

Universidade do Minho

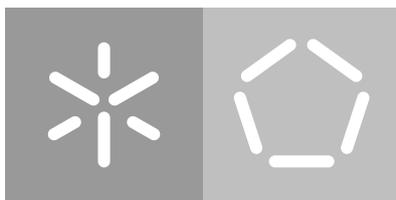
Escola de Engenharia

Departamento de Informática

Rui Filipe Ferreira Araújo

**Especificação e implementação de
um workflow numa plataforma para
classificação e avaliação documental**

Março 2021



Universidade do Minho

Escola de Engenharia

Departamento de Informática

Rui Filipe Ferreira Araújo

**Especificação e implementação de
um workflow numa plataforma para
classificação e avaliação documental**

Relatório de dissertação

Mestrado em Engenharia Informática

Trabalho efetuado sob a orientação do

Professor Doutor José Carlos Ramalho

Março 2021

AUTHOR COPYRIGHTS AND TERMS OF USAGE BY THIRD PARTIES

This is an academic work which can be utilized by third parties given that the rules and good practices internationally accepted, regarding author copyrights and related copyrights.

Therefore, the present work can be utilized according to the terms provided in the license bellow.

If the user needs permission to use the work in conditions not foreseen by the licensing indicated, the user should contact the author, through the RepositóriUM of University of Minho.

License provided to the users of this work



Attribution-NonCommercial

CC BY-NC

<https://creativecommons.org/licenses/by-nc/4.0/>

STATEMENT OF INTEGRITY

I hereby declare having conducted this academic work with integrity. I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration.

I further declare that I have fully acknowledged the Code of Ethical Conduct of the University of Minho.

Signature:

AGRADECIMENTOS

Este espaço é dedicado a todos aqueles que tornaram esta caminhada muito mais especial, no qual pretendo transmitir um profundo sentimento de gratidão e reconhecimento para com todos.

Agradeço ao meu orientador, Professor Doutor José Carlos Ramalho, não só pelo apoio, dedicação e disponibilidade demonstrados, por forma a ultrapassar as dificuldades e dúvidas que iam surgindo, mas também pelo seu espírito humanista, o qual permitiu que se estabelecesse um excelente ambiente de trabalho.

Aos meus amigos, Miguel Brito, Leandro Fernandes, Eduardo Machado e Paulo Caldas, por todos os bons momentos que passamos, momentos esses que guardo com imenso carinho e saudade.

Aos meus amigos José Carlos Martins, Frederico Pinto e Alexandre Teixeira, os quais me acompanharam no desenvolvimento do projeto, sendo parte integrante desta plataforma. A vocês, muito obrigado.

Um agradecimento especial à minha namorada, Elsa Pereira, pela paciência e apoio prestado nesta etapa, bem como pela sua ajuda na hora da revisão do relatório.

Um agradecimento especial para a minha família, nomeadamente para os meus pais, José e Clementina, os quais me apoiaram desde o início da caminhada e acreditaram em mim, incondicionalmente, mostrando-me que, com esforço e dedicação, tudo é possível. E aos meus irmãos, Catarina e Daniel, pela paciência que tiveram durante este percurso.

Àqueles que, de uma forma ou de outra, me deram apoio e ânimo, contribuindo para a realização do presente projeto.

A todos, o meu sincero agradecimento.

RESUMO

A substituição do papel pelo formato digital nas instituições e empresas tornou-se uma prática comum, sendo que algumas já extinguiram a utilização de formatos analógicos. As políticas europeias incentivam a que sejam adotadas medidas para a redução de papel. Desta forma, a administração pública tem abandonado a utilização de suportes analógicos, substituindo-os pelo formato digital, tendo as entidades públicas passado a prestar os seus serviços e a disponibilizar a documentação de forma eletrónica.

A Plataforma “CLAV - Classificação e avaliação da informação pública” desenvolvida pela DGLAB pretende disponibilizar instrumentos, tal como a Lista Consolidada; a mediação desmaterializada da produção de tabelas de seleção; e a prestação de um serviço automatizado de controlo da eliminação da informação pública. A adoção de esquemas de metainformação para a interoperabilidade permite a disponibilização de uma linguagem comum aos vários organismos da Administração existente na Lista Consolidada, através da plataforma CLAV, permitindo ainda a integração com sistemas de informação organizacionais e a troca de informação entre entidades.

Em concordância com os objetivos delineados para o desenvolvimento do projeto, realizou-se um estudo sobre abordagens idênticas ao CLAV noutros países, onde foram identificadas as formas de classificação desenvolvidas e as tecnologias utilizadas para essa implementação, fazendo uma comparação com a abordagem de Portugal. O plano de trabalho do projeto dividiu-se então em três fases. Inicialmente realizou-se um estudo teórico acerca da abordagem de Andaluzia (Espanha) e a de Portugal. Seguiu-se com o desenvolvimento de um modelo gráfico acompanhando as especificações do *Business Process Model and Notation* e, por fim, a implementação e o desenvolvimento desse modelo na plataforma CLAV.

Verificou-se que, após a implementação do *workflow* para a gestão dos pedidos, este tornou todo o processo mais simples e rápido para os responsáveis pelo tratamento destes, uma vez que todo o processo é realizado através da plataforma, não havendo a necessidade de transitar documentos físicos.

Palavras-Chave: Administração Pública, CLAV, Web Semântica, Workflow.

ABSTRACT

The substitution of paper for digital format in institutions and companies has become a common practice, and some have already extinguished the use of analog formats. European policies encourage measures to reduce paper. In this way, the public administration has abandoned the use of analogue media, replacing them with the digital format, with public entities now providing their services and making documentation available electronically.

The Platform “CLAV - Classification and Evaluation of Public Information” developed by DGLAB intends to provide instruments, such as the Consolidated List; the dematerialized mediation of the production of selection tables; and the provision of an automated service to control the elimination of public information. The adoption of metadata schemes for interoperability allows a common language to be made available to the various Administration bodies in the Consolidated List, through the CLAV platform, also allowing integration with organizational information systems and the exchange of information between entities.

In accordance with the objectives outlined for the development of the project, a study was carried out on approaches similar to CLAV in other countries, where the forms of classification developed and the technologies used for this implementation were identified, making a comparison with the approach of Portugal. The project’s work plan was then divided into three phases. Initially, a theoretical study was carried out on the approach of Andalusia (Spain) and that of Portugal. This was followed by the development of a graphic model following the specifications of the Business Process Model and Notation and, finally, the implementation and development of this model on the CLAV platform.

It was found that, after the implementation of the workflow for order management, it made the whole process simpler and faster for those responsible for handling these orders, since the whole process is carried out through the platform, with no need to transit physical documents.

Keywords: CLAV, Public Administration, Semantic Web, Workflow

CONTEÚDO

1	INTRODUÇÃO	1
1.1	Contexto	1
1.2	Motivação	3
1.3	Objetivos	3
1.4	Metodologia	4
1.5	Extrutura do Documento	4
2	ESTADO DA ARTE	6
2.1	Andaluzia (Espanha)	6
2.1.1	Tabela Classificatória	6
2.1.2	Classificação Funcional	7
2.1.3	Classificação Orgânica	10
2.1.4	Solução apresentada	11
2.1.5	Tecnologias e implementação	12
2.2	Portugal	13
2.3	Comparação síntese	15
3	CLAV	16
3.1	Web Semântica	16
3.1.1	Resource Description Framework	17
3.1.2	Resource Description Framework Schema	19
3.1.3	Web Ontology Language	21
3.1.4	RIF/SWRL	22
3.1.5	SPARQL	23
3.1.6	Unifying Logic, Proof e Trust	25
4	DESENVOLVIMENTO	26
4.1	Modelação do <i>workflow</i>	26
4.1.1	Perfis de utilizadores do sistema	27
4.1.2	Especificação das etapas do <i>workflow</i>	28
4.1.3	Níveis de acesso às etapas do <i>workflow</i>	32
4.1.4	Histórico de alterações e linguagem cromática	33
4.1.5	Modelo do <i>workflow</i>	34
4.2	Tecnologias Utilizadas	38
4.2.1	GraphDB	39
4.2.2	MongoDB	39

4.2.3	NodeJS + ExpressJS	39
4.2.4	VueJS + Vuetify	40
4.2.5	Preservação dos dados do workflow	40
4.3	Interface desenvolvida	41
4.3.1	Implementação dos Níveis de Acesso	44
4.3.2	Implementação do Histórico de Alterações e da Linguagem Cromática	45
4.3.3	Implementação das etapas do <i>workflow</i>	49
4.3.4	Interface de Consulta de Pedidos do Utilizador e Notificações	61
5	CONCLUSÃO	69
5.1	Conclusões acerca do Projeto	69
5.2	Trabalho Futuro	70
5.3	Reflexão acerca do valor do projeto no desenvolvimento pessoal e profissional	71
	Referências	73
A	IMAGENS COMPLEMENTARES DO MODELO BPMN DO <i>workflow</i>	75
B	INTERFACES DE CONSULTA DE PEDIDOS DO UTILIZADOR E NOTIFICAÇÕES	78
B.1	Interfaces de Consulta dos Dados de Pedidos do Utilizador	78

LISTA DE FIGURAS

Figura 1	Plataforma M51-CLAV-Arquivo digital: Plataforma modular de classificação e avaliação da informação pública como serviço central para a classificação e avaliação.	2
8figure.captio.3		
9figure.captio.4		
9figure.captio.5		
Figura 5	Exemplo da tabela de classificação funcional-orgânica.	11
Figura 6	Exemplo da tabela de classificação funcional.	12
Figura 7	Exemplo da tabela de classificação orgânica.	12
Figura 8	Processo de criação da Lista Consolidada.	13
Figura 9	Exemplo da lista consolidada.	14
17figure.captio.13		
18figure.captio.14		
18figure.captio.16		
19figure.captio.17		
19figure.captio.18		
22figure.captio.21		
Figura 16	Representação dos vários elementos do BPMN.	27
Figura 17	Interface de consulta de um pedido no <i>workflow</i> .	32
Figura 18	Modelo BPMN da etapa de Pedidos Novos .	35
Figura 19	Modelo BPMN da etapa de Pedidos em Apreciação Técnica .	36
Figura 20	Modelo BPMN da etapa de Pedidos em Validação .	37
Figura 21	Modelo BPMN da etapa de Pedidos Devolvidos .	37
Figura 22	Modelo BPMN da etapa de Pedidos Aprovados .	38
Figura 23	Arquitetura da plataforma.	38
Figura 24	Versão inicial da interface com as várias fases do <i>workflow</i> .	42
Figura 25	Versão final da interface com as várias etapas do <i>workflow</i> .	42
Figura 26	Interface de consulta dos pedidos na etapa de Pedidos em Apreciação Técnica .	43
Figura 27	Interface de consulta do Histórico de alterações de um pedido nos vários estados.	47
Figura 28	Ecrã de escolha das duas etapas do pedido a comparar.	48
Figura 29	Janela de consulta da comparação de duas etapas do histórico.	49

Figura 30	Interface de distribuição do pedido.	51
Figura 31	Interface para adicionar um despacho na distribuição de um pedido.	52
Figura 32	Interface de análise de um pedido.	53
Figura 33	Interface de consulta dos despachos atribuídos ao respetivo pedido.	53
Figura 34	Exemplo de uma nota com a indicação dos dados removidos.	55
Figura 35	Interface de validação de um pedido.	55
Figura 36	Janela de confirmação de finalização do pedido.	56
Figura 37	Janela de confirmação de finalização do pedido com campos inválidos.	57
Figura 38	Janela com os erros da validação de um pedido.	58
Figura 39	Interface de pedido finalizado com sucesso.	58
Figura 40	Interface de consulta dos pedidos do utilizador.	61
Figura 41	Interface de consulta dos dados de um pedido do utilizador.	62
Figura 42	Relatório com os detalhes entre os dados do pedido submetido e os dados finais.	63
Figura 43	<i>Dialog</i> com nota relativa a um campo do pedido.	64
Figura 44	Exemplo de notificação em pdf de pedidos finalizados.	65
Figura 45	Modelo BPMN parcial das últimas fases do <i>workflow</i> .	75
Figura 46	Modelo BPMN parcial das primeiras etapas do <i>workflow</i> .	76
Figura 47	Modelo BPMN do <i>workflow</i> completo.	77
Figura 48	Interface de consulta dos dados de um pedido do utilizador no estado Distribuído .	78
Figura 49	Interface de consulta dos dados de um pedido do utilizador no estado Apreciado .	79
Figura 50	Interface de consulta dos dados de um pedido do utilizador no estado Devolvido .	80
Figura 51	Interface de consulta dos dados de um pedido do utilizador no estado Validado .	81

LISTA DE TABELAS

Tabela 1	Descrição das principais funções da classificação funcional.	10
Tabela 2	Descrição dos níveis da lista consolidada.	13
Tabela 3	Exemplo de prefixos que referenciam IRI.	18
20table.caption.19		
20table.caption.20		
Tabela 6	Lista descritiva dos vários níveis de acessos e a respetiva descrição.	28
Tabela 7	Modelo de dados dos pedidos.	29
Tabela 8	Descrição das operações presentes nas etapas do <i>workflow</i> .	29
Tabela 9	Relação entre os níveis dos utilizadores e as respetivas etapas do <i>workflow</i> .	33
Tabela 10	Operações restritas e os respetivos níveis permitidos.	33
Tabela 11	Descrição das cores no <i>workflow</i> .	34
Tabela 12	Calendarização prevista.	70
Tabela 13	Sugestão de reformulação da Linguagem cromática .	71

SIGLAS

AP	Administração Pública
API	Application Programming Interface
ASIA	Avaliação Supra-Institucional na Administração
BPMN	Business Process Model and Notation
BSON	Binary JSON
CLAV	Plataforma M51-CLAV-Arquivo digital: Plataforma modular de classificação e avaliação da informação pública
CSV	Comma-separated values
DGLAB	Direção-Geral do Livro, dos Arquivos e das Bibliotecas
IRI	Internationalized Resource Identifier
J2EE	Java2 Enterprise Edition
JSON	JavaScript Object Notation
LC	Lista Consolidada
MEF	Macroestrutura Funcional
MEI	Mestrado em Engenharia Informática
MVC	Model-View-Controller
ORM	Object-Relational Mapping
OWL	Web Ontology Language
PN	Plano de Negócio
RCM	Resolução do Conselho de Ministros

RDF	Resource Description Framework
RDFS	Resource Description Framework Schema
REST	Representational state transfer
RIF	Rule Interchange Format
RuleML	Rule Markup Language
SKOS	Simple Knowledge Organization System
SWRL	Semantic Web Rule Language
TS	Tabelas de Seleção
TSV	Tab-separated values
UM	Universidade do Minho
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
W ₃ C	World Wide Web Consortium
Web	World Wide Web
XML	Extensible Markup Language

INTRODUÇÃO

No presente capítulo será apresentado o contexto teórico relativo à plataforma CLAV, responsável por auxiliar na classificação e avaliação da informação pública. Segue-se a motivação para o mesmo. Numa fase final serão apresentados os objetivos da presente dissertação, a metodologia utilizada e a estrutura do presente documento.

1.1 CONTEXTO

Ultimamente tem-se observado a definição de políticas e estratégias governamentais para a metamorfose digital na Administração Pública (AP) até 2020. No âmbito destas está a Resolução do Conselho de Ministros (RCM) n.º 108/2017, de 26 de julho, a qual aprovou a Estratégia TIC 2020 e os respetivos Planos de Setoriais TIC de cada uma das 17 áreas governamentais. Esta estratégia detém 3 núcleos: 1) integração e interoperabilidade; 2) inovação e competitividade; 3) partilha de recursos.

Destacando também a RCM n.º 51/2017, de 19 de abril, a qual tem como objetivo reduzir o consumo de papel na Administração Pública, «promovendo a otimização de processos e a modernização de procedimentos administrativos» (do Conselho de Ministros, 2017), especificamente através da desmaterialização de processos, da promoção da adoção de sistemas de gestão documental eletrónica e da digitalização de documentos destinados a ser arquivados. Junto das medidas previstas estão a adoção de processos de classificação, avaliação e seleção de informação, tendo em consideração, sempre que possível, os princípios de uma Macroestrutura Funcional (MEF) e a Avaliação Supra-Institucional na Administração (ASIA).

Esta transformação implica novas práticas de promoção de interoperabilidade semântica e de gestão da informação arquivística, nomeadamente ao nível da sua classificação e da avaliação, bem como novas formas de atuação do organismo de coordenação nesta área – a Direção-Geral do Livro, dos Arquivos e das Bibliotecas (DGLAB).

A DGLAB já há vários anos que delineava algumas destas novas formas, sendo exemplo disso os projetos MEF e ASIA, os quais deram início à Lista Consolidada (LC) para a

classificação e avaliação da informação pública. Na [secção 2.2](#) será abordada esta temática com mais detalhe.

A plataforma apresenta um sistema tecnológico que permite administrar e providenciar a estrutura concetual, de acordo com o MoReq2010, agregando logicamente a informação produzida pelas diversas entidades públicas, bem como desmaterializar o serviço de avaliação da informação pública e o serviço de controlo de eliminação de informação arquivística. Um exemplo é a submissão de propostas de tabela de seleção ou o envio de autos de eliminação, para apreciação da DGLAB.

Esta plataforma terá também a capacidade de agregação com sistemas de gestão documental (*record systems*) ou com sistemas dirigidos ao negócio (*business systems*), fornecendo e recebendo componentes relativos à classificação e à avaliação, como é demonstrado na [Figura 1](#). Proporciona ainda que as organizações apliquem esquemas hierárquicos normalizados para a organização da informação, garantindo a interoperabilidade semântica.

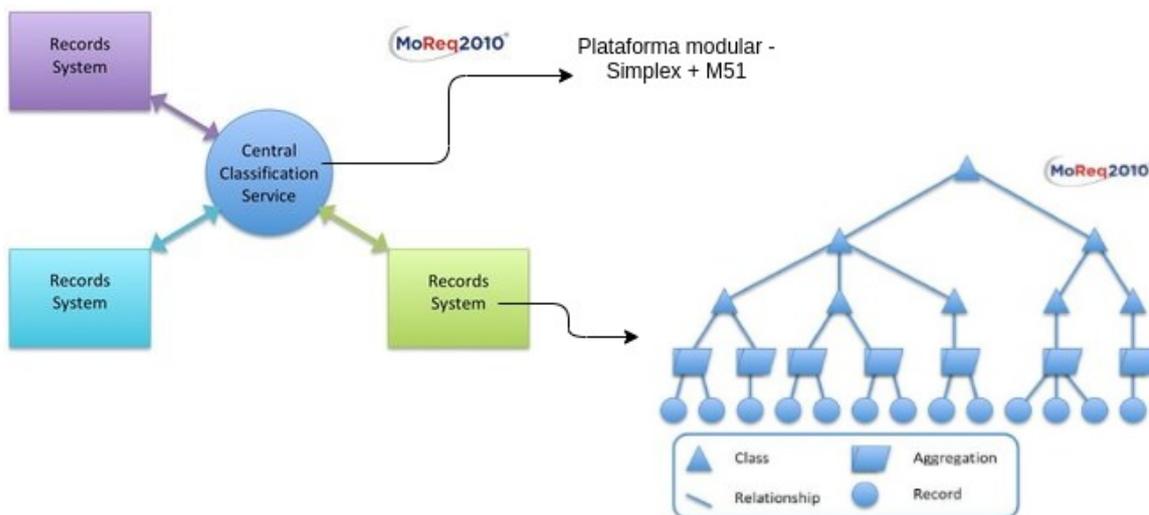


Figura 1: Plataforma M51-CLAV-Arquivo digital: Plataforma modular de classificação e avaliação da informação pública como serviço central para a classificação e avaliação.¹

¹ Imagem adaptada de (MoReq2010, s.d.),(Vieira & Borbinha, 2011).

1.2 MOTIVAÇÃO

A Plataforma M51-CLAV-Arquivo digital, plataforma modular de classificação e avaliação da informação pública (CLAV), simplifica e desmaterializa procedimentos na Administração Pública, fornecendo, ainda, uma agregação de produtos e serviços que, por consequência, alteram consideravelmente o cenário atual. Estes serviços são orientados tanto para a Administração Pública, como também para as empresas e o cidadão. De seguida é apresentada a lista destes produtos e serviços (Lourenço, Ramalho, Gago, & Penteado, 2018):

- Referencial Lista Consolidada para a organização e avaliação da informação pública (ontologia), contemplando a representação das funções do Estado e dos processos de negócio da Administração Pública, das relações entre estes, com identificação das entidades responsáveis pela preservação da informação e do ciclo de vida desses processos;
- Tabelas de seleção (TS), derivadas da Lista Consolidada, com o ciclo de vida da informação associada aos processos de negócios (prazos de conservação/ retenção administrativa e destino final da informação pública) para implementação em contexto organizacional ou pluriorganizacional.

Simplifica também os serviços de (Lourenço et al., 2018):

- Avaliação da informação produzida pela Administração Pública: atividades de recolha de dados, submissão, análise e decisão sobre as propostas, procedimento atualmente sujeito a uma complexa tramitação;
- Controlo de eliminação de documentos, através de submissão de autos diretamente na plataforma.

No entanto, sem um controlo pormenorizado sobre a manipulação da informação, surgem dois problemas. Um deles prende-se com o facto de que toda a manipulação terá que ser realizada pela DGLAB, de forma a minimizar os erros cometidos na introdução ou edição da informações. No entanto, esta abordagem faz com que a DGLAB fique sobrecarregada com processos. O outro problema surge quando se passa essa responsabilidade para as respetivas entidades, uma vez que podem ser cometidos erros, surgindo irregularidades na informação.

1.3 OBJETIVOS

Dadas as características da plataforma CLAV e tendo em conta a grande importância da informação contida nela, é importante garantir que toda a criação, alteração ou eliminação de dados seja analisada pela DGLAB, para que a sua fidedignidade seja validada.

Deste modo, é necessário especificar e implementar na plataforma existente um *workflow* de controlo da informação inserida, editada ou eliminada. Esta implementação encontra-se dividida em duas fases: a modelação de um *workflow* completo, através das especificações do *Business Process Model and Notation* (BPMN), detalhando todas as etapas e operações; e a implementação deste modelo na plataforma CLAV.

É também necessário implementar uma interface de gestão dos pedidos que o utilizador submete na plataforma, de modo a que este visualize o estado em que estes se encontram, mas também a consulta da notificação relativa a cada um desses pedidos. Esta notificação será um resumo do percurso do pedido ao longo das várias etapas do *workflow*.

1.4 METODOLOGIA

Para a realização desta dissertação, adotou-se a metodologia a seguir descrita:

- Estudo teórico acerca das abordagens semelhantes ao CLAV noutros países, nomeadamente a abordagem de Andaluzia (Espanha), percebendo a forma de avaliação e classificação utilizada, mas também as tecnologias utilizadas para a sua implementação;
- Estudo teórico acerca da modelação de um *workflow*, utilizando as especificações do BPMN;
- Estudo acerca das tecnologias e ferramentas utilizadas atualmente no desenvolvimento da plataforma CLAV;
- Desenvolvimento e implementação do *workflow* na plataforma;

De forma a garantir que o trabalho se encontra conforme o esperado, serão realizadas reuniões semanais com o orientador e com a DGLAB.

1.5 EXTRUTURA DO DOCUMENTO

Este documento encontra-se estruturado em cinco capítulos.

O capítulo 1 apresenta uma introdução acerca do projeto, onde é apresentado um contexto sobre o mesmo e a motivação pela qual este deve ser implementado. São apresentados os objetivos a realizar, bem como a metodologia adotada para os cumprir. Por fim, é apresentada a estrutura de todo o documento.

O capítulo 2 sumariza a abordagem do governo de Andaluzia (Espanha) para a construção de um referencial classificativo da informação, bem como as tecnologias e ferramentas utilizadas para a construção da plataforma. Apresenta também a abordagem utilizada na plataforma CLAV e, por fim, é apresentada uma comparação entre as duas.

O capítulo 3 visa, essencialmente, dar a conhecer os fundamentos teóricos que dão suporte ao CLAV, bem como os fundamentos para a construção da sua ontologia.

O capítulo 4 compreende todo o trabalho desenvolvido. Inicialmente é apresentado um resumo sobre o BPMN, seguindo-se a apresentação de todo o processo de modelação do *workflow*, bem como os seus requisitos. Posteriormente são listadas as várias tecnologias e *frameworks* utilizadas, bem como quais as bases de dados utilizadas. No final são apresentadas as diversas interfaces implementadas, nomeadamente todas as interfaces relativas às várias etapas do *workflow* e dos pedidos.

Por último, o capítulo 5, apresenta as conclusões do projeto, bem como algumas melhorias possíveis em futuras intervenções neste âmbito. É ainda apresentada uma reflexão acerca do valor do projeto no desenvolvimento pessoal e profissional.

ESTADO DA ARTE

No contexto da classificação funcional documental, existem várias iniciativas em curso tanto na Europa como na Austrália. A maior parte delas está ainda na fase de recolha de informação e requisitos.

Neste capítulo será apresentado um resumo sobre as abordagens definidas em Andaluzia (Espanha) e em Portugal. Serão destacados alguns tópicos relevantes face às duas abordagens e, no final, será apresentada uma comparação entre ambas.

2.1 ANDALUZIA (ESPANHA)

Quando se fala na função dos arquivistas, fala-se também em papéis e na sua organização. Estando numa era de desenvolvimento tecnológico e, aliado à falta de espaço para guardar os documentos físicos, surge a necessidade de criar um sistema informático para lidar com esse problema.

Para tal, é necessário criar um sistema capaz de guardar os documentos num formato virtual e que possibilite a sua classificação. Como apenas se podem aplicar critérios de classificação sobre o que se pode descrever, pode afirmar-se que a descrição e a normalização são duas tarefas que se encontram fortemente relacionadas.

Este esquema de classificação poderá ser utilizado em inúmeros contextos, tais como:

- Arquivos Centrais de Ministérios;
- Órgãos Autónomos;
- Empresas Públicas.

2.1.1 *Tabela Classificatória*

A necessidade de uma base de classificação universal, como uma tabela classificatória (Tabela de Seleção), era evidente pelo simples facto de esta ser a base para a ferramenta da gestão arquivística.

Após várias análises, por parte do governo de Andaluzia, foram identificadas duas abordagens: a classificação orgânica e a classificação funcional.

A classificação orgânica é uma abordagem pouco segura, uma vez que a sua constante evolução estrutural nas organizações apresenta um ritmo elevado face às exigências da sociedade.

Por outro lado, a classificação funcional garante uma estratégia mais segura. Embora as funções variem, tal como as organizações, esta alteração acontece a um ritmo muito mais lento, mantendo assim uma classificação mais sólida e segura.

2.1.2 *Classificação Funcional*

Assumindo que as funções são atividades que uma instituição deve exercer, com base no cumprimento do fim para a qual foi criada, pode afirmar-se que as funções estão constituídas com uma base de atividades e/ou serviços para a sua realização.

A elaboração de uma tabela de funções e subfunções é a maior dificuldade para este tipo de abordagens, sendo que esta dificuldade deve-se ao facto de existirem vários termos para classificar a mesma atividade, ou seja, uma atividade pode ser classificada como "Agricultura" ou como "Pragas de Campo".

Uma regra fundamental para identificar as diferentes funções é não confundir os elementos de classificação seguintes:

- Organização (Classificação Orgânica);
- Ações (Classificação Funcional);
- Questões ou Assuntos (Classificação Temática).

Uma sugestão dada por García (García, 2004) para a classificação é, no caso de não ser direta qual a função a aplicar, usar a negação, ou seja, encontrar e excluir as funções que não se enquadram.

Na lista¹ que se segue, enumeram-se algumas atividades que devem ser reorganizadas em funções:

- Assessoria política;
- Assessoria jurídica;
- Coordenação da informação institucional;
- Impulso da atividade económica;
- Programação, acompanhamento, avaliação e coordenação de ações de conteúdo económico;
- Fiscalização financeira e tributária;

¹ Adaptado de (García, 2004)

- Gestão, promoção e desenvolvimento do comércio e do artesanato;
- Promoção de uma cultura empresarial;
- Promoção e coordenação de voluntariado;
- Incentivo, promoção e desenvolvimento do desporto.

Olhando de uma forma geral para a lista, é possível extrair alguns adjetivos, tais como **político, legal, financeiro** e os seus genitivos **da atividade económica, do comércio e artesanato, do desporto**. Para identificar ações de forma abstrata, pode manter-se apenas os substantivos **coordenação, impulso e promoção**.

É válido afirmar que as funções são consideradas categorias intelectuais abstratas, como forma de organizar documentos de uma forma lógica e hierárquica, não devendo ser confundidas com as disciplinas e/ou competências que fazem parte das áreas dessas funções.

Uma vez identificadas todas as ações, estas foram divididas em atividades e funções. As atividades são «o conjunto de ações independentes do domínio de competência»¹ (García, 2004) e as funções são «o conjunto de atividades destinadas a um mesmo fim administrativo»¹ (García, 2004).

Para a classificação funcional definiu-se uma codificação numérica com seis níveis. Os primeiros três, sendo estes os que têm mais importância, representam: Função, como primeiro nível; Atividade, como segundo; e Série como terceiro.

Uma vez que a característica das séries vem muitas vezes de grupos documentais menores, foi necessário estabelecer mais três níveis. Assim sendo, estes últimos são designados para alcançar as subséries e/ou séries subordinadas, níveis 5 e 6 respetivamente. A [Figura 2](#) mostra, de um modo geral, os vários níveis.

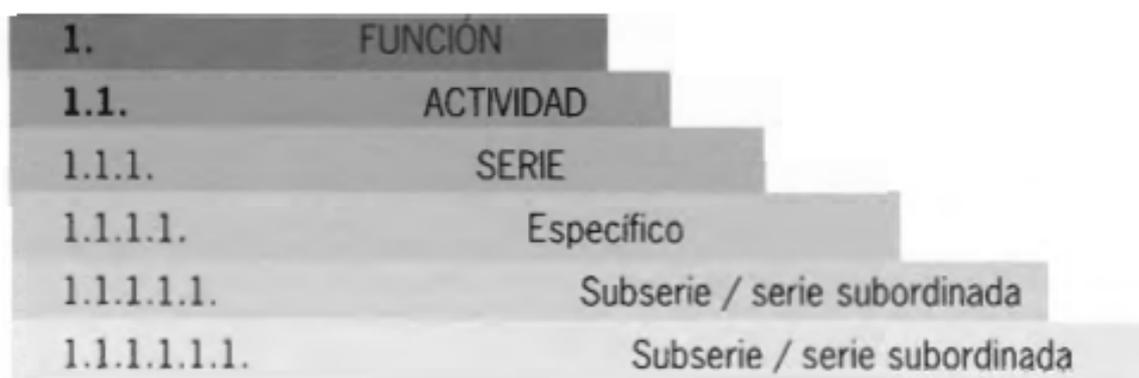


Figura 2: Representação dos vários níveis da classificação funcional.(García, 2002)

¹ Tradução do autor.

O quarto nível é designado de Específico, sendo um recurso auxiliar que permite o crescimento da tabela classificativa (Tabela de Seleção), sem que isso implique a reformulação integral da mesma.

Este dígito é opcional como carácter classificativo. No entanto, não pode ser omitido para manter a estrutura classificativa, ou seja, possibilitar o uso dos níveis seguintes (5 e 6). A solução para este problema passa pela utilização de um 0 (zero) nesta mesma posição. Para exemplificar esta situação, a [Figura 3](#) mostra o uso de uma classificação, no qual se utilizou o quinto nível classificatório, sem que o anterior, nível 4, apresentasse um carácter classificativo.

1.4.	■ PROTOCOLO Y REPRESENTACIÓN
1.4.1.	■ EXPEDIENTES DE CONCESIÓN DE HONORES Y DISTINCIONES
1.4.1.0.1.	■ Premios Andalucía de Agricultura y Pesca
1.4.2.	■ EXPEDIENTES DE CONCURSOS Y CERTÁMENES

Figura 3: Tabela de classificação funcional para o nível 1.4.(García, 2002)

Por outro lado, este dígito pode ser utilizado como identificador característico ou condicional e, nesse caso, aparece entre parêntesis. Observando a [Figura 4](#) verifica-se o uso de parêntesis, uma vez que, algumas séries de processos de contratação administrativa, podem ser geridas como um contrato menor, sendo que pode ser necessário salientar essa situação. Assim sendo, 3.6.1.(1.) representa um contrato menor e, se houver a necessidade, podem definir-se as subséries seguintes, de modo a que 3.6.1.0.1 passa a ser 3.6.1.1.1, o que indica que está a ser processada como um contrato menor.

3.6.	■ GESTIÓN DE GASTOS CORRIENTES, INVERSIONES Y ACTIVOS FINANCIEROS (CAPS. II, VI Y VIII)
3.6.1.	■ EXPEDIENTES DE CONTRATACIÓN DE OBRAS
3.6.1.(1).	■ Contrato menor
3.6.1.(0).1.	■ De primer establecimiento, reforma o gran reparación

Figura 4: Tabela de classificação funcional para o nível quatro "Especificação" como identificador característico.(García, 2002)

Para a criação das principais funções foi realizada uma filtragem das atividades administrativas, o que resultou num total de sete funções descritas na [Tabela 1](#).

Código	Designação	Descrição
1.	Governo	Função que agrega todos os serviços e negócios que dependem de decisões políticas e executivas.
2.	Administração	Função que engloba as atividades que constituem a gestão administrativa da própria instituição.
3.	Economia / Finanças	Função responsável pela gestão económica da instituição como a contratação administrativa, contabilidade e fiscalização.
4.	Gestão / Regulamentação	Função responsável pela organização e regulação do organismo como garantia de que este exerce as funções para as quais foi confiado de forma coordenada e organizada.
5.	Incentivo / Promoção / Desenvolvimento	Função responsável pela promoção, desenvolvimento e incentivo a todas as áreas que se inserem e na sua área de atuação.
6.	Investigação e Formação	Função que inclui a realização de atividades de investigação e formação para as atividades em que a entidade se responsabiliza.
7.	Avaliação / Controlo / Monitorização / Inspeção	Função responsável por acompanhar e avaliar todas as anteriores, como forma de verificar o grau de desempenho e êxito e para assegurar o suporte ao planeamento posterior.

Tabela 1: Descrição das principais funções da classificação funcional.

2.1.3 Classificação Orgânica

Face às circunstâncias de que o arquivo central trabalha melhor com uma tabela funcional, mas as unidades administrativas, por outro lado, fazem melhor uso das tabelas orgânicas, surge a necessidade de implementar uma estrutura que dê conta das necessidades. Para resolver esse problema surgiu a necessidade de trabalhar com uma estrutura híbrida, ou seja, usar uma classificação baseada tanto na orgânica como na funcional.

Segundo Mundet (Mundet, 1994) os requisitos gerais para a construção de uma tabela de classificação integrada são: delimitação; singularidade; estabilidade; simplicidade; e consistência. Além destas, em (García, 2004) são acrescentados os seguintes²:

1. Deve estar organizado numa estrutura hierárquica aplicável a qualquer disciplina, competência ou órgão da Administração Autónoma. Não pode, portanto, ser uma tabela *ad hoc*, mas deve aplicar-se a qualquer fundo, de forma a facilitar a organização de arquivos com diversos fundos.
2. A classificação deve ser baseada na integração e interação dos três elementos básicos da classificação. Deve recolher a ligação entre organismos, funções e assuntos, de forma flexível para poder passar facilmente e de acordo com os interesses, de um modo de classificação para outro, ou combiná-los.
3. Deve permitir o crescimento e evolução da tabela sem traumas, ou seja, deve tolerar o acréscimo de novas séries sem alterar a hierarquia ou a codificação que foi atribuída a uma série, a menos que seja estritamente necessário.

² Traduzido do autor.

4. A codificação e sistematização devem ser realizadas considerando as possibilidades oferecidas pelas ferramentas computacionais, ou seja, devem ser concebidas como requisitos de um sistema integrado de gestão de arquivos.

Dada a necessidade de manter uma correlação entre a posição a que uma entidade pertence na tabela e a sua posição hierárquica na organização, foi definida uma codificação numérica com três níveis de profundidade separados por ponto(.).

Não havendo necessidade de descer para além do terceiro nível, uma vez que esta descida seria um excesso de complexidade para a classificação, descreveram-se os níveis da seguinte forma:

- O primeiro dígito diz respeito ao organismo;
- O segundo ao centro diretivo;
- E o terceiro à unidade administrativa.

Para além da codificação, foi desenvolvido um padrão de códigos, de forma a existir coerência, mantendo-se invariável ao longo do tempo. Alguns destes códigos são:

- Independentemente da direção das entidades, o código **1** diz respeito ao Diretor, **2** ao Vice-Ministro e **3** ao Secretariado Técnico Geral.
- No interior de cada Centro Diretivo, mais propriamente nas Unidades Administrativas, o número **1** deverá sempre representar o Secretariado.

2.1.4 Solução apresentada

Face às abordagens anteriores, nomeadamente a classificação orgânica e classificação funcional, e com o objetivo de desenvolver um sistema de classificação documental, surgiu uma tabela de classificação composta por nove dígitos, como se pode ver na [Figura 5](#), onde seis dizem respeito à classificação funcional e os restantes três refletem a classificação orgânica.

Funcional						Orgânica			
F	A	S	E1	Sb	Ss	/	O	C	U
1.	1.	1.	1.	1.	1.	/	1.	1.	1.

Figura 5: Exemplo da tabela de classificação funcional-orgânica.

Além da classificação funcional-orgânica, pode em alternativa ser usada a classificação funcional, como mostra a [Figura 6](#), ou a classificação estritamente orgânica, representado pela [Figura 7](#).

Funcional					
F	A	S	E	Sb	Ss
1.	1.	1.	1.	1.	1.

Figura 6: Exemplo da tabela de classificação funcional.

Orgânica		
O	C	U
1.	1.	1.

Figura 7: Exemplo da tabela de classificação orgânica.

As abreviaturas **F**, **A**, **S**, **El**, **Sb**, **Ss**, **O**, **C** e **U** representam **Função**, **Atividade**, **Série**, **Específico**, **Subsérie**, **Série subordinada**, **Órgão**, **Centro** e **Unidade**, respetivamente.

Este sistema de classificação proposto baseia-se na classificação de cada série, de acordo com os elementos funcional e orgânico, mantendo sempre a independência entre uma estruturação adequada às necessidades, garantindo a relação entre si. A abordagem utilizada permite identificar de uma forma única e inequívoca uma série documental.

2.1.5 Tecnologias e implementação

Para a implementação da tabela classificativa descrita anteriormente num sistema de informação, foi iniciado o desenvolvimento da plataforma *@archivA*³.

Este sistema é responsável por toda a gestão documental de Andaluza. Além disso, este opera, de forma modular, com outros sistemas como o *W@ndA*⁴ e o *Trew@*⁵. O *W@ndA* é responsável pela gestão do *workflow* na administração de Andaluza e faz uso do *Trew@* como motor de processamento.

O desenvolvimento desta plataforma tem uma arquitetura *Model-View-Controller* (MVC) implementada em Java, mais propriamente Java2 *Enterprise Edition* (J2EE). A base de dados consiste num modelo relacional da *Oracle* e é acedida utilizando uma *framework* de *Object-relational mapping* (ORM), denominada *Hibernate*.

³ Pode ser consultado em <https://ws024.juntadeandalucia.es/ae/adminelec/areatecnica/archiva>

⁴ Pode ser consultado em <https://www.juntadeandalucia.es/repositorio/usuario/listado/fichacompleta.jsf?idProyecto=708>

⁵ Pode ser consultado em <https://ws024.juntadeandalucia.es/ae/adminelec/areatecnica/trewa>

É também utilizada uma *framework Apache Struts*, a qual permite gerar páginas *web* a partir de código Java e a *framework Tomcat* «como o container para a aplicação». (Martínez Force, Melero Casado, Molina González, & Valiente Fabero, 2010)⁶

2.2 PORTUGAL

Como se abordou no capítulo anterior, e pela análise da [Figura 8](#), a lista consolidada resulta da elaboração dos projetos MEF e ASIA como referencial para a classificação e avaliação da informação.



Figura 8: Processo de criação da Lista Consolidada.⁷

A Lista Consolidada é uma ferramenta importante que serve de suporte à classificação, composta por uma estrutura hierárquica de classes pré-definidas, as quais representam as funções, sub-funções e processos de negócio executados pela Administração Pública.

Segue então uma classificação funcional hierárquica, com quatro níveis de profundidade. O primeiro e o segundo nível foram importados do projeto MEF; o terceiro surgiu da harmonização das classes do mesmo projeto em conjunto com o projeto ASIA, denominados de Processos de Negócio (PN). O quarto nível é aplicado apenas quando é necessário atribuir diferentes prazos e destinos finais à informação dos processos de negócio.

A descrição de cada nível é apresentado na [Tabela 2](#):

Nível	Descrição
1.	Representação das funções da administração pública.
2.	Representação das subfunções da administração pública.
3.	Representação dos processos de negócio executados pela administração pública.
4.	Representação de subdivisão dos processos de negócio para efeitos de avaliação.

Tabela 2: Descrição dos níveis da lista consolidada.

⁶ Traduzido do autor.

⁷ Fonte (s.a., s.d.).

Esta estrutura garante a identificação unívoca da informação através da associação de um código, representativo de uma classe do esquema de classificação, atribuindo, assim, a este referencial um papel essencial na contínua gestão da informação.

Na [Figura 9](#) encontra-se um exemplo dos vários níveis da Lista Consolidada. Como primeiro nível observa-se a classe 200 - *Execução da Política Externa*, a qual contém várias subclasses que atuam dentro do mesmo tema. Segue-se a subclasse 200.30 - *Projetos de Execução da Política Externa* e, dentro desta, o nível três representativo do processo de negócio 200.30.001 - *Coordenação e Execução de Ações de Ajuda Humanitária e de Emergência*, a qual contém o quarto nível como critério de avaliação deste, nomeadamente à coordenação e execução.

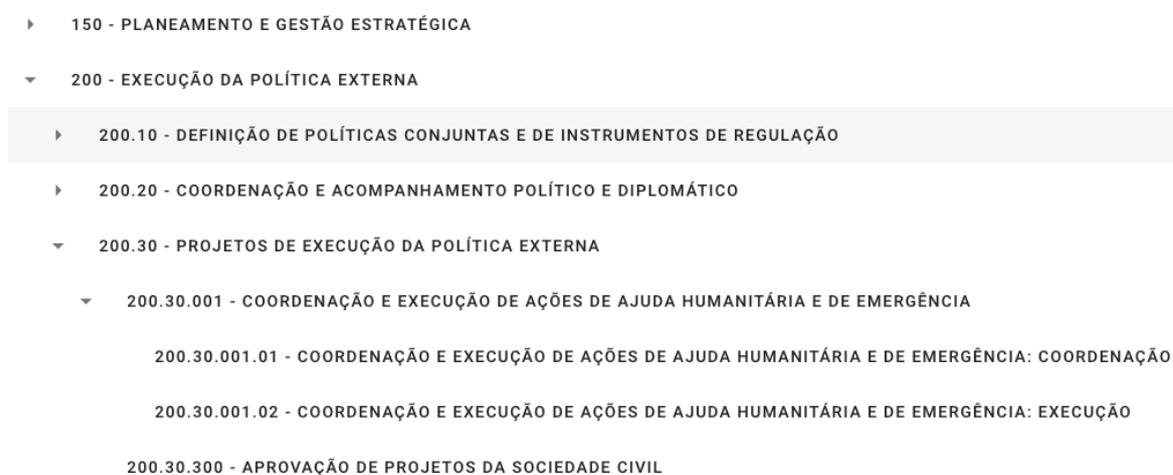


Figura 9: Exemplo da lista consolidada.⁸

Com a disponibilização da lista consolidada, a DGLAB pretende a (s.a., s.d.):

- Integração de sistemas ao nível da interoperabilidade semântica;
- Criação de uma linguagem comum para a Administração Pública;
- Harmonização de perspetivas sobre a natureza das funções e dos processos;
- Avaliação suprainstitucional da informação pública;
- Determinação da entidade responsável pela conservação permanente da informação;
- Controlo mais eficaz nos diferentes ciclos de vida informacional;
- Partilha e rentabilização da informação;
- Racionalização e agilização de processos;

⁸ Exemplo extraído da plataforma <https://clav.dglab.gov.pt/classes>.

- Economia de escala;
- Diminuição de despesas correntes.

2.3 COMPARAÇÃO SÍNTESE

De um modo geral, ambas as abordagens seguem o mesmo princípio, nomeadamente a necessidade de modernizar os processos da administração pública e criar um sistema de classificação documental.

No entanto, a abordagem apresentada por Portugal apresenta um referencial mais sólido devido ao uso de apenas um instrumento de classificação, ao invés da abordagem de Andalusia, a qual apresenta um referencial funcional-orgânico, constituído por dois instrumentos, uma tabela funcional, muito parecida com a abordagem implementada pelo CLAV, e uma tabela orgânica. A escolha da tabela a utilizar é decidida pela entidade, podendo utilizar a funcional-orgânica, composta pela combinação das duas tabelas, o uso da tabela funcional ou da estritamente orgânica.

O uso de apenas quatro níveis hierárquicos também ajuda no momento da classificação da informação, simplificando a pesquisa pela classe pretendida para a classificação.

CLAV

Sendo o CLAV uma plataforma baseada na Web com elevado valor tanto para a Administração Pública como para o Cidadão, é importante seguir boas práticas e padrões para a distribuição dos dados da mesma. Estes dados devem ser distribuídos na Web, preservando o significado dessa informação, de forma a que sejam consumidos por outros sistemas.

Deste modo, o presente capítulo compreende o fundamento teórico relevante para o contexto da plataforma CLAV.

3.1 WEB SEMÂNTICA

Atualmente, e cada vez mais, a *World Wide Web* (Web) contém utilizadores dispersos por todo o globo terrestre, que intervêm com uma enorme fração de informação. À medida que o tempo passa, essa interação ocorre a um ritmo muito superior àquele que era até então concretizável. Um fenómeno que contribui para este acontecimento, é o facto de que cada vez há mais sistemas que se baseiam no acesso a esta informação.

Este fenómeno pode refletir-se como problemático, uma vez que a diversidade de informação presente em diversos formatos dificulta a descoberta de relações entre os dados. Desta forma, e como o exemplo dado por (Librelotto, Ramalho, & Henriques, 2005), «os motores de busca actuais se preocupam apenas com a existência das referidas palavras-chave na página pesquisada e não com a semântica de tal busca».

De forma a garantir a evolução a longo prazo da Web, foi desenvolvida a *World Wide Web Consortium* (W3C) e foi daí que emergiu a Web Semântica. Esta, tem como objetivo a criação de um conjunto de práticas e padrões para a distribuição de dados e respetivos significados na Web, de modo a que estes possam ser manipulados por outros sistemas. (W3C, s.d.)

Algumas destas práticas e padrões propostos pela W3C encontram-se enumerados de seguida:

- RDF;
- RDF Schemas;
- Web Ontology Language (OWL);

- SPARQL.

No seguimento da Web Semântica surgiu a *stack* tecnológica que conjuga estas práticas e padrões, a [Figura 10](#) ilustra esta *stack*.

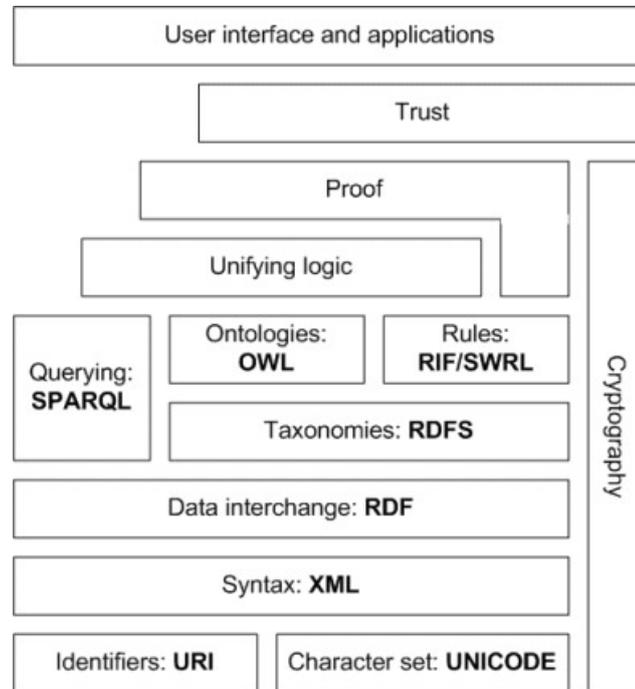


Figura 10: *Stack* tecnológica da Web Semântica. (Obitko, 2007)

Esta *stack* apresenta-se dividida em oito camadas. Começando pela base, tem-se o *Uniform Resource Identifier* (URI) e o *Unicode*, os quais oferecem funções importantes para a Web. O URI é uma sequência de caracteres única que permite identificar recursos, como por exemplo documentos. Já o Unicode é um padrão de codificação de caracteres, permitindo que todos os alfabetos possam ser utilizados na Web.

De seguida, está representada a camada de sintaxe, definida pelo uso de *Extensible Markup Language* (XML), com *namespaces* e *XMLSchemas*, a qual garante o uso de uma sintaxe comum na Web Semântica. XML é uma linguagem de marcação, ou seja, um documento XML apresenta elementos que podem conter atributos e conteúdo, permitindo, assim especificar diferentes vocabulários.

Nos subcapítulos seguintes serão descritas as restantes camadas com maior detalhe.

3.1.1 *Resource Description Framework*

A terceira camada é o *Resource Description Framework* (RDF), sendo este o principal formato para a representação de dados na Web Semântica. O RDF é uma estrutura para representar

dados sob a forma de grafos e é baseada em triplos, nomeadamente Sujeito-Predicado-Objeto, como se exemplifica na [Figura 11](#).

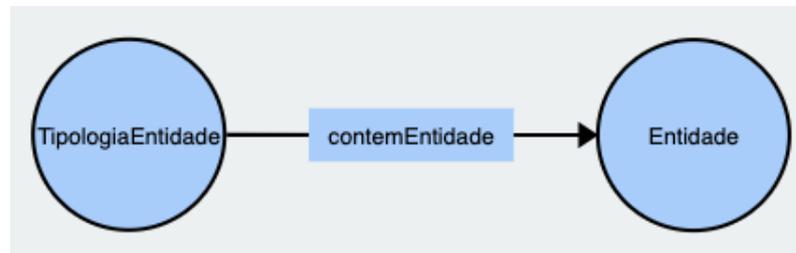


Figura 11: Triplos em RDF. (Wooldridge, s.d.)

Estes triplos podem ser identificados por *Internationalized Resource Identifier* (IRI). Um IRI é uma generalização de URI's, isto é, cada URI ou *Uniform Resource Locator* (URL) é um IRI, no entanto, nem todos os IRI são URI. A [Tabela 3](#) mostra alguns exemplos de prefixos que referenciam IRI's.

Prefixo	IRI
rdf	< http://www.w3.org/1999/02/22-rdf-syntax-ns# >
rdfs	< http://www.w3.org/2000/01/rdf-schema# >
clav	< http://jcr.di.uminho.pt/m51-clav# >

Tabela 3: Exemplo de prefixos que referenciam IRI.

Aplicando o conceito de prefixos que referenciam IRI, podemos obter uma estrutura como a ilustrada na [Figura 12](#) onde se pode observar que o sujeito representado pelo IRI (<http://www.example.org/~joe/contact.rdf#joesmith>), pretende identificar o Joe Smith, que dispõe de uma *homepage* (<http://xmlns.com/foaf/0.1/homepage>) sendo este IRI o predicado, onde o *Friend-of-a-friend* (FOAF) é um vocabulário. Por fim, o objeto é identificado pelo IRI (<http://www.example.org/~joe/>), em que este é também um URI que indica onde se encontra *homepage* do Joe Smith.

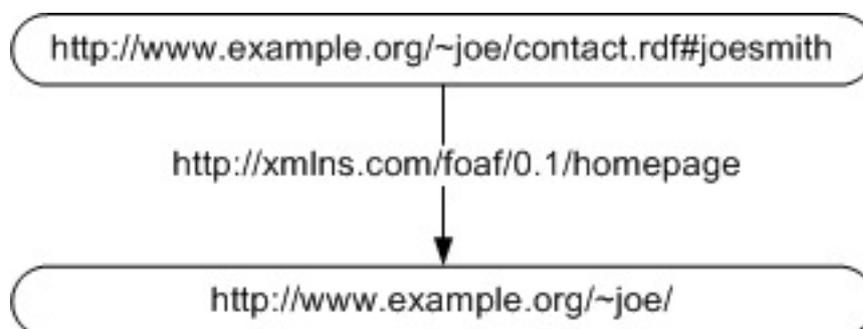


Figura 12: Triplos em RDF com URI. (Obitko, 2007)

O elemento objeto dos triplos, além de poder ser um IRI, pode também ser um elemento literal. Neste contexto, um elemento literal designa-se por uma *string* ou número. Na Figura 13 é representado um objeto (Smith) como um elemento literal.

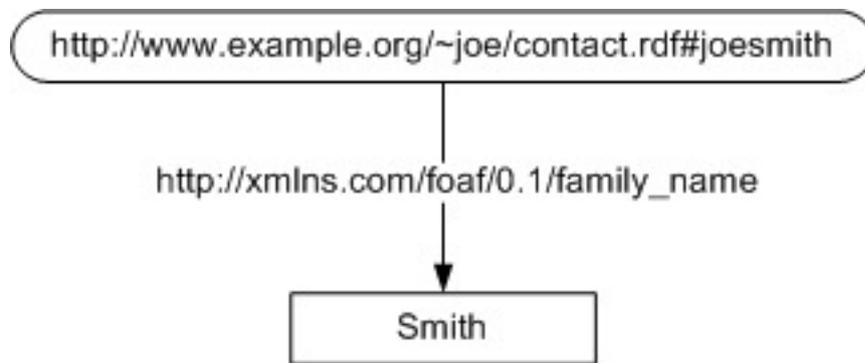


Figura 13: Triplos em RDF com URI e objeto literal. (Obitko, 2007)

Uma vez que o RDF é baseado em grafos, podem-se relacionar todos conceitos abordados anteriormente usando triplos, obtendo assim um grafo. A Figura 14 representa um excerto do grafo da plataforma CLAV.

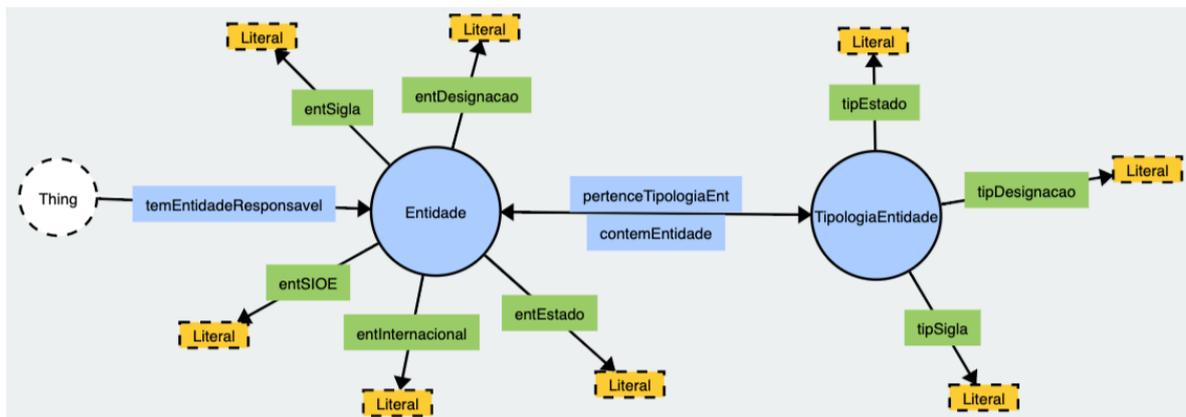


Figura 14: Representação de um grafo em RDF. (Obitko, 2007)

3.1.2 Resource Description Framework Schema

O *Resource Description Framework Schema* (RDFS), sendo esta a quarta camada, é uma extensão de vocabulário do RDF, a qual permite descrever taxonomias de classes e novas propriedades.

As classes que o RDFS acrescenta encontram-se identificadas na Tabela 4.

Class name	comment
<code>rdfs:Resource</code>	The class resource, everything.
<code>rdfs:Literal</code>	The class of literal values, e.g. textual strings and integers.
<code>rdf:langString</code>	The class of language-tagged string literal values.
<code>rdf:HTML</code>	The class of HTML literal values.
<code>rdf:XMLLiteral</code>	The class of XML literal values.
<code>rdfs:Class</code>	The class of classes.
<code>rdf:Property</code>	The class of RDF properties.
<code>rdfs:Datatype</code>	The class of RDF datatypes.
<code>rdf:Statement</code>	The class of RDF statements.
<code>rdf:Bag</code>	The class of unordered containers.
<code>rdf:Seq</code>	The class of ordered containers.
<code>rdf:Alt</code>	The class of containers of alternatives.
<code>rdfs:Container</code>	The class of RDF containers.
<code>rdfs:ContainerMembershipProperty</code>	The class of container membership properties, <code>rdf:_1</code> , <code>rdf:_2</code> , ..., all of which are sub-properties of 'member'.
<code>rdf:List</code>	The class of RDF Lists.

Tabela 4: Lista de classes do RDFS. (Brickley & Guha, 2014)

Na Tabela 5 então listadas as propriedades do RDFS.

Property name	comment	domain	range
<code>rdf:type</code>	The subject is an instance of a class.	<code>rdfs:Resource</code>	<code>rdfs:Class</code>
<code>rdfs:subClassOf</code>	The subject is a subclass of a class.	<code>rdfs:Class</code>	<code>rdfs:Class</code>
<code>rdfs:subPropertyOf</code>	The subject is a subproperty of a property.	<code>rdf:Property</code>	<code>rdf:Property</code>
<code>rdfs:domain</code>	A domain of the subject property.	<code>rdf:Property</code>	<code>rdfs:Class</code>
<code>rdfs:range</code>	A range of the subject property.	<code>rdf:Property</code>	<code>rdfs:Class</code>
<code>rdfs:label</code>	A human-readable name for the subject.	<code>rdfs:Resource</code>	<code>rdfs:Literal</code>
<code>rdfs:comment</code>	A description of the subject resource.	<code>rdfs:Resource</code>	<code>rdfs:Literal</code>
<code>rdfs:member</code>	A member of the subject resource.	<code>rdfs:Resource</code>	<code>rdfs:Resource</code>
<code>rdf:first</code>	The first item in the subject RDF list.	<code>rdf:List</code>	<code>rdfs:Resource</code>
<code>rdf:rest</code>	The rest of the subject RDF list after the first item.	<code>rdf:List</code>	<code>rdf:List</code>
<code>rdfs:seeAlso</code>	Further information about the subject resource.	<code>rdfs:Resource</code>	<code>rdfs:Resource</code>
<code>rdfs:isDefinedBy</code>	The definition of the subject resource.	<code>rdfs:Resource</code>	<code>rdfs:Resource</code>
<code>rdf:value</code>	Idiomatic property used for structured values.	<code>rdfs:Resource</code>	<code>rdfs:Resource</code>
<code>rdf:subject</code>	The subject of the subject RDF statement.	<code>rdf:Statement</code>	<code>rdfs:Resource</code>
<code>rdf:predicate</code>	The predicate of the subject RDF statement.	<code>rdf:Statement</code>	<code>rdfs:Resource</code>
<code>rdf:object</code>	The object of the subject RDF statement.	<code>rdf:Statement</code>	<code>rdfs:Resource</code>

Tabela 5: Lista de propriedades do RDFS. (Brickley & Guha, 2014)

Embora a sintax do RDF padrão seja XML, podem ser utilizadas sintaxes noutros formatos, como por exemplo Turtle¹, TriG¹ ou JSON-LD¹.

¹o Mais informações sobre estas sintaxes podem ser consultadas em (Beckett, Berners-Lee, Prud'hommeaux, & Carothers, 2014), (Bizer & Cyganiak, 2014), (Sporny et al., 2019) respetivamente.

Para ilustrar o uso do vocabulário do RDFS, encontra-se de seguida um pequeno exemplo onde se pode observar o uso das classes e as propriedades deste.

Exemplo da definição de RDFS.

```
@prefix :      <http://www.example.org/sample.rdfs#> .
@prefix rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs:  <http://www.w3.org/2000/01/rdf-schema#> .

:Dog          rdfs:subClassOf :Animal.
:Person       rdfs:subClassOf :Animal.

:hasChild     rdfs:range :Animal;
              rdfs:domain :Animal.
:hasSon       rdfs:subPropertyOf :hasChild.

:Max          a :Dog.
:Abel         a :Person.
:Adam         a :Person;
              :hasSon :Abel.
```

3.1.3 Web Ontology Language

A *Web Ontology Language* (OWL), sendo esta a camada seguinte, é mais expressiva que o XML, o RDF e até mesmo o RDFS, uma vez que oferece um extenso vocabulário. Desta forma, permite criar ontologias com maior detalhe.

Desde 2009 que têm vindo a ser desenvolvidas alterações ao modelo OWL₁, surgindo, assim, a OWL₂, a qual adiciona várias funcionalidades à OWL₁. (Golbreich & Wallace, 2012)

A OWL₁ é dividida em 3 camadas, OWL Lite, OWL DL e OWL Full. Estas 3 sublinguagens são caracterizadas pela sua expressividade, isto é, a OWL Lite é menos expressiva do que a OWL LD e esta, consequentemente, é menos expressiva que a OWL Full. Esta expressividade garante que as ontologias OWL Lite válidas, são também ontologias OWL LD válidas e, à semelhança da sua expressividade, todas as ontologias OWL LD válidas são também ontologias OWL Full válidas.

Dadas as características da OWL₁, a OWL₂ seguiu o mesmo conceito, embora a sublinguagem OWL₂ DL é mais expressiva que a OWL₁ DL, sendo que substitui a OWL₁ Lite por três novos perfis, sendo eles OWL₂ RL, OWL₂ QL e OWL₂ EL, tal como se pode observar na [Figura 15](#).

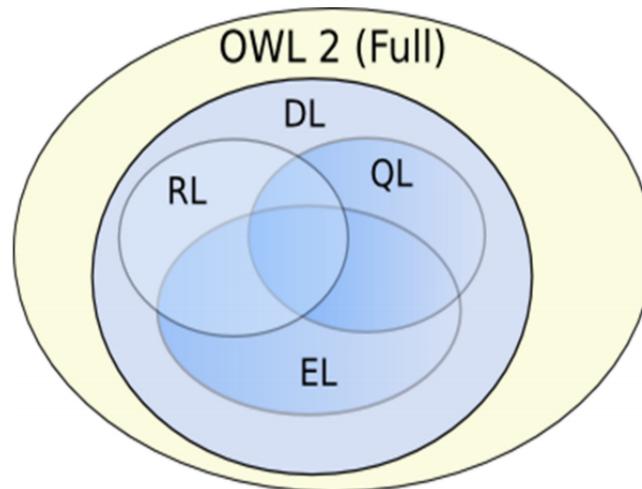


Figura 15: Estrutura da OWL2. (Cardoso & Pinto, s.d.)

A sublinguagem OWL2 Full continua a ser a mais expressiva de todas e herda as características de todas as outras sublinguagens subsequentes. Esta destina-se a ser utilizada quando há a necessidade de grande expressividade, mesmo que isso comprometa a integridade computacional.

OWL2 DL é uma sublinguagem da OWL2 Full, a qual permite uma alta expressividade, embora menor do que a anterior. No entanto, garante a integridade computacional da linguagem.

Os três perfis da OWL2 DL são diferenciados pelos seguintes casos: a OWL2 RL é utilizada para aplicações que exigem um raciocínio escalável, não comprometendo muito o poder da expressividade; a OWL2 EL é usada quando aplicações integram ontologias que definem um número elevado de classes e/ou propriedades; por fim, a OWL2 QL foi planeada para aplicações que integram sistemas de base de dados relacionais, de forma a que estes possam aceder aos dados através de uma ontologia sem perda ou alteração dos mesmos. (Cardoso & Pinto, s.d.)

3.1.4 RIF/SWRL

A camada das regras RIF/SWRL surge uma vez que há a necessidade de definir regras além da semântica existente no RDFS e na OWL.

O *Rule Interchange Format* (RIF) é uma estrutura extensível para linguagens baseadas em regras, incluindo a especificação formal e precisa da sintaxe, da semântica e da serialização do XML. (Boley & Kifer, 2013), (Kifer, 2008)

Por outro lado, a *Semantic Web Rule Language* (SWRL) resulta da combinação entre as sublinguagens OWL DL e OWL Lite da OWL com a sublinguagem *Rule Markup Language*

(RuleML), o que fornece uma sintaxe abstrata de alto nível que estende a sintaxe abstrata do OWL. É também fornecida uma extensão da semântica teórica do modelo OWL que proporciona um significado formal para ontologias OWL. (Horrocks et al., 2004)

3.1.5 SPARQL

Ao nível das três camadas abordadas anteriormente, encontra-se a camada de *Querying* SPARQL. Esta é uma linguagem idêntica ao SQL, embora faça uso de triplos e recursos a RDF para a consulta e apresentação dos dados consultados. Uma vez que o RDFS e a OWL fazem uso do RDF, também o SPARQL pode ser utilizado para consultar bases de conhecimento e ontologias.

O excerto de código seguinte exemplifica o uso da linguagem de *querying* SPARQL:

Query em SPARQL para a consulta de uma Entidade.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX clav: <http://jcr.di.uminho.pt/m51-clav#>
SELECT ?sigla ?designacao {
    ?uri rdf:type clav:Entidade ;
        clav:entDesignacao ?designacao ;
        clav:entSigla ?sigla .
} ORDER BY ?sigla
```

Esta *query* retorna a lista de entidades presentes no sistema, ordenada pela **sigla** e apresentando apenas os dados **designação** e **sigla**.

O SPARQL suporta vários formatos de saída dos dados consultados, nomeadamente XML, JavaScript Object Notation (JSON), Comma-separated values (CSV) e Tab-separated values (TSV), permitindo, assim, uma maior flexibilidade face ao seu uso. (T. W. S. W. Group, 2013)

De forma a expor o resultado da *query* em SPARQL apresentada anteriormente, apresentam-se de seguida três exemplos dos resultados obtidos em três formatos, sendo o primeiro em formato CSV, o segundo em TSV e, o último, em JSON.

Resposta em CSV.

```
sigla,designacao
AC,"Águas de Coimbra, EM"
AISAN,Agência ISAN Portugal
```

Resposta em TSV.

```
?sigla      ?designacao
AC          Águas de Coimbra, EM
AISAN      Agência ISAN Portugal
```

Resposta em JSON.

```
{
  "head" : { "vars" : ["sigla", "designacao"] },
  "results" : {
    "bindings" : [
      {
        "sigla" : {
          "type" : "literal",
          "value" : "AC"
        },
        "designacao" : {
          "type" : "literal",
          "value" : "Águas de Coimbra, EM"
        }
      },
      {
        "sigla" : {
          "type" : "literal",
          "value" : "AISAN"
        },
        "designacao" : {
          "type" : "literal",
          "value" : "Agência ISAN Portugal"
        }
      }
    ]
  }
}
```

3.1.6 *Unifying Logic, Proof e Trust*

Nas últimas camadas é esperado que todas as regras e semânticas tenham sido aplicadas nas camadas inferiores, de forma a que os resultados sejam utilizados para provar deduções. Para os dados confiáveis, devem ser utilizadas criptografias como assinaturas digitais, de forma a verificar tanto a origem como a fonte dos dados.

Dadas as características da plataforma CLAV e analisando os padrões da Web Semântica, pode concluir-se que o seu uso é uma mais valia para o projeto. Desse modo, seguiram-se estes padrões para a construção de toda a base ontológica da plataforma, permitindo assim uma estrutura bastante sólida e completa.

DESENVOLVIMENTO

Neste capítulo será especificado o *workflow* através do uso de *Business Process Model and Notation* (BPMN), nomeadamente quais as etapas e quais as operações em cada etapa do *workflow*, bem como uma explicação do que é e para que serve o BPMN. Segue-se a descrição das tecnologias utilizadas no projeto e as bases de dados utilizadas. É também apresentada uma pequena explicação sobre a preservação dos dados do *workflow*. Será ainda explicado como foi implementado o modelo idealizado e as respetivas interfaces. Por fim, será apresentada a interface para a consulta dos pedidos que o utilizador criou, bem como as notificações de finalização dos pedidos.

4.1 MODELAÇÃO DO *workflow*

Para a modelação do *workflow* utilizou-se uma representação gráfica seguindo os padrões e recomendações do BPMN.

O BPMN é uma especificação que fornece uma notação gráfica para especificar processos de negócio num diagrama de processos de negócio (O. M. Group, 2013). O principal objetivo é dar suporte à modelação de processos de negócio, disponibilizando uma linguagem comum e de simples compreensão para os vários utilizadores dentro de uma instituição, ainda que a sua utilização represente processos complexos.

Uma vantagem do uso do BPMN é o facto de este ter uma notação idêntica a um fluxograma, o que facilita o seu uso, e é independente de qualquer ambiente de implementação.

Desta forma, este diagrama transcreve, no contexto do *workflow* da plataforma CLAV, os processos e as ações que o sistema de informação oferece ao utilizador, sem a necessidade deste entender como os mesmos são executados pelo sistema.

A [Figura 16](#) apresenta os principais elementos da notação do BPMN. Este apresenta quatro elementos base, sendo eles: os eventos; as atividades; os *gateway*; e os conectores ou fluxos.

- Os eventos são representados por círculos e representam ocorrências, isto é, o acontecimento de algo, como o início ou o fim de um processo.
- As atividades são representadas por retângulos e traduzem as tarefas a realizar.

- Os gateway são representados por losangos e indicam condições que, com base na resposta a essa pergunta, determinam o fluxo que o processo deve seguir.
- Os conectores ou fluxos indicam os caminhos que o processo deve seguir durante a sua execução.

Existe ainda outro componente, denominado de *pool*, o qual pode ser dividido em várias *swimlanes*. A *pool* representa processos e os seus participantes, bem como as *swimlanes* que designam os papéis, áreas e responsabilidades no processo.

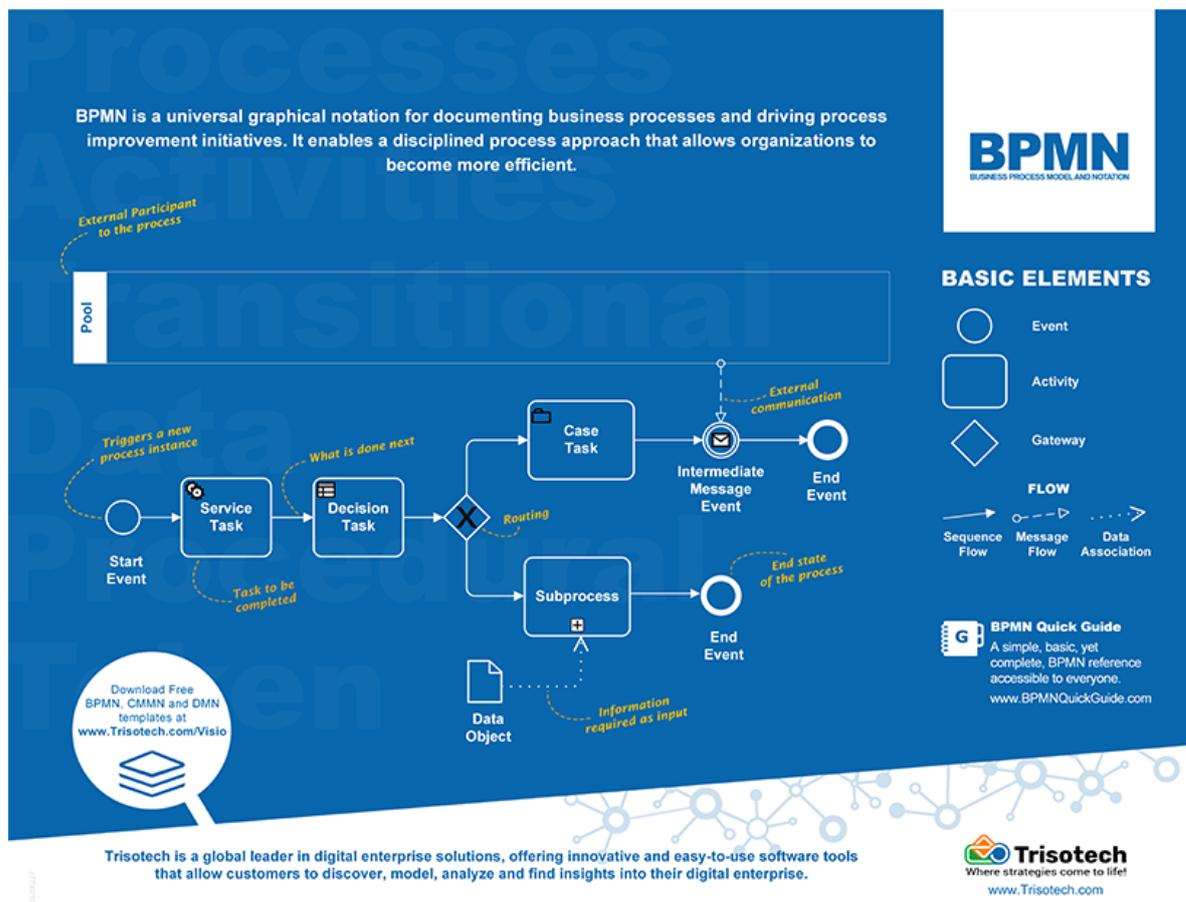


Figura 16: Representação dos vários elementos do BPMN.¹

4.1.1 Perfis de utilizadores do sistema

De forma a garantir a devida utilização da plataforma CLAV, foram criados vários perfis de acesso à mesma. Desse modo, esta consta com um total de dez níveis, pelo que cada um terá a sua limitação. A Tabela 6 mostra esses níveis e a devida descrição.

¹ Fonte: <https://www.trisotech.com/infographics/bpmn-poster>

Nível de Acesso	Descrição
-1	Qualquer utilizador não autenticado.
0	Utilizador com uma chave API.
1	Representante de entidade.
2	Utilizador simples.
3	Arquivo distrital.
3.5	Utilizador avançado.
4	Utilizador validador.
5	Utilizador decisor.
6	Utilizador administrador de perfil funcional.
7	Utilizador administrador de perfil tecnológico.

Tabela 6: Lista descritiva dos vários níveis de acessos e a respetiva descrição.

Esta lista de perfis de utilizadores é importante, uma vez que nem todos os utilizadores podem realizar operações e/ou consultas no *workflow*. Desse modo, apenas uma parte destes utilizadores, mais propriamente os utilizadores com nível superior ou igual a 3.5, pode ter acesso à consulta dos processos que se encontram em processamento no *workflow*.

4.1.2 Especificação das etapas do *workflow*

Após uma análise das características da plataforma, foi evidente a necessidade de controlar a manipulação de dados nesta. Dados estes que apresentam elevado valor para a administração pública, uma vez tratar-se de Legislações, como Portarias e RCM, ou dados de relativos a Entidades. Dessa forma, era necessário prototipar e desenvolver um *workflow* com várias etapas.

Surgiu então a primeira versão deste *workflow*, onde constavam quatro etapas, sendo elas: **Pedidos Novos**; **Pedidos em Apreciação Técnica**; **Pedidos em Validação**; e **Pedidos Devolvidos**.

No entanto, e como forma a manter o registo dos pedidos que eram aprovados, houve a necessidade de adicionar uma nova etapa, sendo ela **Pedidos Aprovados**.

Todas estas etapas e as suas respetivas designações, bem como as operações presentes em cada uma delas, foram definidas em reuniões com elementos da DGLAB e serão descritas de seguida.

Informações e operações nos pedidos

Uma vez que existem informações e operações sobre os pedidos que são comuns às várias etapas, criou-se um modelo de dados que representam os dados relativos aos pedidos e criaram-se algumas operações para os mesmos que serão descritas através da [Tabela 7](#) e da [Tabela 8](#):

Informação	Descrição
Código	Identificador do pedido, composto por dois componentes, o ano em que o pedido foi criado e uma sequência incremental de pedidos nesse ano, composta por sete dígitos. Exemplo: <u>2020-0000001</u>
Tipo	Indicador do tipo do pedido, composto pela descrição do tipo de dado e pelo tipo de operação realizada. Exemplo: <u>Entidade - Criação</u>
Entidade	Sigla da entidade à qual o criador do pedido pertence.
Criador	Utilizador responsável pela criação do pedido.
Data	Data de criação do pedido.
Responsável	Utilizador responsável pelo processo do pedido na etapa à qual este se encontra.
Tarefa	Operações que o utilizador tem acesso para o pedido em questão.

Tabela 7: Modelo de dados dos pedidos.

Operação	Descrição
Ver pedido	Operação que permite consultar os dados detalhados sobre o pedido em questão.
Distribuir pedido	Operação que permite atribuir um responsável ao pedido selecionado e transferir o mesmo para a etapa seguinte.
Devolver pedido	Operação que devolve um pedido, por inconformidade nos dados do mesmo e move-o para a etapa de pedidos devolvidos.
Analisar pedido	Operação que permite um utilizador analisar os dados presentes no pedido selecionado.
Validar pedido	Operação que permite um utilizador validar um pedido que tenha sido analisado.
Substituir responsável	Operação que permite substituir o responsável atual por processar o pedido no estado em que este se encontra.
Reapreciar pedido	Operação que permite que um pedido que esteja na etapa de pedidos em validação, volte à etapa pedidos em apreciação técnica, para ser apreciado novamente.
Ver despachos	Operação que permite consultar as notas que o utilizador escreve sobre o pedido no momento em que o atribuí a outro utilizador, quando este é devolvido ou aprovado.

Tabela 8: Descrição das operações presentes nas etapas do *workflow*.

As operações **Substituir responsável** e **Reapreciar pedido** serão apenas acedidas através da interface de consulta do pedido; e a operação **Ver despachos** será acedida nas interfaces de processamento do pedido.

Sempre que um pedido é distribuído, devolvido, aprovado ou há uma substituição de responsável, o utilizador em questão poderá deixar uma nota sobre o mesmo. Esta nota é chamada de despacho.

De seguida serão descritas as diferentes etapas do *workflow*.

Pedidos Novos

Esta é a etapa onde chegarão todos os novos pedidos realizados na plataforma, sendo o ponto de partida para o início da averiguação dos dados. Aqui será visível uma lista dos pedidos com algumas informações sobre os mesmos e as possíveis operações que se poderão realizar.

As informações a apresentar nesta etapa são: o **Código**; o **Tipo**; a **Entidade**; o **Criador**; a **Data**; e a **Tarefa**.

As operações são **Ver pedido**, onde serão apresentados todos os dados relativos a esse pedido; **Distribuir pedido**, onde permite que determinado pedido seja atribuído a um colaborador interno da plataforma para proceder à sua apreciação técnica; e **Devolver pedido** caso este não se encontre devidamente preenchido ou exista alguma inconformidade no mesmo.

Pedidos em Apreciação Técnica

Neste estado, o responsável poderá ver, à semelhança do anterior, uma lista com os pedidos aos quais foi selecionado para proceder à sua apreciação técnica.

Esta etapa consta com as informações: **Código**; **Tipo**; **Entidade**; **Criador**; **Data**; **Responsável**; e **Tarefa**.

As operações para os pedidos que se encontram neste estado serão **Ver pedido** e **Analisar pedido**, onde este será analisado de forma a garantir a conformidade nos dados inseridos.

Na interface da consulta do pedido, para os pedidos que estejam nesta etapa, terá também a operação **Substituir responsável**, no caso de o atual não conseguir realizar a análise desse pedido; ou no caso de ter sido selecionado o utilizador incorreto.

Pedidos em Validação

Nesta etapa, os pedidos serão validados pelo utilizador validador ou por outro utilizador por ele representado, responsável por averiguar se não foram cometidas falhas durante os processos anteriores, e dar permissão para que os dados do pedido sejam inseridos na plataforma, havendo então uma atualização na base de dados.

As informações para esta etapa serão: **Código**; **Tipo**; **Entidade**; **Criador**; **Data**; **Responsável**; e **Tarefa**.

Já as operações serão **Ver pedido** e **Validar pedido**, onde este será validado de forma a garantir que nenhum erro foi cometido.

Tal como na etapa anterior, na interface da consulta do pedido, para além da operação **Substituir responsável** terá também a operação **Reapreciar pedido**, no caso de existirem erros que necessitem de uma nova apreciação técnica.

Pedidos Devolvidos

No caso dos pedidos serem devolvidos, ou seja, pedidos que contêm dados com erros, os responsáveis pela submissão do pedido poderão analisar a causa pela qual este foi recusado e proceder, eventualmente, à alteração dos dados, corrigindo-os.

Desse modo, o responsável pela criação do pedido terá de realizar uma consulta à validação realizada e alterar o pedido, para que este seja novamente avaliado, voltando a transitar por todo o *workflow*.

Para tal, é necessário guardar o pedido, sendo então nesta etapa que os pedidos devolvidos permanecem. É apresentada uma lista como nas restantes etapas, com as informações: **Código; Tipo; Entidade; Criador; Data; e Tarefa.**

Para esta etapa há apenas uma tarefa, sendo ela a de **Ver pedido**, o que permite consultar os dados do pedido em questão.

Pedidos Aprovados

Este último estado é muito parecido com o anterior, sendo que a sua única diferença é o facto de manter o registo de pedidos inseridos no sistema. Tanto as informações como as tarefas são iguais.

Uma vez que a plataforma CLAV tem vindo a ser desenvolvida ao longo de vários anos, a interface de consulta de pedidos já se encontrava construída. Esta é a interface que apresenta os detalhes de um pedido sempre que se utiliza a operação **Ver pedido**. Desse modo, é apresentada esta mesma interface na [Figura 17](#). Nela é possível observar três áreas distintas. A primeira diz respeito à informação do pedido, nomeadamente o estado em que se encontra, a data de criação, a entidade à qual o criador pertence, o criador e o tipo.

A segunda é respetiva à distribuição do pedido, nomeadamente os despachos que foram atribuídos ao pedido em cada uma das etapas, bem como o estado em que o pedido se encontrava, a data, a hora e o responsável. Serve assim como um registo de etapas.

A terceira e última área apresenta os dados relativos ao pedido em questão. Num pedido de criação são apresentados os dados inseridos e, nos pedidos de alteração, apenas a informação alterada é apresentada.

Consulta do pedido: 2020-0000006

Estado

Data

Entidade

Criado Por

Tipo

Distribuição

Estado	Data	Responsável	Despacho
Submetido	2020-10-21 - 17:30	pg38425@alunos.uminho.pt	Submissão inicial
Distribuído	2020-10-21 - 17:31	pg38425@alunos.uminho.pt	Teste
Apreciado	2020-10-21 - 17:32	pg38425@alunos.uminho.pt	

Alteração da Entidade

i Entidade: AC - Águas de Coimbra, EM

Nome

Internacional

VOLTAR

Figura 17: Interface de consulta de um pedido no *workflow*.

4.1.3 Níveis de acesso às etapas do *workflow*

Uma vez descritos os diversos utilizadores da plataforma e as várias etapas do *workflow*, é importante entender também quais os utilizadores com permissão para consultar/gerir os pedidos que constam em cada uma das fases.

Cada uma destas etapas terá diferentes níveis de acesso e, conseqüentemente, apenas quem tem permissão para o efeito poderá consultar/gerir os pedidos que constam nessas mesmas fases.

Como já foi dito anteriormente, apenas os utilizadores com nível superior ou igual a 3.5 têm acesso ao *workflow*. No entanto, para cada etapa, estes níveis variam, nomeadamente para níveis superiores.

Com base nas etapas do *workflow*, é apresentada a [Tabela 9](#) que traduz a relação entre os níveis dos utilizadores e as operações que poderão ser realizadas por estes na respetiva etapa.

Nível Utilizador	Pedidos Novos	Pedidos em Apreciação Técnica	Pedidos em Validação	Pedidos Devolvidos	Pedidos aprovados
3.5	Consultar	Apreciar	Consultar	Consultar	Consultar
4	Gerir	Apreciar	Validar	Consultar	Consultar
5	Consultar	Consultar	Consultar	Consultar	Consultar
6	Gerir	Apreciar	Validar	Gerir	Gerir
7	Gerir	Apreciar	Validar	Gerir	Gerir
8	Gerir	Apreciar	Validar	Gerir	Gerir

Tabela 9: Relação entre os níveis dos utilizadores e as respetivas etapas do *workflow*.

Além das etapas, também algumas operações terão de ser controladas e, por isso, terá de ser aplicado um conjunto de níveis dos utilizadores que podem realizar essa operação. A Tabela 10 representa as operações possíveis e o conjunto de níveis dos utilizadores que as podem realizar.

Operação	Níveis
Distribuir pedido	4, 6, 7
Devolver pedido	4, 5, 6, 7
Analisar pedido	3.5, 4, 6, 7
Validar pedido	4, 6, 7
Substituir responsável	4, 5, 6, 7

Tabela 10: Operações restritas e os respetivos níveis permitidos.

No entanto, determinadas operações inerentes ao pedido em questão poderão ser realizadas pelo utilizador, ainda que este não tenha permissão para tal. Um exemplo é a situação em que um utilizador de nível 3.5, selecionado como responsável pela apreciação técnica de um pedido, embora não possua permissões para devolver pedidos, sendo este um pedido inválido e estando no seu domínio de apreciação, poderá então devolvê-lo.

4.1.4 Histórico de alterações e linguagem cromática

Após uma consulta mais detalhada a todos os requisitos do workflow descritos anteriormente, verificou-se que faltava um instrumento que facilitasse o processo de análise dos pedidos.

O modelo era suficiente para analisar os pedidos, no entanto, não eram perceptíveis as alterações realizadas, tanto nos pedidos de alteração, submetidos pelos criadores destes, como nas alterações realizadas nas várias etapas do workflow.

Desse modo, surgiram duas novas funcionalidades: uma denominada de **Linguagem cromática** e a outra de **Histórico de alterações**. A primeira funcionalidade tem como objetivo apresentar ao utilizador um esquema de cores que representam a operação realizada, tanto no momento de criação do pedido, como nas várias etapas do *workflow*. Isto é, existem três cores e cada uma representa uma operação tomada. A Tabela 11 descreve o significado de cada cor.

Cor	Descrição
Verde	- Submissão de novos dados no momento de criação de um pedido; - Validação de uma linha de dados relativa a um pedido dentro do <i>workflow</i> .
Amarelo	- Alteração de dados no momento de criação de um pedido de alteração ou extinção; - Alteração de uma linha de dados relativa a um pedido dentro do <i>workflow</i> .
Vermelho	- Invalidação de uma linha de dados relativa a um pedido dentro do <i>workflow</i> .

Tabela 11: Descrição das cores no *workflow*.

A segunda funcionalidade, denominada por **Histórico de alterações**, apresenta ao utilizador o histórico com as alterações realizadas. Estas alterações dizem respeito não só às alterações realizadas durante as várias etapas do *workflow*, mas também às alterações realizadas pelo criador do pedido, no caso de um pedido de alteração de dados. Esta funcionalidade será bastante útil, não só por fazer uso da **Linguagem cromática**, o que permite identificar rapidamente as alterações e as decisões tomadas, mas também por permitir navegar entre as várias etapas do **Histórico de alterações**, representativas das várias etapas do *workflow*, e consultar as possíveis notas que são deixadas pelos utilizadores, responsáveis pela respetiva etapa do *workflow*, relativas a cada linha de dados do pedido.

Ainda no contexto do **Histórico de alterações**, poderá ser necessário comparar duas etapas do **Histórico de alterações** lado a lado, em vez de visualizar os dados em forma de *slideshow*. Desse modo, é necessário implementar duas abordagens desta funcionalidade. Uma onde os dados do histórico relativos a cada etapa do *workflow* são apresentados em forma de *slideshow*, isto é, em cada ecrã são apresentados os dados relativos a uma etapa; e uma outra que permite visualizar, em forma de tabela, os dados relativos a duas etapas distintas do histórico, escolhidas pelo utilizador, implementando também a **Linguagem cromática**.

Uma vez que esta funcionalidade apresenta dados importantes relativamente ao pedido que transita pelo *workflow*, também nesta terá de ser implementado um controlo de acesso. Desse modo, e seguindo tanto a congruência do controlo de acesso descrito na [Tabela 6](#) e na [Tabela 10](#), bem como o facto de que esta funcionalidade poderá ser consultada em todas as etapas do *workflow*, os níveis com acesso deverão ser os 4, 6 e 7. Assim, garante-se que apenas os utilizadores com os respetivos níveis, têm acesso a esta funcionalidade, mantendo toda a gestão interna do *workflow* o mais privada possível, evitando que os utilizadores com níveis inferiores acedam às alterações e decisões tomadas pelos responsáveis do processamento dos pedidos.

4.1.5 Modelo do *workflow*

Após analisadas todas as necessidades e exigências para a construção do fluxo de processos, era evidente a necessidade de um *workflow* específico ao problema em mãos. Trata-se

de um *workflow* produtivo, uma vez que o fluxo é repetido para cada pedido, mas com elevada complexidade. Esta complexidade deve-se ao facto de existirem vários tipos de pedidos que, conseqüentemente, implicam diferentes análises, bem como a existência de vários utilizadores envolvidos nessa análise.

Criou-se então o modelo do *workflow* utilizando as normas do BPMN e seguindo os requisitos descritos anteriormente. Estes modelos mostram de uma forma mais clara todas as operações presentes em cada etapa, bem como o natural fluxo dos dados.

Seguindo a ordem natural do fluxo dos dados, apresenta-se primeiramente na **Figura 18** o diagrama para a etapa de **Pedidos Novos**.

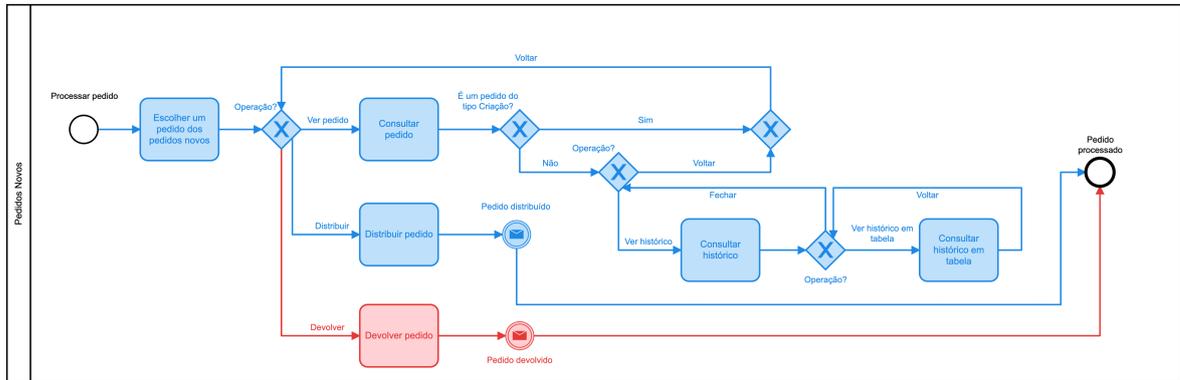


Figura 18: Modelo BPMN da etapa de **Pedidos Novos**.

A etapa seguinte é a de **Pedidos em Avaliação Técnica**, representado na Figura 19.

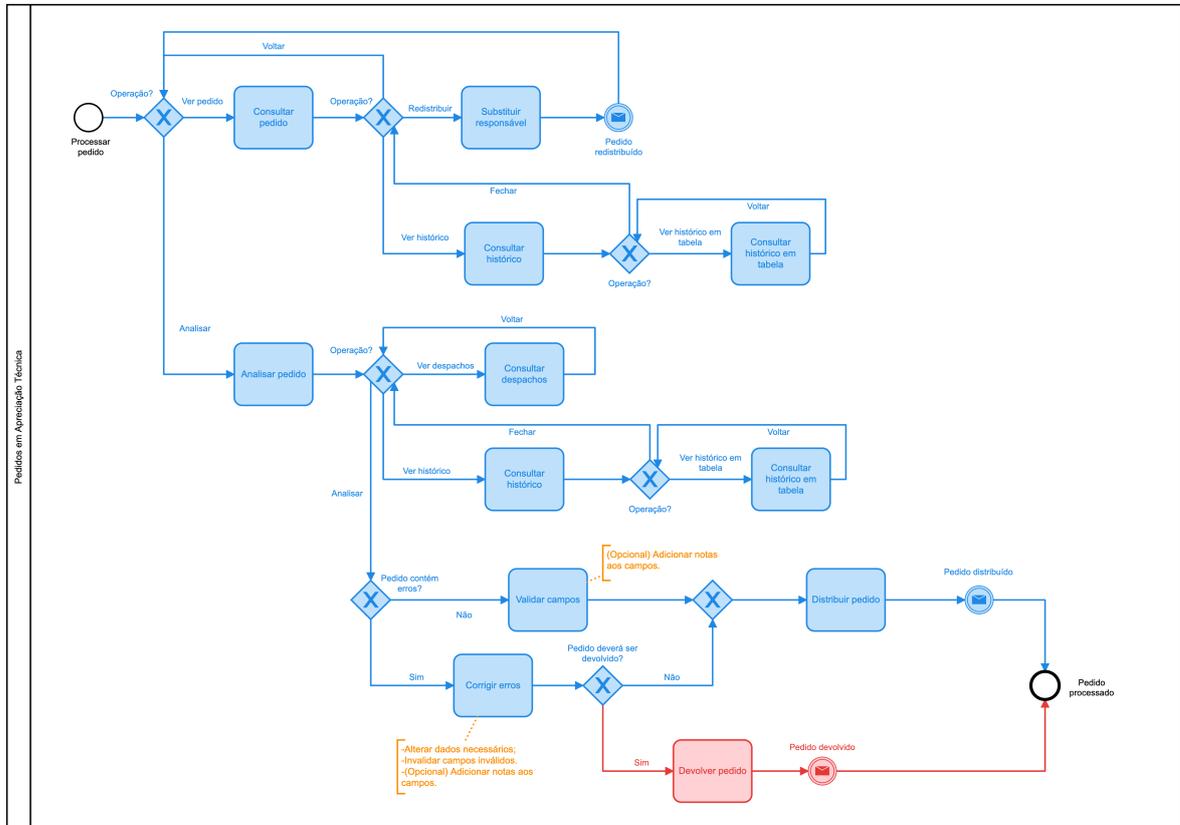


Figura 19: Modelo BPMN da etapa de **Pedidos em Avaliação Técnica**.

Seguidamente, na Figura 20, apresenta-se o modelo da etapa de **Pedidos em Validação**.

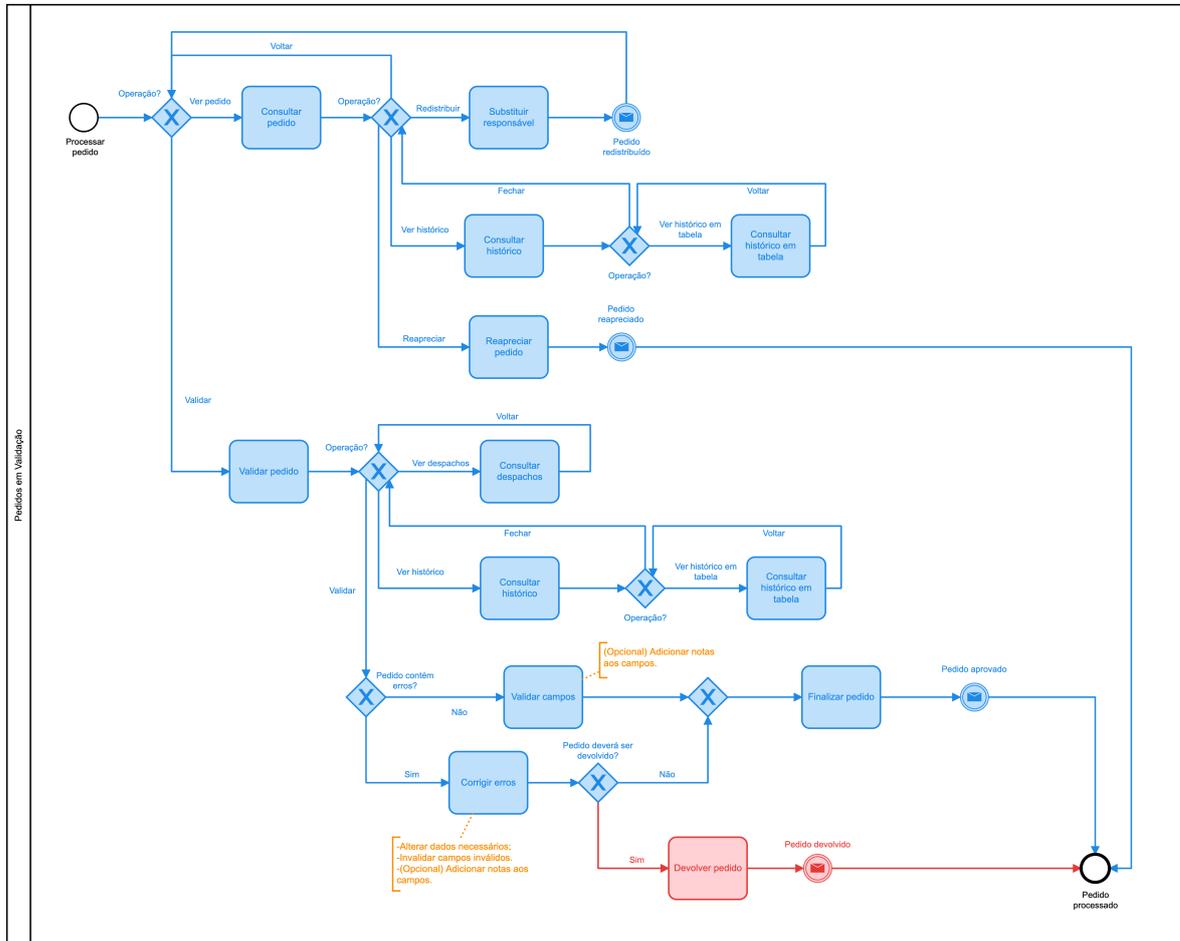


Figura 20: Modelo BPMN da etapa de **Pedidos em Validação**.

Como destino final de um pedido que tenha sido devolvido, a Figura 21 é o modelo para os pedidos que estejam no estado **Pedidos Devolvidos**.

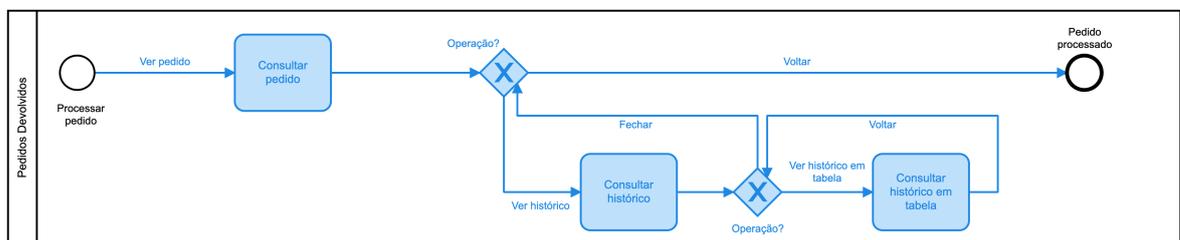


Figura 21: Modelo BPMN da etapa de **Pedidos Devolvidos**.

No caso dos **Pedidos Aprovados**, o modelo BPMN é idêntico ao dos **Pedidos Devolvidos** e é apresentado na **Figura 22**.

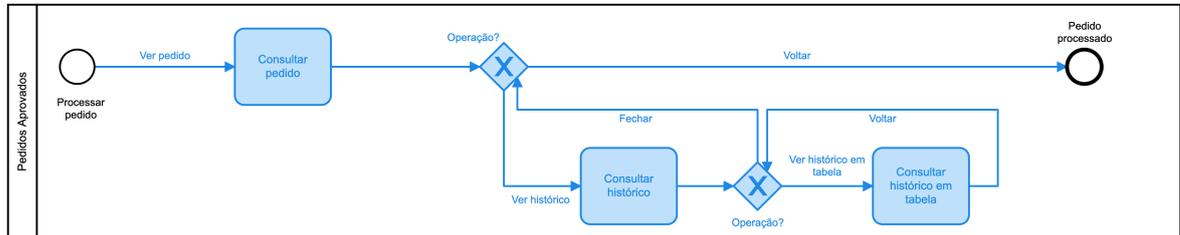


Figura 22: Modelo BPMN da etapa de **Pedidos Aprovados**.

Os restantes diagramas, tais como o modelo em BPMN completo, **Figura 47**, e os modelos complementares, **Figura 45** e **Figura 46**, encontram-se no **Anexo A**.

Para todos os diagramas apresentados, a cor amarela representa as notas associadas às etapas em questão; e a cor vermelha foi escolhida para identificar o fluxo dos pedidos a devolver.

4.2 TECNOLOGIAS UTILIZADAS

Para a realização deste projeto e, uma vez que este já se encontra em desenvolvimento há alguns anos, seguiu-se a arquitetura representada na **Figura 23**, estando esta dividida em três camadas. Como camada central encontra-se a *Application programming interface* (API) de dados, responsável por responder a todos os pedidos da camada da interface e comunicar com a camada da persistência dos dados.

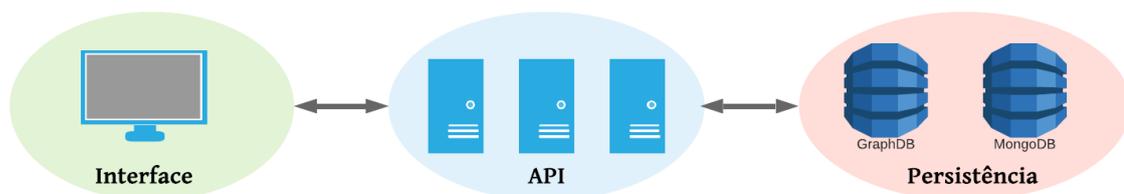


Figura 23: Arquitetura da plataforma.

Cada camada da arquitetura tem uma função específica na plataforma. Começando pela API, esta segue uma arquitetura *Representational state transfer* (REST) e é desenvolvida em NodeJS com ExpressJS, sendo estas duas *frameworks* de JavaScript. A interface, sendo o ponto de acesso disponível aos utilizadores, é desenvolvida em VueJS, sendo também esta uma *framework* de JavaScript. Por fim, a camada de persistência dos dados, é composta por duas bases de dados distintas, uma em GraphDB para os dados principais da plataforma e outra em MongoDB para os dados temporários.

4.2.1 GraphDB

GraphDB é uma base de dados de grafos baseada em RDF que se afirma ser eficiente, robusta e altamente escalável. É caracterizada pela implementação de especificações da W3C, tal como a linguagem SPARQL, sendo esta a linguagem de *query*, e suporta todos os formatos de serialização do RDF. (Ontotex, s.d.)

É também caracterizada pelo facto de lidar com consultas massivas em tempo real e permite que os utilizadores formem novas semânticas a partir dos factos existentes.

Dadas estas características e, uma vez que a plataforma faz uso das especificações e recomendações da W3C, tendo como base uma ontologia definida, bem como o facto de que todos os dados da plataforma estão interligados, como as Entidades com as Tipologias, ou a Legislação com os Processos de Negócio, esta ligação é facilmente representável pela forma de grafos. Dessa forma, esta é a principal base de dados do sistema onde serão armazenados todos os dados gerados.

4.2.2 MongoDB

O MongoDB é uma base de dados *open-source* NoSQL, orientada a documentos, que faz uso do formato *Binary JSON* (BSON), sendo este uma estrutura de dados idêntica ao JSON.

É uma base de dados bastante flexível, permitindo guardar informação, mesmo que esta não se encontre totalmente completa. Esta é uma característica importante para o projeto, uma vez que esta base de dados é utilizada na plataforma para armazenar dados incompletos, como o trabalho ainda não terminado e que tenha sido guardado para continuar mais tarde, ou dados que se encontram em validação para poderem, posteriormente, ser inseridos na base de dados principal.

Além dos dados temporários são também guardados os dados relativos aos utilizadores do sistema nesta base de dados. Esta abordagem permite isolar os dados relativos aos utilizadores da base de dados ontológica, desse modo, a base de dados principal mantém apenas os dados que dizem respeito à ontologia.

Uma vez que toda a plataforma faz uso de *frameworks* de JavaScript, a integração com esta base de dados é simples e rápida.

4.2.3 NodeJS + ExpressJS

NodeJS é uma *framework open-source* baseada em JavaScript, utilizada para criar servidores altamente escaláveis. Esta *framework* funciona de forma assíncrona, o que significa que serve a pedidos mas não fica à espera da resposta, segue para o próximo pedido e apenas responde ao anterior assim que receber um evento com os dados solicitados.

Já o ExpressJS é uma *framework* que faz uso das funcionalidades do NodeJS, permitindo criar API's do lado do servidor com a arquitetura MVC. Estas API's apresentam um elevado grau de robustez e uma grande quantidade de funcionalidades para a Web e para aplicações móveis.

Estas duas *frameworks* juntas permitem criar uma API sólida e robusta, capaz de lidar com todos os pedidos do sistema com grande rapidez.

4.2.4 VueJS + Vuetify

Para a criação da interface do utilizador foi utilizada a *framework* VueJS, também esta baseada em JavaScript e *open-source*. Permite criar interfaces reativas e flexíveis, de forma a que o utilizador aceda e manipule os dados consumidos da API sem grande esforço.

A grande popularidade desta *framework* deve-se à sua simplicidade no desenvolvimento de páginas Web e à facilidade de integração em aplicações existentes.

Já o Vuetify é uma biblioteca de VueJS que apresenta diversos tipos de materiais e estilos, aumentando a velocidade de desenvolvimento uma vez que reduz o tempo necessário com o *design* de componentes.

4.2.5 Preservação dos dados do workflow

Apenas pedidos que impliquem alterações na base de dados, ou seja, pedidos de criação, alteração ou eliminação de informação, são enviados para o *workflow*. Desta forma, após a realização de um pedido que tenha de percorrer o *workflow*, este é armazenado em MongoDB.

O uso desta base de dados para armazenar os dados do *workflow*, tal como foi referido na [subsecção 4.2.2](#), deve-se à sua vantagem de permitir guardar os dados do *workflow*, mesmo que estes se encontrem incompletos ou com erros, evitando, assim, a necessidade de criar transações na base de dados principal com dados por analisar.

Sem o uso de uma base de dados secundária, durante as várias fases do *workflow* descritas anteriormente, na eventualidade da existência de alteração de dados, implicaria uma nova transação na base de dados, o que poderia comprometer a performance do sistema.

Assim que um pedido chegar à última fase de análise do *workflow*, na maioria das situações, nomeadamente sempre que os dados analisados se encontrem corretos, irá implicar uma transação na base de dados principal, estando ela definida em GraphDB, como descrito na [subsecção 4.2.1](#).

Como forma de exemplo, no caso da criação de uma nova Entidade, assim que esta percorra todo o *workflow*, será então introduzida na base de dados, criando, dessa forma, uma transação. A transação para a criação de Entidades encontra-se disponível na API e representa-se de seguida:

Query SPARQL para inserção de uma Entidade na base de dados GraphDB.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX clav: <http://jcr.di.uminho.pt/m51-clav#>
INSERT DATA {
    clav:ent_${ent.sigla} rdf:type owl:NamedIndividual, clav:Entidade ;
    clav:entEstado "${ent.estado}";
    clav:entSIOE "${ent.sioe}";
    clav:entSigla "${ent.sigla}";
    clav:entDesignacao "${ent.designacao}";
    ${ent.tipologias.map(
        tip => "clav:pertenceTipologiaEnt clav:tip_"
            + tip.split("_").pop() + ";" ).join("\n"
        )
    }
    clav:entInternacional "${ent.internacional}";
}
```

Esta é uma *query* em SPARQL e, tal como abordado na [subsecção 3.1.5](#), segue os padrões da W3C. A *query* insere uma entidade no sistema com os campos **estado**, **SIOE**, **sigla**, **designação**, **internacional** e a lista de **tipologias** associadas à mesma. É possível observar tanto o uso de vocabulário dos padrões OWL e RDF, definidos pelos prefixos `rdf` e `owl`, como também o uso de vocabulário da ontologia definida, em trabalhos anteriores, para a plataforma CLAV.

4.3 INTERFACE DESENVOLVIDA

Esta implementação baseia-se, fundamentalmente, no desenvolvimento da interface do utilizador, uma vez que os métodos existentes na API de dados se mostravam ser suficientes, devolvendo a informação necessária para a construção das interfaces inerentes ao *workflow*, bem como a manipulação dos dados dentro deste.

Desse modo, no início da implementação das várias etapas do *workflow*, analisou-se todo o conteúdo existente e reaproveitou-se, sempre que possível, os componentes existentes implementados com as tecnologias **VueJS + Vuetify**.

Assim sendo, já havia sido criada a interface tanto de visualização dos pedidos, descrita anteriormente, como também a interface com a listagem das várias etapas do *workflow*, como se pode observar na [Figura 24](#).

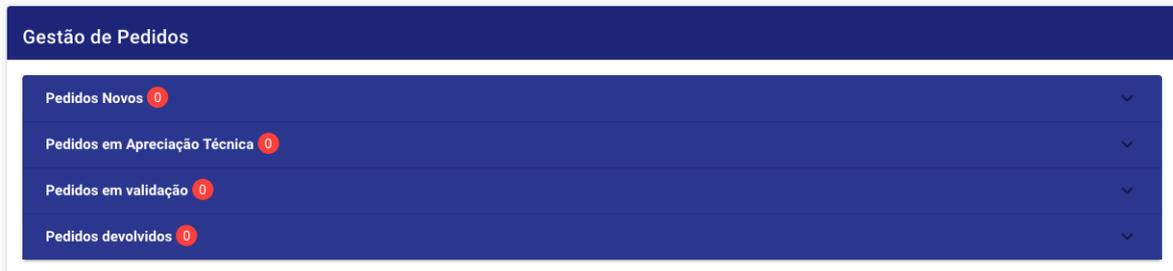


Figura 24: Versão inicial da interface com as várias fases do *workflow*.

No entanto, esta interface era insuficiente, uma vez que não cumpria todas as necessidades do modelo anteriormente definido. Adaptou-se então a interface, de modo a que esta cumprisse com os requisitos, nomeadamente, acrescentou-se a nova etapa **Pedidos Aprovados**, melhorou-se um pouco o *design*, e ainda se adicionou informações em falta nas tabelas dos pedidos em cada etapa, nomeadamente o responsável pelo pedido e as operações possíveis. Além disso, foram implementadas também as permissões de acesso para cada uma dessas operações e uma barra de pesquisa, a qual permite procurar, de forma mais específica, por um pedido na respetiva etapa. A [Figura 25](#) mostra algumas dessas alterações.



Figura 25: Versão final da interface com as várias etapas do *workflow*.

Num primeiro contacto com a funcionalidade da barra de pesquisa, para cada etapa do *workflow*, observou-se a necessidade de implementar uma forma de se preservar o termo pesquisado pelo utilizador, quando este usa a barra de pesquisa para filtrar os pedidos, consulta um deles e volta à interface de gestão dos pedidos. Além disso, era necessário também expandir a etapa referente ao pedido pesquisado. A [Figura 26](#) mostra o bloco

referente à etapa de **Pedidos em Avaliação Técnica** expandido e a barra de pesquisa preenchida com o termo "Criação".

Código	Tipo	Entidade	Criador	Data ↓	Responsável	Tarefa
2020-0000010	Classe_N1 - Criação	DGLAB	matiasjcatia@gmail.com	23-10-2020	Formação DGLAB	🔍
2020-0000042	Classe_N3 - Criação	DGLAB	alexandra.lourenco@dglab.gov.pt	05-11-2020	Cátia Trindade	🔍

Pedidos por página 10 1 - 2 de 2 < >

Figura 26: Interface de consulta dos pedidos na etapa de **Pedidos em Avaliação Técnica**.

Para a implementação desta funcionalidade, utilizou-se um recurso existente nos navegadores, o qual se denomina por *local storage*. A abordagem passa por guardar o número do bloco que foi expandido e qual o termo que foi escrito na barra de pesquisa, sempre que um pedido é consultado. Desse modo, sempre que um utilizador volta para a lista de consulta dos pedidos, vindo da interface de consulta do pedido, o respetivo bloco será expandido novamente e a barra de pesquisa preenchida. O excerto de código seguinte mostra a implementação desse recurso.

Utilização da *local storage* para preservação do estado da interface de gestão de pedidos no *workflow*.

```
showPedido(pedido) {
  localStorage.setItem(
    "pesquisa-pedidos",
    JSON.stringify({
      painel: 0,
      pesquisa: this.procurar,
      pagina: this.paginaTabela,
      limpar: true,
    })
  );
},
```

Esta funcionalidade foi implementada para todas as etapas do *workflow*, uma vez que todas apresentam a mesma estrutura, nomeadamente uma tabela com os diversos pedidos nesse estado e a barra de pesquisa.

4.3.1 Implementação dos Níveis de Acesso

Para a implementação dos níveis de acesso, nomeadamente o controlo das operações e das funções que cada utilizador tem disponíveis, criou-se um ficheiro de configuração com os diversos níveis, refletido na [Tabela 10](#), adicionando também os níveis de acesso ao **Histórico de alterações**. O ficheiro pode ser consultado no excerto abaixo.

Ficheiro de configuração com os vários níveis de acesso.

```
const CONSTS = module.exports;

CONSTS.NIVEIS_DEVOLVER_PEDIDO = [4, 5, 6, 7];
CONSTS.NIVEIS_DISTRIBUIR_PEDIDO = [4, 6, 7];
CONSTS.NIVEIS_ANALISAR_PEDIDO = [3.5, 4, 6, 7];
CONSTS.NIVEIS_VALIDAR_PEDIDO = [4, 6, 7];
CONSTS.NIVEIS_SUBSTITUIR_RESPONSAVEL = [4, 5, 6, 7];
CONSTS.NIVEIS_CONSULTAR_WORKFLOW = [3.5, 4, 5, 6, 7];
CONSTS.NIVEIS_CONSULTAR_HISTORICO = [4, 6, 7];
```

Uma vez que todos os utilizadores autenticados na plataforma apresentam um *token*, onde consta o nível do utilizador, facilmente se consegue verificar se este tem permissão para aceder a alguma funcionalidade ou operação em específico. Um exemplo desta situação pode ser verificado nos excertos seguintes, onde se utilizou o ficheiro de configuração, com os níveis definidos, e se verificou se o nível do utilizador, atualmente autenticado, se encontra na lista de níveis com acesso à operação **Validar pedido**. Caso esta condição seja verdadeira, a operação é apresentada ao utilizador.

Método para verificar se o utilizador tem permissão para validar um pedido.

```
temPermissaoValidar() {
  return NIVEIS_VALIDAR_PEDIDO.includes(this.$userLevel());
},
```

Condição de apresentação da operação Validar pedido.

```
<v-tooltip bottom v-if="temPermissaoValidar()">
  (...)
  <span>Validar pedido...</span>
</v-tooltip>
```

4.3.2 Implementação do Histórico de Alterações e da Linguagem Cromática

Implementou-se numa única estrutura, tanto o **Histórico de alterações** como a **Linguagem cromática**, uma vez que estes são inerentes um ao outro. O **Histórico de alterações** preserva, para cada etapa, um conjunto de informações relativo a cada dado do pedido, incluindo a informação da **Linguagem cromática**. Essa estrutura pode ser consultada de seguida.

Estrutura do Histórico de alterações e da Linguagem cromática².

```
[
  {
    "designacao":{
      "cor":"verde",
      "dados":"Camara Municipal de Arcos de Valdevez",
      "nota":null
    },
    "sioe":{
      "cor":"verde",
      "dados":"1111",
      "nota":null
    }
  },
  {
    "designacao":{
      "cor":"amarelo",
      "dados":"Câmara Municipal dos Arcos de Valdevez",
      "nota":null
    },
    "sioe":{
      "cor":"vermelho",
      "dados":"1111",
      "nota":"Parece-me errado o SIOE."
    }
  }
]
```

Esta estrutura de dados desenvolvida em JSON, sendo constituída por um *array* de objetos, permitiu que cada um desses objetos representasse, sequencialmente, uma etapa do *workflow*. Dessa forma, analisando cada objeto individualmente, observa-se que estão representadas as *keys*, dos campos do pedido e, cada um deles apresenta também uma estrutura com algumas informações. Essas informações indicam: a cor a apresentar na **Linguagem cromática** para esse campo; o valor desse campo para o respetivo estado; e uma nota associada ao respetivo campo, caso se aplique.

² Alguns campos foram omitidos de forma a comprimir o tamanho do exemplo apresentado.

Como cada objeto dentro do *array* representa uma etapa do *workflow*, observa-se que de uma etapa para a outra, os objetos das *keys* "designacao" e "sioe" são diferentes. Observando o objeto da designação, o valor da propriedade "cor" mudou de verde para amarelo, o que significa, segundo a **Linguagem cromática**, que o valor da propriedade "dados" foi alterado. Como se pode verificar, de facto o valor foi alterado de "Camara Municipal de Arcos de Valdevez" para "Câmara Municipal dos Arcos de Valdevez". Não havendo nenhuma nota adicionada, a propriedade "nota" mantém o valor *null*. Por outro lado, o campo "sioe", foi invalidado, ou seja, foi assinalado como vermelho. Isto é, o valor da propriedade "cor" foi alterado de verde para vermelho, não havendo alteração, como expectável, do valor da propriedade "dados". Neste campo, no segundo objeto, foi atribuída uma nota com informações sobre a decisão tomada e, desse modo, o valor da propriedade "nota" passou a ser "Parece-me errado o SIOE."

Com as estruturas apresentadas, criaram-se as interfaces que fazem uso destas e apresentam, de forma simples, o histórico ao utilizador. A [Figura 27](#) mostra a interface de consulta do **Histórico de alterações** de um pedido.

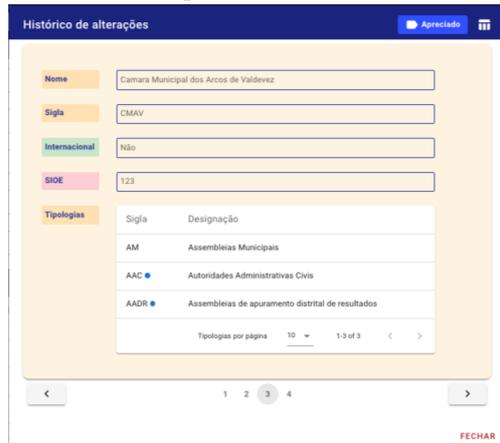
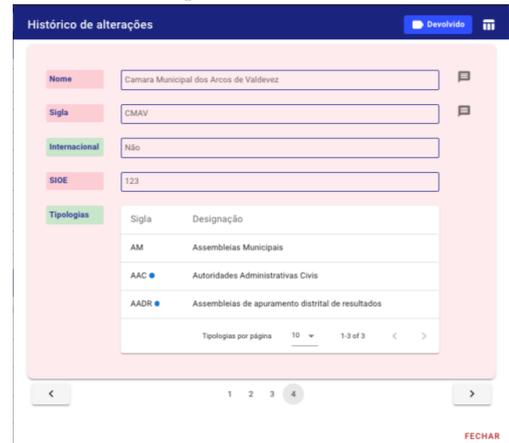
(a) Histórico do pedido no estado **Submetido**.(b) Histórico do pedido no estado **Distribuído**.(c) Histórico do pedido no estado **Apreciado**.(d) Histórico do pedido no estado **Devolvido**.

Figura 27: Interface de consulta do **Histórico de alterações** de um pedido nos vários estados.

Na interface representada pela [Figura 27d](#), é visível um *icon* ao lado do valor dos campos "Nome" e "Sigla". Este *icon*, informa o utilizador que existe uma nota associada a esse campo. Para a consultar, basta clicar sobre o *icon* e surgirá uma janela com a nota.

Nestas interfaces, no canto superior direito, encontra-se um botão que permite consultar o **Histórico de alterações** em formato de tabela. Para tal, o utilizador é deparado com a interface representada pela [Figura 28](#), onde terá de escolher de uma lista de etapas, as duas que pretende comparar.

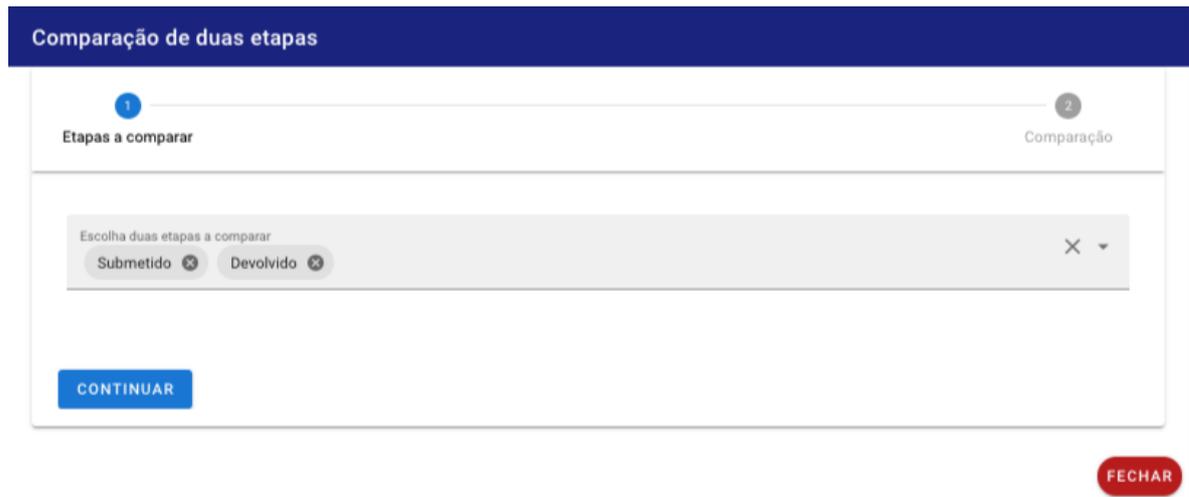


Figura 28: Ecrã de escolha das duas etapas do pedido a comparar.

Assim que o utilizador selecione duas etapas e clique no botão continuar, surgirá a comparação desta em formato de tabela, ou seja, os dados aparecem lado a lado. Desse modo, e através do auxílio da **Linguagem cromática**, é possível comparar, de uma forma simples, as alterações e as ações tomadas. A [Figura 29](#) mostra a janela com essa comparação.

Comparação de duas etapas		
Campo	Submetido	Devolvido
Nome	✓ Camara Municipal de Arcos de Valdevez	✗ Camara Municipal dos Arcos de Valdevez
Sigla	✓ CMA2V	✗ CMAV
Internacional	✓ Não	✓ Não
SIOE	✓ 123	✗ 123
Data de Criação	⊘ Campo não preenchido	⊘ Campo não preenchido
Data de Extinção	⊘ Campo não preenchido	⊘ Campo não preenchido
Tipologias	✓ • AM	✓ • AM • AAC • AADR

VOLTAR

FECHAR

Figura 29: Janela de consulta da comparação de duas etapas do histórico.

4.3.3 Implementação das etapas do workflow

Tal como para o **Histórico de alterações**, também aqui foi criada uma estrutura de dados que preserva os vários estados do pedido, de forma sequencial.

De acordo com a estrutura abaixo apresentada, é possível observar que cada objeto dentro do *array* "distribuicao" possui uma propriedade denominada "estado". Esta propriedade reflete em que etapa o pedido se encontrava em determinado estado no tempo, no entanto, o último objeto representa o estado atual do pedido. Desse modo, utilizou-se esta abordagem para distribuir os pedidos pelas várias etapas na interface de consulta do *workflow* representado pela [Figura 25](#).

Estrutura de dados da distribuição dos pedidos.

```

"distribuicao": [
  {
    "data": "2020-11-04T12:38:31.240Z",
    "_id": "5fa2a0c7ba6cff01a52df6d0",
    "estado": "Submetido",
    "responsavel": "pg38425@alunos.uminho.pt",
    "despacho": "Submissão inicial"
  },
  {
    "proximoResponsavel": {
      "nome": "Rui Araújo",
      "entidade": "ent_II",
      "email": "pg38425@alunos.uminho.pt"
    },
    "data": "2020-11-04T12:39:36.064Z",
    "_id": "5fa2a108ba6cff01a52df6f7",
    "estado": "Distribuído",
    "responsavel": "pg38425@alunos.uminho.pt"
  },
  {
    "proximoResponsavel": {
      "nome": "Rui Araújo",
      "entidade": "ent_II",
      "email": "pg38425@alunos.uminho.pt"
    },
    "data": "2020-11-04T12:39:55.934Z",
    "_id": "5fa2a11cba6cff01a52df707",
    "estado": "Apreciado",
    "responsavel": "pg38425@alunos.uminho.pt"
  }
],

```

Além disso, nesta estrutura, à exceção do primeiro e do último objeto, submissão e finalização do pedido respetivamente, todos os restantes apresentam um outro objeto, denominado de "proximoResponsavel", onde constam informações relativas ao utilizador responsável por processar o pedido no estado seguinte ao atual.

Seguindo o mesmo conceito, a propriedade "responsavel", no primeiro objeto, representa o criador do pedido. No entanto, nos restantes objetos reflete o responsável pelo processo dessa etapa do pedido.

Posto isto, quando um pedido transita de uma etapa para outra, é criado um novo objeto e adicionado ao final da lista da distribuição. Desse modo, garante-se que o pedido em questão é atualizado e o último estado reflete sempre a etapa em que o pedido se encontra.

Sempre que um pedido é submetido na plataforma, é adicionado o primeiro objeto na distribuição. Este objeto preserva a data dessa submissão, o estado, previamente explicado, o responsável pela submissão do pedido e o despacho. Em todos os objetos está também presente a propriedade "_id" que representa o ID da distribuição e é gerado pela API de dados.

Quando um pedido se encontra no estado **Submetido**, este será apresentado na etapa de **Pedidos Novos**. Nesta etapa, o utilizador terá duas opções possíveis, devolvê-lo ou distribuí-lo.

Se a operação for para devolver, será adicionado um novo objeto na distribuição, idêntico ao primeiro objeto, mas com novos dados: uma nova data, sendo ela a data da operação tomada, neste caso a devolução; o estado que reflete a operação realizada, neste caso "Devolvido"; o responsável pela operação executada; e um despacho que pode ou não estar presente, uma vez que é de carácter opcional.

No caso do utilizador distribuir o pedido, terá de selecionar um utilizador de uma lista de utilizadores do sistema, como mostra a [Figura 30](#). Esta interface mostra apenas os utilizadores que podem realizar a etapa seguinte à que o pedido se encontra, neste caso, a análise do pedido.

Distribuição do pedido 2020-000007

ℹ Clique sobre a linha da tabela para selecionar o utilizador a quem deve ser atribuída a análise do pedido 2020-000007.

Procurar utilizador 🔍

Nome↓	Entidade
Zélia Gomes	DGLAB
Rui Araújo	II
Ricardo Canela	BdP
Miguel Ferreira	KEEP
Maria Rita Gago	DGLAB

Utilizadores por página 5 1-5 of 28 < >

CANCELAR DISTRIBUIR

Figura 30: Interface de distribuição do pedido.

Com esta informação, são gerados os dados relativos ao próximo responsável, o qual preserva o seu nome, o ID da entidade a que este pertence e o seu email. Estes dados são inseridos no objeto da nova distribuição adicionada, através da propriedade "proximoResponsavel". Além dessa informação, também as restantes são criadas, nomeadamente, a data da operação, o estado do pedido, neste caso "Distribuído" e o responsável pela realização da operação, podendo também ser deixado um despacho, ainda que este seja opcional. A interface que permite criar um despacho pode ser consultada na [Figura 31](#).

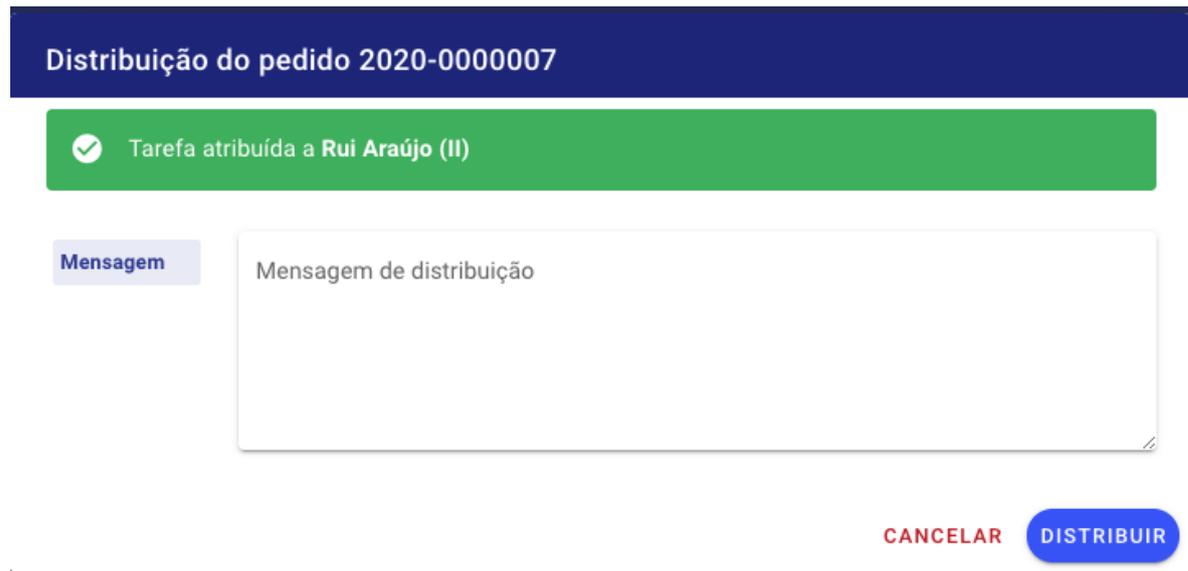


Figura 31: Interface para adicionar um despacho na distribuição de um pedido.

Estes passos são repetidos para todas as restantes etapas do pedido até que seja finalizado ou devolvido. Nessa situação, deixará de ser adicionado o objeto do "proximoResponsavel", adicionando-se apenas as restantes informações.

Após a implementação das estruturas de dados apresentadas e dos respetivos métodos, foi necessário implementar as interfaces para a apreciação e validação dos pedidos, que também fazem uso dessas estruturas.

Estas interfaces têm exigências associadas. A primeira delas, e talvez a mais importante, é garantir que é apresentada a **Linguagem cromática**, de forma a que o utilizador tenha a perceção, não só das ações que por ele são executadas, mas também das que foram executadas antes dele. As outras exigências dizem respeito à consulta dos despachos atribuídos aos pedidos e à consulta do histórico.

Criou-se então a interface, representada pela [Figura 32](#), a qual representa a etapa de análise de um pedido.

Análise do pedido: 2020-000062 - Criação de Entidade

Nome: Câmara Municipal dos Arcos de Valdevez

Sigla: CMAV

Internacional: Sim

SIOE: [Campo não preenchido na submissão do pedido]

Data de Criação: [Campo não preenchido na submissão do pedido]

Data de Extinção: [Campo não preenchido na submissão do pedido]

Tipologias

ADICIONAR TIPOLOGIAS

Sigla	Designação	Operação
AM	Assembleias Municipais	
AAC	Autoridades Administrativas Civas	

Tipologias por página: 10 1-2 of 2

DEVOLVER ENCAMINHAR

Figura 32: Interface de análise de um pedido.

Começando pela barra azul, esta encontra-se dividida em duas áreas importantes. A primeira contém algumas informações acerca do pedido, nomeadamente a operação, o número e o tipo deste. Já na segunda parte, do lado direito, encontram-se dois botões: o primeiro permite abrir a interface de consulta do **Histórico de alterações**; e o segundo abre a interface de consulta dos despachos. Esta é uma interface muito simples, onde é apresentado o estado do pedido em que o despacho foi escrito; o utilizador que o criou; e o despacho propriamente dito. Esta interface pode ser consultada na [Figura 33](#).

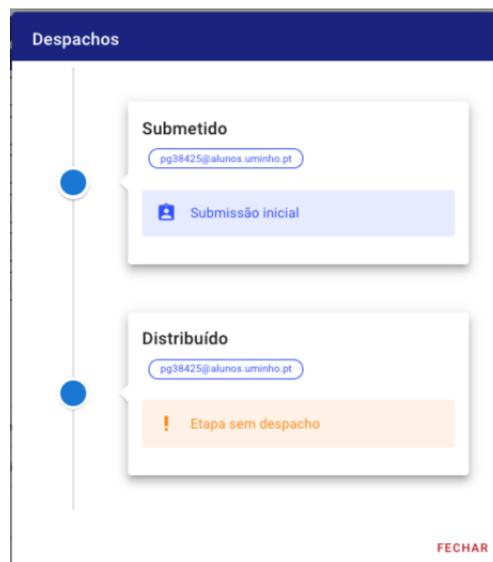


Figura 33: Interface de consulta dos despachos atribuídos ao respetivo pedido.

No corpo da interface há três partes importantes. No lado esquerdo, as *labels* de cada campo apresentam uma cor, sendo que esta representa a **Linguagem cromática** e, como já foi explicado, cada cor tem um significado. Segue-se a linha com o respectivo valor desse campo, podendo ser um texto ou uma lista, sendo que as listas são representadas em forma de tabela. Por fim, do lado direito, encontram-se as operações que podem ser executadas pelo respectivo responsável pela análise do pedido. Estas operações são, respetivamente: validar; invalidar; editar; adicionar nota.

Nesta parte, foram realizadas algumas implementações, de forma a garantir a conformidade dos dados e a melhor experiência do utilizador. Essas implementações são listadas de seguida:

- Quando o utilizador executa uma das três operações possíveis, é executada uma animação de *fade in* na respetiva *label*, de forma a mostrar ao utilizador que a operação foi executada. Além da animação, também a cor é alterada para a respetiva cor da operação executada.
- Quando um novo dado é adicionado na tabela, escolhido de uma lista, este é marcado com um ponto azul, de forma a informar o utilizador que aquele dado foi adicionado, ou seja, não fazia parte do pedido recebido. À semelhança do que acontece neste caso, quando um dado é removido da tabela, esse dado é automaticamente inserido na nota associada a essa *label*, podendo esta situação ser consultada na [Figura 34](#).
- Se o utilizador pretender alterar algum dado, é apresentada uma janela onde este poderá realizar essa alteração. No entanto, com base no tipo de dados, essa janela poderá ser diferente, isto é, se os dados forem texto, aparecerá um bloco de texto. No entanto, se o tipo de dados for controlado com valores pré-definidos, esse bloco será uma *select box*, com todas as opções possíveis. É também importante referir que sempre que uma linha de dados é alterada, as operações de validar e invalidar são omitidas, uma vez que não é possível alterar um dado e validá-lo ou invalidá-lo na mesma etapa do *workflow*. Esta situação é visível na linha de dados para o campo "Internacional".
- Nas situações em que um campo não tenha sido preenchido no momento da criação do pedido, este aparecerá na interface de análise mas identificado com a mensagem "[Campo não preenchido na submissão do pedido]", de forma a indicar ao utilizador que o criador não o preencheu, mas permitindo ao responsável pela análise proceder ao seu preenchimento, se necessário.

Ainda nesta interface, estão presentes dois botões: um para a devolução do pedido, no caso de este ter alguma inconformidade e necessite de ser devolvido, podendo ser escrito um despacho; e outro para o encaminhamento deste para a fase seguinte, neste caso, a validação.

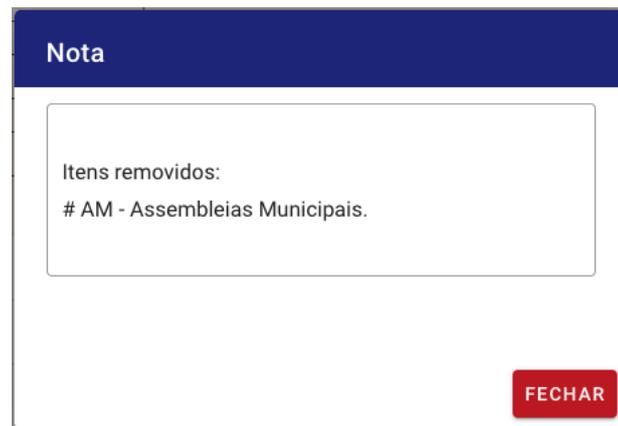


Figura 34: Exemplo de uma nota com a indicação dos dados removidos.

Após clicar no botão de encaminhamento, surgirá novamente a janela para a escolha do próximo responsável e a janela para escrever um despacho.

Apresenta-se de seguida, na [Figura 35](#), a interface de validação de um pedido. Como se pode observar, tanto esta como a interface de análise são idênticas, todas as funcionalidades e operações estão aqui presentes. A única diferença visível é o botão azul, que aqui tem o texto "Finalizar", uma vez que traduz a operação de finalização do pedido.

Figura 35: Interface de validação de um pedido.

No entanto, esta interface tem algumas funcionalidades extra em comparação com a anterior. Nesta, quando o utilizador clica no botão para finalizar o pedido, ocorrem algumas validações, de forma sequencial, descritas de seguida:

1. Surge uma janela para que o utilizador confirme a ação, onde também pode deixar uma mensagem de despacho;
2. Se o pedido contiver algum campo marcado a vermelho, surge uma outra janela com a mensagem "Existem um ou mais campos assinalados a vermelho, deseja mesmo continuar com a submissão do pedido?";
3. Os dados são analisados por um validador, onde constam algumas validações sobre cada linha de dados. Este validador analisa se todos os dados de carácter obrigatório se encontram preenchidos; se não existe nenhuma informação repetida na base de dados, como por exemplo a **Sigla**; e verifica também se alguns campos, como o **SIOE** estão de acordo com as suas regras.

A janela de confirmação de finalização pode ser consultada na [Figura 36](#).

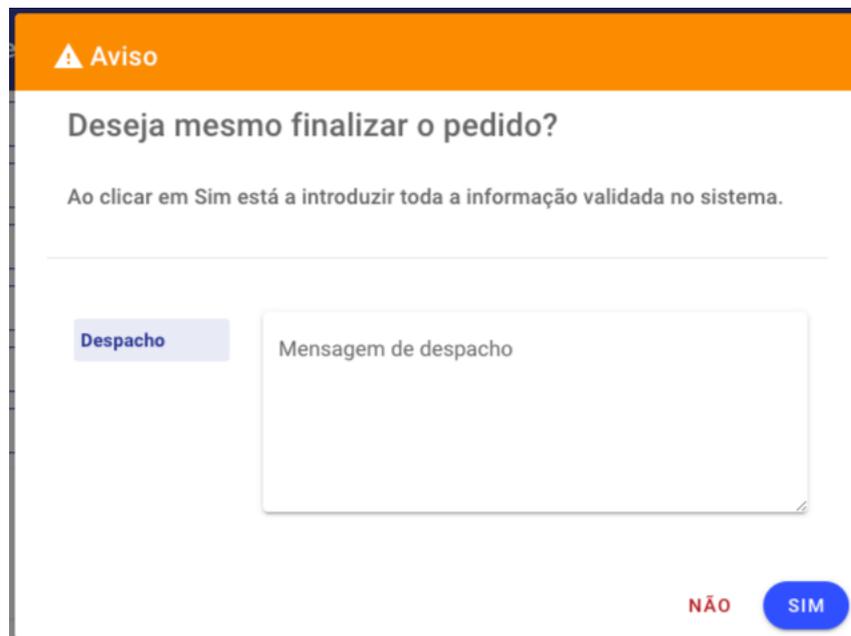


Figura 36: Janela de confirmação de finalização do pedido.

No caso de campos inválidos, é ilustrado na [Figura 37](#) a janela de confirmação.

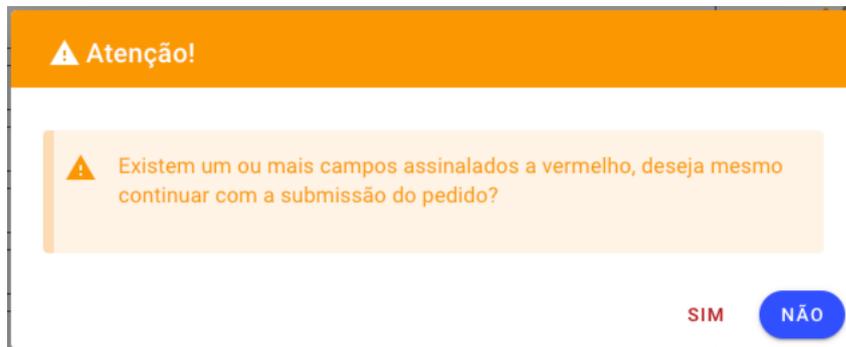


Figura 37: Janela de confirmação de finalização do pedido com campos invalidados.

As regras foram implementadas através de expressões regulares, sendo que o excerto de código seguinte ilustra essa implementação para o campo **SIOE**.

```
Validação do valor do SIOE utilizando expressões regulares.
if (dados.sioe.length > 12) {
  this.erros.push({
    sobre: "SIOE",
    mensagem: "O campo SIOE tem de ter menos
              que 12 dígitos numéricos.",
  });
} else if (!testarRegex(dados.sioe, /^\d+$/)) {
  this.erros.push({
    sobre: "SIOE",
    mensagem: "O campo SIOE só pode ter dígitos
              numéricos.",
  });
}
```

No caso de existirem erros na validação, é apresentada ao utilizador uma janela com os respetivos erros. Desse modo, o utilizador poderá corrigir esses erros e voltar a finalizar, ou então devolver o pedido. A [Figura 38](#) representa essa janela.

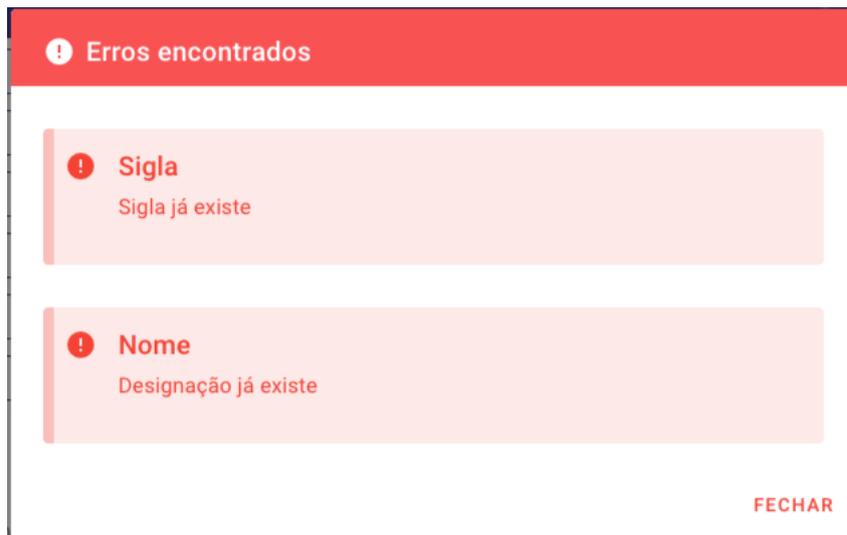


Figura 38: Janela com os erros da validação de um pedido.

Sempre que um pedido seja finalizado com sucesso, é apresentada uma interface que contém algumas informações sobre este, de forma a confirmar essa operação ao utilizador. A Figura 39 representa essa interface.

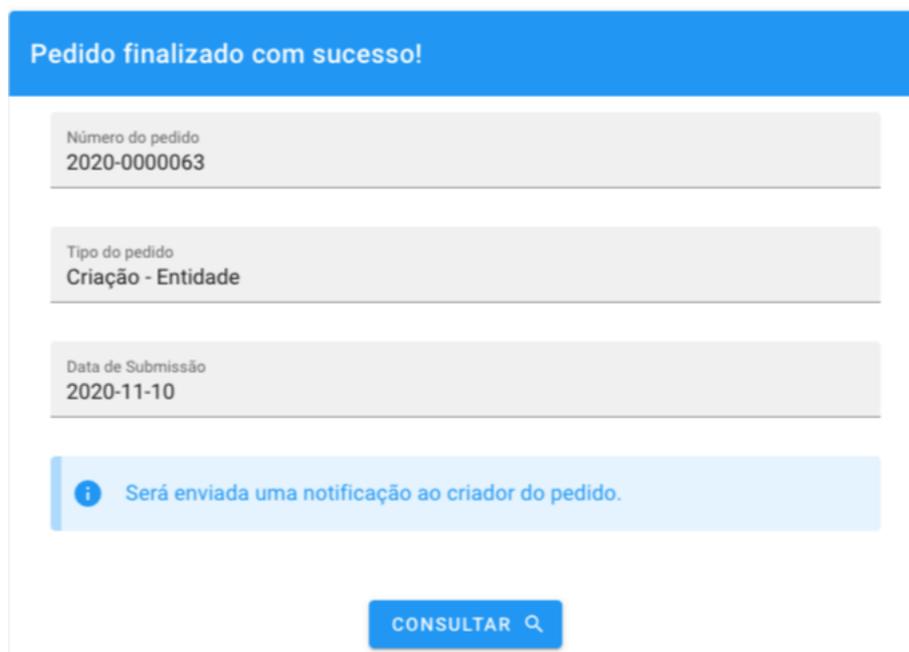


Figura 39: Interface de pedido finalizado com sucesso.

Embora pareçam interfaces simples a níveis visuais, estas são bastante complexas em questões de implementação. Isto porque foi necessário garantir toda a reatividade e dinamismo

das interfaces, uma vez que as interações do utilizador nestas provocam alterações no estado. Estado esse que necessita de ser controlado e renderizado novamente.

No final, assim que um pedido seja aprovado, os dados deste serão introduzidos na plataforma e guardados na base de dados principal do sistema.

A estrutura de dados de um pedido que transita no *workflow* é muito complexa, no entanto, apenas uma pequena parte dos dados da estrutura é guardada na base de dados principal. Esta estrutura é complexa, dado o facto de que suporta todas as funcionalidades e exigências do pedido no processo de transição no *workflow*. Parte desta estrutura já foi apresentada e analisada anteriormente. Apresenta-se agora, no excerto abaixo, a estrutura referente aos dados que são enviados para a base de dados principal.

Estrutura dos dados do pedido a inserir na base de dados principal.

```
"dados": {
  "designacao": "Câmara Municipal dos Arcos de Valdevez",
  "sigla": "CMAV",
  "internacional": "Não",
  "estado": "Ativa",
  "tipologiasSel": [
    {
      "sigla": "AM",
      "designacao": "Assembleias Municipais",
      "id": "tip_AM"
    }
  ]
},
```

A estrutura completa de um pedido pode ser consultada no excerto seguinte.

Estrutura de dados de um pedido completa.³

```

{
  "_id": "5fabbe8c2d246d05472f4452",
  "criadoPor": "pg38425@alunos.uminho.pt",
  "entidade": "ent_II",
  "data": "2020-11-11T10:35:56.546Z",
  "codigo": "2020-0000022",
  "objeto": {
    "tipo": "Entidade",
    "acao": "Criação"
    "dados": { ... },
    "dadosOriginais": {
      "designacao": "Câmara Municipal dos Arcos de Valdevez",
      "sigla": "CMAV",
      "internacional": "Não",
      "sioe": "",
      "dataCriacao": "",
      "dataExtincao": "",
      "estado": "Ativa",
      "tipologiasSel": [
        {
          "sigla": "AM",
          "designacao": "Assembleias Municipais",
          "id": "tip_AM"
        }
      ]
    }
  },
  "historico": [{ ... }],
  "distribuicao": [{ ... }],
}

```

³ As estruturas de dados já apresentadas foram compactadas de forma a manter a estrutura completa numa única página.

4.3.4 Interface de Consulta de Pedidos do Utilizador e Notificações

Uma vez que os utilizadores que criam os pedidos não têm acesso às decisões tomadas nas várias etapas do *workflow*, quer sejam elas alterações ou invalidação de dados, bem como o facto de ser útil para o criador ter a possibilidade de acompanhar o estado dos pedidos submetidos, surgiram duas necessidades evidentes: interface de consulta de pedidos do utilizador; e notificação do pedido.

A interface de consulta de pedidos do utilizador tem como objetivo apresentar ao utilizador uma tabela com todos os pedidos por ele submetidos, com algumas informações úteis, como o **Código** do pedido, o **Tipo** do pedido, **Data** de submissão e o **Estado** do pedido. Para cada pedido selecionado será apresentada uma interface com os dados do próprio pedido.

Ainda nesta interface, será possível consultar uma comparação entre os dados submetidos e os dados finais com o uso da **Linguagem cromática**. Esta comparação tem por base a diferença entre os dados inseridos na criação do pedido e os dados finais, sendo que estes últimos dizem respeito aos dados dos pedidos aprovados ou dos pedidos devolvidos.

Esta operação herdou o termo **Notificação**, uma vez que, para além da interface gráfica, é possível descarregar também um documento em pdf com essa comparação. Também aqui está presente a **Linguagem cromática**, de forma a salientar os campos que foram alterados, validados ou invalidados.

Implementação da Interface de Consulta de Pedidos do Utilizador

Nesta interface foram seguidas as abordagens da interface de gestão dos pedidos do *workflow*, isto significa que os pedidos serão apresentados numa tabela onde são apresentadas algumas informações sobre os mesmos, como se pode verificar na [Figura 40](#).

Código ↑	Tipo	Data	Estado ↑
2020-0000034	Legislação - Revogação	04-11-2020	Submetido
2020-0000033	Tipologia - Alteração	04-11-2020	Apreciado
2020-0000032	Entidade - Alteração	04-11-2020	Distribuído
2020-0000031	Entidade - Criação	04-11-2020	Validado
2020-0000030	Entidade - Criação	04-11-2020	Devolvido

Lista de pedidos submetidos Total de pedidos: 7

Procurar pedido

Pedidos por página 5 1 - 5 de 7

Figura 40: Interface de consulta dos pedidos do utilizador.

No entanto, tal como acontece nos pedidos, também nesta fase a operação será apenas uma, a qual é a consulta dos dados submetidos no pedido. A forma de consultar essa interface será clicando sobre a linha da tabela referente ao pedido pretendido.

Realizando essa operação, o utilizador terá acesso à interface onde serão apresentados todos os dados inseridos no momento de criação do pedido. A [Figura 41](#) mostra essa interface, neste caso, para um pedido que se encontra no estado **Submetido**.

O uso de uma *progress bar* no topo da interface permite ao utilizador, de forma simples e sem a necessidade de ter conhecimento das várias etapas do *workflow*, identificar o quão longe está o pedido de chegar à fase final, sendo ela a inserção dos dados na plataforma ou a devolução.

Informação sobre o pedido: 2020-000030

Submetido

Criação - Entidade Submetido ▶
2020-11-04

Nome
Camara Municipal de Arcos de Valdevez

Sigla
CMAV

Internacional
Não

SIOE
1111

Tipologias

Sigla	Designação
AM	Assembleias Municipais

Tipologias por página 10 1-1 of 1 < >

VOLTAR

Figura 41: Interface de consulta dos dados de um pedido do utilizador.

Mais imagens relativas à interface de consulta dos dados de um pedido podem ser consultadas no [Anexo B](#).

Notificações

De forma a preservar um registo dos pedidos finalizados, surgiu a necessidade de criar uma interface com os detalhes da informação submetida, comparando-a com a informação que chegou ao estado final desse pedido. Esta interface tem uma grande importância, uma vez que durante o processamento de um pedido, no *workflow*, podem ser realizadas alterações aos dados e, desse modo, é importante apresentar essa alteração ao utilizador que criou o pedido. Essa interface pode ser visualizada na [Figura 42](#) e é uma espécie de relatório comparativo entre os dados iniciais e os finais. Estes dados finais podem ser os dados de um pedido **Devolvido** ou de um pedido **Aprovado**, isto porque em ambos os casos é importante a existência de uma comparação dos dados.

Relatório do pedido: 2020-0000030

Criação - Entidade

Data de criação: 2020-11-04

Data de finalização: 2020-11-04

Devolvido !

Campo	Submetido	Finalizado	Nota
Nome	✓ Camara Municipal de Arcos de Valdevez	✎ Câmara Municipal dos Arcos de Valdevez	🗨
Sigla	✓ CMAV	✓ CMAV	🗨
Internacional	✓ Não	✓ Não	🗨
SIOE	✓ 1111	✗ 1111	🗨
Data de Criação	⊘ Campo não preenchido	⊘ Campo não preenchido	🗨
Data de Extinção	⊘ Campo não preenchido	⊘ Campo não preenchido	🗨
Tipologias	✓ • AM	✓ • AM	🗨

Despacho
Corrigir os campos assinalados, nomeadamente o SIEO.

Se possível, preencher também a Data de Criação.

VOLTAR
DESCARREGAR

Figura 42: Relatório com os detalhes entre os dados do pedido submetido e os dados finais.

Como já foi salientado anteriormente, esta interface faz uso da **Linguagem cromática**. Desta forma, é facilmente perceptível que dados foram alterados ou invalidados. Haverá também a possibilidade de consultar, caso exista, um despacho sobre o pedido em questão e consultar a nota relativa a um determinado campo sempre que este campo apresente uma. No exemplo da [Figura 42](#) apenas o campo **SIOE** apresenta uma nota que pode ser consultada clicando sobre o *icon*, onde surgirá um *dialog* com a respetiva nota. Este *dialog* pode ser visualizado na [Figura 43](#).



Figura 43: *Dialog* com nota relativa a um campo do pedido.

Uma vez que muitas instituições têm a necessidade de consultar este relatório em modo *offline*, implementou-se uma funcionalidade que permite a geração de um ficheiro em pdf, de modo a que o criador do pedido tenha a possibilidade de descarregar para o seu computador a denominada **Notificação**.

Um exemplo da **Notificação** em pdf pode ser consultada na [Figura 44](#). Como se pode observar, criou-se este documento com a maior semelhança à interface gráfica da plataforma, onde também se utilizou a **Linguagem cromática**.



NOTIFICAÇÃO

Criação - Entidade

Devolvido

Data de criação: 2020-11-04
Data de finalização: 2020-11-04

Campo	Submetido	Finalizado
Nome	Camara Municipal de Arcos de Valdevez	Câmara Municipal dos Arcos de Valdevez
Sigla	CMAV	CMAV
Internacional	Não	Não
SIOE	1111	1111
Nota	Parece-me errado o SIOE.	
Data de Criação		
Data de Extinção		
Tipologias	• AM	• AM

Despacho final: Corrigir os campos assinalados, nomeadamente o SIEO.

Se possível, preencher também a Data de Criação.

Figura 44: Exemplo de notificação em pdf de pedidos finalizados.

Package Utilizado e Implementação da Notificação

Para a implementação deste documento, utilizou-se o *package* **pdfmake**⁴ sendo este um *package* bastante popular e serviu na perfeição para as necessidades exigidas.

Antes deste, foi analisado um outro, denominado por **jsPDF**⁵ que, embora seja bastante poderoso, o grau de complexidade é maior uma vez que todos os comportamentos tem que ser codificados. Um exemplo disso era o facto das quebras de página não serem automaticamente calculadas, havendo a necessidade de codificar esse comportamento.

Já o **pdfmake** permite não só calcular, de forma automática, as quebras de página sempre que assim o justificasse, como definia as margens, com base numa configuração previamente definida, do documento de forma a que o texto não transpusesse esses limites.

Este *package* tem por base uma estrutura em JSON para a criação do pdf, desse modo facilita a sua utilização em tecnologias como o JavaScript. Esta estrutura tem, como principais elementos, a *key content* que diz respeito a todo o conteúdo gerado no pdf, e a *key styles* responsável pelos estilos do respetivo conteúdo, como cores e tamanho de letra. Um exemplo dessa estrutura pode ser consultado no excerto de código seguinte:

Exemplo da geração de uma Notificação.

```
let pdf = { pageMargins: [40, 160, 40, 80] };
let content = [];
let styles = {
  defaultStyle: {
    fontSize: 15,
    bold: false,
  },
};

content.push({ text: "Notificação" });

const guardarPDF = (numeroPedido) => {
  pdf.content = content;
  pdf.styles = styles;
  pdfMake.createPdf(pdf).download(`notificacao-${numeroPedido}`);
};
```

⁴ A documentação pode ser consultada em <https://pdfmake.github.io/docs/0.1/>.

⁵ A documentação pode ser consultada em <https://rawgit.com/MrRio/jsPDF/master/docs/index.html>.

Como se pode observar, o nome da notificação é gerado através da concatenação entre a palavra "notificação" e o número do pedido em questão, separado por um hífen. Dessa forma, é possível identificar a que pedido pertence sem a necessidade de abrir o ficheiro.

Além da estrutura existente no *package pdfmake*, criou-se uma estrutura interna que guarda algumas informações importantes para a geração da **Notificação** com os dados pretendidos. Esta estrutura tem várias *keys*, no entanto, uma das mais importantes é a *key* "comparacao" que se representa por um *array* de objetos em que guarda, para cada campo, o nome deste, os dados do pedido submetido e os dados do pedido finalizado. Esta estrutura é apresentada no excerto de código seguinte:

Estrutura de dados para a criação da **Notificação**.

```
{
  "alteracaoInfo": "",
  "comparacao": [
    {
      "campo": "Nome",
      "finalizado": {
        "cor": "amarelo",
        "dados": "Câmara Municipal dos Arcos de Valdevez",
        "nota": null
      },
      "submetido": {
        "cor": "verde",
        "dados": "Camara Municipal de Arcos de Valdevez",
        "nota": null
      }
    },
    { ... }
  ],
  "dataFinal": "2020-11-04",
  "dataInicial": "2020-11-04",
  "despacho": "Corrigir os campos assinalados, nomeadamente o SIE0.
  \n\nSe possível, preencher também a Data de Criação.",
  "estadoPedido": "Devolvido",
  "numeroPedido": "2020-0000030",
  "tipoPedido": "Criação - Entidade"
}
```

Realizada a modelação do *workflow*, com a devida atenção aos requisitos, e analisadas as tecnologias utilizadas no sistema, foram implementadas as interfaces que permitem a utilização desse modelo. Deste modo, todas as interfaces implementadas permitem seguir e interagir com os pedidos, desde a entrada na primeira fase do *workflow* até à última.

CONCLUSÃO

Este último capítulo apresenta uma reflexão global face ao projeto e às aprendizagens desenvolvidas.

Num primeiro momento, serão apresentadas as conclusões do projeto, seguindo-se algumas recomendações para futuros desenvolvimentos.

Por fim, será apresentada uma reflexão acerca das dificuldades sentidas, bem como das competências desenvolvidas, a nível pessoal e profissional, ao longo do desenvolvimento do projeto.

5.1 CONCLUSÕES ACERCA DO PROJETO

Tendo em conta os problemas existentes, face à inexistência de um *workflow* capaz de controlar e auxiliar no processo dos pedidos da plataforma CLAV, aliado ao facto da existência de erros nos dados submetidos face a falhas humanas, houve uma grande necessidade de desenvolver um *workflow* para a garantir a eficácia da gestão dos pedidos. Sem a implementação de um sistema destes, os processos seriam mais demorados, dado o facto de que os vários responsáveis pelas várias etapas de um pedido se encontrarem em departamentos diferentes dentro da DGLAB.

Em virtude de criar um sistema o mais completo e robusto possível, foi necessário iniciar o desenvolvimento através de um modelo gráfico, detalhando, para cada uma das diferentes etapas, as diversas ações e funcionalidades, bem como o controlo dos utilizadores que têm acesso às mesmas. Após este modelo ter sido aprovado, seguiu-se com a implementação deste na plataforma CLAV.

Posto isto, todas as etapas do *workflow* foram implementadas, bem como todas as funcionalidades a ele inerentes, tais como o controlo das permissões de acesso, as interfaces de consulta do **Histórico de alterações** e ainda a **Linguagem cromática**.

Além disso, também a interface de consulta dos pedidos submetidos, bem como a geração da notificação do pedido em pdf foram implementadas. Desse modo, todos os objetivos foram cumpridos e toda a implementação está funcional e a ser utilizada pela DGLAB.

De seguida, é apresentada a [Tabela 12](#) onde se pode verificar a calendarização prevista para a realização deste projeto.

Mês	Tarefa
Outubro de 2019	Estudo teórico do contexto do projeto CLAV.
Novembro de 2019	Estudo acerca da abordagem semelhante ao CLAV desenvolvida em Andaluzia (Espanha).
Dezembro de 2019	Estudo teórico sobre BPMN.
Janeiro de 2020	Modelação do <i>workflow</i> usando as especificações do BPMN.
Fevereiro de 2020	Modelação do <i>workflow</i> usando as especificações do BPMN; Análise das tecnologias utilizadas no desenvolvimento da plataforma CLAV e familiarização com as mesmas.
Março de 2020	Estrutura de dados de suporte aos pedidos. Implementação das interfaces das várias etapas do <i>workflow</i> .
Abril de 2020	Estrutura de dados de suporte aos pedidos; Implementação das interfaces das várias etapas do <i>workflow</i> .
Maió de 2020	Implementação das interfaces das várias etapas do <i>workflow</i> .
Junho de 2020	Estrutura de dados de suporte ao histórico de alterações e da linguagem cromática; Implementação das interfaces das várias etapas do <i>workflow</i> .
Julho de 2020	Estrutura de dados de suporte ao histórico de alterações e da linguagem cromática; Implementação das interfaces do histórico de alterações.
Agosto de 2020	Implementação das interfaces do histórico de alterações; Implementação da Interface de Consulta de Pedidos do Utilizador.
Setembro de 2020	Implementação da Interface de Consulta de Pedidos do Utilizador; Ajustes na implementação.
Outubro de 2020	Geração de notificações em pdf; Ajustes na implementação.

Tabela 12: Calendarização prevista.

De salientar que toda a calendarização foi cumprida conforme o previsto, à exceção do mês de maio de 2020, que devido a problemas com a estrutura de suporte aos pedidos, foi necessário realizar uma reformulação nesta, prolongando-se também pelo mês de junho de 2020.

5.2 TRABALHO FUTURO

Ao longo do desenvolvimento do projeto foram surgindo algumas necessidades extra, as quais não puderam ser implementadas, face ao tempo que seria necessário dedicar a estas. Desse modo, serão apresentadas de seguida algumas dessas necessidades, bem como algumas sugestões de como melhorar a plataforma, de forma a que estas possam ser implementadas em trabalhos futuros.

A primeira sugestão relaciona-se com a necessidade de reformular a estrutura de dados relativa à **Linguagem cromática**, de forma a que esta fosse identificada pela sua ação e não pela cor. Ou seja, substituir a propriedade "cor" por "acao", onde o valor passaria de "vermelho" para "invalidado", podendo também ser utilizada apenas uma letra. A [Tabela 13](#) exemplifica esta reformulação.

Atual	Sugestão
"verde"	"validado" ou "V"
"vermelho"	"invalidado" ou "I"
"amarelo"	"alterado" ou "A"

Tabela 13: Sugestão de reformulação da **Linguagem cromática**.

Outra sugestão relevante seria a criação de uma interface de estatísticas do *workflow*. Isto é, uma interface que mostrasse aos administradores da plataforma algumas métricas, nomeadamente:

- Percentagem de pedidos aprovados e devolvidos;
- Tempo médio que um pedido permanece no *workflow*;

Através destas, muitas outras métricas podem ser aplicadas utilizando os dados dos pedidos.

Uma funcionalidade importante para toda a plataforma, não apenas para o *workflow*, seria a implementação de um sistema de notificações que permitisse, em tempo real, mostrar a notificação aos utilizadores. Estas poderiam ser notificações relativas aos pedidos submetidos que tenham sido aprovados ou devolvidos.

Uma vez que a plataforma já se encontra em produção, a adoção de um *changelog* acompanhado de um sistema de versão numérico era importante, uma vez que permitia, de forma rápida e simples, informar os utilizadores acerca das novas funcionalidades implementadas ou quais os *bugs* corrigidos.

5.3 REFLEXÃO ACERCA DO VALOR DO PROJETO NO DESENVOLVIMENTO PESSOAL E PROFISSIONAL

Fazendo uma retrospectiva de toda a preparação e implementação do projeto, torna-se necessário identificar quais as dificuldades sentidas e as aprendizagens desenvolvidas ao longo deste percurso.

Começando pelas dificuldades, a maior delas foi o contacto com uma temática totalmente nova para mim, nomeadamente a Web semântica. Esta tem associado um conjunto de padrões que até então desconhecia. Aliado também ao facto de a base de dados ser ontológica, outro conceito totalmente novo para mim, sendo então necessário um maior empenho e realizar uma vasta pesquisa acerca do tema.

A segunda maior dificuldade, mas que desempenhou um papel fundamental para o meu crescimento profissional, foi o facto de trabalhar numa equipa grande, em comparação com os trabalhos académicos. A equipa era composta não só por *developers*, mas também por órgãos da DGLAB, tais como arquivistas e administradores.

Esta situação contribuiu para um desenvolvimento tanto pessoal como profissional, na medida em que este projeto se assemelhou a um contexto empresarial, possibilitando uma melhor preparação para o mercado de trabalho.

Além disso, possibilitou o desenvolvimento das minhas *soft skills*, nomeadamente a colaboração, a flexibilidade, a comunicação, a criatividade, e ainda organização do tempo.

REFERÊNCIAS

- Beckett, D., Berners-Lee, T., Prud'hommeaux, E., & Carothers, G. (2014). Rdf 1.1 turtle. W3C. Retrieved from <https://www.w3.org/TR/2014/REC-turtle-20140225/>
- Bizer, C., & Cyganiak, R. (2014). Rdf 1.1 trig. W3C. Retrieved from <https://www.w3.org/TR/2014/REC-trig-20140225/>
- Boley, H., & Kifer, M. (2013). *Rif basic logic dialect (second edition)*. Retrieved from <https://www.w3.org/TR/2013/REC-rif-bld-20130205/>
- Brickley, D., & Guha, R. (2014). Rdf schema 1.1. W3C. Retrieved from <https://www.w3.org/TR/2014/REC-rdf-schema-20140225/>
- Cardoso, J., & Pinto, A. M. (s.d.). The web ontology language (owl) and its applications. Retrieved from <https://eden.dei.uc.pt/~jcardoso/Research/Papers/BC-2014-031-ISR-OWL-and-Its-Applications.pdf>
- do Conselho de Ministros, P. (2017). *Resolução do conselho de ministros n.º 51/2017*. (<https://dre.pt/home/-/dre/106896737/details/maximized>)
- García, M. A. P. (2002). *El cuadro de clasificación integrado: normalización de la clasificación archivística*. Retrieved from <https://www.juntadeandalucia.es/organismos/agriculturaganaderiapescaydesarrollosostenible/servicios/publicaciones/detalle/43566.html>
- García, M. A. P. (2004). Criterios: El cuadro de clasificación integrado: normalización de la clasificación archivística. Retrieved from <https://doi.org/10.33349/2004.47.1691>
- Golbreich, C., & Wallace, E. K. (2012). Owl 2 web ontology language new features and rationale (second edition). W3C. Retrieved from <https://www.w3.org/TR/2012/REC-owl2-new-features-20121211/>
- Group, O. M. (2013). Business process model and notation. *omg*. (<https://www.omg.org/spec/BPMN/2.0.2/PDF>)
- Group, T. W. S. W. (2013). Sparql 1.1 overview. W3C. Retrieved from <https://www.w3.org/TR/2013/REC-sparql11-overview-20130321/>
- Horrocks, I., Patel-Schneider, P. F., Boley, H., Tabet, S., Grosz, B., & Dean, M. (2004). *Swrl: A semantic web rule language combining owl and ruleml*. Retrieved from <https://www.w3.org/Submission/2004/SUBM-SWRL-20040521/>
- Kifer, M. (2008). Rule interchange format: The framework. In *Web reasoning and rule systems*.
- Librelotto, G. R., Ramalho, J. C., & Henriques, P. R. (2005). Representação de conhecimento na semantic web. *Researchgate*. Retrieved from https://www.researchgate.net/publication/277179661_Representacao_de_conhecimento_na_semantic_web
- Lourenço, A., Ramalho, J. C., Gago, M. R., & Penteado, P. (2018). Transformação digital: novas políticas e procedimentos para a classificação e a avaliação da informação. In *Transformação digital: novas políticas e procedimentos para a classificação e a avaliação da infor-*

- mação. Retrieved from <https://www.bad.pt/publicacoes/index.php/congressosbad/article/view/1861>
- Martínez Force, E., Melero Casado, A., Molina González, A., & Valiente Fabero, M. I. (2010). Sistema de información @rchiva de la junta de andalucía para la gestión de documentos y archivos. *Ibersid: revista de sistemas de información y documentación*. (<https://www.ibersid.eu/ojs/index.php/ibersid/article/view/3811>)
- MoReq2010. (s.d.). *What is moreq's service based architecture?* Retrieved from <https://moreq.info/faq/23-what-is-moreq-s-service-based-architecture>
- Mundet, J. R. C. (1994). *Manual de archivística*.
- Obitko, M. (2007). *Ontologies and semantic web*. Retrieved from <https://www.obitko.com/tutorials/ontologies-semantic-web/>
- Ontotext. (s.d.). *Graphdb documentation*. Retrieved from <http://graphdb.ontotext.com/documentation/standard/>
- s.a. (s.d.). *Lista consolidada*. Retrieved from <http://arquivos.dglab.gov.pt/programas-e-projectos/modernizacao-administrativa/macroestrutura-funcional-mef/lista-consolidada/>
- Sporny, M., Longley, D., Kellogg, G., Lanthaler, M., Champin, P.-A., & Lindström, N. (2019). *Json-ld 1.1. W3C*. Retrieved from <https://www.w3.org/TR/2019/CR-json-ld11-20191212/>
- Vieira, R., & Borbinha, J. (2011). *Moreq2010 – uma apresentação*. In *Moreq2010 – uma apresentação*. (https://www.researchgate.net/publication/235922146_MoReq2010_-_Uma_Apresentacao)
- W3C. (s.d.). *Semantic web*. Retrieved from <https://www.w3.org/standards/semanticweb/>
- Wooldridge, M. (s.d.). *Making new connections with marklogic semantics*. Retrieved from <https://www.marklogic.com/blog/making-new-connections-ml-semantics/>

IMAGENS COMPLEMENTARES DO MODELO BPMN DO WORKFLOW

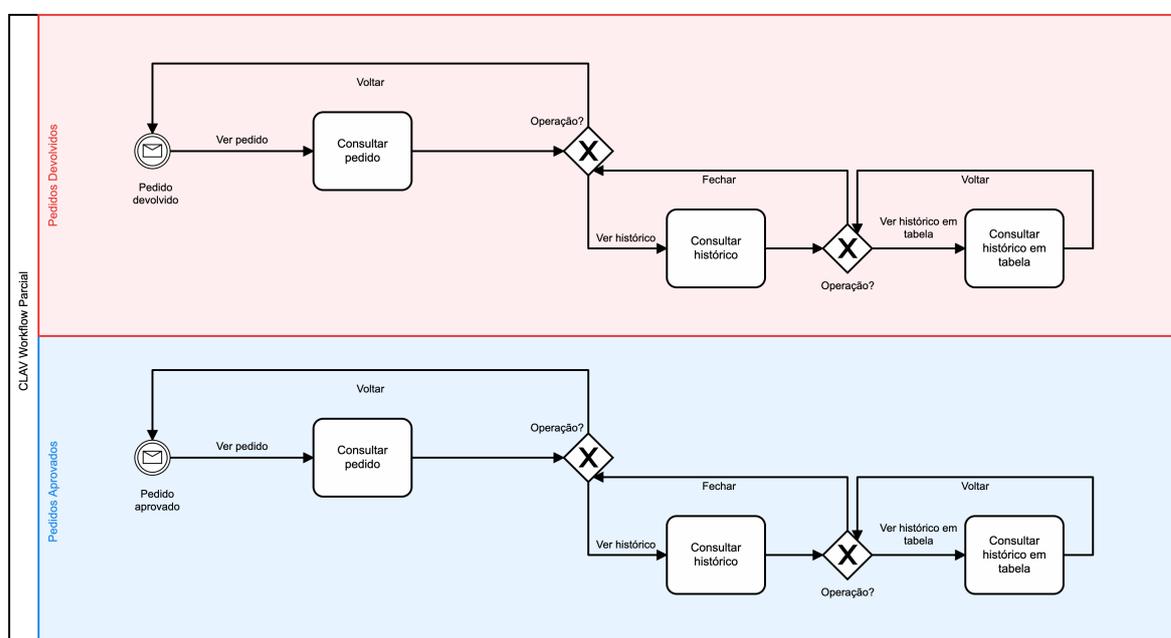


Figura 45: Modelo BPMN parcial das últimas fases do *workflow*.

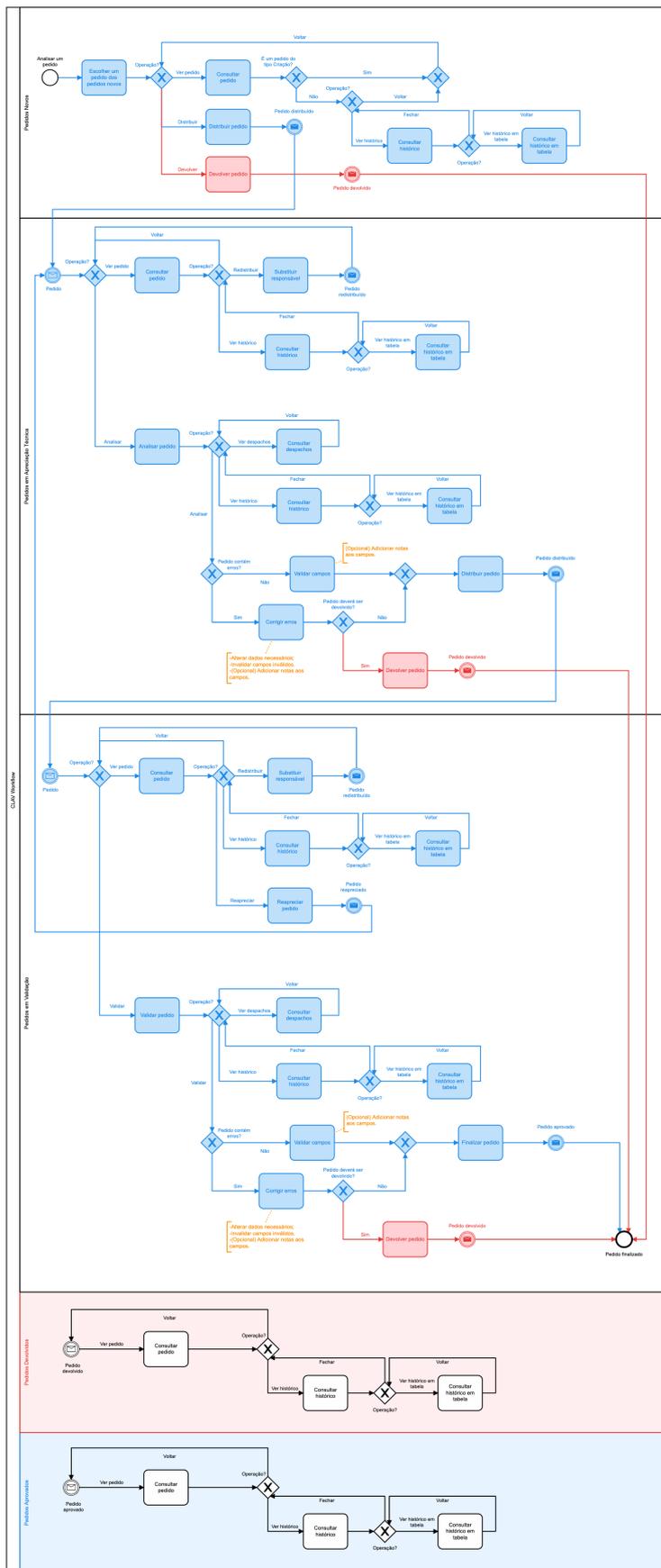


Figura 47: Modelo BPMN do workflow completo.

B

INTERFACES DE CONSULTA DE PEDIDOS DO UTILIZADOR E NOTIFICAÇÕES

B.1 INTERFACES DE CONSULTA DOS DADOS DE PEDIDOS DO UTILIZADOR

Informação sobre o pedido: 2020-000030

Distribuído

Criação - Entidade Distribuído
2020-11-04

Nome
Camara Municipal de Arcos de Valdevez

Sigla
CMAV

Internacional
Não

SIOE
1111

Tipologias

Sigla	Designação
AM	Assembleias Municipais

Tipologias por página 10 1-1 of 1 < >

VOLTAR

Figura 48: Interface de consulta dos dados de um pedido do utilizador no estado **Distribuído**.

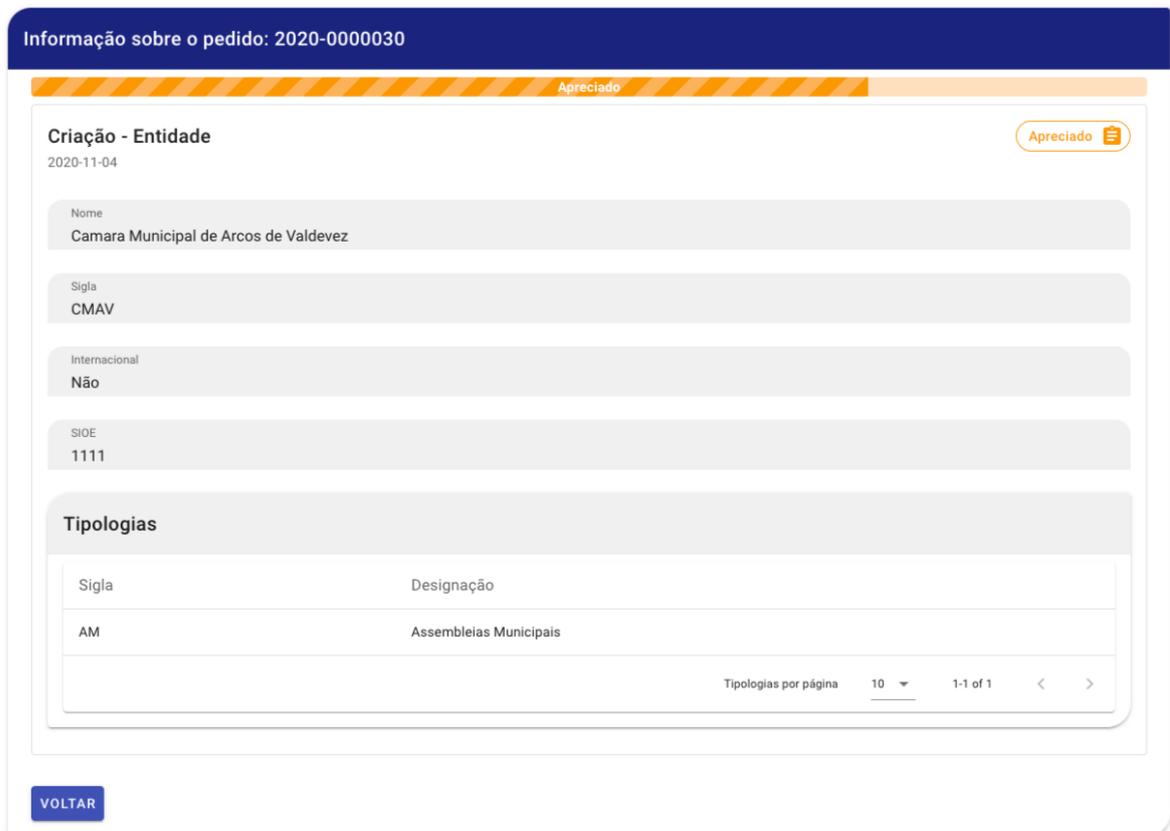


Figura 49: Interface de consulta dos dados de um pedido do utilizador no estado **Apreciado**.

Informação sobre o pedido: 2020-000030

Devolvido

Criação - Entidade

2020-11-04

Devolvido ⓘ

Nome
Câmara Municipal dos Arcos de Valdevez

Sigla
CMAV

Internacional
Não

SIOE
1111

Estado
Ativa

Tipologias

Sigla	Designação
AM	Assembleias Municipais

Tipologias por página 10 1-1 of 1 < >

VOLTAR VER RELATÓRIO

Figura 50: Interface de consulta dos dados de um pedido do utilizador no estado **Devolvido**.

Informação sobre o pedido: 2020-0000031

Validado

Criação - Entidade Validado

2020-11-04

Nome
Câmara Municipal dos Arcos de Valdevez

Sigla
CMAV

Internacional
Não

Estado
Ativa

Tipologias

Sigla	Designação
AM	Assembleias Municipais

Tipologias por página 10 1-1 of 1 < >

VOLTAR VER RELATÓRIO

Figura 51: Interface de consulta dos dados de um pedido do utilizador no estado **Validado**.