



**ANTONIO MENEGHETTI FACULDADE - AMF
CURSO DE SISTEMAS DE INFORMAÇÃO**

EDUARDO GIACOMINI CIROLINI

APLICAÇÃO ABAP PARA GERENCIAR ACESSOS A AMBIENTES SAP

RESTINGA SECA/RS

2016

EDUARDO GIACOMINI CIROLINI

APLICAÇÃO ABAP PARA GERENCIAR ACESSOS A AMBIENTES SAP

Trabalho de Conclusão apresentado ao curso de Sistemas
de Informação como requisito parcial para obtenção do
título de Bacharel em Sistema de Informação

Orientador: Profº. Ms. Fábio Sarturi Prass

RESTINGA SECA/RS

2016

AGRADECIMENTOS

Agradeço ao meu Orientador Fábio Sarturi Prass, pelo incentivo, presteza e paciência no auxílio às atividades, principalmente sobre o andamento e normatização deste Trabalho de Conclusão de Curso, onde com toda certeza seus conhecimentos foram compartilhados.

Agradeço os demais Professores da ANTONIO MENEGHETTI FACULDADE – AMF, que foram corresponsáveis pelo nosso crescimento intelectual.

“O único lugar onde o sucesso vem antes do trabalho é no dicionário”.

Albert Einstein

RESUMO

O uso de sistemas de gestão é cada vez mais unânime no mundo, principalmente o sistema de gestão da SAP, gerando vários benefícios para quem o implanta. Além das facilidades que o sistema traz por padrão, é possível customizá-lo para gerar inúmeros benefícios com base na necessidade de quem o utiliza. Desta forma, o presente trabalho tem por objetivo o desenvolvimento de uma aplicação que gerencia acessos a ambientes SAP. Iniciando com explicações de conceitos sobre sistemas de gestão como um todo. Após direcionando para o sistema de gestão da SAP e entendendo seu funcionamento. A aplicação foi desenvolvida usando a linguagem de programação ABAP, própria da SAP. O impacto desta aplicação é eliminar o cadastro manual, gerando organização, praticidade e agilidade no acesso a ambientes SAP.

Palavras-chave: Sistema de Gestão; SAP; ABAP.

ABSTRACT

The use of management systems is more and more unanimous in the world, especially the SAP management system, generating several benefits for those who implement it. In addition to the facilities that the system brings by default, it is possible to customize it to generate numerous benefits based on the need of those who use it. In this way, the present work has the objective of developing an application that manages access to SAP environments. Starting with explanations of concepts about management systems as a whole. After directing to the SAP management system and understanding its operation. The application was developed using SAP's proprietary ABAP programming language. The impact of this application is to eliminate the manual cadastre, generating organization, practicality and agility in accessing SAP environments.

Keywords: Management System; SAP; ABAP.

LISTA DE ILUSTRAÇÕES

Figura 1. Modelo conceitual do Sistema ERP.....	15
Figura 2. Modelo conceitual do sistema gestão da SAP.	17
Figura 3. Modelo conceitual do SAP NetWeaver.	19
Figura 4. Visão lógica do NetWeaver AS ABAP.	20
Figura 5. Visão orientada a Software do NetWeaver AS ABAP.	23
Figura 6. Visão orientada a Usuário do NetWeaver AS ABAP.....	24
Figura 7. Exemplo de linguagem ABAP.....	25
Figura 8. Exemplo de Processo no SAP.....	26
Figura 9. Exemplo de um Arquivo XML.....	28
Figura 10. Tela da aplicação SAP GUI.....	30
Figura 11. Dados de acesso de um ambiente no SAP GUI.....	31
Figura 12. Tela de acesso a um ambiente SAP.....	32
Figura 13. Por dentro de um ambiente conectado.....	33
Figura 14. Executando aplicações no SAP.....	34
Figura 15. Tela inicial do programa ZTSAPLOGON.....	34
Figura 16. Funções do programa ZTSAPLOGON.....	35
Figura 17. Cadastrando um cliente.....	35
Figura 18. Registro de um cliente salvo na tabela ZTCLIENTS.....	36
Figura 19. Cadastrando informações de acesso ao ambiente do cliente.....	36
Figura 20. Registro de um cliente salvo na tabela ZTENVIROMENTS.....	37
Figura 21. Cadastrando informações de usuário para acesso ao ambiente.....	37
Figura 22. Registro de um cliente salvo na tabela ZTCREDENTIALS.....	38
Figura 23. Criando o arquivo XML com a nova entrada.....	38
Figura 24. Exemplo de cadastrado de cliente que não está contido no XML.....	39
Figura 25. Exemplo de cadastrado de cliente que está contido no XML.....	39
Figura 26. Selecionando um ambiente para gerar uma entrada no XML.....	40
Figura 27. Criando um XML com a nova entrada.....	41
Figura 28. SAP com o novo ambiente disponível.....	42

Figura 29. Estrutura do XML gerado para o novo ambiente.....	43
Figura 30. Classe ZCL_SAPLOGON_CREATE.....	44
Figura 31. Método CREATE_SAPLOGON.	44

LISTA DE ABREVIATURAS

ABAP - Advanced Business Application Programming

BAPI - Business Application Programming Interface

DBMS - Data Base Management System

ERP - Enterprise Resource Planning

ITS - Internet Transaction Server

SAP - Systeme, Anwendungen und Produkte in der Datenverarbeitung

XML - Extensible Markup Language

SUMÁRIO

RESUMO	5
ABSTRACT	6
LISTA DE ILUSTRAÇÕES	7
LISTA DE ABREVIATURAS	9
1. INTRODUÇÃO.....	11
1.1 Objetivos	11
1.1.1 Objetivo Geral	11
1.1.2 Objetivos Específicos	12
2. JUSTIFICATIVA	13
3. REFERENCIAL TEÓRICO.....	14
3.1 Sistemas de Gestão Empresarial (ERP).....	14
3.2 Ferramentas SAP.....	15
3.2.1. SAP Enterprise Planning.....	15
3.2.2. SAP NetWeaver	18
3.2.2.1. Visão Geral do SAP NetWeaver	19
3.2.2.1.1. Visão Lógica	19
3.2.2.1.2. Visão Orientada a Software	21
3.2.2.1.3. Visão Orientada a Usuário.....	23
3.2.3. Linguagem ABAP.....	25
3.3. Arquivo XML	26
4. METODOLOGIA	29
5. ESTUDO DE CASO	30
6. CONSIDERAÇÕES FINAIS	45
7. REFERÊNCIAS BIBLIOGRÁFICAS.....	46

1. INTRODUÇÃO

A cada dia que passa, a informática vem tomando um espaço cada vez maior no dia a dia das organizações. Uma pesquisa realizada nas empresas permite analisar a importância e os fatores que levou a mesma organização a aderir o sistema de informatização, sendo alguns deles a agilidade e rapidez no processamento das informações. (ALENCAR, 2013).

Uma das tecnologias mais usadas pelas empresas é o sistema de gestão (ERP), um sistema que gerencia as informações de todas as áreas da empresa: como o Financeiro, Contábil, Recursos Humanos, etc. Fazendo com que as empresas tenham uma visão geral e mais correta de suas informações, gerando uma melhor administração e, conseqüentemente, uma melhor produtividade e lucratividade.

Existem vários sistemas de gestão no mercado, e um deles é o sistema de gestão da SAP, uma multinacional alemã com foco em sistemas gerenciais. O sistema da SAP é dividido em vários módulos como qualquer outro sistema ERP, mas como suas particularidades, como a linguagem de programação, o ABAP, e também a forma de acesso aos seus ambientes.

No sistema de gestão da SAP, não existe uma aplicação que gerencie esses acessos aos ambientes. Assim, um colaborador que utilize o sistema de gestão da SAP, precisará inserir manualmente os dados necessários para acessar o ambiente escolhido, que pode ser o de desenvolvimento, qualidade ou produção. Além de inserir manualmente, os acessos ficarão na sua máquina, assim, se outro colaborador precisar uma entrada que não tenha, precisará pedir os dados de acesso também, gerando uma perda de tempo e produtividade desnecessária. Nesse ensejo, como tornar o procedimento de acesso aos ambientes SAP uma atividade simples e rápida?

1.1 Objetivos

Segue abaixo os objetivos para solucionar a problemática proposta:

1.1.1 Objetivo Geral

Criar uma aplicação que facilite o gerenciamento de ambientes SAP de forma organizada e ágil.

1.1.2 Objetivos Específicos

- Eliminar inserção manual e repetitiva.
- Organizar as informações.
- Tornar o processo mais ágil.

2. JUSTIFICATIVA

No meio empresarial, onde há uma mudança contínua, a agilidade e praticidade nos processos têm um grande impacto. Uma boa organização pode gerar um ganho de produtividade e lucratividade para a empresa.

Um bom exemplo é que, hoje em dia, as empresas que eram referências há cerca de dez anos já nem mesmo existem. E a causa disso está justamente no fato de que tais corporações não conseguiram se adaptar às grandes e muitas mudanças do cenário mundial, que destacam a agilidade empresarial como item fundamental para a sobrevivência de qualquer corporação (CONVENIA, 2013).

Nesse contexto, para melhorar a agilidade e praticidade dos processos, tem-se a ajuda de softwares, que são a alma das organizações. Um desses softwares usados nesse processo é o ERP, *Enterprise Resource Planning*, próprio, um sistema de gestão de negócios desenvolvido pela empresa alemã SAP. A sigla SAP tem como significado *Systeme, Anwendungen und Produkte in der Datenverarbeitung*, em português, Sistemas, Aplicativos e Produtos para Processamento de Dados.

No mercado atual, fazer o uso das tecnologias mais recentes e ter a capacidade de adaptação constante é o que mais traz pontos positivos a uma empresa. Esta, que deve estar sempre conectada ao seu consumidor e ter em mente qual é a reação destes clientes em relação ao que produz – podendo, desta forma, realizar todas as alterações e interações necessárias para sair ganhando no jogo corporativo. (CONVENIA, 2013).

No sistema de gestão da SAP, não existe nenhuma ferramenta que gere a forma de acessar os vários ambientes que nele existem, sendo feitos de forma manual. Desta forma há uma perda de produtividade e desorganização interna por não existir um local centralizado para essas informações estarem contidas, pelo fato de, a cada colaborador que ingressar e utilizar o sistema de gestão da SAP, precisará inserir manualmente os acessos de onde irá trabalhar. Assim, um programa desenvolvido dentro deste sistema de gestão, supriria essa necessidade, pois nele ficariam todas as informações dos acessos SAP para gerenciamento e também a geração de um XML, *Extensible Markup Language*, pronto com todas as entradas que o colaborador precisa, sem que o mesmo precise digitá-los na mão.

3. REFERENCIAL TEÓRICO

Neste capítulo, serão apresentados os itens que permeiam a aplicação. Iniciando pelos conceitos que norteiam o ERP, elencando seus conceitos, funções e vantagens. Posteriormente entrando no sistema de gestão da SAP, na qual a aplicação será desenvolvida e o detalhamento dos seus módulos. Já na parte técnica, será abordada a explicação da linguagem ABAP, *Advanced Business Application Programming*, que é proprietária e será usada no desenvolvimento da aplicação. E, por fim, uma explicação sobre como funciona, o que é como será utilizado o arquivo XML, que será gerado pelo programa.

3.1 Sistemas de Gestão Empresarial (ERP)

Em termos gerais, ERP é uma plataforma de software desenvolvida para integrar os diversos departamentos de uma empresa possibilitando a automação e armazenamento de todas as informações de negócios; o ERP possibilita um fluxo de informações único, contínuo e consistente. (GASPAR, 2012).

Na Figura 1 observa-se a atuação do ERP. Onde o Sistema ERP (localizado no centro da imagem) interage com todas as partes da empresa: Financeiro, Recursos Humanos, etc, representados pelos círculos de fora. Essa integração faz com que as informações sejam reunidas e relacionadas entre si para dar uma melhor visão e um melhorando na gestão da organização.

Desta forma, são vantagens de implementar um sistema ERP:

- Ajudar na comunicação interna;
- Agilizar a execução de processos internos;
- Diminuir a quantidade de processos internos;
- Evitar erros humanos - em cálculos de tributos e pagamentos, por exemplo;
- Ajudar na tomada de decisões;
- Auxiliar na elaboração de estratégias operacionais;
- Agilizar a obtenção de dados referentes a determinados cenários;
- Diminuir o tempo de entrega do produto ou serviço ao cliente;
- Ajudar a lidar com grandes volumes de informação;
- Evitar trabalho duplicado;

- Fazer com que a empresa se adapte melhor a mudanças no mercado e na legislação. (ALECRIN, 2013).



Figura 1. Modelo conceitual do Sistema ERP

Fonte: SOUSA(2013).

3.2 Ferramentas SAP

A SAP é uma empresa multinacional de origem alemã que trabalha com softwares de gestão de empresas. A multinacional possui uma forte presença no Brasil onde possui o SAP Labs Brasil, situado na cidade de São Leopoldo, Rio Grande do Sul, primeiro centro da companhia na América Latina.

Na atualidade, a empresa segue líder mundial no fornecimento de softwares de gestão empresarial com projetos e atuações em diversos países, oferecendo uma diversidade de produtos e, entre eles, o SAP ERP, sistema de planejamento dos recursos empresariais.

3.2.1. SAP Enterprise Planning

O SAP Enterprise Planning é o software de gestão dos recursos empresariais SAP. É constituído de quatro soluções:

- SAP ERP Financials: oferece suporte corporativo para áreas de contabilidade, geração de relatórios financeiros, gestão de desempenho e governança corporativa, com suporte para diversos processos específicos de mercados verticais e suporte local a requisitos de diferentes mercados, idiomas e moedas;
- SAP ERP Operations: oferece soluções com a finalidade de automatizar e a integrar as execuções de aquisições e logística, desenvolvimento de produtos e manufatura e vendas e serviços;
- SAP ERP Human Capital Management (HCM): oferece recursos de gestão de capital humano, tais como gestão de talentos, gestão de processos da força de trabalho e a alocação pessoal, oferecendo recursos analíticos da força de trabalho, seguindo as conformidades com a legislação e as regulações locais e globais;
- SAP ERP Corporate Services: oferece recursos para o gerenciamento de imóveis, bens corporativos, portfólios de projetos, viagens corporativas, conformidade a regras ambientais, de saúde e de segurança, qualidade e serviços de comércio global.

Como o SAP ERP é um sistema de grande porte, acaba por abranger muitas áreas, setores e mercados. Sendo assim, para um melhor atendimento, o sistema foi dividido em módulos, conforme Figura 2.

Sales & Distribution (SD) é o módulo de vendas e distribuição. É responsável pelos processos e atividades que envolvem venda de produtos, bem como a sua distribuição, pedidos, impostos, notas;

Material Management (MM) pertence ao módulo de administração de materiais. Lida com o gerenciamento da empresa no que envolve materiais de modo geral. Suprimentos, planejamento e controle de materiais, compras e recebimento de mercadorias, manutenção de estoques, etc.;

Production Planning (PP) faz parte do módulo de planejamento de produção. Lida com o planejamento e controle dos processos que envolvem a produção;

Quality Management (QM) é do módulo de administração da qualidade. Ele abrange os processos envolvidos com a qualidade do produto: controle de qualidade, planejamento dos processos de qualidade, controle de produção e qualidade;

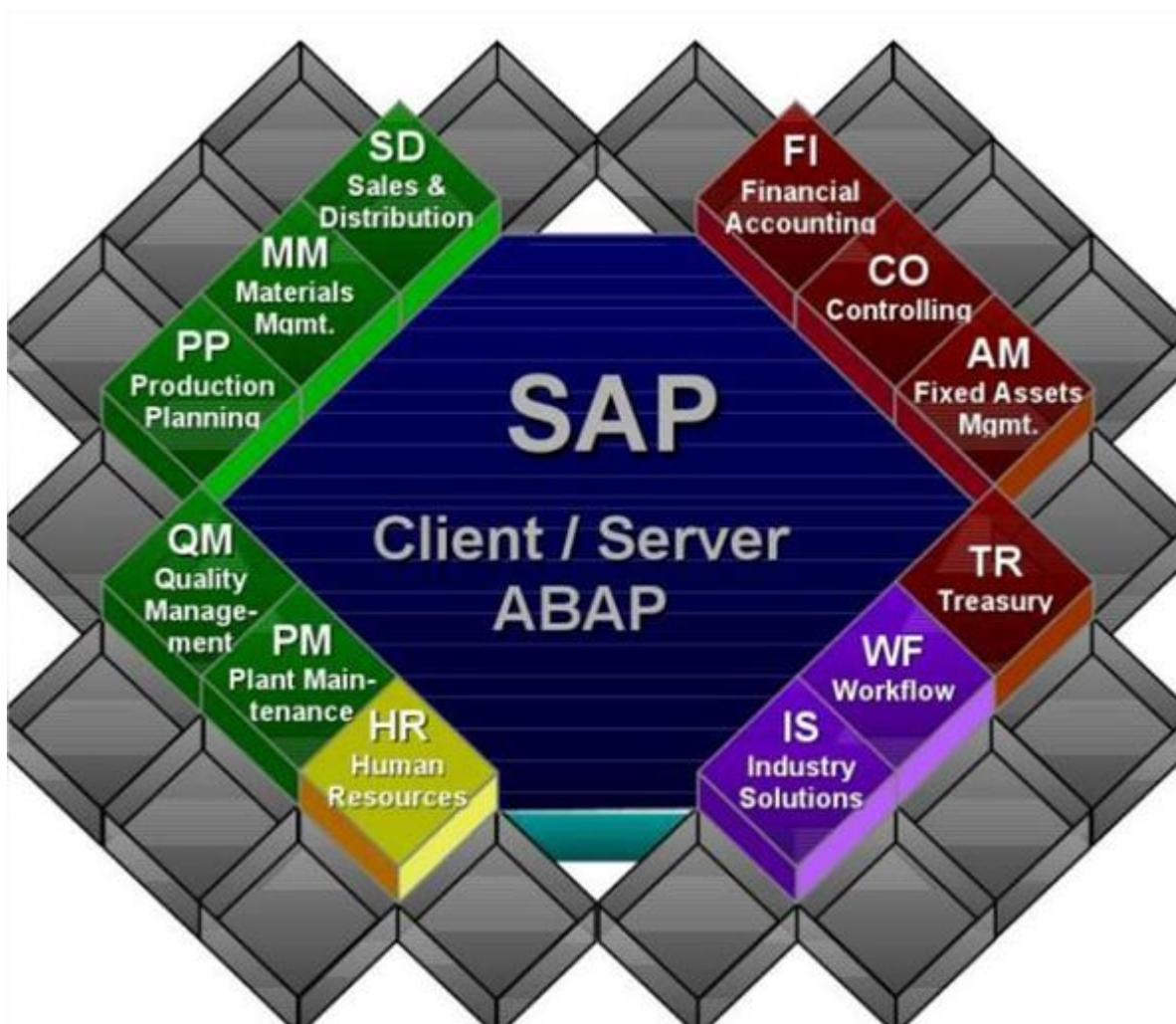


Figura 2. Modelo conceitual do sistema gestão da SAP.

Fonte: MOURA(2012).

PlantMaintenance (PM) pertence ao módulo de manutenção. Como o nome sugere, abrange todo o processo, controle e execução de tarefas que envolvam manutenção de equipamentos, produtos, etc.;

HumanResources (HR) é do módulo de recursos humanos. Lida com os dados relativos aos colaboradores da empresa, como registro, avaliação, pagamento, etc.;

Financial Accounting (FI) é do módulo de contabilidade financeira. Abrange a contabilidade de clientes e fornecedores, a contabilidade financeira principal, a administração de contas, bem como relatórios envolvendo as informações de contabilidade financeira;

Controlling (CO) faz parte do módulo de contabilidade de custos. Lida com os dados das receitas da empresa e as movimentações e controle dos custos;

FixedAssets Management (AM) é do módulo de contabilidade do imobilizado. Administra o controle dos aspectos do ativo imobilizado, tais como máquinas, terrenos, prédios, equipamentos, veículos e etc.;

Treasury (TR) ou ainda TreasuryandRisk Management (TRM), Administração de Tesouraria e Riscos,lida com os módulos FI e CO, realizando o controle de tesouraria e riscos, como depósitos e outras transações financeiras;

Workflow (WF) realiza a ligação dos módulos do SAP ERP para com ferramentas, serviços e tecnologias para todas as aplicações;

IndustrySolutions (IS) pertence ao módulo de soluções setoriais. Responsável pela ligação entre os módulos de aplicação do SAP ERP com soluções específicas de setores da indústria, como por exemplo, óleo e gás;

3.2.2. SAP NetWeaver

O SAP NetWeaver é camada técnica do SAP. Que contém, além do ABAP, várias outras capacidades que permitem qualquer sistema da SAP oferecer um servidor web, conectividade com dispositivos móveis, colaboração entre usuário e executar códigos ABAP. A Figura 3 ilustra o que é o SAP NetWeaver inserido no contexto de trabalho. (PAGOTI, 2013).



Figura 3. Modelo conceitual do SAP NetWeaver.

Fonte: PAGOTTI(2013).

3.2.2.1. Visão Geral do SAP NetWeaver

O entendimento do SAP NetWeaver será abordado de três formas. Uma visão lógica, outra orientada a software e por fim a visão do usuário. (SAP, 2014).

3.2.2.1.1. Visão Lógica

Conforme figura 4, o componente de Administração de Serviços e Kernel é um ambiente de tempo de execução para todas as aplicações ABAP: que é um hardware, sistema operacional e um banco de dados independente. O tempo de execução ABAP é escrito principalmente em C e C++.

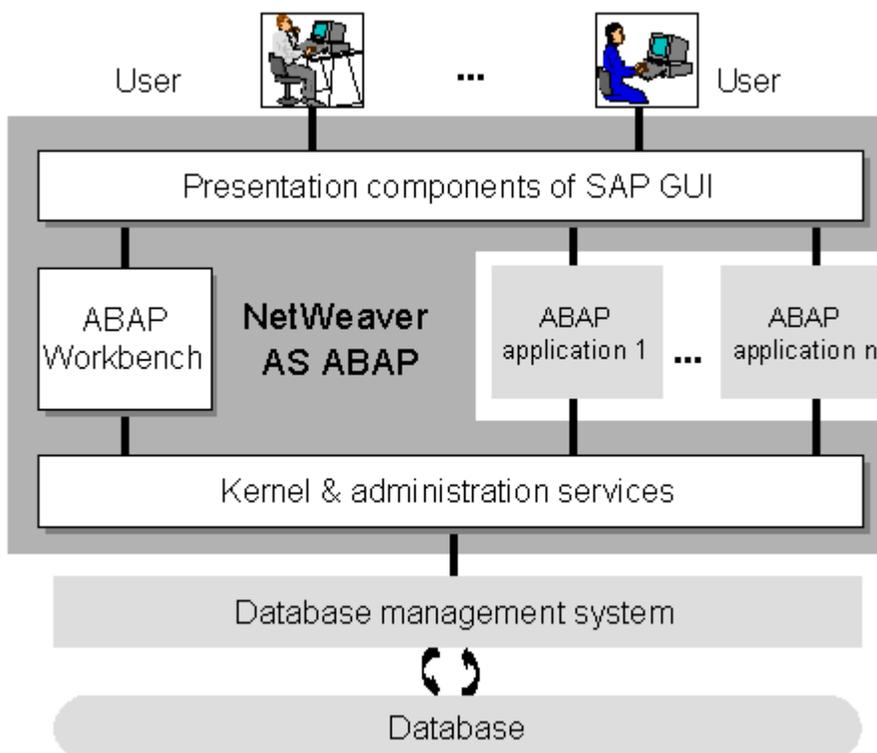


Figura 4. Visão lógica do NetWeaver AS ABAP.

Fonte: SAP(2014).

No entanto, algumas partes de baixo nível também são escritas em ABAP. As tarefas do componente da Administração de Serviços e Kernel são:

- **Aplicações em Execução:** todas as aplicações ABAP são executadas em máquinas virtuais dentro deste componente.
- **Usuário e processo de administração:** NetWeaver AS ABAP é um ambiente multiusuário, onde cada usuário pode executar diversas aplicações ABAP independentes. Em suma, este componente é responsável pelas tarefas que normalmente pertencem a um sistema operacional. Os usuários entram no NetWeaver AS ABAP e executam aplicações ABAP dentro dele. Deste modo, eles não entram em contato com o sistema operativo real. NetWeaver AS ABAP é o único usuário sistema operacional do usuário.
- **Acesso a Banco de Dados:** cada NetWeaver AS ABAP está ligado a um sistema de banco de dados, que consiste em um sistema de gerenciamento de banco de dados (DBMS) e o

próprio banco de dados. As aplicações ABAP não se comunicam diretamente com o banco de dados. Em vez disso, eles usam os serviços de administração.

- Comunicação: Aplicações ABAP podem se comunicar tanto com outros sistemas SAP como com sistemas externos. Também é possível acessar aplicações ABAP de sistemas externos usando uma interface BAPI, *Business Application Programming Interface*. Os serviços necessários para a comunicação fazem parte do componente do kernel e serviços de administração.
- Controle e administração de NetWeaver AS ABAP: O componente contém programas que permitem monitorar e controlar a NW AS enquanto ele está executando, e mudar seus parâmetros de tempo de execução.

O componente ABAP Workbench é o ambiente de desenvolvimento para aplicações na linguagem ABAP. Com ele, pode-se criar, editar, testar e organizar esses programas aplicativos. É totalmente integrado no NetWeaver AS ABAP e, assim como outras aplicações ABAP, é também escrito em ABAP.

3.2.2.1.2. Visão Orientada a Software

Um sistema SAP é um sistema cliente/servidor *multi-tier*. Os componentes individuais de software estão dispostos em fileiras como um cliente para os componentes abaixo delas ou um servidor para os componentes acima deles.

- Camada de banco de dados: A camada de banco de dados, que é acessado por NetWeaver AS, consiste em um sistema de banco de dados central, que por sua vez é composta de sistema de gestão de banco de dados (DBMS) e o próprio banco de dados. O banco de dados não contém apenas os dados mestre e dados de transações de seus programas de aplicação ABAP. Todos os dados do NetWeaver AS ABAP são armazenados lá. Por exemplo, o banco de dados contém o controle e os dados de como o NetWeaver AS executa as aplicações ABAP. Os componentes (textos do programa, definições de tela, menus, módulos de função, e assim por diante) são armazenados em uma parte especial do banco de dados conhecido como o Repositório. Assim se trabalha com eles no ABAP Workbench.

- **Camada de Aplicação:** Os componentes da camada de aplicação do NetWeaver AS ABAP de software consistem em um ou mais servidores de aplicação ABAP e um servidor de mensagens. Cada servidor de aplicativos contém um conjunto de serviços utilizados para executar o NW AS. Teoricamente, só se precisa de um servidor de aplicativos para executar um NW AS. Na prática, os serviços são distribuídos em mais de um servidor de aplicativos. Isto significa que nem todos os servidores de aplicações irão fornecer toda a gama de serviços. O servidor de mensagens é responsável pela comunicação entre os servidores de aplicativos. Ele passa solicitações de um servidor de aplicativos para outro dentro de um NW AS. Ele também contém informações sobre os grupos de servidores de aplicação e a carga atual de equilíbrio dentro deles. Ele usa essas informações para escolher um servidor de forma adequada quando um usuário faça logon no sistema.
- **Camada de Apresentação:** Esta camada é a interface entre o sistema SAP e seus usuários. Usando seus componentes de software referidos como SAP GUI (Graphical User Interface) esta camada fornece uma interface gráfica intuitiva para entrada e exibição de dados. A camada de apresentação envia a entrada do usuário no servidor de aplicativos, e recebe dados para exibição a partir dele. Enquanto um componente GUI SAP está sendo executado, ele permanece ligado à sessão de terminal de um usuário no NetWeaver AS ABAP. Esta visão orientada a software pode ser expandida para incluir outras camadas, como uma Internet Transaction Server (ITS).
- **Vantagens da arquitetura *multi-Tier*:** A distribuição do sistema SAP ao longo de três camadas significa que a carga do sistema, também é distribuído. Isto leva a um melhor desempenho do sistema. Outro benefício é a alta escalabilidade alcançada devido ao fato de que se pode distribuir os componentes de um sistema SAP entre as diferentes unidades de hardware de software praticamente sem quaisquer restrições. Isto é particularmente valioso na camada de aplicação, onde se pode facilmente adaptar NetWeaver AS ABAP para atender à crescente demanda pela instalação de novos servidores de aplicação ABAP. A arquitetura do sistema SAP permite instalar a camada de aplicação e a camada de banco de dados em máquinas separadas e deixá-los se comunicar usando a rede. Isto reduz consideravelmente a carga colocada sobre o sistema de banco de dados que contém o armazenamento de dados central para NetWeaver AS ABAP e, portanto, deve cumprir exigências elevadas enquanto o sistema está em execução.

Conforme **Erro! Fonte de referência não encontrada.**, ocorre à evidência da configuração clássica de um sistema SAP com suas camadas explicadas acima.

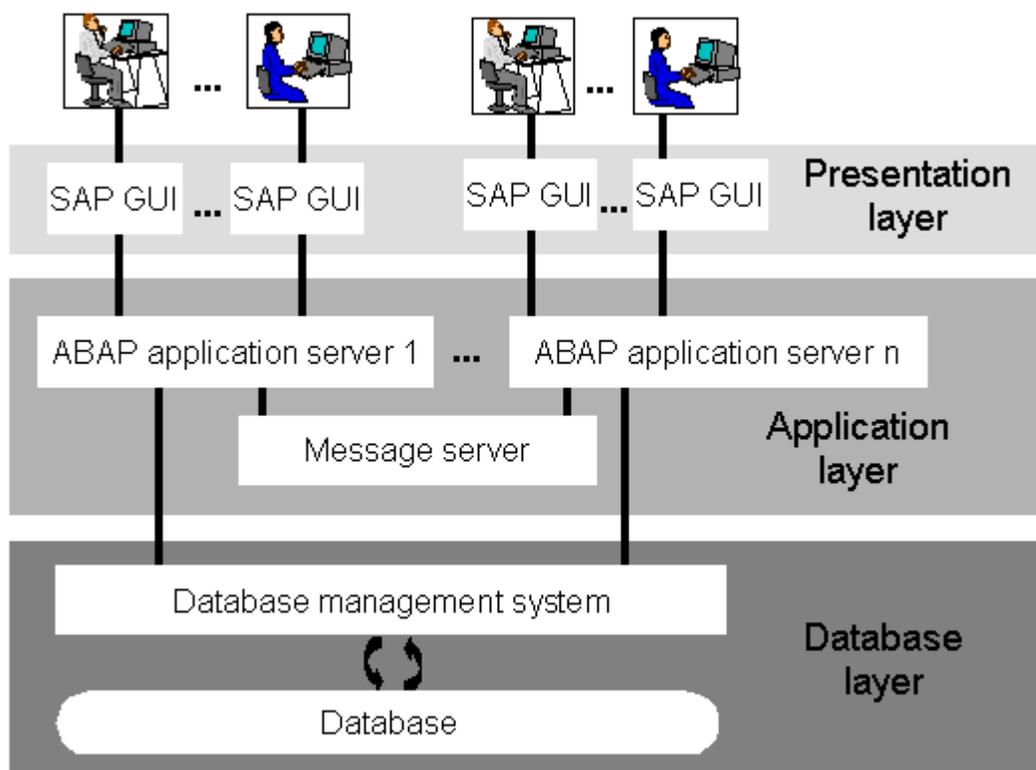


Figura 5. Visão orientada a Software do NetWeaver AS ABAP.

Fonte: SAP(2014).

3.2.2.1.3. Visão Orientada a Usuário

Para o usuário, os componentes visíveis do sistema SAP são aquelas que aparecem como uma janela na tela. Estas janelas são criadas pela camada de apresentação do NetWeaver AS ABAP, conforme Figura 6.

Antes de o usuário efetuar login no sistema SAP, ele deve iniciar um utilitário SAP GUI chamada SAP Logon, que é instalado na extremidade frontal. Em SAP Logon, o usuário escolhe um dos sistemas SAP disponíveis. O programa se conecta ao servidor de mensagens da NetWeaver AS ABAP no sistema SAP selecionado e obtém o endereço de um servidor de aplicação adequado (mais levemente usado) ABAP. Em seguida, ele inicia uma GUI SAP, ligado

a esse servidor de aplicativos. O programa SAP Logon, então, não é mais necessário para esta conexão.

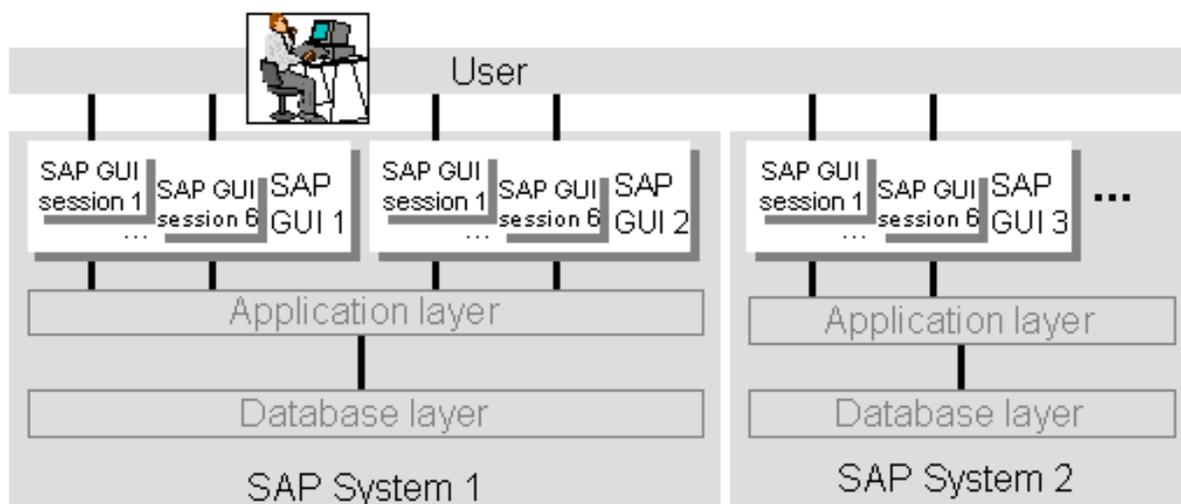


Figura 6. Visão orientada a Usuário do NetWeaver AS ABAP

Fonte: SAP(2014).

SAP GUI inicia a tela de logon. Depois que o usuário conectado com êxito, você verá a tela inicial do sistema SAP em uma janela na tela. Dentro SAP GUI, cada janela é representada como uma sessão. Depois de iniciar a sessão, o usuário pode abrir até mais cinco sessões dentro da única GUI SAP. Estes comportam quase como aplicações SAP independentes. As diferentes sessões permitem que você execute aplicativos diferentes em paralelo, independentemente um do outro.

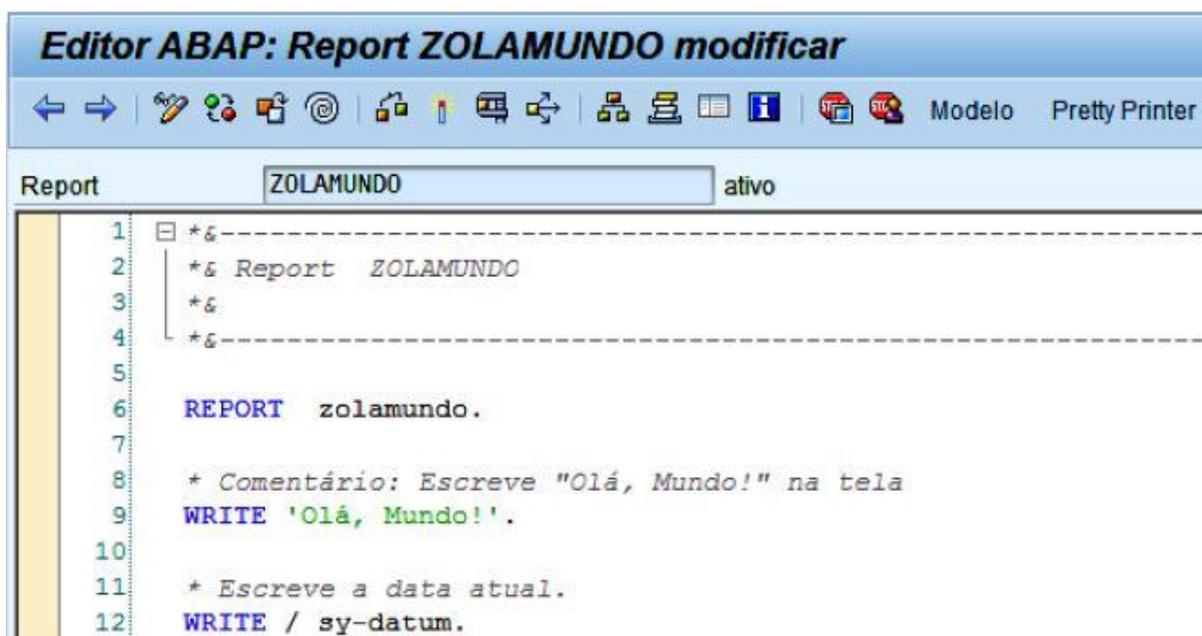
Dentro de uma sessão, o usuário pode executar aplicativos que se chamam mais janelas (tais como caixas de diálogo e janelas gráficas). Essas janelas não são independentes - que pertencem a sessão a partir do qual eles foram chamados. Estas janelas podem ser modal (a janela original não está pronto para entrada) ou amodal (ambas as janelas estão prontos para entrada e interagir uns com os outros).

O usuário pode abrir outras GUIs SAP, usando SAP Logon, para efetuar *logon* no mesmo sistema ou de outro sistema SAP. As GUIs SAP individuais e sessões de terminal correspondente de um usuário são totalmente independentes. Isso significa que você pode ter GUIs SAP

representam as camadas de apresentação de vários sistemas SAP aberto no seu computador *desktop*.

3.2.3. Linguagem ABAP

A linguagem ABAP, Advanced Business Application Programming, é uma linguagem de programação que tem declarações ABAP e comentários. Onde a primeira palavra é chave, seguido o operando, adições de mais palavras chaves e um ponto final, que encerra o comando. A Figura 7 exemplifica um código ABAP:



```

1  *-----
2  * Report  ZOLAMUNDO
3  *
4  *-----
5
6  REPORT  zolamundo.
7
8  * Comentário: Escreve "Olá, Mundo!" na tela
9  WRITE 'Olá, Mundo!'.
10
11 * Escreve a data atual.
12 WRITE / sy-datum.

```

Figura 7.Exemplo de linguagem ABAP

Fonte: MOURA(2012).

Existe também o chamado ABAP Workbench, ferramentas ABAP para desenvolvimento de objetos de uma aplicação ABAP, ou seja, criar tabelas, programas, transações, classes, etc.

Na Figura 8 ocorre a exemplificação de onde linguagem ABAP está inserida dentro do contexto SAP. Ao acessar o SAP via sistema ou WEB, camada de apresentação que se mostra para o usuário. Todo o manuseio feito pelo usuário é processado pela camada de aplicação, ou

seja, pela linguagem ABAP. É por meio dela que ocorre o acesso a camada de banco para busca e tratamento dos dados salvos.

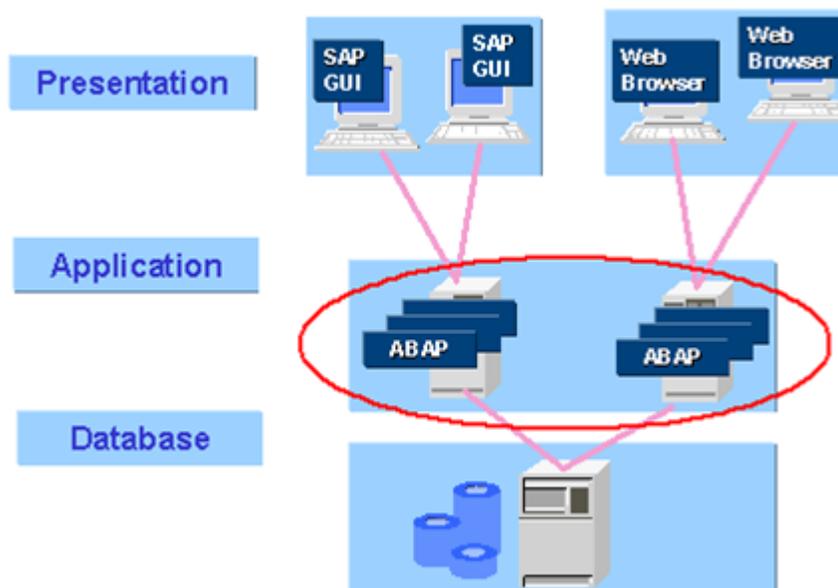


Figura 8. Exemplo de Processo no SAP

Fonte: SAP(2014).

Entretanto, a aplicação desenvolvida não é específica para um módulo específico, conforme detalhado acima. É desenvolvida dentro do SAP ERP.

3.3. Arquivo XML

A aplicação desenvolvida gera um arquivo XML com os acessos do SAP. Então, para um melhor entendimento, abaixo uma explicação detalhada de o que é um arquivo XML, para que ele serve e um exemplo para ver sua simplicidade e praticidade.

O XML é uma linguagem de marcação recomendada pela W3C para a criação de documentos com dados organizados hierarquicamente, tais como textos, banco de dados ou desenhos vetoriais. A linguagem XML é classificada como extensível porque permite definir os elementos de marcação. (MOREIRA, 2009).

Linguagens de marcação são utilizadas para definir formatos, maneiras de exibição e padrões dentro de um documento qualquer. Normalmente, elas não possuem qualquer estrutura de controle como as linguagens de programação tradicionais (por exemplo, comandos

condicionais ou de repetição). Dessa forma, elas servem basicamente para definir como um determinado conteúdo será exibido na tela ou como os dados estarão estruturados ao trafegar entre os diferentes módulos de um sistema. (PORTAL, 2013).

As linguagens de marcação se utilizam do conceito de marcador ou *tag*, que já trazem algum significado e que quando forem visualizados por algum sistema que as reconheça, irão saber como o conteúdo deve ser exibido. Por exemplo, suponha que se queira deixar todas as palavras começadas pela letra “A” em negrito. Para executar essa operação em HTML, seria necessário colocar as palavras começadas por esta letra entre os símbolos “” e “”, sendo que esses símbolos são um exemplo de marcadores válidos na linguagem. (PORTAL, 2013).

Principais usos para o XML

O XML oferece um modo extremamente flexível para fazer transferência de dados. A lista seguinte apresenta os principais exemplos onde o XML pode ser usado:

- Um documento comum.
- Um registro estruturado, como um registro de compromisso ou pedido de compra.
- Aplicações *Web* (Internet/intranet) que transferem dados.
- Um objeto com dados, como o formato persistente de um objeto ou controle ActiveX.
- Um registro de dados, como o conjunto de resultados (*dataset*) de uma consulta SQL.
- Meta-conteúdo sobre um local de Web, como Formato de Definição de Canal (CDF).
- Apresentação gráfica, como a interface de usuário de uma aplicação.
- Ligações entre informações e pessoas na Web. (REIS, 2011)

Na Figura 9, um exemplo de XML, como descrito anteriormente, é uma forma fácil e prática de compartilhar dados, já que é uma linguagem lida por qualquer software, mesmo de diferentes tecnologias, ou seja, como se fosse um idioma comum para culturas diferentes. Por isso é uma das formas mais usadas quando se fala em distribuição de dados, principalmente se tratando de tecnologias diferentes. Onde, tem-se informações de uma “receita” de bolo. E estão organizados em *tags*, representados pelos símbolos “<>”. A *tag* <receita> é o agrupador, tudo que está dentro, faz parte da receita, que tem subagrupadores, “<ingredientes>” e “<instruções>”,

onde tem informações dos ingredientes para fazer a receita da forma de fazer. Como se vê, é uma forma fácil e intuitiva de compartilhamento de dados:

Exemplos de Código XML

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<cd>
<titulo>Death Magnetic</titulo>
<artista>Metallica</artista>
<ano>2008</ano>
</cd>

<?xml version="1.0" encoding="iso-8859-1"?>
<receita nome="Pão Caseiro" tempo_de_preparo="10 minutos" tempo_assar="40 minutos">
<titulo>Pão Caseiro Simples</titulo>
<ingredientes>
<ingrediente quantidade="3" unidade="xícaras">Farinha</ingrediente>
<ingrediente quantidade="7" unidade="gramas">Fermento</ingrediente>
<ingrediente quantidade="1" unidade="xícaras" estado="morna">Água</ingrediente>
<ingrediente quantidade="1" unidade="xícaras">Açúcar</ingrediente>
<ingrediente quantidade="1" unidade="colheres de chá">Sal</ingrediente>
</ingredientes>
<instrucoes>
<passo>Misture o fermento com o açúcar e espere aproximadamente cinco minutos.</passo>
<passo>Misture todos os ingredientes, e dissolva bem.</passo>
<passo>Cubra com um pano e deixe por uma hora em um local morno.</passo>
<passo>Misture novamente, modele o formato e coloque na forma. Asse em forno médio.</passo>
</instrucoes>
</receita>
```

Figura 9. Exemplo de um Arquivo XML.

Fonte: PEREIRA(2009).

4. METODOLOGIA

O método usado neste trabalho consiste no desenvolvimento de uma aplicação de dentro do sistema de gestão de recursos da SAP. É uma aplicação independente de módulo e fica disponível no ambiente de desenvolvimento.

A aplicação possui a finalidade de oferecer uma função de gerenciamento de entradas do SAP e também a criação de um arquivo XML com essas entradas.

A linguagem utilizada para este desenvolvimento é o ABAP, linguagem esta proprietária ERP da SAP.

O programa contém telas para realizar o cadastro das entradas, informações de clientes, seus mandantes, informações de servidor de aplicação, etc. O mandante, falado anteriormente, são os ambientes que um cliente pode ter, por exemplo: o cliente X tem o mandante 100, 400 e 500. Onde o mandante 100 é o ambiente de desenvolvimento, o mandante 400 é o ambiente de testes e o mandante 500 é o ambiente de produção. Dessa forma, a aplicação possui o controle de todas as entradas de todos os clientes concentradas nessa aplicação.

Também podem ser visualizadas as informações cadastradas, em forma de relatório, onde são filtradas por cliente. Nesse trabalho também existe a opção de editar as informações inseridas, bem como também excluí-las. Isso através de botões no próprio relatório, onde a tabela do banco de dados será atualizada a partir das ações do usuário.

Por fim, a opção de gerar um XML com os registros selecionados pelo usuário, que é o tipo de arquivo que o sistema de gestão da SAP cria quando as entradas dos clientes são inseridas manualmente no sistema. Além de criar o arquivo XML, a aplicação salva automaticamente na pasta onde o sistema de gestão busca as informações das entradas para carregá-las ao acessá-lo.

5. ESTUDO DE CASO

Neste item será apresentada a aplicação para gerenciar os acessos aos ambientes SAP. Iniciando-se do acesso ao SAP, pela aplicação e seu funcionamento até o objetivo final, a geração do XML com os acessos.

Conforme Figura 10, o acesso ao ambiente SAP é feito pela aplicação do SAP GUI, aplicação está que armazenará todas as entradas cadastradas. Neste caso, existe apenas um ambiente cadastrado, o de desenvolvimento.

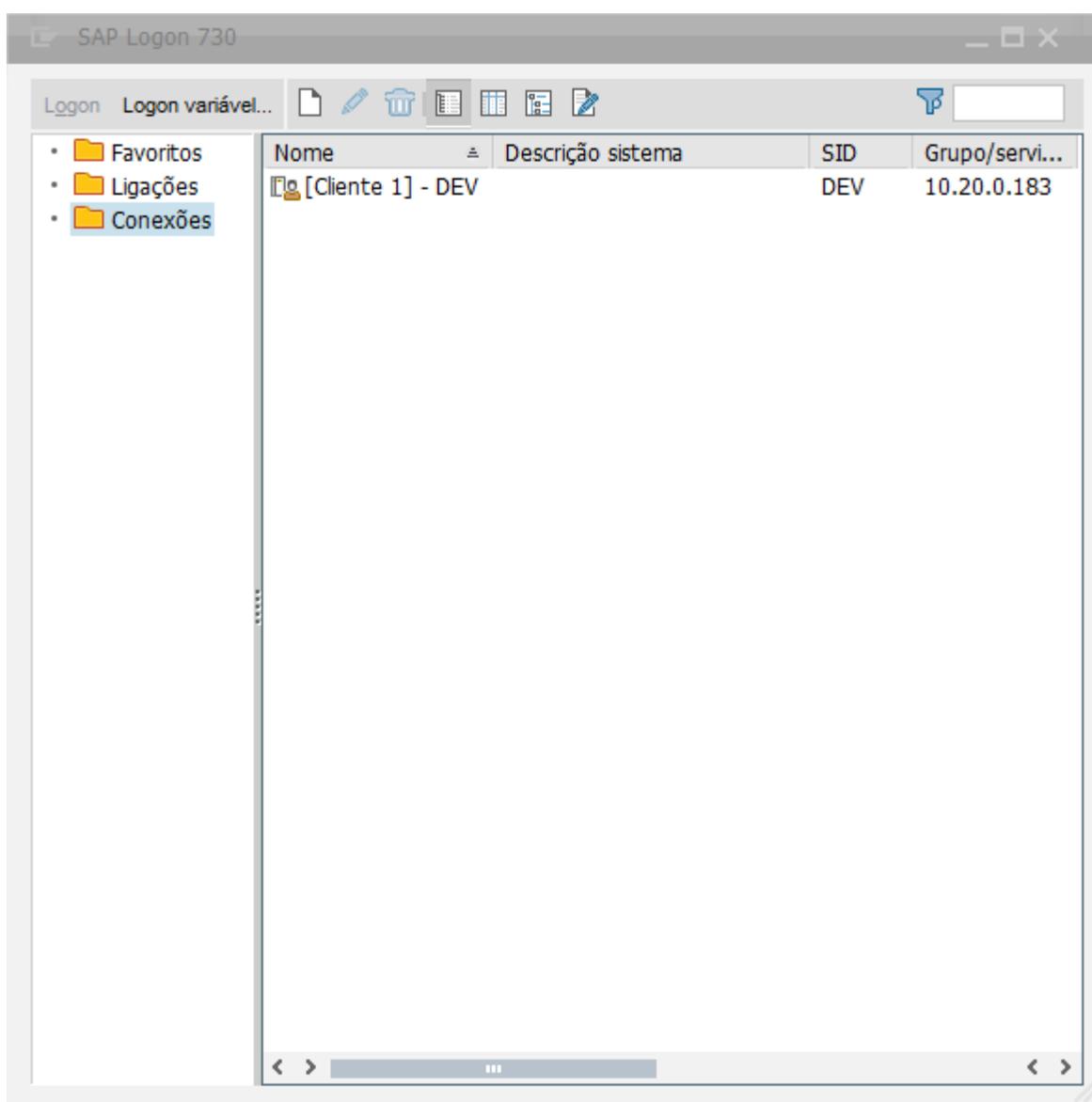


Figura 10. Tela da aplicação SAP GUI.

Conforme Figura 11 visualiza-se os dados de acesso ao ambiente Cliente 1. A Descrição é apenas para identificar o ambiente, é de entrada livre. O Número da Instância, que é a porta de acesso para o ambiente e o ID Sist. (SAP Sistema de Identificação) contém um código de três caracteres, como, por exemplo, C11, PRD, E56, etc. Para o exemplo apresentado é usado o DEV.

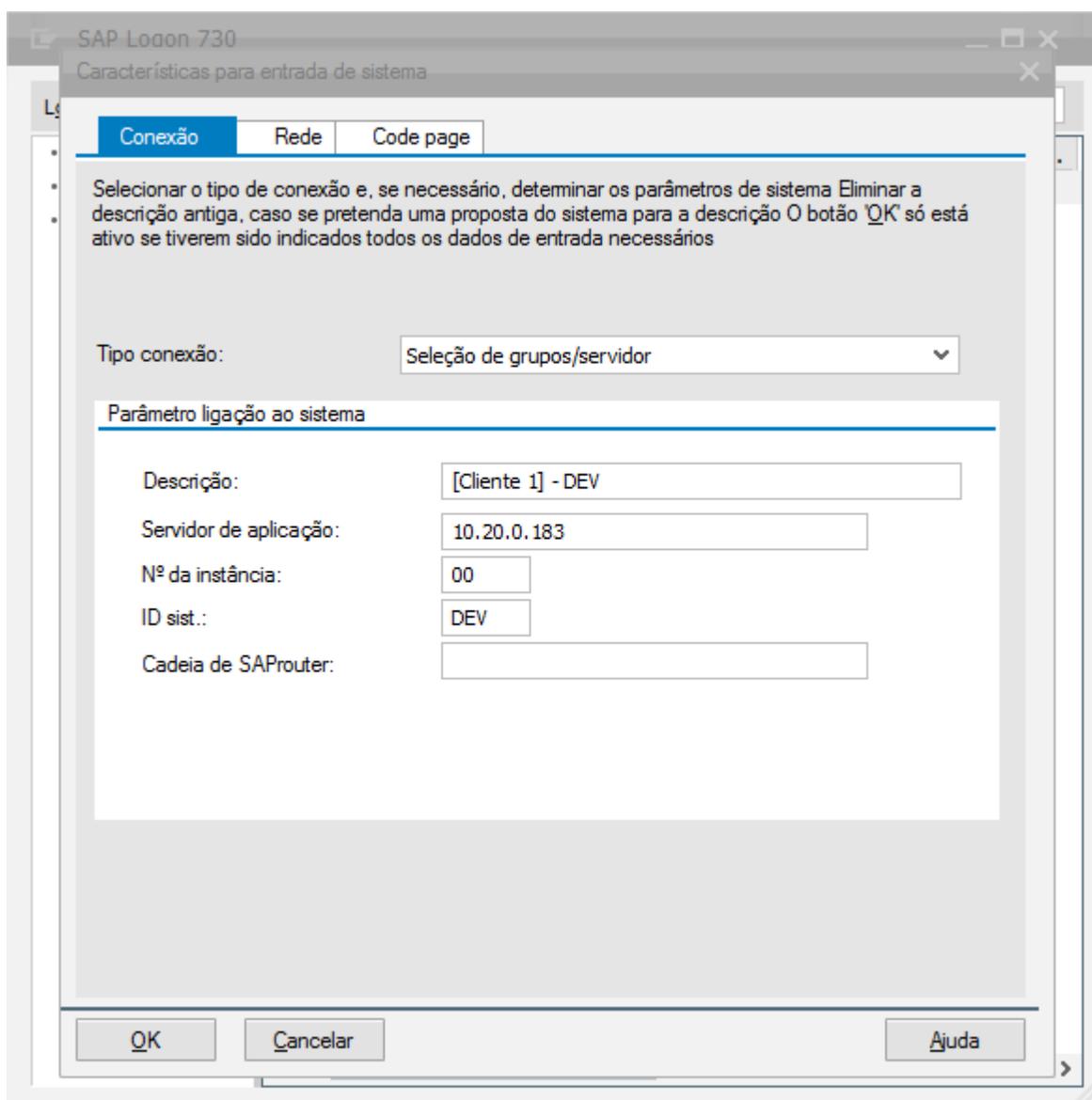


Figura 11. Dados de acesso de um ambiente no SAP GUI.

Após o cadastro do cliente, basta dar um duplo clique sobre o novo cadastro para que o acesso ao ambiente seja feito, conforme Figura 12. A tela do ambiente vem com o *Client*, para um ambiente, pode haver vários clientes. Este campo serve para separar dados sem haver a

necessidade de duplicar tabelas, sendo assim todas as tabelas no SAP contém esses campos como chave. O usuário e a senha são enviados pelo cliente, que é configurado pelo BASIS (Profissional responsável pelas configurações, acessos, etc., do sistema SAP). E o último campo é o idioma de exibição.

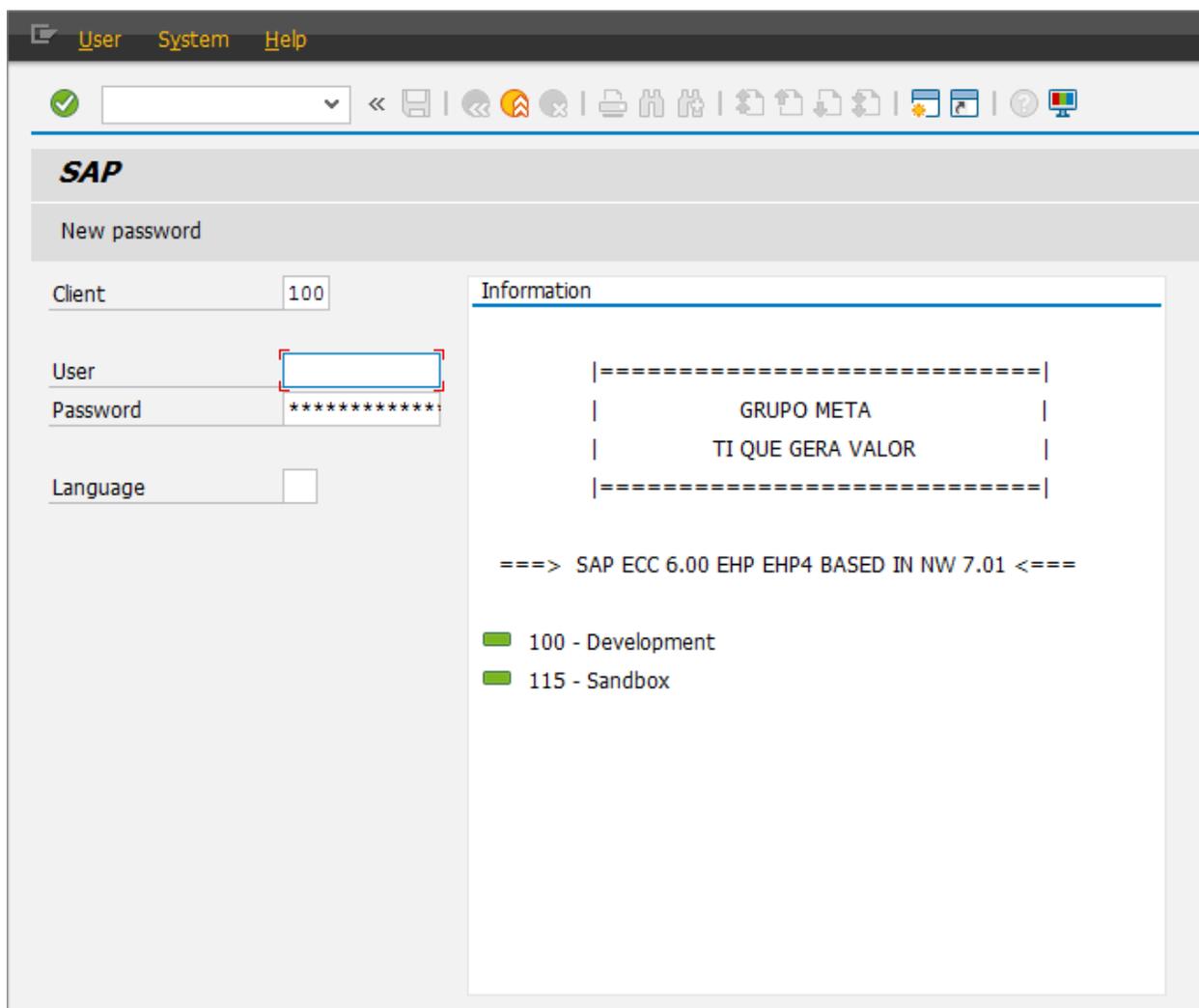


Figura 12. Tela de acesso a um ambiente SAP.

Após inserir as informações necessárias para acesso, finalmente se está dentro do ambiente do cliente, conforme Figura 13.

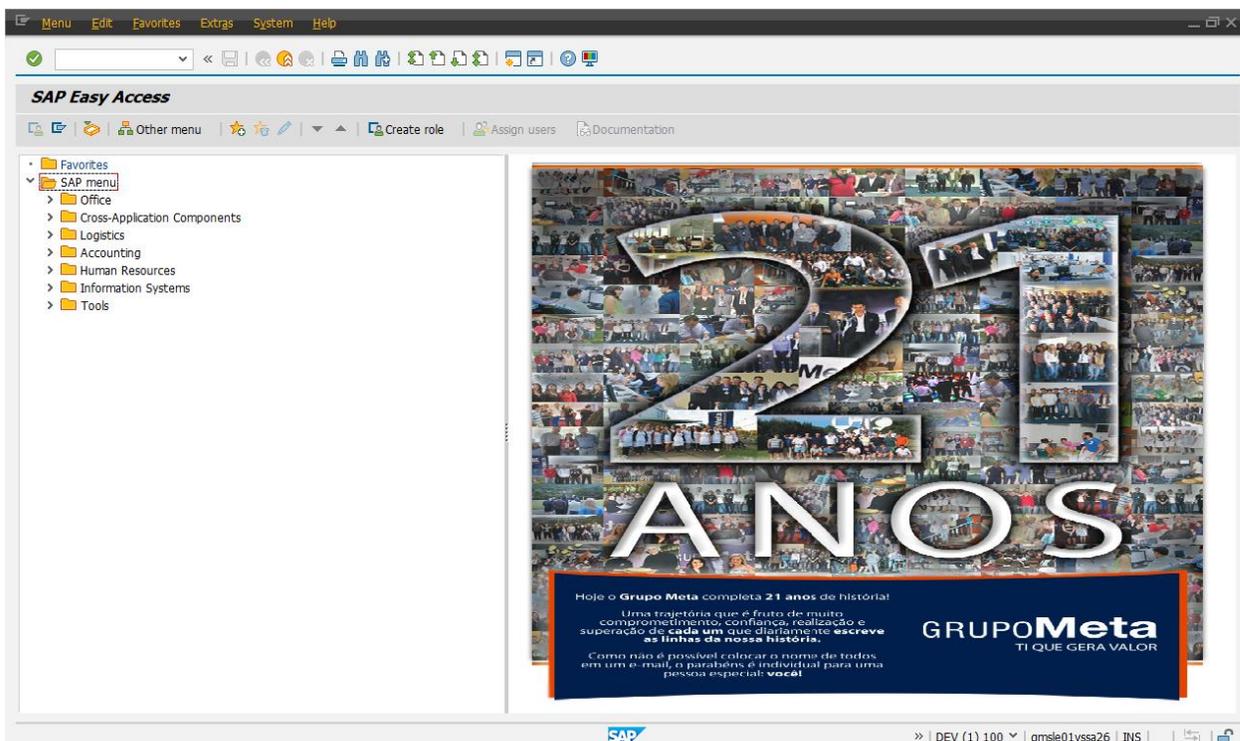


Figura 13. Por dentro de um ambiente conectado.

Após conexão ao ambiente, podem-se executar as aplicações disponíveis. Para tal, é necessário digitar o nome da transação na barra de comandos e pressionar “Enter”. Uma transação nada mais é que um nome fantasia para um programa, que é escolhido pelo cliente. Sendo assim, a transação pode ter o nome diferente do programa em si. Na figura 14, contém o nome da transação deste trabalho, o ZTSAPLOGON.

No SAP, todas as aplicações customizadas no *standard*, devem ser iniciadas com a letra Z. Esta regra serve também para criação de tabelas, campos de tabelas, etc. Ao tentar criar um programa customizado sem a letra Z no início, será requisitada uma chave de acesso onde somente a SAP pode fornecer. Mas só é usada em últimos casos, pois própria SAP não recomenda a utilização.

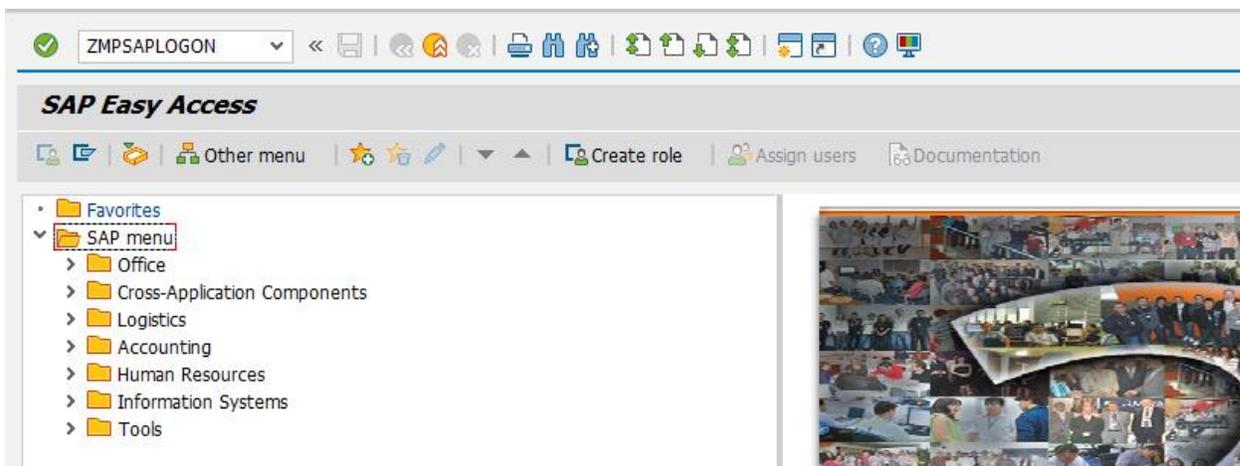


Figura 14. Executando aplicações no SAP.

Ao acessar a transação ZTSAPLOGON, carrega-se a primeira tela, onde é possível gerenciar Clientes, Ambientes e Credenciais de Acesso e realizar criação do XML (SAPLOGON.INI) com essas informações, conforme Figura 15.

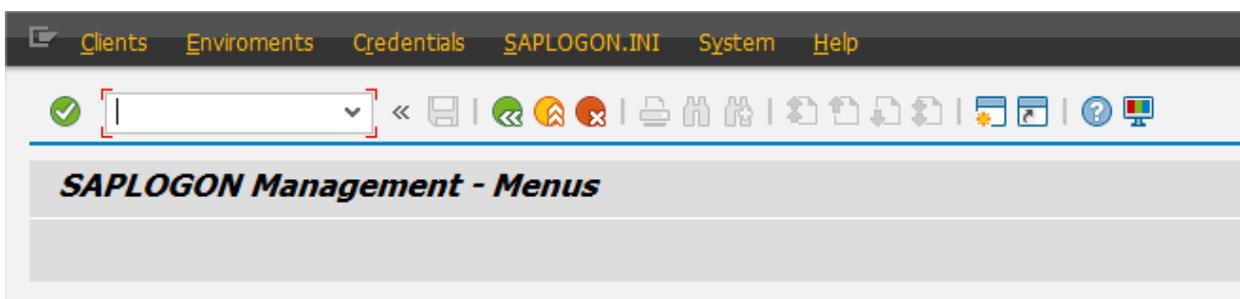


Figura 15. Tela inicial do programa ZTSAPLOGON.

Para cadastrar um novo cliente, basta selecionar a opção “Clients” no topo da tela e selecionar a opção “Create” e, para modificar, selecionar a opção “Update”. Essas duas opções estão presentes também para as opções “Enviroments” e “Credentials”, conforme Figura 16.

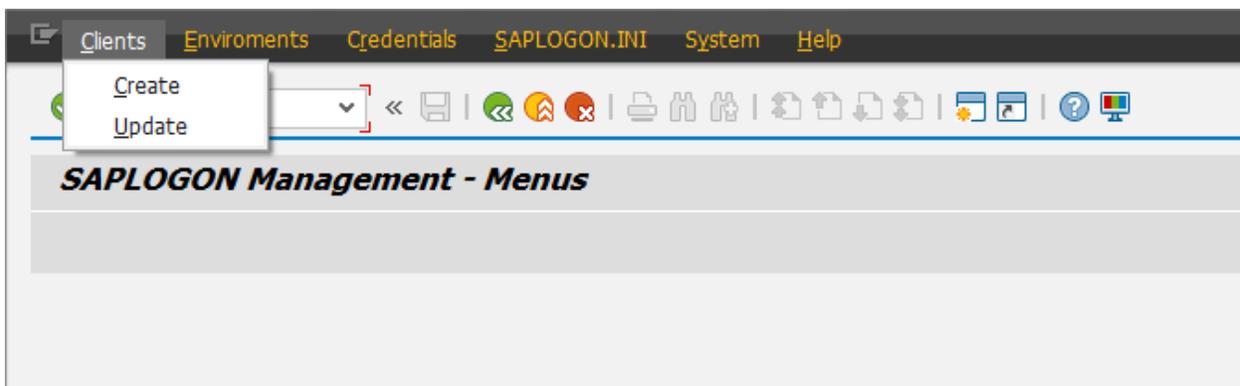


Figura 16. Funções do programa ZTSAPLOGON.

Na Figura 17, será cadastrado um cliente. Basta informar o nome do cliente, se for VPN (Virtual Private Network), marcar a *flag* e inserir a documentação referente à instalação desta VPN e marcar também se é um cliente ativo. Após preencher as informações, basta pressionar o botão de criação.

The image shows a screenshot of the 'SAPLOGON Management - Client Create' form. The form has a title bar with the text 'SAPLOGON Management - Client Create'. Below the title bar, there are several input fields and checkboxes. The 'Client:' field contains the number '1'. The 'Name:' field contains the text 'Cliente 2'. The 'VPN:' checkbox is checked. The 'Active:' checkbox is also checked. The 'VPN Documentation:' field contains the path 'C:\SAP\VPN.pdf'. At the bottom left of the form, there is a 'Create' button.

Figura 17. Cadastrando um cliente.

Na Figura 18, a evidência de que os dados foram salvos na tabela ZTCLIENTS, criada para receber os registros dos clientes.

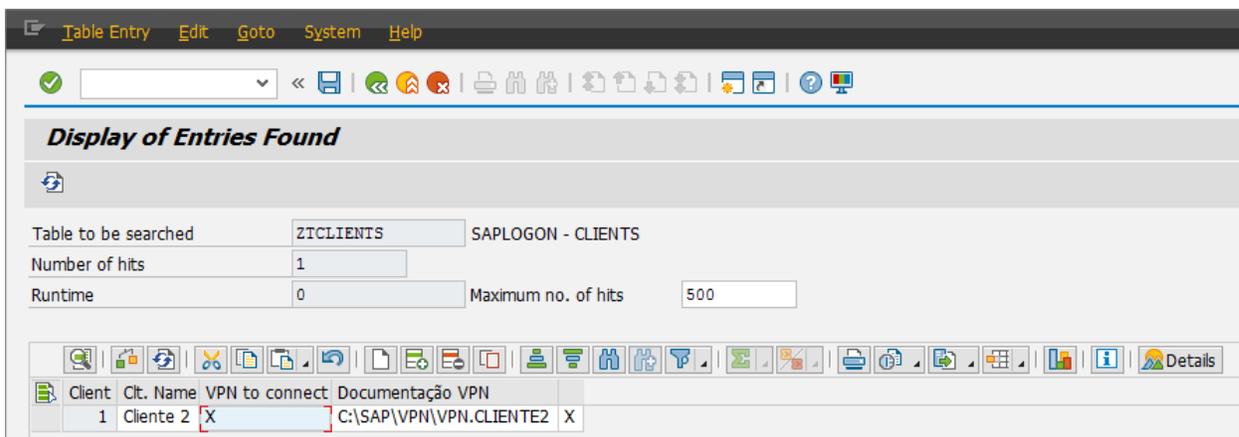


Figura 18. Registro de um cliente salvo na tabela ZTCLIENTS.

Na Figura 19, são cadastradas as informações de acesso ao ambiente do cliente cadastrado anteriormente. Com isso, basta inserir o código do cliente cadastrado. A descrição, que é de livre entrada. Já as informações de instância, servidor de aplicação, o ID do sistema e o SAPRouter são fornecidos pelo cliente. O SAPRouter não é um campo obrigatório, tanto que para o exemplo este campo não está sendo preenchido.

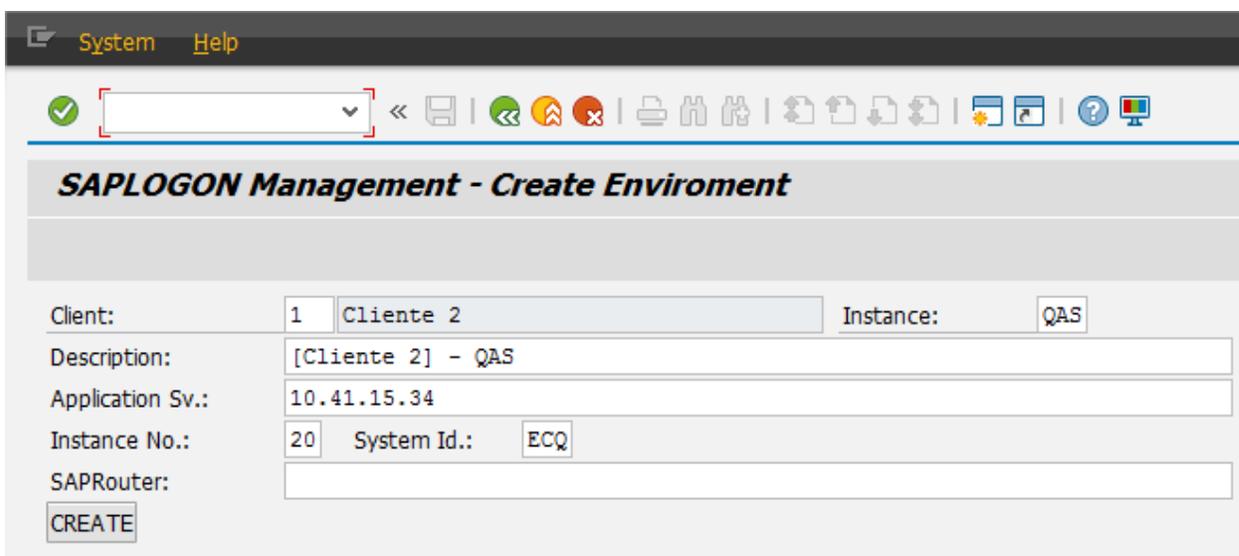


Figura 19. Cadastrando informações de acesso ao ambiente do cliente.

Na Figura 20 a evidência de que os dados foram salvos na tabela ZTENVIROMENTS, criada para receber os registros dos ambientes.

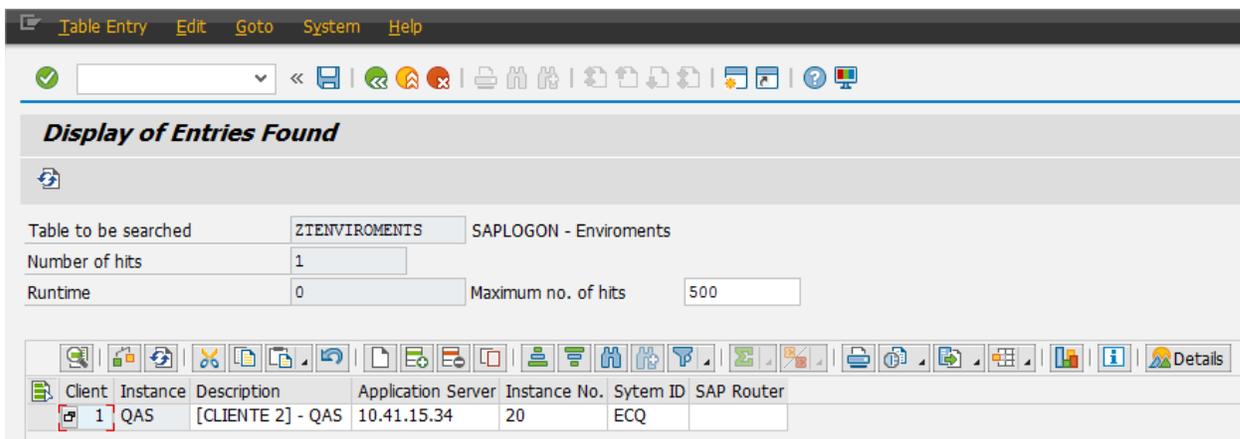


Figura 20. Registro de um cliente salvo na tabela ZTENVIROMENTS.

Na Figura 21 ocorre o cadastramento do usuário e senha do ambiente cadastrado. Ao inserir o código do cliente, os campos da descrição do cliente, instância e o mandante são cadastrados automaticamente, ficando apenas o nome do usuário e senha para cadastramento.

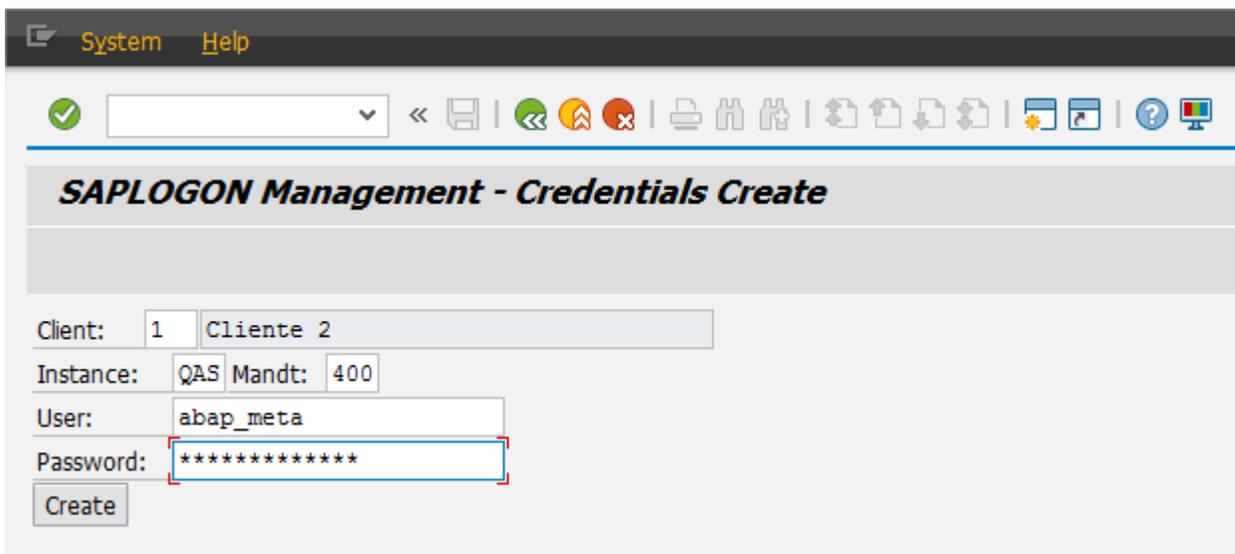


Figura 21. Cadastrando informações de usuário para acesso ao ambiente.

Na Figura 22 a evidência de que os dados foram salvos na tabela ZTCREDENTIALS, criada para receber os registros de informações de usuário.

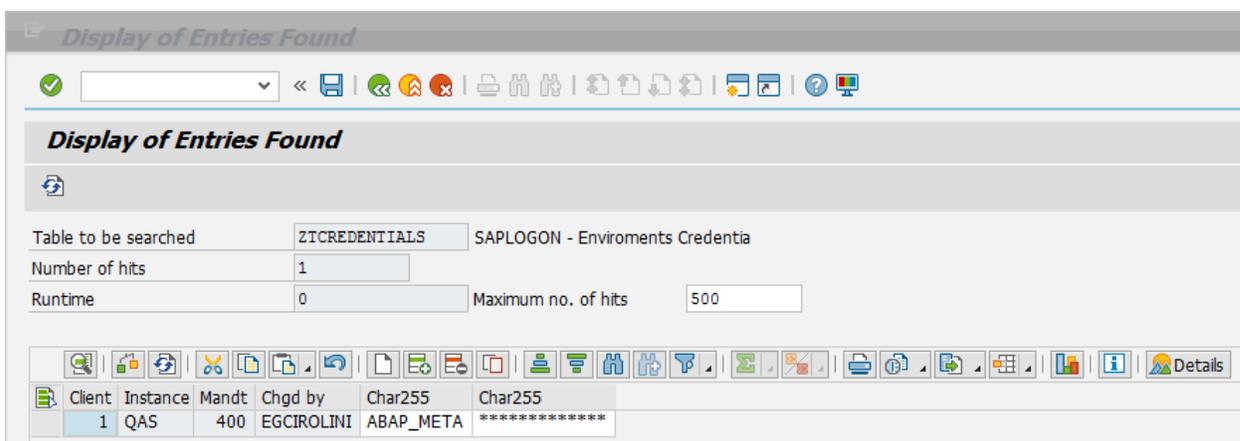


Figura 22. Registro de um cliente salvo na tabela ZTCREDENTIALS.

Para criar o arquivo XML com a entrada criada, basta acessar SAPLOGON.INI e selecionar a criação, conforme Figura 23.

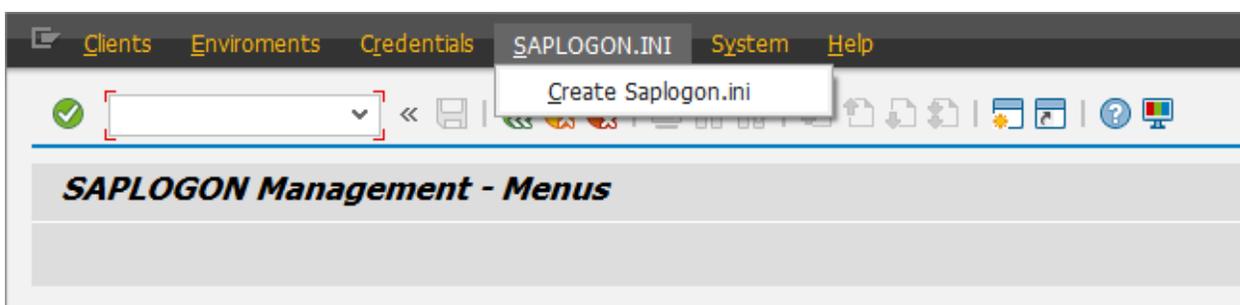


Figura 23. Criando o arquivo XML com a nova entrada.

Após será exibida uma tela com dois relatórios. No primeiro relatório, contém a lista e ambientes cadastrados com informações como o “ID”, “nome” e se já foi está inserido no XML ou não. O que define está diferença é coluna “Check”. Se ícone representativo for um sinal negativo, ainda não está contido no XML. Se estiver representado com um sinal de mais, está contido no XML, conforme Figura 24 e Figura 25.

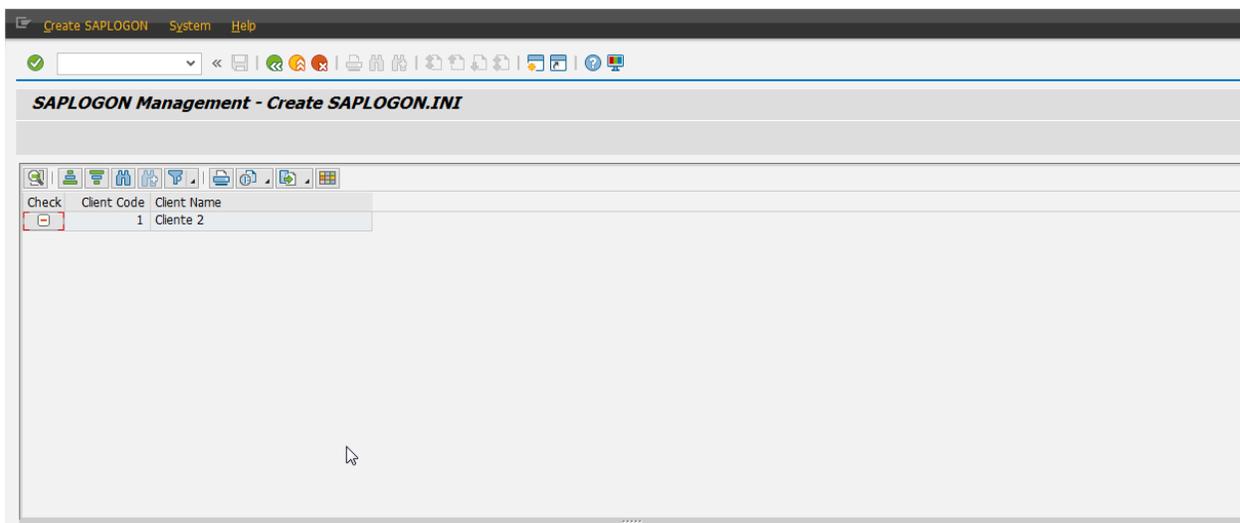


Figura 24. Exemplo de cadastrado de cliente que não está contido no XML.

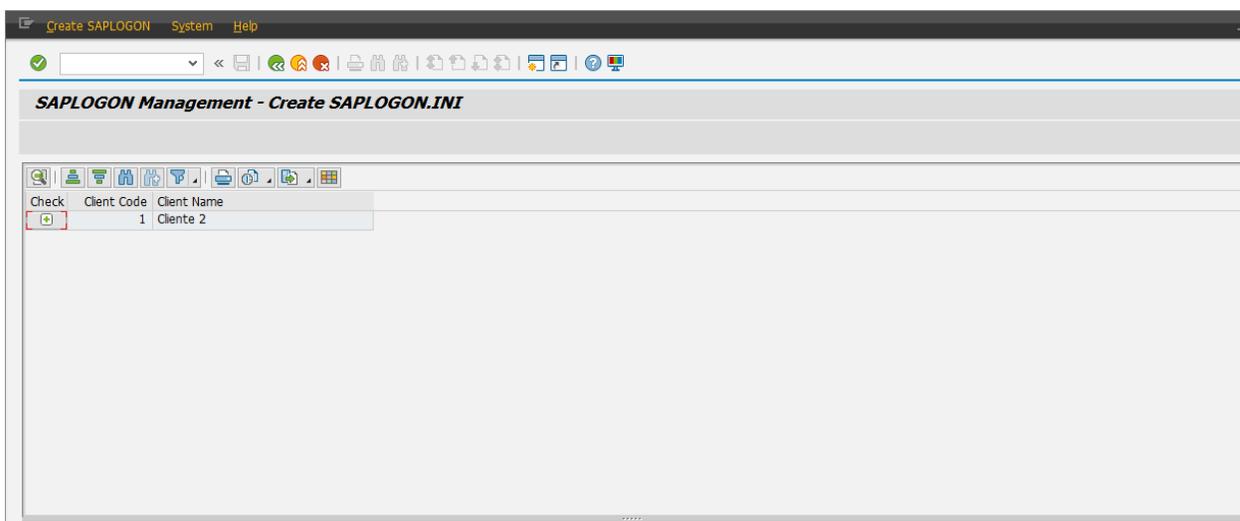


Figura 25. Exemplo de cadastrado de cliente que está contido no XML.

Para criar o XML com a com o novo ambiente, basta clicar em cima do campo “Check” que estiver com o sinal negativo. Após este comando, um relatório na parte debaixo será exibido com mais algumas informações, para saber exatamente qual ambiente vai ser criado. Para este exemplo, e um ambiente de teste (QAS), conforme Figura 26. Para remover o novo ambiente, basta clicar novamente no campo “Check” onde estiver com o valor positivo.

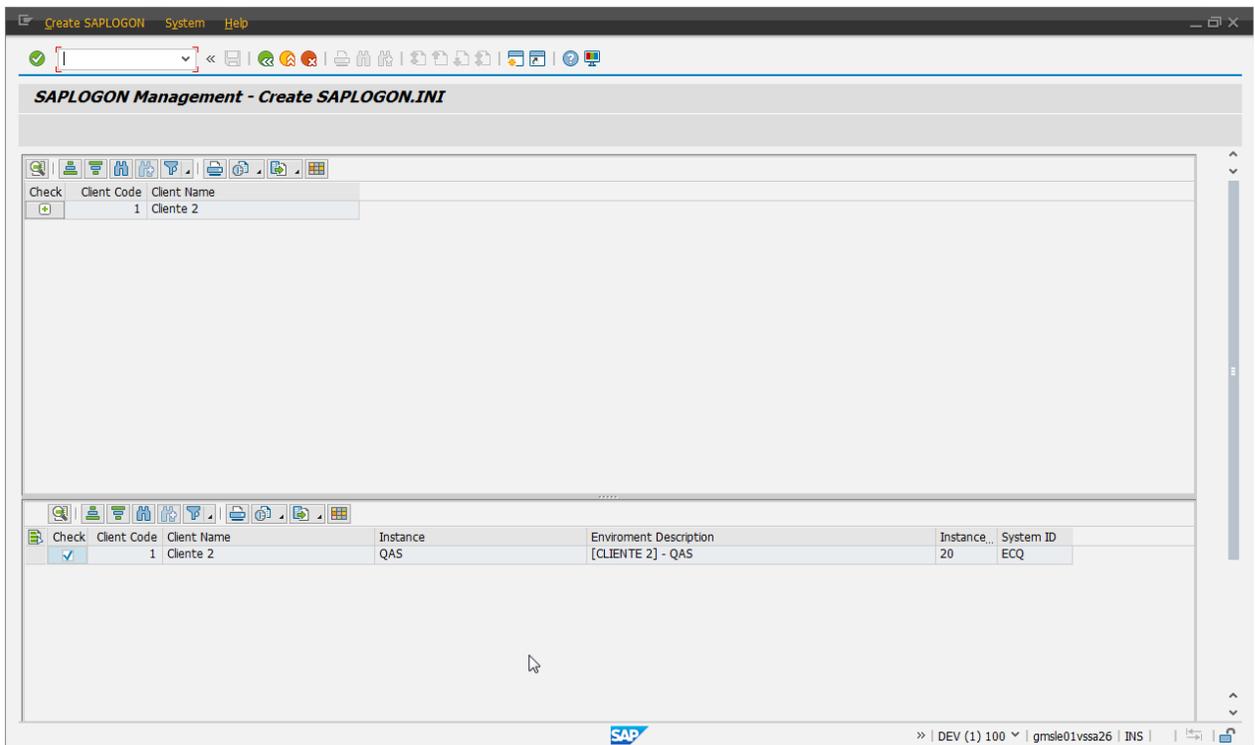


Figura 26. Selecionando um ambiente para gerar uma entrada no XML.

Para gerar efetivamente o XML com o novo ambiente, basta selecionar a opção “Create” SAPLOGON, conforme Figura 27.

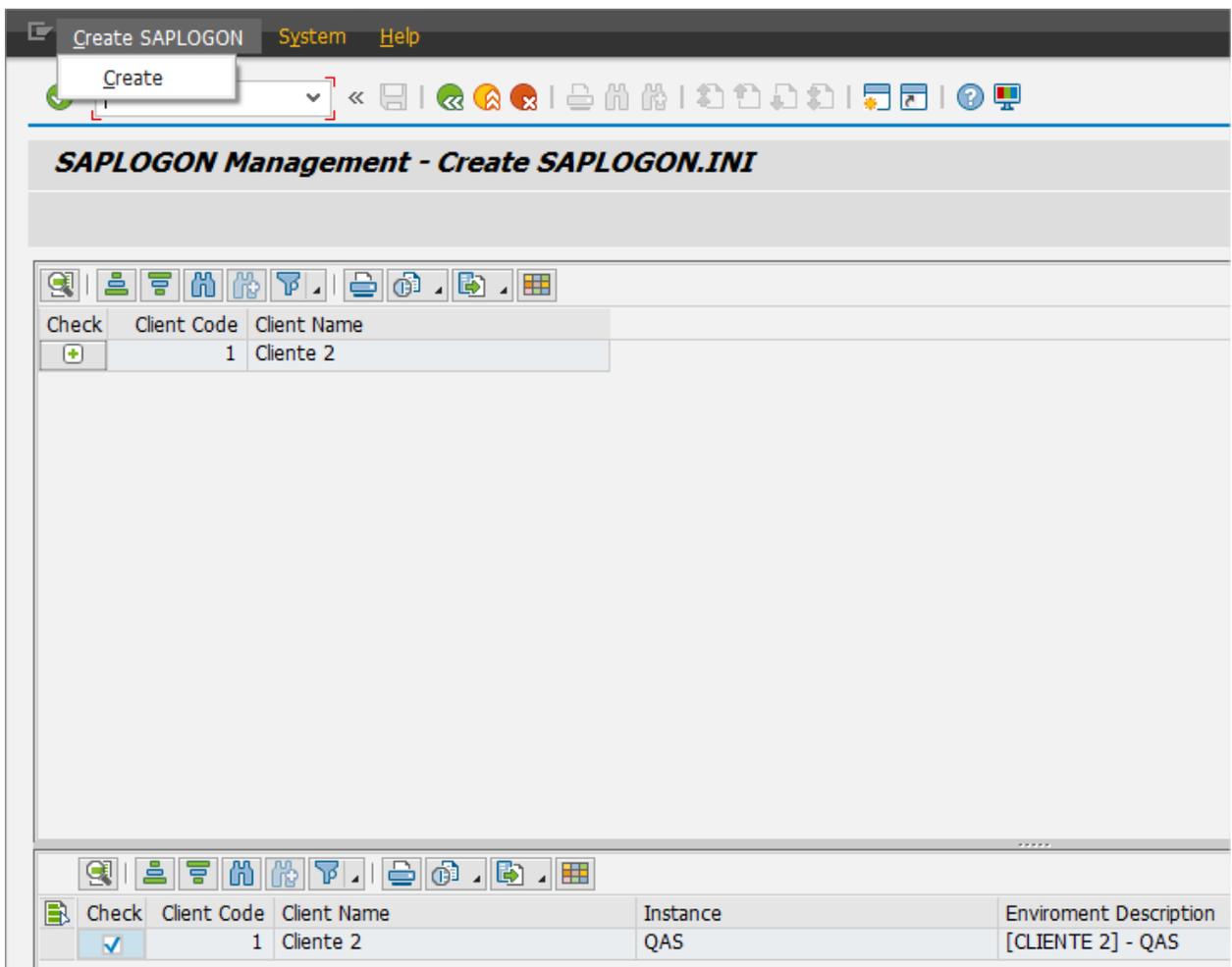


Figura 27. Criando um XML com a nova entrada.

Após criar o XML, basta acessar o SAP GUI para confirmar que o novo ambiente foi inserido e pronto para acesso, conforme Figura 28.

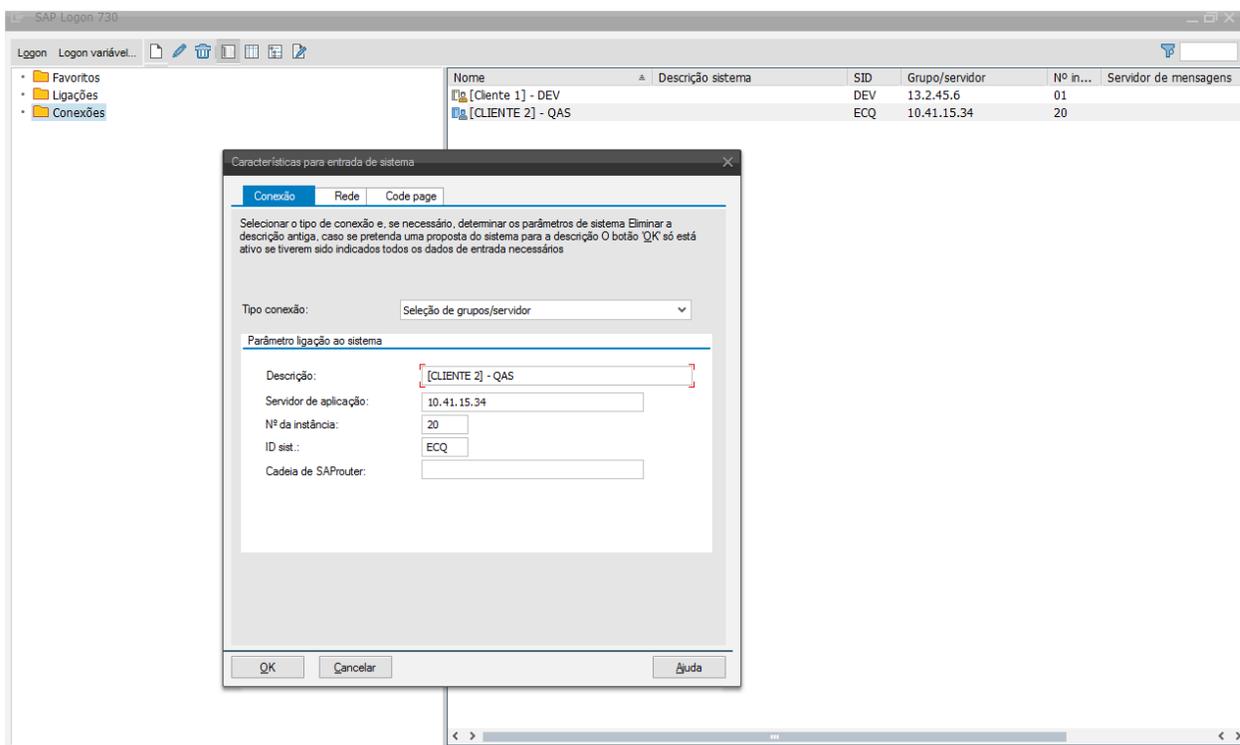


Figura 28. SAP com o novo ambiente disponível.

Na Figura 29 a estrutura do XML que foi gerado para o novo ambiente para que o SAP GUI consiga interpretar. Esta estrutura foi gerada automaticamente pelo método disponibilizado pelo próprio sistema de gestão. Desta forma, não há preocupação com a montagem das tags do XML, e sim apenas os dados necessários para a criação do XML.

```

[Configuration]
SessManNewKey=02
[EntryKey]
Item1=
[Router]
Item1=
[Router2]
Item1=
[RouterChoice]
Item1=0
[Server]
Item1=10.41.15.34
[Database]
Item1=20
[System]
Item1=3
[Description]
Item1=[CLIENTE 2] - QAS
[Address]
Item1=
[MSSysName]
Item1=ECQ
[MSSrvName]
Item1=
[MSSrvPort]
Item1=
[SessManKey]
Item1=-1
[SncName]
Item1=
[SncChoice]
Item1=-1
[SncNoSSO]
Item1=0
[Codepage]
Item1=1100
[CodepageIndex]
Item1=-1
[Origin]
Item1=USEREDIT
[LowSpeedConnection]
Item1=0
[Utf8Off]
Item1=0
[EncodingID]
Item1=DEFAULT_NON_UC
[ShortcutType]
Item1=0
[ShortcutString]
Item1=
[ShortcutTo]
Item1=
[ShortcutBy]
Item1=

```

Figura 29. Estrutura do XML gerado para o novo ambiente.

Para gerar o XML foi usado o método CREATE_SAPLOGON da classe ZCL_SAPLOGON_CREATE. Na figura 30 visualiza-se a classe e seus métodos. Na Figura 31 tem-se o código do método CREATE_SAPLOGON.

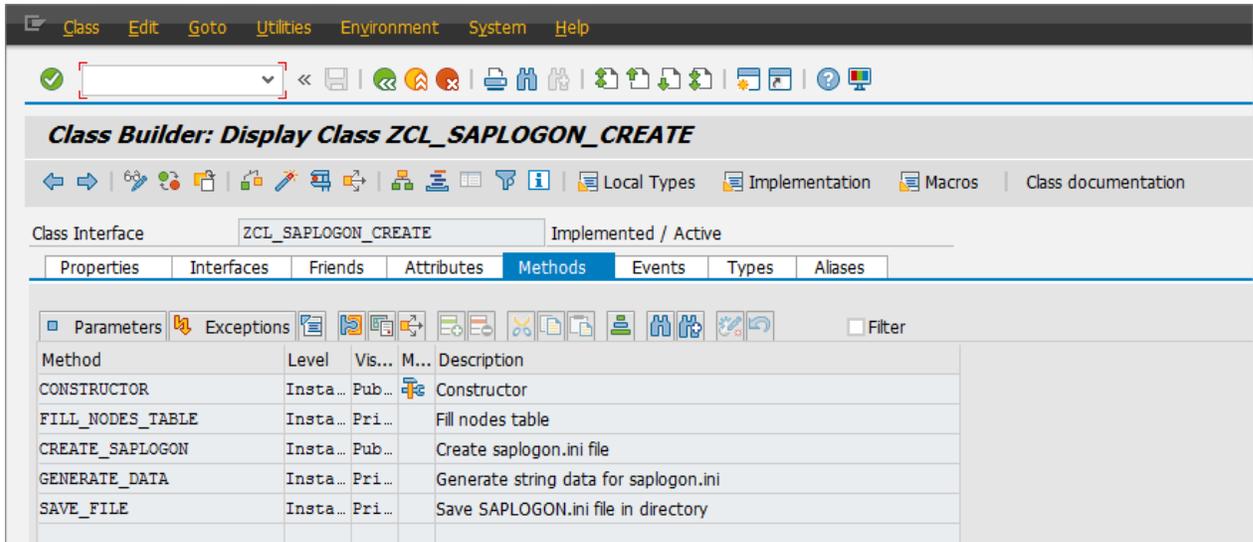


Figura 30. Classe ZCL_SAPLOGON_CREATE.

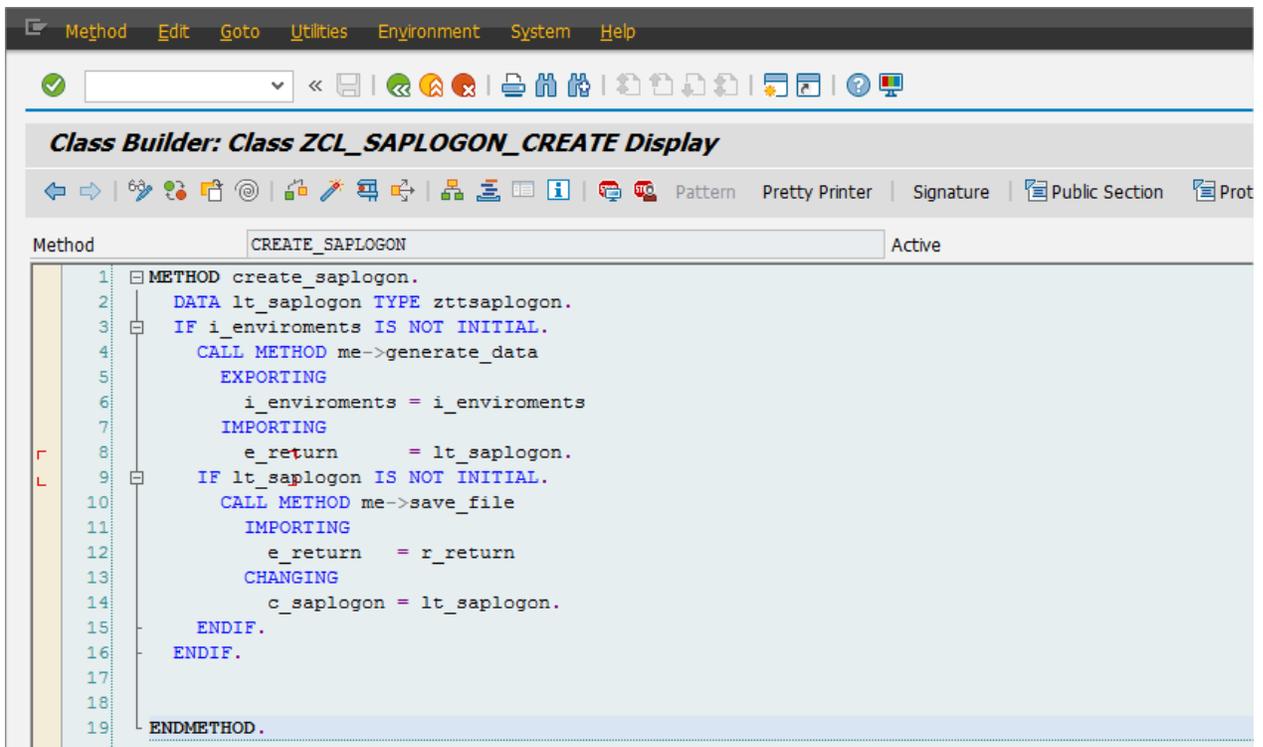


Figura 31. Método CREATE_SAPLOGON.

6. CONSIDERAÇÕES FINAIS

Este trabalho teve como objetivo substituir um trabalho realizado manualmente e de forma repetitiva por uma aplicação que gerencie todos esses acessos, gerando uma organização, agilidade e praticidade no acesso a informações sobre ambientes de acessos ao sistema SAP.

O diferencial desta aplicação é reunir em um único lugar todas as informações necessárias para acessar qualquer cliente nele cadastrado. Assim não se faz necessário ficar pedindo a cada usuário se o mesmo tem as informações necessárias para acessar tal ambiente, gerando um ganho de performance.

Por fim, uma sugestão de melhoria seria criar uma nova função para criar repositórios de programas por clientes. Todo programa desenvolvido ou que tenha alguma relevância técnica que se queira guardar para pesquisa futura, grava-se no programa de gerenciamento de ambientes para fim de consulta.

7. REFERÊNCIAS BIBLIOGRÁFICAS

ALECRIN, O que é ERP (Enterprise Resource Planning)?. Disponível em <<http://www.infowester.com/erp.php>>. Acesso em: 04 de Outubro de 2016.

ALENCAR, Tecnologia nas empresas. Disponível em: <<http://www.administradores.com.br/artigos/tecnologia/tecnologia-nas-empresas/70427/>>.

Acesso em: 04 de Outubro de 2016.

CONVENIA, A importância da agilidade empresarial no mercado de hoje. Disponível em <<http://www.convenia.com.br/blog/a-importancia-da-agilidade-empresarial-no-mercado-de-hoje/>>. 04 de Outubro de 2016.

GASPAR, O que é sistema ERP?. Disponível em: <<http://www.pwi.com.br/blog/o-que-e-sistema-erp/>>. 04 de Outubro de 2016.

MOREIRA, O que é XML?. Disponível em: <<http://www.tecmundo.com.br/programacao/1762-o-que-e-xml-.htm>>. 04 de Outubro de 2016.

MOURA, Ferramenta ABAP para gerenciamento de casos de teste e gestão de defeitos, 2012, São Leopoldo: UNISINOS, 2012.

PAGOTI, Netweaver – Suas versões e descobrindo suas funcionalidades. Disponível em: <<http://abap101.com/2013/11/29/netweaver-suas-versoes-descobrimdo-suas-funcionalidades/>>.

Acesso em: 25 de Agosto de 2016.

PORTAL, Linguagem de Marcação. Disponível em: <<http://www.portaleducacao.com.br/informatica/artigos/31639/linguagem-demarcacao#ixzz3cDioljoq>>. Acesso em: 04 de Outubro de 2016.

REIS, O que é esse tal de XML? Disponível em <<http://programandofacil.com.br/artigos/xml/o-que-e-esse-tal-de-xml/>>. Acesso em: 04 de Outubro de 2016.

SAP. Empresa alemã criadora do Software de Gestão Empresarial SAP ERP. Disponível em <<http://www.sap.com>>. Acesso em: 04 de Outubro de 2016.

SAP. Overview of NetWeaver AS ABAP. Disponível em<http://help.sap.com/saphelp_nw70/helpdata/en/fc/eb2e97358411d1829f0000e829fbfe/content.htm> Acesso em: 04 de Outubro de 2016.

SAP. The ABAP Programming Language. Disponível em <http://help.sap.com/saphelp_nw70/helpdata/en/c9/5472f6787f11d194c90000e8353423/content>.

htm?frameset=/en/c3/225b5354f411d194a60000e8353423/frameset.htm¤t_toc=/en/d3/2e974d35c511d1829f0000e829fbfe/plain.htm&node_id=36&show_children=false>.. Acesso em: 11 de Setembro de 2016.

SOUSA,O que é Sistema ERP?. Disponível em <<http://www.focadoemti.com.br/o-que-e-sistema-erp/>>. Acesso em: 11 de Setembro de 2016.