

Escola Superior de Tecnologia de Tomar

Logística: Automação de Encomendas e Otimização com Inteligência Artificial

Relatório de Estágio

Pedro Miguel Monteiro Gaspar

Engenharia Informática – Internet das Coisas

Tomar. Novembro, 2025



Esta página foi intencionalmente deixada em branco.

Escola Superior de Tecnologia de Tomar

Logística: Automação de Encomendas e Otimização com Inteligência Artificial

Relatório de Estágio

Pedro Miguel Monteiro Gaspar

Orientado por:

Prof. Renato Panda, Instituto Politécnico de Tomar
Dir. de Operações Rui Paiva, petMaxi

Relatório de Estágio apresentado ao Instituto Politécnico de Tomar para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Informática – Internet das Coisas

Esta página foi intencionalmente deixada em branco.

Informação sobre o Autor, Orientador e Financiamento

Autor: Pedro Gaspar
pedrogaspar2002@gmail.com

Orientadores: Prof. Renato Panda
renato.panda@ipt.pt
Dir. de Operações Rui Paiva
Rui.Paiva@petmaxi.pt

**Instituição de acolhi-
mento:** petMaxi

**Instituição que concede
o grau:** Instituto Politécnico de Tomar

Esta página foi intencionalmente deixada em branco.

Agradecimentos

A realização deste estágio e a redação do presente relatório só foram possíveis graças ao apoio e colaboração de diversas pessoas e instituições, a quem deixo o meu mais sincero agradecimento.

Expresso a minha profunda gratidão à empresa petMaxi, pela forma acolhedora e disponível com que me recebeu. Um agradecimento especial ao meu orientador, Diretor de Operações Rui Paiva, pelo acompanhamento constante, pela partilha de conhecimentos e pela dedicação demonstrada ao longo de todo o estágio. A sua paciência e disponibilidade foram essenciais para o meu crescimento, tanto a nível profissional como pessoal.

Agradeço igualmente ao Professor Renato Panda, pelo apoio e orientação prestados na elaboração deste relatório.

Deixo também um agradecimento à Escola Superior de Tecnologia de Tomar, por me ter proporcionado uma experiência valiosa ao longo dos cinco anos em que me acolheu.

Por fim, um agradecimento muito especial à minha mãe e à minha namorada, pelo apoio constante e pela força que sempre me transmitiram, especialmente durante esta fase decisiva da minha jornada académica. Sem elas, nada disto teria sido possível.

Esta página foi intencionalmente deixada em branco.

Resumo

Este relatório documenta o estágio curricular de 39 semanas realizado na petMaxi, empresa portuguesa de produção de alimentos para animais de companhia. O estágio adotou uma abordagem multi-projeto focada em três vertentes principais: 1) continuação do desenvolvimento da plataforma web NETPET (*Django/Bootstrap*) para gestão interna de múltiplos departamentos da empresa; 2) automação de processos logísticos através da implementação de um sistema de extração automática de encomendas provenientes do Outlook usando Power Automate com modelos GPT 3.5/4 e técnicas de *fuzzy matching* para processamento de dados; e 3) melhorias na infraestrutura DevOps com implementação de *Docker*, *Traefik* para gestão de certificados SSL, *Redis* e *Celery* para tarefas assíncronas.

O NETPET foi expandido com funcionalidades transversais como *WebSockets* para atualização em tempo real, sistema de notificações, e funcionalidades específicas para os departamentos de Logística, Qualidade, Financeiro, Comercial e Área Fabril. A automação com Inteligência artificial (IA) permitiu extrair dados de encomendas recebidas por email, eliminando quase por completo os processos manuais de integração das mesmas no ERP Primavera. A infraestrutura foi complementada com ferramentas como o GLPI para gestão de equipamentos, *Netbox* para inventariação de redes, e desenvolvimento de uma API *REST* com *Flask* e *Swagger*.

Palavras-chave: Automação de Processos, NETPET, Inteligência Artificial, Desenvolvimento Web, Infraestrutura DevOps.

Esta página foi intencionalmente deixada em branco.

Abstract

This report documents a 39-week curricular internship at petMaxi, a Portuguese pet food manufacturing company. The internship adopted a multi-project approach focused on three main areas: continuation of the NETPET web platform development (Django/Bootstrap) for internal management across multiple company departments; automation of logistics processes through the implementation of an automatic order extraction system from Outlook using Power Automate with GPT 3.5/4 models and fuzzy matching techniques for data processing; and improvements to the DevOps infrastructure with implementation of Docker, Traefik for SSL certificate management, Redis and Celery for asynchronous tasks.

NETPET was expanded with cross-functional features such as WebSockets for real-time updates, a notification system, and specific functionalities for the Logistics, Quality, Financial, Commercial, and Manufacturing departments. AI-powered automation enabled the extraction of order data from emails, almost completely eliminating manual integration processes from these orders into the Primavera ERP system. The infrastructure was complemented with tools such as GLPI for equipment management, Netbox for network inventory, and development of a REST API with Flask and Swagger.

Keywords: Process Automation, NETPET, Artificial Intelligence (AI), Web Development, DevOps Infrastructure.

Índice

Agradecimentos	v
Resumo	vii
Abstract	ix
Índice	x
Lista de Figuras	xiii
Lista de Tabelas	xv
Lista de Abreviaturas	xvii
1 Introdução	1
1.1 Contexto do Estágio	1
1.1.1 Enquadramento Empresarial	1
1.1.2 Desafios Logísticos Atuais	2
1.1.3 Motivação	2
1.1.4 Integração	3
1.1.5 Duração e Cronologia	3
1.2 Objetivos e Âmbito	5
1.2.1 Evolução dos Requisitos	5
1.2.2 Âmbito de Trabalho	5
1.3 Estrutura do Relatório	6
2 Enquadramento e Metodologia	9
2.1 Análise de Necessidades	9
2.2 Metodologia de Desenvolvimento	10
2.2.1 Linguagem de Programação e <i>Frameworks</i>	10
2.2.2 Disponibilização para Produção	11
2.2.3 Modelos de IA	12
3 Desenvolvimento e Implementação	15
3.1 Plataforma NETPET	15
3.1.1 Infraestrutura e Funcionalidades Transversais	15
3.1.2 Departamento de Logística	21
3.1.3 Departamento da Qualidade	26
3.1.4 Departamento Financeiro	30

3.1.5	Departamento Comercial	30
3.1.6	Área fabril	33
3.2	Automação com IA	34
3.3	Infraestrutura DevOps	39
4	Análise de Impacto e Reflexões	45
4.1	Avaliação do Desempenho do Estagiário	45
4.2	Avaliação do NETPET	47
4.3	Impacto na Duração dos Processos	52
5	Conclusão	55
5.1	Síntese do Trabalho Desenvolvido	55
5.2	Trabalho Futuro	56
	Referências	59

Esta página foi intencionalmente deixada em branco.

Lista de Figuras

1.1	Atividades Realizadas - Setembro a Fevereiro	4
1.2	Atividades Realizadas - Março a Junho	4
2.1	Arquitetura do <i>Docker</i> (Docker Inc., 2025)	12
3.1	Estrutura da diretoria onde se localizam os ficheiros de <i>WebSockets</i>	16
3.2	Página das Notificações	17
3.3	Filtro estilo Excel	18
3.4	Imagem de perfil na dropdown de opções	19
3.5	Uptime Kuma	19
3.6	Encomenda destacada	20
3.7	Modal dos Custos ECL	20
3.8	Declaração 2018/1910	22
3.9	Página da Documentação	23
3.10	Recolhas de Paletes	23
3.11	Mapa das recolhas de paletes	24
3.12	Página de importação de documentos	24
3.13	Listagem do Stock das Grandes Superfícies	25
3.14	Dashboard dos dados da balança	26
3.15	Lista de Fornecedores	27
3.16	Página de registo e acompanhamento dos resíduos	27
3.17	Página da listagem das informações dos resíduos	28
3.18	Página de listagem das análises	28
3.19	Diagrama de matérias primas	29
3.20	Parâmetros de cada matéria prima	29
3.21	Página das Vendas COSEC	30
3.22	Página de visualização do Stock de Sílica	31
3.23	Página de visualização dos dados das marcas próprias	31
3.24	Gráfico Gantt de análise da durabilidade das ECFs	32
3.25	Mapa de Clientes	33
3.26	Planeamento – Ordens de Produção	33
3.27	Planeamento - Ordens de Ensaque	34
3.28	<i>To Do</i> Ordens de Produção	34
3.29	Fluxo - Parte Inicial	35
3.30	Fluxo - Parte Intermédia	36
3.31	Fluxo - Parte Final	37
3.32	Dashboard das encomendas	38
3.33	Listagem de encomendas	38
3.34	Página de confirmação da encomenda	39

3.35	Interface do GLPI	40
3.36	Interface do Netbox	41
3.37	Página de manutenção	42
3.38	API da petMaxi	43
4.1	Clareza de Comunicação	46
4.2	Iniciativa e Proatividade	46
4.3	Contribuição positiva para a equipa	46
4.4	Abertura a feedback	47
4.5	Capacidade de resolução de problemas	47
4.6	Frequência de utilização de dispositivos tecnológicos	48
4.7	Nível de competência digital	48
4.8	Frequência de utilização da plataforma	48
4.9	Perceção de complexidade da plataforma	49
4.10	Avaliação da facilidade de uso da plataforma	49
4.11	Necessidade de apoio técnico para utilização da plataforma	49
4.12	Integração das funcionalidades da plataforma	50
4.13	Perceção de inconsistência na plataforma	50
4.14	Facilidade de aprendizagem da plataforma	50
4.15	Perceção de lentidão da plataforma	51
4.16	Confiança na utilização da plataforma	51
4.17	Necessidade de conhecimentos prévios para utilização adequada	51

Lista de Tabelas

2.1	Vantagens e desvantagens dos Modelos de IA	14
4.1	Comparação entre o método antigo e o método NETPET por destino.	52
4.2	Análise do Impacto Semanal por destino	53

Esta página foi intencionalmente deixada em branco.

Lista de Abreviaturas

Acrónimo	Designação	Páginas
API	<i>Application Programming Interface</i>	pag. vii, pag. 9, pag. 12, pag. 18, pag. 42, pag. 56
ASGI	<i>Asynchronous Server Gateway Interface</i>	pag. 15
BERT	<i>Bidirectional Encoder Representations from Transformers</i>	pag. 5, pag. 12–14, pag. 35
CN	<i>Common Name</i>	pag. 41
CSS	<i>Cascading Style Sheets</i>	pag. 10, 11, pag. 17, pag. 21, 22
CSV	<i>Comma-separated values</i>	pag. 30
DNS	<i>Domain Name System</i>	pag. 9, pag. 41
ECF	Encomenda de Fornecedor	pag. 31
ECL	Encomenda de Cliente	pag. 20
ERP	<i>Enterprise Resource Planning</i>	pag. vii, pag. 2, pag. 9, pag. 22, pag. 24, pag. 30, pag. 39, pag. 56

GLPI	<i>Gestionnaire Libre de Parc Informatique</i>	pag. vii, pag. 9, pag. 39–41
GPT	<i>Generative Pre-training Transformer</i>	pag. vii, pag. 5, 6, pag. 12–14, pag. 35–37, pag. 55, 56
HTML	<i>Hypertext Markup Language</i>	pag. 17, pag. 21–23, pag. 35
HTTP	<i>Hypertext Transfer Protocol</i>	pag. 9, pag. 15, 16
HTTPS	<i>Hypertext Transfer Protocol Secure</i>	pag. 41
IA	Inteligência artificial	pag. vii, pag. xv, pag. 2, pag. 6, pag. 9, 10, pag. 12–14, pag. 55, 56
ID	<i>Identifier</i>	pag. 16
IRS	Imposto sobre o Rendimento das Pessoas Singulares	pag. 21
JSON	<i>JavaScript Object Notation</i>	pag. 35, 36
LLM	<i>Large Language Model</i>	pag. 13
MVC	<i>Model-View-Controller</i>	pag. 10
MVT	<i>Model-View-Template</i>	pag. 10
NER	<i>Named Entity Recognition</i>	pag. 13
NLP	<i>Natural Language Processing</i>	pag. 14
PDF	Portable Document Format	pag. 6, pag. 21–24, pag. 26, pag. 32
PHP	<i>Hypertext Preprocessor</i>	pag. 39
PM	petMaxi	pag. vii, pag. 1–3, pag. 5, pag. 9, pag. 19, pag. 26, pag. 37, pag. 55

SaaS	<i>Software as a Service</i>	pag. 12
SAML	<i>Security Assertion Markup Language</i>	pag. 40
SAN	<i>Subject Alternative Name</i>	pag. 41
SQL	<i>Structured Query Language</i>	pag. 42
SSL	<i>Secure Sockets Layer</i>	pag. vii, pag. 6, pag. 9, pag. 41, pag. 56
TLS	<i>Transport Layer Security</i>	pag. 41
URL	<i>Uniform Resource Locator</i>	pag. 42
VGM	<i>Verified Gross Mass</i>	pag. 9, pag. 23
VM	<i>Virtual Machine</i>	pag. 11, pag. 40

Esta página foi intencionalmente deixada em branco.

Capítulo 1

Introdução

Este capítulo enquadra o estágio curricular desenvolvido na petMaxi (PM), delineando os objetivos iniciais propostos e detalhando a evolução do trabalho ao longo do tempo.

1.1 Contexto do Estágio

1.1.1 Enquadramento Empresarial

A PM é uma empresa portuguesa, localizada em Ferreira do Zêzere, no distrito de Santarém. Inaugurada a 13 de fevereiro de 2015, com o objetivo de concorrer com as principais marcas internacionais no mercado, dedica-se à produção de alimentos de alta qualidade para animais de companhia, mais especificamente cães e gatos.

A empresa destaca-se pela utilização de ingredientes frescos, como os ovos e a carne, mantendo os nutrientes e o sabor do produto. Tem como principal objetivo melhorar a qualidade de vida e a longevidade dos animais domésticos através da produção de alimentos distintos. A PM exporta atualmente para mais de 40 países de diferentes continentes, sendo as exportações uma parte significativa da sua produção.

Esta é também certificada pela IFS Food¹, um padrão internacional que garante a qualidade e segurança dos produtos, desde 2017, obtendo sempre classificações de *Higher Level*, o nível mais elevado desta certificação. Além disto a empresa tem também um compromisso com a sustentabilidade ambiental, social e económica, de forma a reduzir a pegada de carbono, promovendo práticas ecológicas.

A empresa procura constantemente inovar-se, diferenciando-se de outras empresas do mercado, como é possível verificar observando uma das suas marcas em destaque, happyOne Mediterranean, que utiliza ovos frescos na sua composição, uma inovação europeia, para além de ter sido a primeira gama sem cereais a ser

¹<https://www.ifs-certification.com/en/>

produzida em Portugal.

1.1.2 Desafios Logísticos Atuais

A gestão manual de encomendas e a inserção de dados em processos logísticos apresentam uma série de desafios no contexto empresarial atual. Estes métodos são lentos e são frequentemente suscetíveis a erros humanos, o que acaba por comprometer a precisão e a eficiência de quem está a realizar tal tarefa. A complexidade e o tempo necessários para gerir estes processos de forma manual acaba por impactar diretamente a produtividade da empresa. Perante este problema, a solução encontrada reside na automação e otimização dos processos através da IA. A partir desta podemos então facilitar a gestão de encomendas, melhorar e automatizar a logística da empresa, e ainda reduzir erros humanos que ocorrem na execução manual destas tarefas, como anteriormente referido, entre outros.

1.1.3 Motivação

A logística desempenha um papel crucial no sucesso operacional das empresas que dependem de processos eficientes para satisfazer a procura do mercado e dos seus clientes. Na PM, como noutras organizações, o fluxo de encomendas, produção e entrega é essencial para manter a competitividade num mercado desafiador como este. Todavia, os processos logísticos manuais, como a inserção de encomendas em sistemas de gestão empresarial (como é o caso do ERP Primavera), apresentam desafios regularmente, tais como:

- Erros humanos: A inserção manual de dados é suscetível a erros, que podem acabar por causar atrasos e custos adicionais;
- Eficiência operacional reduzida: Tarefas repetitivas e manuais consomem muito tempo, acabando este por não ser aproveitado para outras atividades igualmente importantes;

Diante deste cenário, o projeto *Logística: Automação de Encomendas e Otimização com IA* surge como uma solução lógica e inovadora para reformar a área logística da PM de forma a corrigir os desafios acima descritos.

A implementação deste projeto é motivada pela necessidade de ultrapassar estas limitações, promovendo a eficiência operacional, reduzindo erros e retificando a logística da PM com as necessidades de um mercado cada vez mais competitivo e ativo. Algumas destas motivações são:

- Otimização do tempo e recursos humanos: Com a automação da extração e inserção dos dados no ERP Primavera, os funcionários poderão dedicar-se a outras tarefas;
- Redução de custos e aumento da produtividade: Ao minimizar os erros humanos e maximizar o processo logístico, a PM pode reduzir custos relacionados a atrasos e melhorar a distribuição de recursos.

Diante disto o projeto não apenas aborda os desafios operacionais como também impulsiona a empresa para um novo nível de eficiência e competitividade, refletindo-se em melhores resultados para a empresa.

1.1.4 Integração

A integração na PM foi um processo relativamente simples, dado que houve um excelente acolhimento por parte da equipa, o que permitiu compreender rapidamente a estrutura e o funcionamento da empresa, facilitando a adaptação ao ambiente empresarial e às exigências da minha função.

Pertenço ao Departamento de Informática da empresa em conjunto com 2 outros trabalhadores, tendo como função o desenvolvimento de diversas tarefas relacionadas com software, em especial a continuação do desenvolvimento do NETPET, uma aplicação Web realizada no âmbito de um projeto de mestrado de Engenharia Informática - Internet das Coisas, no ano 2023/2024 pelo Diretor de Operações da PM, que é um dos constituintes do meu Departamento.

Embora a maioria das tarefas sejam desempenhadas de forma independente, é necessário por vezes trabalhar em equipa quando surgem tarefas de uma maior complexidade. Este equilíbrio entre colaboração e autonomia tem sido essencial para o desenvolvimento das tarefas, permitindo a contribuição de soluções mais engenhosas, suportadas pelos restantes colegas do Departamento, quando necessário.

O ambiente de trabalho da PM é, portanto, caracterizado por um forte foco na inovação, comunicação entre colaboradores e procura constante por melhorias. Tudo isto permitiu uma rápida adaptação e contribui para a motivação durante a realização das tarefas propostas.

1.1.5 Duração e Cronologia

Este estágio teve a duração de 39 semanas, nas quais foram desenvolvidas diversas funcionalidades que irão ser abordadas posteriormente, como ilustram os seguintes gráficos Gantt (Figura 1.1 e Figura 1.2):

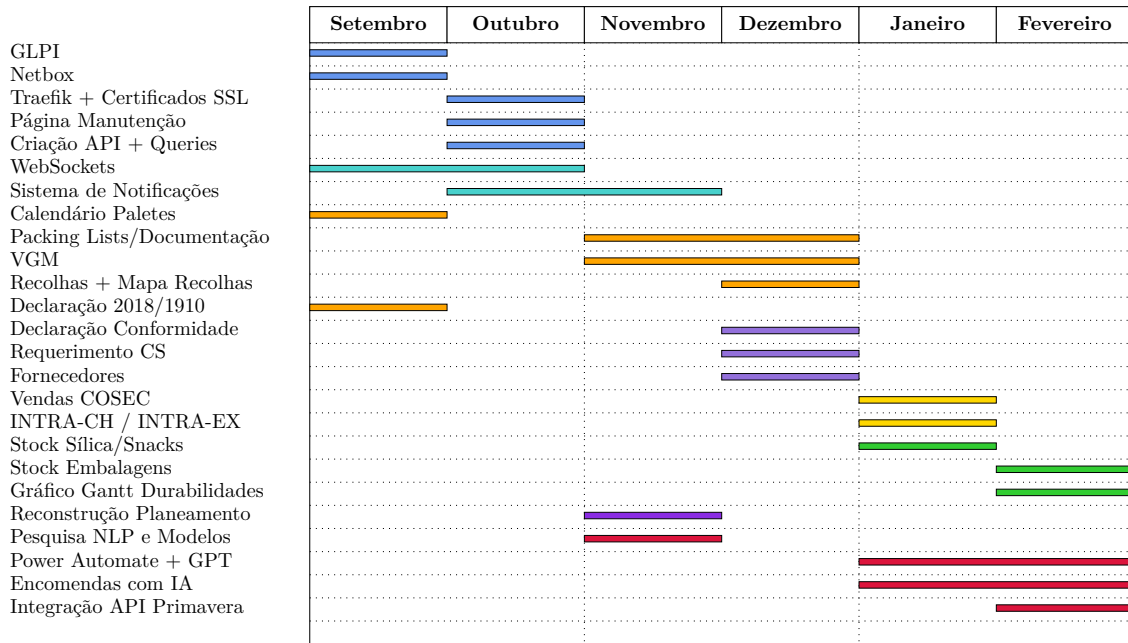


Figura 1.1: Atividades Realizadas - Setembro a Fevereiro

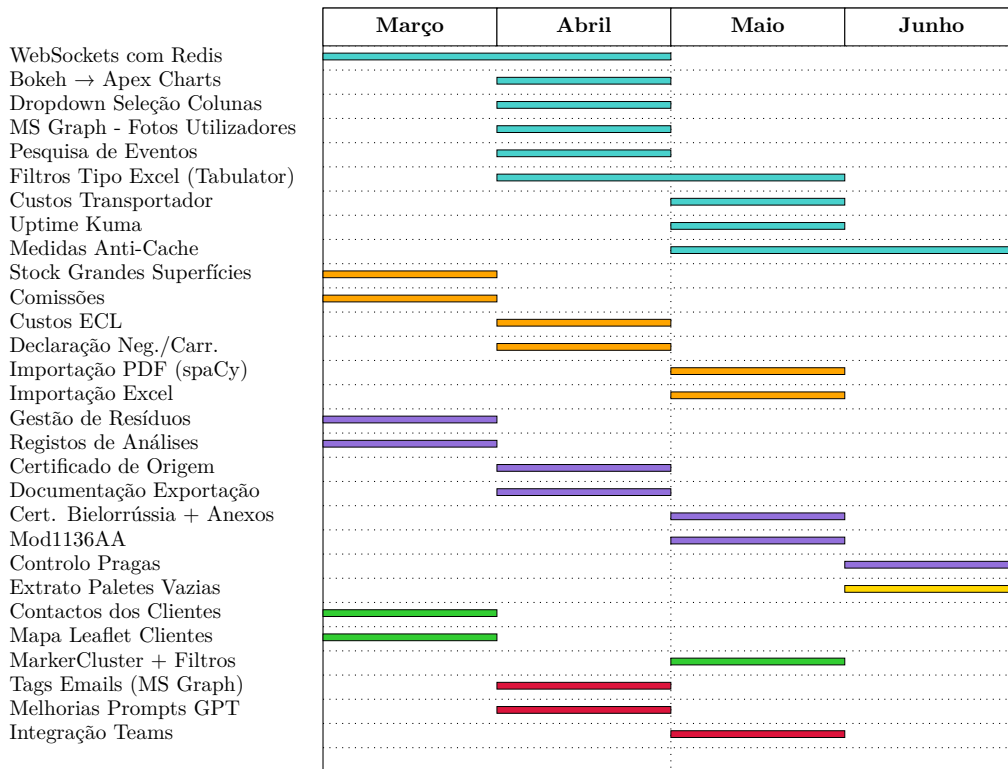


Figura 1.2: Atividades Realizadas - Março a Junho



1.2 Objetivos e Âmbito

1.2.1 Evolução dos Requisitos

O projeto surgiu, como já anteriormente referido no Capítulo 1.1.2, em resposta a necessidades logísticas concretas detetadas internamente na empresa, nomeadamente a elevada dependência de processos manuais para a inserção de dados e gestão de encomendas, o que levava frequentemente a erros humanos, atrasos e uma utilização pouco eficiente dos recursos humanos disponíveis.

Ao longo do estágio, os objetivos do projeto foram sendo ajustados conforme foram surgindo os desafios, de forma a melhor responder à realidade encontrada no dia a dia da empresa. Numa fase inicial, foi considerada a utilização de modelos de inteligência artificial como o *Bidirectional Encoder Representations from Transformers* (BERT). No entanto, a elevada variabilidade dos dados, a complexidade do pré-processamento e as exigências em termos de capacidade computacional acabaram por tornar essa abordagem pouco viável. Como resultado, foi escolhida uma solução mais prática e integrada: a utilização do Power Automate² em conjunto com o modelo *Generative Pre-training Transformer* (GPT) 3.5/4. Esta escolha revelou ser mais eficiente e alinhada com o ecossistema Microsoft já adotado na empresa, o que facilitou a integração com o Outlook e outras ferramentas da Microsoft.

Para além dos desafios próprios do projeto, ao longo do estágio foram também exploradas, ajustadas ou substituídas diversas bibliotecas e *frameworks* — como a troca do *Plotly*³ pelo *Bokeh*⁴ ou do *xhtml2pdf*⁵ pelo *PDFKit*⁶ — sempre com o objetivo de melhor responder às necessidades concretas dos utilizadores e dos diferentes departamentos.

A arquitetura da solução também foi progressivamente refinada, culminando numa integração plena com o sistema NETPET, garantindo centralização de funcionalidades, uniformidade no acesso aos dados e maior eficiência na gestão de processos.

1.2.2 Âmbito de Trabalho

O trabalho desenvolvido durante o estágio envolveu três projetos principais integrados com objetivos e tecnologias distintas, mas com uma visão comum: a modernização da PM.

²<https://www.microsoft.com/pt-pt/power-platform/products/power-automate?market=pt>

³<https://plotly.com/>

⁴<https://bokeh.org/>

⁵<https://github.com/xhtml2pdf/xhtml2pdf>

⁶<https://pdfkit.org/>

- NETPET: Uma aplicação Web desenvolvida fazendo uso das *frameworks Django*⁷ e *Bootstrap*⁸, que serve como interface central para gestão interna de processos. Foram desenvolvidas diversas funcionalidades como dashboards, planejamento de ordens de produção e ensaque, criação de PDFs, notificações em tempo real usando *WebSockets*, assim como páginas dedicadas ao tratamento de encomendas, facilitando o trabalho de diversos departamentos.
- Automação com IA: Este projeto foi dedicado à criação de um sistema de extração automática de dados de encomendas recebidas por email, recorrendo a fluxos no Power Automate e aos modelos GPT 3.5/4. Os dados extraídos eram posteriormente processados através de técnicas de *fuzzy matching*, implementadas com o apoio da biblioteca *TheFuzz*⁹, e integrados numa base de dados interna, para posterior validação e envio. Esta solução reduz o esforço manual, aumenta a eficiência dos colaboradores e minimiza erros humanos.
- DevOps e Infraestrutura: Para assegurar a estabilidade e a escalabilidade do sistema, foram implementadas melhorias na infraestrutura já existente - baseada em *Docker*¹⁰. Foram integradas tecnologias como o *Traefik*¹¹, utilizado como gestor de certificados digitais de segurança, do inglês *Secure Sockets Layer* (SSL) e de tráfego; o *Redis*¹², para armazenamento de tarefas em *background*; e o *Celery*¹³, responsável pela gestão de tarefas assíncronas. Estas ferramentas, entre outras, permitiram distribuir melhor as cargas de trabalho, automatizar processos e garantir que os serviços se mantêm disponíveis de forma contínua e fiável.

1.3 Estrutura do Relatório

Este relatório está organizado em cinco capítulos, estruturados de forma a apresentar de forma clara e coerente as diferentes fases do estágio e do trabalho realizado.

No Capítulo 1 é apresentado o enquadramento geral do estágio, incluindo o contexto empresarial, os objetivos definidos, a motivação, o âmbito do trabalho e a presente descrição da estrutura do relatório.

O Capítulo 2 descreve as necessidades identificadas, as metodologias de desenvolvimento adotadas e as tecnologias selecionadas para a execução do projeto.

O Capítulo 3 detalha as soluções desenvolvidas durante o estágio, abrangendo as três vertentes principais do estágio: a plataforma NETPET, a automação de processos logísticos com IA e as melhorias na infraestrutura na empresa.

No Capítulo 4 são analisados os resultados obtidos, com o objetivo de avaliar

⁷<https://www.djangoproject.com/>

⁸<https://getbootstrap.com/>

⁹<https://github.com/seatgeek/thefuzz>

¹⁰<https://www.docker.com/>

¹¹<https://traefik.io/traefik>

¹²<https://redis.io/>

¹³<https://docs.celeryq.dev/en/stable/>

o impacto das soluções implementadas, o desempenho do estagiário e a percepção dos colaboradores quanto às melhorias verificadas.

Por fim, o Capítulo 5 apresenta as considerações finais, refletindo sobre o trabalho desenvolvido, as competências adquiridas e as possíveis melhorias a implementar no futuro.

Esta página foi intencionalmente deixada em branco.

Capítulo 2

Enquadramento e Metodologia

Neste capítulo vamos abordar a escolha das diversas tecnologias e ferramentas utilizadas durante estágio.

2.1 Análise de Necessidades

Durante a fase inicial de desenvolvimento e integração na empresa PM, foi possível identificar algumas lacunas na infraestrutura que impactavam diretamente a eficiência operacional da instituição. Uma das primeiras limitações observadas foi a ausência de sistemas centralizados para a gestão de equipamentos e suporte técnico. Esta necessidade motivou a implementação do *Gestionnaire Libre de Parc Informatique* (GLPI)¹ e do *Netbox*², softwares de gestão de inventário e de infraestrutura, respetivamente, que suprimiram estas carências. Outro ponto crítico identificado foi o uso de certificados SSL distintos para cada aplicação interna da empresa, aliado à ausência de uma solução centralizada para redirecionamento adequado aos DNS internos. Para resolver essa limitação, foi implementado o *Traefik*, um *reverse proxy* HTTP moderno e *load balancer*, que para além disto efetua também a administração dos certificados SSL para os serviços, de forma centralizada.

A plataforma NETPET, sendo o núcleo de diversas operações da PM, apresentou também diversas oportunidades de melhoria. Uma das principais foi a integração com o ERP Primavera, que permite acelerar diversos processos e reduzir os riscos de erro humano, como é o caso da inserção das encomendas com ajuda da IA, que foi o tema principal abordado no estágio. Para isto foi necessária a utilização de uma interface de programação de aplicações Web, do inglês *Application Programming Interface* (API), do ERP. Outra necessidade latente foi a automação de tarefas repetitivas. Verificou-se que a criação de documentos, como Declarações, *Packing Lists* e *Verified Gross Mass* (VGM), era realizada manualmente através de ficheiros Excel, o que não só era demoroso como também penso a

¹<https://www.glpi-project.org/en/>

²<https://netboxlabs.com/>

falhas. A adoção de bibliotecas como *PDFKit* permitiu automatizar estas etapas, trazendo ganhos substanciais de produtividade. Por fim, foram realizados avanços significativos na área de usabilidade e gestão de tarefas, com a introdução de dashboards que passaram a exibir indicadores relevantes, como o número de encomendas editadas, os principais clientes e os artigos mais encomendados (no caso das encomendas que fazem uso da IA). Além disto, foram desenvolvidas funcionalidades complementares, como notificações, que fazem uso de *WebSockets*, também implementados na plataforma em diversas áreas de visualização de dados, de forma a obter os dados em tempo real. Tudo isto aumentou significativamente o valor da plataforma NETPET para os diferentes departamentos da empresa.

2.2 Metodologia de Desenvolvimento

2.2.1 Linguagem de Programação e *Frameworks*

A linguagem de programação utilizada para a continuação do desenvolvimento do NETPET será o *Python*. Em conjunto com este irá também ser utilizada a *framework Django*, que utiliza o *Python* como linguagem base e também o *Bootstrap* como *framework CSS*. É importante destacar que todas estas já estavam a ser utilizadas anteriormente no projeto, o que reforça a escolha para a sua continuidade.

O componente Web do projeto das Encomendas com IA foi integrado no NETPET para garantir que tudo esteja centralizado no mesmo local. Esta integração permite uma maior uniformidade, facilitando a manutenção, desenvolvimento e acesso às funcionalidades de forma coesa.

- *Python*: É uma linguagem de programação muito utilizada em aplicações Web, desenvolvimento de software, ciência de dados e *machine learning*. A sua alta utilização é derivada da eficiência e curva de aprendizagem acessível que esta oferece, tendo uma sintaxe muito semelhante ao inglês e podendo ser executada em múltiplas plataformas. Para além disto, o *Python* oferece ainda uma grande variedade de bibliotecas (pacotes de códigos prontos para serem utilizados, evitando que quem está a fazer desenvolvimento precise de programar o código todo do zero) em constante desenvolvimento por parte da comunidade desta linguagem, que é bastante ativa.
- *Django*: É um software utilizado para desenvolvimento de aplicações Web de forma rápida e eficaz. Enquanto a maior parte das aplicações Web possui diversas funções comuns, como autenticação ou gestão de cookies, o que frequentemente exige a codificação de funcionalidades semelhantes para cada projeto, o *Django* agrupa as distintas funções num grande conjunto de módulos reutilizáveis, facilitando o trabalho dos desenvolvedores. É por este motivo que o *Django* é denominado de *framework*, uma vez fornece uma estrutura base com componentes pré-desenvolvidos que servem de suporte à criação de aplicações Web. O *Django* utiliza uma arquitetura *Model-View-Template* (MVT), muito semelhante ao padrão *Model-View-Controller* (MVC), amplamente utilizado

no desenvolvimento de software, especialmente em *frameworks*, pois facilita a organização do código, separando o em diferentes camadas, estruturando as aplicações de forma mais eficiente e consistente.

- *Bootstrap*: É uma *framework front-end* que ajuda com a estilização de interfaces Web, utilizando componentes de CSS e *JavaScript* pré-construídos para elementos comuns como navegação, formulários e botões. Este possui também um *Grid System* reativo, permitindo o ajuste da aplicação Web a diferentes *layouts* e tamanhos de ecrã. Destaca-se ainda pela sua simplicidade, responsividade, vasta e clara documentação e pela disponibilidade de componentes prontos a usar. Neste projeto foi utilizada a versão 4 desta *framework*.

2.2.2 Disponibilização para Produção

A solução adotada para disponibilizar as aplicações desenvolvidas, como o NETPET, aos utilizadores foi o *Docker*. A escolha desta solução deve-se à sua estabilidade, facilidade de uso, escalonamento, manuseamento, entre outros. Esta plataforma permite realizar alterações de forma simples, consumindo menos recursos em comparação com outras soluções, e oferecendo uma manutenção relativamente fácil. O facto de já ser utilizada na disponibilização do NETPET e de outras aplicações anteriores ao estágio foi um fator determinante para a continuidade do uso da mesma.

- *Docker*: É uma plataforma *open-source*, utilizada para o desenvolvimento, disponibilização e execução de aplicações. O *Docker* permite que tenhamos diferentes aplicações completamente isoladas em termos de ambiente de desenvolvimento, à semelhança das *Virtual Machine* (VM), onde cada aplicação possui as suas próprias configurações e versionamento de módulos, eliminando possíveis problemas de compatibilidade, permitindo ainda assim que as aplicações interajam entre si como se estivessem no mesmo sistema. O *Docker* utiliza uma arquitetura cliente-servidor, onde o cliente (*Docker Client*) comunica com o *Docker Daemon*, que acaba por gerir as operações do *Docker*, solicitadas pelo cliente, tais como a construção, execução e distribuição dos contentores. As solicitações do cliente são realizadas através de comandos tais como os da Figura 2.1.

- **docker run**: Iniciar um novo *container* a partir de uma imagem.
- **docker build**: Construir uma nova imagem a partir de um *Dockerfile*.
- **docker pull**: Baixar uma imagem através de um repositório remoto.

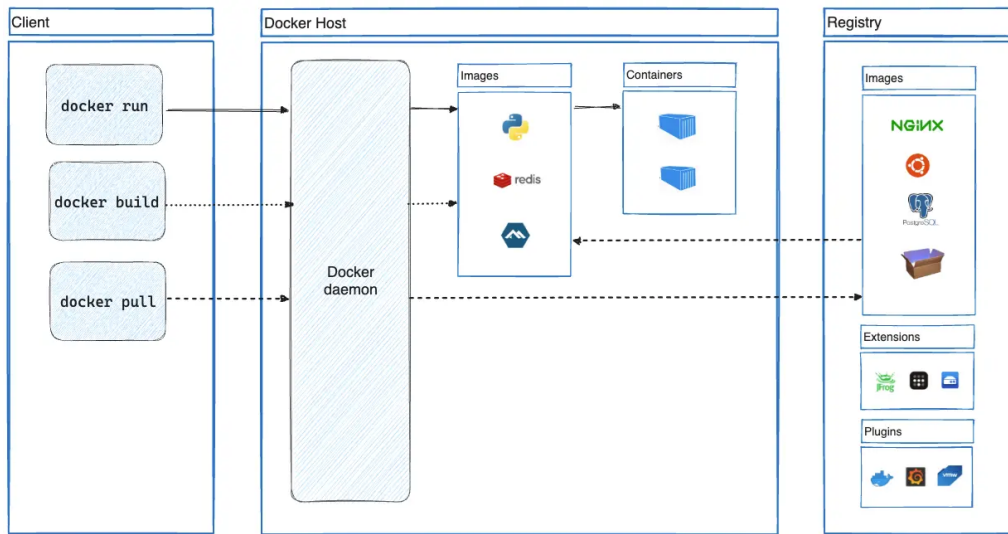


Figura 2.1: Arquitetura do *Docker* (Docker Inc., 2025)

2.2.3 Modelos de IA

A seleção adequada do modelo de IA desempenha um papel fundamental para que a extração dos dados relevantes seja realizada de forma precisa e eficiente. No domínio dos modelos de IA as escolhas são diversas, sendo que cada modelo possui pontos fortes e limitações.

Inicialmente, considerou-se a utilização do modelo BERT para realizar a extração dos dados das encomendas recebidas por email, utilizando o Microsoft Graph, uma API unificada da Microsoft que conecta múltiplos serviços, para que fosse possível realizar a conexão com a caixa de entrada do Outlook, para então realizarmos a extração dos conteúdos dos emails.

Esta solução apresentava, no entanto, alguns desafios. Os e-mails possuem uma grande variedade de formatos e estruturas, pelo que era necessário realizar o pré-processamento dos dados e *fine-tuning* do modelo, processo de treinamento do mesmo com dados específicos, o que iria exigir uma quantidade significativa de dados disponibilizada ao modelo. Para além disto, como é um modelo bastante grande, é também exigente em termos de recursos computacionais. Portanto foi necessário continuar a procura por outras soluções mais viáveis.

Durante a mesma, encontrou-se o Power Automate, uma plataforma *Software as a Service* (SaaS) da Microsoft com o objetivo de otimizar e automatizar *workflows* e processos de negócio, que, embora não seja um modelo de IA, oferece ferramentas integradas de IA, além de disponibilizar recursos da Microsoft que vão de encontro às necessidades da empresa, como a conexão à caixa de entrada da conta de Outlook onde vão ser recebidas as encomendas, sem ser necessário o Microsoft Graph.

Dentro das ferramentas de IA disponibilizadas, destaca-se a ação de extrair palavras-chave de um texto usando o modelo GPT 3.5/4, auxiliada por um *prompt*

indicando a intenção desejada, que acabou por ser a solução escolhida.

Abaixo segue uma descrição do modelo escolhido e do seu funcionamento, assim como descrições breves de outros modelos que foram encontrados durante a pesquisa por uma solução.

GPT

GPT são um tipo de *Large Language Model* (LLM) que utiliza aprendizagem profunda (*Deep Learning*) para processar informações e produzir textos e conteúdos de maneira semelhante aos humanos. As redes neurais, que são um tipo de algoritmo de *machine learning* inspirado no cérebro humano, são treinadas com grandes conjuntos de dados o que permite que estas entendam e produzam respostas pertinentes e coerentes.

Estes modelos representam um avanço significativo na IA, sendo fundamentais para o desenvolvimento de aplicações de IA generativa, como é o caso do ChatGPT.

Os modelos operam como uma rede complexa de neurônios artificiais, estando organizados por camadas para processar informações de forma profunda, assim como o cérebro humano. São construídos na arquitetura *transformer*, um design de rede neural, inventada por pesquisadores da Google e tornada *open-source* em 2017. O *transformer* permite analisar frases inteiras simultaneamente, analisando possíveis relações entre palavras, independentemente da posição das mesmas na frase.

Esta capacidade de captar relações entre palavras é proveniente da “auto-atenção”, um procedimento que permite ao modelo atribuir um peso à importância de cada palavra na frase, assemelhando-se à forma de pensar humana, onde é dada atenção a diferentes partes da frase, de forma a entender o contexto.

BERT

BERT é um modelo de *machine learning* para processamento de linguagem natural. Foi desenvolvido por pesquisadores da Google em 2018 e serve como solução para diversas tarefas de linguagem comuns, tais como análise de sentimentos e reconhecimento de entidades.

spaCy

O *spaCy*³ é uma biblioteca gratuita *open-source* para processamento de linguagem natural em *Python*. Lançada em 2015, esta oferece ferramentas como o *Named Entity Recognition* (NER), *POS tagging*, vetores de palavras, entre outros.

³<https://spacy.io/>

Na Tabela 2.1 apresentam-se as vantagens e as desvantagens dos modelos acima descritos.

Modelo	Vantagens	Desvantagens
GPT	<ul style="list-style-type: none"> - Alta capacidade de entendimento de texto e contexto - Personalização via <i>prompts</i> 	<ul style="list-style-type: none"> - Custo por uso - Dependência de serviços externos
BERT	<ul style="list-style-type: none"> - <i>Open-source</i> e adaptável - Ótimo para tarefas específicas de NLP 	<ul style="list-style-type: none"> - Complexidade na implementação - Requer infraestrutura computacional para o treinamento e ajustes
<i>spaCy</i>	<ul style="list-style-type: none"> - Biblioteca leve e eficiente para NLP - <i>Open-source</i> e personalizável 	<ul style="list-style-type: none"> - Menor capacidade generativa - Exige maior esforço de configuração e integração

Tabela 2.1: Vantagens e desvantagens dos Modelos de IA

Face às características apresentadas na Tabela 2.1, o modelo GPT integrado no Power Automate foi selecionado como a solução mais adequada. Os detalhes da sua implementação e integração na arquitetura do sistema serão abordados no Capítulo 3.2.

Capítulo 3

Desenvolvimento e Implementação

Neste capítulo, é apresentada uma visão detalhada do trabalho realizado ao longo do estágio, destacando as principais tarefas desempenhadas, projetos desenvolvidos e competências adquiridas durante a experiência profissional, com início a 2 de setembro de 2024 e fim a 2 de junho de 2025.

3.1 Plataforma NETPET

3.1.1 Infraestrutura e Funcionalidades Transversais

O trabalho desenvolvido na plataforma começou por se focar na implementação de *WebSockets* em locais específicos do NETPET, com o objetivo de exibir em tempo real as alterações de dados pertinentes. Para este efeito, foi criado um novo *branch* no repositório do *GitHub*¹ onde iriam ser guardadas todas as alterações relacionadas com este tópico.

Foi realizada, de seguida, uma pesquisa sobre os módulos do *Django* mais adequados para a integração de *WebSockets* no projeto. Desta análise, conclui-se que o *Django Channels*² seria a escolha mais adequada às necessidades exigidas. Este módulo possui suporte nativo para o *Asynchronous Server Gateway Interface* (ASGI), permitindo que o *Django* continue a gerir o HTTP tradicional, enquanto o *Django Channels*, utilizando o *Daphne* como servidor, se responsabiliza pela gestão de conexões de *WebSockets* assíncronas, tirando proveito das capacidades do ASGI para uma comunicação mais eficiente e escalável.

Foram então criados três ficheiros essenciais à configuração: `asgi.py`, `routing.py` e `consumers.py`. No primeiro foi configurada a aplicação *Django* de forma a usar o *Django Channels* com suporte a *WebSockets*, além de gerir as

¹<https://github.com/>

²<https://github.com/django/channels/>

requisições HTTP tradicionais. No segundo ficheiro foi definida a rota para os *WebSockets*, indicando o *consumer* da conexão. Por fim, no terceiro ficheiro é especificado o *consumer* `EncomendasConsumer`, que gere as conexões dos clientes e envia dados periódicos sobre as encomendas. A estrutura da diretoria onde se encontram os ficheiros é apresentada na Figura 3.1.

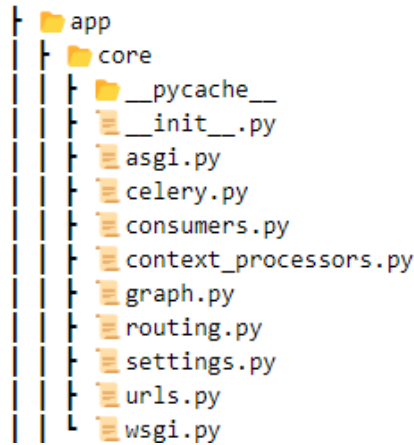


Figura 3.1: Estrutura da diretoria onde se localizam os ficheiros de *WebSockets*

Após a configuração do *WebSockets* no *back-end*, foi utilizado *JavaScript* para criar as conexões no *front-end*. Estas foram integradas, numa fase inicial, no Calendário das Encomendas e no Dashboard Inicial, permitindo observar alterações de encomendas para diferentes dias da semana e o fluxo de encomendas atual em tempo real, respetivamente.

Durante este processo, foi também corrigido um bug no calendário das encomendas que causava a duplicação das mesmas sem motivo aparente. Inicialmente, as encomendas estavam a ser guardadas a partir da série e do número da encomenda. Descobriu-se, contudo, que o *Django* se equivocava ao guardar o número de encomenda indicado em séries diferentes, causando a duplicação. A solução passou por guardar as encomendas a partir do seu *Identifier* (ID), que é único.

Com o objetivo de implementar notificações em tempo real no NETPET, iniciou-se o desenvolvimento de toda a lógica por trás dos *toasts*, notificações pequenas e temporárias, permitindo que estas surgissem em formato *pop-up* com o respetivo conteúdo. Esta implementação revelou-se necessária porque o template utilizado no projeto, *Datta Able*³, não suporta nativamente este tipo de funcionalidade. Paralelamente, foi criada uma página dedicada onde o utilizador pode visualizar todas as suas notificações, com opções para as eliminar, marcar como lidas e até mesmo marcar como importantes para impedir a sua remoção acidental. Neste contexto, foi também reformulada toda a componente visual das notificações já existente, acessível a partir do ícone no canto superior direito. Após a conclusão deste processo, foi criada uma rota de *WebSockets* exclusiva para as notificações, desenvolvendo-se de seguida o *consumer* `NotificationsConsumer`, juntamente com toda a lógica de obtenção assíncrona de informações que constituem a notificação.

³<https://github.com/app-generator/django-datta-able>

A conexão foi estabelecida usando *JavaScript* no *front-end*, completando assim a integração do sistema. A página de Notificações, juntamente com a respectiva notificação em *pop-up*, encontra-se representada na Figura 3.2.

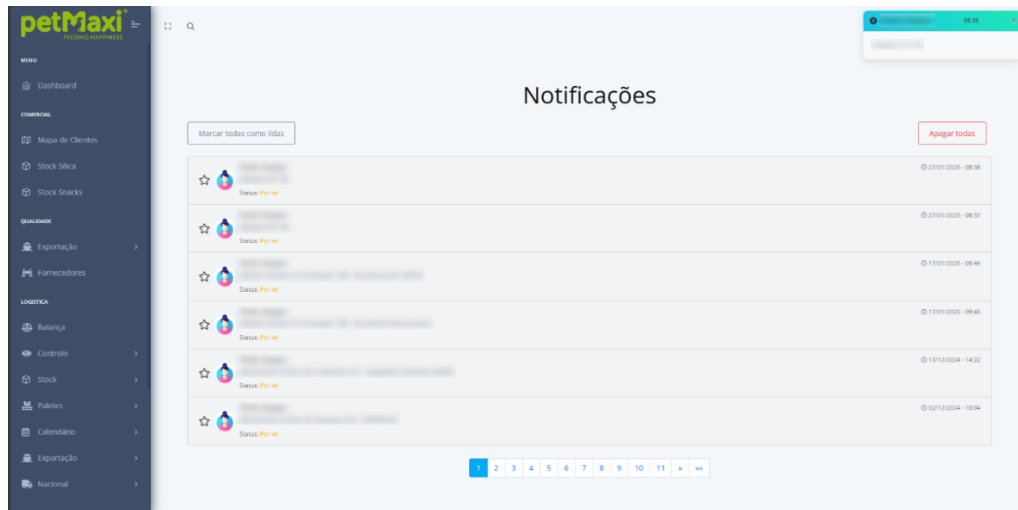


Figura 3.2: Página das Notificações

De seguida avançou-se para um conjunto de melhorias estruturais e visuais. Uma das principais ações foi a uniformização da plataforma, acompanhada de diversas alterações no CSS e HTML em todas as páginas. Exemplos incluem a remoção dos títulos que anteriormente estavam posicionados no canto superior esquerdo de cada vista, passando estes a ser exibidos de forma centralizada na barra de navegação. Esta alteração não só homogeneizou a interface, como também libertou espaço adicional nas vistas, permitindo a inclusão de novos elementos, como botões e filtros personalizados.

No que diz respeito à interatividade e à atualização em tempo real, foram integrados *WebSockets* nas vistas do Stock Primavera, Encomendas e Stock de Grandes Superfícies. Estas implementações possibilitaram uma consulta mais rápida dos dados das tabelas e evitaram a necessidade de constantes atualizações manuais da página.

Dada a crescente complexidade das funcionalidades implementadas, foram posteriormente realizadas alterações estruturais no funcionamento dos *WebSockets*. Até este momento, o *back-end* recorria à memória local para o funcionamento dos mesmos, o que implicava que estes estavam restritos a uma única instância, limitando a sua utilização em determinados cenários. Para ultrapassar esta limitação, optou-se pela adoção de um servidor *Redis*, permitindo que diferentes instâncias do servidor partilhassem informações de forma centralizada. Esta alteração trouxe ainda uma vantagem adicional: o *Celery*, que corre separado do servidor principal, passou também a poder comunicar através dos *WebSockets* — algo que não era possível quando a comunicação estava limitada à memória local. Complementarmente, foram ainda introduzidos layer groups, um mecanismo do *Django Channels* que gere as ligações *WebSocket* agrupando-as em canais ou grupos identificados por nome. Esta adição permitiu a otimização do código já existente e a redução

significativa de erros.

Para melhorar a experiência do utilizador nas tabelas, foi desenvolvido um dropdown ao estilo do Excel (Figura 3.3), com checkboxes que possibilitam a aplicação de múltiplos filtros em simultâneo. Adicionalmente, os filtros podem ser guardados em cache, caso o utilizador assim o deseje, garantindo maior flexibilidade e eficiência no uso das tabelas da biblioteca *Tabulator*⁴. A criação desta funcionalidade foi derivada da não existência de uma filtragem deste tipo nativa da biblioteca.

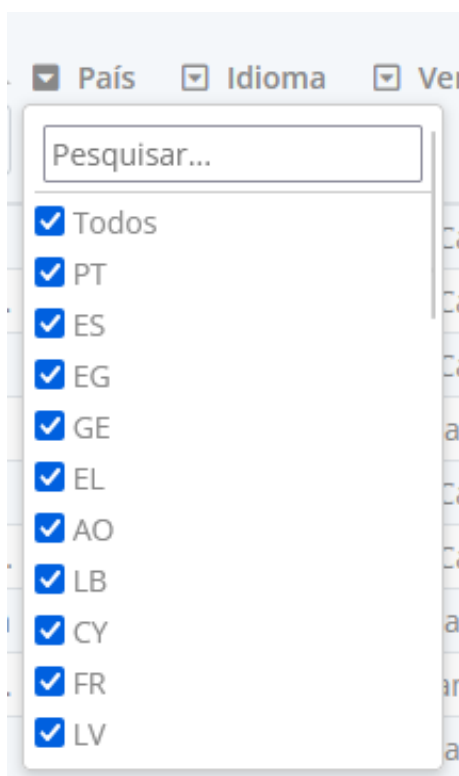


Figura 3.3: Filtro estilo Excel

Outra melhoria significativa foi a integração das fotografias dos utilizadores no NETPET. Estas são obtidas através da API Microsoft Graph e armazenadas em cache para melhorar o desempenho e a velocidade de carregamento, tornando a interface mais personalizada e profissional. Esta implementação pode ser observada na Figura 3.4.

⁴<https://tabulator.info/>

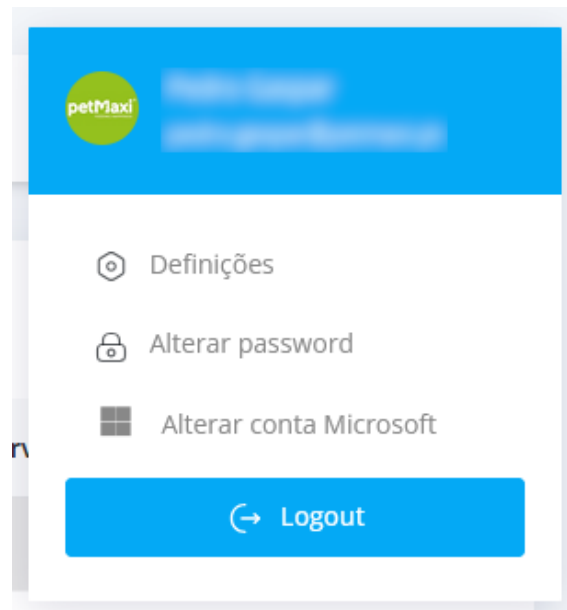


Figura 3.4: Imagem de perfil na dropdown de opções

Para aumentar a resiliência do sistema, foram introduzidas verificações condicionais na base de dados do MultiDOS - software de produção desenvolvido pela empresa JCE e utilizado pela PM - assegurando que, em caso de falha de ligação, a utilização do NETPET não seja comprometida. Como os dados desta base de dados são apenas informativos, a sua indisponibilidade não afeta o funcionamento dos restantes recursos. A monitorização da disponibilidade é feita através do *Uptime Kuma* (Figura 3.5), permitindo a rápida deteção de falhas de ligação. Em complemento, foi utilizada a API do *Docker* para automatizar o reinício do *container* onde está alojada a plataforma, aplicando uma variável de ambiente que define se a base de dados deve ou não ser utilizada, por meio de um endpoint intermédio. Esta abordagem garante a continuidade do serviço de forma automática, eliminando a necessidade de intervenção manual.

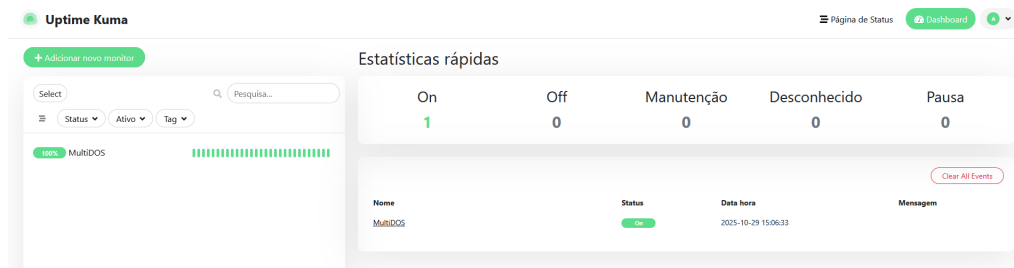


Figura 3.5: Uptime Kuma

Para facilitar a navegação, a pesquisa de eventos no Calendário das Encomendas foi aprimorada, destacando visualmente o evento encontrado. Esta funcionalidade contribui para uma experiência de utilização mais intuitiva e eficiente, observável na Figura 3.6.

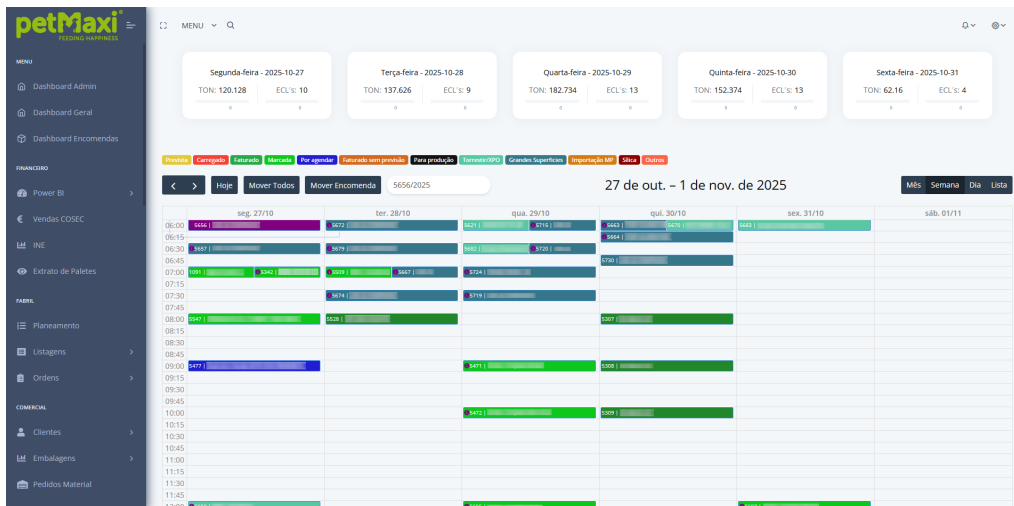


Figura 3.6: Encomenda destacada

Foi criada, em seguida, uma nova vista designada Custos de Transportador, que integra duas funcionalidades distintas: as listagens dos Custos Torrestir e XPO. Adicionalmente, esta vista contempla ainda uma listagem com os dados referentes às demais transportadoras não incluídas nas anteriores. Em paralelo, foi desenvolvida a funcionalidade Custos ECL (Figura 3.7), acompanhada por uma reestruturação visual da modal de edição das encomendas, que passou a incluir novos campos. Foi também criada uma página dedicada à visualização dos diferentes custos agrupados por encomenda. Adicionalmente, a modal de visualização e edição dos Custos ECL foi integrada na listagem de Custos do Transportador, bem como nas funcionalidades de Controlo Pós-venda e CCE, assegurando a consistência da informação em toda a plataforma.

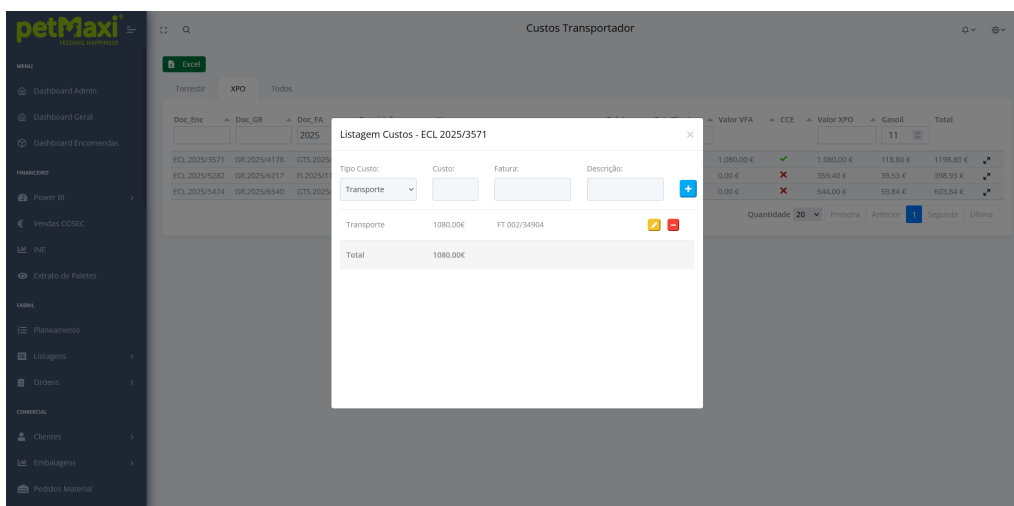


Figura 3.7: Modal dos Custos ECL

Por fim, durante o desenvolvimento da plataforma, identificou-se a necessidade de garantir a correta atualização das funcionalidades dependentes de *JavaScript* após a realização de novos *deploys* em ambiente de produção. Para resolver este

problema, foi implementado o *WhiteNoise*⁵, uma biblioteca do *Python* que efetua o controlo do cache através da atribuição de *hashes* únicos a cada ficheiro estático. Desta forma, sempre que ocorre uma nova atualização, o navegador é forçado a descarregar a versão mais recente dos recursos, evitando o carregamento de ficheiros desatualizados. Esta solução assegura que elementos críticos da aplicação não permaneçam em versões antigas, evitando assim o surgimento de erros e aumentando a estabilidade do sistema em ambiente de produção.

3.1.2 Departamento de Logística

Neste departamento a primeira funcionalidade desenvolvida consistia na adição de uma página onde possam ser visualizados os dados da Declaração 2018/1910, um documento relacionado com questões fiscais ou tributárias que contém orientações sobre o cumprimento de regras fiscais, assim como o IRS, para cada cliente. O objetivo principal é permitir que, através de poucos cliques, seja possível enviar um email para os clientes selecionados contendo em anexo um PDF com os dados relevantes para uma determinada data (tipo e número do documento das encomendas, número de toneladas, entre outros).

Para a visualização dos dados, foi utilizado o *Tabulator*, uma biblioteca *JavaScript open-source* que permite a criação de tabelas interativas com diversas funcionalidades incluídas de base, como ordenação e filtros das colunas. No entanto, apesar do vasto conjunto de funcionalidades que oferece, a biblioteca não possui checkboxes para seleção de linhas nativamente, tendo sido necessário desenvolver esta funcionalidade de raiz.

Posteriormente, foi criado um template HTML com o formato e as informações que o PDF deveria incluir. Para a transformação do HTML em PDF, recorreu-se ao módulo *xhtml2pdf*, que inclui a biblioteca *Pisa*⁶, uma ferramenta *Python* que permite a criação de ficheiros PDF a partir de conteúdo HTML e CSS.

No que respeita ao envio dos emails, foi utilizado o *EmailMultiAlternatives*, pertencente à biblioteca de emails incluída no *Django*. Adicionalmente, implementou-se o módulo *django-celery-beat*⁷ em conjunto com o *Celery*, configurado com *Redis* para armazenamento das filas e dados das tarefas. Esta configuração possibilitou a criação de uma tarefa periódica que permite o envio automático de emails para os clientes, desde que o email ainda não tenha sido enviado anteriormente. Para garantir o funcionamento adequado, é necessário inicializar o *Celery Beat*, responsável pelo agendamento das tarefas, juntamente com o *Celery Worker*, que as executa. O resultado final desta funcionalidade encontra-se na Figura 3.8.

⁵<https://github.com/evansd/whitenoise>

⁶<https://github.com/icecube/pisa>

⁷<https://github.com/celery/django-celery-beat>

Declaração 2018/2019

Todos

TipoDoc	Nº Fatura	Data	Cliente	País	TON	Valor	Destino FL...	Transport...	Enviado	CCE
FR	1	2018-01-02	Daniel Portilla Alonso	ES	29.992	418.276,76	ES	Marfina	✓	✓
FR	30	2018-01-14	Daniel Portilla Alonso	ES	23.459	470.213,14	ES		✗	✗
FR	32	2018-02-09	Daniel Portilla Alonso	ES	23.954	420.507,80	ES		✓	✓
FR	102	2018-03-21	Daniel Portilla Alonso	ES	24.12	478.904,78	ES	Marfina	✓	✓
FR	267	2018-03-13	Daniel Portilla Alonso	ES	21.390	478.604,18	ES	Marfina	✓	✓
FR	297	2018-04-03	Daniel Portilla Alonso	ES	24.278	478.211,70	ES	Marfina	✓	✓
FR	279	2018-02-02	Daniel Portilla Alonso	ES	20.974	478.408,87	ES	Marfina	✓	✓
FR	524	2018-06-19	Daniel Portilla Alonso	ES	21.77	420.108,24	ES	Marfina	✓	✓
FR	392	2018-06-19	Daniel Portilla Alonso	ES	24.887	478.014,04	ES	Marfina	✓	✓
FR	438	2018-03-08	Daniel Portilla Alonso	ES	25.605	478.808,96	ES	joan marfina	✓	✓
FR	261	2018-04-10	Daniel Portilla Alonso	ES	14.114	478.269,87	ES	joan marfina	✓	✓
FR	529	2018-08-28	Daniel Portilla Alonso	ES	14.860	420.733,89	ES		✗	✗
FR	442	2018-12-19	Daniel Portilla Alonso	ES	14.872	478.962,67	ES		✓	✗
FR	442	2018-12-19	Daniel Portilla Alonso	ES	14.802	420.907,86	ES		✓	✗
FR	731	2018-11-05	Daniel Portilla Alonso	ES	34.238	478.371,86	ES		✗	✗
FR	368	2018-11-30	Daniel Portilla Alonso	ES	12.910	478.709,02	ES		✗	✗
FR	886	2018-12-10	Daniel Portilla Alonso	ES	22.270	478.588,42	ES		✗	✗
FR	384	2018-10-24	Daniel Portilla Alonso	ES	9.702	420,00	ES		✗	✗

Quantidade 20 | Primeira | Anterior | 1 | 2 | 3 | 4 | 5 | Seguinte | Última

Figura 3.8: Declaração 2018/2019

Dando continuidade ao trabalho no Departamento Logístico, desenvolveu-se outra funcionalidade que visava a criação automatizada de ficheiros PDF a partir de dados das *Packing Lists*. Os dados são apresentados numa tabela criada com a biblioteca *Tabulator*, alimentada por uma *view* existente na base de dados do NETPET, que retorna os dados necessários das tabelas da base de dados do ERP Primavera. O utilizador necessita apenas de selecionar os dados desejados e determinar se o meio de transporte é navio ou comboio, tudo através de poucos passos. A designação desta foi posteriormente renomeada para um nome mais genérico *Documentação*, refletindo a sua expansão, permitindo exportar não apenas *Packing Lists*, mas também diversos documentos em formato PDF, tais como a Declaração Negativa, a Declaração de Carregamento e a Declaração de Cedência de Instalação, todas alimentadas pelos mesmo dados.

Para esta funcionalidade, optou-se pela biblioteca *PDFKit* na conversão do template HTML para PDF, em substituição do *xhtml2pdf* utilizado anteriormente. Esta mudança deveu-se às limitações significativas do *xhtml2pdf* em termos de suporte CSS, o que impedia a criação de um ficheiro exatamente igual ao original pretendido.

A necessidade desta funcionalidade surgiu do facto de a criação anterior dos ficheiros ser realizada manualmente através de um ficheiro Excel, tornando o processo demorado e suscetível a erros. Com esta automação, o processo tornou-se significativamente mais eficiente e menos propenso a erros humanos, uma vez que eliminou a necessidade de introdução manual de dados. O resultado pode ser observado na Figura 3.9.

Cliente	TipoDoc	Serie	NumDoc	Incluir Palette	Incluir Marketing	Navio/Camião
	FP	2025E	1141	✓	✓	☑
	FP	2025E	1109	✓	✓	☑
	FP	2025E	1111	✓	✓	☑
	FP	2025E	1064	✓	✓	☑
	FP	2025E	1096	✓	✓	☑
	FP	2025E	1206	✓	✓	☑
	FP	2025E	1097	✓	✓	☑
	FP	2025E	1204	✓	✓	☑
	FP	2025E	1205	✓	✓	☑
	FP	2025E	1098	✓	✓	☑
	FP	2025E	1116	✓	✓	☑
	FP	2025E	957	✓	✓	☑
	FP	2025E	1048	✓	✓	☑
	FP	2025E	1050	✓	✓	☑
	FP	2025E	1244	✓	✓	☑
	FP	2025E	1051	✓	✓	☑
	FP	2025E	1049	✓	✓	☑
	FP	2025E	1076	✓	✓	☑
	FP	2025E	1247	✓	✓	☑
	FP	2025E	1074	✓	✓	☑

Figura 3.9: Página da Documentação

Seguindo a mesma abordagem da funcionalidade anterior, foi desenvolvida uma solução similar para a geração de documentos VGM. Neste caso, os valores são retirados da tabela VGM, também alimentada pelos dados da *view* da Documentação. Para tal, foi necessário criar um novo template HTML para posterior transformação em PDF. Esta função oferece ainda a possibilidade de alterar algumas informações do ficheiro a gerar, caso estas venham incorretas ou sejam inexistentes nas bases de dados.

Atendendo a outra solicitação do departamento, foi criada uma página dedicada à visualização de todas as recolhas de paletes, organizadas por tipo e apresentando de forma clara o seu estado (pendentes, agendadas, entre outros). Para complementar esta funcionalidade, recorreu-se ao módulo *Leaflet*, que permite a criação de mapas interativos. Foi também integrada a biblioteca de *JavaScript*, *Select2*, que possibilita a seleção intuitiva da entidade desejada ao adicionar ou editar uma recolha. A Figura 3.10 apresenta a listagem de todas as recolhas de paletes registadas no sistema, organizada por tipo e estado.

Estado	Data Pedido	ECL	Nome	Morada	Localidade	Código Postal	Paletes	Transportador	Cliente C. RZ	Observações
Pendente	2024-12-23						1		Sim	🔍

Figura 3.10: Recolhas de Paletes

Na Figura 3.11 podemos observar a respetiva localização das recolhas num mapa interativo, permitindo identificar rapidamente as zonas com maior concentração das mesmas.

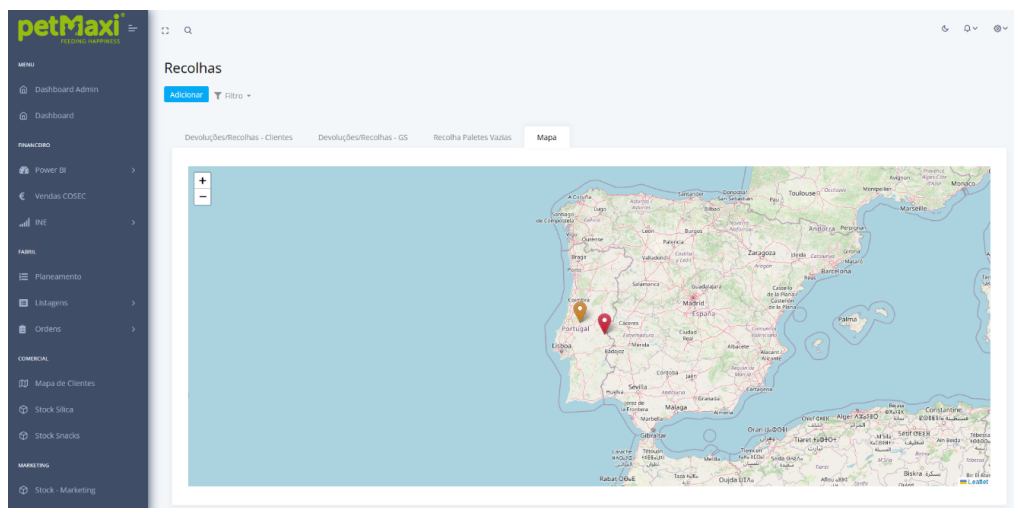


Figura 3.11: Mapa das recolhas de paletes

Posteriormente, foram adicionadas funcionalidades de importação de documentos em PDF e Excel para integração de encomendas no ERP Primavera, conforme ilustrado na Figura 3.12.



Figura 3.12: Página de importação de documentos

No caso específico dos PDFs, recorreu-se inicialmente às bibliotecas *spaCy-Layout* e *spaCy*, que permitem a extração estruturada de dados tabulares contidos nos documentos. Contudo, esta abordagem revelou-se demasiado pesada e ineficiente para o objetivo em causa, além de apresentar limitações na fiabilidade dos dados extraídos. Por esta razão, optou-se pela utilização da biblioteca *pdfplumber*⁸, que se revelou significativamente mais rápida e eficiente. Esta biblioteca

⁸<https://www.pdfplumber.com/>

permite ainda delimitar com precisão a posição das colunas e linhas das tabelas nos ficheiros, garantindo uma maior robustez e fiabilidade no processo de extração.

Seguidamente, foi reconstruída de raiz uma funcionalidade já existente mas que carecia de melhorias significativas: o Stock de Grandes Superfícies (Figura 3.13). Na nova versão, passou a ser possível adicionar ou remover colunas da tabela conforme a necessidade de cada utilizador, tornando a visualização mais flexível e ajustada a diferentes contextos de uso.

Artigo	Descrição	Stock Sacos	Stock - ECL	Quantidade Encomendada	Stock Embalagens	Fórmula
9 Items						
		160	-0.48	320	14610	
		426	1.278		14610	
		790	9.32	324	13921	
		240	1.2	120	5365	
		4938	17.192	640	21869	
		620	1.11	546	9780	
		5169	13.636	1760	26125	
		4561	7.202	960	14372	
		891	5.91	300	13175	
		17795	56.368	4970	133828	
14 Items						
		5925	8.887		31980	
		271	-0.261	300	11249	
		3542	7.084		30569	
		4517	18.068		31622	
		4448	17.792		9696	
		549	7.098	42	3742	
		7289	29.156		62635	
		1289	11.601		4594	

Figura 3.13: Listagem do Stock das Grandes Superfícies

Para implementar esta funcionalidade, foi necessário criar uma dropdown personalizada que permitisse selecionar as colunas a exibir, tendo em conta que esta opção não existe nativamente no *Tabulator*. Foram ainda introduzidas várias melhorias funcionais que aumentaram significativamente a eficiência e a fiabilidade do processo de consulta e gestão de stocks.

Após a conclusão da funcionalidade anterior, procedeu-se à refatoração completa do dashboard da página da Balança. A biblioteca utilizada para a construção dos gráficos foi alterada do *Plotly* para o *Bokeh*, devido aos diversos problemas apresentados pelo *Plotly*, nomeadamente a inconsistência estética dos gráficos após aplicação de filtros e a impossibilidade de alteração das cores das barras, apesar de estarem especificadas no código. A Figura 3.14 apresenta o dashboard da balança refatorado, exibindo as diferenças de peso entre Guia e Balança para diversos produtos.

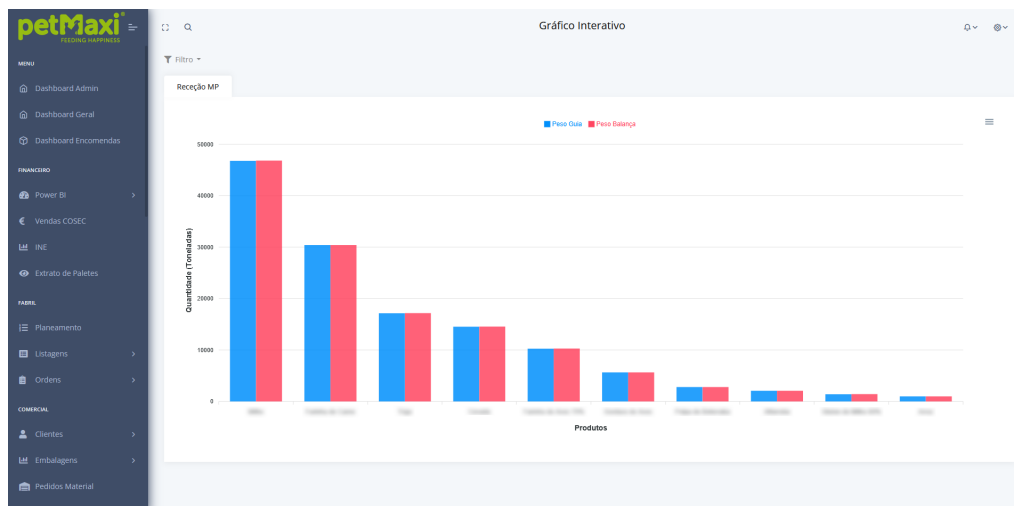


Figura 3.14: Dashboard dos dados da balança

Por fim, foi criada uma página dedicada à listagem de três comissões distintas: Sicarze, Liliana e Serviproduarte. Esta funcionalidade, à semelhança de outras anteriormente implementadas, permite apresentar os dados de forma centralizada e intuitiva, facilitando a análise e oferecendo ainda a possibilidade de exportação para PDF ou Excel, consolidando assim as ferramentas de gestão disponíveis para o Departamento Logístico.

3.1.3 Departamento da Qualidade

Inicialmente, foram solicitadas duas funcionalidades pelo Departamento da Qualidade: o desenvolvimento de uma página que criasse os ficheiros PDF da Declaração de Conformidade e do Requerimento CS, alimentadas também pela *view* da Documentação, e outra página onde fossem apresentados os Fornecedores, divididos por tipo, juntamente com as suas informações.

Para a primeira tarefa, o processo elaborado foi semelhante aos descritos anteriormente, com uma particularidade: era necessário criar ambos os ficheiros simultaneamente. No segundo ficheiro, apenas a primeira página mantém o formato vertical, enquanto as restantes adotam o formato horizontal. Esta especificidade exigiu a utilização de uma biblioteca adicional, o *PyPDF2*⁹, que permite transformar os *templates* utilizados para gerar os PDFs em *streams* e posteriormente juntá-los, criando assim o ficheiro conforme pretendido. Em casos específicos, foi ainda necessária a geração de um terceiro ficheiro simultâneo quando o cliente era de Marrocos.

A função permite ainda selecionar um expedidor diferente do padrão (petMaxi), fornecendo depois as opções de destinatário correspondentes ao expedidor escolhido. Esta funcionalidade tornou a criação destes ficheiros significativamente mais fácil e rápida, automatizando todo o processo e aumentando consideravel-

⁹<https://github.com/py-pdf/pypdf>

mente a sua eficiência. Dado que este processo era particularmente demorado e consumia muito tempo ao funcionário responsável, a sua automatização permitiu otimizar substancialmente o tempo de execução desta tarefa e aumentar a produtividade do utilizador.

Para a segunda tarefa, recorreu-se ao *Bootstrap* para criar as tabelas que apresentam os fornecedores e os seus dados – certificados, datas de validade, etc. – organizados por tipo, como se pode observar na Figura 3.15.

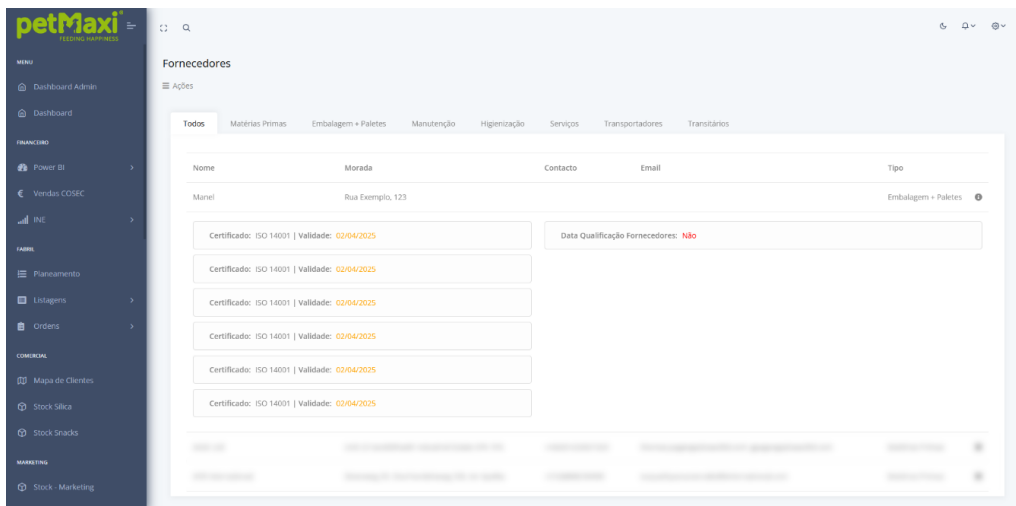


Figura 3.15: Lista de Fornecedores

Numa fase posterior, o Departamento da Qualidade solicitou duas novas funcionalidades: Gestão de Resíduos e Listagem de Registos de Análises.

A primeira consiste no desenvolvimento de um calendário (Figura 3.16) que permite o registo e acompanhamento dos resíduos.

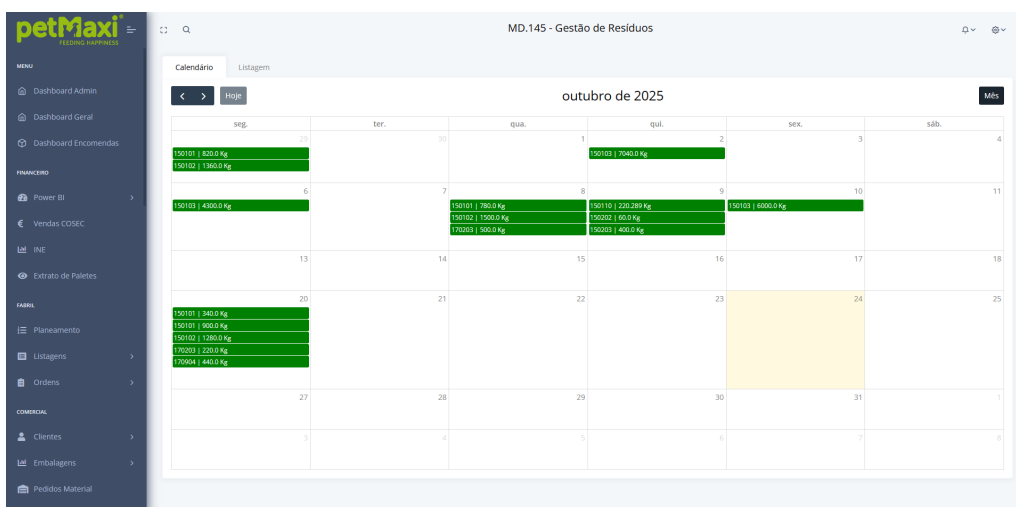


Figura 3.16: Página de registo e acompanhamento dos resíduos

Após o registo, a informação é apresentada numa listagem agrupada por data e

tipo, permitindo uma visualização estruturada e intuitiva dos dados (Figura 3.17).

Código LER	Designação	Quantidade (Kg)	e-Gar	Data de Operação	Data de Registro
2025 - 150101 - Embalagens de Papel e Cartão Total: 30355 Kg					
1º Trimestre					
150101	Embalagens de Papel e Cartão	1320	PT20250109101095	09/01/2025	13/03/2025
150101	Embalagens de Papel e Cartão	1520	PT20250117263263	17/01/2025	13/03/2025
150101	Embalagens de Papel e Cartão	1520	PT20250129430651	29/01/2025	13/03/2025
150101	Embalagens de Papel e Cartão	135	PT20250204046637	04/02/2025	13/03/2025
150101	Embalagens de Papel e Cartão	1620	PT20250212159393	12/02/2025	13/03/2025
150101	Embalagens de Papel e Cartão	840	PT20250221317613	21/02/2025	14/03/2025
150101	Embalagens de Papel e Cartão	1060	PT20250228446981	28/02/2025	14/03/2025
150101	Embalagens de Papel e Cartão	1320	PT20250314187458	14/03/2025	14/03/2025
150101	Embalagens de Papel e Cartão	200	PT20250324334850	24/03/2025	10/04/2025
150101	Embalagens de Papel e Cartão	1200	PT20250326374127	26/03/2025	10/04/2025
10735					
2º Trimestre					
150101	Embalagens de Papel e Cartão	1080	PT20250404068546	04/04/2025	07/04/2025
150101	Embalagens de Papel e Cartão	580	PT20250410161135	10/04/2025	11/04/2025
150101	Embalagens de Papel e Cartão	1260	PT20250515212634	15/05/2025	15/05/2025
150101	Embalagens de Papel e Cartão	680	PT20250523346143	23/05/2025	26/05/2025
150101	Embalagens de Papel e Cartão	300	PT20250612170834	12/06/2025	12/06/2025
150101	Embalagens de Papel e Cartão	880	PT20250627399327	27/06/2025	30/06/2025

Figura 3.17: Página da listagem das informações dos resíduos

A segunda funcionalidade corresponde à criação de uma listagem destinada ao registo das análises realizadas, com o objetivo de substituir a utilização de múltiplos ficheiros Excel e, assim, centralizar toda a informação num único sistema. Adicionalmente, esta funcionalidade possibilita a exportação dos dados para formato PDF, assegurando maior uniformidade, praticidade e eficiência no tratamento e partilha da informação.

A Figura 3.18 apresenta a página principal de listagem das análises, onde é possível visualizar e gerir todos os registos de forma centralizada.

Data do Registro	Designação/Referência	Lote	NIR	Lab.	Requisição	Estado	Observações
24/10/2025		241025.01	Não Aplicável		877/25	Conforme	Q
24/10/2025		201025.08	Não Aplicável		879/25	Pendente	OGM
24/10/2025		231025.10	Não Aplicável		880/25	Pendente	OGM
24/10/2025		231025.09	Não Conforme			Pendente	MGB + CZ
24/10/2025		231025.13	Não Conforme			Pendente	Salmonella
24/10/2025		231025.09	Não Aplicável		881/25	Pendente	F
24/10/2025		241025.02	Não Conforme			Pendente	DNA
23/10/2025		231025.04	Não Aplicável		875/25	Pendente	Q
23/10/2025		231025.08	Não Aplicável		876/25	Conforme	Q
22/10/2025		221025.02	Não Conforme			Pendente	F
22/10/2025		221025.04	Não Conforme			Pendente	MGB
22/10/2025		221025.04	Não Aplicável			Pendente	M
22/10/2025			Não Aplicável		870/25	Conforme	CZ
22/10/2025		221025.05	Não Aplicável		871/25	Pendente	Q + M
22/10/2025		221025.07	Não Aplicável		872/25	Pendente	Q + M
21/10/2025		211025.02	Não Aplicável		861/25	Conforme	
21/10/2025		161025.01	Não Aplicável			Pendente	Salmonella
21/10/2025		211025.07	Não Conforme			Pendente	PB + MGB + CZ + F
21/10/2025		211025.03	Não Aplicável			Pendente	M
21/10/2025		211025.06	Não Aplicável		869/25	Pendente	Q + M

Figura 3.18: Página de listagem das análises

Para uma melhor compreensão da estrutura organizacional dos dados, a Figura 3.19 ilustra o diagrama de matérias primas, demonstrando as relações e hierarquias entre os diferentes elementos do sistema.

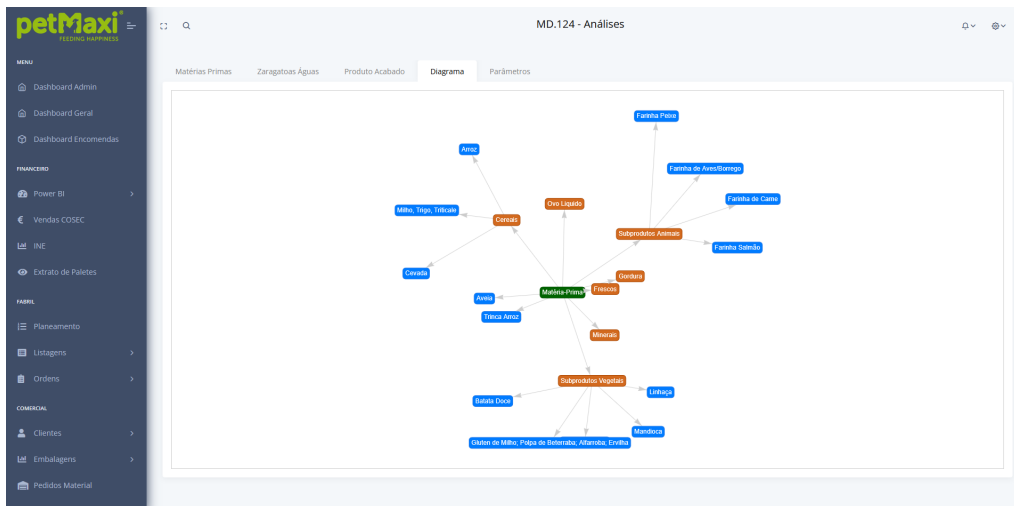


Figura 3.19: Diagrama de matérias primas

Por último, Figura 3.20 detalha os parâmetros associados a cada matéria prima, permitindo um controlo rigoroso e personalizado das características específicas de cada elemento registado no sistema.

Familia	Produto	Parâmetros Químicos	Parâmetros Microbiológicos
Cereais	Milho, Trigo, Triticale	Humidade, Proteína, Celulose, Matéria Gorda, Cinza, Amido	E. Coli, Coliformes, Enterobacterias, Bolores e Leveduras
Cereais	Aveia	Humidade, Proteína, Celulose, Matéria Gorda, Cinza, Amido	E. Coli, Coliformes, Enterobacterias, Bolores e Leveduras
Cereais	Cevada	Humidade, Proteína, Celulose, Cinza, Amido	E. Coli, Coliformes, Enterobacterias, Bolores e Leveduras
Cereais	Arroz	Humidade, Proteína, Cinza, Amido	E. Coli, Coliformes, Enterobacterias, Bolores e Leveduras
Cereais	Trinca de Arroz	Humidade, Proteína, Cinza, Amido	E. Coli, Coliformes, Enterobacterias, Bolores e Leveduras
Digest	Digest	Densidade	-
Frescos	Todos	Humidade, Proteína, Matéria Gorda, Cinza,	Salmonella, Microorganismos a 30°C, Clostridium perfringens, Coliformes, Enterobacteriacæ, E. Coli, Bolores e Leveduras
Gordura	Gordura de Aves	Acidez Oleica, Densidade	-
Gordura	Óleo de Soga	Humidade, Acidez Oleica, Peróxidos, Densidade	-
Gordura	Azeite	Acidez Oleica, Densidade	-

Figura 3.20: Parâmetros de cada matéria prima

Por fim, foi implementada a possibilidade de gerar novos certificados a partir da antiga vista *Requerimento CS e Declaração de Conformidade*, atualmente designada *Documentação de Exportação*, pela mesma razão que motivou a funcionalidade da Documentação a mudar de designação. Entre os documentos contemplados encontram-se o Certificado para a Bielorrússia, o Certificado de Origem, o Anexo de Grupagem e o Requerimento mod1136AA.

3.1.4 Departamento Financeiro

Para este departamento foi iniciado o desenvolvimento de algumas tarefas que até então eram realizadas em Excel, especificamente as funcionalidades Vendas COSEC, INTRA-CH e INTRA-EX. Para a página de Vendas COSEC (Figura 3.21), o objetivo foi criar uma interface intuitiva e eficiente, que permitisse a visualização dos dados de forma rápida e sem necessidade de inserção manual, com a possibilidade de aplicar filtros por ano e mês, além da funcionalidade de exportar os dados para um ficheiro Excel. No caso das páginas do INTRA-CH e INTRA-EX, o objetivo foi semelhante, mas os dados deveriam ser exportados em formato CSV, dado que seriam posteriormente importados para um site que aceita apenas este tipo de ficheiro.

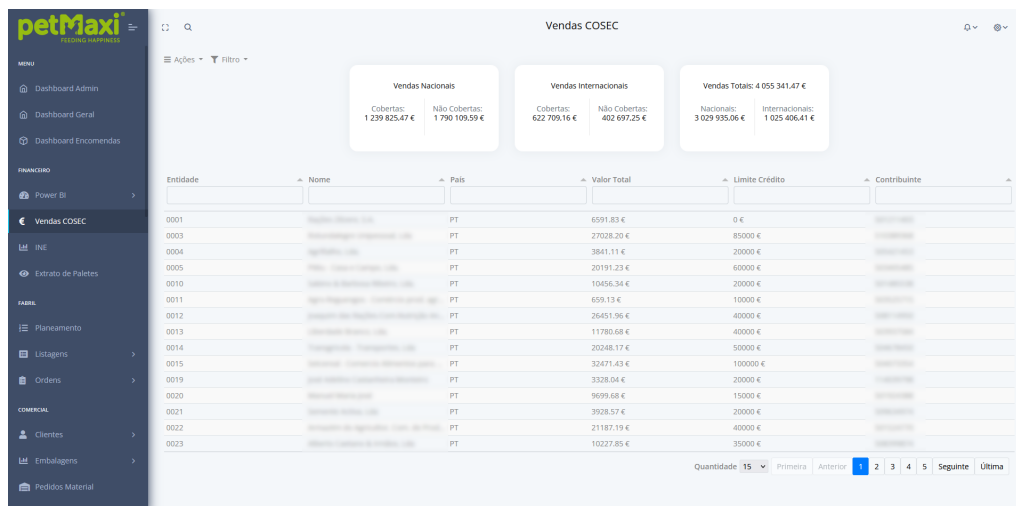


Figura 3.21: Página das Vendas COSEC

3.1.5 Departamento Comercial

No Departamento Comercial, começou por se desenvolver as listagens do stock de sílica e de snacks. Estas funcionalidades foram concebidas com o objetivo de reduzir a utilização de folhas Excel, uma vez que anteriormente era necessário copiar manualmente os dados do ERP Primavera para realizar as análises de stock. Agora, este processo é efetuado de forma automática, garantindo maior eficiência e fiabilidade. Para além disto, a listagem do stock de sílica permite ainda a criação e remoção de simulações, garantindo maior controlo e fiabilidade nas previsões de stock. É possível visualizar esta funcionalidade na Figura 3.22.

Artigo	Descrição	Stock Atual	ECL's	Stock Final	A	B	C	D	E	Observações	F	G
Legenda												
<ul style="list-style-type: none"> A - Consumos Mensais Previstos B - Média Vendas (Mensal) C - Durabilidade Stock Existente tendo em conta média vendas mensal D - Durabilidade Stock Existente tendo em conta consumos mensais previstos E - Encomendas Pendentes (a chegar ou a sair da China) UND F - Durabilidade Stock Existente (média mensal últ. 12 meses) + Encomendas Pendentes G - Durabilidade Stock Existente (Consumos previstos) + Encomendas Pendentes 												

Figura 3.22: Página de visualização do Stock de Sílica

Dando continuidade ao desenvolvimento de ferramentas de gestão de stock, iniciou-se posteriormente a implementação das funcionalidades de Controlo de Stock de Embalagens e de Embalagens de Artigos de Marcas Próprias (Figura 3.23), que permitem a listagem e monitorização dos respetivos níveis de stock, em resposta a uma nova solicitação do Departamento Comercial.

Produto	Stock Atual	Stock Atual + ECL Emb	Média Vendas	Durabilidade Stock	Durabilidade Stock + ECF Emb
...	14450	14695	1680	8,6	8,75
...	15036	15281	320	46,99	47,75
...	14387	14387	942	15,27	15,27
...	5485	12758	1177	4,66	10,84
...	26167	41167	4123	6,35	9,98
...	9854	9854	2423	4,07	4,07
...	29534	29534	5686	5,19	5,19
...	17974	20354	3780	4,76	5,38
...	13766	27402	1044	13,19	26,25

Figura 3.23: Página de visualização dos dados das marcas próprias

De forma a complementar esta funcionalidade e proporcionar uma análise mais intuitiva, foi implementado um gráfico de *Gantt*, concebido para facilitar a visualização da durabilidade das Encomendas de Fornecedor (ECFs) ativas, observável na Figura 3.24.

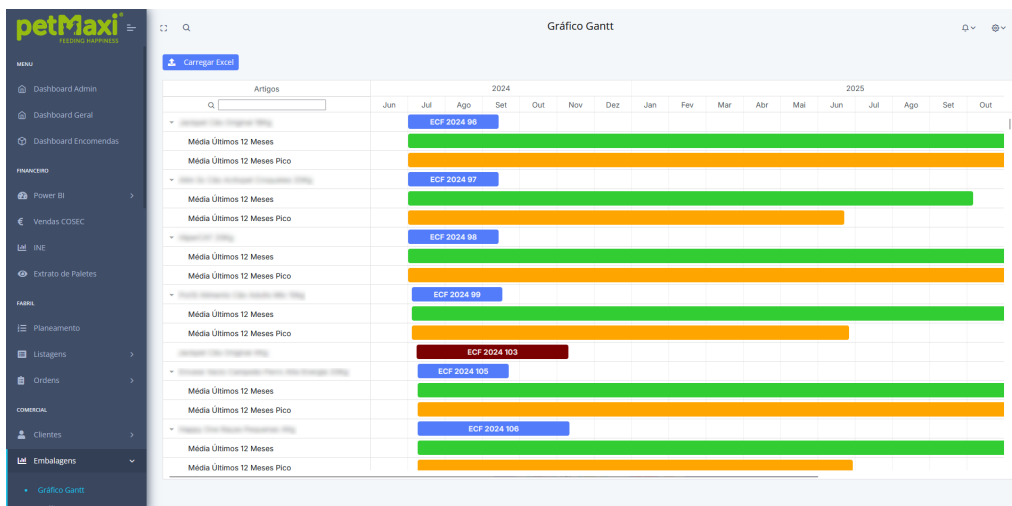


Figura 3.24: Gráfico Gantt de análise da durabilidade das ECFs

À semelhança das funcionalidades anteriores, estas ferramentas foram desenvolvidas com o intuito de reduzir a dependência do Microsoft Excel como principal ferramenta de análise, contribuindo para uma maior eficiência operacional, a diminuição do tempo despendido em tarefas manuais e a libertação de recursos humanos para atividades de maior valor acrescentado. Adicionalmente, foram incorporados filtros dinâmicos, que possibilitam uma análise mais direcionada da informação, bem como a funcionalidade de exportação dos dados para formatos PDF e Excel, assegurando maior versatilidade na utilização e partilha dos resultados.

No âmbito da otimização da gestão comercial, foi criada uma nova vista, Contactos dos Clientes, que centraliza e organiza a informação dos contactos. Desta forma, os dados anteriormente dispersos passaram a estar reunidos num único local, tornando a análise e a gestão desta informação mais eficiente. Seguindo a lógica de personalização já implementada na funcionalidade dos Stocks de Grandes Superfícies, o utilizador pode escolher as colunas apresentadas na tabela, garantindo maior liberdade na definição da informação que pretende visualizar. Para além disto, é disponibilizada a exportação dos dados em PDF e Excel, permitindo que a informação seja facilmente partilha ou trabalhada fora da aplicação. Em complemento à vista de contactos, foi desenvolvido um Mapa de Clientes, que recorre à biblioteca *Leaflet*¹⁰ para a representação geográfica e ao *Leaflet MarkerCluster*¹¹ para o agrupamento dos artigos comprados por cliente em clusters, tornando a navegação mais clara e escalável. Esta funcionalidade, visível na 3.25, permite uma análise espacial dos clientes e dos seus padrões de compra, enriquecendo a capacidade de tomada de decisão do Departamento Comercial.

¹⁰<https://leafletjs.com/>

¹¹<https://github.com/Leaflet/Leaflet.markercluster>

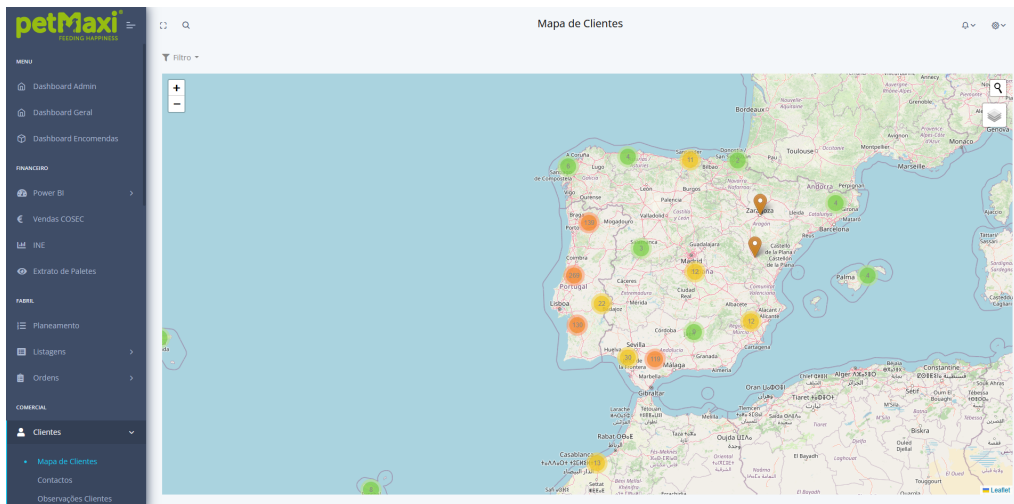


Figura 3.25: Mapa de Clientes

3.1.6 Área fabril

Para a área fabril, procedeu-se à reformulação completa de todas as páginas referentes ao planeamento de ordens de produção e ensaque, com o objetivo de melhorar a usabilidade e a eficiência do sistema.

A Figura 3.26 apresenta a página onde podem ser adicionadas as Ordens de Produção, constituindo o ponto de partida do processo de planeamento.

ID	Fórmula	Data	Linha	Toneladas	Observações
2024-11-13 (2 items)					
109		2024-11-13	1	35	
108		2024-11-13	1	24	
2024-11-14 (3 items)					
111		2024-11-14	1	23	
107		2024-11-14	1	56	
104		2024-11-14	1	3	
2024-11-18 (1 item)					
121		2024-11-18	1	98	
2024-11-19 (1 item)					
122		2024-11-19	1	65	
2024-12-02 (1 item)					
123		2024-12-02	1	98	
2024-12-12 (1 item)					
124		2024-12-12	2	3	oi

Figura 3.26: Planeamento – Ordens de Produção

A partir desta interface, e conforme ilustrado na Figura 3.27, ao clicar numa Ordem de Produção específica, é exibido um *pop-up* que possibilita a criação de Ordens de Ensaque diretamente associadas à ordem selecionada. Esta abordagem garante a rastreabilidade e a ligação direta entre as diferentes fases do processo produtivo.

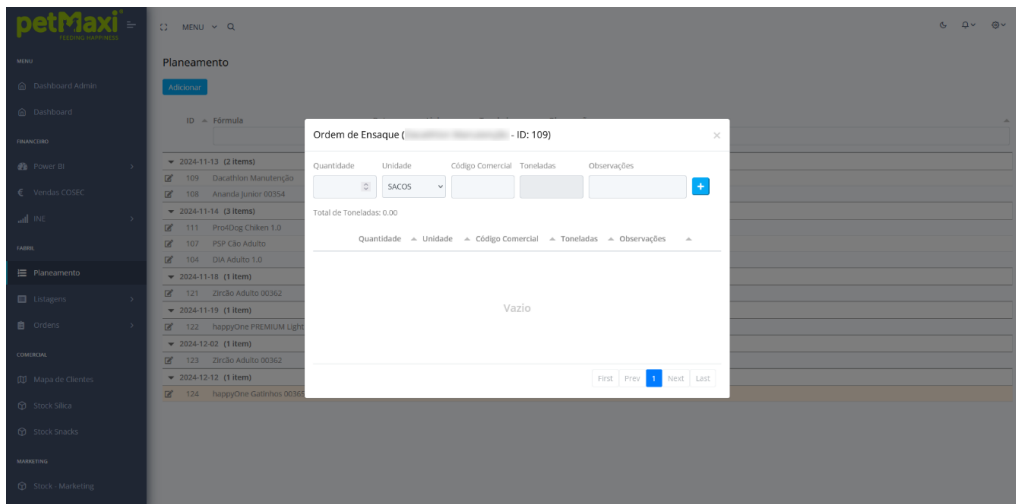


Figura 3.27: Planeamento - Ordens de Ensaque

Para complementar o fluxo de trabalho e facilitar o acompanhamento das tarefas pendentes, foi implementada uma funcionalidade de *To Do*, visível na Figura 3.28 que permite observar de forma clara quais as ordens que ainda estão por concluir e quais as ordens já finalizadas. Embora a figura apresente apenas a página do *To Do* de Ordens de Produção, esta funcionalidade encontra-se igualmente disponível para as Ordens de Ensaque, assegurando uma gestão consistente e integrada de ambos os processos.

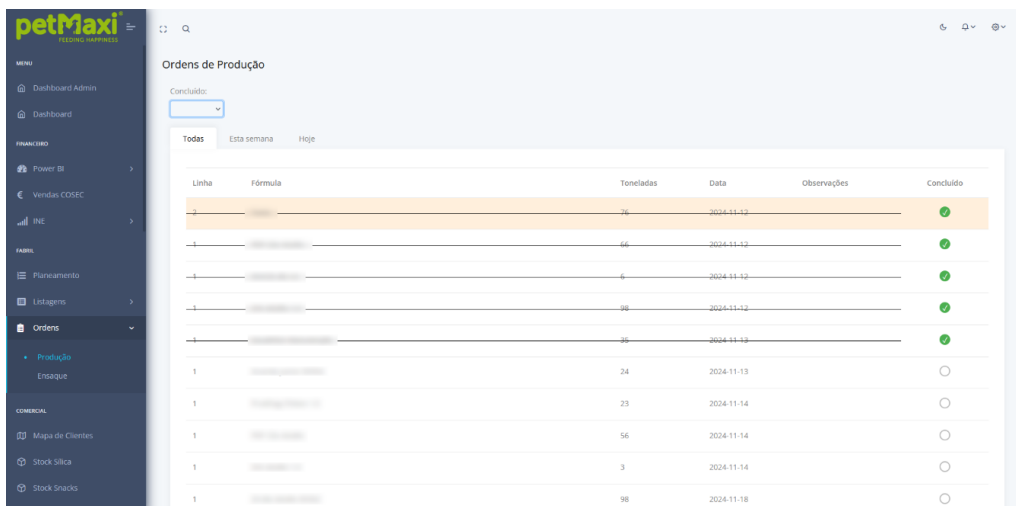


Figura 3.28: *To Do* Ordens de Produção

3.2 Automação com IA

A escolha adequada de um modelo de inteligência artificial constitui um fator determinante para garantir a extração de dados de forma precisa, eficiente e alinhada com as necessidades da empresa. Conforme referido na secção 2.2.3, foi realizada uma análise comparativa de diferentes soluções amplamente reconhecidas, como

o BERT e o *spaCy*, resultando na conclusão de que a alternativa mais adequada seria a utilização do Power Automate em conjunto com o modelo GPT, devidamente integrado nesta plataforma. Para além disso, o Power Automate permite ainda tirar partido de um vasto leque de funcionalidades nativas do ecossistema Microsoft, que se revelaram essenciais ao desenvolvimento do projeto.

Foi então criado um fluxo que é executado sempre que for recebido um novo email na caixa de entrada. Após a receção do email e dos seus dados, é colocada uma tag no mesmo a indicar que este vai ser processado. De seguida, é definida uma variável, *Dados*, que vai armazenar os dados retirados do email, como o nome do cliente, o conteúdo do email, as características dos artigos encontrados, as unidades encomendadas e o peso do artigo, tudo isto em formato JSON. É ainda criada uma variável, *Status*, que define o fim do fluxo, para que sejam evitados erros. Este processo pode ser verificado na Figura 3.29.

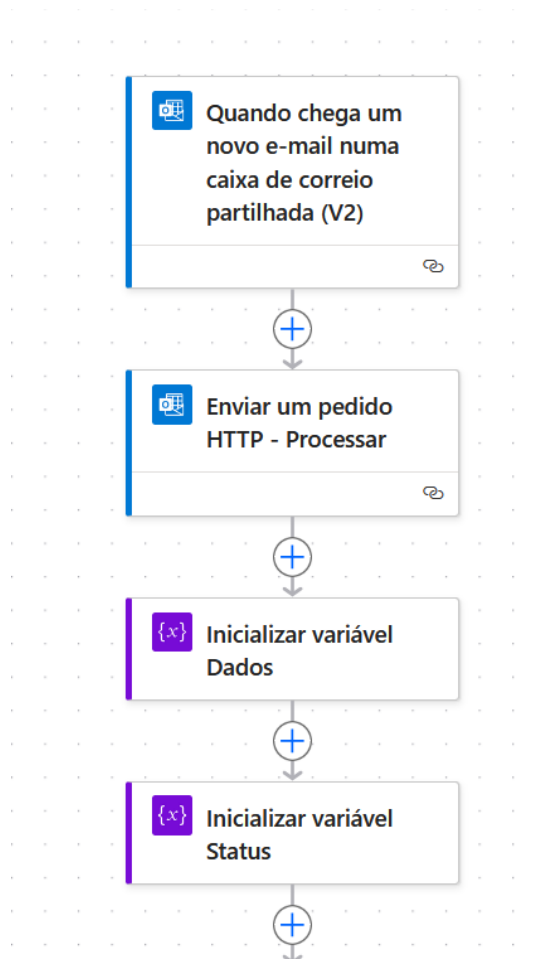


Figura 3.29: Fluxo - Parte Inicial

De seguida é realizada a filtragem pelo tipo de conteúdo que o email pode retornar no fluxo: texto ou anexos. Para cada um destes foi criado um *prompt* diferente especificando necessidades diferentes para cada, como o modelo GPT, utilizando a versão 3.5 para o texto e o 4.0 para os ficheiros. No caso do texto, este é retornado em HTML, pelo que é necessário primeiro haver uma limpeza

do mesmo para depois ser processado pelo modelo GPT, para que as informações pertinentes sejam retiradas. Posteriormente, os dados em formato JSON são guardados na variável *Dados*, como já anteriormente referido, o email recebe uma tag indicando se a extração dos dados foi bem-sucedida ou falhou, de forma a facilitar a monitorização do processo. Por fim, os dados em JSON são enviados para um endpoint intermédio, que atua como ponto de passagem para o destino final. A fase intermédia e a fase final do fluxo podem ser observadas nas Figuras 3.30 e 3.31.

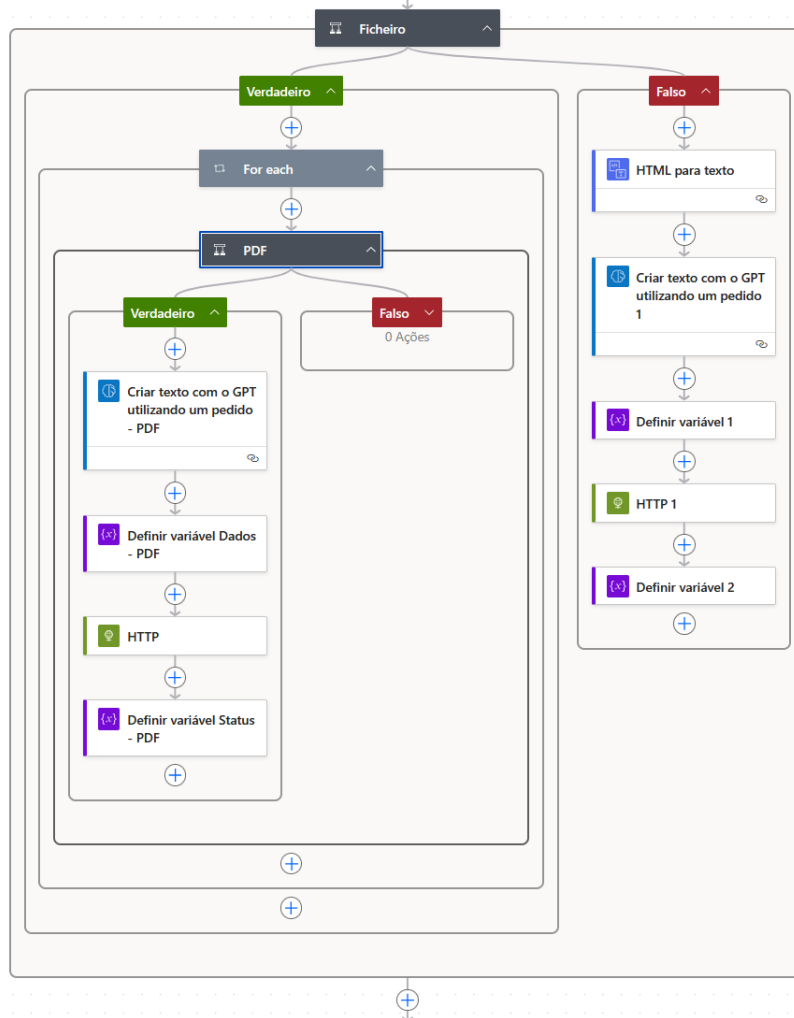


Figura 3.30: Fluxo - Parte Intermédia

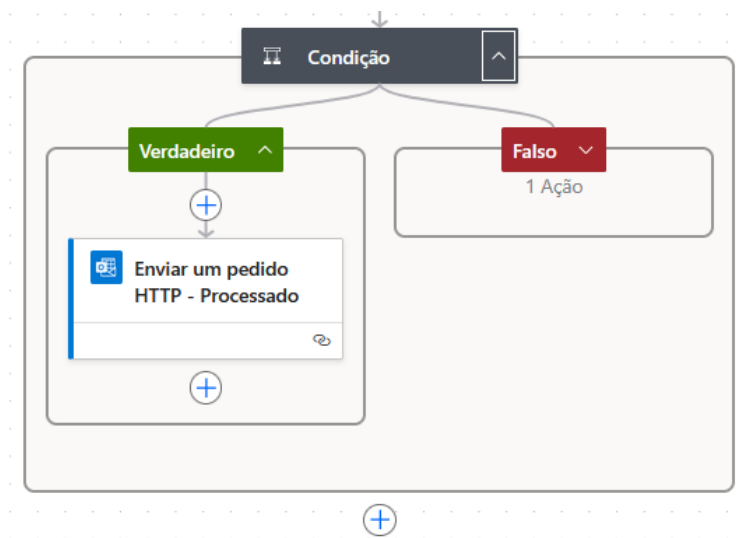


Figura 3.31: Fluxo - Parte Final

Durante o desenvolvimento do fluxo, foram encontradas algumas dificuldades, nomeadamente no *prompt* dado ao modelo GPT para que executasse a extração correta dos dados dos ficheiros. Inicialmente, o *prompt* solicitava que fosse retirado o nome cliente presente no ficheiro, no entanto, devido à presença recorrente de petMaxi nos cabeçalhos do ficheiro, o modelo GPT inferia de forma incorreta que a PM era o cliente. Diante deste cenário, foi necessária a retificação no *prompt*, sendo então este alterado para obter o nome do remetente em vez do nome do cliente, o que solucionou o problema.

Ultrapassadas todas as limitações, deu-se a continuação do desenvolvimento do projeto. Obtidos os dados no endpoint intermédio, estes são guardados numa tabela temporária para posteriormente serem enviados para o endpoint final. Para a obtenção destes, foi criada uma *task* periódica, utilizando a biblioteca *Celery*, que verifica se existem alterações na tabela temporária e, caso existam e estas ainda não tenham sido recebidas, sejam então inseridas no sistema. Antes disso, é necessário primeiramente tratar os dados recebidos. Para isto foi utilizada a biblioteca *TheFuzz*, em conjunto com um sistema de tags. Criou-se uma tabela de tags tanto para artigos quanto para os clientes, onde as descrições dos mesmos foram segmentadas em tags e armazenadas nesta estrutura. Os dados vindos do endpoint intermédio são também processados, normalizados e divididos em tags, permitindo que seja verificada a semelhança entre estas e os registos existentes. Desta forma, é possível identificar no sistema o artigo ou cliente correspondente com base na similaridade calculada segundo os pesos definidos: 70% correspondem à igualdade das tags e 30% à semelhança entre os dados recebidos e as descrições existentes. A inserção no sistema só ocorre quando o nível de similaridade total ultrapassa os 70%. Após a introdução dos dados no sistema, estes são exibidos nas páginas do Dashboard das encomendas, Lista de encomendas e Informações da encomenda.

A página infra, observada na Figura 3.32, mostra o total de encomendas realizadas, encomendas editadas e encomendas por editar. Para além disto, apresenta

também o top 5 de clientes com mais encomendas e o top 5 de artigos mais encomendados. Cada cartão (situado no topo da página) é clicável e redireciona para uma página com a listagem correspondente ao tipo de encomenda apresentado.

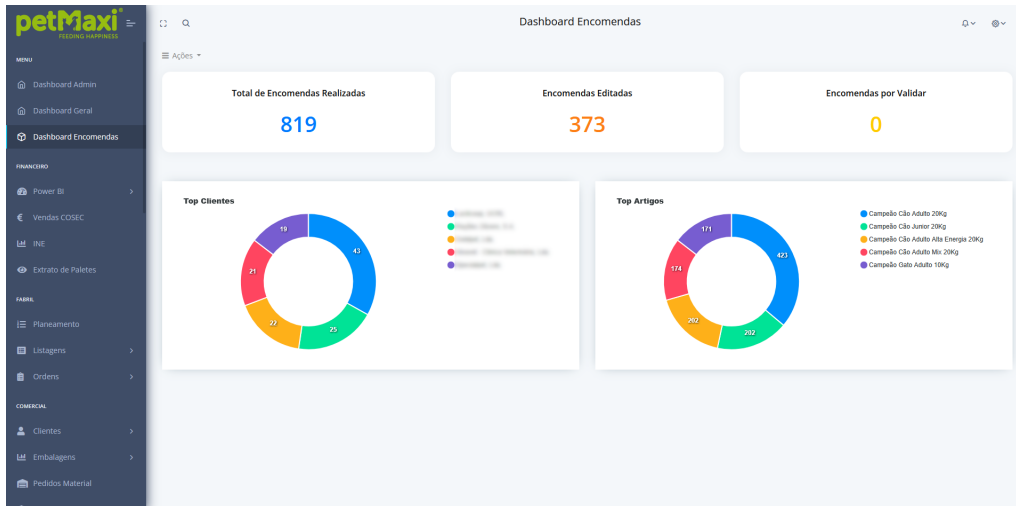


Figura 3.32: Dashboard das encomendas

Adicionalmente, a listagem das encomendas por integrar utiliza *WebSockets* para atualização dos dados em tempo real, garantindo que, sempre que uma encomenda entre no sistema ou o utilizador proceda à sua reentrada, a página seja atualizada automaticamente sem necessidade de recarregamento, visível na Figura 3.33.

Figura 3.33: Listagem de encomendas

Na Figura 3.34 podemos observar a página onde são apresentadas as informações de cada encomenda, sendo possível editar cada um dos campos de input. É também mostrado o conteúdo completo do email recebido, através da utilização dos endpoints do Microsoft Graph, o que possibilita a visualização tanto do corpo

da mensagem como dos seus anexos, facilitando a confirmação da encomenda ao utilizador.

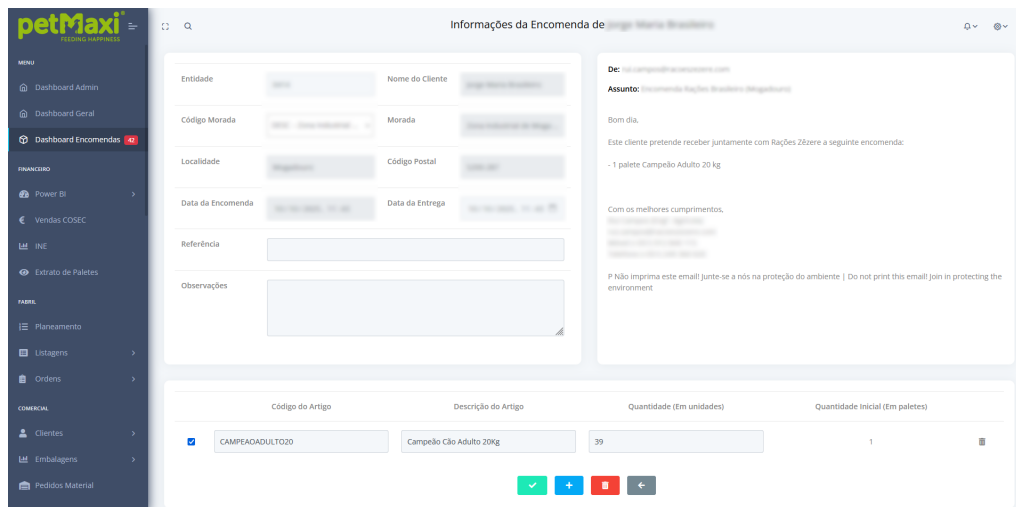


Figura 3.34: Página de confirmação da encomenda

Depois de confirmados os dados inseridos em sistema e verificado que estes coincidem com as informações da encomenda recebida por email, o utilizador pode clicar no botão *check*. Ao fazê-lo, os dados são transmitidos para o ERP Primavera, onde é automaticamente gerada uma encomenda real. Foram também disponibilizadas as opções de adicionar e remover linhas da encomenda, bem como a possibilidade de criar uma encomenda parcial, selecionando apenas algumas das linhas. Desta forma, o utilizador pode dividir uma mesma encomenda em várias, por exemplo, quando um único email contém pedidos destinados a locais diferentes.

Após estas implementações, foram criadas *tasks* no *Celery* para complementar o funcionamento descrito anteriormente. Uma das *task* adiciona tags aos emails das encomendas inseridas com sucesso no ERP Primavera - ou, em alternativa, quando estas são removidas - através do Microsoft Graph. A outra *task* atualiza o número do documento sempre que este é guardado como rascunho, nomeadamente nos casos de bloqueio da encomenda.

Foi ainda adicionada uma funcionalidade que envia automaticamente uma mensagem para um chat do Microsoft Teams sempre que uma encomenda é desbloqueada, facilitando a comunicação entre equipas em tempo real.

3.3 Infraestrutura DevOps

A primeira semana de estágio começou com a criação de um *container* integrando o GLPI, um sistema *open-source* de gestão de equipamentos informáticos e suporte técnico, escrito em PHP e distribuído sob a GNU General Public License, juntamente com uma base de dados mariadb, para guardar os dados.

A necessidade de uso deste software surgiu pois não existia um local centralizado para registar e acompanhar os equipamentos que a empresa possuía. Para além disto, era essencial criar um espaço onde os funcionários pudessem registar os seus tickets, como problemas técnicos, bugs encontrados, pedidos de suporte, entre outras questões. Depois de alguma pesquisa, chegou-se à conclusão de que o GLPI era o software que melhor atendia às nossas necessidades, pois, além de possibilitar uma fácil gestão dos equipamentos, oferece ainda uma plataforma eficiente para tratamento de pedidos dos funcionários.

Para facilitar a criação de tickets por parte dos funcionários foi instalado o plugin *Security Assertion Markup Language* (SAML) no GLPI, que permite a autenticação do utilizador a partir do Microsoft Azure, tornando mais simples a autenticação dos trabalhadores da empresa no sistema. A página inicial do sistema encontra-se na Figura 3.35.

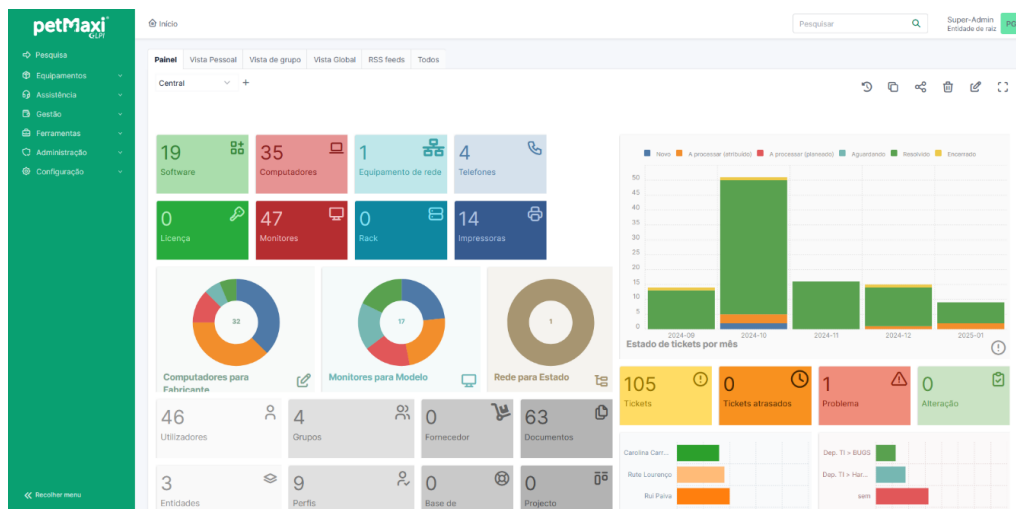


Figura 3.35: Interface do GLPI

Na mesma semana foi também instalado numa máquina virtual, o Netbox, outro sistema *open-source* que permite a inventariação de redes, VMs, entre outros de forma coesa. A instalação deste foi realizada diretamente numa VM e não num *container*, devido à incompatibilidade que alguns plugins apresentam quando dentro de um *container*. Este sistema é demonstrado na Figura 3.36.

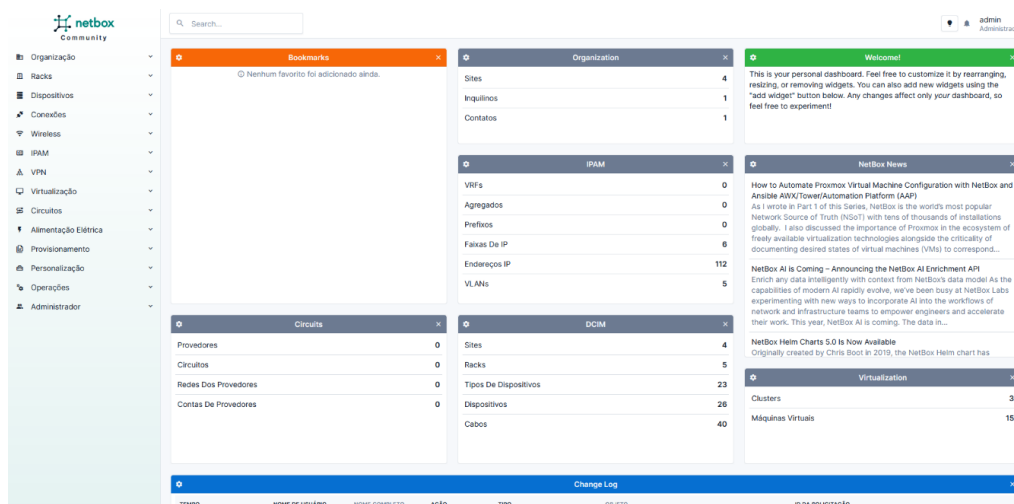


Figura 3.36: Interface do Netbox

Depois da instalação, foi atribuído ao GLPI e ao Netbox um DNS para que fossem facilmente acessíveis pelos funcionários. Para garantir uma ligação segura, foi configurado o protocolo *Hypertext Transfer Protocol Secure* (HTTPS), que exigiu a criação de certificados SSL auto-gerados, carregados posteriormente no servidor que abrange a rede, para que a ligação fosse considerada segura em qualquer local da empresa.

De seguida, foi configurado o *Traefik* para assumir a gestão centralizada de certificados SSL/TLS, garantindo que todas as ligações são estabelecidas de forma segura e simplificando a gestão dos certificados. Para o correto funcionamento, foi necessário disponibilizar os ficheiros dos certificados do servidor a cada serviço configurado no *Traefik*. Desta forma, o *Traefik* não só valida os certificados, como também assegura o redirecionamento para o seu DNS correto, evitando erros associados a certificados baseados apenas no *Common Name* (CN) em vez de *Subject Alternative Names* (SANs).

Adicionalmente, a gestão de DNS dos serviços foi automatizada através de um ficheiro dinâmico: basta editar e guardar o ficheiro com a alteração desejada para que esta seja aplicada imediatamente, sem necessidade de reiniciar o *Traefik* ou interromper os serviços.

Foi ainda criada uma página de manutenção, apresentada sempre que algum serviço se encontra indisponível (Figura 3.37). Esta abordagem melhora a experiência do utilizador e garante uma comunicação mais clara em períodos de paragem planeada ou inesperada.

Voltamos em breve!

Pedimos desculpa pelo inconveniente, mas estamos a realizar uma manutenção neste momento. Se precisar, pode sempre [entrar em contacto conosco](#), caso co>

— Departamento de IT



Figura 3.37: Página de manutenção

Posteriormente, foi desenvolvida uma API recorrendo ao *Flask*¹², uma *framework* web em *Python*, e ao *Swagger*¹³, um conjunto de ferramentas *open-source* para desenvolvimento de APIs. Após a sua implementação, a API foi integrada num *container* juntamente com o servidor *Redis*, permitindo uma maior escalabilidade e facilidade de gestão. O objetivo principal da criação desta API consistiu em suportar as queries utilizadas em ficheiros Excel, de forma a facilitar a interação com a base de dados. Tradicionalmente, seria necessário que o utilizador alterasse manualmente a query SQL sempre que pretendesse, por exemplo, extrair dados relativos a uma data distinta. Com a API, esta operação passou a ser realizada através de parâmetros fornecidos no URL, simplificando o processo de extração de dados (Figura 3.38). Desta forma, a solução reduz a complexidade para utilizadores sem conhecimentos avançados em SQL, garantindo um acesso mais intuitivo e eficiente à informação, ao mesmo tempo que promove uma maior flexibilidade e reutilização das consultas.

¹²<https://flask.palletsprojects.com/en/stable/>

¹³<https://swagger.io/>



Figura 3.38: API da petMaxi

Esta página foi intencionalmente deixada em branco.

Capítulo 4

Análise de Impacto e Reflexões

O presente capítulo tem como propósito analisar e interpretar os dados recolhidos de um questionário enviado para os colaboradores, afim de avaliar o desempenho do estagiário no contexto organizacional e obter uma melhor percepção dos colaboradores relativamente à usabilidade da plataforma digital, NETPET, desenvolvida pelo mesmo. Para além desta análise, serão também abordadas as diferenças observadas entre os períodos anteriores e posteriores à intervenção, nomeadamente no que respeita ao modo de execução das tarefas e ao tempo despendido na sua realização por parte dos colaboradores, permitindo compreender se as alterações introduzidas resultaram em processos mais eficientes ou não.

4.1 Avaliação do Desempenho do Estagiário

A primeira parte do questionário teve como objetivo avaliar o comportamento e o desempenho do estagiário a nível profissional, considerando critérios como a clareza na comunicação, a iniciativa e proatividade, a contribuição para a equipa, a abertura ao feedback e a capacidade de resolução de problemas.

Numa perspectiva geral, as respostas obtidas demonstram uma avaliação positiva em todas as categorias mencionadas. A maioria dos inquiridos atribuiu classificações entre os 4 e 5 pontos (numa escala de 1 a 5).

Relativamente à clareza na comunicação, os resultados revelam que 23,5% dos inquiridos atribuíram a classificação de 4 pontos e 76,5% classificaram com a pontuação máxima de 5 pontos, como se pode observar na Figura 4.1.

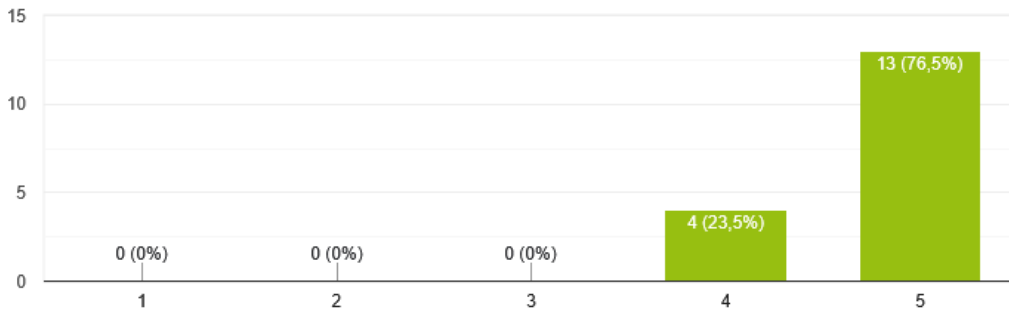


Figura 4.1: Clareza de Comunicação

No que diz respeito à iniciativa e proatividade, os valores obtidos foram igualmente positivos, com 17,6% dos respondentes a atribuírem 4 pontos e 82,4% a atribuírem a classificação máxima de 5 pontos, conforme ilustrado na Figura 4.2.

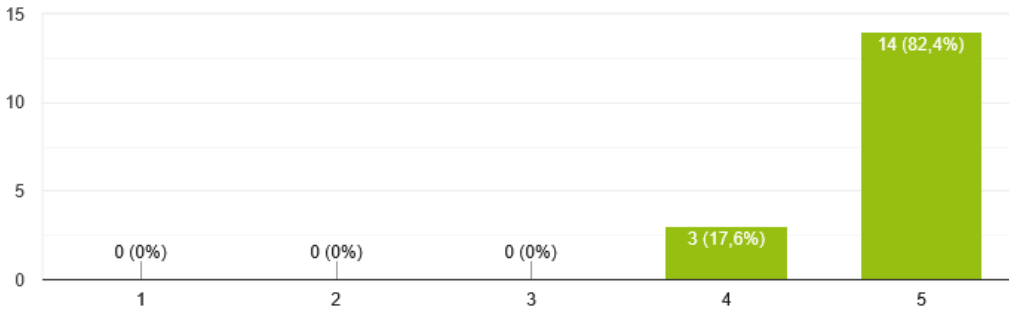


Figura 4.2: Iniciativa e Proatividade

Quanto à contribuição positiva para a equipa, verificou-se uma distribuição semelhante, sendo que 11,8% classificaram com 4 pontos e 88,2% com 5 pontos, tal como apresentado na Figura 4.3.

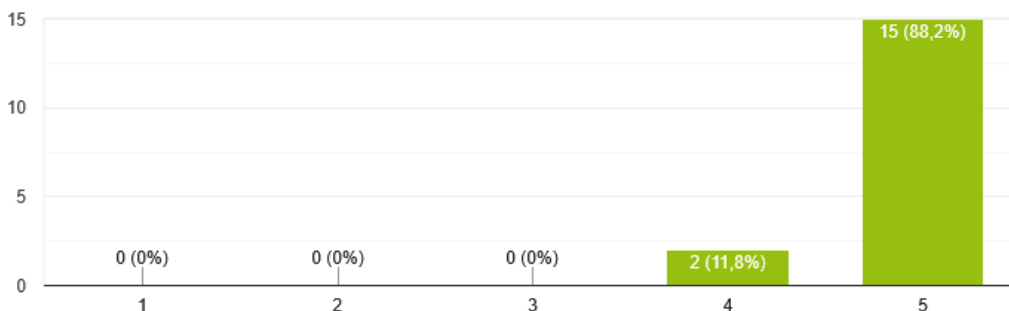


Figura 4.3: Contribuição positiva para a equipa

A abertura ao feedback apresentou resultados ligeiramente diferentes, com 5,9% dos inquiridos a atribuírem 4 pontos e 94,1% a classificarem com 5 pontos, como demonstra a Figura 4.4.

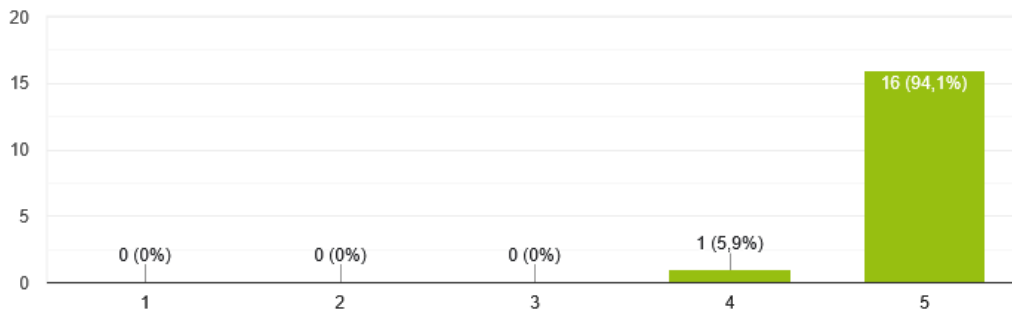


Figura 4.4: Abertura a feedback

Por fim, a capacidade de resolver problemas e encontrar soluções foi igualmente bem avaliada, com 5,9% a atribuírem 4 pontos e 94,1% a pontuarem com 5 pontos, conforme se verifica na Figura 4.5.

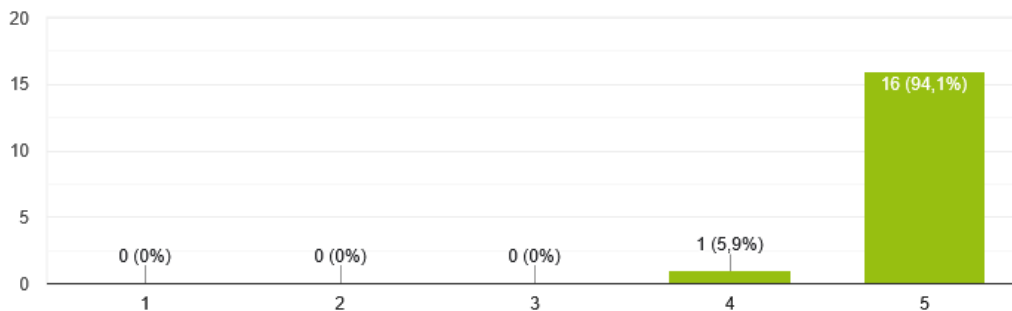


Figura 4.5: Capacidade de resolução de problemas

Nos campos abertos do questionário, os colaboradores destacaram como principais pontos fortes a capacidade de inovação, a disponibilidade para ajudar e a rapidez na execução de tarefas, indicando um alinhamento com as necessidades e dinâmicas de trabalho da equipa. Por outro lado, não foram apontadas áreas de melhoria significativas, o que sugere que o desempenho do estagiário foi globalmente satisfatório e a integração do mesmo foi bem-sucedida no contexto organizacional.

4.2 Avaliação do NETPET

A segunda parte do questionário focou-se na avaliação da plataforma digital utilizada pelos colaboradores, desenvolvida durante o estágio, através da aplicação do questionário SUS (System Usability Scale), sendo o grau de concordância medido numa escala de Likert de 1 a 5, em que 1 corresponde a "discordo totalmente" e 5 a "concordo totalmente", com o intuito de compreender a perceção dos mesmos quanto à sua usabilidade, complexidade e integração funcional.

A maioria dos participantes indicou que utiliza dispositivos tecnológicos diariamente, como telemóveis e computadores, conforme ilustrado na Figura 4.6.



Figura 4.6: Frequência de utilização de dispositivos tecnológicos

Este padrão de uso diário demonstra um perfil de utilizador digitalmente competente, o que se reflete também na autoavaliação das suas competências digitais (Figura 4.7).

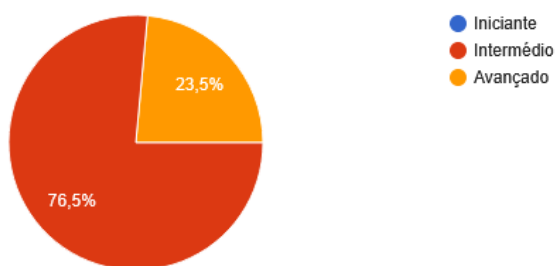


Figura 4.7: Nível de competência digital

Quanto à utilização da plataforma NETPET, os resultados demonstram que os colaboradores recorrem à mesma de forma regular, como evidenciado na Figura 4.8.

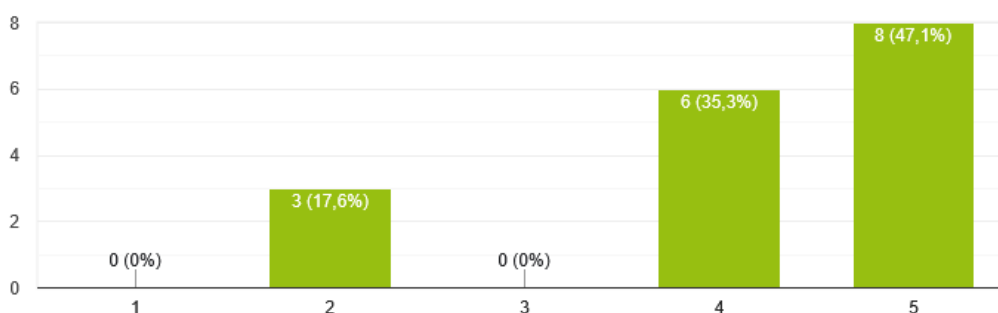


Figura 4.8: Frequência de utilização da plataforma

Relativamente à sua complexidade, as classificações obtidas foram baixas (valores entre 1 e 2), reforçando a ideia de que os utilizadores não têm grandes dificuldades de navegação na plataforma (Figura 4.9).

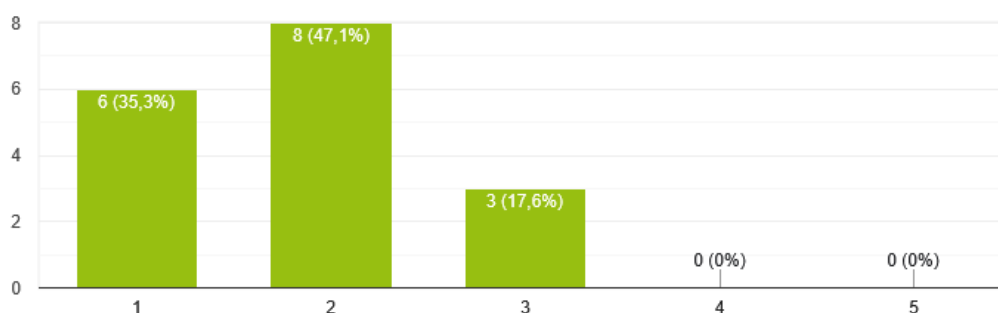


Figura 4.9: Percepção de complexidade da plataforma

A facilidade de utilização foi um dos aspetos mais valorizados, com classificações médias situadas entre 4 e 5 pontos, indicando que a plataforma é, na percepção dos inquiridos, intuitiva e acessível, conforme se observa na Figura 4.10.

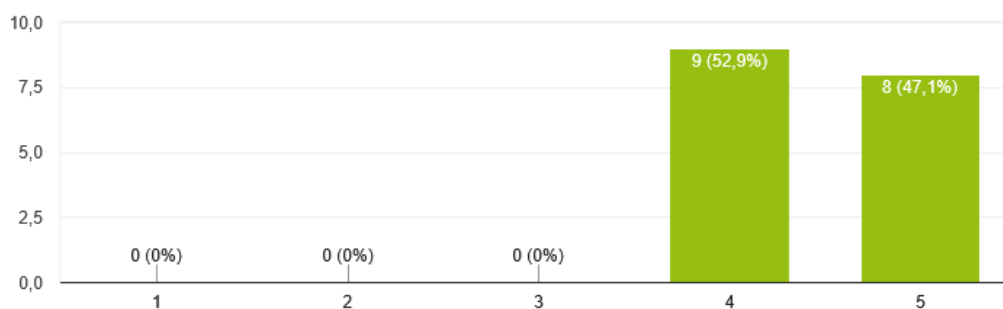


Figura 4.10: Avaliação da facilidade de uso da plataforma

No que diz respeito à necessidade de apoio técnico para a utilização da mesma, a maioria dos inquiridos assinalou valores baixos (entre 1 e 2), o que sugere que o sistema é suficientemente autoexplicativo (Figura 4.11).

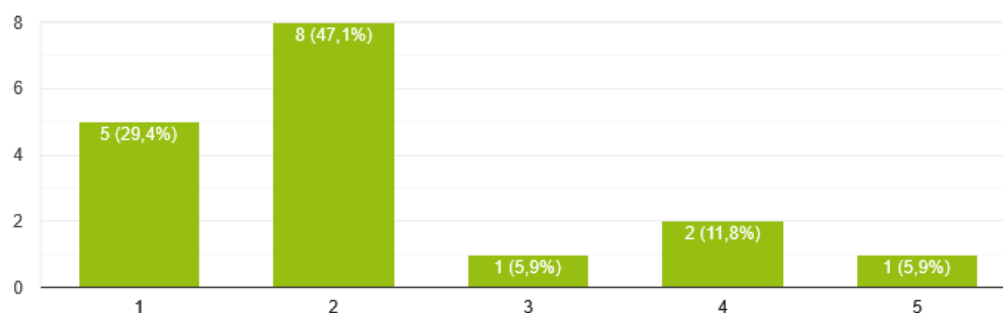


Figura 4.11: Necessidade de apoio técnico para utilização da plataforma

Os resultados relativos à integração de funcionalidades obtiveram classificações altas, demonstrando que as diferentes funcionalidades da plataforma funcionam de forma coesa, como ilustrado na Figura 4.12.

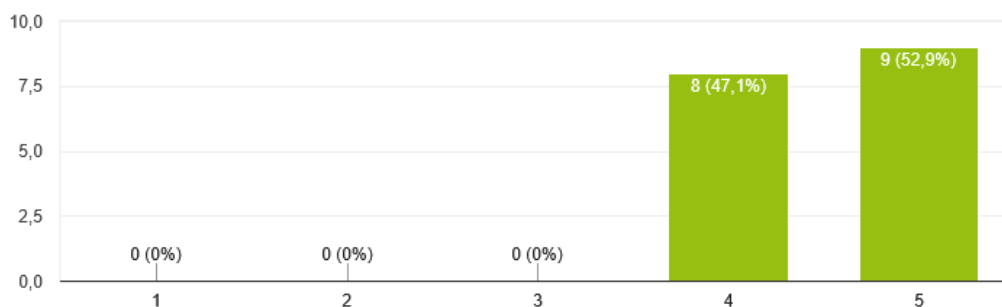


Figura 4.12: Integração das funcionalidades da plataforma

A afirmação sobre inconsistências na plataforma obteve classificações baixas (valores entre 1 e 2), indicando um design coeso e um funcionamento consistente do sistema (Figura 4.13).

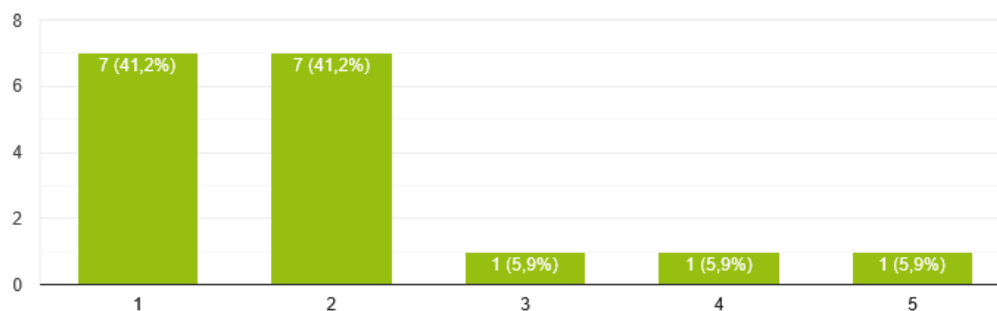


Figura 4.13: Percepção de inconsistência na plataforma

Os participantes consideraram que a maioria das pessoas conseguiria aprender a utilizar a plataforma rapidamente, conforme demonstrado na Figura 4.14.

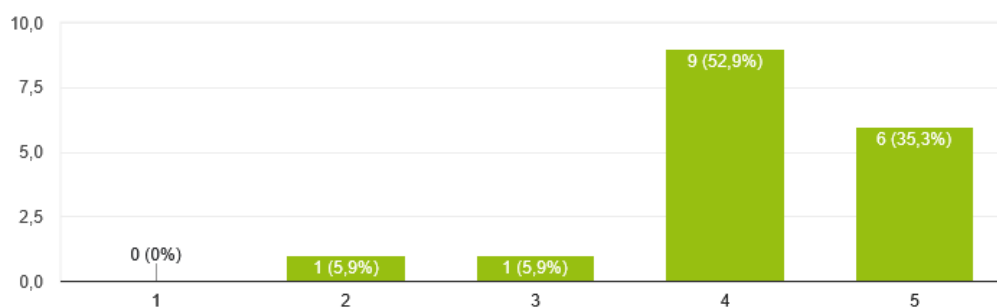


Figura 4.14: Facilidade de aprendizagem da plataforma

Quanto à velocidade da plataforma, os valores obtidos foram baixos, indicando que os utilizadores não sentem lentidão significativa enquanto navegam na mesma (Figura 4.15).

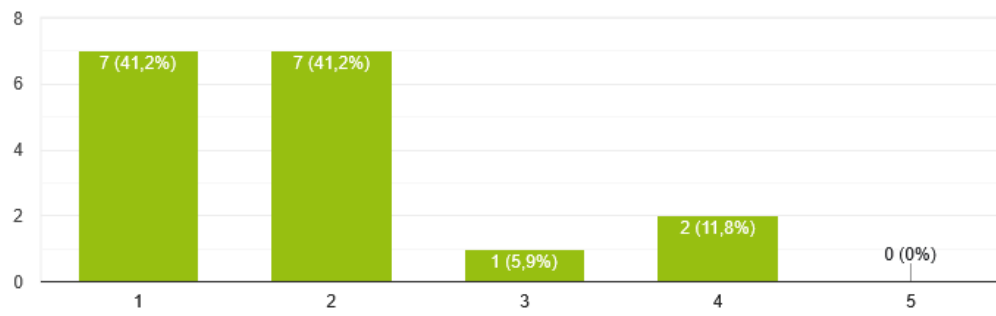


Figura 4.15: Percepção de lentidão da plataforma

A afirmação "senti-me confiante a utilizar a plataforma" obteve classificações elevadas, refletindo um alto nível de satisfação e autonomia digital, como se pode observar na Figura 4.16.

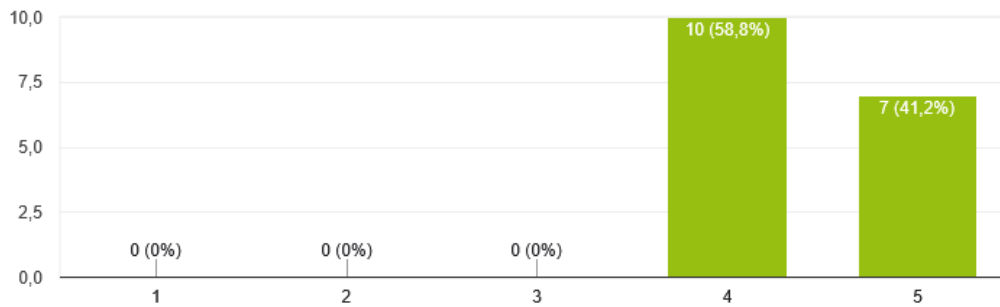


Figura 4.16: Confiança na utilização da plataforma

Relativamente à necessidade de adquirir conhecimentos prévios, os valores obtidos variaram entre o intervalo de 1 a 3, sugerindo que, embora alguns utilizadores tenham sentido necessidade de alguma familiarização inicial, esta não constitui uma barreira significativa à utilização da plataforma (Figura 4.17).

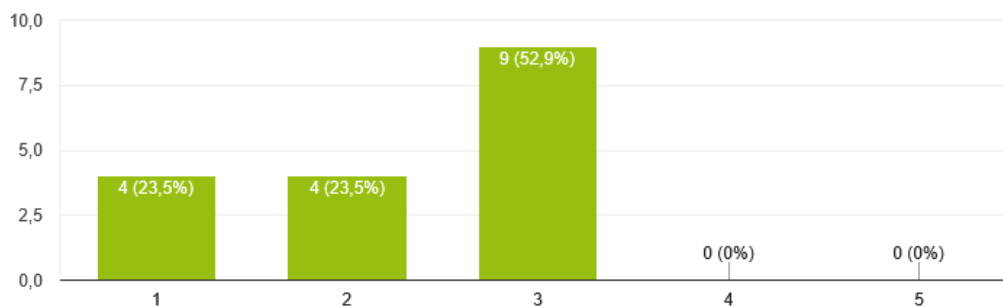


Figura 4.17: Necessidade de conhecimentos prévios para utilização adequada

Em suma, os dados permitem concluir que a plataforma apresenta bons níveis de usabilidade, consistência e integração, cumprindo adequadamente as suas funções e proporcionando uma experiência de utilização positiva.

4.3 Impacto na Duração dos Processos

Os dados apresentados e analisados nesta secção referem-se exclusivamente ao Departamento da Qualidade, escolhido para a recolha de informação e avaliação do impacto da implementação do sistema NETPET, por ser o setor que anteriormente apresentava algumas das tarefas mais demoradas.

A Tabela 4.1 apresenta a variação no tempo de execução dos pedidos de certificados sanitários para diferentes países, comparando os métodos anteriores com a plataforma digital.

Destino	Método antigo	Método NETPET	Ganho de Velocidade (%)
Angola	13m 34s	8m 3s	69%
Bielorrússia	25m 26s	8m 53s	186%
Marrocos	14m 21s	9m 7s	57%
Venezuela	14m 22s	4m 31s	218%
Grupagem	17m 8s	9m 2s	90%
Média	16m 58s	7m 55s	124%

Tabela 4.1: Comparação entre o método antigo e o método NETPET por destino.

Os resultados mostram que o tempo médio necessário para concluir cada pedido diminuiu significativamente em todos os destinos analisados. No método antigo, o tempo variava entre 13 minutos e 34 segundos e 25 minutos e 26 segundos, enquanto que com o NETPET os valores situam-se entre 4 minutos e 31 segundos e 9 minutos e 7 segundos, o que representa uma redução média global de cerca de 52%.

De uma forma mais detalhada, observamos que a maior redução de tempo ocorreu nos pedidos destinados à Bielorrússia e à Venezuela, com reduções de aproximadamente 65% e 69%, respetivamente. No caso da Bielorrússia, o tempo passou de 25 minutos e 26 segundos para apenas 8 minutos e 53 segundos, enquanto que na Venezuela houve uma redução de 14 minutos e 22 segundos para 4 minutos e 31 segundos.

Nos restantes destinos, Angola, Marrocos e Grupagem, também se verificaram melhorias consistentes, com reduções de tempo entre 36% e 47%. Mesmo nestes casos, em que os procedimentos já eram relativamente mais simples, o NETPET contribuiu para otimizar o fluxo de trabalho, eliminando intervenções manuais e facilitando a execução das tarefas.

Para obter uma visão mais abrangente do impacto da plataforma, foi realizada uma avaliação adicional centrada na frequência de emissão dos certificados sanitários. O objetivo foi estimar o impacto global em termos de tempo de trabalho poupado num período operacional típico.

A Tabela 4.2 apresenta um resumo desta análise, evidenciando o tempo poupado por processo, o número de emissões diárias e o tempo total acumulado semanalmente para diferentes tipos de certificados.

Destino	Tempo poupado	Processos/dia	Dias	Tempo total acumulado
Angola	5m 31s	2	5	55m 10s
Bielorrússia	16m 33s	1	3	49m 39s
Marrocos	5m 14s	1	5	26m 10s
Venezuela	9m 51s	1	5	49m 15s

Tabela 4.2: Análise do Impacto Semanal por destino

Com base nestes dados, estima-se que o impacto global semanal da utilização do NETPET represente uma poupança de cerca de 3 horas de trabalho, o que se traduz em aproximadamente 12 horas mensais de tempo recuperado pela equipa da Qualidade.

Ainda que a análise apresentada não inclua o caso da grupagem, devido à dificuldade em quantificar com precisão a sua frequência semanal, é razoável supor que terá uma poupança semanal semelhante aos outros certificados.

É também importante destacar que os resultados apresentados refletem apenas o impacto de uma funcionalidade específica da plataforma. O NETPET automatiza uma variedade de processos tanto do departamento da Qualidade como de outros departamentos, conforme descrito anteriormente. Se considerarmos a totalidade das funcionalidades implementadas e o efeito cumulativo da automação em múltiplas tarefas diárias, é expectável que a poupança total de tempo seja consideravelmente superior às 12 horas mensais estimadas apenas para os certificados sanitários.

Desta forma, o NETPET não só reduz significativamente o tempo de execução de processos individuais, como também aumenta a eficiência operacional, libertando recursos humanos para tarefas de maior valor.

Esta página foi intencionalmente deixada em branco.

Capítulo 5

Conclusão

Este capítulo apresenta uma síntese do trabalho desenvolvido durante o estágio na petMaxi, destacando os resultados obtidos nos diferentes projetos e as oportunidades de desenvolvimento futuro que poderão contribuir para a evolução da plataforma NETPET.

5.1 Síntese do Trabalho Desenvolvido

O estágio na PM resultou numa abordagem multi-projeto que excedeu significativamente a proposta inicial. A plataforma NETPET foi expandida com novas funcionalidades para todos os departamentos empresariais. O projeto de Automação com IA foi implementado através de uma solução diferente da planeada (Power Automate + GPT), mais adaptada às necessidades reais, sendo depois incorporado no próprio NETPET. A Infraestrutura DevOps foi modernizada garantindo estabilidade e escalabilidade da empresa.

Relativamente ao NETPET, a plataforma foi expandida para múltiplos departamentos da empresa, nomeadamente Logística, Qualidade, Financeiro, Comercial e Área Fabril, através da implementação de funcionalidades transversais como *WebSockets* para atualizações em tempo real, sistema de notificações, e automação de processos anteriormente manuais, complementadas por funcionalidades específicas de cada departamento. Os resultados da avaliação de usabilidade demonstraram uma adesão bem-sucedida por parte dos colaboradores, com classificações elevadas em todos os critérios analisados pelo questionário SUS.

A análise de impacto realizada numa das funcionalidades do Departamento da Qualidade revelou resultados significativos, com reduções de tempo entre 36% e 68% em processos específicos deste departamento. Esta otimização representa uma poupança estimada de aproximadamente 12 horas mensais apenas nesta funcionalidade específica. O NETPET consolidou-se assim como um sistema central e fulcral para as operações da empresa.

O projeto de Automação com IA, utilizando Power Automate em conjunto com

modelos GPT 3.5/4, automatizou significativamente o processo de gestão de encomendas recebidas por email. A implementação de técnicas de *fuzzy matching* e a criação de um sistema de *tags*, aliadas à IA, eliminou quase por completo a necessidade de inserção manual das encomendas recebidas por email no ERP Primavera. Esta funcionalidade não só reduziu drasticamente a ocorrência de erros humanos, como libertou recursos humanos para outras tarefas de maior importância.

No âmbito da infraestrutura DevOps, a implementação de ferramentas como *Docker*, *Traefik*, *Redis* e *Celery* garantiu maior estabilidade, escalabilidade e disponibilidade dos serviços. A gestão centralizada de certificados SSL, a criação de uma API *REST* com *Flask* e *Swagger*, e a implementação de sistemas de monitorização como o *Uptime Kuma* contribuíram para uma infraestrutura mais robusta e eficiente.

Do ponto de vista académico, este estágio permitiu consolidar competências multidisciplinares nas áreas de Engenharia de Software, Desenvolvimento Web, DevOps e Gestão de Projetos. A capacidade de adaptação contínua a novos requisitos, a gestão de múltiplos projetos em simultâneo e a capacidade de cumprimento de prazos de entrega constituíram aprendizagens valiosas que transcendem o âmbito específico deste estágio.

Conclui-se que a combinação de tecnologias modernas de Desenvolvimento Web, IA e práticas DevOps pode alterar substancialmente a eficiência operacional de uma organização. Os resultados obtidos confirmam a importância do investimento em automação e digitalização de processos, contribuindo não apenas para o aumento de produtividade, como também para a melhoria qualitativa das condições de trabalho dos colaboradores, libertando-os de tarefas repetitivas e monótonas. A experiência adquirida durante este estágio constitui uma base sólida para o desenvolvimento profissional futuro, reforçando a importância da formação académica como suporte para a inovação e transformação digital nas organizações.

5.2 Trabalho Futuro

Apesar dos resultados positivos alcançados durante este estágio, foram identificadas diversas oportunidades de melhoria e desenvolvimento futuro que poderão melhorar a plataforma NETPET.

Uma destas oportunidades consiste na sua migração de Bootstrap 4 para Bootstrap 5. Esta atualização permitirá não apenas alinhá-la com uma *framework* mais moderna e suportada, como também tirar partido das melhorias significativas de performance e novos componentes disponibilizados pela versão mais recente.

Simultaneamente, será realizada uma otimização e refatoração do código existente. Esta refatoração contribuirá para uma melhor manutenção e crescimento futuro, eliminando redundâncias e melhorando a estrutura do código, tornando-a mais consistente.

Por fim, está previsto uma modernização visual da interface da plataforma,

tornando-a mais intuitiva e responsiva, potenciando assim a produtividade e satisfação dos colaboradores que a utilizam diariamente.

Esta página foi intencionalmente deixada em branco.

Referências

Amazon Web Services. (2025). *What is gpt ai? - generative pre-trained transformers explained*. Retrieved from <https://aws.amazon.com/what-is/gpt/> (Consultado a 20 de janeiro de 2025)

Django Software Foundation. (2025a). *Django channels documentation*. Retrieved from <https://channels.readthedocs.io/en/latest/> (Consultado a 02 de fevereiro de 2025)

Django Software Foundation. (2025b). *Django: The web framework for perfectionists with deadlines*. Retrieved from <https://www.djangoproject.com/> (Consultado a 11 de janeiro de 2025)

Docker Inc. (2025). *Docker overview*. Retrieved from <https://docs.docker.com/get-started/docker-overview/> (Consultado a 15 de janeiro de 2025)

Explosion AI. (2025). *spacy 101: Everything you need to know*. Retrieved from <https://spacy.io/usage/spacy-101> (Consultado a 23 de janeiro de 2025)

GLPI Project. (2025). *Glpi - the most complete open source service management software*. Retrieved from <https://glpi-project.org/> (Consultado a 17 de janeiro de 2025)

Google Cloud. (2025). *What is gpt and how does it work?* Retrieved from <https://cloud.google.com/discover/what-is-gpt> (Consultado a 20 de janeiro de 2025)

Hugging Face. (2025). *Bert 101 - state of the art nlp model explained*. Retrieved from <https://huggingface.co/blog/bert-101> (Consultado a 22 de janeiro de 2025)

NetBox Labs. (2025). *Netbox - the premier network source of truth*. Retrieved from <https://netboxlabs.com/docs/netbox/en/stable/> (Consultado a 17 de janeiro de 2025)

Petmaxi. (2025). *Fábrica portuguesa de ração para cães e gatos*. Retrieved from <https://www.petmaxi.pt/> (Consultado a 03 de janeiro de 2025)

Python Software Foundation. (2025). *What is python? executive summary*. Retrieved from <https://www.python.org/doc/essays/blurb/> (Consultado a 12 de janeiro de 2025)

Esta página foi intencionalmente deixada em branco.

Esta página foi intencionalmente deixada em branco.

