



Aplicação Web para Criação e Gestão de Encomendas de Produtos

ALCIDES MANUEL NUNES TEIXEIRA

novembro de 2018

ISEP

Instituto Superior de Engenharia do Porto

Aplicação Web para Criação e Gestão de Encomendas de Produtos

Tese de Mestrado

Para obter o grau de mestre no
Instituto Superior de Engenharia do Porto,
a defender em Novembro por

Alcides Manuel Nunes Teixeira

Mestrado em Engenharia Electrotécnica e de Computadores,
Automação e Sistemas
Porto, Portugal.

Orientadores da Empresa:

Paulo Matos paulo@indmei.pt

Manuel Silva mvieirasilva@sapo.pt

Orientador do ISEP:

Eng. Carlos Campos crc@isep.ipp.pt

Copyright © 2018

Relatório elaborado para satisfação parcial dos requisitos da Unidade Curricular de Tese/Dissertação do Mestrado em Engenharia Electrotécnica e de Computadores

Email do Autor: 1090349@isep.ipp.pt

Agradecimentos

Em primeiro lugar, gostaria de deixar uma palavra de agradecimento ao Instituto Superior de Engenharia do Porto, por me ter proporcionado todas as condições para desenvolver as minhas competências técnicas e humanas. Foi este Instituto que me viu crescer e evoluir, e onde encontrei excelentes profissionais e amigos que nunca irei esquecer.

Agradecer ao meu orientador, Eng Carlos Campos pelo projeto proposto, e por todo o acompanhamento dado ao longo do período em que este esteve a ser desenvolvido, sempre procurando ter feedback do seu desenvolvimento e nunca deixando uma dúvida por responder.

Quero deixar um agradecimento à empresa para a qual o projeto foi realizado, a INDMEI, em especial ao gestor do departamento de produção, Sr. Paulo Matos e ao colaborador Sr. Manuel Silva por toda a disponibilidade e por todos os esclarecimentos especialmente no que respeita ao funcionamento da empresa.

Gostaria de deixar uma nota de agradecimento a toda a minha família, em especial aos meus pais e ao meu irmão (também ele aluno desta nobre instituição) por todo o apoio e toda a compreensão ao longo de toda a minha vida, especialmente na vida académica.

Finalizo, deixando um agradecimento especial à minha namorada, Cátia Rocha, pela companheira que tem sido ao longo de todo este percurso, e onde pude sempre encontrar a motivação para atingir todos os meus objetivos.

Resumo

A gestão de *stocks* e de encomendas tem sido uma necessidade constante ao longo dos tempos. Trata-se de um método que permite gerir um determinado produto de forma a conseguir saber exatamente qual a sua quantidade em determinado momento. À medida que os tempos foram avançando e com a chegada dos computadores às grandes indústrias, logo se verificou uma tendência em tentar otimizar estes processos, de modo a auxiliar na boa gestão e no aumento da produtividade das empresas.

O projeto desenvolvido vem no sentido de, efetivamente, resolver este problema na empresa INDMEI, empresa responsável pela produção de meias, orientada para a exportação das encomendas, especialmente para países nórdicos. A empresa que, até então, não possuía nenhum mecanismo automático de gestão do armazém e na qual o processo de produção de uma encomenda se inicia com uma deslocação ao armazém, para realizar um levantamento em papel da quantidade de *stock* existente. Também a comunicação entre colaboradores não seria a ideal, dado que não existia um canal único para o tratamento de informação sensível à criação de encomendas: ou se fazia em formato papel, ou através de email, ou através de passa-a-palavra. É um processo que se revela demorado e pouco eficaz que se baseia em mecanismos que poderão ser melhorados substancialmente com a utilização de uma aplicação web. Para este problema, foi desenvolvida uma Aplicação Web, que recorre à framework PHP - Laravel e MySQL.

Esta aplicação tem a capacidade de gerir encomendas, bem como os clientes e os fornecedores da empresa. Permite também a gestão do armazém, criação de amostras assim como consultar a evolução das encomendas ao longo do tempo. Todos estes processos passarão a ter um método de resposta às necessidades da empresa e a otimizar o tempo despendido em tarefas que seriam outrora realizadas manualmente.

Atualmente, a plataforma encontra-se a ser utilizada parcialmente, com o objetivo de se tornar no modelo principal de gestão de toda a empresa dentro de alguns meses.

Palavras-Chave:

Gestão, Stocks, Encomendas, Aplicação Web, PHP, Laravel, MySQL

Abstract

The stock and order management has been a constant need in factory environments over the ages. This is a method that allows the management of a specific product in a way that it is possible to know exactly what is its amount in a specific moment. As time went by, and with the arrival of computers to the bigger industries, the process optimization has become a trend which helped with the management and productivity increase in companies.

The developed project was the attempt of solving this problem on the company INDMEI, responsible for the socks production oriented towards the orders export, especially to northern countries. This company who, until then didn't have an automatic stock management mechanism and which started the process of order production by walking to the warehouse and collecting the existing stock on a piece of paper.

Also the communication between employees was not the ideal, since there was no specific channel to handle information respecting the order creation: it was done by paper, through email, or by talking with each other.

It is a process known to be too long and not very effective who is based in mechanisms that can be substantially improved by using a web application. In order to solve this problem, a Web Application was developed using the PHP framework - Laravel and MySQL.

This application is able to manage orders as well as clients and suppliers. It also allows the stock management, creation of samples and consult the orders evolution throughout the time. All these processes will thus have an answer to all the company needs, optimizing the time spent in tasks that were previously handled manually. Currently, the platform is being used partially, aiming to become the main management model of the whole company in a matter of a few months.

Keywords:

Management, Stocks, Orders, Web Application, PHP, Laravel, MySQL

Índice

Resumo	i
Abstract	iii
1 Introdução	1
1.1 Contextualização	2
1.2 Objetivos	3
1.3 Calendarização	4
1.4 Organização do Relatório	5
2 Estado da Arte	7
2.1 Soluções de Software Disponíveis	8
2.1.1 Software Gratuito	8
2.1.2 Software Pago	12
2.2 Soluções Equacionadas pela Empresa	18
2.2.1 PROtextil	19
2.2.2 MacWin	21
3 Software e Ferramentas de Desenvolvimento	25
3.1 Back-End (PHP - Composer - Laravel Framework)	26
3.2 Base de Dados (MySQL)	29
3.3 Front-End (HTML, CSS, Javascript)	30
3.4 JIRA e Metodologia SCRUM	31
4 Proposta de Protótipo	35
4.1 Análise de Requisitos	36
4.2 Histórias de Utilizador	37
4.3 Tipos de Permissões a Implementar	38
4.3.1 Admin	39
4.3.2 Convidado	40
4.3.3 Gestor de Encomendas	40

4.3.4	Gestor de Amostra de Artigo	40
4.3.5	Gestor de Orçamentação	41
4.3.6	Gestor de Armazém	41
4.3.7	Operário	42
4.4	Fluxograma do Workflow dentro da INDMEI	42
4.5	Atores e Casos de Uso	44
5	Desenvolvimento da aplicação	51
5.1	Estrutura e relações da Base de Dados	51
5.1.1	Modelos de Dados (Diagrama EER)	52
5.1.2	Relação do módulo dos Users e Permissões	54
5.1.3	Relação do módulo da Amostra de Artigo	55
5.1.4	Relações do módulo de Armazém	56
5.1.5	Relações do módulo de Encomendas, Orçamentação e Operário	58
5.1.6	Relações dos módulos Auxiliares	59
5.2	Funções Implementadas	60
5.2.1	Sistema de Login e Registo	62
5.2.2	Flash Messages	67
5.2.3	Table Seeder	69
5.2.4	Sistema de Envio de Emails	71
5.2.5	Tabelas de Dados - Datatables	72
5.2.6	Dados Estatísticos - Chart.js	75
5.2.7	Atualização do Stock do Armazém	78
5.2.8	Upload de Ficheiros	82
5.2.9	Atualização de Custo de uma Amostra	84
6	Demonstração e Análise dos Resultados	87
6.1	Funcionalidades da Aplicação	88
6.1.1	Login	88
6.1.2	Menus	91
6.1.3	Gerir Permissões	91
6.1.4	Gestão de Encomendas	95
6.1.5	Estatísticas	100
6.1.6	Gestão de Fornecedores	101
6.1.7	Gestão de Clientes	103
6.1.8	Gestão de Amostras	105
6.1.9	Gestão de Orçamentação	107
6.1.10	Gestão de Armazém	110

6.1.11	Encomendas em Produção	114
6.2	Gestão dos erros	117
6.3	Notificações via Email	119
6.4	Feedback da Empresa	123
7	Conclusão e Trabalho Futuro	127
7.1	Conclusão	127
7.2	Trabalho Futuro	129
7.2.1	Módulo de solicitação de matéria-prima	129
7.2.2	Módulo de notificações in real time	129
7.2.3	Módulo de duplicar amostra de artigo	130

Índice de figuras

2.1	Aspeto da plataforma.	8
2.2	Características gerais da plataforma	9
2.3	Aspeto da plataforma (1).	10
2.4	Aspeto da plataforma (2).	10
2.5	Aspeto da plataforma.	13
2.6	Exemplo retirado do vídeo ilustrativo da plataforma.	14
2.7	Exemplo da plataforma.	15
2.8	Aspeto da plataforma.	16
2.9	Tabela ilustrativa das características e versões.	17
2.10	Ilustração das funcionalidades do software.	17
2.11	Leitores de código de barras da plataforma.	18
2.12	Modelo de apresentação do software.	20
2.13	Ilustração de funcionamento da plataforma.	21
2.14	Estrutura modular do software.	22
3.1	Página oficial - Composer.	26
3.2	Página oficial - Laravel.	27
3.3	Interesse na framework em comparação com as concorrentes, ao longo do tempo.	28
3.4	Percentagem de projetos encontrados por framework de PHP.	28
3.5	Funcionamento da <i>framework</i> SCRUM.	33
4.1	Interação da aplicação web com a base de dados e os dispositivos.	36
4.2	Funcionalidades da permissão Admin.	40
4.3	Funcionalidades da permissão Gestão de Encomenda.	40
4.4	Funcionalidades da permissão Gestão de Amostra de Artigo.	41
4.5	Funcionalidades da permissão Gestão de Orçamentação.	41
4.6	Funcionalidades da permissão Gestão de Armazém.	41
4.7	Funcionalidades da permissão Operário.	42
4.8	Fluxograma com o ciclo do produto da empresa INDMEI.	43

4.9	Diagrama de casos de uso.	48
5.1	Representaçãode todas as relações da base de dados.	53
5.2	Relações do módulo de utilizadores.	54
5.3	Relações do módulo das amostras de artigo.	55
5.4	Relações do módulo de Armazém.	57
5.5	Relações do módulo de Encomendas, Orçamentação e Operário.	58
5.6	Relações dos módulos Auxiliares.	59
5.7	Diretório de pastas da aplicação web.	61
5.8	Controllers criados com "php artisan make:auth".	63
5.9	Model Users.	64
5.10	Migrations.	65
5.11	Routes de autenticação.	66
5.12	Validações no registo.	66
5.13	Validações no login.	67
5.14	Implementação de mensagem Flash.	68
5.15	View correspondente da mensagem Flash.	69
5.16	Diretório de <i>Seeders</i> desenvolvidos.	70
5.17	Seeder para criar os estados de encomenda.	70
5.18	Configuração de email.	71
5.19	Email Controller.	72
5.20	Relacionamento dos módulos Auxiliares.	72
5.21	Ficheiros auxiliares da biblioteca DataTables.	73
5.22	Implementação de datatable.	74
5.23	Tabela apresentada para ecrãs maiores.	74
5.24	Tabela apresentada para ecrãs menores.	75
5.25	Controller responsável pelo envio dos dados para a view.	75
5.26	Estrutura de dados formatada.	76
5.27	Chamada do script e css da biblioteca chartjs.	76
5.28	Script implementado para um dos gráficos.	77
5.29	Gráfico implementado.	77
5.30	Atualização de valores ao listar Stocks.	78
5.31	Função que verifica o fio gasto por par de meias.	80
5.32	Função que obtém o total líquido de pares de meias, por cor.	80
5.33	Função que obtém o total bruto de pares de meias, por cor.	81
5.34	Método para adicionar stock.	82
5.35	Validações aos ficheiros que são carregados.	83
5.36	Upload de ficheiros.	84

5.37	Atualização do custo de uma amostra.	84
6.1	Ecrã inicial em <i>desktop</i> e <i>mobile</i>	88
6.2	Ecrã de registo em <i>desktop</i> e <i>mobile</i>	89
6.3	Ecrã de login em <i>desktop</i> e <i>mobile</i>	90
6.4	Ecrã de esquecimento de palavra-passe em <i>desktop</i> e <i>mobile</i>	90
6.5	Menu completo para ecrãs maiores.	91
6.6	Menu completo para ecrãs menores.	91
6.7	Criar nova permissão.	92
6.8	Listar Permissões.	92
6.9	Confirmação ao apagar permissão.	93
6.10	Mensagem de erro ao apagar permissão.	93
6.11	Listar utilizadores.	94
6.12	Confirmação ao eliminar utilizador.	94
6.13	Editar utilizador.	95
6.14	Criar encomenda.	96
6.15	Listar encomendas.	97
6.16	Confirmação antes de apagar encomenda.	97
6.17	Produção atual de uma encomenda.	98
6.18	Contactos apresentados ao enviar email.	98
6.19	Enviar email.	99
6.20	Gerir emails.	100
6.21	Estatísticas da empresa.	101
6.22	Criar fornecedor.	102
6.23	Listar fornecedores.	102
6.24	Eliminar fornecedor.	103
6.25	Criar cliente.	103
6.26	Listar clientes.	104
6.27	Confirmação ao eliminar cliente.	104
6.28	Mensagem de erro ao eliminar cliente.	105
6.29	Criar amostra de artigo.	106
6.30	Listar amostras de artigo.	107
6.31	Listar orçamentos.	108
6.32	Criar orçamento.	109
6.33	Email de envio de orçamento.	110
6.34	Dar entrada de stock.	111
6.35	Criar nova matéria-prima.	112
6.36	Listar stock.	113

6.37 Histórico de matéria-prima.	113
6.38 Confirmação ao eliminar matéria-prima.	114
6.39 Solicitar matéria-prima.	114
6.40 Encomenda em produção.	116
6.41 Erro 404 - page not found.	117
6.42 Erro inesperado na plataforma.	118
6.43 Email de erro inesperado na plataforma.	118
6.44 Notificação de encomenda à espera de amostra.	119
6.45 Notificação de nova amostra criada.	120
6.46 Notificação de encomenda à espera de orçamentação.	121
6.47 Notificação de orçamento efetuado.	121
6.48 Notificação de encomenda em produção.	122
6.49 Notificação de encomenda finalizada.	123

Índice de tabelas

1.1	Calendarização do projeto	4
2.1	Custos das diversas versões do software	11
4.1	Tabela de <i>User Stories</i>	38
4.2	Tabela com Atores e Casos de Uso (1)	46
4.3	Tabela com Atores e Casos de Uso(2)	47

Acrónimos

API	Aplication Programming Interface
CORS	Cross Origin Resource Sharing
CSS	Cascading Style Sheet
CSV	Comma-separated values
DB	Database
EDI	Electronic Data Interchange
EER	Enhanced Entity-Relationship
ERP	Enterprise Resource Planning
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
IE9	Internet Explorer 9
IIS	Internet Information Services
ITV	Independent Television
JS	JavaScript
MVC	Model View Controller
NIF	Número de Identificação Fiscal
OAUTH	Open Authentication
ORM	Object Relational Model
OS	Operative System
PALOP	Países Africanos de Língua Oficial Portuguesa
PDF	Portable Document Format
PHP	Hypertext Preprocessor
PME	Pequenas e Médias Empresas
RFID	Radio Frequency Identification
SAFT	Standard Audit File for Tax Purposes

SMTP	Simple Mail Transfer Protocol
UML	Unified Modeling Language
URL	Uniform Resource Locator
USB	Universal Serial Bus
XAMPP	Apache+MariaDB+PHP+Perl

1

Introdução

O controlo e a gestão de *stocks* e de produção têm sofrido uma evolução constante que existe desde o Egito antigo onde era contabilizado o *stock* de grãos e cereais das suas plantações. Também os comerciantes tiveram necessidade de contabilizar as quantidades de produto existente, trabalho esse realizado manualmente. Trata-se de um processo demorado e pouco eficiente e que torna muito fácil cometer erros de origem humana - um produto pode ser mal contabilizado com uns segundos de distração.

Mais recentemente, nos anos 60, o controlo de *stock* sofreu uma mudança mais relevante com a criação e implementação de códigos de barras que identificam os produtos. Esta é uma solução muito utilizada em hipermercados e em vários tipos de lojas de conveniência. Foi de resto uma solução padronizada a partir de 1974, passando a ser utilizada em todo o mundo.

A evolução do mundo tecnológico permitiu que se encontrem soluções diversas para o controlo de *stock* e de produção que têm vindo a evoluir durante os últimos 30 anos até aos dias de hoje.

É muito fácil entender o peso que tem vindo a ser dado à gestão de *stocks*, produtos e encomendas, havendo para isso software cada vez mais capaz e eficaz na mesma gestão. Por outro lado, existe uma enorme quantidade de oferta mas que pode não significar exatamente que exista algum *software* que satisfaça especificamente todas as funcionalidades que se pretendem. Posto isto, um software feito à medida das necessidades de uma empresa torna-se sempre na solução mais adequada, uma vez que

não há necessidade de adaptar conteúdos e tarefas, ou de remover ou adicionar novas tarefas. Esta enorme vantagem pode, em casos particulares, sobrepor-se mesmo aos custos de desenvolvimento que uma solução dedicada acarreta.

Este capítulo pretende enquadrar o projeto desenvolvido para a tese, ou seja, o desenvolvimento de uma aplicação para a criação e gestão de encomendas e produtos.

1.1 Contextualização

A empresa INDMEI - fabrico de meias, lda intitula-se como sendo uma empresa Portuguesa sediada na zona Industrial de Laundos, no concelho da Póvoa de Varzim e que é especializada no fabrico de meias e collants para clientes específicos. Foi fundada no final dos anos 90, conta já com mais de 15 anos de existência e experiência no setor específico em que atua.

É nos dias de hoje uma empresa especialista no setor e que se distingue pela qualidade e inovação dos seus produtos dedicados a mercados muito específicos, como por exemplo o nórdico e o medicinal. O atual processo produtivo da empresa é constituído por máquinas tecnologicamente avançadas, permitindo produções elevadas e com padrões de qualidade superior, satisfazendo assim todas as exigências técnicas que são solicitadas pelos seus clientes.

Cada máquina que trata da produção da meia possui entre 150 e 200 agulhas que trabalham em simultâneo para através da linha inserida produzirem uma meia de acordo com as especificações. Estas máquinas são programadas através de um software proprietário que funciona à base de comandos predefinidos. Estes comandos permitem a estruturação da meia: desde o início da base da meia, passando pelo calcanhar até chegar à parte da perna. Por cada meia produzida são necessários cerca de 2 minutos e 30 segundos, o que resulta numa produção diária, por máquina, de cerca de 200 pares de meias.

Os processos de gestão existente na empresa, de momento, são realizados de forma manual quase na sua totalidade:

- A consulta de *stock* é realizada através do próprio deslocamento do trabalhador ao local. Este processo resulta num tempo dispensado demasiado elevado, uma vez que em todas as encomendas é necessário um processo semelhante. Torna-se um problema ainda maior quando existem várias encomendas em execução nas quais são usadas as mesmas matérias primas. Nestes casos, é necessário calcular valores em *stock*, subtrair matéria prima já "guardada", e verificar efetivamente o valor líquido de matéria ainda existente por processos pouco

expeditos.

- Suportes de informação diversificados e pouco fiáveis. Para a elaboração de uma encomenda é necessário a transmissão de informação entre departamentos. Essa informação é transmitida através de folhas de papel escritas à mão, o que se pode tornar num problema no caso de existir um erro de distração no momento da cópia de valores de papel para papel, ou então de papel para ficheiros excel.
- O cálculo manual de *stocks*, matérias primas necessárias, tempo necessário para a realização de encomendas, percentagem de material extra para garantir a qualidade final do produto. Em qualquer um destes cálculos existe um sem número de fatores que poderão afetar o resultado final.
- A comunicação no geral torna-se difícil e isso origina perdas de tempo útil de trabalho que poderia estar a ser utilizado de forma mais eficiente.

Tendo em conta o mercado agressivo que existe nos dias de hoje, em que cada detalhe pode promover o melhor desenrolar das atividades, é fundamental otimizar todos os processos que permitam uma melhoria ao nível da produção e gestão de tarefas, bem como uma redução do tempo dispendido com a manutenção de *stocks*, e na fase de preparação de tarefas.

Surgiu então a necessidade de otimizar os processos da empresa INDMEI através de uma aplicação web que permite a criação e gestão de encomendas e de produtos.

1.2 Objetivos

O projeto desenvolvido tem como principal objetivo a utilização de um software por parte de todos os intervenientes na empresa que permita um melhor funcionamento de todo o processo de gestão de *stocks*, de encomendas e de produtos.

Para além deste objetivo principal, existem vários objetivos secundários dos quais a aplicação está dependente para um bom funcionamento e também para otimizar todos os processos dos intervenientes.

1.4 Organização do Relatório

No Capítulo 1 é realizada uma breve introdução ao contexto do problema em análise e ao desenvolvimento que será efetuado para resolver esse mesmo problema. É ainda apresentada a calendarização do projeto ao longo do tempo.

O Capítulo 2 representa o estado da arte, onde se começa por apresentar uma lista de soluções que podem ser encontradas online para a gestão de *stocks* e produtos.

Esta lista subdivide-se em duas, mencionando softwares pagos e não pagos. São depois referidas as suas vantagens e desvantagens, passando de seguida para a análise de dois softwares em específico, que foram vistos como alternativas válidas pela empresa INDMEI.

O Capítulo 3 refere o estudo efetuado ao nível do software. São também justificadas as escolhas efetuadas ao nível das linguagens de programação, que por sua vez estão subdivididas em tecnologias Back-End, Front-End e Base de dados.

Depois de selecionadas as ferramentas, no Capítulo 4 é realizada uma proposta de protótipo que será apresentada à empresa INDMEI para validar se corresponde a todos os critérios requisitados. É também identificada a forma como a empresa gere o fluxo de uma encomenda, quais os atores envolvidos e os casos de uso.

O Capítulo 5 diz respeito a todo o desenvolvimento da aplicação web. Tendo por base a *framework* selecionada, este capítulo está subdividido em estrutura e relacionamentos da Base de dados, e a forma como as funcionalidades mais interessantes foram implementadas, de forma a otimizar todo o sistema.

O Capítulo 6 corresponde à forma como as funcionalidades foram implementadas do ponto de vista do utilizador final, com recurso à apresentação e explicação de várias funcionalidades do ponto de vista dos diversos atores. É também realizada uma breve análise dos resultados obtidos.

No último capítulo (7), são fornecidas algumas melhorias a implementar na aplicação, nomeadamente ao nível de notificações, módulos que fazem sentido acrescentar ao sistema e ainda funcionalidades ao nível utilitário para a empresa. O projeto de Tese termina apresentando as conclusões finais a retirar do projeto, tendo por base os objetivos que foram propostos.

2

Estado da Arte

A constante evolução das tecnologias em todas as suas vertentes possibilita que novas ferramentas emergjam com relativa frequência e ferramentas cada vez mais elaboradas.

Este processo acontece também nas ferramentas de gestão empresarial, como a gestão de stocks e de produtos. Embora este procedimento tenha aparecido pela primeira vez, de uma forma arcaica, há centenas de anos atrás, nas culturas egípcias, rapidamente se compreendeu a importância e a necessidade de tornar este processo o mais rápido e eficiente possível. Existe portanto a possibilidade de gerir o tempo com tarefas que resultem efetivamente na produção de material e em produtividade efetiva.

A indústria veio acelerar esta evolução, uma vez que rapidamente aumentou a concorrência entre competidores que produzem os mesmos materiais. Isto obriga as empresas a procurar ferramentas de gestão que acrescentam uma mais valia na gestão na sua área de negócio.

Por este motivo, este capítulo passa por realizar um estudo da quantidade e também da qualidade dos *software* que tornam possível a realização das operações de gestão de stocks e de encomendas, para indústrias em que se enquadra a realização deste projeto. Este estudo permitirá ainda verificar onde se encontra o foco das grandes empresas que produzem este tipo de ferramentas, para verificar aquilo de os produtores procuram encontrar.

2.1. SOLUÇÕES DE SOFTWARE DISPONÍVEIS

Para além de um estudo mais genérico pelas ferramentas mais procuradas *online*, foram depois focadas duas ferramentas que a empresa INDMEI considerou adquirir junto dos criadores e o que motivou inicialmente a sua aquisição em detrimento de outras aplicações do mesmo tipo e com características semelhantes, o que posteriormente levou ao abandono desta ideia.

2.1 Soluções de Software Disponíveis

O fator evolutivo desencadeado pelo aparecimento da internet permite que através da pesquisa online seja encontrado um leque de oferta bastante diverso e para as mais variadas opções de negócio. Podem ser divididas em soluções encontradas gratuitamente, bem como soluções que requerem pagamentos de licenças e também pagamentos mensais.

2.1.1 Software Gratuito

inWork

O inWork é apresentado como sendo um software com um conjunto alargado de ferramentas que se foca na produtividade e eficiência da empresa, respondendo às necessidade dos clientes.

O inWork ERP é o software que a empresa desenvolveu para manter a organização e funcionalidade da empresa e o seu objetivo é o de simplificar a gestão comercial e financeira de pequenas e médias empresas, bem como de empresários em nome individual. Na figura 2.1 é apresentado o *layout* da ferramenta introduzida.

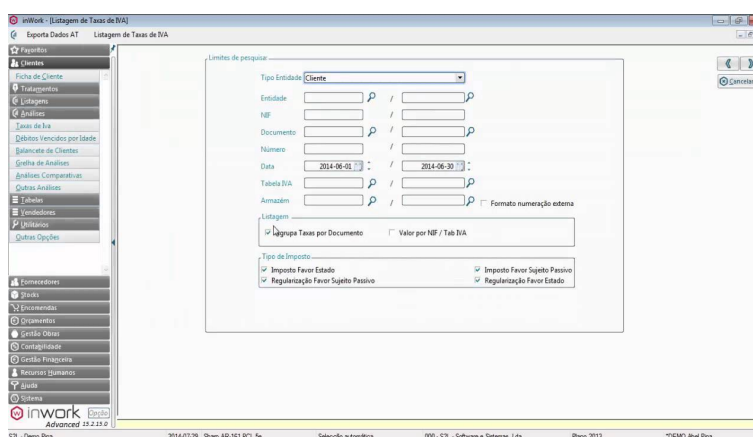


Figure 2.1: *Aspetto da plataforma.*

O software permite ainda a impressão e exportação de documentos em formato csv para reduzir o uso de papel e para melhor permitir arquivar todos os processos.

Em termos de dados estatísticos, o controlo pode ser consultado através de gráficos e relatórios.

O software permite ainda o controlo de acesso por utilizador. Através de uma breve pesquisa, é possível verificar que nem todas as opções estão disponíveis com um pacote *free*, havendo necessidade de realizar o upgrade dependendo das necessidades da empresa.

Características	Free	Starter	Classic	Advanced
Limite de utilizadores	1	4	Ilimitado	Ilimitado
Gestão de Clientes	✓	✓	●	●
Gestão de Fornecedores	✓	✓	●	●
Gestão de Stocks	✓	✓	●	●
Gestão de Lotes	✗	✗	✓	✓
Gestão de acessos	●	●	●	●
Atributos na ficha de Entidades e Produtos.	●	●	●	●
Funções.	●	●	●	●
Eventos	●	●	●	●
Visualização de documentos	✓	✓	✓	✓
Listas	●	●	●	●
Séries de valores de contexto	●	●	●	●
Valores de contexto	●	●	●	●
Importação de dados em excel/openoffice	✓	✓	✓	✓
Validação e obtenção de dados associados ao NIF	✓	✓	✓	✓
Listagens e análises gráficas	●	●	●	●
Grelhas de análise	●	●	●	●
Análises comparativas	●	●	●	●
Documentos configurados	●	●	●	●
Stocks multi-armazém	✗	✗	✓	✓
Tabelas de Preço	●	●	●	●
Preços cliente/produto	✗	✗	●	●
Documentos em rascunho	●	●	✓	✓
Módulo POS	✗	Opcional	✓	✓
Tratamento de produtos compostos	✗	✗	✓	✓
Produtos em cadeia	✗	✗	✓	✓
Tara e vasilhames	✗	✗	Opcional	Opcional
Recebimentos por multibanco	✗	✗	Opcional	Opcional
Recebimentos por transferência bancária	✗	✗	Opcional	Opcional
Módulo para bombas de gasolina	✗	Opcional	Opcional	Opcional
Produtos em promoção	✗	✗	✓	✓
Envio de email interno nos reportes	✗	✓	✓	✓
Importação de documentos em bloco	✗	✗	✓	✓
Gestão de encomendas	✗	Opcional	Opcional	Opcional
Contabilidade	✗	Opcional	Opcional	Opcional

Figure 2.2: Características gerais da plataforma

Existe ainda uma limitação de 100 clientes e/ou fornecedores e também a 50 documentos por mês [1].

2.1. SOLUÇÕES DE SOFTWARE DISPONÍVEIS

EasyForYou

A EasyForYou existe desde 1992 em que a primeira versão para Windows foi lançada em 2002 e oferece uma solução de gestão de Stocks e Faturação. É identificada como sendo adaptável a qualquer ambiente profissional em nome individual ou para PME's.

As figuras 2.3 e 2.4 apresentadas a seguir demonstram o *layout* da plataforma em maior detalhe.

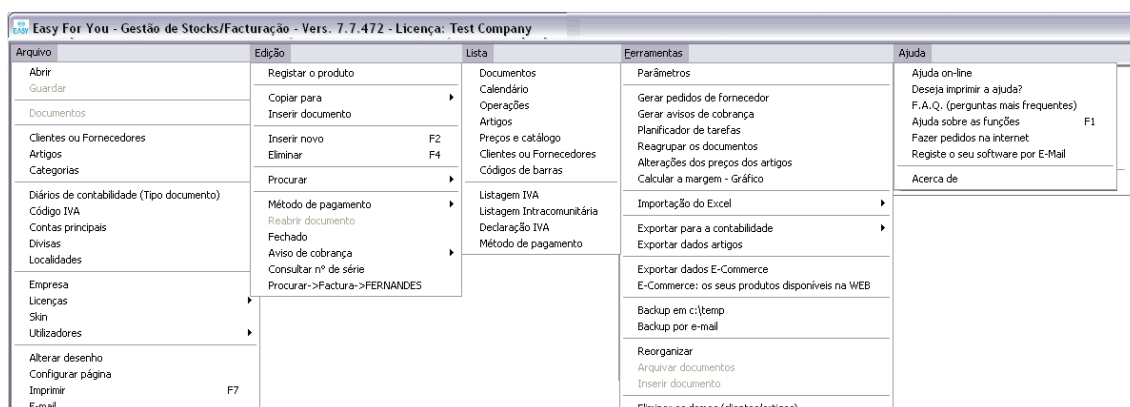


Figure 2.3: *Aspetto da plataforma(1).*

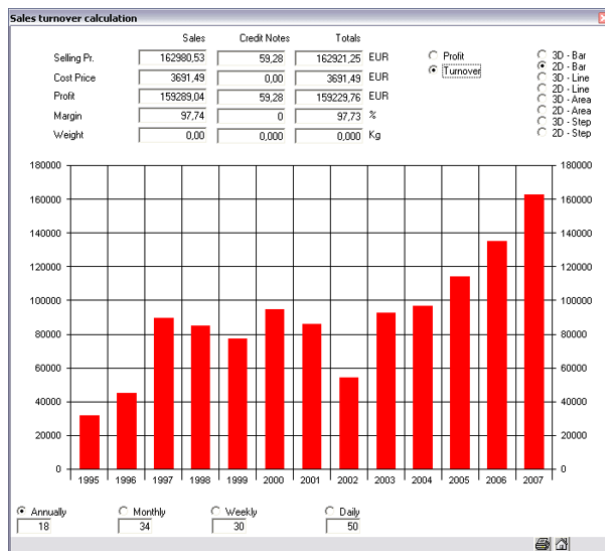


Figure 2.4: *Aspetto da plataforma (2).*

São apontados como pontos de venda:

- Funcionalidades à distância de apenas um clique;

- Diversos módulos à disposição do utilizador incluindo o SAFT export;
- Importação de ficheiros excel;
- Oferece soluções e-commerce;
- Permite o seguimento de cobranças;
- Gerar e fazer a gestão automática de avisos de cobrança;
- Permite a formatação e envio de documentos por email;
- Realiza o cálculo da margem do negócio;
- Permite a gestão do negócio através da web.

Embora bastante multifacetado, possui uma versão free bastante limitada e que termina o período de teste ao final de 30 dias, 120 documentos impressos ou 300 movimentos na plataforma.

Os preços anunciados no website do software variam conforme o quadro 2.1:

Table 2.1: *Custos das diversas versões do software*

Nome Software	Valor Total (eur)	Valor após desconto (eur)
Evolution	169	99
Stock Basic	235	149
Premium	453	299
Start My Shop	643	399
Advanced	984	599
Pro	1984	1353
E-Commercero	894	599

São também vendidos separadamente módulos de software que se pretendam adicionar, bem como os dispositivos para gestão do stock (scanner de código de barras, data terminals, caixas de dinheiro automático, USB Powered Displays e Labels Printer Star)[2].

Com base nos *software* gratuitos analisados, a principal conclusão a retirar está relacionada com a usabilidade dos mesmos. Como é possível verificar, todos eles possuem funcionalidades interessantes. Contudo, as funcionalidades não foram desenvolvidas especificamente para o que é pretendido pela empresa em questão. Outra

importante conclusão a reter é que em softwares como o caso do produto desenvolvido pela EasyForYou possui módulos extra que vão para além do plano base e que por consequência correspondem a um custo associado. O terceiro ponto também importante refere-se ao suporte dado por que desenvolveu os produtos: será muito reduzido, ou então, estará associado a custos que poderão até ultrapassar o custo do próprio *software* a curto prazo.

2.1.2 Software Pago

Na ótica dos *software* pagos, é possível encontrar algumas soluções aceitáveis para a gestão de stock e de produto.

Contudo, este tipo de *software* tem a principal desvantagem de necessitarem de um custo associado, quer pelo software, quer por eventuais adições (pacotes extra), quer pela manutenção e atualizações.

CentralGest

A CentralGest oferece uma solução empresarial totalmente integrada, com 31 anos de história e com a colaboração de milhares de clientes em Portugal e nos países africanos de língua oficial portuguesa (PALOP), tais como Angola, Moçambique e Cabo Verde.

A CentralGest desenvolveu um software de GEstão de Produção com o principal objetivo de acompanhar e registar tudo o que ocorre na produção das empresas industriais de forma extremamente flexível e adaptável.

Trata-se de um complemento a um outro módulo, neste caso o de Logística, o que se associa imediatamente a um custo extra.

O módulo de Gestão de Produção é uma solução de grande abrangência que permite efetuar o planeamento e controlo das atividades necessárias [3].

As principais características são as seguintes:

- Permite visualizar ordens de produção;
- Planear ordens de produção;
- Iniciar, terminar, interromper ou reiniciar ordens de produção;
- Permite acompanhar as recolhas realizadas.

2.1. SOLUÇÕES DE SOFTWARE DISPONÍVEIS

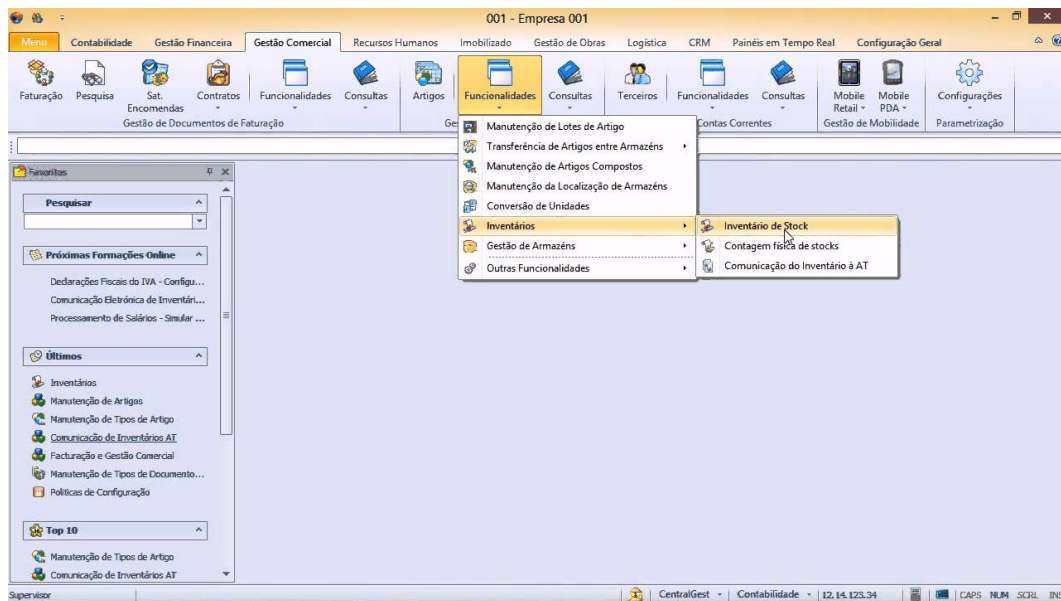


Figure 2.5: *Aspetto da plataforma*

O módulo referido permite ainda multi utilizador com características relevantes para cada perfil existente.

Com este software, a empresa pretende essencialmente incidir em quatro fatores que sairão beneficiados para quem adquirir o produto:

1. Redução de custos operacionais;
2. Consistência, fiabilidade e qualidade de informação;
3. Disponibilidade de informação;
4. Automatização de processos na empresa.

ArtSoft

A ARTSOFT é uma empresa de software que desenvolveu uma ferramenta denominada Gestão de Encomendas e o Rateio ARTSOFT. Este *software* permite automatizar estes processos com base na análise de stocks, processos de produção e encomendas por satisfazer.

Trata-se de um software que se destina a empresas que pretendem efetuar encomendas a fornecedores com base nos pedidos de clientes, garantindo desta forma a reserva e alocação das mercadorias necessárias à venda ou produção, logo no momento da receção.

As principais vantagens da Gestão de Encomendas e Rateio ARTSOFT prendem-se com a eliminação de esquecimentos e diminui a falha humana, o que se converte

2.1. SOLUÇÕES DE SOFTWARE DISPONÍVEIS

na redução dos custos operacionais e aumenta deste modo a satisfação no geral dos clientes. As principais características deste produto são as seguintes:

- Relacionar as encomendas de clientes com os fornecedores;
- Efetua encomendas a fornecedores, com base na lista de encomendas ainda não satisfeitas de clientes;
- Cativa os artigos encomendados pelos clientes, garantindo a reserva dos mesmos.

A empresa ARTSOFT apresenta ainda um vídeo onde demonstra um caso de utilização real do software tal como demonstrado na figura 2.6, um exemplo retirado do vídeo ilustrativo da plataforma [4].

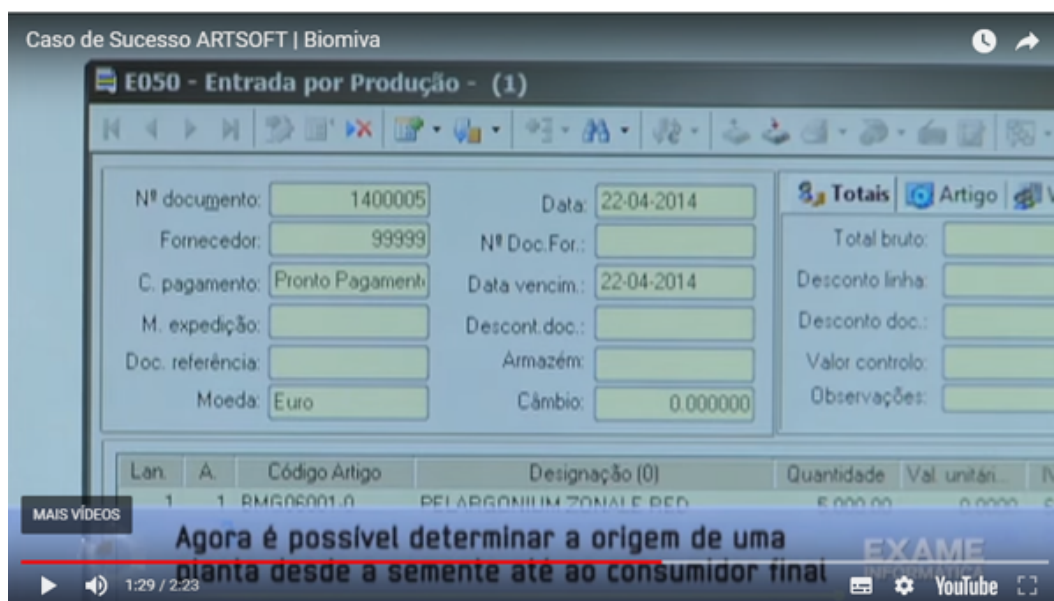


Figure 2.6: Exemplo retirado do vídeo ilustrativo da plataforma.

alvo

A empresa alvo é uma empresa de software que se foca em diversas áreas de desenvolvimento, nomeadamente no software de gestão, faturação, recursos humanos, *business intelligence*, entre outros. Para além disso, ainda apresenta uma distinção por setor, onde compreende o software de administração pública, construção, indústria, distribuição e logística, retalho e restauração.

Ao realizar uma pesquisa sobre os *software* de gestão, é possível verificar a existência de um software orientado para as compras e gestão de stocks - Primavera.

2.1. SOLUÇÕES DE SOFTWARE DISPONÍVEIS

CF	Item	Supr	Artigo	Descrição	Armazém	Quantidade	Preço Unit.	Desconto 1	Preço Líq.	Data Receção	Data Limite Compra	Data Limite Receção	Data Prevista Entrega
				Cenário Compras: GERAL (9)									
	D005			Placa de Son 16000 32 bits		1,00	0,00	0,00	0,00	29/04/2014	29/04/2014	29/04/2014	29/04/2014
	F001			Memória RAM DDR2 (DDR2-2048)		5,00	0,00	0,00	0,00	29/04/2014	15/04/2014	15/04/2014	13/05/2014
	F001			Mesa p/ PC		97,00	30,00	0,00	2.910,00	29/04/2014	12/01/2013	12/01/2013	09/05/2014
	CP05			Gaveta		4,00	0,00	0,00	0,00	29/04/2014	29/04/2014	29/04/2014	29/04/2014
	R007			Rato 4/12B		1,00	0,00	0,00	0,00	29/04/2014	29/01/2013	29/01/2013	29/04/2014
	A005			Secretária 1,50*0,70		30,00	219,70	0,00	6.591,00	29/04/2014	30/01/2013	30/01/2013	29/04/2014
				TAPETE		1,00	0,00	0,00	0,00	29/04/2014	29/04/2014	29/04/2014	29/04/2014
	F001			Memória RAM DDR2 (DDR2-812)		3,00	0,00	0,00	0,00	29/04/2014	15/04/2014	15/04/2014	13/05/2014
	C002			GT-15000 600x1200pp 48bit USB S...		1,00	1.635,00	0,00	1.635,00	29/04/2014	29/04/2014	29/04/2014	29/04/2014

Documento	Entidade	Nome	Artigo	Descrição	Armazém	Data Ref.	Quantidade
ECL A/10	SILVA	Maria José da Silva	A0005	Secretária 1,50*0,70	A1	28/02/2013	30,00
ECL A/9	SOPRJO	Soeiro, Lda	A0005	Secretária 1,50*0,70	A1	30/01/2013	10,00

Figure 2.7: Exemplo da plataforma.

O software de compras e stocks Primavera tem como objetivo facilitar a gestão do aprovisionamento, para comprar na altura certa, na quantidade adequada e com a qualidade desejada ao menor custo.

A alvo foca como um dos principais pontos o aprovisionamento na altura certa explicando o benefício de obter projeções automáticas de necessidades. Permite o acompanhamento da evolução de cada pedido, desde a sua orçamentação, encomenda até à sua receção, bem como o controlo de todos os processos de compras, podendo incluí-los nas previsões de existências em novas projeções de necessidades.

Outros pontos defendidos pelo software de compras e gestão de stocks - Primavera permite as transações eletrónicas via Eletctronic Data Interchange (EDI). Ao tornar o meio de transações eletrónico, é possível agilizar a gestão de notas de encomenda, notas de débito e crédito bem como as guias de remessa e faturas.

A gestão de contratos de compra é também abordada neste software, uma vez que contém uma funcionalidade que permite a gestão de contratos com múltiplos fornecedores e com múltiplas regras definidas com tipo de artigo, fornecedor, quantidade, meios de pagamento ou outros fatores.

O software Primavera suporta ainda a gestão automatizada do ciclo de vida dos contratos de compra, de acordo com as regras do negócio [5].

Outros pontos considerados a favor deste software são os seguintes:

- Permite o comércio externo, ou seja, a importação de matérias-primas ou outro tipo de produtos;
- Intrastat - O mapa Intrastat é obrigatório para as pessoas ou entidades abrangi-

2.1. SOLUÇÕES DE SOFTWARE DISPONÍVEIS

das pelo regime normal de IVA, que intervenham em transações de mercadorias da União Europeia;

- Assistência técnica personalizada;
- Informação acerca do nível ideal de stocks;
- Otimização de inventário;
- Gestão integrada de armazéns;
- Fluxos documentais;
- Indicadores de gestão.

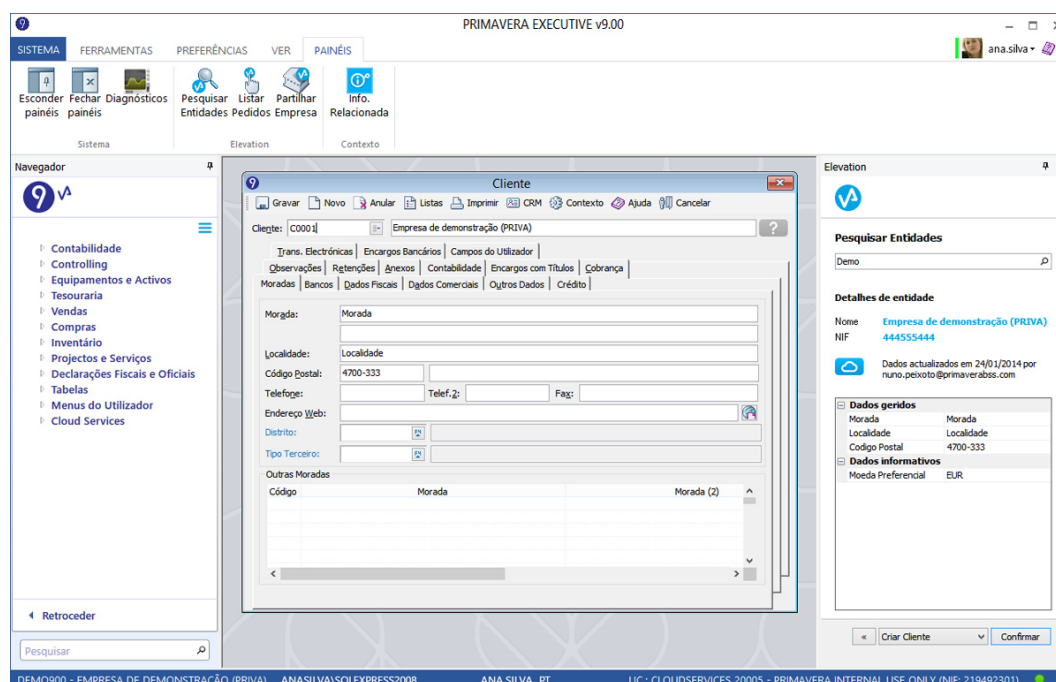


Figure 2.8: *Aspetto da plataforma.*

Tendo em conta a diversidade de opções deste software, uma das desvantagens prende-se com o preço, que poderá variar dos 129eur/ano até aos 299eur/ano, conforme verificado na tabela da imagem 2.9:

2.1. SOLUÇÕES DE SOFTWARE DISPONÍVEIS

	Starter Easy v9	Starter v9	Starter Plus v9	Professional v9	Executive v9	Jasmin
Valor (a partir de)	129€/ano	229€/ano	299€/ano	Para mais informações por favor contacte-nos	Para mais informações por favor contacte-nos	131€/ano (1)
	Cloud ou na sua empresa	Cloud ou na sua empresa	Cloud ou na sua empresa	Cloud ou na sua empresa	Na sua empresa	Cloud
Resumo das Funcionalidades						
Vendas	✓	✓	✓	✓	✓	○
Comunicações de documentos à AT (apenas offline)		✓	✓	✓	✓	
Ecovalor	✓	✓	✓	✓	✓	
Gestão de Retenções	✓	✓	✓	✓	✓	
Intrastat				✓	✓	
Faturação Eletrónica				○	○	
Compras			✓	✓	✓	○
Stocks e Inventário	✓	✓	✓	✓	✓	○

Figure 2.9: Tabela ilustrativa das características e versões.

XSTOCK

O XSTOCK é um Software de Gestão de *Stocks* e Inventário e que permite otimizar recursos, espaço e tempo em todas as operações de gestão de stocks em armazém e controlo de imobilizado.

As principais vantagens deste software estão relacionadas com a praticidade do software, uma vez que funciona em dispositivos móveis, o que confere aos utilizadores realizar um controlo de stock diretamente junto ao produto, naquele momento [6]. Existe ainda uma variada aplicação do seu produto na área industrial como se verifica na imagem a seguir:

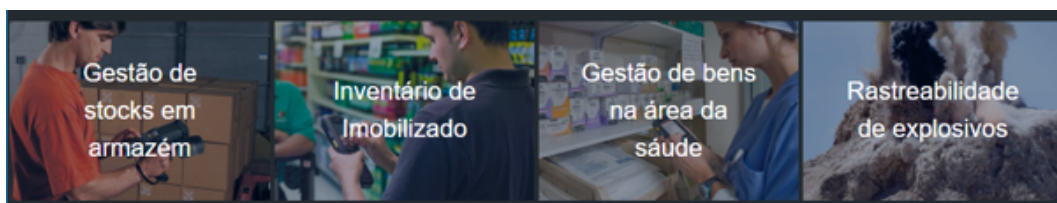


Figure 2.10: Ilustração das funcionalidades do software.

Existe ainda uma solução automatizada de picking móvel para inventário e contagem de produtos (ver figura 2.11):

2.2. SOLUÇÕES EQUACIONADAS PELA EMPRESA

- Gestão de stocks com recolha de listagens e atualização de informação de produtos, com ou sem contagens;
- Picking e validação de cross-checking automático. Comparação de contagens através de cor e avisos de sessão em conformidade;
- Criação de vários utilizadores com acesso à aplicação e sessões de trabalho distintas por cada um;
- Capacidade de realizar o inventário através de código de barras ou tag RFID (conforme o terminal).



Figure 2.11: Leitores de código de barras da plataforma.

2.2 Soluções Equacionadas pela Empresa

Tendo em conta a enorme oferta para o tipo de serviço necessário, a empresa Indmei informou-se acerca das possibilidades e verificou que existiam duas soluções que poderiam ser do seu interesse.

Tendo em conta o facto de existirem diversas ofertas online para as mais diversas áreas e para os diversos tamanhos de empresas, foi estudada a viabilidade de dois *software*: PROTextil e Macwin.

Foram tidos em conta essencialmente dois fatores:

1. Localização: Ambas as empresas que desenvolveram os *software* encontram-se relativamente próximas à empresa em questão (ambas situadas em Barcelos), o que permitiria uma maior facilidade em instalação e manutenção do software. Dada a relativa proximidade entre os criadores de ambos os *software*, também poderia advir daqui algum poder negocial, pelo que seria uma mais valia.

2. Conhecimento do negócio: Apesar de existirem diversas ferramentas de controlo de stock e de produto, nem todas são diretamente orientadas para a produção têxtil. Dada a vasta existência de empresas têxteis na zona em questão, ninguém melhor do que as empresas de software para terem conhecimento das necessidades dos produtores têxteis e, portanto, foi também visto como uma mais valia.

2.2.1 PROtextil

O Software PROTextil foi desenvolvido pela empresa Inforcávada - Informática, Lda. A empresa iniciou a sua atividade em 2002 com o objetivo de desenvolver e implementar sistemas de informação de apoio à gestão.

Com o trabalho realizado ao longo de mais de 15 anos, o software que mais tem adicionado valor à empresa é o PROTextil, sendo utilizado em mais de 50 empresas a nível nacional.

A empresa pretende destacar-se pela Inovação, Qualidade, Disponibilidade, Excelência, Ambição e Flexibilidade.

Apresentando o software em si, este surge no seio da Indústria Têxtil de Vestuário (ITV), cobrindo vários ramos de atividade dentro deste mesmo setor.

A empresa defende como principais vantagens na aquisição do PROTêxtil as seguintes:

1. Flexibilidade em termos de configuração e adaptação às necessidades de cada empresa;
2. Baixo custo de manutenção, quando comparado com outras soluções concorrentes;
3. Design funcional que se traduz numa experiência única em termos de usabilidade;
4. Inovação constante com o lançamento de atualizações semanais; Suporte rápido e eficaz via internet;
5. Capacidade de apoio em termos de consultoria e implementação, resultante do know-how adquirido desde 2002 na ITV.



Figure 2.12: *Modelo de apresentação do software.*

O software permite realizar as seguintes funções:

- Registo de encomendas de clientes;
- Desenvolvimento do Produto;
- Planeamento;
- Gestão de Corte;
- Gestão de *Stocks*;
- Gestão de Subcontratos;
- Gestão de Produção Interna;
- Ferramentas de Controlo de Produção;
- *Packaging-List* (Exportação);
- Gestão de Separação e Logística;
- Faturação;
- Gestão de Tesouraria;
- Análise de custos e Rentabilidade.

Em termos de planeamento, o software PROTextil disponibiliza um conjunto de ferramentas que permitem ao diretos de produção definir de uma forma geral as datas de entrega de cada encomenda e de uma forma mais detalhada as fases de fabrico de cada artigo, tendo em conta os recursos disponíveis e o tempo previsto para produção.

Permite ainda uma divisão de tempo mais específica para cada situação, havendo uma representação gráfica como a imagem 2.13 representa:

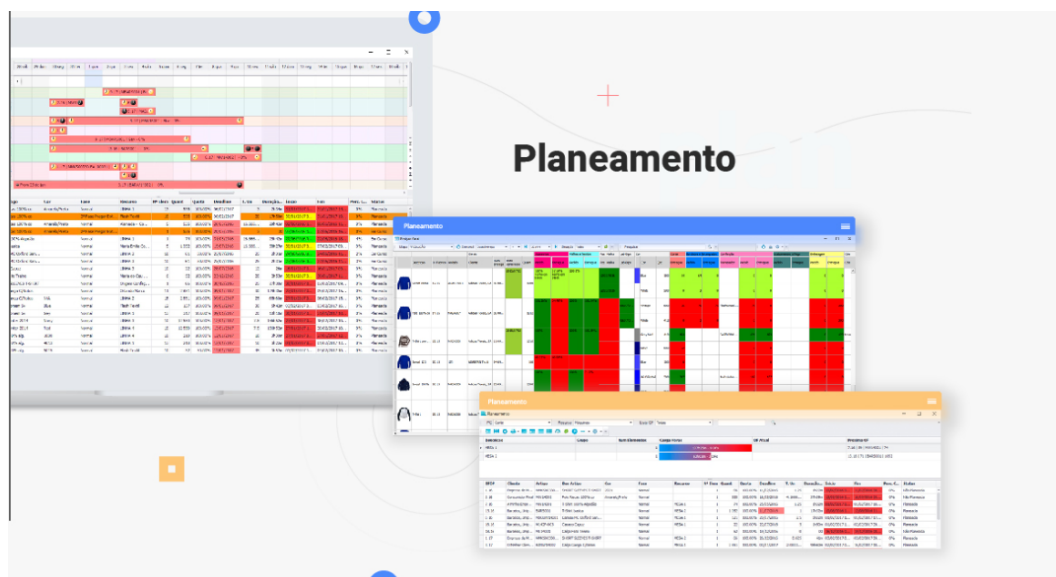


Figure 2.13: *Ilustração de funcionamento da plataforma.*

Há, no entanto, disponíveis vídeos, tutoriais de algumas funcionalidades e até mesmo o download de uma demonstração do software [7].

Existem ainda depois funcionalidades mais específicas da indústria têxtil, tais como a Tinturaria e o Laboratório. Este tipo de funcionalidades são irrelevantes para a empresa Indmei, uma vez que não se aplicam à realidade da empresa.

O facto de não ser utilizado o software na sua totalidade, bem como a falta de algumas funcionalidades que a empresa considera fundamentais fez com que o software PROTêxtil deixasse de ser uma solução compatível com as suas necessidades.

2.2.2 MacWin

O software Enterprise Resource Planning (ERP) MacWin tem 18 anos de existência e intitula-se o software de gestão líder na indústria têxtil, possuindo 80% dos clientes do setor têxtil, contando com técnicos especializados.

A MacWin pretende produzir soluções adequadas a cada cliente, o que torna o trabalho personalizado consoante as necessidades.

Atualmente, a MacWin integra a Holding OSIT SGPS - um grupo de empresas de retalho e serviços, com especial foco nas áreas de desporto e nutrição, e-commerce, tecnologias da informação e comunicação e transportes. Com mais de 250 colabor-

2.2. SOLUÇÕES EQUACIONADAS PELA EMPRESA

adores, o grupo opera no mercado global, com transações comerciais para mais de 100 países.

A MacWin relaciona-se intrinsecamente com a Indústria Têxtil e do Vestuário (ITV), considerando-o mesmo como sendo o seu setor core, onde operam desde 1998. Os principais valores da empresa prendem-se com a cooperação com o cliente, competência técnica, flexibilidade, qualidade e profissionalismo. Relativamente ao ERP MacWin, existe uma diferenciação tendo em conta o setor a gerir associado:

- Têxtil
- Confeção
- Tinturaria
- Tecelagem
- Laboratório

É possível ainda encontrar algumas soluções especializadas nas seguintes áreas:

- Financeiro
- Comercial
- Contabilidade
- Recursos Humanos
- Imobilizado

Ainda destacado como brevemente a apresentar surgem soluções na temática do controlo de qualidade e de receções no armazém.

No que toca ao interesse da empresa Indmei mais especificamente, o GM Têxtil poderia ser a solução a optar, tal como várias outras indústrias exigentes.

Os módulos do GM Têxtil permitem a desmaterialização de todos os processos operacionais, produtivos e administrativos, convertendo-os em informação útil à tomada de decisão [8].



Figure 2.14: *Estrutura modular do software.*

Dos 4 parâmetros do módulo GM Têxtil, a gestão de confeção é aquele que seria do interesse da Indmei, uma vez que não são do âmbito da empresa as questões de tinturaria, tecelagem e laboratório.

A Gestão de Confeção MacWin é a solução adequada e completa de gestão de todo o processo produtivo, que de um modo totalmente integrado, garante um conjunto de

ferramentas ímpares de acesso centralizado a toda a informação necessária à gestão eficaz do negócio.

Desde o desenvolvimento de protótipos, passando pela aquisição de matérias-primas até ao acompanhamento da produção e expedição, a solução Gestão de Confeção reúne toda a informação nevrálgica, garantindo a precisão e a consistência dos dados, eliminando o conteúdo redundante e a duplicação de tarefas.

Este software permite o acesso a um demo do qual se pode fazer o pedido através do preenchimento de um formulário.

Através das duas soluções estudadas pela empresa INDMEI para colmatar a gestão de stocks e encomendas da empresa, é possível verificar que seriam soluções com diversas falhas. Quer no caso do software PROTêxtil quer o ERP MacWin não cumprem os requisitos que a empresa pretende no seu software, sendo que se trata de um software para ambientes têxteis mais complexos e com outro tipo de atividades associadas. O facto de algumas funcionalidades ainda estarem em desenvolvimento tais como as receções no armazém também desfavorecem o software.

Estes *software* deixariam de ser uma mais valia para se tornarem numa sobrecarga em todo o processo. O facto de qualquer uma das medidas estudadas pela empresa serem soluções genéricas significa que qualquer tipo de alteração para as necessidades da empresa tivesse que ser pago. E este processo, dependendo da quantidade de alterações, poderia estar associado a um custo que ultrapassaria o valor de um *software* especificamente desenvolvido para a empresa em questão. Quanto ao suporte constante do software, esse valor também teria que ser tido em conta, o que aumentaria ainda mais o custo total do *software*.

É possível ainda concluir com o estudo para a gestão deste nicho de negócio (ou seja, a produção de meias), existe uma falha na oferta. As soluções estão sempre orientadas para outro tipo de indústria ou então outro tipo de negócio.

3

Software e Ferramentas de Desenvolvimento

A aplicação web a desenvolver tem como objetivo principal uma solução para a gestão de artigos, encomendas e gestão de stocks de matérias primas. Tem também como funcionalidades importantes a desenvolver, alguns medidores de dados estatísticos relevantes através de representações gráficas e tabelas.

De acordo com as especificações do projeto, a aplicação web tem todas as características de um *back-office* de controlo e gestão, com gestão de utilizadores e atribuição de permissões diferenciadoras.

As ferramentas de desenvolvimento foram focadas numa base robusta para a tecnologia *back-end*, passando também por tecnologias amplamente aceites para o desenvolvimento a interface com o utilizador, ou seja, o *front-end*.

Este capítulo divide-se portanto da seguinte forma: será explicada a opção de tecnologia para o desenvolvimento *back-end* no capítulo 3.1.

De seguida no capítulo 3.2. será indicada a tecnologia relacionada com o armazenamento em base de dados.

O subcapítulo 3.3. serão apresentadas as tecnologias front-end para apresentar a interface ao utilizador.

No final do capítulo (3.4), são apresentadas duas tecnologias que irão auxiliar no desenvolvimento do projeto, o software JIRA e a metodologia SCRUM.

3.1 Back-End (PHP - Composer - Laravel Framework)

Para o desenvolvimento do *back-end* foi escolhida a *server script language PHP* associada a uma *framework* - Laravel.

O *Hypertext Preprocessor - PHP* é uma linguagem de *script open source* amplamente conhecida que se adequa especialmente ao desenvolvimento web e que pode ser utilizada dentro de HTML.

O PHP é compatível com qualquer tipo de sistema operativo, nomeadamente Linux, Microsoft Windows e Mac OS. Suporta ainda a maioria dos servidores web na atualidade, incluindo Apache, IIS, lighttpd e o nginx.

O PHP oferece ainda a possibilidade de desenvolver programação estruturada ou então programação orientada a objetos.

Em relação à comunicação com a base de dados, existe uma grande variedade de opções, destacando-se entre outras, MongoDB, MySQL, PostgreSQL, SQLite.[9]

Por forma a realizar uma gestão dos *packages* que vão sendo instalados e dos quais o projeto irá depender, será utilizada uma ferramenta que irá auxiliar na organização de todas as dependências do projeto.

Composer é uma ferramenta de gestão de dependências em PHP. Permite declarar as bibliotecas das quais o projeto depende e irá geri-las por nós, quer seja na sua instalação, *updates*, ou ao eliminar[10].

O Composer lida com *packages* ou com bibliotecas e realiza a sua gestão para o projeto em que se encontra inserido, fazendo as instalações num diretório, como por exemplo a pasta *vendor*, dentro do projeto. Por defeito, não realiza nenhum tipo de instalação globalmente, cingindo-se apenas ao projeto específico. Por este motivo, é visto essencialmente como um gestor de dependências.

Para poder utilizar o Composer, é necessária pelo menos a versão 5.3.2 de PHP. No caso da Framework Laravel, será necessária a versão 4 ou superior.

O Composer é apresentado no seu *website* aos utilizadores como é ilustrado na figura 3.1.

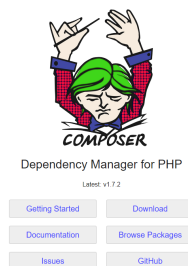


Figure 3.1: *Página oficial - Composer.*

Tendo em conta a elevada reputação da linguagem PHP, os *developers* centraram as suas atenções em desenvolver mecanismos e ferramentas que permitissem retirar o partido de todas as vantagens que o PHP tem para oferecer, mas focando-se nos pontos necessários para melhorar algumas ferramentas e também, para otimizar recursos da plataforma e de tempo e de estruturação de código.

As *frameworks* foram portanto uma mais valia para a melhorar todos estes pontos já mencionados e o Laravel segue essa mesma ideologia. O *website* da *framework* surge como ilustrado pela figura 3.2.

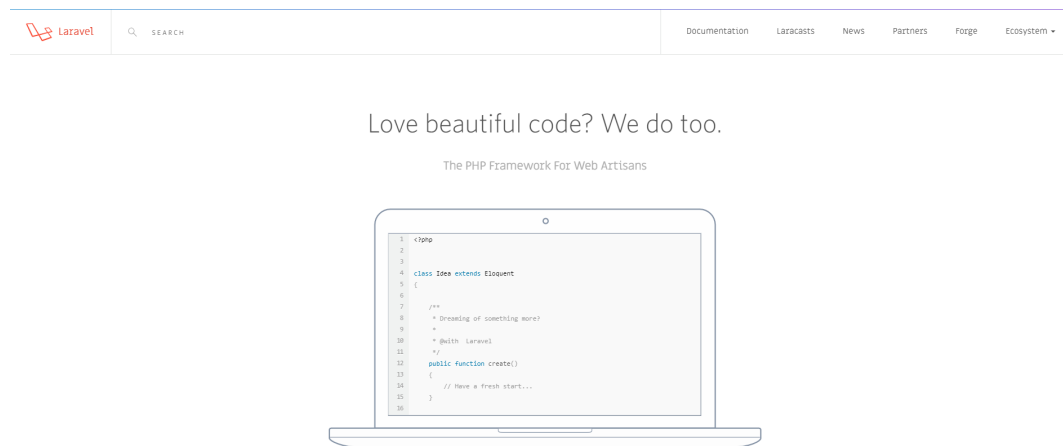


Figure 3.2: *Página oficial - Laravel.*

Laravel foi desenvolvido por Taylor Otwell e é uma *framework open-source* que permite o desenvolvimento de aplicações web seguindo o padrão de arquitetura *model-view-controller* (MVC).

A primeira *release*, ainda em versão beta, foi disponibilizada em Junho de 2011. Ainda no mesmo mês, foi lançada a primeira versão da *framework*, dada a conhecer como Laravel 1.

Depois de 7 anos de desenvolvimento, o Laravel foi-se tornando cada vez mais robusto e o crescente número de utilizadores ao longo dos tempos fizeram com que se tornasse na *framework* mais popular desde o ano de 2015 até à atualidade, à frente de Symfony2, Nette, CodeIgniter, Yii2, Zend e outras. A figura 3.3 ilustra o interesse na *framework* ao longo do tempo.

3.1. BACK-END (PHP - COMPOSER - LARAVEL FRAMEWORK)

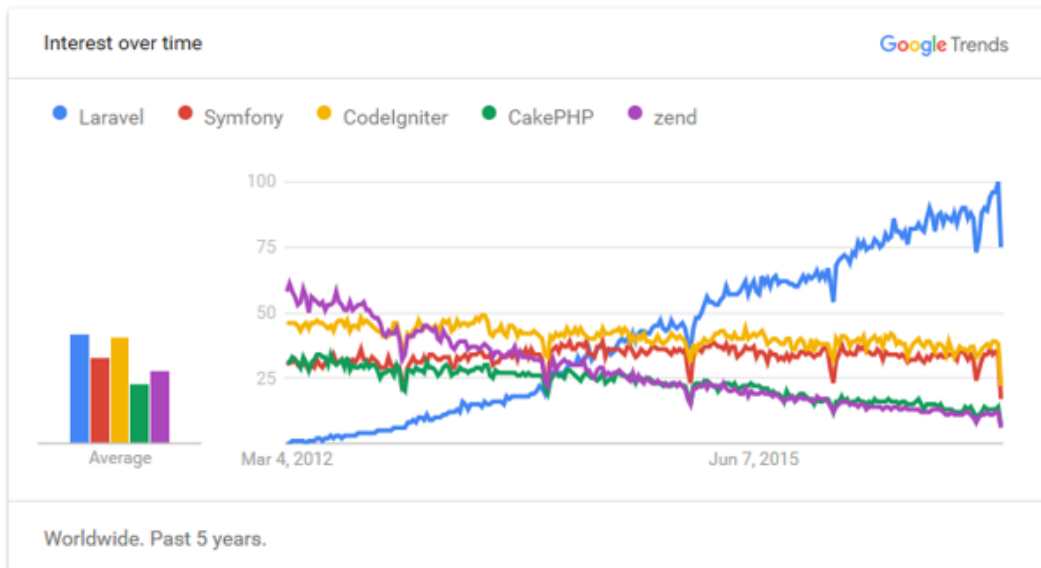


Figure 3.3: Interesse na framework em comparação com as concorrentes, ao longo do tempo.

PHP Framework Used for Project Use

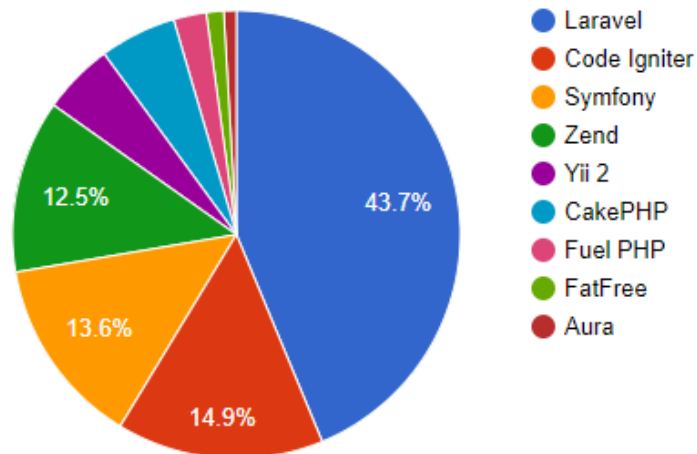


Figure 3.4: Percentagem de projetos encontrados por framework de PHP.

Através de uma pesquisa nas *trends* da Google dos últimos 5 anos, é possível verificar a crescente evolução do interesse que a *framework* Laravel veio a obter ao longo do tempo até ao presente ano.

Um estudo da *coderseye* [11] demonstra isso mesmo e acrescenta num estudo realiz-

ado, que foi aberto um inquérito a 7500 utilizadores de *frameworks* PHP para definir qual seria aquela cuja utilização era a mais comum em termos de autenticação, código de sessão, métodos de *cache* e *routing*. Os resultados apresentados referem-se no entanto ao uso de uma forma geral por esta amostra, no presente ano e é visível que na esmagadora maioria, os utilizadores preferem Laravel com 43.7% de utilização às outras *frameworks*. Vêm de seguida frameworks como Code Igniter (14.9%), Symphony (13.6%) e Zend (12.5%). As principais vantagens do Laravel são as seguintes:

- Maior organização de ficheiros e do próprio código;
- Desenvolvimento de aplicações mais eficaz;
- Utiliza a versão mais recente de PHP (7.0) e a arquitetura MVC;
- Testes unitários;
- Melhor documentação por comparação com as restantes frameworks;
- Abstracionismo de grande nível;
- Recursos de sobrecarga usando métodos dinâmicos;
- Diversas funcionalidades disponíveis *out of the box*;
- Integração com métodos de pagamento com o Stripe;
- Pacotes de encriptação muito fortes;
- Mapeamento de objeto relacional (ORM) muito intuitivo - Eloquent.

A versão mais recente desta *framework* à data de início do projeto é a 5.6 e data o seu lançamento a 7 de Fevereiro de 2018. Nota: Foi lançada a versão 5.7 de Laravel no dia 4 de Setembro de 2018.

3.2 Base de Dados (MySQL)

A comunicação com a base de dados para o desenvolvimento do projeto recaiu sobre MySQL.

Esta escolha deve-se em parte à escolha do XAMPP como ambiente de desenvolvimento PHP para a aplicação web.

O ambiente de desenvolvimento XAMPP permite replicar um servidor web localmente, ou seja, desenvolver aplicações web no próprio computador para mais tarde poderem ser lançadas online.[12]

A sigla XAMPP refere-se aos serviços oferecidos pela plataforma:

- X: uma ideologia de referência à sua utilização multi-plataforma;
- A: Apache HTTP Server Project, que permite uma forma segura e eficiente de oferecer serviços HTTP;
- M: MariaDB, onde anteriormente era utilizado MySQL (até 19-10-2015);
- P: PHP;
- P: PERL.

Sendo este ambiente de desenvolvimento o mais popular atualmente, de fácil acesso, gratuito e de fácil instalação, a escolha recaiu sobre este pacote.

Para além destes motivos, outro fator a favor desta decisão foi a forte aceitação de MySQL como uma das mais populares sistemas de bases de dados relacionais usados hoje na *web* [13]. Algumas das vantagens a destacar são as seguintes:

- MySQL é fácil de utilizar, contudo é extremamente poderoso, rápido, seguro e escalável;
- MySQL é compatível com uma variada gama de sistemas operativos, incluindo UNIX ou Linux, Microsoft Windows, Apple Mac OS X, entre outros;
- MySQL suporta a normal SQL - Structured Query Language;
- MySQL é a solução de base de dados ideal para pequenos e grandes aplicações;
- MySQL é atualmente desenvolvido e distribuído pela Oracle Corporation;
- MySQL inclui camadas de segurança de dados que protegem os dados mais sensíveis de intrusos.

3.3 Front-End (HTML, CSS, Javascript)

De um modo geral, as linguagens que complementam o *front-end* de um projeto com interface web, não variam muito. Tendo em conta que o projeto apresentado centra-se maioritariamente no *back-end* e nas suas funcionalidades, a parte de *front-end* será realizada tendo por base as seguintes tecnologias:

- HTML5: Hypertext Markup Language - Linguagem de estruturação e apresentação do conteúdo. A quinta versão existe desde 2014 e promove alterações essencialmente ao nível de novas API's; controlo de conteúdos multimédia; melhoramentos no uso offline e melhoria no debugging[14].

- JavaScript: Linguagem de programação compilada, conhecida como linguagem de *scripting* para as páginas web. Permite uma experiência ao utilizador mais completa e versátil, com recurso a várias funcionalidades dentro da própria página[15].
- CSS3: Cascading Style Sheet - Utilizada para definir estilos para páginas web com efeitos de transição, imagens e outros que afetam todo o design das páginas web. Os *browsers* mais comuns já suportam esta nova versão de CSS, nomeadamente: Chrome, Opera, IE9, Safari e Firefox[16].
- Bootstrap: Permite a integração de HTML, JS e CSS de modo a desenvolver layouts responsivos. Isto significa que os layouts poderão ser desenvolvidos para ecrãs de qualquer tipo de resolução, sem que se desformatem e por consequência, tornam a experiência do utilizador mais fraca[17].
- JQuery: Trata-se de uma das mais rápidas, pequenas e ricas em características bibliotecas de JavaScript. Tornam ainda mais simples a manipulação dos componentes HTML do documento, a gestão dos eventos, animações bem como chamadas Ajax[18].

3.4 JIRA e Metodologia SCRUM

No caso da INDMEI e do projeto para realização de uma aplicação web para criação e gestão de encomendas e produtos especificamente, é utilizado o software JIRA da empresa Atlassian.

O software JIRA é utilizado especificamente para a gestão de projetos e rastreamento de possíveis problemas ou bugs no produto. É utilizado atualmente por mais de 75 mil clientes em 122 países pelo mundo. A versão do software utilizada será a Jira Software, que inclui o software base e ainda características da gestão de projeto Agile. Todo o conteúdo será armazenado no domínio `indmei.atlassian.net`.

O projeto seguirá a metodologia SCRUM. A metodologia SCRUM é uma estrutura na qual as pessoas podem abordar problemas adaptativos complexos, entregando, de maneira produtiva e criativa, produtos do maior valor possível. O SCRUM é considerada uma *framework* simples para colaboração efetiva de equipas em produtos complexos. Os co-criadores do Scrum Ken Schwaber e Jeff Sutherland escreveram um guia desta metodologia para explicar de forma clara e sucinta em que consiste. Este guia contém também a sua definição. Essa definição consiste nos seus papéis, eventos, artefatos e regras do SCRUM que os unem [19].

Dentro da metodologia SCRUM, o local onde se encontram todas as tarefas a realizar

é chamado de *Product Backlog*, e é fundamentalmente onde é possível analisar todos os requisitos necessários à realização de um projeto. O *Product Backlog* representa uma lista ordenada de tudo o que poderá ser necessário no produto e é a única fonte de requisitos para qualquer tipo de alteração ao produto. Qualquer alteração no produto, tem de constar no *Product Backlog* e não em variados locais.

O *Product Backlog* é criado a partir de *Epics*, que são gerados através de ideias que são demasiado densas para individualizar. Esses *Epics* formam depois ideias mais concisas e diretas que entram então no *Product Backlog*. Os produtos que fazem parte do *Product Backlog* podem ser divididos da seguinte forma:

- *User Stories* - A melhor forma de obter os requisitos para o projeto;
- *Technical Requirements* - Requisitos que têm de ser construídos para outras *User Stories* poderem ser construídas em cima;
- *Code Spikes* - Provas de conceito ou itens de pesquisa que têm de ser realizados para posteriormente desenvolver uma *User Story* ou *Technical Requirement*;
- *Technical Debt* - Tomada de decisões quando uma falha ocorre num design do software e onde se torna necessário realizar alterações;
- *Bugs* - Erros que necessitam de resolução ao longo do desenvolvimento do produto.

Regra geral, uma pessoa é a responsável pela gestão do *Product Backlog* e é denominada de *Product Owner*. A metodologia SCRUM é considerada importante para o desenvolvimento deste projeto principalmente porque defende que para chegar ao projeto pretendido, é importante receber *feedback* constante daquilo que está a ser realizado. Este processo faz com que a cada iteração do projeto seja acrescentado valor ao mesmo. Uma vez que o *feedback* é transmitido constantemente ao longo do projeto, é possível assim atingir o objetivo pretendido (e que muitas vezes varia daquele transmitido inicialmente, quando o projeto estava ainda a ser idealizado). A figura 3.5 representa o funcionamento da framework previamente descrita.

SCRUM FRAMEWORK

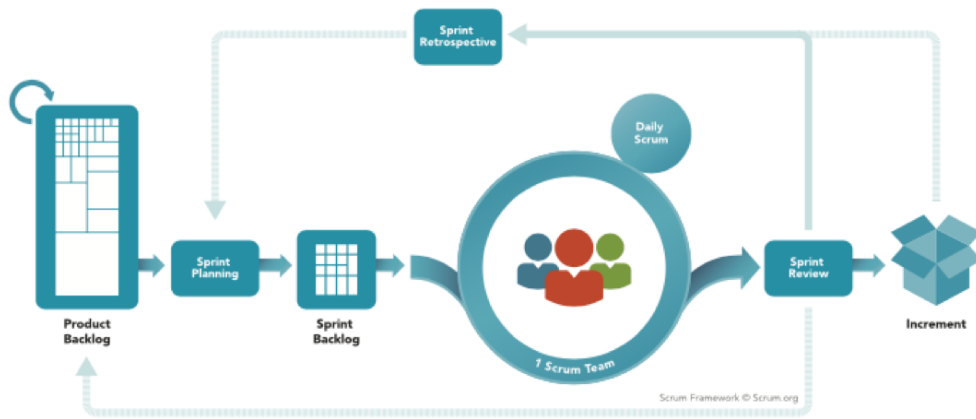


Figure 3.5: *Funcionamento da framework SCRUM.*

Neste capítulo foram apresentadas as linguagens e ferramentas utilizadas no desenvolvimento do projeto, em particular atenção na secção 5.3. Ao longo deste capítulo foram apresentados estudos do uso das diferentes linguagens e tecnologias que acabaram por suportar a escolha prévia para o desenvolvimento do trabalho.

4

Proposta de Protótipo

A proposta de protótipo apresentada segue uma estrutura composta pela aplicação web com os seus diversos tipos de utilizadores, bem como a ligação à base de dados onde serão armazenados todos os dados introduzidos. Esta relação terá comunicações nos dois sentidos, uma vez que por um lado, haverá chamadas para consulta de informação da base de dados e por outro, haverão dados para introduzir, editar ou eliminar a base de dados.

O esquema fica completo com a sua utilização através de qualquer um dos dispositivos disponíveis na rede de instalação de todo o sistema. Isto significa que o seu acesso por parte dos diversos utilizadores poderá ser realizado através de computadores, tablets ou telemóveis, desde que consigam aceder à rede da INDMEI.

A figura 4.1 representa as relações descritas anteriormente e a forma como se relacionam todos os componentes.

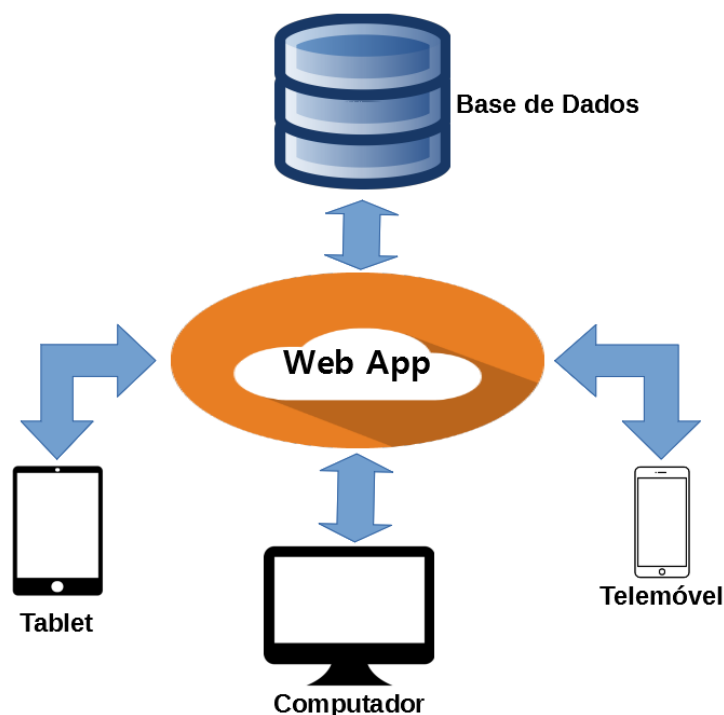


Figure 4.1: *Interação da aplicação web com a base de dados e os dispositivos.*

4.1 Análise de Requisitos

Depois de identificado o problema da ineficácia na gestão do armazém e do *stock* de cada matéria-prima, bem como a dificuldade em obter uma norma para o tratamento de processos na INDMEI, foram enumerados os diversos requisitos que a plataforma terá de possuir para colmatar as falhas encontradas.

O projeto concebido será uma plataforma de gestão de processos para permitir à empresa INDMEI otimizar as principais funções da empresa. Para cada uma das funcionalidades a implementar, são definidos os seus requisitos principais:

Gestão de Encomendas:

- Associar clientes a cada encomenda com um módulo adicional para gerir clientes. Também um módulo para gerir os fornecedores será de igual importância;
- Gerir os emails recebidos de clientes, bem como ser capaz de enviar emails;
- Alterar o estado de uma encomenda, de modo a fazer avançar ou recuar conforme necessário;

- Armazenar ficheiros de forma organizada que são associados a cada encomenda;
- Gerir o estado de produção das encomendas e observar os dados estatísticos.

Gestão de Armazém:

- Introduzir e consultar matérias-primas por quilos;
- Solicitar matéria-prima no momento em que fica abaixo do *threshold*;
- Gerir *stock* bruto e *stock* líquido;

Gestão de Amostras:

- Incluir o processo de criação de amostras de artigo que a empresa já realiza na aplicação web, para interligar com a gestão dos restantes processos;

Gestão de Orçamentos:

- Criar orçamentos que se baseiem na encomenda e no custo das matérias-primas utilizadas de forma automática;

Gestão de Permissões:

- Gerir os utilizadores da plataforma e verificar quais os utilizadores existentes;

4.2 Histórias de Utilizador

Seguindo aqui a terminologia identificada no capítulo 3.4 quando mencionada a metodologia usada para desenvolver o projeto, ou seja, o SCRUM, as Histórias de Utilizador, geralmente denominadas pela terminologia em inglês *User Stories* foram desenvolvidas no sentido de abranger todas as funcionalidades pedidas e pensando na possibilidade de desenvolver novas necessidades da empresa no futuro. Foram depois criadas tarefas para cada uma das *stories*, que posteriormente foram divididas em *Sprints*, ou seja, tempos específicos de desenvolvimento.

O exemplo de uma *user story* desenvolvida é apresentado de seguida:

”Como administrador, eu devo possuir a opção de listar todos os utilizadores da plataforma e ter uma opção de definir quais as permissões que cada um deles possui.”

Todas as User Stories poderão então ser agrupadas pelas características comuns entre elas. O quadro da tabela 4.1 apresentado demonstra como foram agrupadas essas mesmas stories e, posteriormente, desenvolvidas num período de tempo previamente definido.

Table 4.1: *Tabela de User Stories.*

Sprint 1	Sprint 2	Sprint 3
01/05/2018 a 20/06/2018	22/06/2018 a 01/07/2018 e 15/07/2018 a 31/07/2018	01/08/2018 a 20/08/2018
Gestão de Utilizadores e Permissões;	Bugfixes (adicionados após nova reunião com clientes);	Bugfixes (adicionados após testes nas tarefas já realizadas);
Gestão do Armazém (Stock);	Gestão de uma Encomenda;	Gestão de Encomendas em Produção;
Gestão de Amostras de Artigos;	Gestão de Fornecedores; Gestão de Clientes;	Gestão de Orçamentação; Gestão de Emails;

Por forma a executar todas as tarefas no tempo estipulado, foi realizado um exercício de número de horas previsto para cada tarefa e, em seguida, foram estimadas as horas de desenvolvimento nos períodos de Sprint necessárias.

O objetivo passa por permitir um fluxo de desenvolvimento o mais otimizado possível, sem que o tempo apertado de desenvolvimento prejudique a qualidade das funcionalidades apresentadas.

4.3 Tipos de Permissões a Implementar

Por forma a separar as permissões e as principais responsabilidades de cada utilizador da plataforma, foi criada uma estrutura de Roles para que desta forma todas as funcionalidades ficassem atribuídas ao trabalhador correto.

Os Roles estão então definidos da seguinte forma:

- **Admin:** O Admin será capaz de aceder a qualquer funcionalidade presente na plataforma. É um tipo de utilizador que estará disponível apenas para *developers* e responsáveis pela manutenção correta da plataforma;
- **Convidado:** O Convidado é um tipo de utilizador que apenas terá acesso à plataforma e na qual poderá registar-se e realizar login. Contudo, não terá acesso a qualquer funcionalidade;
- **Gestor de Amostra de Artigo:** O utilizador que possua esta permissão, será capaz de gerir todo o processo de criação, edição, listagem e eliminação de amostras de artigos;
- **Gestor de Armazém:** O gestor de armazém será capaz de controlar todas as funcionalidades relacionadas com o *stock* da empresa. Fazem parte desta permissão as ações de criar nova matéria prima, listar, editar e eliminar

matérias primas do armazém. Existe ainda a possibilidade de dar entrada de *stock* através de uma lista de matérias-primas, todas de uma só vez;

- **Gestor de Encomenda:** O utilizador que possua a permissão de gerir encomendas, terá também associadas a si a gestão de fornecedores e de clientes (criação de fichas de fornecedores/clientes, edição dos seus dados, listagem e eliminação). Terá também acesso à produção da encomenda à medida que é realizada. Para além disto será capaz de criar novas encomendas e também de listar, editar e eliminar encomendas já existentes;
- **Gestor de Orçamentação:** O utilizador com este tipo de permissão, será capaz de Listar as encomendas existentes e produzir orçamentos para cada uma das encomendas. Será também capaz de listar os orçamentos já existentes, editá-los ou eliminá-los. Por forma a agilizar o processo de orçamentação e de comunicação com os clientes, também possui as funcionalidades de criar e-mails e de abrir diretamente a caixa de entrada de emails através da plataforma;
- **Operário:** O operário terá acesso à lista de encomendas criadas, para desta forma poder adicionar a produção diária de determinada encomenda à medida que é realizada.

As funcionalidades implementadas de acordo com a permissão que faria mais sentido que tivesse acesso às mesmas. Deste modo, nem todos os utilizadores terão acesso às mesmas funções como se verifica na lista acima, sendo que apenas uma permissão permite aceder a todos os módulos em simultâneo (Admin) e uma outra permissão não terá acesso a módulo algum (Convidado).

4.3.1 Admin

Tal como já foi referido, a permissão Admin é a única capaz de aceder a todas as funcionalidades. Contudo, a principal funcionalidade que esta permissão permite e que mais nenhuma o faz, está relacionada com a gestão de utilizadores e de permissões.

Assim, um utilizador que seja Admin poderá, para além de todas as funcionalidades que os restantes tipos de permissões possuem, permite ainda realizar as funcionalidades descritas na figura 4.2.

4.3. TIPOS DE PERMISSÕES A IMPLEMENTAR

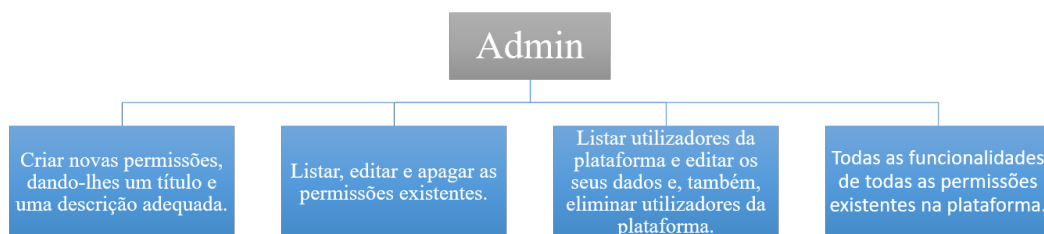


Figure 4.2: *Funcionalidades da permissão Admin.*

4.3.2 Convidado

A permissão Convidado serve como uma permissão básica, apenas para aceder à plataforma através do registo e login na mesma. É uma permissão base para qualquer utilizador novo da plataforma.

4.3.3 Gestor de Encomendas

A permissão Gestor de Encomendas é a principal responsável por todas as encomendas solicitadas na empresa. Será ainda responsável pela manutenção da Gestão de Fornecedores e de Clientes da empresa. Um utilizador que possua esta permissão terá acesso às funcionalidades da figura 4.3.

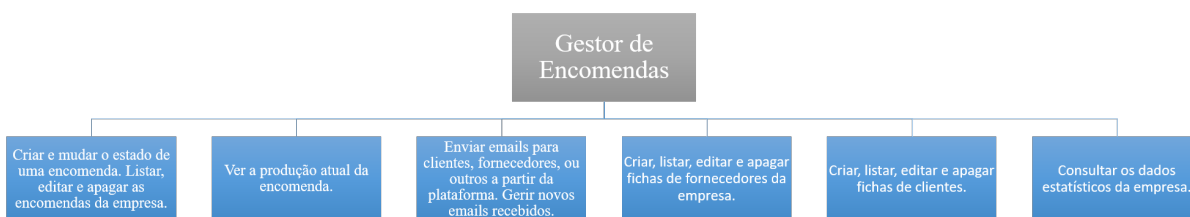


Figure 4.3: *Funcionalidades da permissão Gestão de Encomenda.*

4.3.4 Gestor de Amostra de Artigo

A permissão Gestor de Amostra de Artigo permite verificar as encomendas existentes e a partir delas trabalhar na amostra de artigo respectiva.

As funcionalidades são então listadas da forma apresentada da figura 4.4.



Figure 4.4: *Funcionalidades da permissão Gestão de Amostra de Artigo.*

4.3.5 Gestor de Orçamentação

O a permissão Gestor de Orçamentação será capaz de realizar todas as operações relacionadas com a orçamentação de uma encomenda. Terá ainda acesso às encomendas da empresa, para que a partir das mesmas, possa então recolher todos os dados necessários.

As funcionalidades listadas estão representadas pela figura 4.5.

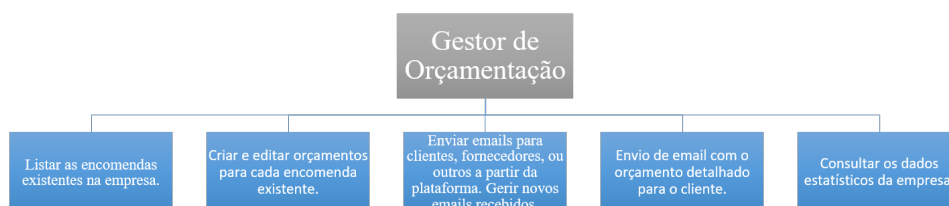


Figure 4.5: *Funcionalidades da permissão Gestão de Orçamentação.*

4.3.6 Gestor de Armazém

A permissão Gestor de Armazém terá acesso às funcionalidades necessárias para a gestão de stocks. O gestor não é o responsável pelo pedido de mais matéria-prima (esse papel pertence ao gestor da encomenda), pelo que o envio de emails não se torna uma prioridade para esta permissão.

A lista de funcionalidades presentes no utilizador com a permissão Gestor de Armazém são identificadas pela figura 4.6.

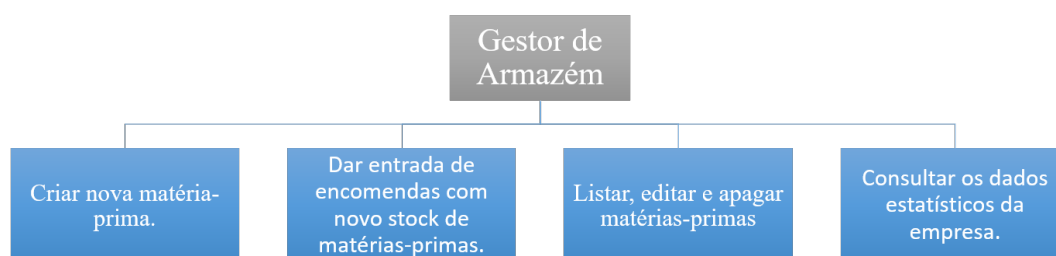


Figure 4.6: *Funcionalidades da permissão Gestão de Armazém.*

4.3.7 Operário

A permissão Operário é certamente aquela com menos permissões. Contudo, é aquela que torna possível que as restantes tenham variações diárias de dados. Esta permissão, identificada na figura 4.7 tem como principais funcionalidades as descritas.

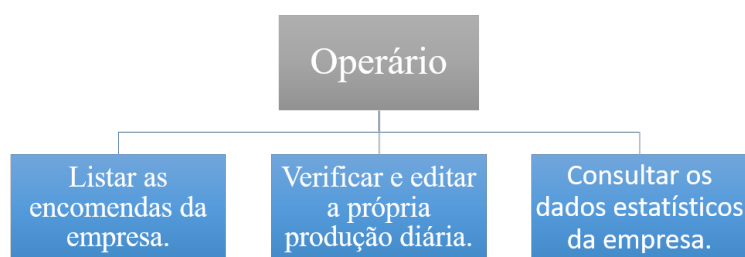


Figure 4.7: *Funcionalidades da permissão Operário.*

4.4 Fluxograma do Workflow dentro da INDMEI

A empresa INDMEI segue um fluxo de trabalho específico desde o momento em que é realizado um pedido de uma amostra de encomenda, até que a mesma é expedita para o seu cliente final.

No fluxograma a seguir apresentado, é demonstrado o ciclo do produto para uma melhor demonstração de todos os passos que são levados a cabo pelas pessoas intervenientes na produção da encomenda.

Também desta forma, torna-se mais fácil compreender de que modo todas as permissões do software se relacionam.

4.4. FLUXOGRAMA DO WORKFLOW DENTRO DA INDMEI

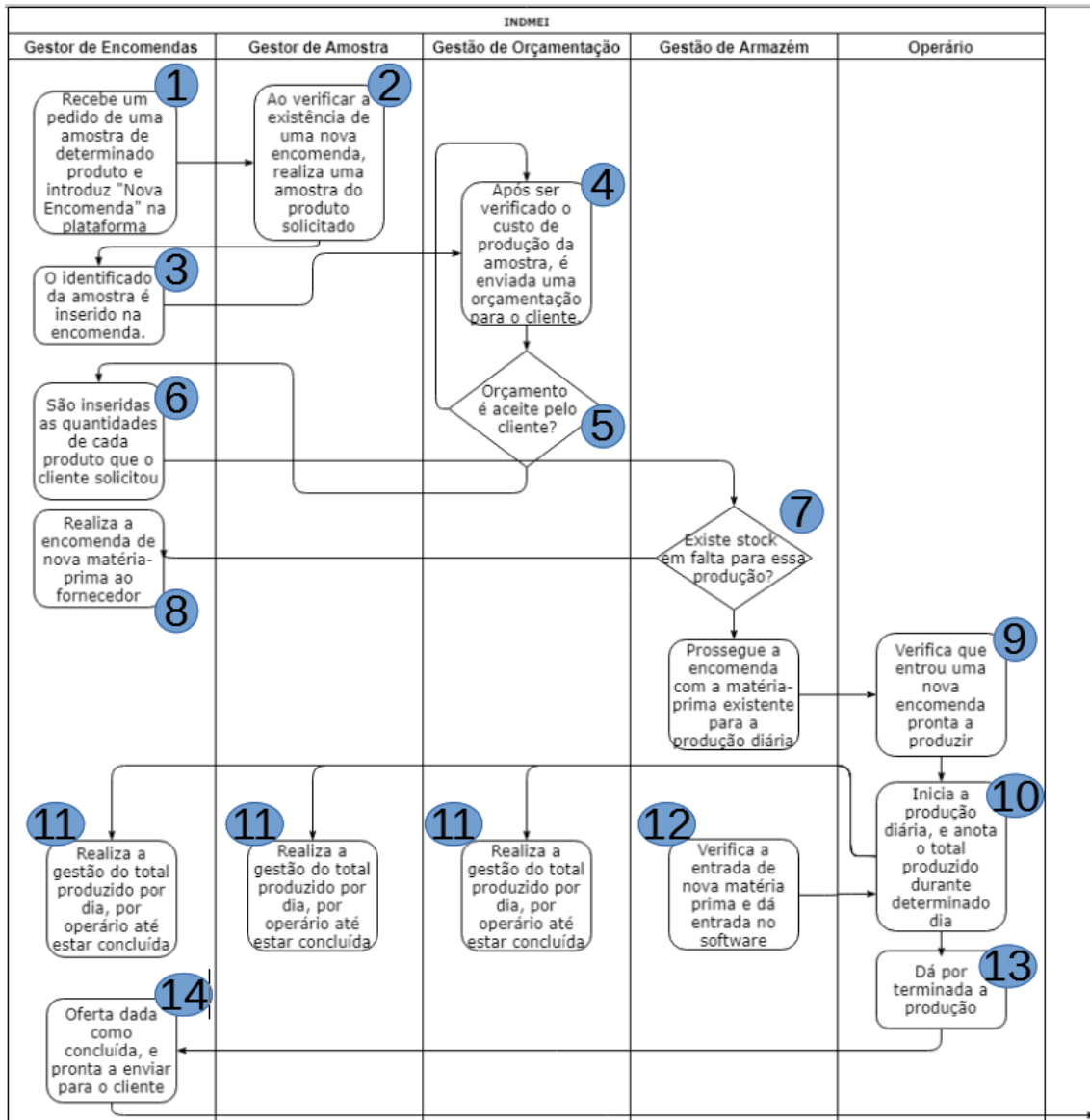


Figure 4.8: Fluxograma com o ciclo do produto da empresa INDMEI.

Tendo por base o fluxograma do ciclo de uma encomenda da figura 4.8, é possível verificar que uma encomenda inicia-se através da chegada de um pedido de um fornecedor ao gestor de Encomendas (1). É neste momento que uma encomenda é introduzida na plataforma, mais especificamente na lista de encomendas (1). Determinadas informações base acerca da encomenda são inseridas e é então passada toda essa informação ao gestor de amostra da encomenda (2).

O gestor de amostra da encomenda irá realizar uma amostra do produto, na qual irá identificar os materiais necessários e as suas quantidades. Irá também definir a

maquinaria necessária e em que momento são utilizadas na produção de uma meia. Uma vez determinada a amostra, é necessário realizar um certo número de pares de meias para fazer chegar ao cliente final para que sejam verificadas. Assim que a amostra esteja aprovada, o gestor de encomenda tem de associar a amostra correta à encomenda (3). Este processo permite que uma determinada amostra possa ser reutilizada por diversas encomendas, caso seja necessário.

Uma terceira entidade (4), responsável pela gestão de orçamentação irá fazer chegar um orçamento ao cliente, geralmente através do envio de email da plataforma onde irá contemplar todos os custos que resultam da produção da encomenda solicitada pelo cliente: valor total de uma amostra multiplicada pelo número de pares pedidos; etiquetas; peças com defeito; mão-de-obra; outros custos.

Assim que o orçamento é aceite pelo cliente (5), são então colocadas as quantidades finais de produto (6). Neste momento, o gestor de armazém tem acesso à informação recebida após introdução da encomenda, se terá necessidade de requerer mais *stock* ou se por outro lado, existe *stock* suficiente para avançar com a encomenda (7). O gestor de armazém tem ainda a possibilidade de verificar se existe quantidade suficiente para a produção diária ir avançando enquanto é esperada a chegada de mais matéria prima, o que permite maior eficácia na produção da encomenda. Caso seja, efetivamente necessário requisitar mais matéria prima, esta informação será passada ao gestor da encomenda que fará este pedido ao fornecedor pretendido (8).

Assim que a encomenda esteja pronta a realizar, o operário irá verificar o acesso à mesma e irá dar início à sua produção (9). Este processo passa por um registo diário do valor produzido e do tipo (10). O processo de produção dos operários será controlado pelos utilizadores com permissões de gestor de produção, gestor de amostra e gestor de orçamentação (11). Enquanto o processo de produção ocorre, o gestor de armazém está responsável pela verificação de nova entrada de matéria-prima que será utilizada na encomenda em questão, e que dará a sua entrada em armazém (12). Uma vez atingidos o total de meias necessárias, a encomenda ficará pronta a entregar ao cliente(13).

Assim que a produção esteja terminada, o gestor da encomenda dá a mesma como concluída e pronta a enviar para o cliente (14).

4.5 Atores e Casos de Uso

A maioria dos problemas encontrados em sistemas orientados a objetos tem sua origem na construção do modelo, ou seja, no desenho do sistema. Por vezes as empresas não dão muita ênfase à essa fase do projeto e acabam por cometer diversos

erros de análise e modelagem.

O UML (acrónimo para a expressão Unified Modeling Language) representa uma linguagem que define uma série de artefatos e que ajuda na tarefa de modelar e documentar os sistemas orientados a objetos que são desenvolvidos. O diagrama de casos de uso representa aquilo que o sistema é capaz de fazer, do ponto de vista do utilizador. Por outras palavras, irão ser descritas as principais funcionalidades do sistema e a sua interação com os utilizadores do mesmo. O objetivo do diagrama não será mergulhar nos detalhes mais técnicos de como é que o sistema faz a tarefa, mas sim representar a tarefa em si.

O diagrama de casos de uso é representado pelos seguintes atores:

- Administrador
- Convidado
- Gestor de Encomendas
- Gestor de Amostra de Artigo
- Gestor de Armazém
- Gestor de Orçamentação
- Operário

Se verificarmos os casos de uso de cada um dos atores, estes podem ser definidos pelas tabelas 4.3 e 4.2:

Após a análise da lista de casos de uso, o diagrama da figura 4.9 representa estes mesmos casos do ponto de vista dos diversos utilizadores da plataforma.

Table 4.2: *Tabela com Atores e Casos de Uso (1)*

Ator	Use Cases
Gestor de Armazém	Dar Entrada de Stock Criar Nova Matéria-Prima Listar Stock de Armazém Dar Saída de Stock Pedir novo Stock Editar detalhes da Matéria-Prima Apagar Matéria-Prima Verificar estado da encomenda Verificar dados Estatísticos
Gestor de Orçamentação	Ver detalhes da Encomenda Listar Encomendas Enviar Email Gerir Emails Criar Orçamento Editar Orçamento Verificar estado da encomenda Verificar dados Estatísticos
Operário	Listar Encomendas Criar entrada de Produção Editar entrada de Produção do próprio dia Verificar estado da encomenda Verificar dados Estatísticos

Table 4.3: *Tabela com Atores e Casos de Uso(2)*

Ator	Use Cases
Administrador	Criar nova permissão Listar permissão Editar permissão Apagar permissão Listar utilizadores Editar permissões de utilizador Apagar utilizadores Todas as funcionalidades listadas abaixo existentes na plataforma
Convidado	Nenhuma funcionalidade excetuando o Login
Gestor de Encomenda	Criar nova encomenda Listar encomendas Ver detalhes da encomenda Editar encomenda Apagar encomenda Verificar estado da encomenda Enviar Email Gerir Emails Criar Fornecedor Listar Fornecedores Editar Fornecedor Apagar Fornecedor Criar Cliente Listar Clientes Editar Cliente Apagar Cliente Verificar dados Estatísticos
Gestor de Amostra de Artigo	Criar Amostra de Artigo Listar Amostras de Artigo Ver detalhes da Amostra de Artigo Editar Amostra de Artigo Apagar Amostra de Artigo Ver detalhes da Encomenda Listar Encomendas Enviar Email Gerir Emails Verificar estado da encomenda Verificar dados Estatísticos

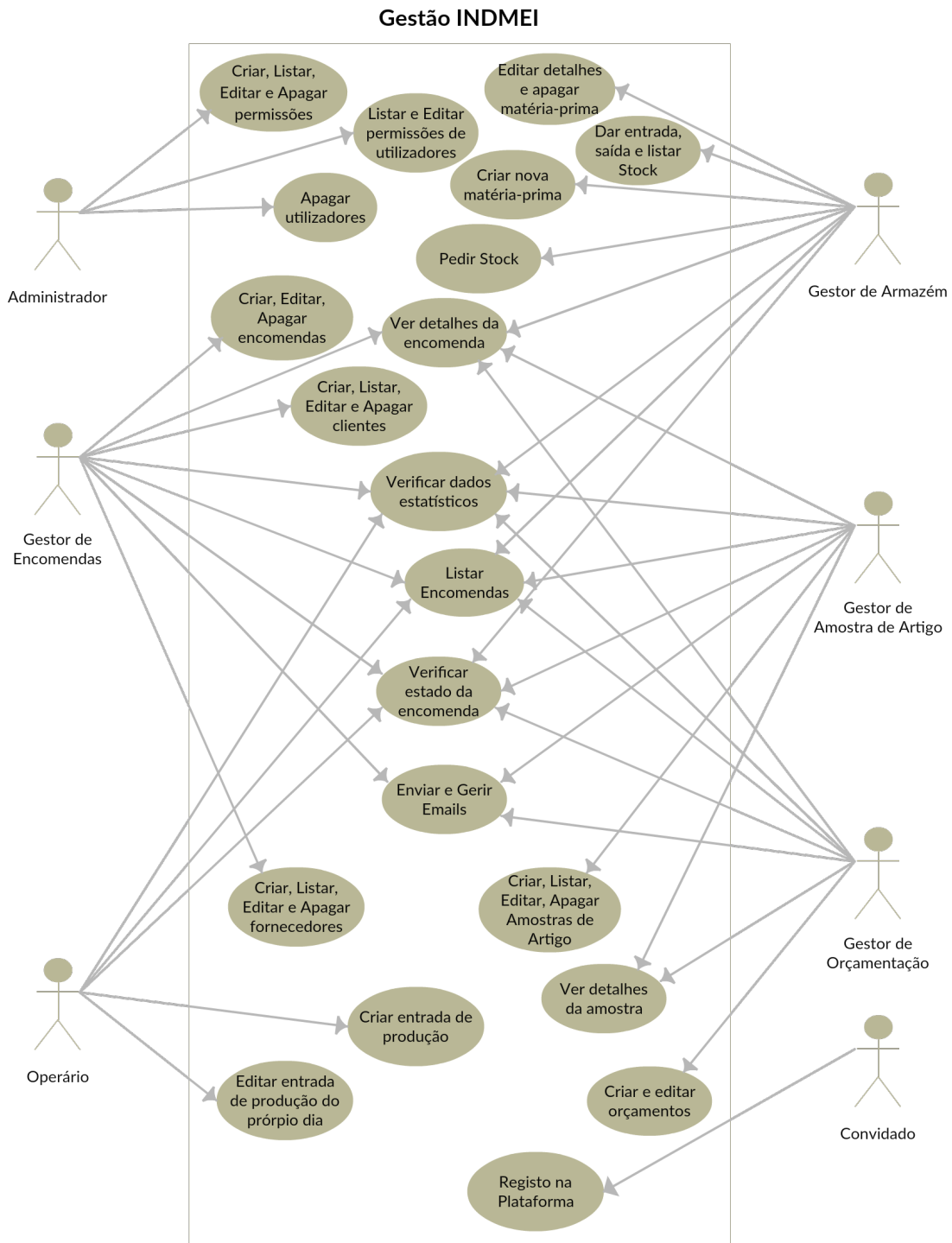


Figure 4.9: Diagrama de casos de uso.

Através do diagrama (4.9), é possível verificar que algumas das funcionalidades são comuns a diversos tipos de atores. Isto deve-se ao facto de vários atores depend-

erem do estado em que uma determinada encomenda se encontra para se verificar se será a sua vez de intervir no processo de produção da mesma.

Outras situações que contribuem para as diversas funcionalidades comuns a vários atores são o facto de vários necessitarem de enviar email ou consultar emails através da plataforma, ou ainda o facto de existirem estatísticas através de uma interface gráfica que é disponibilizada a qualquer tipo de utilizador.

5

Desenvolvimento da aplicação

Este capítulo pretende focar concretamente o projeto desenvolvido, dando foco às funcionalidades de maior relevo.

A sua organização parte inicialmente da base de dados, das relações nela existentes. A forma mais prática de realizar esta análise será através de um diagrama Entidade Relação (EER) que será explicado no subcapítulo 5.1.1.

Segue depois uma análise das funções implementadas, passando um pouco pelo funcionamento do modelo MVC e das principais funcionalidades utilizadas da framework Laravel. Serão apresentadas as classes desenvolvidas e ainda as bibliotecas JavaScript utilizadas.

Por fim, ainda serão demonstradas as notificações via email implementadas e em que casos é que estas são despoletadas.

5.1 Estrutura e relações da Base de Dados

O software MySQL Workbench oferece a possibilidade de criar este tipo de diagramas, quer a partir das tabelas da base de dados pré-existentes ou mesmo de novo. É possível depois criar as diferentes relações um-para-um, um-para-muitos e muitos-para-muitos, as ferramentas fornecidas, criando de seguida o campo relacional de cada tipo de relação.

5.1.1 Modelos de Dados (Diagrama EER)

O diagrama EER (Enhanced Entity-Relationship) foi gerado por forma a ter uma visão geral de como são os relacionamentos (do inglês *relationships*) entre as diversas tabelas da base de dados. Todas as tabelas estão relacionadas entre si de modo a evitar o armazenamento de informação em duplicado e também para estruturar melhor toda a informação adquirida.

Como é possível verificar pelo diagrama, todas as tabelas têm uma relação direta, excetuando a tabela *suppliers*, que pode passar por uma relação de um-para-um, um-para-muitos, ou muitos-para-muitos. Tendo em conta o sentido da esquerda para a direita do diagrama, é possível distinguir as tabelas pelo tipo de dados e por permissão que cada utilizador tem dentro da plataforma:

- Mais à esquerda: Permissões do Armazém, com as tabelas *warehouse_product_specs*, *warehouse_products* e *warehouse_products_history*;
- Na área central-esquerda : Permissões da amostra do Artigo, com as tabelas *sample_article_colors*, *sample_article_wires*, *sample_article_guiafios*, *sample_article_steps* e *sample_articles*;
- Na área central-direita: Permissões de Administrador (Utilizadores e Permissões), com as tabelas *users*, *role_user* e *roles*;
- Na área um pouco mais à direita: Permissões de Encomendas, com as tabelas *orders*, *order_files* e *order_statuses*;
- Na área à direita, na parte superior: Permissões de Operário, com a tabela *order productions*;
- Na área à direita, na parte inferior: Permissões de Orçamentação com a tabela *quotations*;
- No fundo do diagrama, são apresentadas tabelas auxiliares ao funcionamento da aplicação web: *suppliers*, *password_resets* e *migrations*.

A imagem 5.1 representa de um modo geral todas as relações entre todas as tabelas.

5.1.2 Relação do módulo dos Users e Permissões

Neste módulo estão representadas as tabelas *users*, *roles* e a tabela relacional *role_users*. Os campos da tabela que privilegiam a relação são destacados com o símbolo do losango vermelho. Esta relação permite armazenar toda a informação necessária dos utilizadores, e também toda a informação necessária das permissões criadas.

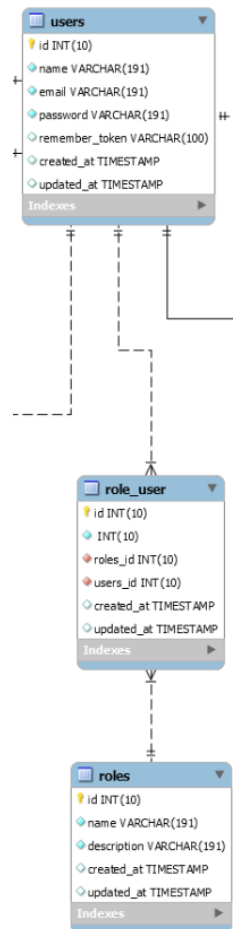


Figure 5.2: Relações do módulo de utilizadores.

Na figura 5.2, a relação que existe entre os Utilizadores e os Roles é de muitos-para-muitos. Isto permite que um utilizador possua várias permissões diferentes em simultâneo e também que outros utilizadores possam possuir essas mesmas permissões.

Isto prepara o *software* para que possa ser utilizado em ambientes pequenos, mas também para que seja aplicado a equipas maiores, com maior número de encomendas, maior número de operários.

A tabela central (role-user) é utilizada para armazenar os pares de identificadores do utilizador e da permissão que lhe é atribuída.

5.1.3 Relação do módulo da Amostra de Artigo

Este módulo apresenta a tabela com maior número de campos de todo o sistema implementado, a *sample_articles*. Isto porque terá de armazenar todas as informações de como uma amostra é criada desde ferramentas utilizadas, referência, descrição, tamanhos criados, custo de matérias-primas intervenientes e até uma imagem referencial. A esta tabela juntam-se quatro outras que se relacionam. Todas as relações apresentadas têm os campos identificados com um símbolo vermelho imediatamente antes do campo relacionado.

As quatro tabelas são responsáveis por armazenar informação das cores dos fios, de quais os fios utilizados e a sua quantidade, e ainda tabelas com todas as ferramentas de guiafios e todos os passos que uma produção de amostra poderá conter.

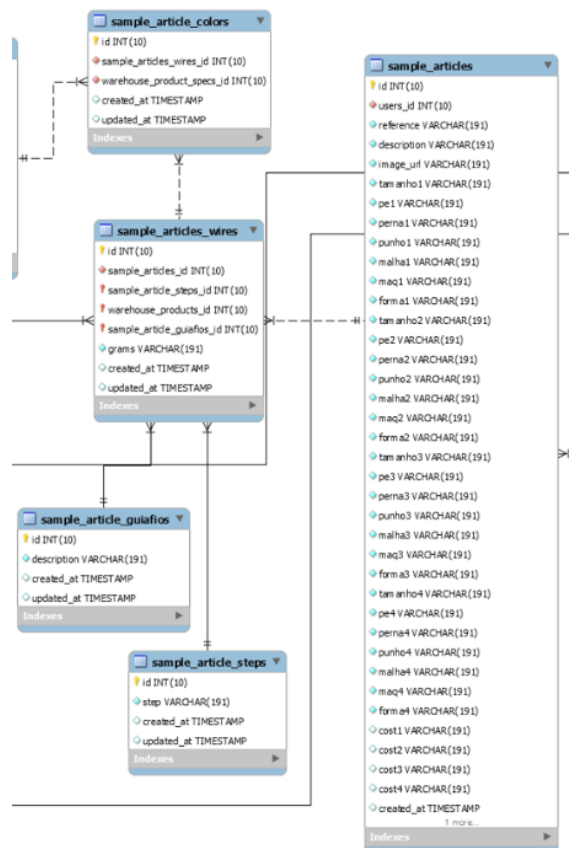


Figure 5.3: Relações do módulo das amostras de artigo.

Através da figura 5.3, é possível verificar que a funcionalidade que permite armazenar os dados referentes à amostra de artigo é das mais exigentes na medida em que contempla diversas informações distintas. É no fundo, onde são armazenados todos os passos para a produção de um produto, a ordem pela qual são realizados e as matérias primas que serão utilizadas.

A tabela principal (`sample_articles`), armazena a maioria dos valores. Contudo, possui uma relação de um-para-muitos com a tabela de fio a utilizar (`sample_article_wires`), uma vez que é utilizado mais do que um fio por amostra.

Cada fio utilizado na amostra necessitará de armazenar qual o passo da produção em que será utilizado e portanto, foi realizada uma relação de muitos-para-um com o passo respectivo (`sample_article_steps`).

Para cada passo da produção, é ainda necessário atribuir a ferramenta da produção a utilizar e, portanto, resulta daqui uma nova relação muitos-para-um com essa mesma ferramenta (`sample_article_guiafios`).

Da mesma forma, tendo em conta que um fio possui até quatro cores diferentes, resulta daqui uma relação um para muitos com a cor utilizada (`sample_article_colors`). Todos os fios estão diretamente ligados com o Armazém, módulo este que será explicado no subcapítulo seguinte.

5.1.4 Relações do módulo de Armazém

O módulo do armazém representa a informação guardada para toda a matéria-prima que é guardada em *stock*. Permite armazenar todas as especificações de cada matéria prima permanentemente atualizada, tais como a descrição, cor, *stock* líquido e bruto, o custo e o valor de aviso mínimo. A relação permite ainda obter um histórico detalhado de entrada e saída de matéria-prima.

O módulo do armazém é responsável pelo armazenamento de produtos que chegam à empresa, representado na figura 5.4. É também através deste módulo que a empresa, mais concretamente um gestor de armazém, poderá controlar se uma matéria prima existe em quantidade suficiente para a produção de determinada encomenda.

Os valores de *stock* são armazenados em *stock* bruto e em *stock* líquido:

- *Stock* Bruto: Corresponde ao *stock* armazenado. Este stock é atualizado no momento exato em que um operário introduza a quantidade de produtos que produz naquele dia de trabalho. Este valor pretende refletir qual a quantidade efetiva de produto que existe no armazém, todos os dias.
- *Stock* Líquido: Corresponde ao *stock* armazenado, subtraindo as quantidades totais das encomendas listadas para produção.

Existe ainda um histórico de armazenamento e de saída de produto, para desta forma o gestor consiga realizar estudos de gastos e custos, quantidades de entrada e saída de produtos de modo a vir a otimizar todas estas variáveis associadas à gestão de *stocks*.

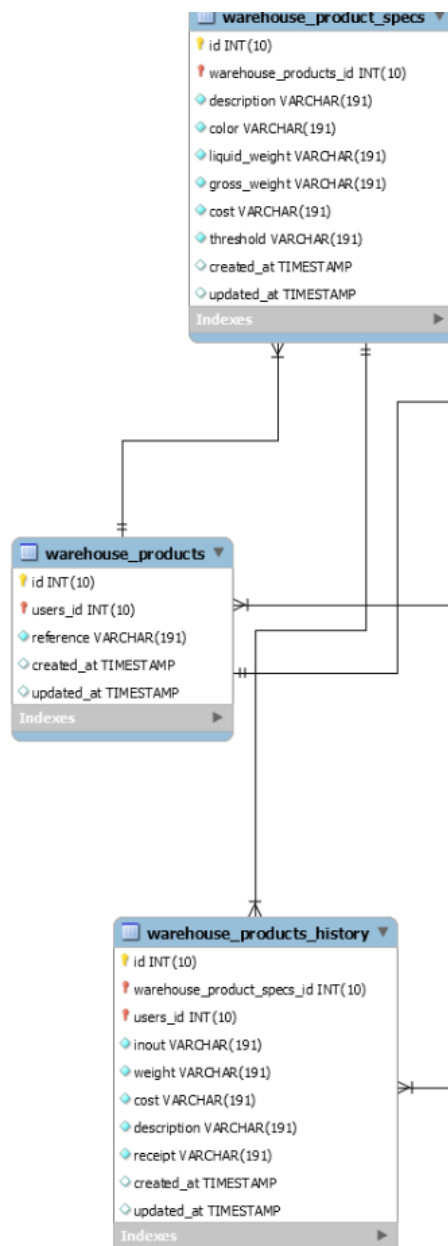


Figure 5.4: Relações do módulo de Armazém.

5.1.5 Relações do módulo de Encomendas, Orçamentação e Operário

O módulo de encomendas é o módulo que irá gerir todo o processo da encomenda, desde que é feito o seu pedido, até que a encomenda sai da produção.



Figure 5.5: Relações do módulo de Encomendas, Orçamentação e Operário.

É na tabela "orders" onde ficarão armazenadas a maioria das informações acerca da encomenda, desde quantidades solicitadas pelo cliente, data de entrega, cores principais da encomenda.

Existe uma relação de muitos-para-um com os clientes da empresa, sendo que um cliente poderá ter várias encomendas em simultâneo.

Existe também uma relação de um-para-muitos com os ficheiros anexados a cada encomenda (order_files), uma vez que cada encomenda pode ter diversos ficheiros auxiliares que poderão ser guardados juntamente com a encomenda.

Este módulo está representado em maior detalhe na figura 5.5. Para verificar o estado da encomenda a cada passo, é utilizada uma tabela auxiliar (order_statuses) que possui uma relação muitos-para-um, uma vez que um estado poderá ocorrer em várias encomendas ao mesmo tempo.

Existe depois aqui uma relação um-para-muitos com a produção da encomenda. Esta relação existe para que cada operário, a cada dia, possa introduzir as quantidades realizadas de determinada encomenda.

Uma relação um-para-um ocorre entre a encomenda e o seu orçamento. Isto porque uma encomenda possui apenas um e só um orçamento que lhe pertence. Para qualquer nova encomenda, um orçamento novo tem de ser criado.

5.1.6 Relações dos módulos Auxiliares

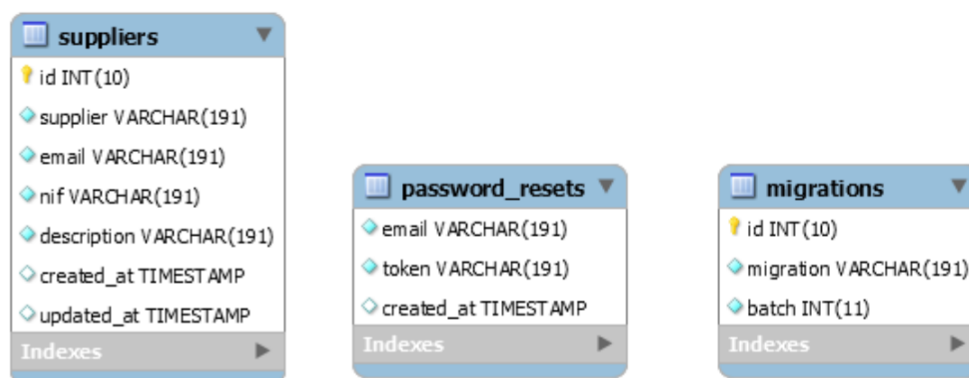


Figure 5.6: Relações dos módulos Auxiliares.

Existem três módulos auxiliares (ver figura 5.6) que desempenham tarefas distintas:

- **suppliers:** Este módulo pretende armazenar informações relacionadas com os fornecedores da empresa. Serve de auxílio para uma melhor gestão de fornecedores, bem como contacto para que seja possível a troca de emails entre a empresa e os fornecedores de forma mais automatizada.

- **password_resets:** Este módulo, criado automaticamente pelo projeto laravel no momento da criação de autenticação, serve o propósito de armazenar tokens de reset de password no momento em que um utilizador se esquece da sua password.
- **migrations:** Este módulo é criado de forma automática pelo projeto Laravel pretendendo armazenar as migrações de um projeto. Cada registo na tabela representa umadas tabela da base de dados e todas as suas características.

5.2 Funções Implementadas

De modo a compreender melhor as funcionalidades implementadas, a figura 5.7 representa o diretório de pastas da aplicação web.

São destacadas e numeradas as pastas que contém os ficheiros de maior relevo para o projeto, sendo eles:

- **1 Http:** Tendo por base o modelo MVC, aqui estão contidos todos os *controllers* com todos os métodos desenvolvidos.
- **2 Mail:** Contém o template base do email enviado pelos utilizadores.
- **3 Models:** Tendo por base o modelo MVC, este ponto contém todos os Models necessários para o projeto.
- **4 database:** Este diretório contém as migrações para criação da base de dados e ainda os *seeders* que irão ser apresentados a seguir.
- **5 views:** Com base no modelo MVC, este diretório contém as views do projeto, onde é desenvolvido todo o *front-end* da aplicação.
- **6 web:** Este ficheiro permite verificar todas os diretórios, ou *routes* do projeto.
- **7 .env:** Neste ficheiro é possível analisar as ligações com a base de dados utilizada bem como as configurações do servidor de email.

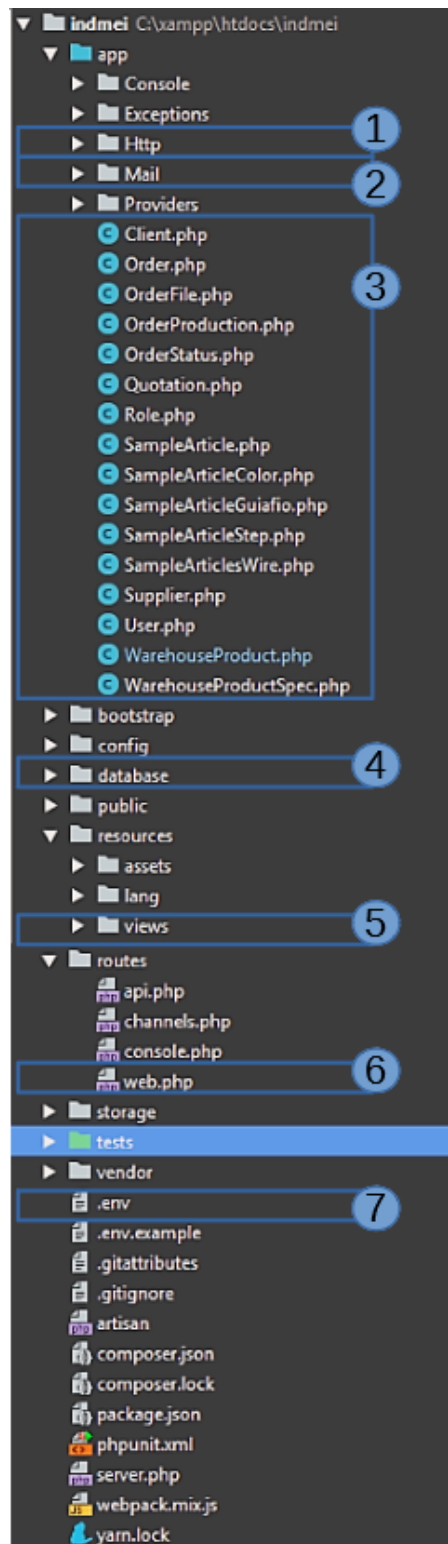


Figure 5.7: Diretório de pastas da aplicação web.

As funções implementadas ao longo do desenvolvimento da aplicação web utilizam "pacotes" adicionais que promovem uma utilização mais prática de determinadas funcionalidades bem como permitem uma maior otimização em termos de fluidez da plataforma e organização de código.

De entre as subsecções descritas a seguir, estas podem ser divididas em três diferentes implementações:

- Os capítulos **5.2.1**, **5.2.2**, **5.2.3** e **5.2.4** correspondem a *packages* Laravel que podem ser implementados recorrendo ao *Composer*, sendo que depois terão de ser cuidadosamente configurados e em certos casos criadas novas funções, para terem a finalidade pretendida;
- Os capítulos **5.2.5** e **5.2.6** recorrem a bibliotecas JQuery, ou seja, dependem do uso da *framework* JQuery para que funcione corretamente. Estas bibliotecas são bastante versáteis, contudo necessitam de diversas configurações, funções adicionais e manipulação de dados;
- Os capítulos **5.2.7**, **5.2.8** e **5.2.9** são funções ou classes desenvolvidas no decorrer da aplicação web que permitem auxiliar em diversos módulos desenvolvidos, recorrendo aos princípios da programação orientada a objetos e também ao modelo MVC.

5.2.1 Sistema de Login e Registo

Para criar o sistema de login a utilizar na plataforma, recorreu-se ao Artisan. O Artisan é o nome dado à interface da linha de comandos que vem incluída no Laravel. Esta interface permite uma vasta quantidade de comandos que assistem quando a aplicação está a ser desenvolvida.

É possível aceder a todos os comandos existentes através do comando:

php artisan list

Para além de listar todos os comandos, cada um destes possui uma secção de ajuda que permite apresentar e descrever cada um dos comandos individualmente e ainda os seus possíveis argumentos. Para visualizar a ajuda do comando *migrate*, por exemplo, é inserido o comando:

php artisan help migrate

Posta esta breve definição, para a criação do sistema de login, recorreu-se ao comando:

php artisan make:auth

Ao executar o comando acima mencionado, são criadas uma série de estruturas que auxiliam o sistema de login e registo. Como o Laravel se trata de uma estrutura que segue o padrão MVC, o *model Users* é criado, juntamente com um diretório de *controllers* chamado *Auth* (pode ser observado na figura 5.8) e ainda as *views* de login e registo.

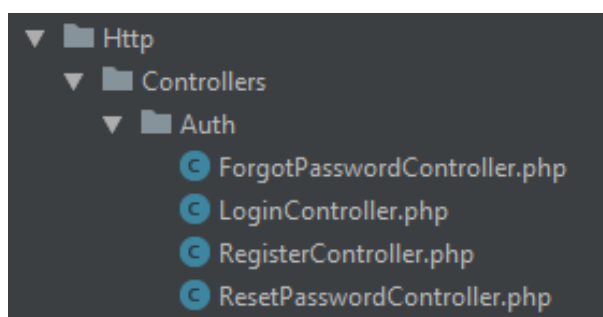


Figure 5.8: *Controllers criados com "php artisan make:auth".*

Os *controllers* de autenticação são compostos pelas quatro ações possíveis: Login, Registo, Esquecer Palavra-Passe e Recuperar Palavra-Passe.

O *model Users* é apresentado na figura 5.9, representa as suas definições. Trata-se de uma classe *filha* de uma outra classe *Authenticable* e portanto possuirá todas as suas propriedades. Nesta classe *Users*, é inserida também a *trait Notifiable* para que deste modo esteja preparada para a possibilidade de ser necessária a utilização de notificações para utilizadores, como por exemplo o envio de um email, ou de uma notificação para a plataforma slack, entre outras. Neste caso específico, esta *trait* seria desnecessária.

Por fim, são indicadas as variáveis utilizadas pelo *model* e que podem ser divididas sendo que o name, email e password são do tipo *protected fillable* e as variáveis password e remember_token são *protected hidden*. As variáveis do primeiro tipo são as únicas que poderão ser atualizadas via *mass assignment*, ou seja, são as únicas que poderão ser enviadas como fazendo parte de um array para depois serem atualizadas ou criadas.

```
<?php

namespace App;

use Illuminate\Notifications\Notifiable;
use Illuminate\Foundation\Auth\User as Authenticatable;

class User extends Authenticatable
{
    use Notifiable;

    /**
     * The attributes that are mass assignable.
     *
     * @var array
     */
    protected $fillable = [
        'name', 'email', 'password',
    ];

    /**
     * The attributes that should be hidden for arrays.
     *
     * @var array
     */
    protected $hidden = [
        'password', 'remember_token',
    ];
}
```

Figure 5.9: Model Users.

O comando mencionado para criação do sistema de login e registo cria ainda um ficheiro de migração (chamado *migration*) para os utilizadores. Este tipo de ficheiro é criado como um controlo da base de dados, ou seja, permite a qualquer pessoa que tenha acesso ao projeto, fazer alterações à base de dados quer ao nível de colunas, ordem das mesmas, adicionar ou remover novas colunas, bem como ao tipo de campo que irá guardar.

Torna-se especialmente útil quando o projeto é desenvolvido num ambiente distinto do ambiente de produção, uma vez que permite depois fazer o *deploy* da base de dados com apenas a utilização de um novo comando *artisan*.

Tendo como base a figura 5.10, as *migrations* criadas para gerir a autenticação de utilizadores referem-se à primeira e à segunda linha do diretório: *create_users_table* e *create_passwords_resets_table*.

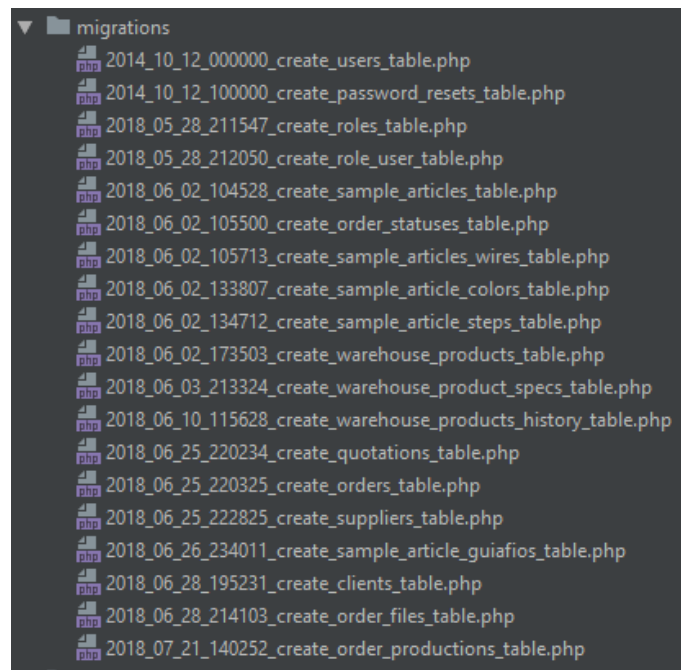


Figure 5.10: Migrations.

As *routes* associadas ao sistema de login e registo também ficarão disponíveis após a utilização do comando **php artisan make:auth** e encontram-se listadas na imagem 5.11. Cada *route* corresponde a um url disponibilizado depois ao utilizador para as mais diversas funcionalidades, tais como:

- **Route::get('/samples/list', 'SampleArticleController@index');** corresponde ao *url* /samples/list para listar as amostras de artigo;
- **Route::get('/samples/create', 'SampleArticleController@create');** corresponde ao *url* /samples/create para criar uma nova amostra de artigo;
- **Route::post('/samples/create', 'SampleArticleController@store');** corresponde ao *url* /samples/list, mas na qualidade de um pedido *POST*, que irá permitir armazenar os seus dados na base de dados e tabelas correspondentes.

Para cada uma das linhas, que representa uma *route* distinta, é possível verificar se se trata de um método GET ou POST e depois dentro de parêntesis o primeiro parâmetro diz respeito ao *url*; o segundo é o *controller* e o método para onde é redirecionado o utilizador.

```
public function auth()
{
    // Authentication Routes...
    $this->get('login', 'Auth\LoginController@showLoginForm')->name('login');
    $this->post('login', 'Auth\LoginController@login');
    $this->post('logout', 'Auth\LoginController@logout')->name('logout');

    // Registration Routes...
    $this->get('register', 'Auth\RegisterController@showRegistrationForm')->name('register');
    $this->post('register', 'Auth\RegisterController@register');

    // Password Reset Routes...
    $this->get('password/reset', 'Auth\ForgotPasswordController@showLinkRequestForm');
    $this->post('password/email', 'Auth\ForgotPasswordController@sendResetLinkEmail');
    $this->get('password/reset/{token}', 'Auth\ResetPasswordController@showResetForm');
    $this->post('password/reset', 'Auth\ResetPasswordController@reset');
}
```

Figure 5.11: *Routes de autenticação.*

Em termos de validações, no momento do registo são solicitados ao utilizador quatro campos: nome, email, password e confirmação de password. As validações são efetuadas recorrendo à classe *Validation* que vem na instalação de um novo projeto Laravel. É apresentado a seguir na imagem 5.12 o seu modo de funcionamento e as validações que são realizadas no momento do registo:

```
/**
 * Get a validator for an incoming registration request.
 *
 * @param array $data
 * @return \Illuminate\Contracts\Validation\Validator
 */
protected function validator(array $data)
{
    return Validator::make($data, [
        'name' => 'required|string|max:255',
        'email' => 'required|string|email|max:255|unique:users',
        'password' => 'required|string|min:6|confirmed',
    ]);
}
```

Figure 5.12: *Validações no registo.*

É possível verificar portanto, que todos os campos carecem de algumas validações:

nome:

- É obrigatório;
- Tem de ser do tipo *string*;
- Tem de possuir até 255 caracteres.

email:

- É obrigatório;

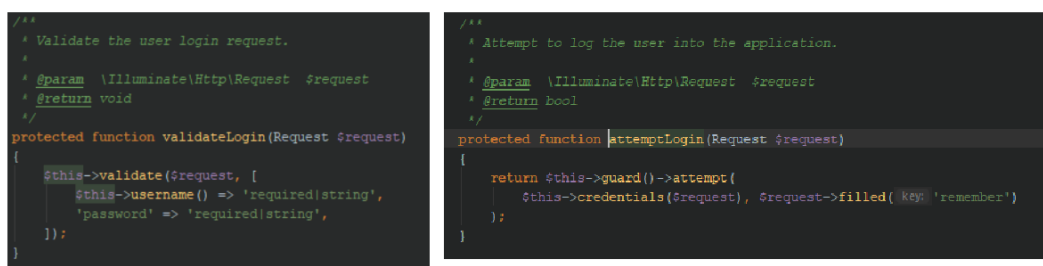
- Tem de ser do tipo *string*;
- Tem de ser do tipo email;
- Tem de possuir até 255 caracteres;
- Tem de ser um campo único, ou seja, não poderá ser registado caso já exista nessa coluna da base de dados.

password:

- É obrigatório;
- Tem de ser do tipo *string*;
- Tem de ser do tipo email;
- Tem de possuir pelo menos 6 caracteres;
- Terá de bater certo com o que o utilizador introduza no campo "confirmação de palavra-passe".

A validação no momento do login utiliza também a classe *Validation*, sendo que apenas valida deste modo se ambos os campos foram preenchidos e se utilizam o tipo *string*.

A validação mais importante deste passo utiliza o método *attempt*, que irá utilizar os campos inseridos de email e palavra-passe e verificar primeiro se o email introduzido existe na base de dados e, por fim, irá verificar se a palavra passe está associada ao email inserido. A figura 5.13 abaixo apresenta a validação de login utilizando o *validation* e o *attempt*:



```

/**
 * Validate the user login request.
 *
 * @param \Illuminate\Http\Request $request
 * @return void
 */
protected function validateLogin(Request $request)
{
    $this->validate($request, [
        $this->username() => 'required|string',
        'password' => 'required|string',
    ]);
}

/**
 * Attempt to log the user into the application.
 *
 * @param \Illuminate\Http\Request $request
 * @return bool
 */
protected function attemptLogin(Request $request)
{
    return $this->guard()->attempt(
        $this->credentials($request), $request->filled('remember')
    );
}

```

Figure 5.13: Validações no login.

5.2.2 Flash Messages

A necessidade de implementar Flash Messages adveio da tentativa de manter o utilizador da plataforma a par das alterações que vai realizando. Quando uma inserção

é realizada com sucesso, uma mensagem a anunciar isso mesmo é apresentada. Por outro lado, se uma ação não é concluída corretamente dentro dos parâmetros configurados, uma mensagem de erro é apresentada ao utilizador.

O Laravel Flash é um *package composer* que permite utilizar o Bootstrap otimizado para implementar um sistema de mensagens flash no desenvolvimento de aplicações. Este package é disponibilizado na plataforma github do utilizador Laracasts. Através do gestor de dependências "Composer", é realizada a instalação do *package*:

composer require laracasts/flash

Para utilizar esta funcionalidade depois de implementada, é preciso certificar de que o ficheiro css do Bootstrap também está a ser chamado.

Posto isto, a chamada de uma mensagem flash é realizada como se apresenta na figura 5.14:

```
/**
 * Store a newly created resource in storage.
 *
 * @param \Illuminate\Http\Request $request
 * @return \Illuminate\Http\Response
 */
public function store(Request $request)
{
    Auth::user()->authorizeRoles(['1']);

    $role= new Role();
    $role->name = $request->get( 'key: 'name');
    $role->description = $request->get( 'key: 'description');
    $role->save();

    flash( message: 'Role ' . $role->name . ' foi criado com sucesso!')->success();

    return redirect()->action( action: 'RoleController@index');
}
```

Figure 5.14: Implementação de mensagem Flash.

Neste exemplo, é apresentado o *controller*, mais especificamente a função que faz o armazenamento de uma permissão. No momento em que é finalizada esta operação, é invocada a mensagem flash que dará a informação ao utilizador de que a permissão foi criada com sucesso.

A mensagem flash necessita apenas de um parâmetro que será a mensagem a apresentar ao utilizador. A seguir à mensagem, poderá ser especificado o tipo de mensagem a ser apresentada, ou seja, se é uma mensagem de sucesso, alerta, erro, aviso, entre outros tipos.

Este último parâmetro irá fazer variar as classes javascript apresentadas na mensagem flash (como verificado na figura 5.15). Por exemplo, o parâmetro *success* irá despoletar a classe *alert-success* que dará um aspeto à mensagem representativa do sucesso da ação (fundo verde, com letras e borders verde-escuro).

```
@foreach (session('flash_notification', collect())->toArray() as $message)
    @if ($message['overlay'])
        @include('flash::modal', [
            'modalClass' => 'flash-modal',
            'title'       => $message['title'],
            'body'        => $message['message']
        ])
    @else
        <div class="alert
            alert-{{ $message['level'] }}
            {{ $message['important'] ? 'alert-important' : '' }}"
            role="alert"
        >
            @if ($message['important'])
                <button type="button"
                    class="close"
                    data-dismiss="alert"
                    aria-hidden="true"
                ></button>
            @endif
            {!! $message['message'] !!}
        </div>
    @endif
@endforeach
{{ session()->forget('flash_notification') }}
```

Figure 5.15: View correspondente da mensagem Flash.

Este *package* é utilizado na plataforma, de forma a dar feedback positivo ou negativo ao utilizador, sempre que este execute uma ação.

5.2.3 Table Seeder

O Seeder é uma funcionalidade que é disponibilizada pelo Artisan do Laravel e que tem como principal objetivo poder popular uma base de dados com dados no momento em que é criada.

Isto tem especial interesse no momento em que a aplicação está a ser desenvolvida, na qual é necessário testar diversas funcionalidades e por vezes ter que inserir dados sempre que se pretende testar algo torna-se bastante monótono e moroso.

Por outro lado, para além desta utilidade, permite ainda que quando se realiza uma instalação da aplicação em qualquer servidor, este fique imediatamente com os dados iniciais imprescindíveis para a aplicação, tais como os status disponíveis ou até mesmo as permissões de utilizadores.

O comando artisan a executar para criar um seeder é o:

```
php artisan make:seeder Tableseeder
```

Todos os seeders são armazenados num diretório, como ilustra a figura 5.16:

5.2. FUNÇÕES IMPLEMENTADAS

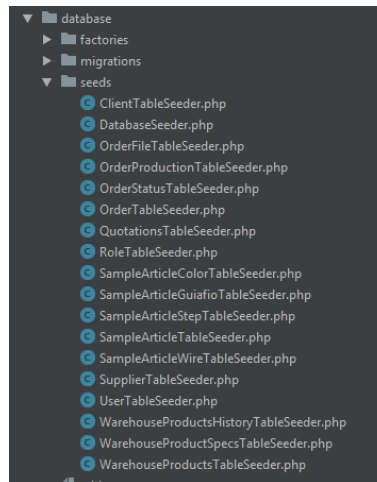


Figure 5.16: Diretório de Seeders desenvolvidos.

```
1 <?php
2
3 use App\OrderStatus;
4 use Illuminate\Database\Seeder;
5
6 class OrderStatusTableSeeder extends Seeder
7 {
8     /**
9      * Run the database seeds.
10     *
11     * @return void
12     */
13     public function run()
14     {
15         $status = new OrderStatus();
16         $status->status = 'Encomenda Recebida';
17         $status->save();
18         $status = new OrderStatus();
19         $status->status = 'A produzir amostra';
20         $status->save();
21         $status = new OrderStatus();
22         $status->status = 'A criar orçamento';
23         $status->save();
24         $status = new OrderStatus();
25         $status->status = 'A aguardar resposta do cliente';
26         $status->save();
27         $status = new OrderStatus();
28         $status->status = 'Em produção';
29         $status->save();
30         $status = new OrderStatus();
31         $status->status = 'A espera de matéria-prima';
32         $status->save();
33         $status = new OrderStatus();
34         $status->status = 'Produzido';
35         $status->save();
36         $status = new OrderStatus();
37         $status->status = 'Em distribuição';
38         $status->save();
39     }
40 }
```

Figure 5.17: Seeder para criar os estados de encomenda.

Depois de todos os seeders criados, é possível ainda unir as funcionalidades de migração de base de dados (já previamente mencionadas) com as funcionalidades do *seeder* e assim criar uma nova instalação de base de dados juntamente com os dados correspondentes através do comando:

```
php artisan migrate:fresh --seed
```

5.2.4 Sistema de Envio de Emails

A framework Laravel disponibiliza uma API simples e prática de envio de emails e que se baseia na biblioteca de envio de emails *"SwiftMailer"* já existente em PHP. Esta biblioteca permite o uso de vários protocolos, tais como SMTP, Mailgun, SparkPost, AmazonSES e ainda as funções *mail* e *sendmail* do PHP. Por forma a fazer o envio, é necessário instalar a biblioteca guzzlehttp, uma vez que se trata de um client PHP HTTP que facilita o envio de *requests HTTP* e é facilmente integrável com qualquer *web service*. Esta instalação é realizada a partir do comando:

```
composer install guzzlehttp/guzzle
```

Posto isto, é necessário implementar as definições do servidor de email que iremos utilizar. Uma vez que durante o desenvolvimento vários emails seriam enviados e recebidos, optou-se por implementar um sistema de emails "virtual", com recurso ao mailtrap. Este serviço permite criar uma caixa de emails fictícia fornecendo todos os dados necessários para implementar o serviço.

A figura 5.18 representa a configuração utilizada a título de testes:

```
MAIL_DRIVER=smtp
MAIL_HOST=smtp.mailtrap.io
MAIL_PORT=2525
MAIL_USERNAME=186d05e3dc7272
MAIL_PASSWORD=ad2b672e51c0dc
MAIL_ENCRYPTION=null
```

Figure 5.18: Configuração de email.

A partir das configurações mencionadas, foi criada classe que irá ser utilizada para o envio de emails da plataforma. Esta classe receberá dois parâmetros no *constructor*, sendo eles o assunto do email (subject) e o conteúdo do email (content). A função *build* fará depois o envio do email, colocando o email de origem como parâmetro *from*, o mesmo valor do email será colocado em bcc para receber uma cópia do mesmo. Irá ser ainda colocado o subject com o seu respetivo parâmetro. Dado que o conteúdo do email seria enviado a partir de uma *view* criada especificamente para o este fim, é enviado o conteúdo do email como parâmetro. A figura 5.19 ilustra a classe que trata do envio do email.

```

1  <?php
2
3  namespace App\Mail;
4
5  use ...
6
11
12  class sendSimpleEmail extends Mailable
13  {
14      use Queueable, SerializesModels;
15
16      public $subject;
17      public $content;
18
19
20      /**
21       * Create a new message instance.
22       *
23       * @return void
24       */
25      public function __construct($subject, $content)
26      {
27          $this->subject = $subject;
28          $this->content = $content;
29      }
30
31
32      /**
33       * Build the message.
34       *
35       * @return $this
36       */
37      public function build()
38      {
39          return $this->from(Auth::user()->email, Auth::user()->name)
40             ->bcc(Auth::user()->email)
41             ->subject($this->subject)
42             ->view('view:emails.emailContent')->with(['content' => $this->content]);
43      }
44  }
45

```

Figure 5.19: Email Controller.

A título de exemplo, é possível identificar no envio de orçamento para o cliente, é um dos diversos usos da classe de envio de email. Tal como foi descrito acima, o Mail necessitará do email de destino como parâmetro, sendo que depois realiza a chamada da classe criada (SendSimpleEmail) que necessita dos parâmetro *subject* e *content* (ver figura 5.20).

```

//Enviar email para criadores de encomendas indicando que um orçamento acabou de ser criado
$users = Role::find(4)->users()->orderBy('name')->get();
$subject = "Novo orçamento efetuado.";
$body = "Foi efetuada a orçamentação do produto descrito abaixo, e enviada ao cliente:
<br>Identificador do Cliente: ". $order->client_identifier ."
<br>Cliente: ". $order->client->client ."
<br><br>
Para aceder à encomenda, dirija-se à plataforma, ou clique
<a href="'.url( path: "/orders/list/")"' target='_blank'>aquí</a>.";
foreach($users as $user) {
    Mail::to($user->email)->send(new sendSimpleEmail($subject, $body));
}

```

Figure 5.20: Relacionamento dos módulos Auxiliares.

5.2.5 Tabelas de Dados - Datatables

As tabelas de dados existentes na aplicação desenvolvida necessitavam de cumprir algumas funcionalidades já existentes na empresa, nomeadamente a sua exportação para formato de ficheiro excel. Havendo esta necessidade, foi utilizada a biblioteca

JQuery JavaScript chamada *DataTables*. É uma ferramenta muito flexível e que permite implementar a exportação noutros formatos para além do ficheiro excel (xls), tais como CSV, PDF ou até mesmo imprimir diretamente para uma impressora.

```
<!-- DataTables-->
<link rel="stylesheet" href="http://cdn.datatables.net/1.10.16/css/jquery.dataTables.min.css">
<link rel="stylesheet" href="https://cdn.datatables.net/buttons/1.5.2/css/buttons.dataTables.min.css">
<script src="http://cdn.datatables.net/1.10.16/js/jquery.dataTables.min.js"></script>
<script src="https://cdn.datatables.net/buttons/1.5.2/js/dataTables.buttons.min.js"></script>
<script src="https://cdn.datatables.net/buttons/1.5.2/js/buttons.flash.min.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/jszip/3.1.3/jszip.min.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/pdfmake/0.1.36/pdfmake.min.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/pdfmake/0.1.36/vfs_fonts.js"></script>
<script src="https://cdn.datatables.net/buttons/1.5.2/js/buttons.html5.min.js"></script>
<script src="https://cdn.datatables.net/buttons/1.5.2/js/buttons.print.min.js"></script>
```

Figure 5.21: *Ficheiros auxiliares da biblioteca DataTables.*

A figura 5.21 representa os ficheiros que são necessários incluir no projeto para poder usufruir da biblioteca de DataTables e de todas as funcionalidades que são importantes para o projeto.

Tendo em conta a figura 5.22, é possível verificar a forma como a implementação surge na página onde a tabela é apresentada. A funcionalidade DataTable recebe uma série de parâmetros, sendo que vários deles são opcionais.

O primeiro parâmetro recebido é a definição das colunas, onde no caso de Stocks, não se permite a reordenação das três últimas colunas, por se tratarem dos botões que não fariam sentido serem ordenáveis. A seguir é apresentado o número de linhas visíveis por página da tabela, ou seja, 10.

Com base na necessidade de exportação da tabela, a *string* "IBfript" acrescenta esta mesma necessidade à tabela, para além das funcionalidades já disponibilizadas. O parâmetro "buttons" a seguir, permite identificar quais os formatos possíveis de exportação, seguidos do nome que aparecerá no botão respetivo.

Por último, com a necessidade de desenvolver uma plataforma que nesta fase fosse totalmente em português e dado que a ferramenta DataTables está por defeito em inglês, foi totalmente traduzida para que o utilizador final conseguisse tirar partido da sua utilização.

Após terminada, a tabela ficará reforçada por uma série de opções: Escolher número de resultados a apresentar; os vários tipos de formatos de exportação; uma caixa para pesquisa nas várias colunas da tabela; a informação de quantas páginas de tabela existem; e uma navegação facilitada para qualquer página da tabela.

5.2. FUNÇÕES IMPLEMENTADAS

```

$('#stock').DataTable({
  columnDefs: [ { orderable: false, targets: [-1, -2] } ],
  "pageLength": 10,
  dom: 'lBfrtip',
  buttons: [
    { extend: 'csv', text: 'CSV' },
    { extend: 'excel', text: 'Excel' },
    { extend: 'pdf', text: 'PDF' },
    { extend: 'print', text: 'Imprimir' }
  ],
  "language": {
    "lengthMenu": "Apresentar MENU resultados por página",
    "zeroRecords": "Nenhum resultado encontrado.",
    "info": "Página PAGE de PAGES",
    "infoEmpty": "Sem resultados disponíveis",
    "infoFiltered": "(Filtrado de MAX resultados totais)",
    "paginate": {
      "first": "Primeira",
      "last": "Última",
      "next": "Seguinte",
      "previous": "Anterior"
    },
    "loadingRecords": "A pesquisar...",
    "processing": "A processar...",
    "search": "Pesquisar:"
  }
});

```

Figure 5.22: Implementação de datatable.

Ao implementar a funcionalidade da forma apresentada acima, o seu resultado final encontra-se identificado na figura 5.23.

Referência	Cor	Stock Bruto (Kg)	Stock Líquido (Kg)	Alerta mínimo (Kg)	Custo (€/Kg)	Atualizado Por	Última Atualização			
ca67	castanho escuro	2511571	2511314	100	1.03	Cátia Rocha	2018-09-27 22:40:15	Editar	Apagar	
e22abcodef/1	pt101e2	49.965	-22625.427	502	15	Alcides Teixeira	2018-09-26 23:14:46	Editar	Apagar	Pedir stock
e22abcodef/1	az3505f2	0	-1.004	500	0	Alcides Teixeira	2018-09-26 23:14:46	Editar	Apagar	Pedir stock
e22abcodef/1	pt101f2	0	-18138.532	500	0	Alcides Teixeira	2018-09-26 23:14:46	Editar	Apagar	Pedir stock
e22pa78/1	pt101e1	-338.85	-86507.481	500	0	Alcides Teixeira	2018-09-26 23:14:46	Editar	Apagar	Pedir stock
e22pa78/1	az3505f1	181185	180.827	181	11	Alcides Teixeira	2018-09-26 23:17:50	Editar	Apagar	
e22pa78/1	pt101f1	0	-0.251	500	0	Alcides Teixeira	2018-09-26 23:14:46	Editar	Apagar	Pedir stock
nylom70/1	pt101e3	-391.44	-400.057	500	23	Alcides Teixeira	2018-09-26 23:14:46	Editar	Apagar	Pedir stock
nylom70/1	az3505f3	-0.03	-49878.959	500	0	Alcides Teixeira	2018-09-26 23:14:46	Editar	Apagar	Pedir stock
nylom70/1	pt101f3	0.135	-9073.071	500	100	Alcides Teixeira	2018-09-26 23:14:46	Editar	Apagar	Pedir stock

Figure 5.23: Tabela apresentada para ecrãs maiores.

The screenshot shows a web interface for a product page. At the top, there's a dropdown menu set to '10' with the text 'resultados por página'. Below it are buttons for 'CSV', 'Excel', 'PDF', and 'Imprimir'. A search bar labeled 'Pesquisar' is present. The main content is a table with the following data:

Referência	ca67
Cor	castanho escuro
Stock Bruto (Kg)	2511571
Stock Líquido (Kg)	25113514
Alerta mínimo (Kg)	100
Custo (€/Kg)	1.03
Atualizado Por	Cátia Rocha
Última Atualização	2018-09-27 22:40:15

Below the table are two buttons: 'Editar' (highlighted with a yellow box) and 'Apagar' (highlighted with a red box). At the bottom, there's a pagination control showing 'Página 1 de 2' and buttons for 'Anterior', '1', '2', and 'Seguinte'.

Figure 5.24: Tabela apresentada para ecrãs menores.

5.2.6 Dados Estatísticos - Chart.js

A biblioteca *Chart.js* é uma ferramenta *open-source* bastante simples mas flexível que permite o desenvolvimento de informação numa representação gráfica. Permite uma série de variados gráficos, desde os mais simples até aos mais complexos.

```
//Obter dados para gráfico 4
//Primeiro obter quais os clientes existentes
$clientes = Client::all()
->toArray();

$i = 0;
$g4Data = [];
//Para cada cliente, obter o valor enviado no orçamento, por dia, para os clientes
foreach($clientes as $client) {
    $quotPerDate = [];

    $clientsData = DB::table( table: 'clients')
        ->leftJoin( table: 'orders', first: 'clients.id', operator: '=', second: 'orders.client_id')
        ->leftJoin( table: 'quotations', first: 'orders.id', operator: '=', second: 'quotations.order_id')
        ->select( columns: 'quotations.value_sent')
        ->selectRaw( expression: 'DATE(quotations.created_at) as date')
        ->selectRaw( expression: 'sum(quotations.value_sent) as total_value')
        ->where( column: 'clients.id', $client['id'])
        ->groupBy( ..groups: 'date')
        ->pluck( column: 'total_value', key: 'date');

    //dd($clientsData);
    //Obter um array que possibilitasse o incremento do valor ao longo dos dias.
    $prev = 0;
    foreach($label as $date) {
        $quotPerDate[$date] = @$clientsData[$date] ? @$clientsData[$date]+$prev : $prev;
        $prev = @$clientsData[$date] ? @$clientsData[$date]+$prev : $prev;
    }
    //dd($quotPerDate);

    $g4Data[$i] = $quotPerDate;
    $i++;
}
```

Figure 5.25: Controller responsável pelo envio dos dados para a view.

dos parâmetros *labels* (as datas a inserir no eixo das abcissas) e também os *datasets* (os valores de gastos com a INDMEI, por cada um dos seus clientes). As opções disponíveis como o título não foram necessárias uma vez que este foi inserido fora do gráfico, por cima do mesmo.

```
//Gráfico 4
var ctx4 = document.getElementById("myChart4").getContext('2d');
var myChart4 = new Chart(ctx4, {
  type: 'line',
  data: {
    labels: label,
    datasets: arrayData4
  },
  options: {
    title: {
      display: false,
      text: ''
    }
  }
});
```

Figure 5.28: Script implementado para um dos gráficos.

A figura 5.29 representa o gráfico implementado a partir da função acima descrita.

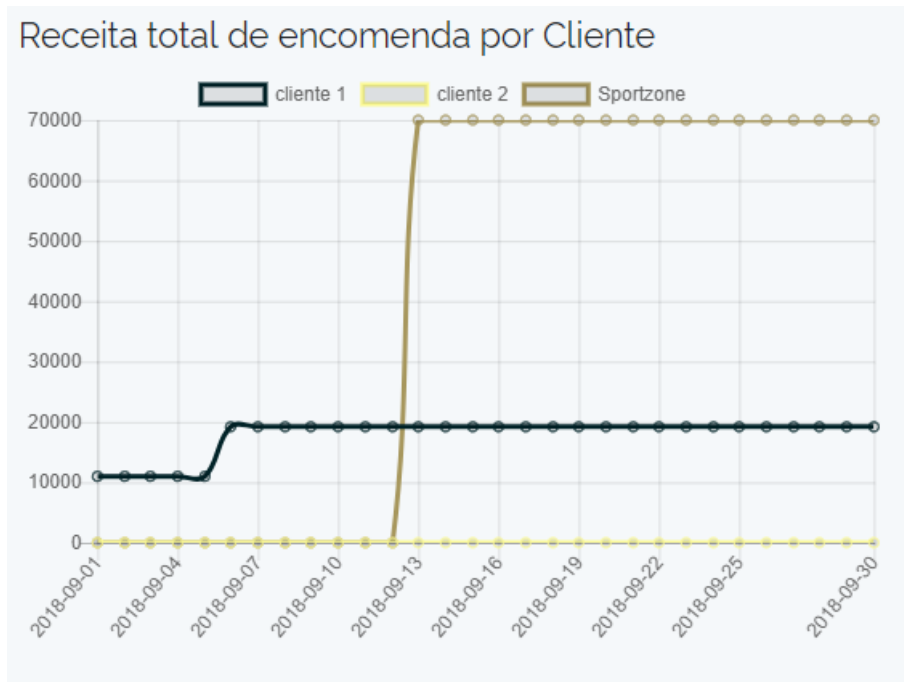


Figure 5.29: Gráfico implementado.

5.2.7 Atualização do Stock do Armazém

A atualização de stock do armazém é uma funcionalidade fulcral para o bom desempenho da aplicação. Isto porque, para além de ser um dos principais objetivos do projeto, é também uma ação que tem de ocorrer em diversas ocasiões do projeto.

A atualização de stock é efetuada em dois momentos distintos: 1º No momento em que uma matéria-prima em específico sofre uma alteração de quantidade; 2º quando a lista de matérias-primas no armazém é consultada.

```

public function updateStocks () {
    $history = DB::table( table: 'warehouse_products_history' )
        ->get()->toArray();
    $products = WarehouseProductSpec::all();
    //Para cada produto, percorre o array de histórico, e calcula o valor:
    //Se o valor de histórico tiver IN, soma; Se o valor histórico tiver OUT, subtrai.
    //Stock Líquido -> valor em stock menos o q foi associado para cada encomenda.
    //Stock Bruto -> valor em stock no momento -> corresponde ao stock menos o valor utilizado pelos operadores todos os dias.
    foreach($products as $product) {
        $total_liquid = $total_bruto = 0;
        $cost = 0;
        //Cálculo de stock líquido: total menos as encomendas criadas
        foreach ($history as $key => $val) {
            if ($val->warehouse_product_spec_id == $product->id) {
                if ($val->inout == 'IN') {
                    $total_liquid += $val->weight;
                    $cost = $val->cost;
                }
                elseif ($val->inout == 'OUT_LIQUID') {
                    $total_liquid -= $val->weight;
                }
            }
        }
        //Cálculo de stock bruto: total menos o que já foi produzido, ou seja, o que existe efectivamente no armazém
        foreach ($history as $key => $val) {
            if ($val->warehouse_product_spec_id == $product->id) {
                if ($val->inout == 'IN') {
                    $total_bruto += $val->weight;
                }
                elseif ($val->inout == 'OUT_GROSS') {
                    $total_bruto -= $val->weight;
                }
            }
        }
        //Apenas atualiza os valores de stock que sofreram alterações
        $currentValsStored = WarehouseProductSpec::where('id', $product->id)->first();
        if (strcmp($currentValsStored->liquid_weight, $total_liquid) ||
            strcmp($currentValsStored->gross_weight, $total_bruto) ||
            strcmp($currentValsStored->cost, $cost)) {
            WarehouseProductSpec::where('id', $product->id)
                ->update([
                    'liquid_weight' => $total_liquid,
                    'gross_weight' => $total_bruto,
                    'cost' => $cost
                ]);
        }
    }
    return "done";
}

```

Figure 5.30: Atualização de valores ao listar Stocks.

Este 2º momento, ou seja, a consulta de matérias-primas em armazém, corresponde a uma ação de atualização dos valores totais de todas as matérias-primas que sofreram alterações no primeiro momento. A figura 5.30 representa a função que realiza a atualização de stock e será instanciada no momento em que qualquer utilizador lista o stock do armazém.

Quando esta ação é realizada, é despoletado o cálculo do histórico de cada um dos produtos em armazém, somando as matérias-primas que entraram e subtraindo o valor que está a ser gasto para stock líquido. O mesmo processo ocorre para o stock bruto.

Quando o valor destes cálculos difere em termos de valor líquido, valor bruto, ou custo da matéria-prima, a matéria em questão terá estes três parâmetros atualizados. Desta forma os valores apenas serão atualizados se for estritamente necessário e a data de atualização corresponderá à data correta em que a matéria-prima foi atualizada.

O 1º momento mencionado acima, ou seja, quando a matéria-prima específica sofre alterações, acontece em 4 ocasiões no projeto:

- Ao criar uma encomenda nova - Quando a encomenda é criada, é inserido o total de pares de meias a produzir, o que afetará a matéria-prima a ser gasta para essa encomenda;
- Ao atualizar uma encomenda - A matéria-prima poderá ou não sofrer alterações, na medida em que a quantidade de pares de meias pode ser aumentada ou reduzida;
- Ao alterar uma amostra de artigo, que poderá ou não estar associada a uma encomenda - Quando uma amostra de artigo é alterada, poderá afetar a gramagem de uma qualquer matéria-prima e, portanto, é necessário atualizar os stocks em armazém;
- Ao atualizar o valor diário produzido por um operador em qualquer uma das encomendas - Neste momento, o valor de stock bruto, ou seja, o total em armazém subtraindo o valor das encomendas, irá certamente sofrer alterações e por isso, o valor em stock irá ser atualizado também neste momento.

A atualização da matéria-prima irá ser efetuada portanto, tendo como base o histórico de cada uma delas.

O histórico é controlado pela função da figura 5.34. Este método, instanciado nos 4 momentos descritos anteriormente, encontra-se dependente de 3 funções.

A primeira função (`checkWireSpentInOnePair`) evidenciada na figura 5.31 recebe como parâmetro a encomenda em questão e irá retornar um objeto com a quantidade de fio gasto num par de meias, para todas as cores utilizadas. Trata-se de um *array* bastante completo, com o identificador do fio e a quantidade em gramas, gasta.

5.2. FUNÇÕES IMPLEMENTADAS

```
public function checkWireSpentInOnePair ($request) {
    //Add Row to Stock History with the order to subtract:
    $sampleArticleId = $request->sample_article_id;
    //SELECIONAR o total GASTO, POR FIO, numa amostra, por COR de UM PAR DE MEIAS:
    // Multiplos de 4 -> somando 0 sao cor 1;
    // Multiplos de 4 -> somando 1 sao cor 2;
    // Multiplos de 4 -> somando 2 sao cor 3;
    // Multiplos de 4 -> somando 3 sao cor 4;
    $wires = DB::table('sample_articles_wires')
        ->select( columns: 'sample_articles_wires.id', 'sample_article_id', 'warehouse_product_id', 'grams', 'a.sample_articles_wire_id', 'a.warehouse_product_spec_id')
        ->selectRaw( expression: 'COUNT(sample_articles_wire_id) AS total_samples')
        ->selectRaw( expression: 'MIN(a.id) AS minColorID')
        ->selectRaw(
            expression: 'CASE
                WHEN MOD(a.id, 4) = 0 THEN 4
                ELSE MOD(a.id, 4)
            END as cor')
        ->leftJoin( table: 'sample_article_colors AS s', with: 'sample_articles_wires.id', operation: '=', second: 'a.sample_articles_wire_id')
        ->where( column: 'sample_article_id', $sampleArticleId)
        ->groupBy( groups: 'sample_articles_wires.id', 'a.id')
        ->orderBy( column: 'cor')
        ->orderBy( column: 'minColorID')
        ->get()
        ->toArray();
    //FIM SELECIONAR o total GASTO numa amostra, por COR de UM PAR DE MEIAS:
    return $wires;
}
```

Figure 5.31: Função que verifica o fio gasto por par de meias.

Tendo este valor por par de meias, é depois executada a segunda função acessória deste processo (pairsPerColorLiquid) evidenciada na figura 5.32 que retornará para cada uma das 4 cores, dos diferentes tipos de pares de meias, qual a quantidade de pares de meias totais que a encomenda possui. Esta função retorna um *array* com 4 combinações key-value, com as 4 cores e os seus valores totais de pares de meias da encomenda.

```
/**
 * Função que retorna o total de pares para cada cor
 * Usada para o cálculo do STOCK LÍQUIDO: TOTAL MENOS AS ENCOMENDAS CRIADAS
 */
public function pairsPerColorLiquid ($request) {
    //Selecionar as quantidades de pares de meias, POR COR
    dd($request->all());
    $paresPorCor = [
        'cor1' => $request->tamanho1+$request->tamanho2+$request->tamanho3+$request->tamanho4,
        'cor2' => $request->tamanho21+$request->tamanho22+$request->tamanho23+$request->tamanho24,
        'cor3' => $request->tamanho31+$request->tamanho32+$request->tamanho33+$request->tamanho34,
        'cor4' => $request->tamanho41+$request->tamanho42+$request->tamanho43+$request->tamanho44
    ];
    //dd($paresPorCor);
    return $paresPorCor;
    //FIM Selecionar as quantidades de pares de meias, POR COR
}
```

Figure 5.32: Função que obtém o total líquido de pares de meias, por cor.

A terceira função acessória (pairsPerColorGross) evidenciada na figura 5.33 recebe como parâmetro o id da encomenda em questão e irá posteriormente calcular o valor total produzido de meias dessa mesma encomenda, por cor, até ao momento. Isto porque o cálculo do valor bruto representa o valor que existe efetivamente no armazém. No final, irá retornar um array de 4 combinações key-value, com as 4 cores e os seus valores de pares de meias produzidos até ao momento.

```

/**
 * Função que retorna o total de pares para cada cor
 * Usada para o cálculo do STOCK BRUTO: TOTAL MENOS OS TOTAIS DE ENCOMENDAS JÁ EXECUTADOS PELOS OPERADORES
 */
public function pairsPerColorGross ($order_id) {
    //Selecionar as quantidades de pares de meias, POR COR, já executados
    $currentProduction = OrderProduction::where('order_id', $order_id)->get();
    $sparesPorCor = [];

    foreach ($currentProduction as $newInsertion) {
        if(array_key_exists( $key: 'cor', $newInsertion->cor, $sparesPorCor) ) {
            $sparesPorCor['cor' . $newInsertion->cor] = intval($sparesPorCor['cor' . $newInsertion->cor]) + intval($newInsertion->value);
        } else {
            $sparesPorCor['cor' . $newInsertion->cor] = intval($newInsertion->value);
        }
    }

    //fazer verificação para as 4 cores: caso não tenha, coloca valor a zero
    if(count($sparesPorCor) != 4) {
        for($i = 1; $i <= 4; $i ++ ) {
            if (!array_key_exists( $key: 'cor' . $i, $sparesPorCor) ) {
                $sparesPorCor['cor' . $i] = '0';
            }
        }
    }

    // * 0.97 / 2 uma vez que nos referimos a meias e não a pares
    foreach ($sparesPorCor as $key => $par) {
        //dump($key);
        $sparesPorCor[$key] = round( $par * 0.97 / 2);
    }

    //dd($sparesPorCor);
    return $sparesPorCor;
    //FIM Selecionar as quantidades de pares de meias, POR COR, já executados
}

```

Figure 5.33: Função que obtém o total bruto de pares de meias, por cor.

Assim que todos os dados anteriores forem obtidos, as quantidades totais de *stock* bruto e líquido serão atualizadas tendo por base todas as movimentações que ocorreram com a matéria-prima ao longo do tempo.

Serão então adicionadas novamente as linhas de histórico correspondentes à encomenda em questão: no caso de se tratar de uma atualização de *stock* líquido, a *tag* "OUT_LIQUID" será associada. Por outro lado, se se tratar de uma atualização de *stock* bruto, a *tag* "OUT_GROSS" será indicada.

Juntamente com a *tag* respectiva, segue ainda o utilizador que realizou a atualização, o peso de matéria-prima a reduzir, o custo (com a identificação N/A), uma imagem da amostra do artigo e ainda uma descrição com a informação da encomenda e cliente que gastou determinada quantidade de matéria-prima (Encomenda para o cliente ... com o identificador ...) juntamente com a data e hora da alteração.

5.2. FUNÇÕES IMPLEMENTADAS

```
public function addRowtoStockHistory ($request, $id) {
    $wires = $this->checkWireSpentInOnePair($request);
    $clientName = Client::where('id', $request->client_id->first()->client);
    $sparesPorCorLiquido = $this->pairsPerColorLiquid($request);
    $sparesPorCorBruto = $this->pairsPerColorGross($id);

    //Delete das entradas antes de atualizar.
    DB::table('warehouse_products_history')
        ->where('column: 'description', 'operator: 'Encomenda para o cliente: ' . $clientName . ', com o identificador: ' . $request->client_id)
        ->delete();

    //Obter imagem do produto
    $orderImage = SampleArticle::where('id', $request->sample_article_id->first()->image_url);

    //Editar para conter historico de liquid weight e gross weight!!!!
    //Adicionar as quantidades de fio, dependendo das meias selecionadas: stock
    for($i = 1; $i <= 4; $i++) {
        $cor = 'cor'.$i;
        foreach ($wires as $wire) {
            if($wire->cor == $i || $wire->grams != '0') {
                //Inserir valor liquido multiplicado pelos pares
                DB::table('warehouse_products_history')->insert([
                    'warehouse_product_spec_id' => $wire->warehouse_product_spec_id,
                    'user_id' => Auth::id(),
                    'inout' => 'OUT_LIQUID',
                    'weight' => $wire->grams * $sparesPorCorLiquido[$cor],
                    'cost' => 'N/A',
                    'receipt' => $orderImage,
                    'description' => 'Encomenda para o cliente: ' . $clientName . ', com o identificador: ' . $request->client_id,
                    'created_at' => Carbon::now(),
                    'updated_at' => Carbon::now(),
                ]);

                //Inserir valor bruto multiplicado pelos pares
                DB::table('warehouse_products_history')->insert([
                    'warehouse_product_spec_id' => $wire->warehouse_product_spec_id,
                    'user_id' => Auth::id(),
                    'inout' => 'OUT_GROSS',
                    'weight' => $wire->grams * $sparesPorCorBruto[$cor],
                    'cost' => 'N/A',
                    'receipt' => $orderImage,
                    'description' => 'Encomenda para o cliente: ' . $clientName . ', com o identificador: ' . $request->client_id,
                    'created_at' => Carbon::now(),
                    'updated_at' => Carbon::now(),
                ]);
            }
        }
    }
}

//END - Add Row to Stock History with the order to subtract
return $clientName;
```

Figure 5.34: Método para adicionar stock.

O utilizador irá depois poder consultar o valor atualizado de todas as encomendas ao clicar na linha da matéria-prima pretendida, na lista de *stock*.

5.2.8 Upload de Ficheiros

O upload de ficheiros é essencial no desenvolvimento do projeto de estágio. Este processo repete-se em diversas ocasiões, sendo as mais importantes: 1 - O momento de entrada de uma encomenda na empresa, pois é importante fazer *upload* da fatura para fazer o seu armazenamento; 2 - O momento em que é criada uma nova encomenda, uma vez que geralmente vêm associados diversos ficheiros a um pedido de encomenda, quer sejam *mockups* do que é pretendido, quer seja o pedido solicitado em papel.

No momento em que são submetidos os ficheiros, é realizada uma validação do tipo de ficheiro, permitindo apenas o upload de imagens das extensões mais comuns (jpg, png e gif) e também de ficheiros no formato PDF. A figura 5.35 evidencia esta mesma validação, procurando por qualquer uma destas extensões no nome do ficheiro. Caso

algum dos ficheiros enviados para a plataforma não cumpra estes requisitos, o pedido não será submetido e é indicado ao utilizador quais são as extensões permitidas.

```
if($request->order_files_id) {
    foreach ($request->order_files_id as $order_files_id) {
        if (!str_contains($order_files_id->getClientOriginalName(), ['pdf', 'jpg', 'png', 'gif', 'JPG', 'PNG', 'GIF'])) {
            flash( message: 'Por favor, insira uma extensão válida. (.pdf, .jpeg, .png, .gif)'->error());

            return redirect()->back();
        }
    }
}
```

Figure 5.35: Validações aos ficheiros que são carregados.

Caso os ficheiros efetivamente cumpram os requisitos de extensões, é então momento de fazer o seu *upload* para o servidor.

Este processo inicia-se pela definição da extensão a utilizar. Os ficheiros armazenados passam então a ter a extensão de ".pdf" ou de ".jpg" caso se trate de um ficheiro PDF ou de uma imagem. O nome da imagem depois será também alterado. É-lhe adicionado ao nome original a data e hora. Isto acontece como método de precaução para o caso de existir o *upload* de dois ficheiros diferentes, mas com nomes iguais. Havendo a adição de data e hora, até ao segundo, apenas se duas imagens tiverem o mesmo nome e forem enviadas para o servidor na mesma data e hora é poderá haver algum problema e portanto o risco é praticamente nulo neste caso.

O método que fará o armazenamento da imagem é o *Storage* do Laravel. Primeiramente terá de ser executado o comando que se apresenta a seguir, que criará um *symbolic link* a partir do qual se poderá aceder à imagem através de um link público:

php artisan storage:link

Este método permite fazer o armazenamento local, configurando a pasta de destino previamente. Serão depois necessários dois parâmetros para o método, sendo o primeiro o nome com o qual o ficheiro será armazenado e o segundo será o ficheiro em si (obtido através do método *File* da framework Laravel).

Este armazenamento do ficheiro acresce ainda o armazenamento do seu nome na tabela da base de dados e o identificador da encomenda. Desta forma poderá ser apresentado ao utilizador quando este pretender consultar a encomenda novamente. A figura 5.36 representa o processo descrito anteriormente.

5.2. FUNÇÕES IMPLEMENTADAS

```
//Store Image
foreach ($request->order_files_id as $order_files_id) {
    $orderFile= new OrderFile();
    $file = $order_files_id;
    if($file) {
        $extension = str_contains($file->getClientOriginalName(), 'needle:' . 'pdf') ? 'pdf' : 'jpg';
        $filename = 'orders/' . explode('/', $file->getClientOriginalName())[0] . '-' . Carbon::now('Europe/London')->format('YmdHis') . $extension;
        Storage::disk('public')->put($filename, File::get($file));
        $orderFile->order_id = $order->id;
        $orderFile->url = $filename;
        $orderFile->save();
    }
}
//End Store Image
```

Figure 5.36: Upload de ficheiros.

5.2.9 Atualização de Custo de uma Amostra

O custo de uma amostra é uma funcionalidade importante essencialmente ao gestor de orçamentos, uma vez que terá de ter acesso a esta informação para uma orçamentação adequada. Posto isto, o valor de custo de uma amostra é atualizado sempre que o gestor de orçamentos cria ou atualiza um orçamento e ainda sempre que uma amostra de artigo é criada ou atualizada.

```
/**
 * recebe o id da amostra e calcula os valores para 4 cores
 * retorna um array com os 4 valores
 * o método corre ao criar ou ao fazer update de uma amostra
 */
public function getValuePerSample ($sample) {

    //Obter valor de cada fio (por Kg)
    $wireCost = WarehouseProductSpec::all()->pluck('value:cost')->toArray();

    //Para cada cor, pegar na quantidade de fio e multiplicar pelo preço/100 (pois o fio está em gramas)
    //ir somando sempre ao valor anterior
    $valueCor = [
        'cor1' => 0,
        'cor2' => 0,
        'cor3' => 0,
        'cor4' => 0
    ];

    for ($j = 1; $j <= 4; $j++) {
        for ($i = 1; $i <= $sample['rowCount']; $i++) {
            //Para o caso dos valores vazios, colocar aqui condição
            $keyForWireCost = intval(@$sample['row-' . $i . '-color' . $j]) - 1 == -1 ? 0 : intval(@$sample['row-' . $i . '-color' . $j]) - 1;
            $valueCor['cor' . $j] += intval(@$sample['row-' . $i . '-grams']) * $wireCost[$keyForWireCost] / 100;
        }
    }

    return $valueCor;
}
```

Figure 5.37: Atualização do custo de uma amostra.

O método *getValueForSample* (através de um parâmetro - a amostra de artigo) apresentado na figura 5.37 realiza o cálculo do custo da amostra. Uma amostra de artigo possui sempre 4 cores diferenciadoras da amostra (segundo os requisitos da empresa) e portanto a matéria-prima a utilizar é diferente para cada uma das cores, que por consequência o custo também será. Deste modo, uma amostra possui 4 valores de custo distinto que serão calculados neste momento e que corresponder às cores "cor1", "cor2", "cor3", "cor4".

Para cada uma destas cores, são então percorridos todos os passos de produção de uma amostra e multiplicados o número de gramas pelo custo da amostra, dividindo posteriormente por 100 (dado que o custo é guardado por Kg e não por grama). Ao incrementar os custos de todos os passos de produção, obtemos o custo total da amostra, separado por cor, tal como pretendido para realizar o orçamento da encomenda.

6

Demonstração e Análise dos Resultados

A plataforma desenvolvida teve um período no qual o principal foco foi a gestão de possíveis erros ao longo do uso da mesma. Durante esse período, foram corrigidos pequenos *bugs*, que decorrem do uso não adequado da plataforma, ou seja, quando os dados introduzidos não são aqueles que a plataforma espera obter, ou quando existem falta de dados para prosseguir.

Dentro do período referido, a plataforma foi utilizada por utilizadores externos ao projeto, que não conheciam a plataforma, no sentido de verificar quais seriam as dificuldades de um utilizador ao ver a plataforma pela primeira vez.

Nesse processo, foram otimizados alguns menus que foram considerados menos intuitivos, foram adicionadas algumas *tooltips* em certas funcionalidades e até mesmo foram reformuladas funcionalidades para que não fosse necessário recorrer a mais interações (cliques) quando o mesmo processo poderia ser realizado com menos (por exemplo, na funcionalidade que atualiza o *stock* no momento em que uma amostra é alterada; inicialmente seria necessário passar pelo processo de edição da encomenda antes de passar diretamente para a orçamentação; atualmente a orçamentação pode ser realizada imediatamente após uma atualização de *stock*). A seguir neste capítulo irá fazer-se uma apresentação e demonstração das funcionalidades desenvolvidas no capítulo 5.

6.1 Funcionalidades da Aplicação

A interface em si foi baseada na *skin default* que a *framework* Laravel oferece aos utilizadores. Isto significa que possibilita a utilização das *frameworks* Bootstrap e JQuery, tornando mais eficaz o desenvolvimento da *user interface*.

Foram ainda utilizadas bibliotecas dependentes de JQuery, nomeadamente na criação das tabelas de dados, por forma a aumentar o número de funcionalidades possíveis, tais como a visualização separada por paginação, as diferentes formas de exportação e até mesmo a pesquisa na tabela.

Há ainda a ressaltar que a interface foi pensada para permitir a sua utilização em qualquer tipo de dispositivo, computador, *tablet* ou telemóvel. É, portanto, uma aplicação web em que foi tido em consideração o *responsive design*, o que melhora a sua experiência de utilização e também a flexibilidade de uso. Isto permite ainda que um utilizador possa aceder à aplicação em qualquer dispositivo sem perda de informação útil, à custa de um *layout* menos intuitivo, o que garante também uma longevidade na utilização desta ferramenta por parte da empresa que a irá implementar.

São evidenciadas a seguir as interfaces do *software*, independente do dispositivo de visualização, quer seja visualizado num computador, quer seja visualizado num dispositivo móvel.

6.1.1 Login

No momento em que um utilizador acede à aplicação pela primeira vez, possui a identificação da Empresa imediatamente à sua frente, bem como uma ação de "Login" e uma ação de "Registo" na parte superior do ecrã.

Logo ao centro, é apresentada uma imagem da empresa, clicável, que redireciona o utilizador para o ecrã de login. Esta ação é especialmente criada para dispositivos móveis, em que se pretendem obter botões facilmente alcançáveis e responsivos. A imagem 6.1 representa uma vista *desktop* e mobile do ecrã inicial da aplicação.



Figure 6.1: Ecrã inicial em desktop e mobile.

No caso de o utilizador aceder à plataforma pela primeira vez, terá de realizar o registo na mesma. O registo é um processo relativamente simples, em que o utilizador apenas terá que preencher os campos apresentados: Nome, Email, Password e Confirmação de Password.

Assim que o registo é terminado, o utilizador é encaminhado para o interior da aplicação como convidado, sendo que das próximas vezes que pretenda aceder novamente, terá as suas credenciais já criadas.

O acesso não necessita de validação por parte de administradores da aplicação, uma vez que todos os utilizadores que criam o registo têm atribuída a permissão de "Convidado" e portanto, não terão acesso a qualquer funcionalidade, a não ser que um Administrador da plataforma conceda uma permissão diferente.

O ecrã de registo é apresentado de seguida (6.2), em versão desktop e mobile.

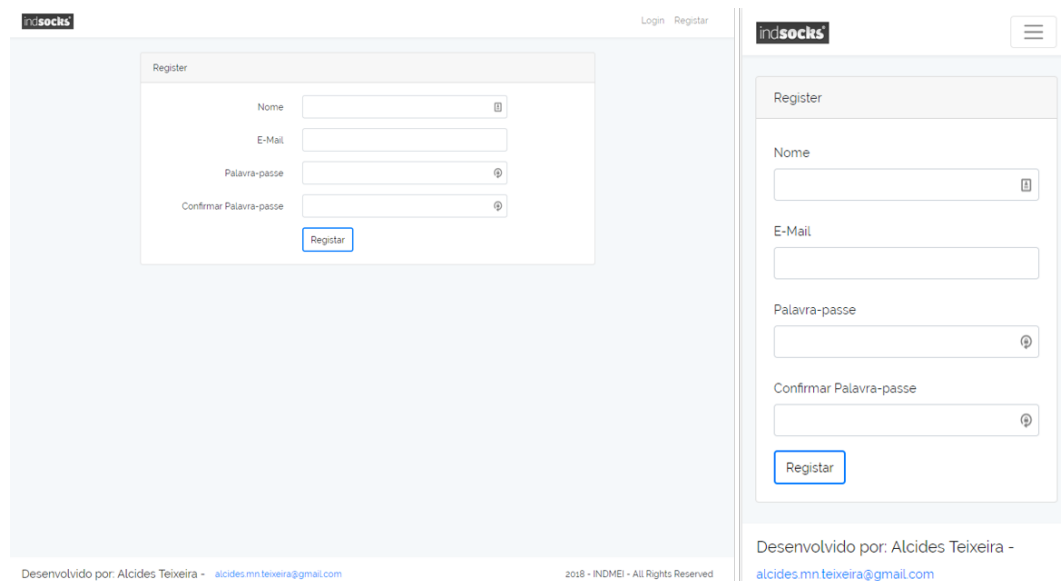


Figure 6.2: *Ecrã de registo em desktop e mobile.*

Todos os campos são de preenchimento obrigatório, sendo o campo do nome do tipo *text*, o campo email do tipo *email* e os campos da *password* serão do tipo *password*.

A password é encriptada no momento em que é armazenada na base de dados, para que não seja possível obter a password, nem mesmo acedendo à base de dados diretamente.

No momento em que se pretende realizar o login, é apresentado o ecrã de login ao utilizador. Neste momento, o utilizador terá de inserir o email e a *password* correta-

6.1. FUNCIONALIDADES DA APLICAÇÃO

mente.

Existe ainda a possibilidade de relembrar os dados de login para o caso de aceder sempre a partir do computador ou de outro qualquer dispositivo pessoal.

No caso de se ter esquecido da palavra-passe, é possível repor uma nova palavra passe através de um email que será enviado para o email do respetivo utilizador, evitando o falseamento de dados. Este processo é representado pelas figuras 6.3 e 6.4.

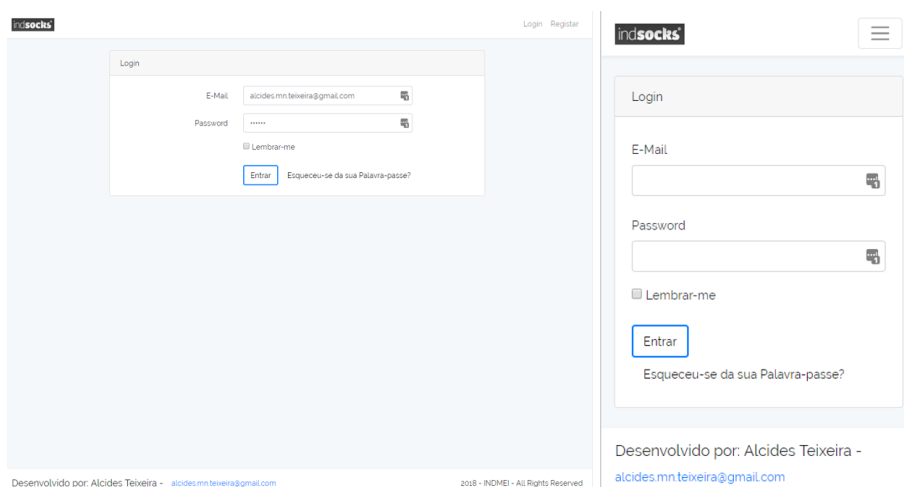


Figure 6.3: *Ecrã de login em desktop e mobile.*

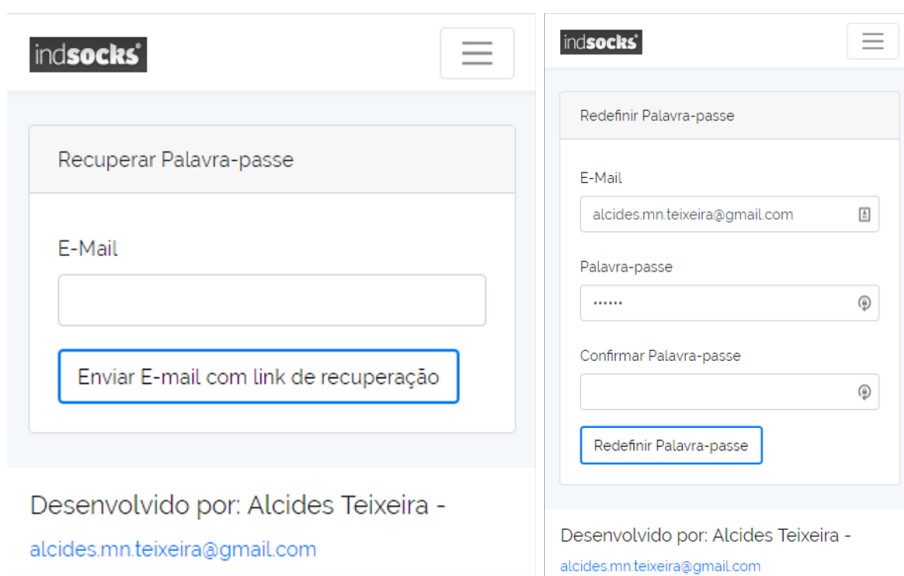


Figure 6.4: *Ecrã de esquecimento de palavra-passe em desktop e mobile.*

6.1.2 Menus

O menu principal de utilizador administrador destaca todas as funcionalidades de gestão possíveis. Nem todos os utilizadores terão acesso a tudo. Contudo, o menu de um utilizador que seja Administrador, terá um aspeto semelhante ao apresentado nas figuras 6.5 e 6.6.



Figure 6.5: Menu completo para ecrãs maiores.



Figure 6.6: Menu completo para ecrãs menores.

Através deste tipo de organização, é possível facilmente diferenciar as funcionalidades que cada permissão terá acesso.

6.1.3 Gerir Permissões

Criar Permissão

Criar permissão permite ao Administrador criar uma nova Permissão que depois poderá associar a utilizadores, conforme pretenda. Para isso, basta apenas preencher o campo Nome e Descrição, conforme a imagem 6.7 apresenta.

The screenshot shows a web form titled 'Criar Nova Permissão' from the 'indsocks' application. It features two text input fields labeled 'Nome:' and 'Descrição:'. Below the inputs are two buttons: a green 'Criar' button and a blue 'Voltar' button.

Figure 6.7: *Criar nova permissão.*

Listar Permissões

A listagem de permissões permite ao Administrador consultar todas as permissões existentes, bem como oferecer a possibilidade de editar uma permissão, ou até mesmo apagar (6.8).

Lista de Permissões disponíveis

Apresentar resultados por página Pesquisar:

Nome	Descrição	Data de Alteração		
Admin	O role admin tem acesso a todas as funcionalidades.	2018-08-29 22:17:02	Editar	Apagar
Convidado	O role convidado tem acessos restritos e apenas pode visualizar alguma info.	2018-08-29 22:17:02	Editar	Apagar
Gestor de Amostra de Artigo	O role de gestor de amostra de artigo tem acesso a todas as funcionalidades relacionadas com criar amostras de artigo.	2018-08-29 22:17:02	Editar	Apagar
Gestor de Armazém	O role de gestor de armazém tem acesso a todas as funcionalidades relacionadas com a gestão de Stock.	2018-08-29 22:17:02	Editar	Apagar
Gestor de Encomenda	O gestor de encomenda de artigo poderá realizar todas as operações relacionadas com amostras de artigos.	2018-08-29 22:17:02	Editar	Apagar
Gestor de Orçamentação	O gestor de orçamentação poderá aceder às amostras e ao seu valor, para depois gerar o seu orçamento.	2018-08-29 22:17:02	Editar	Apagar
Operário	O role operário terá acesso às encomendas prontas a produzir, e a todas as suas características.	2018-08-29 22:17:02	Editar	Apagar

Página 1 de 1 Anterior Seguinte

Figure 6.8: *Listar Permissões.*

Caso o Administrador pretenda apagar uma Permissão, é apresentada uma se-

gunda mensagem de confirmação da acção ”Tem a certeza que pretende apagar a permissão ...?”

Para além desta medida de segurança indicada, apagar uma permissão está ainda dependente do facto de existirem ou não utilizadores com essa permissão. Caso existam, primeiro é necessário desassociar essa permissão dos utilizadores e só depois então sim, é possível apagá-la. A figura 6.9 representa esse mesmo processo, onde na figura 6.10 é exemplificado o alerta de erro respetivo.

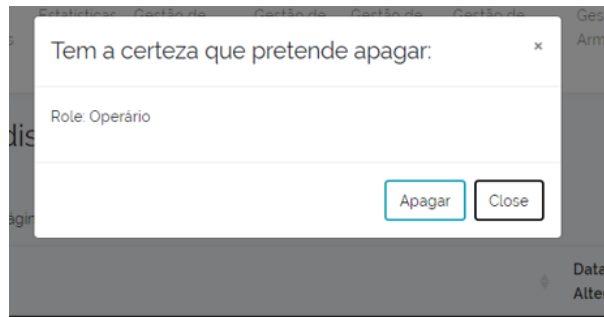


Figure 6.9: *Confirmação ao apagar permissão.*

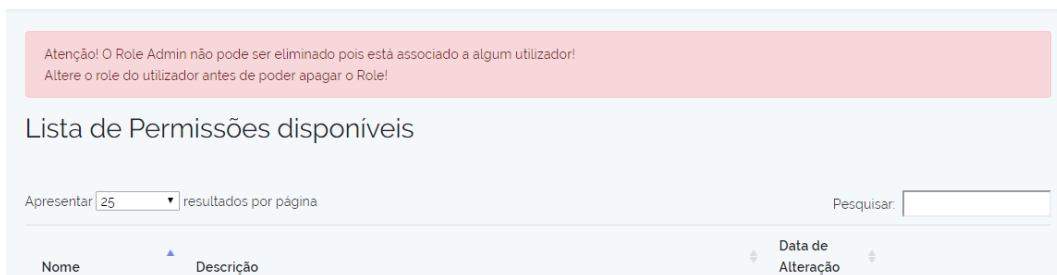


Figure 6.10: *Mensagem de erro ao apagar permissão.*

Listar Utilizadores

Listar utilizadores permite ao Administrador verificar quais os utilizadores inscritos na plataforma. Permite ainda editar os dados de qualquer utilizador e até mesmo apagar um utilizador do sistema (6.11).

6.1. FUNCIONALIDADES DA APLICAÇÃO

Lista de Utilizadores e das suas Permissões

Apresentar resultados por página Pesquisar:

Nome	Email	Permissões	Data de Alteração		
Alcides Teixeira	alcides.mn.teixeira@gmail.com	Admin, Gestor de Amostra de Artigo, Operário.	2018-09-28 23:07:06	<input type="button" value="Editar"/>	<input type="button" value="Apagar"/>
Cátia Rocha	catia_rocha@live.com.pt	Gestor de Orçamentação.	2018-09-30 21:48:17	<input type="button" value="Editar"/>	<input type="button" value="Apagar"/>
Daniel Rodrigues	alcides.mn.teixeira+2@gmail.com		2018-09-19 20:58:25	<input type="button" value="Editar"/>	<input type="button" value="Apagar"/>
Gestor de Encomenda	alcides.mn.teixeira+10@gmail.com	Gestor de Encomenda.	2018-08-29 22:17:02	<input type="button" value="Editar"/>	<input type="button" value="Apagar"/>
José Costa	cides26@gmail.com	Gestor de Amostra de Artigo.	2018-09-06 18:50:43	<input type="button" value="Editar"/>	<input type="button" value="Apagar"/>

Página 1 de 1 Anterior Seguinte

Figure 6.11: *Listar utilizadores.*

Apagar um utilizador do sistema carece de uma confirmação numa segunda mensagem de aviso que surge antes mesmo de se apagar (6.12).

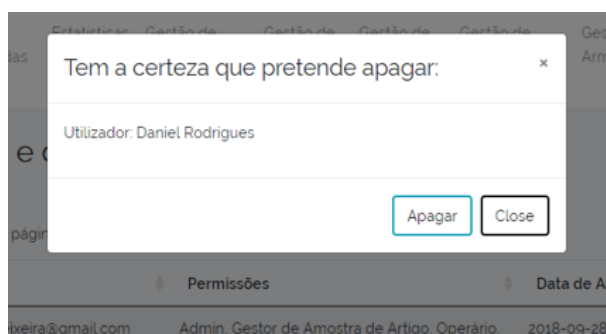


Figure 6.12: *Confirmação ao eliminar utilizador.*

Editar Utilizador

Editar um utilizador permite ao Administrador editar todos os dados de um qualquer utilizador: Nome, Email e as suas Permissões.

Será utilizado essencialmente para destacar as permissões de um utilizador, ou seja, a que funcionalidades terá acesso na sua plataforma como é possível verificar através da figura 6.13.

Editar Utilizador: Alcides Teixeira

Nome:
Alcides Teixeira

Email:
alcides.mn.teixeira@gmail.com

Permissões Seleccionadas:

- Admin
- Convidado
- Gestor de Amostra de Artigo
- Gestor de Encomenda
- Gestor de Armazém
- Operário
- Gestor de Orçamentação

Atualizar Voltar

Figure 6.13: *Editar utilizador.*

6.1.4 Gestão de Encomendas

Criar Encomenda

O processo de criar uma nova encomenda subdivide-se em dois momentos. Isto porque não é certo que a Amostra de Artigo necessária para associar a uma qualquer encomenda já esteja criada no momento em que a encomenda é criada. Inicialmente os campos a preencher serão o Nome do Cliente, o Identificador do Cliente (ou seja, o nome ou código pelo qual o cliente denomina a encomenda solicitada), a Descrição da Encomenda, Upload de Ficheiros (poderão ser diversos ficheiros do tipo png, jpg, gif ou pdf enviados pelo cliente no momento do pedido) e a Data de Entrega da encomenda, como é demonstrado pela figura 6.14.

A encomenda poderá então ser guardada tendo os campos referidos anteriormente preenchidos.

6.1. FUNCIONALIDADES DA APLICAÇÃO

The screenshot shows the 'Criar Nova Encomenda' form. At the top, there is a navigation menu with 'iStocks' and various management options. Below the menu is a progress bar with seven steps. The form fields are: 'Nome do Cliente' (dropdown), 'Identificador do Cliente' (text), 'Descrição' (text), 'Upload Ficheiros' (button), 'Data de entrega' (date), and 'Identificador INDEMEI' (dropdown). Below these are four columns for 'Cor #1' to 'Cor #4', each with a quantity input field. A 'Pedido' row at the bottom shows the total quantity for each color. 'Criar' and 'Limpar' buttons are at the bottom.

Figure 6.14: *Criar encomenda.*

No momento em que a amostra é criada e poderá ser associada a uma encomenda, no campo Identificador INDEMEI do formulário. Imediatamente, serão preenchidos os números associados aos pares de meias (exemplo: 35-38, 39-42, 43-46 e 47-50, por cada uma das 4 linhas zeradas evidentes na imagem 6.14). São também visíveis os campos em que se poderão inserir as quantidades de pares de meias solicitados pela empresa por baixo das colunas da imagem com os nomes Cor #1, Cor #2, Cor #3, Cor #4.

Na última linha da imagem, iniciada com o termo "Pedido", irá surgir o somatório total de pares solicitados, por cada uma das cores.

Ao criar uma nova encomenda e ao serem inseridos os pares de meias totais dessa mesma encomenda, serão atualizados os stocks líquidos do armazém. Esta atualização de stock ocorre sempre que exista uma alteração de quantidades de pares de meias na encomenda (para o caso de a encomenda ser editada mais do que uma vez).

Listar Encomendas

O *layout* apresentado ao listar encomendas permite obter todas as encomendas existentes, bem como algumas das informações gerais e ainda o status em que cada encomenda se encontra.

A partir deste ecrã apresentado na figura 6.15 é ainda possível a edição da encomenda, apagá-la, ou então verificar a produção atual, filtrada por cada um dos

operários.

Lista de Encomendas

Apresentar resultados por página Pesquisar:

Cliente	Id do Cliente	Id da Amostra	Descrição	Criado Em	Data de Entrega	Criado Por	Status			
cliente 1	qwe123	a1	desc çÁ	2018-08-30	2018-09-06	Alcides Teixeira	A criar orçamento	<input type="button" value="Editar"/>	<input type="button" value="Apagar"/>	<input type="button" value="Produção Atual"/>
cliente 1	abc123dfdzfs	a1	descricao	2018-09-06	2018-09-30	Alcides Teixeira	Em produção	<input type="button" value="Editar"/>	<input type="button" value="Apagar"/>	<input type="button" value="Produção Atual"/>
cliente 1	awd		awda	2018-09-26	2018-09-27	Alcides Teixeira	Encomenda Recebida	<input type="button" value="Editar"/>	<input type="button" value="Apagar"/>	<input type="button" value="Produção Atual"/>
Sportzone	Spo1	0001	Encomenda prioritária	2018-09-05	2018-10-31	Cátia Rocha	Produzido	<input type="button" value="Editar"/>	<input type="button" value="Apagar"/>	<input type="button" value="Produção Atual"/>

Página 1 de 1 Anterior Seguinte

Figure 6.15: *Listar encomendas.*

Apagar Encomenda

Apagar uma encomenda carece, tal como no momento de apagar qualquer registo da plataforma, de dupla confirmação antes de poder ser efetivamente eliminada (6.16).

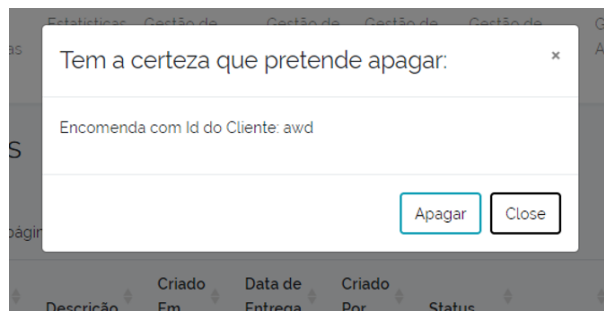


Figure 6.16: *Confirmação antes de apagar encomenda.*

Produção Atual

Ao clicar em "Produção Atual", na Listagem de Encomendas, surge uma tabela com os operários que trabalharam na Encomenda específica e onde é possível clicar no botão "Ver" para verificar o desenvolvimento da encomenda, por operário. Este processo está demonstrado na imagem 6.17.

6.1. FUNCIONALIDADES DA APLICAÇÃO

Cliente	Id do Cliente	Id da Amostra	Descrição	Data de Entrega	Produzido Por	
Sportzone	4	2	Encomenda prioritária	2018-10-31	Alcides Teixeira	Ver
Sportzone	4	2	Encomenda prioritária	2018-10-31	Cátia Rocha	Ver

Figure 6.17: *Produção atual de uma encomenda.*

Enviar Email

O processo de envio de emails, comum a várias permissões de utilizadores (Gestor de Encomendas, Gestor de Amostras de Artigo e Gestores de Orçamentação), permite a possibilidade de enviar um email automático para qualquer pessoa associada à gestão do processo. No caso de um email tiver que ser enviado para qualquer fornecedor ou cliente já inseridos na plataforma, o acesso torna-se mais simples, uma vez que através de um select qualquer um deles poderá ser selecionado (6.18).

Os restantes campos para além do destinatário são a inserção de um endereço ainda não presente no sistema, o Assunto do email e o conteúdo do mesmo, tal como está evidenciado na figura 6.19.

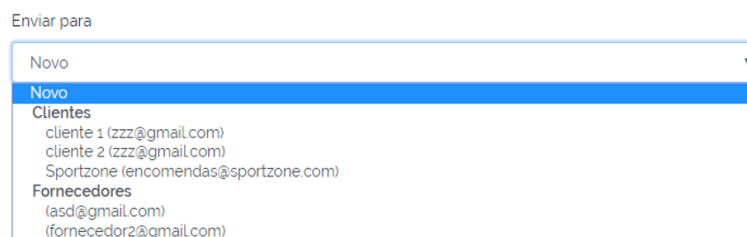


Figure 6.18: *Contactos apresentados ao enviar email.*

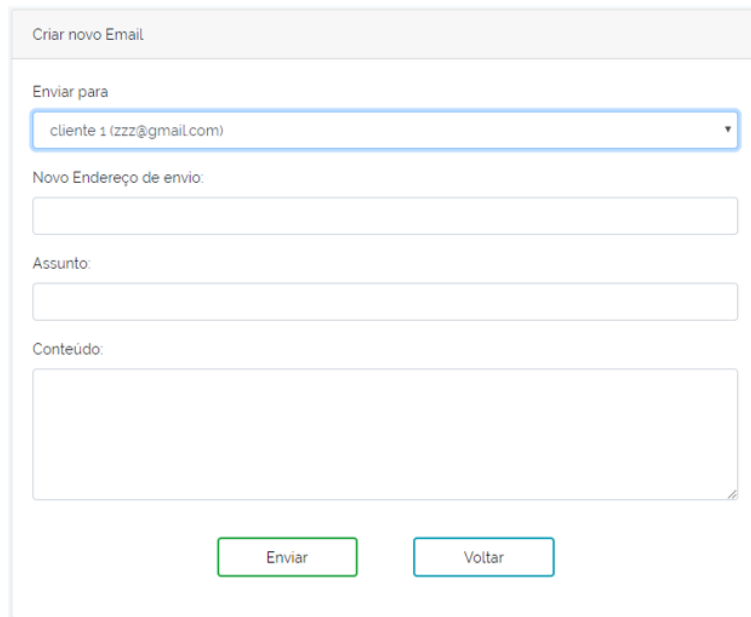


Figure 6.19: *Enviar email.*

Gerir Emails

A gestão de emails por parte da empresa INDMEI é realizada por diversos utilizadores com diversos tipos de permissão (Gestor de Encomendas, Gestor de Amostras de Artigo e Gestor de Orçamentação). Esta gestão é efetuada através de contas criadas no motor de busca "Sapo" e "Outlook". No entanto, antecipando um pouco as necessidades da empresa, foi implementado o acesso para além dos dois motores de busca já referidos, o acesso através do "Gmail" e ainda através do "Webmail" (para o url <http://webmail.indmei.pt/>).

Importa referir que nenhum dos gestores de email permitem de uma forma simples e dinâmica a gestão do seu conteúdo a partir de uma outra plataforma (como o caso da aplicação web que foi aqui desenvolvida - teria de ser desenvolvido algum tipo de acesso com recurso a *OAuth*, ou seja, obter acesso limitado a uma conta de uma plataforma através de outra aplicação web que não a original com recurso a um *token* de acesso) por se tratarem de diferentes domínios da aplicação criada - normalmente referido como *Cross-Origin Resource Sharing* (CORS). Deste modo, como é possível verificar a partir da imagem 6.20, ao clicar em qualquer um dos botões apresentados com o logo de cada um dos motores de busca, o utilizador será redirecionado numa nova aba para a sua caixa de entrada do seu email, onde poderá realizar a gestão de emails recebidos.

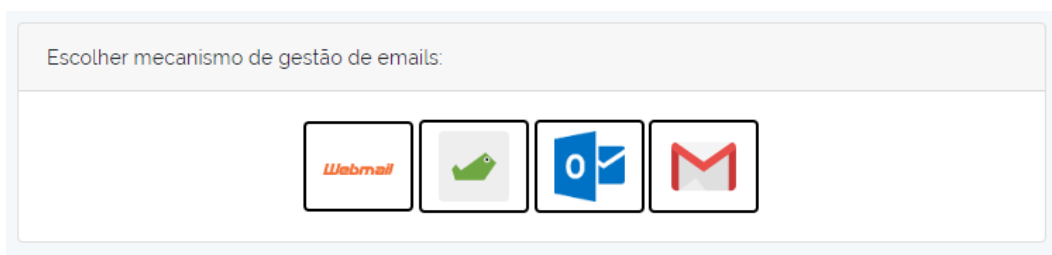


Figure 6.20: *Gerir emails.*

6.1.5 Estatísticas

O *layout* disponibilizado pretende visar de uma forma geral alguns dados estatísticos sob a forma de alguns gráficos.

É possível interpretar os dados ao longo do tempo, sendo que a data inicial e final são editáveis, bem como a forma como a medição será feita, ou seja, por dia, por mês ou por ano.

Entre os gráficos apresentados na imagem 6.21, estes estão organizados da seguinte forma:

1. Gráfico linear, que representa a quantidade total de meias produzidas pela empresa, no período selecionado;
2. Gráfico circular que representa o total de Matéria-Primas utilizada na empresa, desde sempre, até ao corrente dia, em quilos;
3. Gráfico de barras, que pretende representar quais são os clientes da empresa para os quais são produzidas mais meias, desde sempre, até ao corrente dia;
4. Gráfico linear, em que cada linha corresponde a um cliente e que representa a receita total gerada por esse mesmo cliente ao longo do tempo. A janela temporal variará no período selecionado no topo da página.

6.1. FUNCIONALIDADES DA APLICAÇÃO

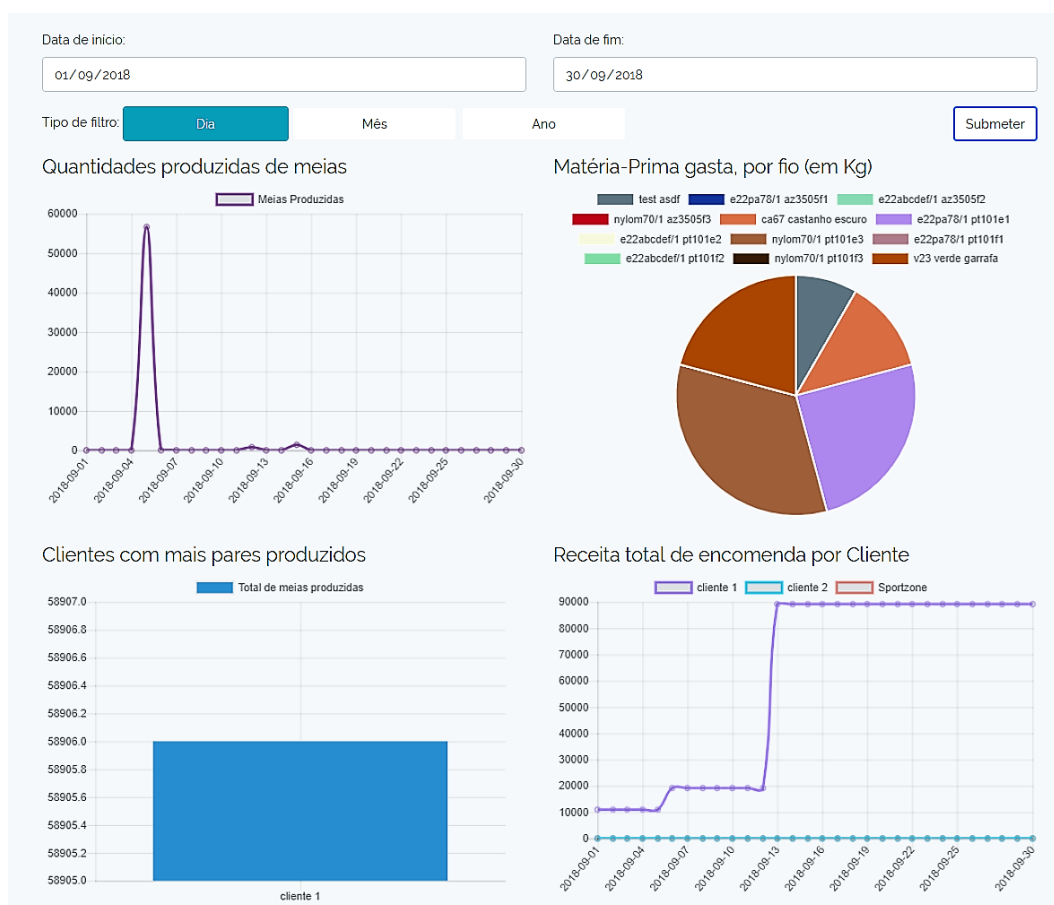


Figure 6.21: Estatísticas da empresa.

É possível interagir com os gráficos apresentados, fazendo variar a data de início e de fim do período que se pretende analisar, fazendo um estudo mais detalhado de cada período. É ainda possível alterar a unidade apresentada, entre dias, meses ou anos e centralizar os resultados por diferentes períodos de tempo.

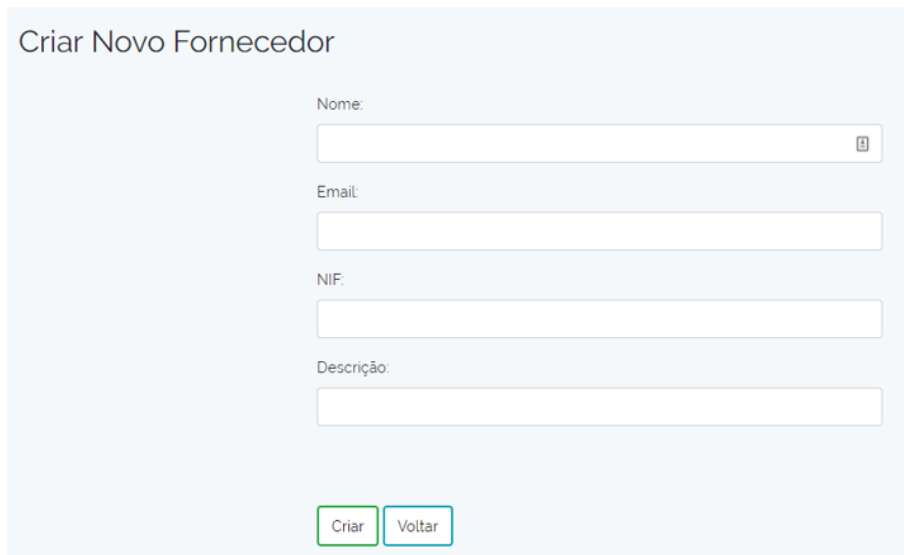
Cada um dos gráficos permite, ao passar o cursor por cima de cada valor, verificar qual o valor exato que está a ser marcado. Para além desta funcionalidade, caso um gráfico possua vários dados a apresentar, ao clicar na cor das legendas é possível mostrar/esconder alguns dos resultados.

6.1.6 Gestão de Fornecedores

Criar Fornecedor

Para criar um novo fornecedor, são necessárias as seguintes informações: Nome, Email, NIF e Descrição (ver figura 6.22). O Fornecedor estará depois disponível para ser contactado mais facilmente através de email da plataforma. Para além

disso ficarão com uma lista de todos os fornecedores num só local, mais organizada e eficiente.



The image shows a web form titled "Criar Novo Fornecedor". It contains four input fields: "Nome:" (with a small icon on the right), "Email:", "NIF:", and "Descrição:". At the bottom of the form are two buttons: "Criar" (green border) and "Voltar" (blue border).

Figure 6.22: *Criar fornecedor.*

Listar Fornecedores

Ao listar todos os fornecedores da empresa, é possível ter acesso às principais informações sobre cada um deles. É ainda possível editar cada um dos fornecedores previamente adicionados e também apagá-lo (6.23).



The image shows a table titled "Lista de Fornecedores". At the top left, there is a dropdown menu for "Apresentar" set to "25" and the text "resultados por página". At the top right, there is a search box labeled "Pesquisar:". The table has five columns: "Fornecedor", "Email", "NIF", "Descrição", and "Data de Alteração". There are two rows of data. Each row has two buttons: "Editar" (yellow border) and "Apagar" (red border). At the bottom left, it says "Página 1 de 1". At the bottom right, there are navigation buttons: "Anterior", "1" (highlighted), and "Seguinte".

Fornecedor	Email	NIF	Descrição	Data de Alteração		
fornecedor 1	asd@gmail.com	123456789	fornecedor de braga	2018-08-29 22:17:02	Editar	Apagar
fornecedor 2	fornecedor2@gmail.com	123456789	fornecedor de viana do castelo	2018-09-30 20:20:49	Editar	Apagar

Figure 6.23: *Listar fornecedores.*

Eliminar Fornecedor

No momento em que se pretende apagar um fornecedor, é realizada uma dupla confirmação antes de ser eliminado permanentemente da base de dados (6.24).

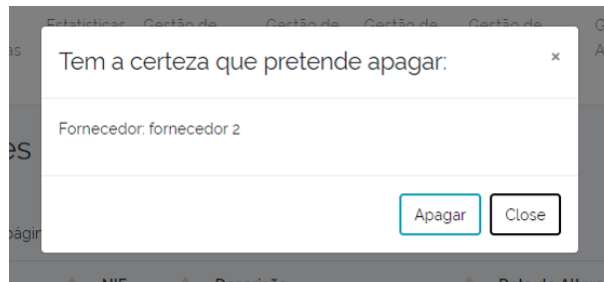


Figure 6.24: *Eliminar fornecedor.*

6.1.7 Gestão de Clientes

Criar Cliente

De um modo muito semelhante ao da criação de um fornecedor, para criar um novo cliente, são necessárias as seguintes informações: Nome, Email, NIF e Descrição (6.25). O Cliente estará depois disponível para ser contactado mais facilmente através de email da plataforma, ou então para ser associado a qualquer encomenda que seja criada em seu nome.

A screenshot of a web form titled "Criar Novo Cliente". The form has a light blue background and contains four input fields: "Nome do Cliente:", "Email:", "NIF:", and "Descrição:". Each field is a simple white rectangle with a thin border. At the bottom of the form, there are two buttons: "Criar" (highlighted in green) and "Voltar" (highlighted in blue).

Figure 6.25: *Criar cliente.*

Listar Clientes

Listar Clientes da empresa permite obter uma visão geral dos dados de cada cliente. É ainda possível editar os dados de cada um deles, bem como eliminar, através dos botões disponibilizados para o efeito (6.26).

6.1. FUNCIONALIDADES DA APLICAÇÃO

Lista de Clientes

Apresentar resultados por página Pesquisar:

Cliente	Email	NIF	Descrição	Data de Alteração		
cliente 1	zzz@gmail.com	123456789	cliente de braga	2018-08-29 22:17:02	<input type="button" value="Editar"/>	<input type="button" value="Apagar"/>
cliente 2	zzz@gmail.com	123456789	cliente de faro	2018-08-29 22:17:02	<input type="button" value="Editar"/>	<input type="button" value="Apagar"/>
Sportzone	encomendas@sportzone.com	506086380	Fornecedor importante. Dar prioridade	2018-09-05 21:10:38	<input type="button" value="Editar"/>	<input type="button" value="Apagar"/>

Página 1 de 1 Anterior Seguinte

Figure 6.26: *Listar clientes.*

Eliminar Cliente

Para que um cliente seja permanentemente eliminado da base de dados, este terá de ser duplamente confirmado no momento em que se clica no botão eliminar, tal como é demonstrado na mensagem de confirmação da imagem 6.27.

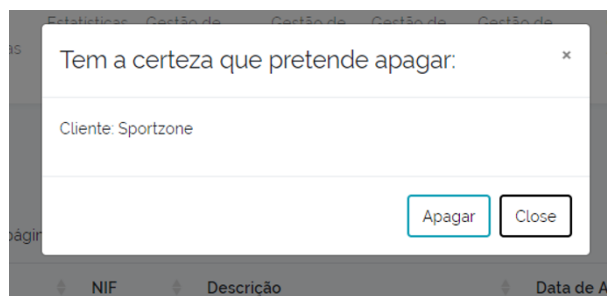


Figure 6.27: *Confirmação ao eliminar cliente.*

No caso de existirem encomendas pertencentes a um determinado cliente que se pretende apagar, o sistema considera que se poderá tratar de um erro e então o cliente neste caso não será eliminado, dando uma indicação ao utilizador. É solicitado ao utilizador que substitua o cliente de todas as encomendas a que está associado por outro, para que depois sim, possa ser eliminado. A imagem 6.28 ilustra este mesmo alerta.

Atenção! O Cliente não pode ser eliminado pois está associado a alguma encomenda!
 Altere o fornecedor da encomenda antes de poder apagar o fornecedor!

Lista de Clientes

Apresentar resultados por página Pesquisar:

Cliente	Email	NIF	Descrição	Data de Alteração	
cliente 1	zzz@gmail.com	123456789	cliente de braga	2018-08-29 22:17:02	<input type="button" value="Editar"/> <input type="button" value="Apagar"/>
cliente 2	zzz@gmail.com	123456789	cliente de faro	2018-08-29 22:17:02	<input type="button" value="Editar"/> <input type="button" value="Apagar"/>
Sportzone	encomendas@sportzone.com	506086380	Fornecedor importante. Dar prioridade	2018-09-05 21:10:38	<input type="button" value="Editar"/> <input type="button" value="Apagar"/>

Página 1 de 1 Anterior Seguinte

Figure 6.28: Mensagem de erro ao eliminar cliente.

6.1.8 Gestão de Amostras

Criar Nova Amostra de Artigo

Criar uma amostra de artigo segue um *layout*, representado na figura 6.29 que à partida poderá ser um pouco "complexo". Esta estrutura de página foi mantida da seguinte forma pelo facto de possuir uma organização idêntica àquela já utilizada pelos colaboradores da empresa. Sendo esta uma característica a ter em conta para a realização de algumas estruturas de páginas ao longo do projeto, não poderia ter sido deixada de parte e, portanto, foram adaptadas algumas medidas para alinhar um pouco a organização, sendo que a forma se manteve, na sua maioria.

A estrutura da página de criação de uma amostra necessita então de diversas informações para que fique o mais completa possível. Inicialmente são introduzidas a referência e descrição do novo artigo, sendo depois solicitada uma imagem que represente a meia a ser criada.

A segunda parte da página apresenta campos para quatro tamanhos distintos da meia (ex: 35-38, 39-42, 43-36, 47-50) e as características de seis componentes da criação da amostra para cada um dos tamanhos: pé, perna, punho, malha, máquina e forma.

O quadro final da página (ainda na figura 6.29) representa os passos necessários para criar a meia. As colunas representam as ferramentas, quantidades, passos e matérias primas necessárias para cada uma das meias:

6.1. FUNCIONALIDADES DA APLICAÇÃO

- guiafios;
- step;
- gramas;
- referência do fio;
- cor #1;
- cor #2;
- cor #3;
- cor #4.

Guiafios	Step	Gramas	Referência do Fio	Cor #1	Cor #2	Cor #3	Cor #4
crava	G1	0	e22pa78/1	pt101e1	pt101e1	pt101e1	pt101e1
crava	G2	0	e22pa78/1	pt101e1	pt101e1	pt101e1	pt101e1
crava	G3	0	e22pa78/1	pt101e1	pt101e1	pt101e1	pt101e1

Figure 6.29: Criar amostra de artigo.

Os *steps* representados na tabela do fundo da figura 6.29 são a quantidade máxima de passos enumerados pela empresa, ou seja desde o G1 a G8, Punho e BR1 a BR8. Uma medida evidente a inserir neste ponto poderia passar pela adição de uma linha à medida que a amostra fosse preenchida. Esse passo foi tido em conta e considerou-se não ser o ideal para este caso uma vez que a empresa se encontra efectivamente habituada a este modelo de apresentação e preenchimento dos steps da encomenda.

Listar Amostras de Artigos

Listar as amostras de artigo permite ter uma noção das amostras que o gestor de amostras de artigos possui.

Permite ainda ver algumas informações gerais sobre as mesmas, nomeadamente a sua referência, descrição, a imagem associada à amostra e que a realizou. Juntamente com a data da última atualização, existe a possibilidade de executar duas

ações: editar a amostra ou então eliminá-la. A figura 6.30 representa todas as ações indicadas anteriormente.

Lista de Amostras de Artigos disponíveis

Apresentar 25 resultados por página

CSV Excel PDF Imprimir

Pesquisar:

Referência	Descrição	Imagem	Executante	Última Atualização		
0001	amostra para sportzone		Cátia Rocha	2018-09-08 15:57:31	Editar	Apagar
a1	amostra da encomenda x		Alcides Teixeira	2018-09-27 00:15:36	Editar	Apagar

Página 1 de 1

Anterior 1 Seguinte

Figure 6.30: *Listar amostras de artigo.*

Da mesma forma que na gestão de encomendas, também aqui a tentativa de apagar uma amostra terá uma dupla confirmação, para que não seja erradamente eliminada.

Também a gestão de emails (através do redirecionamento para a sua conta de email), bem como a sua criação (dentro da aplicação web) são processos que são permitidos ao Gestor de Amostras de Artigos.

Ao Gestor de Amostras é permitido também listar encomendas existentes. A funcionalidade de editar uma encomenda encontra-se ativa. Contudo, não pode ser alterada por este tipo de permissão, servindo apenas como um modo de ver detalhes de uma determinada encomenda sem que tenha a possibilidade de a poder alterar.

6.1.9 Gestão de Orçamentação

Listar Orçamentos

A listagem de orçamentos (figura 6.31) permite listar todas as encomendas e verificar em que estado é que se encontram. Caso uma encomenda esteja "A Criar Orçamento", significa que o gesto de orçamentação terá de criar um orçamento e enviar ao cliente e que a encomenda está a aguardar a sua resposta para poder prosseguir.

6.1. FUNCIONALIDADES DA APLICAÇÃO

Lista de Encomendas

Apresentar 25 resultados por página

Pesquisar

Cliente	Id do Cliente	Id da Amostra	Descrição	Criado Em	Data de Entrega	Criado Por	Status					
cliente 1	qw123	a1	desc çA	2018-08-30	2018-09-06	Alcides Teixeira	A criar orçamento	Editar	Apagar	Produção Atual	A minha produção	Ver Orçamentação
cliente 1	Spo1	0001	Encomenda prioritária	2018-09-05	2018-10-31	Cátia Rocha	Produzido	Editar	Apagar	Produção Atual	A minha produção	Ver Orçamentação
cliente 1	abc123dfdzfs	a1	descricao	2018-09-06	2018-09-30	Alcides Teixeira	Em produção	Editar	Apagar	Produção Atual	A minha produção	Ver Orçamentação
cliente 1	awd	0001	awda	2018-09-26	2018-09-27	Alcides Teixeira	Em produção	Editar	Apagar	Produção Atual	A minha produção	Ver Orçamentação

Página 1 de 1

Anterior 1 Seguinte

Figure 6.31: Listar orçamentos.

Criar Orçamento

Ao criar um orçamento de uma determinada encomenda, o gestor de orçamentação terá acesso a diversos campos previamente preenchidos pelo gestor da encomenda e que poderão auxiliar a ter uma melhor ideia da encomenda a orçarmentar. São estes campos compostos por:

- Status da encomenda;
- Nome do cliente;
- Descrição;
- Identificador do Cliente;
- Data de Entrega.

É possível ainda ver qual a amostra de artigo e em que consiste a mesma. Depois de toda a informação assimilada, são apresentados então os campos com os valores correspondentes.

O cálculo do orçamento total poderá ser feito através da plataforma, sendo que cada amostra possui um valor que depois é multiplicado pelo seu total de pares. É então depois somado o valor de diversas variáveis que serão incluídas no orçamento. São elas o custo das etiquetas, o custo das caixas, a percentagem total de peças com defeito a adicionar, o custo da mão-de-obra e ainda outros custos. A soma deste valor é apresentado no campo "Total" logo a seguir.

Por um motivo ou por outro, o gestor de orçamentação poderá querer enviar um valor total diferente do resultado obtido através do cálculo. Por esse motivo é dada a possibilidade de adicionar o valor de orçamentação a enviar para o cliente. A figura

6.32 demonstra o processo de criação do orçamento.

Atualizar Orçamentação

Status:
Produzido

Nome do Cliente: cliente 1 Identificador do Cliente: Sp01

Descrição: Encomenda prioritária Data de entrega: 31/10/2018

Identificador INDMEI: Ver Amostra de Artigo

Amostra de Cor 'preto':	2.19	€	Total Pares:	506
Amostra de Cor 'rosa':	2.19	€	Total Pares:	214
Amostra de Cor 'verde':	2.19	€	Total Pares:	1751
Amostra de Cor 'amarelo':	2.19	€	Total Pares:	234
Custo das etiquetas:	155	€		
Custo das Caixas:	233	€		
Peças com defeito:	11	%		
Mão-de-Obra:	1235	€		
Outros Custos:	22	€		
Total:	69467,11	€		
Total a Enviar:	70000	€		

Atualizar e ir para Enviar Email Voltar

Figure 6.32: Criar orçamento.

Ao clicar em "Atualizar e ir para Enviar Email", é apresentado um template de email onde constam todas as informações que o gestor de orçamentação inseriu anteriormente, sendo apenas necessário editá-las da forma que mais convier. A imagem 6.33 apresenta este template de email.

Enviar Orçamentação

Enviar para
Novo

Novo Endereço de envio:

Assunto:

Conteúdo:
Custo total: 61101.
Etiquetas: 155.
Caixas: 233.
Defeito: 11.
Mão de obra: 1235.
Outros Custos: 22.
Total: 70000

Enviar Voltar

Figure 6.33: *Email de envio de orçamento.*

O Gestor de orçamentação poderá ainda utilizar funcionalidades que já foram descritas anteriormente no gestor de encomendas, tais como:

- Enviar Emails (através da aplicação web desenvolvida);
- Gerir Emails (através do redirecionamento existente na aba "Gerir emails" da aplicação).

6.1.10 Gestão de Armazém

Dar Entrada de Stock

O ecrã que permite dar a entrada de novo stock no armazém da empresa permite não só atualizar stock existente, mas também adicionar nova matéria-prima para o caso de esta nunca ter sido necessária anteriormente (clicando no botão "Criar nova Matéria-Prima").

Os campos a preencher são em ambos os casos: descrição, referência e cor, peso em Kg e o seu custo. Caso seja uma nova matéria-prima, é solicitado o valor mínimo de aviso para não haver o risco de que alguma matéria-prima se esgote enquanto necessária. A imagem 6.34 representa a entrada ou saída de stock do armazém.

Esta funcionalidade que permite dar a entrada de stock, permite ainda também a sua saída. Isto porque também poderá haver a necessidade de em algum momento a empresa necessite de libertar stock, ou por falhas de contagem humana, ou por

material que vai para quebras de stock, entre outros motivos.

À medida que são preenchidos e adicionados, é formada uma lista na parte inferior da página. Esta lista será a lista que se irá submeter com todas as matérias-primas a dar entrada.

É ainda solicitada uma imagem ou ficheiro PDF para associar à entrada de stock. Isto porque o stock vem associado a um documento, uma factura ou outro tipo de validação que confirma as quantidades e as matérias-primas encomendadas.

Dar Entrada de Stock

Entrada/Saida:

Descrição:

Referência:

Cor:

Peso (Kg):

Custo por Kg:

Entrada/Saida	Referencia	Cor	Qty (Kg)	Custo (€/Kg)	Descrição	Limite minimo
<input type="text" value="IN"/>	<input type="text" value="e22pa78/1"/>	<input type="text" value="az3505f1"/>	<input type="text" value="110"/>	<input type="text" value="2"/>	<input type="text" value="descricao"/>	<input type="text"/>
<input type="text" value="IN"/>	<input type="text" value="v23"/>	<input type="text" value="verde garrafa"/>	<input type="text" value="500"/>	<input type="text" value="3"/>	<input type="text" value="amostra para sportzone"/>	<input type="text"/>

Carregar Fatura (.pdf, .png, .jpeg, .gif): 300.png

Figure 6.34: *Dar entrada de stock.*

Criar Nova Matéria-Prima

Criar uma nova matéria-prima consiste no preenchimento de alguns campos, nomeadamente: descrição da matéria prima, referência, cor, peso bruto e peso líquido em Kg e ainda o valor de aviso mínimo no qual se pretende receber um aviso de que a matéria-prima estará prestes a esgotar-se. A imagem 6.35 representa a criação de matéria-prima nova.

Form for creating a new raw material. The form contains the following fields and buttons:

- Descrição:
- Referência:
- Cor:
- Peso Bruto (gramas):
- Peso Líquido (gramas):
- Valor de aviso mínimo (Kg):
- Custo por Kg:
- Buttons:

Figure 6.35: *Criar nova matéria-prima.*

Listar Stock

Listar o Stock de Armazém possui uma série de informações relevantes para o Gestor de Armazém, quer visuais, quer ao nível quantitativo.

De um modo geral, é possível observar todas as matérias-primas existentes no armazém, a sua referência e a sua cor, o stock bruto (ou seja, o stock que existe efectivamente em armazém e que é subtraído diariamente no momento em que as encomendas são atualizadas), o stock líquido (o stock em armazém, subtraindo o valor total necessário para terminar todas as encomendas) e ainda o custo por Kg de cada matéria-prima. Surge ainda a data da última atualização e quem a realizou. Existe ainda a possibilidade de editar, apagar e pedir reforço de stock.

Salienta-se o facto de determinadas linhas da tabela possuírem diferentes cores. As linhas marcadas a tons de vermelho simbolizam que o stock bruto se encontra abaixo do valor de alerta mínimo indicado para cada matéria-prima. Isto serve de precaução para que o gestor de stock possa pedir atempadamente stock e assim nenhuma encomenda necessite de parar devido a falta do mesmo.

A imagem 6.36 ilustra a tabela descrita previamente.

6.1. FUNCIONALIDADES DA APLICAÇÃO

Stock de produtos disponíveis

Apresentar 10 resultados por página

CSV Excel PDF Imprimir

Pesquisar:

Referência	Cor	Stock Bruto (Kg)	Stock Líquido (Kg)	Alerta mínimo (Kg)	Custo (€/Kg)	Atualizado Por	Última Atualização			
ca67	castanho escuro	2514571	25113514	100	1.03	Cátia Rocha	2018-09-27 22:40:15	Editar	Apagar	
e22abcdef/1	pt101e2	49.955	-22525.427	502	15	Alcides Teixeira	2018-09-26 23:14:45	Editar	Apagar	Pedir stock
e22abcdef/1	az3505f2	0	-1.004	500	0	Alcides Teixeira	2018-09-26 23:14:45	Editar	Apagar	Pedir stock
e22abcdef/1	pt101f2	0	-18138.532	500	0	Alcides Teixeira	2018-09-26 23:14:45	Editar	Apagar	Pedir stock
e22pa78/1	pt101e1	-338.85	-86507.481	500	0	Alcides Teixeira	2018-09-26 23:14:45	Editar	Apagar	Pedir stock
e22pa78/1	az3505f1	181.185	180.827	181	1.1	Alcides Teixeira	2018-09-26 23:17:50	Editar	Apagar	
e22pa78/1	pt101f1	0	-0.251	500	0	Alcides Teixeira	2018-09-26 23:14:45	Editar	Apagar	Pedir stock
nylom70/1	pt101e3	-391.44	-400.057	500	23	Alcides Teixeira	2018-09-26 23:14:45	Editar	Apagar	Pedir stock
nylom70/1	az3505f3	-0.03	-49878.959	500	0	Alcides Teixeira	2018-09-26 23:14:45	Editar	Apagar	Pedir stock
nylom70/1	pt101f3	0.135	-9073.071	500	100	Alcides Teixeira	2018-09-26 23:14:45	Editar	Apagar	Pedir stock

Página 1 de 2

Anterior 1 2 Seguinte

Figure 6.36: *Listar stock.*

Ao clicar em cada uma das linhas da figura 6.36, é especificada a informação sobre cada matéria-prima através de um histórico de entradas e saídas da mesma. As entradas serão obtidas através dos pedidos de stock e as saídas estão associadas à sua utilização para encomendas. A imagem 6.37 ilustra este mesmo histórico.

Histórico de Matéria-Prima

Entrada/Saída	Quantidade (g)	Custo (€)	Descrição	Atualizado Por	Última Atualização	Fatura
OUT_LIQUID	0	N/A	Encomenda para o cliente: cliente 1 com o identificador: qwe123	Alcides Teixeira	2018-10-02 21:28:19	jpg
OUT_GROSS	0	N/A	Encomenda para o cliente: cliente 1 com o identificador: qwe123	Alcides Teixeira	2018-10-02 21:28:19	jpg
OUT_LIQUID	251	N/A	Encomenda para o cliente: cliente 1 com o identificador: abc123dfdzfs	Alcides Teixeira	2018-10-02 21:28:19	jpg
OUT_GROSS	0	N/A	Encomenda para o cliente: cliente 1 com o identificador: abc123dfdzfs	Alcides Teixeira	2018-10-02 21:28:19	jpg

Fechar

pt101f1 0 -0.251 500 0 Alcides Teixeira 2018-09-26 23:14:45 Editar Apagar

Figure 6.37: *Histórico de matéria-prima.*

Eliminar Matéria-Prima

Havendo a necessidade de eliminar uma amostra, existe uma dupla confirmação para que um artigo não seja acidentalmente eliminado da base de dados (6.38).

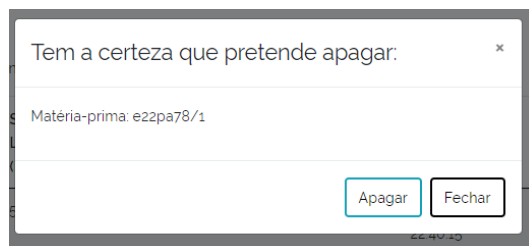


Figure 6.38: Confirmação ao eliminar matéria-prima.

Solicitar Matéria-Prima

Ao clicar em "Pedir Stock", surge um *template* de email, mas com a ressalva de que o conteúdo virá pré-preenchido com a amostra da qual se solicitou novo *stock*. Este email poderá posteriormente ser editado para a mensagem que mais convier ao gestor de armazém. Possui ainda um campo para escolher o destinatário do email, bem como o assunto do email (figura 6.39).



Figure 6.39: Solicitar matéria-prima.

6.1.11 Encomendas em Produção

Produção

No momento em que o Operário abre a sua produção de uma encomenda, tem a possibilidade de ver diversas informações acerca da encomenda que está a produzir.

No topo da página representada na figura 6.40 surgem informações gerais da encomenda: a imagem da amostra; o identificador da amostra; e um quadro com o total de pares de meias a produzir que foram solicitados pelo cliente, bem como o número de pares em falta atualmente.

De seguida, surgem informações sobre como é produzida cada uma das meias, identificadas com um quadro de criação de uma amostra.

Na parte mais próxima do final (ainda na figura 6.40), surge um quadro que representa a produção diária do operário. Neste quadro, a cada dia surge uma nova linha onde o operário poderá colocar o valor diário. Uma linha nova deixará de surgir caso a data de entrega da encomenda seja atingida. Neste momento uma de duas coisas poderá acontecer:

1. Se a quantidade de meias produzidas em falta atingir o valor zero, então a célula é marcada a verde;
2. Se a quantidade de meias produzidas em falta for superior a zero quando a data de entrega da encomenda é atingida, então a célula é pintada de vermelho.

Caso todos tamanhos e cores de meias sejam terminadas e a linha ficar a verde (como na imagem abaixo), é enviado um email para o gestor de encomendas, gestor de orçamentação e gestor de amostras com a informação de que uma encomenda foi terminada.

Refira-se que o layout desta página encontra-se estruturado da seguinte forma uma vez que se trata do formato que a empresa reconhece desde o momento em que ainda era utilizada em folha excel. Por esse motivo, as alterações de estrutura efetuadas foram no sentido de organizar a informação melhor, mas mantendo a estrutura original o mais semelhante possível.

Por forma a desenvolver um *layout* orientado para o utilizador final específico, a página foi mantida desta forma, apesar de possuir uma grande quantidade de informação num só local.

6.2 Gestão dos erros

De forma inevitável, a plataforma poderá vir a dar alguns erros. Isto acontece pois não é possível prever que acções poderá o utilizador tomar no futuro, deixando a plataforma suscetível a erros.

Por forma a combater, ou então minimizar estes mesmos erros, foram desenvolvidas duas medidas.

A plataforma beneficia de uma página "404", ou seja, no momento em que um utilizador é desviado do link que pretendia seguir, o utilizador é reencaminhado para uma página preparada para o efeito. A página, que se encontra na figura 6.41, possui depois um botão em que o utilizador poderá clicar para ser redirecionado novamente para dentro da plataforma.



Figure 6.41: *Erro 404 - page not found.*

Uma segunda medida preventiva e de auxílio na gestão de possíveis erros foi desenvolvida e trata-se de um sistema de envio de mensagens de erro, com a sua descrição e mais algumas informações acerca do mesmo.

O ecrã apresentado na figura 6.42 fornece ainda ao utilizador do software um ambiente *user friendly*, com algumas indicações.

É indicada qual a descrição principal do erro, seguida de uma mensagem a indicar o envio de um email de forma automática para um email pessoal do desenvolvedor. É ainda adicionada a indicação de que o utilizador poderá insistir no envio de um print screen do erro para o caso de se tratar de um erro que impeça o utilizador de proceder com a funcionalidade que pretende executar.

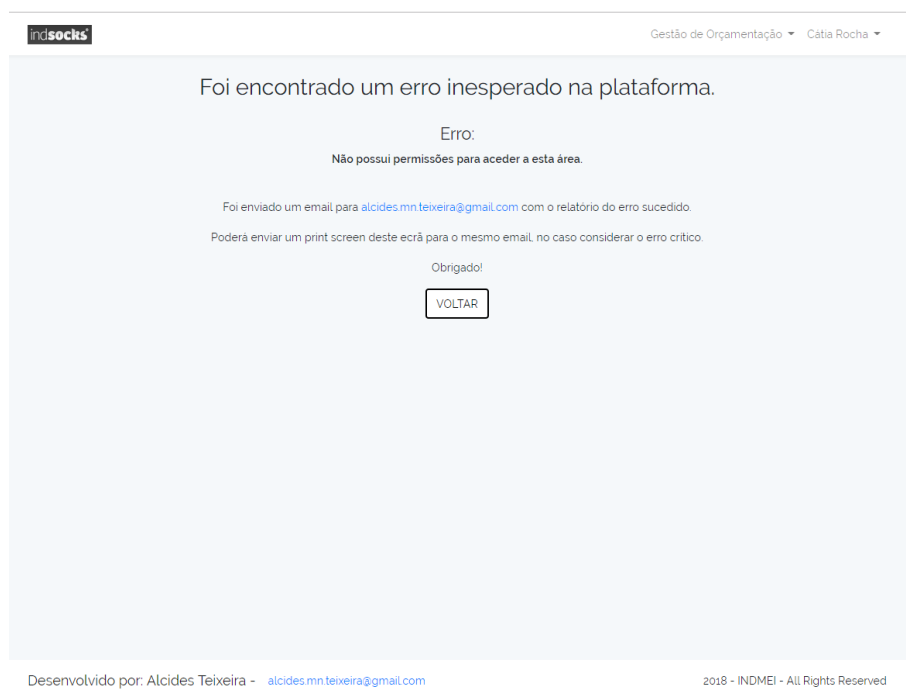


Figure 6.42: *Erro inesperado na plataforma.*

O email recebido por parte do *developer* apresenta algumas informações para que o erro possa ser replicado por ele e assim tentar resolver a fonte do problema de forma mais eficiente. As informações contempladas na mensagem enviada são:

1. Uma mensagem do erro;
2. O ficheiro em que o erro ocorre;
3. A linha do ficheiro, para conseguir determiná-lo mais especificamente;
4. O URL que o utilizador estaria a navegar no momento em que o erro ocorreu.

Todos estes dados podem ser visualizados na imagem 6.43.



Figure 6.43: *Email de erro inesperado na plataforma.*

6.3 Notificações via Email

Um utilizador poderá sempre seguir o estado de uma encomenda através da plataforma, desde que para isso tenha permissões, ou seja, um utilizador com permissões de Admin, Gestor de Encomenda, Gestor de Amostra de Artigo, Gestor de Orçamentação e Operário.

De outro modo e caso existam demasiadas encomendas para identificar o estado de todas, foram gerados alertas de notificação via email para uma melhor gestão das encomendas, como se ilustra na figura 6.44.

Notificação de Encomenda à espera de amostra:

Esta notificação consiste num email enviado para todos os utilizadores com a permissão de Gestor de Amostra de Artigo, quando o Gestor de Encomenda muda o estado de uma encomenda para "A Produzir Amostra". O email contém as informações gerais da encomenda que se encontra à espera de uma amostra para ser associada, juntamente com um link de redirecionamento para a encomenda específica. A imagem 6.44 representa este mesmo email.

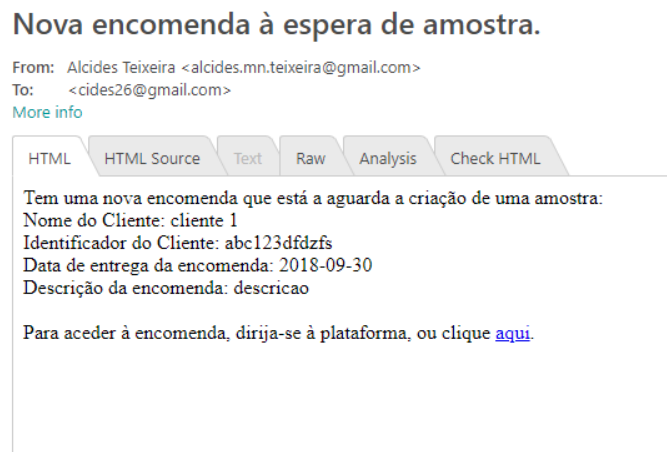


Figure 6.44: *Notificação de encomenda à espera de amostra.*

Notificação de Nova amostra criada:

Quando um Gestor de Encomenda possui encomendas à espera de uma amostra para poder ser associada, irá receber uma notificação via email com a indicação de que uma amostra foi criada e encontra-se pronta a utilizar nas suas encomendas.

É enviado para todos os utilizadores com a permissão Gestor de Encomenda, no momento em que a amostra é criada pelo Gestor de Amostra.

Este email contém as informações básicas da amostra produzida, juntamente com

uma imagem da amostra da meia produzida, que é associada no momento em que a amostra é criada. É ainda fornecido o link para aceder rapidamente à amostra finalizada. A imagem 6.45 representa o email recebido pelo Gestor de Encomendas.

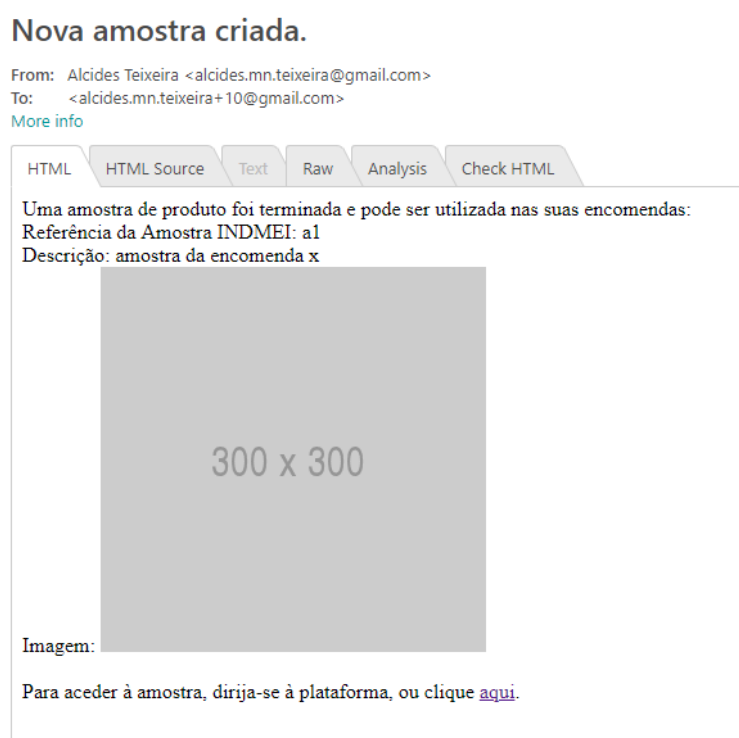


Figure 6.45: *Notificação de nova amostra criada.*

Notificação de encomenda a aguardar orçamentação:

Esta notificação consiste num email enviado para todos os utilizadores com a permissão de Gestor de Orçamentação, quando o Gestor de Encomenda muda o estado de uma encomenda para "A Criar Orçamento". O email contém as informações gerais da encomenda que se encontra à espera de uma amostra para ser associada, juntamente com um link de redirecionamento para a encomenda específica.

A imagem 6.46 representa esse email recebido por um Gestor de Orçamentação.

Nova encomenda à espera de orçamentação.

From: Alcides Teixeira <alcides.mn.teixeira@gmail.com>

To: <catia_rocha@live.com.pt>

[More info](#)

HTML

HTML Source

Text

Raw

Analysis

Check HTML

Tem uma nova encomenda que está a aguarda o envio de orçamentação:

Nome do Cliente: cliente 1

Identificador do Cliente: abc123dfdzfs

Data de entrega da encomenda: 2018-09-30

Descrição da encomenda: descricao

Para aceder à encomenda, dirija-se à plataforma, ou clique [aqui](#).

Figure 6.46: Notificação de encomenda à espera de orçamentação.

Notificação de Orçamento efetuado:

Quando um Gestor de Encomenda possui encomendas à espera de um orçamento efetuado, irá receber uma notificação via email com a indicação de que um orçamento foi associado a determinada encomenda.

É enviado para todos os utilizadores com a permissão Gestor de Encomenda, no momento em que o orçamento é associado pelo Gestor de Orçamentação.

Este email contém as informações básicas da encomenda com orçamento associado, juntamente com o link para aceder rapidamente à encomenda.

A imagem 6.47 representa um exemplo de email recebido pelo Gestor de Encomendas.

Novo orçamento efetuado.

From: Alcides Teixeira <alcides.mn.teixeira@gmail.com>

To: <alcides.mn.teixeira+10@gmail.com>

[More info](#)

HTML

HTML Source

Text

Raw

Analysis

Check HTML

Foi efetuada a orçamentação do produto descrito abaixo, e enviada ao cliente:

Identificador do Cliente: abc123dfdzfs

Cliente: cliente 1

Para aceder à encomenda, dirija-se à plataforma, ou clique [aqui](#).

Figure 6.47: Notificação de orçamento efetuado.

Notificação de encomenda em produção:

Esta notificação tem o objetivo de indicar aos utilizadores com a permissão de Operário que uma encomenda está aprovada e pronta para iniciar a produção.

Esta notificação é despoletada no momento em que o Gestor de Encomendas muda o estado de uma encomenda para "Em Produção" e será recebida por todos os operários.

O email enviado contém informações acerca da encomenda, bem como o link para aceder mais facilmente à encomenda referida.

A imagem 6.48 exemplifica a notificação descrita.

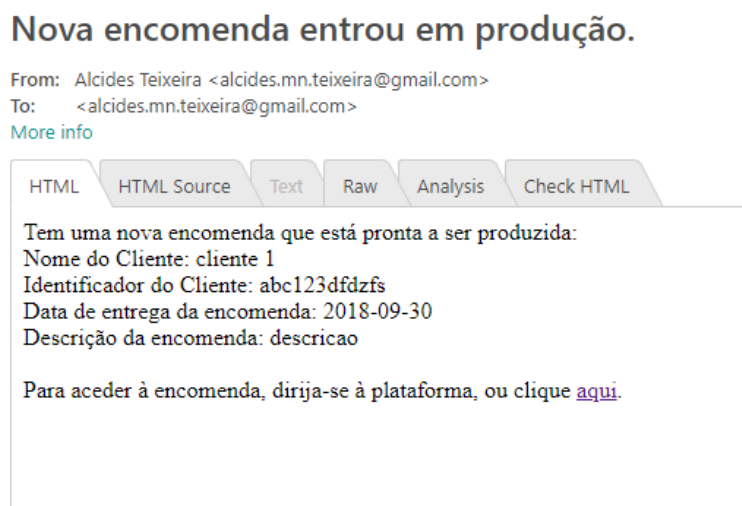


Figure 6.48: *Notificação de encomenda em produção.*

Notificação de encomenda finalizada:

Este email de notificação é recebido pelo Gestor de Encomendas, no momento em que uma encomenda é terminada. No momento em que um operário insira na sua produção diária um valor que resulte numa quantidade de meias em falta igual a zero, isto significa que a encomenda está terminada e o email despoleta.

O email consiste em informações relacionadas com a encomenda acabada, bem como um link direto para a mesma encomenda.

A imagem 6.49 descreve esta mesma notificação.

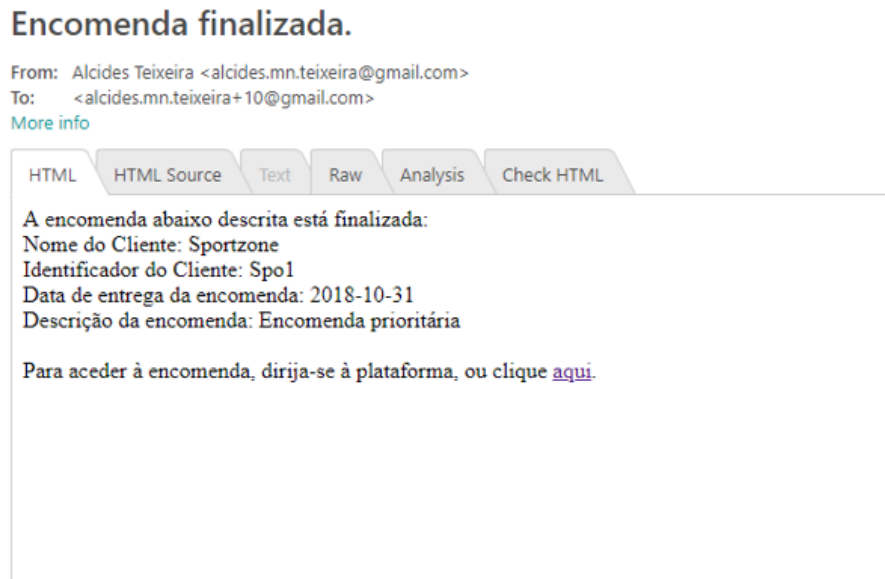


Figure 6.49: Notificação de encomenda finalizada.

6.4 Feedback da Empresa

A empresa INDMEI foi mantida sempre a par de todo o desenvolvimento realizado. O principal objetivo era tornar o *software* uma ferramenta de uso diário da empresa e que permitisse otimizar todos os processos envolvidos. Por este motivo, a obtenção de *feedback* constante permitiu criar uma aplicação web à medida das necessidades da empresa e dos seus utilizadores.

À medida que as funcionalidades iam sendo criadas e apresentadas aos colaboradores da empresa, as primeiras reações à partida foram bastante positivas! Inicialmente ao apresentar as primeiras funcionalidades desenvolvidas, foram solicitadas certas características acessórias, que resultaram por exemplo no desenvolvimento mais completo da gestão de clientes e de fornecedores.

No momento em que o *software* foi instalado no ambiente da empresa, e preparado para realização de testes intensivos por parte da mesma, alguns problemas de compatibilidade surgiram. Estes problemas foram facilmente solucionados, passando por um pormenor de pouca importância, como o uso do *Google Chrome*, uma vez que é o *browser* onde todos os testes foram realizados ao longo do desenvolvimento. No entanto, isto não invalida que possa ser usado outro *browser*.

Foi enviado um breve formulário para a INDMEI para realizar um estudo de validação, com diversas questões que focavam principalmente o projeto desenvolvido e as suas funcionalidades. O público alvo deste estudo passa por toda a empresa, em

especial os intervenientes que lidam com o *software* com maior regularidade. As respostas possíveis eram na sua grande maioria do tipo fechado, para que fossem possíveis de analisar de forma mais eficiente. Os resultados apresentados de seguida representam a visão da empresa relativamente à aplicação web desenvolvida. De entre as respostas obtidas, os intervenientes trabalham como Gestor de Encomendas, Fornecedores e Clientes, Gestor de Amostras, Gestor de Armazém e Operário.

Quando questionados:

Os módulos que utiliza possui as principais funcionalidades pretendidas?, a resposta foi **Sim** por parte de todos os questionados.

O funcionamento da aplicação é fácil e intuitivo?, a resposta foi **Sim** por parte de todos os questionados.

Seguiram-se uma série de questões também de resposta fechada, com o objetivo de selecionar uma resposta num intervalo de "1" a "5" (sendo que 1 corresponde a "Discordo totalmente" e 5 corresponde a "Concordo totalmente").

Na generalidade a aplicação possui todas as funcionalidades pretendidas, foi obtida uma média de **4.0** nas respostas dadas.

A aplicação é de fácil utilização nos diferentes dispositivos (computadores, tablets e smartphones)?, foi obtida uma média de **4.0** nas respostas dadas.

De forma geral, como avalia o projeto desenvolvido?, foi obtida uma média de **5.0** nas respostas dadas.

As respostas seguintes seriam também dadas numa escala de "1" a "5" (sendo que 1 significava "Nada melhor" e 5 significava "Significativamente melhor")

Quão melhor torna a nova plataforma o processo de gestão? obteve uma resposta com valor médio de **4.3** nas respostas dadas.

Quão otimizado torna a nova plataforma o processo de gestão? obteve

uma média de 4.7 nas respostas dadas.

Como classifica a evolução do processo de gestão com a nova plataforma relativamente ao modelo anterior? obteve uma média de 4.7 nas respostas dadas.

Na resposta à pergunta **Qual ou quais as funcionalidades que acarretaram maiores melhorias para o processo de gestão?**, todas as funcionalidades foram identificadas com as exceção das funcionalidades do tipo de administrador. Destacam-se as funcionalidades de Encomendas, Armazém, Orçamentos e de Operário.

Nas respostas do tipo aberto à pergunta **Quais as melhorias mais importantes a destacar às funcionalidades atuais?**, foram destacadas a maior produtividade e maior rapidez como melhorias.

Quando questionados sobre **Quais as novas funcionalidades que gostariam de ver implementadas?**, no momento das respostas obtidas, nenhuma foi indicada.

O facto de esta ferramenta possuir todas as funcionalidades pretendidas demonstra a versatilidade das diferentes funcionalidades e a exigência imposta no momento do seu desenvolvimento para que cumprisse tudo aquilo que a empresa necessitava, nunca descurando a sua utilização e responsividade de modo a torná-la de utilização fácil e intuitiva.

A avaliação do projeto desenvolvido obteve por parte da empresa a classificação máxima, o que permite verificar a satisfação dos seus utilizadores e a garantia de que existe, de facto, uma diferença nos processos implementados na empresa diariamente e um impacto positivo na sua otimização.

Comparativamente com os processos anteriormente implementados na empresa, a evolução é notória e foi registada pela empresa também no questionário solicitado. Os processos de armazém, encomendas, orçamentos e de utilização por parte dos operários foram os mais destacados. Estes processos primam por ser bastante práticos e de rápida adaptação por parte dos intervenientes tendo melhorado especialmente a produtividade e a rapidez.

Nenhuma futura melhoria foi indicada por parte da empresa, o que é entendido como um indicador do cuidado no desenvolvimento da plataforma para que não fossem descuradas funcionalidades necessárias para o seu uso diário que se pretende robusto

e fiável.

Tendo por base as respostas obtidas por parte da empresa, é possível concluir a satisfação na utilização da nova aplicação web. Isto demonstra que conjuntamente com uma empresa, é possível desenvolver uma ferramenta que contém todas as funcionalidades necessárias, com claras vantagens na otimização dos processos.

7

Conclusão e Trabalho Futuro

O capítulo final visa apresentar as conclusões finais a reter do projeto da tese desenvolvido. É possível também confrontar os objetivos traçados inicialmente para o seu desenvolvimento e validar se foram atingidos.

Sendo a aplicação web desenvolvida uma ferramenta que a empresa pretende utilizar como auxílio no processo de gestão da mesma, existem funcionalidades úteis que foram tidas em conta como melhorias a implementar. Estas mesmas melhorias são identificadas no subcapítulo seguinte.

7.1 Conclusão

Fazendo uma retrospectiva dos objetivos traçados aquando o início do projeto, era realçado um objetivo principal que visava o desenvolvimento de uma aplicação, essencialmente para a gestão de *stocks* e encomendas. Esta base cimentada da plataforma foi desenvolvida e implementada, tendo em conta os princípios referidos pela empresa INDMEI e, de onde se destaca o facto de alguns dos *layouts* de informação que tiveram de ser seguidos e um fluxograma de produto à medida da empresa.

Em adição a estes princípios, também de destacar a necessidade de ter sempre presentes os princípios de uma boa programação, ou seja, código limpo e comentado, código com indentação, minimizar o uso de *spaghetti code* (ou seja, repetir o mesmo código em diversos pontos do projeto) definindo variáveis identificativas do que se pretende definir.

A este objetivo principal, associaram-se outros que foram tidos em conta e, foi verificada a sua importância para que o uso da plataforma pudesse estar o mais independente possível de outras ferramentas, como por exemplo folhas excel.

O sistema de gestão de permissões foi desenvolvido, ponderando a utilização de diferentes tipos de utilizadores, que teriam acesso a diferentes funcionalidades da plataforma. Existe portanto uma vasta gama de funcionalidades implementadas em que a sua atribuição varia de acordo com as funções do funcionário (utilizador).

Os módulos auxiliares para gestão de clientes e fornecedores, embora possam parecer dos mais simples da plataforma, são um acessório muito útil e que permite à empresa organizar uma lista completa com as principais informações de cada cliente e fornecedor. Esta lista encontra-se ainda preparada para filtragem e ordenação de forma a adequar-se às necessidades de pesquisa da empresa.

Com o objetivo traçado de exportação de dados para formato excel (ou xls) e em caso de necessidade, esta funcionalidade passou a estar implementada nas matérias-primas do armazém e ainda nas amostras produzidas pela empresa. Para além desta implementação, a exportação poderá ser realizada não só em formato excel, mas também nos formatos csv, pdf, ou imprimir diretamente a partir do ecrã respetivo da aplicação.

A apresentação de dados de forma visual era um outro objetivo inicialmente proposto e que também este foi cumprido. Os utilizadores da plataforma que possuam permissão que não seja convidado, poderão consultar os dados estatísticos e obter rapidamente informações sobre a evolução da empresa em termos de produção, tais como: quais os clientes que mais encomendas solicitam, quais são os clientes que geram maior receita e até mesmo quais as matérias-primas em que devem apostar para uma compra antecipada, uma vez que são aquelas mais utilizadas.

O envio e a gestão de emails através da plataforma tornou-se numa ferramenta importantíssima para os diversos tipos de utilizadores, embora em alguns casos, esta funcionalidade esteja a ser utilizada para funções específicas (ex. envio de orçamentos, solicitação de matéria-prima), trata-se de uma funcionalidade bastante interessante e uma ferramenta de comunicação interna. A consulta de emails através dos atalhos disponibilizados na plataforma também se traduz numa forma rápida e simples de otimizar a gestão de tempo e de tarefas.

Fazendo uma análise ao *feedback* obtido pela empresa INDMEI, considera-se que foi extremamente positivo. A empresa ficou satisfeita com a aplicação web desenvolvida e acredita que irá melhorar significativamente os seus processos, tratando-se de uma plataforma muito boa, prática e eficiente. Segundo indicado pela empresa, a sua interface é de fácil entendimento e, portanto, é bastante intuitiva.

Os processos de gestão de encomendas e armazém foram destacados como tendo uma melhoria muito acentuada relativamente aos métodos utilizados anteriormente, o que me deixa muito satisfeito.

Trata-se, contudo, de um projeto que possui melhorias possíveis de implementar. No entanto, foi feito um esforço para que todas as funcionalidades desenvolvidas pudessem ser utilizadas pela empresa e, portanto, trata-se de um projeto terminado e pronto a utilizar considerando-se todos os objetivos inicialmente propostos. Assim é possível concluir que os objetivos do projeto estágio foram cumpridos com êxito.

7.2 Trabalho Futuro

Uma vez que a aplicação web desenvolvida possui margem de evolução e, dado que à medida que a mesma é utilizada pela empresa novas oportunidades de melhoria vão surgindo, foram reunidas algumas melhorias que se poderiam implementar no futuro, de modo a tornar a aplicação mais completa.

7.2.1 Módulo de solicitação de matéria-prima

Este módulo provém de uma necessidade de segunda linha, que pretende agilizar os processos de solicitação de matéria-prima por parte de um utilizador com permissões de Gestor de Armazém. Uma vez que a aquisição de nova matéria-prima é realizada pelo utilizador com permissão de Gestor de Encomenda, existe atualmente a necessidade de este processo ser realizado manualmente, havendo a necessidade das duas entidades se encontrarem para proceder a este pedido.

O módulo a implementar como uma melhoria futura do software irá disponibilizar uma lista de pedidos a ser gerida pelo gestor de encomendas e na qual o gestor de armazém poderá criar novos pedidos.

Este módulo irá resultar num novo estado para os produtos em armazém, que será representado por uma cor específica.

Atualmente, existe a cor **vermelha** para indicar que o *stock* bruto se encontra abaixo do valor que seria de esperar. No momento em que este módulo seja implementado, pretende-se incluir uma outra cor, o **amarelo**, que irá sinalizar o pedido de novo stock já efetuado.

7.2.2 Módulo de notificações in real time

O módulo de notificações in real time tem o objetivo principal de indicar a qualquer utilizador, com qualquer tipo de permissão, que possui uma nova ação à espera de ser realizada.

Para cada permissão, diversas notificações distintas estarão disponíveis. Segue uma lista inicial de notificações que poderão vir a ser implementadas, divididas por permissão:

1. Gestor de Encomendas:
 - Alerta quando uma encomenda tem uma amostra associada;
 - Alerta quando é iniciada e terminada a produção de uma encomenda.
2. Gestor de Orçamentação:
 - Alerta quando uma encomenda passa para o estado "A aguardar orçamentação";
 - Alerta quando é iniciada e terminada a produção de uma encomenda.
3. Gestor de Amostra de Artigo
 - Alerta quando uma encomenda passa para o estado "A aguardar amostra de artigo";
 - Alerta quando é iniciada e terminada a produção de uma encomenda.
4. Gestor de Armazém
 - Alerta quando ocorrem movimentações de *stock* abaixo do *threshold* de aviso;
 - Alerta quando um pedido de nova matéria-prima é realizada.
5. Operário
 - Alerta quando uma encomenda passa para o estado "Em Produção";
 - Alerta quando faltarem 30 dias, 7 dias e 1 dia para a entrega da encomenda.

7.2.3 Módulo de duplicar amostra de artigo

Esta melhoria tem como objetivo otimizar o tempo de criação de uma amostra de artigo. Como foi possível verificar no submenu 5.4.2, a criação de uma amostra de artigo contempla o preenchimento de diversos dados e apesar de alguns deles já estarem facilitados pela escolha de opções através de *select*, ou então o preenchimento de alguns dados de uma linha no caso de apenas um campo ser preenchido, existe por vezes a necessidade de criar amostras de artigo idênticas, em que existe apenas uma ou outra diferença.

Neste contexto, uma melhoria a implementar será a possibilidade de duplicar uma amostra pré-existente. O fluxo desta funcionalidade seria o seguinte:

1. Clicar na opção "Duplicar Amostra";
2. Solicitar o novo nome da Amostra duplicada;
3. Ir para um layout que permitiria efetuar as alterações na amostra duplicada;
4. Guardar amostra duplicada.

Esta opção ficará disponível apenas para o gestor de amostra de artigo, para otimizar o seu ritmo de trabalho.

Por fim, para concluir, com a utilização diária da aplicação, e com a evolução natural dos processos das empresas, novas melhorias e até novas funcionalidades serão apresentadas e questionadas.

Bibliografia

- [1] inWork, “disponível em: <https://inwork.software/solucoes/picking/>,” acessido pela última vez a: 01.10.2018.
- [2] EasyForYou, “disponível em: <https://www.easyforyou.be/PT/gestion-de-stock.htm>,” acessido pela última vez a: 01.10.2018.
- [3] CentralGest, “disponível em: <https://www.centralgest.com/software/comercial/gestao-de-encomendas>,” acessido pela última vez a: 01.10.2018.
- [4] ARTSOFT, “disponível em: <https://www.artsoft.pt/encomendas-rateio>,” acessido pela última vez a: 01.10.2018.
- [5] alvo, “disponível em: <https://www.alvo.com/software-de-gestao/software-de-compras-e-stocks>,” acessido pela última vez a: 01.10.2018.
- [6] XSTOCK, “disponível em: <https://www.xstock.pt/>,” acessido pela última vez a: 01.10.2018.
- [7] PROTextil, “disponível em: <https://inforcavado.com/protexil/>,” acessido pela última vez a: 01.10.2018.
- [8] MACWIN, “disponível em: <https://www.macwin.pt/pt/1-erp-macwin/3-solucoes-gm-textil/1-gestao-de-confecao/>,” acessido pela última vez a: 01.10.2018.
- [9] php, “disponível em: <http://php.net/manual/en/intro-what-is.php>,” acessido pela última vez a: 01.10.2018.
- [10] composer, “disponível em: <https://getcomposer.org/doc/00-intro.md>,” acessido pela última vez a: 01.10.2018.
- [11] S. Reigns, “disponível em: <https://coderseye.com/best-php-frameworks-for-web-developers/>,” acessido pela última vez a: 01.10.2018.
- [12] XAMPP, “disponível em: <https://www.apachefriends.org/index.html>,” acessido pela última vez a: 01.10.2018.
- [13] MySQL, “disponível em: <https://dev.mysql.com/doc/refman/8.0/en/introduction.html>,” acessido pela última vez a: 01.10.2018.
- [14] HTML5, “disponível em: <http://tableless.github.io/iniciantes/manual/html/>,” acessido pela última vez a: 01.10.2018.

BIBLIOGRAFIA

- [15] JavaScript, “disponível em: <https://developer.mozilla.org/en-US/docs/Learn/JavaScript/>,”, acessido pela última vez a: 01.10.2018.
- [16] CSS3, “disponível em: <https://developer.mozilla.org/pt-PT/docs/Web/CSS/CSS3>,”, acessido pela última vez a: 01.10.2018.
- [17] BOOSTRAP, “disponível em: <http://getbootstrap.com/>,”, acessido pela última vez a: 01.10.2018.
- [18] JQuery, “<http://www.tutorialsteacher.com/jquery/what-is-jquery/>,”, acessido pela última vez a: 01.10.2018.
- [19] SCRUM, “<https://www.scrum.org/resources/what-is-scrum/>,”, acessido pela última vez a: 01.11.2018.