações standard devem integrar a nova solução standard, contendo os devidos campos para input/output de informação, terem atualização responsiva e consistência nos dados;

5.1.6. Assinação em massa de Master Contracts a Ordens de Venda/Contratos Standard

- Ator: Utilizador geral;
- Tecnologia: Programa Report SAP ABAP;
- Descrição: O report deve conseguir ler um ficheiro EXCEL contendo combinações de 1 Opportunity Number para 1 contrato/ordem de venda. De seguida terá de assignar os vários Master Contract referentes ao Opportunity Number em causa e assigná-lo á ordem de venda/contrato standard presente na combinação. No fim terá de entregar um registo de logs com o resultado de cada linha do ficheiro, sucesso ou erro. Em caso de erro terá de devolver uma mensagem explicativa do mesmo.

5.2. Requisitos Não-Funcionais

5.2.1. Requisitos Gerais

5.2.1.1. Performance

O sistema deve garantir tempos de resposta eficientes e consistentes, assegurando uma experiência fluida para os utilizadores, mesmo sob carga elevada.

5.2.1.2. Eficiência

A aplicação ABAP deve otimizar o uso dos recursos do sistema SAP, reduzindo o tempo de execução e o consumo de memória durante o processamento, garantindo uma execução rápida e economizando recursos do servidor.

5.2.1.3. Interface Gráfica Amigável

A aplicação ABAP deve apresentar uma interface gráfica intuitiva e fácil de utilizar, que facilite a navegação e minimize a necessidade de formação dos utilizadores. Os elementos visuais devem ser claros e organizados, garantindo uma experiência de utilização eficiente e agradável.

5.2.1.4. Robustez

A aplicação ABAP deve ser resistente a falhas e erros, garantindo a continuidade das operações mesmo em situações inesperadas. Deve ser capaz de identificar, gerir e recuperar-se de exceções sem comprometer a integridade dos dados ou interromper o fluxo de trabalho do utilizador.

5.2.1.5. Coerência e consistência de dados

A aplicação ABAP deve assegurar que os dados mantêm integridade e precisão ao longo de todas as operações e processos. Alterações ou atualizações nos dados devem refletir-se uniformemente em todas as transações e interfaces, evitando discrepâncias e garantindo que todos os utilizadores acedem a informações fiáveis e sincronizadas.

5.2.2. Tecnologias

5.2.2.1. ERP SAP

SAP ERP é uma solução integrada de gestão empresarial que centraliza processos de negócio, permitindo uma gestão eficiente de recursos, informações e operações em tempo real, promovendo a colaboração entre diferentes áreas da organização e melhorando a tomada de decisões.

5.2.2.2. SAP ABAP

SAP ABAP (Advanced Business Application Programming) é uma linguagem de programação desenvolvida pela SAP, utilizada para personalizar e expandir as funcionalidades dos sistemas SAP, permitindo a criação de relatórios, interfaces e aplicações específicas que atendem às necessidades empresariais.

5.3. Diagrama de Classes

Na Figura 9, apresenta-se o diagrama de classes, o qual desempenha um papel fundamental, dado que as classes nele representadas contêm os dados essenciais para o correto funcionamento da aplicação.

Na classe Utilizador representamos o interveniente principal, que pode criar, modificar e visualizar Master Contracts, bem como assignar ou anular a assignação de um Master Contract a um contrato/ordem de venda standard.

Na classe Master Contract representa o objeto principal criado, o contrato mestre. Nela podemos ver os atributos dum Master Contract, bem como as atividades que nele podemos executar

Na classe Contract Header temos representado a informação obrigatória para o cabeçalho dum ordem de venda/contrato na implementação do Master Contract. Nesta podemos assignar ou anular a assignação do contrato mestre.

Na classe Contract Item temos representado a informação obrigatória para o item dum ordem de venda/contrato na implementação do Master Contract. Nesta podemos assignar ou anular a assignação do contrato mestre.

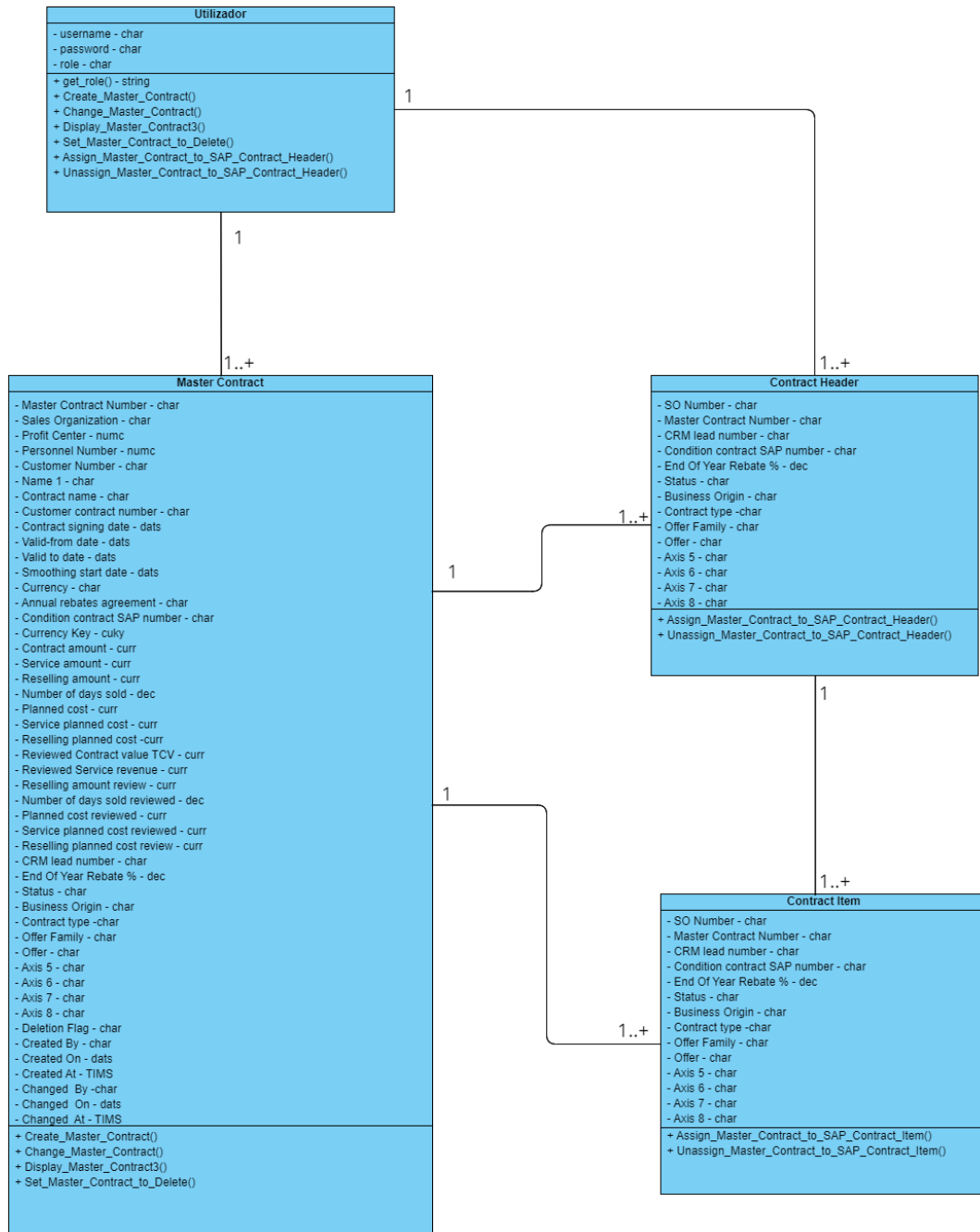


Figura 9 - Diagrama de Classes

5.4. Diagrama de Atividades

Um diagrama de atividades é uma representação visual que descreve o fluxo de ações num sistema, evidenciando a sequência e lógica das operações para facilitar a compreensão dos processos internos.

No contexto do desenvolvimento, podemos ver as diversas atividades que podem ser executadas, sendo:

- Visualizar um Master Contract, ilustrado na Figura 10;
- Modificar um Master Contract, ilustrado na Figura 11;
- Criar um Master Contract, ilustrado na Figura 12;
- Assignar um Master Contract a um contrato standard, ilustrado na Figura 13;

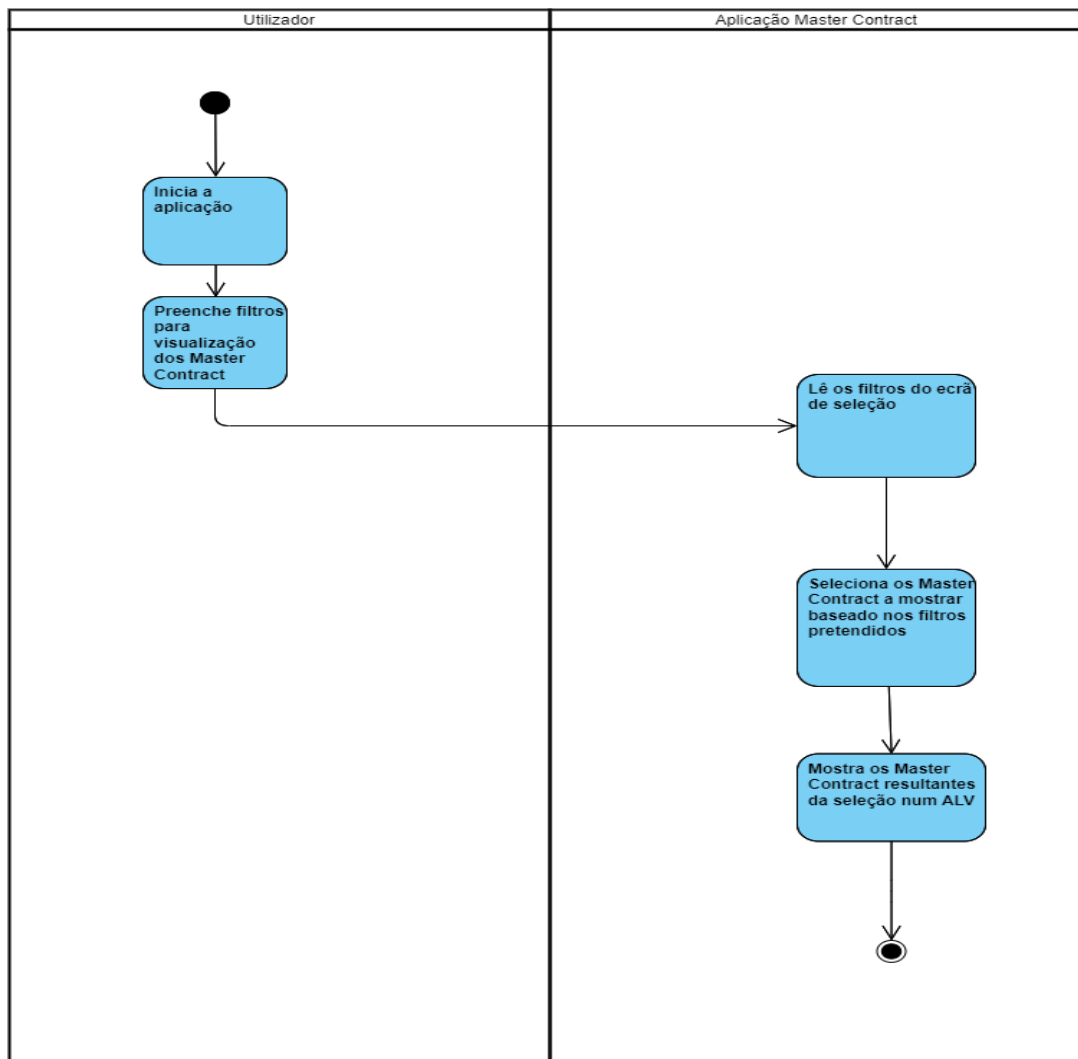


Figura 10 - Diagrama de Atividades: Visualizar um Master Contract

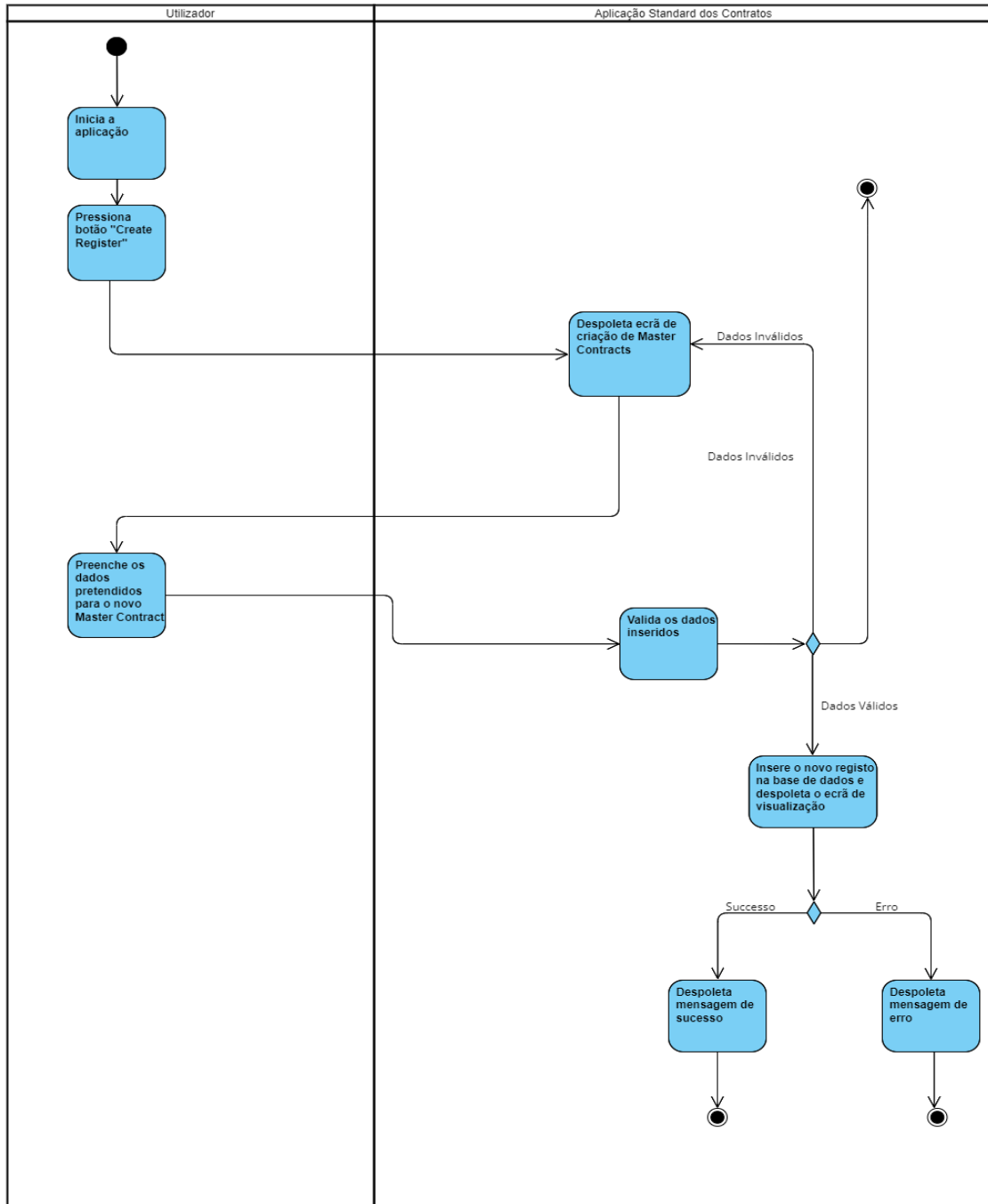


Figura 11 - Diagrama de Atividades: Criar um Master Contract

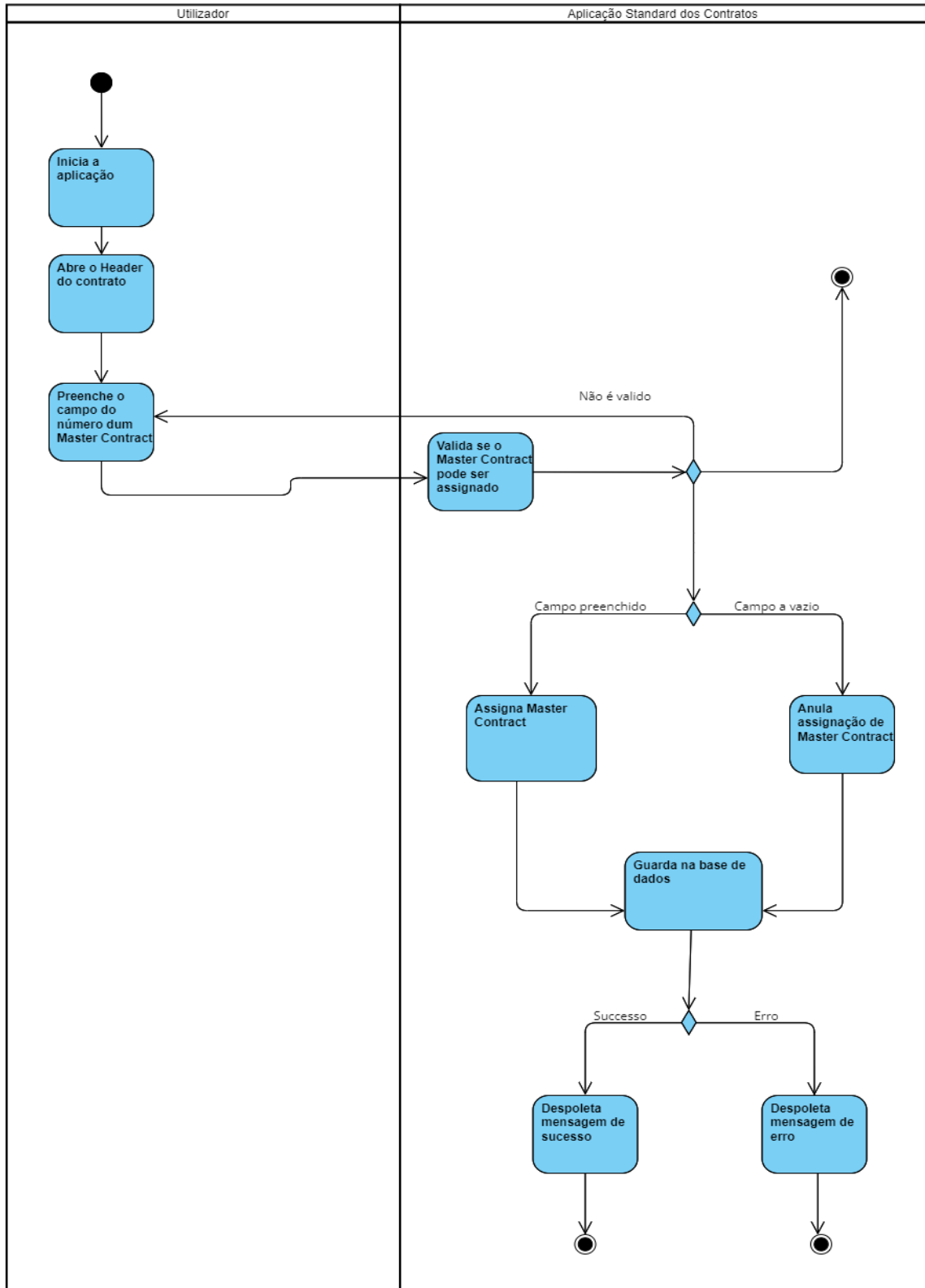


Figura 12 - Diagrama de Atividades: Modificar um Master Contract

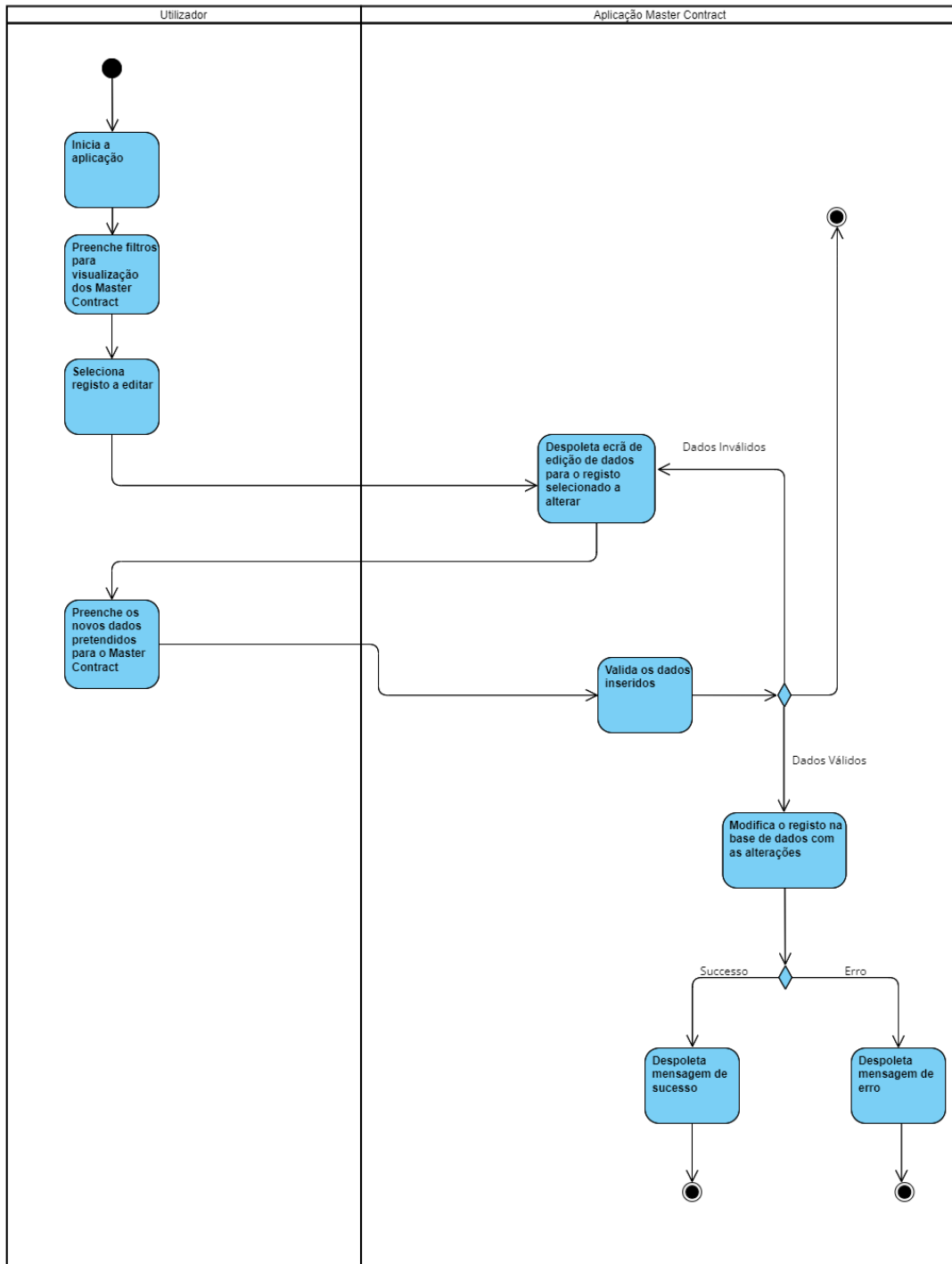


Figura 13 - Diagrama de Atividades: Assignar um Master Contract a um contrato standard

5.5. Diagrama de Casos de Uso

O diagrama de Casos de Uso destaca as funcionalidades que o utilizador da aplicação pode realizar. Visto que o desenvolvimento se concentrou na implementação das funcionalidades da solução, apenas está representada a entidade Utilizador.

O utilizador que tenha o *role* necessário para aceder á transação pode executar três ações principalmente: criar, modificar e visualizar Master Contracts. Para criar, o utilizador terá de fornecer os dados do novo contrato mestre necessários e válidos perante as condições do negócio. Já para a visualizar e modificar Master Contracts, o utilizador terá primeiro de pedir uma lista desses contratos baseados nos filtros que achar relevantes. Para modificar terá, adicionalmente, de selecionar o registo pretendido e modificar os dados com os novos valores.

Por outro lado, o utilizador terá também a possibilidade de assignar um Master Contract a vários contratos standard ao preencher os campos *custom* desenvolvidos para o efeito nas transações.

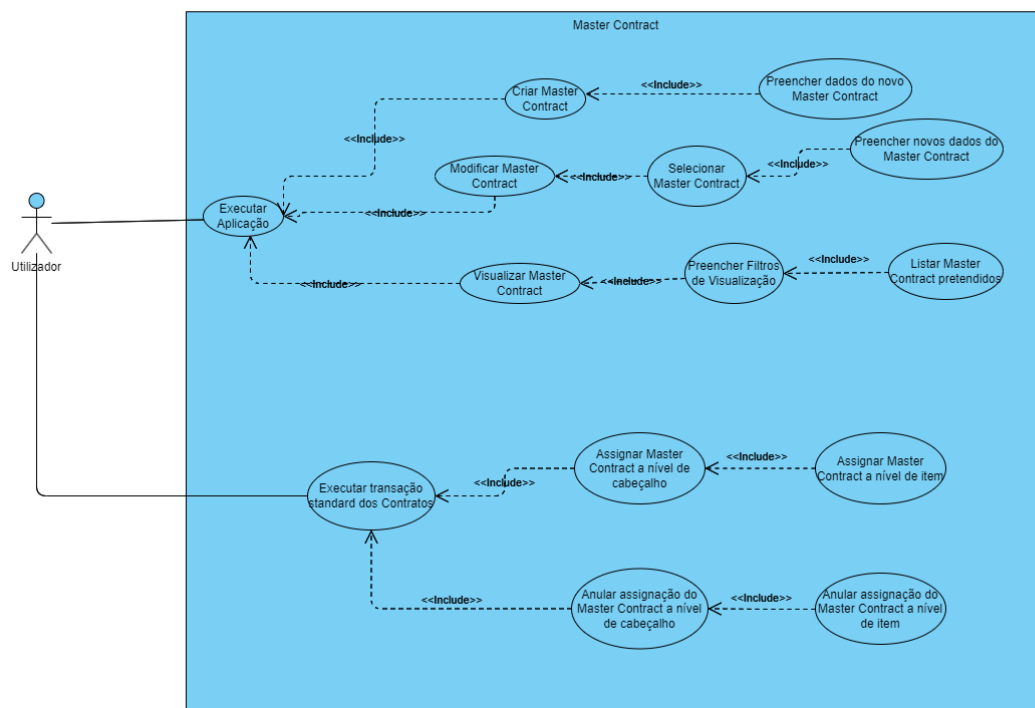


Figura 14 - Diagrama de Casos de Uso

6. Desenvolvimento

O desenvolvimento do projeto foi orientado por Scrum e Kanban com o Jira como ferramenta de suporte. As sprints curtas do Scrum e a visualização de tarefas do Kanban, geridas pelo Jira, facilitaram o controlo e acompanhamento do progresso de forma ágil e organizada. Assim foi possível de manter a equipa funcional de França a par dos últimos desenvolvimentos e receber feedback dos mesmo sobre possíveis alterações/correções.

6.1. Implementação dos Objetos do Dicionário de Dados

Para oferecer o melhor suporte e coerência possível na aplicação, é extremamente importante ter uma boa base de dados, bem estruturada e otimizada. Para tal, foi amplamente discutido com a equipa funcional quais os campos chave, as ligações com chaves externas, ajudas de pesquisa etc. De todos os objectos do dicionário de dados destacam-se os seguintes:

6.1.1. Tabela

Nome: ZS4FR_MAST_CONTR (Figura 15)

Descrição: Master Contract data table

A tabela em questão é responsável por guardar todas as informações do Master Contract, tendo como campo chave o ZMC_NUMBER (número único atribuído ao Master Contract).

Field	Key	Inti...	Data element	Data Type	Length	Decim...	Coordinate	Short Description
MANDT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	MANDT	CLNT	3	0		0 Client
ZMC_NUMBER	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	ZS4FR_MC_NUMBER	CHAR	10	0		0 Master contract number
VKORG	<input type="checkbox"/>	<input type="checkbox"/>	VKORG	CHAR	4	0		0 Sales Organization
PRCTR	<input type="checkbox"/>	<input type="checkbox"/>	ZS4FR_MC_PRCTR	NUMC	10	0		0 Profit Center
PERNR	<input type="checkbox"/>	<input type="checkbox"/>	PERNSO	NUMC	8	0		0 Personnel Number
KUNNR	<input type="checkbox"/>	<input type="checkbox"/>	KUNNR	CHAR	10	0		0 Customer Number
NAME1	<input type="checkbox"/>	<input type="checkbox"/>	NAME1_GP	CHAR	35	0		0 Name 1
ZMC_NAME	<input type="checkbox"/>	<input type="checkbox"/>	ZS4FR_MC_NAME	CHAR	40	0		0 Contract name
ZMC_CUSTCN	<input type="checkbox"/>	<input type="checkbox"/>	ZS4FR_MC_CUSTCN	CHAR	10	0		0 Customer contract number
ZMC_VUNDAT	<input type="checkbox"/>	<input type="checkbox"/>	ZS4FR_MC_VUNDAT	DATS	8	0		0 Contract signing date
ZMC_GUEBG	<input type="checkbox"/>	<input type="checkbox"/>	ZS4FR_MC_GUEBG	DATS	8	0		0 Valid-from date
ZMC_GUEEN	<input type="checkbox"/>	<input type="checkbox"/>	ZS4FR_MC_GUEEN	DATS	8	0		0 Valid to date
ZMC_STSMT	<input type="checkbox"/>	<input type="checkbox"/>	ZS4FR_MC_STSMT	DATS	8	0		0 Smoothing start date
ZMC_CURRENCY	<input type="checkbox"/>	<input type="checkbox"/>	ZS4FR_MC_CURR	CHAR	4	0		0 Currency
ZMC_ARA	<input type="checkbox"/>	<input type="checkbox"/>	ZS4FR_MC_ARA	CHAR	3	0		0 Annual rebates agreement
ZMC_CCN	<input type="checkbox"/>	<input type="checkbox"/>	ZS4FR_MC_CCN	CHAR	10	0		0 Condition contract SAP number
ZMC_WAERS	<input type="checkbox"/>	<input type="checkbox"/>	WAERS	CURY	5	0		0 Currency Key
ZMC_NETWR	<input type="checkbox"/>	<input type="checkbox"/>	ZS4FR_MC_NETWR	CURR	15	2		0 Contract amount
ZMC_NETSER	<input type="checkbox"/>	<input type="checkbox"/>	ZS4FR_MC_NETSERV	CURR	15	2		0 Service amount
ZMC_NETRES	<input type="checkbox"/>	<input type="checkbox"/>	ZS4FR_MC_NETRESE	CURR	15	2		0 Reselling amount
ZMC_NBDAYS	<input type="checkbox"/>	<input type="checkbox"/>	ZS4FR_NB_DAYS	DEC	15	2		0 Number of days sold

Figura 15 - Tabela ZS4FR_MAST_CONTR

6.1.2. Estrutura

De modo a implementar as tabelas standard para acomodar a nova ferramenta do Master Contract foi optado por adicionar os campos encapsulados numa estrutura *append*. Desta forma podemos isolar os novos campos da nova ferramenta dos restaurantes campos standard sem correr o risco de afetar a coesão da base de dados standard. As tabelas standard em questão são a tabela de dados do Cabeçalho de Ordem de Venda, VBAK (Figura 16), e a tabela de dados do Item de Ordem de Venda (Figura 17).

Dictionary: Display Table

Transparent Table: VBAK Active

Short Description: Sales Document: Header Data

Attributes | Delivery and Maintenance | Fields | Input Help/Check | Currency/Quantity Fields | Indexes

Field	Key	Inti...	Data element	Data Type	Length	Decim...	Coordinate	Short Description
.APPEND			ZS4FR_MC_VBAK	STRU	0	0		0 Master Contract Header Data
ZZMC_NUMBER			ZS4FR_MC_NUMBER	CHAR	10	0		0 Master contract number
ZZMC_LEAD			ZS4FR_MC_LEAD	CHAR	18	0		0 CRM lead number
ZZMC_CCN			ZS4FR_MC_CCN	CHAR	10	0		0 Condition contract SAP number
ZZMC_EOYR			ZS4FR_MC_EOYR	DEC	5	2		0 End Of Year Rebate %
ZZMC_STATUS			ZS4FR_MC_STATUS	CHAR	3	0		0 Status
ZZAXIS01_CODE			ZS4FR_CODE	CHAR	10	0		0 Business Origin
ZZAXIS01_DESC			ZS4FR_DESCR	CHAR	50	0		0 Description
ZZAXIS02_CODE			ZS4FR_AXIS02_CO..	CHAR	10	0		0 Contract type
ZZAXIS02_DESC			ZS4FR_DESCR	CHAR	50	0		0 Description
ZZAXIS03_CODE			ZS4FR_MC_AXIS03..	CHAR	20	0		0 Offer Family
ZZAXIS03_DESC			ZS4FR_DESCR	CHAR	50	0		0 Description
ZZAXIS04_CODE			ZS4FR_MC_AXIS04..	CHAR	20	0		0 Offer
ZZAXIS04_DESC			ZS4FR_DESCR	CHAR	50	0		0 Description
ZZAXIS06_CODE			ZS4FR_AXIS06_CO..	CHAR	10	0		0 Axis 6
ZZAXIS06_DESC			ZS4FR_DESCR	CHAR	50	0		0 Description
ZZAXIS07_CODE			ZS4FR_AXIS07_CO..	CHAR	10	0		0 Axis 7
ZZAXIS07_DESC			ZS4FR_DESCR	CHAR	50	0		0 Description
ZZAXIS08_CODE			ZS4FR_AXIS08_CO..	CHAR	10	0		0 Axis 8
ZZAXIS08_DESC			ZS4FR_DESCR	CHAR	50	0		0 Description
ZZAXIS09_CODE			ZS4FR_AXIS09_CO..	CHAR	10	0		0 Axis 7

Figura 16 - Tabela VBAK do Cabeçalho da Ordem de Venda

Dictionary: Display Table

Transparent Table: VBAP Active

Short Description: Sales Document: Item Data

Attributes | Delivery and Maintenance | Fields | Input Help/Check | Currency/Quantity Fields | Indexes

Field	Key	Inti...	Data element	Data Type	Length	Decim...	Coordinate	Short Description
.APPEND			ZS4FR_MC_VBAP	STRU	0	0		0 Master Contract Item Data
ZZMC_NUMBER			ZS4FR_MC_NUMBER	CHAR	10	0		0 Master contract number
ZZMC_LEAD			ZS4FR_MC_LEAD	CHAR	18	0		0 CRM lead number
ZZMC_CCN			ZS4FR_MC_CCN	CHAR	10	0		0 Condition contract SAP number
ZZMC_EOYR			ZS4FR_MC_EOYR	DEC	5	2		0 End Of Year Rebate %
ZZMC_STATUS			ZS4FR_MC_STATUS	CHAR	3	0		0 Status
ZZAXIS01_CODE			ZS4FR_CODE	CHAR	10	0		0 Business Origin
ZZAXIS02_CODE			ZS4FR_AXIS02_CO..	CHAR	10	0		0 Contract type
ZZAXIS03_CODE			ZS4FR_MC_AXIS03..	CHAR	20	0		0 Offer Family
ZZAXIS04_CODE			ZS4FR_MC_AXIS04..	CHAR	20	0		0 Offer
ZZAXIS06_CODE			ZS4FR_AXIS06_CO..	CHAR	10	0		0 Axis 6
ZZAXIS07_CODE			ZS4FR_AXIS07_CO..	CHAR	10	0		0 Axis 7
ZZAXIS08_CODE			ZS4FR_AXIS08_CO..	CHAR	10	0		0 Axis 8
ZZAXIS09_CODE			ZS4FR_AXIS09_CO..	CHAR	10	0		0 Axis 7
ZZAXIS10_CODE			ZS4FR_AXIS10_CO..	CHAR	10	0		0 Axis 8
.APPEND			ZS4FR_REBATE_AG..	STRU	0	0		0 Customer structure Rebate Agreement
ZZRFA			ZZRFA	CHAR	1	0		0 Rebate Agreement (RFA)
.APPEND			ZS4FR_VBAP_CUST..	STRU	0	0		0 Custim Fields to retrieve the price and cost of the material
ZZQUOTEID			ZS4FR_QUOTEID	CHAR	20	0		0 Original Quotation ID
ZZSBO			ZS4FR_SBO	CHAR	40	0		0 SBO Agreement Number.
ZZBQUODATE			ZS4FR_BQUODATE	DATS	8	0		0 Quotation SBO Price valid from

Figura 17 - Tabela VBAP do Item da Ordem de Venda

Como é possível ver nas imagens acima, a estrutura que suporta os dados do Master Contract a nível de cabeçalho, na tabela VBAK, portanto, é ZS4FR_MC_VBAK. E a estrutura que suporta os dados do Master Contract a nível de item, na tabela VBAP, portanto, é ZS4FR_MC_VBAP.

Estas contemplam apenas as informações relevantes a mostrar a nível da transação, e não o registo completo do Master Contract associado. Os campos nelas presentes são os que serão disponibilizados na janela de input nas transações standard e que guardarão as informações do Master Contract assignado.

Nome: ZS4FR_MC_VBAK (Figura 18)

Descrição: Master Contract Header Data

Dictionary: Display Append Structure

Append Structure: ZS4FR_MC_VBAK Active

Short Description: Master Contract Header Data

Attributes Components Input Help/Check Currency/quantity fields

Component	Typing Method	Component Type	Data Type	Length	Decim...	Coordinate	Short Description
ZZMC_NUMBER	1 Types	ZS4FR_MC_NUMBER	CHAR	10	0		0 Master contract number
ZZMC_LEAD	1 Types	ZS4FR_MC_LEAD	CHAR	18	0		0 CRM lead number
ZZMC_CCN	1 Types	ZS4FR_MC_CCN	CHAR	10	0		0 Condition contract SAP number
ZZMC_EOYR	1 Types	ZS4FR_MC_EOYR	DEC	5	2		0 End Of Year Rebate %
ZZMC_STATUS	1 Types	ZS4FR_MC_STATUS	CHAR	3	0		0 Status
ZZAXIS01_CODE	1 Types	ZS4FR_CODE	CHAR	10	0		0 Business Origin
ZZAXIS01_DESC	1 Types	ZS4FR_DESCR	CHAR	50	0		0 Description
ZZAXIS02_CODE	1 Types	ZS4FR_AXIS02_CO_	CHAR	10	0		0 Contract type
ZZAXIS02_DESC	1 Types	ZS4FR_DESCR	CHAR	50	0		0 Description
ZZAXIS03_CODE	1 Types	ZS4FR_MC_AXIS03_	CHAR	20	0		0 Offer Family
ZZAXIS03_DESC	1 Types	ZS4FR_DESCR	CHAR	50	0		0 Description
ZZAXIS04_CODE	1 Types	ZS4FR_MC_AXIS04_	CHAR	20	0		0 Offer
ZZAXIS04_DESC	1 Types	ZS4FR_DESCR	CHAR	50	0		0 Description
ZZAXIS06_CODE	1 Types	ZS4FR_AXIS06_CO_	CHAR	10	0		0 Axis 6
ZZAXIS06_DESC	1 Types	ZS4FR_DESCR	CHAR	50	0		0 Description
ZZAXIS07_CODE	1 Types	ZS4FR_AXIS07_CO_	CHAR	10	0		0 Axis 7
ZZAXIS07_DESC	1 Types	ZS4FR_DESCR	CHAR	50	0		0 Description
ZZAXIS08_CODE	1 Types	ZS4FR_AXIS08_CO_	CHAR	10	0		0 Axis 8
ZZAXIS08_DESC	1 Types	ZS4FR_DESCR	CHAR	50	0		0 Description
ZZAXIS09_CODE	1 Types	ZS4FR_AXIS09_CO_	CHAR	10	0		0 Axis 7
ZZAXIS09_DESC	1 Types	ZS4FR_DESCR	CHAR	50	0		0 Description
ZZAXIS10_CODE	1 Types	ZS4FR_AXIS10_CO_	CHAR	10	0		0 Axis 8

Figura 18 - Estrutura Master Contract na tabela VBAK

Nome: ZS4FR_MC_VBAP (Figura 19)

Descrição: Master Contract Item Data

Dictionary: Display Append Structure

Append Structure: ZS4FR_MC_VBAP Active

Short Description: Master Contract Item Data

Attributes Components Input Help/Check Currency/quantity fields

Component	Typing Method	Component Type	Data Type	Length	Decim...	Coordinate	Short Description
ZZMC_NUMBER	1 Types	ZS4FR_MC_NUMBER	CHAR	10	0		0 Master contract number
ZZMC_LEAD	1 Types	ZS4FR_MC_LEAD	CHAR	18	0		0 CRM lead number
ZZMC_CCN	1 Types	ZS4FR_MC_CCN	CHAR	10	0		0 Condition contract SAP number
ZZMC_EOYR	1 Types	ZS4FR_MC_EOYR	DEC	5	2		0 End Of Year Rebate %
ZZMC_STATUS	1 Types	ZS4FR_MC_STATUS	CHAR	3	0		0 Status
ZZAXIS01_CODE	1 Types	ZS4FR_CODE	CHAR	10	0		0 Business Origin
ZZAXIS02_CODE	1 Types	ZS4FR_AXIS02_CO_	CHAR	10	0		0 Contract type
ZZAXIS03_CODE	1 Types	ZS4FR_MC_AXIS03_	CHAR	20	0		0 Offer Family
ZZAXIS04_CODE	1 Types	ZS4FR_MC_AXIS04_	CHAR	20	0		0 Offer
ZZAXIS06_CODE	1 Types	ZS4FR_AXIS06_CO_	CHAR	10	0		0 Axis 6
ZZAXIS07_CODE	1 Types	ZS4FR_AXIS07_CO_	CHAR	10	0		0 Axis 7
ZZAXIS08_CODE	1 Types	ZS4FR_AXIS08_CO_	CHAR	10	0		0 Axis 8
ZZAXIS09_CODE	1 Types	ZS4FR_AXIS09_CO_	CHAR	10	0		0 Axis 7
ZZAXIS10_CODE	1 Types	ZS4FR_AXIS10_CO_	CHAR	10	0		0 Axis 8

Figura 19 - Estrutura Master Contract na tabela VBAB

6.2. Desenvolvimento programa principal Master Contract

Após solidificar a base de dados, é necessário desenvolver o programa principal da nova ferramenta. O programa principal chama-se ZS4FR_MASTER_CONTRACT, foi guardado no pacote do módulo Sales and Distribution com a descrição “Maintenance of Master Contract information”.

Como já explicado anteriormente, a funcionalidade deste relatório é possibilitar a criação, modificação e visualização dos Master Contract. Existe também a possibilidade de customizar as tabelas dos eixos de afetação. Esta funcionalidade extra é importante pois as tabelas dos eixos são responsáveis por conter os valores válidos para os respetivos eixos no master contract. Quando se despoleta a ajuda de pesquisa para cada eixo, a mesma procura as entradas possíveis nestas tabelas.

Numa abordagem mais profunda, verificamos que o código standard do SAP funciona muito á base de *performs*, isto é, extratos de código tipo módulos. Contudo, como o paradigma de programação em SAP tem evoluído cada vez mais para POO (Programação Orientada a Objetos), decidi desenvolver o programa baseado em classes e métodos. A classe usada é local, isto é, apenas pode ser usada no programa do Master Contract pois está apenas declarada no mesmo. O SAP permite declarar classes globais para serem reutilizadas noutras instâncias através da transação SE24.

6.2.1. Report ZS4FR_MASTER_CONTRACT

O report ZS4FR_MASTER_CONTRACT(Figura 20) é o include principal, nele fazes a inclusão dos restantes includes necessários á execução do programa. O include é um objecto em sap que contém pedaços de código que foram agregados, normalmente, pela sua funcionalidade. Para o programa principal do Master Contract temos os seguintes includes:

- INCLUDE zs4fr_master_contract_dat, com as tipificações e declarações de variáveis globais ao programa;

- INCLUDE zs4fr_master_contract_scr, onde definimos o ecrã de seleção para o utilizador poder filtrar a pesquisa por Master Contracts, criar uma Master Contract ou alternar entre janelas e poder manter registos das tabelas de eixos;
- INCLUDE zs4fr_master_contract_cla, neste include definimos a classe local ao programa, onde declaramos os seus métodos e implementamos os mesmos;
- INCLUDE zs4fr_master_contract_f01, para o desenvolvimento das rotinas necessárias.
- INCLUDE zs4fr_master_contract_pbo, para o processamento da lógica antes de iniciarmos os sub-ecrãs, sendo estes os de criação, modificação e visualização;
- INCLUDE zs4fr_master_contract_pai, para o processamento de lógica depois do utilizador interagir com os ecrãs anteriormente especificados;

```

1 | *~
2 | *~ Report ZS4FR_MASTER_CONTRACT
3 | *~
4 | *~
5 | *~
6 | *~ Author.....: Hugo Borges
7 | *~ Company.....: INETUM
8 | *~ Description..: OTC - MASTER CONTRACT
9 | *~ Jira.....: GFSHC-7143: OTC - MASTER CONTRACT
10 | *~ Date.....: February 2023
11 | *~ Transaction..: ZS4_MASTER_CONTRACT
12 | *~
13 | *~
14 | REPORT zs4fr_master_contract MESSAGE-ID zs4fr_master_contr.
15 |
16 |
17 | INCLUDE zs4fr_master_contract_dat. "Local Tipif. and Global Decl.
18 |
19 | INCLUDE zs4fr_master_contract_scr. " Selection Screen
20 |
21 | INCLUDE zs4fr_master_contract_cla. "Local class
22 |
23 | INCLUDE zs4fr_master_contract_f01. "Rotines
24 |
25 | INCLUDE zs4fr_master_contract_pbo. "PBO
26 |
27 | INCLUDE zs4fr_master_contract_pai. "PAI
28 |
29 | INITIALIZATION.
30 |
31 |     CALL METHOD main=>initialization.
32 |
33 |
34 | START-OF-SELECTION.
35 |
36 |     CALL METHOD main=>principal.

```

Figura 20 - Report ZS4FR_MASTER_CONTRACT

Como manter os contratos mestre da nova funcionalidade é uma tarefa importante e crítica á nova ferramenta, foi pedido a aplicação de controlo de acessos á transação pelo uso de *roles*. Para tal tive de usar a transação PFCG para criar o role desejado com os privilégios de acesso ao programa.

Antes de criar o role foi necessário criar uma transação associada ao programa principal, a transação ZS4_MASTER_CONTRACT (Figura 21).

Transaction code	ZS4_MASTER_CONTRACT
Package	ZS4FRSD
Transaction text	Master Contract
Start Options	
Program	ZS4FR_MASTER_CONTRACT
Selection screen	1000
Start with variant	
Authorization Object	Values
Classification	
Transaction classification	
<input checked="" type="radio"/> Professional User Transaction	
<input type="radio"/> Easy Web Transaction	Service
<input type="checkbox"/> Pervasive enabled	
GUI support	
<input checked="" type="checkbox"/> SAP GUI for HTML	
<input checked="" type="checkbox"/> SAP GUI for Java	
<input checked="" type="checkbox"/> SAP GUI for Windows	

Figura 21 - Transação ZS4_MASTER_CONTRACT

Depois de criada a transação associada ao programa ZS4FR_MASTER_CONTRACT, foi possível então criar o role de autorização para o acesso á mesma.

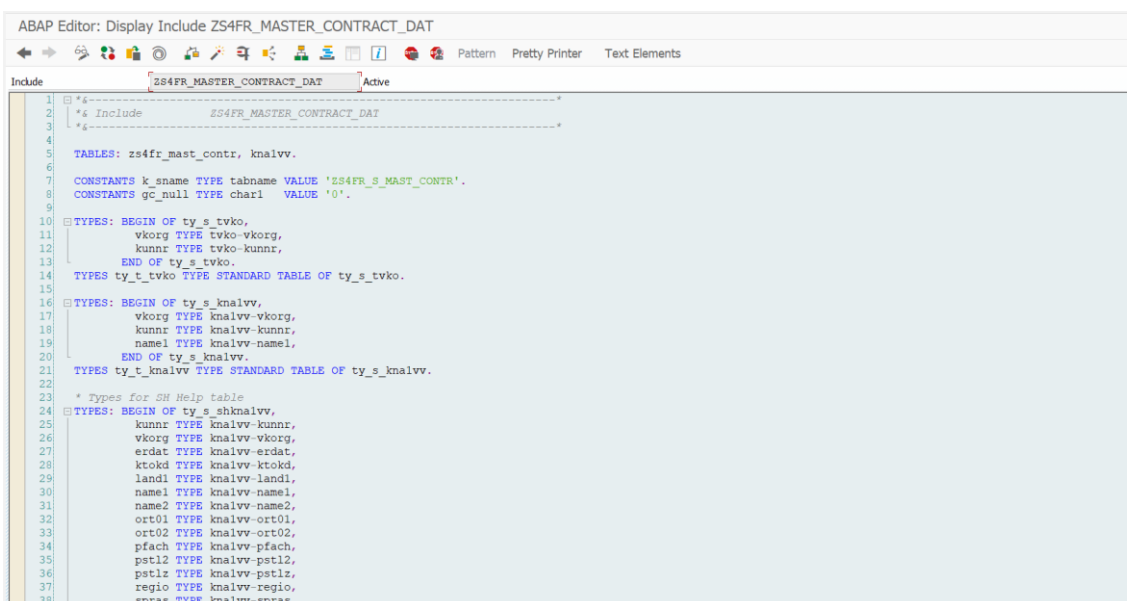
O role definido é ZS4FR_MASTER_CONTRACT_ROLE (Figura 22), e apenas os utilizadores associados a ela podem aceder á transação.

Display Roles					
Other role Inheritance					
Role					
Role	ZS4FR_MASTER_CONTRACT_ROLE				<input type="checkbox"/> Obsolete
Description	Role to access Master Contract t-code ZS4_MASTER_CONTRACT				
Target System	No destination				
Description Menu Applications Workflow Authorizations User MiniApps Personalization					
Created		Last Changed		Last Profile Generation	
User	MSFONSECA	User	MSFONSECA	User	MSFONSECA
Date	19.07.2024	Date	19.07.2024	Date	19.07.2024
Time	10:33:15	Time	10:33:15	Time	10:33:15
Information About Authorization Profile					
Profile Name	T-SD070472				
Profile Text	Profile for role ZS4FR_MASTER_CONTRACT_ROLE				
Status	Authorization profile is current				
Edit Authorization Data and Generate Profiles					
Display Authorization Data					

Figura 22 - Role ZS4FR_MASTER_CONTRACT_ROLE para o acesso á transação ZS4_MASTER_CONTRACT

6.2.2. Include ZS4FR_MASTER_CONTRACT_DAT

Neste include, como já referido acima, procede-se á declaração de variáveis globais necessárias ao tratamento de dados. Nestas, vale realçar a declaração dos tipos de tabela para a tabela de visualização do ALV, das tabelas necessárias ás ajudas de pesquisa e da estrutura para o ALV. Esta última partilha a estrutura da tabela principal do Master Contract com a adição do campo muito importante, o BOXFD. Este campo é necessário para a funcionalidade de selecionar um registo da tabela de visualização e poder editar o mesmo.



```

ABAP Editor: Display Include ZS4FR_MASTER_CONTRACT_DAT
-----
Include ZS4FR_MASTER_CONTRACT_DAT Active
-----
1  *~
2  *~ Include ZS4FR_MASTER_CONTRACT_DAT
3  *~
4
5  TABLES: zs4fr_mast_contr, knalvv.
6
7  CONSTANTS k_sname TYPE tabname VALUE 'ES4FR_S_MAST_CONTR'.
8  CONSTANTS gc_null TYPE char1 VALUE '0'.
9
10 TYPES: BEGIN OF ty_s_tvko,
11         vkorg TYPE tvko-vkorg,
12         kunnr TYPE tvko-kunnr,
13         END OF ty_s_tvko.
14 TYPES ty_t_tvko TYPE STANDARD TABLE OF ty_s_tvko.
15
16 TYPES: BEGIN OF ty_s_knalvv,
17         vkorg TYPE knalvv-vkorg,
18         kunnr TYPE knalvv-kunnr,
19         name1 TYPE knalvv-name1,
20         END OF ty_s_knalvv.
21 TYPES ty_t_knalvv TYPE STANDARD TABLE OF ty_s_knalvv.
22
23 * Types for SH Help table
24 TYPES: BEGIN OF ty_s_shknalvv,
25         kunnr TYPE knalvv-kunnr,
26         vkorg TYPE knalvv-vkorg,
27         erdat TYPE knalvv-erdat,
28         ktokd TYPE knalvv-ktokd,
29         land1 TYPE knalvv-land1,
30         name1 TYPE knalvv-name1,
31         name2 TYPE knalvv-name2,
32         ort01 TYPE knalvv-ort01,
33         ort02 TYPE knalvv-ort02,
34         pfach TYPE knalvv-pfach,
35         pstl2 TYPE knalvv-pstl2,
36         pstlz TYPE knalvv-pstlz,
37         regio TYPE knalvv-regio,
38         sdras TYPE knalvv-sdras,

```

Figura 23 - Include ZS4FR_MASTER_CONTRACT_DAT

6.2.3. Include ZS4FR_MASTER_CONTRACT_SCR

Neste bloco de inclusão definimos o ecrã de seleção. Este é responsável por apresentar os filtros que o negócio considera relevantes. A partir desses filtros procedemos á seleção dos registos pretendidos pelo utilizador. Como pedido pela equipa funcional, foi adicionada também uma *tab* para a manutenção e customização das tabelas de eixos de afetação.

Para o acesso á segunda *tab*, dos eixos de afetação, devido á importância das tabelas nela presente, foi pedido um controlo sobre os utilizadores que podem acessá-la.

Semelhante ao que foi feito para controlar o acesso á transação, para este cenário foi preciso criar outro role. No entanto, como no contexto da ferramenta, estamos a tentar

controlar o acesso a uma tabe específica e não um objeto regular como uma transação, foi necessário criar um objeto de autorização.

Objetos de autorizações são objetos que controlam o tipo de autorizações que um utilizador com um role pode executar em *runtime* num certo pedaço de código. Para o efeito usamos as autorizações de editar e visualizar. A transação necessária para tal é a SU21 e o objeto de autorização criado foi Z_ADMIN_MC (Figura 24).

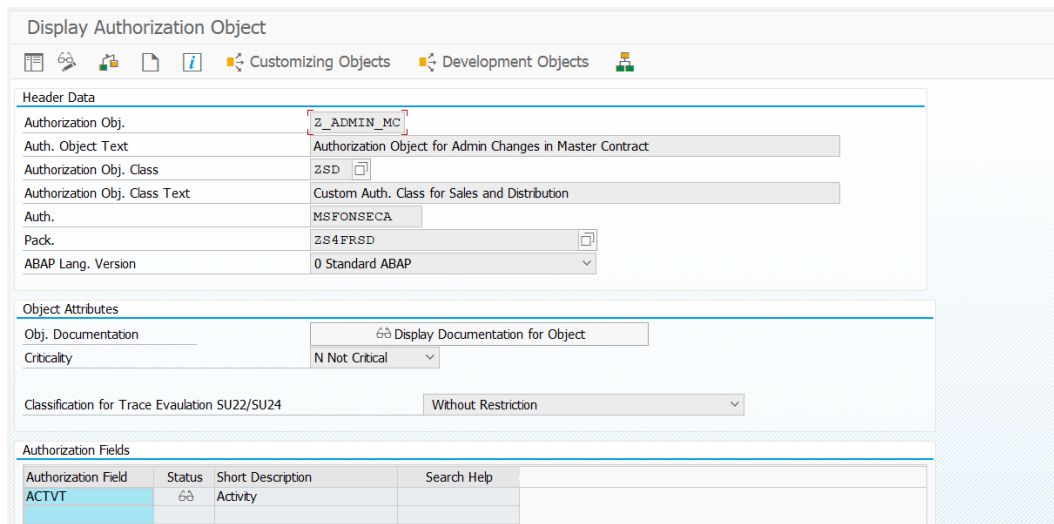


Figura 24 - Objeto de Autorização Z_ADMIN_MC

Depois de criado o objeto de autorização, precisamos de assigná-lo a um role para posteriormente definirmos quais os utilizadores que tem a autorização de customizar as tabelas de eixos. O role definido é ZMCS4_MC_ADMIN (Figura 25).

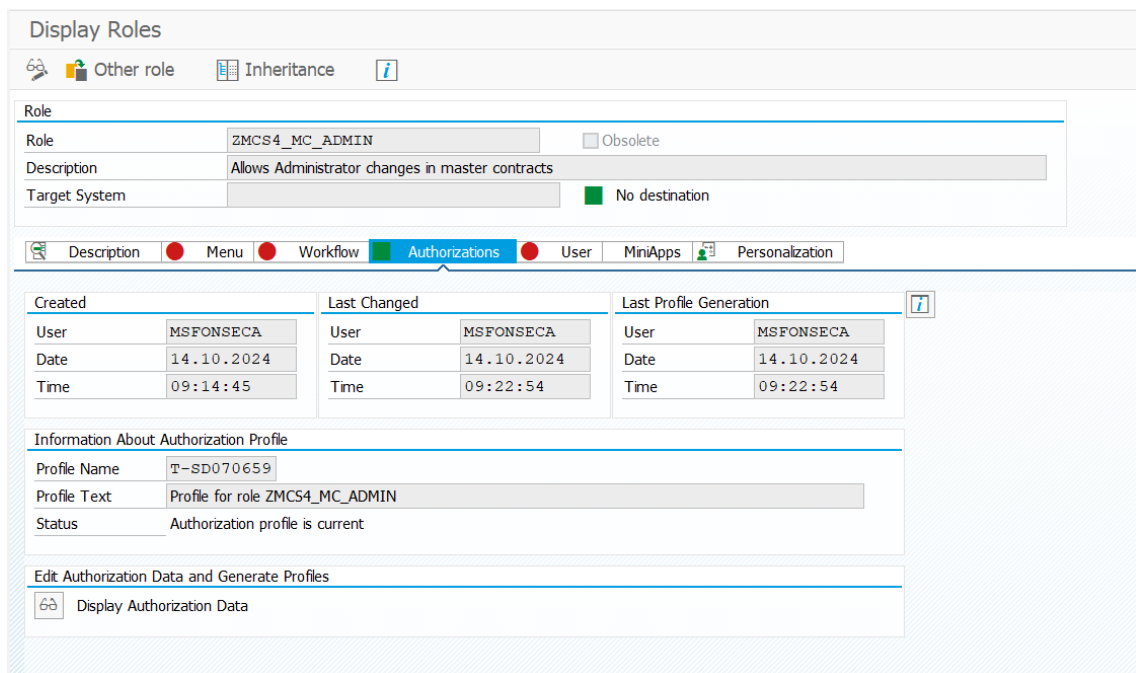


Figura 25 - Role com autorização para alterar as tabelas de eixo

Com os objetos necessários criados, podemos então proceder ao controlo do acesso via código. Para tal usamos o comando abap

Listagem 1 - Código Authority-Check

```

AUTHORITY-CHECK OBJECT 'Z_ADMIN_MC'
  ID 'ACTVT' DUMMY.
  IF sy-subrc = 0.

    selscrtab-dynnr      = 200.
    selscrtab-activetab = 'TAB2'.

  ELSE.

    MESSAGE e031 WITH sy-uname.

  ENDIF.
    
```

No caso do utilizador não tiver permissões de aceder á tab, uma mensagem é despoletada avisando o mesmo de tal situação.

Assim sendo obtemos o seguinte resultado:

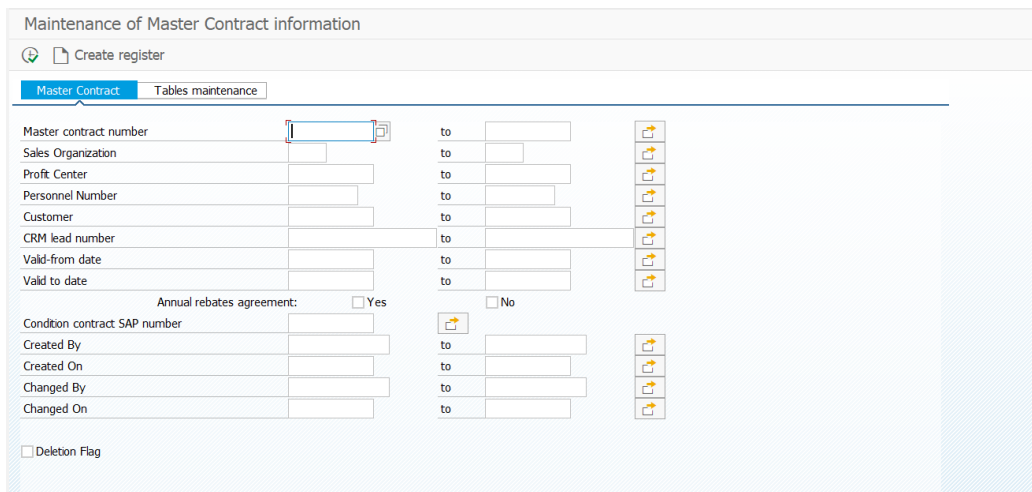


Figura 26 - Ecrã de Seleção da transação ZS4_MASTER_CONTRACT

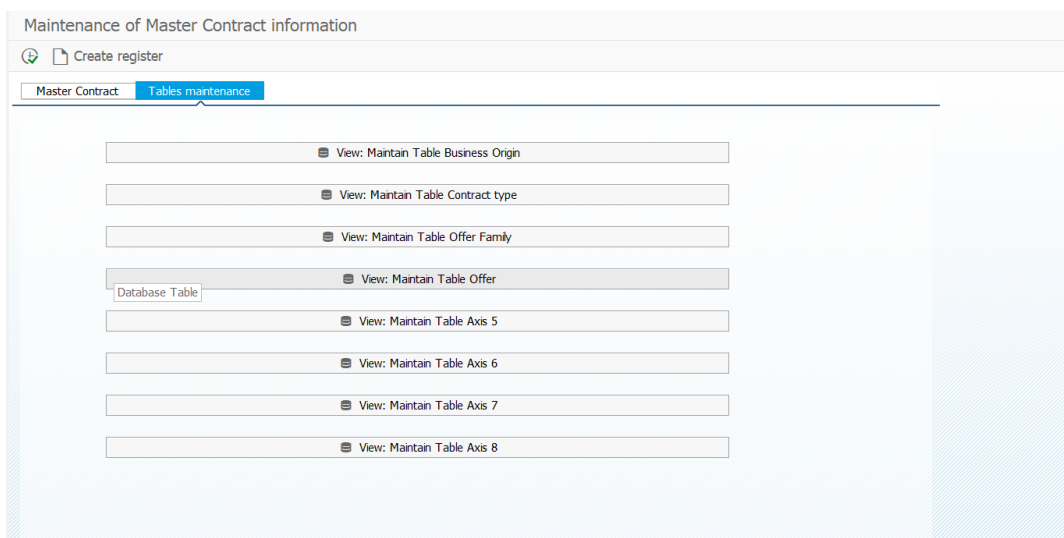


Figura 27 - Tab do ecrã de seleção para manutenção das tabelas de eixos

De realçar que o desenho dos ecrãs é standard, interpretado pelo SAP e dificilmente passível de ser alterado a posição dos campos, apenas a sua ordem de apresentação.

6.2.4. Restantes Includes

Os restantes includes, embora com finalidades diferentes, exercem a mesma função, desenvolver o código fonte necessário para as diversas funcionalidades.

Baseando nos diagramas de atividades, vou agora explicar as principais funcionalidades/fluxo de execução.

Como método de execução primário, temos a listagem de um ALV com os dados dos Master Contract pretendidos. Primeiramente o programa lê os filtros aplicados pelo utilizador no ecrã de seleção e executa uma *query* a base de dados aplicando os mesmos, com especial enfase nos filtros do “Número de Master Contract”, “CRM lead number” e “Annual rebates agreement”.

Depois de obter os dados copia os mesmos para a tabela de visualização, onde contém o campo extra do BOXFD já explicado acima. Antes de fazer o display, procedemos ao tratamento de algumas propriedades para tal. No contexto de ABAP, será o catálogo de campos, definições de layout, entre outras pequenas definições.

6.2.4.1. Método “trata_fieldcatalog”

Neste método procedemos ao tratamento do catálogo de campos. Para tal executamos a função 'LVC_FIELDCATALOG_MERGE' para gerar o catálogo de campos automaticamente a partir da estrutura de dados especificada na variável *k_sname*.

Dentro do loop, uma estrutura CASE é utilizada para verificar o nome do campo (fieldname) e aplicar configurações específicas para cada campo. Por exemplo, para o campo 'BOXFD', a coluna é fixada e excluída da saída, enquanto para 'ZMC_NUMBER', o campo é marcado como chave, fixado, e configurado com opções adicionais como hotspot e textos de cabeçalho.

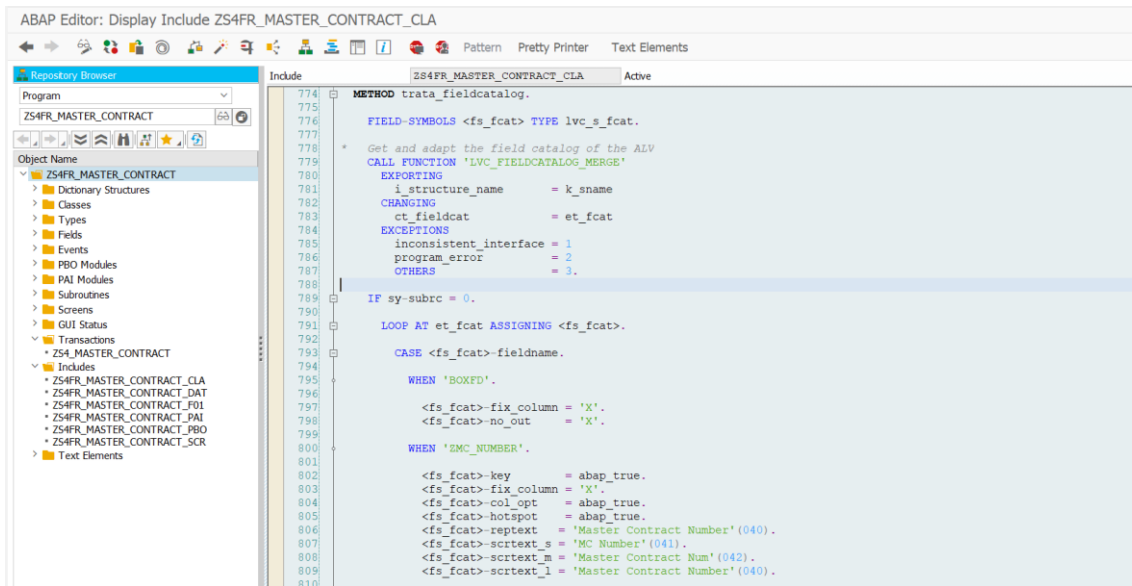


Figura 28 - Método trata_fieldcatalog

6.2.4.2. Método trata_layout

O método trata_layout foca em definir três aspectos específicos do layout do ALV:

es_layout-zebra: Ativa a exibição de linhas zebradas (alternando cores de fundo para as linhas da tabela) para melhorar a legibilidade do relatório. O valor abap_true indica que a funcionalidade está habilitada.

es_layout-box_fname: Define o nome do campo que será utilizado para a caixa de seleção (checkbox) em cada linha do relatório ALV. Neste caso, o campo 'BOXFD' é designado para essa função.

es_layout-sel_mode: Estabelece o modo de seleção das linhas no ALV. O valor 'D' indica que a seleção padrão é desativada, ou seja, o usuário não poderá selecionar linhas no relatório ALV.

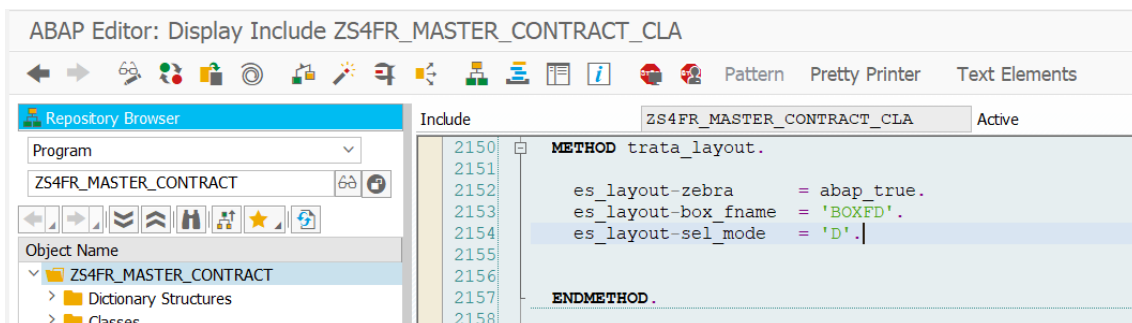


Figura 29 - Método trata_layout

6.2.4.3. Módulo de função

'REUSE_ALV_GRID_DISPLAY_LVC'

A função 'REUSE_ALV_GRID_DISPLAY_LVC' é chamada para exibir a lista ALV (ABAP List Viewer) utilizando os parâmetros fornecidos.

A chamada desta função é essencial para a construção e exibição de uma lista ALV, permitindo a visualização de dados em um formato tabular com diversas funcionalidades interativas e visuais.

Os parâmetros de exportação configuram o comportamento e a aparência da lista ALV:

- **i_callback_program:** Especifica o programa que contém as rotinas de callback para eventos do ALV, como v_repid.
- **i_callback_pf_status_set:** Define a rotina de callback para ajustar o status da GUI (Interface Gráfica do Usuário), neste caso, 'PF_STATUS_SET'.
- **i_callback_user_command:** Indica a rotina de callback para comandos do usuário, aqui representada por 'USER_COMMAND'.
- **i_structure_name:** Nome da estrutura de dados (k_sname) que corresponde às colunas da lista ALV.
- **i_grid_title:** Título do ALV, contido na variável v_title.
- **is_layout_lvc:** Estrutura que contém as configurações de layout do ALV, referenciada por ls_layout.
- **it_fieldcat_lvc:** Tabela interna que contém o catálogo de campos (lt_fcat) para definir as colunas do ALV.
- **i_save:** Define se as configurações do usuário para o layout do ALV devem ser salvas, com 'A' indicando que serão salvas automaticamente.

O parâmetro de tabela t_outtab passa os dados a serem exibidos na lista ALV, representados pela tabela interna gt_alv. Esta tabela foi alimentada previamente pelos dados selecionados a partir do ecrã de seleção.

6.2.4.4. Ecrã de Visualização 0300

O ecrã 0300, como já foi explicado, serve o propósito de mostrar informações dum Master Contract. Para tal foi desenhado com os campos necessários e a lógica para o preenchimento dos mesmos. Esta lógica encontra-se no PBO, Process Before Output.

Para interpretar as ações do utilizador, foi necessário desenvolver o PAI, Process After Input. No ecrã, o utilizador pode fechar a janela ou entrar no modo edição, onde alguns campos ficam passíveis de ser editados mediante o nível de autorização do utilizador.

Em contexto, PBO é um evento que é disparado antes de uma tela ser exibida ao utilizador. Durante o PBO, o sistema prepara os dados que serão mostrados na tela. Isso inclui inicializar ou atualizar campos da tela, esconder ou mostrar elementos da tela com base em certas condições, e realizar cálculos ou determinar dados adicionais que precisam ser apresentados.

O PAI é acionado após o utilizador realizar uma ação que requer processamento, como inserir dados num campo de entrada, selecionar uma opção de um menu ou clicar num botão. Quando o evento PAI ocorre, o sistema processa as entradas do utilizador. Isso pode envolver a validação dos dados inseridos, atualização de registos na base de dados, chamada de outras telas ou relatórios, e a execução de lógica de negócios complexa.

Como existe a possibilidade de entrar em modo de edição e esta é semelhante á do ecrã de modificação, identificado com o número 0500, irei abordar essa funcionalidade no próximo tópico.

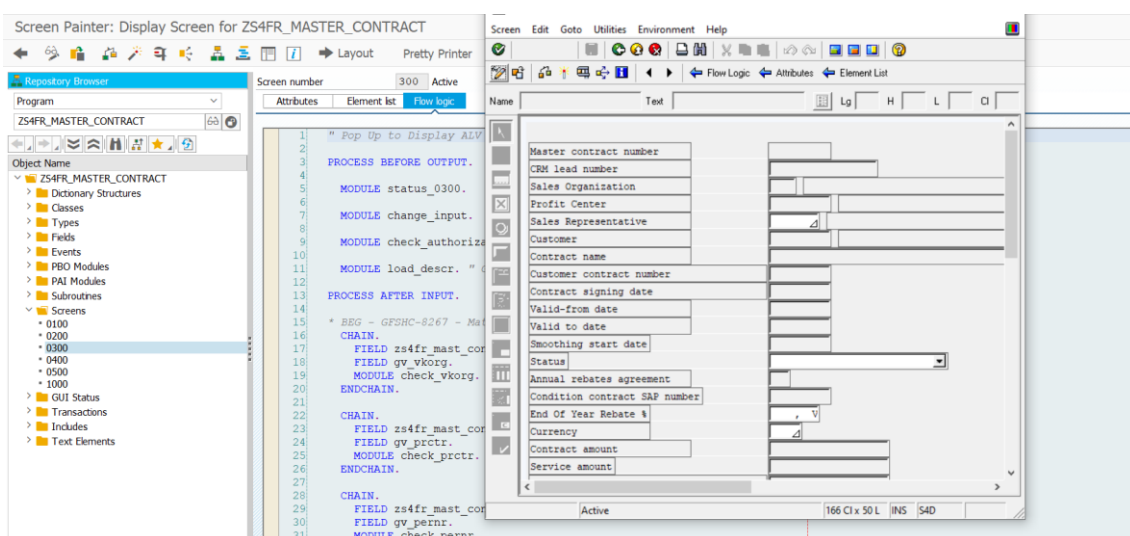


Figura 32 - Ecrã 0300 para visualização do Master Contract

6.2.4.5. Ecrã de Modificação 0500

Outra funcionalidade que o programa principal oferece é a possibilidade de alterar um registo específico através do hotspot do ALV. Este ecrã foi construído baseado no ecrã de visualização, mas iniciado com os campos em modo de edição.

Para este modo de edição o cliente pediu mais uma vez que houvesse um certo nível de rigor em quem poderia editar e quais campos a editar. Para isso foi necessário criar mais um objeto de autorização, Z_MAINT_MC, e o respetivo role ZMCS4_MC_AMOUNTS.

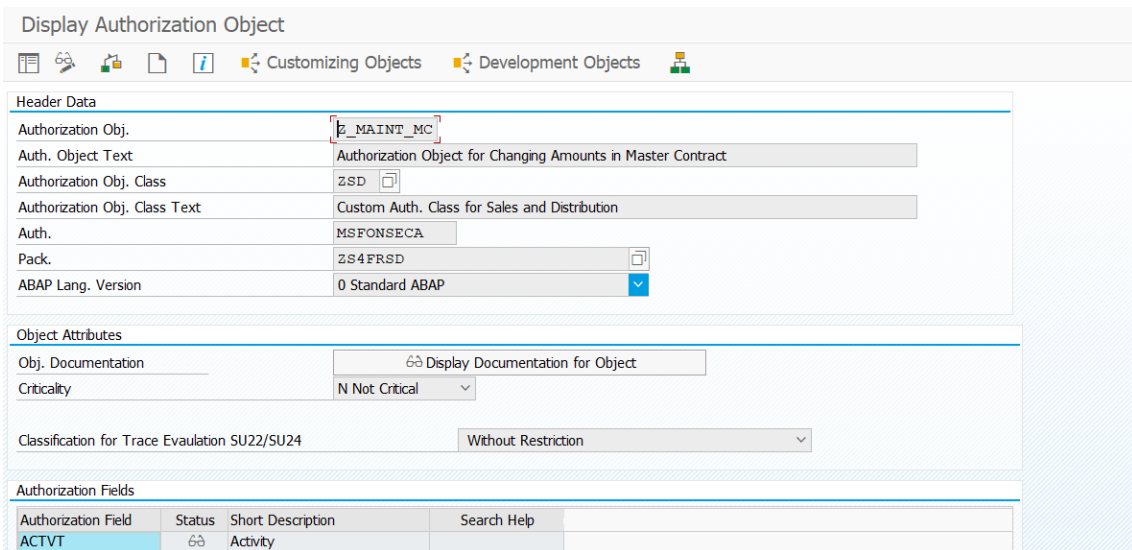


Figura 33 - Objeto de Autorização Z_MAINT_MC

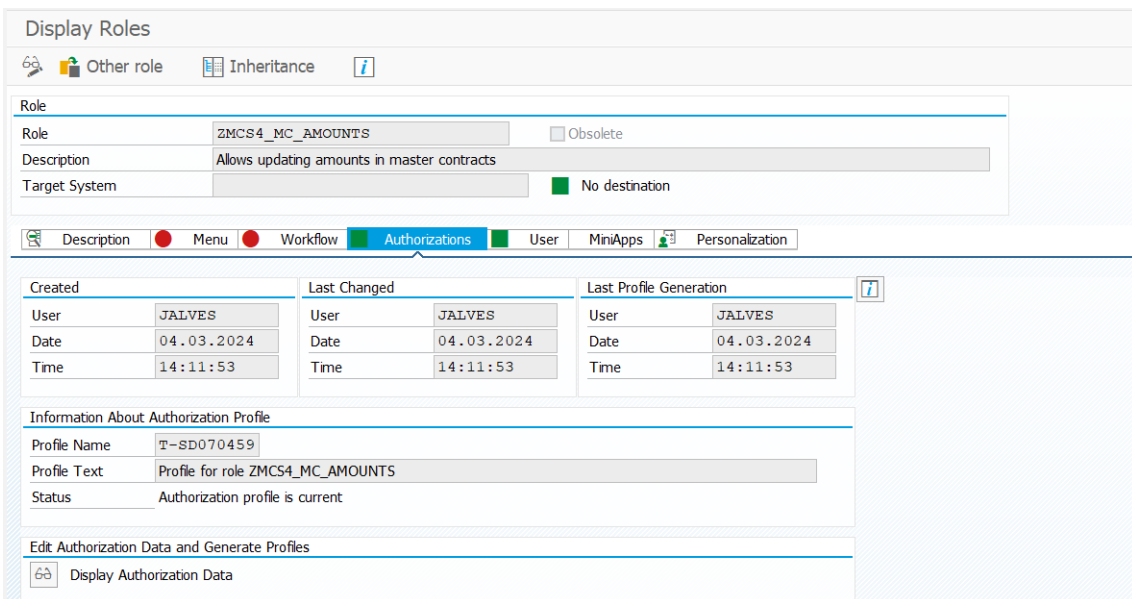


Figura 34 - Role ZMCS4_MC_AMOUNTS

Assim sendo, temos 3 níveis para editar:

- Nível 1, sem qualquer role, apenas pode editar campos básicos como textos descritivos, datas...;
- Nível 2, com role ZMCS4_MC_AMOUNTS, pode editar tudo o nível 1 consegue e alguns campos referentes a montantes, mas dados como centro de custo, vendedor, comprador, nome do contrato ainda continuam inacessíveis para edição;
- Nível 3, com role ZMCS4_MC_ADMIN, pode editar todos os campos à exceção da organização comercial;

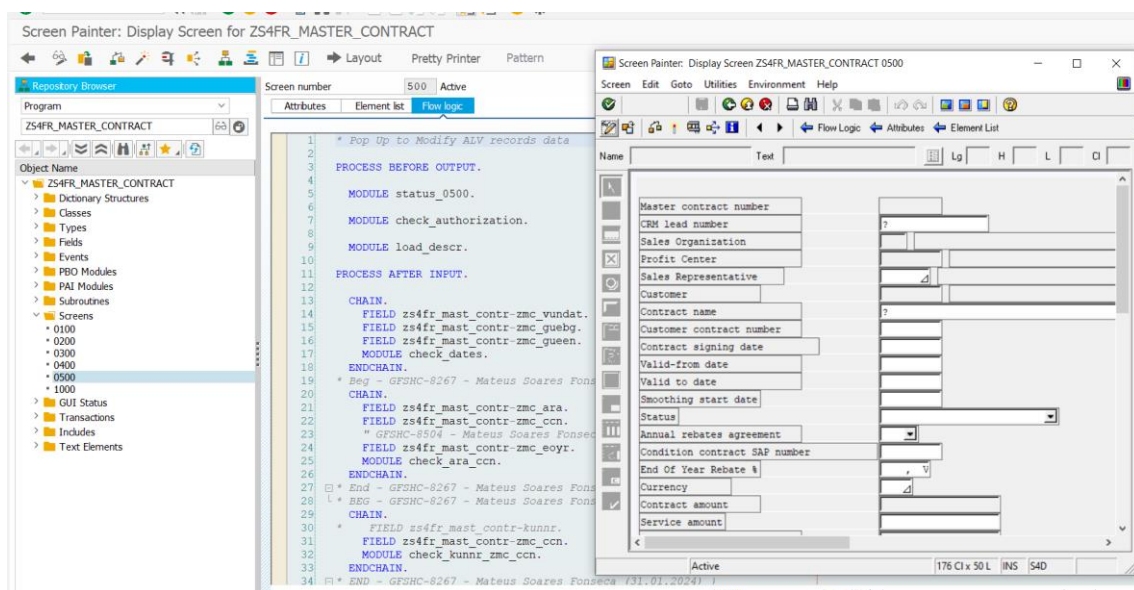


Figura 35 - Ecrã 0500 para modificação dum Master Contract

Tanto no ecrã de modificação como no de visualização, a ação de modificar um registo é processada no método `update_data`.

Este opera da seguinte maneira;

- Extrai o registo original para uma estrutura temporária;
- Extrai o registo atual com as novas atualizações, devidamente validadas;
- Verifica se existem diferenças entre os dois registos para confirmar se é necessária uma atualização á base de dados;
- Se sim, tenta modificar o registo específico com recurso á chave primário Número de Master Contract;
- Em caso de sucesso, despoletar mensagem e dar *commit* á base de dados;
- Em caso de erro, despoletar mensagem e dar *rollback* á base de dados;

Para comparar os registos campo a campo, recorreu-se ao módulo de função 'CTVB_COMPARE_TABLES'. Este compara os registos fornecidos, os campos chave, e retorna o registo diferente, quais campos diferem e em que valor.

6.2.4.6. Ecrã de Criação 0400

Finalmente, a ferramenta também possibilita a criação manual dum Master Contract, ao carregar no botão “Create Register” no ecrã de seleção. Semelhante aos dois ecrãs anteriores, o ecrã de criação também apresenta os campos a preencher, inicializados a vazio. No processo de criação não foi requerido o controlo por autorizações.

No entanto, alguns dos dados tem de ser validados para estarem em conformidade uns com os outros. Mais especificamente a organização comercial, o centro de custo, o vendedor e o cliente. As validações foram fornecidas pelo negócio e aplicadas no código tanto a este ecrã como aos ecrãs anteriores de visualização e edição. Assim como no ecrã de modificação, depois de validadas as entradas com as regras do negócio, o sistema tenta inserir o registo na base dados. Se tiver sucesso, executa o commit á base de dados e manda a mensagem de sucesso o número do novo Master Contract. Se não obtiver sucesso, faz rollback e despoleta a mensagem de erro.

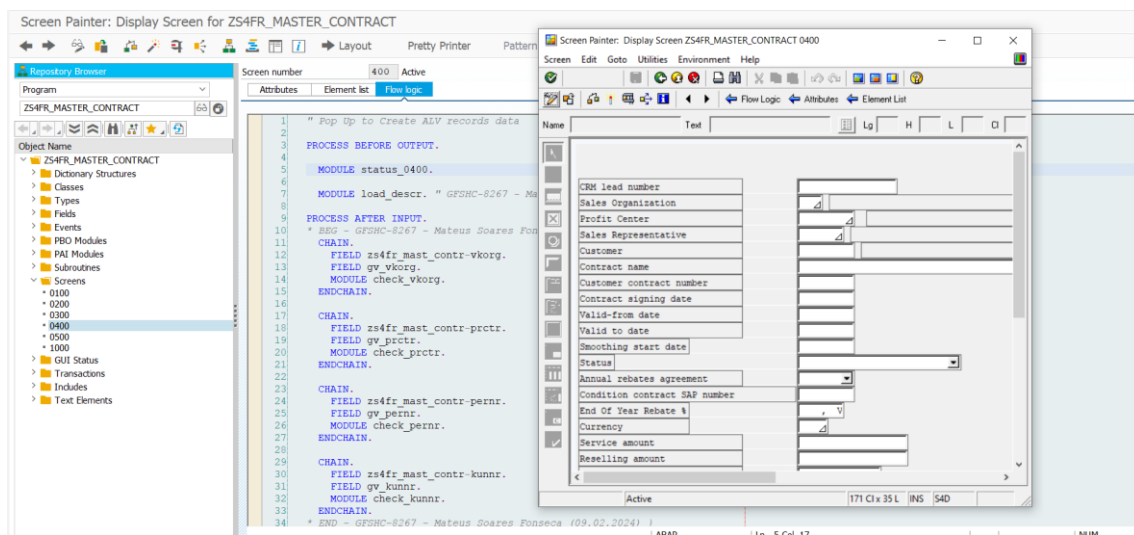


Figura 36 - Ecrã 0400 para criação dum Master Contract

6.3. Adaptação das Transações Standard

Após desenvolvido o programa principal, e ter os elementos do dicionário de dados bem modificados, no contexto deste capítulo as tabelas VBAK e VBAP, procedeu-se então a integração da nova ferramenta nas transações standard VA01/02/03 e VA41/42/43.

Esta nova modificação tem de ser feitas nos ecrãs que a SAP disponibiliza aos desenvolvedores para tal efeito. Após uma pesquisa, chegou-se á conclusão que seria na tab “Additional Data B”, tanto a nível de cabeçalho como a nível de item.

Embora seja a mesma tab, esta está desenhada em dois sub-ecrãs diferentes, um para o cabeçalho e outro para o item. Para o ecrã do cabeçalho, o ecrã é o 8309 (Figura 37), e para o ecrã a nível de item é o 8359 (Figura 38).

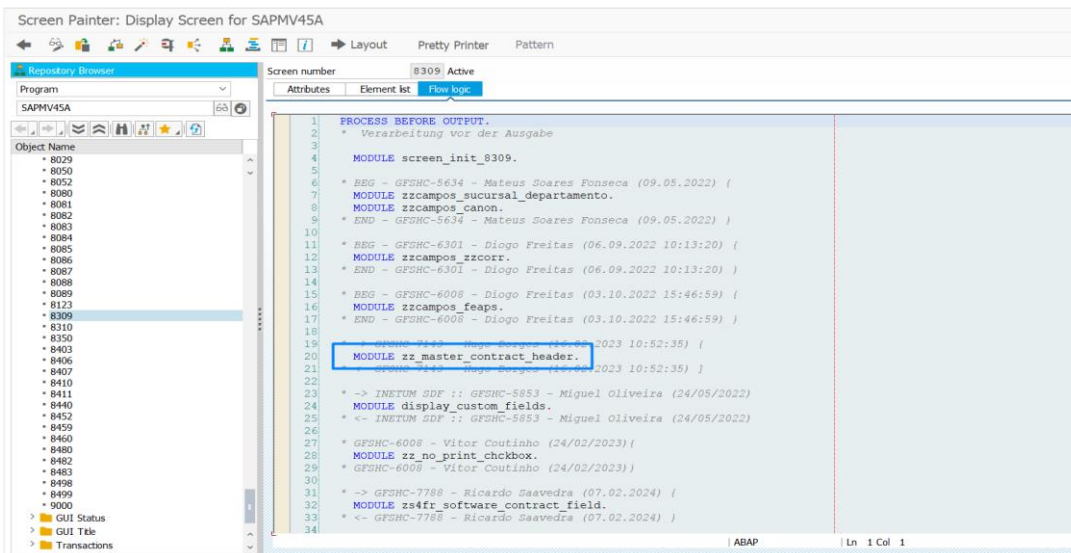


Figura 37 - Ecrã 8309 para o Master Contract a nível de cabeçalho

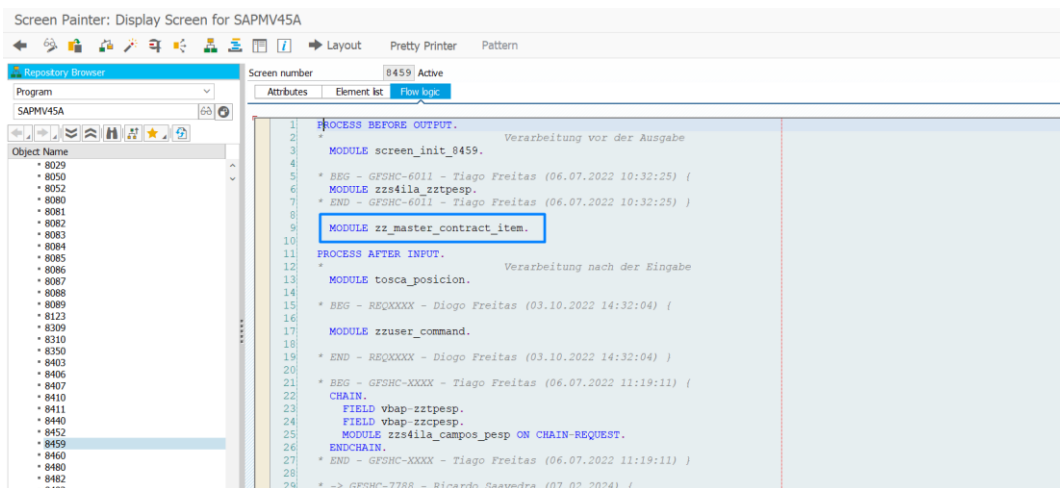


Figura 38 - Ecrã 8459 para o Master Contract a nível de item

Em ambos foram adicionados os campos que o negócio achou pertinente mostrar, e foi também desenvolvida a lógica no PBO de cada um dos ecrãs.

No PBO de ambos os ecrãs, o processamento é semelhante, alternando apenas as tabelas donde os dados são obtidos. Lá é validado se o tipo de contrato standard permite assinatura de um Master Contract, se os campos estão disponíveis para edição ou não dependendo da transação, atualizar os campos consoante são editados, e preenche automaticamente as descrições necessários.

É importante realçar os seguintes pontos:

- Assegurar a coerência dos dados na base de dados;
- Ter atenção nas flags de atualização que são obrigatórias ou então as atualizações não serão registadas;

Para cobrir este dois pontos precisamos atualizar a flag UPRKZ na tabela VBAK/VBAP (U para atualizar o registo do contrato mestre já assignado ou I para assignar um contrato mestre novo) como se pode ver na Figura 39.

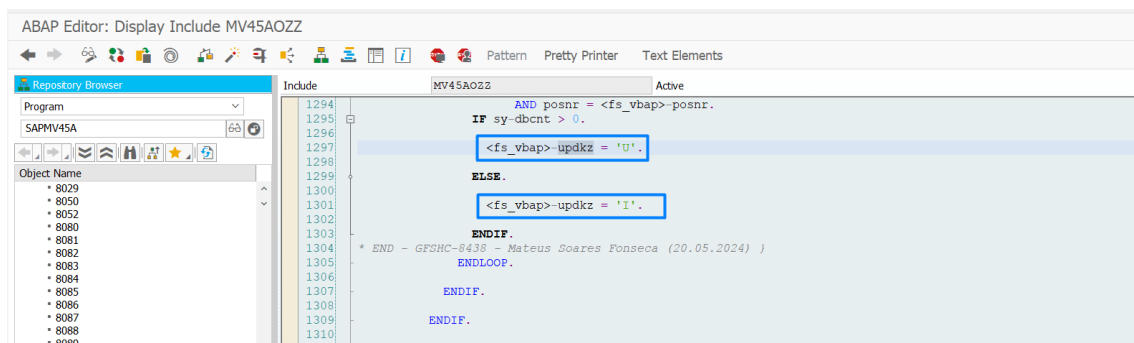


Figura 39 - Atualização da flag para submeter alterações á tabela standard

Já para evitar a existência de erros de execução, no fim de processamento do PBO foi necessário alinhar as tabelas temporárias standard com os dados a atualizar como pode ser vista na Figura 40.

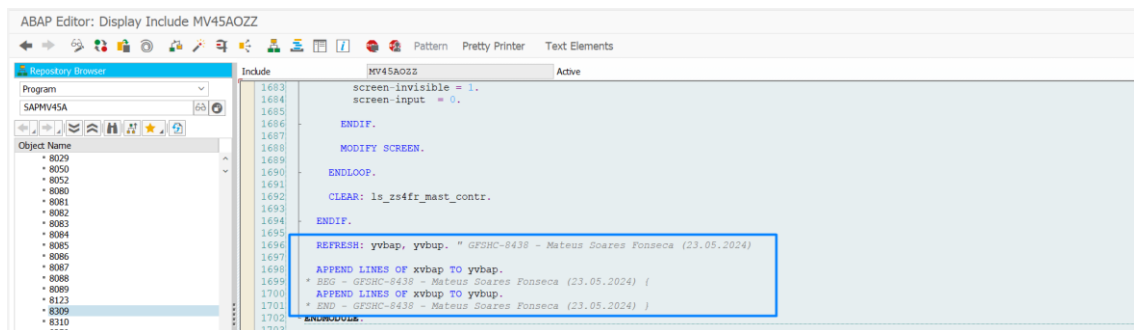


Figura 40 - Alinhamento das tabelas standard

6.4. Implementação de Rotinas Standard VOFM

As rotinas VOFM são um conjunto de funcionalidades no sistema SAP que permitem a personalização de processos de negócio, sobretudo na área de vendas e distribuição. Estas rotinas são utilizadas para adaptar cálculos de preços, impostos e outras operações comerciais específicas, de acordo com as necessidades individuais de cada empresa.

Desenvolvidas em ABAP, as rotinas VOFM oferecem uma forma de ajustar o sistema sem alterar o código padrão, assegurando flexibilidade e conformidade com requisitos comerciais e regulatórios.

No contexto da ferramenta do Master Contract, esta alteração será necessária para quando o cliente decidir criar ordens de venda/contratos por referência a outro contrato/ordem de venda já existente ou então se desejar faturar esse mesmo contrato/ordem de venda.

A informação do Master Contract, se existente, deverá ser transferida para a ordem de venda/contrato ou fatura precedente.

Para tal foram identificadas 5 rotinas standard que precisam ser adulteradas:

- 051 - General header data
- 052 - Billing doc.header
- 151 - General item data
- 153 - Item from bill.doc.
- 601 - Header data copy

Embora sejam rotinas diferentes usadas em ocasiões distintas, como são aplicadas no mesmo tipo de objetos, a sua constituição em termos de tabelas e estruturas é semelhante, salvo a rotina 052 onde é preciso usar variáveis guardadas na stack de memória para aplicar algumas validações.

Considerando os pontos anteriores, a lógica a ser aplicada nas rotinas é semelhante nas 5 e pode ser vista como uma transferência direta de dados como ilustrado na Figura 41.

```

ABAP Editor: Display Include FV45C051
Enhancement Implementation ZS4FR_MASTERCONTRACTUPDATECPRL Active
169 ENHANCEMENT 3 ZS4FR_MASTERCONTRACTUPDATECPRL. "active version
170
171 SELECT COUNT(*)
172 FROM tvko
173 WHERE vkorg = cvbak-vkorg
174 AND vkokl = 'FR'.
175
176 IF sy-dbcnt GT 0.
177
178 vbak-zzmc_number = cvbak-zzmc_number.
179 vbak-zzmc_lead = cvbak-zzmc_lead.
180 vbak-zzmc_ccn = cvbak-zzmc_ccn. " GFSHC-8267 - Mateus Soares Fonseca (11.01.2024)
181 * BEG - GFSHC-8504 - Mateus Soares Fonseca (11.07.2024) (
182 vbak-zzmc_eoyr = cvbak-zzmc_eoyr.
183 vbak-zzmc_status = cvbak-zzmc_status.
184 * END GFSHC-8504 - Mateus Soares Fonseca (11.07.2024) )
185 vbak-zzaxis01_code = cvbak-zzaxis01_code.
186 vbak-zzaxis01_desc = cvbak-zzaxis01_desc.
187 vbak-zzaxis02_code = cvbak-zzaxis02_code.
188 vbak-zzaxis02_desc = cvbak-zzaxis02_desc.
189 vbak-zzaxis03_code = cvbak-zzaxis03_code. " GFSHC-8504 - Mateus Soares Fonseca (11.07.2024)
190 vbak-zzaxis03_desc = cvbak-zzaxis03_desc. " GFSHC-8504 - Mateus Soares Fonseca (11.07.2024)
191 vbak-zzaxis04_code = cvbak-zzaxis04_code.
192 vbak-zzaxis04_desc = cvbak-zzaxis04_desc.
193 ** vbak-zzaxis05_code = cvbak-zzaxis05_code. " GFSHC-8504 - Mateus Soares Fonseca (11.07.2024)
194 ** vbak-zzaxis05_desc = cvbak-zzaxis05_desc. " GFSHC-8504 - Mateus Soares Fonseca (11.07.2024)
195 vbak-zzaxis06_code = cvbak-zzaxis06_code.
196 vbak-zzaxis06_desc = cvbak-zzaxis06_desc.
197 vbak-zzaxis07_code = cvbak-zzaxis07_code.
198 vbak-zzaxis07_desc = cvbak-zzaxis07_desc.
199 vbak-zzaxis08_code = cvbak-zzaxis08_code.
200 vbak-zzaxis08_desc = cvbak-zzaxis08_desc.
201 vbak-zzaxis09_code = cvbak-zzaxis09_code.
202 vbak-zzaxis09_desc = cvbak-zzaxis09_desc.
203 vbak-zzaxis10_code = cvbak-zzaxis10_code.
204 vbak-zzaxis10_desc = cvbak-zzaxis10_desc.
205
    
```

Figura 41 - Exemplo de implementação de rotina VOFM (051)

Toas as melhorias feitas às rotinas standard da VOFM estão guardas num *Enhancement Implementation*.

Enhancement Spot	Program	Overwrite	Enhancement Implementation Type	Enhancement Implementation Point/Section	FULL_NAME of Superior Enhancement
SAPFV45C			Dynamic Enhancement Point/Section	VPR:SAPFV45C/FO:DATEN_KOPIEREN_601/SE:END/IEI	
SAPFV45C			Dynamic Enhancement Point/Section	VPR:SAPFV45C/FO:DATEN_KOPIEREN_151/SE:END/IEI	
SAPFV45C			Dynamic Enhancement Point/Section	VPR:SAPFV45C/FO:DATEN_KOPIEREN_051/SE:END/IEI	
SAPFV45C			Dynamic Enhancement Point/Section	VPR:SAPFV45C/FO:DATEN_KOPIEREN_052/SE:END/IEI	
SAPFV45C			Dynamic Enhancement Point/Section	VPR:SAPFV45C/FO:DATEN_KOPIEREN_153/SE:END/IEI	

Figura 42 - Melhoria de Implementação que guarda as alterações às rotinas da VOFM

6.5. Programa de criação em massa de Master Contracts

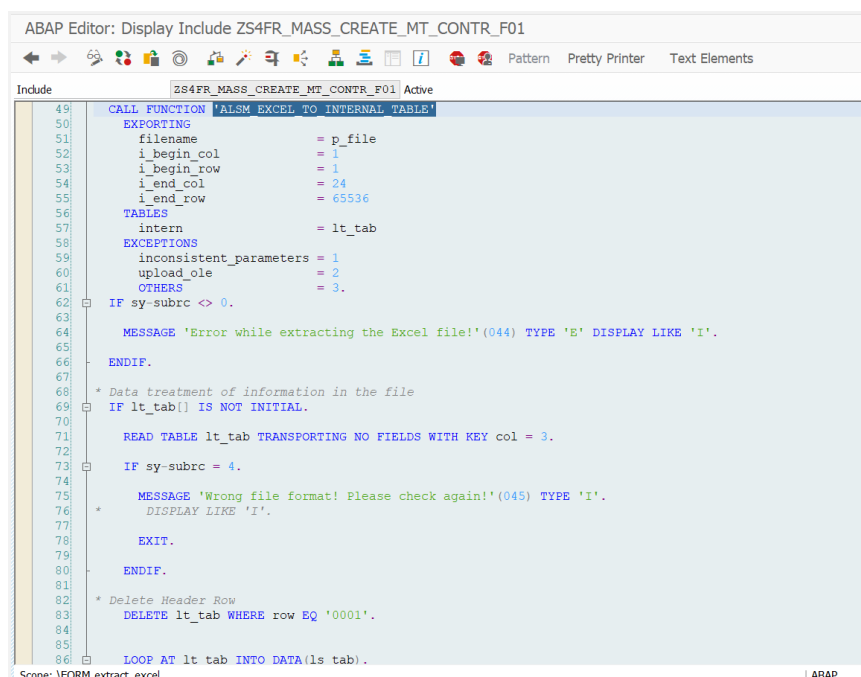
De modo a automatizar o processo de criação de Master Contracts, o negócio pediu também a criação de um programa que permitisse a leitura dum ficheiro Excel onde seriam carregados diversos Master Contracts a ser criados.

Para suprimir tal necessidade, foi criado um programa de criação em massa de Master Contracts. O programa lê o ficheiro Excel como o cliente pretende, faz as mesmas validações que são aplicadas aquando da criação manual e no fim retorna um relatório do resultado da operação. Neste programa foram usadas duas abordagens diferentes ao display dum ALV.

Para mostrar o resultado de sucesso foi adotado o processo normal usando o módulo de função 'REUSE_ALV_GRID_DISPLAY' para um simples *display* em lista. Para apresentar os resultados de erro foi usado o módulo de função 'REUSE_ALV_HIERSEQ_LIST_DISPLAY' pois há a necessidade de ter um *display* em árvore. Neste, o nodo principal seria a informação do Master Contract que não foi possível criar e o nodo secundário seria a tabela de mensagens com os erros responsáveis pela falha. Para interligar os nodos optei por usar uma chave que representava a linha do Excel.

O programa começa por apresentar o ecrã de seleção onde podemos abrir uma janela de pesquisa e escolher o ficheiro que pretendemos usar como fonte da informação dos novos Master Contract. Depois de ter o Excel, o programa extrai o mesmo para uma tabela interna através do módulo de função 'ALSM_EXCEL_TO_INTERNAL_TABLE'.

Este extrai o Excel e preenche uma tabela interna previamente declarada. Para usar o módulo de função temos de fornecer o nome do ficheiro (guardado na variável do ecrã de seleção), o número de colunas e linhas máximo e a tabela de destino.



```

ABAP Editor: Display Include ZS4FR_MASS_CREATE_MT_CONTR_F01
ZS4FR_MASS_CREATE_MT_CONTR_F01 Active
Include
49: CALL FUNCTION 'ALSM_EXCEL_TO_INTERNAL_TABLE'
50: EXPORTING
51:   filename           = p_file
52:   i_begin_col       = 1
53:   i_begin_row       = 1
54:   i_end_col         = 24
55:   i_end_row         = 65536
56: TABLES
57:   intern             = lt_tab
58: EXCEPTIONS
59:   inconsistent_parameters = 1
60:   upload_ole           = 2
61:   OTHERS               = 3.
62: IF sy-subrc <> 0.
63:
64:   MESSAGE 'Error while extracting the Excel file!'(044) TYPE 'E' DISPLAY LIKE 'I'.
65:
66: ENDIF.
67:
68: * Data treatment of information in the file
69: IF lt_tab[] IS NOT INITIAL.
70:
71:   READ TABLE lt_tab TRANSPORTING NO FIELDS WITH KEY col = 3.
72:
73:   IF sy-subrc = 4.
74:
75:     MESSAGE 'Wrong file format! Please check again!'(045) TYPE 'I'.
76:     * DISPLAY LIKE 'I'.
77:
78:     EXIT.
79:
80:   ENDIF.
81:
82: * Delete Header Row
83:   DELETE lt_tab WHERE row EQ '0001'.
84:
85:
86: LOOP AT lt_tab INTO DATA(ls_tab).

```

Figura 43 - Uso do módulo de função 'ALSM_EXCEL_TO_INTERNAL_TABLE'

Depois de extraída e tratada a informação do Excel para uma tabela interna, inicializamos o objeto standard de log para registrar, visualizar e guardar no sistema para futura consulta os logs de execução do programa. No contexto do nosso programa o nome do objeto é “ZS4FR_CRT_MC” e do subobjecto é” ZS4FR_CRT_MC” e podem ser criados via transação SLG0 e consultados via transação SLG1.

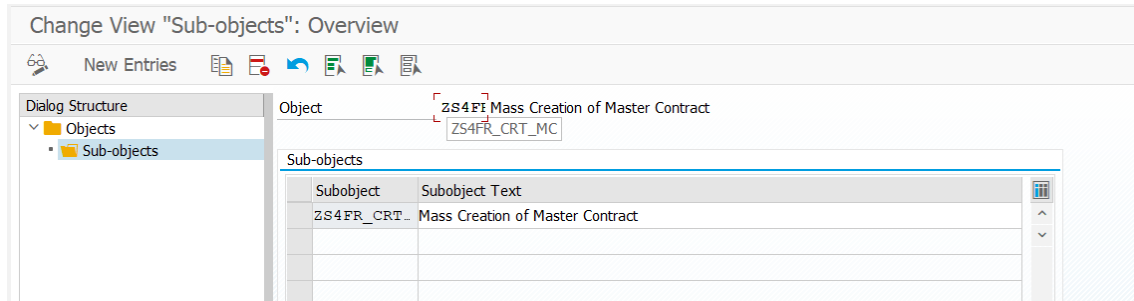


Figura 44 - Objetos de Log criados na SLG0

Para inicializar os mesmo recorremos ao módulo de função standard 'BAL_LOG_CREATE' onde fornecemos os nomes do objeto e subobjecto, declarados como constantes no include das variáveis globais do programa.

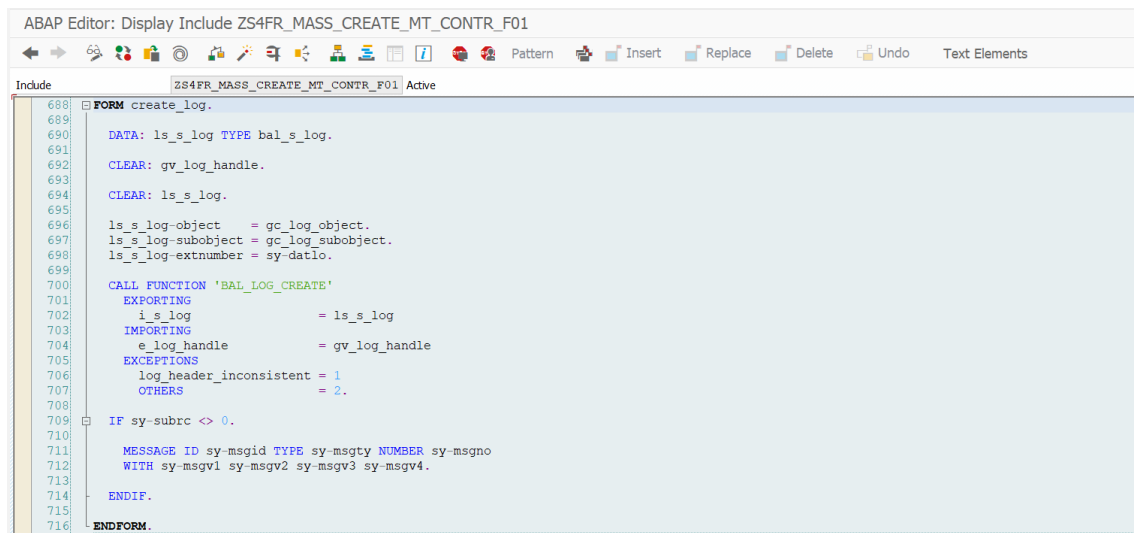


Figura 45 - Inicialização do objeto de Log

Feito isto, pode-se começar a processar a informação dos Master Contract a serem criados. Para tal iteramos a tabela com a informação extraída e aplicamos as mesmas validações que são usadas na criação manual, a nível de código de empresa, centro de custo, vendedor, comprador, datas, etc.... Qualquer erro que surja é guardado na tabela das mensagens de erro com o identificativo da linha e também adicionado ao log. No fim de validar todos os dados, e se não forem encontrados erros, adicionamos o registo á tabela do ALV de sucesso para mostrar nos registos de sucesso, e modificamos a base de dados com a inserção do novo registo. Se der erro em alguma validação, acumula-se as mensagens de erro e cancela-se a inserção na base de dados.

No fim de processar todos os registos, sugeri ao cliente para que aprovasse uma nova funcionalidade. A minha proposta é gerar um ficheiro Excel novo, com a mesma estrutura do ficheiro de input, mas apenas com os registos de erro, para que possam mais facilmente serem identificados e corrigidos. O nome do ficheiro também será

diferente com a adição das *strings* “-MC_error”, data e hora de execução. O cliente aprovou e agradeceu a sugestão.

Para tal foi necessário extrair o nome do ficheiro inicial e adicionar a nova terminação, construir o header semelhante ao original, e realizar o download para o mesmo caminho do Excel de origem sem que a execução do programa fosse interrompida.

Essa implementação foi possível com o código da Figura 46

```

ABAP Editor: Display Include ZS4FR_MASS_CREATE_MT_CONTR_F01
Include ZS4FR_MASS_CREATE_MT_CONTR_F01 Active
797 FORM save_excel.
798
799 DATA: lt_path_components TYPE TABLE OF string,
800       lv_file_name       TYPE string,
801       lv_file_name2     TYPE string,
802       lv_file_nameaux   TYPE string,
803       lv_new_file_name  TYPE string,
804       lv_file_ext       TYPE string.
805
806 " Split the file path
807 SPLIT p_file AT '\' INTO TABLE lt_path_components.
808
809 " Extract the file name with extension
810 READ TABLE lt_path_components INTO lv_file_name2 INDEX lines( lt_path_components ).
811
812 " Check if the file name was extracted successfully
813 IF sy-subrc = 0.
814   " Split the file name and extension
815   " Check if it is a reprocessed file
816   SPLIT lv_file_name2 AT '.' INTO lv_file_nameaux lv_file_ext.
817   SPLIT lv_file_nameaux AT '-' INTO lv_file_name lv_file_name2.
818
819
820 " Concatenate the new file name with the original file extension
821 CONCATENATE lv_file_name 'MC_error' sy-datum ' ' sy-zeitit ' .XLS' INTO lv_new_file_name. " lv_file_ext
822
823
824
825 * Build Excel Header
826 PERFORM build_header.
827
828 CALL FUNCTION 'GUI_DOWNLOAD'
829 EXPORTING
830   filename           = lv_new_file_name
831   filetype           = 'DBF'
832   TABLES
833     data_tab         = gt_tab_error
834     fieldnames       = lt_head

```

Figura 46 - Geração do ficheiro Excel com entradas de erro

Posteriormente, uma vez que está terminado o processamento dos dados, salvamos e fechamos o objeto de log através dos módulos de função 'BAL_LOG_MSG_ADD' e 'BAL_DB_SAVE'.

```

ABAP Editor: Display Include ZS4FR_MASS_CREATE_MT_CONTR_F01
Include ZS4FR_MASS_CREATE_MT_CONTR_F01 Active
717 *%-----*
718 | *% Form handle_log
719 | *%-----*
720 FORM handle_log.
721
722 LOOP AT gt_messages INTO gs_messages.
723
724 CALL FUNCTION 'BAL_LOG_MSG_ADD'
725 EXPORTING
726   i_log_handle = gv_log_handle
727   i_s_msg      = gs_messages
728 EXCEPTIONS
729   log_not_found = 1
730   msg_inconsistent = 2
731   log_is_full = 3
732   OTHERS = 4.
733
734 ENDMETHOD.
735
736 * Save the LOG
737 PERFORM save_log.
738
739 ENDFORM.
740 *%-----*

```

Figura 47 - Módulo de função 'BAL_LOG_MSG_ADD'

```

ABAP Editor: Display Include ZS4FR_MASS_CREATE_MT_CONTR_F01
Include ZS4FR_MASS_CREATE_MT_CONTR_F01 Active
740 | *~
741 | *~ Form save_log
742 | *~
743 | FORM save_log.
744 |
745 | APPEND gv_log_handle TO gt_handle.
746 |
747 | CALL FUNCTION 'BAL_DB_SAVE'
748 | EXPORTING
749 |   i_client      = sy-mandt
750 |   i_in_update_task = ' '
751 |   i_save_all    = 'X'
752 |   i_t_log_handle = gt_handle
753 | EXCEPTIONS
754 |   log_not_found      = 1
755 |   save_not_allowed  = 2
756 |   numbering_error   = 3
757 |   OTHERS            = 4.
758 |
759 | IF sy-subrc <> 0.
760 |
761 |   MESSAGE ID sy-msgid TYPE sy-msgty NUMBER sy-msgno
762 |     WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4.
763 |
764 | ENDIF.
765 |
766 | ENDFORM.
767 | *~

```

Figura 48 - Módulo de função 'BAL_DB_SAVE'

Por fim, chamo o sub-ecrã 0100 que criei para mostrar estatísticas dos resultados da execução. Nele podemos ver quantos registos se tentou criar, quantos tiveram sucesso e quantos obtiveram erro. O ecrã tem o layout demonstrado na Figura 49.

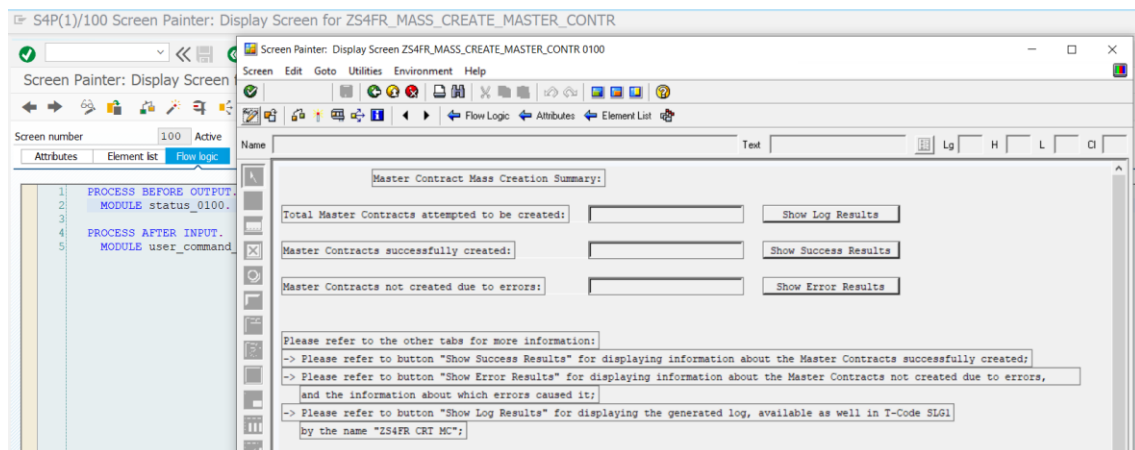


Figura 49 - Layout do ecrã 0100

Neste ecrã há a possibilidade de navegar para mais três subecrãs dependendo do botão que o utilizador pressionar:

- Botão “Show Log Results”: usa o módulo de função standard 'BAL_DSP_LOG_DISPLAY' para mostrar o objeto de log criado;

```

ABAP Editor: Display Include ZS4FR_MASS_CREATE_MT_CONTR_F01
Include ZS4FR_MASS_CREATE_MT_CONTR_F01 Active
770 FORM display_log.
771
772 CHECK sy-batch IS INITIAL.
773
774 CHECK qt_handle[] IS NOT INITIAL.
775
776 CALL FUNCTION 'BAL_DSP_LOG_DISPLAY'
777 EXPORTING
778   i_t_log_handle      = qt_handle
779 EXCEPTIONS
780   profile_inconsistent = 1
781   internal_error      = 2
782   no_data_available   = 3
783   no_authority        = 4
784   OTHERS              = 5.
785
786 IF sy-subrc <> 0.
787
788 MESSAGE ID sy-msgid TYPE sy-msgty NUMBER sy-msgno
789 WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4.
790
791 ENDIF.
792
793 ENDFORM.
794

```

Figura 50 - Módulo de função 'BAL_DSP_LOG_DISPLAY'

- Botão “Show Success Results”: constrói e mostra um ALV com a tabela dos registos que obtiveram sucesso;

```

ABAP Editor: Display Include ZS4FR_MASS_CREATE_MT_CONTR_F01
Include ZS4FR_MASS_CREATE_MT_CONTR_F01 Active
1360 * Form build_success_alv
1361 *
1362 FORM build_success_alv.
1363
1364 DATA: ls_layout_gr TYPE sls_layout_alv.
1365
1366 ** DATA: alv_grid TYPE REF TO cl_gui_alv_grid,
1367 ** alv_container TYPE REF TO cl_gui_custom_container.
1368 **
1369 *** For Success Display
1370 ** CREATE OBJECT alv_container
1371 ** EXPORTING
1372 **   container_name = 'ERROR_CONTAINER'.
1373 **
1374 ** TRY.
1375 **
1376 **   CREATE OBJECT alv_grid
1377 **   EXPORTING
1378 **     i_parent = alv_container.
1379 **
1380 **   CATCH cx_root.
1381 **
1382 **     MESSAGE 'Error occurred while creating ALV grid' TYPE 'E'.
1383 **
1384 **   ENDTRY.
1385 **
1386 ** For Success Display
1387 ** IF alv_grid IS BOUND.
1388
1389   ls_layout_gr-colwidth_optimize = abap_true.
1390
1391   CALL FUNCTION 'REUSE_ALV_GRID_DISPLAY'
1392   EXPORTING
1393     i_callback_program      = sy-repid
1394     i_callback_user_command = 'USER_COMMAND'
1395     i_structure_name        = 'ZS4FR_MC_MASSCREA_SUCCESS'
1396     i_grid_title            = 'Master Contract Success Log' (020)
1397     ls_layout              = ls_layout_gr

```

Figura 51 - ALV de sucesso

- Botão “Show Error Results”: constrói e despoleta o ALV hierárquico em árvore com os registos de error e as respetivas mensagens. As dependências já foram explicadas previamente (página 55);

```

ABAP Editor: Display Include ZS4FR_MASS_CREATE_MT_CONTR_F01
Include ZS4FR_MASS_CREATE_MT_CONTR_F01 Active
1450 * ls_layout-sebra = abap_true.
1451 *
1452 * Field Catalog
1453 ** lt_fieldcat-fieldname = 'LINENUM'.
1454 ** lt_fieldcat-tabname = 'GT_ALV_ERROR'.
1455 ** lt_fieldcat-seltext_s = 'Ln. Number'.
1456 ** APPEND lt_fieldcat.
1457 *
1458 * Table Connection
1459 ls_keyinfo-header01 = 'LINENUM'.
1460 ls_keyinfo-item01 = 'LINENUM'.
1461 *
1462 CALL FUNCTION 'REUSE_ALV_HIERSEQ_LIST_DISPLAY'
1463 EXPORTING
1464   i_callback_program      = sy-repid
1465   i_callback_user_command = 'USER_COMMAND'
1466   is_layout              = ls_layout
1467 **   lt_fieldcat           = lt_fieldcat
1468   i_tabname_header       = 'GT_ALV_ERROR'
1469   i_tabname_item         = 'GT_ERROR_TXT'
1470   i_structure_name_header = 'ZS4FR_MC_MASSCREA_ERROR'
1471   i_structure_name_item  = 'ZS4FR_MC_MASSCREA_ERROR_TXT'
1472   is_keyinfo            = ls_keyinfo
1473 TABLES
1474   t_outtab_header       = gt_alv_error
1475   t_outtab_item         = gt_error_txt
1476 EXCEPTIONS
1477   program_error        = 1
1478   OTHERS                = 2.
1479
1480 IF sy-subrc <> 0.
1481 * Implement suitable error handling here
1482 ENDF.
1483 ** ENDF.
1484

```

Figura 52 - ALV Hierárquico em Árvore de Erros

6.6. Programa de assinatura em massa de Master Contracts

Semelhante á criação em massa, o negócio também achou pertinente poder assinar vários Master Contract a contratos standard através dum ficheiro Excel.

Assim como no programa de criação em massa, o funcionamento pretendido é conseguir ler um ficheiro Excel, que no contexto específico trará o Número de Oportunidade CRM e o Número da Ordem de Venda/Contrato, validar se a assinatura é possível dependendo de certos campos e no fim retornar um log

O objeto para o log da assinatura em massa é ZS4FR_ASSIGNCONTRACT, como ilustrado na Figura 53.

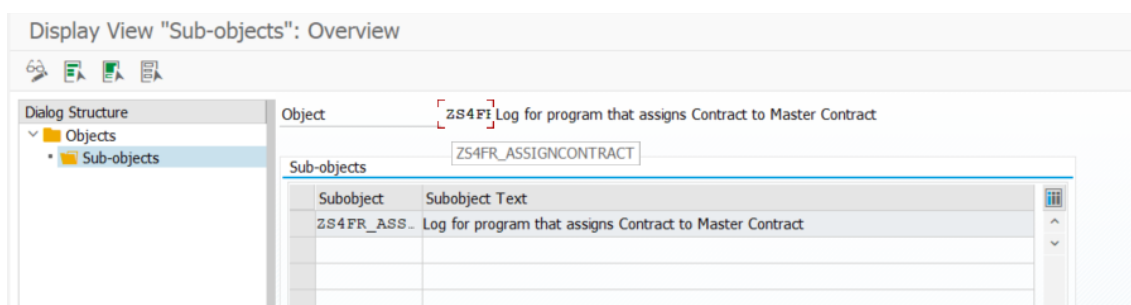


Figura 53 - Objeto de log para o programa de assinatura em massa

Os módulos de função para inicialização, guardar, fechar e visualizar são os mesmos usados no programa anterior. A diferença é que o relatório final será somente e obrigatoriamente o log.

A forma de extrair a informação do Excel é semelhante também, usando o módulo de função 'ALSM_EXCEL_TO_INTERNAL_TABLE'.

Depois de extraída e tratada a informação do Excel para uma tabela interna, iteramos a mesma e validamos os dados, como por exemplo se a Organização Comercial é compatível ou o Grupo de Vendas é mesmo nos dois contratos, mestre e standard.

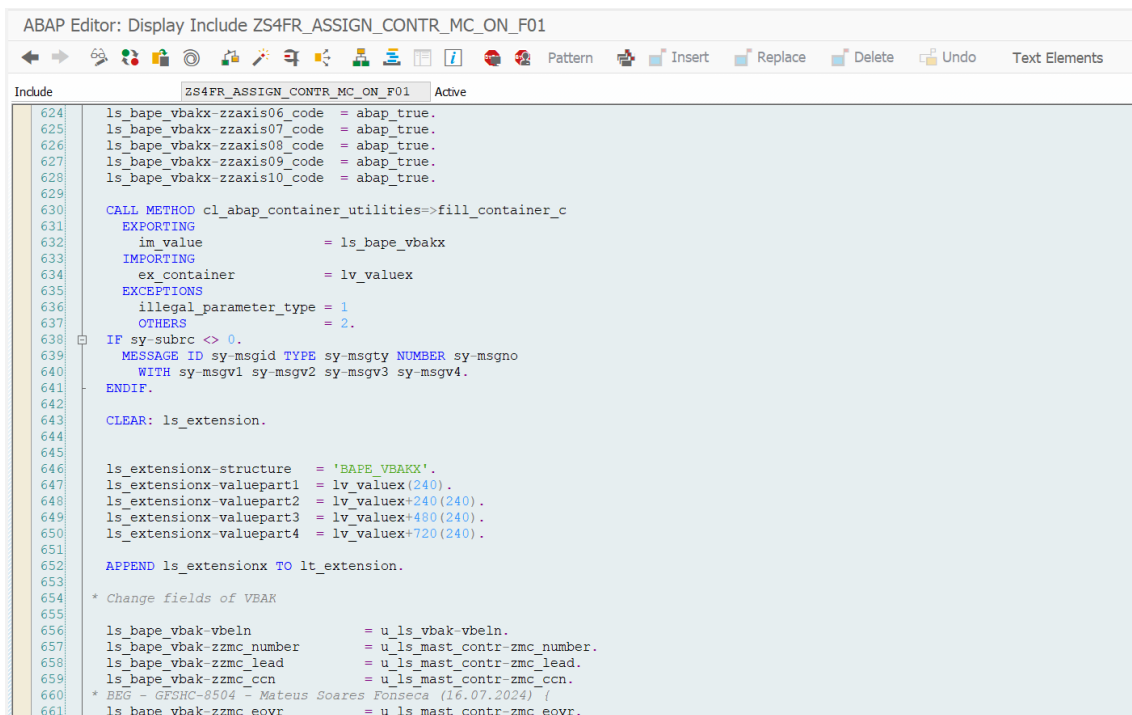
Feitas estas validações, passamos á fase mais crítica do processo do programa, a assinatura em si. Como não é um processo comum nem o sistema foi desenhado para tal, esta operação é muito sensível e qualquer erro pode corromper a base de dados.

O processo inicia-se com a atribuição de dados aos campos pertinentes do cabeçalho e dos itens de um documento de vendas/contrato, no nosso caso os campos essenciais do

Master Contract, utilizando para tal as estruturas BAPE_VBAK, BAPE_VBAKX, BAPE_VBAP e BAPE_VBAPX. As estruturas com terminação 'X' são empregues para sinalizar quais os campos necessitam de ser atualizados, sendo estes marcados com o valor `abap_true`.

A transformação destas estruturas em formato binário é efetuada recorrendo à classe `cl_abap_container_utilities`, o que facilita a geração de extensões que serão posteriormente enviadas para a BAPI `BAPI_BAPI_SALESORDER_CHANGE`. Esta BAPI tem como função implementar as modificações requeridas no documento de vendas, com base nas extensões previamente preparadas.

Seguidamente à invocação da BAPI, procede-se à verificação de potenciais erros e, caso se verifique erros, é desencadeado um `ROLLBACK` para preservar a integridade dos dados. Caso a operação decorra com sucesso, efetua-se um `COMMIT`.



```

ABAP Editor: Display Include ZS4FR_ASSIGN_CONTR_MC_ON_F01
Include ZS4FR_ASSIGN_CONTR_MC_ON_F01 Active
624 ls_bape_vbakx-zzaxis06_code = abap_true.
625 ls_bape_vbakx-zzaxis07_code = abap_true.
626 ls_bape_vbakx-zzaxis08_code = abap_true.
627 ls_bape_vbakx-zzaxis09_code = abap_true.
628 ls_bape_vbakx-zzaxis10_code = abap_true.
629
630 CALL METHOD cl_abap_container_utilities=>fill_container_c
631 EXPORTING
632   im_value           = ls_bape_vbakx
633 IMPORTING
634   ex_container       = lv_valuex
635 EXCEPTIONS
636   illegal_parameter_type = 1
637   OTHERS              = 2.
638 IF sy-subrc <> 0.
639   MESSAGE ID sy-msgid TYPE sy-msgty NUMBER sy-msgno
640   WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4.
641 ENDIF.
642
643 CLEAR: ls_extension.
644
645
646 ls_extensionx-structure = 'BAPE_VBAKX'.
647 ls_extensionx-valuepart1 = lv_valuex(240).
648 ls_extensionx-valuepart2 = lv_valuex+240(240).
649 ls_extensionx-valuepart3 = lv_valuex+480(240).
650 ls_extensionx-valuepart4 = lv_valuex+720(240).
651
652 APPEND ls_extensionx TO lt_extension.
653
654 * Change fields of VBAK
655
656 ls_bape_vbak-vbeln           = u_ls_vbak-vbeln.
657 ls_bape_vbak-zzmc_number     = u_ls_mast_contr-zmc_number.
658 ls_bape_vbak-zzmc_lead       = u_ls_mast_contr-zmc_lead.
659 ls_bape_vbak-zzmc_ccn        = u_ls_mast_contr-zmc_ccn.
660 * BEG - GFSHC-8504 - Mateus Soares Fonseca (16.07.2024) {
661 ls_bape_vbak-zzmc_eovr       = u_ls_mast_contr-zmc_eovr.

```

Figura 54 - Exemplo de preenchimento do cabeçalho da ordem de venda/contrato

```

ABAP Editor: Display Include ZS4FR_ASSIGN_CONTR_MC_ON_F01
Include ZS4FR_ASSIGN_CONTR_MC_ON_F01 Active
723 ls_bape_vbapx-zzaxis07_code = abap_true.
724 ls_bape_vbapx-zzaxis08_code = abap_true.
725 ls_bape_vbapx-zzaxis09_code = abap_true.
726 ls_bape_vbapx-zzaxis10_code = abap_true.
727
728 CALL METHOD c1_abap_container_utilities=>fill_container_c
729 EXPORTING
730   im_value           = ls_bape_vbapx
731 IMPORTING
732   ex_container      = lv_valuex
733 EXCEPTIONS
734   illegal_parameter_type = 1
735   OTHERS              = 2.
736 IF sy-subrc <> 0.
737   MESSAGE ID sy-msgid TYPE sy-msgty NUMBER sy-msgno
738     WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4.
739 ENDIF.
740
741 CLEAR: ls_extensionx.
742
743 ls_extensionx-structure = 'BAPE_VBAPEX'.
744 ls_extensionx-valuepart1 = lv_valuex(240).
745 ls_extensionx-valuepart2 = lv_valuex+240(240).
746 ls_extensionx-valuepart3 = lv_valuex+480(240).
747 ls_extensionx-valuepart4 = lv_valuex+720(240).
748
749 APPEND ls_extensionx TO lt_extension.
750
751 * Change fields of VBAF
752
753 ls_bape_vbap-vbeln      = ls_vbap-vbeln.
754 ls_bape_vbap-posnr     = ls_vbap-posnr.
755 ls_bape_vbap-zzmc_number = u_ls_mast_contr-zmc_number.
756 ls_bape_vbap-zzmc_lead  = u_ls_mast_contr-zmc_lead.
757 ls_bape_vbap-zzmc_ccn   = u_ls_mast_contr-zmc_ccn.
758 * BEG - GFSHC-8504 - Mateus Soares Fonseca (16.07.2024) /
759 ls_bape_vbap-zzmc_eoyr  = u_ls_mast_contr-zmc_eoyr.
760 ls_bape_vbap-zzmc_status = u_ls_mast_contr-zmc_status.

```

Figura 55 - Exemplo de preenchimento do/s item/s da ordem de venda/contrato

```

ABAP Editor: Display Include ZS4FR_ASSIGN_CONTR_MC_ON_F01
Include ZS4FR_ASSIGN_CONTR_MC_ON_F01 Active
804 CALL FUNCTION 'BAPI_SALESORDER_CHANGE'
805 EXPORTING
806   salesdocument      = u_ls_vbak-vbeln
807 *   order_header_in  =
808   order_header_inx  = ls_headerx
809 * SIMULATION        =
810 * BEHAVE_WHEN_ERROR = ' '
811 * INT_NUMBER_ASSIGNMENT = ' '
812 * LOGIC_SWITCH      =
813 * NO_STATUS_BUF_INIT = ' '
814 TABLES
815   return      = gt_messages
816   order_item_in = lt_items
817 *   order_item_inx =
818 * PARTNERS        =
819 * PARTNERCHANGES =
820 * PARTNERADDRESSES =
821 * ORDER_CFGS_REF  =
822 * ORDER_CFGS_INST =
823 * ORDER_CFGS_PART_OF =
824 * ORDER_CFGS_VALUE =
825 * ORDER_CFGS_ELOB  =
826 * ORDER_CFGS_VR    =
827 * ORDER_CFGS_REFINST =
828 * SCHEDULE_LINES  =
829 * SCHEDULE_LINESX =
830 * ORDER_TEXT       =
831 * ORDER_KEYS       =
832 * CONDITIONS_IN    =
833 * CONDITIONS_INX   =
834   extensionin     = lt_extension.
835 *   extensionx     = lt_extensionx.
836 *   NFMETALLIMS    =
837
838 READ TABLE gt_messages INTO gs_messages WITH KEY type = 'E'.
839
840 IF sy-subrc = 0.
841 * BEG - GFSHC-8504 - Mateus Soares Fonseca (01.10.2024) /

```

Figura 56 - Uso do Módulo de Função 'BAPI_SALESORDER_CHANGE'

6.7. Desenvolvimento do Módulo de Função ZS4FR_WS_MASTER_CONTRACT que alimentará o Webservice

Por fim, foi requerido que fosse possível á equipa de CRM criar Master Contracts via webservice baseado em SOAP. De modo a suportar o webservice, foi necessário desenvolver um simples, mas robusto módulo de função que será executado em background quando o webservice for chamado.

O webservice recebe um payload com a informação do Master Contract a ser criado que será traduzido numa estrutura de importação ao módulo de função. O código faz então validações sobre os campos fornecidos, verificando se os dados necessários foram preenchidos corretamente. Posteriormente, faz uma seleção á tabela do Master Contract para encontrar o último contrato mestre criado e assim adicionar um número para encontrar o novo master contract.

Depois de termos o novo número do Master Contract, os dados da estrutura de input são transferidos para uma estrutura temporária e depois tenta inserir o novo registo na base de dados. Se a operação teve sucesso, registamos uma mensagem de sucesso na variável de exportação lv_message e o número do Master Contract criado na variável de exportação lv_mc_number.

Estas variáveis de exportação são importantes pois representam o retorno do módulo de função que serão transferidas para o webservice e conseqüentemente devolvidas ao cliente, nomeadamente á equipa de CRM.

No caso de erro, apenas a variável lv_message será preenchida por uma mensagem de erro.

```

4  ** IMPORTING
5  **   VALUE(IS_INPUT) TYPE ZS4FR_WS_MASTER_CONTRACT
6  ** EXPORTING
7  **   VALUE(E_MESSAGE) TYPE STRING
8  **   VALUE(MASTER_CONTRACT_NUMBER) TYPE ZS4FR_MC_NUMBER
9  ** TABLES
10 **   ET_RETURN TYPE BAPIRET2_T OPTIONAL
11 **-----
12 * DEVELOPER.....: INETUM-SDF - Hugo Borges *
13 * DATE.....: 13.04.2023 *
14 * DESCRIPTION...: INETUM - S4 HANA Cloud JIRA - (GFSHC-7515) *
15 *-----
16
17 CLEAR: gs_master_contract.
18
19 * Check if we have all the mandatory fields filled
20 PERFORM check_mandatory_fields TABLES et_return
21                                CHANGING is_input
22                                e_message.
23
24 * Create new Master contract number
25 PERFORM get_master_contract_data USING is_input.
26
27 * Pass all the values to global structure
28 PERFORM add_reg_table TABLES et_return
29                                USING is_input.
30
31 * Update the register in the table
32 PERFORM update_table TABLES et_return
33                                CHANGING e_message
34                                master_contract_number.
35
36 ENDFUNCTION.
    
```

Figura 57 - Módulo de Função ZS4FR_WS_MASTER_CONTRACT

Depois de criado o módulo de função foi necessário criar o webservice em si e criar a binding com o WSDL via transação SOAMANAGER.

O webservice cria-se via transação SE37, a mesma usada para a criação do módulo de função, mas através da opção de utilidades extra como ilustrado na Figura 58.

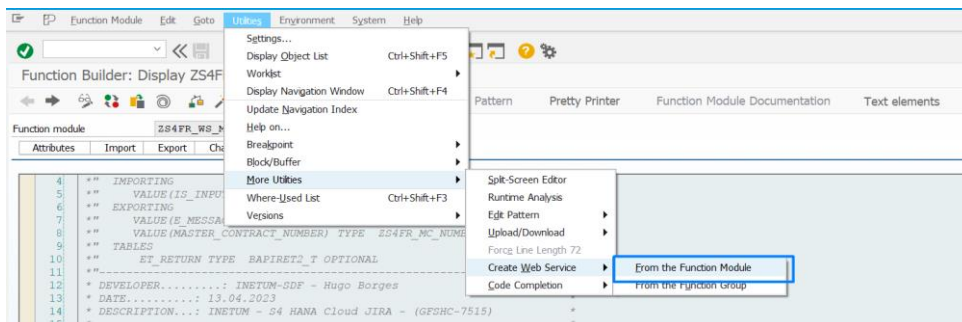


Figura 58 - Criação do Webservice

Na Figura 59 vemos o webservice definido.

Display Service Definition ZS4FR_WS_MASTER_CONTRACT											
Service Definition: ZS4FR_WS_MASTER_CONTRACT Active											
<table border="1"> <thead> <tr> <th colspan="2">Identification</th> </tr> </thead> <tbody> <tr> <td>Technical Name</td> <td>ZS4FR_WS_MASTER_CONTRACT</td> </tr> <tr> <td>Porttype Name</td> <td>ZS4FR_WS_MASTER_CONTRACT</td> </tr> <tr> <td>Short Description</td> <td>ZS4FR_WS_MASTER_CONTRACT</td> </tr> <tr> <td>SOAP Namespace</td> <td>urn:sap-com:document:sap:rfc:functions</td> </tr> </tbody> </table>		Identification		Technical Name	ZS4FR_WS_MASTER_CONTRACT	Porttype Name	ZS4FR_WS_MASTER_CONTRACT	Short Description	ZS4FR_WS_MASTER_CONTRACT	SOAP Namespace	urn:sap-com:document:sap:rfc:functions
Identification											
Technical Name	ZS4FR_WS_MASTER_CONTRACT										
Porttype Name	ZS4FR_WS_MASTER_CONTRACT										
Short Description	ZS4FR_WS_MASTER_CONTRACT										
SOAP Namespace	urn:sap-com:document:sap:rfc:functions										
<table border="1"> <thead> <tr> <th colspan="2">Endpoint</th> </tr> </thead> <tbody> <tr> <td>Endpoint Type</td> <td>Function Module</td> </tr> <tr> <td>Referenced Object</td> <td>ZS4FR_WS_MASTER_CONTRACT</td> </tr> </tbody> </table>		Endpoint		Endpoint Type	Function Module	Referenced Object	ZS4FR_WS_MASTER_CONTRACT				
Endpoint											
Endpoint Type	Function Module										
Referenced Object	ZS4FR_WS_MASTER_CONTRACT										
<table border="1"> <thead> <tr> <th colspan="2">General Data</th> </tr> </thead> <tbody> <tr> <td>Package</td> <td>ZS4FRSD</td> </tr> <tr> <td>Original Language</td> <td>EN</td> </tr> <tr> <td>Created by</td> <td>HSBORGES on 17.04.2023 16:04:20</td> </tr> <tr> <td>Changed by</td> <td>MSFONSECA on 24.10.2024 15:58:55</td> </tr> </tbody> </table>		General Data		Package	ZS4FRSD	Original Language	EN	Created by	HSBORGES on 17.04.2023 16:04:20	Changed by	MSFONSECA on 24.10.2024 15:58:55
General Data											
Package	ZS4FRSD										
Original Language	EN										
Created by	HSBORGES on 17.04.2023 16:04:20										
Changed by	MSFONSECA on 24.10.2024 15:58:55										

Figura 59 - Webservice ZS4FR_WS_MASTER_CONTRACT

Na Figura 60 podemos ver a definição da binding do webservice ZS4FR_WS_MASTER_CONTRACT configurada no SOAMANAGER. A partir deste ponto o webservice está disponível para ser consumido.

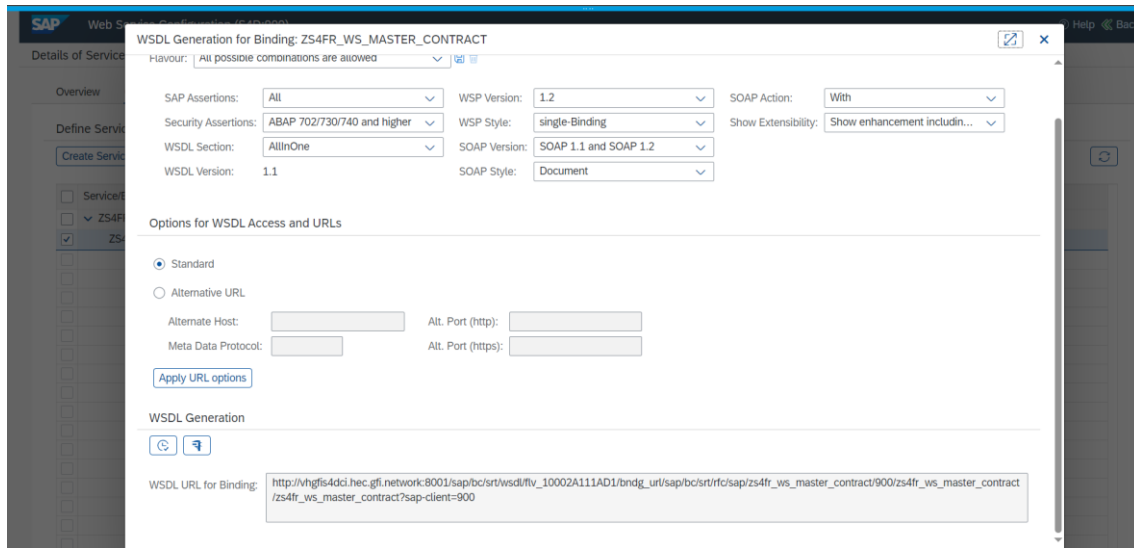


Figura 60 - Definições da binding do webservice via SOAMANAGER

7. Análise e Discussão de Resultados

7.1. Introdução

No capítulo presente serão abordados os testes necessários para comprovar as funcionalidades implementadas, bem como possíveis melhorias a serem desenvolvidas.

Nos subcapítulos seguintes serão então testados os diversos programas e os resultados devidamente analisados. Em suma os contratos mestre terão de ser devidamente criados, modificados, visualizados e assinados corretamente, independente de qual programa usado.

De ressaltar dois pontos:

- Nos testes do Webservice teremos de usar um payload diretamente no módulo de função pois não temos acesso á ferramenta do CRM que chama o webservice;
- Todos os programas já foram testados pela equipa funcional e pelo cliente, com aprovação e feedback positivo;

7.2. Teste do Programa principal do Master Contract ZS4FR_MASTER_CONTRACT

7.2.1. Criação manual dum Master Contract

Neste teste vamos criar um contrato mestre via criação manual (Figura 61). Para tal preenchemos alguns campos e despoletamos a criação.

Field	Value
CRM lead number	FAE1524KOE1
Sales Organization	1100 INETUM
Profit Center	0000311311
Sales Representative	8000103 FREDERIC REYNIER
Customer	1000298 HACHETTE LIVRE
Contract name	TESTE DE CRIAÇÃO MANUAL MC A38912
Contract signing date	29.10.2024
Valid-from date	29.10.2024
Valid to date	29.10.2025
Status	A Draft
Currency	EUR
Service amount	100,00
Reselling amount	50,00
Business Origin	201 NEW BUSINESS
Contract type	2003 SERVICE MANAGED
Offer Family	OF_01 CONSULTING
Offer	O_004 CIO CONSULTING

Figura 61 - Teste da função de criar um Master Contract

O resultado foi a criação do Master Contract 1000002434, o qual comprovaremos no próximo capítulo onde vamos testar a funcionalidade de visualizar (Figura 63 e Figura 64).

7.2.2. Funcionalidade de Visualizar

Neste teste iremos tentar visualizar os contratos mestre com filtro aberto (Figura 62) e um contrato específico (Figura 63).

Maintenance of Master Contract information

Change register Create register

Master Contract Data

Deletion	MC Number	CRM lead number	S.Org.	Profit Center	Pers.No.	Customer	Name 1	Contract name	Cus.Con.Nr	Contr.Date	Valid from	Valid to date	Smth St Dt
	1000002434	FAEIS24KOEt	1100	0000311311	8000103	1000298	HACHETTE LIVRE	TESTE DE CRIAÇÃO MANUAL MC A38912		29.10.2024	29.10.2024	29.10.2025	
	1000002433	FAEIS24KOEt	1100	0000207210	8014476	1000297	NEXT VENTURE LIMITED	TEST DATE LISSAGE CREA WEBSERVICE	R2A41	01.06.2024	01.06.2024	01.06.2026	01.01.2025
	1000002432	FZERKT45292424	1100	0000311311	8005263	1000298	HACHETTE LIVRE	TOTALENERGIES ONETECH	TGP_AC0404	01.06.2024	01.01.2024	01.12.2026	
	1000002431	FZERKT45292424	1100	0000311311	8005263	1000297	NEXT VENTURE LIMITED	TOTALENERGIES ONETECH	TGP_AC0404	01.06.2024	01.01.2024	01.12.2026	01.01.2025
	1000002430	GJER8ddee432R4e24	1100	0000207210	8000103	1000297	NEXT VENTURE LIMITED	TEST NEW FIELD DATE LISSAGE	8357387	11.11.2023	11.11.2023	11.11.2026	01.06.2025
	1000002428	GJER8ddee432R4e24	1100	0000304034	8001443	1000298	HACHETTE LIVRE	NOUVELLE MODIFICATION 5	999999	11.11.2024	11.11.2024	11.11.2033	
	1000002427	GJER8ddee432R4e24	1100	0000207210	8001443	1000298	HACHETTE LIVRE	TEST LOT1 CREATION MODIF	111111	11.11.2024	11.11.2024	11.11.2028	
	1000002426	F44Z20E63	1100	0000311311	8005263	1000297	NEXT VENTURES Ltd	TOTALENERGIES ONETECH	TGP_AC0404	01.06.2019	01.06.2019	31.05.2024	
	1000002425	F44Z20E63	1100	0000204500	8000099	1000298	NEXT VENTURES Ltd	TOTALENERGIES ROLE ADMIN TEST	1111111111	11.11.2023	11.11.2024	11.11.2026	
	1000002424	GJER8ddee42dz3e	1100	0000311311	8000165	1000297	NEXT VENTURES Ltd	SOFTWARE INFOGRANCE	TEST	01.01.2024	01.01.2024	17.07.2028	
	1000002423	GJER8ddee42dz3e	1100	0000311311	8000165	1000297	NEXT VENTURES Ltd	SOFTWARE INFOGRANCE	TEST	01.01.2024	01.01.2024	17.07.2028	
	1000002422	FAEIS38458E	1100	0000311311	8000019	1000235	HP FRANCE SAS	TOTALENERGIES ONETECH	TGP_AC0404	01.06.2019	01.06.2019	31.05.2024	01.06.2024
	1000002421	GJER8ddee432R4e24	1100	0000311311	8000165	1000235	HP FRANCE SAS	SOFTWARE INFOGRANCE	TEST	01.01.2024	01.01.2024	17.07.2028	
	1000002420	GJER8ddee432R4e24	1100	0000204500	8000094	1000297	NEXT VENTURES Ltd	SOFTWARE INFOGRANCE9999	TEST5999	09.09.2024	09.09.2024	09.09.2028	24.10.2025
	1000002419	FAEIS38458E	1100	0000311311	8005263	1000297	NEXT VENTURES Ltd	TOTALENERGIES ONETECH	TGP_AC0404	01.06.2019	01.06.2019	31.05.2024	
	1000002418	356181	1100	0000311311	8005263	1000297	NEXT VENTURES Ltd	TOTALENERGIES ONETECH	TGP_AC0404	01.06.2019	01.06.2019	31.05.2024	
	1000002417	ea99999TTUgaccrFFF	1100	0000207210	8000094	1000298	HACHETTE LIVRE	SOFTWARE INFOGRANCE	TEST	01.01.2024	01.01.2024	31.12.2025	
	1000002416	FAE41222R2424242	1100	0000207210	8000094	1000298	HACHETTE LIVRE	SOFTWARE INFOGRANCE	TEST	01.01.2024	01.01.2024	31.12.2025	
	1000002415	00654000007pAAS	1100	0000322040	8001414	1000008	BNP PARIBAS INTERVENTIONS IMMOBILIE	CR ARA TMA		31.07.2024	01.10.2024	30.09.2024	
	1000002414	TZER3454313534	1100	0000207210	8000018	1000008	BNP PARIBAS INTERVENTIONS IMMOBILIE	TEST		21.09.2026	30.09.2020	10.03.2026	
	1000002413	EZA7123C-8050	1100	0000372703	8000018	1000013	BL Software	ONATI - MAINTENANCE C35 - S1 2023.		19.07.2023	02.08.2023	02.02.2024	
	1000002412	EZA7123C-804933333	1100	0000372703	8000018	1000013	BL Software	NEW NAME OF MC MODIF	9999999999	11.10.2023	11.11.2023	11.11.2024	
	1000002411	EZA7123C-8048	1100	0000372703	8000018	1000013	BL Software	Formation DPF V1		12.09.2023	17.09.2023	17.03.2024	
	1000002410	EZA7123C-8047	1100	0000372703	8000018	1000013	BL Software	IN AMOA S2 2023 (PFU, PJA)		02.10.2023	16.10.2023	16.02.2024	
	1000002409	EZA7123C-8046	1100	0000372703	8000018	1000013	BL Software	AMOA Jazz -Ateler Eng Systems - P3 - 5		09.10.2023	23.10.2023	23.04.2024	
	1000002408	EZA7123C-8045	1100	0000372703	8000018	1000298	HACHETTE LIVRE	AM Hitsau + autres		09.10.2023	02.11.2023	02.03.2024	
	1000002407	EZA7123C-8044	1100	0000372703	8000018	1000298	HACHETTE LIVRE	CP AMOA MES_DLA_Q4 2023 Q1 2024		12.10.2023	13.10.2023	13.04.2024	
	1000002406	EZA7123C-8043	1100	0000372703	8000018	1000298	HACHETTE LIVRE	DOT Cumulocity Q42023 Q12024		09.11.2023	10.11.2023	10.11.2024	
	1000002405	EZA7123C-8040	1100	0000372703	8000018	1000013	BL Software	BODRI remédations sécurité		15.11.2023	15.11.2023	15.02.2024	
	1000002404	EZA7123C-8039	1100	0000372703	8000018	1000013	BL Software	EXPERTISE RÉSEAUX ET SÉCURITÉ - PROLONGA		15.11.2023	03.12.2023	03.07.2024	
	1000002403	EZA7123C-8038	1100	0000372703	8000018	1000013	BL Software	AVIS-PLM-FLX Sécurisation données restrei		21.11.2023	23.11.2023	23.04.2024	
	1000002402	EZA7123C-8037	1100	0000372703	8000018	1000297	NEXT VENTURES Ltd	CP Connexion machines 2023 - 24 - Corbel		21.11.2023	27.11.2023	27.11.2024	
	1000002401	EZA7123C-8036	1100	0000372703	8000018	1000297	NEXT VENTURES Ltd	AMO Resp applications ThingWorx - LBA e		21.11.2023	02.01.2024	02.06.2024	

Figura 62 - Teste da função de visualizar vários Master Contract

54D(1)/900 Maintenance of Master Contract information

Master contract number: 1000002434

CRM lead number: FAEIS24KOEt

Sales Organization: 1100 INETUM

Profit Center: 0000311311

Sales Representative: 8000103 FREDERIC REYNIER

Customer: 1000298 HACHETTE LIVRE

Contract name: TESTE DE CRIAÇÃO MANUAL MC A38912

Customer contract number: [Empty]

Contract signing date: 29.10.2024

Valid-from date: 29.10.2024

Valid to date: 29.10.2025

Smoothing start date: [Empty]

Status: A Draft

Annual rebates agreement: [Empty]

Condition contract SAP number: [Empty]

End Of Year Rebate %: 0,00

Currency: EUR

Contract amount: 150,00

Service amount: 100,00

Reselling amount: 50,00

Number of days sold: 5,00

Planned cost: 0,00

Service planned cost: 0,00

Reselling planned cost: 0,00

Reviewed Contract value TCV: 150,00

Reviewed Service revenue: 100,00

Reselling amount review: 50,00

Number of days sold reviewed: 5,00

Planned cost reviewed: 0,00

Service planned cost reviewed: 0,00

Reselling planned cost review: 0,00

Business Origin: Z01 NEW BUSINESS

Contract type: Z003 SERVICE MANAGED

Offer Family: OF_01 CONSULTING

Continue Cancel

Figura 63 - Teste da função de visualizar um Master Contract

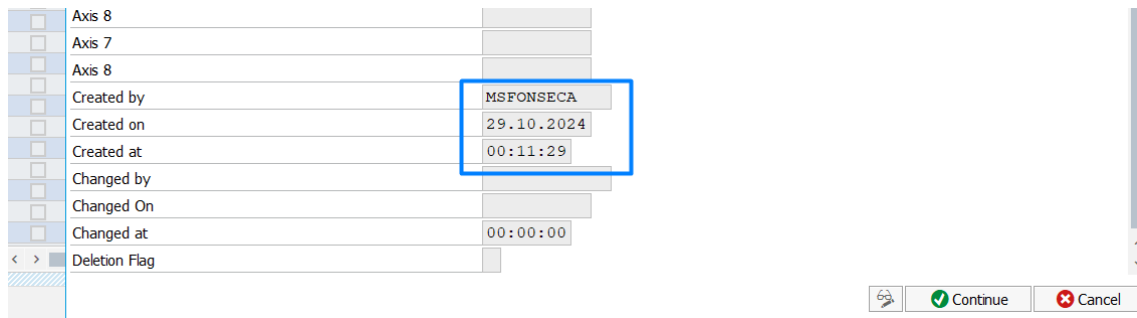


Figura 64 - Continuação do teste da função de visualizar um Master Contract

7.2.3. Funcionalidade de Modificar

Neste teste iremos modificar o Master Contract criado manualmente para os nossos testes (Figura 65 e Figura 66). Para verificarmos que foi devidamente atualizado, o campo de Contract Name será modificado assim como as datas. Os valores da informação relativas á última modificação serão atualizados automaticamente.

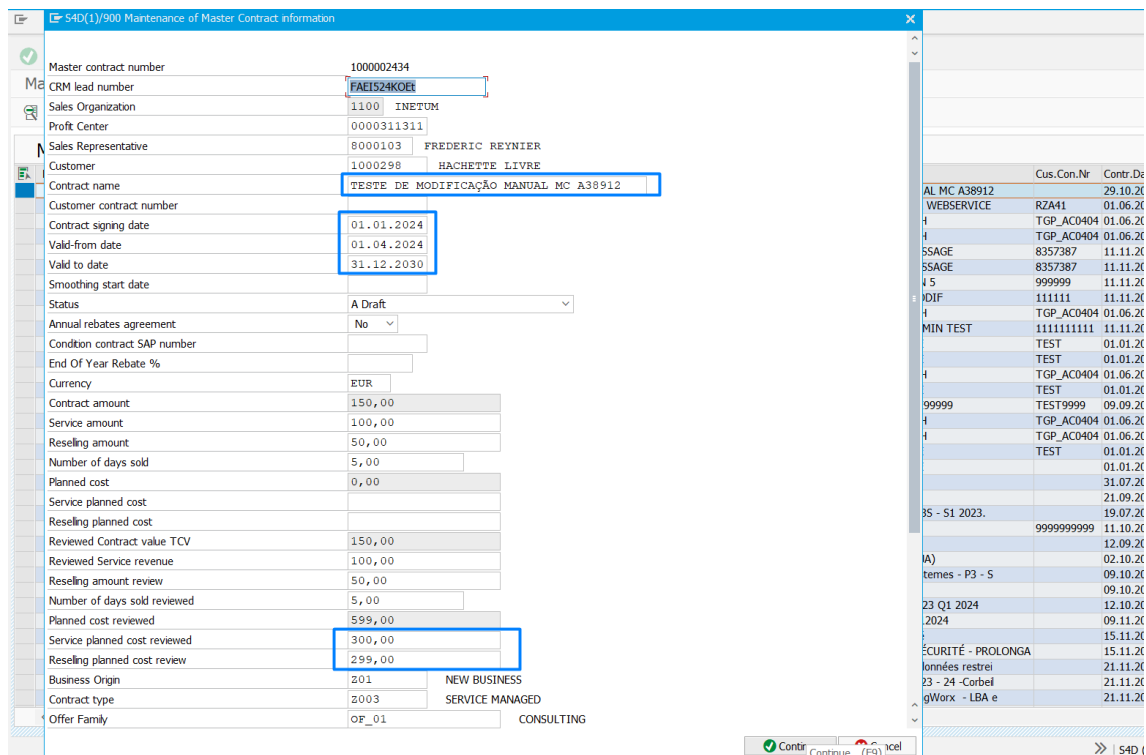


Figura 65 - Teste da função de modificar um Master Contract

Reviewed Contract value TCV	150,00	
Reviewed Service revenue	100,00	
Reselling amount review	50,00	
Number of days sold reviewed	5,00	
Planned cost reviewed	599,00	
Service planned cost reviewed	300,00	
Reselling planned cost review	299,00	
Business Origin	Z01	NEW BUSINESS
Contract type	Z003	SERVICE MANAGED
Offer Family	OF_01	CONSULTING
Offer	O_004	CIO CONSULTING
Axis 7		
Axis 8		
Axis 7		
Axis 8		
Created by	MSFONSECA	
Created on	29.10.2024	
Created at	00:11:29	
Changed by		
Changed On		
Changed at	00:00:00	
Deletion Flag	<input type="checkbox"/>	

DDIT	11111:
H	TGP_A
MIN TEST	11111:
	TEST
	TEST
H	TGP_A
	TEST
99999	TEST9
H	TGP_A
	TEST
35 - 51 2023.	
	99999:
IA)	
termes - P3 - 5	
23 Q1 2024	
2024	
ÉCURITÉ - PROLONGA	
lonnées restrei	
23 - 24 - Corbeil	
gWorx - LBA e	

Continue Cancel

Figura 66 - Continuação do teste da função de modificar um Master Contract

Como podemos ver na Figura 66, os campos “Changed” não são atualizados até submetermos as novas modificações. Assim que guardamos, os campos serão atualizados, tanto os campos alterados manualmente como os campos que registam a última alteração ao contrato mestre.

Vou agora usar a função de visualizar e comprovar a eficácia da funcionalidade de modificar.

Master contract number	1000002434
CRM lead number	FAE1524KOEt
Sales Organization	1100 INETUM
Profit Center	0000311311
Sales Representative	8000103 FREDERIC REYNIER
Customer	1000298 HACHETTE LIVRE
Contract name	TESTE DE MODIFICAÇÃO MANUAL MC A38912
Customer contract number	
Contract signing date	01.01.2024
Valid-from date	01.04.2024
Valid to date	31.12.2030
Smoothing start date	
Status	A Draft
Annual rebates agreement	
Condition contract SAP number	
End Of Year Rebate %	0,00
Currency	EUR
Contract amount	150,00
Service amount	100,00
Reselling amount	50,00
Number of days sold	5,00
Planned cost	0,00
Service planned cost	0,00
Reselling planned cost	0,00
Reviewed Contract value TCV	150,00
Reviewed Service revenue	100,00
Reselling amount review	50,00
Number of days sold reviewed	5,00
Planned cost reviewed	599,00
Service planned cost reviewed	300,00
Reselling planned cost review	299,00
Business Origin	201 NEW BUSINESS

Figura 67 - Registo Modificado

Axis 8	
Axis 7	
Axis 8	
Created by	MSFONSECA
Created on	29.10.2024
Created at	00:11:29
Changed by	MSFONSECA
Changed on	29.10.2024
Changed at	00:28:56
Deletion Flag	

Figura 68 - Registo Modificado

7.2.4. Teste das tabelas dos eixos.

Para efeitos de teste, apenas será alterado o eixo “Offer Family”. Vamos introduzir um novo registo e verificar que este existe na tabela posteriormente.

Change View "Offer Family": Overview

New Entries

Offer Family

Offer Family	Language	Description
OF_01	EN	CONSULTING
OF_02	EN	RESALE
OF_03	EN	SOLUTIONS - BUSINESS APPLICATIONS
OF_04	EN	SOLUTIONS - DIGITAL ENABLERS
OF_05	EN	SOLUTIONS - VERTICAL BUSINESSES
OF_06	EN	TECHNOLOGIES - APPLICATION
OF_07	EN	TECHNOLOGIES - CLOUD
OF_08	EN	TECHNOLOGIES - CYBERSECURITY
OF_09	EN	TECHNOLOGIES - DATA
OF_10	EN	TECHNOLOGIES - INFRASTRUCTURE
OF_11	EN	TECHNOLOGIES - INNOVATION
OF_12	EN	TECHNOLOGIES - WORKPLACE

Figura 69 - Registos da tabela "Offer Family" antes da nova adição

New Entries: Overview of Added Entries

Offer Family

Offer Family	Language	Description
OF_24	EN	TEST EIXO OF A38912

Figura 70 - Visão de inserção na tabela

Change View "Offer Family": Overview

New Entries

Offer Family

Offer Family	Language	Description
OF_01	EN	CONSULTING
OF_02	EN	RESALE
OF_03	EN	SOLUTIONS - BUSINESS APPLICATIONS
OF_04	EN	SOLUTIONS - DIGITAL ENABLERS
OF_05	EN	SOLUTIONS - VERTICAL BUSINESSES
OF_06	EN	TECHNOLOGIES - APPLICATION
OF_07	EN	TECHNOLOGIES - CLOUD
OF_08	EN	TECHNOLOGIES - CYBERSECURITY
OF_09	EN	TECHNOLOGIES - DATA
OF_10	EN	TECHNOLOGIES - INFRASTRUCTURE
OF_11	EN	TECHNOLOGIES - INNOVATION
OF_12	EN	TECHNOLOGIES - WORKPLACE
OF_24	EN	TEST EIXO OF A38912

Figura 71 - Registos do Eixo Offer Family atualizados

7.3. Teste dos novos campos num contrato standard

Para contexto de teste, vamos usar o contrato de exemplo 41107084 e vamos assinar o master contract criado manualmente 1000002434.

7.3.1. A nível de cabeçalho

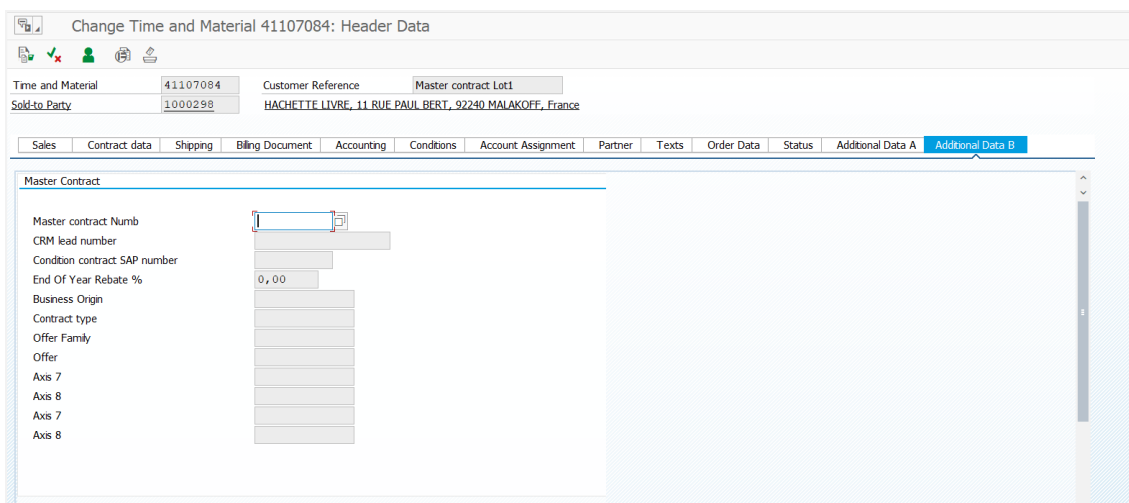


Figura 72 - Contrato 41107084 antes da assinatura do MC 1000002434

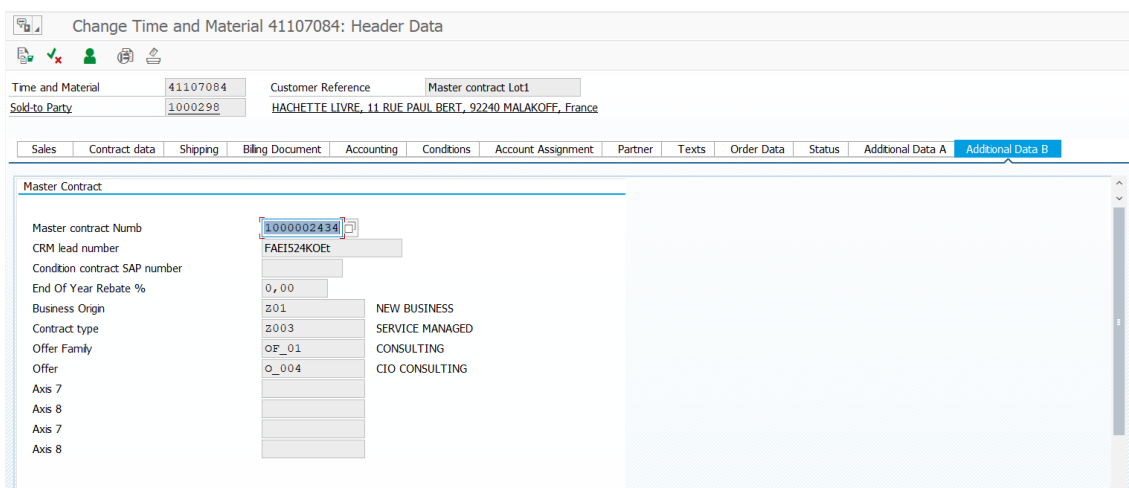


Figura 73 - Contrato 41107084 após a assinatura do MC 1000002434

Como podemos vêr nas imagens acima, o Master Contract, MC, foi devidamente assignado. Após o teste fui consultar a tabela de cabeçalho, VBAK, e esta possui a informação correta.

VBAK: Display of Entries Found

Search in Table: VBAK Sales Document: Header Data
 Number of Hts: 1
 Runtime: 0 Maximum No. of Hts: 500

Insert Column

VBELN	ZZMC_NUMBER	ZZMC_LEAD	ZZMC_CCN	ZZMC_FOYR	ZZMC_STATUS	ZZAXIS01_CODE	ZZAXIS01_DESC	ZZAXIS02_CODE	ZZAXIS02_DESC	ZZAXIS03_CODE	ZZAXIS03_DESC	ZZAXIS04_CODE	ZZAXIS04_DESC	ZZAXIS05_CODE	ZZAXIS05_DESC
41107084	1000002434	FAE1524KOE1		0,00	A	Z01	NEW BUSINESS	Z003	SERVICE MANAGED	OF_01	CONSULTING	O_004	CIO CONSULTING		

Figura 74 - Tabela VBAK após atribuição do contrato mestre

7.3.2. A nível de item

O item deve refletir a informação a nível de cabeçalho, permitindo apenas a alteração do eixo Offer.

Change Time and Material 41107084: Item Data

Sales Document Item: 10 Item Category: ZSER Auto. Service Deliv.
 Material: S0001 Assistance technique en Jours

Schedule lines Partner Texts Order Data Status Structure Risk Distribution Plan Free Characteristics Additional Data A Additional Data B Number of days sold

Master Contract

Master Contract Number: 1000002434
 CRM lead number: FAE1524KOE1
 Condition contract SAP number:
 End Of Year Rebate %: 0,00
 Business Origin: Z01 NEW BUSINESS
 Contract type: Z003 SERVICE MANAGED
 Offer Family: OF_01 CONSULTING
 Offer: O_004 CIO CONSULTING
 Axis 7:
 Axis 8:
 Axis 7:
 Axis 8:

Para efeito de teste iremos consultar a tabela do item, VBAP, e verificar se podemos alterar o eixo Offer mantendo o número de Master Contract.

Change Time and Material 41107084: Item Data

Sales Document Item: 10 Item Category: ZSER Auto. Service Deliv.
 Material: S0001 Assistance technique en Jours

Schedule lines Partner Texts Order Data Status Structure Risk Distribution Plan Free Characteristics Additional Data A Additional Data B Number of days sold

Master Contract

Master Contract Number: 1000002434
 CRM lead number: FAE1524KOE1
 Condition contract SAP number:
 End Of Year Rebate %: 0,00
 Business Origin: Z01 NEW BUSINESS
 Contract type: Z003 SERVICE MANAGED
 Offer Family: OF_01 CONSULTING
 Offer: O_019 FINANCIAL SERVICES
 Axis 7:
 Axis 8:
 Axis 7:
 Axis 8:

Figura 75 - Campo Offer alterado

VBELN	POSNR	ZZMC_NUMBER	ZZMC_LEAD	ZZMC_CCN	ZZMC_EOYR	ZZMC_STATUS	ZZAXIS01_CODE	ZZAXIS02_CODE	ZZAXIS03_CODE	ZZAXIS04_CODE	ZZAXIS05_CODE	ZZAXIS06_CODE	ZZAXIS07_CODE	ZZAXIS08_CODE
41107084	1	1000002434	FAEIS24KOE	0,00	A	Z01	2003	OF_01		O_019				

Figura 76 - Tabela do Item VBAP com o campo Offer modificado e mesmo número de MC

7.4. Teste das rotinas VOFM para criação por referência

De modo a testar esta funcionalidade, usamos o contrato 41107084 com o Master Contract 1000002434 para criar um novo contrato por referência. Como podemos ver na Figura 78, o contrato standard ainda nem tem número atribuído e as informações do Master Contract já foram transferidas.

Figura 77 - Criação dum contrato por referência do 41107084 com MC já atribuído

Figura 78 - Informações do MC transferidas com sucesso

7.5. Testo do Módulo de Função para criação

De forma a testar o webservice/módulo de função, e não tendo acesso á ferramenta da equipa de CRM, foi um usado um *payload* de exemplo enviado pela mesma equipa. É necessário ter atenção que o webservice recebe também o email da pessoa que requiere a criação, e que este é usado para validações dentro do módulo de função.

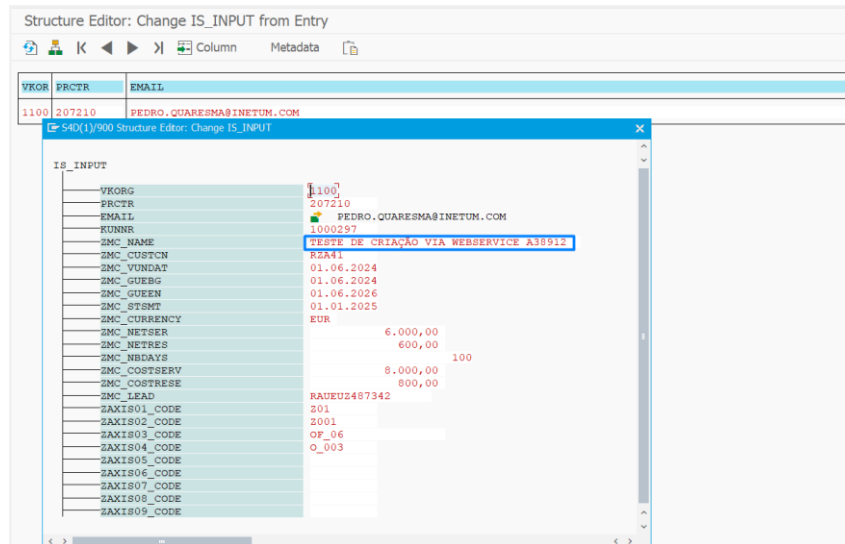


Figura 79 - Payload usado no exemplo de teste para o Webservice

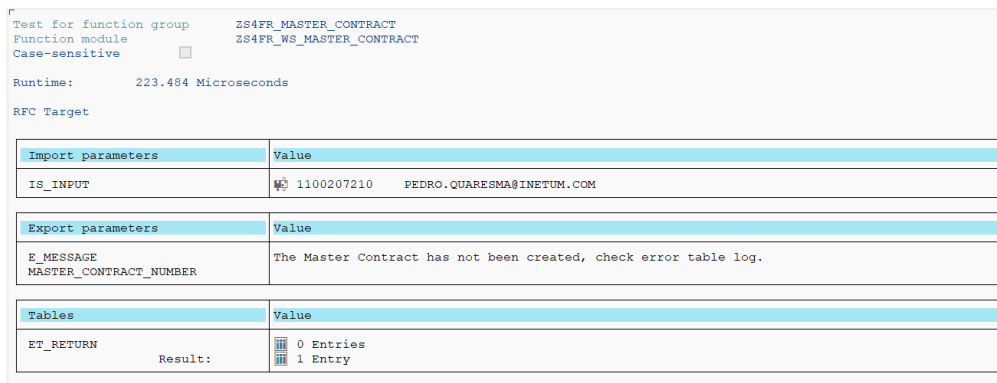


Figura 80 - Retorno de Erro

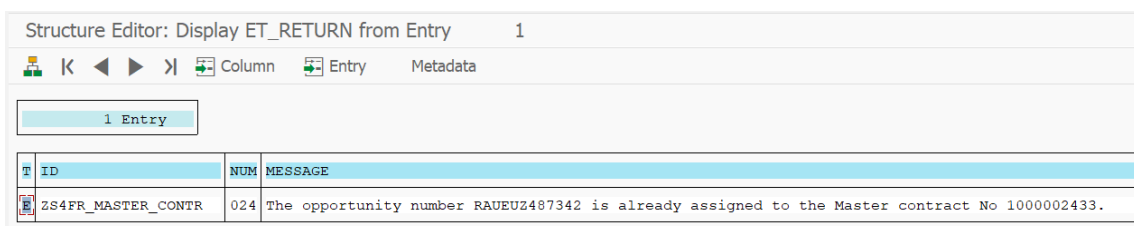


Figura 81 - Mensagem explicativa do erro

Como vemos, a mensagem ilustrada na Figura 81 Figura 81 - Mensagem explicativa do erro seria retornada á equipa de CRM. A mensagem explica que o Opportunity Number já é usado noutro contrato mestre. Foi então alterado o valor e a criação bem sucedida (Figura 82).

RFC Target

Import parameters	Value
IS_INPUT	1100207210 PEDRO.QUARESMA@INETUM.COM

Export parameters	Value
E_MESSAGE MASTER_CONTRACT_NUMBER	The Master Contract Number 1000002435 has been created. 1000002435

Tables	Value
ET_RETURN Result:	0 Entries 0 Entries

Figura 82 - Execução bem-sucedida do webservice

Ao recorrer á ferramenta de visualização de master contracts:

Master contract number	1000002435
CRM lead number	RAUEU248734238912
Sales Organization	1100 INETUM
Profit Center	0000207210 JDE
Sales Representative	8014476 PEDRO RAMOS QUARESMA
Customer	1000297 NEXT VENTURE LIMITED
Contract name	TESTE DE CRIAÇÃO VIA WEBSERVICE A38912
Customer contract number	R2A41
Contract signing date	01.06.2024
Valid-from date	01.06.2024
Valid to date	01.06.2026
Smoothing start date	01.01.2025
Status	A Draft
Annual rebates agreement	
Condition contract SAP number	
End Of Year Rebate %	0,00
Currency	EUR
Contract amount	6.600,00
Service amount	6.000,00
Reselling amount	600,00
Number of days sold	100,00

Figura 83 - Visualização do Master Contract criado via webservice

AXIS 6	
Created by	MSFONSECA
Created on	29.10.2024
Created at	10:36:44
Changed by	
Changed On	
Changed at	00:00:00
Deletion Flag	

Figura 84 - Visualização do Master Contract criado via webservice

7.6. Teste da Criação em massa

A fim de testar a criação em massa, foi criado um ficheiro de teste (Anexo 1) com informações para 3 Master Contract.

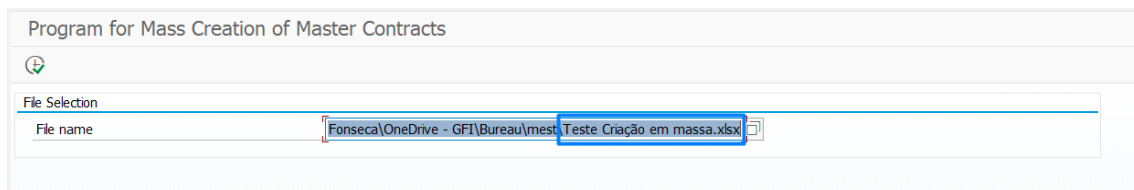


Figura 85 - Seleção do ficheiro a usar na criação em massa

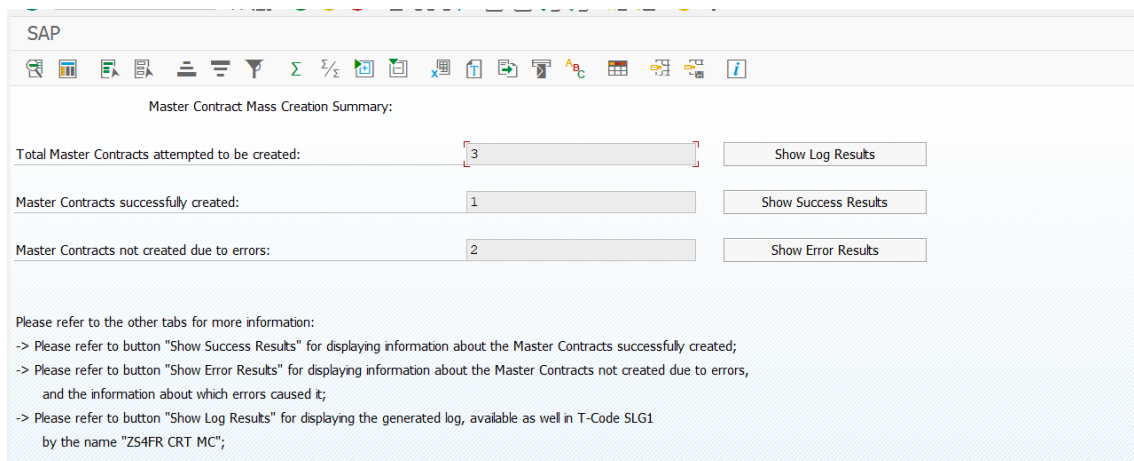


Figura 86 - Resultado da Criação em massa

No registo de Logs verificamos que o ficheiro continha informações para 3 Master Contract. 1 obteve sucesso enquanto 2 continham erros. De seguida vamos testar os três botões: para logs, para o sucesso e para os erros.

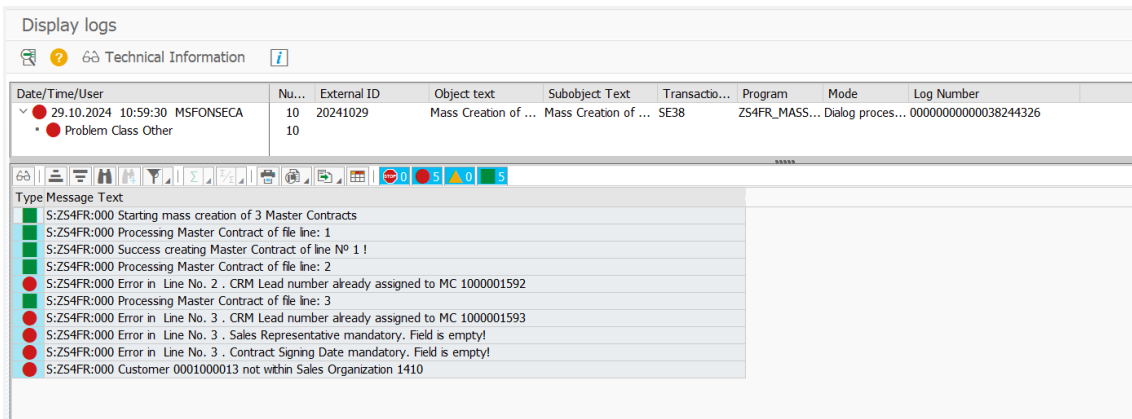


Figura 87 - Log geral após pressionar " Show Log Results"

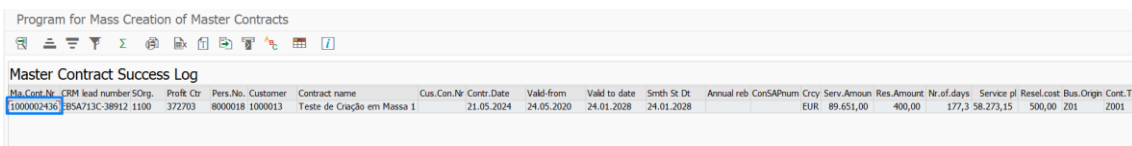


Figura 88 - Log de sucesso obtido após pressionar o botão "Show Success Results"

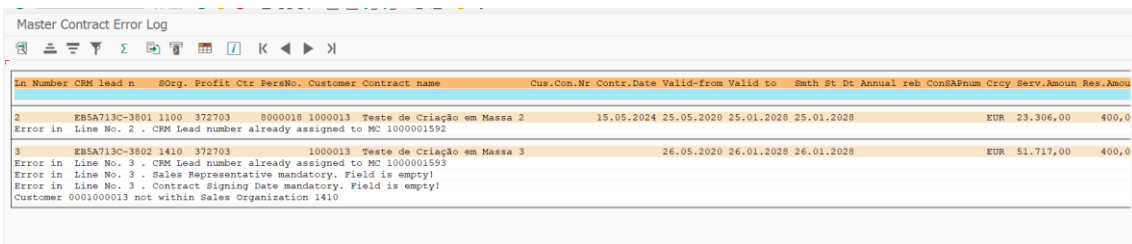


Figura 89 - Log de erros obtido após pressionar o botão "Show Error Results"

Finalmente, para verificarmos se o programa de criação em massa funciona corretamente, temos de verificar se o ficheiro com os registos em erro foi criado, se o contrato mestre 1000002436 foi inserido corretamente na tabela e se o log está disponível na transação SLG1.

Como podemos ver na Figura 90 o ficheiro com a terminação nova e que contém os registos de erro foi devidamente criado (Anexo 2)

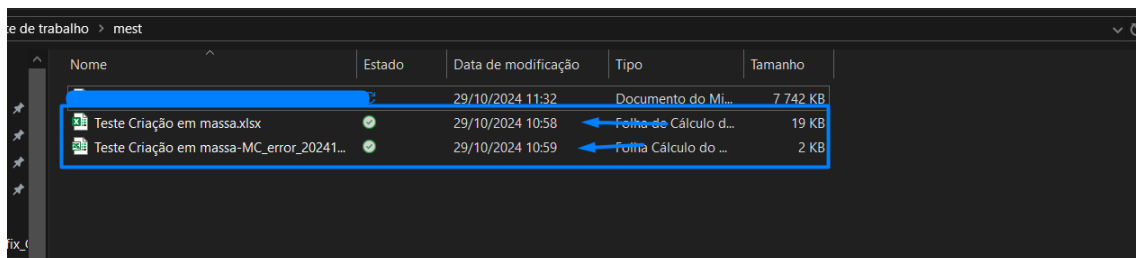


Figura 90 - Ficheiro de erro corretamente criado

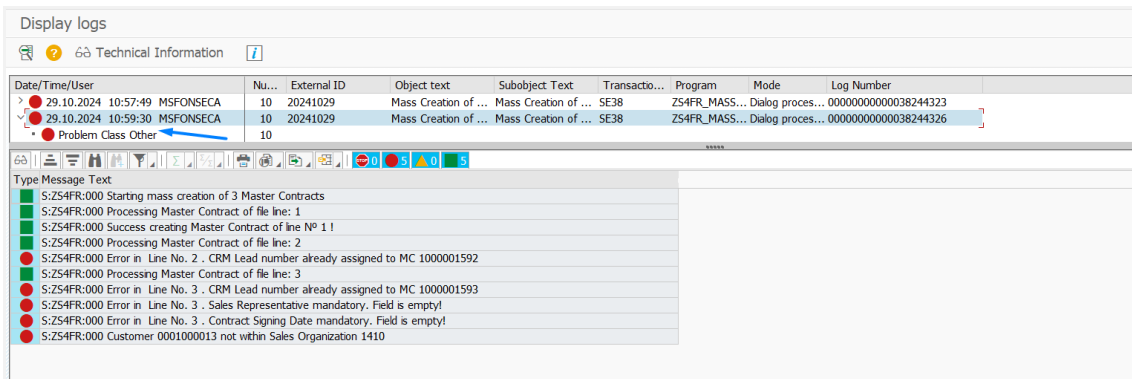


Figura 91 - Log disponível na SLG1

Na Figura 91 podemos ver que o log pode ser consultado após a execução do programa e deste ter sido encerrado.

Nas Figura 92, Figura 93 e Figura 94 podemos verificar que o contrato mestre 1000002436 foi devidamente criado.

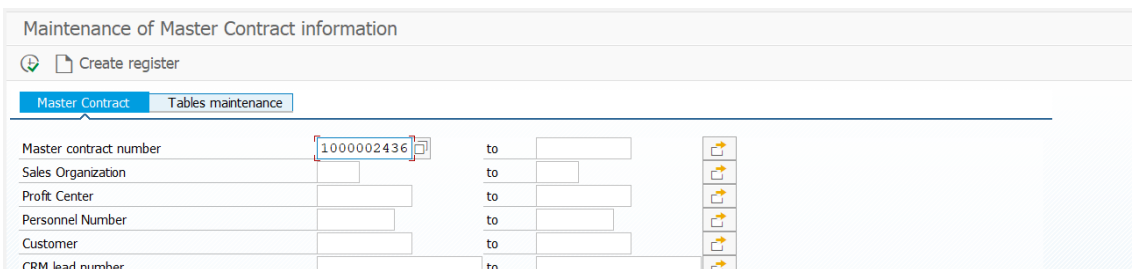


Figura 92 - Pesquisa de apenas 1 Master Contract para ser visualizado

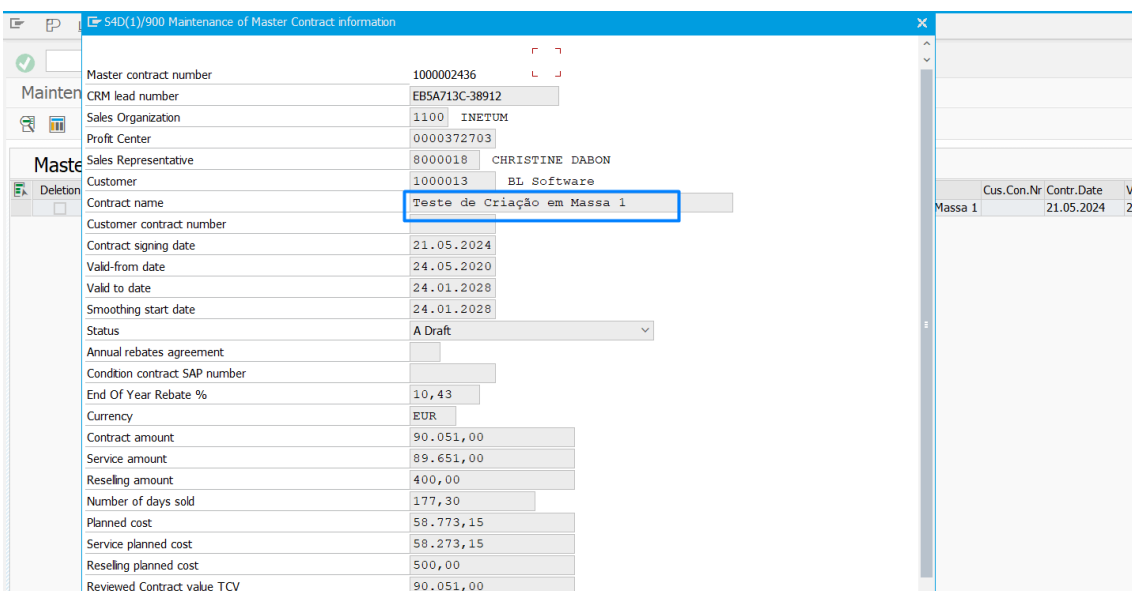


Figura 93 - Visualização do Master Contract criado pelo programa de criação em massa

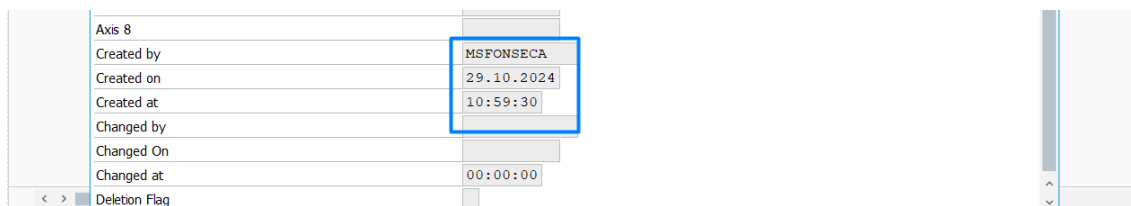


Figura 94 - Visualização do Master Contract criado pelo programa de criação em massa

7.7. Teste da Assinação em Massa

Por fim, para testar a última funcionalidade, a assinação em massa, foram escolhidos 2 contratos mestre para dois contratos standard.

Os contratos mestre são:

- 1000002435;
- 1000002434;

Os contratos standard são:

- 41107084;
- 41107083;

O ficheiro está anexado no Anexo 3.

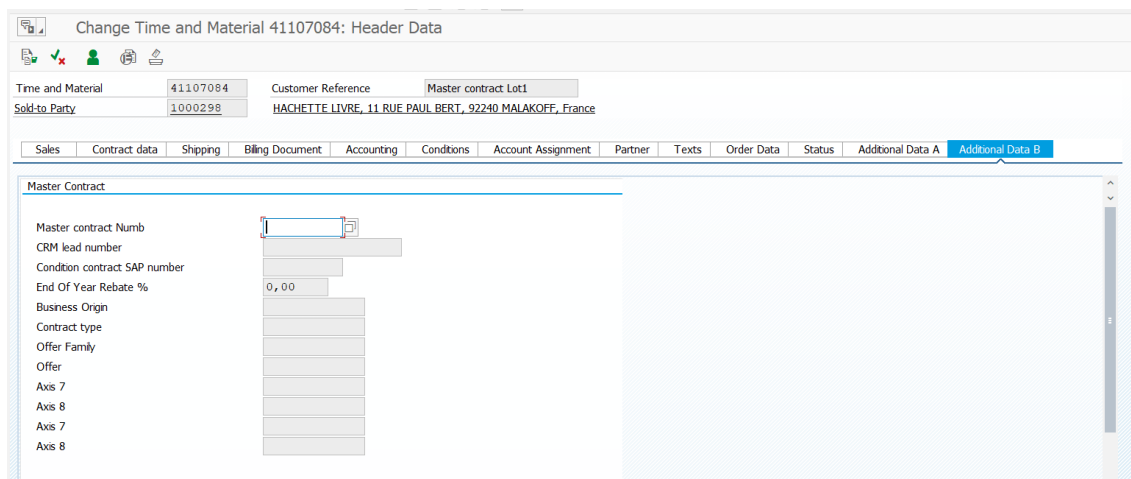


Figura 95 - Contrato 41107084 antes de executar o programa de assinação em massa

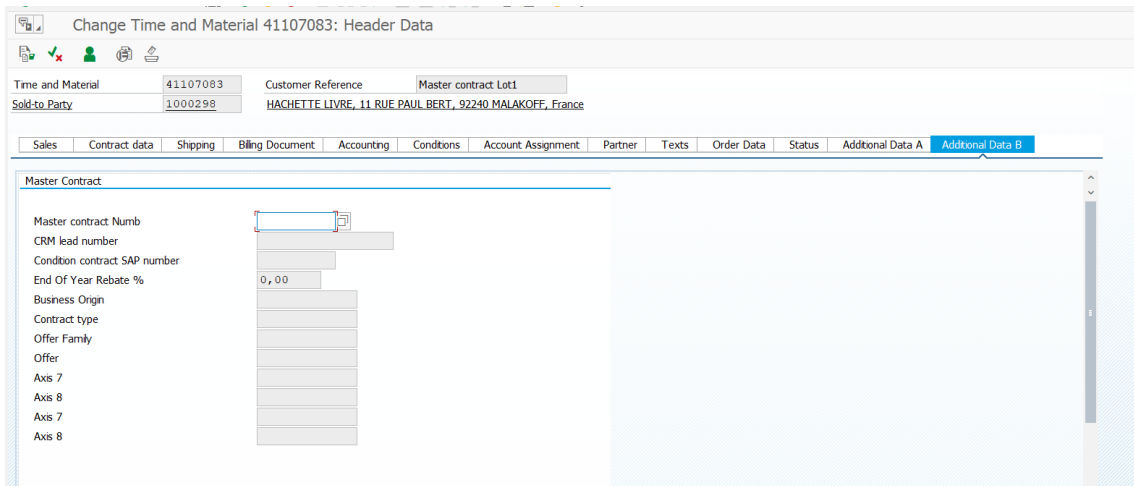


Figura 96 - Contrato 41107083 antes de executar o programa de assinatura em massa

Como podemos verificar, os contratos standard escolhidos não têm qualquer master Contract associado antes da execução do programa alvo dos testes (Figura 95 e Figura 96).

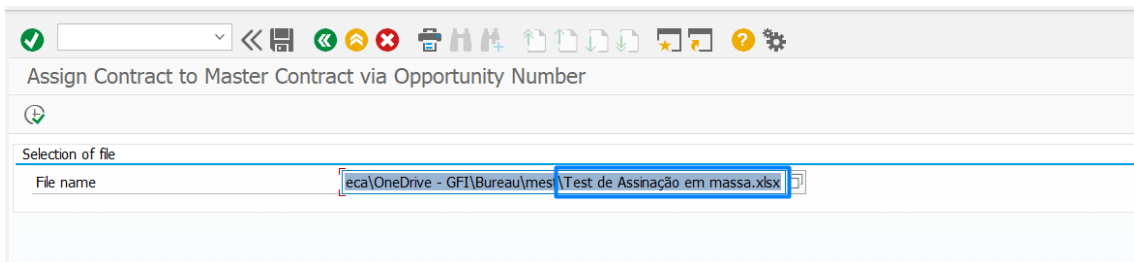


Figura 97 - Seleção do ficheiro a ser usado para a Assinação em Massa

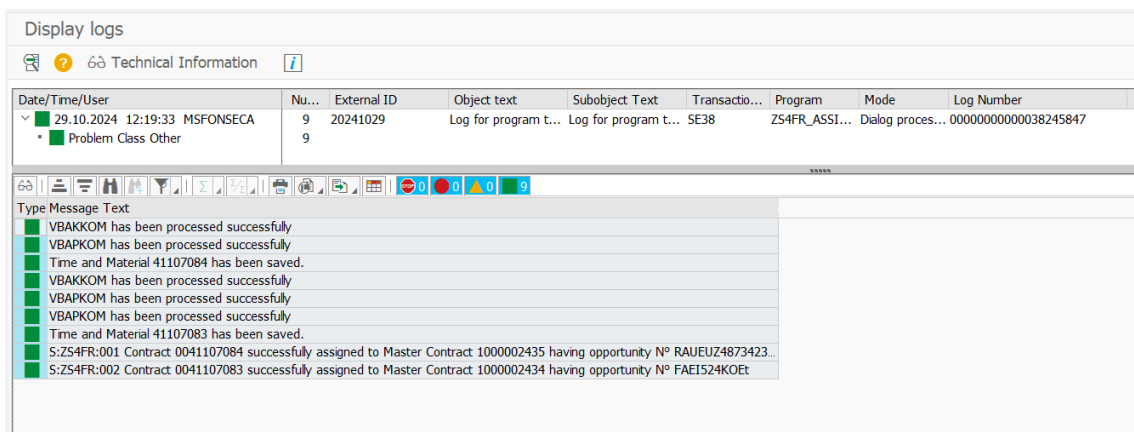


Figura 98 - Log do resultado da execução do programa de assinatura em massa

Como podemos ver na Figura 97 e na Figura 98, o programa executou com sucesso.

De seguida vamos confirmar nos contratos standard se o Master Contract foi devidamente assinado.

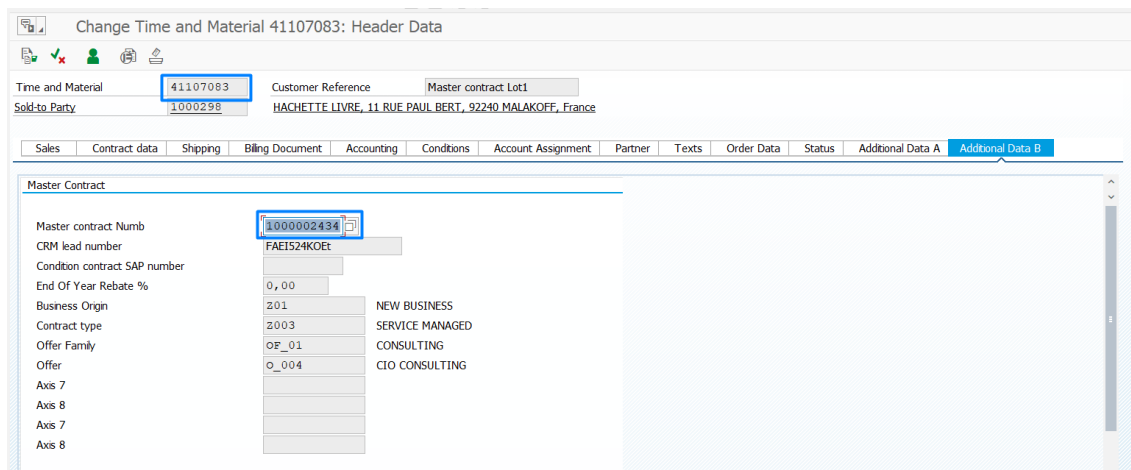


Figura 99 - Master Contract 1000002434 assignado ao contrato 41107083

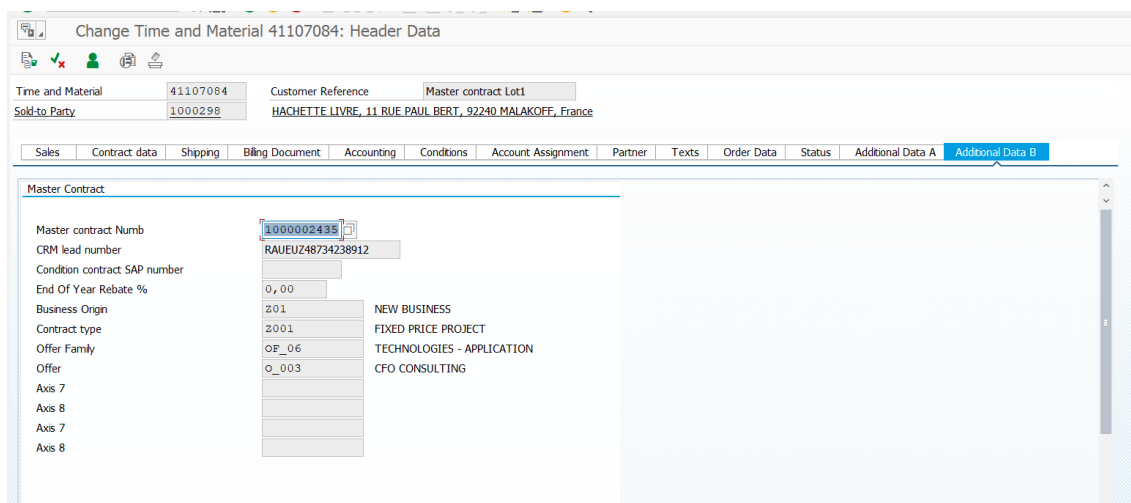


Figura 100 - Master Contract 1000002435 assignado ao contrato 41107084

7.8. Proposta de trabalho futuro

Após entregar a proposta de projeto com os objetivos propostos atingidos com sucesso, acredito que á pontos que passíveis de ser melhorados e novas funcionalidades adicionadas.

A empresa da GFI França já submeteu esta ferramenta para o seu ambiente produtivo para disponibilizar aos seus clientes finais, mas conversações constantes são mantidas para dar suporte á solução e discutir novos desenvolvimentos.

Algumas ideias propostas por mim já foram discutidas, dentro das quais:

- A possibilidade de gerar um ficheiro PDF com as informações do Master Contract pretendido e a quais o mesmo está assignado;
- Integrar no layout já existente das faturas dos contratos standard as novas informações do Master Contract;
- Permitir que o utilizador possa corrigir os erros nos registos do programa da criação em massa em tempo de execução e não gerar o ficheiro apenas;

8. Conclusões

A título de conclusão, o objetivo do desenvolvimento proposto era implementar uma nova ferramenta para gerir documentos de SD, mais concretamente criar o conceito do Master Contract. Para tal é necessário criar um programa principal para as ações de criação, modificação e visualização. É de igual modo, é importante disponibilizar essas mesmas informações nos contrato standard a quais foram assinados contratos mestre.

Desenvolver um programa para criação em massa e assinatura massa estava também compreendido no leque de funcionalidades que o cliente pediu, assim como a possibilidade de consumir um webservice para criar o Master Contract.

Durante o desenvolvimento foram encontradas algumas adversidades tais como manter a coerência dos dados entre cada momento de ação do utilizador, implementar as tabelas standard com os novos campos, alinhar todas as validações perante os diversos programas e métodos de alterar um master contract ou a sua assinatura.

Este projeto exigiu também um profunda investigação e crescimento profissional na vertente da administração de sistemas devido á necessidade de implementar objetos de autorização, roles, gestão de utilizadores e permissões...

Surgiram também algumas dificuldades na leitura e escrita dos ficheiros Excel, uma vez que o SAP ainda não está otimizado para o novo formato XLSX, e o método de leitura dum ficheiro não é direto de tabela para tabela, mas de tabela para uma tabela de índices. Converter do formato do Excel para uma tabela interna comum demonstrou ser um processo que requer agilidade e precisão nas estruturas usadas.

Por último, o uso de uma ALV hierárquico em árvore não é uma abordagem que os desenvolvedores utilizem, nem mesmo a própria SAP, pelo que requereu alguma pesquisa mais específica sobre esse módulo de função. Já a badi para a modificação dum contrato foi outra adversidade, perceber como preencher as extensões, inicializar o processo e submeter as modificações não é um processo que a SAP recomende fazer pelo que não existe muito suporte para tal.

Embora estas dificuldades e adversidades, todas as funcionalidades pretendidas foram atingidas com sucesso. Os programas cumprem todos os requisitos propostos pelo negócio, tanto que o mesmo já disponibilizou a solução em ambiente produtivo.

Assim como a GFI France, outras empresas podem adotar esta ferramenta pois a mesma proporciona otimização e melhoria dos processos de negócio. Resta agora que consideram as minhas propostas de melhoria e a ferramenta continuará a evoluir e, no futuro, ser abrangente nas empresas e nos tipos de negócio que esta impactua.

Bibliografia

- [1] Kumar, K., & Van Hillegersberg, J. (2000). ERP experiences and evolution. *Communications of the ACM*, 43(4), 23-26.
- [2] Jacobs, F. R., & Weston, F. C. (2007). Enterprise resource planning (ERP)—A brief history. *Journal of Operations Management*, 25(2), 357-363.
- [3] Scapens, R. W., & Jazayeri, M. (2003). ERP systems and management accounting change. *European Accounting Review*, 12(1), 201-233.
- [4] Koch, C. (2011). ERP software solutions. Technology Evaluation Centers.
- [5] Monk, E., & Wagner, B. (2012). Concepts in enterprise resource planning. Cengage Learning.
- [6] O'Leary, D. E. (2000). Enterprise resource planning systems: Systems, life cycle, electronic commerce, and risk. Cambridge University Press.
- [7] Hedman, J., & Borell, A. (2004). Narratives in ERP systems evaluation. *Journal of Enterprise Information Management*, 17(4), 283-290.
- [8] Esteves, J., & Pastor, J. (2001). Enterprise resource planning systems research: An annotated bibliography. *Communications of the Association for Information Systems*, 7(1), 1-52.
- [9] Al-Mashari, M., & Zairi, M. (2000). Supply-chain re-engineering using enterprise resource planning systems. *International Journal of Physical Distribution & Logistics Management*, 30(3), 296-313.

- [10] Somers, T. M., & Nelson, K. G. (2004). A taxonomy of players and activities across the ERP project life cycle. *Information & Management*, 41(3), 257-278.
- [11] Fasnacht, D. (2018). *Open Innovation Ecosystems: Creating New Value for Business and Society*. Springer.
- [12] Röglinger, M., Pöppelbuß, J., & Becker, J. (2018). Maturity models in business process management. *Business Process Management Journal*, 19(4), 606-627.
- [13] SAP SE. (2023). *ABAP Keyword Documentation*.
- [14] Gordon, R. (2019). *ABAP Development for SAP S/4HANA*. SAP Press.

Anexos



Teste Criação em
massa.xlsx

Anexo 1 - Ficheiro de Criação em Massa



Teste Criação em
massa-MC_error_2024

Anexo 2 - Ficheiro de erro da Criação em Massa



Test de Assinação
em massa.xlsx

Anexo 3 - Ficheiro de Assinação em Massa