Dissertação

Mestrado em Engenharia Informática – Computação Móvel

Electronic Transmission of Chemical Occurrence Data: Semantic Integration to Assist in Identifying Food Data

Sidney Ricardo Brasil Tomé

Leiria, Setembro de 2013

Dissertação

Mestrado em Engenharia Informática – Computação Móvel

Electronic Transmission of Chemical Occurrence Data: Semantic Integration to Assist in Identifying Food Data

Sidney Ricardo Brasil Tomé

Dissertação de Mestrado realizada sob a orientação do Doutor Vitor Manuel Basto Fernandes, Professor da Escola Superior de Tecnologia e Gestão do Instituto Politécnico de Leiria.

Leiria, Setembro de 2013

Agradecimentos

Primeiramente gostaria de agradecer ao Professor Doutor Vitor Manuel Basto Fernandes, orientador desta dissertação, pelo suporte que nos deu ao longo deste projeto, demonstrando sempre grande interesse no tema e no trabalho em si. Certamente contribuiu de forma positiva para o avanço da dissertação.

Agradeço também a todos os meus colegas de mestrado mas em especial ao colega de projeto João Pereira pela motivação que mostrou desde o início do projeto, e também pelo esforço e empenho que sempre colocou no trabalho que era realizado.

Por fim, agradeço a todos os membros do INSA que trabalharam em conjunto connosco, sempre dispostos a ajudar a ultrapassar qualquer barreira. A disponibilidade que constantemente apresentavam e o apoio que nos deram seguramente contribuíram para conseguirmos alcançar todos os objetivos.

Esta página foi intencionalmente deixada em branco

Resumo

Este documento tem como principal objetivo apresentar o desenvolvimento de uma plataforma que será utilizada a nível nacional por várias entidades portuguesas para a recolha de dados relativos a contaminantes químicos contidos em alimentos e em outros tipos de produtos.

A plataforma web PT.ON.DATA é constituída por diversas ferramentas de análise e tratamento de dados, desenvolvidas por forma a criar a possibilidade para todas as autoridades competentes que tenham acesso a ela, submeterem com o mínimo de esforço possível os seus relatórios que contêm os dados analíticos. A submissão dos dados deverá ser feita à entidade reguladora europeia EFSA, utilizando um modelo específico de dados com o nome de Standard Sample Description, o que levou à necessidade de criação de um sistema que consegue mapear e integrar as fontes de dados (altamente heterogéneas) das entidades nacionais com as fontes de dados da EFSA.

PT.ON.DATA é a solução encontrada para Portugal com o objetivo de satisfazer as necessidades da EFSA. Como ela, existem outras soluções implementadas pelos restantes estados membros, mas a plataforma desenvolvida tenta ser o próximo passo em relação às suas antecessoras, ao simplificar as tarefas de tratamento de dados e resolver os problemas que existem para esta área, problemas tais como o mapeamento de texto livre para vocabulário controlado.

Uma ferramenta foi proposta com o intuito de auxiliar na tarefa de mapeamento. Esta ferramenta utiliza técnicas de tratamento léxico e gramatical, as quais mostraram resultados promissores ao efetuar com sucesso a relação entre uma secção do vocabulário controlado pertencente ao padrão SSD e um conjunto de entradas fornecidas por uma das autoridades portuguesas.

A plataforma PT.ON.DATA, juntamente com os seus diferentes módulos, surge como uma solução para Portugal e algo com que os restantes estados membros poderão ver como exemplo do que poderão alcançar com a evolução dos seus próprios projetos.

Palavras-chave: Processamento de informação, Text-Mining, Integração de Sistemas.

Esta página foi intencionalmente deixada em branco

Abstract

This document has the main objective of presenting the development of a platform which will be used at a national level by several Portuguese organizations for collecting data related to chemical contaminants contained in foods and other products.

The implemented PT.ON.DATA web platform consists of several analysis tools, data processing tools, developed with the of purpose of giving the possibility for all competent authorities which have access to it, to submit with minimum amount of effort their own reports that contains the required analysis data. The submitted data should be delivered to the European regulatory authority EFSA, through the use of a specifically created data model named Standard Sample Description, which led to the need of building a system that can map and integrate (highly heterogeneous) data sources from the national authorities with EFSA's data source.

PT.ON.DATA is the solution created for Portugal with the objective of satisfying all EFSA's necessities. Just like this project, there are many other solutions implemented by other member states, but the developed platform tries to be the next step in relation to their predecessors, by simplifying the data processing tasks and solve the problems that exist in this area of work, problems such as the difficult task of mapping free text to a controlled vocabulary.

A tool has been proposed in order to assist the mapping task. This uses a set of lexicon and grammar techniques, which presented promising results when performing a successful relationship between a section belonging to the controlled vocabulary of the SSD standard and a set of input data provided by one of the Portuguese authorities.

The PT.ON.DATA platform along with its different modules emerges as a solution to

Portugal and something that all member states will be able to see as an example of what can be achieved with further development and iterations of their own projects.

Key-Words: Information Procesing, Text-Mining, Systems Integration.

Esta página foi intencionalmente deixada em branco

Índice de Figuras

Esta página foi intencionalmente deixada em branco

Índice de Quadros

Esta página foi intencionalmente deixada em branco

Lista de Siglas

AC	Autoridade Competente
API	Application Programming Interface
DBMS	DataBase Management System
DCF	Data Collection Framework
EM	Estado Membro
LIMS	Laboratory Information Management System
POS	Part-Of-Speech
SQL	Structured Query Language
SSD	Standard Sample Description
XML	Extensible Markup Language
XP	eXtreme Programming
XSD	XML Schema Definition

Esta página foi intencionalmente deixada em branco

Índice

Esta página foi intencionalmente deixada em branco

Introdução

Existe uma necessidade de avaliar e comunicar todos os riscos associados com a cadeia alimentar, para tal surgiu a entidade europeia EFSA (European Food Safety Authority) que tem como principal objetivo recolher e analisar dados que permitam a caracterização e o controlo dos riscos que tenham impacto direto ou indireto na segurança dos géneros alimentícios ou dos alimentos para animais. A EFSA necessita que cada um dos países, denominados de Estados Membros (EM), coloque em vigor a legislação alimentar e que procedam ao controlo e à verificação da observância dos requisitos relevantes dessa mesma legislação. Cada país (ou Estado Membro) possui diversas Autoridades Competentes (AC) responsáveis pela recolha de dados relativos a contaminantes químicos em diferentes matrizes, as quais realizam deferentes planos de controlo.

A recolha de dados analíticos é uma importante tarefa da EFSA e a avaliação de riscos permite responder de forma substanciada e rápida a todas as solicitações, de modo a que os potenciais riscos possam ser rapidamente avaliados e para que os gestores do risco possam agir os mais rapidamente possível, se necessário.

Em Portugal o Instituto Nacional de Saúde Dr. Ricardo Jorge (INSA), entidade pública integrada na administração indireta do Estado, é responsável pela agregação dos dados recolhidos pelas diferentes ACs para serem posteriormente enviados à EFSA. O INSA é constituído por vários departamentos, de entre os quais encontra-se o Departamento de Alimentação e Nutrição que, como o nome indica, desenvolve atividades nas áreas da segurança alimentar e nutrição. Este é o departamento encarregue por colaborar diretamente com os organismos nacionais (ACs) e internacionais (EFSA).

O controlo oficial dos alimentos está atualmente coordenado pela Direção-Geral de Alimentação e Veterinária (DGAV) do Ministério da Agricultura, do Mar, do Ambiente e do

Ordenamento do Território (MAMAOT). O controlo oficial é baseado em um Plano Nacional de Controlo Plurianual Integrado, que compreende 36 planos setoriais incluindo amostragem direcionadas e aleatórias que pretendem responder às exigências da Legislação Europeia e é executado por várias autoridades nacionais competentes, quer pertencentes ao MAMAOT ou ao Ministério da Economia e do Emprego (MEE). Os planos de controlo setoriais que incluem análise química são executados pelo:

- Autoridade de Segurança Alimentar e Económica (ASAE/MEE);
- Direção Geral de Alimentação e Veterinária (DGAV/MAMOT);
- Instituto Português do Mar e da Atmosfera (IPMA/MAMAOT);

As análises químicas são executadas essencialmente pelos Laboratórios Oficiais, onde a informação analítica necessária para a transmissão de dados para EFSA residem em fontes como LIMS (Laboratory Information Management System), bases de dados Access, Excel ou relatórios em papel. O fluxograma abaixo descreve de uma forma simplificada as relações entre as entidades envolvidas nos programas de controlo oficiais.

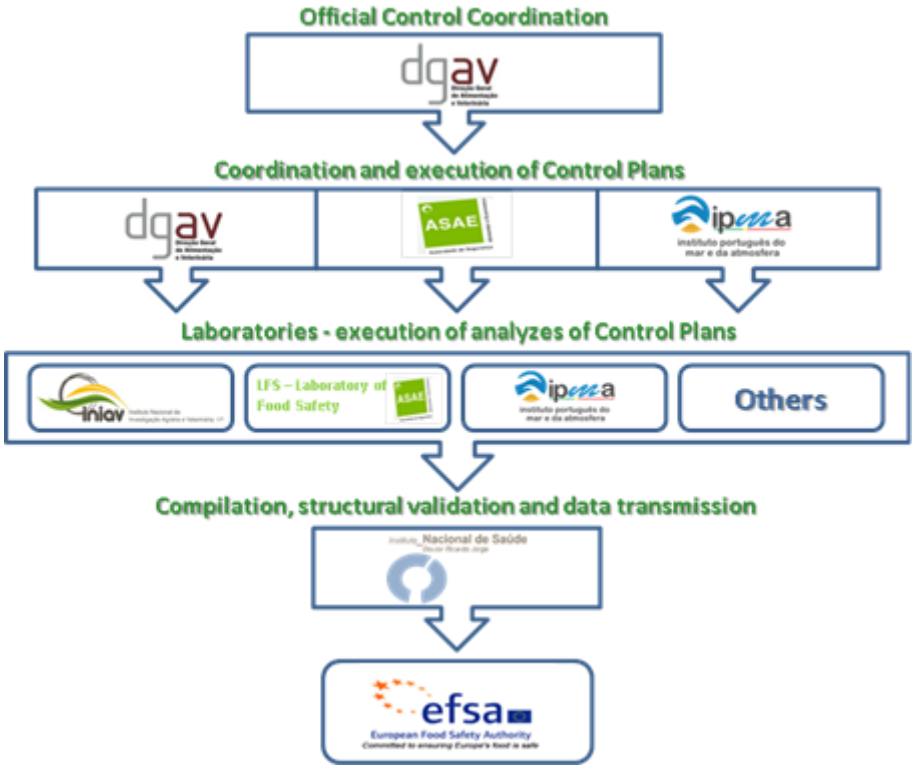


Figura 1. Diagrama que demonstra o fluxo de dados entre as várias entidades.

1.1 Objetivos

Das análises efetuadas resultam um novo conjunto de informação sobre cada uma das amostras bem como os seus respetivos resultados, que por sua vez são agregados e armazenados de forma diferente mediante a autoridade encarregue e o laboratório a elaborar as análises. O formato e o modelo adotado por cada uma das entidades tornam o processo de coleção de dados para o envio à EFSA mais complicado, comprometendo o uso desses mesmos dados e contribuem para dificuldade de comparação, compilação e preparação dos registos. Devido a forma com que os meios financeiros estão condicionados, tentar aplicar alterações aos vários processos internos de uma entidade não poderá ser visto como uma opção. Por estes motivos, e também com o objetivo de melhorar a comparabilidade dos dados técnicos recebidos e analisados e facilitar a sua transmissão, a EFSA decidiu criar em 2010 um modelo de dados que fosse considerado um *standard* para a forma com que cada registo de amostra e cada resultado das análises fosse reportado [1]. Este novo *standard,* denominado de "Standard Sample Description for Food and Feed", está atualmente a ser respeitado pelos vários estados membros. Para explicar de forma sintetizada a constituição do padrão SSD, pode-se dizer que é composto por atributos, cada um destes atributos podem representar uma tabela, e esta tabela é composta por códigos que simbolizam uma descrição. Um exemplo disto poderá ser o atributo "País de Origem da Amostra" que, em exemplo prático, pode vir preenchido com o código PT, o qual representa a descrição de "Portugal". O capítulo mais a frente de Projeto de Software irá explicar de forma mais aprofundada o padrão SSD.

Até ao ano de 2012 não existia nenhuma base de dados a nível nacional que possuísse dados detalhados sobre ocorrências químicas nos alimentos. Portugal, através do INSA, decidiu concorrer à proposta CFP/EFSA/DATEX/2011/01[1] intitulada de "Implementation of Electronic Transmission of Chemical Occurrence Data in Portugal" criada pela EFSA da qual iria ter como mentores a Food Safety Authority of Ireland (FSAI). Com este projeto em mente, Portugal conseguiria então resolver o problema de comunicação de dados entre as várias autoridades competentes, seria agora também possível possuir um ponto central onde seriam armazenados todos os registos referentes às ocorrências químicas nos alimentos, e por fim, Portugal iria pertencer ao grupo de países que aceitariam internamente o novo SSD *standard* como forma de comunicação de dados entre si e a entidade europeia EFSA.

[1] http://www.efsa.europa.eu/en/datex201101/docs/cfpefsadatex201101guide.pdf

O objetivo principal do projeto foi então a criação e manutenção de uma base de dados nacional, que compila dados sobre os planos de controlo realizado em Portugal para a ocorrência da contaminação química nos alimentos e géneros alimentícios. Outro objetivo é recolher todos os dados nacionais (controlos oficiais) existentes a partir de 2009 até 2011 e levar o LIMS existente às autoridades nacionais para que, no futuro, possam incluir campos correspondentes ao SSD que estavam anteriormente ausente nos seus registos, melhorando assim a qualidade dos dados nacionais.

Para o desenvolvimento do projeto foi necessário criar, implementar e testar um sistema para a recolha de dados de diferentes fontes nacionais, transformando-os de seguida no padrão definido pela EFSA (SSD), fazendo uso também do sistema de classificação e descrição de alimentos intitulado de FoodEx [2] e proceder a validação e subsequente envio dos dados à EFSA, após transformados, em formato XML. O processo de envio centra-se primeiramente nos dados de contaminantes químicos, mas tem que ser flexível o suficiente para suportar outros tipos de dados que possam estar definidos no *standard* criado.

Os objetivos iniciais e específicos deste projeto foram:

- Desenvolver uma base de dados nacional de acordo com SSD;
- Mapear os diferentes sistemas nacionais (LIMS nacionais) para o formato SSD e vocabulários controlados;
- Desenvolver ferramentas de tradução e de mapeamento;
- Popular a base de dados nacional com dados históricos nacionais desde 2009;
- Transmitir os registos provenientes da base de dados nacional para a EFSA no formato XML através da plataforma de recolha de dados da EFSA (DCF) [3];

1.2 Estrutura do documento

O documento está dividido em cinco capítulos, tendo início no capítulo corrente de introdução onde foi dado a conhecer um pouco sobre o âmbito do projeto juntamente com os aspetos principais que contribuíram para a sua origem. De seguida no mesmo capítulo foram descritos os principais objetivos focados pelo projeto ao longo do seu desenvolvimento.

O segundo capítulo diz respeito à revisão da literatura onde são apresentados os projetos desenvolvidos pelos outros países que aderiram à mesma proposta criada pela EFSA. Aqui neste capítulo será explicado como estes outros projetos contribuíram para a realização deste.

O terceiro capítulo irá servir descrever o projeto de *software* em si, mostrando o modelo de desenvolvimento utilizado, a gestão e planeamento do projeto, as normas utilizadas e outros pontos de mesma importância.

O quarto capítulo irá expor o sistema desenvolvido onde serão descritos todos os serviços relacionados com a solução encontrada, bem como as ferramentas criadas a partir de várias análises e iterações realizadas sobre a plataforma PT.ON.DATA onde conseguiu-se identificar vários problemas e encontrar-se soluções possíveis que vieram a ser implementadas.

Por fim serão apresentadas as conclusões tendo em conto tudo que tinha sido exigido no início do projeto, os objetivos definidos, os resultados obtidos e o quanto é que este projeto contribuiu para que a proposta criada pela EFSA seja concluída.

Revisão da literatura

Este capítulo pretende ilustrar o trabalho de pesquisa efetuado como auxílio no desenvolvimento do projeto. Como enunciado anteriormente no capítulo de introdução, existem outros países que aderiram à proposta criada pela EFSA resultando na criação dos seus próprios projetos. Estes projetos foram publicados e ajudaram na criação da solução para Portugal. De seguida irá ser explicado a forma com que os vários aspetos dos diferentes projetos influenciaram este.

2.1 Situação atual nos outros países

2.1.1 Solução desenvolvida pela Irlanda

Para ajudar na implementação deste projeto, Portugal recebeu como mentores a entidade irlandesa FSAI que não só já tinham desenvolvido a sua solução para o seu país, como também estaria em funcionamento ao longo de alguns anos, portanto faria bastante sentido seguir as suas recomendações e experiência como forma de modelo para se seguir.

A FSAI possuía os mesmos objetivos principais que podem ser encontrados neste projeto. A adaptação dos dados nacionais para o padrão SSD criado pela EFSA continuava a ser considerado como fim a alcançar, classificado com elevada importância.

No entanto a entidade irlandesa decidiu evitar por completo o facto de serem eles a terem de aplicar a transformação dos dados que possivelmente iriam ser enviados pelas várias organizações de controlo existente no seu país. Isto significa que antes de preencherem a base de dados nacional, controlada pela FSAI, todas as entidades envolvidas teriam de ter utilizar um formato acordado entre eles para a exportação dos registos das amostras presentes nos

seus LIMS. O processo de criar um acordo entre todas as entidades irlandesas demorou cerca de 10 anos para ser concluído.
O desenvolvimento do projeto da FSAI começou pela adaptação da base de dados nacional de forma a estar preparada para receber registos com as linguagens controladas e novos campos definidos no padrão SSD.

Como o padrão SSD foi construído a base de códigos para melhor conseguir-se identificar uma caraterística de uma amostra, a FSAI desenvolveu diversas ferramentas para conseguirem aplicar os códigos aos vários atributos de um registo. Visto que o padrão SSD será explicado em muito mais detalhe mais a frente neste relatório, e que existe agora uma necessidade de dar a entender a estrutura de um registo em SSD, consideremos a seguinte entrada na base de dados:

Tabela 1. Exemplo de uma porção de uma entrada em SSD

Parameter Code (R.06)	Parameter Text (R.07)	Parameter Description	Parameter Type (R.08)	Analytical Method Code (R.09)
		Aflatoxinas	Individual	F005A

A tabela acima mostra apenas cinco atributos de registo SSD e é possível notar que nem todos os atributos estão preenchidos com valor. Isto deve-se ao facto de que quando uma entidade efetua análises à uma amostra, esta entidade não obtém todos os valores que são requeridos pela EFSA e necessários no padrão SSD. É óbvio que uma entidade ao fazer um certo tipo de análise e saber que uma amostra apresenta vestígios de, por exemplo, Aflatoxinas, é impossível saber que código é que a EFSA atribuiu a esta microtoxina. Para tal a FSAI desenvolveu a ferramenta FoodEx Coding Application [4], que é basicamente uma aplicação web criada para que quando recebessem registos similares ao apresentado na Tabela 1, conseguissem preencher de uma forma simplificada os códigos pertencentes ao atributo R.06 e ao atributo R.07.

Ao analisar a ferramenta FoodEx Coding Application implementada, notou-se que a sua utilização é demasiada complicada pela forma com que é feita a pesquisa por níveis para encontrar um certo código SSD, e, apesar de fazer sentido ser complexa mas ao mesmo tempo ser apenas utilizada por elementos da FSAI que teriam por sua vez formação suficiente para a

utilizar, decidimos que não poderia ser desenvolvido algo semelhante para o projeto do INSA visto que o produto desenvolvido é para ser utilizado pelas entidades externas ao INSA que não possuem conhecimentos quaisquer sobre o padrão SSD. Abaixo encontra-se então uma imagem que mostra a interface de utilização do FoodEx Coding Application e que pretende servir de auxílio para que se perceba melhor o porquê da decisão tomada.

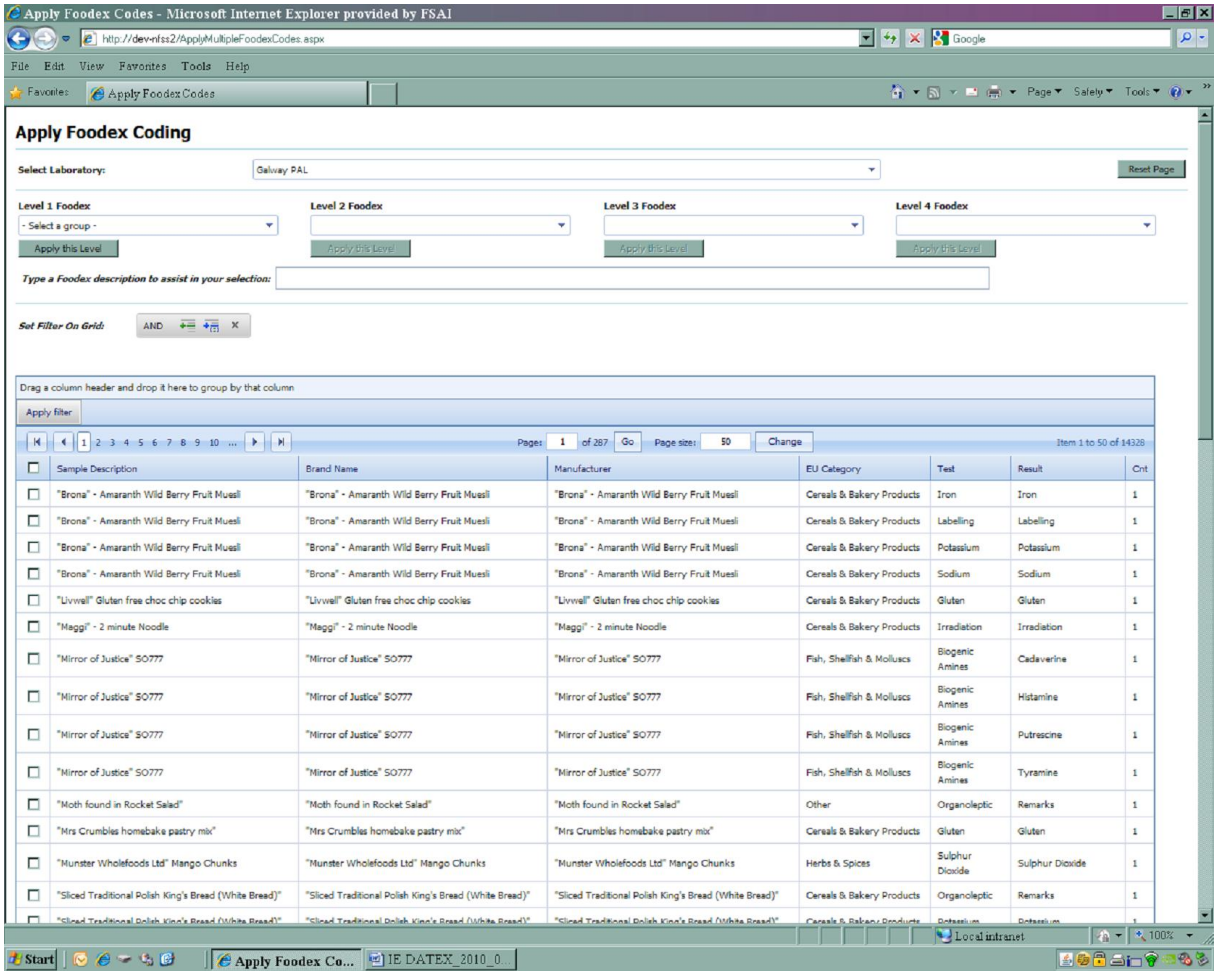


Figura 2. Screenshot da FoodEx Coding Application [4].

À parte da outras ferramentas criadas que basicamente pretende resolver problemas básicos, como por exemplo, envio dos relatórios para a EFSA através de *web services*, existe mais uma ferramenta que chamou a atenção para a criação da nossa solução. Esta ferramenta destina-se a ser utilizada com o intuito de aplicar a transformação dos dados presentes em tabelas e vistas para o formato XML, visto que a EFSA apenas aceita os relatórios das amostras em XML [3]. A abordagem que tomaram para aplicarem a descrita transformação dos dados difere das outras todas. A FSAI decidiu utilizar uma ferramenta comercial

denominada de Altova MapForce[2], muito bem reconhecida por empresas que já fizeram uso dela, como também pelos críticos ao receber uma vasta quantidade de prémios.

Ao analisar melhor o produto Altova MapForce é possível notar de que se trata de um conjunto vasto de ferramentas que conseguem lidar com inúmeros aspetos presentes em uma base de dados, permitindo efetuar qualquer tipo de mapeamento. Esta solução é ideal para o problema de como passar dados armazenados em tabelas para um ficheiro com o formato XML.

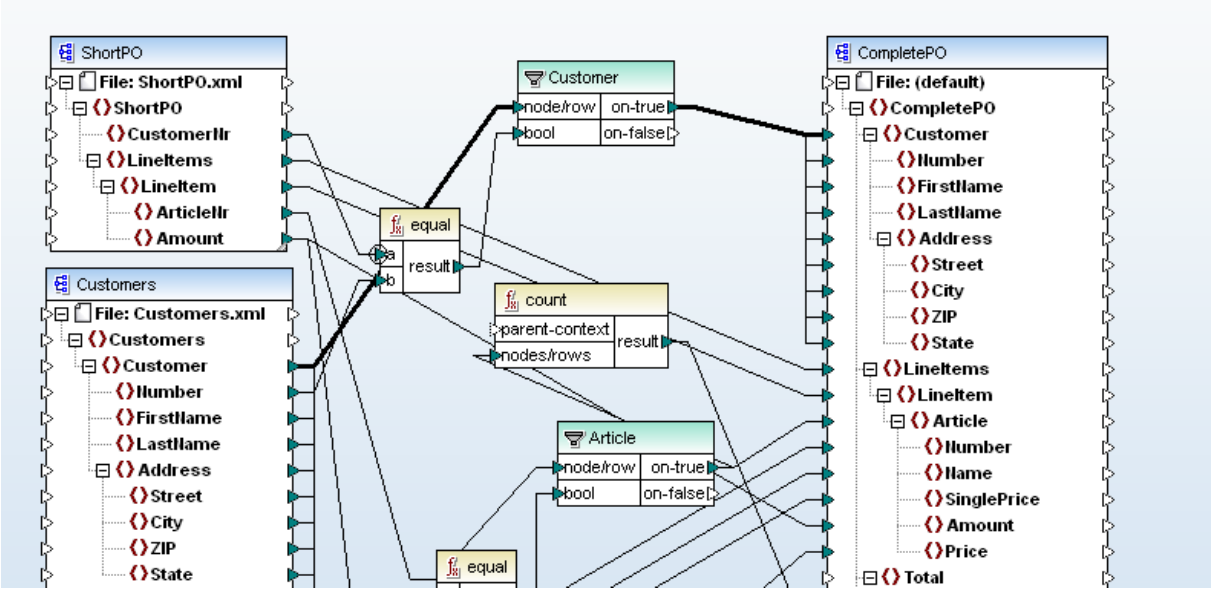


Figura 3. Vista geral do produto Altova MapForce 2

Apesar de a MapForce ser uma excelente ferramenta, esta não irá ser utilizada para a solução de Portugal, uma vez que optou-se por evitar produtos comerciais que não possuam licenças livres, resultando então na decisão de implementar algo de raiz que fizesse o mapeamento necessário e que ao mesmo tempo fosse simples de forma a evitar ter um grande número de funcionalidades onde seria necessário apenas uma.

De forma a resumir a proposta de projeto feita pela Irlanda resta analisar as dificuldades e conclusões que obtiveram. Se formos resumir os problemas que enfrentaram podemos dividi-los em duas categorias. A primeira categoria seria então a barreira física com que com que se depararam ao tentar criar um acordo entre as várias entidades irlandesas para que utilizassem apenas um formato de envio dos dados para a FSAI. Este processo demorou vários anos para

[2] http://www.altova.com/mapforce.html

estar completamente definido. No entanto, esta dificuldade não surgiu na elaboração da solução para Portugal visto que tentou-se ao máximo evitar criar esforços de qualquer tipo para as entidades envolvidas. Esta foi uma das principais preocupações que foram tidas em conta desde o início do processo de produção deste trabalho, a de tentar facilitar ao máximo as funções executadas pelas entidades Portuguesas, funções estas que estivessem relacionadas com as necessidades impostas pela EFSA. Foi percetível na altura do início do projeto de que ao tentar proteger as várias instituições iria trazer, e realmente trouxeram, grandes dificuldades que afetaram o desenvolvimento. Apesar destas complicações decidimos continuar já que a alternativa iria atrasar bastante o projeto.

Resta então analisar a segunda categoria de problemas que encontraram. Grande parte desta categoria é composta problemas relacionados com a estrutura do padrão SSD, como por exemplo, foi relatado que houve valores que não conseguiam mapear diretamente para um código SSD, ou porque não havia detalhe suficiente ou porque se tratava de uma versão antiga do SSD não possuindo todos os tipos de códigos, pelo menos os mais importantes. O relatório que fizeram a estes tipos de problemas ajudou no desenvolvimento no projeto para Portugal contribuindo de forma positiva para concluirmos a solução com o mínimo de obstáculos possíveis.

2.1.2 Solução desenvolvida pela Alemanha

A seguinte solução proposta pela Alemanha apresenta dois pontos interessantes que não foram colocados por parte da Irlanda pelo facto destes dois estados membros terem seguido caminhos completamente diferentes ao planearem e desenvolverem os seus projetos. Possuindo os mesmos objetivos que os restantes países que aceitaram a proposta criada pela EFSA, a Alemanha decidiu seguir a mesma abordagem presente no trabalho que está a ser descrito ao longo deste relatório.

A entidade alemã BVL (Federal Office of Consumer Protection and Food Safety, trandução para inglês) encarregue pelo projeto, possuía uma base de dados nacional onde iria receber os dados das várias instituições alemãs, dados estes que iriam chegar no seu estado original sem qualquer tipo de tratamento. Esta base de dados, construída sobre tecnologias da Oracle®, foi basicamente o sistema raiz de todo o projeto, de onde surgiram as várias ferramentas descritas no relatório elaborado por eles.

O processo de automatização da receção e envio dos dados foi dividido em várias etapas:

- Análise da informação presente na base de dados da BVL;
- Construção de tabelas de mapeamento/tradução;
- Implementação de uma aplicação para a seleção, reconstrução e validação dos dados;
- Implementação de uma aplicação para gerar o ficheiro XML;
- Envio dos dados à EFSA.

A primeira etapa que a BVL descreveu aplica-se a qualquer um dos projetos. A análise à informação que chega é indispensável, é necessário sempre saber a forma com que os dados estão estruturados assim que vêm das suas fontes, logo torna-se bastante complicado implementar uma solução deste tipo sem efetuar esta etapa.

A segunda etapa já é por si só um tópico de grande importância, e como poderá ser visto mais a frente neste relatório, foi de grande auxílio para a criação de um dos módulos deste projeto. A BVL começa a falar sobre um aspeto que pode afetar de forma negativa tanto o desenvolvimento de um projeto como também a sua manutenção. Este aspeto diz respeito a quantidade de técnicos e pessoal especializado que deverão estar envolvidos para efetuaram certas tarefas específicas como, utilizando este caso, a criação manual de tabelas de mapeamento.

Como os dados chegavam à base de dados da BVL sem qualquer tipo de tratamento para satisfazerem as necessidades da EFSA, era necessário que alguém aplicasse as ditas transformações mas a um grande volume de dados. Se uma organização possuísse poucos recursos humanos ter-se-ia de fazer um grande esforço despendendo grandes quantidades de tempo para tentar concluir a tarefa, podendo até comprometer o projeto, mas pelo outro lado se uma organização possuir um número elevado de recursos humanos, esta terá de despender não tempo mas sim das suas fontes monetárias para os pagar. O cenário ideal seria o menor número de pessoas envolvidas juntamente com ferramentas que lhes facilitassem a vida, já que será sempre necessário ter algum suporte humano pode detrás de tarefas onde requerem alguma cautela como é o caso da criação de tabelas de mapeamento.

Tabela 2. Tabela de mapeamento criada pela BVL [5]

BVL Catalogue	Number of Entries	Content	EFSA Catalogue	Number of Entries	Used in fields of SSD
Gemeinde-kennziffern	470 districts	Regions, towns, municipalities…	Nuts	488 for DE	S05, S07
Matrix	roughly 5000	Food code	FOODEX	roughly 3700	S12, S14, S17
Zusatzangaben	34	Production method …	Prodmd	14	S15
Verpackungen	46	Packing materials	Prodpac	46	S16
Verarbeitung	56	Product treatment..	Prodtr	39	S17
Probenahmegründe	30	Reasons for sampling	Sampstr Srtype Smpmd	8 7 5	S33 S34 S35
Betriebsarten	280	Sampling points	Smpnt	64	S39
Parameter	roughly 9500	Parameters	Param Partype	5207 3	R06, R07, R08
Methodensammlung	75	Methods	Anlymd Mdstat	113 4	R10, R11 R12
Maßeinheiten	130 (only 6 recoded)	Dimensions	Unit	190	R13
Bezug	4	Expression of Result	Expres	4	R25

Esta tabela de mapeamento que foi retirada do relatório técnico da BVL é descrita como sendo uma lista de tradução de códigos, em que o seu objetivo é de traduzir os códigos originais de cada um dos registos para códigos respectivos definidos no padrão SSD. Para que se perceba melhor a função desta tabela podemos analisar o significado de cada uma das colunas.

As três primeiras colunas na Tabela 2 descrevem, respetivamente, o nome da tabela presente na base de dados, a quantidade de registos la contidos, e o tipo de informação que cada registo representa. Por sua vez, as três últimas colunas formam um grupo que representam informações relativas ao padrão SSD e descrevem para qual tabela SSD devem os registos ser traduzidos, o número de registos que se conseguem traduzir, e a qual atributo do SSD é que irão estar associados.

Logo de início consegue-se identificar um pormenor bastante importante na tabela acima. Se compararmos o número de entradas inalteradas apresentadas na coluna "Number of Entries" que está adjacente à coluna "BVL Catalogue", com o número de entradas que foram traduzidas com sucesso indicadas pela coluna "Number of Entries" que está a seguir a coluna "EFSA Catalogue", é possível notar a grande discrepância. Existem dois casos que poderão explicar a diferença de valores entre estas duas colunas, o primeiro sendo a grande quantidade de detalhe ou informação presente em um registo.

A princípio, uma entrada deverá representar informação sobre um resultado de uma amostra, no entanto isto poderá ser verdade para a entidade que a criou e não para a EFSA, uma vez que ao analisar uma amostra, vários resultados poderão originar e a informação de cada um destes resultados poderá estar contida em uma só entrada, indo contra as necessidades definidas pela EFSA através do SSD. Portanto, uma só entrada em uma tabela da BVL poderá originar uma ou mais entradas em uma tabela designada para o SSD, sendo assim possível explicar porque em algumas linhas na Tabela 2 o número de entradas aumenta ao fazer-se a tradução.

O segundo caso que poderá contribuir para a diferença no número de registo ao aplicar a tradução é considerado o mais importante para o projeto desenvolvido para o INSA. Ao contrário do primeiro caso, as entradas presentes nas tabelas poderão não possuir informação necessária para se conseguir criar uma entrada que respeite as normas do padrão SSD. Quando não se consegue executar a tradução de uma entrada por causa da falta de informação, então procede-se por entrar em contato com a entidade que deu origem aos dados. Este processo e dificuldade está descrito no relatório elaborado pela Alemanha [5] e eles afirmam que, após o pedido ser feito, os dados que recebem como reposta continuam a ser insuficientes, onde a informação necessária continua a não estar presente. Para além do

tempo e recursos humanos empregados neste processo, existe informação essencial e necessária que não é transmitida à EFSA, comprometendo o projeto e todas as entidades envolvidas. A única solução que é apresentada pela BVL é esperar que programas e procedimentos que irão ser criados no futuro consigam impor às várias entidades que todos os dados necessários para o padrão SSD sejam enviados por elas.

Finalizando a análise e discussão do ponto sobre a criação de tabelas de tradução, segue-se então o ponto de aplicação criada para a seleção, reconstrução e validação dos dados. Pelo que foi descrito pela BVL, não houve implementação de uma ferramenta mas o que fizeram assemelha-se ao procedimento realizado pela Irlanda. Devido ao facto de terem mais de 4 anos de experiência com as tecnologias Oracle associadas às bases de dados, a BVL decidiu fazer uso uma ferramenta denominada de Oracle Discoverer[3]. Esta ferramenta foi adaptada para o projeto para que se conseguisse selecionar as diferentes variedades de dados presentes nas várias tabelas. Visto que o padrão SSD possui várias formas de se conseguir obter um determinado código, faz todo o sentido que exista a necessidade de se ter de se adequar uma ferramenta, seja ela qual for, ao ambiente em que irá trabalhar, neste caso o ambiente será o padrão SSD e as suas regras. As tabelas de tradução, anteriormente discutidas, também foram utilizadas pelo Discoverer para aplicar a reestruturação dos dados.

No mesmo ponto onde é discutida a utilização do Oracle Discoverer, a BVL descreve uma vantagem que conseguiu obter ao utilizar esta ferramenta. Um dos vários processos que são realizados com os dados originais é a validação dos mesmos. O processo de validação consiste em aplicar certas verificações aos dados, estas verificações estão todas registadas em um documento com o nome de EFSA Guidance on Data Exchange [3] onde nele são descritas uma grande de quantidade regras com que os dados deverão obedecer. Existem regras que obrigam que certos valores não possam ter um mais que um certo tamanho definido em termos do número de caracteres, como por exemplo, a descrição de uma amostra não poderá ter mais que 100 caracteres. No entanto existem regras mais sofisticadas que têm em conta valores de múltiplos atributos, como por exemplo, se o AtributoX tiver um valor superior ao valor do AtributoY então o AtributoZ terá de estar preenchido com o valor "X000A".

A BVL conseguiu implementar estas regras na ferramenta Discoverer, e a vantagem de

[3] http://www.oracle.com/technetwork/developer-tools/discoverer/overview/

utilização desta ferramenta, vantagem esta acima exposta, surge quando ao pôr a aplicação em execução, consegue-se listar quais as entradas que satisfazem e quais não satisfazem as regras definidas. Os dados inválidos poderão então ser alocados em uma tabela distinta para caso queira-se excluir estes dados ou então utiliza-los para procedimentos adicionais de modo a que possam estar de acordo com as regras.

Para o processo de gerar o ficheiro XML para que seja posteriormente enviado à EFSA, a entidade alemã decidiu por desenvolver a sua própria ferramenta. A razão pela qual justificam esta decisão é a de não conseguirem encontrar um produto no mercado adequado para os fins do projeto. Como foi analisado anteriormente na secção onde se estudou o projeto da Irlanda, existe realmente uma solução no mercado para o problema, mas talvez o que a BVL tenha seguido o mesmo princípio aplicado ao projeto do INSA, o princípio de tentar evitar ao máximo utilizar produtos não fossem livres, mesmo que implique desenvolver uma ferramenta de raiz.

Portanto a BVL optou por desenvolver uma aplicação utilizando a plataforma de gestão de bases de dados Microsoft Access[4]. Esta aplicação em Access iria importar os dados válidos selecionados pelo Oracle Discoverer e produzia então um ficheiro XML com a estrutura definida pela EFSA. O processo de gerar o ficheiro aplica ainda dois tipos de validações, a validação à estrutura do XML através de um XML Schema disponibilizado pela EFSA, e uma nova validação dos dados la contidos de modo a certificar-se que nenhuma informação é perdida no processo.

À parte da utilização da plataforma Access para a criação de uma ferramenta e do Oracle Discoverer para a seleção dos dados, o projeto desenvolvido para INSA apresentado neste relatório executa um processo idêntico para gerar o ficheiro XML. As duas validações efetuadas, tanto a estrutura do ficheiro como aos seus dados, são realmente necessárias de se aplicar e o que comprova isto é a qualidade dos resultados obtidos pela Alemanha e a experiência obtida ao desenvolver o projeto do INSA, onde por várias vezes ocorreram erros em que a causa era exatamente a perda ou transformações indesejadas que eram aplicadas aos dados no processo criação do ficheiro XML.

[4] http://office.microsoft.com/en-us/access/

O procedimento que a BVL executa para o envio dos dados à EFSA não apresenta nenhuns pontos que requeiram uma análise. Basicamente decidiram efetuar um envio manual dos dados gerados, através da plataforma web DCF disponibilizada pela EFSA.

2.1.3 Solução desenvolvida pela Suécia

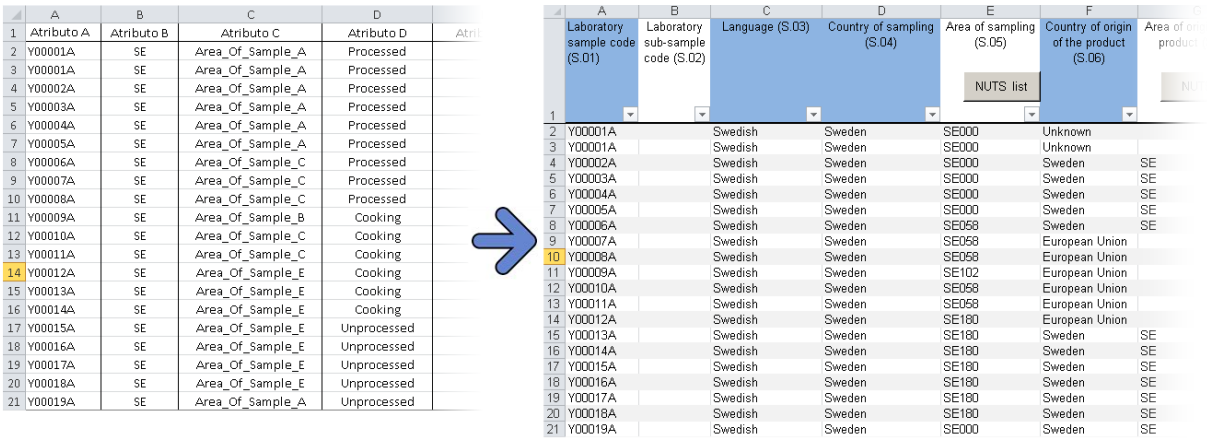
Para finalizar o estudo de soluções apresentadas por outros países, analisaremos o último documento que descreve a solução desenvolvida pela Suécia. A semelhança dos outros países, a Suécia tem definido os mesmos objetivos para o projeto. Para não estar a focar nos mesmos pontos já referidos e analisados para as outras entidades, irei apenas descrever aqueles que ainda não foram expostos ou são de grande importância que faça sentido menciona-los novamente.

Mais uma vez, trata-se de uma entidade que já possuía uma base de dados nacional e que decidiu para o projeto, criar uma extensão desta base de dados iria estar destinada a armazenar apenas dados que estivessem relacionados com a norma SSD.

O processo de transformação de dados para o formato SSD é o que distingue esta solução das outras todas. Começam por referir que os dados no seu estado natural estão arquivados em ficheiros Excel. Os ficheiros são então carregados para uma aplicação desenvolvida por eles sobre a Framework .NET[5] com a linguagem de programação C#[6]. Esta aplicação faz uma conversão dos dados que estão em um formato Excel desconhecido, para um formato conhecido pela EFSA. Para que seja mais fácil de se perceber este tipo de conversão, consideremos a imagem abaixo.

[5] http://msdn.microsoft.com/en-EN/vstudio/aa496123

[6] http://msdn.microsoft.com/en-us/library/67ef8sbd.aspx



	A	B	C	D
1	Atributo A	Atributo B	Atributo C	Atributo D
2	Y00001A	SE	Area_Of_Sample_A	Processed
3	Y00001A	SE	Area_Of_Sample_A	Processed
4	Y00002A	SE	Area_Of_Sample_A	Processed
5	Y00003A	SE	Area_Of_Sample_A	Processed
6	Y00004A	SE	Area_Of_Sample_A	Processed
7	Y00005A	SE	Area_Of_Sample_A	Processed
8	Y00006A	SE	Area_Of_Sample_C	Processed
9	Y00007A	SE	Area_Of_Sample_C	Processed
10	Y00008A	SE	Area_Of_Sample_C	Processed
11	Y00009A	SE	Area_Of_Sample_B	Cooking
12	Y00010A	SE	Area_Of_Sample_C	Cooking
13	Y00011A	SE	Area_Of_Sample_C	Cooking
14	Y00012A	SE	Area_Of_Sample_E	Cooking
15	Y00013A	SE	Area_Of_Sample_E	Cooking
16	Y00014A	SE	Area_Of_Sample_E	Cooking
17	Y00015A	SE	Area_Of_Sample_E	Unprocessed
18	Y00016A	SE	Area_Of_Sample_E	Unprocessed
19	Y00017A	SE	Area_Of_Sample_E	Unprocessed
20	Y00018A	SE	Area_Of_Sample_E	Unprocessed
21	Y00019A	SE	Area_Of_Sample_A	Unprocessed

	A	B	C	D	E	F
1	Laboratory sample code (S.01)	Laboratory sub-sample code (S.02)	Language (S.03)	Country of sampling (S.04)	Area of sampling (S.05) NUTS list	Country of origin of the product (S.06)
2	Y00001A		Swedish	Sweden	SE000	Unknown
3	Y00001A		Swedish	Sweden	SE000	Unknown
4	Y00002A		Swedish	Sweden	SE000	Sweden SE
5	Y00003A		Swedish	Sweden	SE000	Sweden SE
6	Y00004A		Swedish	Sweden	SE000	Sweden SE
7	Y00005A		Swedish	Sweden	SE000	Sweden SE
8	Y00006A		Swedish	Sweden	SE058	Sweden SE
9	Y00007A		Swedish	Sweden	SE058	European Union
10	Y00008A		Swedish	Sweden	SE058	European Union
11	Y00009A		Swedish	Sweden	SE102	European Union
12	Y00010A		Swedish	Sweden	SE058	European Union
13	Y00011A		Swedish	Sweden	SE058	European Union
14	Y00012A		Swedish	Sweden	SE180	European Union
15	Y00013A		Swedish	Sweden	SE180	Sweden SE
16	Y00014A		Swedish	Sweden	SE180	Sweden SE
17	Y00015A		Swedish	Sweden	SE180	Sweden SE
18	Y00016A		Swedish	Sweden	SE180	Sweden SE
19	Y00017A		Swedish	Sweden	SE180	Sweden SE
20	Y00018A		Swedish	Sweden	SE180	Sweden SE
21	Y00019A		Swedish	Sweden	SE000	Sweden SE

Figura 4. Conversão para o formato Excel disponibilizado pela EFSA.

O ficheiro Excel que se encontra no lado esquerdo da imagem é disponibilizado pela EFSA sem estar preenchido, juntamente com os restantes ficheiros que descrevem o padrão SSD.

Ao estudar este processo, acabamos por não concordar com a sua última parte. Dados que estejam armazenados em ficheiros Excel, o que é algo bastante comum neste tipo de projeto, terão de ser extraídos e transformados para o padrão SSD, portanto a primeira parte de extração efetuada pela entidade sueca faz todo o sentido existir, por ser uma etapa que consideramos fundamental. Por outro lado, não se consegue perceber o que levou a que os dados agora extraídos fossem escritos novamente para um ficheiro. A única razão que consegue-se encontrar e que justifique a decisão tomada é a de quererem manter sempre duas versões dos dados, uma versão em estado de ficheiro Excel e a outra versão mantida na base de dados.

Para o projeto desenvolvido para o INSA decidimos extrair os dados do ficheiros e escrevê-los diretamente para a base de dados, não houve nenhuma necessidade de guarda-los em ficheiro após a sua transformação para SSD. O processo tornou-se muito mais simples e eliminou a possibilidade de perda de informações que poderiam originar com transformações sucessivas.

Para concluir a análise ao relatório da Suécia terminemos então com o estudo do procedimento com que utilizam para gerar o ficheiro XML. Incluído com o pacote de ficheiros enviados a cada entidade responsável pelo projeto está presente um ficheiro do tipo XML Schema. Este ficheiro tem como principal objetivo ser utilizado para validar a formação e estrutura do ficheiro XML a ser enviado à EFSA. No entanto a Suécia propôs utiliza-lo para

outro fim que conseguisse ajudar também a gerar o ficheiro XML. O que fizeram foi *reverse engineering* ao XML Schema, criando um ficheiro XML com apenas os vários nós sem estarem preenchidos com informação e de seguida utilizaram os dados que estão na base de dados para preencher o esqueleto do ficheiro XML.

Esta técnica traz três grandes vantagens sendo elas a sua grande eficácia, a redução do tempo que se iria consumir ao tentar seguir as regras presentes manual do SSD enviado pela EFSA e passa-las para código, e finalmente também permite que o projeto em si evoluía juntamente com as várias versões do padrão SSD sendo apenas necessário atualizar o ficheiro com o XML Schema para se conseguir gerar a última versão do ficheiro XML necessitado.

Em síntese, para esta solução apresentada pela Suécia conseguimos retirar excelentes ideias quanto a transformação dos dados inalterados contidos em ficheiros Excel para o padrão SSD, e a formação incomum do ficheiro XML, que por sua vez acabou por ser uma grande ajuda no projeto desenvolvido para o INSA.

2.2 Mapeamento Assistido/Automático

Após efetuar o estudo dos vários relatórios elaborados pelas diversas entidades europeias e retirar conclusões dos mesmos, deu-se início a um novo processo de investigação que de seguida, neste subtópico, irá ser explicado.

O problema referente à quantidade de esforço excessivo realizado por parte dos recursos humanos foi identificado na maioria dos relatórios submetidos pelas várias entidades, senão todas elas. Este elevado esforço concentrou-se em volta de uma tarefa em particular, tarefa esta que consistia em criar as tabelas de mapeamento.

Como foi explicado anteriormente de forma sintetizada, a tarefa de se criar uma tabela de mapeamento (ou tabela de tradução) é uma operação bastante onerosa, em que alguém especializado na área da saúde terá que ir manualmente a cada um dos registos, onde o número de registos poderá ultrapassar facilmente as dezenas de milhar, e examina-los de forma a saber que código SSD é que um certo atributo deverá apresentar como valor. O facto de ter-se de trabalhar um elevado número de entradas já dificulta bastante o processo, mas

para além disto existem ainda 73 diferentes atributos pertencentes ao padrão SSD que deverão ser preenchidos. Portanto, se tomarmos como exemplo um ficheiro Excel com 3000 entradas (ou linhas) e com 20 atributos (ou colunas), para que este ficheiro seja mapeado corretamente, 73 atributos terão de ser preenchidos para cada uma das 3000 entradas. É de fácil constatação que esta tarefa não poderá ser realizada por uma só pessoa e que representa custos significativos neste tipo de projetos.

Identificado assim o problema, conseguimos encontrar uma solução. A solução seria desenvolver um módulo para o projeto e este módulo seria constituído por uma ou mais ferramentas a serem desenvolvidas que permitissem auxiliar o técnico encarregado, no mapeamento dos dados.

Foi assim necessário proceder ao levantamento do estado da arte na área da Integração Semântica, com particular ênfase para as normas, tecnologias e ferramentas de tratamento léxico (*lexers*), sintático (*parsers*) e semântico (tecnologias da Web Semântica), bem como proceder à seleção do conjunto de normas, tecnologias e ferramentas mais adequadas à resolução do problema formulado. Nas secções seguintes da presente dissertação serão abordados e apresentados com detalhe estes aspetos do trabalho.

Projeto de Software

Durante este capítulo serão apresentadas as decisões e procedimentos tomados relacionados com o planeamento do projeto. Este projeto foi realizado em colaboração com a equipa de desenvolvimento do INSA, o que levou que fossem adotados métodos adequados para a implementação da solução de modo a que a comunicação entre as diferentes partes fosse feita de forma harmoniosa.

3.1 Gestão de projeto

Uma pequena equipa do INSA foi estabelecida para orientar o projeto a ser desenvolvimento pelos dois bolseiros selecionados, os estudantes Sidney Tomé e João Pereira. A primeira tarefa em que concordou-se realizar foi o planeamento do projeto. Este planeamento iria permitir perceber a realidade do projeto, os seus domínios, os diferentes percursos que irá tomar e os objetivos a alcançar, bem como também ajudará a diminuir a quantidade de riscos envolvidos nas diferentes atividades.

O planeamento provou ser bastante útil para manter um certo nível de controlo sobre o projeto, e este grau de estabilidade foi verificado pelas duas equipas. A outra vantagem que também se verificou foi no nível superior de eficácia com que se terminava e iniciava um processo, visto que era sempre conhecida a duração da tarefa corrente e qual seria a próxima a ser realizada, mantendo um fluxo de trabalho bastante estável e constante. Ambas as equipas tinham sempre conhecimento do estado do projeto a qualquer altura do seu desenvolvimento, no entanto isto não se deve apenas ao planeamento mas também é resultado das várias reuniões realizadas ao longo da produção do projeto. O tópico que descreve as reuniões feitas entre o INSA será descrito mais a frente no ponto da Gestão do Projeto.

ID	Task Name	Duration	Start
1	**Definição**	**7 days**	**Tue 17-07-12**
2	**Definição de Requisitos**	**4 days**	**Tue 17-07-12**
3	Elaborar documento de Requisitos Funcionais	3 days	Tue 17-07-12
4	Elaborar documento de Requisitos Não-Funcionais	3 days	Tue 17-07-12
5	Elaborar documento de Requisitos Desenvolvimer	3 days	Tue 17-07-12
6	Estabelecer as prioridades para os Requisitos	1 day	Fri 20-07-12
7	**Definição de Casos de Uso**	**4 days**	**Fri 20-07-12**
8	Elaborar documento com os Casos de Uso	4 days	Fri 20-07-12
9	**Planeamento do Projeto**	**7 days**	**Tue 17-07-12**
10	Estabeler os objetivos do Projeto	3 days	Tue 17-07-12
11	Definir as deadlines para as tarefas	2 days	Fri 20-07-12
12	Elaborar documento de Planeamento do Projeto	7 days	Tue 17-07-12
13	**Definição de Metodologia de Desenvolvimento**	2 days	Tue 24-07-12
14	**Requisitar Aprovação**	0 days	Thu 26-07-12
15	**Investigação**	**21 days**	**Thu 26-07-12**
16	**Análise de Trabalhos/Projetos existentes**	**12 days**	**Thu 26-07-12**
17	Análise de documentações	12 days	Thu 26-07-12
18	Estudo de projetos já desenvolvidos	12 days	Thu 26-07-12
19	**Investigação por soluções para problema similar**	**5 days**	**Mon 13-08-1**
20	Pesquisa por técnicas, métodos e algoritmos assoc	5 days	Mon 13-08-1
21	**Documentação**	**4 days**	**Mon 20-08-1**
22	Elaborar documento com resultados encontrados	4 days	Mon 20-08-1
23	Elaborar documento com conclusões obtidas	4 days	Mon 20-08-1
24	**Desenho**	**36 days**	**Fri 24-08-12**
25	**Funcional**	**13 days**	**Fri 24-08-12**
26	Identificar interfaces necessárias	6 days	Fri 24-08-12
27	Identificar métodos de segurança	6 days	Fri 24-08-12
28	Identificar processos	5 days	Mon 03-09-1
29	Identificar serviços	5 days	Mon 03-09-1
30	Especificar tipos de acessos	2 days	Mon 10-09-1
31	**Técnico**	**33 days**	**Wed 29-08-1**
32	Desenhar Base de Dados	23 days	Wed 29-08-1
33	Desenhar Interfaces de utilizador	11 days	Wed 29-08-1
34	Definir Especificações da aplicação	12 days	Thu 13-09-12
35	Definir ferramentas adicionais	10 days	Mon 17-09-1
36	Preparar/Estruturar dados	5 days	Mon 01-10-1
37	Planificar a fase de testes	5 days	Mon 08-10-1
38	Requisitar aprovação		

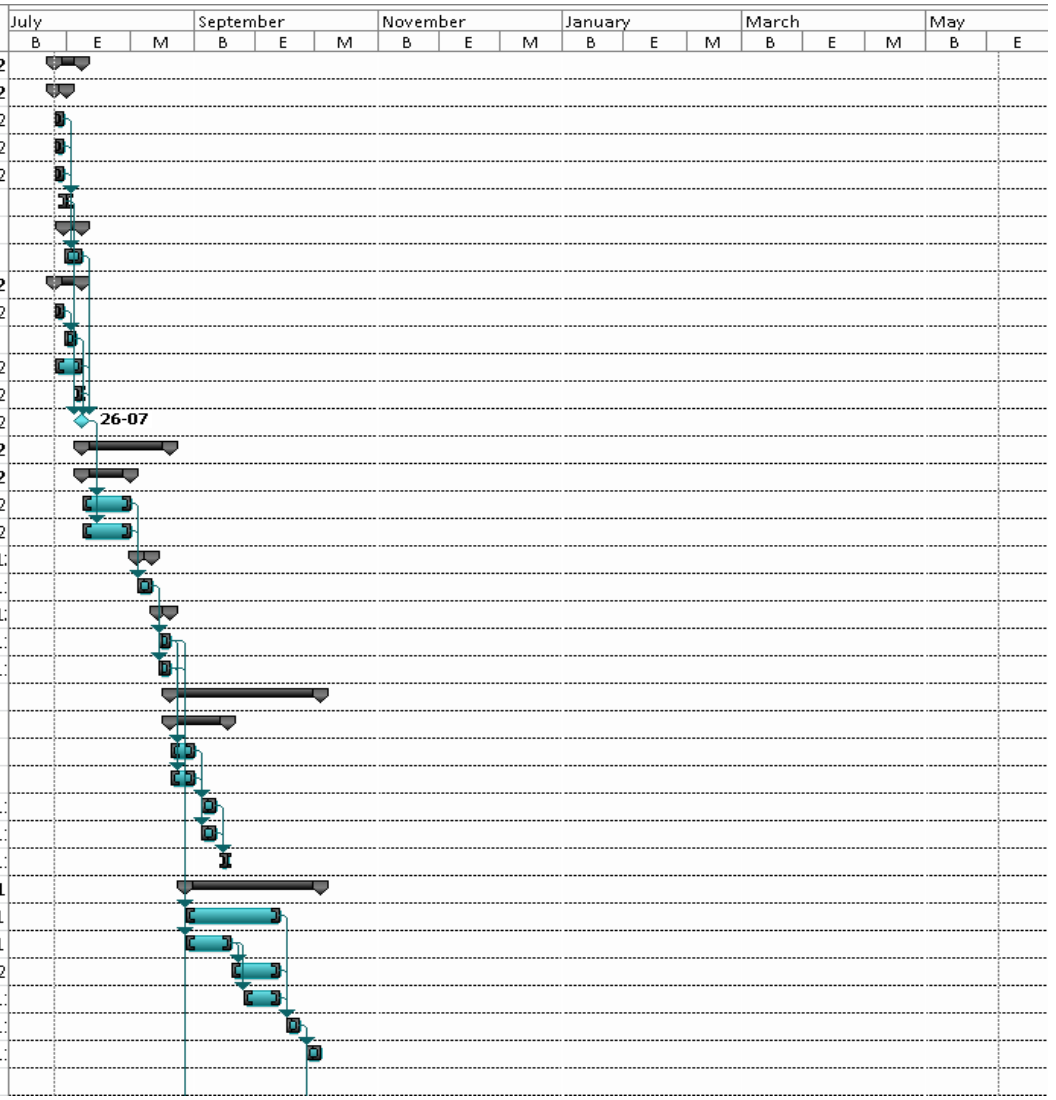


Figura 5. Gráfico de Gantt que ilustra o planeamento do projeto

39	**Implementação**	**158 days**	**Wed 29-08-1**
40	**Programação**	**158 days**	**Wed 29-08-1**
41	Implementar interfaces de utilizador	158 days	Wed 29-08-1
42	Implementar o backoffice	130 days	Mon 08-10-1
43	Desenvolver Sistema Base	158 days	Wed 29-08-1
44	Construir o sistema Base de Dados	150 days	Mon 10-09-1
45	Desenvolver ferramentas de tratamento de dados	140 days	Mon 24-09-1
46	Criar protótipo	43 days	Wed 29-08-1
47	Criar testes unitários	5 days	Mon 29-10-1
48	Criar testes do sistema	7 days	Mon 05-11-1
49	Gerar documentação	12 days	Mon 29-10-1
50	Requisitar aprovação	0 days	Wed 14-11-1
51	**Deployment**	**6 days**	**Fri 16-11-12**
52	Configuração do cenário	3 days	Fri 16-11-12
53	Colocar protótipo em funcionamento	3 days	Wed 21-11-1
54	**Testes**	**24 days**	**Mon 26-11-1**
55	Finalizar plano de testes	3 days	Mon 26-11-1
56	Criar dados de teste	8,5 days	Thu 29-11-12
57	Analisar o fluxo dos dados	5 days	Wed 12-12-1
58	Realizar testes	7 days	Wed 19-12-1
59	Entrada em funcionamento	0 days	Fri 28-12-12
60	**Operação**	**35 days**	**Mon 08-04-1**
61	Operar o sistema	5 days	Mon 08-04-1
62	**Revisão**	**7 days**	**Mon 22-04-1**
63	Estabelecer plano de revisão	4 days	Mon 22-04-1
64	Rever desempenho e funcionamento	3 days	Fri 26-04-13
65	Rever desempenho e funcionamento	7 days	Wed 01-05-1
66	Obter aprovação	0 days	Fri 10-05-13
67	**Conclusões do projeto**	**10 days**	**Mon 13-05-1**
68	Elaborar documento com trabalho futuro	5 days	Mon 13-05-1
69	Elaborar documento com conclusões	5 days	Mon 20-05-1

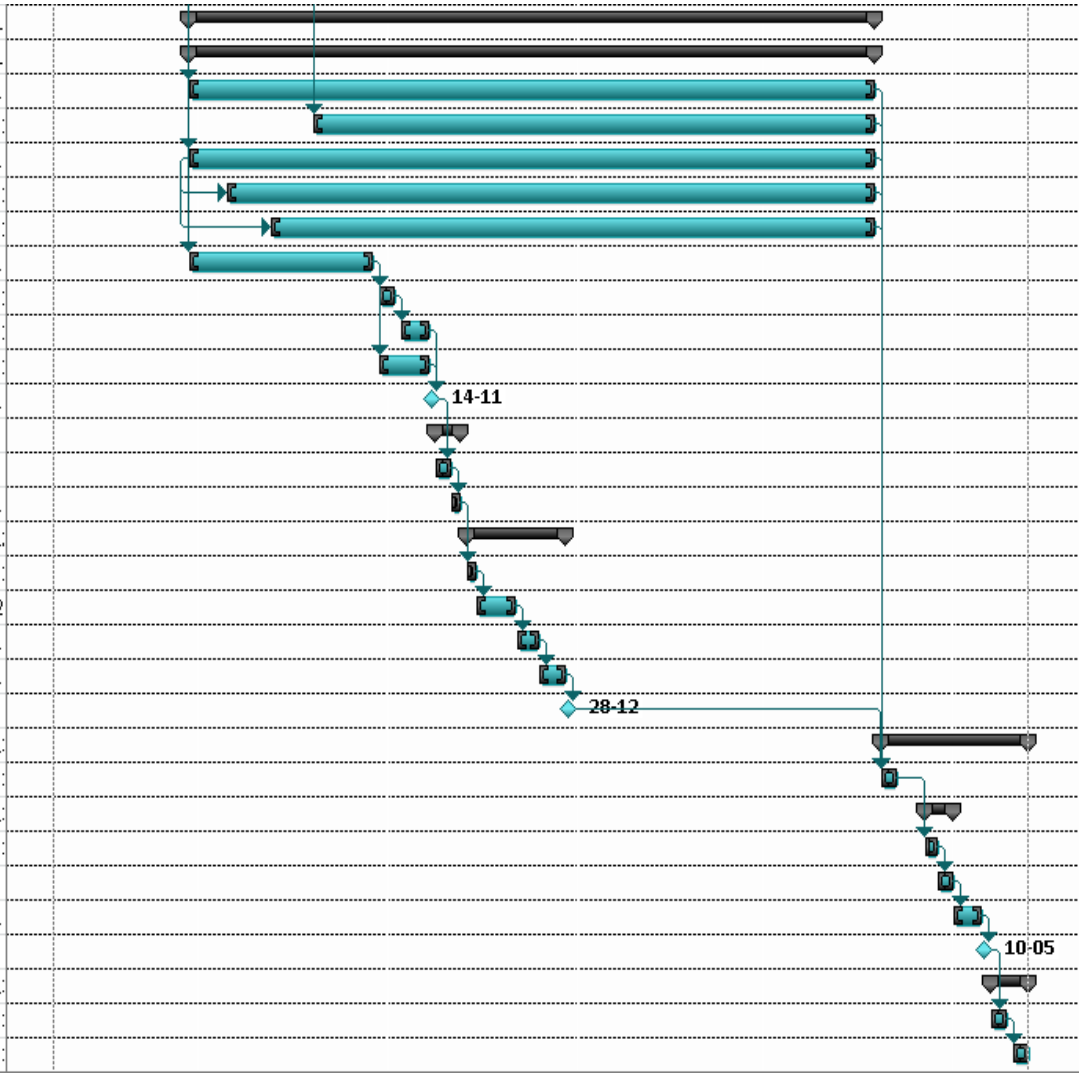


Figura 6. Gráfico de Gantt que ilustra o planeamento do projeto (cont.)

O projeto é constituído por 69 tarefas distintas e 5 tarefas principais, como é possível observar-se através da Figura 5 e Figura 6. O planeamento foi o documento criado que mais alterações sofreu ao longo do projeto, e o que contribuiu para a sua constante modificação foi exatamente dois seguintes fatores: a necessidade de se ter de criar um protótipo para testar a plataforma DCF da EFSA; e a constante criação de novas ferramentas a medida que o projeto evoluía.

A EFSA notificou que os países que estivessem envolvidos na proposta CFP/EFSA/DATEX/2011/01 teriam até a data de 14 de Novembro de 2012 para submeterem dados de teste através da plataforma web, o DCF. Para os estados membros que já possuíssem algo produzido até a data, estes não iriam experienciar qualquer tipo de problemas. Contudo, o estado membro português tinha aderido à proposta meses antes da data de testes com a EFSA, isto levou a que o INSA tivesse de desenvolver uma solução em estado de protótipo até a data estabelecida.

O desenvolvimento do protótipo em curto espaço de tempo implicou várias alterações no planeamento pelo facto de não se possuir, em dada altura, uma visão da envergadura total do projeto. Sabia-se quais seriam os módulos que teriam de ser desenvolvidos para concluir o projeto mas não se sabia ao certo quais destes módulos teriam de estar concluídos, ou pelo menos funcionais, até a data do protótipo, o que resultou numa constante alteração de foco sobre as diferentes tarefas e, consequentemente, na alteração da data de realização e conclusão das outras tarefas afetadas.

O segundo fator que também contribui para a constante alteração do planeamento do projeto foi, basicamente, a tentativa de satisfazer ao máximo as necessidades do INSA, ou seja, após concluir a fase de testes com o protótipo criado, começaram a surgir propostas de novas ferramentas para ajudar nas tarefas mais complicadas.

Como foi brevemente debatido no capítulo de Revisão da Literatura, existem tarefas que requerem muito mais esforço que outras, e para estas tarefas nasceram ideias de ferramentas que poderiam auxiliar na sua finalização. Grandes partes destas ferramentas acabaram por ser implementadas mas com a desvantagem de ter de alterar o curso do projeto sempre que estas eram expostas.

A comunicação entre os bolseiros e a equipa do INSA foi feita de forma constante ao longo do processo de produção. Diferentes meios foram utilizados, incluindo comunicação por videochamadas, *emails* e múltiplas reuniões presenciais.

As reuniões foram realizadas com uma frequência acordada no início do projeto, e foram concretizadas em média de 1 em 1 mês. Nestas reuniões era discutido o progresso do projeto, iniciando-se pela apresentação prática das funcionalidades já elaboradas, de seguida era dado início a uma sessão de avaliação do que foi apresentado, mencionando os pontos fortes e os pontos fracos. Para finalizar a reunião era feito um planeamento das tarefas seguintes a serem concretizadas, baseado no que foi dito e apreciado durante a reunião.

3.2 Modelo de processos de desenvolvimento de *software*

Apesar de existir uma grande equipa envolvida na elaboração do projeto atribuído ao INSA, o seu desenvolvimento, em termos do projeto informático, foi efetuado por dois elementos.

Uma metodologia de desenvolvimento teria de ser adotada para um cenário em que se possui um pequeno grupo de pessoas a produzir código, mas ao mesmo tempo teria também de se enquadrar às necessidades da EFSA e do INSA, necessidades estas que passam pela criação de vários tipos de documentações e reuniões que ocorrem regularmente.

Decidiu-se então utilizar um misto das seguintes metodologias de desenvolvimento: ICONIX, Extreme Programming, Pair Programming e SCRUM.

A junção das metodologias SCRUM, Extreme Programming e Pair Programming é conhecida por ser uma combinação considerada harmoniosa [6] [7]. Estas três metodologias complementam-se umas as outras, da forma que será explicado de seguida, e o facto de também fazerem parte dos mesmo grupo de desenvolvimento ágil de *software* ajuda a serem postas em prática.

A metodologia SCRUM pretende focar-se mais na gestão e nas práticas de organização do projeto, auxiliando no planeamento das tarefas futuras. Com esta metodologia queríamos obter um plano de trabalho devidamente detalhado e evitar ao máximo ficar sem o

conhecimento de quais as próximas tarefas a serem realizadas. Para nós, o estabelecimento de várias datas de entregas de múltiplas interações que constituíam o processo de desenvolvimento era a estratégia ideal para se conseguir concluir o projeto da melhor forma possível, e com o SCRUM foi possível seguir esta estratégia.

De modo a dar a conhecer a forma com que o SCRUM foi utilizado neste projeto, irá ser descrito de forma sintetizada alguns métodos de utilização da metodologia.

Os papéis principais existentes na metodologia SCRUM foram devidamente distribuídos pelos vários elementos da equipa do INSA, e na tabela abaixo é apresentada a atribuição efetuada para a equipa responsável pelo desenvolvimento informático do projeto.

Tabela 3. Atribuição dos vários papéis da metodologia SCRUM à equipa

Elemento	Papel
Eng. João Galhardo	Scrum Master
Sidney Tomé	Team
João Pereira	Team
INSA	Product Owner

Tal como foi explicado no tópico de Planeamento do Projeto, frequentemente eram realizadas as reuniões com toda a equipa de desenvolvimento do INSA. Estas reuniões são realmente Sprints que o projeto possuiu, tendo a duração de 1 mês. Da mesma forma que é especificado na metodologia SCRUM, nestes Sprints eram sempre planeados quais as próximas funcionalidades a serem implementadas, e após concluído um Sprint, era feita a apresentação do resultado do Sprint anterior.

Devido ao tamanho reduzido da equipa responsável pela parte informática do projeto, constituída por apenas 2 elementos, foi bastante simples de se executar e cumprir com os Scrums diários. Basicamente, todos os dias, durante uma Sprint, seria realizado um diálogo inicial com os dois elementos, tendo uma duração em média de 10 minutos, sobre o que teria sido concluído no dia anterior e o que deverá ser implementado naquele dia.

Sempre que houvesse um certo tipo de complicação com alguma funcionalidade definida para a Sprint, o Scrum Master seria contactado através de *emails*, de forma a chegar-se a uma

solução que impedisse que a Sprint realizada naquele momento fosse comprometida.

A metodologia Extreme Programming foi utilizada com o propósito de se tentar obter resultados visíveis com uma boa qualidade, e quanto mais rapidamente conseguisse obter algo tangível, mais rapidamente conseguia-se testar para que estivesse conforme o que foi acordado.

O fator de comunicação descrito na metodologia XP nunca foi um problema no cenário com que foi realizado o projeto, já que em um trabalho feito por duas pessoas teremos apenas um canal de comunicação e, ao contrário do que acontece nas grandes equipas de desenvolvimento, era possível criar um ambiente em que existe a certeza de que qualquer tipo de informação iria chegar a cada uma dos elementos.

Também devido ao tamanho da equipa de desenvolvimento, decidiu-se aplicar a técnica de desenvolvimento ágil Pair Programming. Esta técnica não foi utilizada de uma forma persistente ao longo do desenvolvimento do projeto pela razão de que ao manter um elemento do grupo com a função de Observador para uma tarefa simples seria, na nossa opinião, um péssimo investimento do tempo, no entanto foi realizada uma pequena investigação sobre o assunto e de facto existem estudos que mostram que Pair Programming torna o desenvolvimento 15% mais lento[7].

Para as tarefas mais simples, aplicou-se então um desenvolvimento em paralelo, onde cada elemento focava-se em uma tarefa distinta, auxiliando assim na conclusão de mais tarefas em menos tempo. No entanto, Pair Programming foi utilizado para as funcionalidades mais complexas, visto que existem estudos que comprovam que programação utilizando esta técnica aumenta em cerca de 70% à 80% a produção de código sem erros [8], logo a utilização de Pair Programming faria mais sentido para as tarefas mais críticas e mais complicadas [9].

A combinação das duas metodologias acima descritas permitiu fazer a junção dos benefícios da gestão do projeto especificada pela SCRUM, com a qualidade superior de produção de código descrita pela Extreme Programming. Como consequência, obtivemos planos de

[7] http://www.economist.com/node/779429

trabalho que foram bastantes produtivos e ao mesmo tempo era sempre produzido resultados funcionais que estariam prontos para apresentação e para a avaliação.

Para finalizar este tópico resta explicar a aplicação da metodologia ICONIX no projeto. Optou-se por descrever primeiramente a utilização da metodologia SCRUM e XP pelo facto de serem aquelas que estão mais próximas do resultado produzido pelo desenvolvimento. Contudo existe um processo bastante importante existente na maioria dos projetos, o processo de planeamento do desenvolvimento de uma solução. Este planeamento inclui a análise de requisitos, análise e desenho preliminar do projeto, desenho detalhado do projeto, e por fim a sua implementação. É possível visualizar a existência dos 4 processos que constituem a metodologia ICONIX na WBS que poderá ser consultada através do apêndice A.

A metodologia permitiu a criação de um conjunto de documentos que descreviam com grande detalhe o processo de criação da solução concebida pelo estado membro português. Esta documentação foi enviada à EFSA com o objetivo de auxiliar os outros países envolvidos na proposta, para a criação das suas próprias soluções.

3.3 Análise de requisitos

De seguida serão apresentados os resultados do processo elaborado de levantamento de requisitos. Este processo foi fundamental para dar a conhecer aos envolvidos, os vários aspetos da plataforma PT.ON.DATA que foi desenvolvida. Tratou-se então de um processo interativo em que sofreu várias alterações de acordo com o que era discutido nas reuniões (ou no início de cada Sprint).

3.3.1 Requisitos Funcionais

O levantamento dos requisitos funcionais foi o processo que melhor permitiu dar a conhecer as diversas funcionalidades que se pretendia obter com o desenvolvimento deste projeto. A criação de um documento que descreveu cada um destes requisitos auxiliou na melhor compreensão do tipo de produto que se queria obter, tanto para os elementos externos ao desenvolvimento informático como também para os próprios programadores. Abaixo

encontram-se então ilustrados em uma tabela alguns dos requisitos considerados mais importantes.

Tabela 4. Tabela com uma seleção dos Requisitos Funcionais mais relevantes.

ID	Requisitos Funcionais	Prioridade	Piloto
1	O Sistema deve conter quatro tipos de perfis de utilização: ADMINISTRADOR [1], SUPERVISOR [2], RESPONSAVEL_AC [3] e VISITANTE [4].	Obrigatório	
2	Os utilizadores com perfil ADMINISTRADOR têm acesso a todas as funcionalidades do Sistema (*upload* de ficheiros, mapeamento manual de ficheiros, visualização e alteração dos dados existentes, envio dos ficheiros XML à EFSA, configuração do Sistema, gestão de utilizadores, etc.).	Obrigatório	
3	Os utilizadores com perfil SUPERVISOR têm acesso às mesmas funcionalidades do ADMINISTRADOR, no entanto não possuem privilégios para realizar configurações no Sistema e efetuar a gestão de utilizadores. Este utilizador tem como principal função supervisionar toda a atividade do Sistema, realizar o mapeamento dos campos dos ficheiros e submeter os ficheiros XML à EFSA.	Obrigatório	
4	Os utilizadores com perfil RESPONSAVEL_AC têm acesso às funcionalidades básicas do Sistema (*upload* de ficheiros, visualização e alteração dos dados existentes). Este utilizador é responsável pelo *upload* de ficheiros afetos à respetiva AC.	Obrigatório	
5	Os utilizadores com perfil VISITANTE apenas possuem privilégios de visualização de dados e estatísticas no Sistema.	Obrigatório	
6	O Sistema deve permitir ao utilizador [1] registar novos utilizadores.	Obrigatório	
7	O Sistema deve permitir ao utilizador [1] realizar a gestão de utilizadores (atualização, eliminação, alteração de perfis, ativação	Obrigatório	

	e desativação de contas).		
8	O Sistema deverá incluir um procedimento de autenticação de utilizadores, onde cada utilizador se deve identificar através de um *username* e *password*. Apenas os utilizadores autorizados poderão utilizar a plataforma.	Obrigatório	
10	O Sistema deverá apresentar um formulário para que o utilizador possa adicionar informação das recolhas e análises diretamente no Sistema.	Obrigatório	
11	O Sistema deverá validar automaticamente todas as entradas adicionadas através do formulário, guardando a informação na tabela SSD da base de dados.	Obrigatório	
12	O Sistema deverá permitir ao utilizador exportar os dados introduzidos no formulário para o formato Excel.	Obrigatório	
13	O Sistema deverá permitir ao utilizador adicionar novos campos no formulário.	Desejável	
14	O Sistema deve possuir um diretório central que contêm todos os ficheiros enviados pelas AC´s.	Obrigatório	SIM
15	O Sistema deve permitir aos utilizadores [1], [2] e [3] carregar ficheiros de vários formatos (XLS, CSV, TXT, etc.) e guardá-los no diretório central.	Obrigatório	
16	O Sistema deve realizar um pré-validação dos ficheiros (extensão inválida, ficheiros vazios, campos obrigatórios em falta, etc.) antes de realizar o *upload.*	Obrigatório	SIM
17	O Sistema deve notificar o utilizador se existirem ficheiros inválidos.	Obrigatório	SIM
20	O Sistema deve permitir que o utilizador [3] possa remover ficheiros que tenha adicionado e que ainda não tenham sido mapeados para o SSD.	Obrigatório	

21	O Sistema deve permitir notificar o utilizador [2] da existência de ficheiros para mapear.	Obrigatório	
22	O Sistema deve efetuar a leitura da informação dos ficheiros existentes no diretório central.	Obrigatório	SIM
23	O Sistema deve permitir mapear automaticamente os atributos (campos) dos ficheiros para os atributos definidos pelo SSD.	Obrigatório	SIM
24	O Sistema deve verificar a validade do SSD e a data das amostragens de modo a mapear os atributos para a versão correta do SSD.	Obrigatório	
25	O Sistema deve permitir ao utilizador [2] realizar uma pré-validação do mapeamento, podendo visualizar e alterar a associação automática efetuada pelo Sistema.	Obrigatório	
26	O Sistema deve permitir alertar o utilizador sempre que não consiga efetuar corretamente o mapeamento dos atributos.	Obrigatório	
27	O Sistema deve apresentar aos utilizadores [1] e [2] uma tabela com todos os ficheiros existentes no diretório central e o respetivo estado (não mapeado, mapeamento automático, mapeamento manual, número de erros).	Obrigatório	
32	O Sistema deve guardar de modo persistente o estado do mapeamento de cada ficheiro existente no diretório central.	Obrigatório	
33	O Sistema deve notificar o utilizador [3] acerca dos ficheiros que estejam mapeados e que necessitem de validação da informação presente nos vários campos.	Obrigatório	
34	O Sistema deverá efetuar a validação do conteúdo dos campos apenas aos ficheiros que tenham sido corretamente mapeados.	Obrigatório	SIM
35	O Sistema deve invalidar entradas repetidas (que já tenham sido adicionadas).	Obrigatório	

36	Caso existam entradas repetidas, o Sistema deverá apresentar uma janela a alertar o utilizador, para que este decida qual das entradas pretende adicionar e qual deverá ser descartada.	Obrigatório	
37	O Sistema deve permitir ler e validar o conteúdo de cada campo de cada entrada de acordo com as normas definidas pelo SSD (ex. tipo de dados, formato da data, etc.).	Obrigatório	SIM
38	O Sistema deve permitir alertar o utilizador sempre que existirem campos errados (ex. tipo de dados inválido).	Obrigatório	
39	O Sistema deve mapear o conteúdo de cada campo de linguagem controlada, para o correspondente valor definido pelo SSD.	Obrigatório	
40	O Sistema deve guardar na tabela "quarentena" da base de dados, todas as entradas inválidas.	Obrigatório	
41	O Sistema deve guardar na base de dados SSD todas as entradas válidas.	Obrigatório	SIM
42	O Sistema deve apresentar uma tabela com todas as entradas inválidas (entradas em quarentena).	Obrigatório	
43	O Sistema deve permitir ao utilizador alterar o valor dos campos das entradas inválidas, de modo a corrigi-las.	Obrigatório	
44	O Sistema deve remover automaticamente uma entrada da tabela "quarentena" da base de dados e colocá-la na tabela "SSD" da base de dados quando esta estiver válida.	Obrigatório	
45	O Sistema deve apresentar a tabela SSD com todas as entradas válidas e prontas para serem reportadas à EFSA.	Obrigatório	
46	O Sistema deve permitir que o utilizador "coloque" entradas da tabela SSD em quarentena, retirando a respetiva entrada da tabela "SSD" da base de dados e colocando-a na tabela "quarentena" da base de dados.	Obrigatório	

47	O Sistema deve permitir apresentar cada entrada das tabelas (tabela SSD e tabela quarentena) numa vista de detalhe (para facilitar a visualização e a alteração dos dados da entrada).	Obrigatório	
48	O Sistema deve permitir criar um ficheiro em formato XML com a informação presente na tabela "SSD" da base de dados,	Obrigatório	SIM
49	O Sistema deve validar o ficheiro XML de acordo com as normas definidas pelo SSD, antes de ser remetido à entidade responsável.	Obrigatório	SIM
50	O Sistema deve permitir aos utilizadores [1] e [2] enviar ficheiros XML à entidade responsável.	Obrigatório	
51	O Sistema deve permitir guardar em disco o ficheiro XML.	Obrigatório	SIM
52	O Sistema deve apresentar ao utilizador uma interface para enviar ficheiros XML à entidade responsável.	Obrigatório	SIM
53	O Sistema deve permitir que o utilizador possa escolher para que entidade pretende enviar os dados em XML.	Obrigatório	
54	O Sistema deve notificar o utilizador do sucesso ou insucesso do envio dos ficheiros XML à entidade responsável.	Obrigatório	SIM
55	O Sistema deve guardar um histórico com todos os ficheiros enviados à entidade responsável.	Obrigatório	
56	O Sistema deve alterar o estado para "remetido", adicionar o nome da entidade e colocar a data de envio em todas as entradas da base de dados que tenham sido remetidas à entidade responsável através do ficheiro XML.	Obrigatório	

Quanto à tabela apresentada dos requisitos funcionais, falta realçar a existência da terceira coluna com o título de Piloto. Esta coluna foi criada com o propósito de dar a conhecer quais os requisitos listados que deverão ser implementados para a criação do protótipo funcional. Estes requisitos representam funcionalidades indispensáveis que auxiliam na conclusão dos vários testes definidos pela EFSA.

3.3.2 Requisitos Não-Funcionais

ID	Requisitos Não-Funcionais	Prioridade	Piloto
1	O Sistema deverá possuir uma interface web de controlo e gestão.	Obrigatório	
2	O Sistema deverá impedir utilizadores não autorizados acederem à plataforma web.	Obrigatório	
3	O Sistema deverá ter uma interface simples e funcional.	Obrigatório	
4	O Sistema deverá permitir a persistência dos dados e configurações efetuadas.	Obrigatório	
5	O Sistema deverá possibilitar a sua utilização em qualquer plataforma.	Obrigatório	

3.3.3 Requisitos de Desenvolvimento

ID	Requisitos de desenvolvimento	Prioridade	Piloto
1	O projeto será desenvolvido sobre a tecnologia Microsoft .NET utilizando o IDE Microsoft Visual Studio 2010.	Obrigatório	
2	As base de dados serão construídas em Microsoft SQL Server 2008 R2.	Obrigatório	
3	Será utilizado o software de controlo de versões Microsoft Team Foundation Server.	Obrigatório	
4	O projeto será realizado com recurso às seguintes metodologias de desenvolvimento: ICONIX; SCRUM; XP; Pair Programming;	Opcional	

3.3.4 Prototipagem

A abordagem utilizada para especificar os vários requisitos do projeto foi essencialmente a prototipagem. Porém, no momento inicial em que se procedeu a análise de requisitos, houve uma tentativa de fazer uso de *storyboards* para o levantamento de requisitos, que se realizou com toda a equipa do INSA.

O resultado que se obteve com a utilização da abordagem denominada de *Storyboarding* foi o menos desejável, onde parte da equipa que não possuía conhecimentos informáticos necessários para este processo, encontrava-se com um nível elevado de dificuldade para perceber quais as funcionalidades que estariam a ser discutidas no momento. Basicamente, o que ocorreu foi o levantamento de requisitos, realizado por apenas metade da equipa.

Para que conseguíssemos contornar este problema decidimos seguir pela abordagem da prototipagem, em que, ao apresentar algo que fosse semelhante ao resultado final, conseguíamos pelo menos obter um conjunto de opiniões ou críticas que pudessem levar à especificação de alguns requisitos.

O que se obteve como resultado foi melhor do que o esperado. Grande parte dos elementos conseguiram identificar as necessidades e serviços que deveriam estar presente no produto final, e isto, em consequência, levou então ao aumento do número de requisitos que se obteve em cada uma das reuniões iniciais.

Este tipo de resultado ao utilizar a prototipagem é completamente aceitável para o cenário em que foi obtido, visto que os elementos, aqueles que tiveram mais dificuldades ao serem expostos aos *storyboards*, poderão ser considerados como clientes ou utilizadores finais da aplicação, e para eles é bastante mais simples identificaram funcionalidades que desejam obter ao serem apresentados com algo que se assemelha com o produto que irão utilizar, do que apontarem estas mesmas funcionalidades em "ideias" expostas em papel. Existem estudos que comprovam a eficácia de utilização de protótipos para a análise e avaliação de um projeto pelo simples facto de existir uma melhor interação entre o utilizador e o produto [10].

3.4 Normas e Modelo de Dados

Tal como foi introduzido no capítulo de Revisão da Literatura, a EFSA produziu um conjunto de documentos que pretendem ajudar na implementação do padrão SSD. O pacote de ficheiros enviados à INSA com a documentação do SSD contém 4 ficheiros julgados de maior importância:

- **Guidance of EFSA - Guidance on Data Exchange.pdf:** Trata-se de um documento semelhante a um manual que descreve os vários aspetos presentes no padrão SSD. Estes aspetos incluem os formatos de ficheiros aceites pelos serviços disponibilizados pela EFSA, medidas de segurança necessárias para a utilização destes mesmos serviços, protocolo para a transmissão de dados, regras que deverão ser impostas aos dados, e vários outros pontos.

- **StandardSampleDescription.xls:** Este ficheiro contém informações que dizem respeito a cada um dos 73 atributos (ou variáveis, termo utilizado pela EFSA). O tipo de informação contido neste ficheiro varia desde a descrição de cada um dos vários atributos, até a apresentação do próprio conteúdo que poderá ser atribuído a estes atributos.

- **StandardSampleDescription.1.0.xsd:** Este XML Schema tem como objetivo validar os ficheiros XML gerados pela solução de modo a que esteja de acordo com as regras definidas pelo padrão SSD e que possa ser aceite pela plataforma de recolha de dados criada pela EFSA, denominada de DCF (Data Collection Framework).

- **GenericReportingFormat.xls:** Este ficheiro apresenta uma estrutura que seria a ideal que fosse adotada pelas várias autoridades competentes. Este ficheiro contém um conjunto de macros incorporadas para que, quem estiver a introduzir os dados das amostras, consigam localizar com facilidade os códigos e os valores das linguagens controladas que constituem o SSD.

A melhor forma para dar a conhecer as normas impostas pelo padrão SSD seria começar por mostrar realmente o que é o SSD, como é constituído e como deverá ser utilizado. Portanto o que irá ser explicado de seguida será baseado no conteúdo nos 4 ficheiros já referidos.

O padrão SSD possui no total um conjunto de 73 atributos, estes atributos representam as várias características que estão presentes nas amostras. Quando uma autoridade de controlo desloca-se a um determinado local para recolher amostras de alimentos e outros géneros alimentícios, esta autoridade terá de preencher uma ficha sobre estas determinadas amostras. Esta ficha que é preenchida varia de autoridade para autoridade.

As autoridades e os seus respetivos laboratórios, em conjunto, definem os parâmetros analíticos a serem examinados para cada amostra. No local de recolha da amostra são preenchidos alguns destes parâmetros, como por exemplo, a região do país em que foi recolhida ou a identificação do estabelecimento. Os restantes parâmetros são preenchidos pelos laboratórios após obterem os resultados dos exames.

Os parâmetros definidos pelas autoridades diferem em vários aspetos dos parâmetros presentes no padrão SSD. O principal aspeto, que normalmente é a causa de problemas ao aplicar a transformação dos dados, é o nível de detalhe. A melhor forma para demonstrar este caso será através da análise de uma das linguagens controladas que poderá ser atribuída a um determinado parâmetro.

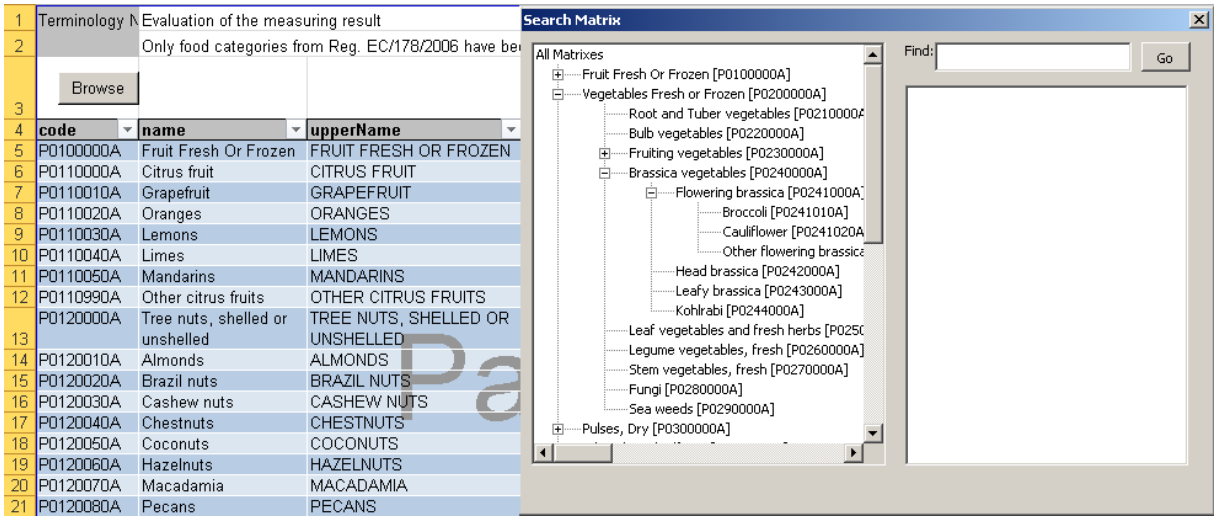


Figura 7. Conteúdo do parâmetro/atributo MATRIX

Do lado esquerdo da Figura 7 é apresentado o conteúdo para a linguagem controlada MATRIX. Este conteúdo indica quais os valores que poderão ser atribuídos quando é efetuado o preenchimento dos dados das amostras ou a transformação dos dados para o padrão. É de salientar também que esta é a única forma com que o conteúdo das várias

linguagens é apresentado, isto é, estas linguagens controladas estão contidas no ficheiro Excel "StandardSampleDescription.xls" distribuídas por folhas de cálculo.

Se analisarmos o lado direito da Figura 7, é possível notar uma estrutura em hierarquia para a escolha dos valores. No exemplo dado para a linguagem MATRIX, a hierarquia tem a profundidade máxima de 4 níveis. O problema que diz respeito ao nível de detalhe, introduzido acima, deve-se ao facto de as autoridades, ou seus laboratórios, não preencherem os dados das amostras com detalhe o suficiente para que se consiga encontrar o código, por exemplo da linguagem MATRIX, que represente o valor introduzido pela autoridade/laboratório.

A autoridade quando estiver no processo de preencher os dados de uma determinada amostra, esta deverá introduzir os valores com informação suficiente para que se consiga percorrer a árvore da hierarquia até encontrar-se o nó correspondente ao valor introduzido. Este é um dos requisitos principais impostos pela EFSA através do padrão SSD.

Na Figura 7 foi apresentado um dos 73 parâmetros presentes no padrão SSD. Como já foi referido, muitos dos parâmetros possuem uma linguagem controlada. Estas linguagens ou terminologias controladas poderão estar estruturadas em forma de árvore, como já foi visto para o MATRIX, ou poderão estar estruturadas em forma de uma lista simples de valores, como por exemplo, uma lista de países.

4	Element Code	Element Name	Element Label	Type[7]	Controlled terminology
5	S.01	labSampCode	Laboratory sample code	xs:string (20)	
6	S.02	labSubSampCode	Laboratory sub-sample code	xs:decimal (4,0)	
7	S.03	lang	Language	xs:string (2)	LANG
8	S.04	sampCountry	Country of sampling	xs:string (2)	COUNTRY
9	S.05	sampArea	Area of sampling	xs:string(5)	NUTS
10	S.06	origCountry	Country of origin of the product	xs:string (2)	COUNTRY
11	S.07	origArea	Area of origin of the product	xs:string (5)	NUTS
12	S.08	origFishAreaCode	Area of origin for fisheries or aquaculture activities code	xs:string (10)	FAREA
13	S.09	origFishAreaText	Area of origin for fisheries or aquaculture activities text	xs:string (250)	
14	S.10	procCountry	Country of processing	xs:string (2)	COUNTRY
15	S.11	procArea	Area of processing	xs:string (5)	NUTS
16	S.12	EFSAProdCode	EFSA Product Code	xs:string (250)	FOODEX
17	S.13	prodCode	Product code	xs:string (20)	MATRIX
18	S.14	prodText	Product full text description	xs:string (250)	
19	S.15	prodProdMeth	Method of production	xs:string (5)	PRODMD
20	S.16	prodPack	Packaging	xs:string (5)	PRODPAC
21	S.17	prodTreat	Product treatment	xs:string(5)	PRODTR

Figura 8. Alguns parâmetros e a suas linguagens controladas

Apesar de existir um total de 73 parâmetros, apenas 23 destes parâmetros são obrigatórios de preenchimento. Isto deve-se ao facto de que o padrão SSD ter sido criado com o objetivo de ser mais genérico possível, isto é, este padrão deverá aceitar dados de qualquer tipo de alimentos ou géneros alimentícios.

Muitas das autoridades competentes estão apenas focadas em domínios específicos às suas funções, como por exemplo o LNIV (Laboratório Nacional de Investigação Veterinárias), que tem principal foco o domínio da sanidade animal e higiene pública. Para o LNIV, os parâmetros que se enquadram na área da pesca e recursos do mar não irão ser preenchidos. Isto verifica-se para todas as autoridades nacionais que fazem parte deste projeto, contudo, desde que estas autoridades preencham ou disponibilizem, nos seus relatórios, informação o suficiente para se conseguir satisfazer os parâmetros obrigatórios, então estas autoridades estão a cumprir com os primeiros requisitos do padrão SSD, e consequentemente da plataforma PT.ON.DATA desenvolvida.

Satisfazendo então estes requisitos mínimos, a EFSA descreve ainda os restantes conjuntos de condições a cumprir de forma a respeitar por completo o padrão SSD. Estas condições estão todas detalhadas no documento "Guidance of EFSA - Guidance on Data Exchange", e grande parte delas dizem respeito à formação, utilização e envio do ficheiro XML. Mas para além destas especificações, existem um outro conjunto de regras que merecem atenção. Estas regras são identificadas como regras de validação onde cada uma descreve, como o nome indica, quais as validações que deverão ser aplicadas aos dados para que sejam considerados válidos.

Tabela 5. Regras de validação para o padrão SSD

Param	Rule Code	Rule	Variables	Error Type	Error Code
R.04	BR09A	"Day of analysis" has to be between 1 and 31	analysisD$1$31	E	ER04A
R.06	BR01A	Where paramCode <> "RF-XXXX-XXX-XXX" (Not in list) then (paramCode, labSampCode, labSubSampCode) must be unique for a data sender;	paramCode$^=$ "RF-XXXX-XXX-XXX"$labSampCode paramCode orgId	E	ER06A

A Tabela 5 exibe duas das múltiplas regras contidas no documento, e ao analisar estas duas entradas é possível claramente compreender o que cada uma destas regras impõe, as suas descrições são simples e estão devidamente identificadas. Porém existe um problema no que diz respeito à implementação destas regras.

Quando os dados são reportados à EFSA através da DCF, esta plataforma também efetua um processo de validação onde as dezenas de validações são aplicadas aos dados. Este processo geralmente tem a duração de várias horas, no entanto pode chegar a durar dias até receber um resultado. Caso o ficheiro esteja válido, este é apresentado com um estado de sucesso e poderá então seguir para os próximos procedimentos exclusivos á EFSA. Contudo, se o ficheiro apresentar o estado de rejeitado ou inválido, um relatório de erros é enviado à entidade que o submeteu indicando todas as regras que não respeitou.

Após receber um relatório de erros gerado pela invalidação dos dados, a equipa de desenvolvimento teria de examinar o relatório de forma a encontrar as entradas inválidas para que se proceda então a correção dos dados. Todo este processo tem um custo demasiado elevado e afeta as restantes tarefas de forma negativa.

Por forma a solucionar este problema, decidiu-se implementar uma validação semelhante na plataforma PT.ON.DATA para quando fosse feita a produção do ficheiro XML a ser enviado à EFSA, os responsáveis pelo tratamento dos dados pudessem verificar a existência de erros e procederem a correção dos mesmos de uma forma simples e instantânea. No momento inicial do desenvolvimento desta nova funcionalidade teve-se conhecimento de que implementar cada uma das regras manualmente e introduzi-las diretamente no código fonte não seria a melhor solução, em termos de tempo despendido e evolução do padrão SSD. Optou-se por fazer recurso à linguagem específica para estas regras, que está por sua vez presente em todas as tabelas das regras de validação e que pode ser observada na terceira coluna da Tabela 5.

Um *parser* para este tipo de linguagem foi implementado para o módulo de validação de dados, e o carregamento de todas as expressões presentes nas tabelas foi realizado através de uma ferramenta também desenvolvida para este propósito. Seguindo esta abordagem conseguiu-se manter-se a base de dados atualizada com todas as regras de validação existente que também poderão surgir no futuro.

3.4.1 Evolução do padrão

A EFSA a qualquer momento poderá produzir uma nova versão do padrão SSD que terá de ser utilizado pelos diferentes estados membros. O processo de transição de uma versão do padrão para outra mais recente requer uma alteração completa da plataforma PT.ON.DATA, dado que uma versão mais recente poderá trazer novas linguagens controladas e outros conjuntos de valores que não existem no momento na base de dados.

Caso existam estas alterações radicais do padrão e não apenas correções de valores, a plataforma terá de possuir um conjunto vasto de ferramentas para o carregamento de novas tabelas que contém as linguagens controladas, para o carregamento de novas regras de validação, e ferramentas para outras funcionalidades igualmente necessárias.

Para que se consiga evitar o desenvolvimento de instrumentos adicionais sempre que exista a necessidade de atualizar o padrão, seria ideal que o próprio padrão possuísse uma estrutura que permitisse uma fácil evolução. Esta estrutura teria de possibilitar apenas a alteração dos objetos que pertencem à versão anterior do padrão, como por exemplo as tabelas da base de dados e formato do ficheiro XML a ser gerado, para que transite para a nova versão. Esta sugestão foi enviada à EFSA para que consigam criar a nova especificação do padrão SSD 2 sem impor demasiado esforço aos estados membros.

3.5 Mapeamento

De acordo com os requisitos em mão estabelecidos para a solução de Portugal, a plataforma desenvolvida surgiu com a finalidade de efetuar mapeamentos de ficheiros de relatórios que possuíam formatos desconhecidos e distintos entre eles, para um formato devidamente definido e estabelecido. Procurou-se por seguir a abordagem que mais facilitasse a tarefa dos elementos envolvidos, que por sua vez veio dar origem a um tipo de mapeamento que não se encontrava implementado por nenhum dos estados membros. No entanto, este processo de mapeamento sofreu várias transformações ao longo do desenvolvimento do projeto.

A primeira aproximação que se poderia tomar para efetuar o processo de mapear um ficheiro seria, de forma manual, realizar uma leitura a cada uma das entradas de um relatório e traduzir

estas entradas para o padrão SSD, tentando encontrar os valores das linguagens controladas presentes nas várias tabelas disponibilizadas que mais se aproximassem aos valores introduzidos pela entidade autora do relatório.

O processo manual de mapeamento foi realizado por vários elementos da equipa do INSA durante o período inicial do desenvolvimento do projeto informático. Este processo produziu resultados poucos satisfatórios, o que levou a mudança do foco do projeto para centralizar-se mais no auxílio ao mapeamento.

Como resultado, decidiu-se fazer uso de um *template* também disponibilizado pela EFSA mas criado com ajuda da entidade irlandesa FSAI. Este *template* no formato Excel possui numerosas macros e simples ferramentas de procura que facilitam o preenchimento células com valores e códigos definidos pelas linguagens do SSD. Devido a utilização do *template*, o procedimento para efetuar o mapeamento dos relatórios estaria então mais simplificado, onde a pessoa encarregue desta tarefa apenas teria de analisar os registos contidos no relatórios e fazer uso das ferramentas de pesquisa localizadas no *template*, ignorando quase por completo a consulta às numerosas tabelas de terminologias controladas.

Após estar preenchido, o *template* seria enviado à plataforma PT.ON.DATA para que fosse realizado a transformação para o formato XML seguindo as normas definidas pelo SSD. Este foi o processo adotado para a criação do protótipo funcional, uma vez que a EFSA necessitava dos resultados até a fase de testes com todos os estados membros.

3.5.1 Dificuldades no mapeamento

Apesar da utilização do *template* disponibilizado ter sido uma grande melhoria na produção de relatórios no formato SSD, este continuou a ser um procedimento que necessitava ainda de um grande período de tempo para ser concluído. Dependendo da complexidade e tamanho do ficheiro a ser mapeado, este poderia demorar de 2 a 3 meses até estar concluído por um grupo de 3 elementos a trabalhar em conjunto na tarefa.

Com o intuito de resolver este problema, decidiu-se modificar as ferramentas existentes e adicionar novas funcionalidades ao protótipo criado. Para tal foi realizado uma análise às vantagens da utilização do *template* e à complexidade e estrutura dos relatórios recebidos.

	Amostra	Unidade Laboratorial	Tipo de Amostra	DataResultado	Método	Parâmetro	Resultado	Res.Calc.	Res.Trat.	Unidade
2	3019	LFQ - Laboratório de Físico-Química	Mel de néctar ou mel de flores, não tropical	03-05-2012	QMI 04 de 01/06/2009	Água (LFQ)	18	18	18,0	%
3	3019	LFQ - Laboratório de Físico-Química	Mel de néctar ou mel de flores, não tropical	03-05-2012	QMI 112 de 19/07/2011	Hidroximetilfurfural (LFQ)	15.2151	15.2151	15,2	mg/kg

Relatório recebido diretamente da entidade: ~20 atributos/colunas

	Result code (R.01)	Year of analysis (R.02)	Month of analysis (R.03)	Day of analysis (R.04)	Parameter code (R.06)	Parameter code (Text preview) (R.07)	Parameter full text description	Parameter type (R.08)
1					Parameter List			
2	PT-ASAE-2011-00001	2011	7	13	RF-00000110-CHE	Sulfur and derivatives	Anidrido Sulfuroso Total	Individual
3	PT-ASAE-2011-00002	2011	7	26	RF-00000085-CHE	Nitrate	Nitratos	Individual
4	PT-ASAE-2011-00003	2011	7	26	RF-00000085-CHE	Nitrate	Nitratos	Individual
5	PT-ASAE-2011-00004	2011	7	26	RF-00000085-CHE	Nitrate	Nitratos	Individual
6	PT-ASAE-2011-00005	2011	7	15	RF-00000210-CHE	Melamine	Melamina	Individual
7	PT-ASAE-2011-00006	2011	7	15	RF-00000210-CHE	Melamine	Melamina	Individual
8	PT-ASAE-2011-00007	2011	7	15	RF-00000210-CHE	Melamine	Melamina	Individual
9	PT-ASAE-2011-00008	2011	7	21	RF-00000122-CHE	Sulfur Dioxide	Dióxido de enxofre total	Individual

Relatório manualmente mapeado: 73 atributos/colunas, onde 23 são obrigatórios

```
<message xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <header>
    <type>dcfmsg</type>
    <version>0.01</version>
    <code>code1</code>
    <receiverCode>receiverCode1</receiverCode>
    <senderCode>senderCode1</senderCode>
    <sentDate>1900-01-01T01:01:01.0000000+00:00</sentDate>
  </header>
  <dataTrx receiverTrxCode="receiverTrxCode1" senderTrxCode="senderTrxCode1" opType="01" dcCode="dcCode1" dcName="dcName1" trxComm="trxComm1">
    <dataset>
      <sample>
        <labSampCode>HP-11-04033</labSampCode>
        <lang>pt</lang>
        <sampCountry>PT</sampCountry>
        <sampArea>PT18</sampArea>
        <origCountry>XX</origCountry>
        <procCountry>XX</procCountry>
        <EFSAProdCode>G.14.1.6</EFSAProdCode>
        <prodCode>XXXXXXA</prodCode>
        <prodText>Alimento composto para suínos engorda/acabamento</prodText>
        <prodProdMeth>Z0215</prodProdMeth>
        <prodPack>H999A</prodPack>
        <prodTreat>T100A</prodTreat>
        <sampY>2011</sampY>
        <sampM>4</sampM>
        <sampD>29</sampD>
        <progCode>CAA Plano 2011</progCode>
        <progLegalRef>Controlo de Alimentação Animal</progLegalRef>
        <progSampStrategy>ST10A</progSampStrategy>
        <progType>K005A</progType>
        <sampMethod>N014A</sampMethod>
        <sampPoint>E980A</sampPoint>
```

Ficheiro XML gerado para o envio à EFSA

Figura 9. Transformação dos dados

Um dos aspetos que se identificou após efetuado a análise aos relatórios das diversas entidades, foi que o volume de dados varia bastante de entidade para entidade, onde observaram-se ficheiros que possuíam por volta das 5000 entradas de amostras e seus resultados, mas ao mesmo tempo foram identificados ficheiros com apenas dezenas de registos neles contidos.

Como é ilustrado pela Figura 9, os relatórios recebidos pelas entidades possuíam um número reduzido de atributos, por volta de 20. Esta escassez de atributos provou ser uma dificuldade para o processo de mapeamento, visto que, a partir de 20 atributos teria de se tentar preencher 73 pertencentes ao SSD. No entanto este problema foi superado de duas formas:

- **Comunicação com a autoridade de controlo:** A autoridade de controlo, não tendo conhecimento do padrão SSD, envia os relatórios com os dados mais relevantes, no seu ponto de vista, sobre as amostras. Após entra-se em contato com a autoridade, esta recolhe dos seus laboratórios os restantes dados necessários.

- **Extração de informação agrupada:** Por vezes as autoridades de controlo tendem a introduzir em alguns dos atributos informação com demasiado detalhe ou vários tipos de informação inseridos em um só atributo. Um exemplo deste caso poderá ser o atributo da "Data da Colheita", onde as várias autoridades têm a tendência por preencher o valor do dia, mês e ano na mesma célula, mas é necessário que esses 3 diferentes valores sejam repartidos para os seus respetivos atributos do padrão SSD.

Existe também uma grande diferença entre a taxonomia e linguagens utilizadas pelas entidades e a EFSA. Os parâmetros analisados, métodos de análise, entre outros atributos apresentam por vezes terminologias diferentes daquelas utilizadas pela EFSA, isto requer que intervenção humana seja tomada para que se aplique a devida tradução dos termos.

3.7 Worflow

Por forma a introduzir a solução desenvolvida, será agora apresentado nesta secção o *workflow* geral do sistema, juntamente com os diagramas de atividade das mais importantes funcionalidades presentes na solução.

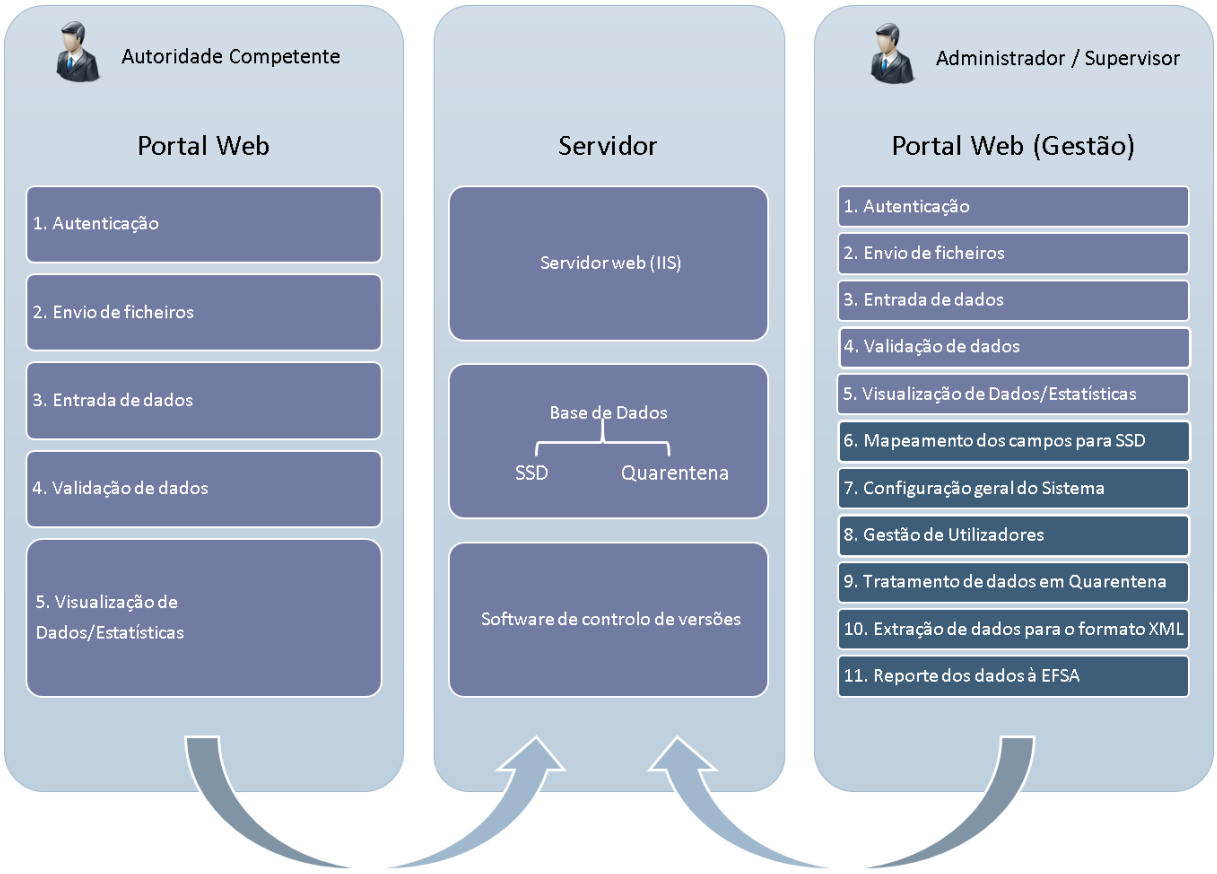


Figura 10. Workflow geral do sistema

A Figura 10 apresenta as três componentes principais que constitui o sistema. O primeiro componente é designado de portal ou plataforma web, onde nele são realizadas, principalmente, as operações orientadas aos dados. Este é o ponto de entrada dos ficheiros que contém registos de amostras e é o componente dispõe de interfaces específicas para comunicação com as autoridades de controlo. Para usufruir dos serviços disponibilizados por este módulo, cada autoridade que irá fazer uso da plataforma PT.ON.DATA terá de possuir uma conta associada, em que esta conta é utilizada para a autenticação no sistema e auxílio na identificação de quem originou determinadas ações na plataforma, ações estas relacionadas com os dados ou os diversos processos. A Figura 10 apresenta, para este constituinte, as principais ações que disponibiliza para as autoridades de controlo.

A plataforma é essencialmente uma aplicação web que necessita de comunicar com o servidor sempre que exista a necessidade de efetuar os processamentos requisitados pelo cliente. Este servidor é considerado como o segundo componente do sistema e tem nele contido uma aplicação que atua com servidor web, disponibilizando as várias interfaces para a interação com os utilizadores. Para além da aplicação de servidor web, está presente um dos elemento

mais importante de todo o sistema, a base de dados. Esta base de dados construída sobre a tecnologia Microsoft SQLServer 2008 R2[8] armazena toda a informação que circula pelo sistema. As tabelas de linguagens controladas criadas para o padrão SSD, bem como as restantes tabelas utilizadas para a gestão da plataforma web, como por exemplo as tabelas com informações dos utilizadores, estão localizadas nesta base de dados. Dos vários objetos nela contida destacam-se duas tabelas, SSD e Quarentena, que têm como objetivo armazenar, respetivamente, as entradas válidas e as entradas inválidas dos vários relatórios de amostras carregadas para o sistema.

Por fim, o último componente constituinte do sistema desenvolvido é semelhante ao primeiro, isto é, trata-se da plataforma web porém com todas as funcionalidades disponíveis para os administradores do sistema. A partir desta vista da plataforma PT.ON.DATA, os supervisores e administradores gerem todos os aspetos presentes nela, e efetuam várias tarefas críticas de forma a garantir um correto fluxo de funcionamento das várias operações que definem o padrão SSD, como por exemplo, são os supervisores que estão encarregados por proceder ao mapeamento dos dados.

3.7.1 Diagrama de atividades

Grande parte dos processos que são realizados através da plataforma construída, que em conjunto pretendem atingir os objetivos impostos pela proposta e pela EFSA, estes requerem um certo nível de atenção pelo facto de serem bastantes distintos dos processos com os mesmos objetivos contidos nas soluções desenvolvidas pelos diferentes estados membros.

Como foi anteriormente discutido nos passados capítulos, para a implementação desta solução foi seguida uma abordagem diferente baseada nos resultados e nas experiências obtidas por cada um dos estados membros na criação das suas soluções. De forma a auxiliar na descrição nos processos que resultaram da escolha desta específica abordagem, serão apresentados de seguida diagramas de atividades. No entanto, serão apenas descritos os processos que constituem o trajeto principal para a criação e envio do ficheiro XML, uma vez que através da análise deles será possível efetuar uma comparação mais exata entre as várias soluções já referidas, conseguindo-se ao mesmo tempo obter uma melhor perceção da complexidade,

[8] http://www.microsoft.com/en-us/sqlserver/product-info.aspx

dimensão do projeto, bem como as várias ações que devem ser tomadas em cada uma das etapas.

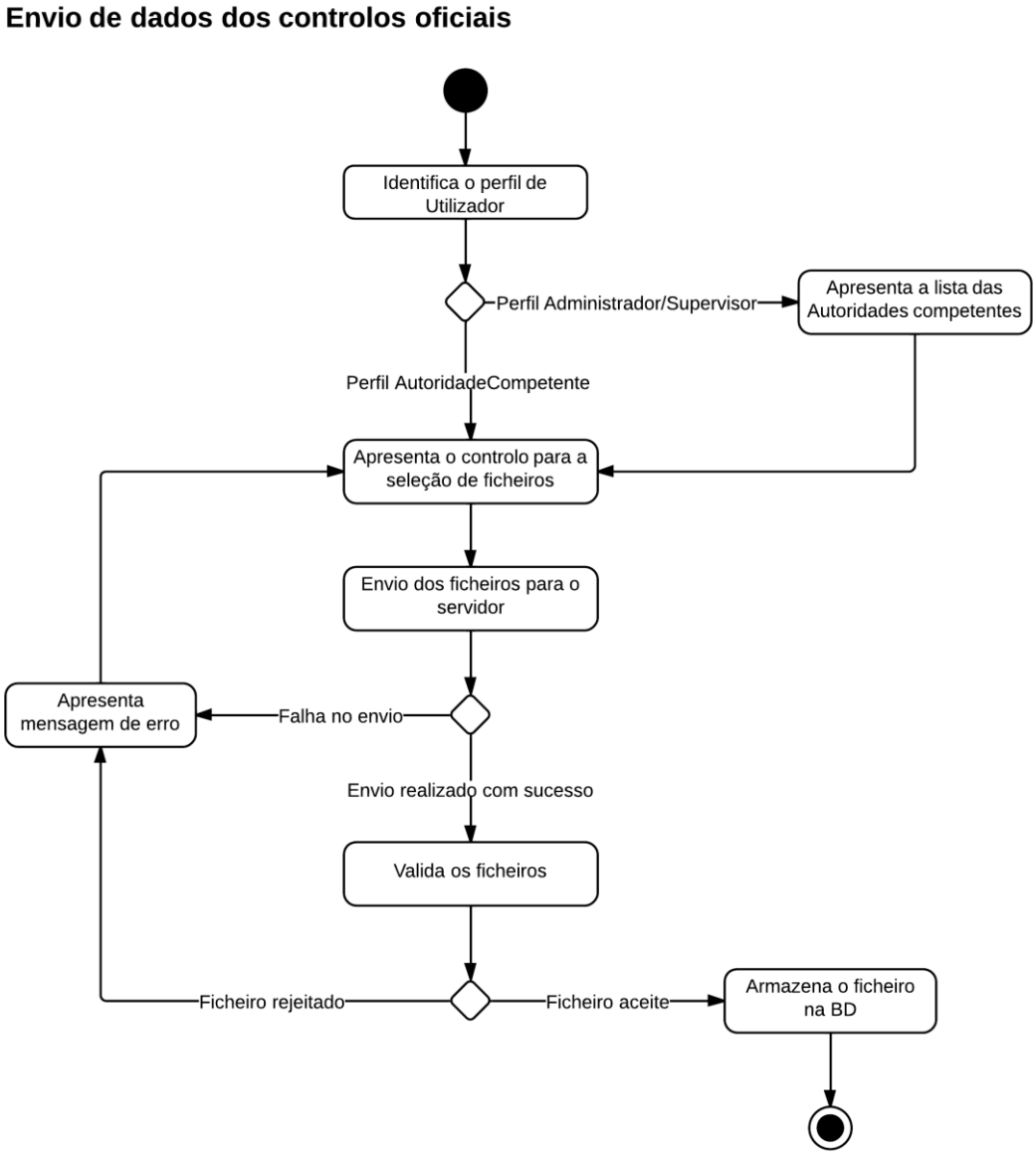


Figura 11. Diagrama de atividade para o processo de carregamento de relatórios

O diagrama de atividade da Figura 11 descreve o processo que é realizado por grande parte dos utilizadores da plataforma web PT.ON.DATA. Dependendo do perfil associado à conta do utilizador, diferentes opções serão apresentadas para o *upload* dos ficheiros que representam os relatórios criados pelas várias autoridades competentes. Inicialmente teria sido determinado que apenas os utilizadores com o perfil de responsável para o carregamento dos ficheiros da autoridade em que está associado, teria a possibilidade de enviar os relatórios através da plataforma, no entanto verificou-se, com base nas reuniões realizadas com as

próprias autoridades, que os administradores e supervisores teriam também de possuir esta funcionalidade para situações em que os relatórios seriam enviados por *email* para serem posteriormente tratados pelos membros do INSA encarregues para esta tarefa. Este tratamento consiste apenas em uma limpeza bastante superficial dos dados, e é somente realizado para detetar erros de fácil correção e que foram salientados pela autoridade que o enviou. Após possuir os relatórios corrigidos estes seriam então carregados para o sistema pelos administradores.

Após realizada a seleção de um ou mais ficheiros a serem enviados, estes ficheiros são então remetidos para o servidor onde serão armazenados temporariamente em formato Excel com a versão semelhante em que foi enviado, de forma a proceder-se a uma validação inicial em que são verificados alguns aspetos relacionados com o ficheiro. Seria importante salientar que para que fosse possível efetuar a leitura e réplica do ficheiro enviado ao servidor, foi realizada uma análise de quais seriam as ferramentas mais apropriadas para estas funções, considerando que se estaria a lidar com o formato Excel. Esta análise foi revelante pelo facto de grande parte dos documentos serem criados e remetidos com formato anteriormente mencionado, e visto a obtenção dos dados de forma correta é um processo em que não são toleradas falhas, a utilização e tratamento destes ficheiros teriam de ser realizados com as ferramentas mais apropriadas.

Por forma a encontrar a melhor ferramenta para a manipulação de ficheiros em formato Excel, verificou-se que existiam duas APIs disponíveis, Microsoft Excel Primary Interop Assembly (PIA)[9] e Microsoft OLE DB[10]. Visto que este formato pertence à Microsoft®, faria todo o sentido utilizar algo concedido por eles, negando assim a utilização de ferramentas criados por terceiros. Com o estudo feito da documentação disponível para as duas APIs, confirmou-se que a utilização da Microsoft OLE DB seria a melhor escolha, uma vez que esta não necessita que uma instalação da aplicação Microsoft Excel esteja presente no servidor, possuindo também a vantagem de ser mais rápida no acesso ao conteúdo do ficheiro e utilizar um número vasto de funcionalidades presentes em DBMS (DataBase Management System), como por exemplo a utilização de *queries* na linguagem SQL para o acesso aos dados.

[9] http://msdn.microsoft.com/en-us/library/ff597926(v=office.14).aspx

[10] http://msdn.microsoft.com/en-us/library/windows/desktop/ms722784(v=vs.85).aspx

A seguir a possuir o ficheiro do relatório presente no servidor, é então feita uma pequena validação que consiste em verificar se o ficheiro já existe no sistema, através do cálculo do seu *hash* utilizando o algoritmo MD5. De seguida certifica-se se o ficheiro possui colunas e registos. Caso o ficheiro não satisfaça estas condições, este é rejeitado pelo sistema e uma mensagem com o erro respetivo é enviada ao cliente. No entanto, se o ficheiro for classificado como válido, o seu conteúdo é então armazenado na base de dados do servidor, finalizando com o envio de uma notificação aos supervisores e administradores de que foi efetuado uma nova entrada de relatório no sistema.

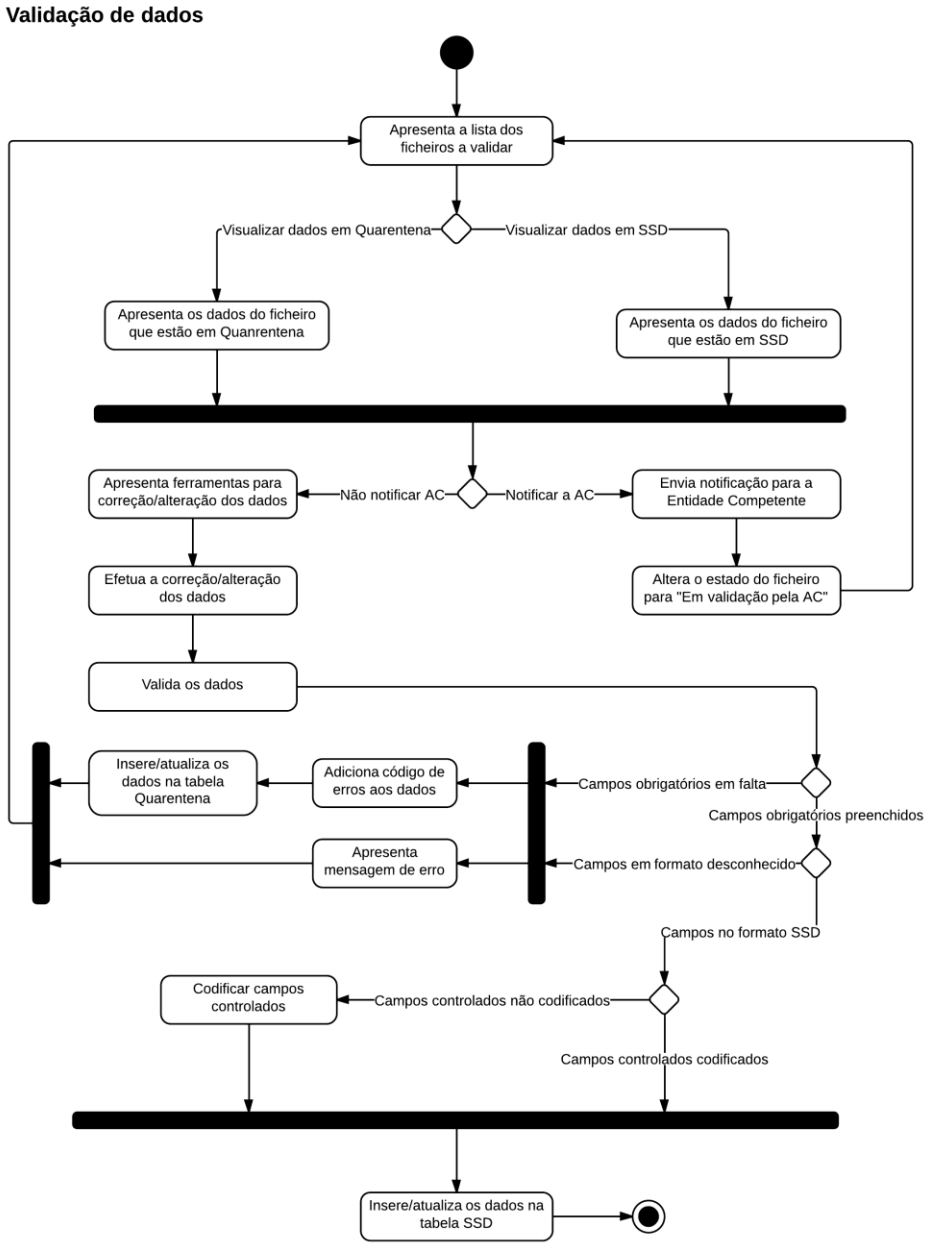


Figura 12. Diagrama de atividade para o processo de validação dos dados

Em seguida, após concluir o processo de envio dos relatórios, os supervisores da plataforma poderão proceder à validação dos dados. Este processo é ilustrado pela Figura 12, e como é possível concluir através da análise a figura, este é um dos processos mais complexos existentes no sistema.

Esta atividade inicia com a apresentação de todos os ficheiros que estão prontos para o processo de validação, e partir da seleção de um, o supervisor terá a opção de visualizar os registos que estão na tabela SSD ou então na tabela Quarentena. Como já anteriormente explicado, os registos que se encontram na tabela SSD são aqueles que passaram pelo processo inicial de validação e consequentemente foram classificados como válidos, em oposição, registos que encontram-se na tabela Quarentena são aqueles que foram classificados como inválidos pelo mesmo processo.

O procedimento de validação de dados a ser descrito tem como o objetivo de certificar que a informação, contida nos registos, foi validada corretamente. A autoridade competente envia o relatório através da plataforma mas não tem conhecimento do resultado da avaliação dos seus dados, já que para os seus laboratórios os dados estão corretamente registados e igualmente formados, mas o mesmo poderá não ser considerado para as normas da EFSA. Múltiplos registos poderão ser classificados como inválidos e serem enviados para a tabela Quarentena por possuírem erros pouco óbvios, no entanto simples de proceder a sua correção, como é o caso comum cometido por várias autoridades, onde é feita a concatenação do dia, mês e ano de uma certa data e o resultado desta concatenação é colocado em um só célula do relatório, mas para a EFSA estas unidades deverão estar presentes em células específicas. Este processo é utilizado para a identificação destes tipos de erros e o supervisor poderá então notificar a autoridade encarregue pelo envio do relatório a ser examinado, para que esta possa então corrigir os dados em questão.

Caso o supervisor não queira notificar a AC por concluir que os erros são de fácil correção e que a sua correção não irá afetar a integridade do relatório, este utilizador tem à sua disponibilidade ferramentas nesta etapa que o auxiliam na correção. Após os erros serem retificados, a validação é novamente aplicada aos novos dados, e dependendo do resultado que tiveram desta nova validação, estes serão armazenados em Quarentena para o caso em que erros ainda persistam nos registos, ou então serão armazenados na tabela SSD caso a correção aos dados tenha sido efetuada com sucesso.

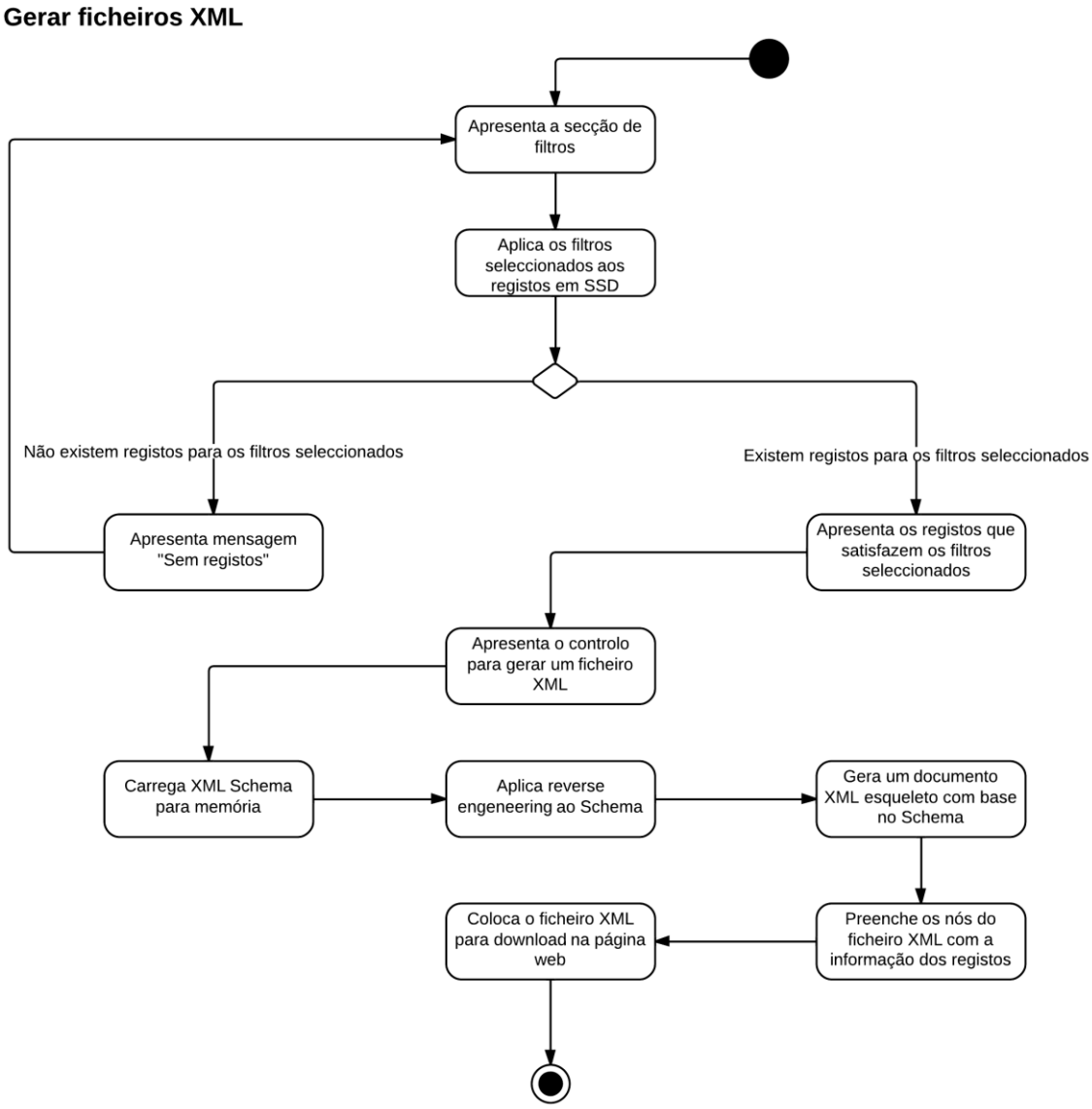


Figura 13. Diagrama de atividade para o processo de criar o ficheiro XML

Esta é a última atividade a ser executada na plataforma PT.ON.DATA de modo a finalizar o procedimento de criação de um relatório em formato XML para que seja analisado pela EFSA. Ao aceder à página de Gerar Ficheiro XML, é apresentado ao supervisor um conjunto de filtros que permitem criar o XML para um determinado conjunto de registos. Este grupo de dados é definido por três filtros que poderão ser utilizados pelo utilizador:

- **Filtro por Grupo de Parâmetro:** Existe uma ferramenta em particular que foi desenvolvida para a plataforma de modo a que permitisse criar grupos constituídos por diferentes parâmetros analíticos. Estes parâmetros são substâncias analisadas ou identificadas em uma amostra. Um determinado laboratório poderá estar focado na

análise de metais pesados para as suas amostras recolhidas e ao concluir as análises poderá encontrar vestígios de, por exemplo, chumbo em um conjunto de amostras, com o auxílio desta ferramenta, o utilizador poderá criar um grupo que contenha apenas o parâmetro analítico "Chumbo", este novo grupo passará a estar presente no filtro de Grupo de Parâmetro para que o utilizador possa criar um ficheiro XML apenas para as amostras com este parâmetro analítico. Este tipo de filtro é necessário pelo motivo de que a entidade EFSA requer que os relatórios em formato XML sejam enviados separados por grupos, grupos estes definidos por eles e que poderão sofrer alterações ao passar do tempo;

- **Filtro por Ano de Colheita:** Este filtro é utilizado para situações em que é necessário o envio de dados de um ano em específico, normalmente requisitado pela EFSA para uma possível reavaliação das amostras para aquele ano. Existe também uma outra possibilidade para o envio de dados de um determinado ano de colheita, que é o caso em uma nova autoridade é introduzida no sistema e torna-se necessário reportar os todos os seus relatórios, efetuado ao longo dos anos, à EFSA;

- **Filtro por Entidade a Reportar:** Este filtro foi criado para a eventualidade em que pudesse surgir uma nova entidade reguladora semelhante à EFSA, que fosse também necessário efetuar o reporte dos dados. O filtro encontra-se apenas preenchido com um item que representa entidade EFSA para a seleção, no entanto está preparado para a adição de novas entidades.

Após feita a configuração dos filtros, os registos que se encontram na tabela SSD serão selecionados com base na filtragem criada. É de relembrar que apenas os registos que estão armazenados na tabela SSD são enviados à EFSA, visto que são estes que foram classificados como válidos pelo sistema. Os registos selecionados são então apresentados juntamente com um controlo que, ao ser receber ordem do utilizador, este irá dar início o procedimento de criação do ficheiro XML através do processo de *reverse engineering* do *schema* disponibilizado pela EFSA, resultando em um estrutura em memória que se assemelha a um esqueleto de um ficheiro XML que será preenchido com os dados dos registos selecionados.

Por fim o ficheiro gerado será enviado ao utilizador/cliente que originou o pedido, em forma de *download*.

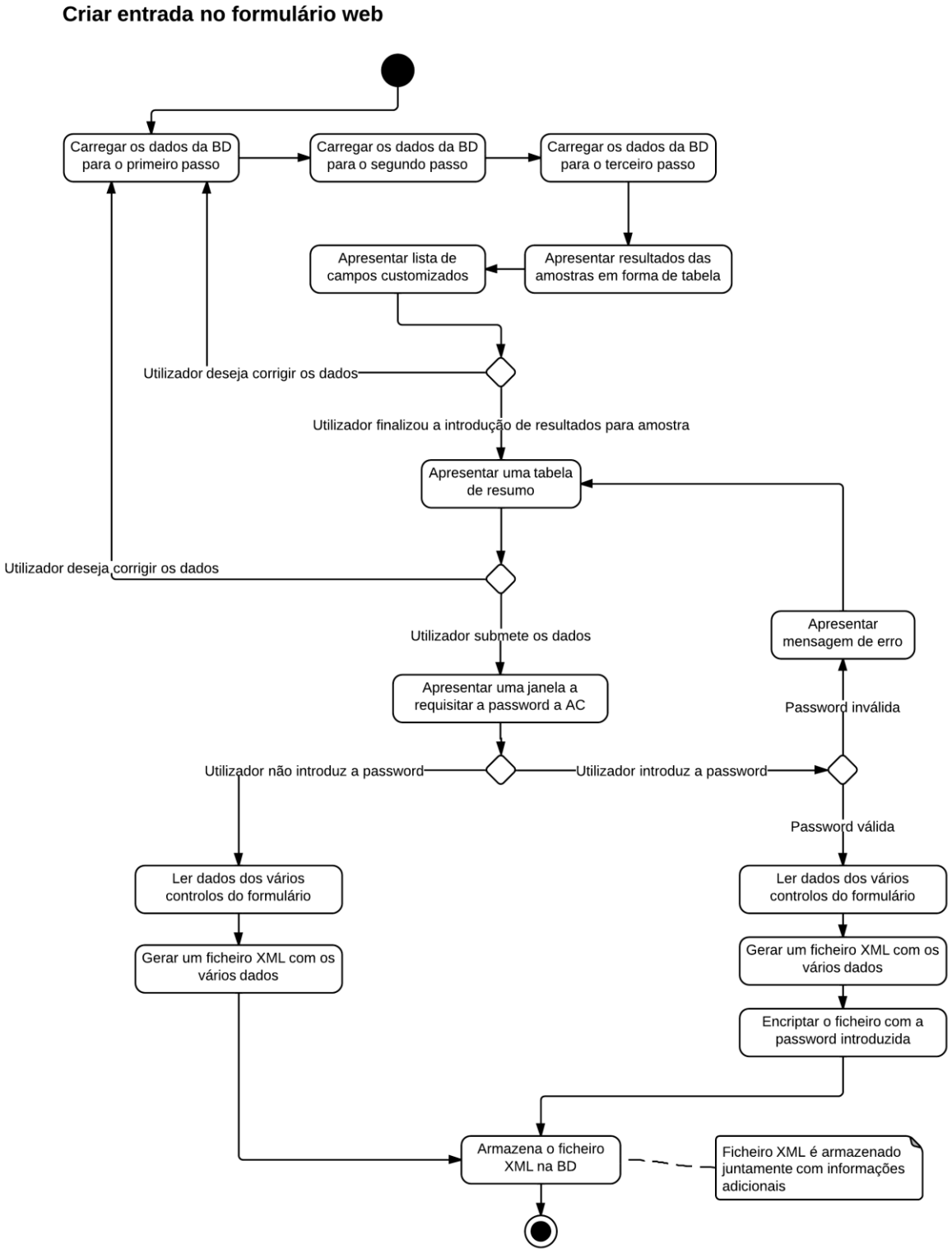


Figura 14. Diagrama de atividade que descreve o processo de criar uma entrada no formulário web.

Por fim, é ilustrado e descrito o diagrama de atividade para o processo de inserção de registos de amostras através do formulário web desenvolvido. Como enunciado em um capítulo anterior, esta ferramenta foi desenvolvida para as autoridades competentes que não possuem

um sistema informático para a introdução e armazenamento de relatórios que se encontram em formato de papel. Pode-se concluir que esta ferramenta é acessível por todos os perfis de utilizador presentes no sistema.

O formulário em si está dividido em 4 etapas, em que cada uma representa o tipo de dados que serão inseridos naquela etapa. O facto de o formulário estar dividido por conteúdo facilita bastante a tarefa do utilizador e o desempenho do servidor ao enviar as diferentes páginas ao cliente, visto que em cada etapa existe um grande número de controlos, como por exemplo *DropDownLists* e *TreeViewControls*, que são preenchidos com conteúdo de múltiplas tabelas que contém um grande volume de entradas, logo o cenário de múltiplas etapas permite que menos controlos sejam apresentados por página, implicando que haja menos leitura às tabelas da base de dados e menor tráfego de dados entre servidor e cliente.

As duas primeiras etapas do formulário são bastantes simples, apenas possuem controlos de fácil preenchimento, no entanto a terceira etapa do formulário, denominada de "Dados Laboratoriais", difere bastante das anteriores porque esta tenta facilitar a inserção registos. Uma amostra recolhida por uma autoridade competente poderá originar em múltiplos resultados, dependendo das análises efetuadas pelos seus laboratórios. A informação da amostra juntamente com os dados dos vários resultados é registada digitalmente de uma forma específica em que, para cada um dos resultados, a informação da amostra teria de estar repetida, isto é, se consideramos uma amostra que apresenta cinco resultados diferentes, este conjunto de dados irá resultar em cinco entradas em um ficheiro Excel de relatório, onde cada uma destas entradas iria possuir a mesma informação sobre a amostra, como por exemplo o valor do local da colheita ou o país da colheita irá estar repetida para as cinco entradas, mas os campos que dizem respeito aos resultados iriam ser diferentes.

De forma a evitar que o utilizador tenha de preencher múltiplas vezes os campos que dizem respeito à amostra para cada resultado, a terceira etapa do formulário assume que as duas etapas anteriores possuem informação sobre a amostra, portanto todos os campos que sejam preenchidos nesta terceira etapa irão estar associados com o resultado. Esta etapa realmente difere das antecedentes por possuir um botão que permite a adição de mais resultados para a mesma amostra, mantendo o utilizador na etapa corrente mas com os dados do resultado adicionado em um tabela na página que permite a visualização dos resultados já adicionados.

Existe também a possibilidade de adicionar, nesta terceira etapa, campos personalizados por cada uma das autoridades competentes. Esta funcionalidade foi sugerida em uma das reuniões com as diferentes ACs que levantaram a questão da necessidade de a adição de campos que não existam no padrão SSD e que fazem sentido existir para a autoridade competente, visto que a autoridade a inserir os dados poderá necessitar de mais informação do que aquela presente no padrão SSD para atingir os seus objetivos.

O utilizador poderá a qualquer momento navegar pelo formulário para efetuar correções em outras etapas. Uma vez concluída a inserção de informação sobre as amostras, este irá proceder para a quarta e última etapa do formulário em que é apresentada uma tabela de resumo que mostra alguns dos campos de maior importância que permitem dar uma visão do que foi introduzido. Aqui o utilizador poderá finalizar todo o processo, no entanto é lhe apresentada uma janela onde é feito o pedido por uma palavra-passe para a encriptação dos dados introduzidos. Esta palavra-passe é gerada aquando da criação da conta do utilizador, e dependendo da sua inserção na janela apresentada, um ficheiro XML será gerado com os dados presentes em todos os campos do formulário e este ficheiro será encriptado se a palavra-passe tiver sido inserida e se for declarada válida, caso contrário se a palavra-passe não for introduzida o ficheiro será armazenado na base de dados em formato de texto legível.

Sistema Desenvolvido e Mapeamento Automático

Os capítulos anteriores foram criados de forma a descreverem os processos e aspetos que contribuíram para o desenvolvimento do projeto, aspetos estes relacionados com o planeamento, organização e gestão de todo o trabalho. Ao longo destes capítulos tentou-se, ao mesmo tempo que iriam sendo descritos as várias tomadas de decisões e procedimentos realizados, dar uma ideia do produto que foi desenvolvido, para que houvesse uma melhor perceção do que realmente era planeado obter.

Este capítulo pretende apresentar os pormenores mais técnicos relacionados com a plataforma desenvolvida. Várias funcionalidades que foram construídas encontram-se caracterizadas em partes anteriores deste relatório, como é o caso do Envio de Relatórios em Excel e do Formulário Web entre outras. Estes módulos, apesar de serem vitais constituintes da plataforma, não são foco principal deste relatório já que para a sua implementação não foram utilizados métodos considerados inovadores para além daqueles já detalhados previamente, mas salienta-se novamente que grande parte do tempo destinado ao desenvolvimento do projeto foi consumida com a criação destes mesmos módulos, o que indica o quanto árduo foi este processo.

Para melhor introduzir o tópico que irá ser detalhado por diante, será feita uma breve síntese do que teria sido pretendido atingir com este projeto e o que foi efetivamente produzido como resultado do desenvolvimento realizado, para que se tenha uma visão geral da solução.

4.1 Síntese introdutória

Uma plataforma web, posteriormente intitulada de PT.ON.DATA, foi desenvolvida com o

objetivo de satisfazer uma necessidade imposta pela entidade europeia que controla riscos associados com alimentação e géneros alimentícios. A EFSA necessita que anualmente sejam efetuados envios de relatórios por entidades responsáveis para este cargo presente em cada um dos países. Para o caso de Portugal, o INSA é a entidade responsável por recolher os vários relatórios gerados pelas restantes autoridades competentes portuguesas. Para que esta tarefa de recolha, tratamento e envio desta coleção de dados possa ser feita de forma mais eficiente por todos os envolvidos, decidiu-se criar uma plataforma que deveria possuir um vasto conjunto de ferramentas com o objetivo de facilitar as diferentes tarefas a serem realizadas pelos técnicos do INSA.

Uma aplicação web foi então criada utilizando a tecnologia ASP.NET Framework, que por sua vez ficaria disponibilizada por um servidor de desenvolvimento que pode ser acedido por entidades e elementos externos para auxiliar no desenvolvimento e, dependendo do estado da plataforma, esta poderá ser utilizada quase como um produto final mas não na sua totalidade, apenas para certas funcionalidades. Foi permitida esta possibilidade uma vez que, ao mesmo tempo que era desenvolvido o projeto, era também requisitado pela EFSA envio dos registos de anos anteriores mas já em formato SSD, formato este definido para este projeto. Portanto, os módulos principais teriam de estar em completamente funcionais para estas ocasiões.

O primeiro módulo a ser implementado foi o carregamento de ficheiros em formato Excel, que realmente não carrega apenas os ficheiros de relatórios mas também cria uma relação na base de dados entre os ficheiros armazenados, utilizador que fez o envio e a entidade que este representa.

O módulo que foi desenvolvido de seguida foi o de validação dos registos contidos na base de dados. Este módulo tem como principal propósito aplicar as regras que os dados deverão respeitar para que sejam considerados válidos perante a EFSA, para que esta consiga então avaliar os riscos com informações precisas. Este módulo também possui um mecanismo de identificação dos erros que invalidaram os dados, e pretende auxiliar os técnicos a procederem o mais rapidamente possível na correção destes mesmos dados.

Por fim, o último módulo a ser concebido para esta etapa foi o que permitia gerar um ficheiro em formato XML com a estrutura definida pelo padrão SSD. Inicialmente este módulo possuiu apenas uma tarefa que executava, mas com a evolução da plataforma, este sofreu

múltiplas alterações que vierem a disponibilizar uma maior quantidade de funcionalidades, tais como a disponibilização de um histórico de envios já efetuados à EFSA que possibilita um reenvio destes mesmo dados caso seja requisitado.

Após a fase de envio de relatórios à EFSA que ocorreu no início do desenvolvimento do projeto, procedeu-se à implementação de ferramentas e outros módulos que, apesar não serem considerados críticos para o projeto como foram considerados os anteriores mencionados, permitem assistir na realização de múltiplas tarefas, sejam elas já existentes ou ainda por existir até a conclusão do projeto. O formulário web e o mapeamento manual foram os seguintes módulos a requerem maior importância, contudo ferramentas adicionais foram implementadas em conjunto sempre que fosse constatado que haveria a necessidade da existirem.

4.2 Serviços integrados com as autoridades

Existem vários tipos de serviços que foram criados para ajudarem as diversas autoridades envolvidas em funções distintas. Para as autoridades de controlo portuguesas foi planeado criar serviços que não fossem demasiados intrusivos, pois ao analisar as experiências e resultados obtidos pelos outros estados membros ao introduzirem algo de novo nos sistemas das autoridades, notou-se que a quantidade de tempo que se necessita para concluir esta operação é demasiada elevada, e que normalmente exige bastante esforço do lado das autoridades.

Todos os serviços que deverão ser utilizados pelas autoridades portuguesas estarão disponíveis pela plataforma web. Um dos serviços, também já discutido anteriormente, é o formulário web, que através de uma conta de utilizador associada a uma autoridade competente, poderá ser utilizado a qualquer momento para a criação de relatórios digitais com linguagens controladas definidas pelo padrão SSD. Os relatórios criados por este serviço poderão ser importados para o sistema que a autoridade possui, como também poderá enviado através da plataforma para proceder ao envio anual dos relatórios à EFSA.

Existe também um serviço integrado com as autoridades que permite que estas estejam sempre a par do estado do processo de tratamento dos seus dados. Este tipo de serviço de notificação é bastante importante para a conclusão do processo mencionado, visto que, caso

exista algum problema com os dados, seja por possuírem valores incorretos ou algum tipo de rejeição que tenha ocorrido devido às regras de validação contidas no SSD, a autoridade terá sempre acesso a estes pormenores e poderá ser alertada que é necessário a sua intervenção para que se resolva o problema em causa. Este alerta é normalmente enviado pelos Supervisores da plataforma, pelo facto de serem estes os utilizadores a efetuarem o mapeamento dos dados e a executarem o processo de validação dos dados. Atualmente este serviço utiliza notificações presentes na plataforma e notificações enviados sobre forma de *email*. Os alertas e requisitos de intervenção são também realizados sobre forma de *email*, sendo posteriormente necessário que a autoridade notificado proceda à plataforma online para uma melhor avaliação da situação.

Existe também o serviço que está integrado diretamente com a EFSA mas que está de momento ainda a ser desenvolvido. Foi referido anteriormente que o envio dos relatórios em formato XML era feito manualmente através da plataforma DCF, com o propósito de testar esse mesmo sistema. No entanto estava planeado desde o início do projeto que a uma dada altura do desenvolvimento, a integração com o DCF teria de ser implementada fazendo uso dos *web-services* que foram feitos disponíveis pela EFSA. Este novo serviço integrado com a EFSA irá substituir o processo manual e irá também ser possível adicionar outras entidades reguladoras para além da EFSA que também necessitam destes tipos de dados.

Por último tem-se um serviço que ainda não foi mencionado mas a sua presença no projeto traz grandes vantagens. A tarefa de transformação de dados em vocabulários utilizados pelas autoridades para o vocabulário utilizado pela EFSA através do padrão SSD necessita de intervenção humana e uma grande quantidade de tempo para ser realizada. Estas dificuldades também foram encontradas no desenvolvimento de soluções para os outros estados membros, e até ao momento a única solução seria a de forçar as autoridades competentes a adaptarem o vocabulário do padrão SSD. Para além dos objetivos principais estabelecidos para este projeto, uma solução foi proposta por um dos elementos (Sidney Tomé) que está a desenvolver a plataforma, que permitisse automatizar o processo descrito. A solução consiste em criar um sistema que fizesse a leitura do vocabulário de uma determinada autoridade e de seguida produzisse o mapeamento desse vocabulário para os códigos correspondentes, presentes nos vários parâmetros do padrão SSD. Tendo o conhecimento do tipo de vocabulário utilizado e o facto de que os dados são introduzidos por elementos pertencente a própria autoridade, o que pode levar a que a probabilidade de ocorrer erro humano seja

elevada, decidiu-se utilizar um conjunto de ferramentas de análise léxica e gramatical, entre outras, para a criação deste sistema, com o objetivo de conseguir-se obter resultados aceitáveis independentemente da qualidade do vocabulário a ser analisado.

4.3 Abordagem seguida

Para um relatório que se encontra em linguagem natural e que seja necessário mapeá-lo para o padrão SSD requer que seja efetuada uma leitura dos vocabulários controlados de forma a encontrar o mapeamento correto. Para certos parâmetros analíticos (ou variáveis, terminologia utilizada pela EFSA) o processo de mapear poderá ser realizado sem dificuldades e de forma rápida, no entanto existem parâmetros que possuem um vocabulário que é constituído por um número bastante elevado de entradas, como é o caso do FoodEx e FoodEx2 que contém por volta das 3000 diferentes entradas.

O FoodEx é um dos parâmetros que mais esforço requer por parte dos técnicos do INSA pelo facto de possuir grande detalhe, como tal decidiu-se inicialmente por utilizar este parâmetro e o seu conteúdo para a criação da ferramenta de análise de vocabulário, mas devido ao seu estado de decadência e ao surgimento de uma versão mais atualizada Foodex2 que por sua vez possui mais entradas e correções efetuadas, optou-se então por utilizar esta versão.

Possuindo o vocabulário alvo com que se irá trabalhar, apenas resta selecionar um conjunto de dados testes. De acordo com os elementos do INSA que efetuam os mapeamentos manuais aos dados, sugeriram que fosse utilizado os registos provenientes da autoridade competente ASAE, uma vez que esta autoridade também utiliza um vocabulário controlado interno a própria instituição, os seus relatórios possuem um número elevado de registos e são considerados dos relatórios que maior tempo consome para criar manualmente um mapeamento.

Portanto, tendo feita a escolha do parâmetro a analisar e possuindo um conjunto de registos para teste da ferramenta a ser desenvolvida, apenas resta investigar e selecionar ferramentas existentes que poderão auxiliar na análise léxica e gramatical dos dados.

4.4 Ferramentas de análise léxica e gramatical

Visto que estamos a lidar com dois vocabulários controlados, a possibilidade de ocorrerem variações da forma com que os dados são formados, pelo menos para o parâmetro analítico que irá ser mapeado para o FoodEx2, é mínima. Após uma análise preliminar a um conjunto de entradas provenientes da lista de dados da ASAE, os mesmos que irão ser utilizados para os testes desta ferramenta, verificou-se que seguiam uma estrutura bastante regular em que cada uma das entradas estava dividida em várias partes com diferentes níveis de detalhe, como pode ser verificado na Figura 15.

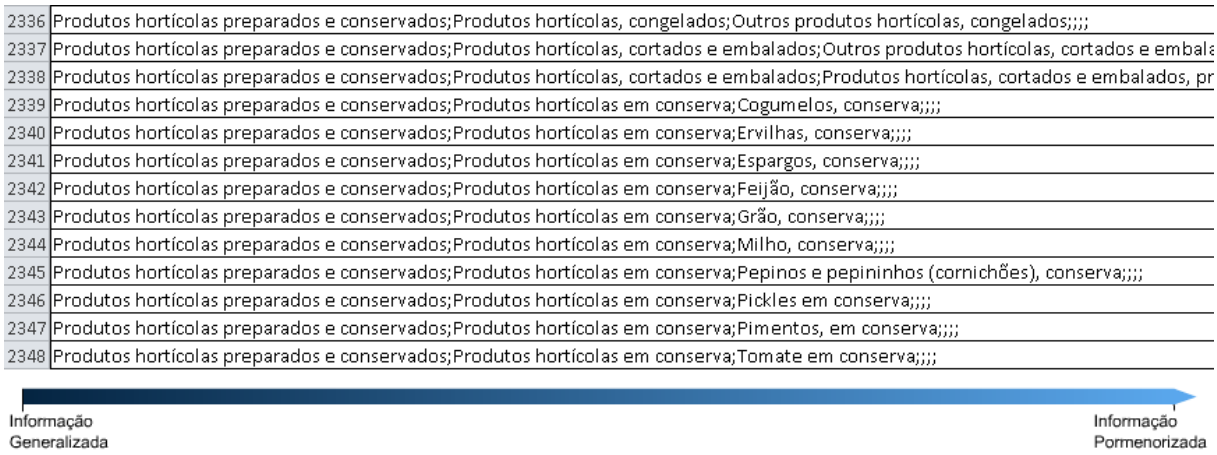


Figura 15. Conjunto de entradas que demonstram o tipo de estrutura do vocabulário utilizado pela ASAE.

O FoodEx2 também apresenta uma estrutura semelhante do vocabulário utilizado pela ASAE. Com este conhecimento adquirido ao comparar as duas linguagens pode-se concluir que para encontrar a correta correspondência entre os vocabulários, um processo terá de ser criado para percorrer as duas estruturas em forma de árvore e que ao mesmo tempo efetue uma comparação entre os diferentes níveis de modo a encontrar o último nó que seja a melhor correspondência entre as duas árvores.

O primeiro desafio que surgiu para a criação deste processo, foi o de organizar a informação proveniente da ASAE para uma estrutura que permitisse manusear os dados de uma certa forma em que se pudesse efetuar diversas operações e análises com maior eficácia possível. Uma investigação inicial revelou que a utilização de processos de análise léxica aos dados poderiam auxiliar na extração da informação necessária com que se pudesse trabalhar. Vários modelos foram encontrados como resultado da investigação, neles são descritos as várias

etapas e operações que poderão estar contidas em um processo de análise léxica a um conjunto de dados [11] [12]. Um modelo em específico não foi utilizado para a criação desta ferramenta pelo facto de possuir-se vocabulários já trabalhados no sentido de já terem uma estrutura definida, evitando assim aplicar certas operações aos dados. No entanto o modelo utilizado irá ser detalhado em um ponto mais a frente neste capítulo.

O seguinte objetivo a ser atingido seria a aquisição de meios existentes que fossem mais apropriadas para a realização das transformações e outros tipos de operações necessárias aos dados. Felizmente existe uma grande quantidade de ferramentas e bibliotecas que permitem efetuar as diferentes tarefas de análise léxica de dados.

4.4.1 Lex e Flex

Lex[11] é uma aplicação que tem como propósito de ser utilizada para criar analisadores léxicos. Estes analisadores léxicos permitem, a partir de um texto dado como entrada na aplicação, produzir uma série de *tokens*, que basicamente são um conjunto de caracteres que possuem um significado no contexto do texto em que foi extraído. Os *tokens* são criados com base nas regras utilizadas pelo analisador léxico, no caso do Lex estas regras são definidas por expressões regulares.

Para utilizar o Lex basta apenas criar um ficheiro de texto que deverá ser divido em 3 secções, onde a primeira é destinada para a definição de *macros* e ficheiros de cabeçalho em linguagem C; na segunda secção deverão estar presentes as regras criadas a partir de expressões regulares e código em C para operações adicionais; por fim, na última secção deverá conter puramente funções novamente em C que irão ser invocadas pelas regras criadas na secção anterior.

Normalmente o Lex é utilizado juntamente com uma outra ferramenta denominada de Yacc[12]. O Yacc utiliza o conjunto de *tokens* gerados pelo Lex e com eles tenta perceber o significado e a relação que os *tokens* têm entre si. Basicamente pode-se considerar o Yacc como um analisador de sintaxes.

[11] http://dinosaur.compilertools.net/#lex

[12] http://dinosaur.compilertools.net/#yacc

Estas duas aplicações possuem as funcionalidades necessárias para a criação das etapas iniciais da ferramenta de análise do vocabulário das autoridades competentes que se pretende desenvolver. No entanto, o Lex e o Yacc são aplicações consideradas antigas e desatualizadas, como tal foram desenvolvidas durante o seu tempo de existência outras aplicações que possuem as mesmas capacidades dos seus antecessores mas tendo em adição novas características que as tornam superiores. Estas aplicação mais recentes, designadas de Flex[13] e Bison[14] uma melhor opção para o desenvolvimento da ferramenta.

4.4.2 OpenNLP

O OpenNLP[15], ao contrário do Lex e Yacc, é uma biblioteca que permite efetuar um conjunto vasto de tarefas que são comuns para o processamento de linguagem natural. Para além das tarefas principais, como é o caso da criação de *tokens* e da reconhecimento de frases, este também permite efetuar algo denominado de *part-of-speech tagging* (POS Tagging), que tem a função de classificar gramaticalmente uma palavra presente no texto, isto é, indica se o *token* é um nome, verbo adjetivo, etc.. O POS Tagger classifica as palavras com base em um dicionário de *tags*, em que cada *tag* representa as várias classificações possíveis para um *token*, e juntamente com o modelo previamente treinado as *tags* serão aplicadas aos *tokens* correspondentes.

Existem ferramentas adicionais contidas no OpenNLP que possibilitam resolver problemas mais complexos, contudo pelo facto de possuir as mesmas funcionalidades das aplicações Lex/Flex e Yacc/Bison e ao mesmo tempo ter ao dispor instrumentos tão poderosos como é o caso POS Tagger, faria mais sentido utilizar o OpenNLP para a criação da ferramenta de mapeamento automático de vocabulários.

4.4.3 RapidMiner

RapidMiner[16] é um sistema extremamente poderoso que faz uso de processos de *data mining*

[13] http://dinosaur.compilertools.net/#flex

[14] http://dinosaur.compilertools.net/#bison

[15] http://opennlp.apache.org/

[16] http://rapid-i.com/content/view/181/190/lang,en/

para extrair informação contida em um conjunto de dados e posteriormente aplica transformações de forma a criar uma estrutura que seja de fácil compreensão. Estes mesmos processos de *data mining* podem ser aplicados também para *text mining*, área em que a ferramenta que pretende-se desenvolver se enquadra.

RapidMiner opera de forma desigual em comparação com as alternativas discutidas anteriormente. Ao contrário do *Lex &Yacc* e do OpenNLP que necessitam da produção de código para obter os resultados desejados, o RapidMiner pode ser utilizado sem codificar uma única linha de código, em virtude da coleção de operadores que possui. Cada um destes operadores podem ser vistos como objetos físicos, isto é, são objetos que o utilizador poderá interagir ao ponto de conseguir alterar os seus parâmetros de modo a obter diferentes resultados de acordo com que se pretende obter. Poderá também criar ligações entre eles, definindo assim quais os valores de saída de um certo operador deverão ser considerados como de entrada para o seguinte operador na relação.

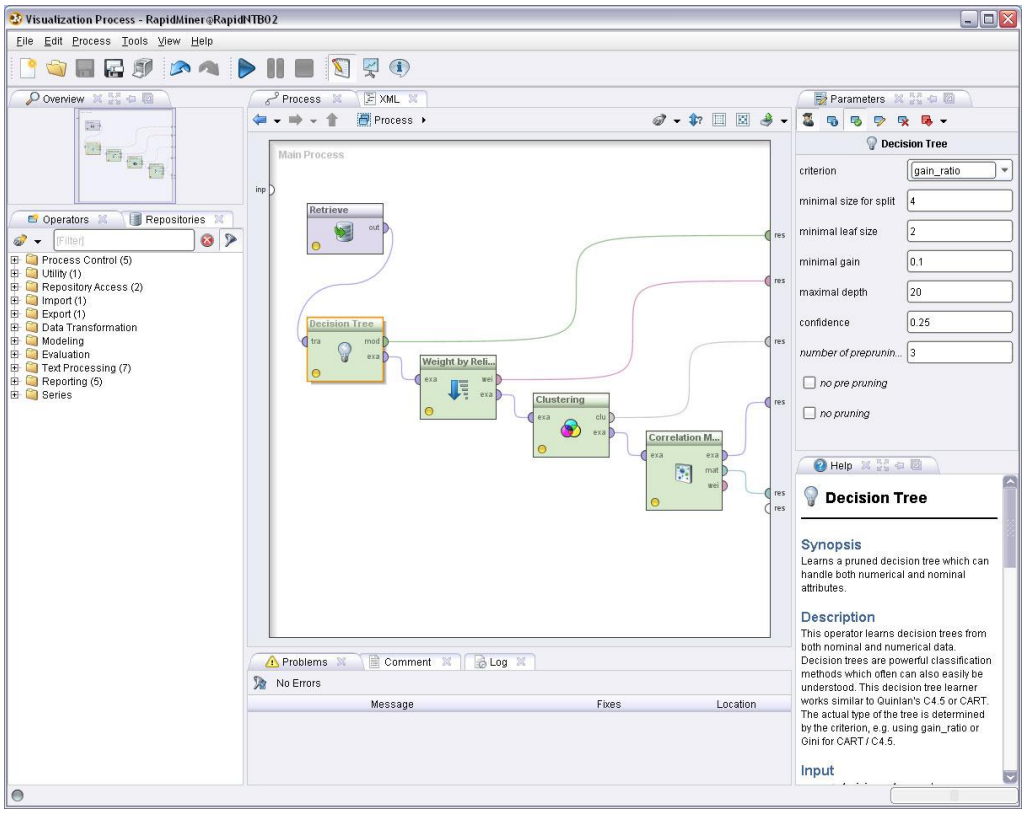


Figura 16. Interface gráfica que expõe a utilização de operadores para a produção de resultados.

Esta forma de programação trás duas grandes vantagens que foram identificadas após a utilização deste sistema. A primeira vantagem trata-se da possibilidade de expandir as

funcionalidades do RapidMiner com a simples adição de novos operadores, em oposição as ferramentas *Lex & Yacc* e OpenNLP que para se conseguir obter o mesmo resultado teriam de ser desenvolvidas estas novas funcionalidades ou algoritmos através da produção de código. A segunda vantagem é a capacidade de visualizar os resultados de um processo em segundos, evitando assim efetuar qualquer tipo de complicação e proceder-se posteriormente à execução de uma aplicação com os dados necessários para entrada. Isto tornou-se bastante útil para a fase inicial da produção da ferramenta para o mapeamento dos vocabulários em que várias alterações como, por exemplo, a utilização de diferentes conjuntos de valores para certos operadores por forma a obter melhores resultados.

OpenNLP, como escolha inicial, foi rapidamente substituído pelo RapidMiner para a implementação da ferramenta devido as suas vantagens enunciadas, como também por possuir o próprio OpenNLP e Flex integrado no seu sistema mas também pode ser expandido com adição de novas extensões. O RapidMiner é também reconhecido como uma das melhores ferramentas de análise[17], utilizada em numerosos projetos que abrange diversas áreas, e possui também uma grande comunidade por detrás a suporta-lo o que auxilia projetos que estejam pela primeira vez a entrar na área de *datamining*, como é o caso deste que está a ser discutido neste capítulo.

4.5 Outras ferramentas

Para além das ferramentas que resultaram da pesquisa efetuada, foram também utilizadas outras que auxiliaram na produção do instrumento de mapeamento automático. A EFSA ao anunciar que a nova versão Foodex2 deveria ser utilizada para o padrão SSD, esta também disponibilizou uma ferramenta com o nome de FoodEx2Browser que permite encontrar o código no formato Foodex2 para uma determinada descrição de um género alimentício. Uma vez que decidiu-se focar em apenas uma determinada categoria de alimentos contida na estrutura do Foodex2, esta ferramenta serviu de auxílio por forma a melhor compreender a estrutura e conteúdo dessa categoria. Esta ferramenta também dispõe de uma função que permite exportar a hierarquia para um ficheiro com o formato HTML e que foi utilizada para depois importar a informação de uma determinada categoria para o RapidMiner.

[17] http://rapid-i.com/content/view/404/1/

Normalmente, quando o sistema RapidMiner é adquirido pela primeira vez, este possui alguns operadores de *datamining* que vêm por omissão com o pacote, no entanto como já explicado anteriormente, para a tarefa proposta de mapear automaticamente vocabulários seria necessário operadores de *textmining*. Para adquirir estes novos tipos de operadores, uma nova pesquisa centrada por volta da aplicação RapidMiner foi realizada com o objetivo de encontrar extensões e *plugins* que satisfaçam a necessidade.

Como resultado, foram encontrados duas extensões e um *plugin* focados nas tarefas de *textmining*. As extensões Text Processing e Web Mining[18] foram encontradas e estas adicionam novos operadores que, primeiramente para o caso do Text Processing, permitem importar documentos de diversas fontes em diversos formatos, permitem também transformar o conjunto de dados com auxílio a diversas técnicas e filtros, como é o caso de *tokenization, stemming* e *stopword filtering.* Para o caso da extensão Web Mining, esta poderá levantar alguma confusão uma vez que aparentemente não seria necessário a utilização de utensílios para tratar de dados provenientes da Web, no entanto foi anteriormente referido que a ferramenta FoodEx2Browser possuía a opção de exportar o vocabulário do FoodEx2 e esta exportação faz com que um ficheiro no formato HTML seja criado com todo o vocabulário. Por forma a extrair a informação que está presente entre os elementos da linguagem HTML seria necessário uma ferramenta que fizesse tal tarefa, e com o conjunto de operadores presentes na extensão Web Mining, esta tarefa seria possível de ser realizada.

A extensão denominada de Information Extraction Plugin[19], ao contrário das anteriores, não foi adquirida através do *marketplace* associado ao RapidMiner, trata-se de uma extensão *opensource* que possui um conjunto de operadores que possibilitam extrair a informação considerada relevante em documentos importados, utilizando certos métodos de *datamining* já presentes no RapidMiner. Esta extensão traz a vantagem de ser possível efetuar uma melhor análise dos resultados que cada um dos processos produz.

4.6 Implementação

A primeira abordagem para implementação da ferramenta de mapeamento automático foi a de

[18] http://rapid-i.com/content/view/202/206/lang,en/

[19] http://sourceforge.net/projects/ieplugin4rm/

definir um domínio inicial com que se iria trabalhar.

O vocabulário FoodEx2 é constituído por 20 categorias onde cada uma representa diferentes tipos de alimentos. Estas categorias por sua vez são compostas por várias subcategorias por forma a abranger o maior número de alimentos possível, o que faz com que o FoodEx2 tenha mais detalhe que o seu antecessor. No entanto se optarmos por aceitar todas as categorias para o desenvolvimento da ferramenta de mapeamento, uma grande quantidade de regras e diferentes procedimentos terão de ser criados para cada uma das categorias, apenas para provar que o mapeamento automático é possível, onde o mesmo pode-se também provar ao aceitar apenas uma categoria.

A partir dos ficheiros que contém as entradas pertencentes a ASAE, fornecido pelo INSA para testes e consequentemente implementação da ferramenta, juntamente com uma breve análise ás diferentes categorias presentes no FoodEx2, decidiu-se por construir a ferramenta para um ponto inicial em que se iria focar apenas na categoria "Fruit and Fruit Products [A01BS]". Esta categoria, como o nome poderá indicar, possui os códigos para classificar produtos alimentares que sejam considerados principalmente frutos ou que estejam relacionados com frutos.

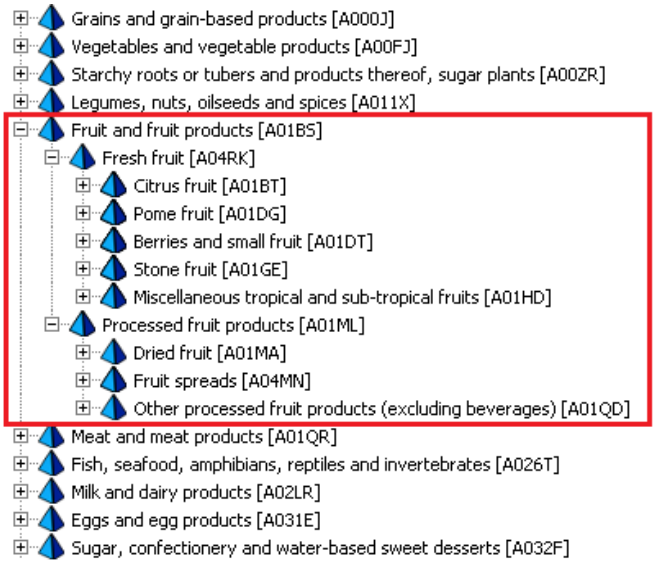


Figura 17. Categoria selecionada para o desenvolvimento da ferramenta para o mapeamento automático.

Ao definir-se um segmento exato do vocabulário com que se irá focar, grande parte das entradas presentes no ficheiro de teste terão de ser excluídas, visto que, neste ficheiro existem cerca de 2500 registos onde a maioria destes registos não estão relacionados com a categoria definida. A razão pela qual estes registos deverão ser excluídos é porque, ao serem

adicionados ao RapidMiner para o processamento da informação, os operadores irão aplicar os métodos independentemente da informação que recebem, no entanto ao fim de todos os operadores terem concluído as suas funções, estes dados não irão satisfazer a primeira regra de pertencer a categoria de frutas e produtos relacionados com frutas.

A partir do ficheiro com as entradas da ASAE, 30 registos foram escolhidos. Com o domínio agora estabelecido para ambos os lados, vocabulário e dados de teste, podia-se proceder com a implementação inicial no sistema RapidMiner. Antes de se apresentar o método de desenvolvimento e a solução obtida, primeiro será detalhado o diagrama de arquitetura que descreve os vários processos criados para resolver o problema em causa. Este diagrama e a sua descrição não engloba a integração e interação entre a ferramenta de mapeamento e a plataforma PT.ON.DATA, este apenas abrange os vários procedimentos criados no RapidMiner e alguma lógica adicional.

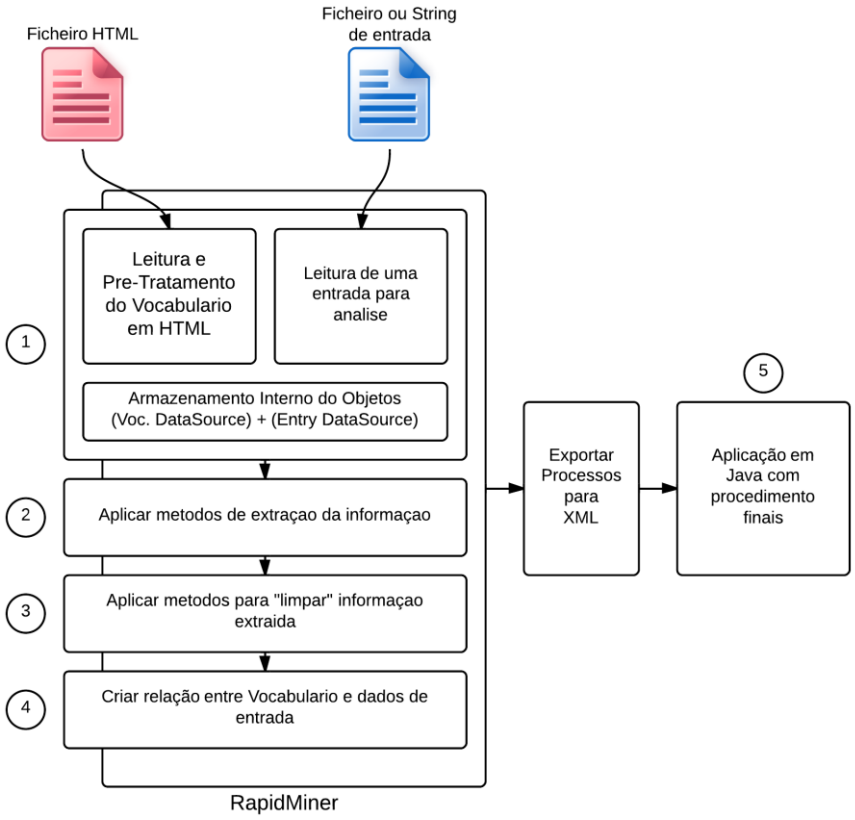


Figura 18. Diagrama de arquitetura para a ferramenta de mapeamento.

O desenvolvimento da ferramenta começa com a criação de vários processos que executam tarefas distintas entre eles. No âmbito do sistema RapidMiner, cada processo representa um conjunto de operações que são aplicadas a determinados dados. A Figura 18 apresenta os 4

diferentes processos que constituem o módulo de análise e tratamento da informação. O primeiro módulo, identificado pelo número 1 na Figura 18, possui duas tarefas que partilham o mesmo objetivo, a leitura e armazenamento interno dos dados para análise. Antes de detalhar cada um das tarefas, resta apenas clarificar o que entende-se por armazenamento interno.

Para que os operadores presentes no RapidMiner consigam tratar a informação que recebem, esta informação necessita de estar armazenada de uma certa forma que permita que estes mesmos operadores saibam como agir com os dados que recebem. Entende-se então por armazenamento interno o procedimento de arquivar a informação em objetos designados de DataTables.

DataTables são estruturas pertencentes ao núcleo do RapidMiner e podem ser interpretadas como tabelas normais, onde as colunas destas tabelas são designadas de atributos, que por sua vez têm o propósito de adicionar mais informação relativamente aos dados. As DataTables serão utilizadas por todos os processos e, a medida que progridem ao longo destes mesmos processos, os dados contidos nelas serão cada vez mais refinados até chegar ao ponto de conseguir-se efetuar um mapeamento.

Retomando a descrição do processo inicial, a primeira tarefa contém um operador que faz a leitura de um ficheiro em HTML que deverá ter presente a informação da categoria “Fruit and Fruit Products [A01BS]” do vocabulário FoodEx2. Para que o processo não consuma demasiado tempo, o ficheiro HTML que é introduzido deverá apenas conter a categoria desejada. O operador de leitura transforma o ficheiro num objeto que permite a que o operador seguinte, desta vez para a tarefa de HTML Processing, consiga extrair a informação que se encontra entre os elementos HTML.

Após receber o conteúdo do ficheiro ainda com o formato HTML, o operador de Web Mining com o nome de “Extract Content” executa o *parse* dos dados e retira o conteúdo que está entre as diferentes *tags*. A informação extraída é semelhante aquela que está apresentada na Tabela 6, e esta é armazenada em uma DataTable com auxílio do operador Store, que também faz parte do núcleo do RapidMiner.

Tabela 6. Informação retirada do primeiro processo que constitui a ferramenta de mapeamento.

Code	Level	Name	Parent Code
A01BS	1	Fruit and fruit products	ROOT

A segunda tarefa que este processo executa em paralelo a extração da informação em do vocabulário, é mais simplificada. Um ficheiro de texto que contém várias entradas para serem mapeadas é lido pelo operador de Text Processing com a designação de "Read Document", e tal como na tarefa anterior, este operador transforma o conteúdo do ficheiro para objetos que são de fácil manipulação pelos restantes operadores. As entradas agora em objetos são diretamente armazenadas para DataTables contendo apenas um só atributo que representa cada uma das entradas que quaisquer alterações, isto porque o processo que se segue está destinado e possui todos os operadores necessários para analisar e consequentemente transformar os dados.

O segundo processo, identificado com o número 2 na Figura 18, inicia com a leitura da DataTable que tem armazenada as entradas que serão mapeadas. Este processo utiliza uma grande variedade de operadores que têm origem interna e externa ao RapidMiner, como consequência surgiu uma grande dificuldade em transformar os dados para o formato correto requeridos por alguns dos operadores, como por exemplo, os operadores que pertencem ao núcleo do RapidMiner recebem para leitura e produzem resultados em formatos diferentes dos operadores das extensões de Text Mining e Information Extraction, logo foi necessário utilizar um conjunto de operadores que permitem realizar a conversão correta. À parte dos operadores de conversão e após possuir as entradas retiradas da DataTable, o processo segue com a tarefa de transformar todos os caracteres para maiúsculas e remover aqueles que representam pontuação no texto, por forma a tentar eliminar ao máximo a ocorrência de erros nas comparações de *strings*.

Com as entradas contendo apenas palavras em sequência, pode então seguir-se para a utilização de operadores mais centrados na área de Text Mining. O primeiro operador a ser executado é o "Filter Stopwords" com o propósito de eliminar das entradas as palavras consideradas *stopwords.* Entende-se por *stopword* as palavras que pretendem criar relações entre ideias mas que por si só não adicionam mais informação ao texto, como é o caso das palavras "de", "a", "o" e "que". Infelizmente o RapidMiner não possui um operador para filtrar *stopwords* portuguesas, no entanto este possui um operador alternativo que faz recurso

a um dicionário de *stopwords* criado manualmente ou adquirido de outras fontes[20]. Existem também técnicas mais avançadas que permitem a criação automática de uma lista de *stopwords* de acordo com o texto a ser analisado [13], contudo estas técnicas não serão necessárias para esta fase inicial de desenvolvimento da ferramenta de mapeamento.

De seguida é feita a separação de cada uma das palavras para *tokens* com a utilização do operador "Tokenize". Estes *tokens* gerados continuam a representar palavras mas permitem que os próximos operadores possam aplicar transformações a cada uma das palavras de forma individual. Os *tokens* seguem assim para o último operador de Text Mining designado de "Stemmer". Este operador tenta obter a raiz das palavras que recebe, eliminando os sufixos presentes que poderão ter originados de algum tipo de derivação. Como é possível perceber, este operador está bastante dependente da linguagem em que as palavras encontram-se escritas, mas a extensão de Information Extraction contém este mesmo operador em diversas implementações para um conjunto de linguagens, em que uma delas é a linguagem portuguesa. Após concluída a tarefa de *stemming* executada para cada um dos *tokens*, os dados encontram-se então prontos para a criação de uma nova DataTable mas desta vez com a informação relevante visível.

Os processos seguintes, identificados pelos números 3 e 4 na Figura 18, foram criados para remoção de atributos adicionais que surgiram com a utilização dos vários operadores e que não adicionam informação relevante para o mapeamento, como é o caso do atributo que identifica o tipo de dados do *token* respetivo indicando se trata-se de um valor numérico ou texto. Quanto ao processo número 4, este foi concebido para realizar um mapeamento direto caso fosse possível de se efetuar com os dados que possuísse nesta etapa. Este último processo raramente conseguiria mapear um conjunto de valores por razões que serão explicadas mais a frente, no entanto a sua existência permite atenuar a execução de tarefas mais complexas para o mapeamento automático, ao tratar imediatamente dos mapeamentos que são obviamente diretos com auxílio de operadores básicos de comparação.

4.6.1 Aplicação Java

Existe algumas razões pela qual a utilização apenas do sistema RapidMiner não é o suficiente para a criação da ferramenta de mapeamento. Para que fosse possível introduzir os dados

[20] http://snowball.tartarus.org/algorithms/portuguese/stop.txt

provenientes da plataforma PT.ON.DATA, de uma forma automática, para dentro dos processos criados no RapidMiner, seria necessário algo que fizesse a ligação entre estes dois sistemas. Seria evidente que uma aplicação teria de ser desenvolvida para transportar os dados, no entanto por fim a evitar os pontos de falha que poderão surgir ao adicionar mais uma ferramenta no modelo, decidiu-se por unir o módulo pertencente ao RapidMiner com a aplicação que iria ser construída.

Juntamente com o sistema em si, RapidMiner também dispõe de bibliotecas que permitem replicar exatamente o que foi obtido ao utilizar a interface gráfica. Todos os operadores podem ser acedidos por código permitindo recriar os mesmos processos, mas as bibliotecas também possuem uma funcionalidade mais poderosas que possibilita a importação dos processos criados no RapidMiner.

A primeira etapa da implementação desta aplicação iniciou com a exportação dos processos presentes no RapidMiner para ficheiros XML, onde no conteúdo destes ficheiros estava descrito os operadores utilizados e os parâmetros definidos para cada um deles. Procedeu-se depois com a criação de um projeto em Java, visto que a API do RapidMiner foi criada para este ambiente, e ao adicionar os objetos da API necessários para leitura dos ficheiros, a aplicação estaria assim pronta para receber os dados para mapeamento.

Decidiu-se por implementar uma comunicação através de *sockets* utilizando o protocolo TCP/IP. Este canal de comunicação será utilizado quando um cliente efetua um pedido para encontrar o mapeamento de um certo valor, que por sua vez será então encaminhado da plataforma em forma de *string* para a aplicação. Após receber um valor, os processos que estão carregados em memória serão postos em execução tal como seria feito no sistema RapidMiner.

Apesar de aparentar possuir-se os requisitos necessários para efetuar um mapeamento automático, existe um problema que surgiu desde a utilização do RapidMiner para a implementação da ferramenta. O problema está relacionado com o vocabulário FoodEx2, mais concretamente associado com a linguagem com que foi criado. O vocabulário atualmente está definido em inglês mas como os dados provenientes das autoridades competentes estão em português, será impossível efetuar um mapeamento correto.

Por forma a resolver o problema da diferença de linguagens, decidiu-se por utilizar um serviço que fizesse a tradução do vocabulário para a linguagem portuguesa. Esta abordagem aparentou ser viável uma vez que a secção selecionada do vocabulário possui essencialmente nome de frutos ou nome de produtos relacionados com frutos, logo uma tradução não irá produzir um resultado que não o desejado. Para que seja mais claro perceber a lógica encontrada, pode-se considerar o valor "Orange" proveniente do vocabulário, a única tradução possível para este valor será a palavra "Laranja".

Fez-se uso do serviço disponibilizado pela Microsoft intitulado de Microsoft Translator[21] por essencialmente possuir a vantagem de ser gratuito, ter um sistema de tradução que produz resultados bastante aceitáveis e uma API simples de utilização.

Com uma solução encontrada para o problema das linguagens, o vocabulário poderia ser então carregado para memória, seguido da invocação dos métodos via HTTP REST disponibilizados pelo serviço de tradução, resultando na produção de uma DataTable com as várias categorias do vocabulário em português.

Ao fim de todos os processos concluírem as suas tarefas resta apenas aplicar o algoritmo para encontrar o melhor mapeamento possível. O algoritmo utilizado é bastante simples e consiste na análise de cada um dos *tokens* com origem da *string* de entrada, com o objetivo de se encontrar um *token* semelhante armazenado na DataTable que pertence ao vocabulário. Caso um ocorra uma comparação que prove que ambos os *tokens* são semelhantes, é adquirido o nível a que pertence a categoria que representa o *token* comparado com sucesso. Este nível, que indica se trata-se da raiz ou um dos nós da hierarquia, está presente em um dos vários atributos da DataTable. Este processo é repetido até percorrer o número máximo de níveis possíveis na hierarquia, retornando o código FoodEx2 do nó com maior profundidade encontrada.

Para situações em que não é encontrado um *token* semelhante e o nó corrente possui filhos, é feita uma nova tradução aos *tokens* originais do vocabulário que ainda estão em inglês e que podem ser obtidos através de um dos atributos da DataTable. O que difere da tradução inicial é o facto de se utilizar o método "GetTranslationsArray" do serviço de tradução que retorna

[21] http://www.microsoft.com/en-us/translator/developers.aspx

um *array* de traduções alternativas. O processo de comparação será realizado novamente, no entanto será tido em consideração o conjunto de traduções alternativas.

Uma vez obtido um código FoodEx2 que represente uma correspondência encontrada com sucesso, este código será enviado para a plataforma de forma a ser apresentado ao cliente que requisitou o mapeamento.

4.7 Testes e Resultados

Os vários testes efetuados ao longo do processo de desenvolvimento foram sempre realizados em conjunto com os elementos da equipa pertencente ao INSA. Devido a constante pressão para conseguir-se concluir o protótipo inicial, testes de aceitação e usabilidade eram os únicos meios que se tinham de modo a certificar o que estaria a ser desenvolvido era exatamente o que se pretendia obter. Pelo menos um teste de aceitação e de usabilidade era realizado por mês, em conjunto com um *workshop* efetuado juntamente com algumas autoridades portuguesas para que pudessem fornecer algum *feedback* a respeita da utilização que fizeram da plataforma PT.ON.DATA. Como resultado dos testes, as próximas etapas e iterações seriam sempre definidos até a próxima reunião definida.

Para a ferramenta de mapeamento desenvolvida foram realizados testes que pretendiam avaliar a qualidade e precisão dos resultados obtidos. Como referido anteriormente, a categoria “Fruit and Fruit Products [A01BS]” do vocabulário FoodEx2 foi utilizada e continha todos os código que se pretendia obter. Como dados de teste, foram utilizadas 30 entradas do relatório fornecido pela ASAE, previamente mapeadas pelos técnicos do INSA, logo seria possível saber se o mapeamento feito pela ferramenta estaria correto ou não. Após realizado o teste à ferramenta, obtiveram-se os seguintes resultados apresentados na Figura 19.

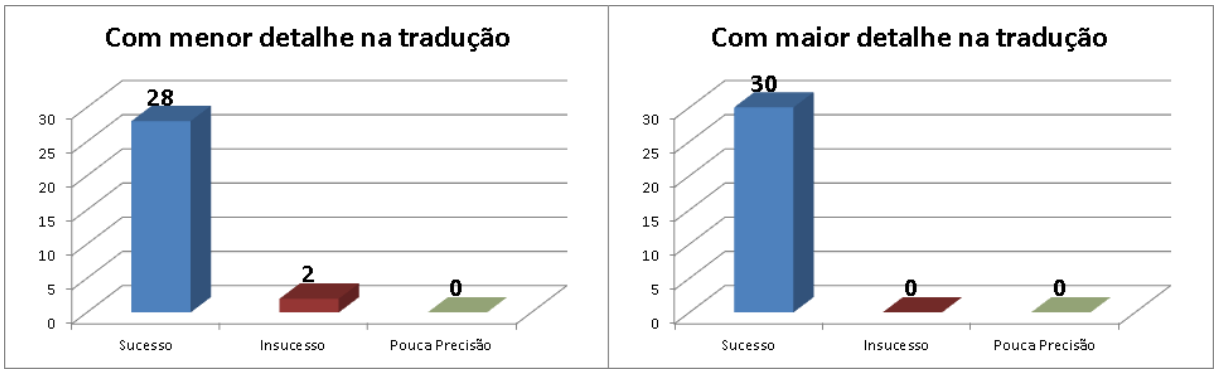


Figura 19. Gráfico de resultados para a primeira fase (esquerda) e segunda fase (direita) de testes de mapeamento.

Como é possível verificar pela Figura 19, na primeira fase de testes identificou-se que duas entradas não conseguiram ser mapeadas pela ferramenta. Após uma análise efetuada aos diferentes processos, confirmou-se que o problema que originou este insucesso tem origem na tradução da categoria do vocabulário. Na primeira fase de testes foi utilizada uma tradução direta, isto significa que apenas uma palavra gerada da tradução para português foi utilizada na comparação de valores. O erro ocorreu ao tentar mapear as duas entradas que estavam relacionadas com frutas cristalizadas. Ao utilizar apenas uma tradução direta, o serviço devolvia como tradução de "cristalizada" a palavra "crystalized", no entanto a correspondência correta para frutas cristalizadas seria a subcategoria com a descrição de "Candied Fruits [A01PS]". Para resolver este problema decidiu-se por utilizar o método do serviço de tradução "GetTranslationsArray" que, em vez de retorna apenas uma palavra como resultado, retorna um conjunto de candidatos para a tradução de "cristalizadas".

Como é possível visualizar através do segundo gráfico presente na Figura 19, as duas entradas que não foram mapeadas anteriormente agora encontram-se no conjunto de entradas mapeadas com sucesso. Apesar de não se verificar mas para ao mesmo tempo esclarecer o último pormenor destes testes, as entradas poderiam ser classificadas como mapeadas com "Pouco Precisão", isto significaria que uma entrada possuía informação suficiente para obter-se o nó correspondente com maior nível no vocabulário mas que por algum motivo esse nó não foi encontrado e como resultado o código do pai do nó foi escolhido como mapeamento correto.

Com base nos resultados apresentados acima, pode-se concluir que em geral a ferramenta produz mapeamentos corretos e que, com mais desenvolvimento e tempo investido, esta

poderá tornar-se em um utensílio bastante importante para realização das tarefas principais presentes no processo de submissão de dados à EFSA.

4.8 Integração com a plataforma PT.ON.DATA

A forma com que é feita a comunicação entre a ferramenta de mapeamento e a plataforma web é através de *sockets* onde as duas aplicações encontram-se na mesma máquina a partilhar os dados de entrada e resultados obtidos. Apesar de esta descrição dar uma certa ideia da interação que estas duas aplicações têm entre si, a comunicação que o supervisor ou administrador efetua com a plataforma por forma a invocar a ferramenta não é evidente.

Uma interface gráfica foi criada de modo a simular uma interação direta entre o utilizador e a ferramenta de mapeamento. Esta interface está localizada na secção administrativa, indicando que apenas os utilizadores com perfil de supervisor ou acima poderão ter acesso a ela. Com auxílio de um conjunto de filtros, o utilizador poderá rapidamente selecionar qual a entidade e que parâmetro analítico possui um valor para mapear. Após a correta seleção dos filtros, um pedido com o valor selecionado será enviado a aplicação a ser executada em *background* enquanto a plataforma web fica a espera de uma resposta para apresentar ao cliente. Visto que se trata de uma ferramenta de auxílio à decisão, o utilizador irá visualizar o possível código FoodEx2 devolvido como resposta e irá decidir se deseja realmente substituir o parâmetro analítico pelo código.

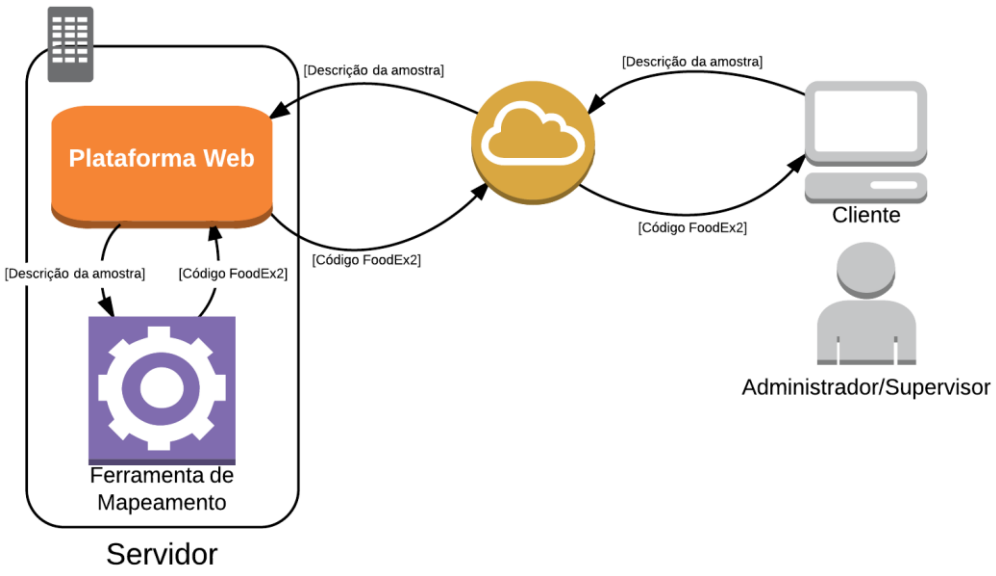


Figura 20. Workflow que apresenta a interação do utilizador com a ferramenta de mapeamento.

Embora esteja visível a interface gráfica que comunica indiretamente com a ferramenta, o estado de desenvolvimento com que se encontra o processo de mapeamento automático levou a que apenas os programadores da plataforma tenham acesso a ferramenta em si.

Conclusão

Com a finalização dos vários tópicos expostos nos dos capítulos anteriores, resta apresentar uma síntese dos pontos mais relevantes e apresentar ao mesmo tempo as conclusões que foram obtidas como resultado do desenvolvimento deste projeto.

Será realizada uma avaliação dos objetivos que foram estipulados na proposta do projeto CFP/EFSA/DATEX/2011/01 criado pela EFSA para que se tenha uma apreciação geral de quais os objetivos que foram ou não devidamente cumpridos. De seguida serão apresentadas as possíveis contribuições que a solução encontrada poderá trazer para futuros projetos que possuem os mesmos objetivos. Por último são apresentados os vários pontos do projeto que acreditamos possuírem potencial de evolução ou melhoria.

5.1 Objetivos alcançados

Um conjunto de objetivos estava definido para os estados membros que aceitassem a proposta de projeto da EFSA. Os objetivos, de um modo geral, requeriam que fosse construído um sistema que fizesse a adequação de dados analíticos presente em vários sistemas nacionais ao modelo de dados Standard Sample Description.

Para além dos objetivos originais ao projeto, ao alongo do processo de desenvolvimento surgiram outros que estariam mais orientados às autoridades competentes do que aos dados, como é o caso da criação de interfaces e implementação de ferramentas para facilitar as tarefas de introdução dos dados. Destes objetivos resultou a criação do módulo de importação de relatório em formato Excel que facilita a aquisição dos dados analíticos de outros sistemas. Para as entidades que não possuem um sistema informático mas os seus dados analíticos necessitam de estar na base de dados do sistema, foi também desenvolvido o módulo Formulário Web, que possibilita a inserção de dados respeitando o modelo SSD.

Após finalizada a implementação do sistema base que engloba todas as funcionalidades indispensáveis, pode-se concluir que os objetivos principais foram alcançados. Foi também possível obter uma confirmação por parte da EFSA de que as várias funcionalidades teriam sidas implementadas com sucesso, uma vez que os dados analíticos com o padrão SSD submetidos ao DCF para aprovação foram aceites.

Objetivos secundário também foram estabelecidos, como é o caso de desenvolver uma ferramenta para auxiliar no mapeamento de vocabulário livre para vocabulário controlado especificado pelo padrão SSD. Como resultado, foi criada uma ferramenta capaz de selecionar corretamente um código do vocabulário FoodEx2, dada uma descrição criada pela ASAE. Apesar de esta ferramenta estar restringida para mapear uma categoria do FoodEx2, o facto de fazê-lo corretamente para uma categoria completa prova que, ao expandir os processos de modo a aceitar as restantes categorias, o mesmo irá verificar-se para todo o vocabulário.

5.2 Contribuição

Da mesma forma com que outros projetos criados pelos estados membros contribuíram para o desenvolvimento da plataforma PT.ON.DATA, o processo e os aspetos relacionados com a criação deste projeto poderá não só ajudar a aperfeiçoar os que já existem, como também poderá servir de base para os estados membros que estejam a implementar algo de raiz.

Algumas das soluções existentes detalham que uma próxima etapa que deveriam tomar seria o desenvolvimento de um módulo orientado a *web*, como é o caso da entidade alemã BVL que propõe criar uma aplicação para assistir na transição de versões do padrão SSD [5]. A plataforma PT.ON.DATA e os seus constituintes poderão dar uma ideia do que conseguirão obter ao caminharem para o ambiente *web*.

Os desafios e resultados encontrados ao tentar resolver o problema de mapeamento de vocabulários que se verifica na grande parte dos projetos propostos pelos estados membros, contribuem de forma positiva ao apresentar uma possível solução.

Essencialmente, a plataforma PT.ON.DATA tentou ser o próximo passo que os projetos já existentes poderiam tomar.

5.3 Trabalho Futuro

Com o anúncio feito pela EFSA a respeito de uma nova versão do padrão SSD que deverá ser adotada pelos estados membros enquadrados no projeto, este seria a necessidade principal a ser considerada como implementação futura.

A principal função da EFSA é a de obter dados analíticos com o máximo de detalhe possível para que possam avaliar de uma forma mais precisa a situação alimentar nos vários países. A nova iteração do padrão SSD vem especialmente fornecer um maior nível de detalhe aos relatórios que forem criados com destino à EFSA, através da adição de novos vocabulários e ao aprimorar os já existentes.

Posto isto, a adaptação da plataforma PT.ON.DATA e da sua base de dados para o novo padrão seria o próximo procedimento de maior importância a ser executado.

Quanto aos vários módulos presentes na plataforma, grande parte deles poderão ser alvo de melhorias, não só no sentido de tentar obter melhor performance, visto que muitas das vezes trabalha-se com grandes volumes de dados, mas também no sentido de continuar a facilitar as tarefas realizadas pelos utilizadores do sistema, como é o caso do mapeamento manual e a inserção de dados através do formulário *web*.

A ferramenta de mapeamento automático deverá também sofrer alterações de modo a expandir o domínio de funcionamento com que se encontra no momento. Deverá ser possível efetuar o mapeamento, pelo menos, para todo o vocabulário FoodEx2, o que implica aceitar todas as suas categorias e subcategorias. Este vocabulário é um dos mais importantes e mais árduo de se mapear porque requer a análise de texto livre criado por autoridades que não possuem linguagens controladas internamente. O que se poderia propor como melhoramento futuro para esta ferramenta seria, por exemplo, a utilização de melhores algoritmos de comparação de texto como é o caso do Levenshtein Distance Algorithm [14] e Boyer–Moore String Search Algorithm [15], de modo a melhor identificar uma melhor correspondência. Também por forma a obter melhores resultados, um diferente serviço de tradução do vocabulário que está em inglês para português deveria ser utilizado. Este serviço terá de possuir um dicionário interno mais completo de ambas as linguagens.

Bibliografia

[1] "GUIDANCE OF EFSA: Standard sample description for food and feed," *EFSA Journal 2010; 8(1):1457,* 2010.

[2] "Evaluation of the FoodEx, The food classification system applied to the development of the EFSA Comprehensive European Food Consumption Database," *EFSA Journal 2011; 9(3):1970,* 2011.

[3] "GUIDANCE OF EFSA: Guidance on Data Exchange," *EFSA Journal 2010; 8(11):1895,* 2010.

[4] O'Dea, E., Webster, S., McCoy, D., "Electronic Transmission of Chemical Occurrence Data in Ireland," *Supporting Publications 2012:EN-313,* 2012.

[5] Gunter, S., Frost, M., "Electronic Transmission of Chemical Occurrence Data CFP/EFSA/DATEX/2009/01," 2011.

[6] Gunter, S., Frost, M., "Scrum with XP," *informit.com,* 2002.

[7] Kniberg, H., "Scrum and XP from the Trenches," *C4Media,* 2007.

[8] Cockburn, A., Williams L., "The Costs and Benefits of Pair Programming," *Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA,* 2001.

[9] Arisholm, E., Gallis, H., Dybå, T., Sjøberg, D., "Evaluating Pair Programming with Respect to System Complexity and Programmer Expertise," *IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. 33, NO. 2,* 2007.

[10] Crinnion, J., Evolutionary systems development: A practical guide to the use of prototyping within a structured systems methodology, Plenum Press, 1991, p. 18.

[11] Thurmair, G., Aleksic, V., Schwarz, C., "Large-scale lexical analysis, European Language Resources Association," 2012.

[12] Jungermann, F., "Information Extraction with RapidMiner," *Artificial Intelligence Group, TU Dortmund,* p. 56, 2009.

[13] Lo, R., He, B., Ounis, I., "Automatically Building a Stopword List for an Information Retrieval

System," 2005.

[14] Black, P., ""Levenshtein distance", in Dictionary of Algorithms and Data Structures," CRC Press LLC, 1999. [Online]. Available: http://xlinux.nist.gov/dads//HTML/Levenshtein.html. [Acedido em 2013].

[15] Boyer, R., Moore, J., "A Fast String Searching Algorithm," *Communications of the ACM,* 1977.

Apêndices

Apêndice A - WBS