



Sistema de Gestão e Análise de Dados Para o Retalho de Produtos Óticos

BRUNO FILIPE PINHEIRO FERRAZ

Setembro de 2019

Sistema de Gestão e Análise de Dados

Retalho de Produtos Óticos

Bruno Ferraz

**Dissertação para obtenção do Grau de Mestre em
Engenharia Informática, Área de Especialização em Sistemas
Computacionais**

Orientador: Paulo Jorge Machado Oliveira

Júri:

Presidente:

Dr. ..., Professor, DEI/ISEP

Vogais:

Dr. ..., Professor, DEI/ISEP

Dr. ..., Professor, DEI/ISEP

Dr. ..., Professor, DEI/ISEP

Resumo

Em qualquer organização, é importante monitorizar os seus processos de negócio e a sua evolução, para que estes assegurem o sucesso da mesma. Numa organização que se dedica à comercialização de produtos a retalho, para que consiga acompanhar essas evoluções e os movimentos dos seus concorrentes, surge constantemente a necessidade de melhorar os seus atuais processos, tornando-os mais eficientes para que prevaleçam perante a concorrência.

Atualmente, um comercial da organização, na qual se insere este trabalho de mestrado, efetua o registo das suas vendas através do preenchimento de um ficheiro *excel*. Para proceder ao processamento desta venda, este tem que fazer chegar o documento criado a um armazenista, que por sua vez terá que dar seguimento ao processo de venda. Este comercial, de toda a vez que tem a necessidade de executar ações de sumário ou de apresentação de resultados, tem também que, realizar manualmente todas as operações que achar necessário para obter a informação pretendida.

Definiram-se então os objetivos para a criação de uma solução para o registo, armazenamento, manutenção e análise dos dados gerados pelas ações quotidianas desta organização. Inicialmente, fez-se um levantamento do estado da arte, apresentando alguns conceitos essenciais à compreensão do tema desta dissertação, estudando possíveis arquiteturas e ferramentas possíveis de adotar na solução (ferramentas de apoio à gestão, ferramentas capazes de gerir e armazenar os dados gerados no quotidiano da mesma, e ferramentas de apresentação e análise de dados).

Neste contexto, é apresentada uma possível solução para o problema descrito, solução esta capaz de resolver as insuficiências quanto ao registo das atividades de negócio da organização, tendo também a capacidade de gerir elevadas quantidade de dados e efetuar a sua respetiva manutenção quanto ao histórico. A solução contempla também a necessidade da análise aos dados que armazena, disponibilizando para o efeito um conjunto de análises definidas de acordo com as necessidades identificadas.

Palavras-chave: *Business Intelligence*, Armazém de dados, *ETL*, Análise de dados

Abstract

In any organization, it is important to monitor the business processes and their evolution to ensure success. In an organization that markets retail products, so that it can keep up with these developments and follow the movements of its competitors, there is an endless need to improve its current processes, making them more efficient, to prevail towards competition.

Currently, a commercial from the organization, where this master's work is inserted, records his sales by filling in an *excel* file. To process this sale, the file must reach a storekeeper who will have to follow up the sale process. This commercial, whenever he needs to perform summary actions or results presentation, also has to manually perform all the operations he deems necessary to obtain the desired information.

The objectives were defined as the creation of a solution for the recording, storing, maintaining and analysing the data generated by the daily actions of this organization. Initially, a state-of-the-art survey is presented, showing some essential concepts for understanding the theme of this dissertation, studying possible architectures and possible tools to adopt in the solution (management support tools, tools capable of managing and storing the data generated in this organization, presentation and data analysis tools).

In this context, a possible solution to the described problem is presented, a solution capable of solving the insufficiencies regarding the registration of the business activities of the organization, as well as able to manage large amounts of data and to maintain them as its history. The solution also addresses the need for analysis of the data it stores, providing a set of analyzes defined according to the identified needs.

Keywords: *Business Intelligence, Data Warehouse, ETL, Data Analysis.*

Agradecimentos

Ao Professor Doutor Paulo Jorge Machado Oliveira, pela orientação e ajuda ao longo deste projeto, pela sua disponibilidade, pela motivação e partilha de conhecimento que proporcionou. Sem o seu contributo este projeto não seria o mesmo.

Aos meus pais e irmão, o meu profundo agradecimento por todos os sacrifícios e me terem proporcionado esta oportunidade, pela compreensão sempre que não estive disponível, pela força e apoio que me deram ao longo de todo o meu percurso académico. Obrigado por terem feito de mim o que sou.

À minha namorada, pela constante motivação, confiança e valorização, pela paciência e compreensão, pelos conselhos, pelo carinho, amizade e toda a felicidade proporcionada. Obrigado.

Índice

Lista de Figuras	xiii
Lista de Tabelas	xv
Glossário	xvii
Lista de Acrónimos	xix
1 Introdução	1
1.1 Problema	1
1.2 Motivação	1
1.3 Objetivos	2
1.4 Estrutura do Documento	2
2 Análise de Valor	5
2.1 Processo de Inovação	6
2.2 Desenvolvimento do Novo Conceito	6
2.2.1 Identificação de Oportunidades	7
2.2.2 Análise de Oportunidades	8
2.2.3 Geração de Ideias	9
2.2.4 Seleção de Ideias	9
2.2.5 Definição de Conceito	10
2.3 Valor	11
2.3.1 Valor Para o Cliente	11
2.3.2 Valor Percecionado	11
2.3.2.1 Perspetiva Longitudinal de Valor	12
2.3.3 Rede de Valor	13
2.3.4 Método Multicritério <i>Analytic Hierarchy Process</i>	13
2.3.5 Proposta de Valor	16
3 Estado da Arte	17
3.1 Armazenamento de Grandes Volumes de Dados	17
3.2 Conceitos Base Para a Criação de Armazéns de Dados	18
3.2.1 <i>Business Intelligence</i> e Armazéns de Dados	18
3.2.2 Armazenamento de Dados	19
3.2.2.1 <i>Data Mart</i>	20
3.2.3 <i>Online Analytical Processing</i>	21
3.2.4 Arquiteturas de Armazéns de Dados	22
3.2.4.1 Arquitetura de <i>Ralph Kimball</i>	23

3.2.4.2	Arquitetura de <i>Bill Inmon</i>	24
3.2.4.3	Comparação Entre Arquiteturas	25
3.2.5	Modelação Dimensional	27
3.3	Análise de Mercado	29
3.3.1	Modelo Básico: Supermercado	29
3.3.2	Indicadores Chave de Performance	31
3.4	Ferramentas de Gestão	33
3.4.1	Tecnologias para a Construção de Ferramenta de Gestão	33
3.4.1.1	<i>ASP.NET Core 2</i>	33
3.4.1.2	<i>NodeJs</i>	34
3.4.1.3	<i>Java</i>	34
3.4.1.4	Comparação Entre as Tecnologias Apresentadas	35
3.4.2	Ferramentas Existentes de Apoio à Gestão	35
3.4.2.1	Primavera <i>BSS</i>	35
3.4.2.2	<i>PHC</i>	36
3.4.2.3	<i>Sage</i>	38
3.4.2.4	Comparação Entre Ferramentas	38
3.5	Ferramentas de <i>ETL</i>	39
3.5.1	<i>Microsoft SQL Server Integration Services</i>	40
3.5.2	<i>Pentaho Data Integration</i>	41
3.5.3	<i>Talend Open Studio for Data Integration</i>	42
3.5.4	Comparação entre Ferramentas de <i>ETL</i>	43
3.6	Ferramentas de Processamento Analítico de Dados	44
3.6.1	<i>Microsoft Power BI</i>	44
3.6.2	<i>Phocas</i>	45
3.6.3	<i>Kibana</i>	46
3.6.4	Comparação entre Ferramentas de Processamento Analítico de Dados	47
4	Análise e Desenho	49
4.1	Modelação de Negócio	49
4.2	Engenharia de Requisitos	50
4.2.1	Partes Interessadas	51
4.2.2	Atores do Sistema	51
4.2.3	Requisitos Funcionais	51
4.2.4	Requisitos Não Funcionais	53
4.3	Design e Arquitetura da Solução	54
4.3.1	Fontes de Informação	54
4.3.2	Definição de Indicadores de Performance	57
4.3.3	Análise do Negócio	57
4.3.4	Arquitetura da Solução	58
4.3.4.1	<i>Extração de Dados</i>	59
4.3.4.2	<i>Staging Area</i>	60
4.3.4.3	<i>Armazém de Dados</i>	63
5	Implementação	67
5.1	Ferramenta de Gestão	67
5.1.1	Idioma	67
5.1.2	Autenticação e Autorização	68
5.1.3	Gestão de Utilizadores	69

5.1.4	Gestão de Artigos	70
5.1.5	Criação de Encomendas	70
5.1.6	Validação de Encomendas	71
5.1.7	Geração de Documentação	71
5.1.8	Atualização de <i>Stocks</i>	72
5.1.9	Gestão de <i>Template</i> de Documento	72
5.1.10	Criação de Alertas	73
5.2	Criação do Armazém de Dados	73
5.2.1	Ficheiro de Propriedades	74
5.2.2	Processo de <i>ETL</i>	74
5.2.3	Migração de Dados do Sistema Legado	80
5.3	Ferramenta de Processamento Analítico de Dados	81
5.3.1	Autorização e Autenticação	81
5.3.2	Ficheiro de Integração	82
5.3.3	Consultas Realizadas	83
5.3.3.1	Valor Líquido de Artigos Vendidos	83
5.3.3.2	Valor Bruto de Artigos Vendidos	84
5.3.3.3	Valor em descontos por artigos vendidos	85
5.3.3.4	Valor em impostos por artigos vendidos	86
5.3.3.5	Número de artigos vendidos	86
5.3.3.6	Outras consultas	87
6	Avaliação de Resultados	91
6.1	Grandezas em Avaliação	91
6.2	Hipóteses a Avaliar	91
6.3	Indicadores e Fontes de Informação	92
6.4	Metodologia de Avaliação	92
6.5	Experiências e Resultados Obtidos	93
6.5.1	Testes de Exatidão	93
6.5.2	Testes de Desempenho	95
7	Conclusões	97
7.1	Objetivos atingidos	97
7.2	Limitações da Solução	98
7.3	Análise de Alternativas e Trabalho Futuro	99
7.4	Apreciação final	101
	Bibliografia	103
A	Versão Completa da Estrutura Usada na Staging Area	107
B	Versão Completa da Estrutura Usada na Staging Area DQP	109
C	Versão Completa do Modelo Dimensional	111
D	Ficheiro de Indexação no Elasticsearch	113
E	Dashboard Exemplo	115
F	Tipos de Consultas Disponíveis no Kibana	117

G	Estratégia de Slowly Changing Dimension	119
H	Ficheiro de Venda Exemplo	123
I	Exemplo de Dados Informativos Produzidos Pelo Kibana	125

Lista de Figuras

1	Modelo <i>Canvas</i>	6
2	Modelo de <i>New Concept Development</i> (Koen, Bertels e Kleinschmidt 2014)	7
3	Perspetiva Longitudinal de Valor segundo <i>Tony Woodall</i> (Woodall 2003).	12
4	Cubo Online Transaction Processing (OLAP).	21
5	Divisões lógicas de um Armazém de Dados (AD) segundo <i>Ralph Kimball</i> (Kimball e Ross 2013).	23
6	Divisões lógicas de um AD segundo <i>Bill inmon</i>	25
7	Modelo Dimensional - Supermercado.	30
8	Modelo Dimensional - <i>Snapshot</i> do Inventário de um Supermercado.	31
9	<i>Primavera Business Software Solutions (BSS) Enterprise Resource Planning (ERP)</i>	36
10	Ecrã de faturação do <i>ERP PHC</i>	37
11	<i>Software Sage</i>	38
12	<i>Microsoft SQL Server Integration Services Integrated Development Environment (IDE)</i>	41
13	<i>Pentaho Data Integration IDE</i>	42
14	<i>Talend Open Studio IDE</i>	43
15	Elementos Principais do <i>Microsoft Power BI</i>	45
16	<i>Dashboard</i> de Vendas Criado no <i>Phocas</i>	46
17	Gráfico de Séries Temporais em <i>Kibana</i>	47
18	Modelo de Domínio Previsto Para a Solução.	50
19	Diagrama de Casos de Uso Previsto Para a Solução.	52
20	Exemplo de Ficheiro de Clientes.	55
21	Exemplo de Ficheiro de Artigos Para Vendas.	55
22	Exemplo de Ficheiro Utilizado Para o Registo de Venda.	56
23	Arquitetura Geral do Armazém de Dados Previsto Para a Solução.	58
24	Versão Simplificada da Estrutura de Armazenamento de <i>Staging</i>	61
25	Versão Simplificada das Tabelas <i>Data Quality Problems (DQP)</i> de <i>Staging</i>	62
26	Modelo Dimensional Simplificado.	65
27	Ecrã de Configuração Para Escolha de Idioma.	68
28	Ecrã de Login.	68
29	Ecrã de Configuração de Acessos.	69
30	Ecrã de Funcionalidade Disponíveis Para o Utilizador Armazenista.	69
31	Ecrã Para Gestão de Utilizadores da Aplicação.	70
32	Ecrã Para Gestão de Artigos da Aplicação.	70

33	Ecrã Para Criação de Encomenda.	71
34	Ecrã Para Validação de Encomenda.	71
35	Ecrã de Consulta de Documentos Relativos a Vendas.	72
36	Ecrã Para o Registo de Entradas de <i>Stock</i>	72
37	Ecrã Para o Registo <i>Templates</i> de Documento.	73
38	Ecrã de Alerta.	73
39	Ficheiro de propriedades <i>Pentaho</i>	74
40	<i>Overview</i> geral do processo <i>ETL</i> implementado.	74
41	Fluxo de <i>DQP</i> Implementado Para a Tabela Artigo.	75
42	Registos Criados Pelo Processo de <i>DQP</i>	76
43	Estratégia de <i>SCD</i> aplicadas à dimensão "DimArtigo"no <i>Pentaho</i>	77
44	Resultado das Operações de <i>SCD</i> à "DimArtigo"e "DimCliente"no <i>Pentaho</i>	78
45	Visão Geral do Processo de Carregamento da Tabela de Factos.	80
46	Processo de Migração e dados no <i>Pentaho</i>	81
47	Criação de um utilizador e respetivo acesso no <i>kibana</i>	82
48	Criação do índice no <i>kibana</i>	83
49	Valor líquido de artigos vendidos.	84
50	Valor bruto de artigos vendidos.	85
51	Valor em descontos por artigos vendidos.	85
52	Valor em impostos por artigos vendidos.	86
53	Número de artigos vendidos.	87
54	Mapa de Vendas de Artigos.	88
55	<i>Dashboard Exemplo</i>	89
56	<i>Comando SQL Desenvolvido Para Teste</i>	94
57	<i>Análise à Venda em Teste na Ferramenta Kibana</i>	94

Lista de Tabelas

1	Tabela Comparativa de Critérios do Método <i>AHP</i>	14
2	Tabela Comparativa de Critérios Definidos Para a Solução	14
3	Matriz Normalizada dos Critério Anteriormente Definidos	15
4	Peso Relativo de Cada Critério na Aplicação do Método <i>AHP</i>	15
5	Comparação entre sistemas OLAP e Online Transaction Processing (OLTP).	20
6	Comparação entre arquiteturas de <i>Kimbal</i> e <i>Inmon</i> (Kimball e Caserta 2004) (Kimball e Ross 2013) (Immon 2002).	26
7	Comparação dos requisitos desejados por ferramenta em análise.	39
8	Comparação entre Ferramentas <i>ETL</i>	43
9	Comparação entre Ferramentas de Processamento Analítico de Dados.	48
10	Matriz de Relacionamento de Factos e Dimensões	58
11	Tabela de Tempos Médios Para Geração de Análises Gráficas	95
12	Objetivos definidos e o seu grau de tangibilidade.	98

Glossário

dashboard É basicamente uma forma de visualização de dados em modo gráfico servindo normalmente para realçar pormenores ou ajudar na elaboração de conclusões sobre os dados. ¹.

data mining É o processo de exploração de grandes conjuntos de dados para a identificação de padrões e relacionamentos entre esses mesmos dados para resolver problemas através da sua análise. ².

Eclipse Programa de desenvolvimento de *software*..

Excel Ferramenta criada pela Microsoft para o cálculo, análise e visualização de dados. ³.

script É um programa ou sequência de instruções interpretado(a) ou executado(a) por outro programa. ⁴.

XML A eXtensible Markup Language (XML) é um *standard* usado para descrever um conjunto de dados. O padrão XML é uma forma flexível e estruturada de criar formatos de informação e partilhá-los por meio eletrônico estruturados através de uma rede de comunicação digital. ⁵.

¹<http://www.isocrm.com.br/dashboard-e-decisoes-empresariais/>

²<http://searchsqlserver.techtarget.com/definition/data-mining/>

³<https://www.edx.org/course/analyzing-visualizing-data-excel-microsoft-dat206x-8>

⁴<http://whatis.techtarget.com/definition/script/>

⁵<http://searchmicroservices.techtarget.com/definition/XML-Extensible-Markup-Language/>

Lista de Acrónimos

AD	Armazém de Dados.
AHP	Analytic Hierarchy Process.
API	Application Programming Interface.
BI	Business Intelligence.
BSS	Business Software Solutions.
CIF	Corporate Information Factory.
DQP	Data Quality Problems.
ELT	Extração, Carregamento e Transformação.
ERP	Enterprise Resource Planning.
ETL	Extração, Transformação e Carregamento.
FEI	Front End of Inovation.
FTP	File Transfer Protocol.
HOLAP	Hybrid Online Analytical Processing.
HTTP	JavaScript Object Notation.
IDE	Integrated Development Environment.
JSON	JavaScript Object Notation.
JVM	Java Virtual Machine.
KPI	Key Performance Indicator.
LDAP	Lightweight Directory Access Protocol.
MOLAP	Multidimensional Online Analytical Processing.
NCD	New Concept Development.
NIF	Número de Identificação Fiscal.
NPD	New Product Development.
NPM	Node Package Manager.
OLAP	Online Transaction Processing.

OLTP	Online Transaction Processing.
PDI	Pentaho Data Integration.
PKI	Public Key Infrastructure.
POS	Point Of Sale.
PVP	Preço de Venda ao Público.
ROLAP	Relational Online Analytical Processing.
SCD	Slowly Changing Dimensions.
SMTP	Simple Mail Transfer Protocol.
SQL	Structured Query Language.
SSDT	SQL Server Data Tools.
SSIS	SQL Server Integration Services.
SSMS	SQL Server Management Studio.
TOS	Talend Open Studio.
TRIZ	Theory of Inventive Problem Solving.
TSG	Tecnology Stage Gate.

1. Introdução

Este capítulo tem o objetivo de descrever o tema de estudo realizado no âmbito da unidade curricular Tese de Mestrado em Engenharia Informática, lecionada no Instituto Superior de Engenharia do Porto. Inicialmente será apresentado o problema que deu origem a esta dissertação, a definição em traços gerais daquilo que é pretendido pela solução, a forma como satisfaz as necessidades encontradas e a definição dos objetivos concretos do projeto. Por último, será descrita a estrutura a seguir ao longo de todo o documento.

1.1 Problema

A constante evolução dos sistemas computacionais e das suas tecnologias associadas, assim como das técnicas de análise de dados fazem desses mesmos dados uma ferramenta cada vez mais essencial no mundo profissional pois, extraindo conhecimento relevante pode tornar-se um fator diferenciador e de sucesso para quem o possuir e entender.

Na origem desta dissertação esta uma empresa real, no ramo da venda de produtos óticos, com um problema relativo à informatização das suas vendas e posterior análise. Sempre que é feita uma venda por parte de um comercial da empresa, é feita a introdução dos dados para um documento (ficheiro Excel) com os detalhes dessa mesma venda. Apesar da criação deste documento digital, nenhuma outra ação é realizada informaticamente, não havendo nenhuma plataforma que suporte o processo da venda, obrigando os seus intervenientes a terem que trocar entre si a informação, por exemplo, via *e-mail*.

É preciso também realizar manualmente todos os cálculos que sejam necessários para o resto do processo de venda, ou de análises de dados posteriores que se pretendam realizar, como por exemplo: valores totais líquidos e brutos por artigo, por família de artigo, por colaborador, por localização, entre outras características de negócio. Assim, sempre que é necessária a realização de análises de dados, todo o processo de cálculo de valores tem que ser processado manualmente pelo interessado na análise de dados.

1.2 Motivação

Desta forma, surgiu a necessidade de criar uma solução que resolva as dificuldades encontradas na informatização e automatização dos dados, proporcionando a introdução fácil e rápida dos dados num sistema informático.

A oportunidade de criar um sistema que resolva os problemas apresentados anteriormente, associada ao objetivo de resolver um problema real e concreto constitui, a principal motivação que levou à aposta no presente projeto.

Por outro lado, a possibilidade da criação de um sistema atual, seguindo práticas e padrões aprendidos ao longo do percurso académico, bem como a possibilidade de captação de conhecimento em áreas nunca antes trabalhadas (por exemplo a criação de *dashboards* de vendas), constitui também um enorme desafio pessoal, motivador e que teve grande peso na escolha do projeto atual.

1.3 Objetivos

Tendo em conta o problema apresentado, as necessidades identificadas e as motivações acima descritas tem-se como objetivo o desenvolvimento de uma solução que integre não só a informação relativa às vendas de produtos, mas também a geração de documentação de vendas/faturação e a importação de documentos relativos a operações anteriormente realizadas (atualmente em formato Excel).

A solução a desenvolver deverá também possibilitar a gestão de artigos e de utilizadores da aplicação, funcionando também como meio de comunicação entre os vários intervenientes do processo de negócio.

Informatizado todo o processo atualmente manual, pretende-se o desenvolvimento de uma solução capaz de criar histórico resultante da atividade quotidiana da empresa, de modo a possibilitar análises de dados mais detalhadas das suas operações e do respetivo valor de negócio, bem como a criação de dashboards para fácil visualização dos indicadores chave de performance da empresa. Os indicadores de performance definidos para a organização serão descritos na Secção 4.3.2 do presente documento. Pertende-se também que a solução seja capaz de lidar e de manter a sua performance mesmo quando o volume de dados a trabalhar for elevado, uma vez que se espera a geração de dados em elevada escala.

1.4 Estrutura do Documento

Este documento começa por apresentar um capítulo de Introdução, fazendo uma breve descrição do contexto e do problema que deu origem a esta dissertação.

De seguida, é apresentado um capítulo de Análise de Valor, onde se detalha o valor que este projeto cria ao ser realizado e o modelo de negócio em que se baseia.

No terceiro capítulo, relativo ao Estado de Arte, serão apresentados conceitos relativos à área de estudo em que se enquadra este projeto, diferentes ferramentas de processamento, análise e apresentação de dados, soluções existentes no mercado e indicadores de performance utilizados na área do retalho.

De seguida, no quarto capítulo, de modo a descrever a solução de um ponto de vista tecnológico, é apresentado o capítulo de Análise e Desenho, onde é apresentada a arquitetura desenvolvida para a solução com base no problema apresentado.

O quinto capítulo, Implementação, apresenta a forma utilizada para chegar à solução proposta, servindo para explicar e exemplificar as opções tomadas e que contribuem para a realização das funcionalidades desejadas.

No capítulo seguinte, apresenta-se a metodologia que se pretende seguir para a Avaliação de Resultados, descrevendo as grandezas e as hipóteses a avaliar e, posteriormente, as experiências de avaliação desenvolvidas e os resultados obtidos.

Por fim, o último capítulo surge como uma Conclusão do trabalho elaborado, onde é feita uma reflexão acerca dos objetivos atingidos, limitações, alternativas e trabalho futuro, assim como uma apreciação final de toda a dissertação.

2. Análise de Valor

Com a execução deste projeto, o cliente pretende uma plataforma informática onde possa registar as suas atividades de negócio, bem como a possibilidade de gerar análises aos dados dessas mesmas atividades de forma a retirar o máximo proveito dos seus recursos e do conhecimento que gera no seu dia-a-dia. Na Figura 1, apresenta-se o modelo *Canvas* que descreve de uma forma resumida o negócio.

Começando pela proposta de valor, refere-se a necessidade de um sistema eficiente para a informatização dos dados relativos ao quotidiano da organização, assim como a comodidade oferecida aos intervenientes do negócio com a informatização de processos até agora manuais, longos, demorosos e repetitivos. É apresentada, também, uma solução para a análise dos dados produzidos e anteriormente informatizados, possibilitando de forma rápida e visual a consulta dos dados, com histórico associado, permitindo a percepção de padrões ou hábitos que poderão trazer mais valias ao processo de negócio.

Espera-se que este sistema se traduza numa fonte de receitas, devido aos ganhos associados com a agilização de processos até agora manuais e com a análise providenciada aos dados de negócio da organização, permitindo o aumento dos seus lucros.

O relacionamento com o cliente surgirá através da assistência nas atividades/funcionalidades desenvolvidas e, pelo *feedback* fornecido pelos utilizadores do sistema.

Como canal de distribuição é apresentado o sistema desenvolvido aos seus utilizadores, os intervenientes do negócio.

Estes intervenientes correspondem à vertente de segmentos de clientes presentes no modelo, sendo eles: o comercial, o armazenista e o administrador.

Para o bom funcionamento do sistema conta-se com a parceria das pessoas responsáveis pela área informática da organização para efetuar a manutenção necessária ao *hardware*, e com as informações e *feedback* devolvido pelos intervenientes do negócio de modo ao desenvolvimento contínuo e melhorado da solução.

Como atividades chave identificaram-se o contínuo suporte e manutenção do sistema, quer a nível de *hardware* e *software*, tendo-se identificado também um recurso chave neste processo, o *hardware*. Este, em conjunto com a correta construção da solução, irá permitir a sua alta escalabilidade e performance.

Para o devido funcionamento, a solução apresenta como estruturas de custo os gastos associados à manutenção de equipamentos e do próprio *software* utilizado, uma vez que eventualmente poderá apresentar despesas de licenciamento, manutenção, refinamento e/ou adição de funcionalidades.

Parceiros Chave <ul style="list-style-type: none"> • Pessoal técnico responsável pela área informática da organização: manutenção básica ao hardware (servidores). • Vários intervenientes do negócio: responsáveis pela identificação de novas funcionalidades; 	Atividades Chave <ul style="list-style-type: none"> • Suporte ao sistema desenvolvido; • Manutenção de hardware; • Manutenção de software pela correção de eventuais bugs aplicativos, feedback obtido e novas funcionalidades; • Geração de documentação. 	Proposta de Valor <ul style="list-style-type: none"> • Sistema informático para o registo das atividades de negócio da organização (gestão de artigos e stocks, gestão de utilizadores); • Meio de comunicação entre os intervenientes do processo de negócio; • Metodologia de limpeza e tratamento de dados, assim como a captação das suas alterações; • Ferramenta de análise e visualização gráfica de dados; • Redução de custos e/ou recursos devido à poupança de tempo associada à automatização de processos; • Informação segmentada; • Aumento de performance; • Solução escalável. 	Relacionamento com o Cliente <ul style="list-style-type: none"> • Assistência às atividades quotidianas dos intervenientes do processo de negócio; • Obtenção de feedback e sugestão de melhorias. 	Segmentos de Clientes <ul style="list-style-type: none"> • Vários intervenientes do negócio: comercial, armazenista, administrador.
	Recursos Chave <ul style="list-style-type: none"> • Hardware (servidor/es): permite a escalabilidade e performance do sistema. 		Canais de Distribuição <ul style="list-style-type: none"> • Divulgação do sistema por partes dos órgãos superiores da organização. 	
Estruturas de Custo <ul style="list-style-type: none"> • Manutenção de hardware: servidores e terminais onde o sistema estará disponível para utilização por parte dos intervenientes no sistema; • Manutenção de software: eventual custo relacionado com licenciamento de software, manutenção da solução existente e adição de funcionalidades. 		Fonte de Receitas <ul style="list-style-type: none"> • Ganhos relacionados com a poupança de recursos devido à agilização de processos; • Ganhos adquiridos pela análise dos dados de negócio que permitirão adaptar estratégias de venda. 		

Figura 1: Modelo Canvas.

2.1 Processo de Inovação

Com esta solução pretende-se a implementação de um sistema que traga mais valor ao negócio da organização por via da inovação nos processos até então usados.

O processo de inovação divide-se em três fases: *front end*, *new product development* e *commercialization*. A primeira fase baseia-se num trabalho de pesquisa de oportunidades, seguindo-se uma fase de análise, onde se fazem algumas avaliações à potencialidade do projeto a nível técnico e de marketing. Por fim, a terceira fase, foca-se num âmbito mais comercial. (Koen, Bertels e Kleinschmidt 2014)

Derivado de diversos estudos realizados na área, que determinaram que o sucesso depende tanto das características das organizações como das atividades específicas do projeto, surgiu um novo modelo, com uma nova diferente estrutura para as atividades do *front end*, chamada *New Concept Development (NCD)*, juntamente com a introdução do termo "*front end of innovation*" que pretende substituir o anterior "*fuzzy front end*". O objetivo foi diminuir a imprecisão atribuída à fase de *front end*, identificando atividades específicas e atributos das organizações que contribuiriam para o sucesso desta fase (Koen, Bertels e Kleinschmidt 2014).

2.2 Desenvolvimento do Novo Conceito

NCD, ou Desenvolvimento do Novo Conceito, proporciona uma linguagem comum para perceber as atividades que ocorrem no *front end*. Este modelo encontra-se dividido em três partes: motor (*engine*), a roda (*wheel*) e a jante (*rim*), tal como é apresentado na Figura 2. O motor consiste nos elementos do núcleo e serve para fornecer força ao processo de *front end*. Estes elementos são os atributos organizacionais, tais como os valores, estratégia, cultura e a colaboração entre equipas. A roda é constituída por cinco atividades: identificação

de oportunidades, análise de oportunidades, geração de ideias, seleção de ideias e definição de conceito. Por fim, a jante, consiste nos fatores ambientais externos que influenciam o motor e os elementos de atividade (Koen, Bertels e Kleinschmidt 2014).

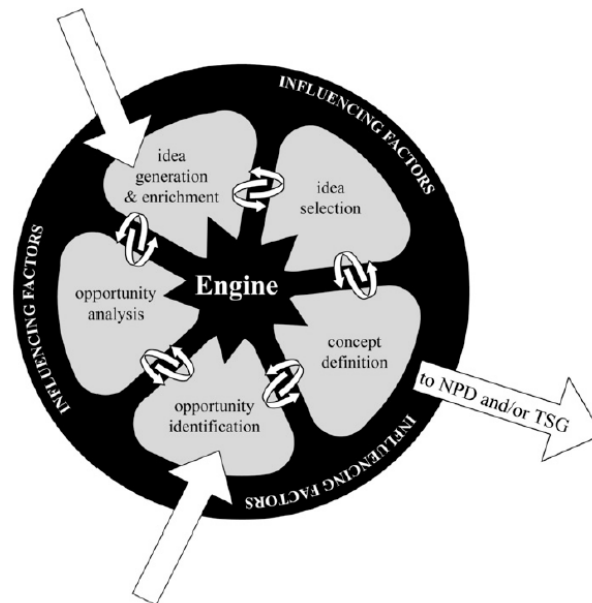


Figura 2: Modelo de *New Concept Development* (Koen, Bertels e Kleinschmidt 2014)

Pela análise da Figura 2 conclui-se que o modelo possui forma circular para demonstrar que os cinco elementos da roda se interligam entre si, ou seja, as oportunidades relacionam-se com as ideias e *vice-versa*. Os projetos que seguem este modelo têm início na identificação de oportunidades ou na fase de geração de ideias, representadas pelas setas de entrada no modelo. Posteriormente, os projetos seguem da atividade de geração de conceito para o processo *New Product Development (NPD)* ou para o *Technology Stage Gate (TSG)*, que ao contrário do *NCD*, são sequenciais. As atividades que constituem o *Front End of Innovation (FEI)* serão descritas nas subsecções a seguir (Koen, Bertels e Kleinschmidt 2014).

2.2.1 Identificação de Oportunidades

A fase de identificação de oportunidades é um dos pontos de entrada do modelo NCD e tem como objetivo identificar oportunidades exploráveis, sejam elas comerciais, no sentido de alocar corretamente os recursos em novos segmentos de mercado, otimização de processos ou aumento de eficiência dos métodos existentes.

Esta atividade é normalmente despoletada devido a objetivos comerciais, por exemplo, para dar resposta a uma nova concorrência, a possibilidade de inovar e obter vantagem competitiva, melhorar as operações e processos internos aumentando a eficiência ou simplesmente reduzir os custos.

Para executar esta atividade existem diferentes métodos, ferramentas e técnicas que possibilitam a execução, nomeadamente (A.Koen et al. 2002):

- **Roadmapping:** capturam as forças de um negócio de forma gráfica, a fim de melhorar a comunicação e o conhecimento.

- **Análise das tendências tecnológicas:** dedica-se à pesquisa e análise de tendências tecnológicas que possam ser relevantes e tornar-se oportunidades competitivas para o projeto.
- **Análise de tendências do cliente:** prende-se com a pesquisa e análise das tendências numa vertente mais direcionada para os clientes.
- **Análise de inteligência competitiva:** elemento essencial na atividade de identificação de oportunidades, uma vez que permite obter, analisar e comunicar informação de tendências competitivas que ocorrem fora da organização, possibilitando a definição de uma estratégia de negócio.
- **Pesquisa de mercado:** pesquisa do mercado em que o projeto se insere.
- **Planeamento de cenário:** proporciona uma abordagem disciplinada para imaginar e preparar o futuro, estimulando as decisões que de outra forma seriam ignoradas. O objetivo será criar diferentes perspetivas do futuro e assim obter conhecimento daquele que poderá ser o ambiente futuro que envolve o projeto.

Com o aumento da competitividade no mercado torna-se cada vez mais difícil a vida de uma organização, por isso é importante que estas possuam metodologias eficientes para a captação de oportunidades de negócio, de modo que as possam aproveitar da forma mais lucrativa.

Uma ferramenta deste tipo em conjunto com o histórico que cria ao longo do tempo poderá assegurar o bom funcionamento da organização ou, em caso de detetar alguma anomalia nos dados, poderá retificar o seu processo de negócio com base no conhecimento adquirido. Devido a este conhecimento adquirido, poderão ser identificados também certos comportamentos ou padrões de atividades por partes dos clientes que até então não foram explorados criando novas janelas de oportunidade de negócio para a organização.

2.2.2 Análise de Oportunidades

Este elemento permite a avaliação de determinada oportunidade no sentido de verificar se a mesma faz efetivamente falta à entidade.

Os métodos para análise de oportunidades segundo (A.Koen et al. 2002) são:

- **Enquadramento estratégico:** forma como determinada oportunidade se enquadra com as forças, fraquezas e ameaças tecnológicas e do mercado da organização.
- **Avaliação do segmento de mercado:** a avaliação do segmento de mercado fornece uma descrição detalhada demonstrando porque é que determinada oportunidade é efetivamente válida.
- **Análise da concorrência:** a análise da concorrência permite identificar quem são os maiores concorrentes em determinado segmento de mercado. Determina ainda que tipo de produtos são necessários para obter vantagem competitiva e permite avaliar as estratégias e capacidades da concorrência.
- **Avaliação do cliente:** permite determinar quais são as principais necessidades dos clientes e que não são atendidas pelos produtos atuais.

Considerando aquilo que se propõe com a implementação deste projeto, a ferramenta proposta será capaz de providenciar aos funcionários da organização uma análise clara, coerente e fiável dos indicadores de performance associados ao seu negócio oferecendo-lhe as formas mais adequadas para a análise dos dados e conseqüentemente das suas oportunidades de negócio.

No entanto, faz sentido estudar o mercado em que a solução se insere e os seus possíveis concorrentes, nomeadamente outros sistemas que permitam implementar esta metodologia nas organizações, no sentido de identificar as suas fraquezas e aproveitar as suas mais-valias para a solução idealizada.

2.2.3 Geração de Ideias

A geração de ideias e o seu conseqüente enriquecimento prende-se com o nascimento, desenvolvimento e maturação de uma ideia em concreto, sendo um dos pontos de entrada do modelo *NCD*, à semelhança da identificação de oportunidades (A.Koen et al. 2002).

As ideias podem percorrer várias iterações e passar por múltiplas alterações à medida que são analisadas, estudadas e discutidas com os restantes elementos do modelo *NCD* (A.Koen et al. 2002).

É comum, nesta fase, haver contacto direto com os clientes e utilizadores, e para a realizar podem ser utilizados processos formais (sessões de *brainstorming*) ou informais (fornecedor a oferecer um novo material ou cliente que faz um pedido incomum), no sentido de gerar ideias novas ou refinadas para determinada oportunidade (A.Koen et al. 2002).

Esta fase é normalmente realizada pela execução de técnicas de *brainstorming* para formação de ideias, ou a aplicação de métodos de enriquecimento que utilizam a *Theory of Inventive Problem Solving (TRIZ)*. A *TRIZ* é uma metodologia que melhora a criatividade levando os indivíduos a pensar além da sua própria experiência e a resolver os problemas utilizando soluções de outras áreas de ciência (A.Koen et al. 2002).

No projeto desenvolvido, a geração da ideia foi realizada em simultâneo com a identificação de um problema e de uma oportunidade, sendo que ambas se encontram bastante relacionadas. No entanto, o seu enriquecimento é construído através da análise do mercado, explorando e analisando outras soluções que através das suas lacunas, permitem obter novas ideias para a solução a ser construída no sentido de enriquecer o projeto realizado, assim como o debate de ideias e requisitos com as partes interessadas ajudaram na identificação de outros potenciais interesses que farão evoluir a respetiva solução.

2.2.4 Seleção de Ideias

A seleção de ideias tem como objetivo selecionar as ideias a perseguir no sentido de obter o maior valor de negócio. Ou seja, o problema, normalmente, não é ter novas ideias, é saber quais são as que possuem maior relevância para o projeto (A.Koen et al. 2002).

Esta seleção pode ser determinante para a evolução do projeto e para o seu sucesso, no entanto não há forma de garantir que a ideia selecionada foi a melhor. A maioria das fases de seleção de ideias é constituída por diferentes iterações de atividades que poderão incluir outras passagens pela identificação de oportunidades, análise de oportunidades, geração de

ideias e enriquecimento, normalmente com novos conhecimentos dos fatores influenciadores (A.Koen et al. 2002).

A fase de seleção de ideias é normalmente caracterizada pela utilização de métodos como o portfólio de metodologias baseado em múltiplos fatores, processo de seleção de ideias formal com *feedback* para aqueles que submeteram a ideia, e a utilização da teoria das opções, embora nenhuma delas possa ser utilizada para todos os tipos de seleção de ideias. Assim, as organizações devem determinar qual o melhor método a adotar considerando o tipo de seleção que pretendem realizar (A.Koen et al. 2002).

Com a solução apresentada pretende-se fornecer aos decisores do processo de negócio todas as informações necessárias para a definição estratégica mais correta e que irá trazer mais valor ao negócio, tendo como base dessa informação os dados armazenados e a seleção de indicadores de performance da organização. Com a análise de soluções já existentes, poderá identificar-se também ideias pré-existentes que contribuam para a introdução de fatores diferenciadores que por sua vez têm como objetivo responder às necessidades da organização.

2.2.5 Definição de Conceito

Como fase final do modelo *NCD* surge a definição de conceito. Esta é a fase em que o inovador realiza um argumento convincente para o investimento na oportunidade escolhida, conhecida por "*win statement*" ou "*gate document*". Esta fase tem por objetivo apresentar um caso de investimento que possui informação qualitativa e quantitativa utilizada convencer os clientes (A.Koen et al. 2002).

Os critérios e requisitos variam, normalmente, dependendo da natureza do tipo de conceito assim como da atitude do decisor face ao risco. Se o caso apresentado não for convincente o suficiente, o "conceito" retorna para o modelo *NCD* em vez de prosseguir para o processo *NPD* ou *TSG*, obrigando a uma nova iteração do processo (A.Koen et al. 2002). O processo de definição de conceito pode ser realizado, através das seguintes metodologias (A.Koen et al. 2002):

- Abordagens de entrega de objetivos, que corresponde ao tempo dispendido em definir cuidadosamente os objetivos do projeto e os resultados esperados;
- Definição de critérios para a organização descrevendo o que será um projeto atrativo (em termos financeiros, crescimento de mercado, dimensão de mercado, etc.);
- Avaliação rápida de inovações com elevado potencial;
- Utilização rigorosa do processo *TSG* para projetos de alto risco;
- Perceber e determinar o limite da capacidade de performance da tecnologia;
- Participação precoce do cliente em testes reais dos produtos, ou seja, envolvimento do cliente mesmo antes do produto estar completo;
- Estabelecer parcerias fora das áreas centrais de competência;
- Perseguir abordagens científicas alternativas;

Para a solução realizada, a abordagem a seguir deverá ser avaliar que inovações, mais concretamente, que funcionalidades têm maior potencial para a solução idealizada e que cumprem

as necessidades dos clientes (organizações), a definição de critérios de interesse para a organização e, a constante participação dos intervenientes do negócio no teste funcional da solução em fase de desenvolvimento da mesma.

2.3 Valor

Valor pode ser definido como aquilo que alguma coisa vale, a sua importância ou o seu mérito, podendo ter diferentes significados consoante o contexto onde seja aplicado. Para as organizações, a criação de valor é algo fundamental para o sucesso das mesmas, seja qual for a atividade de negócio. Qualquer que seja a atividade de negócio exige troca de valor, seja venda de um produto ou um serviço, valor intangível ou tangível, entre a organização e os seus clientes (Nicola, Ferreira e Pinto Ferreira 2012).

Para criar valor, uma organização precisa de perceber quais as necessidades dos clientes que pretende alcançar e adaptar-se a essas necessidades. Para conseguir alcançar este objetivo, as organizações necessitam de definir corretamente que valor pretendem criar/entregar e definir que clientes pretendem atingir, ou seja, definir o seu mercado alvo, no sentido de possibilitar a definição de estratégias para atingir esse mercado. As organizações devem assim assumir que os seus clientes não possuem o suporte necessário para perceber a tecnologia por trás do valor oferecido, no entanto devem perceber esse mesmo valor para que tenham interesse em adquiri-lo.

2.3.1 Valor Para o Cliente

Como referido anteriormente, quando um cliente adquire determinado produto ou serviço, o mesmo irá colocar valor pessoal naquilo que adquiriu. É importante que esse valor pessoal seja próximo daquele que a organização considera que seja o valor do produto/serviço (Woodall 2003).

Assim, o valor para o cliente, surge da comparação que o mesmo faz com outros produtos semelhantes, no sentido de adquirir aquele com mais valor para as suas necessidades. Se o valor que as organizações atribuírem àquilo que oferecem, não corresponder ao valor que efetivamente transmite para o cliente, a expectativa dos clientes não é concretizada, levando uma desilusão da parte do mesmo.

2.3.2 Valor Percecionado

Valor percecionado é o valor que os clientes atribuem a um produto/serviço conforme as suas necessidades, variando de cliente para cliente, ou variando mesmo no valor atribuído pelo próprio fornecedor em relação à perceção que o cliente tem do mesmo (Woodall 2003).

Esta diferença significa que muitas vezes o valor atribuído pelos clientes não depende tanto do valor monetário do produto/serviço, mas sim da capacidade do mesmo de corresponder às necessidades do cliente e expectativas definidas (Woodall 2003).

Com o presente projeto pretende-se efetivamente criar uma solução que traga valor ao cliente, desenvolvendo funcionalidades que tragam melhorias aos atuais processos de negócio da organização, desenvolvendo-as de acordo com os requisitos levantados, construindo um

sistema que facilite as operações de gestão, análise e decisão da organização, de uma forma mais automática, completa, mais inteligente e intuitiva para os seus utilizadores, constituindo uma mais-valia para os seus processos organizacionais.

2.3.2.1 Perspetiva Longitudinal de Valor

Como forma de analisar melhor o valor atribuído pelo cliente à solução, aplicou-se a metodologia de *Woodall* como forma de entender a perspetiva longitudinal de valor. Esta metodologia divide-se em quatro fases, em que cada fase pretende descrever o valor geral que um cliente atribui em cada uma das fases. As fases (Figura 3) são as seguintes: *Ex Ante*, *Transaction*, *Ex Post*, *Disposition* (*Woodall 2003*).

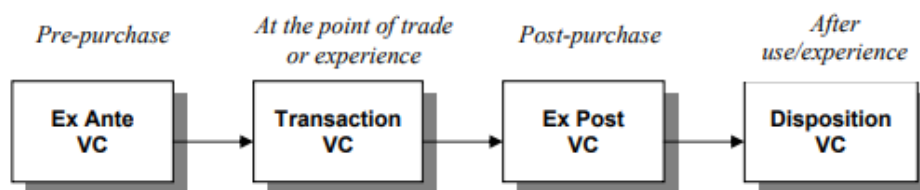


Figura 3: Perspetiva Longitudinal de Valor segundo *Tony Woodall* (*Woodall 2003*).

Numa primeira fase, *Ex Ante* (ou Antes da Compra), espera-se que a solução seja montada do forma a que espelhe o funcionamento da organização e das suas tarefas quotidianas de forma a que os seus processos se tornem mais simples e automáticos, o que se deverá tornar muito benéfico no futuro. Por outro lado, esta fase implica um grande conhecimento do funcionamento da organização e a passagem de muito conhecimento por parte do cliente/utilizador para que seja possível modelar da melhor forma o negócio em questão. Isto poderá implicar custos, seja ao nível de recursos humanos pelo tempo que terão que dispendir seja ao nível da produtividade uma vez que nesse tempo não estarão a produzir.

Após este início, vem a fase de Transição (*Transaction*). Nesta fase, ou seja, no momento da compra, o cliente deverá ter ao seu dispor, um solução adaptada ao seu contexto e ao seu negócio. No entanto, visto que é a fase onde o cliente tem o primeiro impacto com a solução, espera-se que necessite de algum tempo de habituação e aprendizagem para que os utilizadores percebam as suas mais-valias e que estas se reflitam em resultados efetivos.

Posteriormente, numa fase *Ex Post* (Depois da Compra), espera-se que passado um tempo inicial de descoberta e experiência a solução mostre resultados e que estes se traduzam em ganhos efetivos para a organização, seja pela redução de tempo gasto em tarefas quotidianas, seja pela facilidade em executar essas mesmas tarefas, uma vez que nesta fase os utilizadores deverão conhecer a solução na sua plenitude.

Depois disto, e depois de os utilizadores saberem usar a solução (fase *Disposition*), terão condições para apontar melhorias ao processo inicialmente desenvolvido e assim contribuir para uma melhoria da solução atual, sendo que até essas melhorias serem desenvolvidas terão de se adaptar as condições de que dispõem. Apesar das melhorias eventualmente registadas, estas poderão implicar de novo os problemas apresentados na fase *Ex Ante*, e por isso, a organização poderá decidir não obter essas mesmas melhorias.

2.3.3 Rede de Valor

Uma Rede de Valor pode ser definida como um conjunto de relações que gera valor, tangível ou intangível, através de trocas complexas e dinâmicas entre dois ou mais indivíduos, grupos ou organizações (Alle 2008).

Qualquer organização que se envolve em trocas pode ser vista como uma rede de valores. Assim, o valor é uma propriedade da rede e perceber como a rede funciona como um todo é essencial para perceber como o valor é criado (Alle 2008).

Porter Michael definiu o conceito de cadeia de valor (*value chain*) como sendo um conjunto de atividades que uma organização realiza no sentido de criar valor para os seus clientes. *Porter* propôs um modelo geral de cadeia de valor que as organizações podem utilizar para analisar as suas atividades e verificar como estas se relacionam (Porter 2008).

A forma como as atividades de uma cadeia de valor são executadas determina o custo e afeta os rendimentos, por isso, esta ferramenta permite às organizações perceber as fontes de valor para as mesmas.

Tanto a análise de *Verna Allee* como o modelo de *Michael Porter* permitem às organizações analisar e avaliar o seu valor. Na rede de valor de *Verna Allee*, o foco passa pela comunicação dos diferentes participantes da rede, sendo que o valor é retirado das trocas realizadas entre esses mesmos participantes. Por outro lado, no modelo de *Porter*, a análise de valor tem outros fatores em consideração uma vez que é tido em conta o custo e esforço necessário para produzir valor.

Para a solução desenvolvida neste projeto, ambos os métodos poderão ser utilizados para enriquecer a rede valor uma vez que a solução passará também pela comunicação entre os diferentes intervenientes do sistema (de forma direta ou indireta), mas principalmente pela adoção de estratégias que reduzam o tempo e os custos necessários para o aumento da produção de valor para a organização.

2.3.4 Método Multicritério Analytic Hierarchy Process

Apresentadas as formas de como o valor é importante para o cliente e de como este o entende, utilizou-se o método *Analytic Hierarchy Process (AHP)* como forma de captar as funcionalidades e requisitos com mais importância para o cliente, de forma a direcionar um maior esforço para essas mesmas tarefas através da atribuição de critérios e respetivamente dos seus respetivos pesos.

O *AHP* consiste na geração de um peso para cada critério de avaliação, neste caso para cada funcionalidade, de acordo com o seu respetivo valor para o decisor. Quanto maior o peso, mais importante é o correspondente critério (Saaty 1987).

Para calcular os pesos dos diferentes critérios, o *AHP* começa por criar uma matriz de comparação par-a-par. Esta, é uma matriz $m \times m$, onde m é o número de critérios considerados. Cada entrada a_{jk} da matriz representa a importância do critério j_{th} em relação ao critério k_{th} . Se a_{jk} for maior que 1, então o critério j_{th} é mais importante do que o critério k_{th} , enquanto se a_{jk} for menor que 1, então o j_{th} critério é menos importante do que o critério k_{th} . Se dois critérios tiverem a mesma importância, então a entrada a_{jk} é 1. As entradas a_{jk} e a_{kj} satisfazem a seguinte restrição: $a_{jk} \times a_{kj}$ é igual a 1. A importância relativa entre dois

critérios é medida de acordo com uma escala numérica de 1 a 9, como mostra a Tabela 1 (Saaty 1987):

Tabela 1: Tabela Comparativa de Critérios do Método *AHP*

Valor da entrada a_{jk}	Interpretação
1	j e k são igualmente importantes
3	j é ligeiramente mais importante que k
5	j é mais importante que k
7	j é fortemente mais importante que k
9	j é absolutamente mais importante que k
2,4,6,8	valores intermédios

Através do diálogo com o cliente chegou-se à definição dos critérios, ou funcionalidades e respetivos pesos relativos. A Tabela 2 apresenta a matriz de comparação paritária para cada critério identificado:

Tabela 2: Tabela Comparativa de Critérios Definidos Para a Solução

	Facilidade de análise de dados	Visualização gráfica de dados	Geração de documentação	Geração de Alertas/Notificações	Agilização de processos de negócio
Facilidade de análise de dados	1	2	3	4	5
Visualização gráfica de dados	1/2	1	2	3	4
Geração de documentação	1/3	1/2	1	2	3
Geração de Alertas/Notificações	1/4	1/3	1/2	1	2
Agilização de processos de negócio	1/5	1/4	1/3	1/2	1

Pela análise à Tabela 2, podemos concluir que a funcionalidade mais importante para o cliente é a facilidade em analisar os dados para a partir destes extrair conhecimento rico em valor para a organização. De seguida surge a visualização gráfica dos dados, de forma a conseguir perceber rapidamente o panorama geral da organização sem ter que efetuar cálculos matemáticos para o efeito. Em terceiro lugar está a geração de documentos de forma automática, seguida da geração de alertas ou notificações que avisem o utilizador de eventos com interesse para o negócio. Como menos importante considera-se que a agilização de processos inerentes a troca de informação, através do sistema, para com os restantes intervenientes do negócio.

Ainda assim interessa perceber a importância relativa de cada critério definido, pois no caso de realizarmos esta análise associada à seleção de sistemas que apresentem estes critérios, a importância relativa entre cada critério seria essencial para a escolha do sistema a escolher.

Para isso é necessário igualar todos os critérios a uma mesma unidade, para isto cada valor da matriz é dividido pelo total da sua respectiva coluna. A Tabela 3 apresenta os resultados normalizados dos critérios anteriormente definidos, arredondados com duas casas decimais:

Tabela 3: Matriz Normalizada dos Critério Anteriormente Definidos

	Facilidade de análise de dados	Visualização gráfica de dados	Geração de documentação	Geração de Alertas/Notificações	Agilização de processos de negócio
Facilidade de análise de dados	0,31	0,49	0,44	0,38	0,33
Visualização gráfica de dados	0,22	0,24	0,29	0,29	0,27
Geração de documentação	0,15	0,12	0,15	0,19	0,20
Geração de Alertas/Notificações	0,11	0,08	0,07	0,10	0,13
Agilização de processos de negócio	0,09	0,06	0,05	0,05	0,07

Tendo a matriz normalizada, executando a média de cada linha poderá obter-se o peso relativo de cada critério. Estes pesos são apresentados na Tabela 4.

Tabela 4: Peso Relativo de Cada Critério na Aplicação do Método AHP

Critério	Peso Relativo
Facilidade de análise de dados	0,42
Visualização gráfica de dados	0,26
Geração de documentação	0,16
Geração de Alertas/Notificações	0,10
Agilização de processos de negócio	0,06

Pela análise à Tabela 4, conclui-se então que a funcionalidade com mais peso para o sucesso da solução é a facilidade de análise a dados, seguida pela visualização gráfica dos mesmos, geração de documentação, geração de alertas e/ou notificações, e por último a agilização de processos de negócio.

2.3.5 Proposta de Valor

Proposta de valor é descrita como uma ferramenta que descreve a forma como a oferta de uma organização difere dos seus concorrentes e explica porque é que os clientes a compram (Lindic e Marques da Silva 2011).

Sendo nos dias de hoje cada vez mais importante a geração de valor, o elevado número de concorrentes põe em perigo uma organização, fazendo-a adaptar o seu processo de negócio para conseguir fazer face aos seus adversários. Assim, torna-se essencial que uma organização optimize os seus processos mas, para isso tem que conhecer muito bem a sua área de negócio e os seus clientes de modo que consiga perceber quais são as suas fontes de rendimento e que se consiga adaptar às mudanças que estas sofrem.

No âmbito deste projeto, a organização, como utilizador final do sistema que se propõe, deverá ter ao seu dispor uma solução que permitirá à mesma, efetuar um melhor exercício de gestão dos seus recursos tendo em vista o aumento dos seus índices de produtividade e rentabilidade, por via da informatização das suas ações quotidianas libertando os seus recursos humanos de tarefas manuais pouco intuitivas, morosas e repetitivas.

Assim, o conjunto de funcionalidades propostas, interligadas, deverão fornecer à organização um sistema de gestão e planeamento estratégico intuitivo, facilitando o processo de gestão da organização e permitindo, de forma simples, o acompanhamento dos seus indicadores de negócio que poderão contribuir para a evolução da organização e dos seus processos.

3. Estado da Arte

Ao longo deste capítulo será apresentado o estudo que sustenta o trabalho elaborado, começando pela análise de conceitos base para resolução de problemas similares. Serão também identificadas as métricas para a avaliação de dados de negócio, passando pela temática da informatização das atividades quotidianas de uma organização, até à escolha da melhor tecnologia e do melhor método para a criação das estruturas para o armazenamento de dados. Ainda nesta temática, serão avaliadas algumas ferramentas de análise e apresentação dos dados armazenados.

A definição das ferramentas a considerar para a execução deste projeto tiveram como critérios base o suporte e documentação fornecidas, a dimensão da comunidade que as utiliza, a similaridade com funcionalidades pretendidas, usabilidade e o tempo de aprendizagem necessário para o manuseamento das mesmas. Além destes foi também considerado o preço das ferramentas, a sua *performance* e estabilidade assim como a capacidade de se manterem atuais no tempo.

3.1 Armazenamento de Grandes Volumes de Dados

Um dos objetivos definidos inicialmente foi a capacidade da solução desenvolvida ser capaz de lidar com elevados volumes de dados. Como a solução pretende resolver um problema quotidiano de uma organização, que pelo seu uso diário gera diariamente dados que precisam de ser tratados, armazenados e apresentados aos interessados pela informação, prevê-se que o espaço necessário ao seu armazenamento possa ser bastante considerável. Além disto, a solução pretendida prevê a criação de histórico de modo a ter em conta as mudanças nas circunstâncias associadas a cada evento, uma vez que partir destes detalhes se pretende também extrair conhecimento que poderá ser fundamental para o sucesso da organização.

Visto que se pretende ter um registo com histórico dos acontecimentos de negócio, uma comum base de dados de um sistema operacional não será capaz de corresponder a este requisito de modo que terá de ser adotada uma outra solução.

Uma das soluções mais utilizadas para este propósito é a criação de um Armazém de Dados. Estas estruturas favorecem a criação de relatórios, a análise de grandes volumes de dados e a obtenção de informações estratégicas que podem facilitar a tomada de decisão de uma organização. Estas informações quando recolhidas por algum período de tempo são capazes de oferecer análises históricas que preveem tendências e auxiliam no planeamento da organização, seja a longo ou a curto prazo. Graças ao facto de um Armazém de Dados (AD) não ser volátil, este espelha sempre uma imagem do negócio num determinado momento, imagem essa que nunca é apagada ou sobreposta, excepto aquando da necessidade de uma

correção. Assim, é possível efetuar análises sobre um determinado instante de tempo tendo como contexto as características que prevaleciam na altura, tornando possível analisar as mudanças nesse mesmo contexto e determinar se elas têm impacto nos factos registados (Immon 2002).

3.2 Conceitos Base Para a Criação de Armazéns de Dados

Na presente secção, serão apresentados alguns conceitos relacionados com AD, de forma a fornecer conteúdos que permitam a perceção dos assuntos abordados relativamente ao tema.

3.2.1 Business Intelligence e Armazéns de Dados

O *Business Intelligence (BI)*, pode definir-se pela obtenção da informação certa, pelas pessoas certas, no momento ideal. O termo engloba todas as capacidades para transformação dessa informação em conhecimento, de modo a que todos os indivíduos de uma organização possam confiar nesse conhecimento, e este se traduza numa tomada de decisão mais eficaz (Bogza e Zaharie 2008).

Assim, o *BI* pode caracterizar-se pelo conjunto de atividades, técnicas e/ou ferramentas usadas sobre os dados de uma organização, dados esses que podem ter volumes muito elevados, com o objetivo de obter conhecimento sobre o seu negócio quotidiano através da análise da sua atividade passada e corrente. Para isto, é necessário que a organização armazene os dados, que serão a fonte para o conhecimento que deles será extraído, e os prepare de forma a que estes possam ser analisados pelos seus interessados com base nos indicadores que façam sentido para o negócio da organização. Estas análises terão que ser construídas com base nos dados carregados no AD, onde o utilizador deverá ter também ao seu dispor uma série de métricas que lhe permitirá analisar da melhor forma o seu negócio. De uma forma simplificada, um AD pode-se definir como um repositório de dados que passaram previamente por algum tipo de tratamento. Segundo *Bill Inmon*, um AD é a parte principal do sistema onde se insere, disponibilizando informação valiosa com o objetivo de se tornar uma ferramenta de suporte na tomada de decisões. Este, considera que é um repositório de dados centralizados para toda a organização, caracterizando-o como orientado ao assunto mediante o contexto onde está inserido, podendo no entanto disponibilizar várias estruturas de informação dentro do mesmo contexto. Este é não volátil e variante no tempo na medida que defende a inexistência de atualizações diretas sobre os registos para que se consiga ver as diferenças ao longo do tempo. Para esse efeito é sempre adicionado um registo novo relativo ao momento temporal (Immon 2002). Apesar de também o considerar como ferramenta de suporte a tomadas de decisão, Ralph Kimball defende que um AD deve ser um repositório facilmente acessível com dados tratados, consistentes e adaptáveis às mudanças permitindo a atualização direta/indireta de registos. Um AD podendo ser composto por vários assuntos, sendo que cada área tem o seu processo de negócio distinto fazendo parte do contexto onde se insere (Kimball e Ross 2013).

Uma das grandes vantagens deste processo é a rapidez com que os resultados são obtidos. Estes sistemas são preparados para processar grandes quantidades de dados, tornando a tomada de decisões mais fácil e rápida na medida em que o utilizador escolhe que tipo de análises pretende fazer aos dados e os resultados surgem de imediato. Através deste

processo, torna-se possível ter uma visão concreta e fiável sobre o estado da organização, uma vez que os dados que entram no sistema sofreram o tratamento devido, permitindo a visualização rápida e clara dos acontecimentos. Esta, pode ser considerada uma ferramenta muito útil para os responsáveis de empresas, permitindo uma rápida evolução para a mesma não só a nível de tomada de decisões futuras, mas também a nível de análise do comportamento dos seus potenciais concorrentes e clientes, identificando possíveis melhorias nos produtos e segmentos de mercado ainda não explorados pela mesma.

3.2.2 Armazenamento de Dados

No seu quotidiano, as organizações, à medida que efetuam as suas operações de negócio, deparam-se com a necessidade de registar esses mesmos acontecimentos. Estes acontecimentos geram dados, dados esses que precisam de ser armazenados em sistemas ou repositórios de dados especializados, permitindo que estes possam ser preservados. Posteriormente, esta informação poderá tornar-se valiosa, seja pelo interesse da sua simples consulta ou pelo eventual conhecimento que deles possa ser extraído.

Dependendo do tipo de características existentes e do contexto em que se insere, a estrutura do armazenamento pode ser distinta. De uma forma geral, existem dois tipos de sistemas: sistemas orientados à transação, sistemas *Online Transaction Processing (OLTP)* e sistemas orientados a análise/assunto, sistemas *Online Transaction Processing (OLAP)* (OLTP vs. OLAP 2009).

Os sistemas operacionais, ou *OLTP*, têm como base uma arquitetura orientada à transação, isto é, têm como objetivo registar todas as transações efetuadas num determinado momento e em cada domínio. As bases de dados possuem uma estrutura relacional normalizada, onde os dados são armazenados em tabelas de acordo com o contexto e relacionadas entre os restantes artefactos existentes, criando dependências entre si. Desta forma, o número máximo de dependências que podem existir em determinado contexto, e a quantidade de dados adjacente, pode fazer com que este tipo de sistemas possa sofrer grandes perdas de performance na consulta desses mesmos dados (Ilan 2017).

O outro sistema utilizado quando existe a necessidade de grande armazenamento de dados e respetiva análise é o AD. Este é um sistema computacional capaz de armazenar grandes quantidades de dados e manter a performance nas consultas devido à sua estrutura desnormalizada e com poucas dependências entre os dados. Armazena todo o conjunto de dados em modelos multidimensionais/dimensionais, constituídos por dimensões e tabelas de factos. Neste tipo de modelo, as dimensões armazenam todos os dados que pertencem ao seu domínio (Ballard et al. 2006).

Assim, por sistema OLAP pode definir-se um conjunto estruturas de informação criadas a partir do modelo dimensional do AD, onde podem conter a informação armazenada de forma agregada, contabilizando um conjunto de factos e medidas que se pretende de acordo com o contexto. Em alguns casos, as consultas de dados passam a ser feitas diretamente sobre este tipo de estruturas, libertando alguma carga sobre o AD.

Cada um dos tipos de armazenamento enunciados possuem as suas especificidades. Devido ao processo prévio de limpeza e tratamento de dados (técnicas de Extração, Transformação e Carregamento (ETL)), o AD permite despistar em grande alcance as inconsistências, bem como definir qual a melhor resolução do problema perante o caso em questão. No sistema operacional, o utilizador é que necessita de tomar a iniciativa de limpeza das inconsistências

de forma manual. Não possui um processo estruturado que faça parte de um possível carregamento propriamente dito, ficando assim ao critério do responsável pelo sistema garantir a integridade e consistência dos dados.

No caso de um o AD, o acesso à informação nas análises pode atingir tempos de resposta muito expectantes, ao contrário do que acontece com os sistemas operacionais. Neste tipo de sistemas, como já foi referido, as consultas efetuadas pelo consumidor podem não ser efetuadas diretamente sobre o sistema de armazenamento. Depois do sistema se encontrar limpo e carregado, é criada uma camada de abstração para que todas as consultas sobre elas incidam. No caso dos sistemas operacionais, todas as interrogações e consultas que se efetuam ao sistema, são inteiramente efetuadas diretamente. Apesar da execução de análises sobre o AD obter bons resultados, a desvantagem aparece no carregamento/tratamento de dados. Para este sistema e respetiva manutenção, este é um processo lento e trabalhoso. O AD permite que o carregamento de dados seja efetuado a partir de fontes de dados de tipos diferentes, nomeadamente ficheiros *Excel*, ficheiros de texto, bases de dados, etc.

Na Tabela 5 apresentam-se as principais diferenças entre um sistema OLAP e um sistema OLTP (Alves 2013).

Tabela 5: Comparação entre sistemas OLAP e OLTP.

	OLAP	OLTP
Fontes de dados	Sistemas do tipo dimensional e/ou relacional	Sistemas operacionais
Finalidade dos dados	Tornar mais eficaz o processo de tomada de decisão e planeamento	Controlar e executar as operações de negócio
Complexidade das <i>queries</i>	Pode ser bastante complexa	Maioritariamente simples
Velocidade de processamento	Pode variar de acordo com o volume de dados	Processamento rápido
Espaço necessário	Muito	Relativamente pouco
Modelo da Base de Dados	Multidimensional	Relacional
<i>Backup</i> e Recuperação	Raramente	Frequentemente
Idade dos dados	Contém histórico	Dados correntes
Operações mais efetuadas	Maioritariamente leitura	Inserção, atualização, leitura e remoção
O que mostram os dados	Múltiplas vistas sobre os vários tipos de atividades de negócio	Exibe uma imagem do negócio corrente

Para implementar um sistema de armazenamento, é necessário ter em conta as necessidades do meio e o benefício/custo que cada tipo de solução terá e escolher qual a que melhor se ajusta ao caso, não existindo portanto uma solução ideal para todos os casos.

3.2.2.1 Data Mart

Um *Data Mart* é uma forma de divisão lógica de um AD, sendo este mais pequeno, otimizado para servir como base para a tomada de decisões de uma determinada área de negócio (por exemplo: Vendas, Compras, Inventários, etc.). Um AD pode ser dividido/composto por

várias estruturas deste género, tornando-se mais fácil de gerir e manter na medida em que as operações necessárias a serem efetuadas apenas incidem num determinado domínio, mantendo os restantes operacionais e com impacto reduzido (Ballard et al. 2006). Estes podem ser classificados como dependentes ou independentes entre si de acordo com o seu nível de dependência. São dependentes quando estes são parte integrante de um AD, ou seja, significa que a estrutura do armazém é definida por todos os *data marts* nele contidos. Por outro lado, podem ser também independentes quando o AD possui a sua própria lógica e estrutura, sendo que apenas possui ligações com *data marts* externos, sendo assim cada um deles independente do contexto do AD (Oracle 2005). Na secção a seguir será possível perceber os seus detalhes.

3.2.3 Online Analytical Processing

OLAP é o mecanismo de análise de sistemas multidimensionais que possibilita realizar operações complexas e sofisticadas sobre os dados. Este permite aos utilizadores finais realizar pesquisas em múltiplas dimensões ao mesmo tempo, disponibilizando-lhes a informação de que necessitam para tomarem as suas decisões de negócio com mais eficácia. A vantagem de utilizar mecanismos OLAP está na velocidade de acesso à informação armazenada no modelo multidimensional, criando agregações e cálculos muito rapidamente em múltiplos conjuntos de dados. A implementação deste tipo de estrutura depende não só do tipo de software que se está a utilizar mas também do tipo das fontes de dados e dos objetivos do negócio em que se insere (Almeida 2017).

Quanto à sua estrutura, este tipo de metodologia tem como base um cubo de dados dividido em pequenos segmentos (cada segmento representa um determinado valor), em que cada vértice é referente a um ramo de informação específico (dimensão). Na Figura 4 é apresentado o exemplo de um cubo, que relaciona a informação da região, do produto e do dia em que foi adquirido (Suri 2016).



Figura 4: Cubo OLAP.

Para obter a informação pretendida, existe um conjunto de operações que são habitualmente feitas no contexto dos sistemas OLAP sobre esta estrutura de dados que consistem em combinar os valores das dimensões, atuando como filtro para obter a informação desejada. As operações são as seguintes (Nico 2017):

- **Drill-down:** este tipo de operação baseia-se numa procura de dados progressivamente mais restritiva, elevando-se a granularidade da pesquisa para obter informação cada vez mais detalhada a cada nova procura efetuada. Um exemplo seria progredir de uma pesquisa de vendas de dispositivos móveis por períodos de tempo diferentes: no ano

corrente, nos primeiros três meses do ano, num mês em específico, numa semana ou até no dia corrente;

- **Roll-up:** descreve-se como sendo a operação contrária à anteriormente enunciada. Parte-se de um nível de granularidade mais específico, diminuindo-a progressivamente. Neste contexto, o filtro começaria pelo dia, seguido do mês e terminaria no ano;
- **Drill-across:** tipo de operação que incide em dados que tem como origem mais do que uma tabela de factos e que se relacionam a partir das ligações às dimensões em comum. Um exemplo seria a existência da tabela de factos para o processo de vendas de produtos e uma outra tabela de factos para o processo de queixas associadas a determinado produto. Teriam as três dimensões presentes como comuns, dado que a queixa é registada em determinado dia, é associado a determinado produto numa determinada região.
- **Slice:** este tipo de operação cria um novo cubo de informação segmentado, baseando-se em apenas um filtro de uma das dimensões. Ao pesquisar por todas as vendas de produtos da categoria dispositivos móveis, é obtido um conjunto de informação específico com apenas um filtro;
- **Dice:** nesta operação também é criado um novo cubo à semelhança da operação descrita anteriormente, à exceção de que são utilizados dois ou mais filtros. Um exemplo deste tipo de operação seria pesquisar por todas as vendas de produtos do tipo dispositivos móveis de um determinado ano;
- **Rotation:** baseia-se na rotação dos eixos do(s) cubo(s), com o objetivo de permitir efetuar pesquisas em diferentes perspetivas.

Existem três tipos de estruturas distintas para o armazenamento dos dados, denominadas por Multidimensional Online Analytical Processing (MOLAP), Relational Online Analytical Processing (ROLAP) e Hybrid Online Analytical Processing (HOLAP). No MOLAP, os dados estão armazenados num cubo multidimensional, proporcionando a rapidez na obtenção dos dados e nas operações de *slicing* e *dicing* em cubos. Por sua vez, o ROLAP consiste em efetuar análises em dados armazenados num modelo relacional e efetuar as operações como no MOLAP, com a diferença de que todas as interrogações são feitas diretamente ao AD. Por fim, o HOLAP consiste na junção do MOLAP e ROLAP, sendo que o HOLAP aproveita a tecnologia do cubo para um processamento mais rápido (*MOLAP, ROLAP, And HOLAP 2015*).

3.2.4 Arquiteturas de Armazéns de Dados

Um AD, é normalmente desenvolvido para dar resposta a uma necessidade específica, tendo portanto uma estrutura particular e organizada de forma a dar resposta às características do meio no qual se insere. A presente secção servirá para apresentar as duas principais abordagens seguidas no contexto da criação de um AD, sendo estas defendidas por *Ralph Kimball* e *Bill Inmon* (Breslin 2004).

3.2.4.1 Arquitetura de Ralph Kimball

Segundo *Ralph Kimball*, um AD encontra-se dividido em duas áreas lógicas, que podem ou não ser também separadas fisicamente: a camada de implementação (*The Back Room*) e a camada de apresentação (*The Front Room*). Resumidamente, na primeira encontra-se toda a lógica criada incluindo ligações às fontes de dados (*Source Systems*), processos para extração, tratamento e processos desses mesmos dados para armazenamento num sistema de base de dados de *staging* (*The Staging Area*). Esta área de *Staging* inclui também o AD, que vai ser carregado a partir dos dados, depois de efetuado o processo de limpeza e consequente tratamento de dados. Até aqui, o acesso era interdito por parte da camada de apresentação (*The Presentation Area*), sendo que os dados passam a estar disponíveis para os utilizadores finais através da camada de apresentação que engloba todas as aplicações de suporte ao uso e análise dos dados presentes nas estruturas apresentadas anteriormente (Kimball e Ross 2013). A Figura 5 ilustra as divisões descritas.

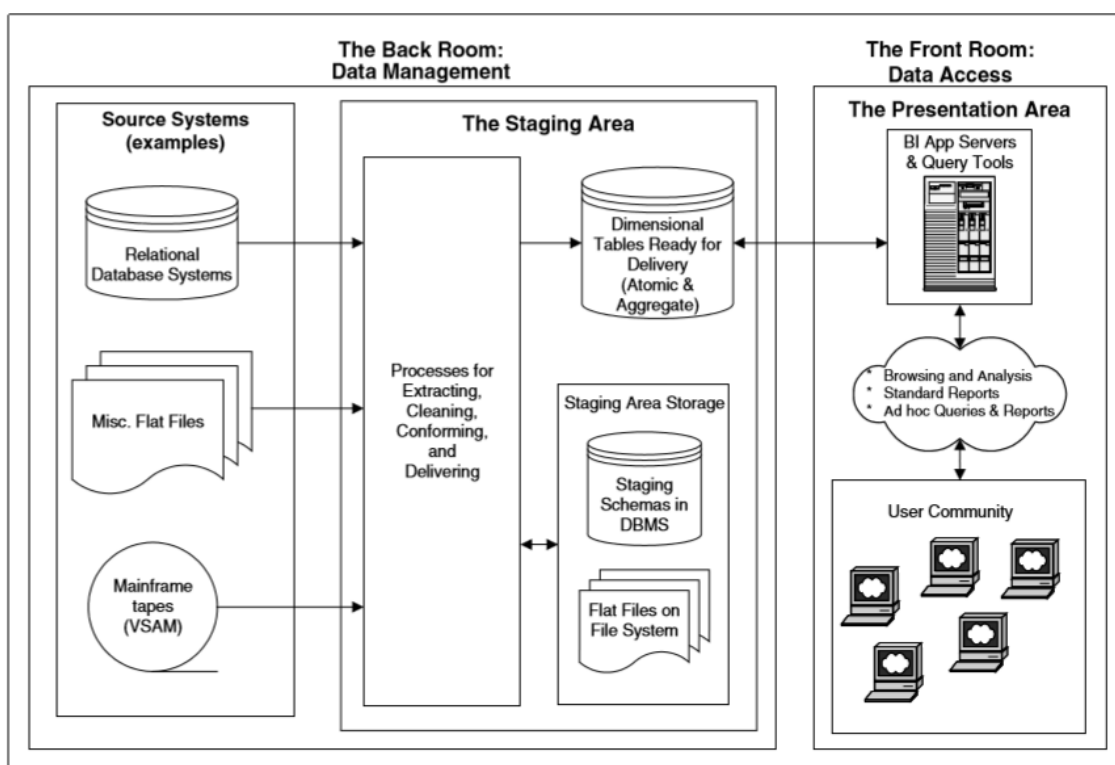


Figura 5: Divisões lógicas de um AD segundo *Ralph Kimball* (Kimball e Ross 2013).

Na primeira camada, o processo começa a partir da extração dos dados das fontes de dados. Estas fontes, possuem os dados que servirão à posteriori para alimentar o AD, podendo assumir formatos variados desde um ficheiros de texto (ou um conjunto de ficheiros) ou a extração a partir de uma base de dados operacional, base de dados essa que poderá ser no limite a utilizada para a gestão do negócio. Então, é feita a extração dos dados em bruto, de uma ou mais fontes, que poderá ser desde logo armazenada. Este armazenamento inicial poderá ser bastante útil, facilitando o recomeço do processo de carregamento sem ter a necessidade que sobrecarregar novamente o sistema na extração seguinte. Depois de extraídos os dados, estes passam para uma área denominada por *Staging Area*. Esta, definida como uma área de limpeza e tratamento dos dados, sendo composta por um processo de extração,

tratamento, carregamento (processo de ETL) e por um sistema de armazenamento interno invisível para o utilizador, com o fim de armazenar os dados tratados antes de passar para o AD final. Após estes dados serem carregados em *staging*, recebem o último tratamento antes do armazenamento no modelo multidimensional, sendo este denominado por *Enterprise Bus Architecture*. Estes dados encontram-se na forma mais atómica possível para permitir dar resposta às mais variadas possibilidades de *query's* sobre os dados armazenados. Além disso, e porque os dados se encontram armazenados na sua forma atómica, é também possível efetuar *query's* com base na agregação dos dados (*rollup*) ou conseguir ir a um maior nível de detalhe (*drill-down*). Depois de estes dados serem inseridos no AD, podem ser feitas manutenções a nível de histórico e atualizações aos registos (Kimball e Caserta 2004).

Esta arquitetura defende uma metodologia do tipo *bottom-up*, considerando que a criação de valor para o AD pode ser incrementada ao longo do tempo através de novos *data marts*, acrescentando assim novos domínios de informação. Não requer que todos estes domínios sejam criados logo na fase inicial, isto faz com que nessa fase inicial o AD possa ser desenvolvido de uma forma mais rápida e menos dispendiosa. Poderão haver casos onde este pormenor poderá ser essencial para que um pequeno desenvolvimento possa comprovar as vantagens de um *data mart*, e com isso validar a implementação de todas as vertentes do AD desejado.

3.2.4.2 Arquitetura de Bill Inmon

Inmon por sua vez defende que, todos os dados operacionais que existem numa organização são parte integrante do AD. Este tipo de arquitetura assenta numa metodologia do tipo *top-down*, onde numa primeira fase se procuram identificar todos os domínios de dados existentes, com o objetivo de criar relações lógicas entre si, resultando assim num modelo de dados essencialmente relacional. Desta forma, a estrutura total do AD é definida e só depois possivelmente se dividirá em diferentes *data marts*, incluindo toda a estrutura numa área denominada por Corporate Information Factory (CIF) (*Inmon, Strauss e Neushloss 2010*).

Este tipo de arquitetura possui algumas especificidades, distintas da arquitetura apresentada anteriormente. O AD é orientado ao assunto, ou seja, a informação encontra-se organizada de acordo com as suas relações, resultando numa base de dados operacional normalizada na terceira forma normal. Estas relações existem dado que esta arquitetura defende que todos os dados que possuam o mesmo domínio de informação devem ser relacionados. Neste sistema, não existe qualquer possibilidade de atualização aos dados. Cada registo é preservado e guardado tal e qual como foi inserido para efeitos de análise futuras, registando-se apenas o momento de inserção para existir a perceção da variável tempo perante os restantes dados. Por último, este tipo de AD é também integrado, dado que pode conter informação de mais do que um sistema relacional existente, tornando a informação nele contida consistente por seguir uma metodologia igualmente relacional. Como apresenta a Figura 6, a primeira camada surge como forma de recolher os dados que alimentarão o AD visto que estes poderão ser provenientes de várias fontes. A nível de extração, tratamento e carregamento de dados, tal como na arquitetura apresentada anteriormente, é utilizada uma área de *staging* para suportar os processos de ETL nesta fase efetuados. Toda a informação passa desta área diretamente para o AD, sendo a partir deste sistema que cada *data mart* é logicamente criado e carregado (*Oracle 2002*).

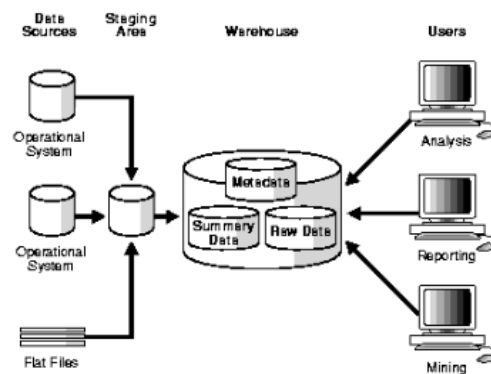


Figura 6: Divisões lógicas de um AD segundo *Bill inmon*.

3.2.4.3 Comparação Entre Arquiteturas

Na Tabela 6, é feita uma comparação entre as duas arquiteturas anteriormente enunciadas, de modo a tentar perceber as diferenças, semelhanças, vantagens e desvantagens de cada uma das abordagens propostas.

Tabela 6: Comparação entre arquiteturas de *Kimball* e *Inmon* (Kimball e Caserta 2004) (Kimball e Ross 2013) (Immon 2002).

	Arquitetura de <i>Kimball</i>	Arquitetura de <i>Inmon</i>
Composição do AD	O AD é composto por vários <i>data marts</i> em que cada um é responsável por um segmento no global do negócio. Todos os <i>data marts</i> resultam assim no AD.	Os dados são parte integrante do AD. É definida uma estrutura global e só depois, se existir essa necessidade, podem ser criados segmentos à parte.
<i>Staging area</i>	Faz parte da arquitetura	Faz parte da arquitetura
Processo de <i>ETL</i>	Faz parte da arquitetura	Faz parte da arquitetura
<i>Data marts</i>	Inicialmente são definidos cada um dos domínios de dados, dando origem a <i>data marts</i> distintos. Só depois é que o AD é definido (abordagem <i>bottom-up</i>).	A estrutura do AD é conhecida à partida. Apenas se necessário, esta estrutura pode ser segmentada em <i>data marts</i> distintos (abordagem <i>top-down</i>).
Mudança ao longo do tempo	Faz parte da arquitetura	Faz parte da arquitetura
Modelo de AD	Segue a modelação multidimensional.	Essencialmente relacional (terceira forma normal).
Orientação aos processos	Orientado ao assunto.	Orientado ao processo
Complexidade de desenvolvimento	Simples, cada estrutura de dados existente é pensada e segmentada logo na fase da construção do AD. Modelo com poucas dependências, baseando-se no geral em relações entre dimensões e tabelas de factos.	Complexa, dado que primeiramente se constrói uma estrutura global para todo o tipo de informação da organização. As relações entre dados podem ser complexas podendo prejudicar a performance a nível de pesquisas de dados, tende a piorar se a dependência entre dados for muita.
Registo de alterações nos dados	Faz parte da arquitetura	Não faz parte da arquitetura
Tempo de desenvolvimento	Menor	Maior
Custo	Menor custo de desenvolvimento inicial. Cada segmento que seja construído numa fase posterior terá aproximadamente o mesmo custo.	Maior esforço inicial. Os desenvolvimentos seguintes terão menor custo de desenvolvimento.
Conhecimentos requeridos	Não são requeridos conhecimentos especialistas, apenas generalistas.	Equipa especializada, dada a complexidade do modelo.

Em suma, cada uma das arquiteturas possui as suas especificidades, especificidades essas que podem ser ambas válidas dependendo do contexto e do âmbito em que são utilizadas,

não existindo uma melhor opção para todos os casos. Aquando da escolha da arquitetura deve então ter-se em conta o contexto e os objetivos desejados, podendo-se optar por uma solução mais progressiva (abordagem de *Kimball*) ou uma solução mais completa na fase de criação (abordagem de *Inmon*).

3.2.5 Modelação Dimensional

A modelação dimensional é uma das técnicas mais utilizadas e mais importantes para modelar um AD (Piton 2019). Atualmente os modelos dimensionais constituem uma base para o armazenamento e de gestão de dados nas soluções de BI. Estes modelos definem o relacionamento dos dados, criando um suporte a consultas em todas as vertentes de negócio em questão, assim como a extração de detalhes sobre esses mesmos dados. Assim, torna-se mais fácil compreender toda a informação de uma organização, esperando-se que com este modelo seja possível assimilar o conhecimento sobre o negócio de uma forma mais simples e eficaz.

Este tipo de modelo segue uma estrutura de dados e uma forma de desenvolvimento do sistema de armazenamento muito específica, distinguindo-se claramente do modelo relacional que é baseado no cumprimento das formas normais. A modelação dimensional defende assim existência de uma ou mais tabelas que registam todas as ocorrências na forma mais resumida possível (tabela de factos), criando tabelas distintas associadas que especificam cada um dos domínios de informação (tabela de dimensão). Pretende-se então identificar o que é necessário registar em cada evento e a informação detalhada que pode ser aglomerada em tabelas diferentes, permitindo que todos os registos que possam vir a ser armazenados se possam relacionar com outros eventos.

As tabelas de factos têm como objetivo armazenar registos associados a operações ou eventos de um determinado contexto previamente identificado. Assim, para cada evento são definidos os campos que o descrevem e que têm interesse para uma posterior análise, assim como a granularidade pretendida, identificando o que se pretende medir numericamente com a informação e qual o nível de detalhe pretendido. Imagine-se um evento no contexto da venda de produtos. De forma geral, sempre que um comerciante efetua a venda de produtos a clientes terá posteriormente interesse em analisar detalhadamente a quantidade de produtos vendidos, o tipo de produto, o valor total da venda, o número de produtos adquiridos, o peso, etc. Posto isto, esta informação terá que ser armazenada na tabela de factos. A tabela de factos contém medidas numéricas resultantes das medidas operacionais utilizadas no mundo real. Com menor granularidade, uma linha da tabela de factos corresponde a um evento e vice-versa (Kimball e Ross 2013).

No entanto, uma venda não é apenas caracterizada pelo produto e pelo comprador em questão pois interessa ter em consideração outros aspetos do evento, como por exemplo a data em que se registou a venda, em que loja ou local se efetuou. Com o registo deste detalhe, e dividindo a informação em vários domínios, cria-se a possibilidade de relacionar, por exemplo, o mesmo cliente ou o mesmo local de venda com mais do que um ato de venda sem que toda a informação detalhada sobre o mesmo se repita e que seja identificada de uma forma simples. Estas estruturas relacionáveis denominam-se por tabelas de dimensão. As dimensões fornecem o contexto "quem, o quê, onde, quando, porquê e como" relacionado com um evento de todo o conjunto dos processos de negócio. As tabelas de dimensão contêm os atributos descritivos utilizados por aplicações BI para filtrar e agrupar os dados (Kimball e Ross 2013).

Uma tabela de dimensão consiste assim, em adicionar o contexto aos eventos existentes numa tabela de factos. Ainda sobre o exemplo apresentado, o domínio cliente teria toda a informação detalhada sobre o mesmo desde o nome, morada, número de identificação fiscal, entre outros tipos de informação possíveis. Este tipo de tabela contém uma relação com a tabela de factos através de uma chave natural (denominada *surrogate key*), onde cada chave identifica de forma unívoca o registo, relacionando-o com cada facto da tabela de factos. Este tipo de tabelas, como são desnormalizadas, possuem um conjunto de campos maior e com menos quantidade de registos armazenados.

À medida que o modelo vai sendo criado vai assumindo uma estrutura muito específica sobre o contexto em que se insere. Essa estrutura pode assentar num dos seguintes tipos de modelos possíveis (Ballard et al. 2006):

- **Modelo em estrela (star-schema)**: é composto por uma única tabela de factos e por um conjunto de dimensões todas relacionadas;
- **Multi-estrela (multi-star model)**: é composto por mais do que uma tabela de factos, ligadas indiretamente através das dimensões que possuem ligação comum;
- **Modelo em floco de neve (snowflake model)**: é composto por uma tabela de factos e por dimensões onde, para além de se relacionarem com a tabela de factos podem relacionar-se entre dimensões.

Como referido anteriormente, estes modelos são compostos por dois tipos de estruturas: tabelas de factos e tabelas dimensões. Estas, foram evoluindo ao longo do tempo, tendo sido criados dentro destes dois grupos específicos, cujo propósito é resolver problemas particulares, servem assim como tratamento padrão para cada tipo de problema. A seguir apresentam-se alguns tipos de tabelas de factos e dimensões.

Tabelas de Factos

- **Tabelas de factos transacionais**: tabela que regista um conjunto de acontecimentos, em que cada linha representa um evento particular;
- **Tabelas de factos agregadas (Agregated fact table)**: tabela que regista a informação armazenada na tabela de factos mas de uma forma agregada, definindo-se no carregamento os níveis de informação pretendidos. O objetivo é reduzir o tempo gasto ao nível de execução de consultas.
- **Tabelas de factos periódicas (Periodic Fact Table)**: tabela que regista factos provenientes de um determinado intervalo de tempo. Contém factos proveniente de uma tabela de factos transacional, mas apenas sobre determinado período de tempo;
- **Tabelas de factos acumulativas (Cumulative Fact Table)**: tabela que regista todos os dados desde que o processo de negócio se iniciou até ao seu fim. Cada registo representa um processo que poderá ser atualizado ao longo do tempo;
- **Factless fact tables**: tabela que regista informação vinda de diferentes dimensões, através da chave de relacionamento. Caracteriza-se normalmente por não conter medidas sobre o evento, mas por apenas juntar informação sobre as dimensões associadas aos factos.

Tabelas de Dimensão

- **Dimensão conforme (Conformed dimensions):** tem como objetivo agrupar um conjunto de informação sobre um domínio específico, podendo este tipo de estruturas ter relacionamentos com várias tabelas de factos ao mesmo tempo e em *data marts* diferentes;
- **Dimensão degenerada (Degenerate dimension):** tem como objetivo armazenar, junto com o facto, a que conjunto de registos pertence. Tendo como exemplo uma tabela de factos para o registo de encomendas, a tabela de factos que armazena linhas de encomenda, para cada linha existente seria adicionado um campo com o identificador da encomenda. Ao fazer a pesquisa pelo identificador da encomenda, é retornado o conjunto de linhas de encomenda associado;
- **Junk Dimension:** tem como objetivo registar informação que, apesar de não constituir um domínio próprio em específico, repete-se entre registos na tabela de factos.

Em suma, a técnica utilizada para definir o modelo de armazenamento de dados depende do contexto em questão. O tipo de estruturas a utilizar prende-se maioritariamente com o negócio e o objetivo do AD em construção, pelo que o uso destas estruturas por si só não irá garantir o sucesso da sua implementação. No entanto, o uso destas estrutura em conjunto com um bom equilíbrio entre o nível de relacionamento entre os dados e o volume desses mesmos dados, poderão determinar a performance que se poderá vir a obter na camada de consulta e apresentação de dados.

3.3 Análise de Mercado

Visto que não foi possível encontrar um exemplo na mesma área de negócio onde este projeto se insere, irá apresentar-se uma solução mais geral na área do retalho, mais propriamente um exemplo de AD ao nível de uma rede de supermercados (Oklahoma s.d.).

Com este exemplo, pretende-se retirar a base de um AD na área do retalho. Assim, adotando este modelo, o AD a desenvolver deverá ser capaz de resolver alguns problemas da mesma natureza daqueles que são abordados neste projeto, contribuindo assim para uma solução mais geral e que poderá ser adaptada a problemas, não só na área da venda de material ótico mas como numa variedade de finalidades no âmbito do mercado retalhista. O exemplo apresentado destaca-se pela sua baixa complexidade de compreensão/uso e alta generalidade.

Serão também apresentados alguns exemplos de indicadores de performance utilizados na área do retalho, sendo que os indicadores a ser utilizados no decorrer do projeto serão abordados na subsecção 4.3.2.

3.3.1 Modelo Básico: Supermercado

Um supermercado armazena uma série de produtos, produtos esses que podem ser comprados a fornecedores externos, e que geralmente possuem códigos de barras associados, gerados pelo fornecedor. Muitos produtos embalados internamente, como a carne, também possuem códigos de barras, mas estes gerados pela própria instituição. Alguns produtos, na sua maioria produzidos, têm os seus preços definidos no sistema de venda ou faturação da organização.

O departamento de gestão deste supermercado está preocupado com a faturação, *stock* e venda de produtos para maximizar o lucro. O lucro vem da cobrança do maior valor monetário possível para um produto, reduzindo custos e aumentando o volume das vendas. Para aumentar o valor gerado, a administração quer um AD para apoiar decisões sobre preços e promoções.

Para a definição do AD segue-se a seguinte modelação:

- **Área de negócio:** Vendas de produtos a clientes;
- **Granularidade:** Vendas por produto, por instante de tempo (dia), por loja e por promoção;
- **Dimensões:** Data, Loja, Produto, Promoção;
- **Factos:** Quantidade vendida, valor da venda em dólares, valor de custo em dólares, lucro bruto em dólares;

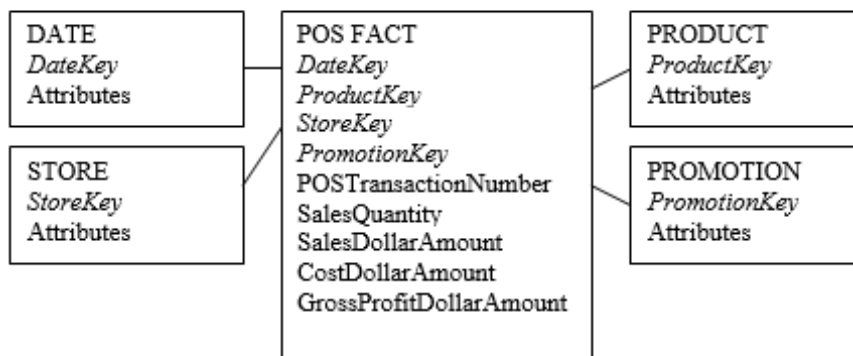


Figura 7: Modelo Dimensional - Supermercado.

A Figura 7 representa o esquema de modelação dimensional em estrela uma vez que todas as dimensões se ligam diretamente à tabela de factos. A utilização deste modelo permite efetuar análises às vendas de produtos, assim como às suas promoções associadas, por loja e data de venda.

Modelo para Snapshots de Inventário

Níveis de inventário/*stocks* otimizados são fundamentais para as operações de gestão na área do retalho. Desta forma, o revendedor precisa de saber e ser capaz de analisar a quantidade diária de produtos que tem em *stock* por loja.

- **Área de negócio:** *Stocks* de um supermercado;
- **Granularidade:** *Stocks* diários por produto e por supermercado;
- **Dimensões:** Data, Produto e Supermercado;
- **Factos:** Quantidade em *stock*.

A Figura 8 representa o esquema de modelação capaz de resolver o problema da gestão de inventários de uma organização. A utilização deste modelo permite efetuar análises aos produtos em *stock*, diariamente por cada loja de uma rede de supermercados.

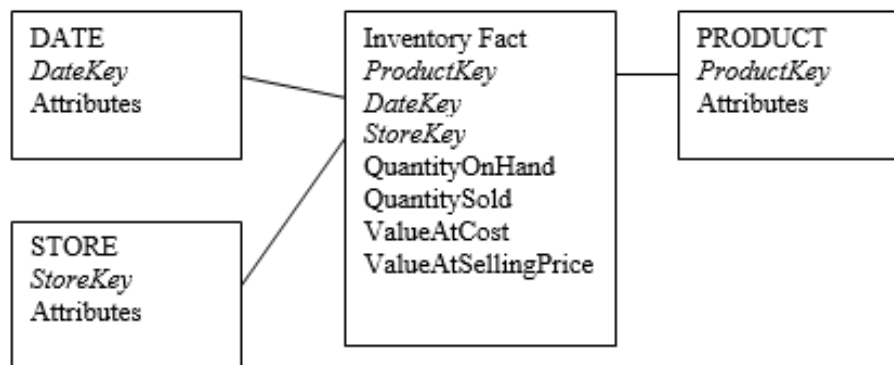


Figura 8: Modelo Dimensional - *Snapshot* do Inventário de um Supermercado.

Slowly Changing Dimensions

Uma dimensão está suscetível a alterações, quer sejam correções a dados introduzidos por engano, ou alguma mudança que ocorra no sistema operacional. Existe então o conceito de Slowly Changing Dimensions (SCD), que especifica uma estratégia para suportar essas alterações.

Existem três tipos de SCD:

- **Tipo 1:** Altera o valor;
- **Tipo 2:** Insere um novo registo e “desativa” o antigo;
- **Tipo 3:** Adiciona uma coluna para o atributo em questão;

O tipo 1 não permite guardar histórico, e deve ser utilizado quando apenas se quer fazer correções aos dados que não influenciem os dados introduzidos anteriormente. O tipo 2 e 3 permitem guardar histórico, sendo que o tipo 3 permite fazer consultas como se a alteração não tivesse ocorrido enquanto que o tipo 2 apenas permite consultas de acordo com uma data de início e de fim, existindo apenas um "registo atual" para um determinado intervalo de tempo.

3.3.2 Indicadores Chave de Performance

Sendo o sistema proposto uma solução de informatização e posterior análise de grandes quantidades de dados na área do retalho, o sistema será constantemente usado para o registo de valores com interesse para o negócio, sejam eles receitas, lucros, vendas, etc. Dada a quantidade destes dados torna-se difícil saber quais são os dados relevantes e que se devem ter em consideração para definir estratégias para o negócio.

É nesta lógica que surgem os Indicadores Chave de Performance, ou *Key Performance Indicator (KPI)*. *KPI* é, uma medida que permite a uma organização, projeto ou indivíduo, classificar o seu desempenho em relação às suas metas e objetivos estratégicos definidos (Marr 2017).

De uma forma geral, os *KPI's* fornecem informação de desempenho importante que permite às organizações (ou partes interessadas) entender se a esta está ou não em progresso em relação aos objetivos definidos. Desta forma, *KPI's* bem projetados são instrumentos de

navegação vitais, dando uma imagem clara dos níveis atuais de desempenho e se o negócio é aquilo que se espera ser. Os *KPI's* também são ferramentas de tomada de decisão muito úteis. Estes ajudam muitas vezes a reduzir complexidade do desempenho organizacional para um número pequeno de indicadores chave facilitando muito a sua gestão (Marr 2017).

Cada negócio é diferente, existindo uma infinidade de *KPI's* associados a cada caso particular. A seguir, apresentam-se aqueles que Nicasio e Stanley 2018 consideram ser os mais significantes na área do retalho.

- **Vendas por metro quadrado:** mede o quão eficiente uma organização é na gestão do espaço de vendas.

Consiste simplesmente na divisão do montante vendido pelo espaço, em metros, de vendas. Poderá ser bastante útil na determinação de localizações mais rentáveis, usando-as por exemplo para armazenamento de *stock*.

- **Número de pessoas em loja:** refere-se ao número de clientes que entram numa loja num determinado período de tempo.

Essas ferramentas não contam apenas o número de pessoas na sua loja, podem também ser usadas para analisar o comportamento do comprador, tempo de permanência e outras métricas.

- **Taxa de conversão:** é a percentagem de clientes que efetivamente comprou algo. Pode calcular-se dividindo o número de vendas pelo tráfego bruto de pessoas. Ajuda na medição de algumas características do negócio, incluindo o serviço ao cliente, ou a eficácia do *merchandising*.

- **Contagem de vendas:** refere-se ao número de transações completas.

O número de transações é uma métrica fundamental que informa quantas vendas foram feitas em determinado período de tempo. Pode ser usada para avaliar o marketing da loja, atendimento ao cliente, experiência do cliente, entre outras.

- **Média de valor de transação:** diz o gasto médio de um cliente. Pode calcular-se dividindo a receita total pelo número de transações.

O valor médio por transação pode dar uma visão macro de quantas pessoas estão a comprar artigos, bem como os tipos e quantidade de artigos que compram. Um valor alto pode significar que os compradores estão a comprar os produtos mais caros ou que estão a comprar quantidades maiores. Por exemplo, ter um valor médio baixo por transação, pode indicar que o preço dos artigos precisa de ser repensado. Pode também significar que é necessário implementar novas táticas de vendas, como pacotes de descontos ou outras ofertas que os compradores valorizem.

- **Margem de lucro:** é a proporção do lucro por receita total. Diz a quantidade de receita obtida deduzindo os custos associados aos dos bens vendidos.

É essencial para a percepção de valor real de lucro obtido no negócio.

- **Taxa de stock:** mede a "velocidade" com que é vendido o stock.

Esta métrica permite ter um melhor controlo sobre o inventário para que se possam fazer decisões de compra mais inteligentes, manter a mercadoria em movimento e vender mais dos produtos que os seus clientes mais desejam.

- **Taxa de produto devolvidos:** mede a percentagem de produtos devolvidos.

Uma alta taxa de retorno poderá indicar problemas na qualidade da mercadoria, cliente, serviço, ou mesmo do marketing.

- **Taxa de vendas:** percentagem de unidades vendidas *versus* o *stock* inicial.

Esta métrica pode ser usada para avaliar a qualidade do produto. Poderá também ser útil para decidir quais produtos devem ser colocados à venda, devolvidos ao fornecedor, ou se é necessário re-encomendar mais produtos.

- **Taxa bruta de retorno de investimento:** mede o valor realizado por cada unidade monetária gasta em *stock*.

Este indicador pode dar uma indicação sólida de como a estratégia está a funcionar. Pode também dizer que produtos ou departamentos específicos estão a funcionar melhor, para que se possam obter informações sobre a otimização da gestão dos artigos.

3.4 Ferramentas de Gestão

A presente secção serve para a identificação e análise de tecnologias comuns ao desenvolvimento de aplicações *web* que poderão ser utilizadas para a construção da aplicação de gestão necessária no presente projeto.

Serão também identificadas potenciais soluções atualmente no mercado, de forma a perceber se, a solução quanto à ferramenta de gestão a utilizar já existe ou, se apesar das ferramentas já existentes o negócio em questão tem alguma necessidade especial que obrigue ao desenvolvimento de uma solução à medida.

3.4.1 Tecnologias para a Construção de Ferramenta de Gestão

3.4.1.1 ASP.NET Core 2

O *ASP.NET Core* é uma *framework open-source* de alto nível e alto desempenho para a construção de aplicações modernas, baseados na *cloud* e conectadas à Internet. Até aos dias de hoje, milhões de *developers* usaram (e continuam a usar) a *framework ASP.NET 4* para a criação de aplicações *web*. O *ASP.NET Core* é uma versão desenhada a partir do *ASP.NET 4*, com mudanças arquiteturais que resultam numa estrutura mais simples e modular (Anderson 2017).

O *ASP.NET Core* é totalmente fornecido como pacotes *NuGet* (gestor de pacotes da tecnologia *ASP.NET*). Isso permite a otimização das aplicações uma vez que incluem apenas os pacotes necessários tendo assim uma segurança mais apertada, manutenção reduzida, contribuindo para um melhor desempenho de cada aplicação (Anderson 2017).

Esta plataforma integra-se perfeitamente com *frameworks* e bibliotecas populares que executam do lado do cliente, incluindo *Angular*, *React* e *Bootstrap*, sendo esta mais uma razão para o elevado uso da tecnologia (Anderson 2017).

3.4.1.2 NodeJs

O *NodeJs* é uma plataforma construída sobre o motor *JavaScript V8* do *Google Chrome* para facilmente construir aplicações *web* rápidas e escaláveis. Usa um modelo, orientado a eventos, de entrada e saída de dados não bloqueante que o torna leve e eficiente, ideal para aplicações em tempo real com grandes trocas de dados (Foundation 2018).

Em linguagens como *Java* ou *PHP*, cada conexão cria uma nova *thread* que terá cerca dois *megabytes* de memória associados a ela. Pensando que a determinado momento um *website* tem ao mesmo tempo milhares de visitantes a realizar pedidos ao servidor facilmente se conclui que será gasta uma vasta quantidade de recursos. Em vez de criar um novo processo a cada conexão (e alocar a memória anexa a ele), cada conexão lança um evento, evento esse que será executado dentro do motor de processos do *Node*. Com o *V8* a *Google* criou um interpretador ultrarrápido escrito em *C++*, com um outro aspeto único: descarregá-lo e incorporá-lo em qualquer aplicação desejada.

O *NodeJs* suporta uma enorme comunidade desenvolvedores. Uma característica chave que torna o *NodeJs* tão convincente é a facilidade em publicar e aceder aos módulos que compõem as aplicações nesta plataforma. Graças ao gestor de pacotes integrado do *Node*, o *Node Package Manager (NPM)*, distribuir e aceder a estes módulos é extremamente fácil. Um colaborador cria um ficheiro *package.json* contendo os seus detalhes e dependências, de seguida simplesmente "envia" o módulo para o repositório público (*NPM*), onde este fica imediatamente disponível para toda a restante comunidade (Whitehead 2017).

3.4.1.3 Java

Java é uma plataforma de computação e linguagem de programação rápida, segura e confiável. Esta linguagem é maioritariamente usada para a implementação de jogos e aplicações *web* (Oracle 2014).

O *Java* foi construído com a filosofia de "escrever uma vez, executar em qualquer lugar". O código *Java* escrito numa plataforma (sistema operacional) será executado em todas as outras plataformas sem modificações.

Para executar *Java*, é utilizada uma máquina abstrata chamada *Java Virtual Machine (JVM)*. A *JVM* executa o código fonte *Java*, e por sua vez, o processador de cada máquina executa a *JVM*. Uma vez que todas as *JVMs* funcionam exatamente da mesma forma, o mesmo código funciona em outros sistemas operacionais, tornando a plataforma *Java* independente.

As versões anteriores do *Java* foram criticadas por serem lentas. No entanto, a evolução das *JVMs* fazem com que estas fiquem significativamente mais rápidas, tornando mais rápidas as aplicações desenvolvidas nesta plataforma. Esta plataforma oferece também várias garantias de seguranças como a gestão automática de memória, redução da corrupção e vulnerabilidades da memória, comunicação segura (protegendo a integridade e a privacidade dos dados transmitidos), assim como a segurança no desenvolvimento e execução de aplicações (Programiz 2018).

Outra razão pela qual o *Java* é amplamente utilizado é pela utilização de uma vasta biblioteca, com centenas de classes e métodos em diferentes pacotes para ajudar desenvolvedores de software na implementação de soluções nesta plataforma (Programiz 2018).

3.4.1.4 Comparação Entre as Tecnologias Apresentadas

Após a análise anteriormente realizada a cada uma das tecnologias, pode concluir-se que apesar das suas diferenças todas são multi-plataforma, quer isto dizer que qualquer sistema que tenha acesso a um *browser web* será capaz de executar a aplicação desenvolvida, seja ele *MacOS*, *Linux* ou *Windows*.

Pode retirar-se também que ambas são *open source*, querendo isto dizer que não haverão custos associados ao seu desenvolvimento, e consideradas tecnologias atuais, com bastante uso, com muita documentação e suporte fornecido graças a vastas comunidades de utilizadores. As tecnologias analisadas apresentam também cuidados ao nível da performance tendo cada uma delas a sua estratégia para a otimização dos processos, sendo que nesse aspeto as tecnologias *Java* e *NodeJs* tem em atenção a gestão de memória utilizada nos seus processos, tendo a última um fator que supera a outra na questão da gestão de *overhead* associada ao processo de criação de *threads* no seu funcionamento.

É importante referir que a tecnologia *ASP.NET Core* implica o alojamento da solução em ambientes *Windows*, contrariamente às restantes, mas, associado a esse ambiente é possível desenvolver toda a solução pretendida nesse ambiente tendo por base ferramentas *Microsoft*. Com isto, poderão aproveitar-se todas as soluções e estratégias de integração entre as ferramentas pensadas pela organização, por outro lado a adoção de todas estas ferramentas poderão trazer custos muito elevados e que poderão não ser suportados pelos interessados.

3.4.2 Ferramentas Existentes de Apoio à Gestão

Nesta secção serão apresentadas algumas soluções existentes no mercado quanto a ferramentas de apoio à gestão do quotidiano de uma organização.

Pretende-se também com o conhecimento adquirido perceber se existe atualmente no mercado soluções capazes de dar resposta aos problemas enunciados previamente e que se encontram em estudo com este projeto. Em função deste estudo irá decidir-se se haverá necessidade de desenvolver uma solução específica à medida das necessidades atuais ou se será mais viável a adoção de uma das soluções já existentes.

Apesar do assunto da presente dissertação assentar num problema relacionado com a venda de produtos a retalho e existirem soluções específicas para a área do retalho, optou-se por escolher para análise soluções que abordassem não só questões diretamente dessa área de negócio mas que possibilitassem a gestão de uma organização na generalidade do seu quotidiano, um Enterprise Resource Planning (ERP). Uma solução deste tipo deverá ser capaz de oferecer uma visão estratégica e detalhada de tudo o que acontece na organização, permitindo a correta distribuição de recursos, a otimização dos processos, o consequente aumento da produtividade e os consequentes lucros. Assim, as soluções escolhidas foram: o Primavera *Professional*, o PHC *CS Enterprise* o Sage *X3*.

3.4.2.1 Primavera BSS

A Primavera *Business Software Solutions (BSS)* é uma empresa que se dedica ao desenvolvimento e comercialização de soluções de gestão e plataformas para integração de processos empresariais, disponibilizando soluções para pequenas, médias e grandes organizações.

A Primavera BSS está presente em Portugal, Espanha, Brasil, Angola, Moçambique, Cabo Verde e Guiné-Bissau, tendo soluções para as mais variadas áreas e funções de uma organização: *softwares* de faturação, Point Of Sale (POS) para a restauração e retalho, sistemas ERP (figura 9), distribuição e logística, gestão de recursos humanos, *software* de manutenção de equipamentos e ativos, *reporting*, entre outros. A variedade de soluções estende-se também às opções quanto ao número de postos de acesso (monoposto e multiposto), número de utilizadores (mono e multi-utilizador) e quanto ao método de implantação da solução, sendo possível a instalação em servidores físicos ou a instalação na *cloud*. Todas estas opções são disponibilizadas em pacotes que tornam o preço da solução bastante acessível uma vez que é possível escolher o produto desejado e minimizar as funções desnecessárias. Ainda assim, a maioria das soluções é disponibilizada numa versão gratuita até atingir um valor de trinta mil euros de faturação, sem limite de tempo associado, o que permite aos interessados utilizar o produto na totalidade das suas funções e com isto ter uma noção mas real das suas mais valias.

O *software* Primavera é um dos principais sistemas usados em Portugal, e nos países de língua portuguesa, como ferramenta de apoio à gestão e ao negócio do retalho e por isso existe também uma vasta comunidade de pessoas capazes de fornecer rapidamente ajuda a quem esteja a começar a utilizar este software. Além disso, a Primavera tem também áreas de suporte e de formação especializadas nos seus produtos, capazes de fornecer a ajuda necessária para quem possa ter dificuldade ao adotar um dos seus produtos (Primavera 2019).

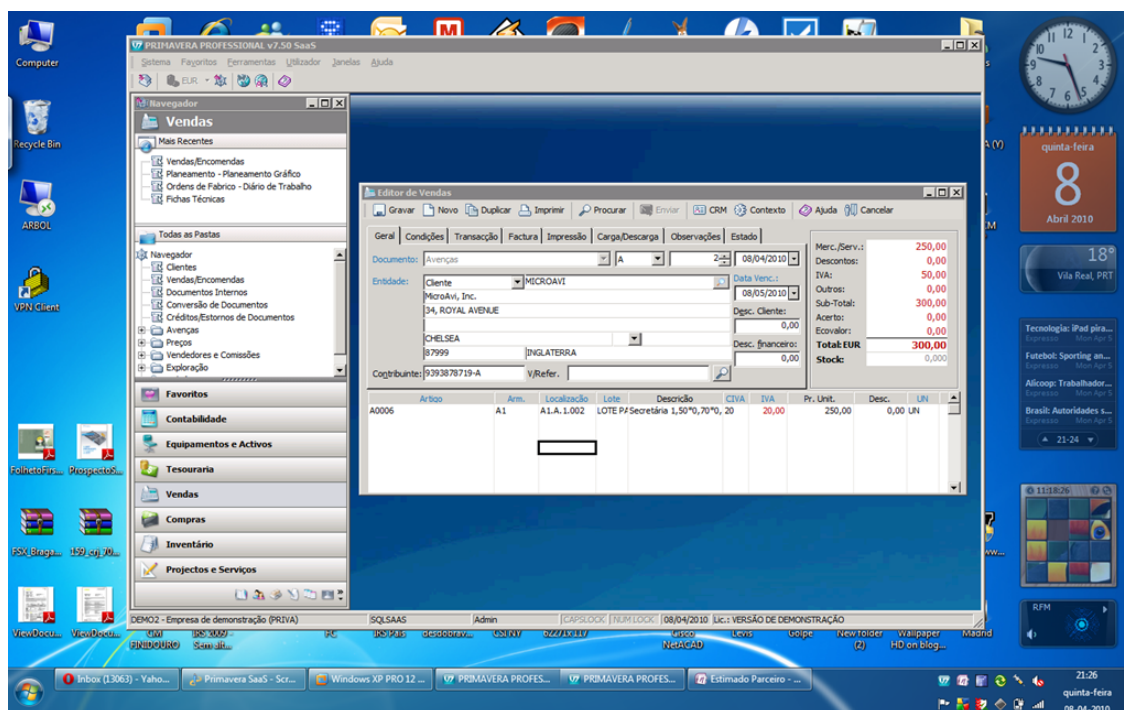


Figura 9: Primavera BSS ERP.

3.4.2.2 PHC

A PHC é também uma solução quanto à gestão da atividade de uma organização, e coloca-se como uma solução capaz de satisfazer os desafios quotidianos seja de organização pequena

ou de grande dimensão. A dimensão da organização é um fator da segmentação dos produtos da PHC, outro é o setor em que a organização se insere. Existem soluções especializadas no retalho, restauração, construção civil, entre outras. A PHC oferece também soluções transversais e mais genéricas para o caso de o desejado ser um sistema de gestão e ERP (Figura 10), um sistema financeiro ou uma solução de suporte técnico (PHC 2019).

Além de versátil, o *software* PHC é apontado como bastante flexível graças às opções de configuração que oferece aos seus utilizadores, fazendo com que o sistema se adapte ao negócio dos seus clientes. Os produtos PHC são também conhecidos por serem feitos boas bases técnicas, sendo grande parte dos seus produtos assentes na *cloud*. Esta qualidade de processos já valeu inclusive um prémio internacional, reputação essa que tem atraído clientes para a empresa. Visto que estes são *softwares* com muitos utilizadores, a PHC tem áreas de formação para instruir os seus utilizadores a usar o sistema da forma mais produtiva, contribuindo desde logo para uma maior produtividade do seu cliente (Sigmacode 2017b).

As soluções PHC, nomeadamente a gama Enterprise, é baseada na tecnologia cliente/-servidor, com *Microsoft SQL Server* como a base de dados relacional. Com o uso desta tecnologia, fica não só assegurada a elevada fiabilidade dos dados (segurança) mas também a capacidade de crescimento para elevados fluxos de informação (escalabilidade).

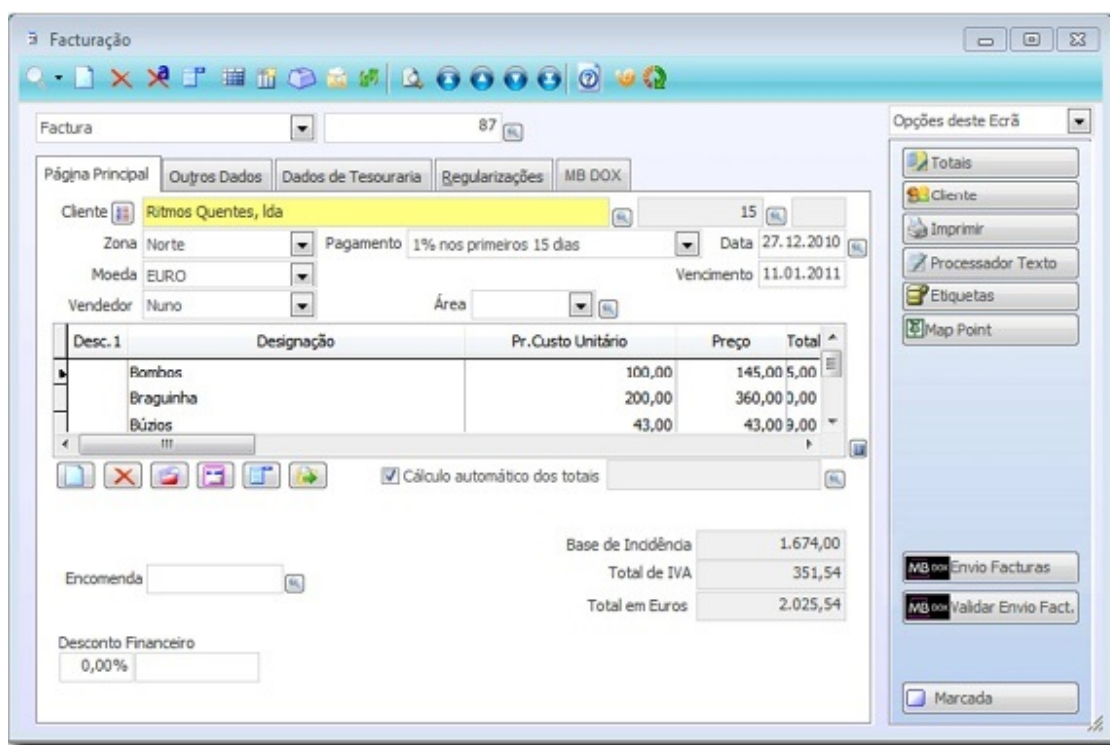


Figura 10: Ecrã de faturação do *ERP PHC*.

Quanto à sua instalação, estes sistemas estão disponíveis sob a forma de alojamento, com a possibilidade de realizar uma implementação local (na sua casa ou no seu escritório), ou então no exterior (num *datacenter* remoto), ou em termos de licenciamento, o seu software pode ser licenciado (isto é, adquirido como em qualquer compra tradicional), ou subscrito (alugado e pago periodicamente).

3.4.2.3 Sage

A Sage é mais uma solução no âmbito de sistemas de apoio à gestão de uma organização. À semelhança dos apresentados anteriormente, esta oferece também uma vasta gama de produtos, ajustados ao tamanho e ao objetivo suposto para o sistema desejado, contando assim como soluções para a gestão de recursos humanos, tesouraria, contabilidade, faturação, etc (Sage 2019).

Além de ter em conta todas as áreas de negócio já identificadas, a Sage é também apontada como uma excelente opção para negócios de elevada mobilidade. Estes sistemas permitem ao cliente que tenha acesso a todas as informações sobre a gestão da sua organização onde quer que esteja. Isto porque este *software* de gestão está otimizado para terminais portáteis como é o caso de *tablets e smartphones* (Sigmacode 2017a).

The screenshot displays the Sage software interface for product management. At the top, there are search and filter fields for 'Category' (containing 'FINIS'), 'Product' (containing 'FIN011'), and 'Product status' (set to 'Active'). Below this is a navigation menu with tabs for 'Identification', 'Management', 'Units of measure', 'Financials', 'Sales', 'After-sales', 'Customers', 'Supply', and 'Suppliers'. The main content area is divided into several sections:

- IMAGE:** A red and black mountain bike is shown in a dashed box. Below it, there are instructions: 'Drop file from explorer or Select it', 'Select file', and 'Remove'.
- IDENTIFICATION:** Contains 'Description 2' (Bike Adult Mountain Bike) and 'Description 3' (empty).
- MISCELLANEOUS:** Includes 'Search key', 'UPC code', 'User access', 'Product line', 'Standard', and 'SAF-T product type' (set to 'None').
- USE:** Features 'Service life start' and 'Service life end' date pickers, and checkboxes for 'Production text' and 'Picking text'.
- EU INFORMATION:** Includes a checked 'Sub Intrastat' checkbox and a 'Customs reference' field.
- STATISTICAL GROUPS:** Contains 'Group 1' and 'Group 2' search fields.

Figura 11: *Software Sage*.

O *software Sage*, Figura 11, apresenta-se como uma alternativa rápida, simples, flexível e poderosa ainda que sem o custo e a complexidade associada aos sistemas de gestão que surgem no mesmo âmbito (Weasy 2019).

3.4.2.4 Comparação Entre Ferramentas

Apresentadas algumas das soluções existentes no mercado, interessa perceber as suas diferenças, assim como as suas vantagens e desvantagens, e se alguma delas oferece todas as funcionalidades e requisitos especificados para a solução pretendida (ver Secção 4.2.3).

Tabela 7: Comparação dos requisitos desejados por ferramenta em análise.

	Primavera	PHC	Sage
Criar Encomenda	Sim	Sim	Sim
Validar Encomendas	Sim	Sim	Sim
Gerir Artigos	Sim	Sim	Sim
Gerir Utilizadores	Sim	Sim	Sim
Gerir <i>Templates</i> de Documentos	Sim	Sim	Sim
Gerar Documentação	Sim	Sim	Sim
Criar Alertas	Sim	Sim	Sim

Pela informação recolhida durante o processo de comparação entre as ferramentas de apoio à gestão anteriormente apresentadas na tabela 7, todas as soluções analisadas apresentam resposta às funcionalidades desejadas, não se excluindo portanto nenhuma opção de início. Nota-se que as soluções em análise são bastante semelhantes seja a nível de similaridade das funções, seja pela capacidade de se manterem estáveis no mercado e ao mesmo tempo atuais, uma vez que todas apresentam versões na *cloud*, no caso da Sage tendo *inclusivé* adaptado a sua solução para dispositivos móveis. No entanto, no que toca à documentação fornecida e, à comunidade que a utiliza e que está disposta a partilhar conhecimento da mesma, a solução Primavera parece destacar-se. Apesar do produto não ser gratuito, foi relativamente fácil encontrar uma versão grátis da ferramenta, e com ela toda a documentação para a sua correta instalação e funcionamento (Moreira 2012). Ainda que limitada a um posto e a uma faturação de trinta mil euros, estas limitações não impedem o uso da mesma para os efeitos de demonstração a que esta dissertação se propõe. Apesar de estar limitada a apenas um posto de trabalho, a versão permite a criação ilimitada de utilizadores e não tem nenhuma limitação de uso a nível temporal, o que permitirá efetuar uma simulação do seu uso como se de uma situação real se tratasse e com isto ter uma noção mais fiável da sua eficácia. Tendo tido resposta positiva a todos os pontos comparativos, esta sólida comunidade de suporte poderá tornar-se muito importante no que toca a adoção da ferramenta de gestão para as funções pretendidas, uma vez que o suporte prestado por esta comunidade e pela documentação fornecida permitirá avançar com a implementação da solução de uma forma mais rápida e apoiada. Assim, a solução Primavera *Professional* é para já a candidata a ferramenta de apoio à gestão pretendida para o presente projeto, não havendo necessidade de desenvolver uma ferramenta específica para este mesmo fim.

3.5 Ferramentas de ETL

Com a implementação de um AD pretende-se não só a elaboração de um sistema funcional, mas também importa garantir a correção e fácil manutenção do próprio AD. Assim, uma boa escolha da ferramenta a utilizar nos processos de *ETL* inerentes à construção do mesmo poderão ser fatores relevantes para o bom sucesso e funcionamento do sistema. Para dar resposta a estas exigências é necessário construir um AD configurável e versátil de modo a que possa evoluir no tempo de forma a dar resposta aos possíveis cenários futuros para o qual é construído.

Nas próximas sub-seções deste documento serão apresentadas algumas ferramentas de *ETL* que estão entre as soluções mais populares neste âmbito, sendo apresentados alguns exemplos de ferramentas de uso gratuito e outros pagos (Alooma 2017a).

3.5.1 Microsoft SQL Server Integration Services

O *Microsoft SQL Server Integration Services (SSIS)*, é uma ferramenta empresarial, especializada na integração de dados, permitindo a sua limpeza, armazenamento e gestão. Esta ferramenta trabalha com pacotes que podem ser executados em simultâneo e suportam dados provenientes de várias fontes de dados como ficheiros de texto, *XML* ou Bases de Dados Relacionais e é capaz de os carregar para um ou mais destinos. Esta plataforma é usada para trabalhar dados com elevada complexidade lógica, pois possui uma variada gama de componentes, como se pode ver na Figura 12, que disponibilizam as mais variadas funções para o manuseamento dos dados, como por exemplo, a cópia ou *download* de ficheiros, o envio de *e-mails* ou o uso de componentes para *data mining*⁶. O *SSIS* permite assim uma vasta oferta de tarefas e transformações possíveis de serem usadas de forma gráfica, permitindo construir soluções avançadas sem escrever uma única linha de código, ainda assim é possível criar tarefas personalizadas via *script*. (Laudenschlager, Guyer et al. 2017)

Para desenvolver no *SSIS* é ainda necessário instalar duas outras ferramentas da Microsoft: o *SQL Server Data Tools (SSDT)* e o *SQL Server Management Studio (SSMS)*. O *SSMS* é uma ferramenta de gestão e execução de instruções sobre um ambiente de produção, que poderá ir desde a gestão da extração dos dados de uma Base de Dados Relacional até à manutenção ou criação da infraestrutura do Armazém de Dados. Por sua vez, o *SSDT* é uma ferramenta moderna, que possibilita facilmente o desenho, construção e implantação de uma solução de *BI*, através da sua constituição em camadas (Laudenschlager e Guyer 2017).

O *SSDT* torna-se muito útil uma vez que possibilita as seguintes opções (Laudenschlager, Guyer et al. 2017):

- Importação e exportação de pacotes para criar cópias de dados, de uma fonte para um destino, através de um assistente (*wizard*) gráfico;
- Criação de pacotes com um fluxo de controlo complexo, incluindo uma lógica orientada, registo de eventos e controlo de exceções;
- Realização de testes e *debug* sobre os diferentes pacotes existentes;
- Criação de configurações que permitem alterar as definições/propriedades dos pacotes em tempo de execução;
- Instalação de pacotes e suas dependências noutras máquinas;
- Gravação de cópias dos pacotes numa Base de Dados específica, do *SSIS* e do sistema de ficheiros.

⁶É o processo de filtragem de grandes conjuntos de dados para a identificação de padrões e relacionamentos entre esses mesmos dados para resolver problemas através da sua análise. <http://searchsqlserver.techtarget.com/definition/data-mining/>

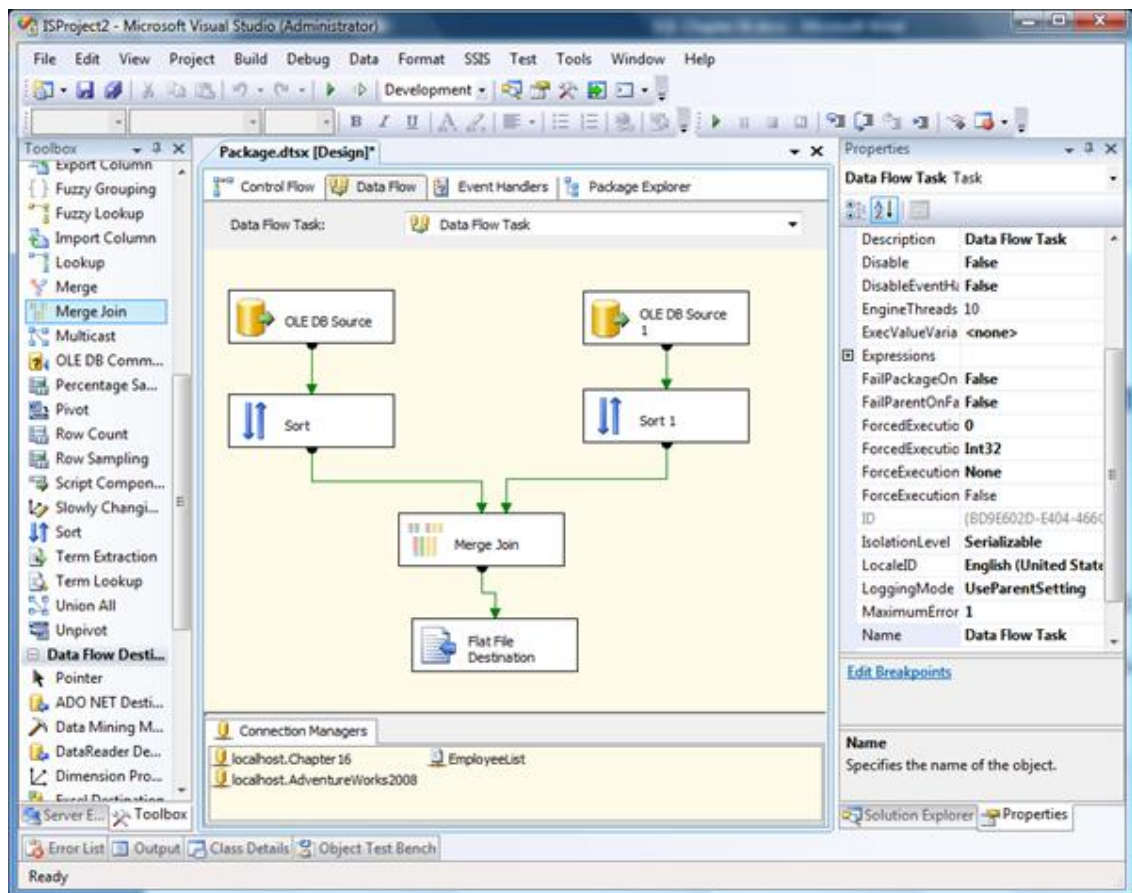


Figura 12: Microsoft SQL Server Integration Services Integrated Development Environment (IDE).

3.5.2 Pentaho Data Integration

O *Pentaho Data Integration (PDI)* é uma das mais conhecidas ferramentas *ETL openSource* do mercado. Esta plataforma faz uso de um conjunto de ferramentas gráficas de forma a eliminar a complexidade da programação via escrita de código e conseguir com isso a aceleração do *design* e da implantação até quinze vezes mais rápida. Graças a esta interface, Figura 13, é possível criar facilmente *pipelines* de *ETL* para os dados via *drag-and-drop*. O *PDI* disponibiliza também uma rica biblioteca para a importação de dados a partir de variadas fontes de informação que podem ir desde os simples ficheiros de texto até às bases de dados relacionais ou a aplicações empresariais (Vantara 2017).

Esta ferramenta apresenta ainda mais algumas características importantes que poderão tornar-se muito úteis, quer em tempo de construção da solução, quer na sua manutenção (Vantara 2017):

- Poder de coordenar e combinar transformações de dados, incluindo a geração de notificações e alertas;
- Ferramentas de *debug* para testes em tarefas do *pipeline*;
- Apresentação de vistas ágeis para a modelação e visualização de dados em tempo de processamento dos dados;

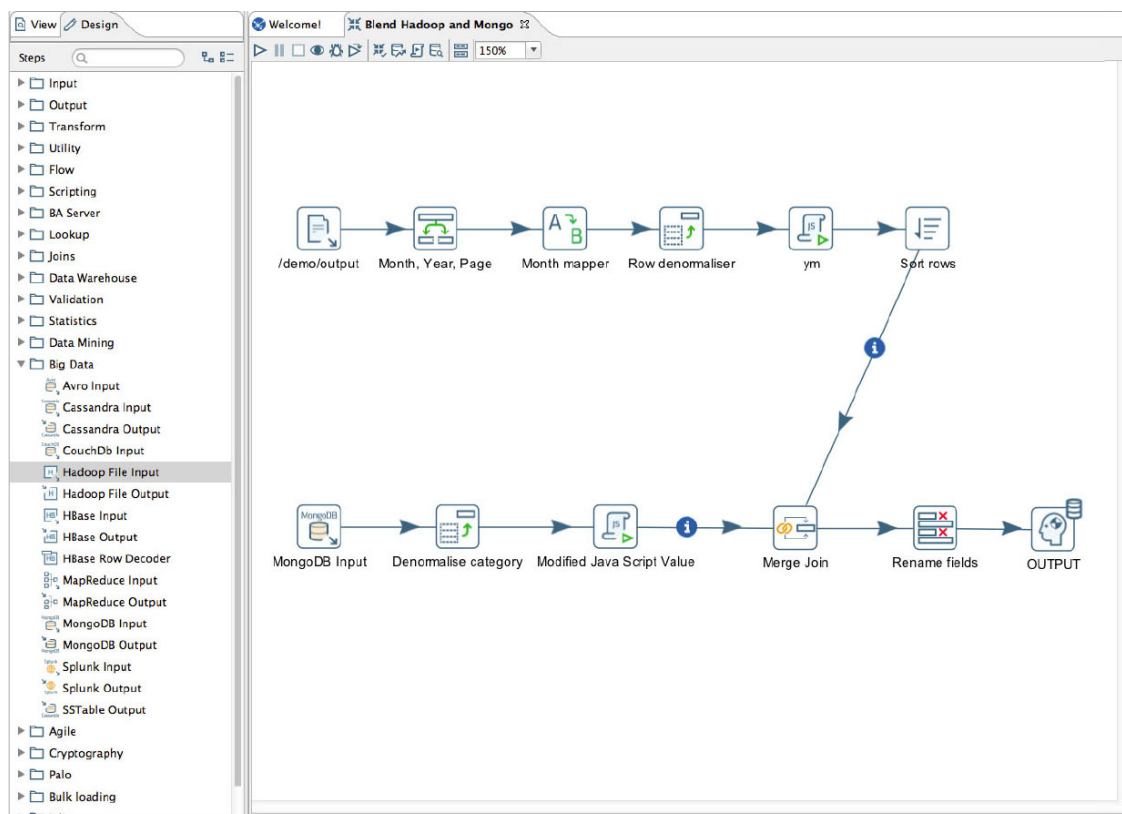


Figura 13: Pentaho Data Integration IDE.

- Mecanismos de Balanceamento de carga;
- Mecanismo para reverter alterações a dados efetuadas por *jobs* (ou tarefas);
- Loja de *plugins* que se podem adicionar à solução, permitindo assim adicionar ainda mais funcionalidades à aplicação disponibilizada pela *Pentaho*.

3.5.3 Talend Open Studio for Data Integration

O *Talend Open Studio (TOS)* é, assim como a solução apresentada na sub-secção anterior, uma proposta *openSource* para a resolução de problemas relacionados com a análise de grandes volumes de dados. Esta ferramenta foi desenvolvida para dar resposta a problemas atuais mas pensando naquilo que se poderá necessitar no futuro (Yin 2017).

Sendo uma aplicação *openSource*, graças aos utilizadores e desenvolvedores que, efetivamente colaboram na melhoria e inovação deste *software*, fazem com que o *TOS* esteja constantemente a ser atualizado de forma automática. A ferramenta possui também um interface gráfico (figura 14), que permite a criação de tarefas de forma fácil, sem ter de conhecer realmente todo o código para a sua implementação. Este interface é baseado no *Eclipse* facilitando a construção de *jobs* de integração aproveitando as últimas inovações das tecnologias *Spark*, *Spark Streaming* e *machine learning*, sendo capaz de fazer versionamento e exportação desses mesmos *jobs* (Talend 2018).

O *TOS* apresenta ainda um leque variado de conectores que permitem fazer a ligação com sistemas de Bases de Dados, *Dropbox*, servidores *Lightweight Directory Access Protocol*

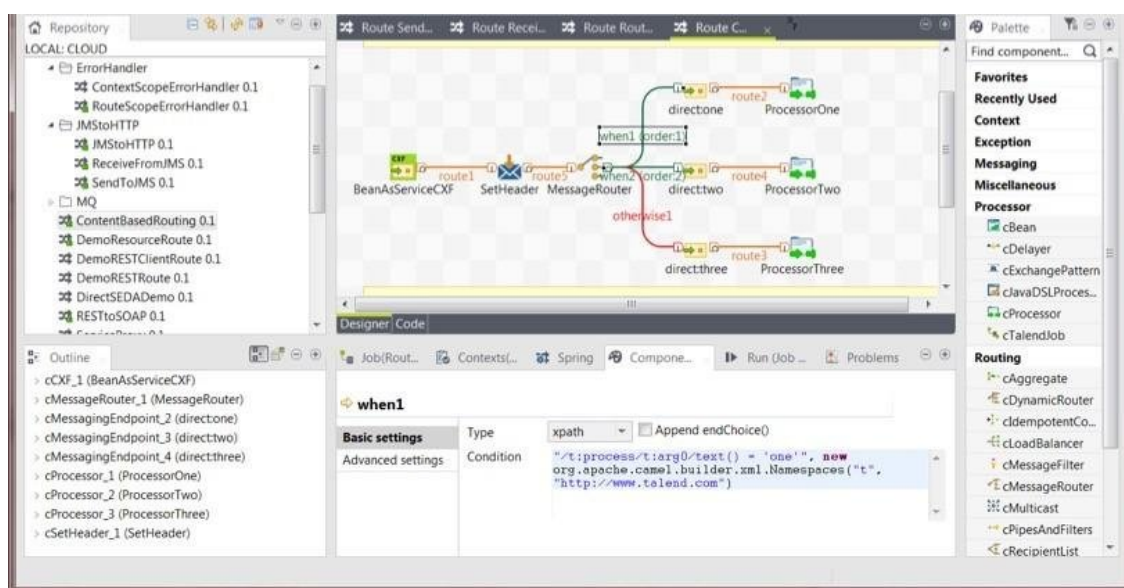


Figura 14: Talend Open Studio IDE.

(LDAP), File Transfer Protocol (FTP), Simple Mail Transfer Protocol (SMTP), entre outros. Além de suportar operações de ETL, permite também Extração, Carregamento e Transformação (ELT), apresentando assim uma alternativa à primeira alargando assim as suas possibilidades de opção (SourceForge 2018).

3.5.4 Comparação entre Ferramentas de ETL

Tendo em conta as considerações apresentadas na secção anterior é apresentada a seguir uma tabela (Tabela 8) comparando as mais-valias e contra-partidas de cada uma das ferramentas descritas (Coleman 2017), (Alooma 2017b), (McBurney 2007).

Tabela 8: Comparação entre Ferramentas ETL

	Vantagens	Desvantagens
<i>Microsoft SQL Server Integration Services</i>	Vasta Documentação e Suporte; Fácil Implementação; Boa Performance; Operações em tempo real.	Problemas em ambientes não <i>Windows</i> ; Visão e estratégia pouca claras.
<i>Pentaho Data Integration</i>	Bom suporte fornecido; Permite integração com outros produtos de <i>BI</i> ; Vários objetivos de transformação e suporte às <i>SCD</i> .	Pouco eficiente no particionamento de dados em paralelo; Baixo desempenho quando necessita de executar vários <i>scripts</i> de <i>SQL</i> em <i>lookup's</i> .
<i>Talend Open Studio</i>	Muito boa interface gráfica; Curto período de habituação; Comunidade extensa e ativa.	Exige alguns recursos computacionais.

Em suma, através deste estudo é possível perceber que todas as ferramentas em análise apresentam uma boa interface gráfica, com recurso a uma vasta coleção de componentes que permitem a cada uma delas efetuar um enorme conjunto de operações, a maioria delas sem ter de escrever o código associado à operação desejada. Estas aplicações são bastante usadas atualmente, com uma comunidade de utilizadores elevada, com suporte técnico e documentação fazendo com que apresentem um fácil e curto período de aprendizagem.

Individualmente, cada solução apresenta alguns pontos que terão de ser analisados e que posteriormente têm de ser tidos em conta para a eleição da ferramenta a utilizar. Ao analisar a performance das ferramentas, o *Microsoft Integration Services* parece ser a melhor opção uma vez que os concorrentes, *Pentaho* e *Talend*, parecem refletir problemas no tratamento de dados complexos e na própria gestão de recursos computacionais respetivamente. No entanto, o *Integration Services*, também tem alguns aspetos negativos como por exemplo o facto de apenas funcionar em ambientes *Windows* e, caso hajam restrições a nível económico, não é *open source*.

3.6 Ferramentas de Processamento Analítico de Dados

Depois de inserir os dados no sistema de gestão de informação, pretende-se o processamento e análise dos mesmos.

Para isso, existem no mercado ferramentas capazes de efetuar análises a grandes quantidades de dados, sendo necessário muito pouco detalhe técnico para a sua adoção. Assim, a introdução deste tipo de ferramentas no sistema a desenvolver, faz da sua possível utilização uma solução bastante fiável.

Nas próximas sub-secções do documento apresenta-se um estudo realizado a algumas das ferramentas mais utilizadas pelo mercado.

3.6.1 Microsoft Power BI

O *Microsoft Power BI* é uma coleção de serviços de software, aplicações e conectores que trabalham juntos para transformar dados em informações coerentes, visualmente agradáveis e interativas. O *Power BI* permite facilmente a criação de conexões às fontes de dados, quer se encontrem simplesmente em ficheiros *Excel* ou numa coleção de AD híbridos, sejam eles locais ou alojados numa *cloud* (Microsoft 2016).⁷

O *Power BI* consiste em três elementos principais, representados na Figura 15 – o *Power BI Desktop* (ferramenta *Desktop*), o serviço do *Power BI* (ferramenta *online*) e o *Power BI Mobile* (aplicação móvel) – que trabalham em conjunto para permitir a criação, interação, partilha e visualização dos dados da maneira que se desejar.

Esta solução usa no seu funcionamento um conjunto de blocos de construção que permitem a criação fácil de formas de visualização de dados. Os blocos são os seguintes (Microsoft 2016):

- **Visualizações:** uma representação visual dos dados, às vezes, chamadas de visuais;

⁷O alojamento em nuvem, ou nuvem, é um modelo de serviço no qual os dados são mantidos, geridos, copiados de forma remota e disponibilizados para os seus utilizadores numa rede (normalmente, a Internet). <http://searchstorage.techtarget.com/definition/cloud-storage>

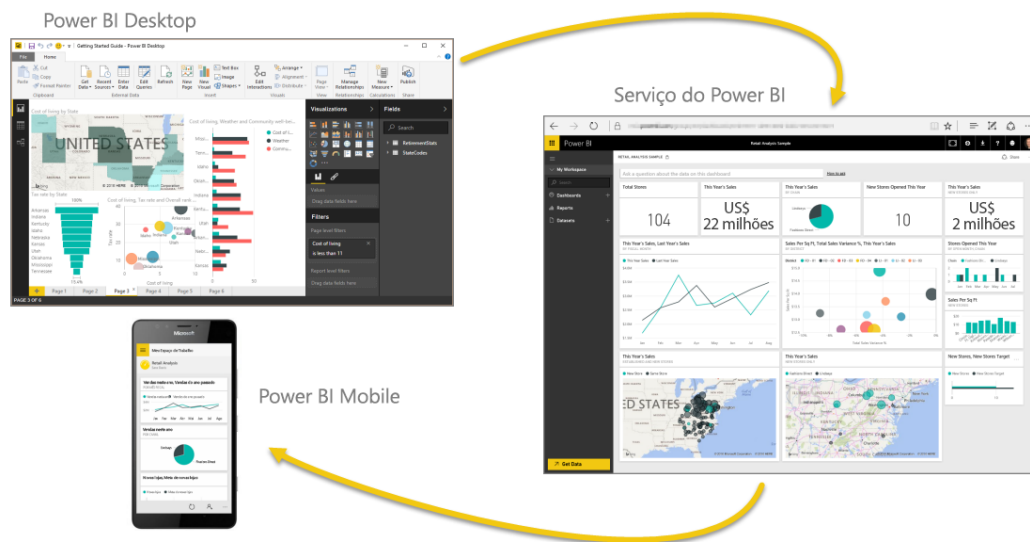


Figura 15: Elementos Principais do *Microsoft Power BI*.

- **Conjuntos de dados:** uma coleção de dados que o *Power BI* usa para criar visualizações;
- **Relatórios:** uma coleção de visuais de um conjunto de dados, abrangendo uma ou mais páginas;
- **Dashboards:** uma coleção de página única de visuais, criados com base num relatório;
- **Blocos:** uma única visualização encontrada num relatório ou *dashboard*.

Assim, simples de usar e disponível em versões gratuitas, este sistema permite a criação de *dashboards* interativos através de conjuntos de informação. Sendo esta uma solução *Microsoft* é também possível a gestão do desenvolvimento a partir do *Visual Studio* (*framework*⁸ de desenvolvimento também da *Microsoft*).

3.6.2 Phocas

O *Phocas* é também um *software* que permite análises analíticas a dados, utilizando a criação de *dashboards*, Figura 16, interativos e de fácil interpretação. Tendo gerados estes *dashboards* é possível aceder aos dados que o compõe através de um simples *click* conseguindo rapidamente aceder à fonte do conhecimento (Ltd 2016).

Este *software* pode também ser considerado fácil e intuitivo de usar, devido à utilização de componentes *drag and drop* que definem o tipo de informação a visualizar, permitindo a fácil utilização até por pessoas sem muitos conhecimentos técnicos.

Esta é uma solução escalável, permitindo a alteração das estruturas e do tipo de informação das análises de forma bastante fácil e facilmente acessível em qualquer dispositivo, via *web*. Os *dashboards* e relatórios produzidos podem também ser partilhados a partir de uma

⁸Uma *framework* de software é uma abstração de *software* que fornece funcionalidades genéricas, possível de ser alterado pelo utilizador, fornecendo software específico da aplicação. <https://www.quora.com/What-is-a-framework-in-programming>

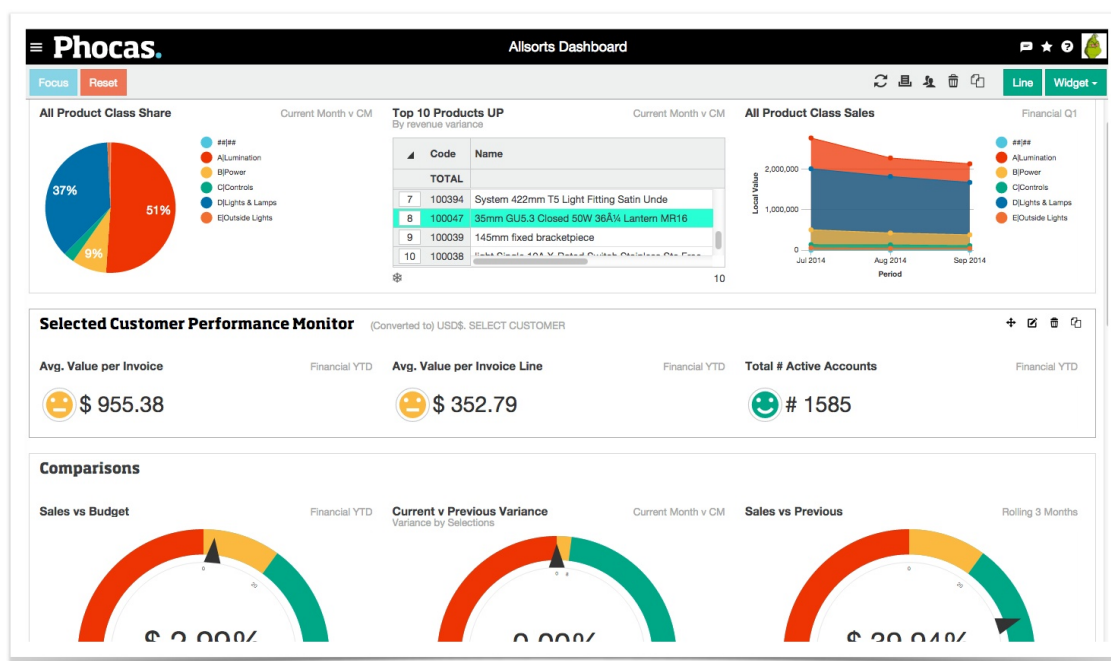


Figura 16: Dashboard de Vendas Criado no Phocas.

funcionalidade disponível pela aplicação. Permite ainda a integração com sistemas ERP e CRM (Ltd 2016).

Apesar de não ser um software Microsoft o seu alojamento só é possível em servidores Windows podendo isto significar a necessidade de despender mais recursos monetários.

3.6.3 Kibana

O Kibana é uma ferramenta de exploração e visualização de dados open source, usada para a monitorização de dados, análise de registos e séries temporais, inclusivé com recurso a técnicas de inteligência artificial que se poderão tornar essenciais para detetar automaticamente alterações fora do normal nos indicadores de performance de uma organização. Este oferece integração com o Elasticsearch⁹, um popular mecanismo de análise e procura de dados, o que torna o Kibana a escolha padrão para visualizar dados armazenados neste sistema. O Kibana é também popular devido aos seus recursos poderosos e fáceis de usar, como histogramas, gráficos de linha, mapas de calor (entre outros) e suporte geoespacial incorporado. É possível ver alguns destes na Figura 17 (Amazon 2018).

Como as soluções anteriormente, este software permite também a partilha de dados, incluindo a sua forma gráfica, em formato de texto (PDF ou Excel), entre outras.

O Kibana oferece também ferramentas muito vantajosas em tempo de construção que poderão ser bastante importantes aquando da construção da solução final. Estas ferramentas, normalmente denominadas de ferramentas de programador ou de debug¹⁰ oferecem funcionalidades como (Elasticsearch 2018):

⁹é um motor de pesquisa e análise distribuído, desenvolvido em servidores Apache. <https://aws.amazon.com/pt/elasticsearch-service/what-is-elasticsearch/>

¹⁰é o ato de tentar remover erros de um programa de computador.

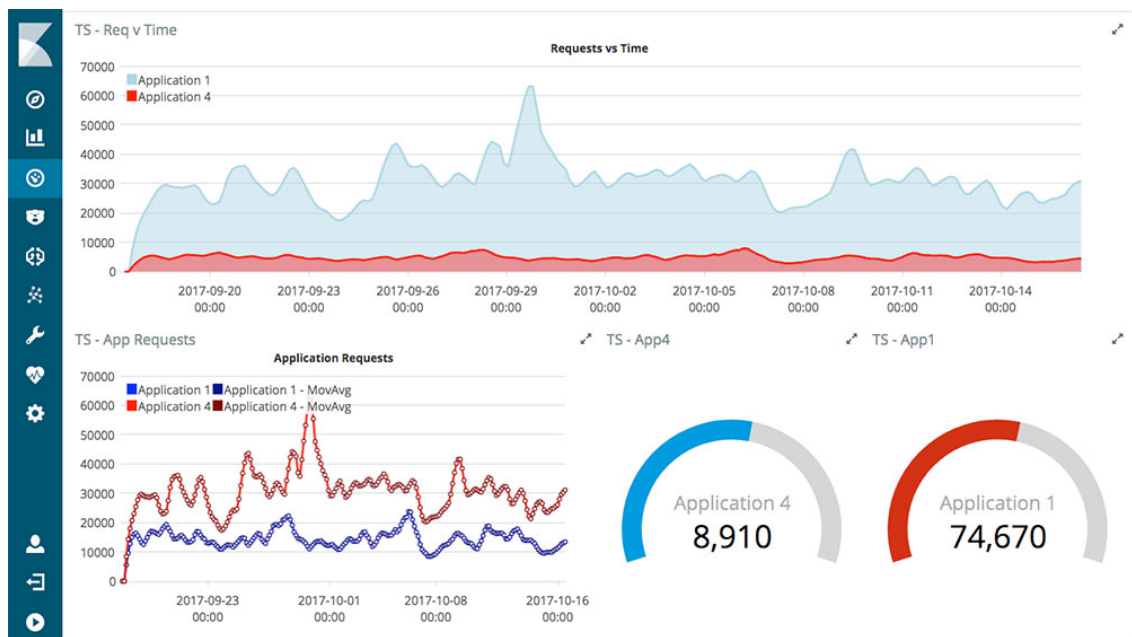


Figura 17: Gráfico de Séries Temporais em *Kibana*.

- **Consola:** interface gráfica para interagir com a *API*¹¹ *REST*¹² da *Elasticsearch*. Tem duas áreas principais: o editor, onde se compõem os pedidos para a *API* e o painel de resposta, que exibe as respostas aos pedidos efetuados.
- **Explorador de Pedidos:** ferramenta de visualização gráfica da performance dos pedidos efetuados pela plataforma permitindo o diagnóstico rápido da performance da solução.
- **Grok:** é uma ferramenta para a análise de padrões usada para a análise de texto e para a sua estruturação. O *Grok* é perfeito para analisar *logs*¹³ em servidores *web*, como por exemplo *Apache* ou *MySQL* e, em geral, qualquer formato de *log* geralmente escrito para a percepção por parte de humanos.

3.6.4 Comparação entre Ferramentas de Processamento Analítico de Dados

Apresentadas algumas das soluções atualmente utilizadas no processamento de dados, será agora feita uma comparação entre as soluções apresentadas de forma a compreender os seus pontos fortes e fracos (Tabela 9) (Advice s.d.[a]) (Advice s.d.[b]) (Crowd 2017).

¹¹Application Programming Interface.<https://techterms.com/definition/api>

¹²REpresentational State Transfer.<https://www.codecademy.com/articles/what-is-rest>

¹³é a informação recolhida cronologicamente de todos os eventos que afetam um sistema informático e de todas as ações das quais resultaram desses eventos. <https://support.ankama.com/hc/pt-br/articles/204188697-0-que-é-um-log>

Tabela 9: Comparação entre Ferramentas de Processamento Analítico de Dados.

	Vantagens	Desvantagens
<i>Microsoft Power BI</i>	Fácil integração com restantes soluções <i>Microsoft</i> no âmbito de <i>BI</i> ; Disponível em versão grátis (mas muito limitada).	Grandes quantidade de dados afetam a performance da ferramenta; Apenas pode ser alojado em servidores <i>Windows</i> .
<i>Phocas</i>	Acesso fácil aos dados a partir dos gráficos criados; Facilidade de uso para novos utilizadores	Apenas pode ser alojado em servidores <i>Windows</i> .
<i>Kibana</i>	<i>Open Source</i> ; Ferramentas de programador muito valiosas; Facilidade em fazer <i>upload</i> de ficheiros e visualização de dados graficamente em tempo real.	A análise de <i>logs</i> pode tornar-se muito complexa.

Após a análise da informação recolhida e que deu origem à Tabela 9, podemos verificar que a única solução *open source* é o *Kibana* ao contrário do *Phocas*. Já o *Microsoft Power BI* apresenta a hipótese de ser utilizado de forma gratuita e paga, sendo que na opção gratuita o seu funcionamento é algo limitado.

O *Microsoft Power BI* apresenta ainda assim algumas limitações, nomeadamente ao nível da performance no tratamento de grandes quantidades de dados, bem como o facto de apenas poder ser alojado em servidores *Windows*, sendo esta última característica comum ao *Phocas*.

O *Kibana* destaca-se pela facilidade da visualização de dados inseridos em tempo real e pelo uso de ferramentas muito úteis aos programadores em fase de construção de soluções usando este *software*. No entanto, estas ferramentas podem tornar-se complexas para programadores não habituados a *logs* deste tipo de sistemas.

4. Análise e Desenho

O presente capítulo tem como objetivo descrever tecnicamente a solução proposta no sentido de contribuir para a resolução do problema descrito.

Numa fase inicial, o capítulo começa por apresentar a modelação do negócio, no qual é apresentado o modelo de domínio desenvolvido para retratar o domínio da solução. De seguida é demonstrado o trabalho realizado para a engenharia de requisitos do projeto, contendo a definição das partes interessadas, atores do sistema e o modelo *FURPS+*¹⁴ com o intuito de captar os requisitos funcionais e não funcionais do sistema a desenvolver.

De seguida apresentam-se os principais indicadores de performance definidos para ter em consideração nas análises a fazer aos dados de negócio, sendo por fim apresentada a arquitetura prevista para a solução.

4.1 Modelação de Negócio

O modelo de domínio é um artefacto essencial e característico da disciplina de modelação de negócio, uma vez que permite representar visualmente o conhecimento adquirido através da análise do negócio, refletindo uma aproximação ao domínio identificado. O modelo de domínio permite analisar a relação entre objetos de forma simples como entidades com significados, sejam elas complexas ou básicas (Fowler 2015).

Para a realização do modelo de domínio, foi utilizado a notação *UML*¹⁵, que permite identificar as classes conceptuais que caracterizam os conceitos de domínio e a relação entre estes. Através da sua classificação baseada em objetos, é possível estabelecer uma ligação entre a análise de requisitos e o design da solução, uma vez que a realização deste modelo permite identificar classes que poderão posteriormente dar origem a classes de software.

Na Figura 18 é possível visualizar todas as classes de domínio identificadas para a solução proposta. É também possível observar a relação entre as classes identificadas.

Na base do problema que deu origem a este projeto está um Comercial cuja função é registar encomendas de artigos para um determinado cliente. Esta encomenda precisa de ser validada por um Armazenista que garante a existência de todos os artigos em *stock*. Depois de validada a encomenda dá origem a uma venda. Por fim, o Administrador, é

¹⁴é um sistema para a classificação de requisitos em engenharia de *software*.<https://businessanalysttraininghyderabad.wordpress.com/2014/08/05/what-is-furps/>

¹⁵é uma linguagem-padrão para a elaboração da estrutura de projetos de software.<http://www.uml.org/what-is-uml.htm>

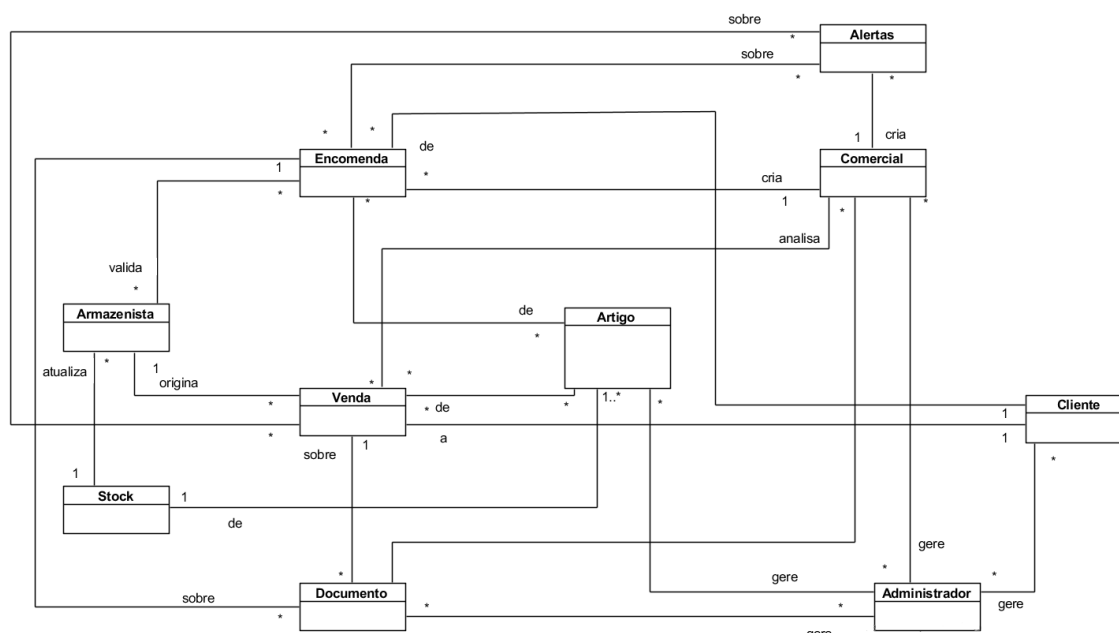


Figura 18: Modelo de Domínio Previsto Para a Solução.

responsável pela gestão de Comerciais, Armazenistas e Clientes inseridos no negócio, assim como a disponibilização do documento que servirá de comprovativo a estas operações.

4.2 Engenharia de Requisitos

A presente secção serve para a identificação e análise de tecnologias comuns ao desenvolvimento de aplicações *web* que poderão ser utilizadas para a construção da aplicação de gestão necessária no presente projeto.

A disciplina de engenharia de requisitos permite identificar, analisar as necessidades e restrições de determinado software a desenvolver, sendo um importante artefacto de análise, para a implementação de qualquer software.

Sendo esta a uma das fases iniciais do processo de desenvolvimento, a execução da atividade de engenharia de requisitos dever ser cuidada, uma vez que as fases seguintes têm por base os resultados obtidos desta fase. Tal como a abordagem *RUP*¹⁶ pressupõe, os resultados de uns artefactos, são utilizados como entrada para os artefactos seguintes, por exemplo, os resultados da análise são utilizados como ponto de partida para o design do software.

No decorrer deste subcapítulo serão apresentadas as partes interessadas na solução proposta, os atores do sistema e os requisitos funcionais e não funcionais do modelo *FURPS+*.

¹⁶é um processo de engenharia de software criado para apoiar o desenvolvimento orientado a objetos, fornecendo uma forma sistemática para se obter vantagens no uso de *UML*. <https://www.infoescola.com/engenharia-de-software/rup/>

4.2.1 Partes Interessadas

Parte interessada, ou *stakeholder*, é uma entidade que poderá ser afetada ou ter interesse pelo sistema.

É importante, por isso, identificar quem são esses interessados, uma vez que poderão influenciar o processo de desenvolvimento do sistema. As partes interessadas para o projeto em questão são:

- **Cliente:** pessoa que compra artigos/produtos.
- **Armazenista:** pessoa responsável pelo material (artigos/produtos) armazenados.
- **Comercial:** pessoa responsável pela criação de encomendas para clientes, assim como pela análise dos dados dessas mesmas encomendas, e pela criação de documentação relativa à mesma.
- **Administrador:** pessoa responsável pela gestão das pessoas associadas ao processo de negócio e pelos artigos a comercializar.

4.2.2 Atores do Sistema

No contexto de desenvolvimento de *software*, um ator do sistema é uma pessoa, que interage com o sistema. Este ator faz uso de funcionalidades que lhe são permitidas pelo sistema, sendo estas atribuídas, pelo sistema ou pelo administrador do mesmo, tendo influencia no seu funcionamento. Um *stakeholder* poderá ser considerado um ator, se interagir com o sistema e usar as suas funcionalidades, o contrário poderá não ser verdade no caso de um *stakeholder* não poder usar o sistema.

Os atores até agora identificados no projeto em questão são:

- **Armazenista:** como apresentado anteriormente em Partes Interessadas, este ator é responsável pela atualização dos *stock* em armazém. Este tem também a função de validar a disponibilidade dos artigos encomendados em *stock*, e caso isto se verifique, validar a respetiva encomenda.
- **Comercial:** este ator, como referido anteriormente, é a pessoa responsável pela criação de encomendas. Além de criar encomendas, este é também responsável pela criação de documentos de análise impostas para a análise do negócio.
- **Administrador:** o Administrador é o responsável pela gestão das pessoas associadas ao processo de negócio, pelos artigos em comercialização pela organização.

4.2.3 Requisitos Funcionais

Os requisitos de um *software* podem ser captados pelo modelo *FURPS+*, sendo que a letra "F" (**F**unctionality) representa os requisitos funcionais, que serão explicados ao longo desta subsecção.

Os requisitos funcionais constituem as funcionalidades que o sistema deve possuir, assim, as funcionalidades da solução apresentada serão representadas em forma de casos de uso e posteriormente explicadas.

Os casos de uso representam ações realizadas por um ou mais atores do sistema com o intuito de atingir um determinado objetivo. Uma forma de apresentar os casos de uso é através da realização de um Diagrama de Casos de Uso em *UML*, diagrama esse que representado pela Figura 19.

O detalhe associado a cada funcionalidade será descrito e apresentado posteriormente.

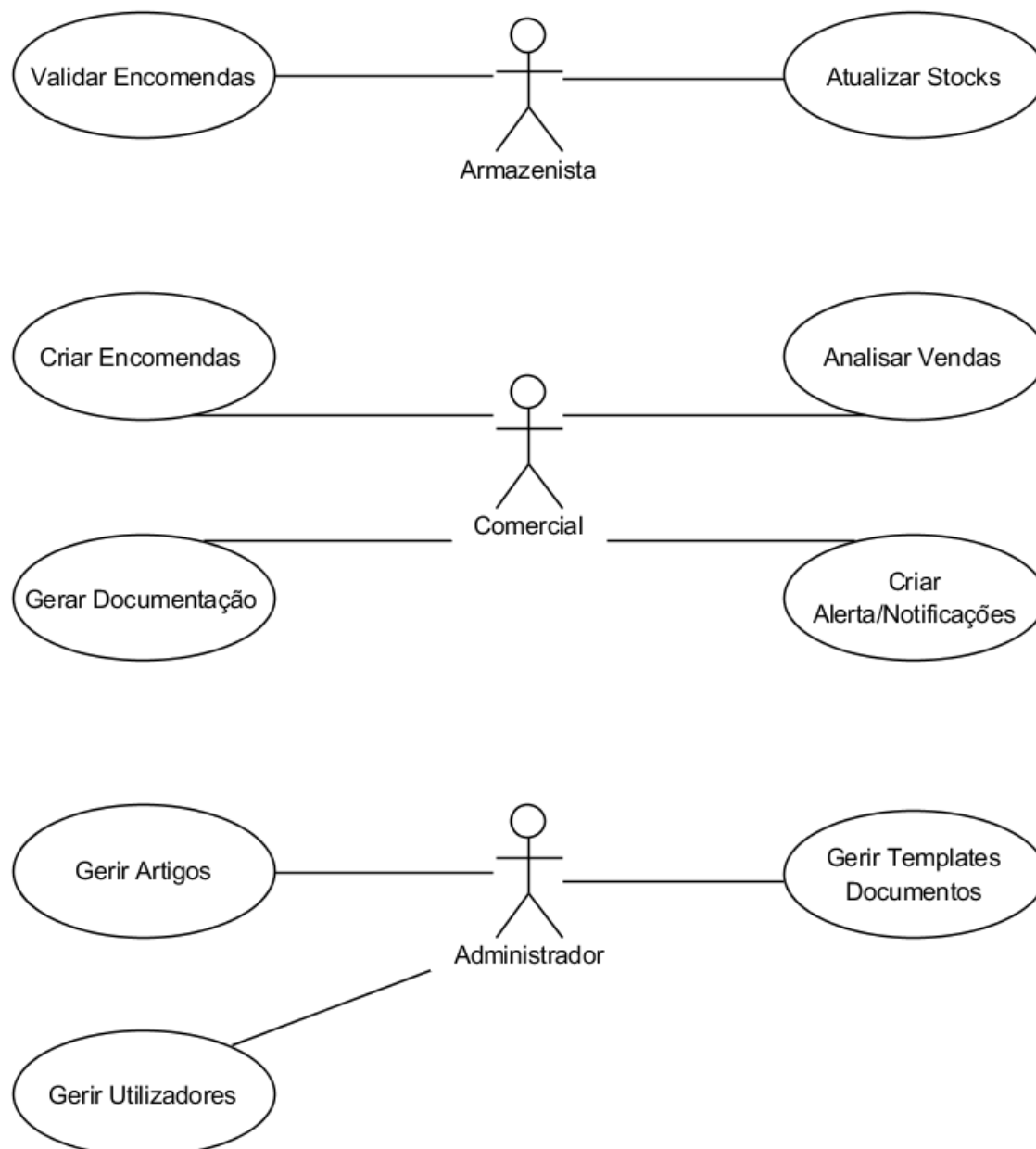


Figura 19: Diagrama de Casos de Uso Previsto Para a Solução.

Armazenista: como apresentado anteriormente em Atores do Sistema, um Armazenista é responsável pela atualização dos *stock* em armazém. Quando é criada uma encomenda, o Armazenista recebe um ficheiro (Nota de encomenda) com os produtos e as respetivas quantidades encomendadas. Este tem que validar a disponibilidade do *stock* e validar a encomenda caso a condição se verifique. Com a implementação da solução em questão, este,

ao validar a encomenda, dará automaticamente início ao processo de venda e atualização do *stock* disponível.

Comercial: este ator, como referido anteriormente, é a pessoa responsável pela criação de encomendas mas, com a implementação da solução, poderá passar a fazê-lo muito mais fácil e intuitivamente num sistema informático desenvolvido também com esse intuito. Além de criar encomendas, este é também responsável pela criação de documentos de análise dessas mesmas encomendas e consequentes vendas, por isso, tem interesse em conseguir de forma automática, rápida e correta, gerar documentos de gestão para fácil análise. Nesta análise inclui-se também a necessidade da visualização gráfica, intuitiva e rápida de alguns indicadores ou métricas essenciais ao processo de gestão que desempenha, tornando-se uma ferramenta essencial a visualização dos mesmos via *dashboard*. Outro requisito importante é a possibilidade de criação de alertas relacionados com estas métricas e a notificação sempre que uma meta está próxima da sua conclusão.

Administrador: o Administrador é o responsável pela gestão das pessoas associadas ao processo de negócio, pelos artigos em comercialização pela organização. Assim, este precisa de ter à sua disposição uma ferramenta que lhe permita a gestão efetiva das pessoas envolvidos no negócio, permitindo-lhe a gestão de permissões dos utilizadores no sistema garantindo o acesso apenas ao que é devido a cada um. É também essencial que este possa definir e manter a lista de artigos em comercialização de forma fácil de modo a garantir a consistência da informação. Sendo este um elemento responsável pelo negócio, poderá também definir os *templates* que serão usados no processo de encomenda e venda de produtos de modo a garantir a criação de documentação uniformizada na organização.

4.2.4 Requisitos Não Funcionais

Os requisitos não funcionais de um sistema, são que vão definir o modo como o sistema se irá comportar, sem definir as suas funcionalidades concretas.

Como referido anteriormente, a letra "F" do modelo *FURPS+*, representa os requisitos funcionais. As restantes letras do acrónimo representam os requisitos não funcionais.

De seguida, é apresentada a descrição de cada uma das restantes do modelo *FURPS+*:

- Usabilidade (**Usability**): características relacionadas com a estética e consistência da interface com o utilizador;
- Confiabilidade (**Reliability**): características relacionadas com a disponibilidade do sistema, exatidão em cálculos efetuados e a capacidade de recuperar em caso de falha;
- Desempenho (**Performance**): características relacionadas com o desempenho do sistema, tempo de resposta, tempo de recuperação, tempo de inicialização e tempo de encerramento;
- Suportabilidade (**Supportability**): características relacionadas com a testabilidade, adaptabilidade, compatibilidade, configurabilidade, instabilidade, escalabilidade e localização;
- +: utilizado para categorias adicionais como requisitos de design, requisitos de implementação, entre outras.

Apresentadas as características gerais que o modelo tem em conta, serão descritos a seguir os requisitos não funcionais associados ao projeto.

Idioma

O sistema a desenvolver deve estar preparado para o seu uso a nível global, não podendo ficar preso a especificidades de uma região ou país em particular. Assim, foi definido um requisito não funcional que permita ao utilizador seleccionar o idioma a utilizar no sistema, seja manualmente ou mediante a sua localização.

Autenticação e autorização

O sistema deve proteger o acesso às funcionalidades e conteúdos tendo por base uma lógica para o acesso a esses mesmos dados. Assim, pretende-se a implementação de um sistema que permita o registo e o login de utilizadores, com um nível de autorização associada, para que se permita o controlo das suas permissões no sistema.

Gestão de utilizadores

O acesso ao sistema deverá ser protegido por métodos de autenticação e autorização que devem ser geridos por alguém com essa responsabilidade. Esta responsabilidade estará a cargo dos administradores, que farão a gestão de utilizadores e dos seus papéis no sistema.

Escalabilidade

O sistema deverá estar preparado para funcionar rápida e corretamente, mesmo quando a carga de pedidos for elevada. Com este requisito pretende-se que sejam implementadas metodologias que garantam a qualidade de performance da solução mesmo nos casos em que a sua taxa de utilização seja elevada.

Plataforma

A solução a desenvolver deverá estar disponível em ambientes *Windows*.

4.3 Design e Arquitetura da Solução

A arquitetura do sistema tem como principal objetivo desenhar uma estrutura geral do sistema a desenvolver, apresentando os diversos componentes do sistema e as suas relações, não sendo uma prioridade a apresentação de detalhes de implementação do sistema.

Nesta secção pretende-se explicar o fluxo da informação desde a sua captação desde os sistemas origem, a sua passagem pelos diferentes componentes do sistema até à sua disponibilização final.

4.3.1 Fontes de Informação

Como se explicou na descrição do problema, e que originou este projeto, o registo de vendas efetuado por um comercial de uma empresa, no ramo do retalho de produtos óticos, é feito tendo por base três ficheiros *excel*: um ficheiro que contém a lista de atuais clientes, outro contendo a lista de atuais artigos para venda e um último que guarda as informações relativas à operação de uma venda. De seguida apresentam-se estes ficheiros dos quais se pretende extrair a informação neles contido de forma a manter no novo sistema o registo

das operações efetuadas no passado e sobre as quais ainda há registo. Os ficheiros a seguir apresentados, são meramente exemplificativo não sendo reais os dados neles contidos.

A Figura 20 mostra a forma como os clientes são guardados e disponibilizados a este comercial, sendo cada cliente identificado por um número que fica associado ao seu respetivo Número de Identificação Fiscal (NIF), nome, morada e endereço eletrónico.

	A	B	C	D	E	F	G	H	I
1	cli.num	cli.nif	Ter.NomeRed	cli.nome	cli.morada	cli.localidade	Ter.CPPais	email	
2	213	135691229	ABREU OPTICO	JOSE PEREIRA ABREU - CENTRO ÓPTICO,	AV. DA AZAMBUZA, 267	PAÇOS DE BRANDÁ	4535-266	PAÇOS D	abreu.oculos@gmail.com
3	239	504879120	BRANQUINHO	AQUAÓPTICA	RUA DO PELOURINHO, Nº 7 - LT.2	MARINHA GRANDE	2430-228	MARINHA	opmarinha@gmail.com
4	111	503789397	CASA AQUA	ANTÓNIO NEVES, MOURÃO, LDA.	RUA DO SOUTO, 120	S. JOÃO DO SOUTO	4700-329	BRAGA	optica.casa.souto@gmail.com
5									

Figura 20: Exemplo de Ficheiro de Clientes.

Os artigos são também mantidos num ficheiro, tendo cada um deles um código de barras, um código, uma descrição e dois Preço de Venda ao Público (PVP) recomendados.

	A	B	C	D	E	F
1	cbarras	codigo	descricao	pvp2	pvp4	
2	1000002508015	CSADELAIDE153	ARO COPPE SID ADELAIDE 1 MASSA/METAL PRETO 15	78,00	195,00	
3	1000002507810	CSCHARLOTTE345	NYLOR COPPE SID CHARLOTTE 3 MASSA VERMELHO 15	78,00	195,00	
4	1000002507889	CSCLEVELAND150	NYLOR COPPE SID CLEVELAND 1 MASSA PRETO 15	78,00	195,00	
5	1000002541340	CSFIRENZE353	ARO COPPE SID FIRENZE 3 MASSA AZUL/CINZA PADRÃO 15	78,00	195,00	
6	1000002323526	FF510253	1/2 LUA CAMPANHA 510 C2 MASSA CASTANHO	6,00	40,00	
7	1000002323557	FF510353	1/2 LUA CAMPANHA 510 C3 MASSA AZUL	6,00	40,00	
8	1000002513255	FFOP965253	ARO STAR OP965 2 METAL DOURADO 15	9,40	70,00	
9	1000002298848	PJS70714	SOL PJ CAMILLE 7071 4 MASSA ROSA 10	38,00	112,00	
10						

Figura 21: Exemplo de Ficheiro de Artigos Para Vendas.

Por último, como mostra a Figura 22, as vendas são registadas também em formato *excel* inicialmente composto por um cabeçalho que identifica o vendedor/comercial, e o cliente a quem serão vendidos os produtos. Após este cabeçalho encontram-se descritos em linhas os artigos, as suas respetivas quantidades para venda e um valor recomendado para venda a clientes finais. Este ficheiro contém ainda a quantidade total de artigos da venda, o valor total gerado pela venda, a percentagem de IVA que incide sobre cada artigo e a percentagem de desconto aplicado sobre a venda.

	C	D	E	F	G	I	J	K
1		N.º Encomenda:		12 797				
2		Data:		16/01/2015				
3								
4	António Afonso		Cliente:	ABREU OPTICO				
5	Dep. Comercial		Cod:	213				
6	antonio.afonso@oculos.pt	Telem: 916 068 865	Nome:	JOSE PEREIRA ABREU - CENTRO ÓPTICO, LDA				
7	Alameda da Maia, Lte23,5º andar		Morada:	AV. DA AZAMBUZA, 267				
8	2880-353 Sto António		Localidade:	PAÇOS DE BRANDÃO				
9	Telef: 289 898 989 Fax=269 878 979		Cod.Postal:	4535-266 PAÇOS DE BRANDÃO				
10	Contribuinte=599 233 655		Email:	abreu.oculos@gmail.com				
11								
12	Cod Artigo	Descrição	Qtd	Preço	Total		PVP recomendada	
13	XO6134NB150	1/2 LUA GRADUADA MASSA PRETO	1	9,00	9,00 		45,00 	
14	NYG18300250	1/2 LUA GRADUADA MASSA LILÁS	1	6,00	6,00 		36,00 	
15	FFOP921350	1/2 LUA STAR OP921 C3 METAL CINZA 10	1	6,50	6,50 		42,00 	
16	FFOP921250	1/2 LUA STAR OP921 C2 METAL CINZA CI	1	6,50	6,50 		42,00 	
17	FFOP955252	1/2 LUA STAR OP955 C2 MASSA PRETO/	1	7,90	7,90 		52,00 	
18	MOSS319253	ARO MOSS 319 2 MASSA ROXO/VIOLETA	1	18,00	18,00 		108,00 	
19		SubTotal	6		53,90 			
20		Desconto	0%		0,00 			
21				Total	53,90 			
22								
23		Ao valor total incide a taxa de Iva 23%						
24								

Figura 22: Exemplo de Ficheiro Utilizado Para o Registo de Venda.

Pretende-se com este projeto a adoção de uma ferramenta que permita a informatização de todo este processo de venda e evolução do processo de negócio da organização. Assim, esta ferramenta em conjunto com as estruturas de dados usadas para o armazenamento da informação, irá substituir as fontes de informação até agora utilizadas.

Depois de analisadas as soluções existentes no mercado conhecidas por resolver os problemas de gestão da atividade quotidiana das empresas e tendo-se verificado que estas tratam os problemas identificados no âmbito deste projeto, decidiu-se a adoção de uma solução já existente. A solução utilizada foi o *software* Primavera. Além de permitir todas as funcionalidades requeridas para a aplicação idealizada, esta decisão teve também em consideração o facto de este software, apesar de não ser totalmente gratuito, oferecer uma solução grátis (ainda que com um limite de faturação de 30 mil euros) e sem limite temporal de utilização, que viabiliza a fase de desenvolvimento do projeto. Este produto tem muito boa reputação a nível nacional e internacional, e possui uma considerável comunidade de utilizadores e parceiros integradores que facilita a solução em caso de contrariedade.

Assim sendo, a estrutura de dados do Primavera constituirá uma nova fonte de informação que passará a ser também armazenada no armazém de dados que irá ser criado para o armazenamento e análise das informações acerca das vendas desta organização.

4.3.2 Definição de Indicadores de Performance

No âmbito deste projeto, no sentido de definir os aspetos de negócio que mais necessitam de serem tidos em conta para a análise do negócio da organização, foi disponibilizada informação sobre o tipo de análises mais importantes e mais frequentes no contexto de avaliações. Esta partilha de informação foi feita principalmente através de conversas com os interessados pela análise dos dados e que melhor podem definir as métricas a serem tidas em conta pela solução, e pela análise aos atuais métodos utilizados para o efeito.

Assim, destas análises, foi possível retirar os seguintes indicadores:

- **A - Valor líquido de artigos vendidos:** consiste em obter informação sobre o valor líquido faturado por artigo vendido, por marca de artigo, do modelo de artigo, por família de artigo, por cliente, por área geográfica, por vendedor, por fornecedor e por unidade de tempo;
- **B - Valor bruto de artigos vendidos:** consiste em obter informação sobre o valor bruto faturado por artigo vendido, por marca de artigo, do modelo de artigo, por família de artigo, por cliente, por área geográfica, por vendedor, por fornecedor e por unidade de tempo;
- **C - Valor de desconto por artigo vendido:** consiste em obter informação sobre o valor descontado em artigos vendidos;
- **D - Valor em impostos:** consiste em obter informação sobre o valor de impostos gerado pela venda de artigos;
- **E - Número de artigos vendidos:** consiste em obter informação sobre o número total de artigos vendidos, por marca de artigo, do modelo de artigo, por família de artigo, por cliente, por área geográfica, por vendedor, por fornecedor e por unidade de tempo;

4.3.3 Análise do Negócio

No âmbito do projeto em questão, cada processo de negócio diz respeito a um artigo, produzido por um fornecedor, vendido por um vendedor a um cliente, numa determinada data. Pode-se dizer que todos estes domínios de informação vão ser transformados em dimensões.

Neste processo de venda, interessa à empresa que o vende saber se com estas vendas está a gerar muito ou pouco valor bruto e líquido, de forma a poder concluir se a sua produtividade está a ser ou não lucrativa. Interessa também saber que quantidades de que artigos os seus clientes compram, podendo aliciá-los com boas oportunidades de negócio e inclusive alguns descontos. Com esta informação podem também gerir da melhor forma os seus inventários e a sua necessidade ao nível de *stock*. Outro foco de interesse é saber o índice de rentabilidade dos seus comerciais/vendedores sabendo as suas vendas efetivas e as suas comissões por venda conseguida. Todas estas métricas foram considerados como factos de interesse para o negócio.

Tendo em conta os indicadores de desempenho apresentados na secção 4.3.2, pela a análise da tabela 10 é possível compreender a origem das dimensões e factos quando relacionados com as consultas descritas.

Tabela 10: Matriz de Relacionamento de Factos e Dimensões

Factos Dimensões	Valor Li- quido	Valor Bruto	Valor de Desconto	Valor em Imposto	Quantidades Vendidas
Artigo	A	B	C	D	E
Cliente	A	B			E
Vendedor	A	B			E
Fornecedor	A	B			E
Data	A	B			E

4.3.4 Arquitetura da Solução

Resolvido o problema da informatização da informação de negócio, o objetivo de desenvolvimento passa a ser armazenar e tratar apenas um conjunto de dados relacionados com a venda de produtos. Assim, será criado apenas um *data mart* para conter este domínio de informação, utilizando a estratégia *bottom-up*. Desta forma, os custos associados aos desenvolvimentos são significativamente reduzidos, facilitando o início do desenvolvimento e mantendo-se sempre a possibilidade de acrescentar novas áreas de informação ao longo do tempo que possuem significativamente o mesmo custo de desenvolvimento que o custo do *data mart* inicial, contribuindo para a escalabilidade da solução.

Um fator importante a ter em conta é a dimensão dos dados em questão. Dado que o negócio está em constante evolução, tendo atualmente um número considerável de artigos e clientes, o conjunto de informação a tratar, cresce com grande velocidade à medida que os dias passam. Desta forma, é necessário ter em conta que todo este armazenamento não prejudique o acesso à informação. Esta arquitetura permite a utilização de modelos dimensionais onde normalmente o relacionamento entre dados é baixo, permitindo assim obter uma menor complexidade e rapidez no acesso, dado o número reduzido de *joins* necessários contribuindo também para a fácil manutenção da solução.

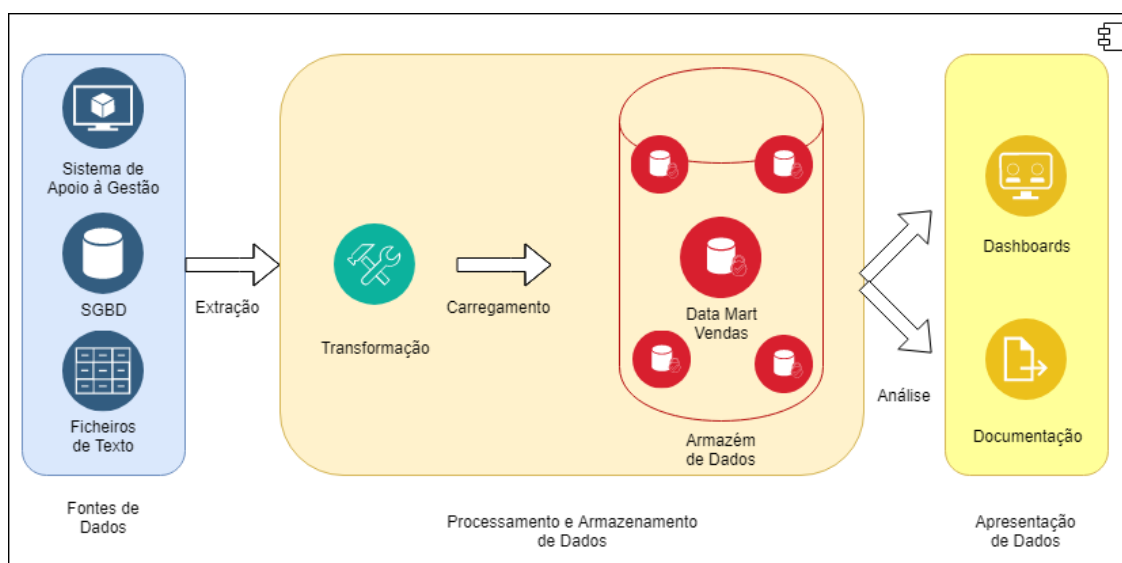


Figura 23: Arquitetura Geral do Armazém de Dados Previsto Para a Solução.

Na Figura 23, pretende representar-se o processo que irá decorrer desde a extração dos dados do sistema de gestão que irá inicialmente guardar os dados da solução até à análise e visualização do conhecimento que será extraído dos mesmos.

Na solução implementada a ferramenta de apoio à gestão utilizada é o Primavera, no entanto, pretende-se também a possibilidade de extração de dados deste sistema, ou daquele que esteja a ser utilizado, seja a partir de um outro repositório de dados previamente existente ou a partir de ficheiros, método até então em utilização. Até chegarem ao AD, os dados passam ainda por um processo de extração e limpeza, com o objetivo de organizar a informação de forma a ser guardada na nova infraestrutura de forma adequada, o *datamart* de vendas onde será carregada.

Com os dados no AD, pretende-se a sua visualização, fácil e rápida de forma a extrair destes dados o conhecimento associado ao negócio, seja de forma mais gráfica (via *dashboard*) ou pela criação de documentos/relatórios associados ao negócio.

No desenvolvimento desta arquitetura irá utilizar-se o *Pentaho Data Integration* para criar todo o *workflow* de carregamento e manutenção do AD. Optou-se por esta ferramenta, pois é grátis, multiplataforma e possui uma vasta comunidade de utilizadores o que se tornou muito útil em tempo de desenvolvimento da solução.

Quanto à visualização e análise destes dados escolheu-se para o efeito a ferramenta *kibana*. Além de ser também grátis e multiplataforma, o *kibana* é uma ferramenta de análise e visualização de dados, capaz de lidar com grandes volumes de dados e, que disponibiliza uma vasta gama de funcionalidades adicionais, entre elas técnicas de *machine learning* que poderão ser usadas, por exemplo para a deteção de anomalias nos dados. Estas funcionalidades apesar de não estarem a ser ainda utilizadas constituem uma excelente perspectiva para o futuro da solução e para o seu contínuo desenvolvimento.

4.3.4.1 Extração de Dados

Como dito anteriormente, irá utilizar-se o *software* Primavera como sistema de apoio à gestão da organização e, no caso em questão, para o registo de vendas.

Uma vez que o software usado é bastante complexo, irá apenas apresentar-se o resultado da análise às suas estruturas de dados e que irá suportar a construção de um armazém de dados. Este por sua vez servirá de base para as análises que se irão fazer sobre os dados destas vendas.

As estruturas de dados tidas em conta foram as seguintes:

- **CabecDoc:** armazena informações gerais sobre a venda, como por exemplo o seu comprador e os totais envolvidos na mesma;
- **LinhasDoc:** guarda a informação sobre cada linha de venda registada no sistema, contendo entre outros a identificação do artigo, a quantidade, desconto e taxas sobre este aplicado;
- **Artigo:** tem informações sobre as características de cada artigo, inclusivé o seu fornecedor, o seu código de barras, peso, volume, etc;

- **ArtigoMoeda:** armazena os dados sobre os possíveis preços de cada artigo; consiste numa desmultiplicação de cada artigo, permitindo que cada um possa ter vários preços e vários tipos de moeda associados;
- **Cliente:** consiste na informação acerca de um cliente no sistema. Tem informações sobre a morada do cliente, nome fiscal, NIF, entre outros;
- **Fornecedor:** informação sobre cada fornecedor, por exemplo a sua localização, contactos ou o seu limite de débito.
- **Vendedor:** guarda as informações de cada vendedor, como por exemplo a sua comissão sobre artigos vendidos.

A estrutura completa das estruturas usadas pelo Primavera podem ser consultadas na página do Primavera na internet.

Apesar de no processo atual de negócio algumas informações não estarem disponíveis, optou-se por ainda assim incluir alguns aspetos importantes para o quotidiano de uma qualquer organização, para que deste modo e estando disponíveis estes passem a ser utilizados e permitam a evolução dos processos atuais.

4.3.4.2 Staging Area

Numa primeira abordagem, é necessário efetuar a extração e o armazenamento dos dados, com o objetivo de os tratar antes de serem carregados para o sistema de armazenamento final. Assim, e tendo em conta a utilização do sistema de gestão Primavera, foi criada uma estrutura de armazenamento muito semelhante à do sistema de origem. Sendo este é um software bastante utilizado e testado no mundo do retalho cobrindo as mais variadas áreas de negócio, optou-se por manter na sua essência o modelo em utilização. Uma vez que o modelo utilizado é algo extenso, por uma questão de facilidade de visualização apresenta-se na Figura 24 uma versão simplificada da estrutura usada. A versão completa poderá ser consultada no anexo A. Na Figura 24, os valores a azul representam uma chave primária, os valores a negrito uma chave estrangeira e o símbolo '?' significa que o campo pode ser *NULL*. Apesar de este modelo apresentar as chaves primárias e estrangeiras, estas são apenas criadas após o carregamento de todas as estruturas, de modo a que a operação seja feita da forma mais rápida/eficiente possível. Após o processo de carregamento será feita a criação dessas mesmas chaves, sendo que, aquando do carregamento seguinte, estas devem ser de novo apagadas e o processo repete-se.

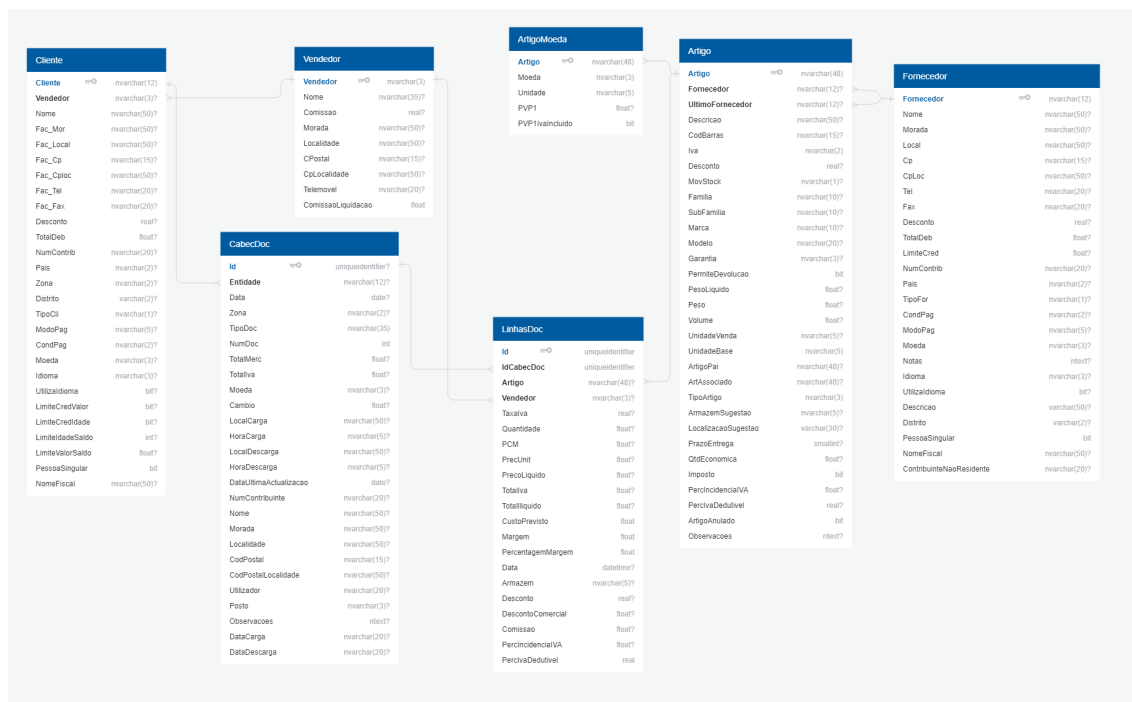


Figura 24: Versão Simplificada da Estrutura de Armazenamento de *Staging*.

Uma vez que a granularidade definida para este processo de negócio é referente às vendas de artigos, foram mantidas na área de *staging* as tabelas do sistema origem que guardam os dados referentes a esses acontecimentos de interesse para o negócio, sendo essas as tabelas "CabecDoc" e "LinhasDoc". Destas, a tabela "LinhasDoc" guarda os dados relativos a cada linha da venda, como por exemplo o preço bruto e o IVA sobre o artigo a que se refere essa linha, enquanto que a tabela "CabecDoc" guarda, entre outros, o comprador dos artigos em questão, o utilizador que registou a venda e os montantes totais envolvidos no processo de venda.

As restantes tabelas consideradas utilizam-se como forma de obter o detalhe que dá significado a cada venda:

- **Cliente:** fornece o detalhe de cada cliente.
- **Vendedor:** apresenta o detalhe acerca de cada vendedor.
- **Fornecedor:** fornece informação sobre cada fornecedor.
- **Artigo:** fornece detalhe sobre os artigos para venda, excepto acerca do preço do artigo que se encontra na tabela "ArtigoMoeda", assim como a moeda a que corresponde esse valor.

Para cada uma das tabelas apresentadas, foi criada uma tabela *Data Quality Problems (DQP)* para armazenar todos os registos problemáticos que surjam no decorrer do processamento de carregamento das respetivas tabelas de *staging*. Desta forma pretende-se conseguir identificar mais facilmente problemas existentes nas fontes de dados, podendo ser vir a ser validados e efetivamente carregados no sistema. Assim, cada tabela existente na área de *staging* possui a sua tabela *DQP* associada. As tabelas criadas para o armazenamento dos erros de carregamento das tabelas de *staging* encontram-se simplificada e ilustradas na Figura 25. A versão completa pode ser consultada no Anexo B.

O modelo de *DQP* desenvolvido apresenta-se sem relacionamentos entre tabelas, de forma a facilitar o carregamento dos dados, sem qualquer tipo de dependência uma vez que cada uma atua apenas como tabela de *log* de registos problemáticos.

Artigo		Cliente		Fornecedor	
Artigo	nvarchar(48)	Cliente	nvarchar(12)	Fornecedor	nvarchar(12)
Fornecedor	nvarchar(12)?	Vendedor	nvarchar(3)?	Nome	nvarchar(50)?
UltimoFornecedor	nvarchar(12)?	Nome	nvarchar(50)?	Morada	nvarchar(50)?
Descricao	nvarchar(50)?	Fac_Mor	nvarchar(50)?	Local	nvarchar(50)?
CodBarras	nvarchar(15)?	Fac_Local	nvarchar(50)?	Cp	nvarchar(15)?
Iva	nvarchar(2)	Fac_Cp	nvarchar(15)?	CpLoc	nvarchar(50)?
Desconto	real?	Fac_Tel	nvarchar(20)?	Tel	nvarchar(20)?
MovStock	nvarchar(1)?	Desconto	real?	Desconto	real?
Familia	nvarchar(10)?	NumContrib	nvarchar(20)?	NumContrib	nvarchar(20)?
SubFamilia	nvarchar(10)?	Pais	nvarchar(2)?	Pais	nvarchar(2)?
Marca	nvarchar(10)?	Distrito	varchar(2)?	Distrito	varchar(2)?
Modelo	nvarchar(20)?	PessoaSingular	bit	PessoaSingular	bit
QtdEconomica	float?	NomeFiscal	nvarchar(50)?	NomeFiscal	nvarchar(50)?
Imposto	bit	NrErros	int?	NrErros	int?
Observacoes	ntext?	DescricaoErro	nvarchar(1000)?	DescricaoErro	nvarchar(1000)?
NrErros	int?	CampoErro	nvarchar(1000)?	CampoErro	nvarchar(1000)?
DescricaoErro	nvarchar(1000)?	CodigoCampoErro	nvarchar(1000)?	CodigoCampoErro	nvarchar(1000)?
CampoErro	nvarchar(1000)?				
CodigoCampoErro	nvarchar(1000)?				

LinhasDoc		ArtigoMoeda		CabecDoc	
Id	uniqueidentifier	Artigo	nvarchar(48)	Id	uniqueidentifier?
IdCabecDoc	uniqueidentifier	Moeda	nvarchar(3)	Entidade	nvarchar(12)?
Artigo	nvarchar(48)	Unidade	nvarchar(5)	Data	date?
Vendedor	nvarchar(3)	PVP1	float?	Zona	nvarchar(2)?
Taxalva	real?	PVP1IvaIncluido	bit	TipoDoc	nvarchar(35)
Quantidade	float?	NrErros	int?	NumDoc	int
PCM	float?	DescricaoErro	nvarchar(1000)?	TotalMerc	float?
PrecUnit	float?	CampoErro	nvarchar(1000)?	TotalIva	float?
PrecoLiquido	float?	CodigoCampoErro	nvarchar(1000)?	NumContribuinte	nvarchar(20)?
TotalIva	float?			Nome	nvarchar(50)?
TotalLiquido	float?			Utilizador	nvarchar(20)?
Margem	float			NrErros	int?
Data	datetime?			DescricaoErro	nvarchar(1000)?
Desconto	real?			CampoErro	nvarchar(1000)?
NrErros	int?			CodigoCampoErro	nvarchar(1000)?
DescricaoErro	nvarchar(1000)?				
CampoErro	nvarchar(1000)?				
CodigoCampoErro	nvarchar(1000)?				

Vendedor	
Vendedor	nvarchar(3)
Nome	nvarchar(35)?
Comissao	real?
Morada	nvarchar(50)?
Localidade	nvarchar(50)?
CPostal	nvarchar(15)?
NrErros	int?
DescricaoErro	nvarchar(1000)?
CampoErro	nvarchar(1000)?
CodigoCampoErro	nvarchar(1000)?

Figura 25: Versão Simplificada das Tabelas *DQP* de *Staging*.

Com este modelo, ao ocorrer um erro no fluxo de carregamento de uma qualquer tabela de *staging*, juntamente com os dados que deveriam ter sido carregados, são também guardados no campo "NrErros" o número de erros detetados, a descrição do(s) erro(s) são registados no campo "DescricaoErro", guarda-se também o campo que originou o erro em "CampoErro", e ainda o código do erro na coluna "CodigoCampoErro". A partir deste conjunto de tabelas torna-se possível detetar erros facilmente podendo-se esta traduzir numa importante ferramenta de captação de erros muito útil em fase de desenvolvimento, e que se poderá traduzir em muito tempo ganho a esse nível. Além disto, posteriormente, e já numa fase após a criação do armazém de dados, estas estruturas serão capazes de registar todos os problemas associados aos dados cujo processo de inserção no mesmo falha. Desta forma, este mecanismo garante a qualidade dos dados que fazem parte do armazém de dados.

4.3.4.3 Armazém de Dados

Depois da extração da informação do sistema operacional para a área de *staging*, essa mesma informação é armazenada no AD. Foi criado o modelo dimensional presente na Figura 26, baseado na topologia em estrela, como resposta aos requisitos identificados na análise da solução (Secção 4.3.3) e para que seja possível dar resposta às necessidades de análise aos indicadores de performance também definidos anteriormente (Secção 4.3.2). A versão completa do modelo dimensional criado pode ser consultada no Anexo C.

Para a definição do AD seguiu-se a metodologia de *Kimball*:

- **Área de negócio:** Vendas de produtos a clientes;
- **Granularidade:** Vendas por produto, por cliente, por instante de tempo (data e hora ao segundo), por vendedor e por fornecedor;
- **Dimensões:** Data, Hora, Fornecedor, Produto, Vendedor, e Cliente;
- **Factos:** Quantidade vendida, valor bruto e liquido da venda, valor unitário, valor em imposto, valor de desconto, comissão, margem e lucro em Euros;

Uma vez que a solução se encontra ainda numa fase de validação, sujeita a correções e melhorias, optou-se por manter as dimensões que constituem o AD com algum detalhe, mantendo em grande parte a estrutura da fase de *staging*, ainda que alguns destes campos não sejam ainda utilizados uma vez que muitos não eram possíveis de ter em conta nos processos até agora utilizados. Como os processos até então utilizados eram muito básico, confinando-se apenas ao essencial do negócio, muitos aspetos tidos em conta pelas estruturas do Primavera não tinham até agora uso no processo da organização. Por isso, optou-se por deixar alguns desse campos nas estruturas do armazém de dados desenvolvido, para que no futuro estes possam fazer parte dos processos da organização.

Além destes, adicionaram-se quatro novos campos. Adicionou-se então, uma chave numérica a cada dimensão, o campo "atual", booleano, que indica se o registo é o mais atual ou não, e ainda dois novos campos do tipo data, que registam as datas de início e de fim de validade de cada registo.

Além das tabelas já existentes na fase anterior, criou-se uma dimensão data de modo a abranger as necessidades de tipos de data. Optou-se por uma granularidade diária.

Quanto à tabela de factos, a tabela "FactVenda" regista todas as vendas consumadas. Definiu-se que um registo de venda é considerado único no sistema apenas a partir da

conjugação do cliente, artigo, data e hora. Assim, a venda de um artigo não pode ter dois registos iguais para o mesmo cliente, artigo e para a mesma data e hora de venda. Esse artigo fica então associado ao vendedor que registou a venda, venda essa realizada a um cliente numa determinada data e hora. É registado também o preço líquido e bruto do artigo vendido, assim como o valor de imposto que sobre ele incide, a quantidade vendida, o seu preço de custo médio, o desconto e a comissão com que foi vendido, a margem em valor monetário e o lucro resultante para a organização dessa venda.



Figura 26: Modelo Dimensional Simplificado.

Neste caso, aquando da criação das estruturas apresentadas para o *data mart*, apenas as chaves primárias de cada tabela são também definidas, sendo as chaves estrangeiras criadas apenas quando o processo de carregamento da tabela de factos acaba. Uma vez que esta estrutura é criada apenas uma vez, de cada vez que se efetua um carregamento da tabela de factos estas chaves estrangeiras são previamente eliminadas e recriadas no fim do processo.

Depois de montado e carregado o AD, passa a ser possível analisar a informação nele contida de modo a extrair o conhecimento do mesmo.

5. Implementação

O presente capítulo tem como objetivo retratar as operações realizadas durante o desenvolvimento deste projeto ao longo das suas diversas fases. Começa-se por descrever os passos seguidos para a configuração da ferramenta de gestão utilizada, de seguida explica-se o modo como foi criado o armazém de dados, área de *staging* e *datamart*, e o processo de migração dos dados contidos nos ficheiros anteriormente utilizados para a estrutura de dados da ferramenta de gestão adotada. Por último, descreve-se o processo que levou à apresentação dos dados contidos no armazém de dados criado.

5.1 Ferramenta de Gestão

Para resolver o problema da informatização da gestão, das atividades da organização em estudo, foi utilizado o *software* de gestão Primavera *Professional* na versão 8.

Visto que esta ferramenta já oferece todas as funcionalidades que eram requisitos para a solução não houve necessidade de implementar nada de novo. Assim, de modo a satisfazer na totalidade as regras de negócio em questão foi apenas necessário configurar a ferramenta de acordo com a análise realizada.

5.1.1 Idioma

Um dos requisitos não funcionais definidos para a aplicação foi a capacidade desta suportar múltiplas linguagens, de modo a permitir alguma flexibilidade quanto à língua dos seus utilizadores. Este requisito é tido em conta pelo Primavera, sendo possível de definir diferentes idiomas, quer ao nível da organização em questão, quer ao nível dos utilizadores da mesma. A Figura 27, mostra a opção que permite escolher o idioma da organização.

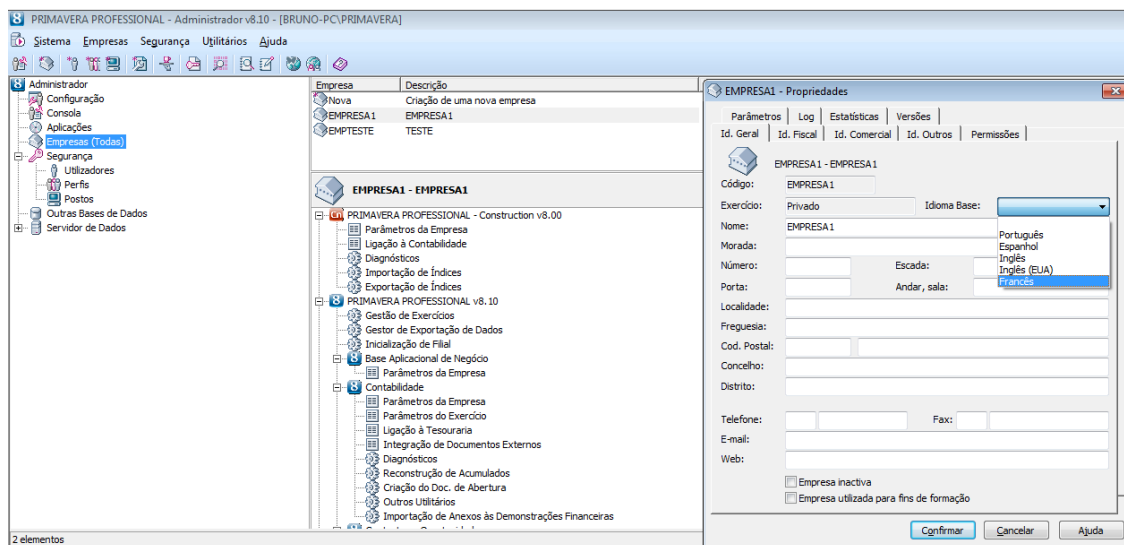


Figura 27: Ecrã de Configuração Para Escolha de Idioma.

5.1.2 Autenticação e Autorização

Outro requisito definido foi a gestão de acessos por parte dos utilizadores ao sistema em questão. Este requisito é também garantido pela solução escolhida. A Figura 28 representa o controlo de acesso à aplicação Primavera.



Figura 28: Ecrã de Login.

Além de garantir a autenticidade dos utilizadores é também possível definir o nível de acessos de cada utilizador. A Figura 29 apresenta a configuração efetuada para um utilizador com o papel de Armazenista na aplicação. Este apenas deverá ter acesso aos módulos de venda e inventário, uma vez que apenas tem a responsabilidade de gerir os *stocks* da empresa (operação suportada pelo inventário) e validar encomendas (operação suportada pelas vendas).

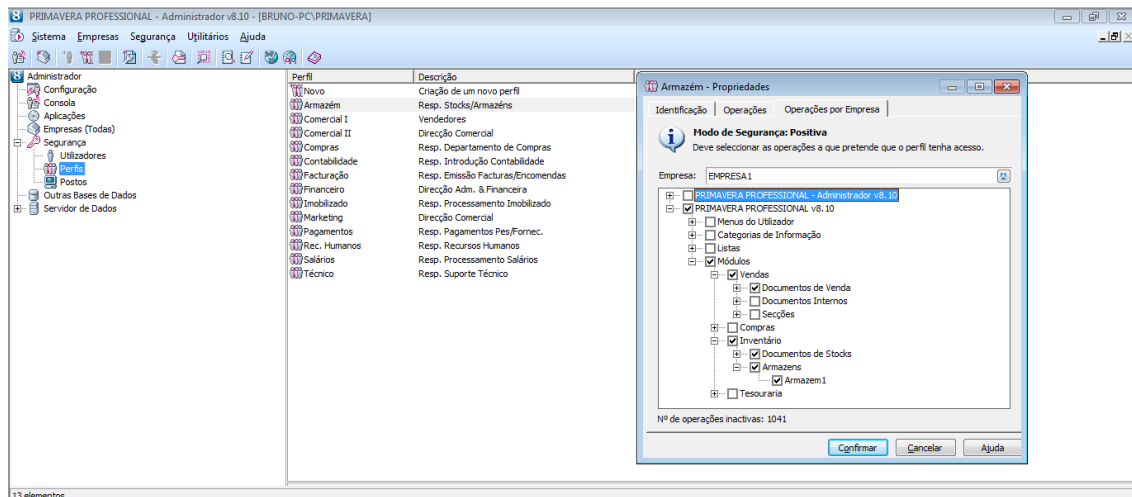


Figura 29: Ecrã de Configuração de Acessos.

A Figura 30 mostra as funcionalidades disponíveis para o utilizador 'Armazenista1' na aplicação.



Figura 30: Ecrã de Funcionalidade Disponíveis Para o Utilizador Armazenista.

5.1.3 Gestão de Utilizadores

O Primavera suporta também a gestão de utilizadores da aplicação, sendo possível atribuir-lhes os perfis e as suas permissões para cada empresa a que têm acesso, como se pode ver na Figura 31.

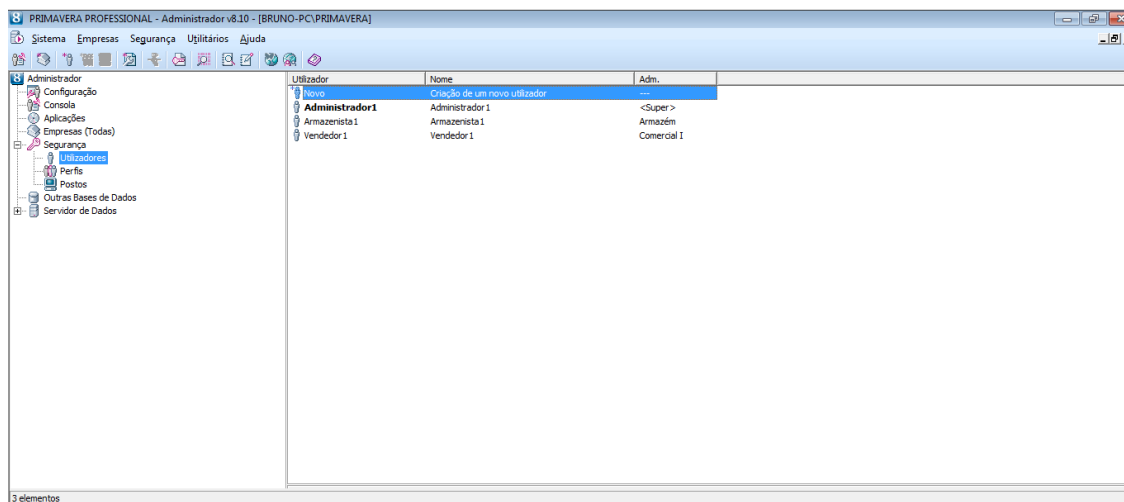


Figura 31: Ecrã Para Gestão de Utilizadores da Aplicação.

5.1.4 Gestão de Artigos

Um requisito apontado aos utilizadores do tipo Administrador é a capacidade de gerir os artigos disponíveis para venda. Essa funcionalidade é demonstrada na Figura 32.

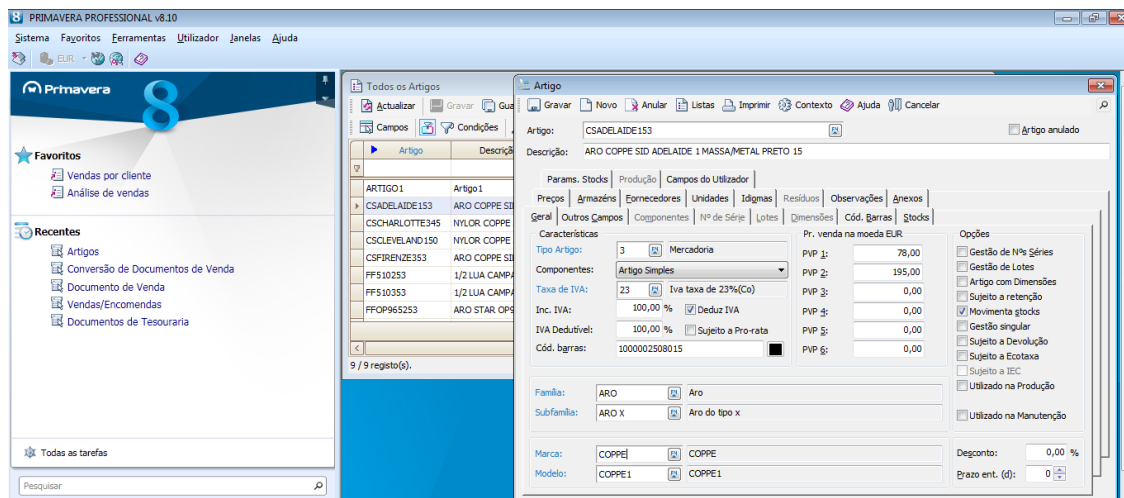


Figura 32: Ecrã Para Gestão de Artigos da Aplicação.

5.1.5 Criação de Encomendas

Mais um requisito fundamental é a criação de encomendas por parte dos comerciais da empresa. O ecrã que permite essa funcionalidade é o representado na Figura 33.

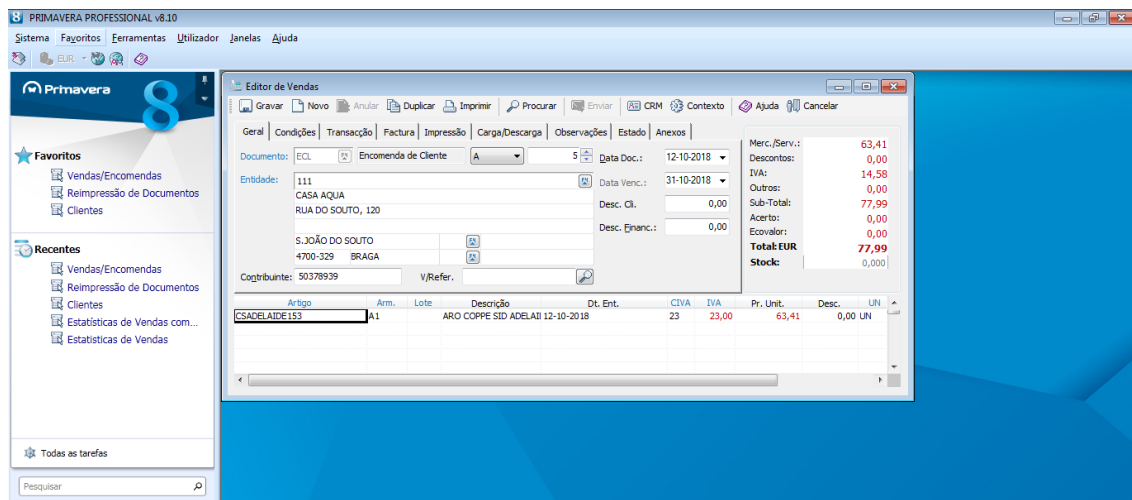


Figura 33: Ecrã Para Criação de Encomenda.

5.1.6 Validação de Encomendas

Após a criação de uma encomenda por parte de um comercial, esta deve ser validada por um armazenista, que deve garantir que os artigos encomendados existem em *stock* para que se possa gerar a fatura referente à venda desses mesmos produtos. O processo que permite validar uma encomenda apresenta-se na Figura 34.

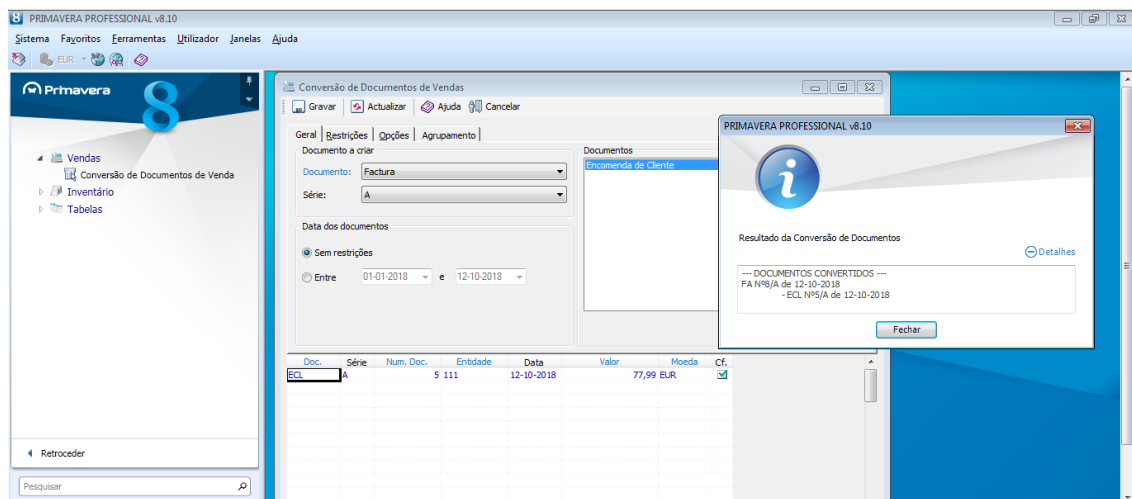


Figura 34: Ecrã Para Validação de Encomenda.

5.1.7 Geração de Documentação

Depois de válida, uma encomenda é então transformada numa fatura, documentos estes que podem ser consultados, analisados e impressos assim que for necessário. Um exemplo de consulta e impressão de um documento de venda é apresentado na Figura 35.

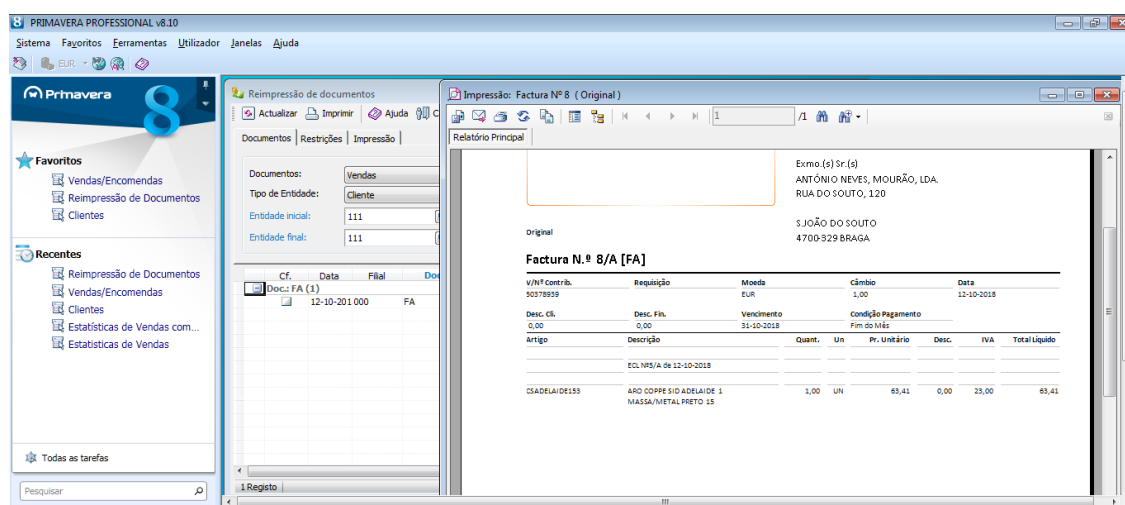


Figura 35: Ecrã de Consulta de Documentos Relativos a Vendas.

5.1.8 Atualização de Stocks

Outra responsabilidade de um armazenista consiste em manter atualizados os *stocks* dos artigos em circulação e disponíveis para venda. A forma disponibilizada a estes utilizadores para o fazer é apresentada pela Figura 36.

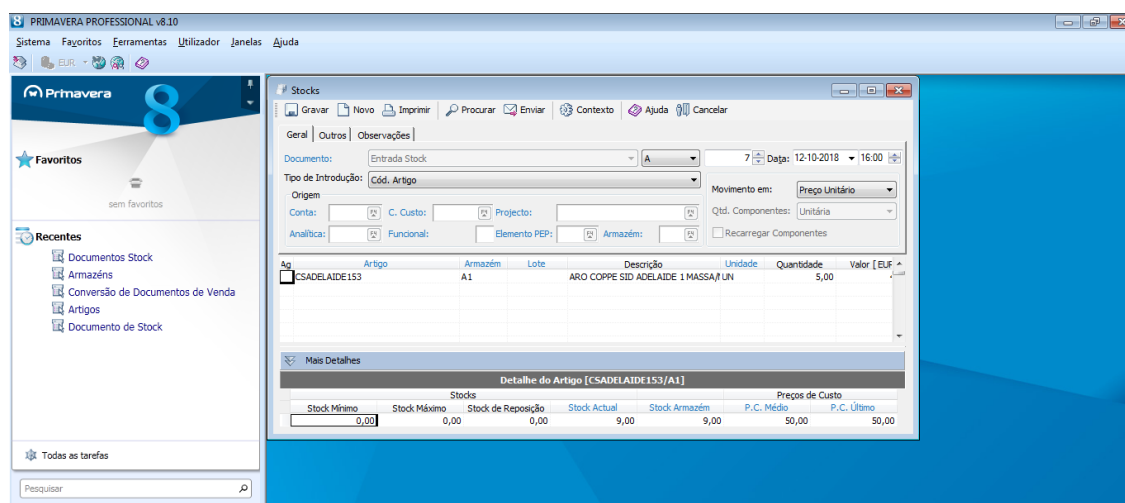
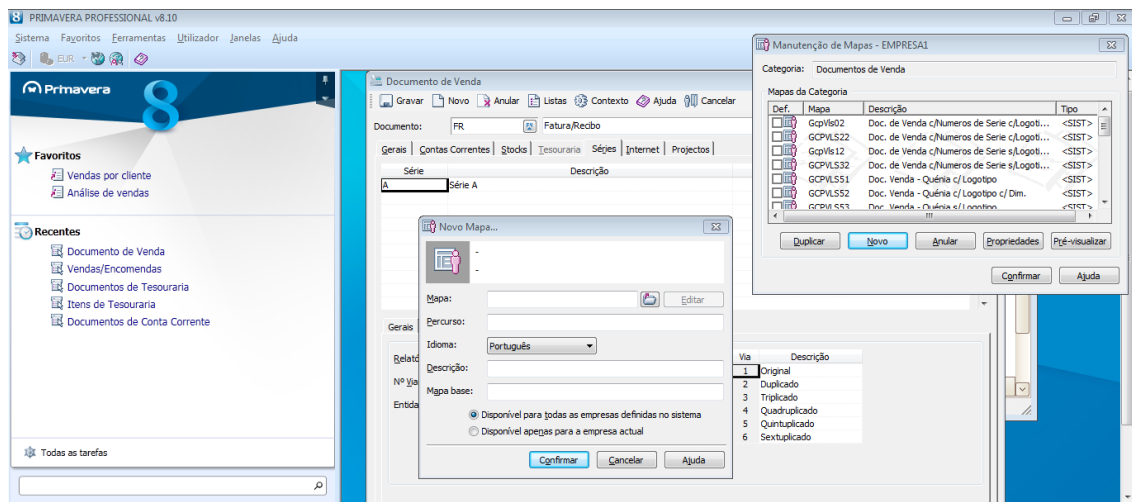


Figura 36: Ecrã Para o Registo de Entradas de *Stock*.

5.1.9 Gestão de Template de Documento

Outra função desejada para a aplicação consistia na possibilidade de gerir os tipos de documento gerados pelas diferentes ações efetuadas no sistema. Para isto, o Primavera possibilita a criação de *templates*, além dos existentes por defeito. A Figura 37 mostra a forma de realizar essa operação.

Figura 37: Ecrã Para o Registo *Templates* de Documento.

5.1.10 Criação de Alertas

Por último, outro objetivo de funcionamento para a aplicação consistia na possibilidade de esta apresentar aos comerciais ou vendedores, uma espécie de alerta que permitisse a estes utilizadores perceberem quando acontecesse um certo evento, ou mais especificamente uma métrica configurada por si fosse atingida ou ainda a segmentação dessa métrica segundo um determinado critério. A Figura 38 mostra um exemplo do tipo de informação pretendida com a definição de um alerta no Primavera.

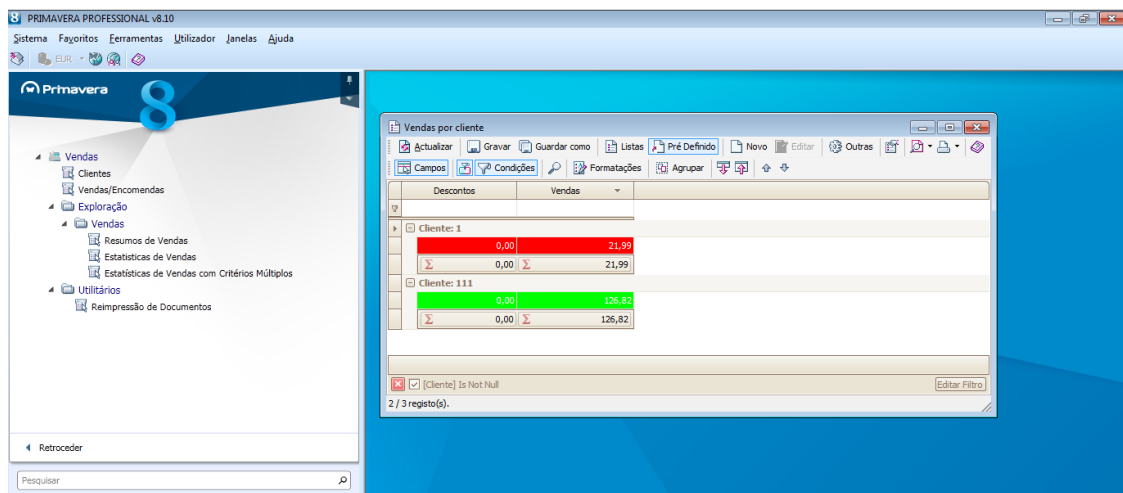


Figura 38: Ecrã de Alerta.

5.2 Criação do Armazém de Dados

No desenvolvimento do Armazém de Dados utilizou-se o *Pentaho Data Integration* para criar todo o *workflow* de carregamento e manutenção do AD. Optou-se por esta ferramenta, pois

é grátis, multiplataforma e possui uma vasta comunidade de utilizadores, o que se tornou muito útil em tempo de desenvolvimento da solução.

5.2.1 Ficheiro de Propriedades

De forma a manter a solução organizada e de fácil manutenção, optou-se por fazer uso de um ficheiro (Figura 39) onde estão definidas algumas propriedades que se irão utilizar ao longo da criação do AD. Estas propriedades funcionam como variáveis globais à aplicação e poderão ser acedidas em qualquer parte do processo. Neste ficheiro inserem-se por exemplo os nomes e acessos às bases de dados envolvidas no processo, o caminho para os *scripts* de criação do AD, entre outros. Desta forma, caso alguma destas variáveis se modifique, a sua alteração será facilmente efetuada garantindo o correto funcionamento do processo.

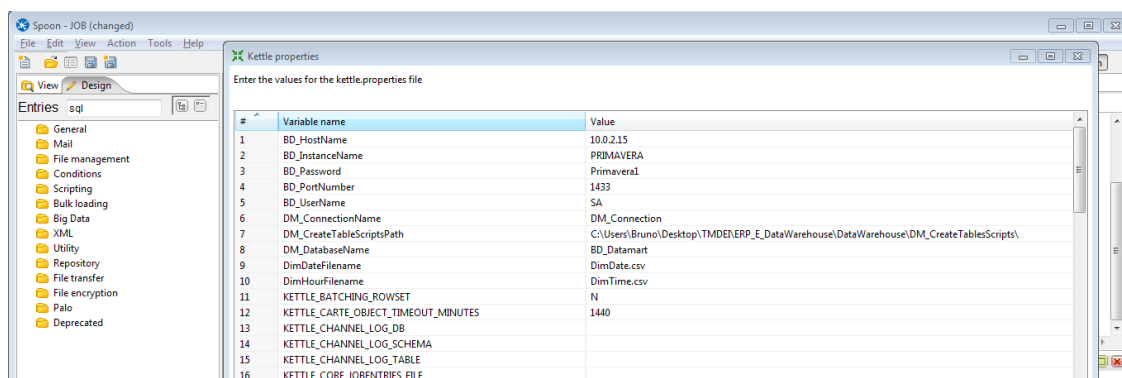


Figura 39: Ficheiro de propriedades *Pentaho*.

5.2.2 Processo de ETL

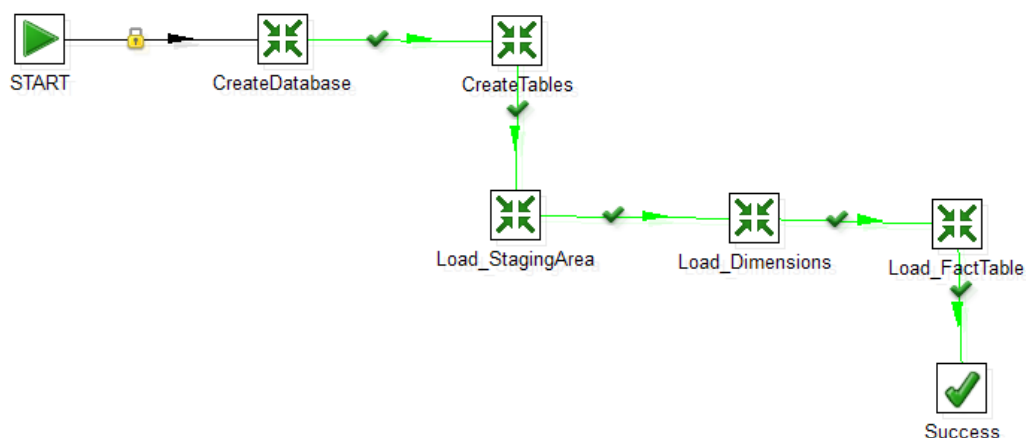


Figura 40: *Overview* geral do processo *ETL* implementado.

O processo (Figura 40) inicia-se pela criação das estruturas de dados, numa fase inicial das bases de dados (*Staging* e *Datamart*) e posteriormente de cada uma das suas tabelas e das

suas especificidades: chaves primárias e estrangeiras, *constraints*, índices, entre outros. Estas operações devem ser realizadas apenas uma única vez, pois uma vez que se pretende manter um histórico das atividades quotidianas da organização, os dados terão que ser mantidos, não podendo ser recriadas as estruturas de cada vez que o processo é executado. No caso do *Datamart*, este processo apresenta a seguinte particularidade: antes de efetuar o carregamento dos dados, eliminam-se as restrições associadas às chaves estrangeiras da tabela de factos, e só após ter ocorrido o carregamento dos dados, é que se deve voltar a recriar estas restrições. Isto faz-se como uma medida de otimização que é feita em cada carregamento de dados.

Criadas as estruturas, procede-se ao carregamento das tabelas da área de *Staging*. Nesta fase, os dados são carregados diretamente das tabelas geridas pelo Primavera para as novas tabelas. Estas contêm também todas as propriedades das tabelas iniciais, pois apesar de atualmente não existirem dados para abastecer grande parte dos campos utilizados no Primavera, espera-se que pela evolução nos processos que atualmente vigoram, graças também à adoção do sistema Primavera, estes processos passem a contemplar os campos já tidos em conta nas referidas estruturas.

Como referido anteriormente numa das secções do capítulo de *Design* e Arquitetura da Solução (Secção 4.3.4.2), definiu-se um modelo de dados com o intuito de captar os eventuais problemas associados aos dados. Assim, devido ao facto do modelo escolhido para a *Staging Area* ser semelhante ao do sistema origem, os dados não deverão apresentar grandes problemas, no entanto quis-se preparar desde já o processo para essa eventualidade. Apresenta-se um exemplo, Figura 41, relativo à captação de problemas associados à tabela Artigo. Os dados são extraídos da tabela origem e são posteriormente adicionados à respetiva tabela da área de *Staging*. Os registos que não puderem ser inseridos nessa tabela terão que apresentar algum problema, o que provocará a falha do processo. Essa falha é captada no fluxo de erro do componente que tenta inserir o registo na tabela destino e é encaminhado para o componente seguinte que fará o seu registo na tabela para o efeito.

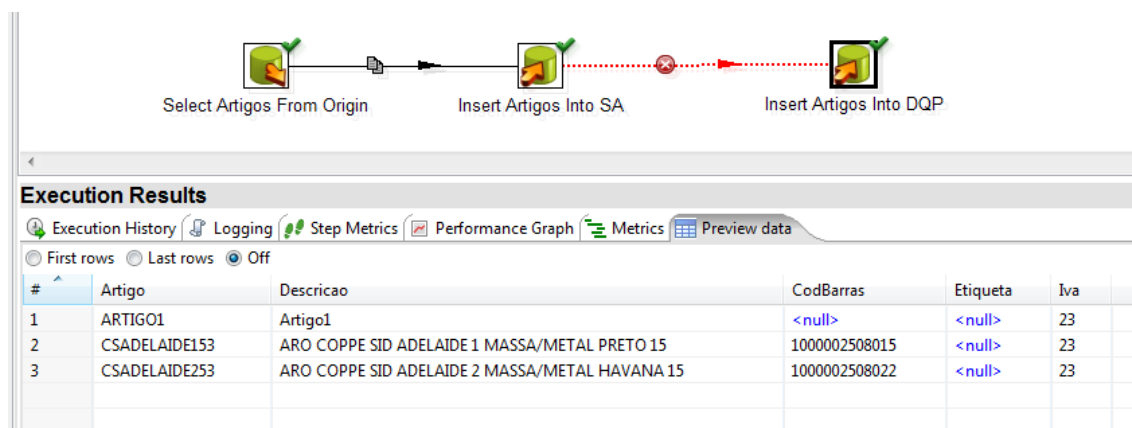


Figura 41: Fluxo de *DQP* Implementado Para a Tabela Artigo.

Além dos campos originais que levaram a essa falha, são registados (sempre que possível) 4 novos campos (Figura 42):

- *NrErrors*: guarda o número de erros que o registo apresenta;
- *ErrorDescription*: guarda a descrição do erro associado à falha;

- *ErrorFieldname*: guarda o nome do campo que originou o erro;
- *ErrorCodesFieldname*: guarda o código do erro para o campo que o originou;

The screenshot shows a SQL query editor with the following query:

```

2 SELECT TOP 1000 [Artigo]
3     , [NrErrors]
4     , [ErrorDescription]
5     , [ErrorFieldname]
6     , [ErrorCodesFieldname]
7     , [Descricao]
8     , [CodBarras]
9     , [Etiqueta]
10    , [Iva]
11    , [Desconto]
12    , [Fornecedor]
13    , [MovStock]

```

Below the query editor, the 'Results' tab is active, displaying a table with the following data:

	Artigo	NrErrors	ErrorDescription	ErrorFieldname	ErrorCodesFieldname
1	ARTIGO1	1	org.pentaho.di.core.exception.KettleDatabaseExce...	NULL	TOP001
2	CSADELAIDE153	1	org.pentaho.di.core.exception.KettleDatabaseExce...	NULL	TOP001
3	CSADELAIDE253	1	org.pentaho.di.core.exception.KettleDatabaseExce...	NULL	TOP001

Figura 42: Registos Criados Pelo Processo de DQP.

Depois de extraídos os dados do sistema origem para a área de *Staging*, estes passarão por uma série de passos, até que sejam carregados no armazém de dados, mais propriamente no *Datamart* relativo a vendas de artigos.

Nesta fase, antes de serem carregados no *Datamart*, os dados devem passar por uma fase de tratamento de modo a que se validem e que sejam detetados potenciais inconsistências nos dados que existem na área de *Staging* e que por sua vez existem no sistema origem. No entanto, não existem de momento regras, nem dados suficientemente validados para definir corretamente uma estratégia de tratamento, pelo que os dados serão aceites como se encontrarem na área de *Staging*.

Um requisito inicial foi a possibilidade dos utilizadores poderem efetuar análises tendo com base o histórico, esta é uma característica importante para o carregamento dos dados no *Datamart*. Assim, é importante definir quais as características para as quais importa guardar histórico. Para isto usou-se o conceito de *SCD*. As dimensões "DimHora" e "DimData" não foram contempladas neste processo uma vez que não irão sofrer alterações ao longo do tempo. As estratégias definidas para as dimensões que o exigem encontram-se em anexo (Anexo G).

Assim, ao carregar os dados da base de dados de *Datamart*, é feita uma comparação entre os registos a inserir e os registo já existentes, operação efetuada no *Pentaho* com recurso a um *Lookup* (Figura 43). Caso o registo não exista, o mesmo é adicionado, sendo-lhe gerada uma *key*, ou identificador, e três campos de controlo: *dataInicio*, *dataFim*, *atual*. Estes têm os seguintes propósitos:

- *dataInicio*: data de início do período de vigência do registo;
- *dataFim*: data de fim do período de vigência do registo;
- *atual*: *flag* que indica se o registo é o mais atual.

Step name: Dimension lookup/update DIM_Artigo

Update the dimension?

Connection: \${DM_ConnectionName} [Edit...] [New...] [Wizard...]

Target schema: [Browse...]

Target table: DimArtigo [Browse...]

Commit size: 1000

Enable the cache?

Pre-load the cache?

Cache size in rows (0 = cache all): []

Keys Fields

Lookup/Update fields

#	Dimension field	Stream field to compare with	Type of dimension update
1	Descricao	Descricao	Update
2	CodBarras	CodBarras	Update
3	Iva	Iva	Insert
4	Desconto	Desconto	Insert
5	Familia	Familia	Insert
6	Marca	Marca	Insert

Technical key field: ArtigoKey [New name:]

Creation of technical key:

- Use table maximum + 1
- Use sequence []
- Use auto increment field

Version field: atual

Stream Datefield: []

Date range start field: dataInicio [Min. year: 1900]

Use an alternative start date? [<Select Option>] []

Table date range end: dataFim [Max. year: 2199]

[OK] [Cancel] [Get Fields] [SQL]

Figura 43: Estratégia de SCD aplicadas à dimensão "DimArtigo" no *Pentaho*.

Caso o registo a inserir já exista, então as estratégias de SCD serão aplicadas. No caso do campo estar marcado com uma estratégia de tipo 1, o registo irá sofrer uma simples atualização, onde será atualizado para o novo valor. No caso se estar definida com tipo 2, o registo será duplicado (terá um novo identificar ou "key") mas, no novo registo, o campo deverá ter o novo valor. Adicionalmente, no registo que passa a ser o mais antigo, o campo "atual" passará a estar a falso, e será adicionada a data em que foi sobreposto, no campo "dataFim". Quanto ao novo registo, ficará preenchida a "dataInicio" com a data em que foi criado, o campo "atual" ficará preenchido de acordo (verdadeiro) e o campo "dataFim" ficará a "null", uma vez que é o mais recente, ou seja, é o que se encontra em vigor..

Ao nível da ferramenta utilizada, o *Pentaho*, o correspondente a uma estratégia de SCD de tipo 1 é o modo "Update" e o tipo 2 é representado pelo modo "Insert".

Na Figura 44 são visíveis as alterações efetuadas pelo processo de *ETL* às dimensões "DimArtigo" e "DimCliente", decorrentes de duas alterações, uma de tipo 2 e outra de tipo 1 às referidas tabelas. Do lado esquerdo, apresenta-se uma alteração ao campo "Marca" assinalado com uma estratégia de *SCD* de tipo 2, relativamente ao Artigo "CSADELAIDE153". Esta alteração fez com que, ao executar o processo de *ETL* desenvolvido, fosse criado um novo registo na tabela "DimArtigo" em tudo semelhante ao anteriormente existente aplicando as alterações associadas ao tipo de *SCD* definido. No registo que passa a ser o antigo, foi preenchido o campo "dataFim" com o valor da data em que deixou de estar em vigor e o campo atual passou de "1" a "0", significando assim que deixou de ser o mais atual. O novo registo, que passa a ser o atual, fica com o campo preenchido dizendo que é o mais atual (valor "1"), o campo "dataInicio" com a data de criação e o campo "dataFim" fica a "NULL" significando que está em vigor. Por sua vez, do lado direito, apresenta-se o resultado de uma alteração do tipo 1, realizada no campo "Fac_Cp" da tabela "DimCliente", ao Cliente "NOVOPTICA". Como se trata de uma operação do tipo 1, não é criado nenhum registo novo, apenas é atualizado o campo referente à alteração necessária.

Desta forma, pela implementação deste sistema, é possível realizar análises às vendas registadas tendo diferentes pontos de vista dos dados em questão. Pelo exemplo apresentado, passa a ser possível ter em consideração a marca do artigo vendido em diferentes circunstâncias, por exemplo passa a ser possível perceber a marca do Artigo em questão aumenta o volume de vendas do mesmo. Desta forma, poderá detetar-se uma preferência, e com base nisso dar prioridade aos artigos com essa característica.

The figure shows two columns of SQL queries and their results. The left column shows a Type 2 SCD operation on the DimArtigo table, and the right column shows a Type 1 update on the DimCliente table.

Left Column (DimArtigo):

- Query 1: `SELECT [Artigo], [Descricao], [CodBarras], [Marca] FROM [dbo].[Artigo] WHERE [Artigo] = 'CSADELAIDE153'`
- Result 1:

Artigo	Descricao	CodBarras	Marca
CSADELAIDE153	ARO COPPE SID ADELAIDE 1 MASSA/METAL PRETO 15	1000002508015	COPPE
- Query 2: `SELECT [Artigo], [Descricao], [CodBarras], [Marca] FROM [dbo].[Artigo] WHERE [Artigo] = 'CSADELAIDE153'`
- Result 2:

Artigo	Descricao	CodBarras	Marca
CSADELAIDE153	ARO COPPE SID ADELAIDE 1 MASSA/METAL PRETO 15	1000002508015	MOSS
- Query 3: `SELECT TOP 1000 [ArtigoKey], [Artigo], [Descricao], [Marca], [dataInicio], [dataFim], [atual] FROM [BD_Datamart].[dbo].[DimArtigo] WHERE [Artigo] = 'CSADELAIDE153'`
- Result 3:

ArtigoKey	Artigo	Descricao	Marca	dataInicio	dataFim	atual
3	CSADELAIDE153	ARO COPPE SID ADELAIDE 1 MASSA/MET...	COPPE	2019-01-09	2019-08-07	0
1051	CSADELAIDE153	ARO COPPE SID ADELAIDE 1 MASSA/MET...	MOSS	2019-08-07	NULL	1

Right Column (DimCliente):

- Query 4: `SELECT TOP 1000 [Cliente], [Nome], [Fac_Mor], [Fac_Cp] FROM [PRIEMPRESA1].[dbo].[Clientes] WHERE [Nome] = 'NOVOPTICA'`
- Result 4:

Cliente	Nome	Fac_Mor	Fac_Cp
102	NOVOPTICA	RUA LUIS DE CAMÕES,nº 5	3750-159
- Query 5: `SELECT TOP 1000 [Cliente], [Nome], [Fac_Mor], [Fac_Cp] FROM [PRIEMPRESA1].[dbo].[Clientes] WHERE [Nome] = 'NOVOPTICA'`
- Result 5:

Cliente	Nome	Fac_Mor	Fac_Cp
102	NOVOPTICA	RUA LUIS DE CAMÕES,nº 5	3750-179
- Query 6: `SELECT TOP 1000 [ClienteKey], [Cliente], [Nome], [Fac_Mor], [Fac_Cp], [dataInicio], [dataFim], [atual] FROM [BD_Datamart].[dbo].[DimCliente] WHERE [Nome] = 'NOVOPTICA'`
- Result 6:

ClienteKey	Cliente	Nome	Fac_Mor	Fac_Cp	dataInicio	dataFim	atual
3	102	NOVOPTICA	RUA LUIS DE CAMÕES,nº 5	3750-179	2019-01-09	NULL	1

Figura 44: Resultado das Operações de *SCD* à "DimArtigo" e "DimCliente" no *Pentaho*.

Carregadas as dimensões poderá iniciar-se o carregamento da tabela de factos, chamada de "FactVenda". A tabela de factos é composta pela referência para as dimensões que compõe o modelo dimensional assim como os factos, adicionados mediante a sua necessidade, necessidade essa gerada pela análise dos requisitos definidos. Para isto, como se pode ver na Figura 45, começou-se por reunir os dados relativos às vendas registadas em *Staging*, incluindo as informações associadas ao cliente que efetuou a compra, o vendedor

que registou a venda, o artigo vendido e o seu respetivo fornecedor, assim como a data e a hora em que se registou essa venda. Para cada um destes realizou-se um *Lookup* com o objetivo de descobrir a *key* associada ao registo da dimensão onde esse registo se encontra armazenado, e que será relacionado com a venda em questão na tabela de factos. Cada um destes *Lookup's*, tem a si associado um passo de validação para garantir que essa *key* é encontrada, caso contrário o registo terá um problema associado. Uma vez que no passo inicial se reúnem todas as informações associadas à venda, é também neste ponto calculado o valor relativo a cada facto que irá compor a tabela "FactVenda". O significado de cada facto e a forma como se encontra o seu valor é apresentado a seguir:

- **PrecoBruto:** preço bruto do artigo. Quer isto dizer que ainda engloba o valor em impostos;
- **PrecoLiquido:** preço líquido do artigo, ou seja, é preço livre de impostos. Reflete o valor que é efetivamente realizado para a organização;
- **ValorIva:** valor gerado em impostos pela venda de um artigo. Pode obter-se pela multiplicação da taxa de IVA do artigo pelo seu preço bruto;
- **PrecUnit:** Preço (bruto) de cada unidade de artigo;
- **Quantidade:** quantidade de artigos vendidos;
- **PrecoCustoMedio:** preço de custo (bruto) médio do artigo;
- **Lucro:** valor efetivo realizado pela venda do artigo (ou artigos, dependendo da quantidade vendida). Este valor é obtido pela diferença entre o preço líquido realizado e a soma do preço de custo médio, o desconto comercial e a comissão do vendedor;
- **Margem:** percentagem de lucro em relação ao preço líquido do artigo;
- **DescontoComercial:** valor de desconto associado ao artigo presente na venda;
- **Comissão:** valor de comissão obtido pelo vendedor que realizou a venda.

A constituição do modelo dimensional desenvolvido para suportar a solução poderá ser consultada no Anexo 26).

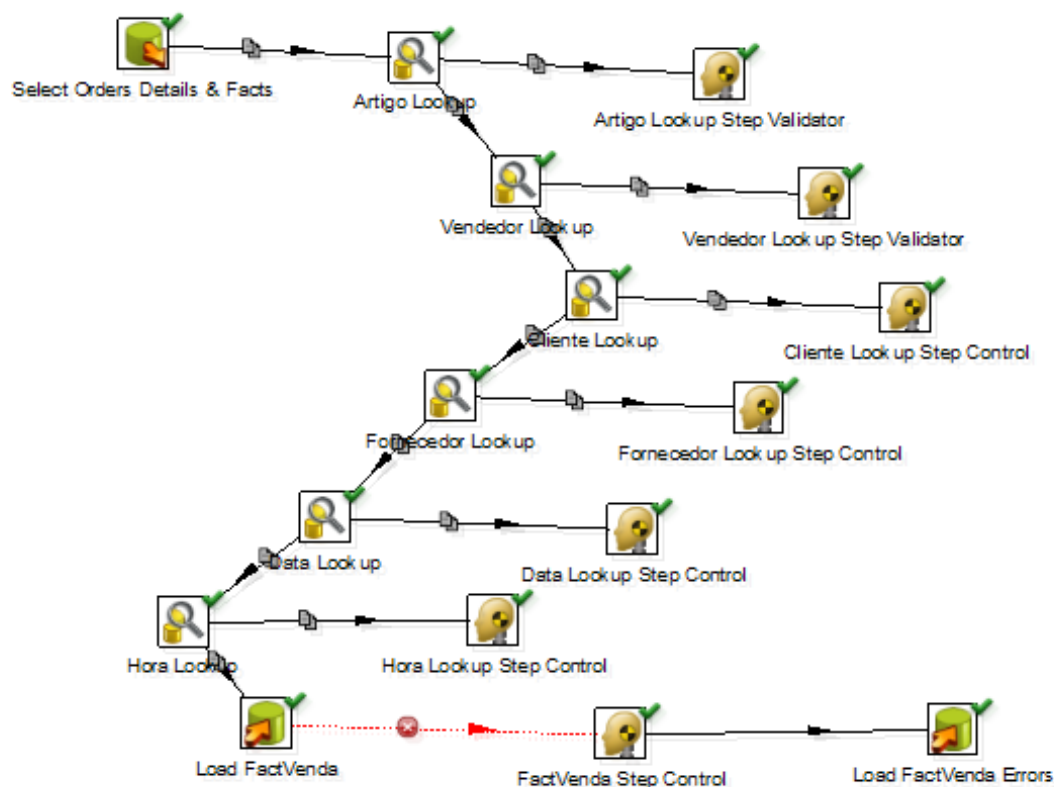


Figura 45: Visão Geral do Processo de Carregamento da Tabela de Factos.

5.2.3 Migração de Dados do Sistema Legado

De modo a garantir a continuidade dos dados até agora produzidos e que servirão de base para a extração do conhecimento associado ao negócio, desenvolveu-se um processo de migração ou importação desses dados do seu formato original (ficheiro *Excel*) para o sistema de dados Primavera. Assim, estes dados estarão também acessíveis ao utilizador via Primavera, com todas as suas funcionalidades associadas. Posteriormente, estes dados irão passar pelo processo de *ETL*, farão também parte do armazém de dados e ficarão disponíveis para consulta no *kibana*. A Figura 46 apresenta o processo de migração desenvolvido.

Começou por desenvolver-se um processo de extração e criação dos clientes e artigos existentes, uma vez que esses não apresentam dependências de outro tipo de dados e serão referenciados nos registos de venda a criar. De seguida extraíram-se, linha a linha, as vendas registadas nos ficheiros existentes, representando assim no novo sistema de dados as vendas efetuadas no sistema legado. Esses registos foram então inseridos nas estruturas criadas pelo sistema Primavera, atualizando os valores totais respetivos a cada encomenda.

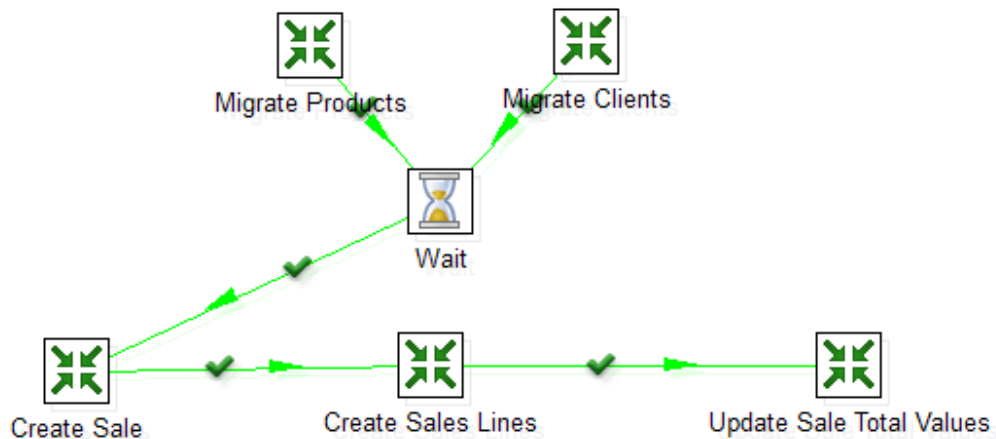


Figura 46: Processo de Migração e dados no *Pentaho*.

Depois de inseridos nestas estruturas os documentos passam a estar todos unificados no mesmo ponto, sendo possível realizar, a partir deste momento, todas as operações definidas nos requisitos da solução, tendo por base um ponto único. Depois de inseridas nestas estruturas, os dados seguirão o processo de *ETL* definido e passarão também a estar disponíveis para análise via *Dashboard* no *Kibana*.

5.3 Ferramenta de Processamento Analítico de Dados

Como ferramenta de apresentação do conhecimento extraído do negócio escolheu-se a ferramenta *kibana*. Além de ser grátis e multi-plataforma, o *kibana* é uma plataforma de análise e visualização de dados, capaz de lidar com grandes volumes de dados e, que disponibiliza uma vasta gama de funcionalidades, entre elas técnicas de *machine learning* para a deteção de anomalias nos dados.

5.3.1 Autorização e Autenticação

O *kibana* é uma ferramenta muito completa principalmente graças à sua integração no *elasticsearch*. Um exemplo é a capacidade de por si só ser capaz de garantir o acesso seguro e controlado aos dados que controla. Além de permitir a integração com sistemas *LDAP*, *Active Directory* e sistemas Public Key Infrastructure (PKI), o *kibana* permite regular o acesso aos dados através da autenticação dos utilizadores permitindo definir ainda *roles* e grupos de utilizadores com permissões semelhantes. Na Figura 47, é criado via *Application Programming Interface (API)*, o utilizador "*jacknich*", com a palavra chave "*t0pS3cr3t*" sendo-lhe atribuída a *role* de "*kibanauser*" (Elastic 2019b).

```
POST /_security/user/jacknich
{
  "password" : "t0pS3cr3t",
  "roles" : [ "kibanauser" ]
}
```

Figura 47: Criação de um utilizador e respetivo acesso no *kibana*.

5.3.2 Ficheiro de Integração

Uma vez garantido o acesso dos devidos utilizadores ao *kibana*, para lhes apresentar o seu *dashboard*, é necessário previamente carregar os dados que serão alvo das consultas.

Para a informação poder ser consultada pelo *kibana* esta tem que ser adicionada e armazenada ao *elasticsearch* ficando associada a um índice. Um índice é identificado por um nome que será usado pelo *elasticsearch* para efetuar operações de indexação, pesquisa, atualização e exclusão aos documentos (dados) que contém. Um documento é a unidade básica de informação indexada pelo *kibana*. Este é expresso em *JavaScript Object Notation (JSON)*. Dentro de cada índice podem ser armazenados quantos documentos forem necessários ((Elastic 2019a)).

Das soluções encontradas destacam-se as seguintes: criação do índice através da *API elasticsearch* ou através do *driver jdbc* desenvolvido para a integração com o *elasticsearch*. Tal como outras funcionalidades já abordadas ao longo deste documento, a criação do índice para alojamento dos dados pode também ser feita através da *API* do *elasticsearch*. Apesar de não se ter optado por este método, este não deixa de ser válido uma vez que é bastante simples de usar e pode ser levado a cabo através de um simples pedido *JavaScript Object Notation (HTTP)*, característica esta que poderá ser muito importante caso no futuro se opte por usar ou desenvolver alguma solução em que se pretenda realizar a operação via Internet.

Além desta, existe a possibilidade de realizar a criação do índice graças à utilização de um *driver jdbc* que efetua o acesso diretamente ao armazém de dados construído. Assim de uma só vez é criado o índice enviando ao mesmo tempo os dados para indexação no *elasticsearch* dando origem aos documentos que serão disponibilizados no *kibana* para a consulta da informação. Este processo acontece graças à execução de um *script batch* que faz uso do *driver* instalado. O ficheiro pode ser consultado em anexo (ver Anexo D).

Deste ficheiro importa destacar as seguintes propriedades:

- *index*: o índice a criar ou atualizar;
- *elasticsearch*: defenição do endereço onde está a ser executado o servidor *elasticsearch*;
- *url*: endereço do servidor de base de dados origem, instância e nome da base de dados;
- *user*: utilizador de base de dados;
- *password*: palavra-chave do utilizador de base de dados;
- *sql*: instrução a ser executada.

O resultado da instrução *Structured Query Language (SQL)* executada dará origem no *elasticsearch* à criação dos documentos correspondentes, documentos esses que serão disponibilizados no *kibana* para que sejam consultados pelos utilizadores da aplicação. Assim, surgiu a necessidade de criar uma nova estrutura de dados que contém as propriedades necessárias a essas pesquisas ou consultas. Foi então criada uma tabela de factos auxiliar que contém os campos de todas as dimensões associadas a cada facto, dando assim origem à tabela "*FactVendaCopy*".

Por último, resta adicionar o índice já existente no *elasticsearch* ao *kibana*. Isso pode ser feito acedendo via *browser* ao *kibana*, escrevendo o nome do índice na zona de pesquisa e uma vez encontrado este será possível de ser adicionado, como mostra a Figura 48.

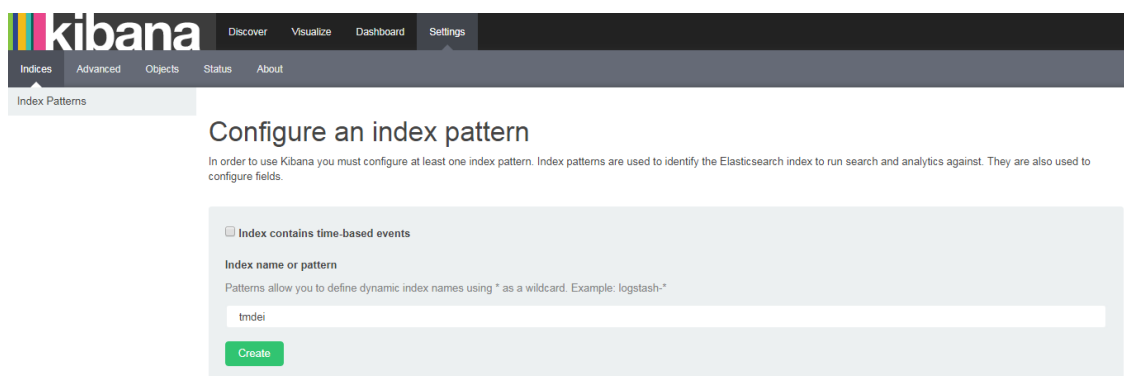


Figura 48: Criação do índice no *kibana*.

5.3.3 Consultas Realizadas

Carregados os dados no armazém de dados e, por sua vez, carregados no *kibana* é agora possível criar as consultas desejadas para uma mais fácil visualização dos indicadores de performance definidos anteriormente. Criou-se, a título de exemplo, um pequeno *dashboard* (Anexo E) para amostra das funcionalidades e mais valias deste tipo de ferramenta. Esta permite a criação de vários tipos de visualização da informação, informação esta que poderá estar agrupada e organizada da forma que mais for conveniente para quem a está a analisar. Neste caso concreto, pela visualização deste *dashboard*, um comercial tem acesso a dados que até agora só tinha disponíveis caso se desse ao trabalho de reunir toda a informação num ficheiro *excel* e construísse ele, com base nisso, as visualizações disponíveis a esse tipo de ficheiro. Cada uma das visualizações que compõe a imagem, é uma visualização ou consulta independente, que pela junção dá origem ao *dashboard*. Assim, estas podem ser consultadas, atualizadas ou eliminadas separadamente. De seguida, apresentam-se algumas das consultas criadas como exemplo e como forma para dar resposta às necessidades identificadas na secção referente aos indicadores de *performance* apontados ao negócio em questão para a sua eficaz análise.

5.3.3.1 Valor Líquido de Artigos Vendidos

A Figura 49 apresenta o total de valor líquido referente à venda de produtos, para o último mês do ano de 2015, ano para o qual existem dados de teste para a solução desenvolvida.

O *kibana* suporta a criação de vários tipos de visualização (consultar Anexo F). Criou-se então a presente visualização, à moda de um contador, para que permita de forma rápida a percepção dos valores gerados no período em análise.

Esta figura apresenta também 4 pontos que importam evidenciar:

- 1): barra de pesquisa que permite criar filtros com os campos e valores desejados. Estes filtros ou expressões seguem uma estrutura criada pelo *elasticsearch* para o efeito.¹⁷;
- 2): a primeira métrica apresenta o total de artigos vendidos. O valor é obtido pela soma da quantidade de todos os artigos vendidos;
- 3): a segunda métrica é o valor total líquido de artigos vendidos. O valor obtem-se pela soma de todos os preços líquidos de artigos vendidos;
- 4): a esta consulta é possível adicionar-se qualquer outra métrica desejada.

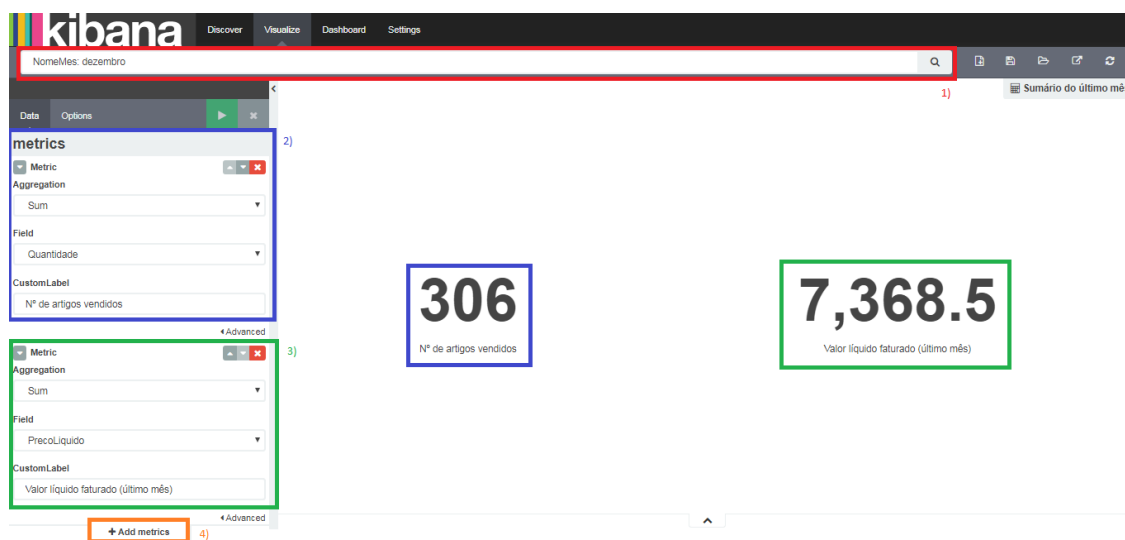


Figura 49: Valor líquido de artigos vendidos.

5.3.3.2 Valor Bruto de Artigos Vendidos

Da mesma forma que na consulta anterior, o valor desta vez bruto de artigos vendido, pode ser conseguido simplesmente pela troca da propriedade que oferece a informação pretendida. Neste caso, apenas é necessário trocar a propriedade "PrecoLiquido" por "PrecoBruto" (e a respetiva legenda), e tem-se a informação desejada (Figura 50), o valor total bruto faturado, neste caso para o mês de dezembro.

¹⁷<https://www.elastic.co/guide/en/elasticsearch/reference/current/query-dsl-query-string-query.html>

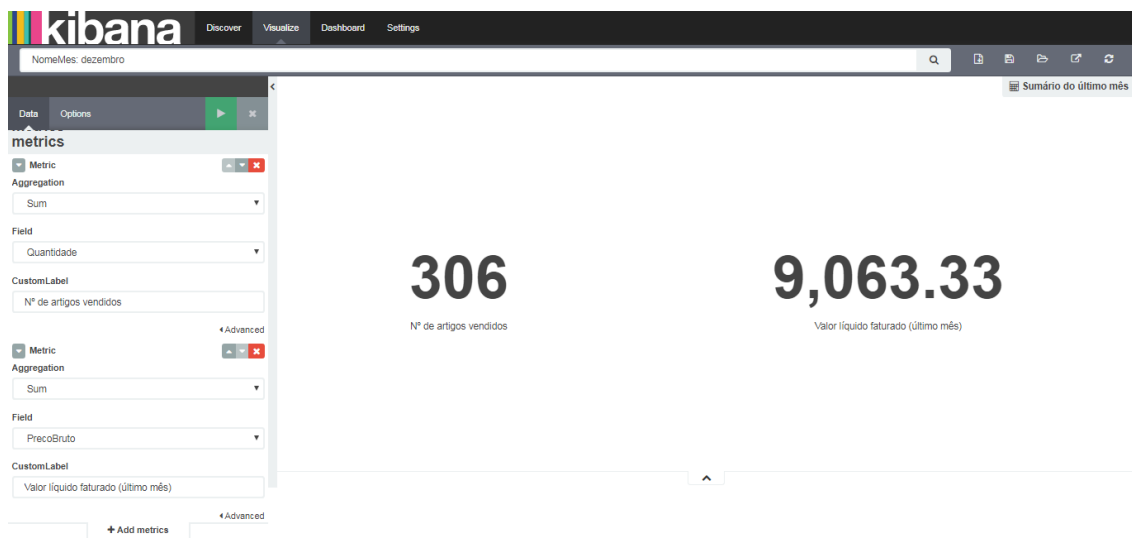


Figura 50: Valor bruto de artigos vendidos.

5.3.3.3 Valor em descontos por artigos vendidos

Na presente consulta pretende-se analisar os artigos vendidos com maior valor de desconto associado a essa venda. Esta informação cruzada com o lucro associado à respetiva venda ou mesmo ao artigo, poderá permitir analisar se apesar de estar a ser vendido com desconto a venda continua a ser lucrativa para a organização, caso contrário, se a situação se tornar um prejuízo por vários períodos de tempo poderá significar que o artigo não representa interesse para o negócio. Assim, com base nestas informações poderá decidir-se se faz sentido ou não o artigo continuar disponível para venda. A Figura 51 apresenta um gráfico de barras que permite a visualização da informação relativa ao *top 5* dos artigos vendidos com maior valor médio de desconto. É possível ver também uma *tooltip*, gerada automaticamente pelo *kibana* permitindo ver em concreto as informações que compõe o gráfico gerado.

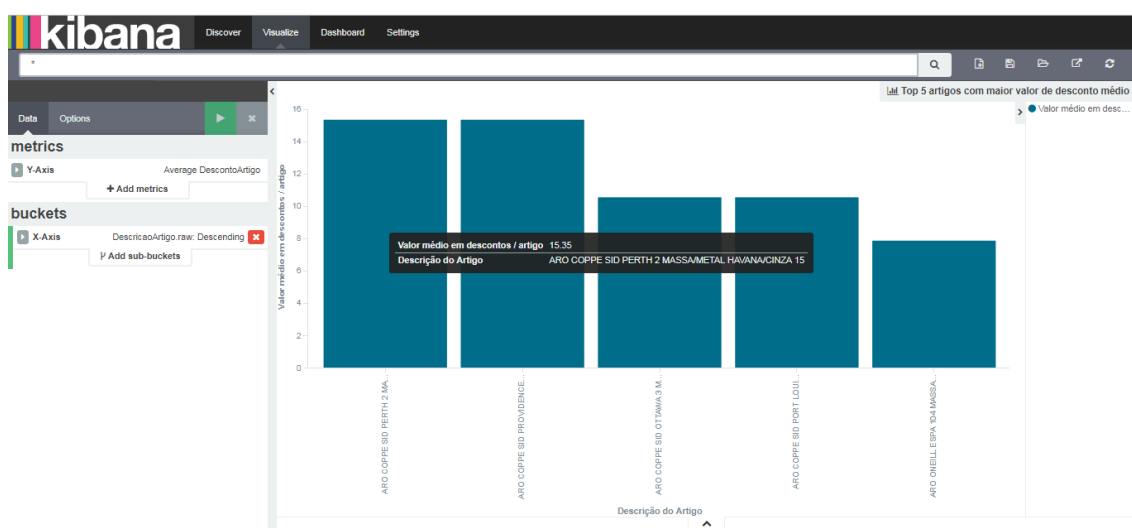


Figura 51: Valor em descontos por artigos vendidos.

5.3.3.4 Valor em impostos por artigos vendidos

Na Figura 52, pode analisar-se o valor gerado em impostos, IVA sobre os artigos vendidos ao longo de cada mês do ano. Uma vez que, dependendo dos volumes de negócio da organização, esta pode gerar valores bastante elevados, parte desse valor tem que ser tributado, o que obriga as organizações a ter em conta os seus fundos disponíveis de forma a não entrarem em situações de incumprimento. Além disso, conseguindo separar o valor em impostos e o valor líquido das suas vendas terão uma melhor noção sobre a atividade efetiva da organização.

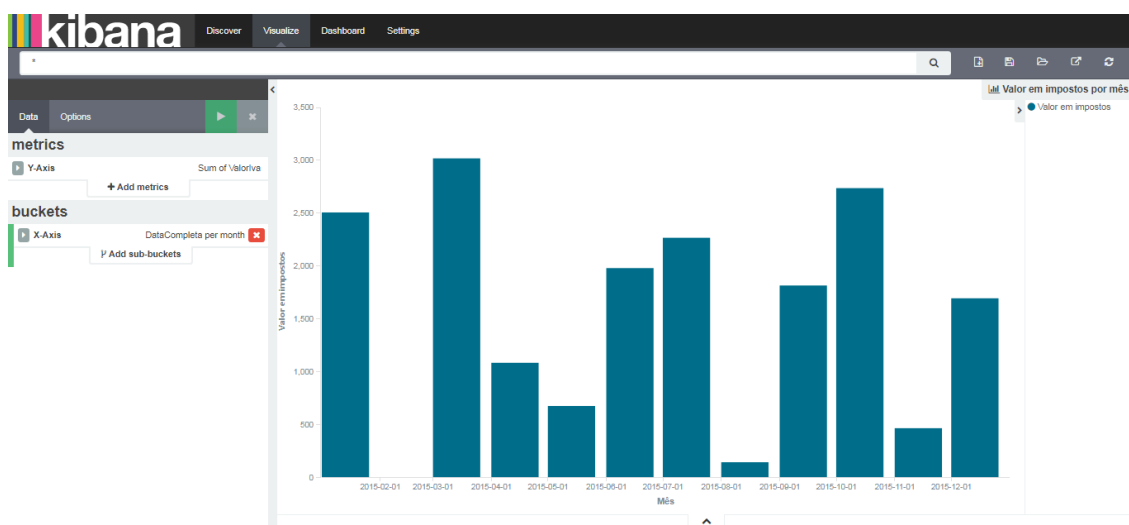


Figura 52: Valor em impostos por artigos vendidos.

5.3.3.5 Número de artigos vendidos

Na Figura 53 apresenta-se graficamente o número de artigos vendidos por mês. Mais uma vez, tal como na consulta realizada na Secção 5.3.3.3, esta informação poderá ser bastante útil ao nível do armazenamento de artigos podendo esta contribuir também para uma gestão mais eficaz dos *stocks*. Poderá também ser muito importante analisar estes dados quando além do número de artigos vendidos sejam aplicados filtros, por exemplo, relacionados com a marca, modelo, estação do ano, cliente ou distrito onde se realizou a venda. Este tipo de configurações permitirá perceber o padrão de venda que levarão a uma gestão dos artigos em comercialização pela organização, evitando a acumulação de artigos sazonalmente menos vendidos, ou mesmo perceber com antecedência o período de decadência de cada artigo e com isso evitar a perda dos seus recursos.

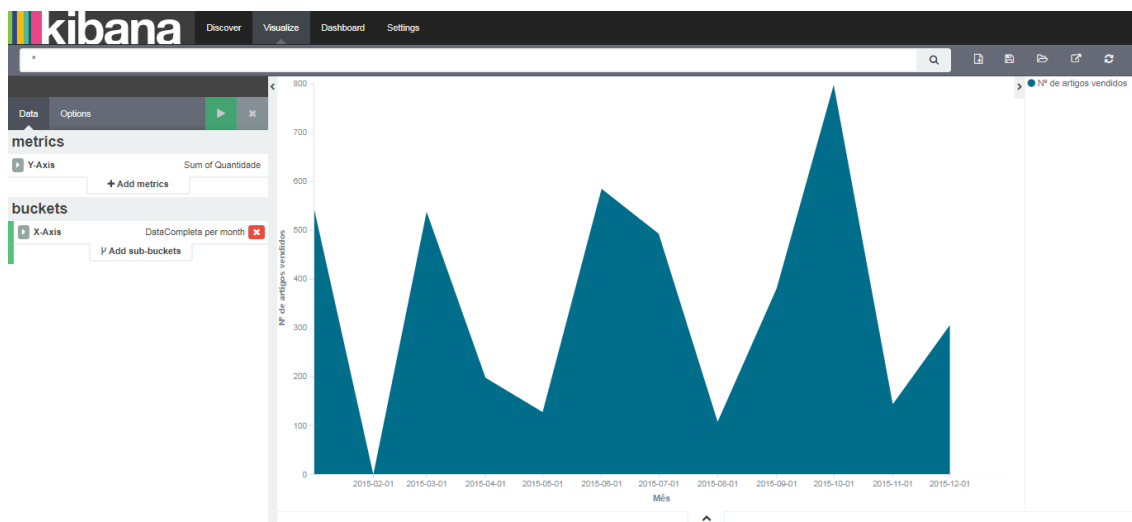


Figura 53: Número de artigos vendidos.

5.3.3.6 Outras consultas

Muitas outras consultas e análises podem ser feitas pelo uso do *kibana*, seja pela criação a partir das anteriormente apresentadas, seja pela criação de análises baseadas nos novos tipos de informação oferecidos pela adoção da solução desenvolvida. Por exemplo, se juntamente com a morada de cada cliente se passar a guardar também as coordenadas *GPS* correspondentes será possível criar de forma gráfica e facilmente perceptível, um mapa baseado no número de vendas de artigos por localização geográfica, como é visível na Figura 54. Com base neste tipo de consulta poderá perceber-se, por exemplo, que clientes poderão ser visitados por dia tendo por base a sua distância, e perceber se essa zona geográfica de venda tem influência no tipo de artigos vendidos, além da zona onde são vendidos mais artigos. Com base nestas informações poderão atualizar-se métodos de negócio, realizar-se campanhas mais eficazes ou até perceber que tipos de artigo e em que quantidade deverão estar em *stock*. Este tipo de análises poderão fornecer às pessoas de uma organização informações essenciais para a gestão e o planeamento e com isso aumentar as hipóteses de sucesso da organização.

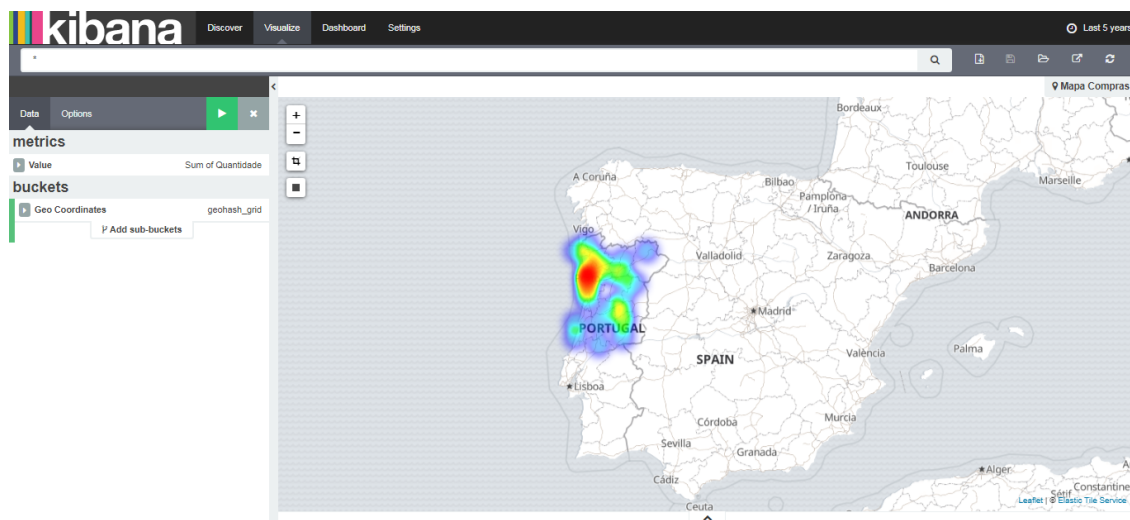


Figura 54: Mapa de Vendas de Artigos.

Como referido anteriormente, criou-se um *Dashboard* exemplo (Figura 55), pensado do ponto de vista de um comercial, que pretende perceber rapidamente como está, de um ponto de vista geral, a sua atividade recente.

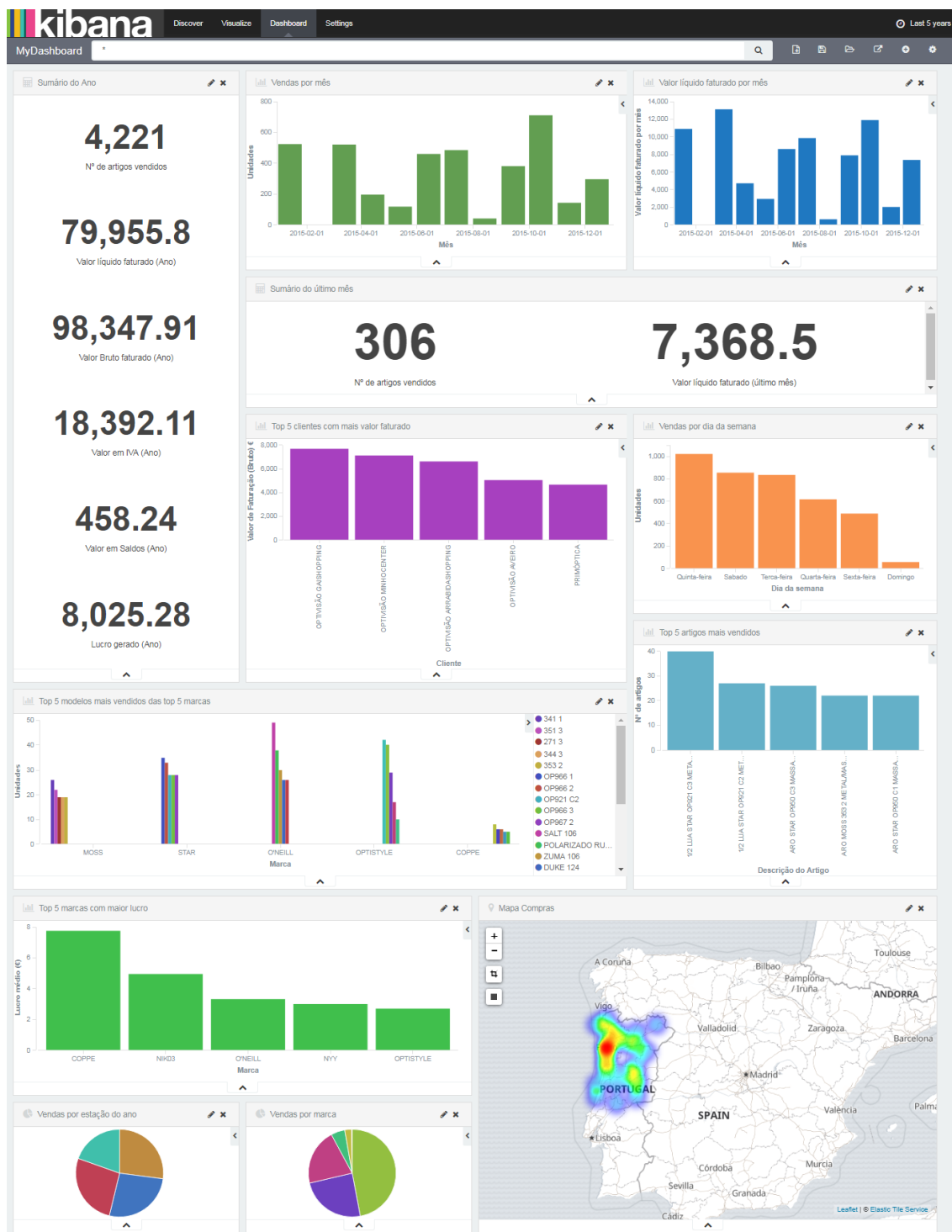


Figura 55: Dashboard Exemplo.

Assim, como mostra a Figura 55, de forma rápida e sem esforço, um comercial poderá perceber quais são os artigos mais vendidos, onde esta a vender mais artigos, perceber destes quais são os que lhes dão mais lucro, quem são os seus melhores compradores entre outras informações.

6. Avaliação de Resultados

Neste capítulo é apresentada a forma como se realizou a avaliação do projeto atual tendo em conta as diferentes grandezas identificadas. Assim, o capítulo começa por identificar as grandezas em avaliação e o que se pretendia avaliar em cada uma delas. De seguida apresenta-se a metodologia de avaliação para cada uma das grandezas e, por fim, descrevem-se as experiências realizadas e os resultados obtidos.

6.1 Grandezas em Avaliação

Tendo em conta o presente projeto e o conjunto de funcionalidades que a solução propõe, as grandezas a considerar prendem-se com a exatidão dos cálculos realizados para as análises pretendidas aos dados e a avaliação dos utilizadores quanto à satisfação para com a solução.

A exatidão de valores, representa a necessidade de apresentar aos utilizadores valores fiáveis, no sentido de possibilitar uma análise correta dos indicadores de desempenho da organização e da sua evolução ao longo do tempo.

Por outro lado, tem também que se ter em conta a satisfação do utilizador, uma vez que a solução propõe agilizar os processos de informatização de dados, bem como a sua posterior análise. Ao garantir a satisfação dos utilizadores, poderá estar-se a contribuir para a minimização do esforço necessário por parte destes, à adoção da nova solução, com isso fazer com que o tempo até ficarem confortáveis com as novas dinâmicas diminua e a organização possa mais rapidamente começar a notar os benefícios da sua adoção.

6.2 Hipóteses a Avaliar

Tendo em conta as grandezas identificadas anteriormente, as hipóteses a testar estão relacionadas com os cálculos a realizar e com a satisfação do utilizador.

Quanto à correção dos dados, estes são validados com testes unitários garantindo a sua validade, de modo que não é testada nenhuma hipótese para o caso. Ainda relativamente aos dados, pretende-se também que a solução apresente fluidez e performance, por isso interessa avaliar essa característica em tempo. Assim pretende-se testar as seguintes hipóteses:

- o tempo médio para a geração de análises é inferior a dois segundos;
- o tempo médio para a geração de um documento é inferior a dois segundos.

Ao nível da satisfação do utilizador, é importante garantir a fácil adoção do sistema por parte dos mesmos, para isso é necessário que estes o considerem adequado às funções que necessitam de desempenhar. Como tal, define-se a seguinte hipótese:

- inquérito de satisfação obter pelo menos quatro numa escala de zero a cinco.

6.3 Indicadores e Fontes de Informação

Tendo em conta as hipóteses definidas, são utilizados diferentes indicadores e fontes de informação para levar a cabo a sua devida avaliação. Sendo esta uma solução que irá manusear dados de carácter financeiro, pretende-se a correta utilização dos mesmos de forma a que estes não contenham erros. Assim, um indicador muito importante é a exatidão dos valores produzidos pela solução.

Uma vez que se pretende garantir a apresentação da informação em tempo útil, para a sua visualização e conseqüente utilização, define-se como indicador o tempo em segundos desde o momento em que o utilizador solicita ao sistema a informação até ao momento em que a mesma fica disponível ao mesmo. Para isso, conta-se com o registo do tempo médio, de uma ação de um utilizador, desde o instante em que o utilizador pede ao sistema a informação até que esta é apresentada ao utilizador, garantindo assim o desempenho da solução. Deve tentar-se executar esta ação em diferentes contextos na tentativa de abranger situações com grandes volumes de informação de modo a que a média se aproxime ao máximo da utilização real do sistema.

Outro indicador da efetividade e eficácia da solução desenvolvida é a satisfação dos seus utilizadores. Pretende-se para isso recolher *feedback* por parte dos utilizadores da solução, *feedback* este que deverá ser captado através do preenchimento de inquéritos de satisfação. Garantida a exatidão dos resultados e a sua produção em tempo útil, é também importante que os seus utilizadores se sintam confortáveis com a solução desenvolvida de modo a que esta seja também por si só um contributo para a sua produtividade. Com estes inquéritos, será possível garantir: a usabilidade, ao avaliar a interação do utilizador com a aplicação; a adaptabilidade, percebendo a facilidade de adaptação do utilizador às funcionalidades da aplicação; a utilidade, ao avaliar se o sistema é útil para a finalidade pretendida e se é eficaz no seu desempenho; e a satisfação ao verificar o grau de contentamento dos mesmos para cada funcionalidade implementada.

6.4 Metodologia de Avaliação

O sistema em geral foi testado utilizando testes quanto à exatidão dos valores gerados, e de integração, no sentido de verificar se cada uma das funcionalidades está a ser corretamente implementada. No caso da avaliação da performance do sistema, pretende-se o uso de grupos selecionados de dados, onde se pretende extrair uma generalização da performance do sistema tendo em conta aquilo que se esperam ser os dados que serão tratados pelo sistema.

Para avaliar a satisfação dos utilizadores quanto à utilização geral do sistema, deverão ser apresentados inquéritos de satisfação a um grupo de utilizadores no sentido de obter uma avaliação das diferentes funcionalidades, utilizando para tal um sistema de classificação para

cada questão do inquérito. Posteriormente deverão ser realizados os devidos cálculos para verificar a classificação da solução. No entanto, nesta fase do projeto não foi possível implementar esta metodologia, questão esta que será abordada ainda na Secção 6.5.

Assim, daquilo que foi anteriormente explicado, conclui-se que as metodologias a aplicar, em particular para a grandeza de satisfação dos utilizadores, são os grupos de controlo/teste, uma vez que um grupo restrito de pessoas deverá realizar as ações definidas para possibilitar a resposta aos inquéritos. No caso dos testes à performance da solução usaram-se métodos de *crossvalidation* com o intuito de garantir a performance atual e futura da mesma com base na tentativa de predição dos dados. Deste modo, ao utilizar testes a conjuntos de dados exemplo, pretendem-se extrair ilações relativas à performance que o sistema terá quando estiver a funcionar com o volume dados completo.

6.5 Experiências e Resultados Obtidos

Apesar de se ter definido a metodologia apresentada anteriormente (Secção 6.4), nem todos os aspetos definidos foram efetivamente executados.

Ainda que se possa validar e dizer com grande certeza que a solução responde e soluciona os problemas que lhe deram origem, a solução não pode ser considerada como acabada, encontrando-se nesta fase num processo de validação. Neste momento não foi possível ainda apresentar a solução a todos os seus utilizadores, pelo que não se justifica realizar a análise à satisfação dos mesmos quanto à solução desenvolvida pois estes não apresentariam significância estatística que o justificasse.

Outra característica que não se pôde garantir foi a geração de documentação no intervalo de tempo definido. Uma vez que na solução desenvolvida foi adotada a ferramenta Primavera, responsável pela funcionalidade em análise, não se conseguiu determinar com certeza que, qualquer documento, leve menos de dois segundos a ser gerado.

No entanto, seguiu-se a metodologia definida para garantir a exatidão dos dados e a performance da solução quanto à apresentação das análises desenvolvidas.

6.5.1 Testes de Exatidão

Pela adoção do sistema Primavera, os dados criados pela ferramenta estão automaticamente validados uma vez que este apresenta as suas próprias regras e validações. No então, contemplou-se no desenvolvimento da solução a hipótese de realizar uma migração dos dados previamente existentes em formato *excel* para as estruturas do Primavera. Por sua vez, os dados presentes nestas estruturas passam pelo processo de *ETL* definido e são também armazenados nas estruturas de *Staging* e *Datamart*. Assim, sentiu-se a necessidade de validar estes dados, desde a sua inserção nas estruturas Primavera até à visualização na ferramenta de análise aos dados. Para isso utilizaram-se alguns ficheiros de venda existentes, sobre os quais incidiu o processo de *ETL* desenvolvido. Depois de ter sido executado o processo, testou-se, utilizando a aplicação adotada para a visualização dos dados, se o resultado era coincidente. Para validar a forma como se implementou o processo de migração dos dados em formato original para as estruturas do Primavera, executou-se um comando em *SQL* com o objetivo de calcular o número de artigos vendidos, o preço líquido e bruto obtidos, assim como o valor em IVA. Na Figura 56 apresenta-se o comando desenvolvido,

neste caso para uma venda realizada no dia dois de novembro de dois mil e cinco ao cliente "OPTIVISÃO GAIA JARDIM".

SQLQuery70.sql - (...master (SA (53)))*

```

1 SELECT SUM(linhas.Quantidade) as 'Artigos Vendidos',
2       SUM(linhas.PrecoLiquido) as 'Preço Líquido',
3       SUM(linhas.TotalIliquido) as 'Preço Bruto',
4       SUM(linhas.TotalIva) as 'Total IVA'
5 FROM [PRIEMPRESA1].[dbo].[CabecDoc] cabec
6      JOIN [PRIEMPRESA1].[dbo].[Clientes] clientes
7          ON clientes.NOME = 'OPTIVISÃO GAIA JARDIM'
8          AND cabec.Entidade = clientes.Cliente
9          AND cabec.Data = '2015-11-02'
10      JOIN [PRIEMPRESA1].[dbo].[LinhasDoc] linhas
11          ON linhas.IdCabecDoc = cabec.Id

```

Results | Messages

	Artigos Vendidos	Preço Líquido	Preço Bruto	Total IVA
1	49	661,9	814,16	152,26

Figura 56: Comando SQL Desenvolvido Para Teste.

De igual forma, tentou validar-se os mesmos valores para a mesma venda mas desta vez na ferramenta utilizada para a realização das análises aos dados.

kibana | Discover | Visualize | Dashboard | Settings

DataCompleta: 2015-11-02 AND NomeCliente: "OPTIVISÃO GAIA JARDIM"

Artigos Vendidos	Preço Líquido	Preço Bruto	Total em IVA
49	661.9	814.16	152.26

metrics

- Metric: Sum of Quantidade
- Metric: Sum of PrecoLiquido
- Metric: Sum of PrecoBruto
- Metric: Sum of valorIva

buckets

Export: Raw | Formatted

Figura 57: Análise à Venda em Teste na Ferramenta Kibana.

Pela análise da Figura 56 e 57, nota-se que os respetivos valores são coincidentes, valores estes que também coincidem com os valores existentes no registo de venda em formato original. Esse registo encontra-se no Anexo H. Este processo foi repetido para vários clientes, em diferentes datas, tendo-se registado em todos os casos o sucesso do mesmo.

6.5.2 Testes de Desempenho

Uma das características da solução que se pretendia validar era a sua rapidez quanto à geração de documentos por parte da solução de apoio à gestão utilizada. A solução adotada, o Primavera, não disponibiliza nenhuma forma de cronometragem quanto a este tempo, nem é garantido nenhum limite temporal associado a esta funcionalidade. Assim, não foi possível garantir a hipótese de o tempo médio de geração de documentos ser inferior a dois segundos. No entanto, no decorrer do projeto e da utilização da ferramenta não foi detetado nenhum problema associado a esta hipótese.

A outra característica de interesse era a rapidez da solução em apresentar graficamente os dados ao utilizador. Como já se tinha descrito na Secção 3.6.3, relativa ao Estado da Arte da ferramenta *Kibana*, a ferramenta adotada oferece características de *debug* muito interessantes e úteis, sendo uma delas a informação relativa ao tempo que o pedido para a geração de uma análise demora desde a sua requisição até ser apresentada no ecrã. Foi com base nesta informação que se realizou o teste a esta característica. Para isto, além do índice que suporta as análises ao armazém de dados criado para a solução, criaram-se dois outros: um índice composto com dados apenas sobre o mês de Agosto (mês com menos registos); e um índice composto com dados apenas sobre o mês de Outubro (mês com mais registos). De seguida, registou-se o tempo necessário para gerar a análise apresentada na Figura 54, o mapa de vendas, vinte vezes para cada índice (o tempo necessário para gerar a análise é dado no *kibana* por *Request Time*).

Para se tentar tirar alguma conclusão em relação aos dados utilizaram-se os três índices apresentados anteriormente, ambos com diferentes quantidades de registos, para tentar perceber a influência da ordem de grandeza dos dados e o seu impacto no tempo necessário à geração da análise gráfica.

No Anexo I apresentam-se alguns exemplos da informação, disponibilizada pelo *Kibana*, utilizados nesta análise de performance.

No fim das vinte experiências a cada índice os resultados foram os seguintes:

Tabela 11: Tabela de Tempos Médios Para Geração de Análises Gráficas

Índice	Nº de registos	Tempo médio (em milissegundos)
Agosto	108	52
Outubro	798	65
Total	4221	75

Como mostram os resultados presentes na Tabela 11, em todos os índices foi registado um tempo médio de geração gráfica inferior a dois segundos (ou dois mil milissegundos). Assim a solução apresentada valida claramente a hipótese em teste. No entanto, o volume de dados atualmente presente na solução não é ainda significativo, pelo que se terão de realizar novamente os testes aqui descritos, assim que se incluam na solução dados relativos ao histórico dos restantes anos de atividade da organização e os dados referentes aos restantes vendedores da mesma.

7. Conclusões

Num mundo cada vez mais exigente e competitivo, as organizações para serem bem sucedidas são obrigadas a estar constantemente à procura de formas para se manterem atualizadas e de otimizarem os seus processos de modo a garantir a maior produtividade possível.

Com os avanços tecnológicos, aparecem cada vez mais ferramentas que se revelam essenciais aperfeiçoar e melhorar os processos de negócio das empresas tornando-se essenciais no sucesso das mesmas. É neste contexto que surge a solução apresentada nesta dissertação.

O sistema desenvolvido apresenta-se como uma solução para a informatização das ações quotidianas de uma empresa que atua no ramo da venda de produtos a retalho. Este, tem como ponto de partida uma aplicação que permite o registo e a gestão das ações que levam à venda destes produtos, incluindo a geração da documentação relativa a essas mesmas vendas, gestão dos *stocks* dos produtos em comercialização, e ainda a configuração e a gestão dos utilizadores que terão acesso à aplicação. Ao adotar esta aplicação, a organização terá ao seu dispor um mecanismo simples, rápido e de fácil utilização que deverá aumentar a fluidez e a produtividade dos seus processos de venda.

Após a contínua informatização das ações quotidianas da organização, a solução proposta disponibiliza uma ferramenta de análise a essa mesma informação. Deste modo, de forma rápida e interativa, será possível extrair conhecimento da informação guardada, nomeadamente sobre o negócio onde se insere a organização, sobre os seus clientes e os seus hábitos, entre outros aspetos do negócio. Esta ferramenta e o conhecimento que irá produzir poderá tornar-se fundamental para o bom desempenho da organização, uma vez que lhe permitirá ajustar os seus processos de negócio de acordo com as necessidades, visando o aumento da produtividade da organização.

7.1 Objetivos atingidos

Na presente secção, serão apresentados os objetivos definidos para a solução desenvolvida e o grau com que estes satisfazem as funções pretendidas. A Tabela 12, servirá para apresentar esses objetivos e o seu grau de tangibilidade.

Tabela 12: Objetivos definidos e o seu grau de tangibilidade.

Objetivo	Grau de Tangibilidade
Criação de Encomendas	Atingido
Validação de Encomendas	Atingido
Criação de Alertas/Notificações	Parcialmente Atingido
Geração de Documentação	Parcialmente Atingido
Criação do <i>Dashboard</i> para Análise de Vendas	Atingido
Possibilidade de Análises Tendo em Conta Histórico	Atingido
Gestão de <i>Stocks</i>	Atingido
Gestão de Artigos	Atingido
Gestão de Utilizadores	Atingido
Gestão de <i>Templates</i> de Documentos	Atingido

O problema que deu origem a esta dissertação foi a forma pouco eficiente como um comercial regista a sua atividade quotidiana, a venda de produtos óticos. Assim, o primeiro objetivo a ser atingido foi a disponibilização de uma solução capaz de oferecer uma forma simples e mais eficaz de registar uma encomenda. Conseguiu-se também que a solução produzida permitisse a este tipo de interveniente do negócio a análise dessas mesmas encomendas, e respetivas vendas. Agora, um comercial tem ao seu dispor um *dashboard* com uma série de informações que lhe permitirá ter uma rápida visão sobre o estado do negócio e das suas vendas. Poderá também a qualquer momento criar novas visualizações que lhe permitirão obter as informações de que necessita. Graças à construção de um Armazém de Dados, conseguiu-se também criar um processo que lhe permitirá realizar análises tendo em conta o histórico dos acontecimentos que ocorrem ao longo do tempo e que influenciam o negócio.

Apesar da solução desenvolvida permitir a geração de documentação e a criação de alertas (ou notificações) sobre as vendas efetuadas, estes não se consideram como objetivos totalmente atingidos. Quanto à geração de documentação, apesar da aplicação que serve como apoio à gestão de encomendas permitir sem restrições a geração de documentação relativa à faturação da sua atividade, a solução desenvolvida não oferece, de forma ilimitada a geração de documentação de sumário dessas mesmas encomendas, considerando-se assim o objetivo como parcialmente atingido. De igual forma, apesar de se garantir ser possível a criação de alertas sobre as vendas registadas, não foi possível a sua implementação efetiva, não se considerando este requisito como totalmente atingido. Estas restrições serão de novo abordadas na secção relativa às limitações da solução (Secção 7.2).

Além dos já objetivos abordados, foi ainda possível garantir aos utilizadores da solução a validação das encomendas criadas, a gestão de *stocks* de artigos, assim como a gestão efetiva dos artigos abrangidos pelo negócio, a gestão dos *templates* de documentos de faturação e a gestão de utilizadores e as suas respetivas permissões no sistema.

7.2 Limitações da Solução

Apesar da solução desenvolvida responder com sucesso praticamente a todos os requisitos definidos para o projeto, este, apresenta ainda assim algumas limitações, seja pelas decisões

tomadas ao longo do projeto ou pelas restrições das tecnologias e/ou ferramentas adotadas.

Uma vez que se decidiu adotar a ferramenta Primavera sendo esta uma solução paga, utilizou-se uma versão grátis com algumas limitações inerentes: a utilização está restringida a apenas um posto de trabalho, e com um teto de trinta mil euros quanto ao valor de faturação. Uma restrição que surge da utilização do *software* Primavera. Esta ferramenta, apesar de responder favoravelmente aos requisitos necessários à sua adoção, apenas pode ser utilizada quando inserida em ambientes cujo sistema operativo é o *Windows*. Uma vez que se optou por fazer uso da solução Primavera, sendo esta uma ferramenta que assenta numa base de dados *SQL Server*, todo o processo de *ETL* foi produzido tendo por base esta especificação.

Como já abordado na secção anterior relativa aos objetivos definidos para a solução, esta apresenta uma limitação quanto à criação de alertas ou notificações sobre as vendas registadas. Apesar de se garantir a possibilidade da ferramenta Primavera oferecer esta funcionalidade, a mesma não foi nesta fase implementada. No entanto, uma vez que a funcionalidade está assegurada pela ferramenta adotada, ainda que não esteja disponível atualmente na versão utilizada, deverá ser adicionada à solução no futuro pela adoção da ferramenta na versão sem restrições. Outro obstáculo encontrado foi a geração de documentação de resumo das vendas efetuadas. A ferramenta de processamento analítico de dados adotada para a análise de vendas, o *kibana*, oferece de forma integrada um *plugin* que permite a geração de documentos, documentos esses que se pretendia que tivesse a informação sumária relativa às vendas efetuadas. Este *plugin* é pago, pelo que não é possível, de forma gratuita, fazer uso desta funcionalidade, o que se revela uma limitação no uso da ferramenta.

Um requisito definido foi a capacidade da solução desenvolvida estar preparada para um uso que não seja apenas a utilização em português. Assim, teve-se em atenção a capacidade de todos os componentes usados manterem esta característica. No entanto, no caso da ferramenta de apresentação e de análise de vendas, este requisito apesar de se verificar não foi plenamente implementado uma vez que a solução apresenta a linguagem apenas na língua inglesa.

A solução desenvolvida tem de ser considerada como um protótipo, estando ainda numa fase de testes e validação. Apesar das limitações detetadas, estas não se consideram impeditivas do correto funcionamento da solução, uma vez que possibilita todos os requisitos propostos. Ainda assim, irão discutir-se estas limitações e as suas potenciais alternativas na secção seguinte.

7.3 Análise de Alternativas e Trabalho Futuro

A presente secção servirá para abordar as limitações apresentadas anteriormente no sentido de apresentar soluções que tentem minimizar e eliminar os problemas detetados.

A solução desenvolvida apresenta-se como um protótipo para o problema apresentado e desta forma, para ser adotada pela organização terá ainda de sofrer alguns ajustes. Um deles seria a subscrição de uma assinatura da versão do *software* Primavera. Esta ferramenta acarreta custos no caso da sua adoção, no entanto graças a esta subscrição, a organização ficaria com uma solução totalmente garantida para as suas necessidades como já se provou no decorrer deste documento. Graças a este vínculo, a solução deverá permitir à organização usar na plenitude todas as suas funções, para todos os seus utilizadores, sendo-lhes garantida a

manutenção da ferramenta, o acesso às mais recentes correções e desenvolvimentos, e desta forma oferecer à organização tudo o que esta necessita para a gestão das suas atividades.

Outra limitação apontada a esta ferramenta é o facto de esta apenas estar preparada para ser alojada em ambiente *Windows*. Este facto não será um problema uma vez que a organização já faz uso de este tipo de sistemas. No entanto, e uma vez que também era um objetivo inicial produzir uma solução genérica, este pormenor poderá ser limitador noutras situações. Das ferramentas adotadas para a solução apresentada, esta é a única que apresenta esta limitação, pelo que quer o *Pentaho*, quer o *Kibana* podem ser utilizadas em ambientes *Windows*, *Linux* ou *Mac*. No caso em que a solução tenha que ser alojada num ambiente que não seja *Windows* uma das seguintes opções terá de ser tomada: adoptar uma solução já existente para esse tipo de ambiente ou desenvolver uma solução à medida. Graças à forma como se desenvolveu a solução, definindo concretamente a interface entre as diferentes ferramentas que compõem a mesma, pode dizer-se que se essas interfaces forem respeitadas a solução continuará a funcionar corretamente. Assim, no caso da substituição da ferramenta *Primavera* por outra que funcione no sistema operativo desejado fica ultrapassada a limitação detetada.

Graças ao avanço tecnológico dos dias em que vivemos, é muito fácil obter uma solução de faturação, peça esta (quase) obrigatória para uma organização. No caso em estudo, esta ferramenta de faturação e de apoio à gestão ainda não existia e como tal, gerou-se em torno da solução adotada todo o processo de criação do armazém de dados que dará à organização mais uma ferramenta para aumentar a sua produtividade e o lucro que daí irá obter. No entanto, não se espera que, uma organização que pretenda criar uma solução de análise de vendas, tenha que mudar a forma como regista as suas vendas e os seus processos de negócio. Portanto, equaciona-se a hipótese de oferecer a potenciais novos clientes formas para que estes possam, mantendo os seus atuais processos de negócio, aproveitar as vantagens que advêm da utilização do armazém de dados criado na atual solução. A primeira abordagem seria disponibilizar uma *API* que teria de ser implementada de acordo com a estrutura de dados suportada na criação do armazém de dados. Esta poderia ser uma *API REST* disponibilizada ao cliente, onde com base nos campos existentes, este seria responsável por enviar os dados que espelham as mesmas características no sistema origem. Outra abordagem poderia ser o desenvolvimento de um processo de *ETL* semelhante ao atualmente utilizado para a criação do armazém de dados, mas que neste caso iria servir para transferir os dados do sistema origem para a base de dados de *staging area* do atual sistema. Depois de inseridos nesta área o processo de criação do AD seguiria os passos atuais.

Outra limitação apresentada prende-se com o facto do *plugin* utilizado pelo *Kibana* para a impressão de documentos ser pago, no entanto além da solução que passa pelo pagamento do *plugin* pode, recorrendo-se à implementação, desenvolver uma pequena aplicação que irá consumir a *API* igualmente usada por este *plugin* para a concepção do conteúdo dos documentos pretendidos, e gerar de igual modo os ficheiros desejados. Outro aspeto menos positivo detetado foi a impossibilidade de trocar, pelo utilizador, a linguagem com que a ferramenta se apresenta, no entanto este facto pode ser resolvido pela criação das traduções pretendidas, traduções essas que substituirão as originais fazendo o efeito desejado.

Apesar de conter limitações, a adoção da ferramenta *kibana* considera-se uma grande mais-valia, seja pela facilidade de configuração e de utilização, seja pela imensidade de funcionalidades que possibilita em conjunto com o motor *elasticsearch*. Entre elas pode destacar-se

a possibilidade de introduzir técnicas de *machine learning* à solução. Sendo esta uma solução de análise de vendas, onde se espera que o seu armazenamento cresça rapidamente mantendo-se a capacidade de analisar facilmente a atividade existente, a hipótese de descobrir eficazmente padrões de comportamento nessas mesmas vendas e as tendências para o futuro torna-se uma funcionalidade que se perspectiva muito aliciante neste âmbito. O *Kibana* oferece este tipo de funcionalidades. A adoção de técnicas deste género, para a análise de vendas, poderá significar uma forma muito eficaz de gerar lucro para uma organização, e uma solução que ofereça estas capacidades poderá ter bastante valor, valor este que se pretende adicionar no futuro à solução existente.

Devido ao facto do projeto estar ainda numa fase inicial, de testes e validação dos processos desenvolvidos, até ao momento, o armazém de dados é ainda constituído apenas por um *Datamart* que se destina à análise das vendas da organização. No entanto, no futuro, caso se validem os desenvolvimentos e o processo implementado poderão ser criados novos *Datamarts* relativos a outros aspetos de negócio que poderão ser muito importantes para a organização, nomeadamente, a possibilidade de criar um *Datamart* relativo à compra de artigos (matérias primas). Desta forma, além de poder analisar-se a forma como se estão a comprar os artigos e/ou matérias primas, e desta forma além de retirar conhecimento da melhor forma para aumentar o lucro em artigos vendidos poderá também reduzir-se o valor gasto na produção dos artigos em venda.

7.4 **Apreciação final**

O trabalho realizado no decorrer desta dissertação permitiu desenvolver uma solução que permite a uma organização, de um modo geral, informatizar todo o seu processo de venda de produtos. Desde a entrada em *stock*, até à sua venda, a solução permite aos seus intervenientes de uma forma simples e rápida registar sem erros o processo de encomenda e posterior venda dos produtos que comercializa.

Depois do processo de venda, a solução permite ainda a análise dessas mesmas vendas. A informática e os processos que utiliza fazem com que trabalhos que no passado eram demorados e trabalhosos possam, com recurso a ferramentas informáticas, ser rápidos e muito mais eficazes. Estes atributos fazem parte da solução desenvolvida. Graças à disponibilização da ferramenta de análise de dados implementada, uma organização deverá ser capaz de, pela análise dos dados estruturados e apresentados, descobrir padrões de comportamento nas vendas registadas e com isso, tomar ações quer de *marketing* quer estratégicas, que lhe permita otimizar os seus processos de negócio.

Num mundo com cada vez mais competitividade e, conseqüentemente, com menores margens de lucro, esta solução possibilita a uma organização tomar medidas que poderão fazer a diferença entre o sucesso ou o insucesso da mesma.

Também a nível pessoal este trabalho foi bastante desafiante e motivador. Uma vez que tem por base um problema real, ter influência na sua resolução em concreto e na daqueles que têm a mesma essência constituiu um fator com muito peso na escolha deste projeto. Por outro lado, a possibilidade de dar continuidade aos conceitos aprendidos ao longo da Licenciatura e do Mestrado foi também muito importante.

Bibliografia

- A.Koen, Peter et al. (2002). *Fuzzy Front End: Effective Methods, Tools, and Techniques*. url: http://www.stevens-tech.edu/cce/NEW/PDFs/FuzzyFrontEnd_Old.pdf.
- Advice, Software (s.d.[a]). *Microsoft Power BI Software*. url: <https://www.softwareadvice.com/bi/microsoft-power-bi-profile/>.
- (s.d.[b]). *Phocas Business Intelligence Software*. url: <https://www.softwareadvice.com/bi/phocas-profile/>.
- Alle, Verna (2008). «Conceptualising 'Value for the Customer': An Attributional, Structural and Dispositional Analysis». Em: *Journal of Intellectual Capital* 9.
- Almeida, Fernando (jan. de 2017). *Concepts and Fundaments of Data Warehousing and OLAP*.
- Alooma, Inc (2017a). *ETL Resources*. url: <https://www.etltools.net/etl-tools-comparison.html>.
- (2017b). *ETL Resources*. url: <https://www.etltools.net/etl-tools-comparison.html>.
- Alves, Allyson (mar. de 2013). *Diferenças entre OLTP x OLAP*. url: <http://all-carvalho.blogspot.com/2013/03/diferencas-entre-oltp-x-olap.html>.
- Amazon (2018). *What is Kibana? – Amazon Web Services*. url: <https://aws.amazon.com/pt/elasticsearch-service/kibana/>.
- Anderson, Rick (dez. de 2017). *Introduction to ASP.NET Core*. url: <https://docs.microsoft.com/en-us/aspnet/core/>.
- Ballard, Chuck et al. (2006). *Dimensional modeling: in a business intelligence environment*. 1st. IBM.
- Bogza, R.M. e Dorin Zaharie (jun. de 2008). *Business intelligence as a competitive differentiator*. url: <http://ieeexplore.ieee.org/document/4588724/>.
- Breslin, Mary (2004). «Data Warehousing Battle of the Giants: Comparing the Basics of the Kimball and Inmon Models». Em: url: <http://olap.it/Articoli/Battle%20of%20the%20giants%20-%20comparing%20Kimball%20and%20Inmon.pdf>.
- Coleman, Chris (fev. de 2017). *Data Integration*. url: <https://www.capterra.com/p/118978/Data-Integration/>.
- Crowd, G2 (nov. de 2017). *Kibana Reviews*. url: <https://www.g2crowd.com/products/kibana/reviews>.
- Elastic (2019a). *Basic Concepts | Elasticsearch Reference [7.1]*. url: <https://www.elastic.co/guide/en/elasticsearch/reference/current/getting-started-concepts.html>.
- (2019b). *Configuring security in Kibana | Kibana User Guide [7.1]*. url: <https://www.elastic.co/guide/en/kibana/current/using-kibana-with-security.html>.
- Elasticsearch (2018). *Kibana*. url: <https://www.elastic.co/products/kibana>.
- Foundation, NodeJs (2018). url: <https://nodejs.org/en/>.

- Fowler, Martin (2015). *Patterns of enterprise application architecture*. Addison-Wesley.
- Ian (out. de 2017). *What is OLTP?* url: <https://database.guide/what-is-oltp/>.
- Immon, W. H (2002). *Building the Data Warehouse*. Wiley Computer Publishing.
- Inmon, William H., Derek Strauss e Genia Neushloss (2010). *DW 2.0: the architecture for the next generation of data warehousing*. Elsevier.
- Kimball, Ralph e Joe Caserta (2004). *The Data Warehouse ETL Toolkit: Practical Techniques for Extracting, Cleaning, Conforming, and Delivering Data*. Wiley Computer Publishing.
- Kimball, Ralph e Margy Ross (2013). *The data warehouse toolkit: the definitive guide to dimensional modeling*. Wiley.
- Koen, Peter A., Heidi M. J. Bertels e Elko Kleinschmidt (mar. de 2014). «Managing the Front End of Innovation-Part I: Results from a Three-Year Study: Senior Management Commitment, Vision, Strategy, Resource Commitment, and Culture Are the Keys to Front-End Success». Em: *Research-Technology Management*.
- Laudenschlager, Douglas e Craig Guyer (mar. de 2017). *Install Integration Services*. url: <https://docs.microsoft.com/en-us/sql/integration-services/install-windows/install-integration-services>.
- Laudenschlager, Douglas, Craig Guyer et al. (mar. de 2017). *SQL Server Integration Services*. url: <https://docs.microsoft.com/en-us/sql/integration-services/sql-server-integration-services>.
- Lindic, Jaka e Carlos Marques da Silva (nov. de 2011). «Value proposition as a catalyst for a customer focused innovation». Em: 49, pp. 1694–1708.
- Ltd, Phocas (2016). *Phocas Software - What is phocas*. url: <https://www.phocassoftware.com/software/what-is-phocas?hsCtaTracking=9acf3644-b33a-478f-80e8-0b46335166c0%7C89f8a3bd-eb3d-48a8-ab20-696d6d6a5cbc>.
- Marr, Bernard (2017). *What are Key Performance Indicators (KPIs)?* url: <https://www.bernardmarr.com/default.asp?contentID=762>.
- McBurney, Vincent (2007). *Wiki Wednesday: comparing Talend and Pentaho Kettle open source ETL tools*. url: <https://it.toolbox.com/blogs/vincentmcburney/wiki-wednesday-comparing-talend-and-pentaho-kettle-open-source-etl-tools-051607>.
- Microsoft (2016). *Documentação do Power BI - Power BI*. url: <https://docs.microsoft.com/pt-br/power-bi/>.
- MOLAP, ROLAP, And HOLAP (2015). url: <https://www.1keydata.com/datawarehousing/molap-rolap.html>.
- Moreira, Carlos (set. de 2012). *Primavera-Instalação do ERP Professional v8*. url: <https://invisibleflamelight.wordpress.com/2012/09/02/primavera-instalao-do-erp-professional-v8/>.
- Nicasio, Francesca e Ross Stanley (2018). *Key Performance Indicators. 10 metrics for guiding and measuring store success*. url: https://www.vendhq.com/images/university/retail-kpi-guide/Key_Performance_Indicators.pdf.
- Nico, Gerard (set. de 2017). *OLAP - Olap Operation (Drill, ...)* url: https://gerardnico.com/olap/olap_operation.
- Nicola, Susana, Eduarda Ferreira e João José Pinto Ferreira (mai. de 2012). «A novel framework for modeling value for the customer, an essay on negotiation». Em: 11.
- Oklahoma, University of (s.d.). *Basic Model: Retail Grocery Store*. url: <http://www.bing.com/cr?IG=9115E2FB2D2D4046B8CEB99862A8ABB0&CID=3C22B6D532256E270B7CBD43338A6F9B&rd=1&h=ijFjfcDIJt8nUZH6oqe9PW39zJVKdqFyj1QCvY5S-Y0&v=1&r=http%3a%2f%>

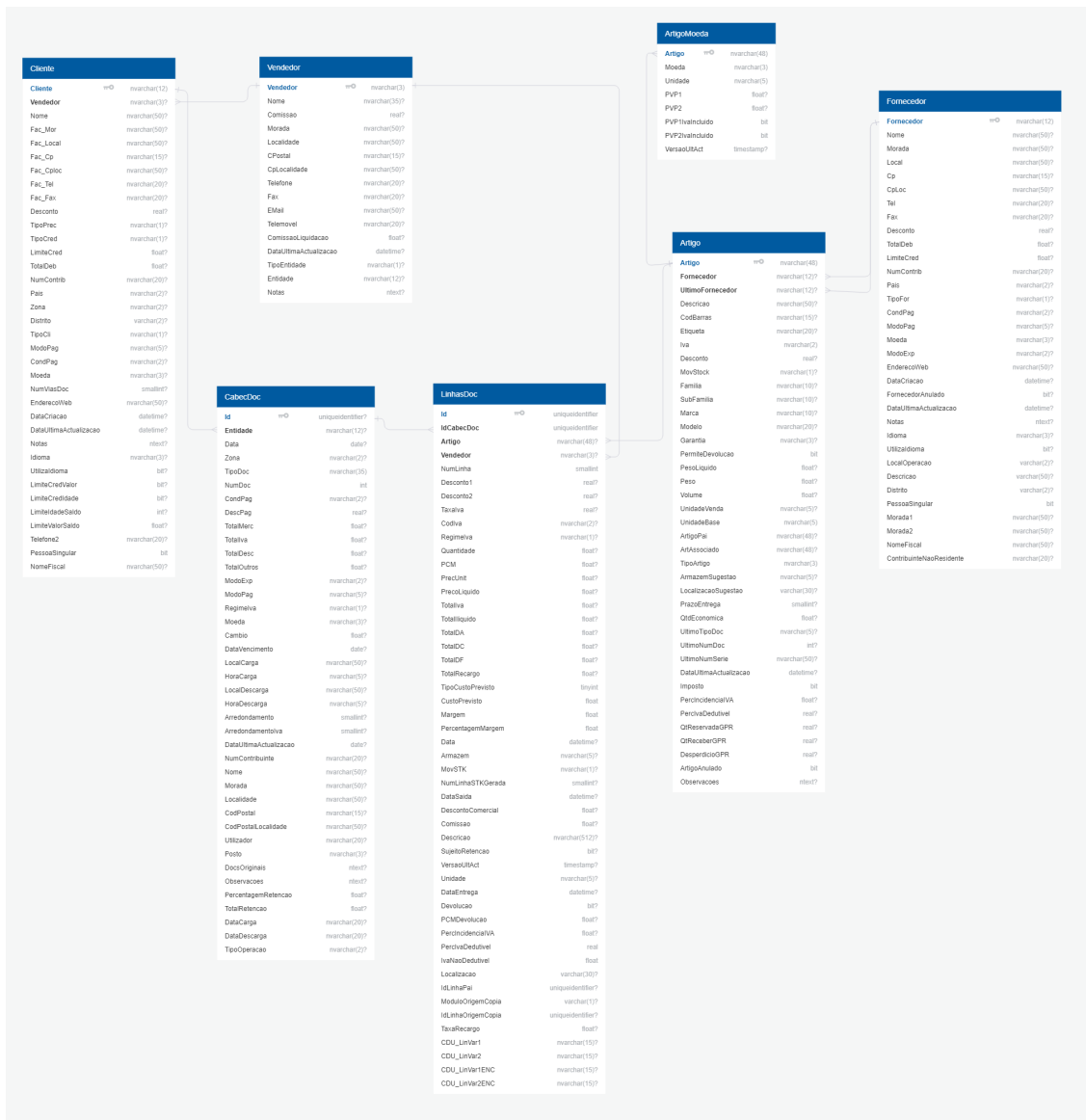
- 2fwww.ou.edu%2fclass%2faschwarz%2fDataWarehouse%2fDimensionalModelTypes.doc&p=DevEx,5068.1.
- OLTP vs. OLAP (2009). url: <https://www.datawarehouse4u.info/OLTP-vs-OLAP.html>.
- Oracle (2002). *Data Warehousing Concepts*. url: https://docs.oracle.com/cd/B10500_01/server.920/a96520/concept.htm.
- (jul. de 2005). *A Data Mart Concepts*. url: https://docs.oracle.com/html/E10312_01/dm_concepts.htm.
- (jul. de 2014). url: https://www.java.com/en/download/faq/whatis_java.xml.
- PHC (2019). *Soluções para a gestão da sua empresa*. url: <https://www.phcsoftware.com/>.
- Piton, Rafael (jan. de 2019). *O que é modelagem dimensional?* url: <https://rafaelpiton.com.br/blog/data-warehouse-modelagem-dimENSIONAL/>.
- Porter, Michael E. (2008). *Competitive Advantage: Creating and Sustaining Superior Performance*. Free Press.
- Primavera (2019). *PRIMAVERA BSS, Software de Gestão, Faturação, ERP e POS*. url: <https://pt.primaverabss.com/pt/>.
- Programiz (2018). *Learn Java Programming*. url: <https://www.programiz.com/java-programming> (acedido em 2018).
- Saaty, R.W. (1987). «The analytic hierarchy process—what it is and how it is used». Em: *Mathematical Modelling* 9.3, pp. 161–176. issn: 0270-0255. doi: [https://doi.org/10.1016/0270-0255\(87\)90473-8](https://doi.org/10.1016/0270-0255(87)90473-8). url: <http://www.sciencedirect.com/science/article/pii/0270025587904738>.
- Sage (2019). *Sage Portugal*. url: <http://www.sage.pt/>.
- Sigmacode (out. de 2017a). *5 Funcionalidades Sage para a Gestão de Empresas*. url: <https://www.sigmacode.pt/funcionalidades-sage-gestao-de-empresas/>.
- (out. de 2017b). *5 Razões para Usar um Software PHC*. url: <https://www.sigmacode.pt/5-raoes-usar-um-software-phc/>.
- SourceForge (jan. de 2018). *Talend Open Studio for Data Integration*. url: <https://sourceforge.net/projects/talend-studio/>.
- Suri, Jyothi (out. de 2016). *Descrição geral dos cubos OLAP para análise avançada*. url: <https://docs.microsoft.com/pt-pt/system-center/scsm/olap-cubes-overview?view=sc-sm-1807>.
- Talend (2018). *Why Talend: Data-Driven Analytics on a Unified Platform*. url: <https://www.talend.com/why-talend/>.
- Vantara, Hitachi (2017). *Data Integration*. url: <http://www.pentaho.com/product/data-integration>.
- Weasy (2019). *SAGE ERP*. url: <https://megavale.pt/pt/sage-erp>.
- Whitehead, Richard (abr. de 2017). *5 reasons Node.js rules for complex integrations*. url: <https://www.infoworld.com/article/3187923/application-development/5-reasons-node-js-rules-for-complex-integrations.html>.
- Woodall, Tony (jan. de 2003). «Conceptualising 'Value for the Customer': An Attributional, Structural and Dispositional Analysis». Em: 12.
- Yin, Fudong (2017). *Why Talend: Data-Driven Analytics on a Unified Platform*. url: <https://www.talend.com/why-talend/>.

A. Versão Completa da Estrutura Usada na Staging Area

Neste anexo apresenta-se a estrutura de tabelas usadas para representar a *Staging Area* desenvolvida para a solução.

Como referido anteriormente, a estrutura segue a representação utilizada pela ferramenta de gestão utilizada, o Primavera. Nesta estrutura, para representar um artigo usam duas tabelas: a tabela "Artigo" e a tabela "ArtigoMoeda". A tabela "Artigo" tem como objetivo representar as características específicas de um artigo como por exemplo a sua descrição, código de barras, peso ou volume. A tabela "ArtigoMoeda" serve apenas para configurar o preço para cada artigo. Além dos artigos apresentam-se também as tabelas onde se persistem os clientes, vendedor e fornecedores que farão parte da venda a registar.

Por último usam-se as tabelas "CabecDoc" e "LinhasDoc" para descrever as operações de negócio registadas. Destas, a tabela "CabecDoc" usa-se para guardar o tipo de operação efetuada (encomenda, venda, pedido de orçamento, entre outras), a entidade a quem se destina, entre outros detalhes de faturação. Por sua vez, a tabela "LinhasDoc" descreve o detalhe associado a cada linha dessa operação. Aqui regista-se, por exemplo, o artigo em questão, a quantidade, o preço de custo, o seu eventual desconto e comissão associado.



B. Versão Completa da Estrutura Usada na Staging Area DQP

Neste anexo apresenta-se a estrutura de tabelas usadas para representar a estratégia de *DQP*, para a *Staging Area* desenvolvida para a solução.

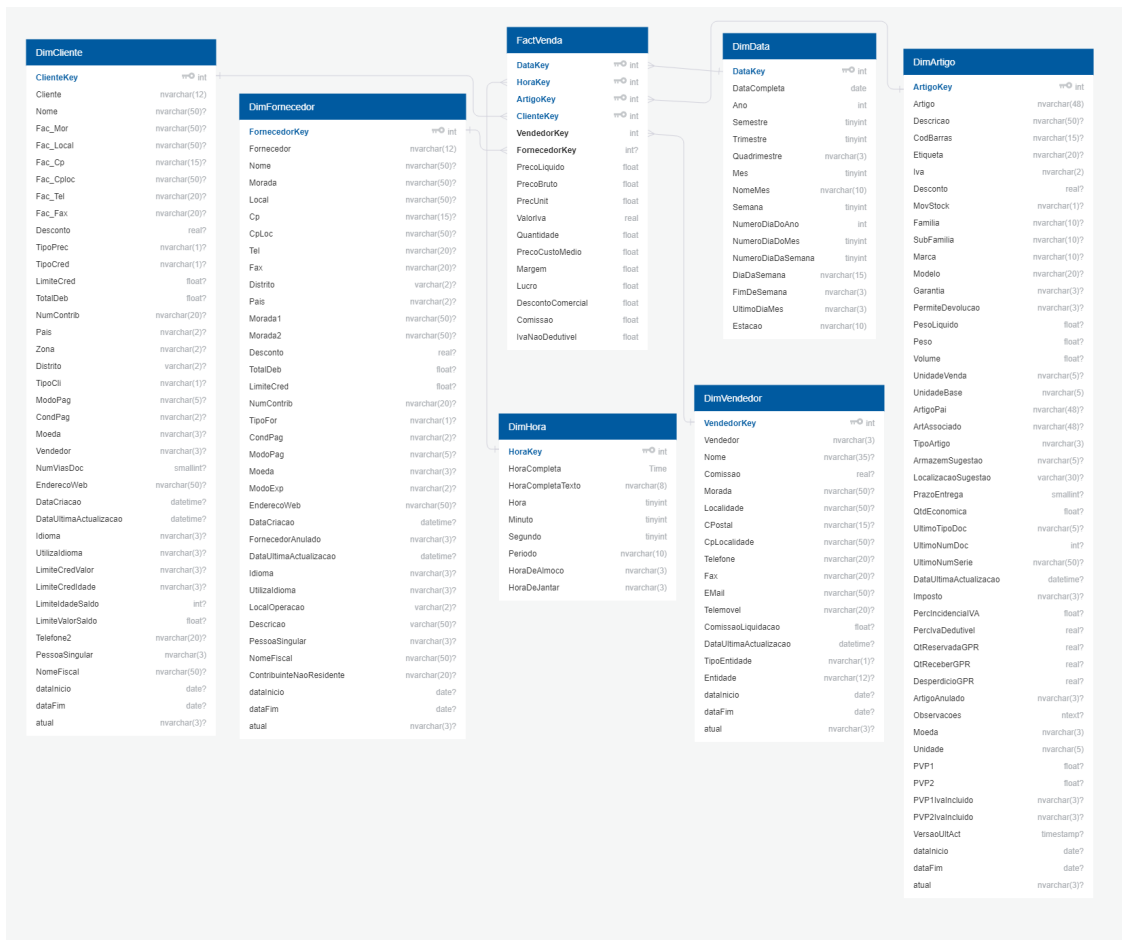
As tabelas *DQP* utilizadas são em tudo semelhantes aquelas definidas para a *Staging Area*, com a exceção de não apresentarem ligações entre si e de apresentarem quatro novos campos: *NrErros*, *DescricaoErros*, *CampoErro*, *CodigoCampoErro*. A estrutura adotada permite, mesmo em casos de eventual erro, guardar as informações que levaram à falha e assim, é possível após a alteração do processo de carregamento ou pela correção dos dados que levaram à falha voltar a tentar a inclusão destes dados nas tabelas da *Staging Area*. Além disso, uma vez que é guardado o detalhe que deu origem ao erro será possível identificar e tratar o erro de forma mais rápida e eficaz.

C. Versão Completa do Modelo Dimensional

Neste anexo apresenta-se a estrutura de tabelas usadas para representar o Modelo Dimensional desenvolvido para a solução.

Para a definição do AD seguiu-se a metodologia de *Kimball*:

- **Área de negócio:** Vendas de produtos a clientes;
- **Granularidade:** Vendas por produto, por cliente, por instante de tempo (data e hora ao segundo), por vendedor e por fornecedor;
- **Dimensões:** Data, Hora, Fornecedor, Produto, Vendedor, e Cliente;
- **Factos:** Quantidade vendida, valor bruto e líquido da venda, valor unitário, valor em imposto, valor de desconto, comissão, margem e lucro em Euros;



D. Ficheiro de Indexação no Elasticsearch

Neste anexo apresenta-se o *script* que permite a indexação dos dados armazenados pelo Modelo Dimensional no *Kibana*.

Graças a este *script*, os dados que passaram pelo processo de *ETL* desenvolvido para a criação do armazém de dados, passam a estar também disponíveis no *Elasticsearch* (processo de indexação). Ficando disponíveis no *Elasticsearch*, pela utilização do *Kibana* passaram a estar disponíveis para visualização por parte desta. Assim, acedendo ao *Kibana* estes dados poderão ser visualizados dando origem às consultas desejadas.

```

1  @echo off
2
3  set DIR=%~dp0
4  set LIB=%DIR%.lib\*
5  set BIN=%DIR%.bin
6
7  REM ???
8  echo {^
9      "type" : "jdbc",^
10     "jdbc" : {^
11         "url" : "jdbc:sqlserver://localhost:49246;instanceName=PRIMAVERA;databaseName=BD_Datamart",^
12         "user" : "sa",^
13         "password" : "Primavera1",^
14         "sql" : "select *, id as _id FROM FactVendaCopy",^
15         "treat_binary_as_string" : true,^
16         "elasticsearch" : {^
17             "cluster" : "elasticsearch",^
18             "host" : "localhost",^
19             "port" : 9300^
20         },^
21         "index" : "tmdei"^
22     }^
23 }^ | "%JAVA_HOME%\bin\java" -cp "%LIB%" -Dlog4j.configurationFile="%BIN%\log4j2.xml" "org.xbib.tools.Runner" "org.xbib.tools.JDBCImporter"

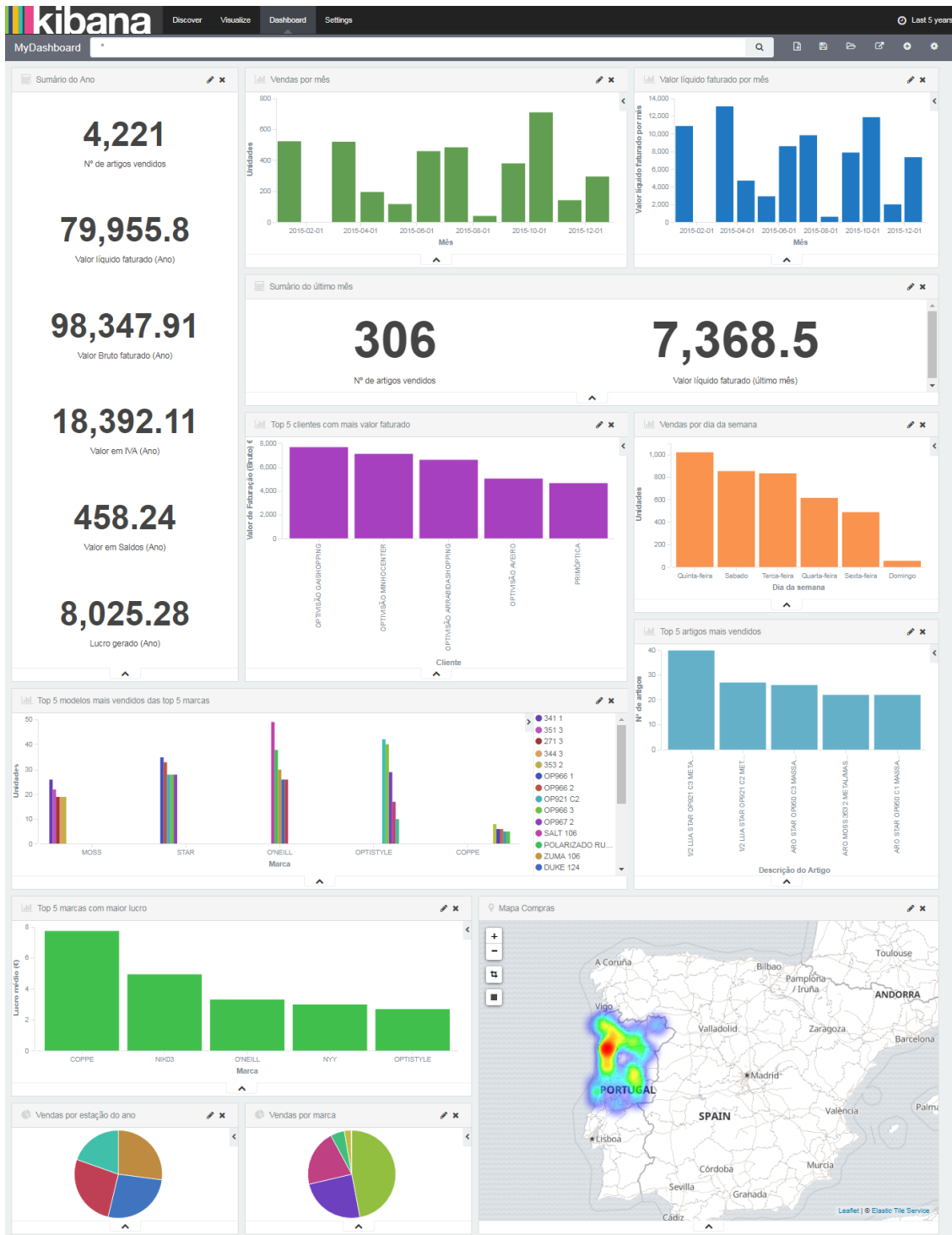
```


E. Dashboard Exemplo

Neste anexo apresenta-se o *Dashboard* de exemplo desenvolvido, na óptica de um vendedor da organização no *Kibana*.

Neste *Dashboard* apresenta-se, do lado esquerdo, algumas métricas que se definiram importantes ao longo do desenvolvimento da solução. Também numa zona inicial, definiram-se, via gráfico e via contador, alguns valores relativos ao último mês de atividade do vendedor, para que este consiga rapidamente ficar com uma ideia do panorama geral da sua atividade.

De seguida, apresentação também recorrendo a gráficos, alguns fatores responsáveis pela geração de valor para a organização: os artigos mais vendidos, os clientes com mais faturação, as marcas e modelos que mais lucros geram à organização, inclusivé um mapa de cores apresentando ao vendedor os locais onde efetua mais vendas (zona avermelhada do mapa).

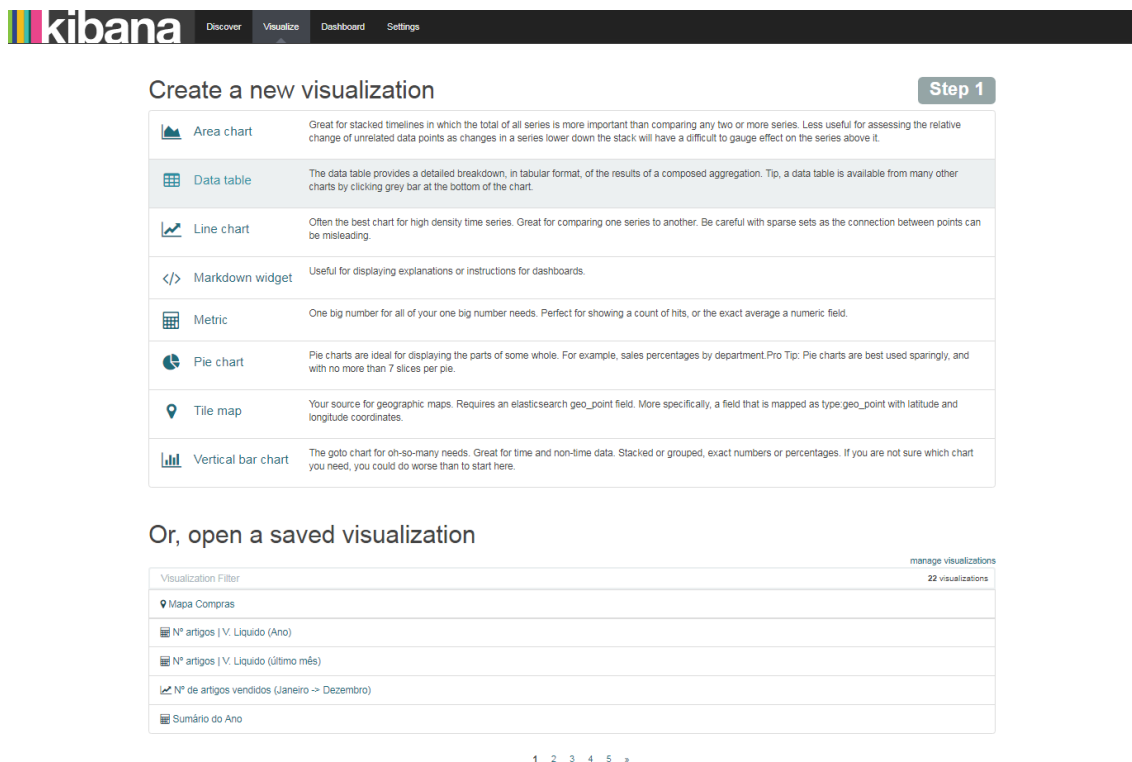


F. Tipos de Consultas Disponíveis no Kibana

Neste anexo apresentam-se os tipos de consultas disponíveis no *Kibana*.

Esta ferramenta mostrou-se muito versátil no decorrer do projeto, mostrando ser capaz de apresentar a informação em análise variadas formas, alargando o leque de possibilidades para que seja possível apresentar ao utilizador da melhor forma e/ou da forma que este desejar.

Esta versatilidade associada à consequente utilização do *elasticsearch*, oferece ao projeto muitas outras características que poderão enriquecer em muito o projeto até agora desenvolvido, como por exemplo a introdução de técnicas de *machine learning*.



The screenshot shows the Kibana interface. At the top, there is a navigation bar with the Kibana logo and tabs for 'Discover', 'Visualize', 'Dashboard', and 'Settings'. Below this, a dialog titled 'Create a new visualization' is displayed, with a 'Step 1' indicator. The dialog lists several visualization types with their descriptions:

- Area chart**: Great for stacked timelines in which the total of all series is more important than comparing any two or more series. Less useful for assessing the relative change of unrelated data points as changes in a series lower down the stack will have a difficult to gauge effect on the series above it.
- Data table**: The data table provides a detailed breakdown, in tabular format, of the results of a composed aggregation. Tip, a data table is available from many other charts by clicking grey bar at the bottom of the chart.
- Line chart**: Often the best chart for high density time series. Great for comparing one series to another. Be careful with sparse sets as the connection between points can be misleading.
- Markdown widget**: Useful for displaying explanations or instructions for dashboards.
- Metric**: One big number for all of your one big number needs. Perfect for showing a count of hits, or the exact average a numeric field.
- Pie chart**: Pie charts are ideal for displaying the parts of some whole. For example, sales percentages by department. Pro Tip: Pie charts are best used sparingly, and with no more than 7 slices per pie.
- Tile map**: Your source for geographic maps. Requires an elasticsearch geo_point field. More specifically, a field that is mapped as type geo_point with latitude and longitude coordinates.
- Vertical bar chart**: The goto chart for oh-so-many needs. Great for time and non-time data. Stacked or grouped, exact numbers or percentages. If you are not sure which chart you need, you could do worse than to start here.

Below the dialog, there is a section titled 'Or, open a saved visualization'. It shows a list of saved visualizations under a 'Visualization Filter' box. The list includes:

- Mapa Compras
- Nº artigos | V. Líquido (Ano)
- Nº artigos | V. Líquido (último mês)
- Nº de artigos vendidos (Janeiro -> Dezembro)
- Sumário do Ano

At the bottom right of the list, there is a 'manage visualizations' link and a count of '22 visualizations'. A pagination bar at the bottom shows '1 2 3 4 5 »'.

G. Estratégia de Slowly Changing Dimension

Neste anexo apresenta-se a estratégia adoptada para a manutenção dos dados quanto ao seu histórico.

Quanto à dimensão "DimArtigo", decidiu-se importante manter histórico quanto aos seguintes campos: Iva, Desconto, Familia, Marca, PrazoEntrega, QtdEconomica, Moeda, PVP1 e PVP1IvaIncluindo.

DimArtigo	Estratégia SCD
ArtigoKey	
Artigo	tipo 1
Descricao	tipo 1
CodBarras	tipo 1
Iva	tipo 2
Desconto	tipo 2
Familia	tipo 2
Marca	tipo 2
PrazoEntrega	tipo 2
QtdEconomica	tipo 2
Imposto	tipo 1
PerclIncidencialVA	tipo 1
PerclvaDedutivel	tipo 1
Moeda	tipo 2
Unidade	tipo 1
PVP1	tipo 2
PVP1IvaIncluido	tipo 2
dataInicio	tipo 1
dataFim	tipo 1
atual	tipo 1

Quanto à dimensão "DimCliente", decidiu-se importante manter histórico quanto aos seguintes campos: FacLocal, FacCp, Desconto, Pais, Zona, Distrito, PessoaSingular e NomeFiscal.

Quanto à dimensão "DimFornecedor", decidiu-se importante manter histórico quanto aos seguintes campos: Local, Cp, Distrito, Pais, Desconto e NomeFiscal.

DimCliente	Estratégia SCD
ClienteKey	
Cliente	tipo 1
Nome	tipo 1
FacMor	tipo 1
FacLocal	tipo 2
FacCp	tipo 2
Desconto	tipo 2
NumContrib	tipo 1
Pais	tipo 2
Zona	tipo 2
Distrito	tipo 2
PessoaSingular	tipo 2
NomeFiscal	tipo 2
dataInicio	tipo 1
dataFim	tipo 1
atual	tipo 1

Quanto à dimensão "DimVendedor", decidiu-se importante manter histórico quanto aos seguintes campos: Comissao, Localidade, CPostal, CpLocalidade e ComissaoLiquidacao.


Quanto às dimensões "DimHora" e "DimData" e à tabela de factos "FactVenda" não se definiram as estratégias de SCD uma vez que não se espera que se alterem ao longo do tempo.

DimFornecedor	Estratégia SCD
FornecedorKey	
Fornecedor	tipo 1
Nome	tipo 1
Morada	tipo 1
Local	tipo 2
Cp	tipo 2
Tel	tipo 1
Fax	tipo 1
Distrito	tipo 2
Pais	tipo 2
Morada1	tipo 1
Desconto	tipo 2
LimiteCred	tipo 1
NumContrib	tipo 1
NomeFiscal	tipo 2
dataInicio	tipo 1
dataFim	tipo 1
atual	tipo 1

DimVendedor	Estratégia SCD
VendedorKey	
Vendedor	tipo 1
Nome	tipo 1
Comissao	tipo 2
Morada	tipo 1
Localidade	tipo 2
CPostal	tipo 2
CpLocalidade	tipo 2
Telefone	tipo 1
Telemovel	tipo 1
ComissaoLiquidacao	tipo 2
dataInicio	tipo 1
dataFim	tipo 1
atual	tipo 1

H. Ficheiro de Venda Exemplo

Neste anexo apresenta-se, de forma resumida, um ficheiro de venda usado para validar o processo de migração e *ETL* desenvolvido, desde o seu formato origem até à sua análise gráfica via *Kibana*.

	C	D	E	F	G	I	J
1			Nº. Encomenda:	12 /	995		
2			Data:	02/11/2015			
3							
4	Amâncio Pinheiro		Cliente:	OPTIVISÃO GAIA JARDIM			
5	Dep. Comercial		Cod:	12			
6			Nome:	Optivisão GAIA I			
7	Alameda Salgueiro Maia, Lte4,2ºandar		Morada:	C.C.GAIA JARDIM, AV. DOS ESCULTORES			
8	2660 Sto António Cavaleiros		Localidade:	CANDELO			
9				VILA NOVA DE GAIA			
10							
11							
12	Cod Artigo	Descrição	Qtd	Preço	Total		PVP recomendado
13	FFOP950351	ARO STAR OP950 C3 MASSA CASTANHO HA	1	7,90 €	7,90 €		52,00 €
14	FFOP950151	ARO STAR OP950 C1 MASSA PRETO MATT 1	1	7,90 €	7,90 €		52,00 €
15	FFOP974152	NYLOR STAR OP974 C1 METAL/MASSA CINZ	1	9,40 €	9,40 €		70,00 €
16	FFOP974252	NYLOR STAR OP974 C2 METAL/MASSA CAST	1	9,40 €	9,40 €		70,00 €
17	FFOP965353	ARO STAR OP965 3 METAL CINZA MATT 15	1	9,40 €	9,40 €		70,00 €
18	FFOP965153	ARO STAR OP965 1 METAL CINZA 15	1	9,40 €	9,40 €		70,00 €
55	MOSS368351	1/2 LUA MOSS 368 3 METAL/MASSA CASTAN	1	19,00 €	19,00 €		112,00 €
56	MOSS368151	1/2 LUA MOSS 368 1 METAL/MASSA CINZA M	1	19,00 €	19,00 €		112,00 €
57	MOSS368251	1/2 LUA MOSS 368 2 METAL/MASSA CINZA/L	1	19,00 €	19,00 €		112,00 €
58	MOSS351357	ARO MOSS 351 3 METAL/MASSA CASTANHO	1	19,00 €	19,00 €		112,00 €
59	MOSS353158	ARO MOSS 353 1 METAL/MASSA CASTANHO	1	19,00 €	19,00 €		112,00 €
60	MOSS352353	ARO MOSS 352 3 METAL/MASSA AZUL 15	1	19,00 €	19,00 €		112,00 €
61	MOSS372354	ARO MOSS 372 3 METAL/MASSA PRETO MAT	1	19,00 €	19,00 €		112,00 €
62		SubTotal	49		661,90 €		
63		Desconto	0%		0,00 €		
64				Total	661,90 €		
65							
66							
67							
68	OBS						

Ao valor total incide a taxa de Iva 23%

I. Exemplo de Dados Informativos Produzidos Pelo Kibana

Neste anexo apresentam-se três exemplos, relativos a três índices diferentes, disponibilizados pelo *Kibana* para a análise aos pedidos de geração de análises gráficas.

