

PM

**Solução de Extração e Visualização  
de dados SAP - Via Remota**

PROJETO DE MESTRADO

**Herlander Augusto Santos Silva**

MESTRADO EM ENGENHARIA INFORMÁTICA



UNIVERSIDADE da MADEIRA  
*A Nossa Universidade*  
www.uma.pt

outubro | 2018

# **Solução de Extração e Visualização de dados SAP – Via Remota**

PROJETO DE MESTRADO

**Herlander Augusto Santos Silva**

MESTRADO EM ENGENHARIA INFORMÁTICA

ORIENTADOR

Leonel Domingos Telo Nóbrega

**Herlander Augusto Santos Silva**

(Licenciado)

Dissertação submetida à Universidade da Madeira para obtenção do grau de mestre em  
Engenharia Informática

Constituição do júri de provas públicas:

Karolina Baras, Prof. Dra. da Universidade da Madeira, Presidente

Leonel Nóbrega, Prof. Dr. da Universidade da Madeira, Vogal

Eduardo Marques, Prof. Auxiliar da Universidade da Madeira, Vogal

Funchal, 25 de março 2018

## Abstract

Nowadays, it is common to deal with large amounts of information which makes the software used to manage and visualize this information, increasingly important for small, medium and large companies. Due to this growing need to deal with large amounts of data, various financial management and business intelligence software appear, such as SAGE, which is widely used in Portugal, Pentaho which was approached in the classes of decision support systems, the SAP system, which is addressed in this thesis, and on which the data extractor is developed, with the help of the Visual Studio WindowsForms.

This thesis focuses on the creation of an extractor that aims to extract large amounts of information from the SAP system and present this information to any user who has the appropriate accesses. It is intended that a person who is unaware, of the ABAP programming language, or the way in which this information is researched within the SAP environment, can obtain this information using the data extractor.

In order to give more context to the reader of how the SAP system works and the data extractor, it is possible to see in chapter four, the system architecture with a brief explanation of how the system extracts said information, it is also possible to visualize a sequence diagram that helps the reader, understand the user's interactions with the extractor by using the methods necessary for obtaining and processing data.

In the tests done to the extractor it is possible to identify the amount of time required to log in to the application, to perform tables lookups, the time required to obtain the fields of a selected table, you can also identify the differences of time between a search with filters and one without filters, and by testing the export methods, it is possible to identify which is the fastest.

## Keywords

ABAP

Data

Extractor

Management

Information

SAP

## Resumo

Nos dias de hoje, é comum lidar com um enorme e variadíssimo leque de informação o que faz com que o *software* utilizado para gerir e veicular a visualização desta informação, seja cada vez mais relevante para pequenas, médias e grandes empresas. Devido a esta crescente e progressiva necessidade de proceder ao tratamento de grandes volumes de dados, surgiram diversos *softwares* de gestão financeira e de inteligência de negócios, tais como a SAGE - que é muito utilizada em Portugal, o Pentaho - o qual foi abordado nas aulas de Sistemas de Apoio à Decisão, e o sistema SAP, o qual é abordado nesta tese, e sobre o qual é desenvolvido o extrator de dados, com o auxílio da plataforma .NET da Microsoft.

Esta tese foca-se na criação de um extrator que tem como objetivo efetuar, tal como o nome indica, a extração de grandes quantidades de informação do sistema SAP, e apresentar esta informação a qualquer utilizador que possua os devidos acessos. Pretende-se que uma pessoa que desconheça a linguagem de programação ABAP, ou a forma como esta informação é pesquisada dentro do ambiente SAP, seja capaz de aceder à mesma através da utilização do extrator de dados.

De forma a dar mais algum contexto ao leitor acerca de como funciona o sistema SAP e o extrator de dados, exploramos, ao longo do capítulo quatro, a arquitetura do sistema com uma breve explicação de como este procede à obtenção de informação; no âmbito destes considerandos, é possível também visualizar um diagrama de sequência que ajuda a compreender as interações do utilizador com o extrator, através da utilização dos métodos necessários para a obtenção e tratamento de dados.

Ao longo dos testes efetuados ao extrator, é possível identificar o tempo necessário para efetuar o *login* na aplicação, efetuar pesquisas de tabelas, apurar o intervalo necessário para obtenção dos campos de uma tabela selecionada, identificar as diferenças de tempo entre uma pesquisa com filtros e uma sem filtros, e através dos testes aos métodos de exportação, torna-se possível identificar qual o mais rápido.

## Palavras-Chave

ABAP

Dados

Extrator

Gestão

Informação

SAP

## Agradecimentos

Em primeiro lugar, agradeço às duas pessoas mais importantes da minha vida, o meu pai e a minha mãe, por todo o apoio e incentivo que me ofereceram durante todos os anos da minha vida académica.

Em segundo lugar, agradeço ao professor Leonel Nóbrega toda a orientação, disponibilidade e conhecimento que me transmitiu durante a elaboração desta tese.

E em terceiro lugar agradeço a todos os meus amigos e colegas, pela sua amizade e apoio.





# Índice

Abstract .....	iv
Keywords.....	v
Resumo.....	vi
Palavras-Chave .....	vii
Agradecimentos .....	viii
Índice de figuras .....	xiv
Índice de Gráficos.....	xvii
Índice de Tabelas.....	xviii
Acrónimos .....	xix
1. Introdução .....	20
1.1 Contexto .....	20
1.2. Motivação.....	22
1.3. Objetivos .....	23
1.4. Estrutura da Dissertação .....	24
2. Revisão da Literatura.....	25
2.1 Visualização de Informação.....	25
2.2. Arquitetura – SAP.....	26
2.2.1. SAP ERP .....	26
2.3.1. Como funciona o SAP? .....	27
2.3.1.1. SAP KERNEL .....	27
2.3.1.2. Exemplo de funcionamento do SAP .....	28
2.4. Landscape - SAP.....	29
2.5. Códigos de Transação.....	30
2.5.1. SE11 .....	30
2.5.2. SE16 .....	32
2.5.3. SE37 .....	33

2.5.4. SE80 .....	34
3. Sistemas de Gestão Financeira e de Inteligência de Negócios .....	35
3.1. Lista de Sistemas .....	35
3.1.1 Lista de ferramentas.....	35
3.2. Dundas BI .....	35
3.2.1 Características Dundas BI .....	36
3.2.2. Painel de controlo personalizado.....	36
3.2.3. Ferramentas de Comunicação e Colaboração .....	37
3.2.4. Visualização de Dados.....	38
3.3. Domo .....	39
3.3.1. Características Domo .....	39
3.3.2. Filtragem de dados.....	39
3.3.3. Software ETL.....	40
3.3.4. Fusão de dados.....	40
3.4. Pentaho .....	41
3.4.1. Características Pentaho.....	41
3.4.2. Análises de Negócio .....	42
3.4.3. Integração de Dados .....	43
3.5. Power BI .....	44
3.5.1. Exemplo de Recursos Humanos para o Power BI: .....	44
3.5.2. Análise de Vendas .....	45
3.6. SAP.....	46
3.6.1. Características SAP .....	46
3.6.2. Visualização de dados e Análise.....	46
3.6.3. Painéis de Dados .....	47
3.6.4. Componentes Principais SAP ERP .....	48
3.7. Comparação entre Sistemas .....	51
3.7.1. Descrição dos critérios de Avaliação.....	52

3.7.2. Tabela de Comparação de Tecnologias.....	53
4. Desenvolvimento .....	55
4.1. Identificação dos Requisitos.....	55
4.1.1. Requisitos Funcionais.....	55
4.1.2. Requisitos Não-Funcionais .....	59
4.2. Arquitetura .....	60
4.2.1 Diagrama de Sequência - Extrator de Dados.....	62
4.3. Extração de dados SAP .....	63
4.3.1. Arquitetura SAP Connector 3.0 .....	63
4.3.2. Desenvolvimento de RFCs .....	67
4.3.3. RFCs utilizados para extração de dados .....	68
4.4. Dificuldades Encontradas na Extração de Dados .....	70
4.4.1. Limitações SQL SAP .....	70
4.5. Pesquisa personalizada .....	71
4.6. Tratamento de Dados Ambiente .Net .....	71
4.6.1 Tratamento do formato TAB512 .....	71
4.6.2. Tratamento do Formato de Exportação do tipo <i>String</i> .....	73
4.7. Paginação .....	75
4.8. Filtragem .....	76
4.9. Exportação de Dados.....	76
4.9.1. iText Sharp (PDF).....	77
4.9.2. XML.....	79
4.9.3. CSV.....	79
4.9.4. HTML .....	80
4.10. Desenvolvimento - Interface.....	83
4.10.1. Login Form.....	83
4.10.2. MainForm .....	84
5. Resultados .....	85

5.1 Testes de Desempenho .....	86
5.1.1. Medição - Tempo de Login.....	87
5.1.2. Medição - Tempo de Obtenção de Campos.....	87
5.1.3. Medição - Tempo de Obtenção de Dados (Sem Filtragem) .....	88
5.1.4. Medição - Tempo de Obtenção de Dados (Com Filtragem).....	89
5.1.5. Medição - Exportação de Dados PDF .....	91
5.1.6. Medição - Exportação de Dados CSV .....	91
5.1.7. Medição - Exportação de Dados XML .....	92
5.1.8. Medição - Exportação de Dados HTML .....	92
6. Conclusão .....	94
6.1 Trabalho Futuro.....	96
7. Referências.....	97
8. Anexos .....	101
8.1. Utilização do Extrator de Dados.....	101
8.1.1. Login .....	102
8.1.2. Formulário Principal .....	103
8.2 Filtragem de valores .....	109
8.3. Exportação de Dados PDF .....	110
8.4. Exportação de Dados CSV .....	111
8.5. Exportação de Dados HTML.....	112
8.6 Exportação de Dados XML (Tabela) .....	113

## Índice de figuras

Figura 1 - Mapa Cartográfico.....	25
Figura 2 - Gráfico de Árvore .....	25
Figura 3 - Arquitetura SAP ERP [1] .....	26
Figura 4 - Instância Central [3] .....	28
Figura 5 – Landscape .....	29
Figura 6 - Janela de Input SE11 .....	30
Figura 7 - Estrutura Tabela SPFLI.....	31
Figura 8 - Janela de Input SE16 .....	32
Figura 9 - Conteúdo tabela SPFLI .....	32
Figura 10 - Módulos de Função.....	33
Figura 11 - Pesquisa de módulos de função.....	33
Figura 12 - SE80 Workbench .....	34
<i>Figura 13 - Características Dundas BI [12] .....</i>	<i>36</i>
Figura 14 - Painel de Controlo Personalizado Dundas BI [13].....	36
Figura 15 - Ferramentas de Comunicação e Colaboração - Dundas BI [13].....	37
<i>Figura 16 - Visualização de Dados – Dundas BI [13] .....</i>	<i>38</i>
Figura 17 - Características – Domo [15] .....	39
Figura 18 - Filtragem de Valores – Domo [16] .....	39
Figura 19- Software ETL – Domo [16].....	40
Figura 20 – Fusão de Dados – Domo [17] .....	40
Figura 21 - Características – Pentaho [18] .....	41
Figura 22 - Painéis de Controlo gráficos e responsivos – Pentaho [20] .....	42
Figura 23 - Integração de dados - Pentaho .....	43
Figura 24 - Painel de Controlo RH – PowerBi [21].....	44
Figura 25 - Exemplo de análise de vendas – Power BI [22].....	45
Figura 26 - Características SAP [23].....	46
Figura 27 - Visualização de dados SAP [23].....	46
Figura 28 - Painel de dados SAP [23].....	47
Figura 29 - SAP ERP Componentes [23].....	48
Figura 30 – Arquitetura do Extrator de dados .....	60
Figura 31 - Arquitetura.....	61
Figura 32 - Diagrama de Sequência.....	62

Figura 33 - Arquitetura SAP Connector 3.0 .....	63
Figura 34 - Declaração objeto RfcConfigParameters .....	64
Figura 35 - Preenchimento de parâmetros .....	64
Figura 36 - Verificação da conexão .....	64
Figura 37 - RFC Remote-Enabled Module .....	65
Figura 38 - Aba de importação .....	65
Figura 39 - Aba Tables .....	66
Figura 40 - Utilização de RFC e Variáveis de importação .....	66
Figura 41 - Exportação de dados .....	67
Figura 42 - Create DATA .....	67
Figura 43 - Pacote de Armazenamento .....	68
Figura 44 - Formato TAB512 .....	68
Figura 45 - Formato tipo String - Z_AS_GET_TABLE_ENTRIES .....	69
Figura 46 - Formato tipo String - Z_AS_RFC_GET_FIELD_DATA .....	69
Figura 47 - Query para obter os nomes dos campos .....	70
Figura 48 - Pesquisa personalizada com variável de controlo .....	71
Figura 49 - Exemplo Paginação .....	75
Figura 50 - Query com uso de MANDT .....	76
Figura 51 - Criação Tabela iTextSharp .....	77
Figura 52 - Adição de cabeçalhos .....	78
Figura 53 - Adição de linhas .....	78
Figura 54 - Gravação PDF .....	78
Figura 55 - Escrita para ficheiro XML .....	79
Figura 56 - DataTable e StringBuilder .....	79
Figura 57 - Ciclo para percorrer colunas .....	79
Figura 58 - Ciclo para percorrer Linhas da tabela .....	80
Figura 59 - Gravar ficheiro CSV .....	80
Figura 60 – Inicializações .....	80
Figura 61 - HTML preenchimento de colunas .....	81
Figura 62 – Preenchimento de colunas HTML .....	81
Figura 63 - Fecho de etiquetas e gravação do ficheiro HTML .....	82
Figura 64 - Form Login .....	83
Figura 65 - MainForm .....	84
Figura 66 - Formulário de Login .....	102
Figura 67 - Conexão com Sucesso .....	102

Figura 68 - Inserção de tabela e obtenção de campos .....	103
Figura 69 - Seleção de Campos .....	104
Figura 70 - Resultados da pesquisa .....	105
Figura 71 - Modificação da pesquisa inicial.....	106
Figura 72 - Exportação de Dados.....	107
Figura 73 - Exportação com sucesso .....	108
Figura 74 – Filtragem de valores .....	109
Figura 75 - Extração PDF .....	110
Figura 76 - Extração CSV.....	111
Figura 77 - Extração HTML .....	112
Figura 78 - Extração XML.....	113



## Índice de Gráficos

Gráfico 1 - Dundas Bi Avaliação .....	53
Gráfico 2 - Domo Avaliação.....	53
Gráfico 3 - Pentaho Avaliação .....	54
Gráfico 4 - SAP Avaliação .....	54
Gráfico 5 - Extrator de Dados Avaliação.....	54
Gráfico 6 - Power Bi Avaliação .....	54
Gráfico 7 - Extrator de Dados Avaliação.....	54
Gráfico 8 - Power Bi Avaliação .....	54
Gráfico 9 - Tempo de pesquisa sem filtragem .....	88
Gráfico 10 -Tempo de pesquisa com filtragem .....	89
Gráfico 11 - Comparação das Pesquisas.....	90

## Índice de Tabelas

Tabela 1 - Parâmetros de Comparação.....	51
Tabela 2 - Descrição critérios de Avaliação.....	52
Tabela 3 - Comparação de Tecnologias.....	53
Tabela 4 - Tempos de Login.....	87
Tabela 5 - Obtenção dos campos de uma Tabela SAP .....	87
Tabela 6 - Obtenção do conteúdo de uma tabela SAP (Sem Filtragem) .....	88
Tabela 7 - Obtenção conteúdo de uma tabela SAP (Com Filtragem).....	89
Tabela 8 - Diferença entre tempos de pesquisa .....	90
Tabela 9 - Tempos Exportação PDF.....	91
Tabela 10 - Tempos Exportação CSV .....	91
Tabela 11 - Tempos Exportação XML.....	92
Tabela 12 - Tempos Exportação HTML.....	92

## Acrónimos

ABAP - Advanced Business Application Programming

CSV - Comma Separated Values

ERP - Enterprise Resource Planning

HTML - Hyper Text Markup Language

ND - Nome Developer

NFS - Network File System

PDF - Portable Document Format

RF - Requisito Funcional

RFC - Remote Function Call

RNF - Requisito não Funcional

SAP - Systems, Applications & Products in Data Processing

SQL - Structured Query Language

XML - Extensible Markup Language

# 1. Introdução

## 1.1 Contexto

Nos dias de hoje, é comum lidar com um enorme e variadíssimo leque de informação o que faz com que o *software* utilizado para gerir e veicular a visualização desta informação, seja cada vez mais relevante para pequenas, médias e grandes empresas. Devido a esta crescente e progressiva necessidade de proceder ao tratamento de grandes volumes de dados, surgiram diversos *softwares* de gestão financeira e de inteligência de negócios, tais como a SAGE - que é muito utilizada em Portugal, o Pentaho - o qual foi abordado nas aulas de Sistemas de Apoio à Decisão, e o sistema SAP, o qual é abordado nesta tese, e sobre o qual é desenvolvido o extrator de dados, com o auxílio da plataforma .NET da Microsoft.

O principal desafio destes sistemas é o tratamento e apresentação de grandes volumes de informação de uma forma segura, rápida e eficiente. Além deste propósito visam, também, dispor os dados de modo a proporcionar uma compreensão fácil e intuitiva, para que o utilizador seja capaz de tirar proveito máximo da informação que lhe é disponibilizada, podendo assim ser capaz de gerir o seu negócio de uma forma mais eficiente, transparente e robusta.

Esta tese consiste na explanação do desenvolvimento de um extrator de dados que será utilizado para efetuar a extração de informação de uma forma rápida e eficiente de empresas que utilizam o sistema de gestão empresarial SAP (Sistemas, Aplicativos e Produtos para Processamento de Dados), o qual é tipicamente utilizado por empresas de grandes dimensões como, por exemplo, a “The Coca-Cola Company”.

Este projeto foi proposto pela Odkas *Factory*, que está localizada no Madeira Tecnopolo, e tem como objetivo a formação de desenvolvedores de software juniores SAP.

Esta *factory* efetua o desenvolvimento de soluções de software para clientes que utilizam o sistema de gestão empresarial e inteligência de negócios SAP.

Pretende-se, com este projeto, tornar possível a visualização de enormes tabelas de informação, e permitir a realização de uma pesquisa personalizada, e ainda proceder à exportação dos dados para quatro tipos de ficheiros, sendo estes os mais utilizados atualmente, nomeadamente: PDF, XML, HTML e CSV.

Estes dados poderão ser acedidos para fins de auditoria, sem que exista a necessidade de gerar um relatório através do ambiente SAP.

Por fim, esta é uma oportunidade para adquirir conhecimentos na linguagem ABAP, utilizada pela SAP para desenvolver as suas soluções de *software*, mediante as necessidades dos clientes.

## 1.2. Motivação

A necessidade de desenvolver um extrator de dados surge do facto da empresa Odkas não possuir uma ferramenta que possibilite efetuar a exportação de grandes volumes de dados do servidor SAP. Esta ferramenta existe e é disponibilizada pela SAP por um preço pouco acessível. Devido ao custo associado a esta ferramenta, urgiu então desenvolver um extrator de dados como uma alternativa mais económica e direcionada a extrações, nomeadamente extração de dados de uma tabela de cada vez.

A extração e visualização de dados poderia ser efetuada através do uso do Excel; no entanto, para utilizar esta ferramenta, é requerida já alguma experiência por parte do utilizador, visto que tem de ser toda configurada de raiz. Por este motivo, o extrator torna-se imperativo, pois através deste será possível visualizar e extrair dados através do uso das suas funcionalidades, e, uma vez que se encontram já implementadas e prontas a utilizar, o utilizador não tem de passar por um período de aprendizagem, reduzindo significativamente o tempo necessário para usufruir dos dados.

Este extrator veicula a facilitação e acessibilidade no acesso aos dados que estão contidos em tabelas de dados do servidor SAP.

É ainda possível efetuar a exportação desses mesmos dados para vários tipos de ficheiro, sem a constante necessidade de efetuar o Login no SAPGUI, e navegar até à transação que possibilita a visualização desta informação.

É também uma alternativa viável para utilizadores que não possuem qualquer tipo de conhecimento em relação à utilização do SAP, pois permite que estes acedam às funcionalidades de visualização e de exportação de dados de uma forma segura e fácil.

### 1.3. Objetivos

- A. Criação de uma ferramenta de extração de dados.
- B. Conceção de RFCs no ambiente SAP, para efetuar a extração de dados.
- C. Visualização dos dados extraídos, através da ferramenta desenvolvida.
- D. Execução da paginação dos valores obtidos.
- E. Implementação de filtros, de forma a auxiliar a extração de dados.
- F. Criação de métodos de exportação dos dados apresentados para determinados tipos de ficheiro, tais como PDF, XML, HTML e CSV.
- G. Desenvolvimento de uma interface de utilização simplificada, de modo a proporcionar aos utilizadores que não possuam qualquer tipo de conhecimento sobre SAP um pleno e assertivo usufruto das suas funcionalidades.
- H. Aquisição de conhecimentos sobre o SAP e a linguagem ABAP.

## 1.4. Estrutura da Dissertação

A estrutura desta dissertação constitui-se pela divisão em sete capítulos. No primeiro capítulo encontra-se a introdução, na qual são apresentados o contexto, a motivação e os objetivos.

O segundo capítulo, por seu turno, consiste numa revisão da literatura que introduz alguns conhecimentos acerca da origem da visualização de informação, e ainda na apresentação da arquitetura do sistema SAP de uma forma resumida, o seu funcionamento e os seus *landscapes* (DEV, QAS, PRD). Assevera, de igual modo, alguns códigos de transação, muito utilizados no desenvolvimento de uma solução SAP.

O terceiro capítulo consiste na exposição de tecnologias semelhantes às do extrator a desenvolver, pese embora com características diferentes, e que serviram de inspiração na implementação de algumas das funcionalidades do extrator de dados.

A descrição dos requisitos funcionais e não funcionais para o desenvolvimento do extrator - a sua arquitetura – é tratada no quarto capítulo. Ao longo deste são, ainda, explanadas as partes mais importantes da sua implementação, nomeadamente a extração de dados em ambos os ambientes (SAP e .Net), o tratamento de dados, o desenvolvimento dos RFC's necessários, a pesquisa personalizada, a filtragem de valores, a paginação dos dados e a implementação dos métodos para a exportação destes para vários tipos de ficheiros.

No quinto capítulo são expostos os resultados da visualização e da exportação dos dados para ficheiros PDF, CSV, HTML e XML.

No decurso dos capítulos anteriores, o sexto capítulo patenteia as conclusões tiradas após o desenvolvimento deste trabalho e abre vetores de trabalho futuro.

Por fim, o sétimo capítulo documenta a exposição da presente dissertação, aditando todos os anexos referentes aos métodos de exportação implementados.



## 2. Revisão da Literatura

### 2.1 Visualização de Informação

Com o início das interfaces gráficas por volta dos anos 1950, altura em que os primeiros gráficos eram gerados pelos computadores, dá-se uma necessidade crescente de desenvolver meios para facilitar a visualização de informação. Em 1987 urgiu o desenvolvimento de novas técnicas de visualização de dados com base nos computadores. Com o aumento rápido do poder de processamento, enormes e complexos modelos numéricos foram desenvolvidos, resultando na geração de conjuntos de dados numéricos colossais [1]. A visualização de dados é atualmente muito ativa e uma área vital de pesquisa, ensino e desenvolvimento.

O sucesso da visualização de dados deve-se:

- À utilização de imagens geradas pelos computadores, de forma a adquirir conhecimento através dos dados e subsequentes padrões e relações.
- À aplicação do sistema sensorial humano na direção e interpretação de processos complexos e simulações envolvendo conjuntos de dados de diversas disciplinas científicas e grandes coleções de dados abstratos de muitas fontes [1].

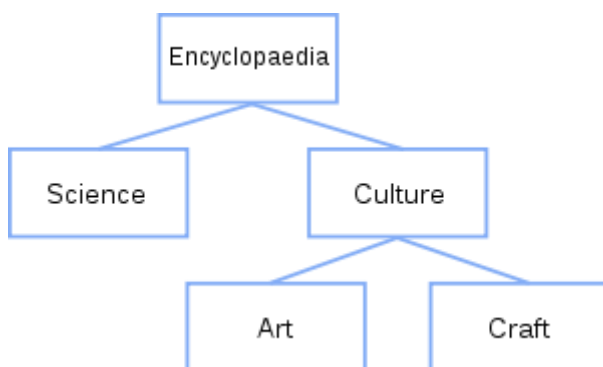


Figura 2 - Gráfico de Árvore

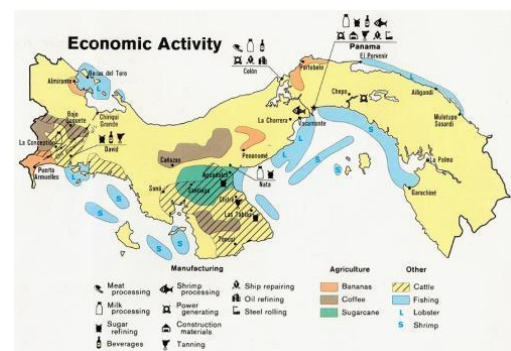


Figura 1 - Mapa Cartográfico

O acesso à observação de dados permite moldar a forma como visualizamos, organizamos e classificamos o mundo. Algumas das representações mais populares e eficazes são as árvores, com frequência utilizadas para representar hierarquias e mapas cartográficos [1].

## 2.2. Arquitetura – SAP

### 2.2.1. SAP ERP

O SAP ERP é tipicamente estruturado em três níveis numa arquitetura cliente/servidor. A arquitetura de três níveis é a mais recomendada, na medida em que permite aceder com grande escalabilidade e flexibilidade, embora possa também ser utilizado com hierarquias de dois e de um nível.

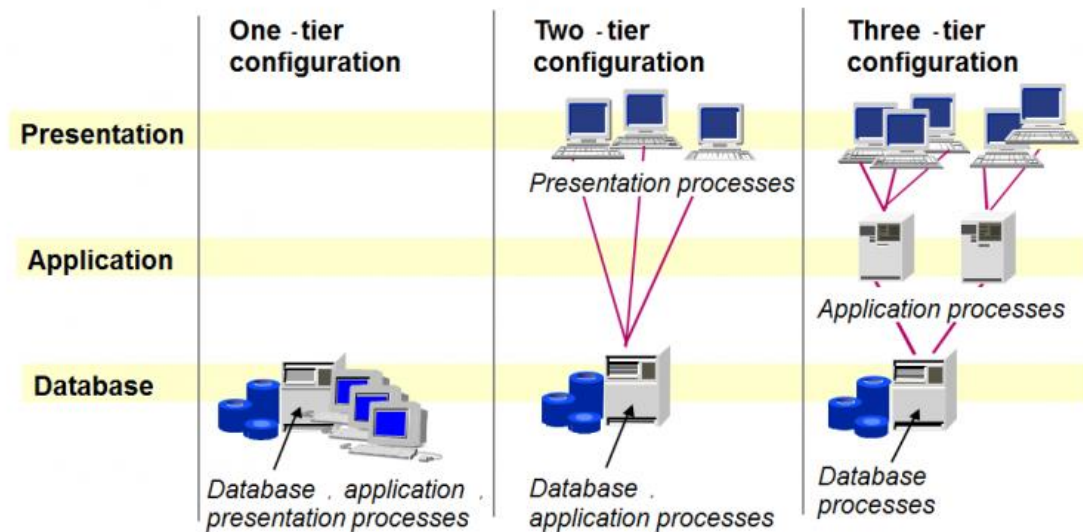


Figura 3 - Arquitetura SAP ERP [1]

Na arquitetura SAP de três níveis, o nível da apresentação providencia a interface ao utilizador, a camada da aplicação efetua o processamento da lógica do negócio, e o nível da base de dados armazena os dados do negócio.

#### 2.2.1.1. Nível de Apresentação

Este é basicamente o núcleo do sistema SAP ERP. Executa a lógica de negócio, é responsável pelo processamento de transações dos clientes, trabalhos de impressão, geração de relatórios, coordena os acessos à base de dados, e veicula a interação com outras aplicações. Torna, também, possível a distribuição da carga do sistema por várias máquinas [2].

### 2.2.1.3. Nível de Base de Dados

A base de dados é utilizada para guardar dois tipos de objetos: os dados gerados pelo negócio e os programas da aplicação SAP. Os dados gerados pelo negócio, representam objetos de dados criados pelos utilizadores como parte integrante de vários processos de negócio como, por exemplo, os registos de vendas ou dos próprios clientes. Os programas SAP são rotinas escritas em ABAP que são carregadas para os servidores SAP da empresa a partir da base de dados, durante o tempo de execução. É possível a utilização de diversas bases de dados [2].

### 2.3.1. Como funciona o SAP?

O servidor da aplicação (KERNEL) obtém o *input* e mostra o *output* no nível da apresentação (SAP GUI), e armazena a informação no nível da base de dados [2].

#### 2.3.1.1. SAP KERNEL

Trata-se de uma coleção de programas executáveis e de ferramentas auxiliares para o processamento da lógica de negócios. Todos os processos iniciados ou parados são chamados instâncias SAP. Cada instância SAP contém um *dispatcher* (expedidor) e vários processos de trabalho. O *dispatcher*, por sua vez, distribui as tarefas para um dos processos de trabalho [2].

Lista de tipos de processos:

D – Processos de diálogo, através dos quais são geridos os pedidos de transações *online* efetuados pelos utilizadores.

B – Processos de trabalho em lote, responsáveis pelo processamento de trabalhos em segundo plano, agendados pelo sistema SAP.

V – Processos de trabalho de atualizações: permitem efetuar atualizações à base de dados.

S – Processos de trabalho de *spool* (Bobina), por meio dos quais se procede à ativação da impressão no sistema SAP.

G – Processos de trabalho de entrada, veículos de ativação da comunicação entre aplicações. É apenas necessária uma porta de entrada por sistema SAP.

O sistema SAP ERP pode conter várias instâncias de aplicações SAP, mas, entre estas, urge a existência de uma instância especial chamada instância principal (CI), a qual possui dois componentes adicionais:

- Processo de Mensagem (M)
- Servidor Enqueue (E)

O processo de mensagem da instância principal é utilizado para estabelecer a comunicação entre várias instâncias incluídas no sistema SAP. Por exemplo, quando um utilizador efetua o seu LOGIN, o processo de mensagem atribui-lhe automaticamente uma das instâncias de aplicação disponíveis. Em seguida, todos os pedidos relacionados com este utilizador são redirecionados para a instância selecionada.

O servidor *enqueue* da instância principal é usado para gerir bloqueios nas tabelas da base de dados. Estes bloqueios garantem que as atualizações na base de dados são efetuadas numa sequência correta, garantindo, deste modo, a consistência dos dados do negócio [2].

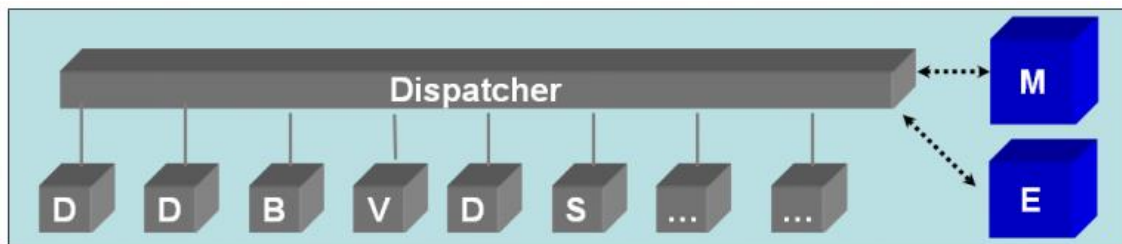


Figura 4 - Instância Central [3]

### 2.3.1.2. Exemplo de funcionamento do SAP

Embora sinoticamente, eis como opera o Sistema SAP:

1. Chega um pedido do nível de apresentação.
2. O pedido é analisado pelo *dispatcher*.
3. O *dispatcher* transfere o pedido para o processo de mensagem (M).
4. O processo de mensagem decide se deve ser efetuado nesta instância ou remetido para uma outra instância.
5. No caso de permanecer na mesma instância, é colocado num dos processos de trabalho do tipo apropriado.
6. O pedido é processado e, caso necessário, o sistema SAP efetua uma atualização da base de dados através do *enqueue server* (E).
7. Por fim, o feedback do desfecho do pedido é entregue à origem através da formulação de um pedido, mas na ordem inversa [2].

## 2.4. Landscape - SAP

Por *landscape* refere-se ao *layout* dos servidores, que se encontram representados da seguinte forma:

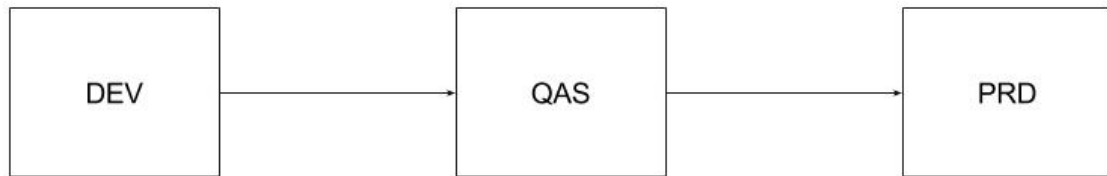


Figura 5 – Landscape

- Desenvolvimento (DEV) – local onde os consultores ABAP efetuam as customizações consoante as necessidades do cliente.
- Qualidade (QAS) – ponto onde os membros principais da equipa testam as personalizações efetuadas.
- Produção (PRD) – lugar onde se encontram alojados os dados da empresa em tempo real.

O desenvolvimento deve ser sempre efetuado da seguinte forma:

- DEV ---> QAS ---> PRD e nunca de forma inversa.

Este *layout* aplica-se a qualquer empresa, no qual o programador desenvolve o seu programa no servidor de desenvolvimento (DEV), posteriormente transportado para o servidor de testes de qualidade (QAS), no qual é exaustivamente testado e finalmente transportado para o servidor de produção (PRD), para futura utilização por parte do cliente [3].

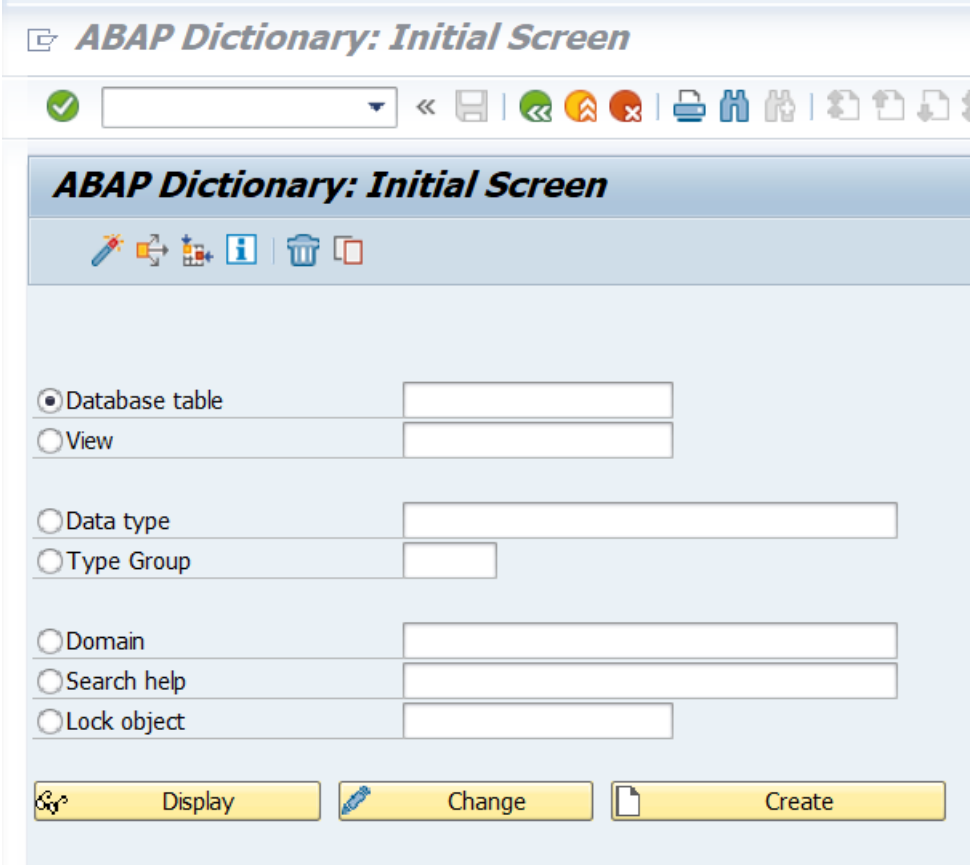
## 2.5. Códigos de Transação

Com o objetivo de auxiliar todo o processo de desenvolvimento do extrator de dados, foram utilizados códigos de transação ou *T-codes*, cujas transações tornam possível a visualização de informação sobre a estrutura das tabelas, os seus tipos de dados e efetuar a criação de RFCs.

T-codes utilizados:

- SE11
- SE16
- SE37
- SE80

### 2.5.1. SE11



The screenshot shows the 'ABAP Dictionary: Initial Screen' (SE11) in SAP. The window title is 'ABAP Dictionary: Initial Screen'. The interface includes a toolbar with icons for search, navigation, and printing. Below the toolbar, there are radio buttons for selecting the object type: Database table (selected), View, Data type, Type Group, Domain, Search help, and Lock object. Each radio button is followed by an input field. At the bottom, there are three buttons: 'Display' (with a magnifying glass icon), 'Change' (with a pencil icon), and 'Create' (with a document icon).

Figura 6 - Janela de Input SE11

Trata-se de um dicionário ABAP, utilizado para criar, alterar e exibir tabelas e as suas estruturas.

No ecrã inicial (Figura 6), é possível inserir o nome de qualquer tabela ou estrutura existente no sistema SAP para visualizar, editar ou criar uma tabela ou estrutura. Esta ferramenta é tipicamente utilizada por programadores ABAP [4].

A título de exemplo expõe-se infra a estrutura da tabela “SPFLI” (Figura 7), tipicamente utilizada para gerir conteúdos relacionados com aeroportos.

**Dictionary: Display Table**

Transparent Table: **SPFLI** Active

Short Description: **Flight schedule**

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

Field	Key	Inti...	Data element	Data Type	Length	Decim...	Short Description
<u>MANDT</u>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<u>S_MANDT</u>	CLNT	3		0 Client
<u>CARRID</u>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<u>S_CARR_ID</u>	CHAR	3		0 Airline Code
<u>CONNID</u>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<u>S_CONN_ID</u>	NUMC	4		0 Flight Connection Number
<u>COUNTRYFR</u>	<input type="checkbox"/>	<input type="checkbox"/>	<u>LAND1</u>	CHAR	3		0 Country Key
<u>CITYFROM</u>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<u>S_FROM_CIT</u>	CHAR	20		0 Departure city
<u>AIRPFROM</u>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<u>S_FROMAIRP</u>	CHAR	3		0 Departure airport
<u>COUNTRYTO</u>	<input type="checkbox"/>	<input type="checkbox"/>	<u>LAND1</u>	CHAR	3		0 Country Key
<u>CITYTO</u>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<u>S_TO_CITY</u>	CHAR	20		0 Arrival city
<u>AIRPTO</u>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<u>S_TOAIRP</u>	CHAR	3		0 Destination airport
<u>FLTIME</u>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<u>S_FLTIME</u>	INT4	10		0 Flight time
<u>DEPTIME</u>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<u>S_DEP_TIME</u>	TIMS	6		0 Departure time
<u>ARRTIME</u>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<u>S_ARR_TIME</u>	TIMS	6		0 Arrival time
<u>DISTANCE</u>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<u>S_DISTANCE</u>	QUAN	9	4	Distance
<u>DISTID</u>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<u>S_DISTID</u>	UNIT	3		0 Mass unit of distance (kms, miles)
<u>FLTYPE</u>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<u>S_FLTYPE</u>	CHAR	1		0 Flight type
<u>PERIOD</u>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<u>S_PERIOD</u>	INT1	3		0 Arrival n day(s) later

Figura 7 - Estrutura Tabela SPFLI

### 2.5.2. SE16

É um navegador de dados e é utilizado para visualizar o conteúdo de uma tabela selecionada, ao contrário da transação SE11 não é possível efetuar, alterações ou anexar novos campos à estrutura já existente [5].

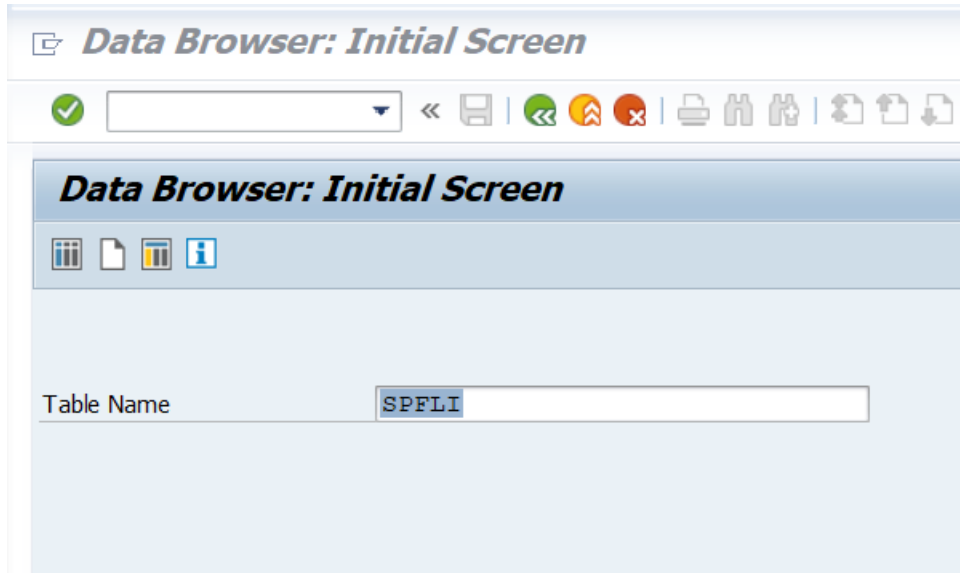


Figura 8 - Janela de Input SE16

A fim de exemplificar o *output* da transação SE16, utilizamos a tabela “SPFLI”. O seu conteúdo pode ser visualizado na Figura 9.

	Client	Airline	Flight Number	Country	Depart.city	Dep. airport	Country
<input type="checkbox"/>	001	AA	0017		NEW YORK	JFK	
<input type="checkbox"/>	001	AF	0820	DE	FRANKFURT/MAIN	FRA	FR
<input type="checkbox"/>	001	DL	1699		NEW YORK	JFK	
<input type="checkbox"/>	001	DL	1984		SAN FRANCISCO	SFO	
<input type="checkbox"/>	001	LH	0400	DE	FRANKFURT	FRA	
<input type="checkbox"/>	001	LH	0402	DE	FRANKFURT	FRA	
<input type="checkbox"/>	001	LH	9981	DE	FRANKFURT	FRA	DE

Figura 9 - Conteúdo tabela SPFLI



### 2.5.3. SE37

Os módulos de função são procedimentos definidos unicamente em programas ABAP específicos, mas podem ser evocados em todos os outros programas ABAP. Permite a criação de grupos de funções (Figura 10), nos quais podem estar contidos vários programas. Os módulos de função permitem o encapsulamento e reutilização de funções globais dentro do sistema SAP [6] [7].

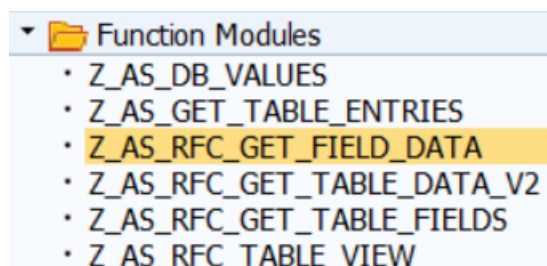


Figura 10 - Módulos de Função

Através desta transação é possível aceder a módulos de função por meio da pesquisa pelo nome da mesma (Figura 11).

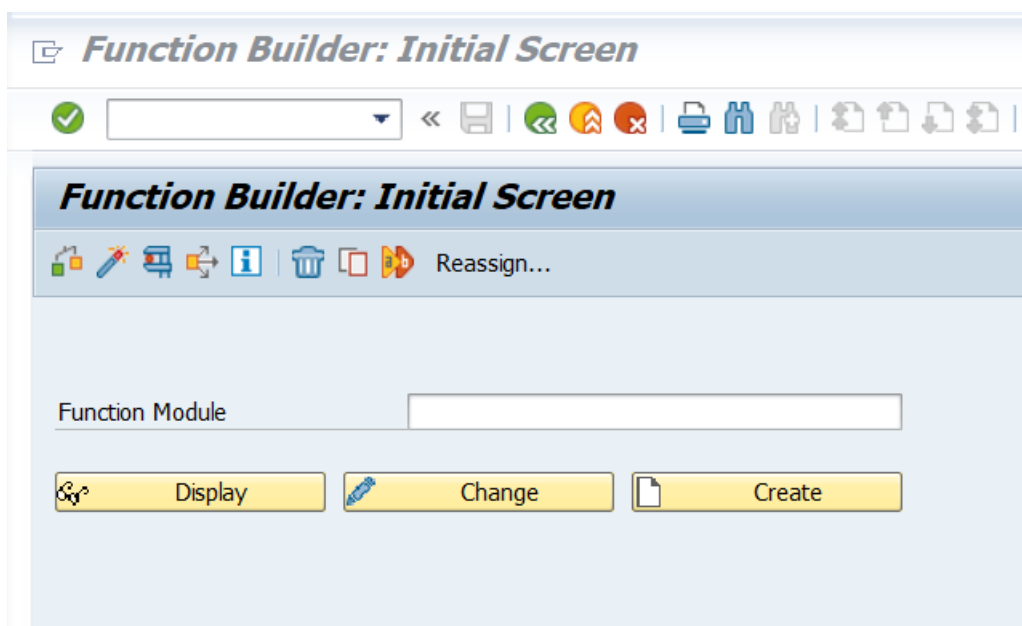


Figura 11 - Pesquisa de módulos de função

## 2.5.4. SE80

O SE80 trata-se de um navegador de objetos que permite:

- Modificar diretamente um programa
- Criar pacotes para conter os programas de um programador
- Conceber funções de módulos
- Facilitar – por meio das suas diversas funcionalidades - o desenvolvimento de um programa em ABAP tais como:
  - Screen Painter – permite o desenvolvimento de uma interface.
  - Menu Painter – utilizado no desenho de interfaces para o utilizador.
  - Function Builder – usado para procurar por módulos de funções existentes.
  - Class Builder – propicia a criação, modificação, definição e teste de classes e interfaces ABAP globais.

Na Figura 12 é possível visualizar a página inicial da transação SE80 [8] [9] [10].

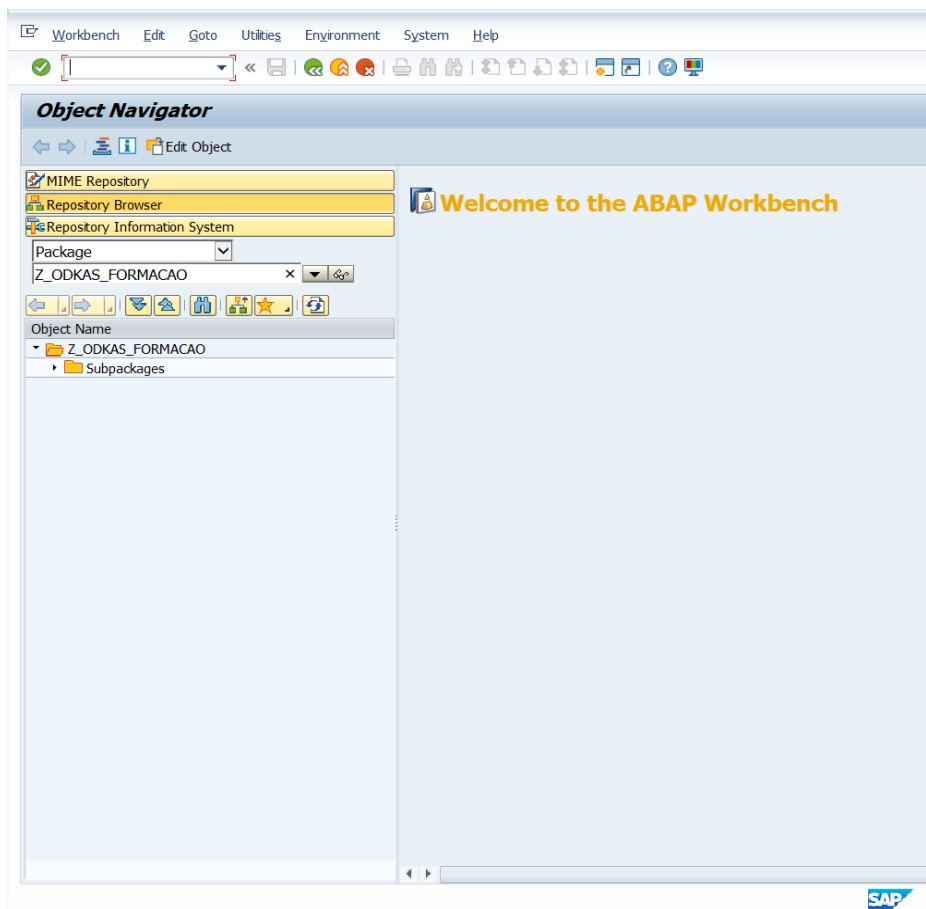


Figura 12 - SE80 Workbench

## 3. Sistemas de Gestão Financeira e de Inteligência de Negócios

Ao longo deste capítulo discorreremos por alguns dos sistemas utilizados como inspiração para o desenvolvimento do extrator de dados; são, igualmente, identificadas algumas características dos mesmos. Por fim, e em jeito de aferição, aproveitamos para proceder a uma comparação das funcionalidades oferecidas pelo extrator com as funcionalidades oferecidas pelos outros sistemas.

### 3.1. Lista de Sistemas

Os *softwares* de inteligências de negócios são um tipo de aplicações de software concebidos para recuperar, analisar, transformar e reportar dados.

#### 3.1.1 Lista de ferramentas

- Dundas BI
- Domo
- Pentaho
- Power BI
- SAP [11]

### 3.2. Dundas BI

O Dundas BI oferece ao utilizador um controlo total sobre os seus dados. Visualiza, explora, prepara e transforma os dados em painéis de controlo, relatórios de múltiplas páginas e análise de dados através de gráficos. Detém muitas funcionalidades integradas, fluxos de dados ágeis adaptados às competências do utilizador e a API é totalmente aberta [12] [13].

### 3.2.1 Características Dundas BI

- ✓ Dashboards
- ✓ Drag-and-Drop Tools
- ✓ Visualizations
- ✓ Personalized Themes
- ✓ Templates
- ✓ Data Filters
- ✓ Navigation
- ✓ Customizable Buttons
- ✓ Hover Interactions
- ✓ Sharing
- ✓ Notifications
- ✓ Notes
- ✓ Email Alerts
- ✓ Comments
- ✓ Reports
- ✓ Slideshows
- ✓ Data Interactions
- ✓ Chart Animated Transitions
- ✓ Re-visualizing
- ✓ Data-brushing
- ✓ Scorecards
- ✓ Tables
- ✓ Charts
- ✓ Gauges
- ✓ Maps
- ✓ Treemap
- ✓ Data Labels
- ✓ Sparkline
- ✓ Indicators
- ✓ Diagrams
- ✓ Mobile Apps
- ✓ Analytics

Figura 13 - Características Dundas BI [12]

### 3.2.2. Painel de controlo personalizado

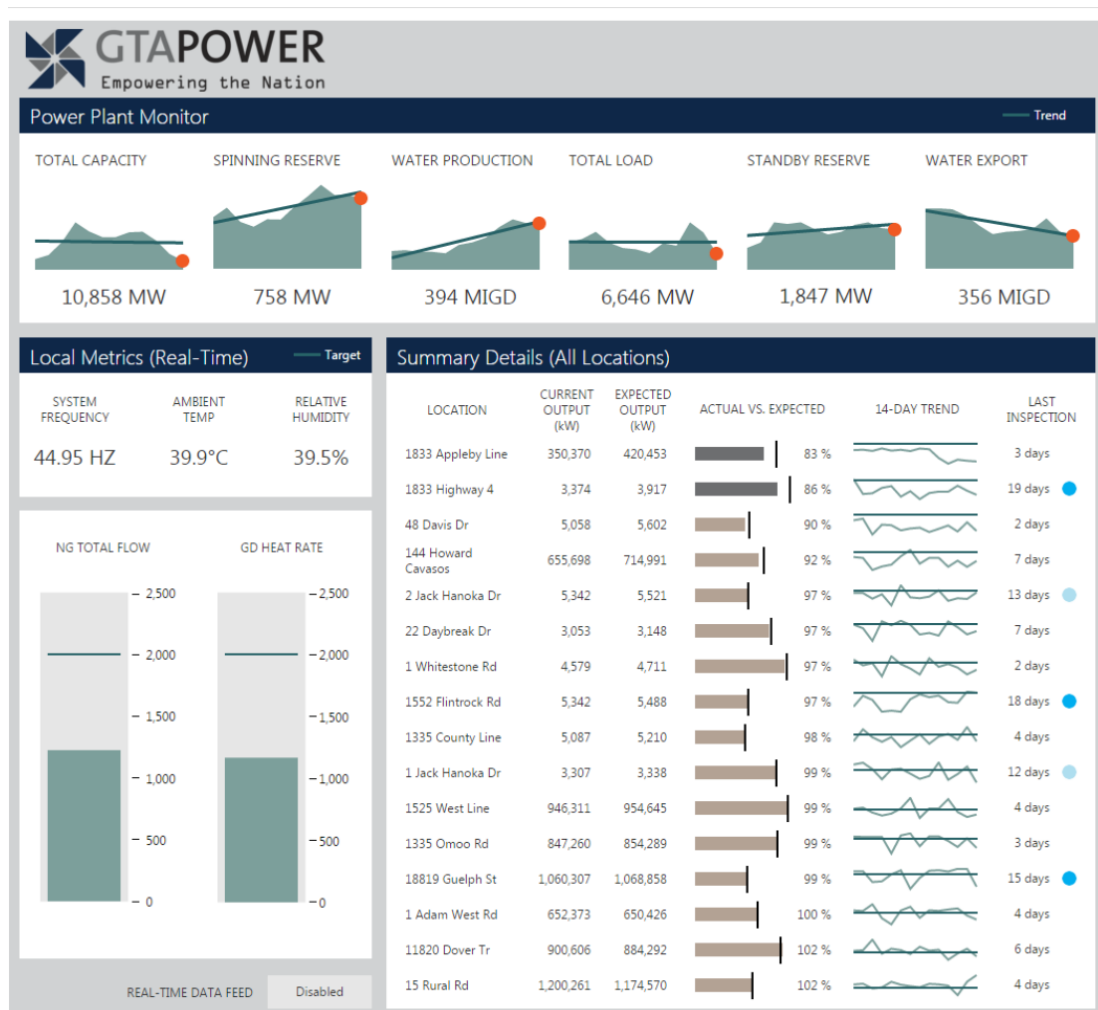


Figura 14 - Painel de Controlo Personalizado Dundas BI [13]

O Dundas BI oferece ao utilizador a opção de personalizar os painéis de controlo, permitindo-lhe monitorizar, efetuar a medição do desempenho e verificar métricas de negócio em tempo real. Procede em período “*drag-and-drop*” para facilitar a criação do painel customizado, proporcionando ao utilizador a hipótese de selecionar aquilo que pretende visualizar no seu painel. Os dados escolhidos podem ser analisados de imediato [13].

### 3.2.3. Ferramentas de Comunicação e Colaboração

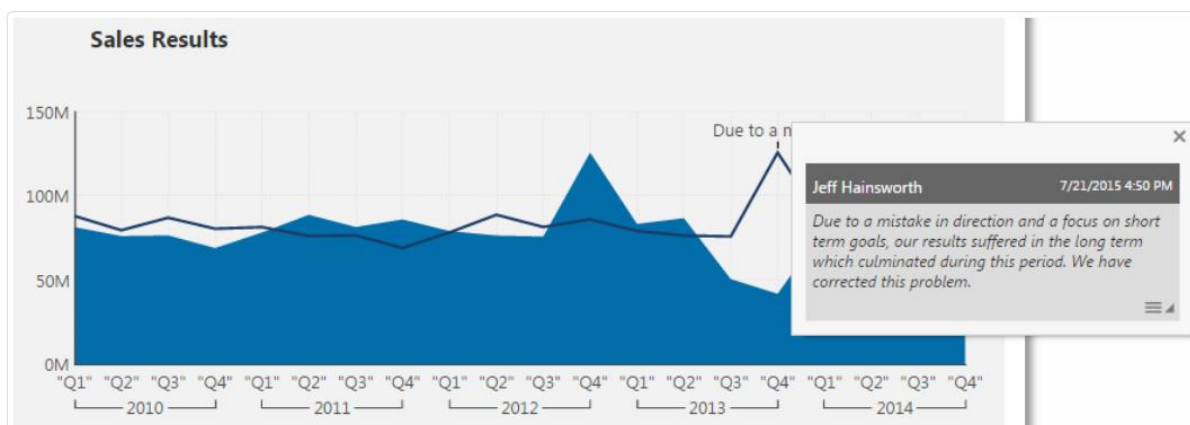


Figura 15 - Ferramentas de Comunicação e Colaboração - Dundas BI [13]

Por vezes, dada a impossibilidade de representação exímia por parte dos números, podem ser encontradas métricas de negócio distorcidas. O Dundas BI oferece a opção de contar a história com precisão, não só através dos dados, mas também através de *email*/notificações de texto, notas e até correções de dados.

O Dundas BI permite ao utilizador a criação uma nota que serve para perguntar e comentar, posteriormente utilizada como ferramenta de fundamentação para a distorção encontrada naquele local. Além de possibilitar a introdução de notas o Dundas BI autoriza a correção dos dados por parte do utilizador [13].

### 3.2.4. Visualização de Dados



Figura 16 - Visualização de Dados – Dundas BI [13]

Na Figura 16 encontra-se representado um exemplo da visualização de dados no Dundas Bi, desta feita referente ao Hotel WEXFORT, e permite-nos analisar o lucro relativo ao aluguer de quartos, consoante a localização escolhida, visto que este grupo hoteleiro detém hotéis espalhados por todo o mundo. Assim sendo, é possível ao utilizador escolher, por exemplo, o continente Asiático, e obter os dados revelantes, de acordo apenas com a localização requerida [13].

### 3.3. Domo

Este *software* de inteligência de negócios reúne única e exclusivamente os indivíduos, os dados e as ideias que os utilizadores das corporações precisam, de forma a fornecer uma visão detalhada do que está a acontecer dentro da sua organização. Permite a conexão de toda a informação crucial do negócio, colaboração com os colegas de trabalho e obtenção de dados úteis para ajudar na tomada de decisões [14] [15].

#### 3.3.1. Características Domo

- ✓ Create your dream dashboard
- ✓ Remain connected anywhere you go
- ✓ Flexible business data connection options and integration
- ✓ Derive true insights into business data
- ✓ Project management
- ✓ Buzz – communication & messaging platform
- ✓ Appstore – App bundles designed by third party developers
- ✓ Data visualization
- ✓ Social sharing tools
- ✓ Optimized for mobile platforms

Figura 17 - Características – Domo [15]

#### 3.3.2. Filtragem de dados



Figura 18 - Filtragem de Valores – Domo [16]

A filtragem de dados é utilizada apenas quando é necessário visualizar uma região específica, um produto, ou outro subconjunto definido [16].

### 3.3.3. Software ETL

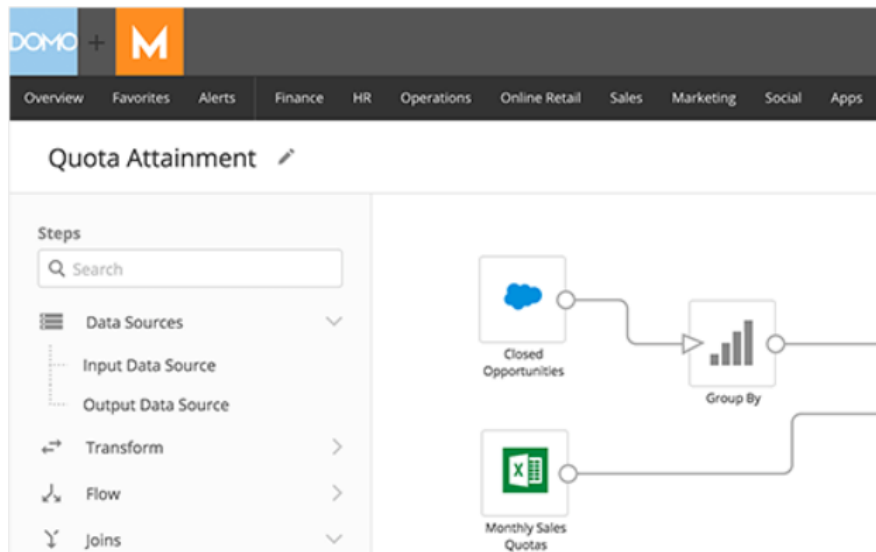


Figura 19- Software ETL – Domo [16]

Permite, desta forma, apurar, combinar e transformar dados, sem qualquer conhecimento de SQL [18].

### 3.3.4. Fusão de dados

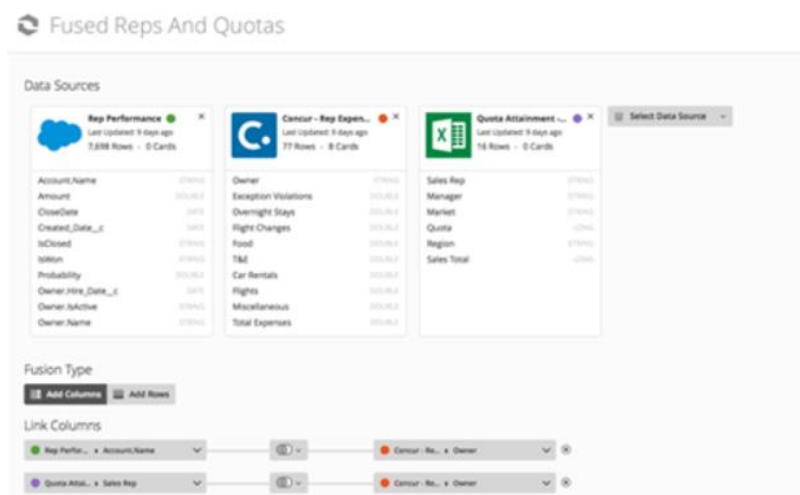


Figura 20 – Fusão de Dados – Domo [17]

A fusão de dados otimiza a integração dos dados, mediante a seleção e junção de fontes de informação para que possam ser visualizados. Permite, deste modo, efetuar a junção de MySQL, Redshift, CSV, Excel, e de outros tipos de dados de uma forma simplificada [17].



### 3.4. Pentaho

O Pentaho trata dos obstáculos que uma empresa pode eventualmente encontrar, o que levaria à obstrução no alcance do seu verdadeiro potencial, ou seja, privá-la-ia de tirar um maior proveito de todos os seus dados. Esta plataforma simplifica a preparação e a mistura de dados e inclui um conjunto de ferramentas para analisar, visualizar, explorar, informar e prever facilmente alterações que possam ocorrer no volume de negócio, bem como descobrir padrões escondidos. Aberto, incorporável e extensível, o Pentaho foi concebido de forma a garantir que qualquer membro da equipa possa traduzir facilmente os dados para informação útil [18] [19].

#### 3.4.1. Características Pentaho

- ✓ Data integration
- ✓ Business Analytics
- ✓ Big Data Analytics
- ✓ Embedded Analytics
- ✓ Cloud Analytics
- ✓ Ad Hoc Analysis
- ✓ Online Analytical Processing (OLAP)
- ✓ Predictive Analysis
- ✓ User-Friendly Interface
- ✓ Ad Hoc Reporting
- ✓ Customizable Features
- ✓ Performance Measurements
- ✓ Intuitive dashboards

*Figura 21 - Características – Pentaho [18]*

### 3.4.2. Análises de Negócio

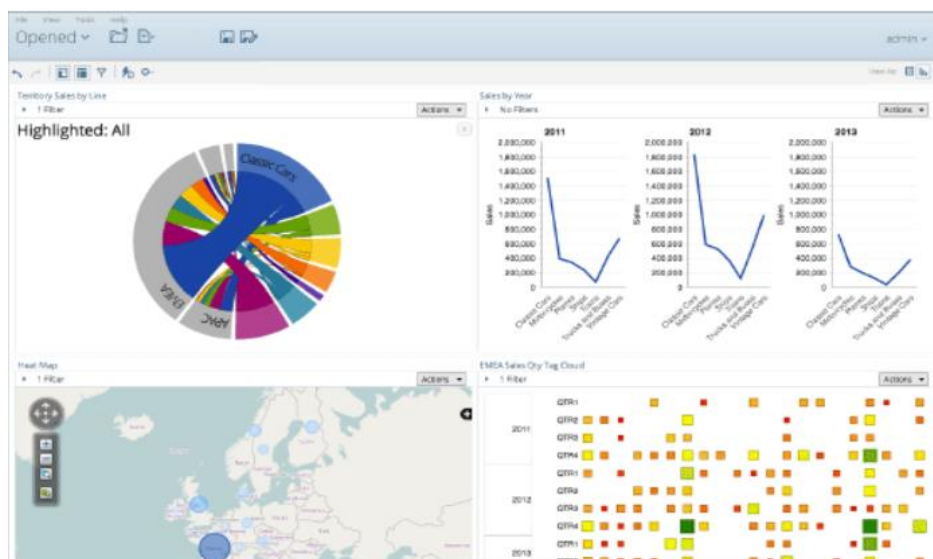


Figura 22 - Painéis de Controlo gráficos e responsivos – Pentaho [20]

Na Figura 22, observamos um exemplo dos painéis de controlo gráfico e responsivos, disponibilizados pelo Pentaho. Esta funcionalidade permite melhorar o desempenho da organização, através do uso de painéis de controlo interativos, os quais oferecem indicadores de desempenho muito importantes aos seus utilizadores, e conseqüentemente, ajudar a identificar quer as áreas críticas, quer as zonas de interesse para o negócio.

Estes painéis são totalmente personalizados, e podem ser adaptados às necessidades da empresa como, por exemplo, restringir a visualização dos dados a uma gama de produtos/vendas/clientes, num determinado local, num determinado espaço de tempo referente a um determinado ano, quadrante, mês ou dia [20].

### 3.4.3. Integração de Dados

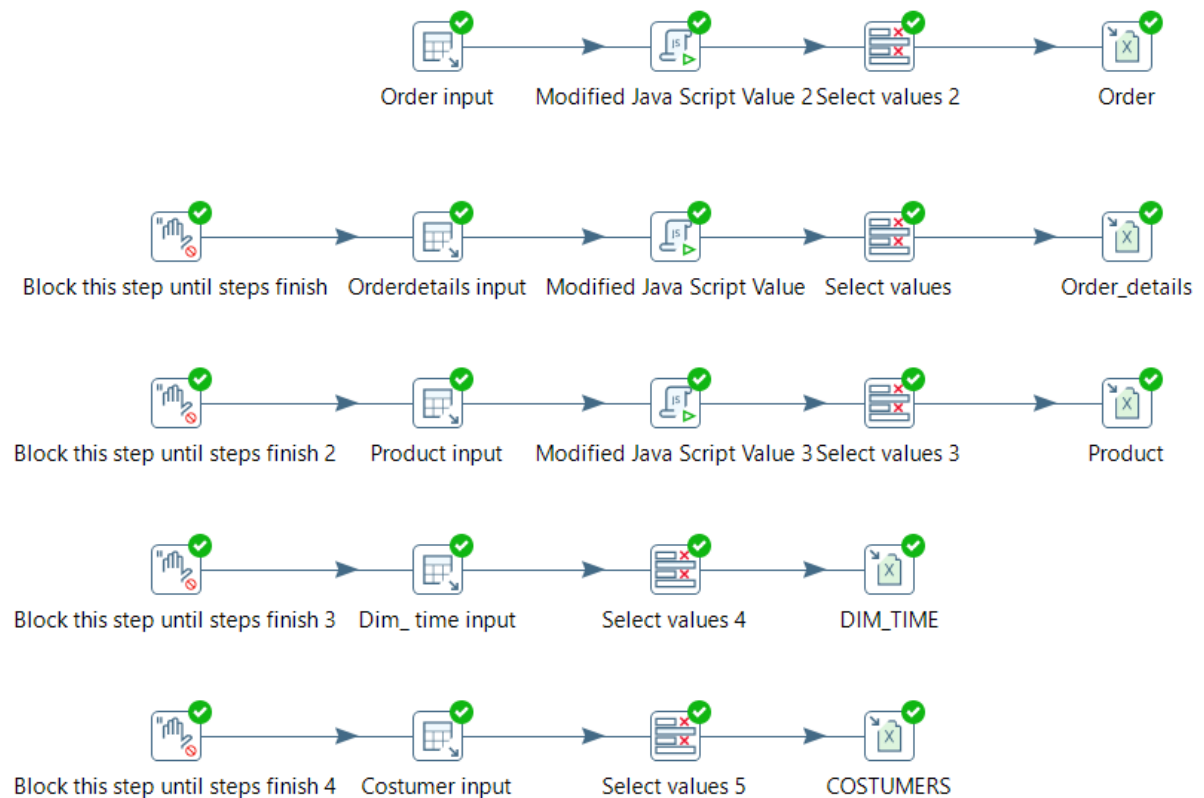


Figura 23 - Integração de dados - Pentaho

A Figura 23 ilustra uma das muitas funções do Pentaho, neste caso indica como é feita a extração de dados no Pentaho. É utilizado um *Table input*, no qual é criada uma *query* (questão) à base de dados para obtenção da informação pretendida. Em seguida essa informação proveniente da base de dados passa por uma série de transformações e, em seguida, é criado um ficheiro CSV, através do uso de um *output* para CSV. É possível efetuar *outputs* para diversos tipos de ficheiros tais como, txt, excel, mdb, etc.

### 3.5. Power BI

Trata-se de um pacote de ferramentas de análise de negócios que oferece informação importante a uma organização. Permite a conexão a centenas de fontes de dados, simplifica a preparação dos mesmos e conduz à análise *ad-hoc*. Permite, de igual modo, a criação de painéis de controlo personalizados [21].

#### 3.5.1. Exemplo de Recursos Humanos para o Power BI:

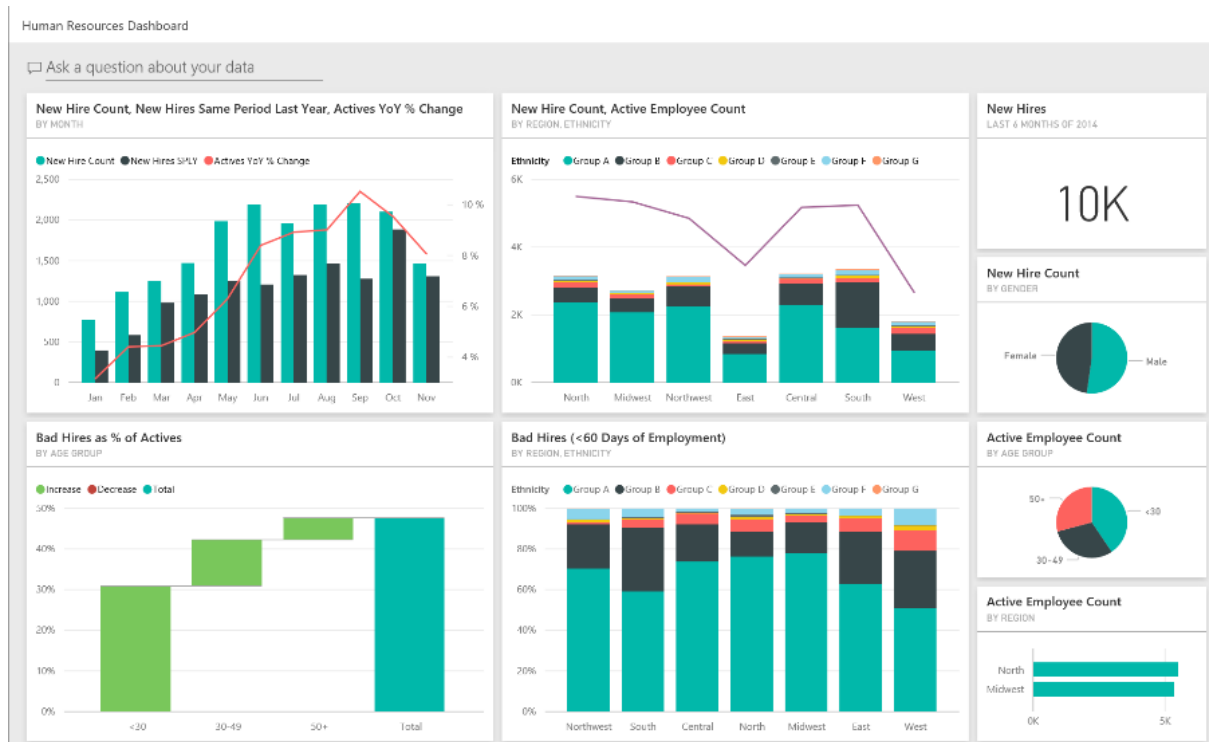


Figura 24 - Painel de Controlo RH – PowerBi [21]

O exemplo supra mostra informação relacionada com novas contratações, o número funcionários ativos, bem como os que saíram da empresa, e tenta aferir tendências na estratégia da contratação.

### 3.5.2. Análise de Vendas

Retail Analysis Sample

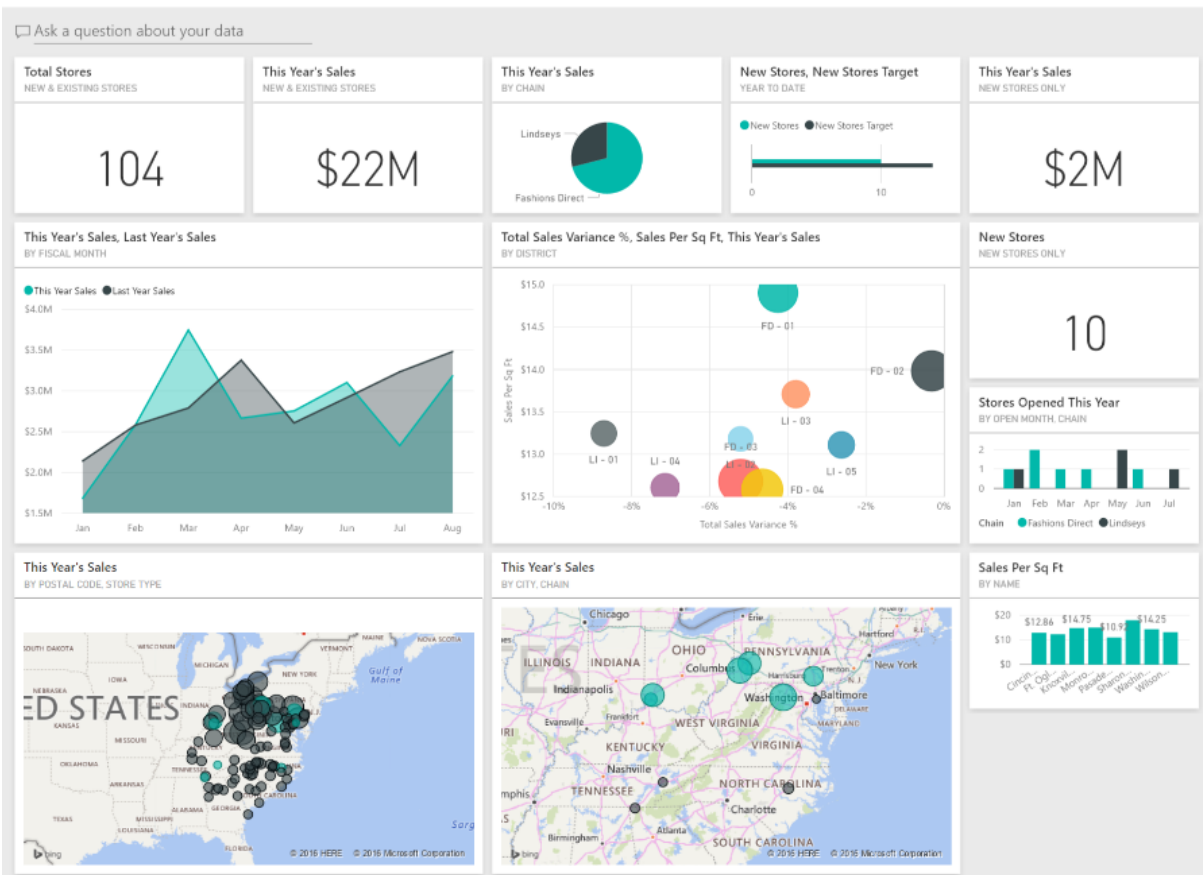


Figura 25 - Exemplo de análise de vendas – Power BI [22]

Este painel analisa os dados de vendas de itens vendidos em várias lojas e distritos. É observável a comparação do desempenho do ano em curso em relação ao ano anterior, nas seguintes áreas: vendas, unidades, margem bruta e variação, bem como uma análise do novo stock. É possível clicar sobre cada um dos gráficos apresentados, para aceder a informação mais detalhada sobre uma determinada área. Desta forma torna-se viável efetuar uma análise mais pormenorizada e precisa que, por sua vez, permitirá identificar as áreas de risco da empresa, através de cuja informação se torna possível identificar as causas de determinados problemas e traçar estratégias para os retificar [22].

### 3.6. SAP

Utilizado para ajudar os tomadores de decisões dos negócios na obtenção do conhecimento necessário para poderem prever novos desenvolvimentos e, desta forma, capitalizarem em tendências futuras, bem como responder a desafios de forma preventiva, isto é, antes que estes aconteçam. O SAP agiliza a extração de informação do Big Data (grandes quantidades de dados, estruturados ou não estruturados, que inundam os negócios no dia-a-dia), e aproveita o poder a linguagem ABAP, criando diversas visualizações de dados com grande facilidade [23].

#### 3.6.1. Características SAP

- ✓ Self Service Data Access
- ✓ Data Transformation
- ✓ Data Visualization
- ✓ SAP HANA Data Discovery
- ✓ Secure Sharing
- ✓ Web/Mobile Support

Figura 26 - Características SAP [23]

#### 3.6.2. Visualização de dados e Análise

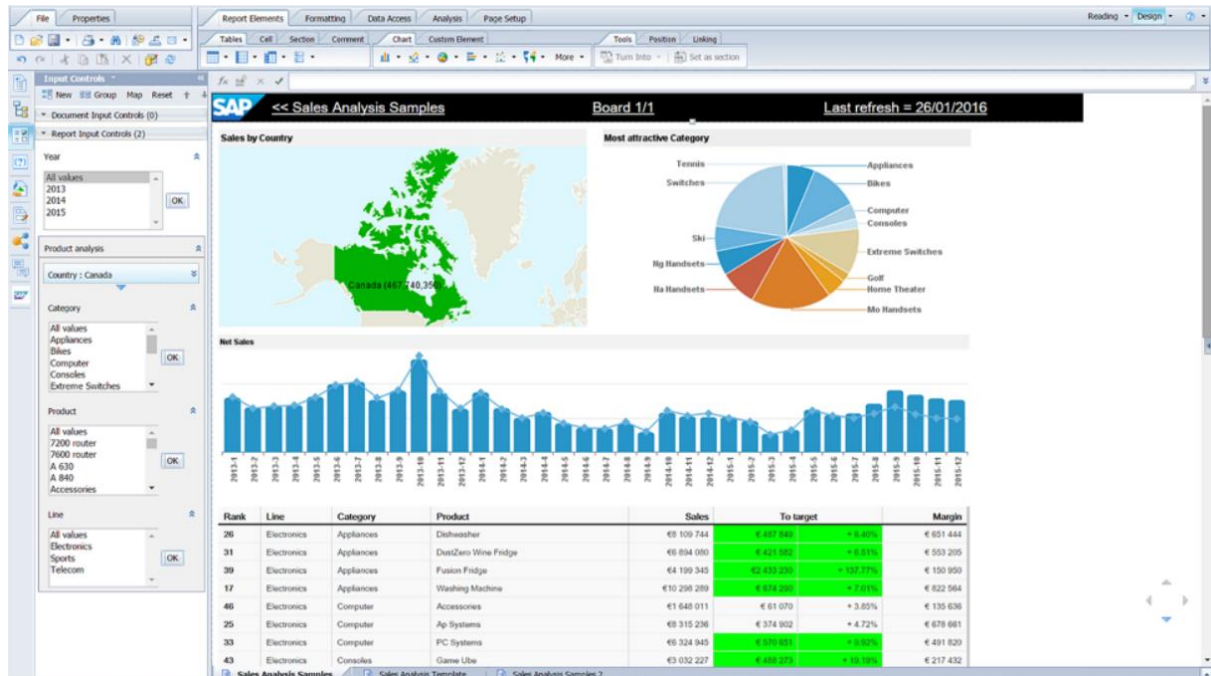


Figura 27 - Visualização de dados SAP [23]

O SAP utiliza ferramentas de visualização amigáveis ao utilizador, que permitem combinar dados vindos de múltiplas fontes, analisar tendências, e contar histórias que cativam as audiências.

Permite a partilha ágil e inteligível de informações comerciais por toda a organização, cujo fluxo se processa de fácil perceção e comunicação [23].

### 3.6.3. Painéis de Dados

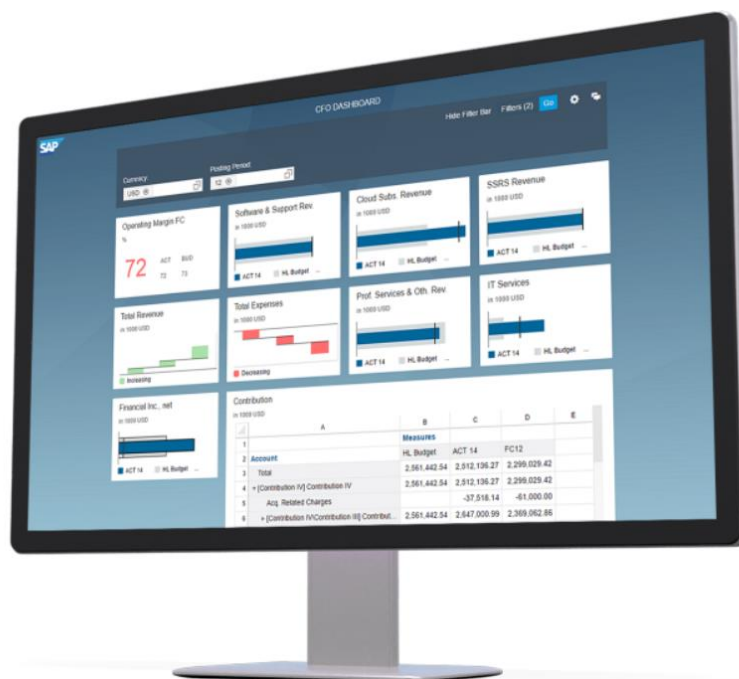


Figura 28 - Painel de dados SAP [23]

Os painéis de dados promovem uma melhor tomada de decisões por toda a organização, através da utilização de painéis de controlo e aplicações muito detalhadas, tornando, naturalmente, mais simples a compreensão de todos os dados apresentados. É possível criar painéis de controlo interativos, que apresentam aos tomadores de decisões dados de confiança e agregados, a qualquer altura e em qualquer ambiente [23].

### 3.6.4. Componentes Principais SAP ERP



Figura 29 - SAP ERP Componentes [23]

- Vendas e Distribuição (SD) - gere, tal como o próprio nome indica, os processos de negócios de vendas e distribuição de uma empresa, incluindo quer a comercialização de produtos ou serviços em mercados nacionais e internacionais, quer a venda direta aos clientes ou transação por meio de uma rede de distribuição.
- Gestão de Materiais (MM) - Gere a aquisição de materiais ou serviços e efetua gestão de stock. A procura pode ser feita através de fornecedores locais ou internacionais. Na gestão de inventário, o MM orienta todo o fluxo (tanto saída como entrada) de mercadorias e transferência de materiais. A contagem e restabelecimento de inventário, é também efetuado pelo MM.
- Plano de Produção (PP) - Esta componente pode ser utilizada para produção discreta, processual ou repetitiva, ou como uma combinação de mais do que um tipo de produção.
- Controlo de Qualidade (QM) – Função integrada, extensível aos processos de aquisição, produção, vendas e manutenção de equipamento.



- Manutenção da Planta (PM) - Ajuda a manter todo o equipamento e os locais funcionais de uma empresa em boas condições de trabalho, ajudando a prevenir eventuais falhas que possam interferir com a produção da empresa.
- Recursos Humanos (HR) – processa e monitoriza o pagamento dos salários, a assiduidade e a pontualidade, as formações e as viagens de negócio. Oferece ainda uma subcomponente subordinada ao Meio Ambiente, Saúde e Segurança no trabalho [24].
- Soluções Industriais (IS) – agiliza a criação de *software* com as melhores práticas de negócio para uma determinada área. Disponibiliza processos e ferramentas específicas, mediante um determinado setor [25].
- Fluxo de Trabalho (WF) - possibilita o desenho e execução de processos de negócio dentro de aplicações SAP. Os clientes podem melhorar os já existentes ou criar os seus próprios fluxos de trabalho [26].
- Tesouraria (TR) - conjunto de soluções voltadas para a análise e otimização de processos de negócio, dentro da área das finanças da companhia [27].
- Gestão de Ativos Fixos (AM) – facilita o agendamento e a alocação de custos dos recursos partilhados pelos funcionários como, por exemplo, os veículos [28].
- Contabilidade (FI) - gere todas as transações financeiras dentro das empresas. Permite a gestão de dados que estejam envolvidos em qualquer transação financeira ou de negócios, dentro de um sistema unificado. Este módulo é muito flexível, o que significa que funciona para qualquer situação económica. Permite, também, obter a posição financeira da empresa no mercado, em tempo real.
- Controlo (CO) – ministra a contabilidade quer dos centros de custo, quer dos centros de lucros, emite ordens internas para gerir projetos de pequeno a médio porte e o custo de produtos.

Todos estes recursos são instalados com base nas necessidades do cliente. O universo SAP é enorme e os custos variam imenso dependendo dos componentes a instalar, da quantidade de utilizadores, das necessidades de existência (ou não) de um consultor, e das manutenções ao sistema que podem ou não ser efetuadas. Vejamos, então, alguns dos pacotes disponíveis no âmbito da SAP:

- **SAP Business One** - Cada licença tem de estar associada a um único utilizador. Estas licenças são compradas de uma só vez, e depois é necessário pagar 18% anualmente sobre o valor total da licença. Normalmente uma licença tem um valor de 1700\$ - 3200\$.
- **SAP Business One Starter Package** - Tem um máximo de cinco utilizadores, é uma versão *lite* do SAP Business One, o que significa que tem menos componentes disponíveis. Cada licença possui um custo de 1400\$, com a importância de 410\$ por utilizador/por ano de subscrição.

### 3.7. Comparação entre Sistemas

De forma a efetuar uma comparação entre os sistemas já existentes e o extrator de dados, foram selecionados quatro parâmetros sobre os quais será avaliada a disponibilidade destes em cada sistema. A tabela 1 identifica os parâmetros selecionados e uma a descrição dos mesmos.

Parâmetro	Descrição
Multilingue	Se o sistema está disponível em mais do que uma língua.
Painel de Controlo	Se o sistema possui um <i>dashboard</i> , e permite a customização de um.
Pesquisa Avançada	Se o sistema permite a pesquisa de informação através da utilização de filtros.
Paginação	Se o sistema permite que os dados sejam paginados, permitindo assim a navegação entre páginas.
Exportação de Dados	Se o sistema permite exportar diversas quantidades de informação para diferentes formatos.
Plataformas Suportadas	Se o sistema é capaz de suportar várias plataformas, tais como MAC OS, Windows, Etc.
Integração de Dados	Se o sistema é capaz de efetuar várias integrações de dados através de, Dropbox, Facebook, Excel, Etc.

Tabela 1 - Parâmetros de Comparação

### 3.7.1. Descrição dos critérios de Avaliação

A avaliação dos parâmetros foi realizada utilizando uma escala de Likert, constituída por cinco níveis com os seguintes critérios correlativos: *um* representa *muito insatisfeito*, enquanto dois, três, quatro e cinco estarão para *insatisfeito*, *neutro*, *satisfeito* e *muito satisfeito*, respetivamente.

	Escala				
	1	2	3	4	5
<b>Parâmetros</b>	Muito Insatisfeito	Insatisfeito	Neutro	Satisfeito	Muito Satisfeito
Multilingue	Monolingue (Português)	Monolingue (Inglês)	Bilingue (Português e Inglês)	Trilingue (Português, Inglês e Francês)	Plurilingue (Português, Inglês, Francês, ...)
Painel de Controlo	Pesquisa de tabelas	Pesquisa de tabelas, seleção de campos	Pesquisa de tabelas, seleção de campos, filtros	Pesquisa de tabelas, seleção de campos, filtros e paginação, Exportações	Pesquisa de tabelas, seleção de campos, filtros e paginação, Exportações, gráficos e estatísticas, etc
Pesquisa Avançada	Não permitido	2 tipos de filtro	3 tipos de filtro	4 tipos de filtro	> 4 tipos de filtro
Paginação	Não permitido	-	-	-	Permitido
Exportação de dados	Não permitido	1 formato	2 formatos	3 formatos	> 4 formatos
Plataformas Suportadas	1 Plataforma	2 Plataformas	3 Plataformas	4 Plataformas	> 4 Plataformas
Integração de Dados	5 Integração	10 Integrações	20 Integrações	30 Integrações	> 30 Integrações

Tabela 2 - Descrição critérios de Avaliação

### 3.7.2. Tabela de Comparação de Tecnologias

Na Tabela 3 e nos gráficos 1-5 encontram-se os resultados obtidos da comparação efetuada entre os sistemas. Adicionalmente, aproveitou-se para incluir o extrator de dados, e encetar a sua comparação com os sistemas já existentes.

Sistemas

Parâmetros	Dundas BI	Domo	Pentaho	Power BI	SAP	Extrator de Dados
Multilingue	5	5	5	5	5	1
Painel de Controlo	5	5	5	5	5	4
Pesquisa Avançada	5	5	5	5	5	5
Paginação	5	5	5	5	5	5
Exportação de dados	4	4	5	4	4	5
Plataformas Suportadas	2	4	4	5	5	1
Integração de Dados	1	5	2	5	5	-

Tabela 3 - Comparação de Tecnologias

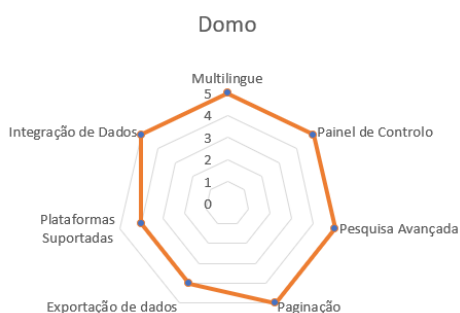


Gráfico 2 - Domo Avaliação

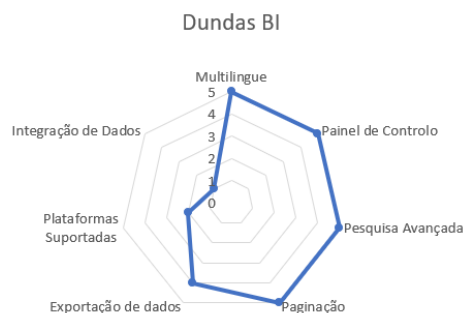


Gráfico 1 - Dundas Bi Avaliação

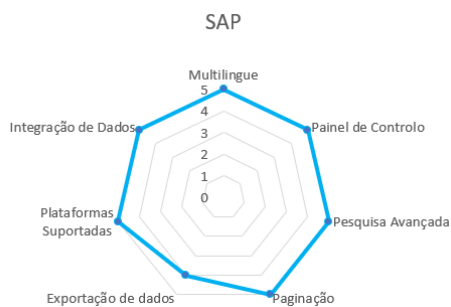


Gráfico 4 - SAP Avaliação

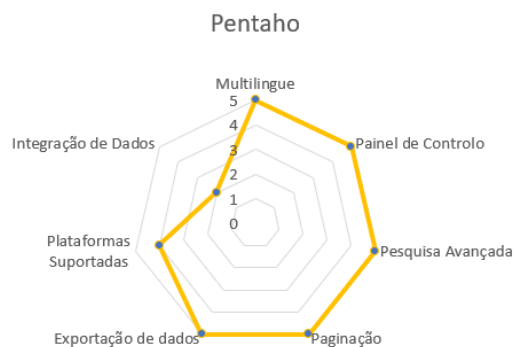


Gráfico 3 - Pentaho Avaliação

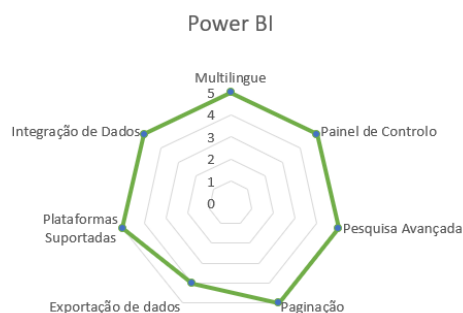


Gráfico 6 - Power Bi Avaliação

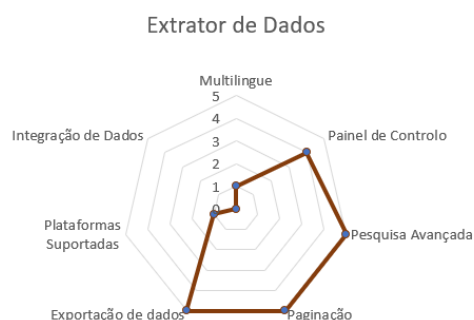


Gráfico 5 - Extrator de Dados Avaliação

No decurso da análise da tabela 3 e dos gráficos 1 - 6, podemos verificar que, no primeiro parâmetro “Multilingue”, o extrator possui uma grande desvantagem visto que apenas está disponível numa língua: o português. No que concerne ao segundo parâmetro “Painel de Controlo”, o extrator é penalizado por não disponibilizar um painel de controlo passível de ser personalizado e que inclua a possibilidade de efetuar análises de probabilidades e estatística. Nos parâmetros “Pesquisa Avançada” e “Paginação” encontra-se num patamar semelhante ao de outras tecnologias. No critério “Exportação de dados”, o extrator de dados consegue superar ligeiramente alguns dos outros sistemas, na medida em que disponibiliza mais alguns formatos de exportação. Em termos de plataformas suportadas, o extrator de dados tem uma enorme desvantagem, visto encontrar-se disponível apenas para *desktop*, e para o sistema operativo Windows. Por fim, no que diz respeito à integração de dados, aferimos ser necessária e imperativa a possibilidade de efetuar a integração com diversos *softwares*. Esta integração permitirá a obtenção de dados através de outras plataformas tais como a Dropbox, Google *Analytics*, ou outro qualquer *software* de armazenamento de informação.

## 4. Desenvolvimento

Ao longo do presente capítulo, identificamos os requisitos necessários para o bom funcionamento do extrator, tal como a arquitetura existente entre o extrator e o sistema SAP. Encontram-se também descritas, de uma forma detalhada, as soluções implementadas para a extração de dados do sistema SAP, paginação, pesquisa avançada e filtração. Por fim, atestamos os métodos de extração de dados implementados no extrator, que se trata de quatro tipos de ficheiro, cuja implementação se encontra aferida ao detalhe, na última secção deste capítulo.

### 4.1. Identificação dos Requisitos

#### 4.1.1. Requisitos Funcionais

**Requisito #:** 1

**Tipo de Requisito:** Formulário Principal

**Prioridade:** Máxima

**Descrição:** O Sistema deve permitir a visualização dos dados por meio de uma grelha.

**Razão:** A informação proveniente da base de dados SAP deve ser visualizada através de uma grelha de dados.

**Origem:** -

**Critério de Ajuste:** Esta informação deve de ser apresentada apenas a utilizadores que possuem os devidos acessos.

**Dependências:**

**Conflitos:** -

**Requisito #:** 2

**Tipo de Requisito:** Formulário Principal

**Prioridade:** Máxima

**Descrição:** O Sistema deve permitir a seleção dos campos mais relevantes da tabela a serem visualizados.

**Razão:** O utilizador tem a opção de selecionar apenas os campos que deseja visualizar.

**Origem:** -

**Critério de Ajuste:** Permite ao utilizador o isolamento de campos desnecessários.

**Dependências:** 1

**Conflitos:** -

**Requisito #: 3**

**Tipo de Requisito:** Formulário Principal

**Prioridade:** Máxima

**Descrição:** O Sistema deve permitir a seleção de filtros.

**Razão:** O utilizador tem a opção de filtrar os seus resultados.

**Origem:** -

**Critério de Ajuste:** Permite que o utilizador obtenha uma informação mais detalhada.

**Dependências:** 1 e 2

**Conflitos:** -

**Requisito #: 4**

**Tipo de Requisito:** Formulário Principal

**Prioridade:** Máxima

**Descrição:** O Sistema deve permitir a exportação dos dados para PDF, XML, HTML e CSV.

**Razão:** Visualização, *a posteriori*, dos dados exportados.

**Origem:** -

**Critério de Ajuste:** A exportação é efetuada para futura utilização dos dados em outras plataformas.

**Dependências:** 1

**Conflitos:** -

**Requisito #: 5**

**Tipo de Requisito:** Paginação

**Prioridade:** Máxima

**Descrição:** O Sistema deve paginar os dados obtidos.

**Razão:** Diminuição do impacto no desempenho.

**Origem:** -

**Critério de Ajuste:** Permite que os resultados da pesquisa sejam fragmentados, tornando a pesquisa mais rápida e eficiente.

**Dependências:** 1

**Conflitos:** -



**Requisito #: 6**

**Tipo de Requisito:** Segurança

**Prioridade:** Máxima

**Descrição:** O Sistema deve verificar se as credenciais do utilizador estão corretas.

**Razão:** Acesso ao sistema apenas a utilizadores autorizados.

**Origem:** -

**Critério de Ajuste:** Caso um utilizador não autorizado efetue o *Login*, é guardado um registo (Log) e é enviada uma mensagem ao administrador do sistema.

**Dependências:** -

**Conflitos:** -

**Requisito #: 7**

**Tipo de Requisito:** Segurança

**Prioridade:** Máxima

**Descrição:** O Sistema deve encriptar toda a informação associada à entrada e à saída do utilizador.

**Razão:** Criar um ambiente seguro, mediante o qual as credenciais do utilizador sejam protegidas.

**Origem:** -

**Critério de Ajuste:** De forma a proteger os dados do utilizador, o sistema deve encriptar toda a informação inserida pelo mesmo aquando do *login*.

**Dependências:** -

**Conflitos:** -

**Requisito #: 8**

**Tipo de Requisito:** Segurança

**Prioridade:** Intermédia

**Descrição:** O Sistema deve manter um registo do login.

**Razão:** Identificar os utilizadores que acederam ou tentaram aceder ao extrator.

**Origem:** -

**Critério de Ajuste:** Permite a identificação de todas as tentativas de *login* no extrator, podendo assim proceder à verificação de uma eventual anomalia.

**Dependências:** 6

**Conflitos:** -

**Requisito #: 9**

**Tipo de Requisito:** Segurança

**Prioridade:** Máxima

**Descrição:** O registo deve estar disponível aos administradores do sistema.

**Razão:** Apenas os administradores devem ter acesso a este registo.

**Origem:** -

**Critério de Ajuste:** A informação deve estar escondida dos utilizadores, e apenas disponível ao administrador para que possa ser averiguada em caso de anomalia.

**Dependências:** 8

**Conflitos:** -

**Requisito #: 10**

**Tipo de Requisito:** Disponibilidade

**Prioridade:** Máxima

**Descrição:** Os utilizadores devem ter acesso a um Painel de Controlo

**Razão:** O utilizador deve ter acesso a um painel de controlo, o qual pode ser personalizado.

**Origem:** -

**Critério de Ajuste:** O utilizador deve ser capaz de personalizar o seu painel de controlo, através do auxílio de opções que permitam adicionar vários tipos de gráficos, grelhas e outras opções que sejam necessárias.

**Dependências:** -

**Conflitos:** -

#### 4.1.2. Requisitos Não-Funcionais

- **Interface**
  - O sistema deve possuir uma interface esteticamente agradável aos utilizadores.
  - O sistema deve possuir uma interface de navegação pouco complexa.
- **Usabilidade**
  - O sistema deve ser fácil de utilizar e compreender.
  - O sistema deve possuir uma interface de fácil memorização.
  - O sistema deve de permitir a navegação através de paginação, por meio de botões.
- **Desempenho**
  - O tempo de resposta do sistema não deve de ultrapassar os 30 segundos.
  - O sistema deve ser capaz de atualizar a informação da grelha de dados sem afetar o seu desempenho.
  - O sistema deve apresentar uma resposta rápida a qualquer pesquisa que seja efetuada.
- **Manutenção e Portabilidade**
  - O sistema poderá ser utilizado em qualquer país.
  - O sistema poderá ser submetido a manutenção sempre que necessário.
- **Segurança**
  - O sistema deve estar protegido a tentativas de acesso por parte de utilizadores desconhecidos, nomeadamente aqueles que não possuem uma conta de utilizador SAP.
  - As autorizações de acesso ao sistema podem ser alteradas apenas pelo administrador do sistema.

## 4.2. Arquitetura

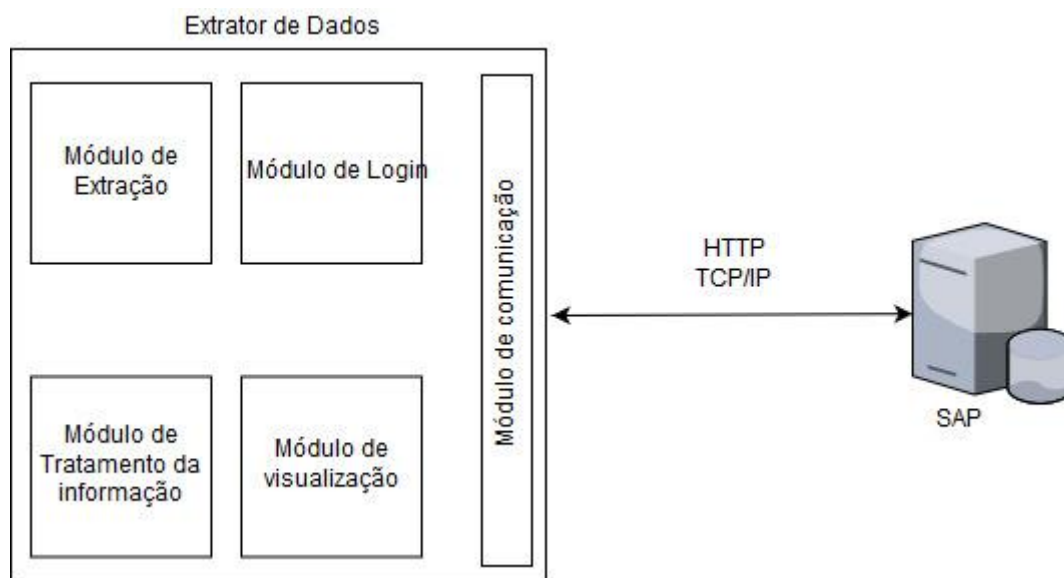


Figura 30 – Arquitetura do Extrator de dados

O extrator de dados é constituído por cinco módulos:

- Módulo de Login - É apresentado um formulário pelo qual é possível introduzir os dados de autenticação do utilizador. A descrição deste módulo encontra-se na secção 4.10.1, este módulo trabalha em conjunto com o módulo de comunicação para proceder à autenticação das credenciais do utilizador.
- Módulo de Comunicação - Trata de efetuar a comunicação entre o sistema SAP e o extrator de dados. (Secção 4.3.1 – 4.3.1.2)
- Módulo de Visualização - Disponibiliza um formulário de pesquisa, o qual permite selecionar o nome da tabela, campos da tabela, filtros, paginação e exportação dos dados para um dos tipos de ficheiro implementados. (Secção 4.10 – 4.10.2)
- Módulo de Extração - Trata de efetuar os pedidos de forma a obter os dados correspondentes à tabela introduzida. (Secção 4.3.3)  
É de referir que este módulo utiliza as funcionalidades disponibilizadas, pelo módulo de comunicação de forma a efetuar a extração da informação.
- Módulo de Tratamento de Informação - Efetua o tratamento da informação recebida através da pesquisa efetuada, preparando os dados para serem apresentados e exportados, consoante a consulta efetuada pelo utilizador através do módulo de visualização. (Secção 4.5 – 4.9.4)

Para que possamos compreender a arquitetura deste sistema, importa observar o exemplo que se segue, por meio do qual se procede a uma pesquisa à tabela “SPFLI”, para retornar todo o seu conteúdo.

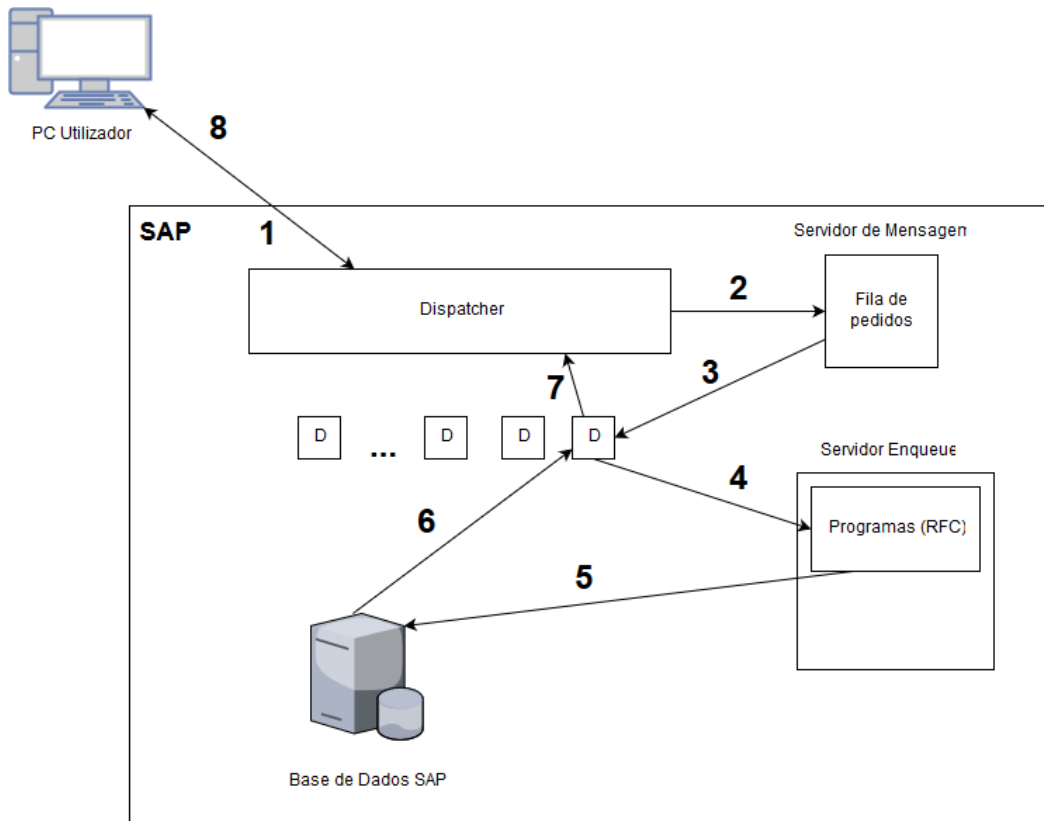


Figura 31 - Arquitetura

1. O *input* (“SPFLI”) é passado para o *dispatcher* da instância central.
2. O *dispatcher* envia o pedido para o Servidor de Mensagem (M).
3. O servidor de Mensagem coloca o pedido dentro de um dos processos de diálogo de trabalho (D).
4. O processo de diálogo (D) procede à leitura da base de dados, através do Servidor *Enqueue* (E).
5. O Servidor *Enqueue* (E) passa o pedido dos dados para a base de dados SAP.
6. A base de dados retorna os dados pretendidos para o processo de diálogo (D) correspondente.
7. O processo de diálogo (D) passa os dados para o *dispatcher*.
8. O *dispatcher* retorna o resultado da operação para a camada de apresentação (Extrator de Dados) e o utilizador pode visualizar os resultados no seu ecrã.

### 4.2.1 Diagrama de Sequência - Extrator de Dados

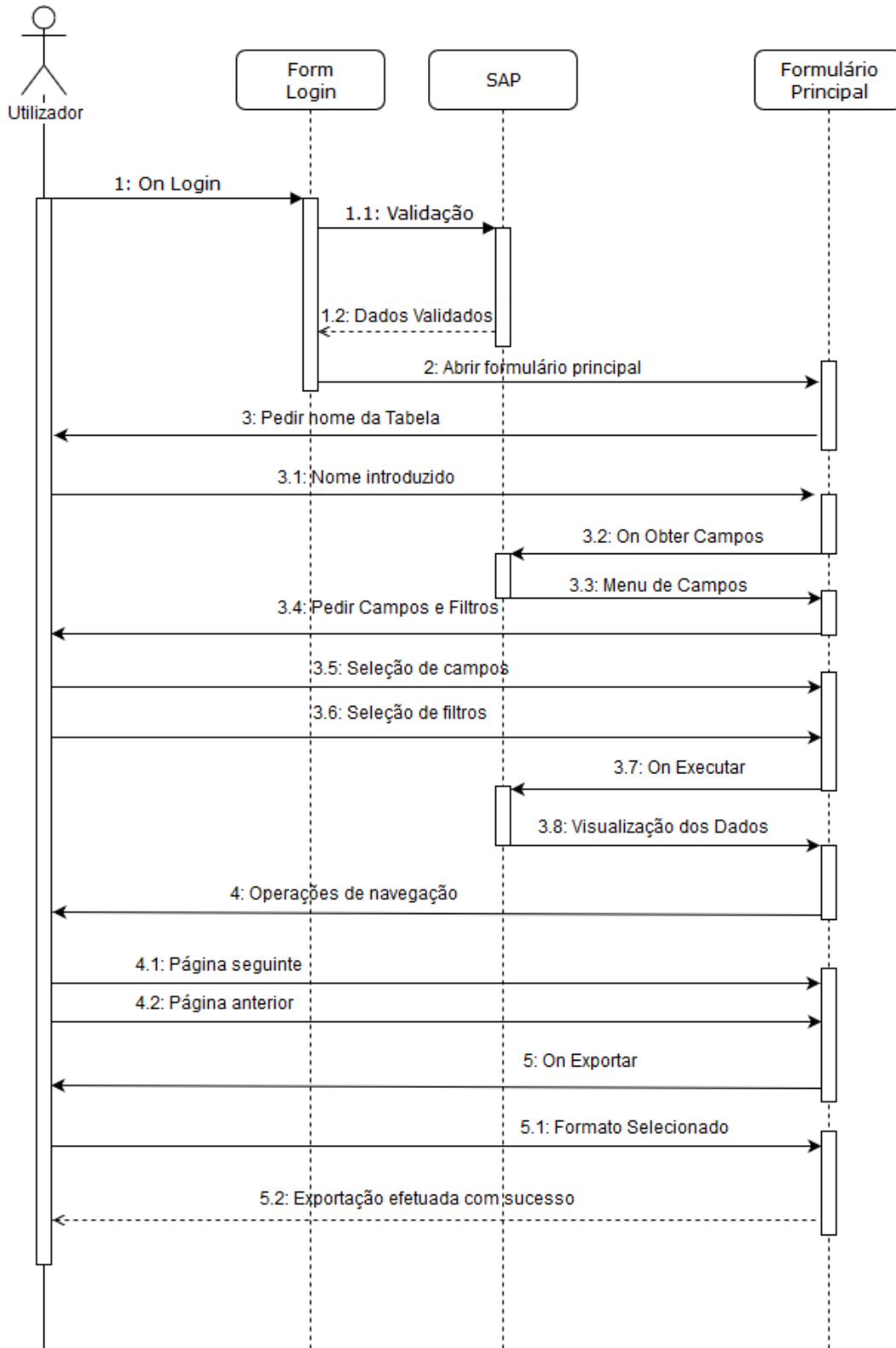


Figura 32 - Diagrama de Sequência

### 4.3. Extração de dados SAP

Antes de efetuar qualquer tipo de extração de dados, é necessário que o utilizador se conecte ao servidor SAP. Esta conexão é estabelecida através da utilização do SAP.NET CONNECTOR 3.0 [30], que envia os dados do utilizador para verificação pelo SAP GUI. Após efetuar a sua autenticação com sucesso, o utilizador acede às funcionalidades do SAP, que se encontram disponíveis no extrator de dados, nomeadamente a visualização de tabelas de dados e extração dos resultados das mesmas. A obtenção dos dados é efetuada do lado do ambiente SAP, onde são utilizados três RFCs. Cada RFC é criado utilizando as regras SAP, onde os RFCs criados devem seguir o seguinte formato:

- Z mais ND mais nome do RFC
- Y mais ND mais nome do RFC.

Este processo permitirá a sua distinção entre os RFCs, já existentes no *software* SAP, e aqueles que foram desenvolvidos pelo programador. É importante notar que Z é utilizado para identificar programas que são personalizados ou implementados para uso do cliente, e Y para programas de testes. Embora seja o formato de identificação recomendado, não significa que seja adotado pela empresa [31].

#### 4.3.1. Arquitetura SAP Connector 3.0

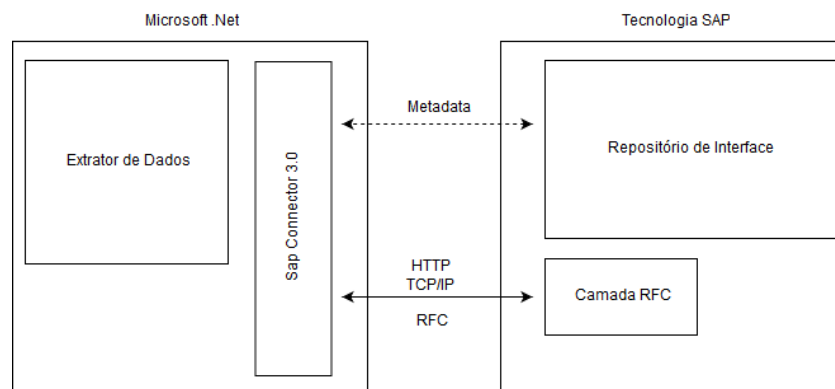


Figura 33 - Arquitetura SAP Connector 3.0

A conexão entre o extrator de dados e a tecnologia SAP é efetuada através do SAP Connector 3.0, a qual é estabelecida através do protocolo RFC da SAP, do tipo HTTP TCP/IP.

O conector processa um pedido da aplicação .NET, que é convertido para o formato RFC e enviado através do protocolo RFC para o sistema SAP selecionado. Esta seleção e conexão é

primeiramente estabelecida por meio da classe **RfcConfigParameter**”, no qual é criado e instanciado um objeto que possui os parâmetros da conexão (Figura 34 e

Figura 35). Seguidamente, é necessária a utilização da classe **RfcDestination** mediante a qual é instanciado um objeto do tipo **Rfc Destination**, posteriormente utilizado em conjunto com o método **GetDestination** da classe **RfcDestination Manager** (Figura 36), de forma a criar o destino para o servidor SAP a empregar [32] [33] [34].

#### 4.3.1.1 Utilização do SAP .NET CONNECTOR 3.0

Para utilizar o SAP .NET CONNECTOR 3.0 é necessário, numa primeira fase, adicionar a sua biblioteca ao projeto a desenvolver. Depois de adicionada, todas as suas funcionalidades ficam imediatamente disponíveis. A utilização das suas funcionalidades é efetuada da seguinte forma:

1. Criação dos parâmetros de conexão através da declaração de um objeto **RfcConfigParameters**.

```
RfcConfigParameters meusap = new RfcConfigParameters();
```

Figura 34 - Declaração objeto RfcConfigParameters

2. Depois de declarado o objeto **meusap** do tipo **RfcConfigParameters**, é necessário efetuar o preenchimento dos parâmetros, de forma a estabelecer a conexão com o servidor SAP pretendido.

```
meusap.Add(RfcConfigParameters.Name, " ");
meusap.Add(RfcConfigParameters.AppServerHost, " : ");
meusap.Add(RfcConfigParameters.SystemNumber, " ");
meusap.Add(RfcConfigParameters.Client, " - ");
meusap.Add(RfcConfigParameters.User, " ");
meusap.Add(RfcConfigParameters.Password, " . ");
```

Figura 35 - Preenchimento de parâmetros

Finalmente, para verificar se o servidor está *online*, e para permitir o acesso à aplicação, é efetuado um **ping**, dentro de um **try/catch**.

```
// Cria destino
RfcDestination meurfc = RfcDestinationManager.GetDestination(meusap);
try
{
    if (meurfc != null)
    {
        // Verifica conexão
        meurfc.Ping();
        MessageBox.Show("SAP - Conectado com sucesso.");
        MainForm.meurfc1 = meurfc;
        DialogResult = DialogResult.OK;
    }
}
catch (Exception ex)
{
    // Erro ao conectar
    MessageBox.Show(ex.ToString(), "Erro - SAP Desconectado. " +
        "Verifique se os dados introduzidos estão correctos.",
        MessageBoxButtons.OK, MessageBoxIcon.Error);
}
```

Figura 36 - Verificação da conexão



### 4.3.1.2. Utilização de RFC – SAP CONNECTOR 3.0

O RFC é o meio de comunicação disponibilizado pela SAP. É através deste protocolo que se torna possível efetuar a troca de informação entre ambientes diferentes. De forma a utilizar um RFC para efetuar a migração de dados para um ambiente externo, é necessário ativar o tipo de processamento “Remote-Enabled Module”. Tal pode ser verificado na Figura 37.

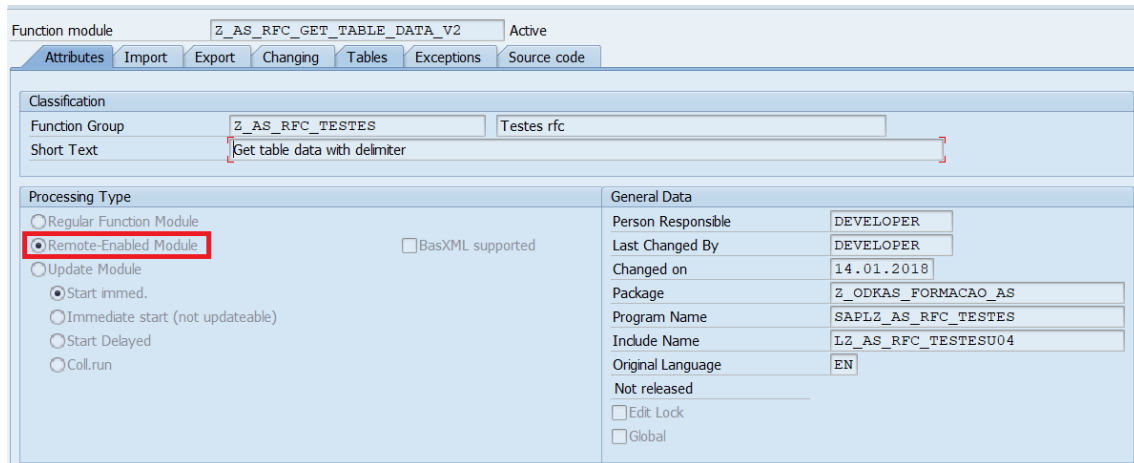


Figura 37 - RFC Remote-Enabled Module

Após a seleção do tipo de processamento, é indispensável selecionar a aba “Import”, de forma a definir quais os tipos de importação a serem utilizados. Este passo da maior importância em todo o processo, pois é através destas variáveis declaradas que são transferidos os parâmetros do ambiente externo para o ambiente SAP, de forma a obter os resultados pretendidos. Exemplos de declaração destes parâmetros estão disponíveis na Figura 38.

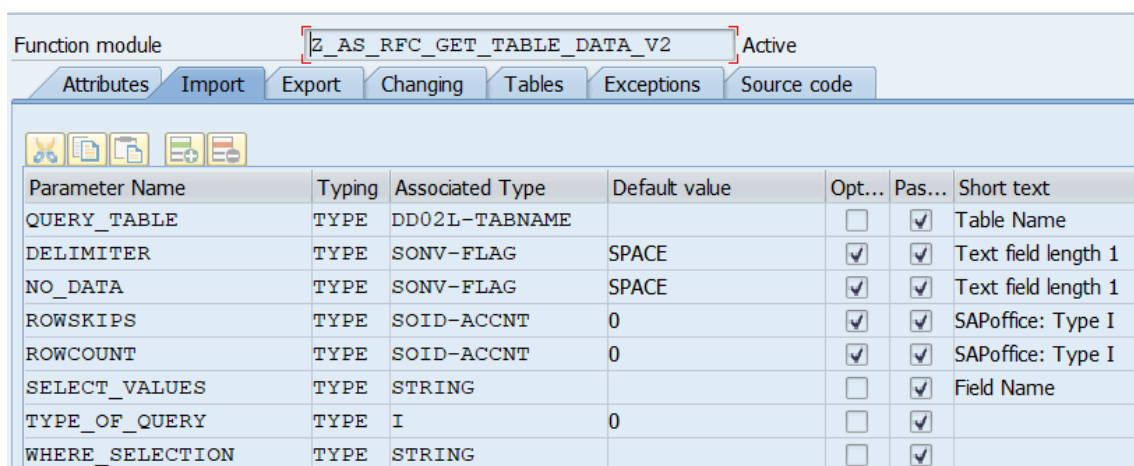


Figura 38 - Aba de importação

Numa última instância, são declaradas as variáveis de exportação. Estas variáveis podem ser declaradas na aba “Export” ou “Tables”, dependendo da quantidade de informação que se pretende obter. Tratando-se de uma quantidade reduzida de informação (como por exemplo os nomes dos campos das tabelas ou o número de registos de uma determinada tabela), bastaria efetuar a declaração na aba “Export”; por outro lado, para o caso de grandes quantidades de informação (nomeadamente dados provindos das tabelas de dados SAP), deve ser declarada a variável de exportação na aba “Tables”. A Figura 39 ilustra um exemplo da sua declaração.

Parameter Name	Typing	Associated Type	Optional	Short text
DATA	LIKE	TAB512	<input type="checkbox"/>	Table with a 512 byte field
FIELDS	LIKE	RFC_DB_FLD	<input type="checkbox"/>	RFC Table Read: Description of Fields to Retri..
OPTIONS	LIKE	RFC_DB_OPT	<input type="checkbox"/>	RFC Table Read: Select Options / WHERE Cla..

Figura 39 - Aba Tables

Concluídas as configurações do RFC, verifica-se que este se encontra apto a ser utilizado pelo ambiente .NET, para acesso à informação disponível no ambiente SAP. Um exemplo da forma de como são utilizados os RFCs no ambiente externo encontra-se representado na Figura 40.

```

RfcRepository repo = meurfc1.Repository;
IRfcFunction funcaoRT = repo.CreateFunction("Z_AS_RFC_GET_TABLE_DATA_V2");
funcaoRT.SetValue("SELECT_VALUES", selected_fields);
funcaoRT.SetValue("QUERY_TABLE", INP_tabname);
funcaoRT.SetValue("DELIMITER", ",");
funcaoRT.SetValue("ROWSKIPS", (CurrentPageIndex - 1) * PageSize);
funcaoRT.SetValue("ROWCOUNT", PageSize);
funcaoRT.SetValue("TYPE_OF_QUERY", query_number);
if (query_number == 1)
{
    funcaoRT.SetValue("WHERE_SELECTION", where_query);
}
    
```

Figura 40 - Utilização de RFC e Variáveis de importação

Como podemos visualizar na Figura 40, para aceder ao RFC pretendido é necessário utilizar o método **CreateFunction**, no qual é passado como parâmetro o nome do RFC que se pretende utilizar. As variáveis que se encontram entre aspas representam alguns dos parâmetros previamente declarados na aba **Imports**. São apenas utilizados os parâmetros que o utilizador considere necessários, aqueles que não têm qualquer tipo de utilidade tomam os seus valores por defeito. A introdução dos valores é feita da seguinte forma **funcaoRT.SetValue("<Parâmetro da aba Import>", <Input correspondente ao parâmetro>)** (Figura 40).

Após a pesquisa, é da maior importância proceder ao tratamento dos valores obtidos, para que estes possam ser facilmente visualizados pelo utilizador. A Figura 41 exemplifica como se processa a sua obtenção.

```
funcaoRT.Invoke(meurfc1);  
IRfcTable getData = funcaoRT.GetTable("DATA");  
string data = getData.ToString();  
string fixed_data = Fixstuff.FixMystring(data);
```

Figura 41 - Exportação de dados

Verificamos, por meio da Figura 41 que, para obter os dados pesquisados, é necessário utilizar o método **GetTable("DATA")**, sendo que **DATA** representa a variável de exportação que contém os dados da pesquisa. Importa declarar uma variável do tipo **IRfcTable**, de forma a empregar o resultado obtido através do SAP. Esta variável nova é primeiramente convertida para *string*, após o qual é utilizado o método **FixMystring**, da classe **Fixstuff**, para efetuar o seu tratamento. A descrição detalhada deste tratamento encontra-se na secção 4.4.3 deste documento.

### 4.3.2. Desenvolvimento de RFCs

Ao longo do processo, verificámos algumas dificuldades na criação dos RFCs visto que, para tratar tabelas que são apenas conhecidas no tempo de execução (runtime) urge a utilização de apontadores que auxiliem na declaração da tabela interna. Esta tabela interna trata-se de, no fundo, de um array, que deve ser declarado antes do runtime; tal não é possível, porém, pois a tabela que será inserida pelo utilizador é, naquele ponto do processo, ainda desconhecida, motivo pelo qual a tabela interna tem de ser imperativamente criada já no tempo de execução através da utilização de um "CREATE DATA". Este é declarado simultaneamente à execução (runtime) do tipo da tabela pretendida, e depois é utilizado como referência para a tabela interna. A Figura 42 ilustra de uma forma muito resumida a aplicação deste princípio. Após efetuar esta implementação, é possível efetuar as operações necessárias para obtenção de todos os dados necessários [35] [36].

```
1 DATA: dref TYPE REF TO data.  
2  
3 FIELD-SYMBOLS: <itab> TYPE any.  
4  
5 CREATE DATA dref TYPE STANDARD TABLE OF (query_table).  
6 ASSIGN dref ->* TO <itab>.
```

Figura 42 – Create Data

### 4.3.3. RFCs utilizados para extração de dados

Para efetuar a extração dos dados são utilizados os seguintes RFCs:

1. Z\_AS\_RFC\_GET\_TABLE\_DATA\_V2 – Utilizado para obter todos os dados da tabela introduzida na pesquisa.
2. Z\_AS\_GET\_TABLE\_ENTRIES – Utilizado para obter o número de resultados da tabela introduzida.
3. Z\_AS\_RFC\_GET\_FIELD\_DATA – Utilizado para obter os nomes dos campos da tabela introduzida.

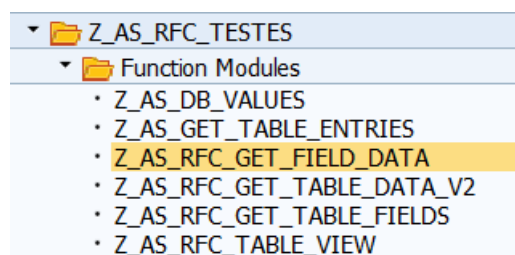


Figura 43 - Pacote de Armazenamento

O RFC Z\_AS\_RFC\_GET\_TABLE\_DATA\_V2 trata de adquirir todos os dados da tabela, consoante os parâmetros introduzidos. Permite, ainda, identificar o número de registos a obter e a partir de que posição devem ser alcançados os dados, mediante a utilização do “ROWCOUNT” e “ROWSKIP” respetivamente. O “ROWCOUNT” indica a quantidade de resultados pretendida (por exemplo 100 registos), e o “ROWSKIP” indica a posição a partir da qual deve obter os dados. Se indicarmos “ROWSKIP=5”, significa que os dados a extrair têm início a partir do quinto registo até ao centésimo quinto registo, pois tal significa que definimos o que pretendíamos: apenas 100 registos.

Os valores obtidos através do RFC Z\_AS\_RFC\_GET\_TABLE\_DATA\_V2 são enviados para o ambiente .net no formato de uma tabela TAB512 (Figura 44).

```
TABLE [STRUCTURE TAB512 { FIELD WA=001, ,0017, ,NEW YORK , , }]
[STRUCTURE TAB512 { FIELD WA=001, ,0820,FRANKFURT/MAIN , , }]
[STRUCTURE TAB512 { FIELD WA=001, ,1699,NEW YORK , , }]
[STRUCTURE TAB512 { FIELD WA=001, ,1984,SAN FRANCISCO , , }]
[STRUCTURE TAB512 { FIELD WA=001, ,0400,FRANKFURT , , }]
[STRUCTURE TAB512 { FIELD WA=001, ,0402,FRANKFURT , , }]
[STRUCTURE TAB512 { FIELD WA=001, ,9981,FRANKFURT , , }]
```

Figura 44 - Formato TAB512

Chegados a esta fase no processo, cumpre-nos proceder ao tratamento deste formato do lado do ambiente .net, de forma a eliminar os valores desnecessários, ficando apenas os dados relevantes utilizados para a visualização.

O mesmo sucede com os RFC Z\_AS\_GET\_TABLE\_ENTRIES e Z\_AS\_RFC\_GET\_FIELD\_DATA, os quais chegam ao ambiente .net com o formato de uma exportação do tipo *String*, que representa o número de entradas e os nomes dos campos das tabelas inseridas em *runtime*, respetivamente (Figura 45 e Figura 46).

```
FUNCTION Z_AS_GET_TABLE_ENTRIES (EXPORT PARAMETER NUMBER_OF_ENTRIES=112057,  
IMPORT PARAMETER QUERY_TABLE=DD02L, TABLES PARAMETER ROW_NUMBER=TABLE )
```

Figura 45 - Formato tipo String - Z\_AS\_GET\_TABLE\_ENTRIES

```
FUNCTION Z_AS_RFC_GET_FIELD_DATA  
(EXPORT PARAMETER EXPORT_DATA=MANDT,CARRID,CONNID,COUNTRYFR,  
CITYFRO4IRPFROM,COUNTRYTO,CITYTO,AIRPTO,FLTIME,  
DEPTIME,ARRTIME,DISTANCE,DISTID,FLTYPE,PERIOD,  
IMPORT PARAMETER QUERY_TABLE=SPFLI)
```

Figura 46 - Formato tipo String - Z\_AS\_RFC\_GET\_FIELD\_DATA

É igualmente necessário efetuar o seu tratamento no ambiente .net, de modo a filtrar e eliminar os valores desnecessários que são também enviados.

A necessidade de criar três RFCs distintos para a extração de dados é sobremaneira importante, pois a utilização de apenas um RFC para extrair uma grande quantidade de informação, pode traduzir-se em problemas de desempenho para a aplicação devido à quantidade de informação que está a ser transportada; desta forma, cada RFC trata de uma parte específica, evitando grandes impactos no desempenho do extrator, sendo que são apenas utilizados quando necessário.

## 4.4. Dificuldades Encontradas na Extração de Dados

### 4.4.1. Limitações SQL SAP

Devido a limitações do OPEN SQL [37] utilizado pelo SAP encontradas ao longo do desenvolvimento do extrator, verificou-se que não é possível obter diretamente os nomes dos campos das tabelas de dados. Para resolver esta situação, utilizou-se a tabela DD03L que contém informações de todas as tabelas existentes no sistema e, a partir desta, é possível obter os campos necessários através da query representada na Figura 46.

```
1 SELECT fieldname
2 FROM dd03l
3 WHERE tabname = @query_table
4 INTO CORRESPONDING FIELDS OF TABLE @it_dd03l
```

Figura 46 - Query para obter os nomes dos campos

Abaixo encontra-se descrita a utilização das palavras-chave e suas variáveis, bem como exemplos dos valores que devem receber.

- No “SELECT” é introduzido o campo pretendido da tabela, através da variável “fieldname”.
- No “FROM” a tabela onde estão contidos os dados (DD03L).
- Por fim, no “WHERE”, é digitado o nome da tabela através da variável “query table”, que representa a tabela sobre a qual efetuamos a pesquisa, por exemplo “SPFLI”.

Após a inserção destes dados, obtemos todos os nomes dos campos da tabela inserida durante o tempo de execução (*runtime*), cujo formato pode ser visualizado na Figura 46.

## 4.5. Pesquisa personalizada

De forma a apresentar apenas dados relevantes para o utilizador, foi necessária a introdução de uma pesquisa personalizada. Devido a limitações por parte do SQL do SAP, não é possível a introdução de uma cláusula “WHERE” na *query* já existente, sem modificar a *query* atual, daí ser necessária a introdução de uma variável de controlo e de duas *queries* diferentes para resolver este problema. Se a variável de controlo tomar o valor 0, significa que a pesquisa é efetuada normalmente sem qualquer tipo de filtragem. No entanto, se a variável de controlo tomar o valor 1, indica que a pesquisa possui filtragem de valores. Esta solução foi implementada em dois RFCs:

- Z\_AS\_RFC\_GET\_TABLE\_DATA\_V2
- Z\_AS\_GET\_TABLE\_ENTRIES

A Figura 47 demonstra uma das implementações da solução, em pseudocódigo.

```
1  IF TYPE_OF_QUERY = 0.  
2    <Query Statement>  
3    <Prepare Output>  
4  ELSE.  
5    <Query Statement>  
6    <Prepare Output>  
7  ENDIF.
```

Figura 47 - Pesquisa personalizada com variável de controlo

Nesta implementação encontra-se representado o requisito número três.

## 4.6. Tratamento de Dados Ambiente .Net

Para procedermos ao tratamento de diversos dados ao longo da execução do extrator de dados, foi necessário implementar uma classe que tenha as funcionalidades necessárias, para retificar o formato de como os dados são recebidos, pelos diferentes RFCs implementados no ambiente SAP. O objetivo desta classe é a limpeza desses dados adicionais, que são desnecessários para o bom funcionamento da aplicação, nomeadamente *Strings*, números ou caracteres.

### 4.6.1 Tratamento do formato TAB512

Na Figura 44, podemos observar o formato TAB512 que contém dados desnecessários para a sua visualização numa grelha de dados [38]. Estes dados precisam de ser tratados, para que os valores corretos possam ser visualizados na tabela de dados presente na interface da aplicação.

O problema consiste na remoção do seguinte padrão, deixando apenas <Dados necessários>:

**“TABLE [STRUCTURE TAB512 { FIELD WA= <Dados necessários> }]  
[STRUCTURE TAB512 { FIELD WA= <Dados necessários> }]”**

De modo a permitir a remoção destes elementos, foram utilizadas expressões regulares (Regex [39]), que recebem um padrão imutável, e efetua uma iteração pela *string* à procura do mesmo, removendo todas as suas ocorrências. No entanto, devido a limitações do *Regex*, foi necessário dividir o padrão em vários padrões diferentes e mais pequenos, nomeadamente:

**1. “TABLE ”**

Resultado: **[STRUCTURE TAB512 { FIELD WA= <Dados necessários> }]  
[STRUCTURE TAB512 { FIELD WA= <Dados necessários> }]**

**2. “STRUCTURE TAB512 “**

Resultado: **[{ FIELD WA= <Dados necessários> }]  
[{ FIELD WA= <Dados necessários> }]**

**3. “ FIELD WA=”**

Resultado: **[{ <Dados necessários> }]  
[{ <Dados necessários> }]**

**4. “[” e “]” efetuando a substituição do parênteses reto direito pelo símbolo “;”.**

Resultado: **{ <Dados necessários> };  
{ <Dados necessários> };**

**5. “{” e “}”**

Resultado: **<Dados necessários>;  
<Dados necessários>;**

Graças a este processo, são eliminados todos os dados desnecessários, permanecendo apenas os dados relevantes para a população da grelha de dados do extrator.



## 4.6.2. Tratamento do Formato de Exportação do tipo *String*

Por meio do Formato de Exportação do tipo *String* (representado nas Figura 45 e Figura 46), perspetivamos, novamente, a remoção de informação desnecessária. Uma vez que o seu formato não é tão complexo como o do TAB512, torna-se mais simples efetuar o seu tratamento, não sendo necessário um grande número de passos no desenrolar do processo.

### 4.6.2.1. Tratamento de Dados RFC - Z\_AS\_GET\_TABLE\_ENTRIES

O objetivo consiste em remover toda a informação identificada pela cor laranja, ficando permanecendo o valor que se encontra identificado pela cor verde. Inicialmente, verificamos o seguinte formato, obtido através da tabela “DD02L”:

```
FUNCTION Z_AS_GET_TABLE_ENTRIES (EXPORT PARAMETER NUMBER_OF_ENTRIES=112057,  
IMPORT PARAMETER QUERY_NUMBER=0, IMPORT PARAMETER QUERY_TABLE=DD02L,  
IMPORT PARAMETER WHERE_SELECTION=, TABLES PARAMETER ROW_NUMBER=TABLE
```

Com o intuito de obter o resultado pretendido, é efetuada uma pesquisa pelo primeiro “=”, e, selecionados todos os dados que se encontram depois deste, obtemos o seguinte:

- Resultado: 112057, IMPORT PARAMETER QUERY\_NUMBER=0, IMPORT PARAMETER QUERY\_TABLE=DD02L, IMPORT PARAMETER WHERE\_SELECTION=, TABLES PARAMETER ROW\_NUMBER=TABLE

Após este passo, é possível efetuar um Split [40] através do símbolo “,”; como queremos apenas o número “112057”, devemos ainda especificar que o valor pretendido se encontra na posição 0, e o resultado é o seguinte:

- Resultado: 112057

Este valor é depois convertido de *string* para número inteiro através de um `Int32.Parse` [41], e pode agora ser utilizado pela classe de paginação para efetuar cálculos.

#### 4.6.2.2. Tratamento de Dados RFC - Z\_AS\_GET\_FIELD\_DATA

Inicialmente, temos o seguinte formato, obtido através da tabela “SPFLI”:

```
FUNCTION Z_AS_RFC_GET_FIELD_DATA (EXPORT PARAMETER  
EXPORT_DATA=MANDT,CARRID,CONNID,COUNTRYFR,CITYFROM,AIRPFROM,COUNTRYTO,CITY  
TO,AIRPTO,FLTIME,DEPTIME,ARRTIME,DISTANCE,DISTID,FLTYPE,PERIOD, IMPORT PARAMETER  
QUERY_TABLE=SPFLI)
```

Pretende-se efetuar a eliminação da informação que se encontra identificada pela cor laranja, de forma a manter apenas a informação identificada pela cor verde. Para obtenção do resultado pretendido, é efetuada uma pesquisa pelo primeiro “=”, e, selecionados todos os dados que se encontram depois deste, obtemos o seguinte:

- Resultado:

```
MANDT,CARRID,CONNID,COUNTRYFR,CITYFROM,AIRPFROM,COUNTRYTO,CITYTO,AIRP  
TO,FLTIME,DEPTIME,ARRTIME,DISTANCE,DISTID,FLTYPE,PERIOD, IMPORT PARAMETER  
QUERY_TABLE=SPFLI)
```

Agora é efetuado um Split através do símbolo “ ” (espaço vazio); como queremos apenas os campos da tabela, devemos especificar que os dados pretendidos se encontram na posição 0, e obtemos o seguinte:

- Resultado:

```
MANDT,CARRID,CONNID,COUNTRYFR,CITYFROM,AIRPFROM,COUNTRYTO,CITYTO,AIRP  
TO,FLTIME,DEPTIME,ARRTIME,DISTANCE,DISTID,FLTYPE,PERIOD
```

Os dados obtidos no resultado final podem agora ser utilizados para efetuar o preenchimento da secção “Seleção de Campos”, na qual o utilizador pode escolher quais os campos que pretende visualizar na grelha de dados.

## 4.7. Paginação

A paginação é uma das partes mais fundamentais deste extrator de dados, pois permite navegar os resultados obtidos, o que significa que podemos obter os dados em parcelas, em vez de tratarmos de uma grande quantidade de informação de uma só vez. Desta forma, o impacto no desempenho da aplicação é reduzido, e mantém um bom grau de eficiência. Esta implementação é referente ao requisito número cinco.

Para determinar o número total de páginas é utilizada a seguinte fórmula:

$$TotalPages = \frac{RowNumber}{PageSize}$$

- TotalPages – Número total de páginas.
- RowNumber – Número de linhas da tabela.
- PageSize – Tamanho da página (valor predefinido).

O tamanho da página tem como valor predefinido 100, o que significa que mostra apenas 100 resultados. Ao mudar de página é necessário efetuar um novo cálculo para obter os 100 novos valores da página seguinte. Este cálculo é designado pela seguinte fórmula:

$$RowSkips = (CurrentPageIndex - 1) \times PageSize$$

- RowSkips – Número a partir do qual devem ser buscados os dados.
- CurrentPageIndex – Número inicialmente definido com o valor 1, ao que depois é sempre incrementado mais 1 ou menos 1, consoante a navegação entre as páginas.

A identificação da página atual onde o utilizador se encontra é dada da seguinte forma:

$$CurrentPageIndex / TotalPages$$

Por exemplo, se o CurrentPageIndex é igual a 30 e o TotalPages é igual a 80, obtemos o seguinte:



Figura 48 - Exemplo Paginação

O que significa que o utilizador está, naquele momento, na página 30 de um total de 80 páginas.

## 4.8. Filtragem

Implementada com o propósito de auxiliar a pesquisa personalizada e de maneira a complementar o requisito número três, a filtragem possibilita ao utilizador a escolha de vários filtros, de forma a restringir os resultados a apresentar. É possível efetuar a filtração por qualquer campo que pertença à tabela inserida inicialmente, com a exceção do campo “MANDT”, pois este é selecionado pelo SAP no momento de login através do Open SQL, que efetua a seleção automática do cliente (“MANDT”). No entanto, é possível utilizar o campo “MANDT”, mas apenas se for especificado depois da cláusula “FROM” a palavra reservada “CLIENT SPECIFIED”. Isto significa que na cláusula “WHERE” é necessário identificar o número de cliente (“MANDT”).

```
SELECT MANDT, <Dados>, <Dados>
  INTO TABLE <Tabela Interna> FROM <Tabela>
  CLIENT SPECIFIED "MANDT foi selecionado,"
                  "é necessário CLIENT SPECIFIED"
 WHERE mandt = '100'.
```

Figura 49 - Query com uso de MANDT

Neste momento, estão apenas disponíveis cinco campos para efetuar a filtração de resultados; no entanto, é sempre possível proceder a uma alteração no código futuramente para permitir uma maior variedade. Na secção de anexos encontra-se disponível uma imagem, que que representa o formulário de filtragem.

## 4.9. Exportação de Dados

A exportação dos dados é vital para o bom funcionamento do extrator e complementa o requisito número quatro. Consiste na exportação dos dados apresentados na grelha de dados para quatro tipos de ficheiro nomeadamente **PDF**, **XML**, **CSV** e **HTML**. Existe sempre a possibilidade de estender a quantidade de opções de exportações disponíveis; porém, foram selecionadas apenas as referidas anteriormente, a pedido da empresa ODKAS e por serem os tipos de ficheiros mais utilizados atualmente.

### 4.9.1. iText Sharp (PDF)

A exportação dos dados para PDF é feita através do auxílio do **iText Sharp**, que se trata de um **software open source** muito utilizado para a conversão de dados para o formato **PDF**. Através do **iText Sharp** é possível personalizar o formato do **PDF**, sendo possível definir várias formatações de letra e tamanhos. É, de igual modo, viável a introdução de imagens no ficheiro como, por exemplo, logotipos de empresas. Passemos, então, à análise de um exemplo de implementação deste *software*:

Para utilizar o **iText Sharp** é necessário efetuar, em primeiro lugar, o *download* do *software* e fazer a sua instalação no *visual studio*, de forma a ficar disponível para todos os projetos a desenvolver.

Posto isto, é necessário criar uma tabela **iText Sharp** através da grelha de dados do extrator. A tabela é criada da seguinte forma:

```
//Creating iTextSharp Table from the DataTable data
BaseFont bf = BaseFont.CreateFont(BaseFont.TIMES_ROMAN, BaseFont.CP1252, BaseFont.EMBEDDED);
Font font = new Font(bf, 10);
PdfPTable pdfTable = new PdfPTable(myDataGridView.ColumnCount);
pdfTable.DefaultCell.Padding = 3;
pdfTable.WidthPercentage = 100;
pdfTable.HorizontalAlignment = Element.ALIGN_LEFT;
pdfTable.DefaultCell.BorderWidth = 1;
```

Figura 50 - Criação Tabela iTextSharp

Começamos por definir o tipo de letra a utilizar através da utilização do método **BaseFont.CreateFont()** - no exemplo supra apresentado, foi escolhido o Times News Roman; a fonte é incorporada dentro do PDF (**BaseFont.EMBEDDED**), pois pretende-se que este seja imprimido de uma forma profissional. Em seguida é preciso criar um novo objeto **Font**, utilizando o **BaseFont (Font(bf, 10))** - o valor 10 representa o tamanho da letra. Ao objeto **Font** é ainda possível adicionar o estilo da letra e a sua cor; caso não seja especificado, toma os valores por defeito: habitualmente cor de letra preta e estilo normal. A tabela é declarada através da instanciação do objeto **PdfPTable pdfTable = new PdfPTable(int columns)**, o qual recebe como parâmetro o número de colunas da tabela, para depois ser preenchida. É ainda possível definir o alinhamento, o tamanho da margem, a largura das células e o preenchimento das células da tabela. Após esta pequena configuração, torna-se exequível efetuar o preenchimento da tabela através de ciclos *foreach*. O primeiro ciclo (Figura 52) trata de preencher os cabeçalhos da tabela e o segundo e terceiro ciclos (Figura 53) estão encarregues de preencher as linhas da tabela [42].

```
//Adding Header row
foreach (DataGridViewColumn column in myDataGridView.Columns)
{
    PdfPCell cell = new PdfPCell(new Phrase(column.HeaderText, font));
    cell.BackgroundColor = new BaseColor(240, 240, 240);
    pdfTable.AddCell(cell);
}
```

Figura 51 - Adição de cabeçalhos

```
//Adding DataRow
foreach (DataGridViewRow row in myDataGridView.Rows)
{
    foreach (DataGridViewCell cell in row.Cells)
    {
        pdfTable.AddCell(new Phrase(cell.Value.ToString(), font));
    }
}
```

Figura 52 - Adição de linhas

Para finalizar, é essencial guardar o ficheiro para que possa ser utilizado mais tarde. A implementação do processo de gravação é simples, como indicado na Figura 53.

```
FileStream stream = new FileStream(tabname + ".pdf", FileMode.Create);
Document pdfDoc = new Document(PageSize.A2, 40f, 40f, 40f, 0f);
PdfWriter.GetInstance(pdfDoc, stream);

pdfDoc.Open();
pdfDoc.Add(pdfTable);
pdfDoc.Close();
stream.Close();
FicheiroGuardadoComSucesso("PDF");
```

Figura 53 - Gravação PDF

Procede-se, em primeiro lugar, à criação do ficheiro através do objeto **FileStream**, onde é definido o caminho e o nome do ficheiro a guardar, é depois necessário criar o documento através do objeto **Document**, dentro do qual é especificado o tamanho da página (A1, A2, A3...) e as medidas das suas margens. Para gerar o ficheiro PDF é necessária uma instância do **PdfWriter**, concebida através da utilização do método **PdfWriter.GetInstance(pdfDoc, stream)**, onde **pdfDoc** representa o documento a escrever e **stream** o local para onde este deve ser escrito. Para finalizar o processo, basta abrir o documento **pdfDoc**, adicionar a **pdfTable** ao documento e fechar o mesmo e o seu local de criação [43].

### 4.9.2. XML

De modo a procedermos à exportação dos dados para o formato **XML** foi utilizada a tecnologia já disponível no .Net, por ser uma alternativa fácil de utilizar e sem custos adicionais. É necessário criar um **DataSet** [44] (conjunto de dados) a partir da grelha de dados do extrator, para seguidamente utilizar esse conjunto de dados com o **XMLwriter** [45], que se trata de um método disponibilizado na .Net Framework que escreve os dados contidos atualmente no conjunto de dados para um ficheiro **XML**. A Figura 54 demonstra a implementação deste método: “**ds**” representa o **DataSet** e **WriteXML** é o método utilizado para a conversão.

```
ds.WriteXml(tabname + ".xml");
```

Figura 54 - Escrita para ficheiro XML

### 4.9.3. CSV

Para efetuar a exportação de dados para o formato **CSV**, foi implementado um método que trata de efetuar a conversão. Uma vez que se trata de uma implementação simples, não houve necessidade de utilizar uma biblioteca externa para o efeito. Ao método utilizado foi atribuído a designação **ExportDataToCSV**, no qual a separação dos valores da grelha de dados é efetuada através de vírgulas. É preciso criar uma tabela de dados (**DataTable**) a partir da fonte de dados da grelha de dados (**dgv.DataSource**) e um **StringBuilder** onde são guardados os resultados.

```
DataTable dt = (DataTable)(dgv.DataSource);  
StringBuilder strCSV = new StringBuilder();
```

Figura 55 - DataTable e StringBuilder

Para iterar sobre os valores é necessário utilizar o ciclo **foreach**. Os ciclos devem ser utilizados separadamente, pois é importante usar um ciclo para percorrer as colunas (Figura 56) e um outro para percorrer as linhas da tabela (Figura 57).

```
foreach (DataColumn dgv_column in dt.Columns)  
{  
    strCSV.Append(dgv_column.ColumnName.ToString() + ", ");  
}
```

Figura 56 - Ciclo para percorrer colunas

```
foreach (DataRow dgv_row in dt.Rows)
{
    foreach (DataColumn dgv_column in dt.Columns)
    {
        strCSV.Append(dgv_row[dgv_column.ColumnName].ToString() + ", ");
    }
}
```

Figura 57 - Ciclo para percorrer Linhas da tabela

Na Figura 56 é demonstrado que se pretende obter apenas os valores dos cabeçalhos, pelo que é apenas necessário efetuar uma simples iteração por todos os cabeçalhos das colunas da tabela, e efetuar um **Append** para cada um dos seus valores a cada iteração, atribuindo os seus valores à variável **strCSV**.

Na Figura 57, porém, verificamos que o esquema é um pouco mais complexo, pois é necessário utilizar as linhas da tabela em conjunto com as suas colunas, de forma a obter o valor da coluna da linha pretendida. Depois é necessário efetuar o seu **Append** à variável **strCSV**. Concluídas estas conversões, é possível efetuar a gravação do ficheiro através da seguinte linha de código:

```
File.WriteAllText(tabname + ".csv", contentsCSV);
```

Figura 58 - Gravar ficheiro CSV

#### 4.9.4. HTML

Para exportar dados para o formato **HTML** foi implementado um método que trata de efetuar o tratamento dos dados. Como é também de fácil implementação não houve necessidade de utilizar uma biblioteca externa que efetue a conversão. A este método foi atribuído o nome de **ExportDataToHTML**, cujo objetivo é a apresentação dos dados no formato de uma tabela HTML. O método foi implementado da seguinte forma:

```
DataTable dt = (DataTable)(dgv.DataSource);
StringBuilder strHTML = new StringBuilder();
strHTML.Append("<html>" + Environment.NewLine);
strHTML.Append("<head>");
strHTML.Append("</head>" + Environment.NewLine);
strHTML.Append("<body>" + Environment.NewLine);
//Start the table.
strHTML.Append("<table border='1px' cellpadding='1'");
//Start first table line with headers.
strHTML.Append("<tr>");
```

Figura 59 – Inicializações



Como indicado pela Figura 59, é necessário efetuar uma declaração de uma tabela de dados (**DataTable**), a partir da fonte de dados da grelha de dados (**dgv.DataSource**), a declaração de um **StringBuilder**, e a abertura do ficheiro HTML consoante a sua estrutura de etiquetas. Seguidamente, é declarada a tabela através da etiqueta “<table> </table>”. Para efetuar o preenchimento da tabela para HTML foram utilizados ciclos *foreach*, implementados da seguinte forma:

```
foreach (DataColumn dgv_column in dt.Columns)
{
    //Table Headers
    strHTML.Append("<th>");
    strHTML.Append(dgv_column.ColumnName);
    strHTML.Append("</th>");
}
// Close first table line.
strHTML.Append("</tr>" + Environment.NewLine);
```

Figura 60 - HTML preenchimento de colunas

Uma vez que se pretende apenas os nomes das colunas, nomeadamente os seus cabeçalhos, basta utilizar um ciclo *foreach*; após a busca de todos os valores é efetuado um **Append** à variável **strHTML**, e esta passa a conter todos os cabeçalhos da tabela.

No passo seguinte é necessário obter os valores das linhas que correspondem às colunas da tabela; para isso são utilizados dois ciclos *foreach*, cuja implementação se encontra na Figura 62.

```
foreach (DataRow dgv_row in dt.Rows)
{
    //Open line of rows
    strHTML.Append("<tr>");
    foreach (DataColumn dgv_columns in dt.Columns)
    {
        //Table Rows
        strHTML.Append("<td>");
        strHTML.Append(dgv_row[dgv_columns.ColumnName].ToString());
        strHTML.Append("</td>");
    }
    //Close line of rows
    strHTML.Append("</tr>" + Environment.NewLine);
}
```

Figura 61 – Preenchimento de colunas HTML

É importante referir o significado das etiquetas utilizadas:

- `<tr></tr>` - Representa uma linha da tabela
- `<th></th>` - Representa os cabeçalhos da tabela
- `<td></td>` - Representa as células de cada linha da tabela

Para finalizar, é necessário fechar todas as etiquetas HTML, e efetuar a gravação do ficheiro, este processo encontra-se na Figura 62.

```
// Close remaining tags
strHTML.Append("</table" + Environment.NewLine);
strHTML.Append("</body>" + Environment.NewLine);
strHTML.Append("</html>");

string contentsHTML = strHTML.ToString();

try
{
    File.WriteAllText(tabname + ".html", contentsHTML);
    FicheiroGuardadoComSucesso("HTML");
}
catch (Exception)
{
    FalhaGravFicheiro("HTML");
}
```

Figura 62 - Fecho de etiquetas e gravação do ficheiro HTML

## 4.10. Desenvolvimento - Interface

Foram desenvolvidos dois Windows Forms, um para tratar do *login* do utilizador (LogIn Form), o qual foi desenvolvido para facilitar a interação com o SAP CONNECTOR 3.0, permitindo ao utilizador a inserção de toda a informação relevante ao servidor a que se pretende aceder, e os dados do utilizador que são utilizados para autenticação de quem utiliza o extrator (este formulário encontra-se na Figura 63. Após efetuar o *login* com sucesso, é necessário um outro formulário para tratar da parte de apresentação de dados ao utilizador (Main Form, Figura 64), mediante o qual o utilizador pode efetuar a pesquisa de uma tabela através da introdução do seu nome e, em seguida, pode selecionar os campos que lhe são relevantes e adicionar filtros, caso seja necessário. Em seguida, executa esta configuração e obtém os dados na grelha de dados. Se existir mais do que uma página de dados, o utilizador tem a possibilidade de navegar entre as páginas à procura da informação relevante, e efetuar a exportação desses dados para quatro tipos de ficheiros, caso necessite. Ambos os formulários foram desenvolvidos com a intenção de poderem ser utilizados de forma simples, clara e assertiva por parte do utilizador.

### 4.10.1. Login Form

A figura 64, representa o formulário disponibilizado ao utilizador para efetuar o seu login na ferramenta de exportação de dados.

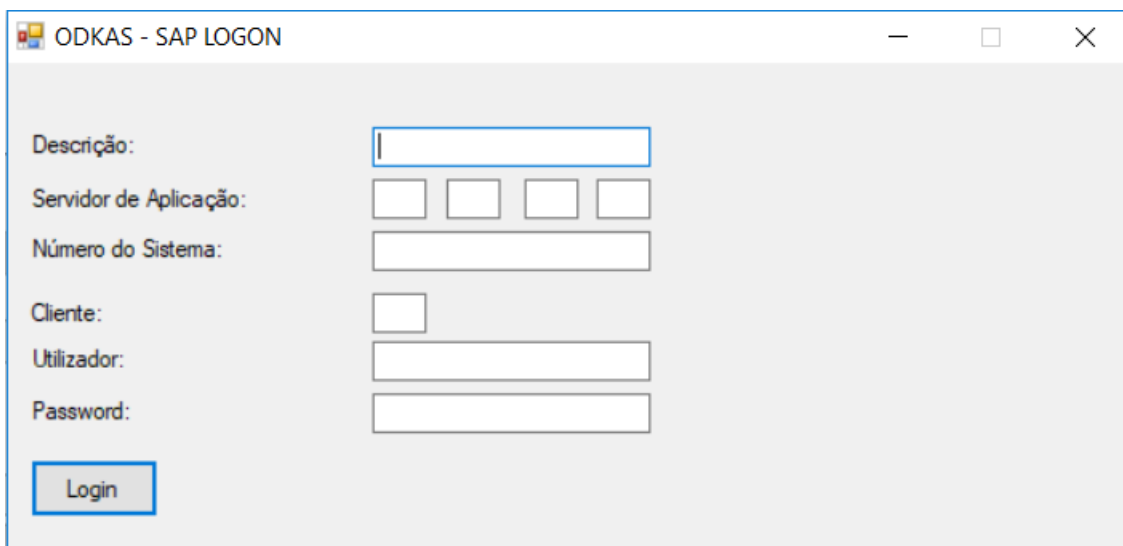


Figura 63 - Form Login

### 4.10.2. MainForm

A implementação deste formulário tem como objetivo a complementação do requisito número um, no qual se define que a informação seja mostrada ao utilizador através do uso de uma grelha de dados. É possível visualizar, de igual modo, a área que diz respeito ao requisito número dois da seleção de campos.

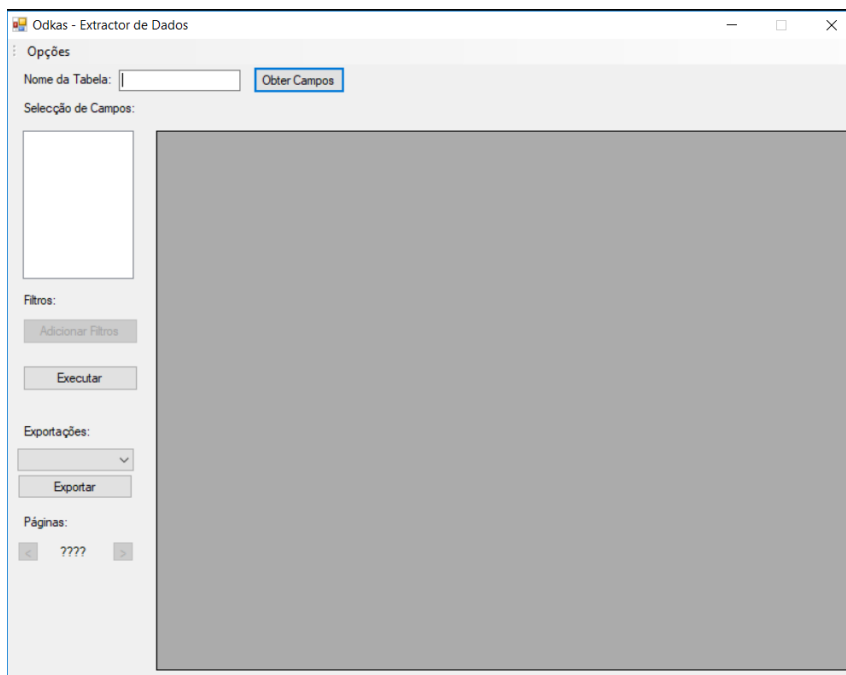


Figura 64 - MainForm

## 5. Resultados

No decurso deste capítulo são, efetuados alguns testes de desempenho, os quais permitem identificar: o tempo necessário para efetuar um *login*, o tempo necessário para efetuar a primeira extração dos campos de uma tabela, o tempo necessário para efetuar a pesquisa dos vários dados disponíveis na base de dados, através da utilização de filtros ou não e, finalmente, o tempo necessário para efetuar uma exportação de dados para um dos tipos de ficheiro disponíveis, de forma a identificar qual o mais rápido.

Devido à falta de acesso ao módulo de extração da SAP, não foi possível efetuar uma comparação entre os dois sistemas, e nem era esse um dos objetivos. O leque de funcionalidades disponibilizadas pela SAP ultrapassa significativamente aquelas que são disponibilizadas pelo extrator de dados, pelo que se espera que este seja utilizado em situações mais diretas, nomeadamente extrações direcionadas para apenas uma tabela, enquanto que a ferramenta da SAP permite efetuar filtragens, integrações e seleções a múltiplas tabelas ao mesmo tempo, pelo que é utilizada para casos que apresentam uma maior complexidade.

## 5.1 Testes de Desempenho

Os testes de desempenho foram efetuados através do uso da classe Stopwatch [46], disponível para o ambiente .Net e é muito utilizada para efetuar testes de desempenho a vários métodos tais como *logins*, pesquisa e tratamentos de dados, etc. Para melhor aferirmos o desempenho dos métodos implementados no extrator, foram efetuados cinco testes para cada um dos métodos mais relevantes. Após estes procedimentos, foi calculada uma média de todos os valores obtidos de forma a apurar um valor mediano, que representa o tempo necessário para a execução dos métodos mais relevantes. Todos os testes sem filtragem foram efetuados através da seleção de cinco campos da tabela “DD02L”. Para os testes com filtragem foram utilizados cinco campos da tabela “DD02L” e dois filtros. “DD02L” [47] trata-se de uma tabela SAP utilizada para armazenar informações de outras tabelas SAP, nomeadamente dados como o seu nome, o seu número máximo e mínimo de registos, as suas ligações, etc. Importa assinalar que estes testes são muito influenciados pela velocidade da internet disponível, bem como pela quantidade de registos que são exportados.

### 5.1.1. Medição - Tempo de Login

Neste primeiro teste pretende-se averiguar o tempo necessário para efetuar o processo de *login*. Foram efetuados cinco testes de *login*, cujos resultados podem ser visualizados pela Tabela 4.

Teste #	1	2	3	4	5
Resultado (ms)	433	433	439	421	428
	421	428	433	433	439
Mediana (s)			433		
Média (s)	$\frac{433 + 433 + 439 + 421 + 428}{5} = 430,8 \cong 0.4308$				
Desvio Padrão (s)	0,006723				

Tabela 4 - Tempos de Login

### 5.1.2. Medição - Tempo de Obtenção de Campos

O segundo teste tem como objetivo o cálculo do tempo médio necessário para efetuar a pesquisa de todos os campos de uma determinada tabela. Os tempos obtidos encontram-se representados na Tabela 5.

Teste #	1	2	3	4	5
Resultado (ms)	753	560	558	567	574
	558	560	567	574	753
Mediana (s)			567		
Média (s)	$\frac{753 + 560 + 558 + 567 + 574}{5} = 602,4 \cong 0.6024$				
Desvio Padrão (s)	0,084423				

Tabela 5 - Obtenção dos campos de uma Tabela SAP

### 5.1.3. Medição - Tempo de Obtenção de Dados (Sem Filtragem)

O terceiro teste visa o apuramento do tempo médio necessário para obter o conteúdo da tabela inserida no momento da pesquisa sem filtros. Os tempos obtidos podem ser visualizados na Tabela 6.

Teste #	1	2	3	4	5
Nº Registos	500	1000	2000	3000	4000
Resultado (ms)	556	631	734	880	998
Mediana (ms)	556	631	734	880	998
Média (s)	$\frac{556 + 631 + 734 + 880 + 998}{5} = 760 \cong 0.760$				
Desvio Padrão (s)	0,180261				

Tabela 6 - Obtenção do conteúdo de uma tabela SAP (Sem Filtragem)

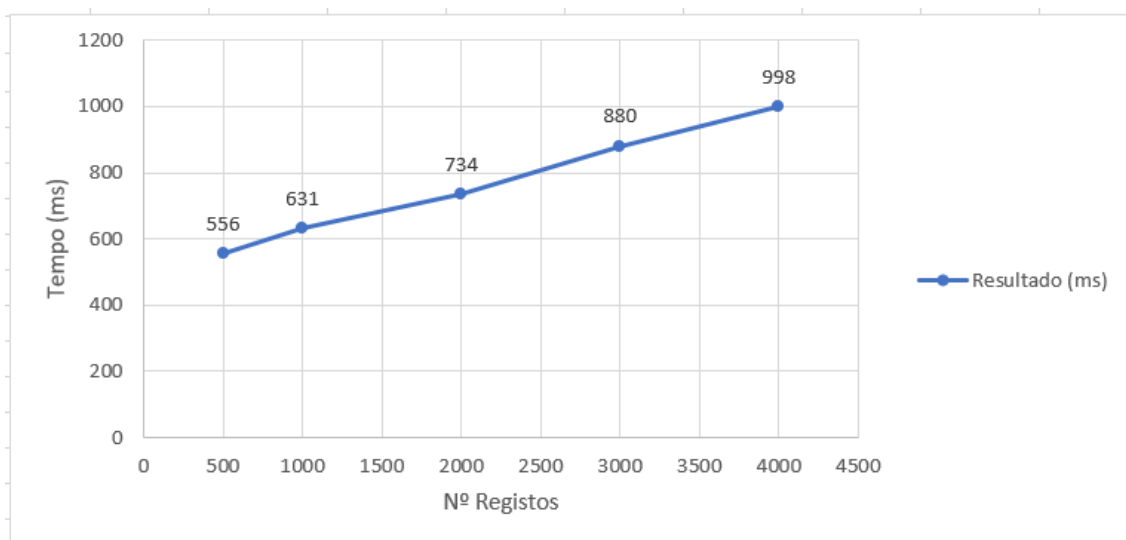


Gráfico 9 - Tempo de pesquisa sem filtragem

Uma análise do Gráfico 9 e da Tabela 6 permite-nos verificar que o aumento do número de registos a obter na pesquisa dos valores da tabela selecionada durante o tempo de execução (*runtime*), é um fator que tem um impacto significativo no tempo necessário à obtenção dos



dados. De facto, quanto maior for o volume dos registos a pesquisar, maior é o tempo necessário para finalizar a pesquisa.

### 5.1.4. Medição - Tempo de Obtenção de Dados (Com Filtragem)

O quarto teste tem com objetivo a obtenção do tempo médio necessário para obter o conteúdo da tabela inserida inicialmente com filtragem de valores. Os tempos obtidos estão presentes na Tabela 7.

Teste #	1	2	3	4	5
Nº Registos	500	1000	2000	3000	4000
Resultado (ms)	440	445	608	731	832
Mediana (s)	440	445	608	731	832
Média (s)	$\frac{440 + 445 + 608 + 731 + 832}{5} = 611,2 \approx 0,6112$				
Desvio Padrão (s)	0,173239				

Tabela 7 - Obtenção conteúdo de uma tabela SAP (Com Filtragem)

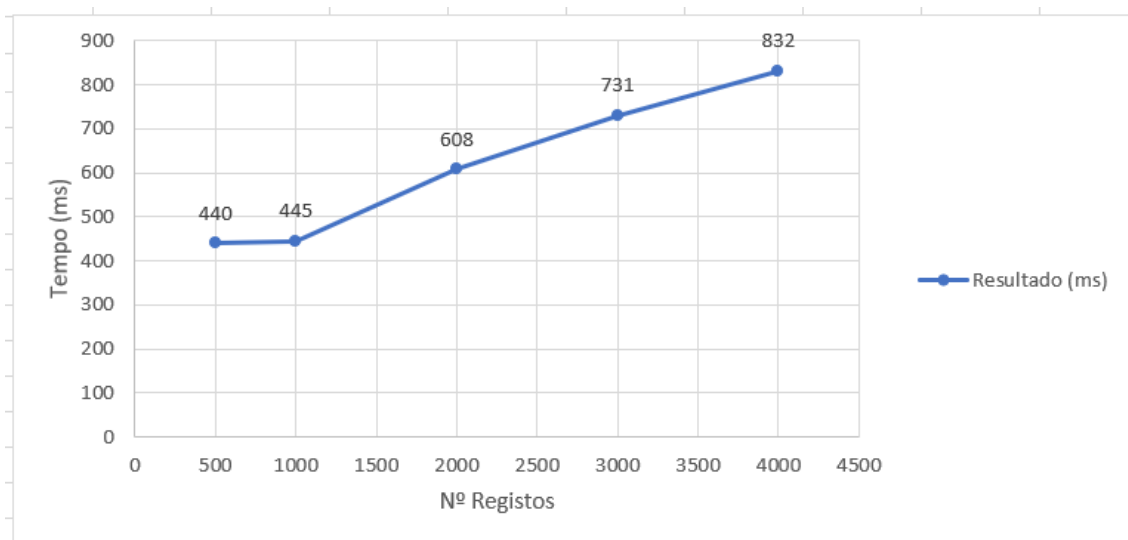


Gráfico 10 -Tempo de pesquisa com filtragem

Através da análise da Tabela 7 e do Gráfico 10 é possível verificar que, tal como na pesquisa sem filtragem, o número de registos influencia o tempo de pesquisa; porém, devido ao facto de serem utilizados filtros, torna-se possível reduzir ainda mais o tempo de pesquisa de informação.

Observemos o Gráfico 11:

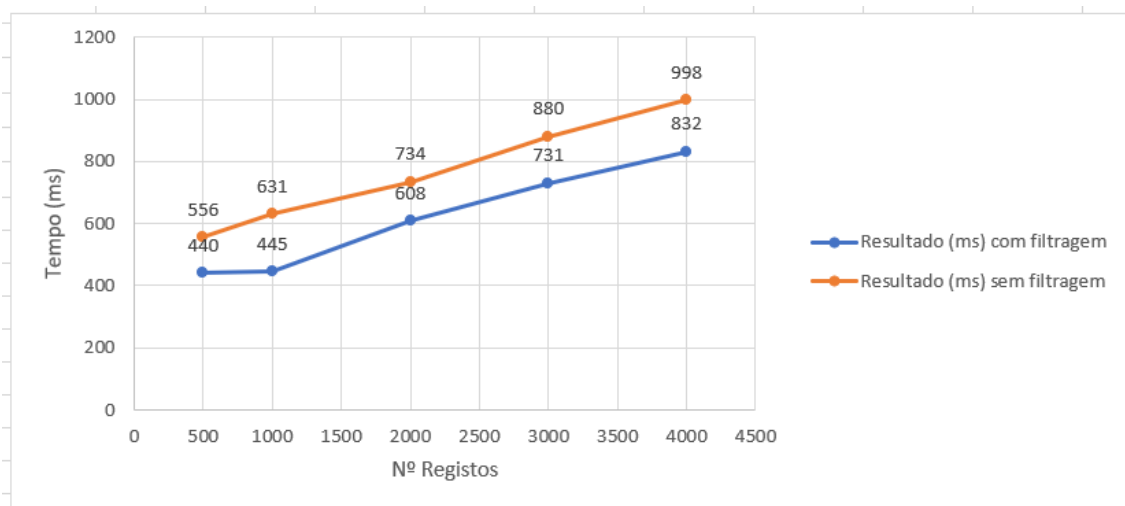


Gráfico 11 - Comparação das Pesquisas

Este gráfico permite-nos aferir, de uma forma mais clara, a diferença entre os tempos de pesquisa. Graças à análise do Gráfico 11, foi possível elaborar a Tabela 8, na qual podemos averiguar qual a diferença média entre os tempos de pesquisa.

<b>Pesquisa sem filtragem (ms)</b>	556	631	734	880	998
<b>Pesquisa com filtragem (ms)</b>	440	445	608	731	832
<b>Diferença de Tempo (ms)</b>	116	186	126	149	166
<b>Média (ms)</b>	148,6				
<b>Desvio Padrão (ms)</b>	28,615				

Tabela 8 - Diferença entre tempos de pesquisa

Como se pode verificar pela análise do Gráfico 11 e dos valores médios obtidos na Tabela 8, é possível chegar à conclusão de que a pesquisa com filtragem de valores é em média 148.6 ms mais rápida que a pesquisa sem filtragem.

### 5.1.5. Medição - Exportação de Dados PDF

O sétimo teste visa, por seu turno, a medição do tempo médio necessário para efetuar a exportação do conteúdo da tabela para PDF.

Teste #	1	2	3	4	5
<b>Nº de Registos</b>	500	1000	2000	3000	4000
<b>Resultado (ms)</b>	394	444	588	688	793
<b>Mediana (s)</b>	394	444	588	688	793
			588		
<b>Média (s)</b>	$\frac{394 + 444 + 588 + 688 + 793}{5} = 581,4 \cong 0,5814$				
<b>Desvio Padrão (s)</b>	0,165966				

Tabela 9 - Tempos Exportação PDF

### 5.1.6. Medição - Exportação de Dados CSV

O oitavo teste tem como objetivo a medição do tempo médio necessário para efetuar a exportação do conteúdo, da tabela selecionada inicialmente para o formato CSV.

Teste #	1	2	3	4	5
<b>Nº de Registos</b>	500	1000	2000	3000	4000
<b>Resultado (ms)</b>	1	1	2	3	3
<b>Mediana (s)</b>	1	1	2	3	3
			2		
<b>Média (s)</b>	$\frac{1 + 1 + 2 + 3 + 3}{5} = 2 \cong 0,002$				
<b>Desvio Padrão (s)</b>	0,001				

Tabela 10 - Tempos Exportação CSV

### 5.1.7. Medição - Exportação de Dados XML

O nono teste visa, por sua vez, demonstrar a medição do tempo médio necessário para efetuar a exportação do conteúdo, da tabela selecionada inicialmente para o formato XML.

Teste #	1	2	3	4	5
Nº de Registos	500	1000	2000	3000	4000
Resultado (ms)	3	2	5	8	9
Mediana (s)	2	3	5	8	9
			5		
Média (s)	$\frac{2 + 3 + 5 + 8 + 9}{5} = 5,4 \cong 0.0054$				
Desvio Padrão (s)	0,00305				

Tabela 11 - Tempos Exportação XML

### 5.1.8. Medição - Exportação de Dados HTML

O intuito do décimo teste é a medição do tempo médio necessário para efetuar a exportação do conteúdo, da tabela selecionada inicialmente para o formato HTML.

Teste #	1	2	3	4	5
Nº de Registos	500	1000	2000	3000	4000
Resultado (ms)	1	1	2	3	4
Mediana (s)	1	1	2	3	4
			2		
Média (s)	$\frac{1 + 1 + 2 + 3 + 4}{5} = 2,2 \cong 0.0022$				
Desvio Padrão (s)	0,001304				

Tabela 12 - Tempos Exportação HTML

Através da comparação dos valores médios das exportações de conteúdo, que se encontram representados nas tabelas 10, 11, 12 e 13, é possível concluir que o método de extração mais rápido é o do formato CSV, seguido pelos formatos XML e HTML, sendo o formato mais lento o do PDF. Importa ter em conta que estas diferenças de tempo são causadas pela complexidade de cada método, sendo que o método utilizado para efetuar a extração para PDF utiliza um maior número de configurações, que são necessárias para a boa estruturação e utilização do documento. No caso do CSV, é apenas necessário separar os valores de cada registo da tabela por vírgulas (“,”), o que implica uma maior rapidez na transformação dos dados. No que diz respeito aos formatos HTML e XML, são ambos também muito rápidos, pois basta utilizar apenas as etiquetas necessárias para a boa estruturação do documento a apresentar, não sendo normalmente necessárias configurações adicionais.

## 6. Conclusão

- O objetivo A, que consistia na criação de um extrator de dados encontra-se concluído, pois é possível (tal como demonstrado ao longo deste trabalho) efetuar a extração de informação das tabelas SAP, no ambiente .net através de uma pesquisa pelo nome da tabela.
- O objetivo B, que objetivava a criação de RFCs no ambiente SAP para efetuar extração de informação, encontra-se plenamente implementado.
- O objetivo C, por sua vez, perspetivava a visualização dos dados extraídos, a qual se demonstrou estar implementada e possibilita a visualização de toda a informação da tabela selecionada no tempo de execução.
- O objetivo D, referente à paginação dos valores obtidos, encontra-se devidamente implementado, sendo possível navegar por várias páginas, de forma a visualizar a informação completa da tabela selecionada. Esta implementação traz grandes benefícios no que diz respeito ao desempenho do extrator, e facilita a visualização dos dados.
- O objetivo E está, de igual modo, implementado e permite a possibilidade de a seleção de filtros tais como a seleção dos campos mais relevantes, de forma a filtrar o número de resultados a apresentar.
- O objetivo F, relativo ao requisito número quatro, encontra-se implementado e viabiliza a extração dos resultados para quatro tipos de ficheiro; é, no entanto, possível aumentar este número limite, caso o utilizador assim necessite.
- O objetivo G, que compreendia simplificar o uso da aplicação para utilizadores que desconheçam a forma de como funciona o SAP, encontra-se parcialmente implementado, visto que a interface foi desenvolvida de uma forma simplificada, motivo pelo qual necessita alguns ajustes. Esta implementação refere-se aos requisitos um, dois, três e cinco.
- Em relação ao objetivo H, é possível afirmar que foram adquiridos conhecimentos relevantes, nomeadamente na criação de RFCs, na forma de como estes são utilizados para extrair informação dos servidores SAP, através do uso de uma *query* implementada em linguagem ABAP, como também conhecimentos importantes na passagem de dados entre duas plataformas distintas, através do uso do SAP Connector 3.0 disponibilizado pela própria SAP aos desenvolvedores ABAP.

A explanação comparativa no âmbito do capítulo três permitiu identificar algumas diferenças entre os sistemas utilizados para inspiração no desenvolvimento do extrator de dados. Estas diferenças consistem na identificação da falta de um painel de controlo personalizável, capaz de

realizar uma análise através de gráficos e de efetuar uma análise de probabilidades e estatística. A análise efetuada possibilitou, portanto, identificar a necessidade de o extrator estar disponível em mais do que uma língua, de forma a abranger um maior número de utilizadores. Este requisito, porém, pode de momento não ser de grande importância, visto que se o extrator for utilizado apenas para uso interno da empresa e para clientes portugueses, em princípio não deverá existir a necessidade de ser disponibilizado em mais línguas.

De modo a facilitar aos utilizadores a visualização de informação, verifica-se também a necessidade de o extrator suportar outras plataformas, tais como Android e MAC OS. E, por fim, no que diz respeito à integração de dados, foi também apurado que existe a necessidade de integrar dados de origem externa, como por exemplo a partir da DropBox, Facebook, Excel, Google Drive, ou de um outro qualquer *software* que seja utilizado para armazenar informação

Os estudos apresentados no capítulo cinco (Resultados) possibilitaram a averiguação da quantidade de tempo necessário para efetuar certas operações como, por exemplo, o tempo despendido para efetuar um *login*. Esta análise de tempos viabilizou, também, a aferição do tempo médio necessário para a obtenção dos campos de uma tabela, para depois serem utilizados na seleção dos valores a apresentar. É, ainda, possível averiguar o tempo necessário para a busca de dados com e sem filtragem da base de dados SAP, no decurso do qual apurámos que o método que faz uso da filtragem é o mais rápido, e que o tempo de pesquisa é influenciado significativamente pelo aumento do número de registos a obter de uma só vez. Para finalizar o capítulo cinco, expusemos o estudo da última medição de tempo, a qual consiste em averiguar qual é o tempo médio necessário para concretizar uma exportação para um determinado tipo de ficheiro. Foram analisados os tempos médios dos quatro tipos de ficheiro disponíveis atualmente no extrator de dados, e é possível concluir perentoriamente que o processo de exportação mais rápido é através de CSV; verificámos, de igual modo que, tal como na pesquisa de informação, a exportação dos dados sofre alterações no desempenho devido ao aumento dos registos a exportar.

## 6.1 Trabalho Futuro

Como trabalho futuro pretende-se retificar algumas partes da interface do formulário principal, de forma a ficar esteticamente mais agradável ao utilizador. Através da análise comparativa, foi possível identificar que existe uma grande vantagem no desenvolvimento de um painel de controlo personalizável, que será utilizado para a visualização de gráficos e análise de probabilidade e estatística, por parte do utilizador.

No que diz respeito à componente da pesquisa avançada, apurámos, por meio da análise comparativa entre sistemas, que a utilização de cinco filtros é suficiente para fornecer o utilizador com uma pesquisa eficiente e robusta; no entanto, deve existir a opção de o utilizador adicionar todos os filtros que considerar relevantes para a sua consulta.

Para o caso da paginação, que de momento se encontra limitada a 100 registos por página, deveria ser adicionado um campo no formulário principal, que permita efetuar a seleção do número de registos que podem ser visualizados. Um exemplo deste tipo de implementação seria criar uma escala a qual poderia evoluir de 100 em 100, com início em 100, tomando depois o valor de 200. Seria, também, necessário implementar um limite na escala de, por exemplo, 500, sendo o valor 100 o mínimo de registos e 500 o número máximo de registos a ser visualizado em cada página.

Concluimos que, apesar do extrator de dados possuir quatro tipos de formatos de exportação disponíveis, podem ser adicionados ainda mais alguns, no caso de existir tal necessidade, ou para o caso de aparecerem novos formatos mais eficientes. Visto que o sistema se encontra disponível em apenas uma língua (Português), pode existir a necessidade, por parte da empresa, de implementar novas línguas, de forma a abranger clientes internacionais.

Por fim, poderá ser considerada a possibilidade de expandir o extrator de dados para outras plataformas, nomeadamente MAC OS, Linux e Android.



## 7. Referências

- [1] Post, Frits H., Gregory M. Nielson, e Georges-Pierre Bonneau. *Data Visualization: The State of the Art*. Springer Science & Business Media, 2012.
- [2] Isco, Cleo. «How Does SAP Work? - Easy Explanation with An Example». *ERProof* (blog), 9 de Janeiro de 2015. <https://erproof.com/how-does-sap-work/>.
- [3] Souza, Denis. «What is SAP Landscape? | SAP Blogs». What is SAP Landscape. Acedido 24 de Janeiro de 2018. <https://blogs.sap.com/2012/06/14/what-is-sap-landscape/>.
- [4] SAP. «SE11 - ABAP Dictionary - Basis Corner - SCN Wiki». Acedido 25 de Janeiro de 2018. <https://wiki.scn.sap.com/wiki/display/Basis/SE11+-+ABAP+Dictionary>.
- [5] Rekha. «Difference between SE16 and SE16N Transaction Codes in SAP». *STechies*. Acedido 25 de Janeiro de 2018. <http://www.stechies.com/difference-between-transaction-codes-se16-se16n/>.
- [6] ITsiti. «SE37: Function Module Builder», 20 de Fevereiro de 2011. <http://itsiti.com/se37-function-module-builder>.
- [7] Srinivasamurthy, Ranganathan. «How to Make Use of SE37- Function Module & How to Find out the Table?» Acedido 25 de Janeiro de 2018. <https://archive.sap.com/discussions/thread/402887>.
- [8] «SE80: Object Navigator: TCODE of the day». Acedido 25 de Janeiro de 2018. [http://newtosap.aditech.info/2012/04/se80-object-navigator-tcode-of-day\\_18.html](http://newtosap.aditech.info/2012/04/se80-object-navigator-tcode-of-day_18.html).
- [9] SAP. «Screen Painter - ABAP Development - SCN Wiki». Acedido 25 de Janeiro de 2018. <https://wiki.scn.sap.com/wiki/display/ABAP/Screen+Painter>.
- [10] SAP. «Menu Painter | SAP NetWeaver 7.5 | SAP Help Portal». Acedido 25 de Janeiro de 2018. <https://help.sap.com/viewer/bd833c8355f34e96a6e83096b38bf192/7.5.9/en-US/d1801ce8454211d189710000e8322d00.html>.
- [11] Guinn, Justin. «Top Business Intelligence Tools - 2018 Reviews & Pricing». Acedido 4 de Fevereiro de 2018. <https://www.softwareadvice.com/bi/>.
- [12] Financesonline. «Dundas BI Reviews: Overview, Pricing, and Features». *Financesonline.com*. Acedido 4 de Fevereiro de 2018. <https://reviews.financesonline.com/p/dundas-bi/>.

[13] Dundas. «Dundas BI Product Features - Dundas Data Visualization». Acedido 4 de Fevereiro de 2018. <http://www.dundas.com/dundas-bi/features>.

[14] Domo. «Data Management – Data Analytics – Data Reporting | Domo». Acedido 4 de Fevereiro de 2018. <https://www.domo.com/>.

[15] Financesonline. «Domo Reviews: Overview, Pricing and Features». Financesonline.com. Acedido 4 de Fevereiro de 2018. <https://reviews.financesonline.com/p/domo/>.

[16] Domo. «Business Intelligence Data Analyzer | Domo». Acedido 4 de Fevereiro de 2018. <https://www.domo.com/product/analyzer>.

[17] Domo. «ETL Tools - Extract, Transform & Load Data Like A Pro | Domo». Acedido 4 de Fevereiro de 2018. <https://www.domo.com/product/magic>.

[18] Daley, Richard. «Home Page». Pentaho. Acedido 4 de Fevereiro de 2018. <http://www.pentaho.com/home>.

[19] Financesonline. «Pentaho Reviews: Overview, Pricing and Features». Financesonline.com. Acedido 4 de Fevereiro de 2018. <https://reviews.financesonline.com/p/pentaho/>.

[20] Daley, Richard. «Business Analytics». Pentaho. Acedido 4 de Fevereiro de 2018. <http://www.pentaho.com/product/business-analytics>.

[21] Microsoft. «Power BI | Ferramentas do BI de Visualização de Dados Interativa». Acedido 4 de Fevereiro de 2018. <https://powerbi.microsoft.com/pt-br/>.

[22] Mihart. «Exemplo de Análise de Varejo para o Power BI: faça um tour - Power BI». Acedido 4 de Fevereiro de 2018. <https://docs.microsoft.com/pt-br/power-bi/sample-retail-analysis>.

[23] SAP. «Business Intelligence (BI) Tools & Software». SAP. Acedido 4 de Fevereiro de 2018. <https://www.sap.com/portugal/products/analytics/business-intelligence-bi.html>.

[24] Akhtar, Jawad. «What Core Components Does the SAP ERP System Include?» SearchSAP. Acedido 15 de Abril de 2018. <https://searchsap.techtarget.com/answer/What-core-components-does-the-SAP-ERP-system-include>.

[25] SAP. «SAP Industry Solutions». Softtek. Acedido 16 de Abril de 2018. <http://www.softtek.com/technologies/sap-industry-solutions>.

[26] mariana.mihaylova, e SAP. «An Introduction to SAP Business Workflow». Acedido 16 de Abril de 2018. <https://archive.sap.com/documents/docs/DOC-31056>.

[27] sarkarinaukriblog. «SAP Treasury management». Educação, 11:26:36 UTC.

<https://pt.slideshare.net/sarkarinaukriblog/sap-treasury-management>.

[28] «How Does SAP Pool Asset Management Improve the Use of Shared Assets?» SearchSAP.

Acedido 16 de Abril de 2018. [https://searchsap.techtarget.com/answer/How-does-SAP-Pool-](https://searchsap.techtarget.com/answer/How-does-SAP-Pool-Asset-Management-improve-the-use-of-shared-assets)

[Asset-Management-improve-the-use-of-shared-assets](https://searchsap.techtarget.com/answer/How-does-SAP-Pool-Asset-Management-improve-the-use-of-shared-assets).

[29] Financesonline. «SAP BusinessObjects Lumira Reviews:Overview, Pricing and Features».

Financesonline.com. Acedido 4 de Fevereiro de 2018.

<https://reviews.financesonline.com/p/sap-businessobjects-lumira/>.

[30] Sapoval, Ion. «SAP – .NET Connector 3.0 – Remote Function Call – RFC – C# |

Desenvolvimento Código Aberto». Html. Desenvolvimento Código Aberto. Acedido 22 de

Janeiro de 2018. [https://desenvolvimentoaberto.org/2014/12/24/sap-net-connector-3-0-](https://desenvolvimentoaberto.org/2014/12/24/sap-net-connector-3-0-remote-function-call-rfc-c/)

[remote-function-call-rfc-c/](https://desenvolvimentoaberto.org/2014/12/24/sap-net-connector-3-0-remote-function-call-rfc-c/).

[31] Wilstroth, William. «Difference b/w Name Beginng with y or Z». Acedido 23 de Janeiro de

2018. <https://archive.sap.com/discussions/thread/284517>.

[32] «SAP Library - Connectivity». Acedido 20 de Abril de 2018.

[https://help.sap.com/saphelp\\_tm81/helpdata/en/5b/617bcd479b46a1829ac0d7886903fb/fra meset.htm](https://help.sap.com/saphelp_tm81/helpdata/en/5b/617bcd479b46a1829ac0d7886903fb/fra meset.htm).

[33] AG, SAP. «The Repository». Text. Acedido 20 de Abril de 2018.

[/doc/saphelp\\_crm700\\_ehp02/7.0.2.17/en-US/0f/8635d6362c4123a37d39b2c8e652b5/content.htm](/doc/saphelp_crm700_ehp02/7.0.2.17/en-US/0f/8635d6362c4123a37d39b2c8e652b5/content.htm).

[34] AG, SAP. «SAP .NET Connector Architecture». Text. Acedido 20 de Abril de 2018.

[/doc/saphelp\\_crm700\\_ehp02/7.0.2.17/en-US/4a/097ee1449836eae10000000a421937/content.htm](/doc/saphelp_crm700_ehp02/7.0.2.17/en-US/4a/097ee1449836eae10000000a421937/content.htm).

[35] Saha, Suhas. «How to Get Field Name Dynamically of a Table». Acedido 22 de Janeiro de

2018. <https://archive.sap.com/discussions/thread/1735929>.

[36] Johnson, Felix. «Dynamic Select, Internal Tables and Work Areas for Beginners | SAP Blogs».

Acedido 22 de Janeiro de 2018. [https://blogs.sap.com/2012/12/06/dynamic-select-internal-](https://blogs.sap.com/2012/12/06/dynamic-select-internal-tables-and-work-areas-for-beginners/)

[tables-and-work-areas-for-beginners/](https://blogs.sap.com/2012/12/06/dynamic-select-internal-tables-and-work-areas-for-beginners/).

[37] SAP. «SAP Library - ABAP Programming (BC-ABA)». Html. Open SQL. Acedido 24 de Janeiro

de 2018.

[https://help.sap.com/saphelp\\_nw70ehp1/helpdata/en/fc/eb3969358411d1829f0000e829fbfe/frameset.htm](https://help.sap.com/saphelp_nw70ehp1/helpdata/en/fc/eb3969358411d1829f0000e829fbfe/frameset.htm).

[38] Microsoft. «DataGridView Class (System.Windows.Forms)». Acedido 28 de Janeiro de 2018.

[https://msdn.microsoft.com/en-us/library/system.windows.forms.datagridview\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/system.windows.forms.datagridview(v=vs.110).aspx).

[39] Microsoft. «Regex Class (System.Text.RegularExpressions)». Acedido 28 de Janeiro de 2018.

[https://msdn.microsoft.com/en-us/library/system.text.regularexpressions.regex\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/system.text.regularexpressions.regex(v=vs.110).aspx).

[40] Microsoft. «String.Split Method (System)». Acedido 28 de Janeiro de 2018.

[https://msdn.microsoft.com/en-us/library/system.string.split\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/system.string.split(v=vs.110).aspx).

[41] Microsoft. «Int32.Parse Method (String) (System)». Acedido 28 de Janeiro de 2018.

[https://msdn.microsoft.com/en-us/library/b3h1hf19\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/b3h1hf19(v=vs.110).aspx).

[42] Brind, Mike. «iTextSharp - Working with Fonts». Acedido 1 de Fevereiro de 2018.

<https://www.mikesdotnetting.com/article/81/itextsharp-working-with-fonts>.

[43] iTextSharp. «PdfWriter (iText 5.5.12 API)». Acedido 1 de Fevereiro de 2018.

<http://itextsupport.com/apidocs/itext5/latest/>.

[44] Microsoft. «DataSet Class (System.Data)». Acedido 31 de Janeiro de 2018.

[https://msdn.microsoft.com/en-us/library/system.data.dataset\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/system.data.dataset(v=vs.110).aspx).

[45] Microsoft. «DataSet.WriteXml Method (Stream) (System.Data)». Acedido 31 de Janeiro de

2018. [https://msdn.microsoft.com/en-us/library/sa57x5f2\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/sa57x5f2(v=vs.110).aspx).

[46] Microsoft. «Stopwatch Class (System.Diagnostics)». Acedido 11 de Fevereiro de 2018.

[https://msdn.microsoft.com/en-us/library/system.diagnostics.stopwatch\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/system.diagnostics.stopwatch(v=vs.110).aspx).

[47] se80. «DD02L SAP SAP Tables Table - ABAP». Acedido 20 de Fevereiro de 2018.

<http://www.se80.co.uk/saptables/d/dd02/dd02l.htm>.

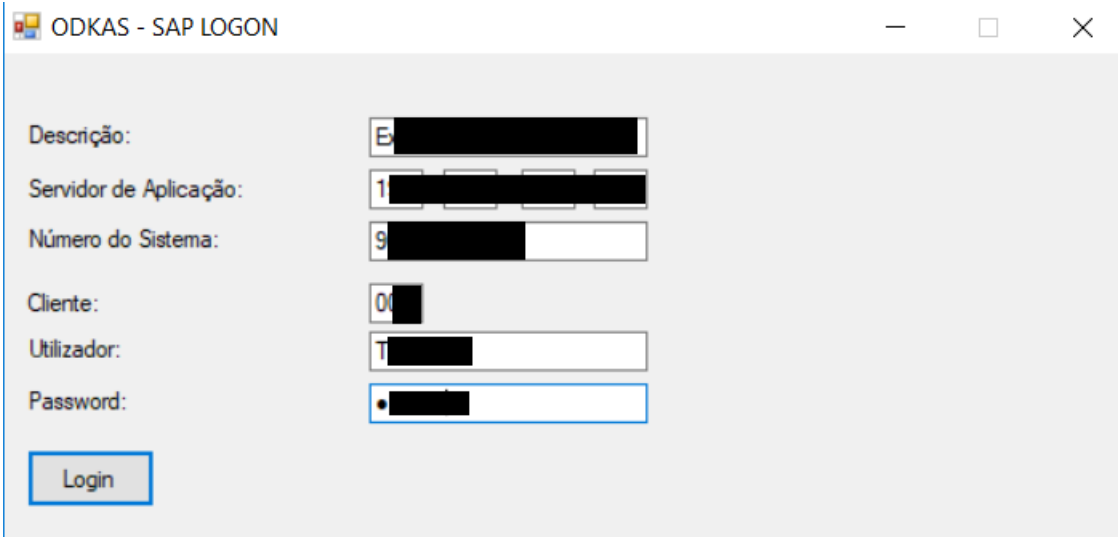
## 8. Anexos

### 8.1. Utilização do Extrator de Dados

A partir da secção 8.1.1 até 8.1.2.5 deste capítulo mostramos um pequeno guia de utilização do extrator, no decurso do qual está indicado como deve ser efetuado um *login* no extrator de dados. Após o *login*, é demonstrada a forma como deve ser utilizado o formulário principal, de forma a introduzir o nome da tabela pretendida, e assim obter os campos da mesma, bem como efetuar a seleção dos campos que o utilizador pretende visualizar. Após esta seleção, basta efetuar a pesquisa através da utilização de filtros (ou não, mediante as necessidades) e, finalmente, utilizar a paginação correta, que direcione à página pretendida e, por fim, efetuar a extração desses mesmos dados, utilizando um dos métodos disponíveis para a sua exportação.

### 8.1.1. Login

Para começar a utilizar o extrator de dados, é necessário efetuar o Login na aplicação através do seguinte formulário:



The image shows a Windows-style dialog box titled "ODKAS - SAP LOGON". It contains several input fields for login information: "Descrição:" (Description), "Servidor de Aplicação:" (Application Server), "Número do Sistema:" (System Number), "Cliente:" (Client), "Utilizador:" (User), and "Password:". Each field contains some characters followed by a black redaction box. A "Login" button is located at the bottom left of the dialog.

Figura 65 - Formulário de Login

Na , são inseridos os dados do servidor e do utilizador. Após a inserção dos dados é efetuada a tentativa de *login*. Este *login* é gerido pelo SAP, pelo que o processo de autenticação é efetuado pelo próprio, dependendo do extrator apenas para a introdução das credenciais do utilizador e identificação do servidor. Se esta for bem-sucedida, a mensagem que se encontra na é mostrada ao utilizador.

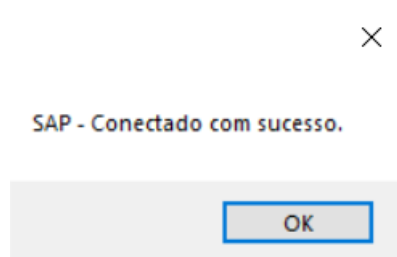


Figura 66 - Conexão com Sucesso

## 8.1.2. Formulário Principal

### 8.1.2.1. Pesquisa para obtenção de campos

Ao autenticar-se com sucesso, o utilizador pode agora utilizar as funcionalidades do formulário principal (Figura 64). Pretende-se pesquisar pela tabela “DD02L”, esta tabela contém um grande número de entradas, pois trata-se de uma das tabelas principais do servidor SAP, na qual está contida informação importante sobre todas as outras tabelas disponíveis no sistema. Para efetuar a pesquisa desta tabela, basta inserir “DD02L” no campo “Nome da Tabela” que se encontra identificado pela cor vermelho, e em seguida é necessário clicar sobre o botão “Obter Campos”, o qual encontra-se identificado pela cor laranja (Figura 67).

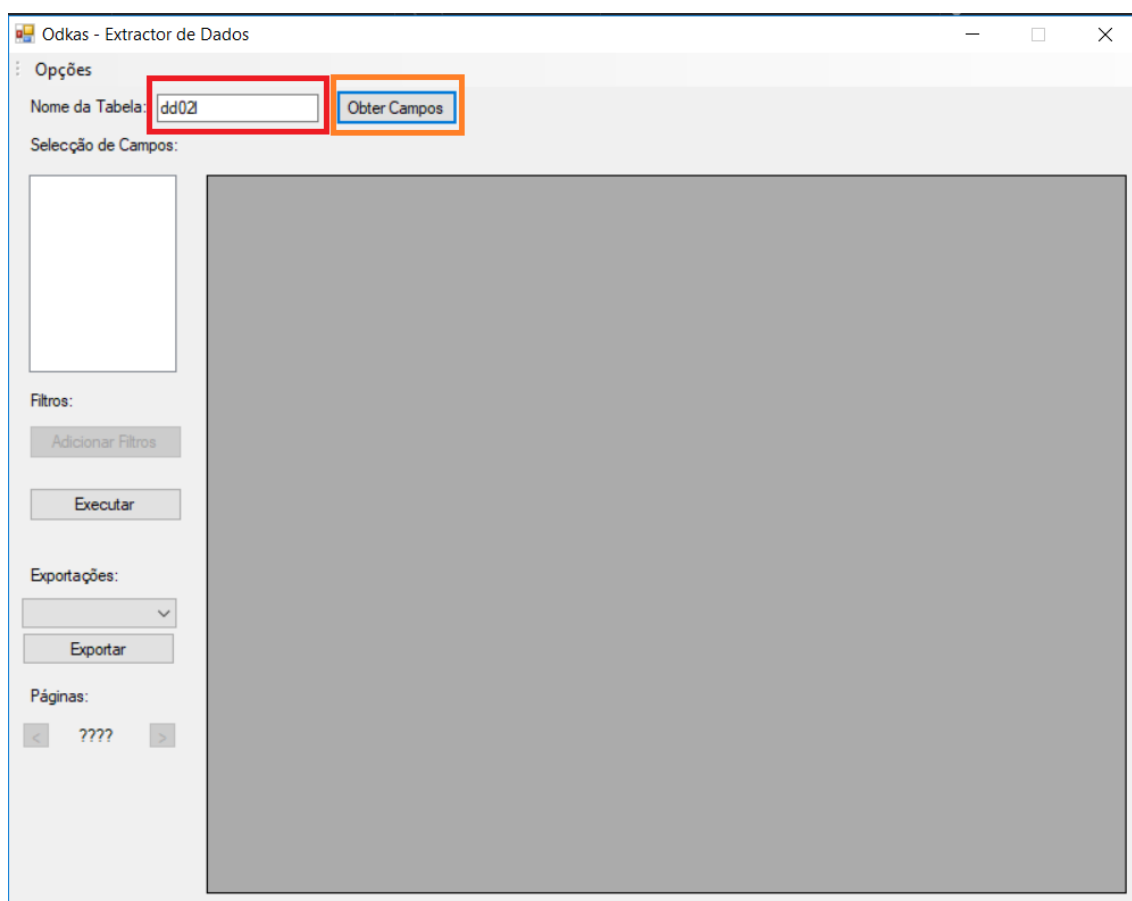


Figura 67 - Inserção de tabela e obtenção de campos

Após o clique no botão “Obter Campos”, todos os campos referentes à tabela “DD02L” podem ser visualizados na “Seleção de Campos” - identificado pela cor azul, na Figura 68.

### 8.1.2.2. Seleção de Campos

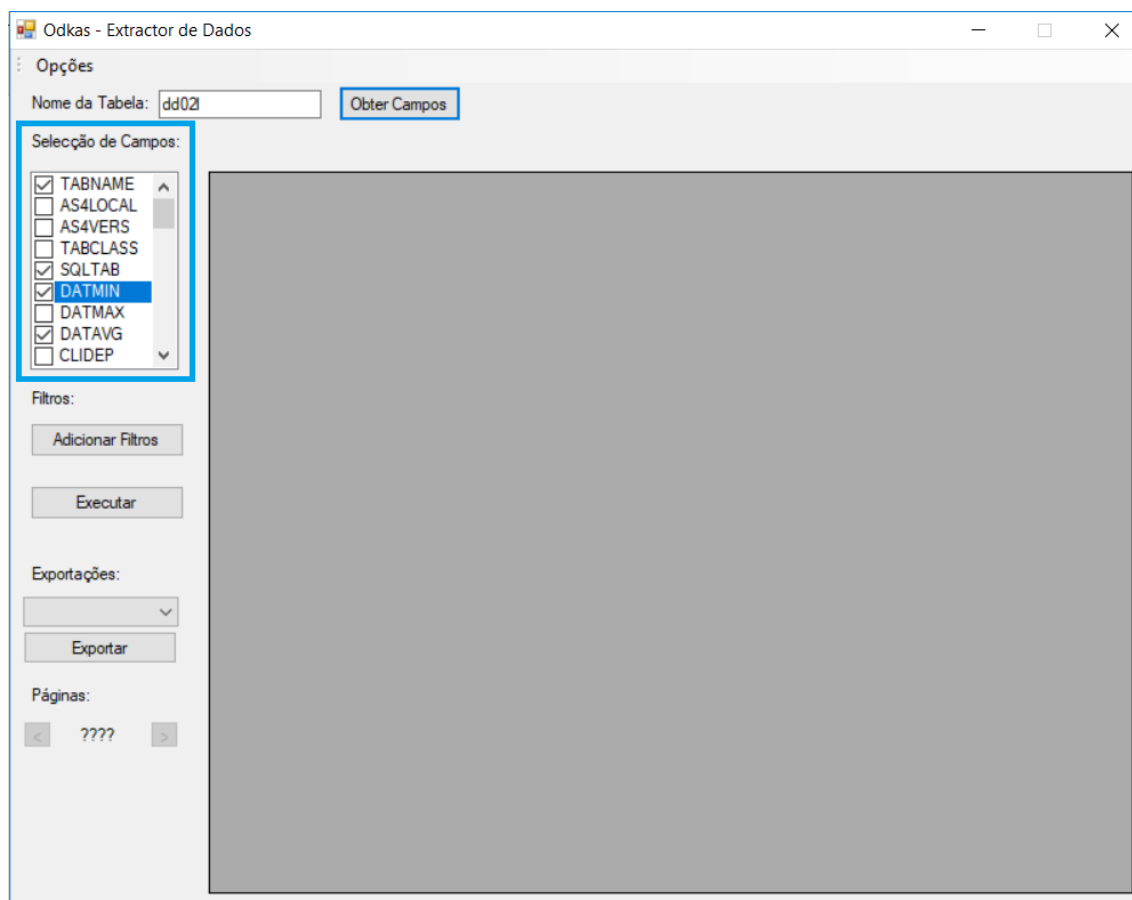


Figura 68 - Seleção de Campos

O utilizador pode efetuar a seleção dos campos através de um clique sobre qualquer uma das caixas, identificando-as com um “✓” (visto). Após esta seleção, torna-se possível proceder à adição de filtros ou - caso o utilizador ache que não é necessário -, pode proceder imediatamente para o executar de forma a obter os dados correspondentes à sua seleção. Os resultados obtidos através da seleção efetuada pelo utilizador podem ser visualizados na Figura 69.



### 8.1.2.3 Resultados (Sem Filtros)

The screenshot shows the 'Odkas - Extrator de Dados' application window. The 'Opções' section at the top has a 'Nome da Tabela' field containing 'dd02' and an 'Obter Campos' button. Below this is the 'Seleção de Campos' section, which includes a list of fields with checkboxes: TABNAME (checked), AS4LOCAL (unchecked), AS4VERS (unchecked), TABCLASS (unchecked), SQLTAB (checked), DATMIN (checked), DATMAX (unchecked), DATAVG (checked), and CLIDEP (unchecked). There is also a 'Filtros' section with an 'Adicionar Filtros' button and an 'Executar' button. The 'Exportações' section has a dropdown menu and an 'Exportar' button. At the bottom left, the 'Páginas' section shows '1/1121' with left and right navigation arrows. The main table displays the following data:

TABNAME	SQLTAB	DATMIN	DATAVG
V_AT30		0000000000	0000000000
CDPOS		0000000000	0000000000
DOKTL		0000000000	0000000000
CVEP11		0000000000	0000000020
CVEP12		0000000000	0000000020
CVEP13		0000000000	0000000020
CVEP14		0000000000	0000000020
CVEP21		0000000000	0000000020
CVEP22		0000000000	0000000020
CVEP23		0000000000	0000000020
CVEP24		0000000000	0000000020
CVER11		0000000000	0000000020
CVER12		0000000000	0000000020
CVER13		0000000000	0000000020
CVER14		0000000000	0000000020
CVER15		0000000000	0000000020
CVER21		0000000000	0000000020
CVER22		0000000000	0000000020
CVER23		0000000000	0000000020
CVER24		0000000000	0000000020

Figura 69 - Resultados da pesquisa

Após a obtenção dos dados, o utilizador pode navegar pelas páginas (as setas de navegação encontram-se identificadas pela cor castanha) à procura da informação que lhe é de maior relevância. Nesta fase do processo, de acordo com a , o utilizador encontra-se na página 1 de 1121 páginas. Se o utilizador tivesse utilizado filtros, o número de páginas teria sido reduzido consideravelmente. De forma a existir uma maior flexibilidade no extrator, é possível efetuar a seleção de campos adicionais e introduzir novos filtros, em qualquer momento da sua execução, sendo apenas necessário carregar novamente no botão “Executar” para carregar as novas alterações. Um exemplo deste tipo de modificação pode ser visto na Figura 70.

### 8.1.2.4 Resultados (Com Filtros e Campo Adicional)

The screenshot shows the 'Odkas - Extractor de Dados' interface. On the left, there are options for selecting fields and filters. The 'TABCLASS' field is checked and highlighted in red. Below the options, there are buttons for 'Adicionar Filtros', 'Executar', and 'Exportar'. At the bottom left, the page indicator shows '1/63', which is highlighted in grey. The main area displays a table of search results.

TABNAME	SQLTAB	DATMIN	DATAVG	TABCLASS
V_AT30		0000000000	0000000000	VIEW
V_AT40		0000000000	0000000000	VIEW
V_TBANK_CD_F...		0000000000	0000000000	VIEW
V_TBANK_CD_F...		0000000000	0000000000	VIEW
V_TBANK_CD_F...		0000000000	0000000000	VIEW
V_TBANK_CD_...		0000000000	0000000000	VIEW
V_TBANK_CD_...		0000000000	0000000000	VIEW
V_TBANK_CD_...		0000000000	0000000000	VIEW
V_TBANK_CD_T...		0000000000	0000000000	VIEW
VW_TBANK_CD...		0000000000	0000000000	VIEW
V_TFK070Q_TN...		0000000000	0000000000	VIEW
V_CLC		0000000000	0000000000	VIEW
V_TBANK_CD_F...		0000000000	0000000000	VIEW
V_HIST_REQ_A...		0000000000	0000000000	VIEW
V_CMV		0000000000	0000000000	VIEW
H_TCP05		0000000000	0000000000	VIEW
V_CPS		0000000000	0000000000	VIEW
ACM_CL_IMPL_IF		0000000000	0000000000	VIEW
ACM_CL_IMPL_I...		0000000000	0000000000	VIEW
ACM_DDDL SRC		0000000000	0000000000	VIEW

Figura 70 - Modificação da pesquisa inicial

Como pode ser observado pela , foi efetuada a introdução de um novo campo “TABCLASS” (Cor Roxo) e adicionados filtros, de forma a reduzir o número de páginas, permitindo chegarmos a uma pesquisa na qual existem apenas 63 páginas (Cor Cinzento) - ao contrário das 1121 páginas da pesquisa anterior, devido à não utilização de qualquer tipo de filtragem. No ponto representado na figura supra referida, o número máximo de filtros disponíveis é 5, mas existe sempre a possibilidade de aumentar este limite, caso seja necessário, por parte do utilizador.

### 8.1.2.5 Exportação de Dados para Ficheiros

Assim que o utilizador der por terminada a sua pesquisa de dados, pode efetuar a extração dos mesmos para quatro tipos de ficheiro, sendo estes PDF, XML, HTML e CSV. O local de extração está identificado na , pela cor verde.

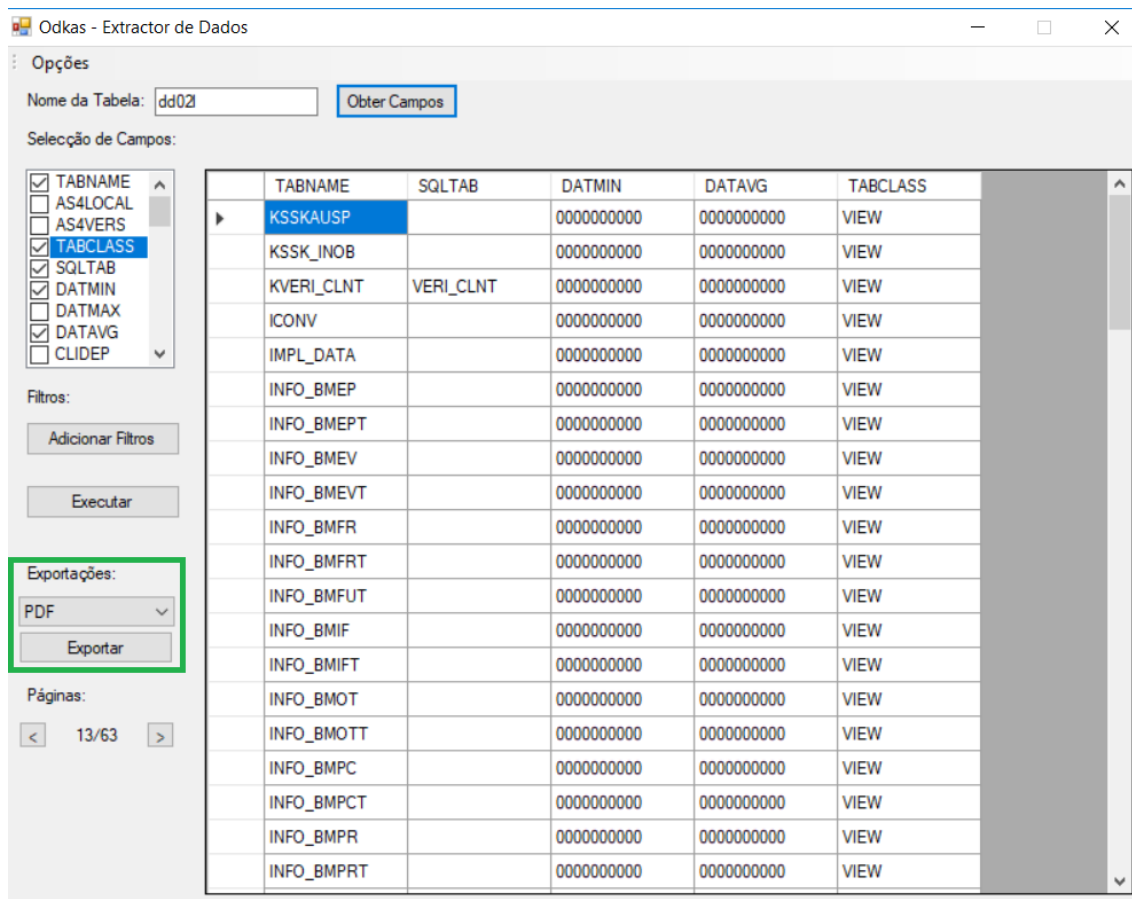
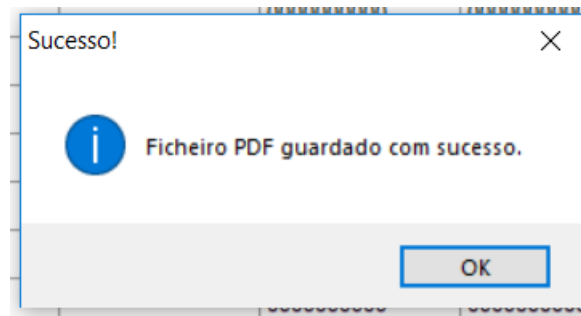


Figura 71 - Exportação de Dados

A extração destes dados pode ser efetuada em qualquer página selecionada pelo utilizador, como pode ser visualizado na figura acima, a qual revela que o utilizador se encontra na página 13. Assim sendo, os dados referentes a essa página são extraídos para o formato escolhido, neste caso, é escolhido o formato PDF. Se a extração for efetuada com sucesso, o utilizador recebe a seguinte informação:



*Figura 72 - Exportação com sucesso*

Em caso de ocorrência de alguma falha, é gerada uma mensagem de erro com o nome do formato, com a instrução de o utilizador efetuar uma nova tentativa de extração mais tarde. O formato e conteúdo dos ficheiros pode ser visualizado na secção de anexos no capítulo oito.

## 8.2 Filtragem de valores

The image shows a software window titled "FilterForm" with a close button (X) in the top right corner. The window contains a section labeled "Filtros" (Filters). There are three rows of filter controls, each consisting of a dropdown menu and a text input field. The first row has a dropdown menu labeled "COUNTRYFR" with a downward arrow and a text input field containing the value "DE". The second row has a dropdown menu labeled "AIRPFROM" with a downward arrow and a text input field containing the value "JFK". The third row has a dropdown menu with a downward arrow and an empty text input field. The dropdown menu is open, displaying a list of filterable fields: CARRID, CONNID, COUNTRYFR, CITYFROM, AIRPFROM, COUNTRYTO, CITYTO, AIRPTO, FLTIME, DEPTIME, ARRTIME, DISTANCE, DISTID, FLTYPE, and PERIOD.

Figura 73 – Filtragem de valores

### 8.3. Exportação de Dados PDF

TABNAME	TABCLASS	SQLTAB	DATMIN	DATAVG
KSSKAUSP	VIEW		000000000	000000000
KSSK_INOB	VIEW		000000000	000000000
KVERI_CLNT	VIEW	VERI_CLNT	000000000	000000000
ICONV	VIEW		000000000	000000000
IMPL_DATA	VIEW		000000000	000000000
INFO_BMEP	VIEW		000000000	000000000
INFO_BMEPT	VIEW		000000000	000000000
INFO_BMEV	VIEW		000000000	000000000
INFO_BMEVT	VIEW		000000000	000000000
INFO_BMFR	VIEW		000000000	000000000
INFO_BMFRT	VIEW		000000000	000000000
INFO_BMFUT	VIEW		000000000	000000000
INFO_BMIF	VIEW		000000000	000000000
INFO_BMIFT	VIEW		000000000	000000000
INFO_BMOT	VIEW		000000000	000000000
INFO_BMOTT	VIEW		000000000	000000000
INFO_BMPC	VIEW		000000000	000000000
INFO_BMPCT	VIEW		000000000	000000000
INFO_BMPR	VIEW		000000000	000000000
INFO_BMPRT	VIEW		000000000	000000000
INFO_BMPV	VIEW		000000000	000000000
INFO_BMPVT	VIEW		000000000	000000000
INFO_DD01L	VIEW	DD01L	000000000	000000000
INFO_DEVC	VIEW		000000000	000000000
INFO_DOMA	VIEW		000000000	000000000
INFO_DOMAT	VIEW		000000000	000000000
INFO_DTEL	VIEW		000000000	000000000
INFO_DTELT	VIEW		000000000	000000000
INFO_ENOFT	VIEW		000000000	000000000
INFO_ENQU	VIEW		000000000	000000000
INFO_ENQUF	VIEW		000000000	000000000
INFO_ENQUT	VIEW		000000000	000000000
INFO_FUGRG	VIEW		000000000	000000000
INFO_FUGRT	VIEW		000000000	000000000
INFO_FUGRZ	VIEW		000000000	000000000
INFO_FUNC	VIEW		000000000	000000000
INFO_FUNCT	VIEW		000000000	000000000
INFO_INDX	VIEW		000000000	000000000
INFO_MCID	VIEW		000000000	000000000
INFO_MCOB	VIEW		000000000	000000000
INFO_MCOBF	VIEW		000000000	000000000
INFO_MCOBT	VIEW		000000000	000000000
INFO_MCOFT	VIEW		000000000	000000000
INFO_MENU	VIEW		000000000	000000000
INFO_MODS	VIEW		000000000	000000000
INFO_MSAG	VIEW		000000000	000000000
INFO_MSNR	VIEW		000000000	000000000
INFO_SCAT	VIEW		000000000	000000000
INFO_SCATA	VIEW		000000000	000000000
INFO_SCATF	VIEW		000000000	000000000
INFO_SHLP	VIEW		000000000	000000000
INFO_SHLP	VIEW		000000000	000000000
INFO_SHLPT	VIEW		000000000	000000000
INFO_SUSO	VIEW		000000000	000000000
INFO_TABFT	VIEW		000000000	000000000
INFO_TABL	VIEW		000000000	000000000
INFO_TABLF	VIEW		000000000	000000000
INFO_TABLS	VIEW		000000000	000000000
INFO_TABLT	VIEW		000000000	000000000
INFO_TABLV	VIEW		000000000	000000000
INFO_TABS	VIEW		000000000	000000000
INFO_TABST	VIEW		000000000	000000000
INFO_TART	VIEW		000000000	000000000
INFO_TABTT	VIEW		000000000	000000000
INFO_TITLE	VIEW		000000000	000000000
INFO_TRAN	VIEW		000000000	000000000
INFO_TRANF	VIEW		000000000	000000000
INFO_TTYP	VIEW		000000000	000000000
INFO_TTYPT	VIEW		000000000	000000000
INFO_TWBA	VIEW		000000000	000000000
INFO_UDMO	VIEW		000000000	000000000
INFO_UDMOT	VIEW		000000000	000000000
INFO_UENA	VIEW		000000000	000000000
INFO_UENAT	VIEW		000000000	000000000
INFO_UENO	VIEW		000000000	000000000
INFO_UENOT	VIEW		000000000	000000000
INFO_VARI	VIEW		000000000	000000000
INFO_VIEF	VIEW		000000000	000000000
INFO_VIEFT	VIEW		000000000	000000000
INFO_VIEF	VIEW		000000000	000000000
INFO_VIEW	VIEW		000000000	000000000
INFO_VIEW2	VIEW		000000000	000000000
INFO_VIEWT	VIEW		000000000	000000000
IRW_VLTARCHDEF	VIEW		000000000	000000000
IRW_VLTARCHOBJ	VIEW		000000000	000000000
IRW_VLTCLDEF	VIEW		000000000	000000000
IRW_VLTCLUSTBL	VIEW		000000000	000000000
IRW_VLTCONFIG	VIEW		000000000	000000000
IRW_VLTIONMAP	VIEW		000000000	000000000
IWDST_EDIT	VIEW		000000000	000000000
IWDST_PRES	VIEW		000000000	000000000
HELP_SEUOBJ	VIEW		000000000	000000000
HFDIR	VIEW		000000000	000000000
HIAPAR	VIEW		000000000	000000000
HLP_OSOUR	VIEW		000000000	000000000
HRV1001	VIEW		000000000	000000000
HRV1016A	VIEW		000000000	000000000
HRV1017A	VIEW		000000000	000000000
HRV1210	VIEW		000000000	000000000
HRV1211	VIEW		000000000	000000000
HRV1212	VIEW		000000000	000000000

Figura 74 - Extração PDF

### 8.4. Exportação de Dados CSV

	A	B	C	D	E	F
1	TABNAME	TABCLASS	SQLTAB	DATMIN	DATAVG	
2	KSSKAUSP	VIEW		0	0	
3	KSSK_INOE	VIEW		0	0	
4	KVERI_CLM	VIEW	VERI_CLN	0	0	
5	ICONV	VIEW		0	0	
6	IMPL_DAT	VIEW		0	0	
7	INFO_BME	VIEW		0	0	
8	INFO_BME	VIEW		0	0	
9	INFO_BME	VIEW		0	0	
10	INFO_BME	VIEW		0	0	
11	INFO_BMF	VIEW		0	0	
12	INFO_BMF	VIEW		0	0	
13	INFO_BMF	VIEW		0	0	
14	INFO_BMI	VIEW		0	0	
15	INFO_BMI	VIEW		0	0	
16	INFO_BMC	VIEW		0	0	
17	INFO_BMC	VIEW		0	0	
18	INFO_BMF	VIEW		0	0	
19	INFO_BMF	VIEW		0	0	
20	INFO_BMF	VIEW		0	0	
21	INFO_BMF	VIEW		0	0	
22	INFO_BMF	VIEW		0	0	
23	INFO_BMF	VIEW		0	0	
24	INFO_DDO	VIEW	DD01L	0	0	
25	INFO_DEV	VIEW		0	0	
26	INFO_DOM	VIEW		0	0	
27	INFO_DOM	VIEW		0	0	
28	INFO_DTE	VIEW		0	0	
29	INFO_DTE	VIEW		0	0	

Figura 75 - Extração CSV

## 8.5. Exportação de Dados HTML

TABNAME	TABCLASS	SQLTAB	DATMIN	DATAVG
KSSKAUSP	VIEW		0000000000	0000000000
KSSK_INOB	VIEW		0000000000	0000000000
KVERI_CLNT	VIEW	VERI_CLNT	0000000000	0000000000
ICONV	VIEW		0000000000	0000000000
IMPL_DATA	VIEW		0000000000	0000000000
INFO_BMEP	VIEW		0000000000	0000000000
INFO_BMEPT	VIEW		0000000000	0000000000
INFO_BMEV	VIEW		0000000000	0000000000
INFO_BMEVT	VIEW		0000000000	0000000000
INFO_BMFR	VIEW		0000000000	0000000000
INFO_BMFRT	VIEW		0000000000	0000000000
INFO_BMFUT	VIEW		0000000000	0000000000
INFO_BMIF	VIEW		0000000000	0000000000
INFO_BMIFT	VIEW		0000000000	0000000000
INFO_BMOT	VIEW		0000000000	0000000000
INFO_BMOTT	VIEW		0000000000	0000000000
INFO_BMPC	VIEW		0000000000	0000000000
INFO_BMPCT	VIEW		0000000000	0000000000
INFO_BMPR	VIEW		0000000000	0000000000
INFO_BMPRT	VIEW		0000000000	0000000000
INFO_BMPV	VIEW		0000000000	0000000000
INFO_BMPVT	VIEW		0000000000	0000000000
INFO_DD01L	VIEW	DD01L	0000000000	0000000000
INFO_DEVC	VIEW		0000000000	0000000000
INFO_DOMA	VIEW		0000000000	0000000000
INFO_DOMAT	VIEW		0000000000	0000000000
INFO_DTEL	VIEW		0000000000	0000000000
INFO_DTELT	VIEW		0000000000	0000000000
INFO_ENQFT	VIEW		0000000000	0000000000
INFO_ENQU	VIEW		0000000000	0000000000
INFO_ENQUF	VIEW		0000000000	0000000000
INFO_ENQUT	VIEW		0000000000	0000000000
INFO_FUGRG	VIEW		0000000000	0000000000
INFO_FUGRT	VIEW		0000000000	0000000000
INFO_FUGRZ	VIEW		0000000000	0000000000

Figura 76 - Extração HTML



## 8.6 Exportação de Dados XML (Tabela)

<> NewDataSet

<> Table1 (100)

<> TABNAME	<> TABCLASS	<> SQLTAB	<> DATMIN	<> DATAVG
<> KSSKAUSP	VIEW		000000000	000000000
<> KSSK_INOB	VIEW		000000000	000000000
<> KVERI_CLNT	VIEW	VERI_CLNT	000000000	000000000
<> ICONV	VIEW		000000000	000000000
<> IMPL_DATA	VIEW		000000000	000000000
<> INFO_BMEP	VIEW		000000000	000000000
<> INFO_BMEPT	VIEW		000000000	000000000
<> INFO_BMEV	VIEW		000000000	000000000
<> INFO_BMEVT	VIEW		000000000	000000000
<> INFO_BMFR	VIEW		000000000	000000000
<> INFO_BMFRT	VIEW		000000000	000000000
<> INFO_BMFUT	VIEW		000000000	000000000
<> INFO_BMIF	VIEW		000000000	000000000
<> INFO_BMIFT	VIEW		000000000	000000000
<> INFO_BMOT	VIEW		000000000	000000000
<> INFO_BMOTT	VIEW		000000000	000000000
<> INFO_BMPC	VIEW		000000000	000000000
<> INFO_BMPCT	VIEW		000000000	000000000
<> INFO_BMPR	VIEW		000000000	000000000
<> INFO_BMPRT	VIEW		000000000	000000000
<> INFO_BMPV	VIEW		000000000	000000000
<> INFO_BMPVT	VIEW		000000000	000000000
<> INFO_DD01L	VIEW	DD01L	000000000	000000000
<> INFO_DEVC	VIEW		000000000	000000000
<> INFO_DOMA	VIEW		000000000	000000000
<> INFO_DOMAT	VIEW		000000000	000000000
<> INFO_DTEL	VIEW		000000000	000000000
<> INFO_DTELT	VIEW		000000000	000000000
<> INFO_ENQFT	VIEW		000000000	000000000
<> INFO_ENQU	VIEW		000000000	000000000
<> INFO_ENQUF	VIEW		000000000	000000000
<> INFO_ENQUT	VIEW		000000000	000000000
<> INFO_FUGRG	VIEW		000000000	000000000
<> INFO_FUGRT	VIEW		000000000	000000000
<> INFO_FUGRZ	VIEW		000000000	000000000
<> INFO_FUNC	VIEW		000000000	000000000
<> INFO_FUNCT	VIEW		000000000	000000000
<> INFO_INDY	VIEW		000000000	000000000
<> INFO_MCID	VIEW		000000000	000000000
<> INFO_MCOB	VIEW		000000000	000000000

Figura 77 - Extração XML