



**Arquitetura de Gestão de Inventário para
Perfis de Alumínio e Retornáveis**

UMinho | 2022 Daniel Ireneu Silva Oliveira



Universidade do Minho
Escola de Engenharia

Daniel Ireneu Silva Oliveira

**Arquitetura de Gestão de Inventário para
Perfis de Alumínio e Retornáveis**

junho de 2022



Universidade do Minho

Escola de Engenharia

Daniel Ireneu Silva Oliveira

A60302

Arquitetura de Gestão de Inventário para Perfis de Alumínio e Retornáveis

Dissertação de Mestrado

Mestrado Integrado em Engenharia e Gestão de Sistemas de
Informação

Trabalho efetuado sob a orientação de

Professor Doutor Miguel Abrunhosa Brito

DIREITOS DE AUTOR E CONDIÇÕES DE UTILIZAÇÃO DO TRABALHO POR TERCEIROS

Este é um trabalho académico que pode ser utilizado por terceiros desde que respeitadas as regras e boas práticas internacionalmente aceites, no que concerne aos direitos de autor e direitos conexos.

Caso o utilizador necessite de permissão para poder fazer um uso do trabalho em condições não previstas no licenciamento indicado, deverá contactar o autor, através do RepositóriUM da Universidade do Minho.

Licença concedida aos utilizadores deste trabalho



Atribuição

CC BY

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

AGRADECIMENTOS

Em primeiro lugar agradeço à minha esposa, que durante todo o meu percurso académico foi muito compreensiva nos momentos em que eu não podia estar presente e deu-me bastante apoio durante os momentos mais difíceis, em que por vezes pensei desistir.

Em segundo lugar, um agradecimento especial ao Professor Doutor Miguel Abrunhosa Brito pela orientação e apoio disponibilizado ao longo do semestre, que me permitiram aprender e concluir esta etapa.

Em terceiro lugar agradeço aos meus colegas de trabalho da Navarra – Extrusão de Alumínio, S.A, que estiveram envolvidos neste projeto e ajudaram a ultrapassar os obstáculos que apareceram ao longo do caminho.

Por fim, agradeço aos meus amigos e à família que me acompanharam durante este percurso, onde a entreaajuda e o companheirismo foram essenciais para escalar esta “montanha”.

DECLARAÇÃO DE INTEGRIDADE

Declaro ter atuado com integridade na elaboração do presente trabalho académico e confirmo que não recorri à prática de plágio nem a qualquer forma de utilização indevida ou falsificação de informações ou resultados em nenhuma das etapas conducente à sua elaboração.

Mais declaro que conheço e que respeitei o Código de Conduta Ética da Universidade do Minho.

RESUMO

As empresas necessitam cada vez mais de melhorar e automatizar os seus processos internos por forma a se tornarem mais eficientes num mercado cada vez mais competitivo. Assim, as tecnologias de informação surgem como um grande aliado neste processo de modernização e contribuição para o crescimento das empresas.

A Navarra – Extrusão de Alumínio S.A, tem realizado investimentos na modernização e na automatização dos seus processos produtivos, através de um maior investimento em novas tecnologias e em mão de obra qualificada na área das TI. Assim, esta dissertação desenvolveu-se segundo esta referência, ou seja, desenvolver uma solução que permitisse à empresa modernizar, agilizar e automatizar o processo de inventário de perfis de alumínio e retornáveis.

Numa primeira fase esta dissertação apresenta uma revisão de literatura onde foram identificados os trabalhos e a experiência de outros autores sobre este assunto e que tecnologias utilizaram.

De seguida, este trabalho passou por escolher as tecnologias a serem utilizadas no desenvolvimento de um protótipo e estudá-las.

Por fim, surgiu o desenvolvimento de uma arquitetura de sistema em camadas, capaz de garantir o desempenho e escalabilidade de uma solução de software feita à medida das necessidades da empresa. Como validação da arquitetura concebida, desenvolveu-se um software capaz de gerir o inventário de perfis de alumínio e retornáveis, e assim garantir à empresa um processo mais ágil, com menos erros e com uma maior monitorização. Apesar de ser um protótipo, a solução desenvolvida obteve um grande sucesso na sua primeira utilização, pelo que passou a ser uma solução integrada na produção e utilizada nos inventários da empresa.

Palavras-chave: Gestão de Inventário, Indústria, Alumínio, Software, Node JS, VUE JS

ABSTRACT

Companies increasingly need to improve and automate their internal processes, in order to become more efficient in an increasingly competitive market. Thus, information technologies emerge as a great ally in this process of modernization and contribution to the growth of companies.

Navarra – Aluminum Extrusion, S.A., has invested in the modernization and automation of its production processes, through greater investment in new technologies and qualified IT staff. As a result, this dissertation was developed according to this reference, that is, to develop a solution that would allow the company to modernize, streamline and automate the inventory process of aluminum and returnable profiles.

In a first phase, this dissertation presented a literature review to identify the work and learn about the experience of other authors on this subject and what technologies they used.

Then, this dissertation went through choosing the technologies to be used in the development of a prototype and studying them.

Finally, the development of a layered system architecture emerged, capable of guaranteeing the performance and scalability of a software solution tailored to the company needs. As a validation of the designed architecture, a software was developed capable of managing the inventory of aluminum and returnable profiles, thus guaranteeing the company a more agile process, with fewer errors and with greater monitoring. Despite being a prototype, the solution developed was very successful in its first use, so it became an integrated solution in production and used in the company's inventories.

Key Words: Inventory Management, Industry, Aluminum, Software, Node JS, VUE JS

CONTEÚDO

Resumo.....	iv
Abstract.....	v
Lista de Figuras.....	viii
Lista de Tabelas.....	x
Acrónimos.....	xii
1. Introdução.....	1
1.1. Motivação.....	1
1.2. Objetivos.....	2
1.3. Enquadramento empresarial.....	2
1.4. Estrutura do Documento.....	3
2. Abordagem Metodológica.....	5
2.1. Metodologia de Investigação.....	5
2.2. Metodologia de Desenvolvimento.....	7
3. Estado da arte.....	10
4. Tecnologias associadas.....	15
4.1. Service Oriented Architecture (SOA).....	15
4.2. Node JS.....	15
4.3. VUE JS.....	16
4.4. MySQL.....	16
4.5. Vuetify.....	16
5. Especificação do sistema.....	17
5.1. Processo atual.....	17
5.1.1. Inventário de perfis de alumínio em curso de fabrico (cestos) e embalado.....	17
5.1.2. Inventário de retornáveis.....	19
5.2. Arquitetura do Sistema.....	20

6.	Análise e Conceção	21
6.1.	Âmbito	21
6.2.	Requisitos funcionais.....	21
6.3.	Requisitos não funcionais	22
6.4.	Atores do projeto.....	23
6.5.	Casos de uso	25
6.5.1.	Casos de Uso de nível 0 – Sistema para gerir inventários	25
6.5.2.	Casos de Uso de nível 1 – Efetuar Inventário de Caixas e Cestos.....	26
6.5.3.	Casos de uso nível 1 – Efetuar inventário de retornáveis	38
6.6.	Estrutura Analítica do Projeto (EAP)	42
6.7.	Product Backlog.....	43
6.8.	Sprint Backlog.....	44
6.9.	Diagrama de Entidades e Relacionamentos (DER).....	47
6.10.	Ferramentas utilizadas	49
7.	Protótipo	50
7.1.	Backend	50
7.2.	Front-end	57
7.2.1.	Inventário de Cestos e de Caixas.....	59
7.2.2.	Inventário de retornáveis.....	63
7.2.3.	Abertura e fecho de inventários.....	64
7.3.	Data Storage	66
7.4.	Client.....	66
7.5.	Validação do protótipo	67
8.	Conclusões.....	71
	Bibliografia	73
	Anexos	76

LISTA DE FIGURAS

Figura 1 - Modelo de Investigação, DSRM (Peppers et al., 2007)	5
Figura 2 - Fluxo do processo Scrum Solo (Pagotto et al., 2016)	8
Figura 3 - Perfis de alumínio em cesto	17
Figura 4 - Perfis de alumínio em caixa.....	17
Figura 5 - Registo de Contagem de Perfis de Alumínio em Curso de Fabrico	18
Figura 6 - Registo de Contagem de Perfis de Alumínio Embalados	18
Figura 7 - Registo de Contagem de Retornáveis.....	19
Figura 8 - Arquitetura do Sistema.....	20
Figura 9 - Organograma do processo	24
Figura 10 - Diagrama de casos de uso de nível 0 – Sistema para gerir inventários.....	25
Figura 11 - Diagrama de casos de uso nível 1 – Efetuar inventário de caixas e cestos.....	26
Figura 12 - Diagrama de Sequência - Registrar contagem de caixa ou cesto.....	28
Figura 13 - Diagrama de Sequência - Consultar Contagem	29
Figura 14 - Diagrama de Sequência – Eliminar contagem de caixa ou cesto	30
Figura 15 - Diagrama de Sequência - Pedir Recontagem.....	31
Figura 16 - Diagrama de casos de uso nível 1 – Efetuar inventário de retornáveis.....	38
Figura 17 - Estrutura Analítica do Projeto (EAP)	42
Figura 18 - Diagrama de Entidades e Relacionamentos	48
Figura 19 - Regra Reserve Proxy no IIS.....	50
Figura 20 - Habilitar proxy no IIS.....	51
Figura 21 - Estrutura do Backend.....	52
Figura 22 - Mensagens de resultados de ações à BD.....	53
Figura 23 - Exemplo de código do tratamento de resultados	53
Figura 24 - Mensagem JSON "No records found"	54

Figura 25 - Script de conexão MySQL.....	54
Figura 26 - Routes localização de um cesto.....	55
Figura 27 - Função localização de um cesto.....	55
Figura 28 – Resultado JSON da consulta da localização do cesto	56
Figura 29 - Página de login	57
Figura 30 - Dashboad de situação do inventário	57
Figura 31 – Notificações.....	58
Figura 32 - Detalhes notificação.....	58
Figura 33 - Estado dos inventários	58
Figura 34 - Página de iniciar inventário de cestos.....	59
Figura 35 - Página de inventário de cesto.....	59
Figura 36 - Introdução dos dados da contagem.....	60
Figura 37 - Consulta da localização do cesto.....	60
Figura 38 - Lista de cestos inventariados.....	61
Figura 39 - Lista de cestos com diferenças de contagem.....	62
Figura 40 - Inventário de retornáveis	63
Figura 41 - Lista de inventário de retornáveis	64
Figura 42 - Abertura de inventários	65
Figura 43 - Fecho de inventários	65
Figura 44 - Dispositivos móveis android	66
Figura 45 - Leitores de códigos de barras.....	66
Figura 46 - Cesto.....	67
Figura 47 - Etiqueta de uma caixa.....	67
Figura 48 - Ata de Reunião	76
Figura 49 - Diagrama de Classes	77

LISTA DE TABELAS

Tabela 1 - U1.1 Registrar Contagem.....	27
Tabela 2 - U.1.2 Consultar Contagem	28
Tabela 3 - U.1.3 Eliminar Contagem	29
Tabela 4 - U.1.4 Consultar diferenças de contagem	30
Tabela 5 -U.1.5 Pedir recontagem.....	31
Tabela 6 - U.1.6 Consultar pedidos de recontagem	32
Tabela 7 - U.1.7 Adicionar resolução a diferenças de contagem	32
Tabela 8 - U.1.8 Consultar resolução de diferença de contagem.....	32
Tabela 9 - U.1.9 Exportar diferenças de contagens.....	33
Tabela 10 - U.1.10 Consultar localização do cesto	34
Tabela 11 - U.1.11 Consultar contagens manuais	34
Tabela 12 - U.1.12 Registrar diferença de contagem	35
Tabela 13 - U.1.13 Registrar recontagem efetuada.....	35
Tabela 14 - U.1.14 Mudar empresa/zona de contagem	35
Tabela 15 - U.1.15 Abrir inventário	36
Tabela 16 - U.1.16 Consultar inventários	36
Tabela 17 - U. 1.17 Fechar inventário	37
Tabela 18 - U.2.1 Registrar Contagem	38
Tabela 19 - U.2.2 Consultar Contagem	39
Tabela 20 - U.2.3 Eliminar Contagem	39
Tabela 21 - U.2.4 Exportar contagens	40
Tabela 22 - U.2.5 Mudar armazém de contagem	40
Tabela 23 - U.2.6 Abrir inventário	41
Tabela 24 - U.2.7 Consultar inventários	41
Tabela 25 - U.2.8 Fechar inventário	41

Tabela 26 - Product Backlog	43
Tabela 27 - Sprint Backlog.....	44
Tabela 28 - Ferramentas utilizadas em desenvolvimento	49
Tabela 29 - Ferramentas utilizadas em produção	49

ACRÓNIMOS

API – *Application Programming Interface*

DER – Diagrama de Entidades e Relacionamentos

DSRM - *Design Science Research Model*

ERP - Enterprise Resource Planning

IoT – *Internet of Things*

MRP - *Material Requirement Planning*

MVC - *Model View Controller*

NFC - *Near Field Communication*

PALOP - Países Africanos de Língua Oficial Portuguesa

RFID - *Radio-Frequency IDentification*

1. INTRODUÇÃO

O tema da dissertação nasceu da necessidade da empresa Navarra – Extrusão de Alumínio, S.A de criar um software para gerir e automatizar o processo de inventários de perfis de alumínio e retornáveis que até então era em grande parte realizado em papel, ou seja, pouco automatizado e muito moroso.

Apesar de ser um protótipo, o software desenvolvido foi capaz de gerir o inventário de perfis de alumínio e retornáveis e garantiu à empresa um processo mais ágil, com menos erros e com uma maior monitorização, pelo que passou a ser uma solução integrada na produção e utilizada nos inventários da empresa.

1.1. MOTIVAÇÃO

Uma gestão de inventários eficiente é fundamental para qualquer empresa, independentemente do modelo do negócio, uma vez que proporciona uma vantagem bastante competitiva relativamente à concorrência e à própria empresa, pois além de reduzir os custos operacionais, uma boa gestão dos inventários também possibilita a introdução de melhorias em termos de produtividade e estratégia, já que boas práticas de gestão de inventários aliadas a um sistema informático capaz e adequado permitem um rápido acesso à informação e análise de stocks.

Atualmente existem sistemas de gestão de inventários desenvolvidos por empresas de *software* precisamente para dar resposta à essa problemática, pelo que alguns destes *softwares* serviram de base para o desenvolvimento do *software* de gestão de inventários apresentado nesta dissertação que foi desenvolvido à medida das necessidades da Navarra – Extrusão de Alumínios por forma a permitir a integração com outras soluções de *software* já existentes na empresa.

No entanto, a gestão de inventários não é tão simples como pode parecer à primeira vista já que consiste numa série de processos com múltiplas funções referentes ao acompanhamento, manuseamento e gestão de materiais em stock, pelo que o desafio de criar de raiz uma ferramenta que trouxesse mais valias para a gestão de inventários foi a principal motivação para este projeto.

A conceção e desenvolvimento do protótipo surge pela necessidade de auxiliar a empresa no seu processo de inventário, nomeadamente no que diz respeito à redução de erros de inserção, redução do tempo do inventário e redução de custos. Assim, esta pesquisa consistiu na construção de um protótipo

feito à medida da empresa por forma a garantir que o mesmo se conecte com o ERP existente, para facilitar todo o processo e assegurar as melhorias que a empresa necessita.

1.2. OBJETIVOS

O presente projeto de dissertação tem como objetivo o desenvolvimento de uma arquitetura de gestão de inventários de alumínio e retornáveis, para auxiliar no processo de inventário.

Tendo em conta o desenvolvimento desta arquitetura, foram definidos os seguintes objetivos:

1. Realizar uma revisão do estado da arte
2. Estudar as tecnologias associadas
3. Conceber uma arquitetura
4. Validar a arquitetura através da conceção e desenvolvimento de um protótipo

Assim, através deste trabalho de desenvolvimento pretende-se melhorar e agilizar o processo de inventário atual, de modo a satisfazer todos os objetivos propostos.

O resultado final é em implementar um protótipo *in loco* e utilizá-lo em todos os inventários periódicos futuros da empresa.

1.3. ENQUADRAMENTO EMPRESARIAL

A empresa Navarra – Extrusão de Alumínio, S.A. pertence ao setor da metalurgia e metalomecânica, sendo as suas principais atividades a extrusão de alumínio e o tratamento de superfície de perfis de alumínio. Também assegura a montagem de perfis de rutura de ponte térmica e a maquinação e corte de precisão de perfis para as mais diversas aplicações. A empresa integra todos os processos produtivos numa única plataforma industrial de 35.000 m^2 e desenvolve perfis de alumínio para os vários setores da indústria, tanto para o mercado nacional como internacional com recurso a práticas de gestão sustentáveis que focam na inovação e melhoria contínua para ir de encontro não só aos desafios dos seus clientes, mas também acrescentar valor para acionistas, colaboradores e para a sociedade em geral.

A Navarra – Extrusão de Alumínio, S.A pertence ao Grupo Navarra que nasceu em 1977 na cidade de Braga e tem representação em várias delegações e redes comerciais por toda a Europa, sendo que a

sua representação também se estende para os PALOP (Angola e Moçambique), onde em Moçambique também conta com uma unidade produtiva de tratamentos de superfície de perfis de alumínio.

Atualmente, a Navarra conta com mais de 700 colaboradores e em 2020 teve um volume de negócios de 85 milhões de euros, é líder de mercado em Portugal onde se encontra no ranking das 500 maiores e melhores empresas portuguesas.

Internacionalmente também é uma empresa de referência onde desenvolve perfis de alumínio para os setores industriais mais exigentes, sendo que cerca de 50% da sua produção é para exportação.

1.4. ESTRUTURA DO DOCUMENTO

A organização deste documento está estruturada da seguinte forma:

O Capítulo 1 é composto por uma introdução ao tema em análise, motivação, objetivos, enquadramento empresarial e a estrutura do presente documento.

Já no Capítulo 2 são descritas as metodologias de investigação e desenvolvimento de software que vão ser utilizadas ao longo do projeto.

O Capítulo 3 é composto pela revisão da literatura do tema escolhido. A revisão da literatura é composta pela análise de vários artigos que abordam temas relacionados com arquiteturas e sistemas de gestão de stocks por forma a compreender melhor determinados conceitos e abordagens inerentes à gestão de inventários.

No Capítulo 4 estão descritas as tecnologias associadas ao projeto de desenvolvimento do protótipo que valida a arquitetura desenhada. Além da descrição, também são fornecidos argumentos técnicos do porquê da escolha dessas tecnologias.

No Capítulo 5 é explicado de que forma é realizado o processo atual de inventários de cestos, caixas e retornáveis. Neste capítulo também é apresentada a arquitetura concebida para dar resposta às problemáticas do processo atual.

No Capítulo 6, designado como análise e conceção, está representada toda a parte de planeamento do projeto, isto é, estão outputs tais como: o âmbito, requisitos funcionais e não funcionais, atores, casos de uso, *product backlog*, *sprint backlog*, etc.

No Capítulo 7, são explicadas as várias componentes do protótipo, ou seja, quais as abordagens e tecnologias utilizadas para o desenvolvimento do *backend* e quanto ao *frontend* são demonstradas e explicadas todas as suas funcionalidades. Ainda neste capítulo é demonstrado quais os equipamentos e suas características, utilizados pelos utilizadores nos vários inventários. Por fim, é abordado o ponto chamado “validação do protótipo”, que é composto sobretudo por testemunhos e indicadores do inventário antes e após o desenvolvimento da arquitetura e protótipo.

No Capítulo 8 são abordadas as conclusões da dissertação, sendo a principal, o fato da solução desenvolvida ter sido um grande sucesso na sua primeira utilização.

Apesar de ser um protótipo, o software desenvolvido foi capaz de gerir o inventário de perfis de alumínio e retornáveis e garantiu à empresa um processo mais ágil, com menos erros e com uma maior monitorização

2. ABORDAGEM METODOLÓGICA

2.1. METODOLOGIA DE INVESTIGAÇÃO

Esta dissertação assenta no desenvolvimento de uma arquitetura e solução de software de gestão de inventários, no entanto a mesma foi apoiada numa análise de artigos relativos às teorias de softwares de gestão de inventários para encontrar a melhor forma de operacionalizar a abordagem previamente estudada.

Foram utilizadas como termos de pesquisa, as seguintes expressões: “Inventory Management”, “Inventory Management Software”, “Inventory Control”, “Inventories”, “Warehouse Inventory Management System”, “Inventory Management Bar Code”.

Estas pesquisas foram efetuadas nos seguintes repositórios: Scopus, Google Scholar, entre outras fontes de informação relevantes.

Dado que esta pesquisa se enquadra na área das tecnologias e sistemas de informação, envolve um problema prático e a criação de novos conhecimentos científicos, nomeadamente uma invenção, a metodologia *Design Science Research* foi utilizada para conduzir o processo de investigação pois auxilia na análise das várias fases da investigação de forma objetiva, e questiona o avanço para a fase seguinte bem como a revisão das fases anteriores, motivado por um problema que afete o seguimento normal do projeto.

Na Figura 1 é apresentado o modelo de processos da metodologia Design Science Research que foi desenvolvida por (Peffer et al., 2007) e onde esta abordagem está dividida em 6 atividades.

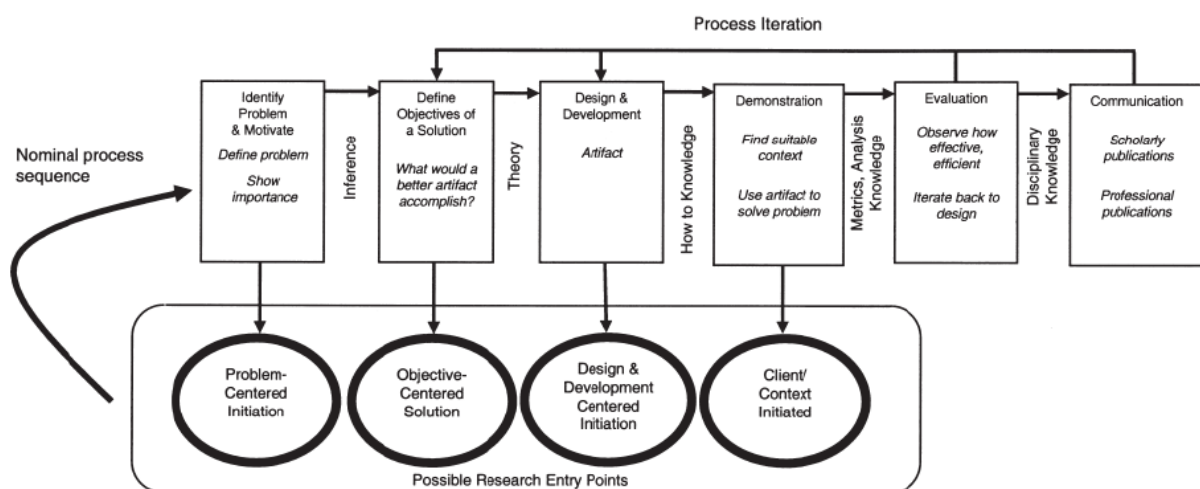


Figura 1 - Modelo de Investigação, DSRM (Peffer et al., 2007)

Atividade 1: Identificação do problema e motivação

Esta atividade define o problema específico e justifica a utilidade da solução. Como a definição do problema irá constar na lista de artefactos do trabalho desenvolvido, é normal atomizar a conceção do problema conceptualmente para que a solução possa capturar a sua complexidade. Justificar o valor da solução apresenta duas grandes vantagens: motiva o investigador a ir em busca da solução e a aceitar os resultados e ajuda a entender o raciocínio associado à compreensão do problema por parte do investigador.

Atividade 2: Definir os objetivos para a solução

Esta atividade tem como objetivo definir os objetivos de uma solução a partir da definição do problema e do conhecimento do que é possível e viável. Os objetivos podem ser quantitativos, onde ter uma solução desejável seria melhor do que a situação atual, ou qualitativos, como uma descrição de como se espera que um novo artefato suporte soluções para problemas não abordados até ao momento. Os objetivos devem ser alcançados a partir da especificação do problema. Os recursos necessários para isso incluem o conhecimento dos problemas e das soluções atuais.

Atividade 3: Conceção e desenvolvimento

Esta atividade consiste na criação de um artefacto que pode ser um produto, modelos, métodos, instanciações ou novas propriedades técnicas, sociais e/ou recursos informacionais.

Esta atividade inclui determinar a funcionalidade desejada da solução, arquitetura e criação da solução real. Os recursos necessários para passar dos objetivos ao desenvolvimento, necessitam de conhecimentos práticos e teóricos para a criação da solução.

Atividade 4: Demonstração

Esta atividade serve para demonstrar o uso do artefacto para resolver uma ou mais instâncias do mesmo problema. Contudo, pode ser necessário uma experiência, uma simulação, um estudo de caso, uma prova de conceito ou outra atividade apropriada. Os recursos necessários para a demonstração incluem conhecimento efetivo de como usar o artefato para resolver o problema.

Atividade 5: Avaliação

Esta atividade pretende medir se o artefacto suporta a solução do problema. Nesta atividade comparamos o objetivo da solução com os resultados reais observados do uso do artefacto na demonstração. Requer conhecimento de métricas relevantes e técnicas de análise.

Atividade 6: Comunicação

Deve ser comunicado o problema e a sua importância, o artefacto, a sua utilidade e novidade, o rigor da concepção, a sua eficácia perante os investigadores e para a audiência interessada. Esta atividade de comunicação requer conhecimento da cultura disciplinar.

(Peffer et al., 2007)

2.2. METODOLOGIA DE DESENVOLVIMENTO

Sabe-se que o processo de desenvolvimento de software sem o recurso a uma metodologia de desenvolvimento é a receita certa para o fracasso, nomeadamente no que diz respeito ao planeamento ou qualidade do artefacto desenvolvido.

Embora a empresa onde se realizou este projeto tenha um departamento de informática, para dar resposta aos pedidos de desenvolvimento de software, não existem equipas de desenvolvimento montadas. Cada elemento vai dando resposta às problemáticas de diferentes setores da fábrica.

A escolha da metodologia Scrum Solo surge como uma escolha natural e que se encaixa perfeitamente na realidade deste projeto, uma vez que esta metodologia surge como uma otimização do processo Scrum, mas voltada para o desenvolvimento individual de software (Pagotto et al., 2016).

Segundo os autores Pagotto et al. (2016), o Scrum Solo possui características muito semelhantes ao Scrum. A produção do *product backlog* e do *sprint backlog*, ocorrem de forma idêntica em ambos os *frameworks*, mas no Scrum Solo é proposto que as durações do *sprint* tenham apenas uma semana e não existem reuniões diárias no final de cada *sprint*. É importante evidenciar também o fato do Scrum Solo integrar as boas práticas do framework Scrum com as prerrogativas delineadas pelo *Personal Software Process* (PSP) que garante qualidade e agilidade na produção do software.



Figura 2 - Fluxo do processo Scrum Solo (Pagotto et al., 2016)

A seguir, será descrita a especificação do Scrum Solo e a sua relação com as atividades, os atores e os artefactos:

A. Atividades do processo

- **Requeriment.** Esta tarefa tem como objetivo definir o âmbito do produto, caracterizar o cliente e definir o *product backlog*.
- **Sprint.** tem como objetivo desenvolver o conjunto de itens selecionados a partir do *product backlog* com a duração máxima de uma semana.
- **Deployment.** esta atividade tem como objetivo disponibilizar o software para o cliente. A planta de desenvolvimento é dada como entrada para esta atividade e os artefactos gerados são o software e as atas de validação.
- **Management.** o objetivo desta atividade é planejar, monitorizar e controlar o desenvolvimento do *software*. O *product backlog* é utilizado como entrada, e os artefactos gerados nessa atividade são a estrutura analítica do projeto (EAP) cronograma, folha de custo e folha de controlo.

B. Atores do processo

- **Product owner:** proprietário do produto.
- **Programador individual:** responsável por executar o processo e construir o produto.
- **Orientador:** caracterizado como um consultor que conhece a fundo o processo.
- **Grupo de Validação:** possíveis utilizadores do produto desenvolvido.

C. Artefactos do processo

- **Âmbito:** descreve os principais pontos do software, o perfil do cliente e os pontos do *product backlog* (requisitos funcionais).

- **Protótipo de Software:** representa todos os mockups feitos ao software e as janelas para acesso e manipulação dos dados do software.
- **Product backlog:** caracteriza uma lista de funcionalidades que devem ser implementadas no software.
- **Repositório do processo:** representa o serviço de armazenamento de arquivos na nuvem, que armazena todos os artefactos gerados durante a execução do processo.
- **Sprint backlog:** armazena o conjunto de funcionalidades que devem ser implementadas durante aquele sprint.
- **Produto ou parte dele em funcionamento:** versão do produto que possibilita ao cliente obter um retorno sobre o investimento realizado aquando da compra do software.
- **Ata:** utilizada para registar a validação da implementação de uma funcionalidade. É utilizada no *sprint* e na entrega.
- **Planta de desenvolvimento:** tem como objetivo juntar os artefactos que foram utilizados na especificação das funcionalidades.
- **Estrutura Analítica do Projeto (EAP):** organograma que representa o âmbito do projeto.
- **Cronograma:** tem o objetivo de organizar sequencialmente as tarefas de trabalho dentro de um determinado espaço de tempo.
- **Folha de custos:** tem a função de mapear o custo efetivo gerado durante a execução do projeto.

Segundo o autor Pagotto et al. (2016), o Scrum Solo foi criado em 2012, sendo que vários programadores solo têm utilizado este processo com bastante êxito.

Esta metodologia foi aplicada pelos autores Brito et al. (2020), num projeto extenso com uma forte componente de integração e em outro com uma forte componente de processo de desmaterialização. Os autores concluíram a metodologia Scrum Solo foi uma boa adaptação das metodologias em que a mesma se baseou. No entanto, o Scrum Solo não está isento de adaptações aos casos específicos de cada indivíduo e de cada projeto.

3. ESTADO DA ARTE

Os inventários são bens que as empresas detêm com o único objetivo de os vender, direta ou indiretamente. Assim, as empresas que pretendem ter as suas contas corretas necessitam de uma boa gestão de inventários, sendo que os recursos a ferramentas informáticas permitem ajudar as empresas a adquirir essa capacidade de gestão dos inventários que como consequência melhoram a tomada de decisão. Na Europa, o investimento em soluções de controlo e suporte já representam uma grande percentagem sobre os lucros das empresas (Akkermans et al., 2003).

Um sistema de gestão de stocks de armazém constitui uma abordagem necessária para todos os armazéns, uma vez que um sistema de armazém informatizado proporciona menos esforço, resultados mais eficientes e estáveis quando comparado com um controlo manual do armazém. A necessidade de automatizar os armazéns tem origem no facto de que os sistemas de movimentação manual podem dar origem a erros humanos, o que pode afetar o seu funcionamento normal (Tejesh & Neeraja, 2018).

Segundo Trunk Christopher (1994), nos dias de hoje o mundo das empresas de armazenamento e distribuição, dependem fortemente de sistemas de códigos de barras para manter o controlo dos stocks e garantir o fluxo eficiente dentro e fora do armazém. Tejesh & Neeraja (2018), também concordam que os códigos de barras representam um grande avanço em relação às etiquetas de texto, pois eliminam a inserção de dados de forma manual.

Tejesh & Neeraja (2018), defendem que embora existam muitas tecnologias de comunicação sem fio, a identificação por radiofrequência (RFID) é o mais adequado para o sistema de gestão de stocks em armazém, assim como Wartha N & Londhe Vaishali (2015), que também defendem que a tecnologia de RFID é mais adequada para a gestão de stocks em armazém, dada esta ser uma tecnologia mais segura, com maior capacidade de armazenamento de dados e mais resistente.

Com o objetivo de obter mais produtividade, um melhor atendimento ao cliente e uma maior precisão de gestão de stocks e custos, Manthou & Vlachopoulou (2001), utilizaram a tecnologia de códigos de barras para apoiar as cadeias de distribuição. Os autores nesta pesquisa estudaram a incorporação da tecnologia de códigos de barras nos sistemas de stocks e desenvolveram um modelo para o desenvolvimento e implementação de um sistema de códigos de barras nos sistemas de gestão de stocks. Os autores aplicaram o modelo conceptual desenvolvido num estudo de caso, com o objetivo de automatizar todas as transações operacionais. Conclui-se que este sistema ofereceu à empresa uma maior eficiência no atendimento dos clientes em comparação com a concorrência, e que o modelo

apresentado pode ser implementado de uma forma estrutural nos sistemas de gestão de inventários existentes, conferindo um papel importante no controlo de stocks da empresa.

Tejesh & Neeraja (2018), desenvolveram um sistema de gestão de stocks que utiliza *Internet of Things* (IoT) e possui uma estrutura de código aberto. Este sistema tem como objetivo localizar e posteriormente guardar qualquer produto e todos os seus detalhes, como por exemplo em que zona de stock se localizam, de forma automática o que poupa bastante trabalho e esforço ao fiel de armazém que normalmente tem que procurar manualmente por todos os produtos nas zonas de stock. O sistema de gestão de stocks foi construído sobre a arquitetura de IoT para conseguir ler as informações das etiquetas RFID e rapidamente fornecer as localizações e detalhes dos produtos.

Após a implementação da metodologia que incorpora a tecnologia de RFID, os autores verificaram que esta não está apenas limitada para uso laboratorial, mas também funciona de forma eficiente em situações do mundo real. O custo da aplicação deste sistema é reduzido em comparação com outras soluções existentes no mercado. Por fim, os autores concluíram que a implementação da interface gráfica foi uma grande mais-valia para os utilizadores conseguirem localizar mais facilmente o produto no armazém sem grandes dificuldades e esforços.

Cheng et al. (2015) realizaram um estudo baseado em tecnologia *Near Field Communication* (NFC) para desenvolver um sistema de gestão de stocks que pode ser implementado em smartphones, pelo que combina dispositivos Android e a tecnologia NFC. Este estudo projeta um módulo de função que utiliza um sensor NFC para substituir a verificação tradicional baseada em papel.

Previamente as tags NFC são carregadas com os dados para realizar a conferência do inventário, sendo que para realizar o inventário, basta aproximar o smartphone da tag e clicar em OK se os dados estiverem corretos, ou realizar a alteração na APP se a informação não conferir. Este estudo mostrou que através da implementação de um sistema de gestão de inventários capaz de ler uma tag NFC, as empresas podem economizar mais tempo e custos com de mão de obra, em comparação com a verificação tradicional baseada em papel ou RFID.

Chin & Chang (2009), realizaram um estudo com o objetivo de desenvolver um sistema capaz de ajudar as empresas a conectar a gestão de stocks, gestão de transporte e seleção de fornecedores, de modo que as mesmas consigam uma melhor coordenação de toda a sua cadeia de abastecimento. Esse estudo passou também pela realização de uma revisão de literatura onde os investigadores verificaram que outros investigadores só se focavam em campos específicos, tais como, sistema de stocks, gestão de

transporte, seleção de fornecedores, mas dificilmente discutiam a questão da tomada de decisão relacionada ao stock de toda a organização.

Assim, os autores propuseram a criação de uma *framework Integrated Inventory Management (IMS)* e do protótipo *Integrated Inventory Management Software (IIMS)* para ajudar as empresas a conectar os vários processos da cadeia de valor, ou seja, o processo de gestão de fornecedores, processo de gestão de ponto de venda, processo de compra, processo de gestão de produto, processo de monitoramento e processo de alocação do ponto de vista da coordenação de toda a empresa.

Chin & Chang (2009) concluíram que através da arquitetura desenhada e do protótipo instalado na empresa, foi possível melhorar problemas de tomada de decisões de gestão de stocks através da flexibilização e automatização de processos.

Atieh et al. (2016) realizaram uma melhoria do processo de gestão de inventários através da automatização da gestão de armazém. O principal objetivo dessa automatização foi conseguir controlar a movimentação dos produtos do armazém, juntamente com o benefício de um manuseio mais rápido e seguro. Os autores concluíram que um sistema automatizado de gestão de armazéns é essencial para as empresas e um substituto necessário a um sistema de gestão manual. O sistema desenvolvido foi capaz de controlar as movimentações e armazenamento dos produtos, juntamente com os benefícios de maior segurança e rapidez de movimentação dos produtos.

Gutiérrez & Jaramillo (2009), realizaram um trabalho de revisão sistemática de todas as ofertas de software de gestão de inventários disponíveis na Colômbia, com o objetivo de proporcionarem um apoio na tomada de decisão para as empresas que enfrentam essa decisão, poderem compreender quais as características dos softwares disponíveis.

A pesquisa concluiu que uma percentagem muito reduzida das aplicações avaliadas permitiam de fato apoiar decisões na gestão de stocks. A principal fragilidade encontrada foi falta de métodos quantitativos, quer no cálculo de custos de pedidos, quer na manutenção de stocks e de vendas perdidas, bem como no registo dos níveis de serviço dos diferentes tipos de stock.

Setemen et al. (2020), realizaram um trabalho que passou por desenvolver uma arquitetura de um sistema de inventários que utilizou dispositivos móveis, códigos de barras 2D e geotagging (GPS e mapa digital). Contudo, antes deste desenvolvimento os autores realizaram pesquisas para encontrar as melhores teorias de levantamento de requisitos e desenvolvimento e implementação de software, e

concluíram que os códigos de barras 2D e o geotagging proporcionam uma experiência interativa para o utilizador que usa um smartphone/tablet e permitem obter informações precisas e fiáveis.

Setemen et al. (2020) também concluíram que estas tecnologias necessitam de ser mais trabalhadas, pois existem um número vasto de oportunidades para melhorá-las.

Arsan et al. (2013), desenvolveram uma arquitetura de um software para um sistema de gestão de stocks. Após o levantamento dos requisitos, decidiram utilizar a linguagem de programação PHP e AJAX para codificar o software e colocar esse código numa estrutura *Model View Controller* (MVC).

Os autores utilizaram a *framework* de desenvolvimento *Code Igniter* para agilizar o desenvolvimento e diminuir o número de linhas de código, o APACHE foi a escolha para o servidor Web e o Mysql para a base de dados. Neste sistema, os autores também utilizaram os códigos de barras para identificar os produtos e assim evitar erros de digitação. Denominaram como *Inventory Management System* (IMS) a arquitetura desenhada que foi composta por 4 módulos: Definições/Configuração, Gestão de Stock, Gestão de Serviços e Gestão de Manutenção. A arquitetura foi dividida em 3 camadas, sendo elas o Frontend, Servidor e Dados.

Arsan et al. (2013) concluíram que o software desenvolvido ficou de acordo com a arquitetura e os requisitos levantados, pelo que foi possível obter uma solução escalável e flexível e que possibilita responder quer a um crescimento rápido do negócio, quer ao aparecimento de novos requisitos.

Mathaba et al. (2017), realizaram uma investigação onde recorreram às sinergias entre a Web 2.0 e as IoTs para aprimorar o controlo de stocks. Assim, desenharam uma arquitetura de software que combinou a tecnologia RFID com a Web 2.0, para o desenvolvimento de um protótipo de gestão de stocks capaz de detetar produtos extraviados, baixos níveis de stocks e enviar notificações do estado do inventário via Twitter para os gestores do stock.

Este estudo propõe um sistema de monitorização de gestão de stocks, de modo que as empresas tenham uma maior visibilidade sobre as informações do inventário e consigam melhorar o processo de tomada de decisão. O estudo também propõe um modelo para detetar os produtos extraviados ou os vencidos que ainda se encontram nas prateleiras das lojas e que têm uma reposição tardia, o que muitas vezes pode levar à perda de vendas. O Twitter possui o papel importante de notificar os gestores de stock em alterações no stock que precisam de uma atenção imediata.

Mathaba et al. (2017), concluíram que o protótipo forneceu informações oportunas para um gestor de stocks através da rede social *Twitter*. As tarefas de notificar o gestor de stocks sobre os extravios e os baixos níveis de stock nas prateleiras, foram cumpridas. As notificações foram sempre recebidas pelos gestores de inventário em tempo real. O estudo demonstrou que as empresas podem beneficiar do uso das IoTs com as ferramentas da Web 2.0 para melhorar a gestão de stocks.

4. TECNOLOGIAS ASSOCIADAS

4.1. SERVICE ORIENTED ARCHITECTURE (SOA)

O desenvolvimento do protótipo para comprovar a arquitetura desenhada foi criado segundo a abordagem *Service Oriented Architecture* (SOA), de forma a promover a reutilização do código e rapidez no desenvolvimento, e garantir com eficiência que os requisitos não funcionais seriam cumpridos.

A arquitetura orientada a serviços (SOA) é um paradigma para organizar e englobar funcionalidades como *web services*, tornando-os disponíveis numa rede para poderem ser invocados e assim serem soluções para problemas de negócios (Laskey & Laskey, 2009).

Segundo Sprott & Wilkes (2004), a adoção de uma arquitetura orientada aos serviços (SOA) é essencial para obter agilidade nos negócios e a flexibilidade que os *Web Services* prometem. Contudo, a adoção de uma arquitetura SOA também exige a criação de um ambiente orientado aos serviços.

4.2. NODE JS

O *Node JS* foi selecionado como opção para o desenvolvimento do *backend* do protótipo. Esta tecnologia foi a escolhida, uma vez que proporciona uma excelente integração de desenvolvimento cliente-servidor e auxilia na reutilização de código no que diz respeito à aplicações web, e é uma ferramenta perfeita para desenvolver de forma rápida aplicações de rede e escaláveis (Chaniotis et al., 2015). Em adição, esta tecnologia tem por base a linguagem JavaScript que é uma das tecnologias *Client-side* mais utilizadas, e que ao mesmo tempo tem uma grande diversidade de plugins prontos a serem utilizados, que são desenvolvidos e atualizados por uma forte comunidade de programadores.

O *Node JS* é uma nova tecnologia *JavaScript* que permite criar facilmente aplicações de uma forma rápida e escalável. O *Node Js* utiliza um modelo de *Input/Output* e é orientado a eventos, o que o torna leve e eficiente, e perfeito para aplicações *real time* com um uso intenso de dados que são executados em dispositivos distribuídos.

Lei et al. (2014) realizaram um teste de *benchmark* entre as tecnologias Node.js, PHP e Python-Web, e concluíram que o Node JS conseguiu lidar com mais solicitações que o PHP ou Python-Web, num determinado espaço de tempo. Também concluíram o Node.js é bastante leve e eficiente, ideal para sites com bastantes requisições de *Input/Output*.

4.3. VUE JS

O Vue JS é o *framework* mais recente em comparação com outros *frameworks* mais conhecidos e utilizados no mercado, e é frequentemente descrito como uma estrutura progressiva que pode ser utilizada para construir interfaces para a web. O Vue JS foi lançado em 2014 por Evan You e a versão 1.0.0 foi lançada em outubro de 2015. Embora não seja estritamente associado ao padrão *Model-View-Viewmodel* (MVVM), os princípios de design do Vue JS foram parcialmente inspirados por ele. (Studiengang Bachelor et al., 2018).

O Vue JS foi a tecnologia escolhida para o *frontend* também por ser esta uma tecnologia mais permissiva, e como consequência, possuir uma curva de aprendizagem mais pequena quando comparado com a do Angular JS ou o React.

4.4. MYSQL

O MySQL foi desenvolvido na Suécia em 1995 e atualmente integra a Oracle Corporation (Letkowski, 2002), é um sistema de gestão de base de dados (SGBD) relacional de código aberto e um dos mais conhecidos e amplamente utilizado devido à sua simplicidade e notável desempenho. Embora não possua alguns dos recursos avançados disponíveis em outros SGBDs do mercado, é uma opção atraente devido à sua facilidade de utilização e tempo de *setup* reduzido (Santillán et al., 2007).

A versão que foi utilizada durante o desenvolvimento do protótipo foi a versão MySQL Community Server 5.5 e o editor de base de dados foi o MySQL Workbench.

4.5. VUETIFY

O *Vuetify* é uma *framework* CSS para desenvolver aplicações web de uma forma mais rápida e otimizada. Esta *framework* foi criada em 2016 e está no lote das *frameworks JavaScript*, e tem como objetivo fornecer vários componentes de estilo para aplicações *Vue JS* e baseia-se na *framework Material Design* da Google.

A escolha desta *framework* para desenvolver os componentes CSS da aplicação, foi a mais natural por estar bastante otimizada para aplicações desenvolvidas em *Vue JS* e também por seguir padrões de estilo bastantes testados e otimizados como são os do *Material Design* da Google.

5. ESPECIFICAÇÃO DO SISTEMA

5.1. PROCESSO ATUAL

Anualmente a Navarra – Extrusão de Alumínio S.A. organiza o inventário de perfis de alumínio em cursos de fabrico (cestos) (Figura 3), embalados (caixas) (Figura 4) e de retornáveis (cestos onde é guardado o material). No dia do inventário, é realizada uma reunião inicial onde são distribuídos os elementos necessários para o inventário (blocos de contagem, etiquetas de cor e canetas) e formadas equipas de duas pessoas, onde para cada uma das equipas é nomeado um coordenador responsável.



Figura 3 - Perfis de alumínio em cesto



Figura 4 - Perfis de alumínio em caixa

5.1.1. INVENTÁRIO DE PERFIS DE ALUMÍNIO EM CURSO DE FABRICO (CESTOS) E EMBALADO

A contagem dos perfis de alumínio em curso é sempre realizada cesto a cesto, referência a referência e por duas pessoas, onde cada uma faz a contagem num dos lados do cesto e após a contagem deverão confrontar as quantidades contadas, caso existam diferenças deverão recontar de imediato o cesto e após chegarem às mesmas quantidades é que deverão registar na folha. Além da quantidade, o inventariante também tem que identificar o local da contagem, o número do cesto, a referência do perfil do alumínio, se o perfil levou algum tratamento (lacado, anodizado, etc) ou se está em bruto e medir

com fita métrica o comprimento do mesmo. Estas contagens e identificações são registadas manualmente numa folha de contagem tal como podemos verificar na Figura 5.

Existe sempre uma folha de contagem por referência de perfil de alumínio no cesto, e após a contagem o inventariante deve colocar uma etiqueta verde para sinalizar que o cesto foi contado, por forma a evitar que outra equipa conte novamente o cesto.

N.º 0427

Secção: Amodificação

N.º de Classificação/Liga: 256

Perfil/Colada: N7659 030

Medida: 6,5 Quantidade: 98

Tratamento Atual: Bruto

Contado por: _____ Conferido por: _____

Cesto - 6m 12m
 U - Azul Cinza

Figura 5 - Registo de Contagem de Perfis de Alumínio em Curso de Fabrico

À semelhança do inventário de perfis de alumínio em curso de fabrico, para o inventário de perfis de alumínio embalados é realizado o mesmo levantamento de informações, apenas com a particularidade de que a contagem é realizada por número de encomenda, ou seja, no cabeçalho da folha de contagem regista-se o número de encomenda e depois todas as caixas que existem para esse número de encomenda, tal como se pode verificar na Figura 6.

Pedidos Embalados Pedido: _____ **876**

Cód. Caixa:	Perfil:	Medida:	Tratamento:	Quantidade:
<u>190430990</u>	<u>N745107</u>	<u>2,6</u>	<u>Bruto</u>	<u>12</u>
<u>190430019</u>	<u>N745109</u>	<u>3,1</u>	<u>Bruto</u>	<u>20</u>
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____

Contado por: _____ Conferido por: _____

Figura 6 - Registo de Contagem de Perfis de Alumínio Embalados

Periodicamente o coordenador de equipa desloca-se junto das suas equipas para recolher as folhas de contagem, para posteriormente fazer chegar ao centro de operações, que está montado para recolher as informações existentes nas folhas de contagem, lançar no ERP e analisar as diferenças. Caso exista alguma diferença quantitativa ou qualitativa, é colada nessa folha de contagem uma etiqueta amarela que sinaliza a necessidade de realizar uma recontagem, pelo que, por forma a garantir imparcialidade, é solicitado a outra equipa a realização da recontagem. Depois desta recontagem, é realizada novamente uma análise de diferenças, caso não existam diferenças então o inventário dessa caixa ou cesto é dada como concluída, e caso continuem a existir diferenças é colada uma etiqueta vermelha na folha de contagem que sinaliza que será necessário fazer um acerto de stock no ERP e reimpressa uma nova folha de classificação ou etiqueta para colocar no cesto ou caixa.

5.1.2. INVENTÁRIO DE RETORNÁVEIS

O inventário de retornáveis realiza-se com o objetivo de saber quantos cestos existem na fábrica e nos clientes. Este inventário faz-se aquando do inventário de perfis de alumínio em curso, ou seja, na mesma folha de contagem dos cestos o inventariante assinala com uma cruz o tipo de cesto que estão a inventariar (Figura 7). Também são enviadas equipas aos clientes com o objetivo de compreender se os mesmos pretendem receber o material embalado ou se preferem o receber em cestos, já que para alguns clientes não é essencial que o material chegue bem-acondicionado. Assim, para esses clientes os cestos funcionam como uma espécie de “tara” e para controlo da Navarra é essencial saber quantos dos seus cestos se encontram nesses clientes. Após todo o levantamento, o colaborador responsável faz a recolha dessa informação e lança num ficheiro Excel onde fica o registo da quantidade de cestos por tipo e a localização dos mesmos.

N.º 0427

Secção: Amodigação

N.º de Classificação/Liga: 256

Perfil/Colada: U/659 030

Medida: 6,5 Quantidade: 98

Tratamento Atual: Prato

Contado por: _____ Conferido por: _____

Cesto - 6m	<input checked="" type="checkbox"/>	12m	<input type="checkbox"/>
U - Azul	<input type="checkbox"/>	Cinza	<input type="checkbox"/>

Figura 7 - Registo de Contagem de Retornáveis

5.2. ARQUITETURA DO SISTEMA

De forma a sustentar uma solução que responda às necessidades e garanta o desempenho e escalabilidade para o protótipo desenvolvido, foi elaborada uma arquitetura de sistema em camadas, conforme se pode verificar na Figura 8.

A camada *Data Storage* que é responsável pela sistematização e permanência dos dados e comunica com o *Back-end* através de um driver de base de dados.

A camada *Back-end* é responsável por garantir as regras de negócio e inserir os dados na camada base de dados. Também é responsável por expor em forma de serviços funcionalidades que permitem à interface gráfica apresentar os dados solicitados, pelo que é responsável por recolher os dados inseridos na interface gráfica e por fim, também é responsável pela obtenção dos dados do ERP da empresa.

A camada de *Front-end*, é responsável pela interação com o utilizador e a camada de *back-end*. Esta camada recebe os *inputs* dados pelos utilizadores através dos seus *devices* e possui a capacidade de se adaptar a dispositivos móveis, sendo que estes através de leitores de códigos de barras realizam a leitura de dados que enviam para a interface gráfica.

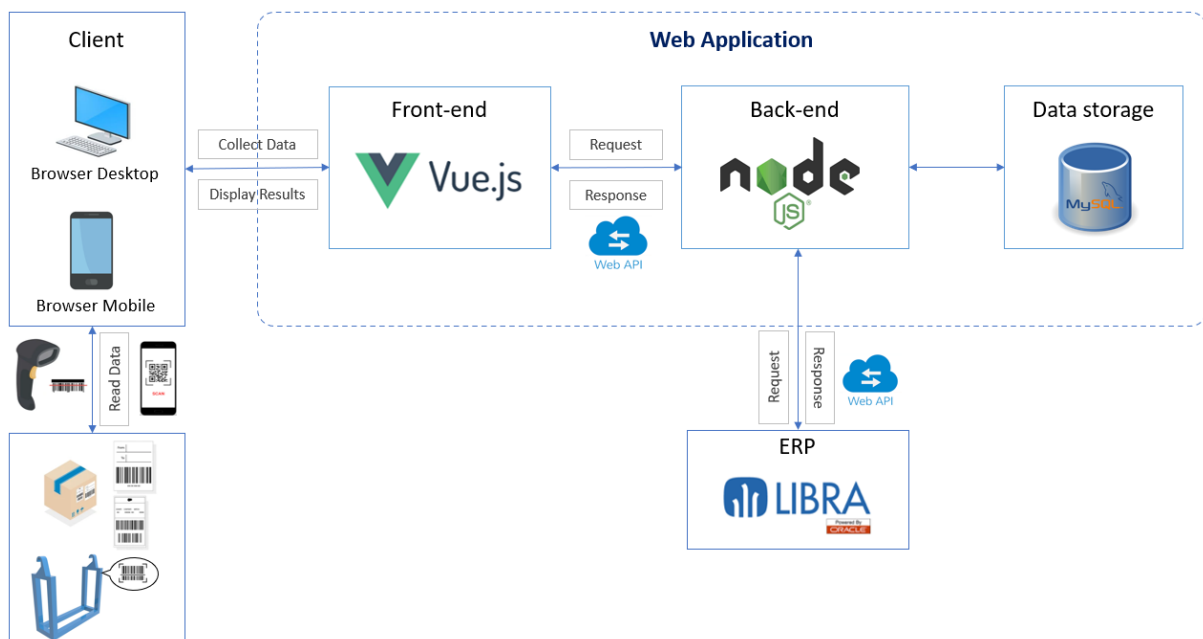


Figura 8 - Arquitetura do Sistema

6. ANÁLISE E CONCEÇÃO

Um desenvolvimento de engenharia de *software* só conseguirá ter sucesso se os objetivos do projeto forem bem definidos e se posteriormente for realizado um acompanhamento rigoroso do cumprimento dos mesmos ao longo do projeto. Assim, a análise e a definição dos requisitos têm um papel crucial para que o projeto não apresente falhas.

Para obter a lista de requisitos para a solução, foram realizadas entrevistas com os vários *stakeholders* ao longo do projeto, onde se obteve uma lista inicial de requisitos e que se foi refinando ao longo da vida do projeto.

Tendo em conta as reuniões realizadas com os *stakeholders*, foram definidos os seguintes requisitos funcionais e não funcionais.

6.1. ÂMBITO

Este projeto tem como âmbito o desenvolvimento de um protótipo capaz de registar a contagem de inventário de caixas, cestos e retornáveis. Também deverá ser possível consultar todas essas contagens e registar automaticamente diferenças de contagem, caso os dados introduzidos pelo utilizador sejam diferentes dos que estão registados no ERP da empresa e notificar automaticamente o gestor de inventário da diferença de contagem. O gestor de inventário também deverá ser capaz de poder consultar e tratar essas diferenças de contagem. O protótipo também deverá ser capaz de enviar e consultar notificações.

6.2. REQUISITOS FUNCIONAIS

Segundo Bourque & Fairley (2014) os requisitos funcionais são uma propriedade que deve ser exibida por algo para resolver algum problema no mundo real. Tanto pode ter o objetivo de automatizar parte de uma tarefa, como dar suporte aos processos de negócios, corrigir problemas de algum software ou controlar um dispositivo. Sendo assim, após reunião (Figura 48) com os *stakeholders* foram definidos os seguintes requisitos funcionais:

1. Página de login, que permita aos utilizadores fazer login com as mesmas credencias de outra plataforma existente chamada navarraApp;
2. Permitir aos operadores criar registos de inventário de cestos, caixas e retornáveis;
3. Listar as várias contagens e recontagens de cestos, caixas e retornáveis;

4. Listar correções a efetuar no inventário de caixas ou cestos e visualizar quais os inputs dados de forma errada;
5. Conseguir adicionar uma ação de correção a uma determinada contagem de cesto ou caixa;
6. Listar a ação de correção atribuída a uma contagem errada;
7. Conseguir enviar notificações entre utilizadores;
8. Listar as notificações enviadas;
9. Conseguir exportar as várias listagens para o Excel;
10. Permitir abrir e fechar inventários;
11. Listar os inventários abertos e fechados;
12. Possibilidade de ter um teclado virtual no ecrã;
13. Alternar entre modo claro e escuro;
14. Conseguir no ecrã inicial visualizar um estado do inventário;
15. Conseguir consultar a localização de um cesto
16. Conseguir mudar de empresa ou zona de inventário

6.3. REQUISITOS NÃO FUNCIONAIS

Os requisitos não funcionais, também conhecidos como restrições ou requisitos de qualidade, podem ser classificados como requisitos de desempenho, requisitos de manutenção, requisitos de segurança, requisitos de confiabilidade, requisitos de segurança, requisitos de interoperabilidade ou um dos muitos outros tipos de requisitos de software (Bourque & Fairley, 2014).

Os requisitos não funcionais definidos foram os seguintes:

Desempenho: A aplicação não deverá ter quebras de desempenho, tanto quanto à recolha de dados por parte do *backend*, assim como na interface web (*frontend*), sendo que a mesma deverá ser capaz de lidar com 300 ligações simultâneas e manter a disponibilidade 24 horas por dia e 365 dias por ano.

Segurança: A aplicação terá vários níveis de acesso para salvaguardar que apenas determinados utilizadores acedem a funcionalidades restritas. As palavras-passe dos utilizadores estão encriptadas através de uma chave md5 para evitar acessos indevidos.

Portabilidade: A aplicação web deverá estar preparada para ser executada nas plataformas Android, IOS e Windows.

Usabilidade: A aplicação web deverá ser uma solução prática e *user friendly* para o utilizador.

Contudo esta aplicação também deverá ser capaz de:

- Analisar todo o processo atual de inventário de caixas, cestos e retornáveis de forma a poder realizar um levantamento de requisitos;
- Eliminar atividades sem valor acrescentado;
- Agilizar o processo de inventário de alumínio;

6.4. ATORES DO PROJETO

Após a realização de reuniões, foram identificados 3 tipos de atores, que também representam 3 níveis de acesso à informação no protótipo desenvolvido.

Operador

Este ator é responsável por efetuar as contagens e recontagens de cestos, caixas e retornáveis.

No que diz respeito às suas permissões no protótipo, o operador pode fazer login, efetuar contagens e recontagem de cestos, caixas, retornáveis e pode visualizar em lista todas as contagens que efetuou.

Gestor de Inventário

Este membro faz a gestão do inventário, ou seja, é responsável por analisar as diferenças de contagens e tratar essas diferenças. Também notifica os operadores acerca de recontagens necessárias a realizar e de abrir ou encerrar inventários.

Este perfil pode fazer tudo o que o perfil de operador faz, mas adicionalmente pode consultar os registos efetuados por si e pelos operadores. Este perfil também permite realizar toda a gestão de ações corretivas e notificações.

Administrador

O administrador é o responsável máximo pelo processo e não terá qualquer restrição a nível de permissões, tendo uma visão mais holística do processo. O seu papel durante o processo será de monitorização e análise de desempenho.

O perfil de administrador pode efetuar todas as ações dos perfis anteriores e adicionalmente pode realizar toda a gestão de perfis de utilizadores e permissões.

A Figura 9 representa a responsabilidade e hierarquia dos atores no processo de inventário.

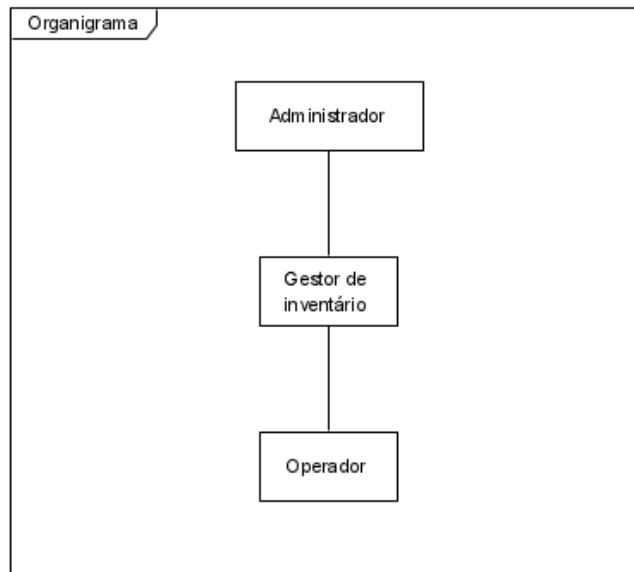


Figura 9 - Organograma do processo

O operador reporta sempre ao gestor do inventário e o gestor de inventário reporta ao administrador. Todos estes atores são internos à empresa.

6.5. CASOS DE USO

Para obter uma melhor compreensão dos requisitos previamente definidos e o modo como os mesmos se podem enquadrar no ponto de vista da cada stakeholder, optou-se pela utilização da linguagem UML (Unified Modeling Language) para descrever as funcionalidades a serem implementadas e os respetivos atores a interagir com as mesmas.

6.5.1. CASOS DE USO DE NÍVEL 0 – SISTEMA PARA GERIR INVENTÁRIOS

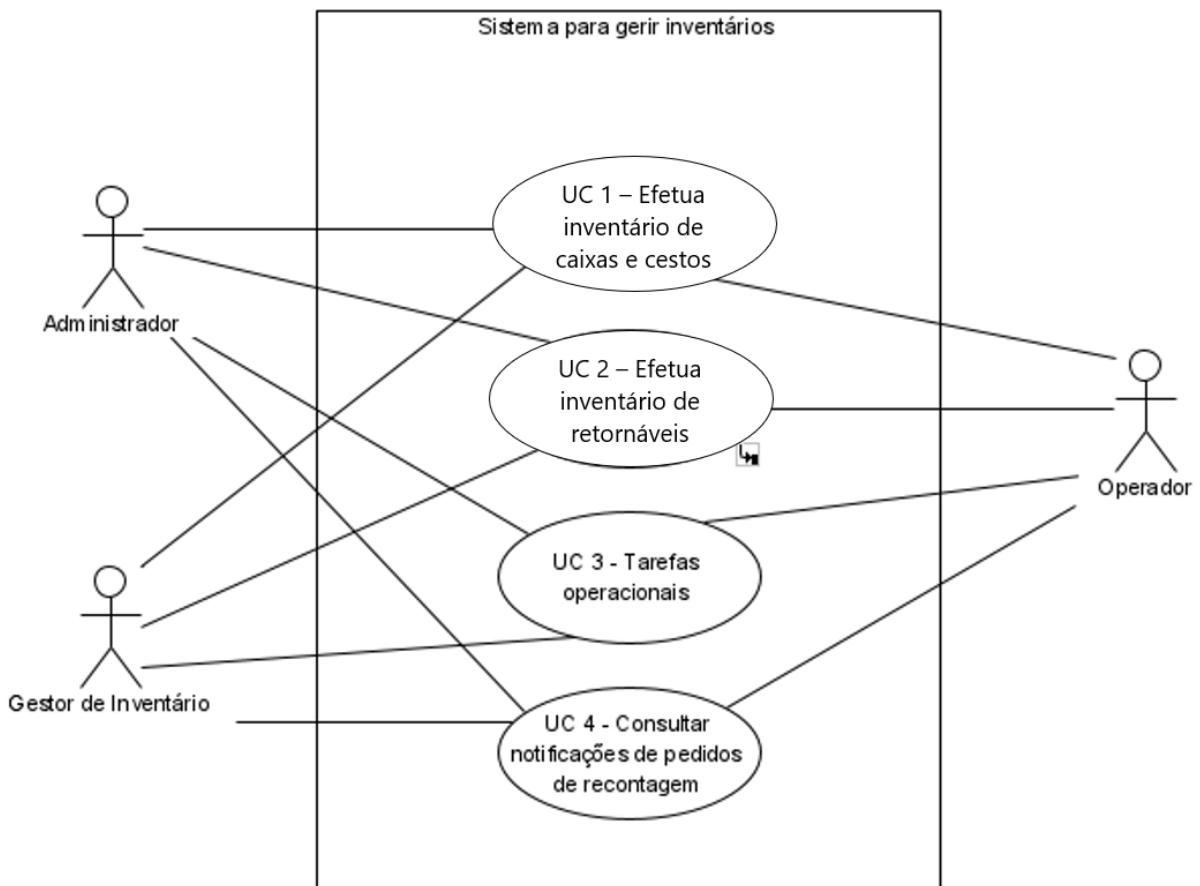


Figura 10 - Diagrama de casos de uso de nível 0 – Sistema para gerir inventários

6.5.2. CASOS DE USO DE NÍVEL 1 – EFETUAR INVENTÁRIO DE CAIXAS E CESTOS

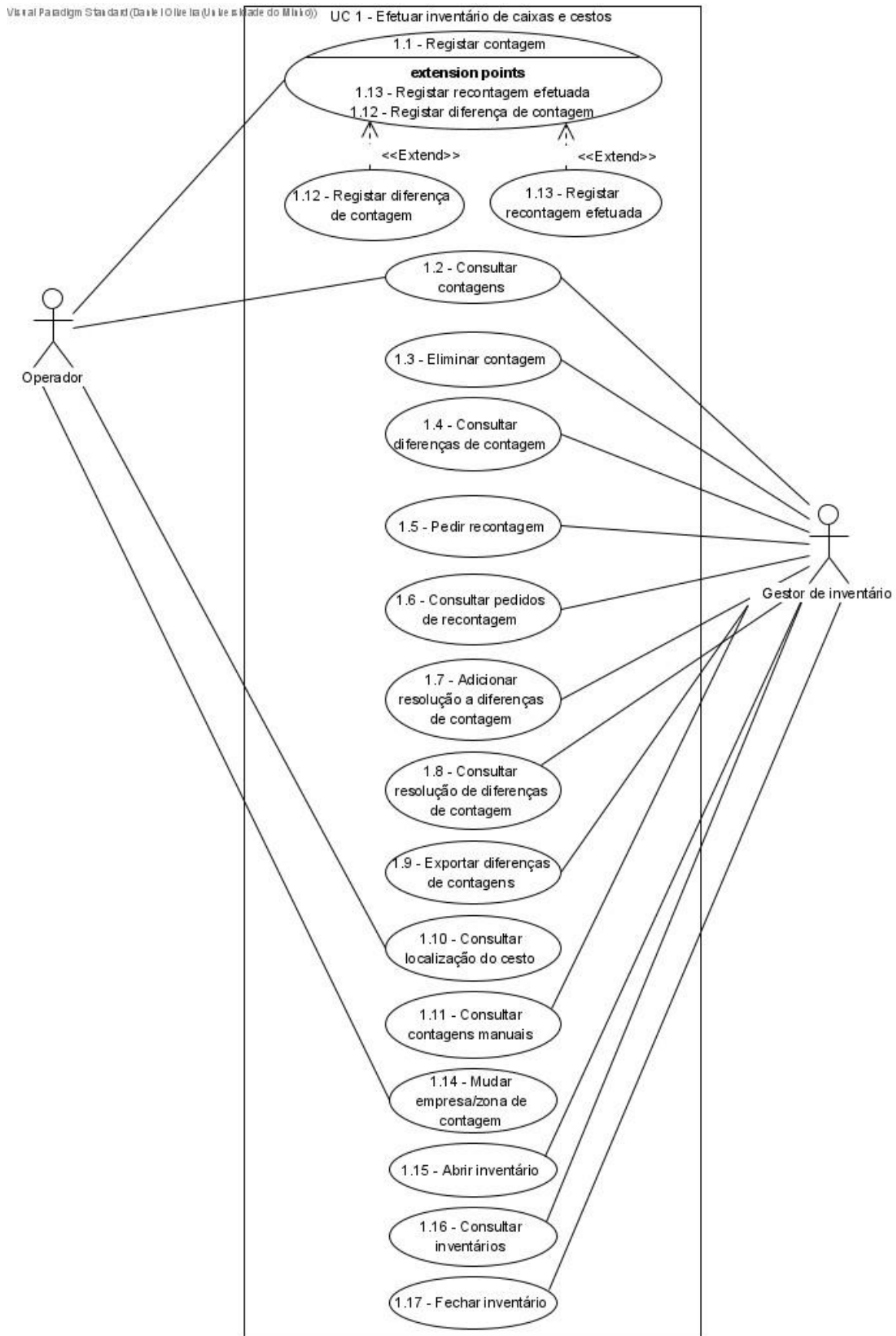


Figura 11 - Diagrama de casos de uso nível 1 – Efetuar inventário de caixas e cestos

Tabela 1 - U1.1 Registrar Contagem

U.1.1 - Registrar Contagem	
Atores	Operador
Descrição	O operador regista no sistema a contagem de uma caixa ou cesto.
Pré-Condições	1 - O utilizador tem de estar autenticado. 2 - Existir um inventário de caixas ou cestos abertos.
Pós-Condição	Contagem registada na base de dados do ERP e na base de dados local.
Fluxo Principal	1- Na interface web o utilizador seleciona a opção inventários de cestos ou caixas e depois clica na opção iniciar. 2 - O utilizador seleciona a empresa e zona onde vai fazer o inventário. 3 - Introduce o n.º de caixa ou cesto a inventariar e clica em pesquisar e retorna a caixa ou cesto 4 - O utilizador clica no botão editar e introduz a medida, tratamento, quantidade de barras e captura uma imagem da caixa ou cesto, e por fim clica em guardar. 5 - O sistema lança uma mensagem de cesto/caixa inventariado com sucesso
Fluxo Alternativo	3.1 – O sistema não encontra a caixa ou cesto como disponível para inventariar 3.2 – O utilizador clica no botão adicionar manualmente e seleciona o perfil, medida, tratamento, quantidade de barras e clica em guardar. 3.3 – O sistema volta para o ponto 5

Diagrama de sequência {U.1.1 - Registrar Contagem}:

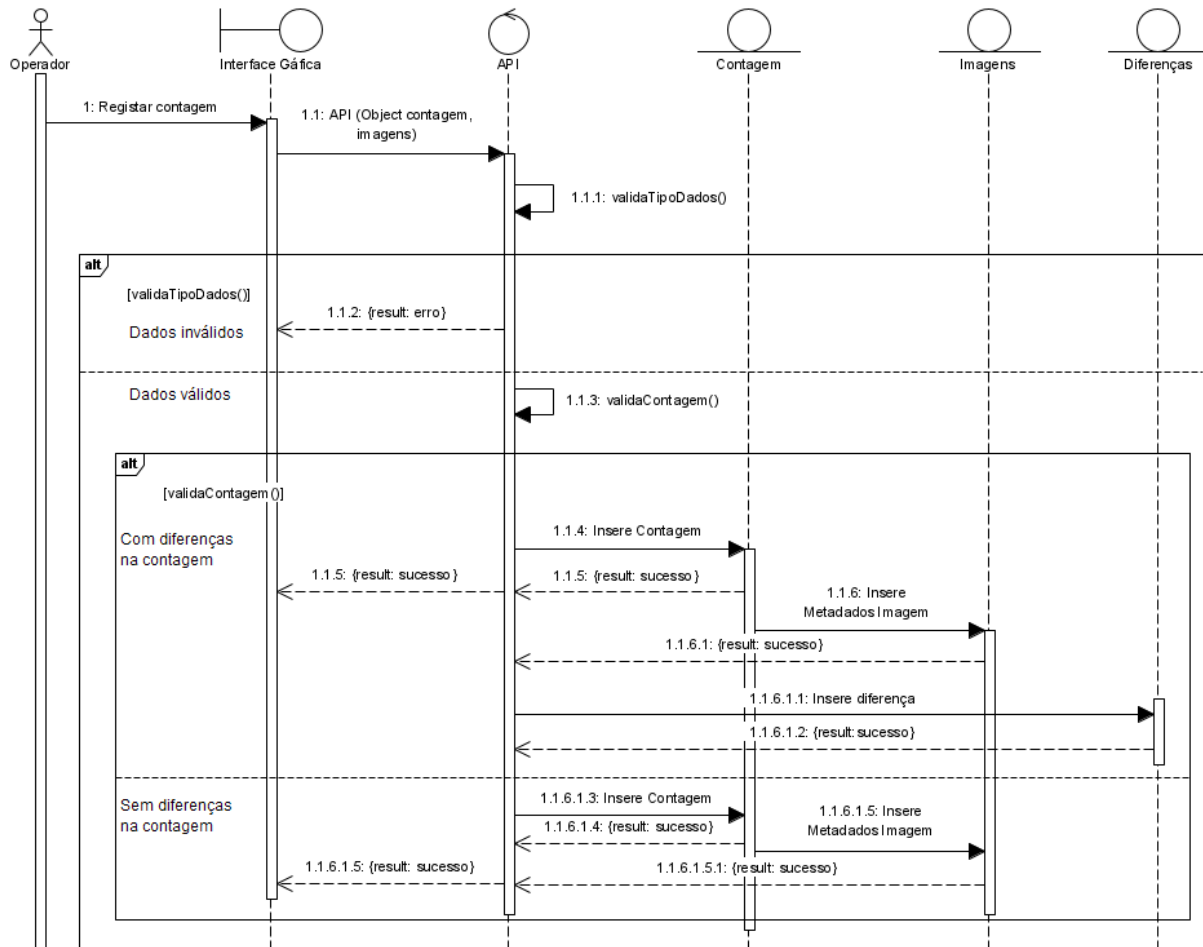


Figura 12 - Diagrama de Sequência - Registrar contagem de caixa ou cesto

Tabela 2 - U.1.2 Consultar Contagem

U.1.2 - Consultar Contagem	
Atores	1 – Operador 2 – Gestor de inventário
Descrição	Um utilizador consulta no sistema as contagens de caixas ou cestos.
Pré-Condições	O utilizador tem de estar autenticado.
Pós-Condição	Mostra uma lista de contagem efetuadas.
Fluxo Principal	1 – Na interface web o utilizador seleciona a opção inventários de cestos ou caixas e depois clica na opção “Listar” 2 – O sistema lista as contagens efetuadas
Fluxo Alternativo	2.1 – O sistema mostra a mensagem “Não foram encontradas contagens”

Diagrama de seqüência {UC 1.2 - Consultar Contagem}:

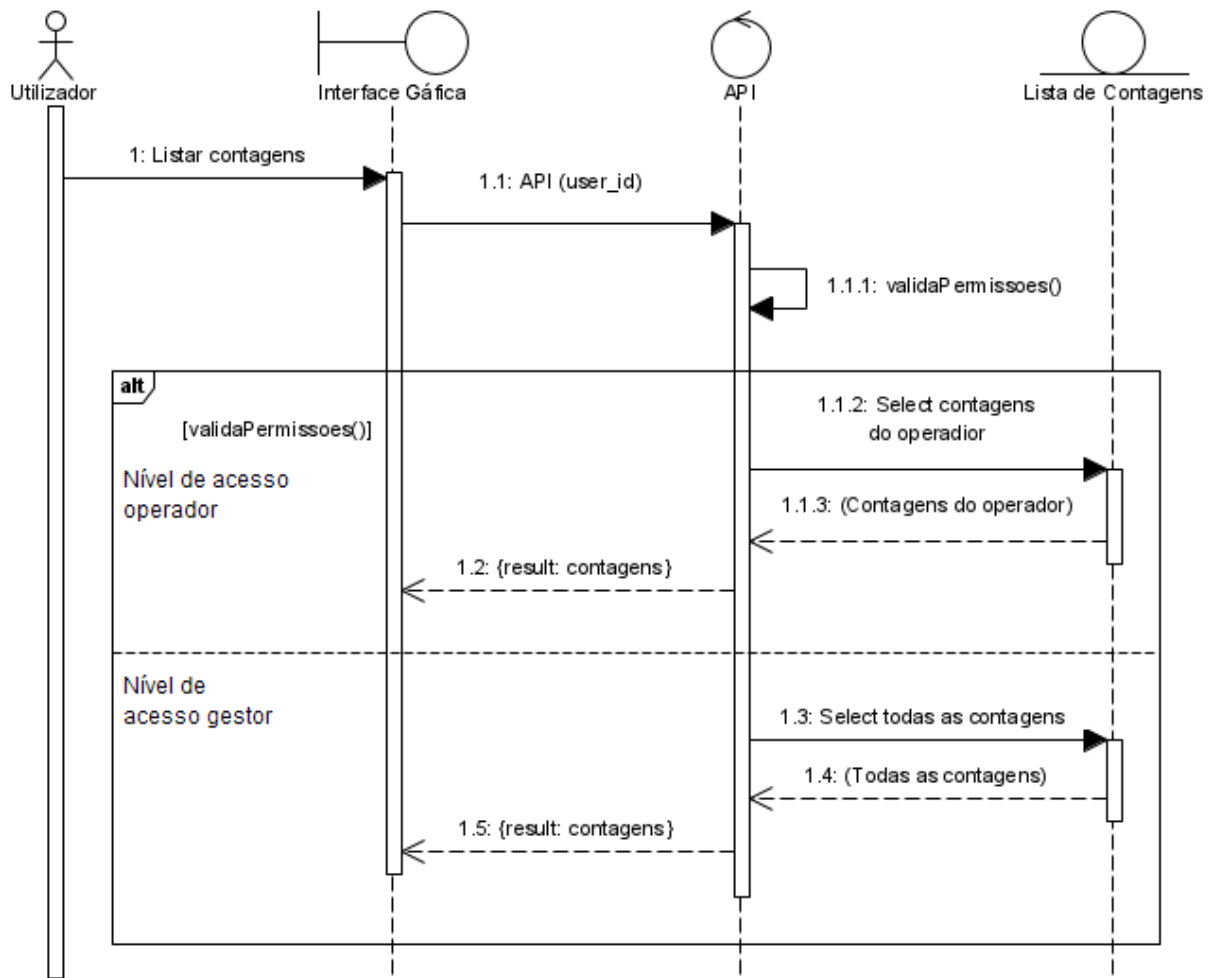


Figura 13 - Diagrama de Sequência - Consultar Contagem

Tabela 3 - U.1.3 Eliminar Contagem

U.1.3 - Eliminar Contagem	
Atores	Gestor de inventário
Descrição	O gestor de inventário elimina uma contagem efetuada de forma errada por parte do operador
Pré-Condições	1 - O utilizador tem de estar autenticado. 2 - O utilizador tem de ter o perfil de gestor
Pós-Condição	Elimina uma contagem efetuada.
Fluxo Principal	1- Na interface web o utilizador seleciona a opção inventários de cestos ou caixas e depois clica na opção "Listar" 2 – O sistema lista as contagens efetuadas 3 – O utilizador clica no botão "Eliminar" 4 - O sistema lança a mensagem "Contagem eliminada com sucesso".
Fluxo Alternativo	2.1 – O sistema mostra a mensagem "Não foram encontradas contagens".

Diagrama de sequência {UC 1.3 - Eliminar Contagem}:

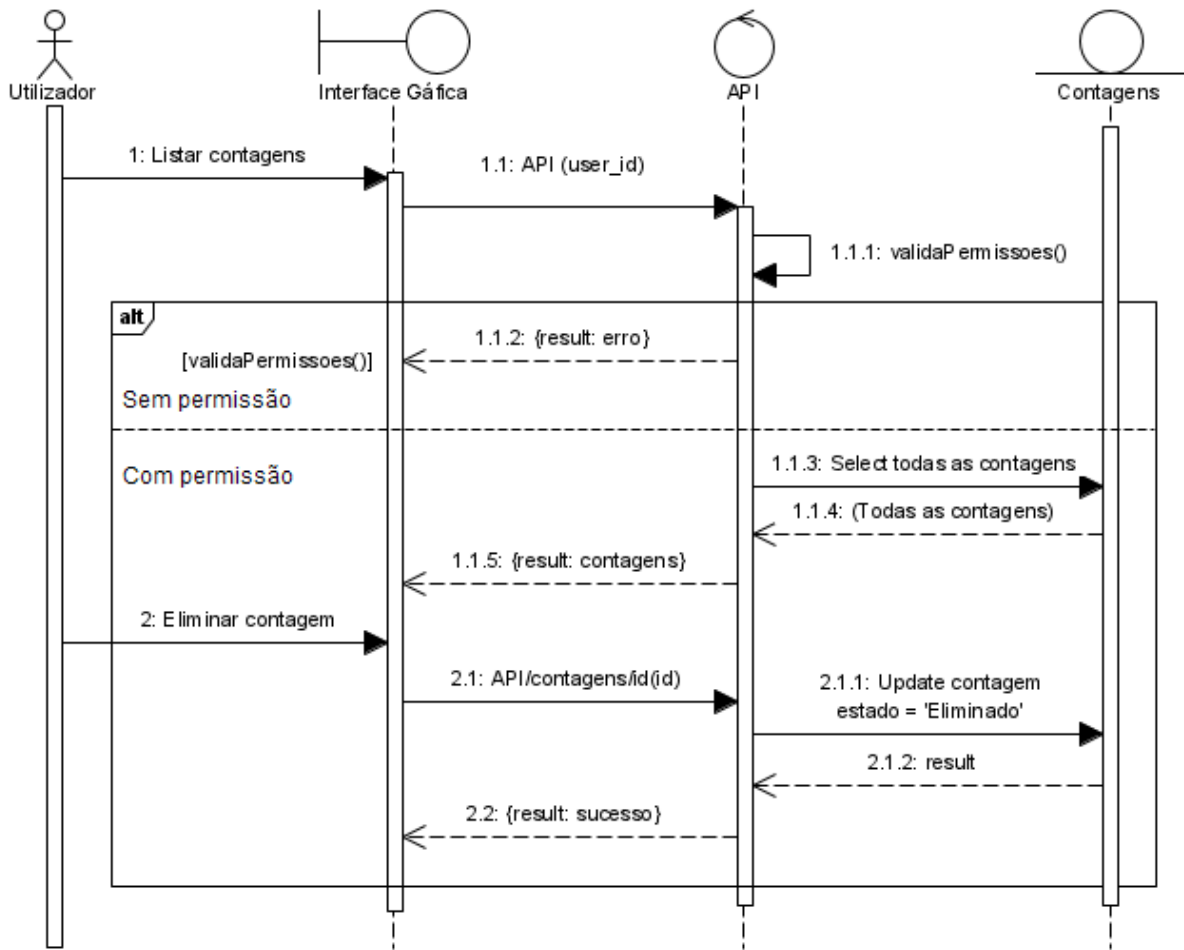


Figura 14 - Diagrama de Sequência – Eliminar contagem de caixa ou cesto

Tabela 4 - U.1.4 Consultar diferenças de contagem

U.1.4 – Consultar diferenças de contagem	
Atores	Gestor de inventário
Descrição	O gestor de inventário consulta a diferenças de contagem.
Pré-Condições	1 – O utilizador tem de estar autenticado. 2 – O utilizador tem de ter o perfil de gestor 3 – Existir contagens com diferenças em relação aos registos do ERP
Pós-Condição	O gestor de inventário obtém uma lista de diferenças de contagem efetuadas pelos operadores.
Fluxo Principal	1 – Na interface web o utilizador seleciona a opção inventários de cestos ou caixas e depois clica na opção “Correções” 2 – O sistema lista as diferenças de contagem
Fluxo Alternativo	2.1 – O sistema mostra a mensagem “Não foram encontradas diferenças de contagens”

Tabela 5 -U.1.5 Pedir recontagem

U.1.5 – Pedir recontagem	
Atores	Gestor de inventário
Descrição	O gestor de inventário pede uma recontagem de uma caixa ou cesto.
Pré-Condições	1 - O utilizador tem de estar autenticado. 2 - O utilizador tem de ter o perfil de gestor
Pós-Condição	Um operador é notificado para fazer uma recontagem.
Fluxo Principal	1 - Na interface web o utilizador seleciona a opção inventários de cestos ou caixas; 2 - Clica na opção “Correções” e seleciona a linha de cesto ou caixa que quer pedir uma recontagem e clica no botão “Notificação”. 3 - Clica na opção “Nova” e o sistema abre um formulário, onde o gestor de inventário seleciona o utilizador que quer notificar.
Fluxo Alternativo	N/A

Diagrama de seqüência {UC 1.5 – Pedir Recontagem}:

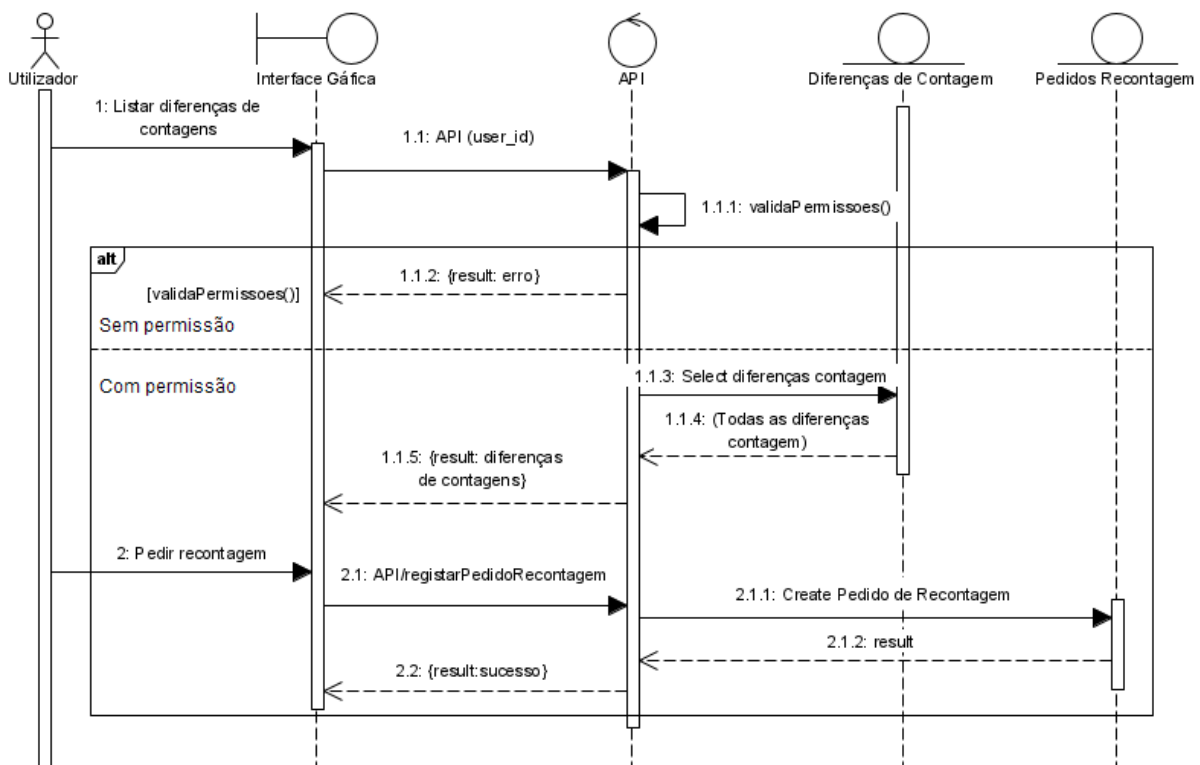


Figura 15 - Diagrama de Sequência - Pedir Recontagem

Tabela 6 - U.1.6 Consultar pedidos de recontagem

U.1.6 – Consultar pedidos de recontagem	
Atores	Gestor de inventário
Descrição	O gestor de inventário obtém uma lista de pedidos de recontagem para efetuados para uma determinada caixa ou cesto.
Pré-Condições	1 - O utilizador tem de estar autenticado. 2 - O utilizador tem de ter o perfil de gestor
Pós-Condição	O gestor de inventário obtém uma lista de pedidos de recontagem para uma caixa ou cesto.
Fluxo Principal	1 - Na interface web o utilizador seleciona a opção inventários de cestos ou caixas; 2 - Clica na opção “Correções” e seleciona a linha de cesto ou caixa que quer pedir uma recontagem e clica no botão “Notificação”. 3 - Clica na opção “Listar” e o sistema lista todos os pedidos de notificação de recontagem para a caixa ou cesto selecionados.
Fluxo Alternativo	3.1 - O sistema mostra a mensagem “Sem dados para apresentar...”

Tabela 7 - U.1.7 Adicionar resolução a diferenças de contagem

U.1.7 – Adicionar resolução a diferenças de contagem	
Atores	Gestor de inventário
Descrição	O gestor de inventário obtém uma lista de pedidos de recontagem para efetuados para uma determinada caixa ou cesto.
Pré-Condições	1 - O utilizador tem de estar autenticado. 2 - O utilizador tem de ter o perfil de gestor
Pós-Condição	Adicionado à caixa ou cesto uma resolução de diferenças de contagem.
Fluxo Principal	1 - Na interface web o utilizador seleciona a opção inventários de cestos ou caixas; 2 - Clica na opção “Correções” e seleciona a linha de cesto ou caixa que quer adicionar uma resolução de recontagem e clica no botão “Correção”. 3 - Clica na opção “Nova” e o sistema abre um formulário, onde o gestor de inventário seleciona a resolução a dar à diferença de contagem.
Fluxo Alternativo	N/A

Tabela 8 - U.1.8 Consultar resolução de diferença de contagem

U.1.8 – Consultar resolução de diferença de contagem	
Atores	Gestor de inventário

U.1.8 – Consultar resolução de diferença de contagem	
Descrição	O gestor de inventário obtém a resolução aplicada a uma diferença de contagem de uma determinada caixa ou cesto.
Pré-Condições	1 - O utilizador tem de estar autenticado. 2 - O utilizador tem de ter o perfil de gestor
Pós-Condição	O gestor de inventário obtém a resolução aplicada a uma diferença de contagem de uma determinada caixa ou cesto.
Fluxo Principal	1 - Na interface web o utilizador seleciona a opção inventários de cestos ou caixas; 2 - Clica na opção “Correções” e seleciona a linha de cesto ou caixa que quer consultar a resolução de diferença de contagem e clica no botão “Resolução”. 3 - Clica na opção “Listar” e o sistema lista a resolução de diferença de contagem para a caixa ou cesto selecionados.
Fluxo Alternativo	3.1 - O sistema mostra a mensagem “Sem dados para apresentar...”

Tabela 9 - U.1.9 Exportar diferenças de contagens

U.1.9 – Exportar diferenças de contagens	
Atores	Gestor de inventário
Descrição	O gestor de inventário obtém uma lista em Excel de todas as diferenças de contagem registadas no sistema, para o inventário de caixas ou cestos abertos.
Pré-Condições	1 - O utilizador tem de estar autenticado. 2 - O utilizador tem de ter o perfil de gestor
Pós-Condição	Lista em Excel de todas as diferenças de contagem.
Fluxo Principal	1 - Na interface web o utilizador seleciona a opção inventários de cestos ou caixas; 2 - Clica na opção “Correções”. 3 - Clica na opção “Exporta” e o sistema exporta para um ficheiro Excel uma lista de todas as diferenças de contagem registadas no sistema.
Fluxo Alternativo	N/A

Tabela 10 - U.1.10 Consultar localização do cesto

U.1.10 – Consultar localização do cesto	
Atores	Operador
Descrição	O operador consulta no sistema a localização de um cesto.
Pré-Condições	1 - O utilizador tem de estar autenticado. 2 - Existir um inventário de caixas ou cestos abertos.
Pós-Condição	Qual a localização do cesto pesquisado.
Fluxo Principal	1- Na interface web o utilizador seleciona a opção inventários de cestos ou caixas e depois clica na opção iniciar. 2 - O utilizador seleciona a empresa e zona onde vai fazer o inventário. 3 - Introduz o n.º de cesto a inventariar e clica em pesquisar e retorna o cesto 4 - O utilizador clica no botão “Localização”. 5 - O sistema abre uma nova janela com as informações da localização do cesto
Fluxo Alternativo	N/A

Tabela 11 - U.1.11 Consultar contagens manuais

U.1.11 - Consultar contagens manuais	
Atores	1 - Operador 2 - Gestor de inventário
Descrição	Um utilizador consulta no sistema as contagens de caixas ou cestos, cujos cestos ou caixas o sistema não encontro registados no ERP.
Pré-Condições	O utilizador tem de estar autenticado.
Pós-Condição	Mostra uma lista de contagem efetuadas de forma manual.
Fluxo Principal	1- Na interface web o utilizador seleciona a opção inventários de cestos ou caixas e depois clica na opção “Manual” 2 - O sistema lista as contagens manuais efetuadas
Fluxo Alternativo	2.1 - O sistema mostra a mensagem “Não foram encontradas contagens”

Tabela 12 - U.1.12 Registrar diferença de contagem

U.1.12 - Registrar diferença de contagem	
Atores	N/A
Descrição	O sistema regista automaticamente uma diferença de contagem, caso o utilizador introduza valores de medida, tratamento ou quantidade de barras, diferentes das que estão registadas no ERP.
Pré-Condições	1 - Existir uma contagem
Pós-Condição	Um registo de diferença de contagem
Fluxo Principal	1- Caso existam diferenças entre os valores que o utilizador introduziu e os que estão no ERP, então o sistema regista automaticamente uma diferença de contagem, para posteriormente ser tratada.
Fluxo Alternativo	N/A

Tabela 13 - U.1.13 Registrar recontagem efetuada

U.1.13 - Registrar recontagem efetuada	
Atores	N/A
Descrição	O sistema envia automaticamente uma notificação para o gestor de inventário caso o operador esteja a fazer uma recontagem de um cesto ou caixa.
Pré-Condições	1 - Já existir uma contagem do cesto ou caixa
Pós-Condição	Uma notificação de registo de recontagem
Fluxo Principal	1 - O sistema envia uma notificação para o gestor de inventário, caso o operador esteja a fazer uma recontagem de um cesto ou caixa e caso o gestor tenha solicitado uma recontagem dessa caixa ou cesto
Fluxo Alternativo	N/A

Tabela 14 - U.1.14 Mudar empresa/zona de contagem

U.1.14 – Mudar empresa/zona de contagem	
Atores	Operador
Descrição	O operador altera no sistema a empresa ou zona de contagem.
Pré-Condições	1 - O utilizador tem de estar autenticado. 2 - Existir um inventário de caixas ou cestos abertos. 3 - O operador já efetuou uma contagem de caixa ou cesto
Pós-Condição	Altera a empresa ou zona de contagem.

Fluxo Principal	<p>1- Na interface web o utilizador clica sobre o avatar do seu nome e obtém o estado do inventário de caixa ou cesto.</p> <p>2 - Caso o inventário esteja aberto e o utilizador já tenha anteriormente selecionado uma empresa ou zona para inventariar, segure um botão chamado “Mudar Empresa / Zona”</p> <p>3 - O utilizador clica nesse botão e o sistema limpa a empresa e zona de inventário e fica disponível para se selecionar nova empresa ou zona</p>
Fluxo Alternativo	N/A

Tabela 15 - U.1.15 Abrir inventário

U.1.15 - Abrir inventário	
Atores	Gestor de inventário
Descrição	O sistema permite ao gestor de inventário abrir um inventário de caixas ou cestos.
Pré-Condições	<p>1 - O utilizador tem de estar autenticado.</p> <p>2 - O utilizador tem de ter o perfil de gestor.</p>
Pós-Condição	Inventário de caixas ou cestos abertos.
Fluxo Principal	<p>1 - Na interface web o utilizador seleciona a opção inventários</p> <p>2 - Depois seleciona a opção “Gerir” e logo a seguir “Abrir”</p> <p>3 - Será aberta uma nova janela, onde o utilizador seleciona o inventário a abrir e depois clica no botão “Abrir Inventário”</p>
Fluxo Alternativo	N/A

Tabela 16 - U.1.16 Consultar inventários

U.1.16 - Consultar inventários	
Atores	Gestor de inventário
Descrição	O sistema permite ao gestor de inventário consultar uma lista de inventário de caixas ou cestos fechados.
Pré-Condições	<p>1 - O utilizador tem de estar autenticado.</p> <p>2 - O utilizador tem de ter o perfil de gestor.</p>
Pós-Condição	Lista de inventários de caixas ou cestos fechados.
Fluxo Principal	<p>1 - Na interface web o utilizador seleciona a opção inventários de cestos ou caixas;</p> <p>2 - Depois seleciona a opção “Gerir” e logo a seguir “Fechar”</p> <p>3 - Será aberta uma nova janela, com todos os inventários abertos.</p>
Fluxo Alternativo	3.1 - O sistema mostra a mensagem “Não foram encontrados inventários abertos”

Tabela 17 - U. 1.17 Fechar inventário

U. 1.17 - Fechar inventário	
Atores	Gestor de inventário
Descrição	O sistema permite ao gestor de inventário fechar um inventário de caixas ou cestos.
Pré-Condições	1 - O utilizador tem de estar autenticado. 2 - O utilizador tem de ter o perfil de gestor. 3 - Existir um inventário de caixas ou cesto aberto.
Pós-Condição	Inventário de caixas ou cestos fechados.
Fluxo Principal	1 - Na interface web o utilizador seleciona a opção inventários de cestos ou caixas; 2 - Depois seleciona a opção “Gerir” e logo a seguir “Fechar” 3 - Será aberta uma nova janela, com todos os inventários abertos. 4 - Seleciona a linha do inventário a fechar e clica no botão “Fechar”
Fluxo Alternativo	N/A

6.5.3. CASOS DE USO NÍVEL 1 – EFETUAR INVENTÁRIO DE RETORNÁVEIS

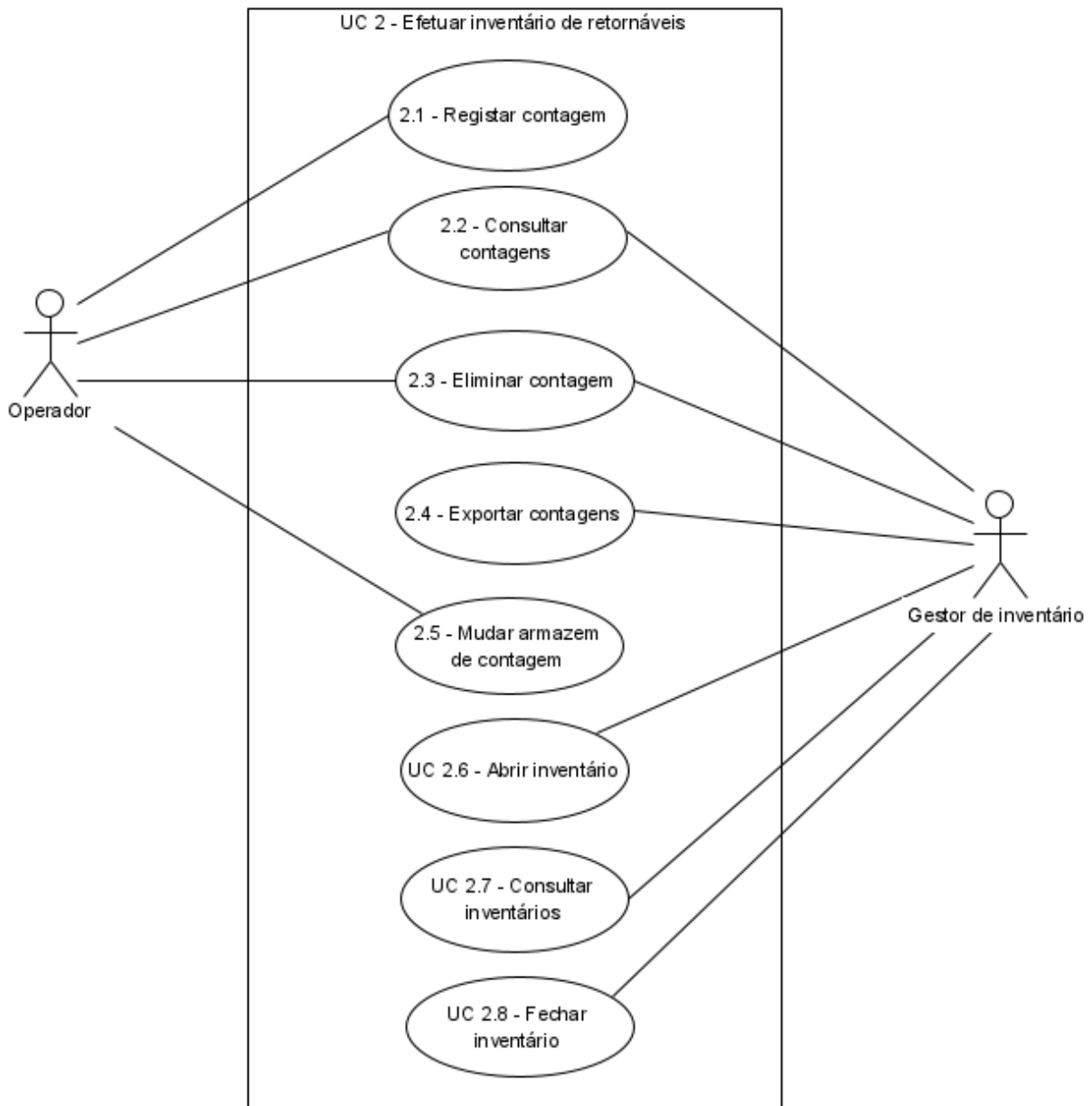


Figura 16 - Diagrama de casos de uso nível 1 – Efetuar inventário de retornáveis

Tabela 18 - U.2.1 Registrar Contagem

U.2.1 - Registrar Contagem	
Atores	Operador
Descrição	O operador registra no sistema a contagem de uma caixa ou cesto.
Pré-Condições	1 - O utilizador tem de estar autenticado. 2 - Existir um inventário de retornáveis aberto.
Pós-Condição	Contagem de um retornável efetuada.

Fluxo Principal	<p>1- Na interface web o utilizador seleciona a opção inventários de retornáveis e depois clica na opção iniciar.</p> <p>2 - O utilizador seleciona o armazém onde vai fazer o inventário.</p> <p>3 - O utilizador introduz o número do cesto, tipo de cesto, se está danificado, a quantidade de chapas, e por fim clica em guardar.</p> <p>4 - O sistema lança uma mensagem de que o retornável foi inventariado com sucesso</p>
Fluxo Alternativo	4.1 - O sistema encontra que aquele n.º de retornável, para aquele armazém e já foi inventariado e lança a mensagem “Retornável já inventariado”

Tabela 19 - U.2.2 Consultar Contagem

U.2.2 - Consultar Contagem	
Atores	<p>1 - Operador</p> <p>2 - Gestor de inventário</p>
Descrição	Um utilizador consulta no sistema as contagens de retornáveis.
Pré-Condições	O utilizador tem de estar autenticado.
Pós-Condição	Mostra uma lista de contagem de retornáveis efetuadas.
Fluxo Principal	<p>1- Na interface web o utilizador seleciona a opção inventários de retornáveis e depois clica na opção “Listar”</p> <p>2 - O sistema lista as contagens efetuadas</p>
Fluxo Alternativo	2.1 - O sistema mostra a mensagem “Não foram encontradas contagens”

Tabela 20 - U.2.3 Eliminar Contagem

U.2.3 - Eliminar Contagem	
Atores	<p>1 - Gestor de inventário</p> <p>2 - Operador</p>
Descrição	O utilizador elimina uma contagem efetuada de forma errada.
Pré-Condições	<p>1 - O utilizador tem de estar autenticado.</p> <p>2 - Existir pelo menos uma contagem de retornáveis</p>
Pós-Condição	Elimina uma contagem de retornáveis efetuada.
Fluxo Principal	<p>1- Na interface web o utilizador seleciona a opção inventários de retornáveis e depois clica na opção “Listar”</p> <p>2 - O sistema lista as contagens efetuadas</p> <p>3 - O utilizador clica no botão “Eliminar”</p> <p>4 - O sistema lança a mensagem “Contagem eliminada com sucesso”.</p>
Fluxo Alternativo	2.1 - O sistema mostra a mensagem “Não foram encontradas contagens”.

Tabela 21 - U.2.4 Exportar contagens

U.2.4 – Exportar contagens	
Atores	Gestor de inventário
Descrição	O gestor de inventário obtém uma lista em Excel de todas as contagens de retornáveis registadas no sistema, para os inventários de retornáveis abertos.
Pré-Condições	1 - O utilizador tem de estar autenticado. 2 - O utilizador tem de ter o perfil de gestor
Pós-Condição	Lista em Excel de todas as contagens de retornáveis.
Fluxo Principal	1 - Na interface web o utilizador seleciona a opção inventários de retornáveis; 2 - Clica na opção “Listar”. 3 - Clica na opção “Exporta” e o sistema exporta para um ficheiro Excel uma lista de todas as contagens de retornáveis registadas no sistema.
Fluxo Alternativo	N/A

Tabela 22 - U.2.5 Mudar armazém de contagem

U.2.5 – Mudar armazém de contagem	
Atores	Operador
Descrição	O operador altera no sistema o armazém onde vai fazer a contagem de retornáveis.
Pré-Condições	1 - O utilizador tem de estar autenticado. 2 - Existir um inventário de retornáveis aberto. 3 - O operador já efetuou uma contagem de retornáveis em algum armazém.
Pós-Condição	Altera o armazém de contagem.
Fluxo Principal	1- Na interface web o utilizador clica sobre o avatar do seu nome e obtém o estado do inventário de retornáveis. 2 - Caso o inventário esteja aberto e o utilizador já tenha anteriormente selecionado um armazém para inventariar, segure um botão chamado “Mudar Armazém” 3 - O utilizador clica nesse botão e o sistema limpa o armazém de inventário e fica disponível para se selecionar um novo armazém.
Fluxo Alternativo	N/A

Tabela 23 - U.2.6 Abrir inventário

U.2.6 - Abrir inventário	
Atores	Gestor de inventário
Descrição	O sistema permite ao gestor de inventário abrir um inventário de retornáveis de um armazém.
Pré-Condições	1 - O utilizador tem de estar autenticado. 2 - O utilizador tem de ter o perfil de gestor.
Pós-Condição	Inventário de retornáveis de um armazém aberto.
Fluxo Principal	1 - Na interface web o utilizador seleciona a opção inventários; 2 - Depois seleciona a opção “Gerir” e logo a seguir “Abrir” 3 - Será aberta uma nova janela, onde o utilizador seleciona o armazém para o qual deseja abrir o inventário de retornáveis e depois clica no botão “Abrir Inventário”
Fluxo Alternativo	N/A

Tabela 24 - U.2.7 Consultar inventários

U.2.7 - Consultar inventários	
Atores	Gestor de inventário
Descrição	O sistema permite ao gestor de inventário consultar uma lista de inventário de armazéns de retornáveis fechados.
Pré-Condições	1 - O utilizador tem de estar autenticado. 2 - O utilizador tem de ter o perfil de gestor.
Pós-Condição	Lista de inventários de armazéns de retornáveis fechados.
Fluxo Principal	1 - Na interface web o utilizador seleciona a opção inventários; 2 - Depois seleciona a opção “Gerir” e logo a seguir “Fechar” 3 - Será aberta uma nova janela, com todos os inventários abertos.
Fluxo Alternativo	3.1 - O sistema mostra a mensagem “Não foram encontrados inventários abertos”

Tabela 25 - U.2.8 Fechar inventário

U.2.8 - Fechar inventário	
Atores	Gestor de inventário
Descrição	O sistema permite ao gestor de inventário fechar um inventário de armazéns de retornáveis.
Pré-Condições	1 - O utilizador tem de estar autenticado. 2 - O utilizador tem de ter o perfil de gestor. 3 - Existir um inventário de armazéns de retornáveis aberto.

Pós-Condição	Inventário de um armazém de retornáveis fechados.
Fluxo Principal	1 - Na interface web o utilizador seleciona a opção inventários; 2 - Depois seleciona a opção “Gerir” e logo a seguir “Fechar” 3 - Será aberta uma nova janela, com todos os inventários abertos. 4 - Seleciona a linha do inventário a fechar e clica no botão “Fechar”
Fluxo Alternativo	N/A

Para mapear de forma ainda mais clara a estrutura do protótipo desenvolvido foi criado um diagrama de classes que pode ser consultado na secção dos anexos, nomeadamente a Figura 49 - Diagrama de Classes Figura 49.

6.6. ESTRUTURA ANALÍTICA DO PROJETO (EAP)

A Figura 17 representa a estrutura de decomposição do trabalho baseada em *deliverables* e demonstra claramente a relação entre os *deliverables* do projeto e o seu âmbito.

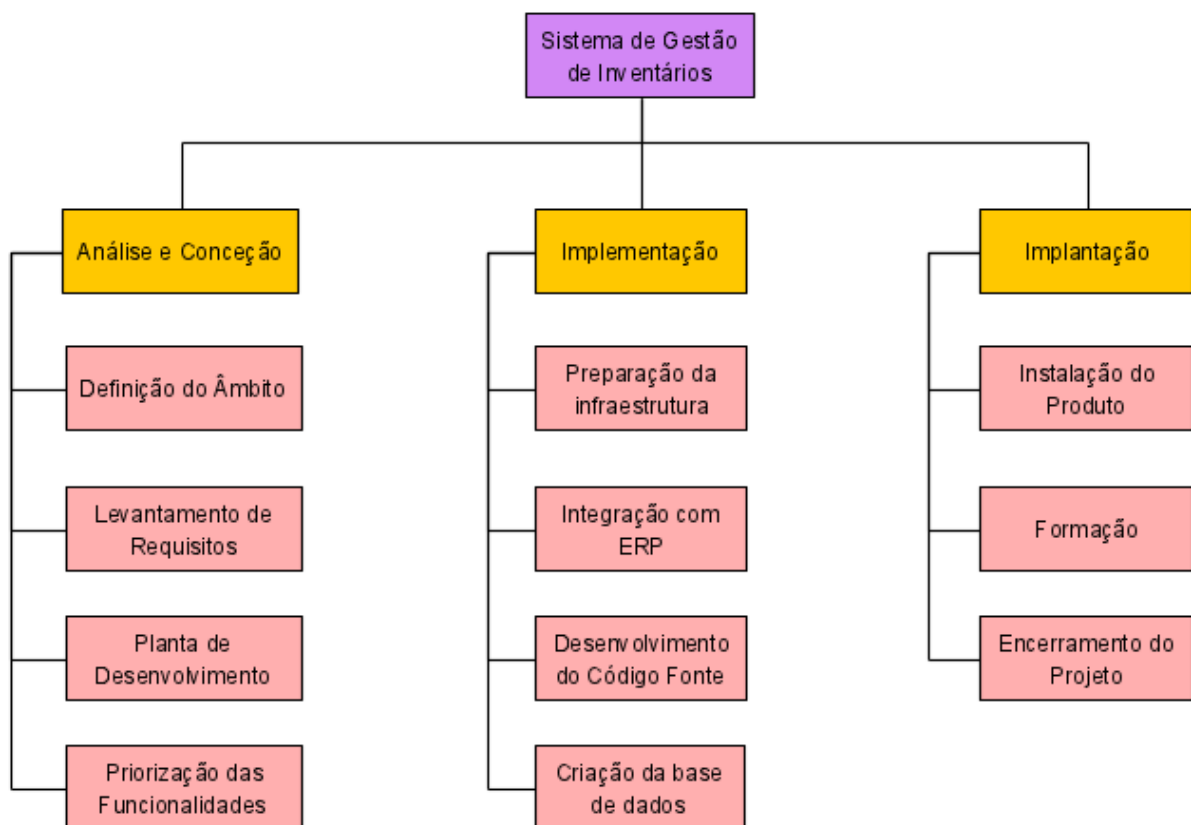


Figura 17 - Estrutura Analítica do Projeto (EAP)

6.7. PRODUCT BACKLOG

O *Product Backlog* apresentado na Tabela 26 abaixo representa a lista de funcionalidades da aplicação e atividades de configurações da infraestrutura e de criação da base de dados. Atividades essas que devem ser executadas ao longo do projeto de forma a conseguir atingir os resultados esperados.

Tabela 26 - Product Backlog

	ID	Descrição	Prio.
High Priority (1- 2)	1	Instalação e configuração da infraestrutura	1
	2	Criação e configuração da base de dados	1
	3	Criar página de login, que permita aos utilizadores realizar login com as mesmas credencias de outra plataforma existente chamada navarraApp	1
	4	Permitir aos operadores poderem criar registos de inventário de cestos, caixas e retornáveis;	1
	5	Listar as várias contagens e recontagens de cestos, caixas e retornáveis;	1
	6	Listar correções a efetuar no inventário de caixas ou cestos e visualizar quais os inputs dados de forma errada;	1
	7	Conseguir adicionar uma ação de correção a uma determinada contagem de cesto ou caixa	1
	8	Listar a ação de correção atribuída a uma contagem errada	2
	9	Permitir abrir e fechar inventários	2
	10	Listar os inventários abertos e fechados	2
Medium Priority (3- 4)	11	Conseguir enviar notificações entre utilizadores	3
	12	Listar as notificações enviadas	3
	13	Conseguir visualizar no ecrã inicial o estado do inventário	4
Low Priority (5+)	14	Conseguir exportar as várias listagens para o excel	5
	15	Conseguir consultar a localização de um cesto	5
	16	Conseguir alterar a empresa ou zona de inventário	5
	17	Possibilidade de ter um teclado virtual no ecrã	6
	18	Alternar entre modo claro e escuro	6

6.8. SPRINT BACKLOG

No *Sprint Backlog* representado na tabela Tabela 27 foram definidos sprint de 3 em 3 semanas e contém uma lista com todas as funcionalidades necessárias a implementar no protótipo desenvolvido. Também apresenta todas as entregas que devem ser realizadas, pelo que permite organizar as tarefas e realizar um ponto de situação a cada sprint.

Tabela 27 - Sprint Backlog

ID Requisito	ID Planta	Tarefa	Estado
Sprint 1 - Semana (14/02/2022 a 04/03/2022)			
1	—	Instalação e configuração do VueJS, NodeJS	Concluído
1	—	Instalação e configuração das extensões e plugins do VueJS e NodeJS necessário para a aplicação	Concluído
1	—	Instalação e configuração do <i>Reverse Proxy</i> no IIS para funcionamento da Web API	Concluído
2	—	Desenvolvimento da base de dados em MySQL	Concluído
Sprint 2 - Semana (07/03/2022 a 25/03/2022)			
1	—	Criação do layout e estilo da aplicação	Concluído
3	—	Criação da página de login	Concluído
13	—	Criação de página que permite verificar o estado do inventário	Concluído
4	U1.1; DS- U.1.1;	Criação da página de registo de inventário de caixas	Concluído
5	U1.2; DS- U.1.2;	Criação da página de consulta de inventário de caixas	Concluído
Sprint 3 - Semana (28/03/2022 a 08/04/2022)			
5	U1.3; DS U1.3	Desenvolvimento de funcionalidade de eliminar contagem no inventário de caixas	Concluído
6	U1.4; DS U1.4	Criação da página de consulta de diferença de contagem do inventário de caixas	Concluído
11	U1.5; DS- U.1.5	Criação da página de pedir recontagem do inventário de caixas	Concluído

ID Requisito	ID Planta	Tarefa	Estado
12	U1.6	Criação da página de consulta de pedidos de contagem do inventário de caixas	Concluído
7	U1.7	Criação da página de criar uma resolução de diferenças de recontagem do inventário de caixas	Concluído
8	U1.8	Criação da página de consulta de resolução de diferenças de contagem do inventário de caixas	Concluído
14	U1.9	Desenvolvimento da funcionalidade de exportação da listagem de diferenças de contagem do inventário de caixas para Excel	Concluído
Sprint 4 - Semana (11/04/2022 a 29/04/2022)			
15	U1.10	Criação de página de consulta de localização de um cesto	Concluído
5	U1.11	Criação de página de consulta de contagens manuais	Concluído
5	U1.12	Criação de página de registo de diferenças de contagem do inventário de caixas	Concluído
12	U1.13	Desenvolvimento de funcionalidade de registo de recontagem do inventário de caixas efetuada com sucesso	Concluído
16	U1.14; U2.5	Criação de página de registo mudança de empresa ou zona de contagem	Concluído
4	U1.1; DS- U.1.1;	Criação da página de registo de inventário de cestos	Concluído
5	U1.2; DS- U.1.2;	Criação da página de consulta de inventário de cestos	Concluído
5	U1.3; DS U1.3	Desenvolvimento de funcionalidade de eliminar contagem no inventário de cestos	Concluído
Sprint 5 - Semana (02/05/2022 a 20/05/2022)			
6	U1.4; DS U1.4	Criação da página de consulta de diferença de contagem do inventário de cestos	Concluído
11	U1.5; DS- U.1.1	Criação da página de pedir recontagem do inventário de cestos	Concluído
12	U1.6	Criação da página de consulta de pedidos de contagem do inventário de cestos	Concluído
7	U1.7	Criação da página de criar uma resolução de diferenças de recontagem do inventário de cestos	Concluído
8	U1.8	Criação da página de consulta de resolução de diferenças de contagem do inventário de caixas	Concluído
Sprint 6 - Semana (23/05/2022 a 10/06/2022)			

ID Requisito	ID Planta	Tarefa	Estado
14	U1.9	Desenvolvimento da funcionalidade de exportação da listagem de diferenças de contagem do inventário de cestos para Excel	Concluído
5	U1.12	Criação de página de registo de diferenças de contagem do inventário de caixas	Concluído
12	U1.13	Desenvolvimento de funcionalidade de registo de recontagem do inventário de cestos efetuada com sucesso	Concluído
Sprint 7 - Semana (13/06/2022 a 24/06/2022)			
4	U2.1	Criação de página de registo de contagem do inventário de retornáveis	Concluído
5	U2.2	Criação de página de consulta de contagens do inventário de retornáveis	Concluído
5	U2.3	Desenvolvimento de funcionalidade de eliminar contagem do inventário de retornáveis	Concluído
14	U2.4	Desenvolvimento da funcionalidade de exportação da listagem de contagens de retornáveis para Excel	Concluído
9	U1.15; U2.6	Criação de página de abertura de inventário	Concluído
10	U1.16; U2.7	Criação de página de consulta de inventários abertos	Concluído
9	U1.17; U2.8	Desenvolvimento de funcionalidade de fecho de inventário	Concluído

6.9. DIAGRAMA DE ENTIDADES E RELACIONAMENTOS (DER)

A Figura 18 representa o diagrama de entidades e relacionamentos da aplicação que irá gerir os inventários de perfis em cursos de fabrico, embalado e cestos retornáveis. A base de dados é composta por um total de 9 tabelas interligadas e 2 tabelas sem interligações.

A base de dados desenhada é composta pelas seguintes tabelas:

- `tbl_app_invent_cab`: guarda os dados relativo ao início, fim e estado do inventário dos cestos e caixas;
- `tbl_app_invent_caixas`: guarda todas as contagens das caixas;
- `tbl_app_invent_caixas_alteracoes`: guarda as contagens das caixas que não coincidem com as que estão no ERP;
- `tbl_app_invent_cestos`: guarda todas as contagens das caixas;
- `tbl_app_invent_cestos_alteracoes`: guarda as contagens de cestos que não coincidem com as que estão no ERP;
- `tbl_app_invent_img`: guarda as capturas de imagens que são realizadas aos cestos;
- `tbl_app_invent_notificacoes`: guarda todas as notificações de contagens necessárias de fazer;
- `tbl_app_invent_retornaveis`: guarda os dados relativo ao início, fim e estado do inventário dos retornáveis;
- `tbl_app_invent_retornaveis_linhas`: guarda todos os dados relativos às contagens de retornáveis;
- `tbl_users`: guarda os dados dos utilizadores da aplicação;
- `tbl_app_user_funcionalidades`: guarda informações relativas às funcionalidades que compõem a aplicação;
- `tbl_app_user_permissoes`: guarda informações relativas às permissões que os utilizadores têm relativamente a uma determinada funcionalidade.

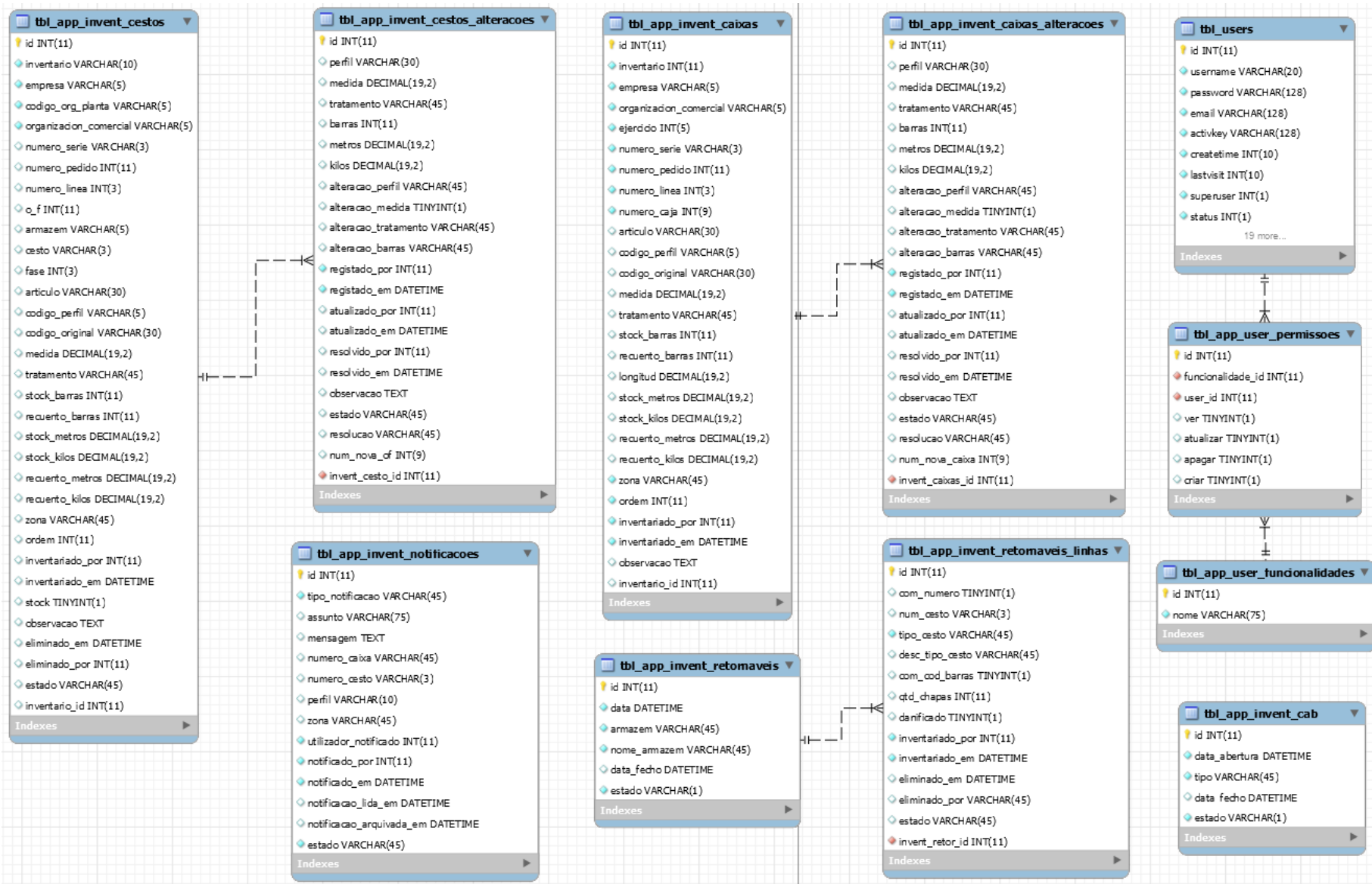


Figura 18 - Diagrama de Entidades e Relacionamentos

6.10. FERRAMENTAS UTILIZADAS

As ferramentas utilizadas no decorrer do desenvolvimento foram escolhidas levando em consideração as experiências obtidas em projetos anteriores. Na Tabela 28 e Tabela 29 estão expostas as ferramentas utilizadas em desenvolvimento e em produção.

Tabela 28 - Ferramentas utilizadas em desenvolvimento

Em desenvolvimento	
Visual Studio Code	Desenvolvimento da aplicação
Mysql Workbenck 8.0	Gestor da base de dados
Postman	Aplicação para testar API's
IIS 8	Servidor Web de Gestão de Aplicações
PHP 7.3.20	Biblioteca da linguagem PHP
MySQL Server 5.5	Base de Dados
NodeJS 14.15.0	Biblioteca da linguagem JavaScript
Nodemon 2.0.13	Ferramenta para desenvolvimento de aplicações nodeJS
VueJS 2.6.11	Framework JavaScript de <i>Open Source</i>

Tabela 29 - Ferramentas utilizadas em produção

Em produção	
IIS 8	Servidor Web de Gestão de Aplicações
PHP 7.3.20	Biblioteca da linguagem PHP
MySQL Server 5.5	Base de Dados
NodeJS 14.15.0	Biblioteca da linguagem JavaScript
PM2	Gestor de execução de processos NodeJS
Vue JS 2.6.11	Framework JavaScript de <i>Open Source</i>

7. PROTÓTIPO

O protótipo aqui apresentado reflete um sistema implementado na empresa que se encontra em produção, sendo que todas as suas funcionalidades serão descritas ao longo deste capítulo.

7.1. BACKEND

O *backend* foi desenvolvido em *NodeJs* e instalada a versão 14.15.0 na máquina de produção *Windows Server 2012 R2 Datacenter*, utilizada para alojar o protótipo desenvolvido. Além desta instalação foi necessário configurar um *Reverse Proxy* no IIS, para capturar o tráfego de entrada e encaminhar para o serviço de *backend*. A configuração do *Reverse Proxy* no IIS 8, foi executada da seguinte forma:

- Passo 1: Instalação do *Routing Extensions* e do *URL rewrite extension*
- Passo 2: Adicionar uma regra de *Reverse Proxy*, onde a API ficou a responder aos pedidos na porta 3000. Assim os pedidos feitos ao endereço público da API, serão encaminhados para o endereço local “http://localhost:3000”. Por exemplo, pedidos feitos a `http://api.myserver.com/API/v1/inventario/localizacao/112` serão encaminhados para `http://localhost:3000` como `http://localhost:3000/API/v1/inventario/localizacao/112`.

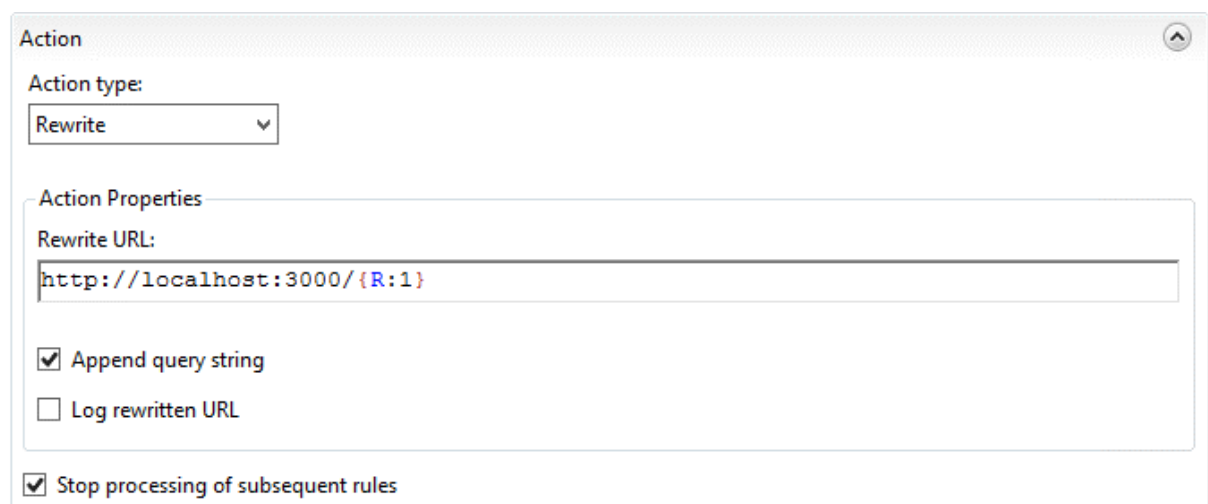


Figura 19 - Regra Reverse Proxy no IIS

- Passo 3: Habilitar as configurações do proxy no IIS

Application Request Routing

Use this feature to configure proxy settings for Application Request Routing.

Enable proxy

Proxy Setting

HTTP version:

Pass through

Keep alive

Time-out (seconds):

120

Reverse rewrite host in response headers

Custom Headers

Preserve client IP in the following header:

X-Forwarded-For

Include TCP port from client IP

Figura 20 - Habilitar proxy no IIS

No desenvolvimento do *backend* foram utilizadas muitas frameworks nativas do *NojeJS* que facilitaram o desenvolvimento e ajudaram a tornar o *backend* mais robusto. O *Express*, foi uma dessas frameworks utilizadas e tem como função gerir os pedidos/respostas HTTP. O *Error Handling* é outro componente utilizado neste projeto e está ligado ao *Express* e ajuda a tratar e a processar todos os erros que ocorrem de uma forma síncrona e assíncrona. Outro componente utilizado é o *body-parse*, responsável por extrair parte do corpo de um pedido de entrada e expõe em um *req.body*. Estes são alguns exemplos, entre muitos, componentes *NojeJS* utilizados que ajudaram a agilizar o desenvolvimento e a responder aos requisitos propostos.

Quanto à estrutura de ficheiros do *backend*, a mesma foi definida para funcionar de uma forma mais otimizada e produtiva com a *framework Express*, já que é uma estrutura que permite manter a consistência do projeto caso seja necessário no futuro trabalhar com equipas de desenvolvimento maiores. A Figura 21 representa a estrutura do *backend* definida para este projeto.

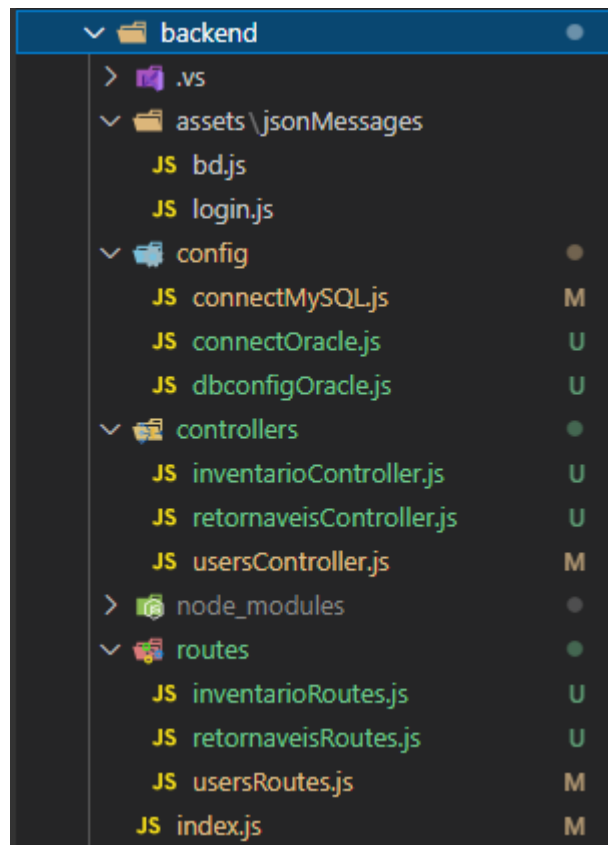


Figura 21 - Estrutura do Backend

A pasta “**assets/jsonMessagem**” contém ficheiros *javascript* onde o seu conteúdo é um JSON de mensagens personalizadas para validação dos dados de entrada na API e respostas da API. Estas validações são geridas através da biblioteca *express-validator* que reúne um conjunto de *middlewares* de validação e limpeza de dados. Além de ser importante fazer uma validação dos dados já do lado do *frontend*, também é extremamente importante realizar a validação dos dados do lado do *backend*. Estas boas práticas ajudam a garantir uma maior confiabilidade da aplicação, prevenir erros do utilizador e a prevenir erros em acesso direto à API.

A pasta “**assets/jsonMessagem**” tem 2 ficheiros, que são eles o “bd.js” que contém mensagens de resultados de ações feitas à base de dados e o “login.js” que possui mensagens de ações no login da aplicação.

Na Figura 22 encontra-se uma parte do código do ficheiro “bd.js” que contém mensagens de validação de resultados da base de dados.

```

module.exports = {
  db: {
    noRecords: {
      msg: "No records found",
      message: {
        eng: "No Records found",
        pt: "Não foram encontrados dados"
      },
      status: 404,
      success: false
    },
    dbError: {
      msg: "Error",
      message: {
        eng: "Invalid data",
        pt: "Os dados que inseriu são inválidos!"
      },
      success: false,
      status: 400,
    },
    successUpdate: {
      msg: "success",
      message: {
        eng: "Records updated with success",
        pt: "Dados alterados com sucesso"
      },
      success: true,
      status: 200,
    },
  },
  (...),
}

```

Figura 22 - Mensagens de resultados de ações à BD

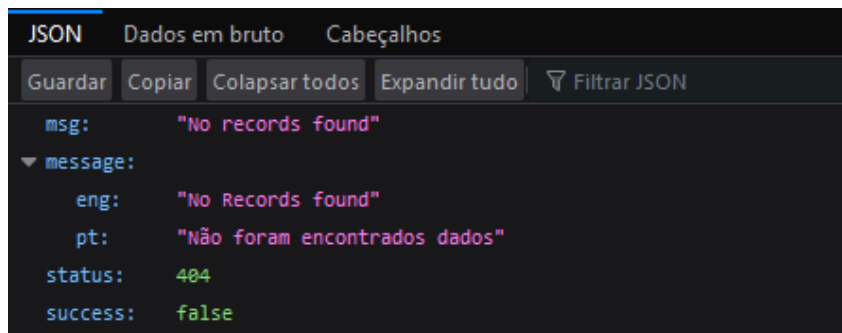
Na Figura 23 encontra-se um exemplo de código de tratamento de resultado de uma consulta à base de dados, ou seja, caso o resultado da consulta tenha um número de linha maior que 0, então a API retorna um *array* com os resultados da consulta, em caso contrário a API apresenta uma mensagem do tipo `jsonMessages.db.noRecords`, tal como ilustrado na Figura 24.

```

if(result.rows.length > 0) {
  res.send(result.rows);
} else {
  res.status(jsonMessages.db.noRecords.status).send(jsonMessages.db.noRecords);
}

```

Figura 23 - Exemplo de código do tratamento de resultados



```
JSON  Dados em bruto  Cabeçalhos
Guardar Copiar Colapsar todos Expandir tudo Filtrar JSON
{
  "msg": "No records found",
  "message": {
    "eng": "No Records found",
    "pt": "Não foram encontrados dados"
  },
  "status": 404,
  "success": false
}
```

Figura 24 - Mensagem JSON "No records found"

Já na pasta “**config**” do *backend* encontram-se os ficheiros de configuração de conexão à base de dados MySQL (*connectMySQL.js*) e à base de dados Oracle (*connectOracle.js* e *dbconfigOracle.js*). Na Figura 25 é apresentado o código fonte do ficheiro *connectMySQL.js* que representa a conexão à base de dados MySQL, sendo que, por questões de segurança foram eliminados os dados da conexão ao servidor MySQL.

```
var mysqlConnection = mysql.createPool({
  connectionLimit: 1000,
  host: process.env.DB_HOST_DEV,
  user: process.env.DB_USER_DEV,
  password: process.env.DB_PASS_DEV,
  database: process.env.DB_NAME_DEV,
  multipleStatements: true,
  dateStrings: true
});
mysqlConnection.getConnection((err, connection) => {
  if (err) {
    if (err.code === 'PROTOCOL_CONNECTION_LOST') {
      console.error('MySQL DB connection was closed.')
    }
    if (err.code === 'ER_CON_COUNT_ERROR') {
      console.error('MySQL DB has too many connections.')
    }
    if (err.code === 'ECONNREFUSED') {
      console.error('MySQL DB connection was refused.')
    }
  }
  if (connection) {
    connection.release();
    console.log('MySQL DB connection was successful!');
  }
  return;
});
```

Figura 25 - Script de conexão MySQL

Na pasta “**controller**” é onde se encontram os ficheiros que contêm o código fonte dos métodos disponibilizados pela API, que fazem alterações ou consultas à base de dados. Já na pasta “**routes**” encontra-se os ficheiros onde são definidos os caminhos da API para aceder aos métodos e onde está definido se o caminho é do tipo GET, PUT ou POST.

Na Figura 26 é apresentado um exemplo de uma rota da API, que neste caso responde com a localização de um cesto, quando é solicitado pelo *frontend*. A Figura 27 contém o código fonte de uma pesquisa à base dados para obter a localização de um cesto dentro da empresa.

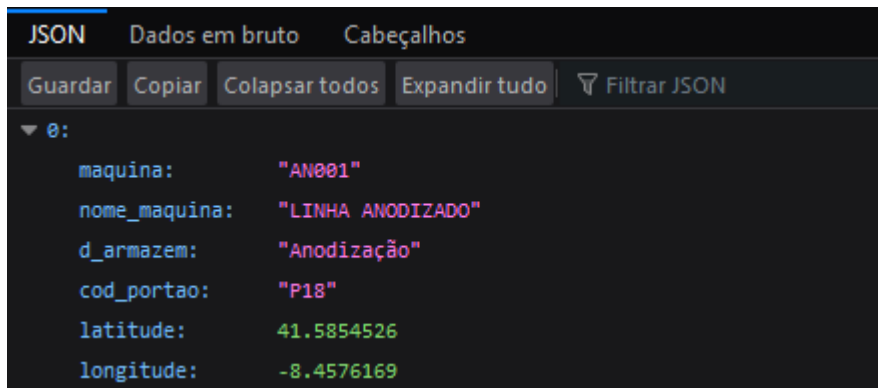
```
//retorna a localização do cesto passado no parâmetro ":id"  
router.get('/inventario/localizacao/:id', InventarioController.localizacaoCesto);
```

Figura 26 - Routes localização de um cesto

```
//retorna a localização do cesto passado no parâmetro ":id"  
controller.localizacaoCesto = (req, res) => {  
  if (!req.params.id) {  
    res.status(jsonMessages.db.requiredData.status).send(jsonMessages.db  
.requiredData);  
  } else {  
    mysqlConnection.query(  
      SELECT o.maquina, (SELECT m.desc_maquina FROM tbl_maquina m WHERE  
m.codigo_maquina = o.maquina) nome_maquina, o.d_armazem, p.cod_portao,  
(SELECT ic.latitude FROM tbl_app_invent_cestos ic WHERE ic.cesto = ?  
ORDER BY ic.inventariado_em DESC LIMIT 1) latitude, (SELECT  
ic.longitude FROM tbl_app_invent_cestos ic WHERE ic.cesto = ? ORDER BY  
ic.inventariado_em DESC LIMIT 1) longitude  
FROM tbl_ofcestos o  
LEFT JOIN tbl_portao p ON p.local = o.maquina  
WHERE o.cesto = ?  
GROUP BY o.maquina', [req.params.id, req.params.id, req.params.id],  
(err, rows, fields) => {  
  if (!err) {  
    if (rows.length > 0) {  
      res.send(rows);  
    } else {  
      res.send('NoDataFound');  
    }  
  }  
});  
}  
};
```

Figura 27 - Função localização de um cesto

Na Figura 28 é apresentada a resposta do *backend* ao *frontend* quando solicitada a localização de um cesto.



```
JSON  Dados em bruto  Cabeçalhos
Guardar Copiar Colapsar todos Expandir tudo Filtrar JSON
0:
  maquina: "AN001"
  nome_maquina: "LINHA ANODIZADO"
  d_armazem: "Anodização"
  cod_portao: "P18"
  latitude: 41.5854526
  longitude: -8.4576169
```

Figura 28 – Resultado JSON da consulta da localização do cesto

Por fim, o **“index.js”** é o ficheiro por onde o projeto do *backend* inicializa e onde são invocadas as camadas de *middleware*, tais como, o *Express*, *Error Handler* e o *Body Parser*. Também é onde se define o *Header* da aplicação e a porta onde a API estará à escuta e onde são importados e invocados os ficheiros das rotas.

7.2. FRONT-END

Na Figura 29 é apresentada a página de login do protótipo, onde através do preenchimento do nome de utilizador e palavra-passe o utilizador se poderá autenticar na aplicação. Os utilizadores que se podem autenticar nesta aplicação são os mesmos que se podem autenticar em outra aplicação da empresa, chamada navarraApp, ou seja, estas duas aplicações partilham a mesma base de dados de utilizadores.

Logo após o utilizador se autenticar na aplicação, surgirá a página de resumo da situação do inventário, conforme se pode verificar na Figura 30. Nesta página o utilizador poderá obter um ponto de situação do inventário, onde pode consultar as caixas ou cestos que são necessários inventariar e quantas caixas ou cestos já estão inventariados.



Figura 29 - Página de login



Figura 30 - Dashboard de situação do inventário

Ainda na página inicial da aplicação, nomeadamente no seu *header* o utilizador pode contar com mais duas funcionalidades.

A primeira diz respeito às notificações de contagens e recontagens de cestos ou caixas que são trocadas na aplicação, acessível através do clique no ícone do "sino". Nesta funcionalidade (Figura 31), o utilizador

poderá consultar as notificações recebidas, enviadas e arquivadas, e quando clica na notificação poderá obter mais informações do pedido de recontagem, tais como, o perfil, zona e a imagem do perfil para que o operador consiga identificar melhor o cesto ou caixa que é necessário recontar (Figura 32). Também é importante referir que após o utilizador visualizar a notificação, fica disponível um botão que permite ao utilizador arquivar a notificação.

Quanto à segunda funcionalidade, permite ao utilizador consultar o estado dos inventários disponibilizados pela aplicação e onde o utilizador pode alterar a empresa ou a zona ao clicar no botão “Mudar Empresa/Zona”, ou seja, quando o utilizador clica neste botão faz um *reset* às variáveis empresa e zona e permite escolher outra empresa ou zona para inventariar. Esta funcionalidade está acessível através do clique sobre o *avatar* que tem as iniciais do nome do utilizador, conforme a Figura 33.



Figura 32 - Detalhes notificação

7.2.1. INVENTÁRIO DE CESTOS E DE CAIXAS

Na Figura 34, é apresentada a página de “*check-in*” do inventário de cestos, ou seja, é a página onde o operador seleciona a empresa e a zona onde vai realizar o inventário de cestos. Ainda nesta fase de “*check-in*”, assim que o operador seleciona a empresa, a aplicação efetua uma validação para verificar se o inventário de cestos está aberto. Caso o inventário esteja fechado a aplicação emite uma mensagem a informar que o inventário de cesto se encontra encerrado e não permite avançar, caso contrário a aplicação habilita uma nova caixa de texto para procurar pelo cesto que deseja inventariar, como se pode verificar na Figura 35.

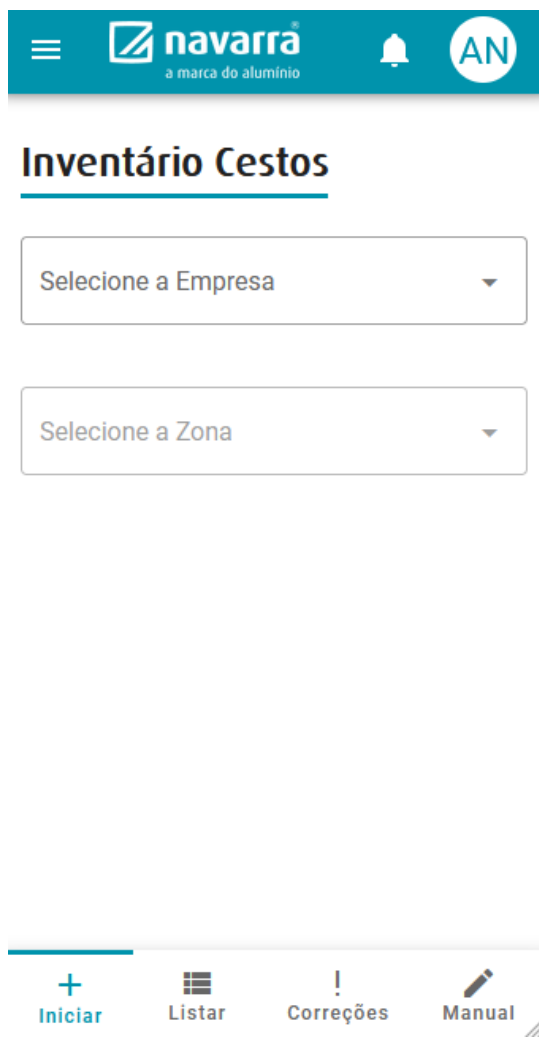


Figura 34 - Página de iniciar inventário de cestos



Figura 35 - Página de inventário de cesto

Após o operador introduzir o número de cesto e clicar em pesquisar, é feito um pedido à *Web API* do ERP da empresa, para verificar se o cesto se encontra disponível para inventariar. Caso esteja disponível, surgem algumas informações do material que se encontra dentro do cesto e um botão que permite ao operador introduzir os dados da contagem que está a realizar no momento (Figura 36). Após introduzir

a medida, o tratamento, as quantidades de barras e clicar em guardar esses dados da contagem são automaticamente guardados no ERP e caso existam diferenças entre os dados que estão no ERP e os dados que o operador está a fornecer também é registada uma diferença de contagem na base de dados da aplicação para posteriormente ser analisada pelo gestor de inventário.

Como podemos verificar na Figura 35, também existe um botão com o nome “Localização”, que tem como objetivo fornecer uma funcionalidade que auxilia os operadores a localizar um cesto. Esta funcionalidade é fundamental quando os operadores são chamados a realizar as recontagens de cestos. As informações apresentadas aos operadores estão refletidas na Figura 37.

The image contains two screenshots of a mobile application interface. The left screenshot, labeled 'Figura 36 - Introdução dos dados da contagem', shows a form for entering basket data. It has a teal header with a back arrow, 'Cesto N.º 003', and a plus icon. Below the header is a dropdown menu for 'Perfil' with the value 'NY659001'. In the center is a circular icon of a basket. Below the icon are two input fields: 'Medida (metros)' with a range selector and 'Tratamento' with a dropdown arrow. At the bottom are two buttons: 'FECHAR' in red and 'GUARDAR' in green. The right screenshot, labeled 'Figura 37 - Consulta da localização do cesto', shows the location information for a basket. It has a teal header with a back arrow, 'Cesto N.º 003', and a plus icon. Below the header are two tabs: 'LOCALIZAÇÃO' (underlined) and 'ÚLTIMO MOV.'. Below the tabs is the location information: 'Máquina CPE001 - Centro Protetivo Estirável', 'Portão P19A', and 'Armazém Embalagem Navarra I'.

Figura 36 - Introdução dos dados da contagem

Figura 37 - Consulta da localização do cesto

Importante referir que o processo de inventário de caixa em tudo é semelhante ao de cestos.

Na Figura 38, é possível visualizar a informação relativa aos cestos inventariados pelo utilizador autenticado na aplicação. Os utilizadores com o perfil de gestor de inventário podem nesta página anular o inventário de um cesto. Sendo também possível procurar por um registo específico e filtrar entre inventário aberto ou fechado.

	Hora	Cesto	Perfil	Barras Recontadas	Zona	Inventariado Por	Ordem	
— data: 2022-03-31 ✕								
✓ ✎	16:53	24A	NY659005	25	RT	admin	3	✕
✓ ✎	16:53	24A	NY659011	48	RT	admin	3	✕
✓	16:47	002	NX855025	25	RT	admin	2	
— data: 2022-03-08 ✕								
✓	16:47	001	NY659087	55	RT	admin	1	

Figura 38 - Lista de cestos inventariados

Na página de correções de cestos ou diferenças de contagem (Figura 39), é onde o gestor de inventário pode tratar as diferenças de contagem, ou seja, o gestor de inventário nesta página pode verificar se a diferença de contagem está nas quantidades, medida ou tratamento de superfície, também pode notificar operadores para fazer recontagens de cestos e pode dar uma resolução para a diferença de contagem.

Correções Cestos

Search Estado do Inventário: **Aberto** Estado da Resolução: **Registado** Estado do Arquivo: **Não Arquiva...** (EXCEL)

Hora	Cesto	Encomenda	Perfil	Registado Por	Zona	Ordem
— registado_em_data: 2022-03-31 ✕						
16:47	002	10889	NX855025	admin	RT	2
— registado_em_data: 2022-03-08 ✕						
16:47	001	1073	NY659087	admin	RT	1

1

Iniciar | Listar | **Correções** | Manual

Figura 39 - Lista de cestos com diferenças de contagem

É importante referir que o inventário de caixas em tudo se assemelha ao inventário de cestos, daí a estratégia de só demonstrar um dos casos para não repetir informação.

7.2.2. INVENTÁRIO DE RETORNÁVEIS

Na Figura 40, está representada a página onde se pode efetuar o inventário de retornáveis. Em primeiro está o campo do armazém, que representa o local onde se vai fazer o inventário de retornáveis. Ficou definido pelo “cliente”, que após o utilizador selecionar o armazém, o valor é afixado para que o utilizador inventarie vários cestos para o mesmo armazém, sem ter que estar constantemente a escolher o armazém. Quando o utilizador pretender mudar de armazém, clica sobre o seu avatar (Ex: DO) e surgirá lá uma secção chamada “Estado do Inventário de Retornáveis” e um botão com o nome “Mudar de Armazém”, conforme já mostrado anteriormente na Figura 33, que permite fazer *reset* ao campo armazém.

navarra
a marca do alumínio

Inventário Retornáveis

Armazém
Expedição Navarra I

O inventário encontra-se aberto

Cesto com número?

Número do cesto

Obrigatório.

Tipo de Cesto

MUDAR TIPO CESTO

Com código de barras?

Qtd. Chapas

Iniciar Listar

Figura 40 - Inventário de retornáveis

Ainda no que diz respeito ao inventário de retornáveis, no protótipo também é disponibilizado ao utilizador uma lista para que este possa controlar os cestos que está a inventariar e eventualmente eliminar alguma linha de inventário que fez por engano. Nesta lista representada na Figura 41 podemos verificar que

também é possível exportar para Excel todas as linhas dos inventários de retornáveis que se encontram em aberto, mas esta funcionalidade, só está disponível para o gestor de inventário.

Cesto		Desc. Armazém			Tipo Cesto		Desc. Tipo Cesto				
Hora	N.º Inventário	Armazém	Desc. Armazém	Com N.º?	Cesto	Tipo Cesto	Desc. Tipo Cesto	Com Cod. Barras?	Qtd. Chapas	Danificado?	Inventariado Por
16:56	60	00191	Expedio Navarra I	✓	002	AC.01.RT.0004	Cestos EXTRUSÃO	✓	2	X	admin
16:56	60	00191	Expedio Navarra I	✓	001	AC.01.RT.0004	Cestos EXTRUSÃO	✓	2	X	admin

Figura 41 - Lista de inventário de retornáveis

7.2.3. ABERTURA E FECHO DE INVENTÁRIOS

Por fim e de forma a dar ao gestor de inventário mais autonomia na gestão dos inventários, foi criada uma interface que lhe permite fazer a gestão dos inventários, nomeadamente abrir e fechar os inventários de caixas, cestos e dos armazéns de retornáveis. Como podemos analisar na Figura 42, é onde o gestor pode escolher qual o tipo de inventário que deseja abrir, e caso escolha retornáveis, surgirá o campo armazém que permite que o utilizador escolha vários armazéns que deseja abrir inventário. Já na Figura 43 está representada a lista de inventários que estão em aberto e caso o utilizador queira fechar algum deles, basta clicar no botão “Fechar”, disponibilizado para cada uma das linhas.

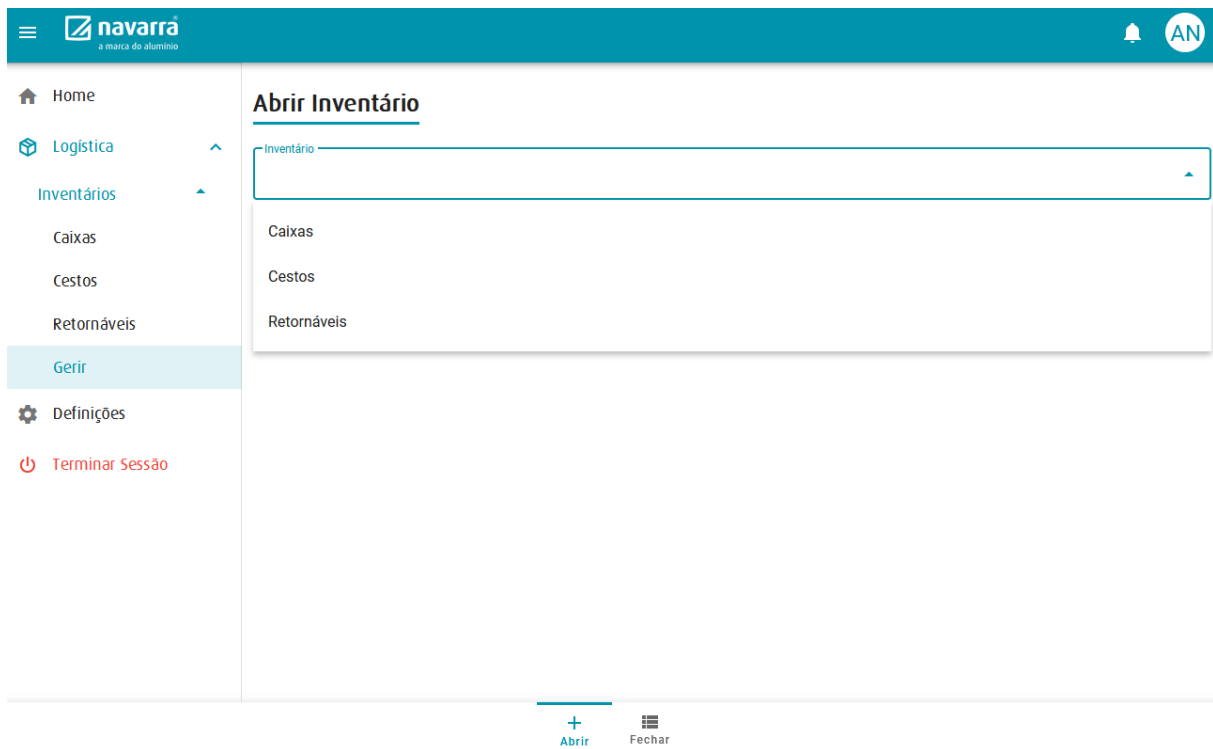


Figura 42 - Abertura de inventários

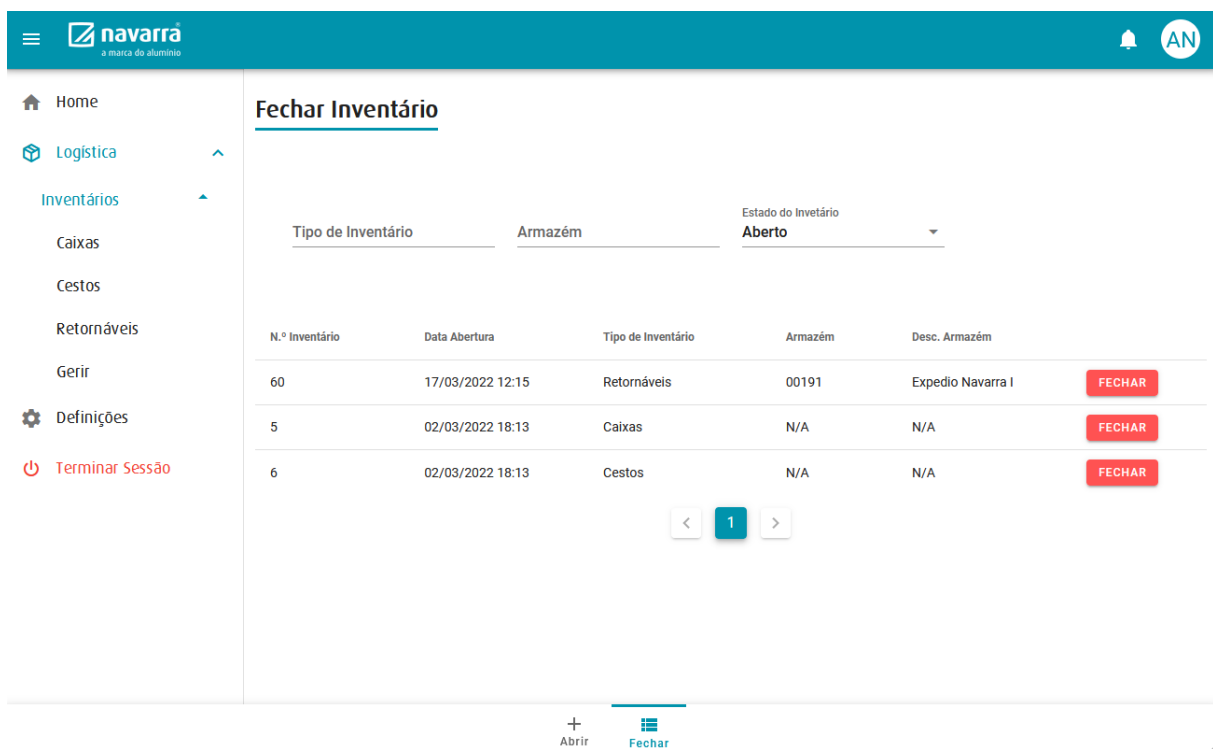


Figura 43 - Fecho de inventários

7.3. DATA STORAGE

Relativamente à camada de *data storage*, esta foi criada com a tecnologia de base de dados em *MySQL*. A base de dados criada em *MySQL* suporta o registo de todos os dados do protótipo. Na Figura 18 podemos consultar o modelo de entidades e relacionamentos desenvolvido para suportar os registos efetuados durante o inventário de cestos, caixas e retornáveis.

7.4. CLIENT

O protótipo desenvolvido tem um design responsivo e está otimizado tanto para trabalhar com dispositivos *desktop* como *mobile*. Os utilizadores que estão mais na parte de *backoffice* do inventário utilizam desktops com o sistema operativo *Windows* e consultam maioritariamente página com maior densidade de dados, relacionadas com a gestão e tratamento dos dados do inventário. Já os utilizadores que andam no terreno a fazer as contagens, utilizam dispositivos móveis (Figura 44) com sistema operativo android e utilizam funcionalidade mais operacionais do protótipo. Os dispositivos utilizados por esses utilizadores, têm integrado um leitor de códigos de barras (Figura 45) que serem para fazer a leitura dos códigos de barras que identificam o número da caixa (Figura 47) e o código de barras do cesto que identifica o número do cesto (Figura 46).

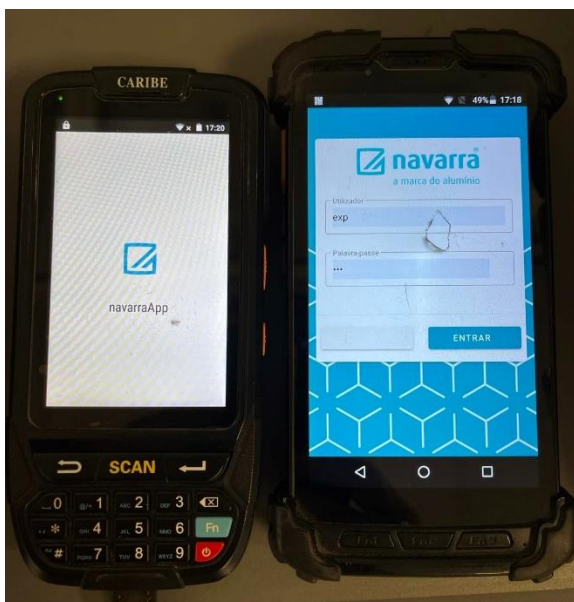


Figura 44 - Dispositivos móveis android



Figura 45 - Leitores de códigos de barras



Figura 46 - Cesto



Figura 47 - Etiqueta de uma caixa

7.5. VALIDAÇÃO DO PROTÓTIPO

Foi realizada uma experiência em produção, onde todos os requisitos expectáveis foram cumpridos. Assim, foram recolhidos alguns testemunhos relativamente à experiência da utilização do protótipo e as mudanças verificadas no processo de inventário de perfis de alumínio e retornáveis por parte de trabalhadores da empresa.

Relativamente às melhorias que o protótipo trouxe para os utilizadores do inventário, **Tiago Martins (Responsável do Dep. Qualidade)** afirma:

“A app desenvolvida pelo Daniel para lançamento do inventário de perfis permitiu um enorme ganho nesta tarefa, eliminando aquilo que na prática era um retrabalho, obrigando a lançamentos manuais em blocos de notas e a um posterior lançamento informático dos dados obtidos no terreno. Desta forma, os colaboradores num simples telemóvel, conseguem, de forma autónoma, realizar o lançamento de dados na aplicação desenvolvida e termos no imediato um output de OK, ou indicação de recontagem. Uma tarefa que, embora simples, se traduzia em inúmeros passos e perdas de tempo, foi assim realizada em meros segundos. Agilizou tremendamente o processo de inventário de perfis em Navarra.”

Já **Filipe Ribeiro (Responsável do Dep. Planeamento e Logística)** aponta que:

O inventário anual era feito com recurso a caneta e pequenos livros preparados para o efeito, onde os colaboradores escreviam, além de outras informações, a referência dos perfis, o cesto onde este estava depositado e a localização do cesto no chão de fábrica.

Após completar cada um desses livros, os mesmos eram levados até ao gabinete onde estavam montados os postos informáticos, onde diversos colaboradores inseriam folha a folha, o inventário no sistema informático, tornando o processo muito lento.

Diversas vezes nos deparávamos com dificuldade em entender a caligrafia dos colaboradores carecendo de nova verificação da informação escrita nos boletins, a não inclusão de toda a informação pedida, levando a nova verificação no chão de fábrica dos cestos indicados, a perda de livros de inventário, esquecimento de lançamento de alguns livros de inventário pela equipa de lançamento no sistema informático.

Com o desenvolvimento desta nova ferramenta, conseguimos extrair informação muito mais rapidamente para validação de dados, uma localização mais precisa de onde se fez o levantamento do inventário, diminuindo o tempo despendido para encontrar novamente o cesto para uma eventual recontagem, eliminamos o fator “caligrafia”, diminuímos o número de verificações necessárias no terreno, reduzimos ao número de colaboradores necessários para esta tarefa e conseguimos reduzir o número de horas necessárias para dar por concluído o inventário.

A **Sandra Caldelas (Responsável do Dep. Controlo de Gestão e Melhoria Contínua e Gestora de Inventário)**, afirma que:

A digitalização dos inventários na Navarra reduziu drasticamente o nº de pessoas envolvidas nos mesmos e permitiu direccionar o foco para uma análise detalhada dos resultados ainda no decorrer do inventário.

Sem a digitalização, tínhamos várias pessoas a efetuar tarefas de desperdício, como o registo em sistema de todas as contagens efetuadas. Registo este que era também suscetível a erros de digitação. Assim existiam vários controlos ao longo do inventário para garantir a fiabilidade do mesmo.

Neste último inventário, passou a ser possível perceber logo após a primeira contagem quais as necessidades de recontagem e de imediato começar a analisar as diferenças existentes. O foco deixou de ser a garantia da fiabilidade da informação para a análise da mesma.

A informação passou a estar disponível no imediato o que simplificou substancialmente todo o processo de inventário.

Em 2021 a Navarra - Extrusão de Alumínio, S.A. conseguiu reduzir o número de colaboradores intervenientes no inventário de 125 para 76 colaboradores, o que correspondeu a uma diminuição de 49 colaboradores face ao período anterior.

A duração do inventário apresentou também um decréscimo, conseguindo-se concluir a fase de contagens e recontagens em 1 dia, ao invés dos anteriores 1,5 dias. Esta redução no número de pessoas envolvidas, bem como no número de horas necessárias para a realização do mesmo, permitiu à organização diminuir drasticamente os custos associados com recursos humanos, ou seja, houve uma redução de 912 horas/homem para 392 horas/homem.

Em termos de custos com economato, em 2021 com a introdução da aplicação verificou-se uma poupança de aproximadamente 400€ por ano, uma vez que deixou de existir a necessidade de adquirir os blocos de etiquetas autocopiativas que se utilizavam anteriormente.

Importa também ressaltar, que a Navarra não teve necessidade de investir em equipamentos tecnológicos para utilização da aplicação durante o inventário, uma vez que todo o material necessário, ou seja, os 53 terminais necessários já existiam na empresa.

A utilização da aplicação permitiu diversas melhorias, em destaque as seguintes:

- Diminuir o número de tarefas de caráter administrativo, relativas ao registo das contagens e recontagens no sistema;
- Garantir o registo de toda a informação necessária nas contagens, através dos campos obrigatórios, simplificando também a formação necessária a todos os envolvidos;
- Garantir a contagem apenas de material Navarra (informando de imediato material já faturado a clientes, por exemplo);
- Aumentar a velocidade da própria contagem de material em caixas, através da leitura do código de barras presente na mesma e da validação simples de dados;
- Associar imagens aos registos dos perfis inventariados, permitindo de imediato perceber se existe alguma troca de artigos;
- Eliminar erros referentes ao registo das contagens escritas manualmente, por caligrafias diferentes, que geravam dúvida e proporcionavam muitas das vezes erros de digitação no sistema;

- Acompanhar “ao momento” as contagens e recontagens efetuadas para uma maior rapidez e fiabilidade na análise e tratamento dos dados;
- No caso do inventário de retornáveis, a introdução da aplicação simplificou largamente os registos efetuados em papel e no sistema, identificando erros de digitação e permitindo detetar no momento da contagem (em diversas localizações do país), a existência de potenciais códigos duplicados, códigos deveras importantes para o controlo de todo o processo produtivo da Navarra.

Perante as afirmações e as melhorias que este protótipo trouxe para a organização e utilizadores do inventário, o software desenvolvido passou de um protótipo para um produto final que foi capaz de gerir o inventário de perfis de alumínio e retornáveis e garantiu à empresa um processo mais ágil, com menos erros e com uma maior monitorização, pelo que passou a ser uma solução integrada na produção que será utilizada nos inventários da empresa.

8. CONCLUSÕES

Este trabalho de dissertação apresentou vários desafios, pois envolveu muito trabalho de investigação, conceção e desenvolvimento de software para garantir que, após a primeira utilização do protótipo durante um processo de inventário, o mesmo correspondesse às necessidades da organização.

Assim, durante a revisão do estado da arte e recorrendo à metodologia *Design Science Research*, foi realizada uma análise crítica de 15 artigos científicos que abordaram temas relacionados com arquiteturas e sistemas de gestão de inventários.

Através do estado da arte foi possível compreender melhor quais as tecnologias e abordagens mais utilizadas para esta problemática, como por exemplo, a utilização de códigos de barras ou etiquetas RFID para identificar os produtos, e no que diz respeito ao desenvolvimento do software também foi possível compreender melhor quais as arquiteturas e as metodologias utilizadas para desenvolver softwares de inventário.

Assim, metodologia Scrum Solo foi a escolhida para ajudar a suportar todo o processo de desenvolvimento de software, sendo que a implementação desta metodologia permitiu adotar uma perspetiva melhor das funcionalidades a implementar, já que permitiu sequenciar e priorizar as tarefas de desenvolvimento do software.

Também foi construída uma arquitetura de software *backend/frontend* que permitiu acelerar o desenvolvimento e melhorar a manutenção e a escalabilidade do produto.

Felizmente, após a primeira utilização do protótipo durante um processo de inventário, a principal conclusão a retirar, foi o fato da solução desenvolvida ter correspondido às necessidades da empresa como demonstrado em detalhe no tópico 7.5 Validação do protótipo, pelo que o software desenvolvido passou de um protótipo para um produto final que será utilizado pela empresa em todos os seus inventários de perfis de alumínio e retornáveis.

As principais vantagens apontadas estiveram relacionadas com a eliminação do papel que além de tornar o processo mais ágil também permitiu eliminar a dificuldade relacionada com a compreensão de algumas caligrafias. Outra vantagem bastante destacada foi a diminuição da duração do processo de inventário que como consequência também permitiu um acesso mais rápido à informação relativa aos perfis de alumínio e retornáveis inventariados.

Relativamente aos números, em 2021 a Navarra - Extrusão de Alumínio, S.A. conseguiu reduzir o número de colaboradores intervenientes no inventário para cerca de metade face ao período anterior.

No que diz respeito à duração do inventário e aos custos horas/homem, houve uma redução de cerca de 50% face ao último inventário.

Em termos de redução de custos, em 2021 com a introdução da aplicação verificou-se uma poupança de aproximadamente 400€ por ano, uma vez que deixou de existir a necessidade de adquirir os blocos de etiquetas autocopiativas que se utilizavam anteriormente e não houve a necessidade de investir em equipamentos tecnológicos para utilização da aplicação durante o inventário, uma vez que todos os terminais necessários já existiam na empresa.

Como sugestão de trabalhos futuros, prevê-se o desenvolvimento de um módulo mais robusto para a gestão de acessos e utilizadores, que será acrescentado ao protótipo.

Por fim, por forma a aprimorar continuamente a arquitetura de gestão de inventário desenvolvida, será fundamental acompanhar a evolução tecnológica, em especial dos dispositivos de RFID, para que o software desenvolvido possa continuar a fornecer à empresa informações mais atualizadas e precisas sobre o inventário atual, e assim ser um suporte importante para a tomada de decisões.

BIBLIOGRAFIA

- Akkermans, H., Bogerd, P., Yücesan, E., & Wassenhove, L. (2003). The impact of ERP on supply chain management: Exploratory findings from a European Delphi study. *European Journal of Operational Research*, *146*, 284–301.
- Arsan, T., Başkan, E., Ar, E., & Bozkuş, Z. (2013). A software architecture for inventory management system. *Lecture Notes in Electrical Engineering*, *152 LNEE*, 15–27. https://doi.org/10.1007/978-1-4614-3535-8_2
- Atieh, A. M., Kaylani, H., Al-Abdallat, Y., Qaderi, A., Ghoul, L., Jaradat, L., & Hdairis, I. (2016). Performance Improvement of Inventory Management System Processes by an Automated Warehouse Management System. *Procedia CIRP*, *41*, 568–572. <https://doi.org/10.1016/j.procir.2015.12.122>
- Bourque, P., & Fairley, R. E. (2014). *Guide to the Software Engineering Body of Knowledge SWEBOK® A Project of the IEEE Computer Society: Vol. Version 3.0*. IEEE Computer Society. <http://www.swebok.org/>
- Brito, J. N., Rebelo, C., & Brito, M. A. (2020). *Scrum Solo Application in a Project with a Strong Integration Component*.
- Chaniotis, I. K., Kyriakou, K. I. D., & Tselikas, N. D. (2015). Is Node.js a viable option for building modern web applications? A performance evaluation study. *Computing*, *97*(10), 1023–1044. <https://doi.org/10.1007/s00607-014-0394-9>
- Chen, H.-M., & Chang, C.-S. (2009). IIMS: an integrated inventory management system based on software agent. In *International Journal of System Science and the European Journal of Operational Research* (Vol. 4, Issue 1).
- Cheng, R.-S., Lin, C.-P., Lin, K. W., & Hong, W.-J. (2015). NFC Based Equipment Management Inventory System. In *Journal of Information Hiding and Multimedia Signal Processing c* (Vol. 6, Issue 6).
- Gutiérrez, V., & Jaramillo, D. P. (2009). Review of the software available for inventory management in supply chains in Colombia. *Estudios Gerenciales*, *25*(110), 125–153. [https://doi.org/10.1016/S0123-5923\(09\)70065-8](https://doi.org/10.1016/S0123-5923(09)70065-8)

- Laskey, K. B., & Laskey, K. (2009). Service oriented architecture. *Wiley Interdisciplinary Reviews: Computational Statistics*, 1(1), 101–105. <https://doi.org/10.1002/wics.8>
- Lei, K., Ma, Y., & Tan, Z. (2014). Performance Comparison and Evaluation of Web Development Technologies in PHP, Python and Node.js. *International Conference on Computational Science and Engineering*.
- Letkowski, J. (2002). Doing database design with MySQL. *Journal of Technology Research*.
- Manthou, V., & Vlachopoulou, M. (2001). *Bar-code technology for inventory and marketing management systems: A model for its development and implementation*. [https://doi.org/10.1016/S0925-5273\(00\)00115-8](https://doi.org/10.1016/S0925-5273(00)00115-8)
- Mathaba, S., Adigun, M., Oladosu, J., & Oki, O. (2017). On the use of the Internet of Things and Web 2.0 in inventory management. *Journal of Intelligent and Fuzzy Systems*, 32(4), 3091–3101. <https://doi.org/10.3233/JIFS-169252>
- Pagotto, T., Fabri, J. A., Lerario, A., & Goncalves, J. A. (2016). Scrum Solo: Processo de software para desenvolvimento individual. *Iberian Conference on Information Systems and Technologies, CISTI, 2016-July*. <https://doi.org/10.1109/CISTI.2016.7521555>
- Peffer, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of Management Information Systems*, 24(3), 45–77. <https://doi.org/10.2753/MIS0742-1222240302>
- Santillán, L., Ginestà, M., & Mora, Ó. (2007). *Bases de datos en MySQL*.
- Setemen, K., Sudirtha, I. G., Marsiti, C. I. R., Dantes, G. R., & Suputra, P. H. (2020). Developing inventory information system using mobile computing with quick response (2d-barcode) and geotagging. *Journal of Physics: Conference Series*, 1516(1). <https://doi.org/10.1088/1742-6596/1516/1/012011>
- Sprott, D., & Wilkes, L. (2004). *Understanding Service-Oriented Architecture*.
- Studiengang Bachelor, im, Prüferin, B., Steeens Zweitgutachter, U., Behrmann geb Knoblauch, M., & Wohlgethan, E. (2018). *Bachelorarbeit eingereicht im Rahmen der Bachelorprüfung*.

- Tejesh, B. S. S., & Neeraja, S. (2018). Warehouse inventory management system using IoT and open source framework. *Alexandria Engineering Journal*, 57(4), 3817–3823. <https://doi.org/10.1016/j.aej.2018.02.003>
- Trunk Christopher. (1994). Using bar codes for warehouse control. *Material Handling Engineering*, 49(10), 48–52.
- Wartha N, & Londhe Vaishali. (2015). Context-Aware Approach for enhancing security and privacy of RFID. *International Journal of Engineering And Computer Science*, 4(1).

ANEXOS

ATAS DE REUNIÃO



Ata de Reunião

Presentes: Daniel Oliveira, Sandra Caldelas, Carlos Teixeira

Data: 07/09/2021

Assunto: Solução informática para inventário de caixas, cestos e retornáveis

Nesta reunião foi transmitida a necessidade de a equipa de informática desenvolver um software de gestão de inventário que fosse de encontro às necessidades abaixo relatadas:

- distinguir material embalado na fábrica de Navarra de material de Navarra II (PS);
- permitir colocar uma quantidade diferente da existente ao picar a caixa (podem existir caixas com quantidades incorretas);
- permitir inserir caixas sem rótulo (podem existir caixas sem rótulo ou rótulo manual);
- deve permitir o registo de caixas supostamente duplicadas (se as caixas estiverem afastadas, se for efetuada por outro terminal ou passado algumas caixas/tempo)(podem imprimir dois rótulos);
- deve permitir o registo de uma caixa que para o sistema já está faturada (caixa faturada não enviada)
- deve haver um registo pela ordem de picagem que é efetuada, por terminal (permite procurar as caixas mais facilmente na recontagem)
- Deve existir uma listagem intermédia que permita a análise das diferenças, assim como efetuar algumas correções (nomeadamente uma caixa duplicada que se trata no fundo de uma impressão duplicada de um rótulo e não de duplicação das quantidades, corrigir quantidades, corrigir nº de caixa); apresenta todas as diferenças: caixas em Libra não contadas/picadas, caixas contadas que não estavam em Libra, diferenças de quantidade, caixas duplicadas, trocas de referência, etc;
- A listagem após análise/correção das diferenças finais deve integrar no Libra automaticamente;
- Se possível, inclusive, eliminar caixas inexistentes, corrigir quantidades e "imprimir" rótulos novos;
- > Deixar de se utilizar os blocos de caixas;
- > Evitar erros de digitação;
- > Evitar cópia de valores (e erros) dos blocos para o Libra;

Sandra Caldelas

07/09/2021

Figura 48 - Ata de Reunião

DIAGRAMA DE CLASSES

A representa o diagrama de classes que mapeia a estrutura conceptual de alto nível do protótipo, nomeadamente no que diz respeito às classes, os seus atributos, operações e relações entre elas.

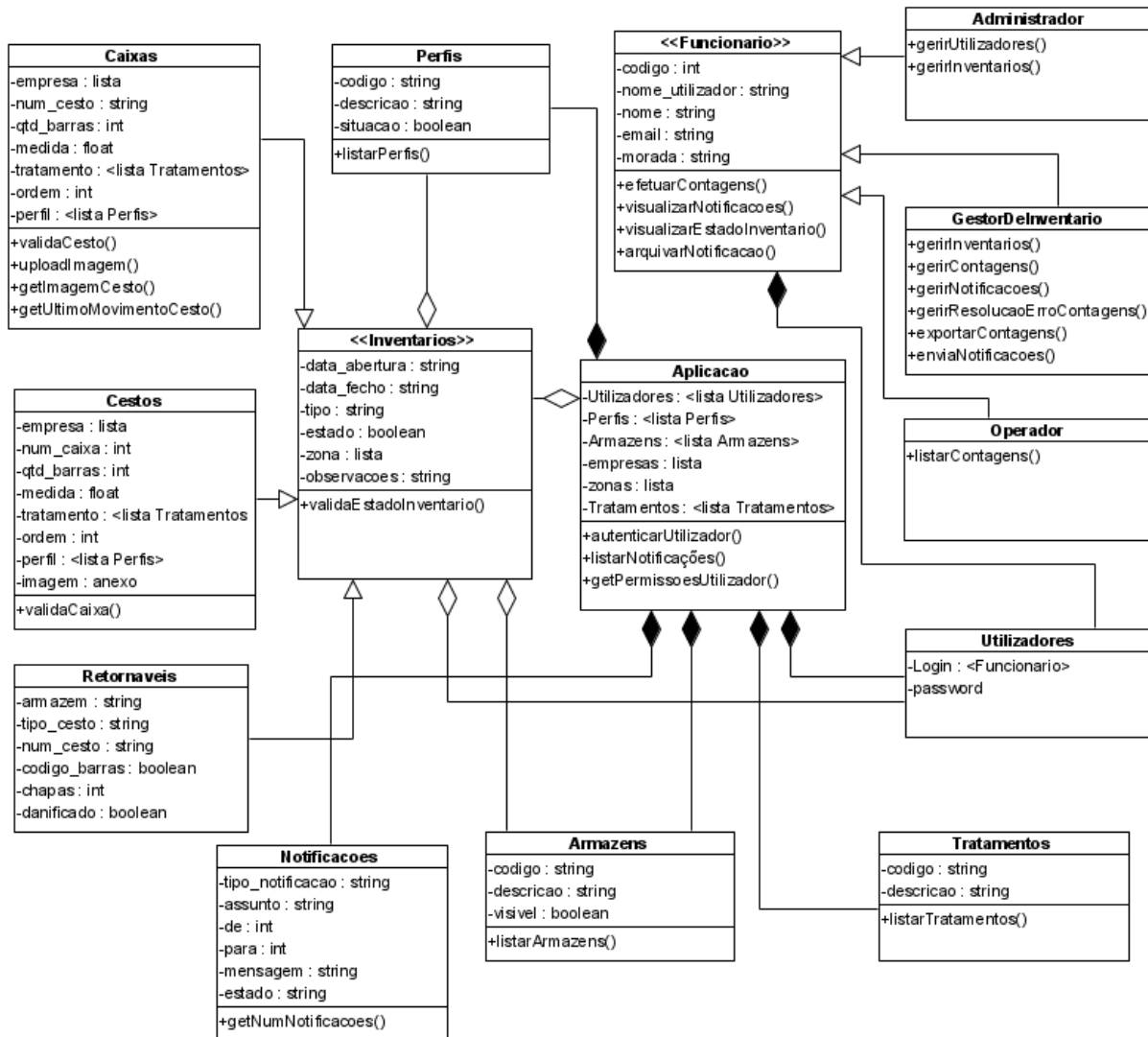


Figura 49 - Diagrama de Classes