

REM

Gestão de Resíduos Sólidos da Estação de Triagem do Funchal

RELATÓRIO DE ESTÁGIO DE MESTRADO

Sérgio Miguel Rodrigues Viúla
MESTRADO EM ENGENHARIA INFORMÁTICA



UNIVERSIDADE da MADEIRA

A Nossa Universidade

www.uma.pt

setembro | 2016

Gestão de Resíduos Sólidos da Estação de Triagem do Funchal

RELATÓRIO DE ESTÁGIO DE MESTRADO

Sérgio Miguel Rodrigues Viúla

MESTRADO EM ENGENHARIA INFORMÁTICA

ORIENTADOR

José Luís Cardoso da Silva

CO-ORIENTADOR

João Miguel Figueira Gomes

Resumo

A imersão tecnológica no nosso quotidiano tem vindo a aumentar significativamente e com o surgimento da internet cada vez mais opta-se por plataformas digitais e aplicações com informação distribuída e acessível a vários utilizadores em simultâneo.

Este documento detalha o trabalho realizado no âmbito do projeto de controlo de movimentos de resíduos sólidos e urbanos da Estação de Transferência e Triagem de Resíduos Sólidos do Funchal.

O projeto surge da necessidade da Estação de Transferência e Triagem obter uma plataforma Web interna ao serviço que interligada com a aplicação de registo de entradas/saídas de viaturas, consiga gerir, consultar e analisar dados estatísticos sobre os registos da estação.

O objetivo do estágio curricular visa a implementação de uma plataforma web interna, que cumpra com as necessidades da Estação de Transferência e Triagem de Resíduos Sólidos do Funchal tal como a importação de informação sensível existente (como por exemplo, informação dos fornecedores). Além das funcionalidades desejadas, foi também adicionado um mecanismo de registo de alterações aos dados existentes na plataforma de modo a detetar possíveis alterações indesejadas, tal como manter um registo histórico dos dados.

Para além dos testes efetuados ao longo do desenvolvimento, realizou-se também na fase final testes à plataforma por parte dos utilizadores da estação, de forma a detetar possíveis erros de usabilidade assim como possíveis anomalias com a plataforma.

No término do estágio, a plataforma foi distribuída aos utilizadores da estação cumprido com as funcionalidades desejadas, tendo esta sido aceite pelo Departamento de Ambiente da CMF.

Palavras-chave

Plataforma Web

Resíduos sólidos e urbanos

Câmara Municipal do Funchal

Rastreamento de Operações

Abstract

Technological presence in our daily lives has increased significantly and with the emergence of the internet the need for digital platforms and applications with distributed information and accessible to multiple users simultaneously has increased.

This document details the process of deploying the project about the control of solid and urban waste in *Estação de Transferência e Triagem de Resíduos Sólidos* of Funchal.

This project arises of the necessity from the *Estação de Transferência e Triagem de Resíduos Sólidos* of Funchal acquiring an internal Web platform that connected with the software responsible of registering vehicles movements, can manage, query and analyze statistical data associated with the station flow.

The goal of the curricular internship aims to develop an internal web platform that meets the needs from the *Estação de Transferência e Triagem de Resíduos Sólidos do Funchal* as well as importing existing sensitive data (such as suppliers). In addition to the desired functionalities, an audit trail mechanism was also implemented in the platform to detect possible unwanted changes on the data and to keep an historical record of all the changes made.

Besides the tests conducted during the development stage, in the final phase of the project tests were conducted in the platform by the users of the station in order to detect possible usability problems as well as possible anomalies in the platform.

By the end of the internship, the web platform was deployed to the users of the stations fulfilling therefore their required needs.

Keywords

Web Platform

Solid and urban waste

Funchal City Hall

Audit Trail

Agradecimentos

Dedico esta seção para agradecer a todos que ajudaram e acompanharam o meu percurso acadêmico.

Em primeiro lugar gostaria de agradecer ao Professor José Luís Silva e ao Dr. João Gomes por todo o apoio, disponibilidade e orientação durante o estágio.

Gostaria de agradecer também ao Departamento de Sistemas de Informação da Câmara Municipal do Funchal e em especial ao Dr. Cesar Rosa chefe do departamento, pelo acolhimento, apoio e condicionarem um excelente ambiente de trabalho.

Agradeço aos meus pais e família, que sem eles esta minha etapa de vida nunca teria sido possível alcançar, com o apoio e paciência que tiveram comigo em todo o meu percurso acadêmico. Agradeço também à minha namorada pela eterna paciência e compreensão nesta minha fase de vida.

Um especial obrigado aos meus colegas Lino Oliveira, Jhair Abreu, Miguel Ribeiro, Mário Ribeiro e Lino Teixeira pela ajuda, opinião e tempo disponibilizado no realizar do projeto.

Índice

Capítulo 1	1
Introdução	1
1.1. Âmbito do estágio	1
1.2. Entidade de acolhimento	2
1.3. Projeto Resíduos Sólidos e Urbanos.....	2
1.3.1. Objetivos principais.....	4
1.4. Estrutura do relatório.....	5
Capítulo 2	7
Análise do funcionamento da estação	7
2.1. Entradas e saídas de viaturas	7
2.2. Registo de movimentos da estação	10
2.2.1. Funcionamento da balança	10
2.2.2. Software da balança.....	11
2.3. Estatísticas.....	12
2.4. Conclusão	14
Capítulo 3	15
Especificação	15
3.1. Stakeholders.....	15
3.1.1. Identificação	15
3.1.2. Perfis.....	17
3.1.3. Permissões de acesso.....	19
3.2. Requisitos Funcionais	19
3.2.1. Casos de uso essenciais.....	20
3.2.1.1. Visualizar Estatísticas	22
3.2.1.2. Gerir Registo de Movimentos	24
3.2.1.3. Gerir Veículos	26
3.2.1.4. Gerir Fornecedores	28
3.2.1.5. Gerir Resíduos	30
3.2.1.6. Gerir Serviços	32
3.2.1.7. Gerir Motoristas	34
3.2.1.8. Gerir Utilizadores	36
3.2.1.9. Gerir Circuitos.....	39
3.2.1.10. Visualizar registo de atividade do sistema.....	41

3.2.2. Resumo de Funcionalidades.....	42
3.3. Requisitos Não-funcionais.....	44
3.4. Restrições tecnológicas.....	45
3.5. Conclusão.....	45
Capítulo 4.....	47
Revisão tecnológica.....	47
4.1. Tecnologias server-side.....	47
4.1.1. JAVA.....	47
4.1.2. Javascript.....	47
4.1.3. PHP.....	48
4.1.3.1. Frameworks PHP.....	49
4.2. Bases de dados.....	51
4.2.1. Relacionais.....	51
4.2.2. Não-Relacionais.....	51
4.2.3. Relacionais vs. Não-Relacionais.....	52
4.3. Conclusão.....	52
Capítulo 5.....	55
Desenho e Implementação.....	55
5.1. Planeamento.....	55
5.2. Modelo de dados.....	56
5.2.1. Diagrama ER - Aplicação.....	57
5.2.2. Descrição das tabelas.....	58
5.3. Plataforma Web.....	59
5.3.1. Arquitetura.....	59
5.3.1.1. Modelo.....	60
5.3.1.2. Controlador.....	62
5.3.1.3. Vista.....	62
5.3.2. Plugins e Extras.....	63
5.3.2.1. Plugins Backend.....	63
5.3.2.2. Plugins Frontend.....	63
5.3.3. Diagrama de componentes.....	65
5.3.4. Aplicação.....	66
5.4. Conclusão.....	71
Capítulo 6.....	73
Testes de usabilidade.....	73
6.1. Método de teste.....	74

6.2. Preparação dos testes	74
6.2.1. Tarefa 1 – crie um fornecedor.....	75
6.2.2. Tarefa 2 – crie um registo de movimento	75
6.2.3. Tarefa 3 - modifique um registo movimento	75
6.2.4. Tarefa 4 – adicione um utilizador à plataforma	75
6.2.5. Tarefa 5 - exporte a tabela de estatísticas do mês atual, fornecedores - resíduos ..	75
6.3. Participantes.....	76
6.4. Resultados	77
6.4.1. Resultado tarefa 1 – crie um fornecedor	77
6.4.2. Resultado tarefa 2 – crie um registo de movimento.....	77
6.4.3. Resultado tarefa 3 – modifique um registo de movimento	78
6.4.4. Resultado tarefa 4 – adicione um utilizador à plataforma.....	78
6.4.5. Resultado tarefa 5 – exporte tabela de estatísticas.....	78
6.5. Conclusão	79
Capítulo 7	81
Conclusão	81
7.1. Trabalho realizado.....	81
7.2. Trabalho futuro	83

Lista de figuras

Figura 1 - Ciclo de recolha de residuos	3
Figura 2 - Fluxo da estação	8
Figura 3 - Entrada e Saída da Estação	9
Figura 4 - Cabine do operador da balança	10
Figura 5 - Software de registo de movimentos	11
Figura 6 - Inconsistência nos dados	12
Figura 7 - Exportação de registos	12
Figura 8 – Excel formatado – dados 1 de janeiro de 2016	13
Figura 9 - Excel formatado – resumo 1 de janeiro de 2016	13
Figura 10 - Diagrama de stakeholders.....	16
Figura 11 - Diagrama Use case. (Nota: Gerir corresponde a operações CRUD).....	20
Figura 12 - Comparação Pedidos por segundo (adaptado de [9])	48
Figura 13 - Linguagens de servidor mais utilizadas (de [9])	49
Figura 14 - Comparação de Frameworks PHP (adaptado de [11])	50
Figura 15 - Sql vs. NoSQL (adaptado de [18])	52
Figura 16 - Planeamento	56
Figura 17 - Diagrama ER - Aplicação.....	57
Figura 18 - Arquitetura MVC	59
Figura 19 - Tabela Veiculo ORM	60
Figura 20 - Doctrine Update	61
Figura 21 - ORM Diagrama	61
Figura 22 - Digrama de Controladores	62
Figura 23 - Diagrama de Componentes.....	65
Figura 24 - Menu de autenticação	66
Figura 25 - Página inicial plataforma web	67
Figura 26 - Adicionar Fornecedor	68
Figura 27 - Listagem de movimentos	68
Figura 28 - Popup informação de registo	69
Figura 29 - Talão de registo	69
Figura 30 - Gráfico resumo mensal	70
Figura 31 - Resumo anual contentor aberto	70
Figura 32 – Gráfico de erros de usabilidade (adaptado de [29])	76

Lista de tabelas

Tabela 1 – Perfil Operador da balança	17
Tabela 2 - Perfil Chefe de Departamento.....	17
Tabela 3 - Perfil Funcionário departamento	18
Tabela 4 - Perfil Developer	18
Tabela 5 - Grupos de utilizadores.....	19
Tabela 6 - Casos de utilização.....	21
Tabela 7 - Fluxo de "Visualizar Estatísticas"	22
Tabela 8 - Slice 1.1 "Pesquisa de Registos"	23
Tabela 9 - Slice 1.2 "Visualizar Estatísticas"	23
Tabela 10 - Slice 1.3 "Exportar como pdf"	23
Tabela 11 - Slice 1.4 "Exportar como xls".....	24
Tabela 12 - Fluxo de "Gerir Registo de Movimentos"	24
Tabela 13 - Slice 2.1 "Pesquisa de Registos"	25
Tabela 14 - Slice 2.2 "Adicionar Registo"	25
Tabela 15 - Slice 2.3 "Modificar Registo"	25
Tabela 16 - Slice 2.4 " Desativar Registo "	26
Tabela 17 - Fluxo "Gerir Veículos"	26
Tabela 18 - Slice 3.1 "Pesquisa de Registos"	27
Tabela 19 - Slice 3.2 "Adicionar Veículo"	27
Tabela 20 - Slice 3.3 "Modificar Registo"	27
Tabela 21 - Slice 3.4 " Desativar Veículo "	28
Tabela 22 - Fluxo de "Gerir Fornecedores"	28
Tabela 23 - Slice 4.1 "Pesquisa de Registos"	29
Tabela 24 - Slice 4.2 "Adicionar Fornecedor"	29
Tabela 25 - Slice 4.3 "Modificar Fornecedor"	29
Tabela 26 - Slice 4.4 " Desativar Veículo "	30
Tabela 27 - Fluxo "Gerir Resíduos"	30
Tabela 28 - Slice 5.1 "Pesquisa de Resíduos"	31
Tabela 29 - Slice 5.2 "Adicionar Resíduo"	31
Tabela 30 - Slice 5.3 "Modificar Resíduo"	31
Tabela 31 - Slice 5.4 " Desativar Resíduo "	32
Tabela 32 - Fluxo "Gerir Serviços"	32
Tabela 33 - Slice 6.1 "Pesquisa de Serviços"	33
Tabela 34 - Slice 6.2 "Adicionar Serviços"	33
Tabela 35 - Slice 6.3 "Modificar Serviço"	33
Tabela 36 - Slice 6.4 " Desativar Serviço "	34
Tabela 37 - Fluxo "Gerir Motoristas"	34
Tabela 38 - Slice 7.1 "Pesquisa de Motoristas"	35
Tabela 39 - Slice 7.2 "Adicionar Motoristas"	35
Tabela 40 - Slice 7.3 "Modificar Motorista"	35
Tabela 41 - Slice 7.4 " Desativar Motorista "	36
Tabela 42 - Fluxo "Gerir Utilizadores"	36
Tabela 43 - Slice 8.1 "Listar Utilizadores".....	37

Tabela 44 - Slice 8.2 "Alterar grupo de utilizador"	37
Tabela 45 - Slice 8.3 "Adicionar novo utilizador"	37
Tabela 46 - Slice 8.4 "Bloquear utilizador"	38
Tabela 47 - Slice 8.5 "Remover conta de utilizador"	38
Tabela 48 - Fluxo "Gerir Circuitos"	39
Tabela 49 - Slice 9.1 "Pesquisa de Circuitos"	39
Tabela 50 - Slice 9.2 "Adicionar Circuitos"	40
Tabela 51 - Slice 9.3 "Modificar Circuito"	40
Tabela 52 - Slice 9.4 " Desativar Circuitos "	40
Tabela 53 - Fluxo "Visualizar registo de atividade do sistema"	41
Tabela 54 - Slice 10.1 "Pesquisa de registo de atividade"	42
Tabela 55 - Slice 10.2 "Visualizar Estatísticas"	42
Tabela 56 - Resumo de Funcionalidades	42
Tabela 57 - Descrição Tabelas ER	58

Abreviaturas e símbolos

ACID – *Atomicity, Consistency, Isolation, Durability*

AJAX – *asynchronous JavaScript and XML*

CMF – Câmara Municipal do Funchal

CRUD – *Create, Read, Update, Delete*

DOM – *Document Object Model*

DSI – Departamento de Sistemas e Informação

ER – Entity Relationship

LCD – *Liquid Crystal Display*

LDAP – *Lightweight Directory Access Protocol*

NoSQL – *Non SQL/Non Relational*

ORM – *Object-Relational Mapping*

PDF – *Portable Document Format*

PHP – PHP Hypertext Preprocessor

POO – Programação Orientada a Objetos

RNF – Requisito Não Funcional

RSU – Resíduos Sólidos e Urbanos

SGBD – Sistemas Gestores de Bases de Dados

SQL – Structured Query Language

UCS – *Use Case Slice*

UML – *Unified Modeling Language*

Capítulo 1

Introdução

Este capítulo aborda o âmbito do estágio e a instituição no qual este se realizou tal como uma breve descrição sobre o projeto e seus objetivos principais.

1.1. Âmbito do estágio

O projeto de estágio realizado na Câmara Municipal do Funchal (CMF) no Departamento de Sistemas e Informação (DSI) tendo como orientador da instituição o Dr. João Miguel Figueira Gomes, enquadra-se no âmbito de Projeto de mestrado pela Universidade da Madeira tendo como orientador o Prof. José Luís Cardoso da Silva. O estágio em questão está subdividido em três fases principais, nomeadamente a de Análise e Modelação do problema seguida das fases de Desenvolvimento e Teste da solução.

O projeto desenvolvido durante o estágio trata-se de uma solução web que permita o controlo e gestão de resíduos que dão entrada e saída na Estação de Transferência e Triagem de Resíduos Sólidos do Funchal, nomeadamente para o Departamento de Ambiente da CMF. O presente projeto foi requisitado pelo Departamento de Ambiente da CMF ao DSI, sendo este realizado na totalidade no DSI, todas as reuniões entre mim e o Departamento de Ambiente foram mediadas pelo DSI.

1.2. Entidade de acolhimento

A entidade de acolhimento do estágio foi a Câmara Municipal do Funchal, situada na Praça do Município, no complexo de edifícios da mesma. Esta é constituída por diversas áreas de competência do município, sendo umas destas o DSI, departamento em que se realizou o estágio.

O projeto desenvolvido destina-se ao Departamento de Ambiente, situado no sítio da Fundoa de Baixo, Freguesia de São Roque. Juntamente ao Departamento de Ambiente encontra-se a Estação de Transferência e Triagem de Resíduos Sólidos do Funchal.

O desenvolvimento do projeto foi realizado inteiramente no DSI, onde se encontra o coorientador do projeto. As instalações do Departamento de Ambiente foram utilizadas para as reuniões com os *stakeholders* do projeto tal como para a fase de análise do processo de funcionamento da Estação de Transferência e Triagem de Resíduos Sólidos.

1.3. Projeto Resíduos Sólidos e Urbanos

Com a era da modernização informática, o Departamento de Ambiente, sentiu necessidade de evolução do processo de registo dos movimentos da estação de triagem para um formato distribuído e acessível a entidades internas da estação. Anteriormente, todo este processo era registado em folhas de cálculo do Microsoft Excel o que tornava o acesso concorrente aos dados atualizados em tempo real uma tarefa difícil de alcançar.

O Departamento de Ambiente necessita de um leque de resumos dos movimentos da estação, sendo este feito manualmente e arduamente à base de diversas folhas de cálculo Excel, tornando este processo propício a erro humano.

O processo de análise de entradas e saídas de veículos contendo resíduos sólidos urbanos (RSU), torna-se extremamente importante para uma limpeza eficaz da cidade, pois com estes dados é possível tomar medidas reforçadas em determinados pontos da cidade. Associado também aos dados descritos, encontra-se uma relação entre reaproveitamento de resíduos e faturação dos mesmos a entidades

externas. A figura 1 esquematiza o ciclo de recolha e processamento de resíduos.

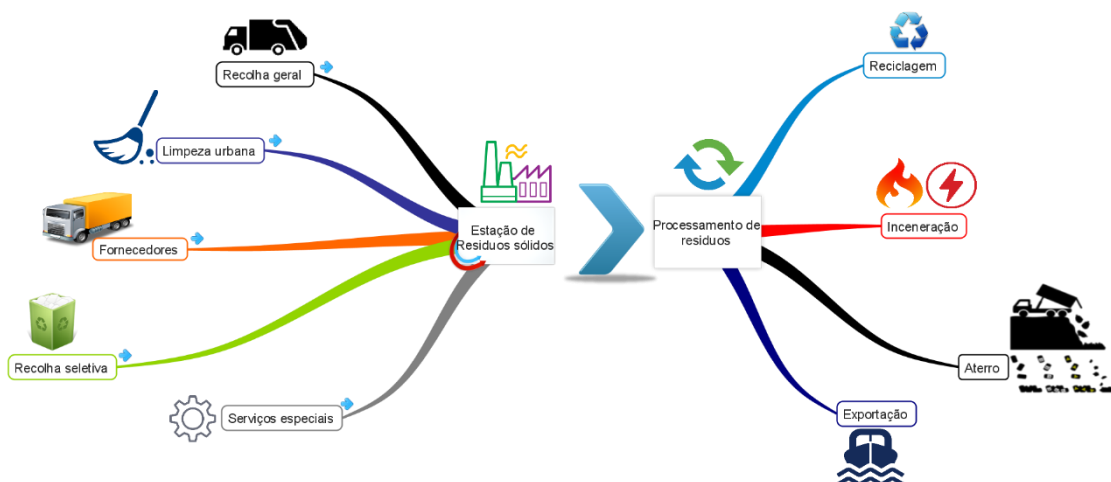


Figura 1 - Ciclo de recolha de resíduos

O processo de aquisição dos dados de movimentos das viaturas é feito através de uma balança de pesagem de veículos ao entrar/sair da estação, associada à balança estava um Software de uma empresa privada, logo o acesso à estrutura do software para possíveis alterações não se encontra acessível. Com o software descrito, o operador da balança preenche os dados do veículo (matrícula, resíduo, etc.) esta informação é então guardada numa base de dados Microsoft Access local, com possível exportação para formato Excel. O Software em questão não garante qualquer tipo de validação não reduzindo possíveis erros humanos no ato de inserção do registo. Ao terminar o dia a folha de cálculo exportada da balança é enviada para a administração do Departamento de Ambiente.

Ao receber o ficheiro Excel, é então feita uma inserção manual dos mesmos dados devidamente formatados numa segunda folha de Excel, com filtros e regras associadas de forma a obter gráficos de resumo, sendo este processo também propício a erro humano.

Com as restrições e problemas previamente descritos, foi então acordado o desenvolvimento de um novo Software para o ato de registo de movimentos por parte do DSI.

1.3.1. Objetivos principais

O objetivo do estágio visa a criação de uma plataforma web para a gestão de todos os movimentos correntes na Estação de Transferência e Triagem de Resíduos Sólidos do Funchal. A solução deverá permitir a gestão de:

- Resíduos: tipo de resíduo sólido que poderá dar entrada/saída na estação;
- Viaturas: identificação de viaturas e que tipo de resíduos estas poderão transportar;
- Fornecedores: entidades que efetuam deslocações de resíduos na estação e que tipo de serviços estas podem efetuar;
- Serviços: tipo de serviço que deu origem à entrada ou saída do resíduo na estação;
- Motoristas;
- Circuitos de remoção de Resíduos;
- Movimentos.

A plataforma terá de validar/restringir a inserção de dados de forma a minimizar ao máximo possíveis erros na inserção de dados.

A plataforma deverá possuir uma componente de administração em que apenas o DSI e os chefes do Departamento de Ambiente, poderão definir a qualquer momento novas restrições, alteração de dados associados a fornecedores, resíduos, motoristas, etc. sem necessidade de proceder a alterações no Software.

O sistema de autenticação terá de ser validado através dos serviços da CMF, tal como gerir níveis de acesso à aplicação através de um mecanismo de permissões.

Em termos estatísticos o Departamento do ambiente pretende obter:

- Resumo de atividade diária;
- Resumo de atividade semanal;
- Resumo de atividade anual;
- Resumo por tipo de movimento;
- Resumo por fornecedor;
- Resumo por tipo de resíduo.

Toda esta informação deverá ser apresentada da forma mais simples e perceptível ao utilizador. Todas as modificações e acessos deverão ser registados, mantendo assim um rastreamento de dados.

1.4. Estrutura do relatório

Este documento é composto por sete capítulos que abordam todo o processo necessário para o desenvolvimento da plataforma web. Descreve-se sucintamente cada capítulo da seguinte forma:

1. Introdução: Aqui é feita uma descrição da entidade acolhimento do estágio em questão tal como os objetivos a alcançar no mesmo;
2. Análise do funcionamento da estação: É feita uma análise completa ao funcionamento da estação desde o processo de pesagem das entradas/saídas de veículos até à obtenção de estatísticas sobre as mesmas;
3. Especificação: São identificados os principais *stakeholders* do projeto, requisitos funcionais, requisitos não-funcionais e restrições tecnológicas. Com base nesta especificação é possível iniciar o desenvolvimento da plataforma;
4. Revisão Tecnológica: É realizado um estudo comparativo entre possíveis tecnologias a utilizar na realização do projeto;

5. Desenho e Implementação: Define o processo de desenho e desenvolvimento da plataforma. Começando pelo planeamento de desenvolvimento segue-se o desenho da base de dados e descrição da implementação;
6. Testes de usabilidade: Descreve-se os testes de usabilidade efetuados tal como os resultados obtidos;
7. Conclusão: Apresenta-se as conclusões acerca do projeto tal como possível trabalho futuro para melhorar o mesmo.

Capítulo 2

Análise do funcionamento da estação

Previamente a qualquer tipo de especificação, foi feita uma análise cuidadosa ao funcionamento da estação, entendendo assim ao seu funcionamento. Este capítulo aborda sucintamente a análise realizada na fase inicial do estágio.

2.1. Entradas e saídas de viaturas

Na fase inicial do estágio, foi realizada uma visita guiada à estação sendo assim possível entender que trabalho se realiza na mesma.

Foi observado o processo de entrada e saída de viaturas. Numa vista inicial o processo demonstrava-se simples até que uma análise aprofundada revelasse problemas associados ao mesmo.

As viaturas, pertencentes à CMF ou a entidades externas, dirigem-se à entrada da estação de modo a serem pesadas na balança. Associada a esta balança estava um Software responsável por registar o movimento de entrada ou saída. Caso se trate de uma entidade singular, do qual a CMF não tem qualquer tipo de informação registada, o registo é anotado numa folha de papel A4, para mais tarde ser introduzido manualmente no Software. Após o veículo se encontrar posicionado na balança, o operador da mesma regista a informação sobre o movimento, como por exemplo, o nome do motorista, matrícula, resíduo, etc.

Após observar o operador da balança, foi detetado que o processo de pesagem era registado de três formas distintas:

1. Pesagem simples: pesagem da qual a CMF conhece à priori a informação do veículo, como por exemplo a sua tara. Se este apenas transportar um único resíduo, é apenas realizada uma única pesagem obtendo assim peso líquido do resíduo transportado;
2. Pesagem dupla: pesagem realizada quando não se conhece à priori a tara do veículo. Sendo então necessárias duas pesagens de forma a obter peso do resíduo transportado. Uma primeira pesagem obtém o peso bruto, i.e., peso do veículo em conjunto com o peso de resíduo. Após o veículo ter descarregado o resíduo é pesado novamente, obtendo assim a sua tara. Com estas duas pesagens é então possível determinar o peso do resíduo transportado;
3. Pesagem de 4: pesagem realizada quando o veículo transporta dois tipos de resíduos, são efetuadas então duas pesagens duplas de forma a obter um valor para cada tipo de resíduo transportado.

Existindo apenas uma balança para todo o fluxo de entrada e saída na estação, é importante que o processo de pesagem se realize o mais rápido possível de forma a não congestionar o portão da estação. A figura 2 ilustra de forma abstrata o fluxo da estação.

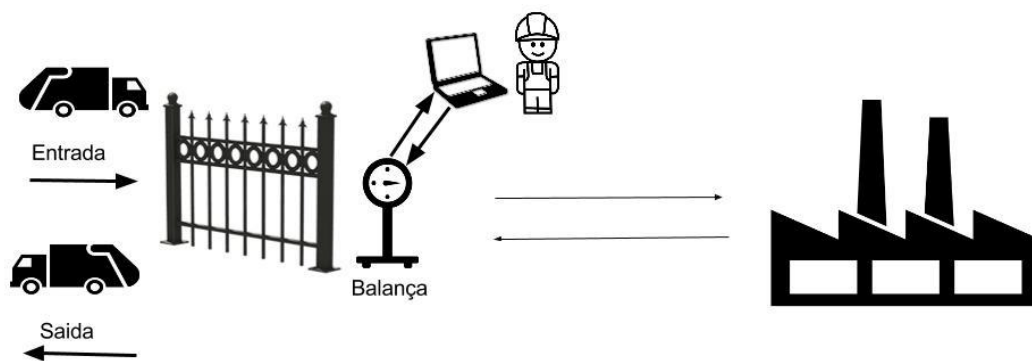


Figura 2 - Fluxo da estação

Como referido anteriormente, existindo apenas uma balança para todo este processo de pesagem, o operador da balança regista os movimentos o mais rápido possível de forma a não congestionar a entrada da estação. A ausência de validação no Software de registo de

movimentos juntamente com a velocidade necessária, por parte do operador, no ato de inserção de dados induz a potenciais erros humanos no ato da inserção do registo.

Presente na figura 3, encontra-se a entrada da estação.



Figura 3 - Entrada e Saída da Estação

2.2. Registo de movimentos da estação

A estação possui protocolos internos definindo assim que veículos podem efetuar certos serviços, por exemplo, um veículo com contentor aberto não poderá efetuar o serviço de transporte para a Meia Serra onde os resíduos são lançados para aterro ou inceneração. Porém no Software existente, tais situações seriam possíveis de registar.

2.2.1. Funcionamento da balança

Os veículos ao passar pelo portão são pesados na balança, junto à balança encontra-se a cabine do operador da mesma. Na chegada de um veículo o operador introduz no Software a matrícula do veículo, identificação do motorista, resíduo transportado e serviço proveniente.

A balança em questão trata-se de uma balança de grande porte com uma sensibilidade de 10kg. Esta encontra-se conectada a um LCD demonstrando assim o peso atual. O LCD é então conectado ao computador através de uma porta *serial* sendo esta a ponte para o Software obter os dados, demonstrado na figura 4.



Figura 4 - Cabine do operador da balança

2.2.2. Software da balança

Esta seção visa explicitar o Software, que interligado ao LCD da balança, regista os movimentos ocorrentes na estação.

Será apresentado o Software utilizado para o registo de movimentos. Este Software trata-se de uma aplicação Microsoft Windows desenvolvida por uma entidade externa à CMF. A figura 5 ilustra o Software em funcionamento na cabine do operador.

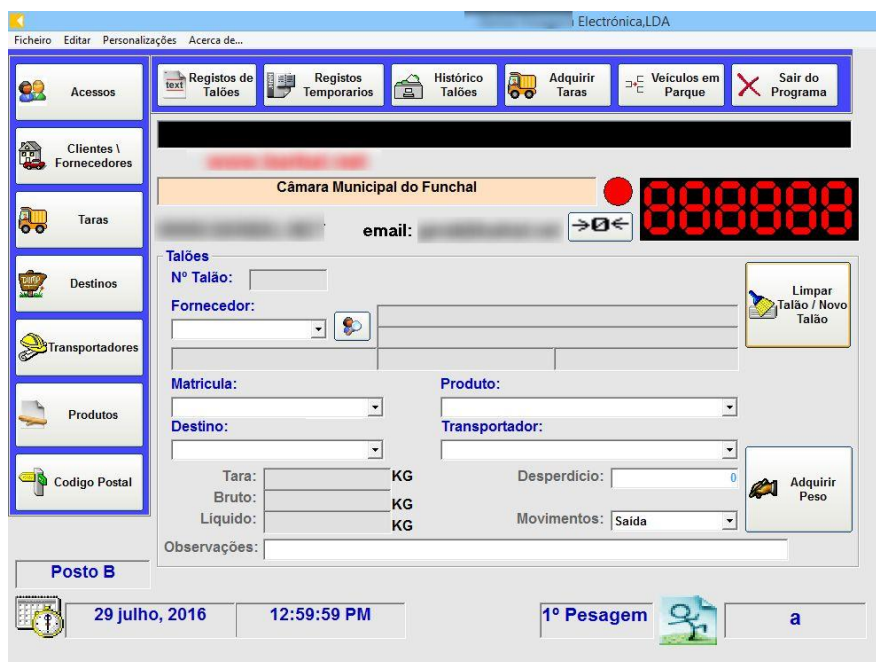


Figura 5 - Software de registo de movimentos

Uma das principais limitações do uso da aplicação é sendo esta desenvolvida por uma entidade externa à CMF e sem período de manutenção, impossibilita assim qualquer tipo de modificação ou necessidade de mudança futura do mesmo.

Os dados do Software, encontram-se centralizados numa base de dados local, Microsoft Access (*.mdb). O Software permite ao operador da balança inserir os dados base, i.e., Fornecedores, Resíduos, Serviços, etc. o que pode resultar em inconsistência dos mesmos, tal como é possível verificar na figura 6.

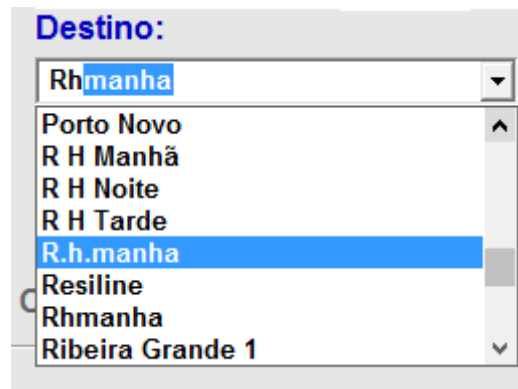


Figura 6 - Inconsistência nos dados

Na figura acima é possível verificar, que existe o destino R H Manhã inserido de três formas diferentes, havendo assim inconsistência e dificultando o trabalho estatístico.

Outro ponto importante a referir, é que o Software inicial não possui qualquer tipo de validação no ato da inserção do registro. Um simples erro de inserção, como por exemplo, registrar que o veículo transportou um resíduo não sendo este o real, terá impacto no resumo mensal de movimentos e conseqüentemente na tabela de faturação mensal.

Este Software não possui qualquer tipo de resumos ou estatísticas sobre os movimentos registrados, sendo este visto apenas como uma ferramenta de inserção de dados. Para acesso ao histórico de registros, é possível exportar os mesmos em formato Microsoft Excel.

2.3. Estatísticas

Como referido previamente, o Software inicial permite a exportação de dados em formato Microsoft Excel, como ilustra a figura 7.

NTalão	Forneced/Matricula	Residuo	Movimen	Tara	Bruto	Líquido	Contentor	Operador	Serviço	Motorista	Data 1ªPesagem	Hora 1ªPe	Data 2ªPesagem	Hora 2ªPe	Observaç	Posto
3810	Madeira C58-EV-84	Indiferent	Entrada		12400	13200	800	0	a	Madeira C Cláudio Fé	15-02-2016	08:40:38	15-02-2016	08:40:48		Posto B
3811	Madeira C00-MA-34	Indiferent	Entrada		7700	11200	3500	0	a	Madeira C Diamantir	15-02-2016	08:45:10	15-02-2016	08:45:16		Posto B
3812	Pingo Doc00-MA-34	Indiferent	Entrada		7700	7880	180	25524886	a	Pingo Doc Diamantir	15-02-2016	08:46:15	15-02-2016	08:46:21		Posto B
3813	C M Funch 18-99-VR	Indiferent	Entrada		3050	4120	1070	0	a	Limpeza U 9697 - Ivo	15-02-2016	08:58:32	15-02-2016	08:58:32		Posto B
3814	C M Funch 94-74-VC	Indiferent	Entrada		1140	1520	380	0	a	R H Manhã 1096 - Ant	15-02-2016	09:06:21	15-02-2016	09:06:21		Posto B
3815	C M Funch 72-85-VH	Cartão	Entrada		5200	6380	1180	0	a	R H Manhã 10449 - Jo	15-02-2016	09:07:17	15-02-2016	09:07:17		Posto B
3816	C M Funch 19-16-VR	Indiferent	Entrada		3050	3690	640	0	a	Limpeza U 9712 - Carl	15-02-2016	09:20:06	15-02-2016	09:20:06		Posto B
3817	C M Funch 89-MJ-13	Indiferent	Entrada		3520	5330	1810	0	a	R H Manhã 10876 - Fil	15-02-2016	09:21:16	15-02-2016	09:21:16		Posto B
3818	C M Funch 06-70-TV	Indiferent	Entrada		5210	7290	2080	0	a	R H Manhã 9346 - Joã	15-02-2016	09:48:54	15-02-2016	09:48:54		Posto B

Figura 7 - Exportação de registos

As estatísticas eram conseguidas ao copiar os dados exportados, presentes na figura acima descrita, para um segundo ficheiro Microsoft Excel, possuindo este um conjunto de regras predefinidas obtendo-se assim as tabelas de resumo.

Ilustra-se então como eram conseguidas as estatísticas pelo Departamento de Ambiente, tenhamos como exemplo, o dia 1 de janeiro de 2016

Apresenta-se uma pequena amostra dos dados exportados do Software da balança. Dados estes, que foram copiados e colados manualmente, para um ficheiro Excel com regras associadas, como é possível visualizar na figura 8.

NTalão	Fornecedor	Matricula	Resíduo	Moviment	Tara	Bruto	Íquido	Serviço	Motorista	Data 1ªPe	Hora 1ªPe	Data 2ªPe	Hora 2ªPe
1	C M Funchal	68-MF-77	Vidro	Entrada	1100	1380	280	Limpeza Urbana	1016 - José Teixeira	01-01-2016	08:09:13	01-01-2016	08:09:13
2	C M Funchal	68-MF-77	Vidro	Entrada	1100	1410	310	Limpeza Urbana	1016 - José Teixeira	01-01-2016	08:48:17	01-01-2016	08:48:17
3	C M Funchal	68-MF-78	Vidro	Entrada	1100	1520	420	Limpeza Urbana	1096 - António	01-01-2016	09:04:41	01-01-2016	09:04:41
4	C M Funchal	18-99-VR	Indiferenciado	Entrada	3050	4610	1560	Limpeza Urbana	9697 - Ivo Pereira	01-01-2016	09:14:23	01-01-2016	09:14:23
5	C M Funchal	54-NX-19	Indiferenciado	Entrada	2650	3250	600	Limpeza Urbana	8478 - Ricardo Ramos	01-01-2016	09:17:59	01-01-2016	09:17:59
6	C M Funchal	19-16-VR	Indiferenciado	Entrada	3050	4010	960	Limpeza Urbana	9712 - Carlos Alberto	01-01-2016	09:55:41	01-01-2016	09:55:41
7	C M Funchal	68-MF-77	Vidro	Entrada	1100	1490	390	Limpeza Urbana	1016 - José Teixeira	01-01-2016	10:04:29	01-01-2016	10:04:29
8	C M Funchal	54-NX-19	Indiferenciado	Entrada	2650	3180	530	Limpeza Urbana	8478 - Ricardo Ramos	01-01-2016	10:43:22	01-01-2016	10:43:22

Figura 8 – Excel formatado – dados 1 de janeiro de 2016

Após inseridos os dados na folha de Excel formatada, então eram obtidos os resumos sobre os movimentos, presente numa segunda folha de cálculo do mesmo ficheiro. A figura 9 representa o resumo diário, do dia 1 de janeiro de 2016.

ENTRADAS - Fornecedores/Resíduos					
Entrada	C M C Lob	C M Funchal	Particular	Kg	Nº Entrads
Baterias	0	0	0	0	0
Cartão	0	1.040	0	1.040	1
Diversos	0	0	0	0	0
Diversos Aterro	0	0	0	0	0
Diversos Incineração	0	0	0	0	0
Embalão	0	2.270	0	2.270	3
Esferovite	0	0	0	0	0
Indiferenciado	0	29.040	0	29.040	20
REEE	0	0	0	0	0
Roupas e Brinquedos	0	0	0	0	0
Vidro	0	5.650	0	5.650	7
	0	0	0	0	0
Sub-Total em Kg	0	38.000	0	38.000	31

Figura 9 - Excel formatado – resumo 1 de janeiro de 2016

Sendo o exemplo apresentado apenas um resumo diário. Quando se tratava de resumos mensais era então copiado o conteúdo de 31 ficheiros Excel, tendo em conta um mês de 31 dias, para o ficheiro Excel com regras associadas. Caso, sejam resumos anuais, é então copiado o conteúdo dos 12 ficheiros Excel de resumo mensal.

2.4. Conclusão

Neste capítulo, analisou-se o funcionamento da estação, como são efetuados os registos e como se obtêm as estatísticas necessárias.

Com esta análise foi possível identificar vários problemas no funcionamento da estação, problemas estes originados maioritariamente no ato de registo de movimentos.

O processo estatístico utilizado, consome imenso tempo e caso os dados não sejam devidamente copiados, as estatísticas obtidas seriam imprecisas.

Após alguma discussão com o DSI sobre o Software da balança e suas limitações, foi então decidido que, a melhor solução seria, o desenvolvimento de um novo Software para a balança. Com o desenvolvimento de um novo Software, será então possível reduzir potenciais erros humanos (utilizando validações de inserção) e simplificar a obtenção de dados estatísticos. O novo Software da balança seria então desenvolvido pelo DSI.

Capítulo 3

Especificação

Após a análise do funcionamento da estação, este capítulo aborda a especificação da plataforma web requisitada pelo Departamento de Ambiente da CMF.

São apresentados o mapa de utilizadores, *stakeholders* e suas restrições, os requisitos funcionais recorrendo a diagramas casos de utilização. A especificação é realizada com na metodologia Use Case 2.0 [1].

Após serem descritos os requisitos funcionais segue-se então à análise sobre os requisitos não-funcionais e restrições tecnológicas da solução proposta.

3.1. Stakeholders

Define-se por *stakeholder*, toda a pessoa ou organização que influencia os requisitos de um sistema ou é afetada por este. É importante compreender quem são os *stakeholders*, quais as suas necessidades e manter contacto com os mesmos de forma a compreender as suas necessidades, obtendo assim um produto satisfatório [2].

3.1.1. Identificação

A identificação dos *stakeholders*, pode ser considerada uma das tarefas mais importantes de Engenharia de Requisitos, caso esta etapa seja ignorada é muito provável o fracasso do projeto.

De forma a identificar os *stakeholders* relevantes devemos procurar por pessoas ou organizações que [3]:

1. Tenham um interesse ativo no sistema pois serão estas que eventualmente o utilizarão;
2. Tenham que operar, manter e gerir o sistema depois deste ter sido desenvolvido;
3. Tenham um papel no desenvolvimento do sistema, sejam estes os programadores ou até mesmo os *testers*;
4. Tenham responsabilidades sobre o processo que o sistema irá suportar ou automatizar.

Com base nos pontos acima referidos foram identificados os principais *stakeholders* do projeto, representados na figura 10.

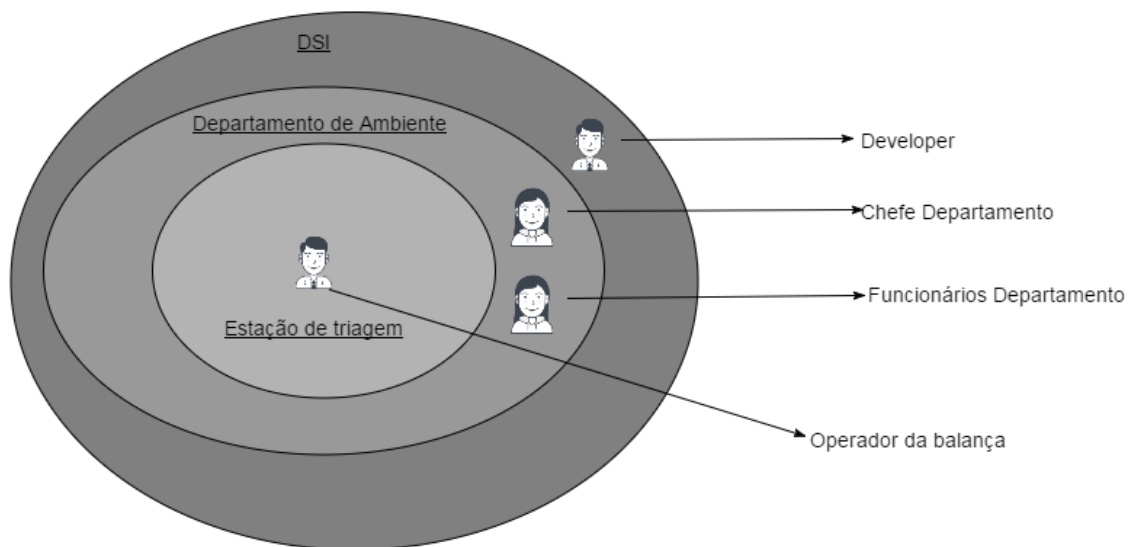


Figura 10 - Diagrama de stakeholders

3.1.2. Perfis

Após identificação dos *stakeholders*, define-se o perfil dos mesmos.

Na tabela seguinte é representado um perfil do operador da balança.

Tabela 1 – Perfil Operador da balança

<i>Cargo do utilizador</i>	Registrar os movimentos ocorrentes na estação.
<i>Experiencia no domínio</i>	Experiente, controla todo o fluxo da estação.
<i>Experiencia tecnológica</i>	Baixa, conhecimentos tecnológicos básicos.
<i>Outras características</i>	Barreiras linguistas, apenas domina a língua Portuguesa. Desejo reduzido de aprender novas tecnologias e sistemas.

Presente na tabela 2, encontra-se o perfil detalhado do Chefe do Departamento.

Tabela 2 - Perfil Chefe de Departamento

<i>Cargo do utilizador</i>	Gerir o departamento, definir circuitos de recolha de resíduos, analisar dados dos movimentos.
<i>Experiencia no domínio</i>	Experiente, conhece e define toda a lógica de funcionamento da estação
<i>Experiencia tecnológica</i>	Alta, facilidade de manuseamento tecnológico.

Em relação ao funcionário do departamento, resume-se o seguinte perfil presente na tabela 3.

Tabela 3 - Perfil Funcionário departamento

<i>Cargo do utilizador</i>	Verificar e corrigir possíveis anomalias nos registos de movimento. Inserir registos presentes na lista de movimentos particulares (folha A4 entregue pelo operador da balança).
<i>Experiencia no domínio</i>	Alta, capaz de identificar facilmente anomalias nos dados.
<i>Experiencia tecnológica</i>	Média, trabalha regularmente com computadores com média destreza.

Entende-se por *developer*, qualquer pessoa responsável pela área de programação e/ou manutenção informática do DSI.

Tabela 4 - Perfil Developer

<i>Cargo do utilizador</i>	Manutenção e alteração de qualquer software presente na CMF.
<i>Experiencia no domínio</i>	Experiência moderada, não estando ligado diretamente ao Departamento de Ambiente.
<i>Experiencia tecnológica</i>	Avançada.

3.1.3. Permissões de acesso

Como referido no Capítulo 1, terá de existir áreas de acesso restrito acessível apenas a determinados utilizadores.

De forma a garantir o mesmo, definiu-se quatro tipos de grupo de utilizadores aos quais foram associados cada utilizador/ *stakeholder* [4]. São assim identificados na tabela 5 os quatro grupos de utilizadores da plataforma, sendo cada nível mais alto uma extensão ao anterior.

Tabela 5 - Grupos de utilizadores

<i>Grupo</i>	<i>Permissões</i>
1. <i>User</i>	Apenas com acesso a visualizar estatísticas
2. <i>Editor</i>	Acesso ao Painel de movimentos, possibilidade de adicionar ou editar registos de movimentos.
3. <i>Admin</i>	Gerir entidades base da plataforma, tais como adicionar e desativar fornecedores.
4. <i>Super Admin</i>	Gerir utilizadores da plataforma, acesso a <i>logs</i> do sistema.

3.2. Requisitos Funcionais

Requisitos funcionais são considerados as funcionalidades desejadas do sistema. De forma a transmitir de forma simples e eficaz as funcionalidades desejadas aos *stakeholders* do projeto, recorreu-se então a um diagrama de casos de uso.

Após terem sido aceites os casos de utilização por parte do Departamento de Ambiente, passa-se então a detalhar cada um dos mesmos explorando assim seu comportamento base, comportamento alternativo e eventuais requisitos especiais necessários.

Passaremos finalmente a uma fase de *slicing*, definindo assim funcionalidades associadas ao caso de utilização. Para toda a *slice* é

também definindo o seu *test case* assim como o critério de teste para cada um deles [1].

3.2.1. Casos de uso essenciais

Casos de uso essenciais, ou *use case essentials* [1], são considerados os casos de utilização da plataforma ou sistema em questão. São definidos de forma abstrata, simples e independentes de tecnologia capturando apenas a intensão do utilizador ao realizar uma determinada tarefa [5].

Segue-se, na figura 11, o diagrama de casos de uso que representa os casos de utilização do sistema.

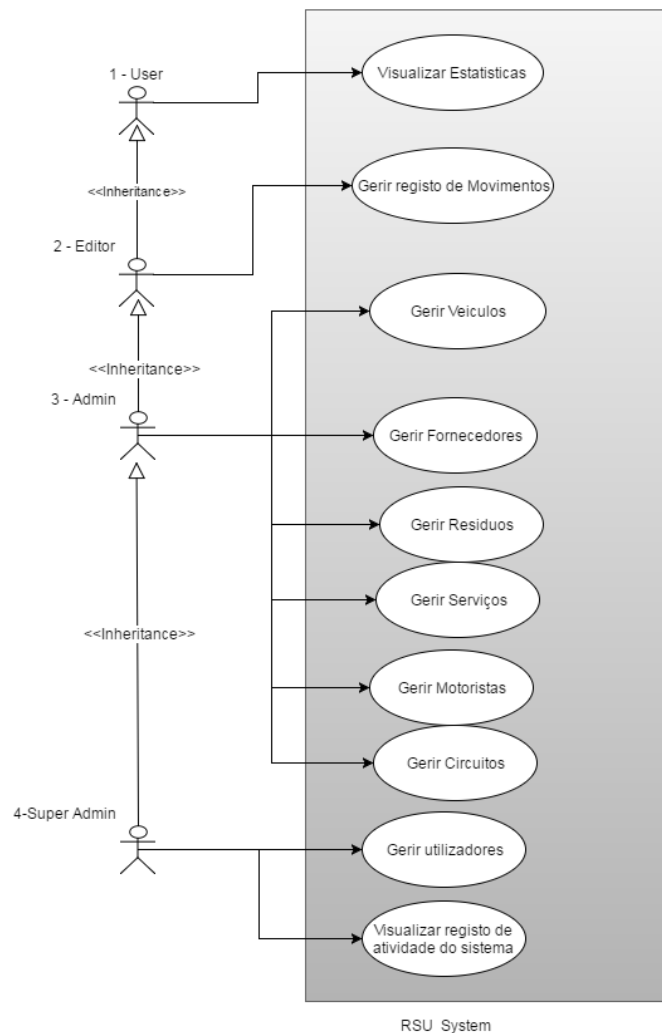


Figura 11 - Diagrama Use case. (Nota: Gerir corresponde às operações CRUD)

Casos de utilização essenciais não apresentam muito detalhe, mas iremos abordar cada um deles e apresentar cada subfuncionalidade dos mesmos.

Para identificar cada caso de utilização, é apresentada uma tabela enumerando cada um deles. Assim sendo, a qualquer ponto do projeto, é facilmente identificável a que caso de utilização uma determinada funcionalidade pertence, com tal obtém-se requisitos rastreáveis [6].

Tabela 6 - Casos de utilização

<i>Número</i>	<i>Caso de utilização</i>
1	Visualizar Estatísticas
2	Gerir Registo de Movimentos
3	Gerir Veículos
4	Gerir Fornecedores
5	Gerir Resíduos
6	Gerir Serviços
7	Gerir Motoristas
8	Gerir Utilizadores
9	Gerir Circuitos
10	Visualizar registo de atividade no sistema

Definida a numeração dos casos de utilização, presente na tabela 6, passa-se agora detalhar com mais rigor cada um deles, recorrendo então à metodologia Use Case 2.0 [1]. Assim sendo, define-se uma descrição do caso de utilização, fluxo básico, fluxo alternativo e possíveis requisitos especiais.

3.2.1.1. Caso de Utilização - Visualizar Estatísticas

Explora-se nesta secção, o caso de uso “Visualizar Estatísticas”. Apresenta-se assim, na tabela 7, o caso de uso e posteriormente definem-se as suas *slices*.

Tabela 7 - Fluxo de "Visualizar Estatísticas"

<i>Caso de utilização</i>	<i>Visualizar Estatísticas</i>
<i>Descrição</i>	O Sistema deve apresentar as estatísticas desejadas ao utilizador
<i>Fluxo Básico</i>	<ol style="list-style-type: none"> 1. Utilizador seleciona o critério estatístico desejado 2. O sistema pesquisa com base dos critérios selecionados 3. O sistema apresenta as estatísticas ao utilizador
<i>Fluxo alternativo</i>	
<i>Requisitos Especiais</i>	<ol style="list-style-type: none"> S1. Exportar estatística como pdf S2. Exportar estatísticas como xls

Define-se agora as *slices* provenientes do caso de utilização apresentado, identificando assim funcionalidades necessárias para cumprir o objetivo inicial e seus respetivos testes de aceitação.

Para cada *slice* identifica-se a sequência de fluxo que originou o mesmo, utiliza-se as siglas BFX identificando *Basic Flow x*, Ax sendo *Alternative Flow x*, e por último Sx surgindo dos *Special Requirements* [1].

Tabela 8 - Slice 1.1 "Pesquisa de Registos"

UCS 1.1 *Pesquisa de Registos*

<i>Fluxo</i>	BF2
<i>Teste de Aceitação</i>	Dados devidamente formatados e corretos
<i>Condições de teste</i>	Filtros selecionados

A *slice 1.1* resume a responsabilidade do sistema de pesquisar e resumir os dados requisitados pelo utilizador.

Apresenta-se a *slice* seguinte representando, "Visualizar Estatísticas".

Tabela 9 - Slice 1.2 "Visualizar Estatísticas"

UCS 1.2 *Visualizar Estatísticas*

<i>Fluxo</i>	UCS 1.1 + BF3
<i>Teste de Aceitação</i>	Dados estatísticos devidamente exibidos
<i>Condições de teste</i>	Dados recebidos corretos

A *slice* anterior, resume o ato em que o utilizador visualiza as estatísticas desejadas, para tal é necessário recorrer à *slice 1.1(UCS 1.1)*, da qual os dados são obtidos para apresentar a estatística.

As *slices* seguintes, representam o desejo adicional por parte dos *stakeholders* terem a possibilidade de exportar os seus dados estatísticos para um ficheiro de fácil transporte, visto a plataforma web estar unicamente acessível internamente à instituição.

Tabela 10 - Slice 1.3 "Exportar como pdf"

UCS 1.3 *Exportar Estatísticas como pdf*

<i>Fluxo</i>	UCS 1.2 + S1
<i>Teste de Aceitação</i>	Ficheiro Pdf obtido com seu respetivo conteúdo
<i>Condições de teste</i>	Existir uma visualização estatística.

A seguinte *slice* torna-se muito semelhante à anterior, apenas diferindo o formato de exportação desejado.

Tabela 11 - Slice 1.4 "Exportar como xls"

UCS 1.3

Exportar Estatísticas como xls

<i>Fluxo</i>	UCS 1.2 + S2
<i>Teste de Aceitação</i>	Ficheiro xls obtido com seu respetivo conteúdo
<i>Condições de teste</i>	Existir uma visualização estatística.

3.2.1.2. Caso de Utilização - Gerir Registo de Movimentos

Uma vez mais, seguiremos a estrutura do Caso de Utilização anterior, detalhando assim o seguinte caso “Gerir Registo de Movimentos”.

Tabela 12 - Fluxo de "Gerir Registo de Movimentos"

Caso de utilização

Gerir Registo de Movimentos

<i>Descrição</i>	O utilizador pode efetuar operações de CRUD sobre os registos de movimentos
<i>Fluxo Básico</i>	<ol style="list-style-type: none"> 1. Utilizador seleciona a operação desejada 2. Utilizador efetua a operação 3. Sistema valida a operação 4. Sistema apresenta notificação sobre o sucesso da ação
<i>Fluxo alternativo</i>	A1. Dados inválidos

De seguida apresenta-se as *slices* necessárias a realizar o caso de utilização anterior.

Tabela 13 - Slice 2.1 "Pesquisa de Registos"

UCS 2.1	Pesquisa de Registos
<i>Fluxo</i>	BF1
<i>Teste de Aceitação</i>	Listagem de registos exibida
<i>Condições de teste</i>	Existir pelo menos um registo

Antes de qualquer tipo de modificação, é necessário pesquisar pelo registo em questão. A *slice* presente na tabela 13 representa tal ação.

Tabela 14 - Slice 2.2 "Adicionar Registo"

UCS 2.2	Adicionar Registo
<i>Fluxo</i>	BF2 + BF3 + BF4 + A1
<i>Teste de Aceitação</i>	Registo adicionado com sucesso
<i>Condições de teste</i>	Dados válidos

Além do operador da balança poder adicionar registos de movimentos pelo software da balança, foi requisitado também a plataforma conceder tal utilidade.

Tabela 15 - Slice 2.3 "Modificar Registo"

UCS 2.3	Modificar Registo
<i>Fluxo</i>	UCS 2.1 + BF2 + BF3 + BF4 + A1
<i>Teste de Aceitação</i>	Registo modificado com sucesso
<i>Condições de teste</i>	Dados válidos, Existir Registo

A funcionalidade de modificar registo, é necessário estar presente, de modo a corrigir qualquer erro humano não detetável pela plataforma.

Os registos não podem ser eliminados, a pedido do Departamento de Ambiente, apenas dando possibilidade de os desativar, assim não sendo contabilizados nas estatísticas.

Tabela 16 - Slice 2.4 " Desativar Registo "

UCS 2.4

Desativar Registo

<i>Fluxo</i>	UCS 2.1 + BF2 + BF3 + BF4
<i>Teste de Aceitação</i>	Registo marcado como desativo
<i>Condições de teste</i>	Existir Registo

3.2.1.3. Caso de Utilização - Gerir Veículos

Segue-se o caso de utilização "Gerir Veículos".

Tabela 17 - Fluxo "Gerir Veículos"

Caso de utilização

Gerir Veículos

<i>Descrição</i>	O utilizador pode efetuar operações de CRUD sobre a entidade de Veículos
<i>Fluxo Básico</i>	<ol style="list-style-type: none"> 1. Utilizador seleciona a operação desejada 2. Utilizador efetua a operação 3. Sistema valida a operação 4. Sistema apresenta notificação sobre o sucesso da ação
<i>Fluxo alternativo</i>	A1. Dados inválidos

De seguida apresenta-se as *slices* necessárias a realizar o caso de utilização anterior.

Tabela 18 - Slice 3.1 "Pesquisa de Registos"

UCS 3.1

Pesquisa de Veículos

<i>Fluxo</i>	BF1
<i>Teste de Aceitação</i>	Listagem de veículos exibida
<i>Condições de teste</i>	Existir pelo menos um registo

Antes de qualquer tipo de modificação, é necessário pesquisar pelo veículo.

Tabela 19 - Slice 3.2 "Adicionar Veículo"

UCS 3.2

Adicionar Registo

<i>Fluxo</i>	BF2 + BF3 + BF4 + A1
<i>Teste de Aceitação</i>	Veículo adicionado com sucesso
<i>Condições de teste</i>	Dados válidos

A funcionalidade presente na *slice* 3.2, permite adicionar novos veículos à plataforma.

Tabela 20 - Slice 3.3 "Modificar Registo"

UCS 3.3

Modificar Veículo

<i>Fluxo</i>	UCS 3.1 + BF2 + BF3 + BF4 + A1
<i>Teste de Aceitação</i>	Veículo modificado com sucesso
<i>Condições de teste</i>	Dados válidos, Existir Registo

Representado na tabela 20, define-se a possibilidade de modificar informação referente aos veículos tais como os resíduos que o mesmo pode transportar.

A plataforma não deverá permitir a remoção de veículos, apenas disponibilizando a possibilidade de os desativar, seja este por razões de avaria ou até mesmo abate.

Tabela 21 - Slice 3.4 " Desativar Veículo "

UCS 3.4

Desativar Veículo

<i>Fluxo</i>	UCS 3.1 + BF2 + BF3 + BF4
<i>Teste de Aceitação</i>	Veículo marcado como desativo
<i>Condições de teste</i>	Existir Registo

3.2.1.4. Caso de Utilização - Gerir Fornecedores

Apresenta-se o caso de utilização “Gerir Fornecedores”, representando as organizações que transportam resíduos para a estação.

Tabela 22 - Fluxo de "Gerir Fornecedores"

Caso de utilização

Gerir Fornecedores

<i>Descrição</i>	O utilizador pode efetuar operações de CRUD sobre a entidade de Fornecedores
<i>Fluxo Básico</i>	<ol style="list-style-type: none"> 1. Utilizador seleciona a operação desejada 2. Utilizador efetua a operação 3. Sistema valida a operação 4. Sistema apresenta notificação sobre o sucesso da ação
<i>Fluxo alternativo</i>	A1. Dados inválidos

De seguida apresenta-se as subfuncionalidades necessárias a realizar a tarefa previamente descrita.

Tabela 23 - Slice 4.1 "Pesquisa de Registos"

UCS 4.1

Pesquisa de Fornecedor

<i>Fluxo</i>	BF1
<i>Teste de Aceitação</i>	Listagem de fornecedores exibida
<i>Condições de teste</i>	Existir pelo menos um registo

A funcionalidade base, previamente a qualquer modificação é a de encontrar o próprio registo.

Tabela 24 - Slice 4.2 "Adicionar Fornecedor"

UCS 4.2

Adicionar Fornecedor

<i>Fluxo</i>	BF2 + BF3 + BF4 + A1
<i>Teste de Aceitação</i>	Fornecedor adicionado com sucesso
<i>Condições de teste</i>	Dados válidos

Representa-se acima, a ação de adicionar um novo fornecedor ao sistema.

Tabela 25 - Slice 4.3 "Modificar Fornecedor"

UCS 4.3

Modificar Fornecedor

<i>Fluxo</i>	UCS 4.1 + BF2 + BF3 + BF4 + A1
<i>Teste de Aceitação</i>	Fornecedor modificado com sucesso
<i>Condições de teste</i>	Dados válidos, Existir Registo

Representado na tabela 25, define-se a possibilidade de modificar informação referente aos fornecedores tais como os serviços que o mesmo pode realizar.

A plataforma não deverá permitir a remoção de Fornecedores, apenas desativar os mesmos.

Tabela 26 - Slice 4.4 " Desativar Veículo "

UCS 3.4

Desativar Veículo

<i>Fluxo</i>	UCS 4.1 + BF2 + BF3 + BF4
<i>Teste de Aceitação</i>	Fornecedor marcado como desativo
<i>Condições de teste</i>	Existir Registo

3.2.1.5. Caso de Utilização - Gerir Resíduos

Apresenta-se o caso “Gerir Resíduos”.

Tabela 27 - Fluxo "Gerir Resíduos"

Caso de utilização

Gerir Resíduos

<i>Descrição</i>	O utilizador pode efetuar operações de CRUD sobre a entidade de Resíduos
<i>Fluxo Básico</i>	<ol style="list-style-type: none"> 1. Utilizador seleciona a operação desejada 2. Utilizador efetua a operação 3. Sistema valida a operação 4. Sistema apresenta notificação sobre o sucesso da ação
<i>Fluxo alternativo</i>	A1. Dados inválidos

De seguida apresenta-se as subfuncionalidades necessárias a realizar a tarefa previamente descrita.

Tabela 28 - Slice 5.1 "Pesquisa de Resíduos"

UCS 5.1

Pesquisa de Resíduo

<i>Fluxo</i>	BF1
<i>Teste de Aceitação</i>	Listagem de resíduos exibida
<i>Condições de teste</i>	Existir pelo menos um registro

É necessário encontrar o próprio registro, antes de o poder modificar.

Tabela 29 - Slice 5.2 "Adicionar Resíduo"

UCS 5.2

Adicionar Resíduo

<i>Fluxo</i>	BF2 + BF3 + BF4 + A1
<i>Teste de Aceitação</i>	Resíduo adicionado com sucesso
<i>Condições de teste</i>	Dados válidos

Apresenta-se acima, a ação de adicionar um novo resíduo ao sistema. Referindo então, um novo resíduo que poderá dar entrada na estação.

Tabela 30 - Slice 5.3 "Modificar Resíduo"

UCS 5.3

Modificar Resíduo

<i>Fluxo</i>	UCS 5.1 + BF2 + BF3 + BF4 + A1
<i>Teste de Aceitação</i>	Resíduo modificado com sucesso
<i>Condições de teste</i>	Dados válidos, Existir Registro

Representado na tabela 30, define-se a possibilidade de modificar informação referente ao resíduo.

A plataforma não deverá permitir a remoção de Resíduos, apenas desativar os mesmos.

Tabela 31 - Slice 5.4 " Desativar Resíduo "

UCS 5.4	Desativar Resíduo
<i>Fluxo</i>	UCS 5.1 + BF2 + BF3 + BF4
<i>Teste de Aceitação</i>	Resíduo marcado como desativo
<i>Condições de teste</i>	Existir Registo

3.2.1.6. Caso de Utilização - Gerir Serviços

Apresenta-se o caso “Gerir Serviços”.

Tabela 32 - Fluxo "Gerir Serviços"

Caso de utilização	Gerir Serviços
<i>Descrição</i>	O utilizador pode efetuar operações de CRUD sobre a entidade de Serviços
<i>Fluxo Básico</i>	<ol style="list-style-type: none"> 1. Utilizador seleciona a operação desejada 2. Utilizador efetua a operação 3. Sistema valida a operação 4. Sistema apresenta notificação sobre o sucesso da ação
<i>Fluxo alternativo</i>	A1. Dados inválidos

De seguida apresenta-se as subfuncionalidades necessárias a realizar a tarefa previamente descrita.

Tabela 33 - Slice 6.1 "Pesquisa de Serviços"

UCS 6.1

Pesquisa de Serviços

<i>Fluxo</i>	BF1
<i>Teste de Aceitação</i>	Listagem de serviços exibida
<i>Condições de teste</i>	Existir pelo menos um registo

A funcionalidade de pesquisar Serviços.

Tabela 34 - Slice 6.2 "Adicionar Serviços"

UCS 6.2

Adicionar Serviço

<i>Fluxo</i>	BF2 + BF3 + BF4 + A1
<i>Teste de Aceitação</i>	Serviço adicionado com sucesso
<i>Condições de teste</i>	Dados válidos

Representa-se acima, a ação de adicionar um novo Serviço ao sistema.

Tabela 35 - Slice 6.3 "Modificar Serviço"

UCS 6.3

Modificar Serviço

<i>Fluxo</i>	UCS 6.1 + BF2 + BF3 + BF4 + A1
<i>Teste de Aceitação</i>	Serviço modificado com sucesso
<i>Condições de teste</i>	Dados válidos, Existir Registo

Representado na tabela 35, define-se a possibilidade de modificar informação referente ao serviço.

A plataforma não deverá permitir a remoção de Serviços, apenas desativar os mesmos.

Tabela 36 - Slice 6.4 " Desativar Serviço "

UCS 6.4

Desativar Serviço

<i>Fluxo</i>	UCS 6.1 + BF2 + BF3 + BF4
<i>Teste de Aceitação</i>	Serviço marcado como desativo
<i>Condições de teste</i>	Existir Registo

3.2.1.7. Caso de Utilização - Gerir Motoristas

Apresenta-se o caso “Gerir Motoristas”.

Tabela 37 - Fluxo "Gerir Motoristas"

Caso de utilização

Gerir Motoristas

<i>Descrição</i>	O utilizador pode efetuar operações de CRUD sobre a entidade de Motorista
<i>Fluxo Básico</i>	<ol style="list-style-type: none"> 1. Utilizador seleciona a operação desejada 2. Utilizador efetua a operação 3. Sistema valida a operação 4. Sistema apresenta notificação sobre o sucesso da ação
<i>Fluxo alternativo</i>	A1. Dados inválidos

De seguida apresenta-se as subfuncionalidades necessárias a realizar a tarefa previamente descrita.

Tabela 38 - Slice 7.1 "Pesquisa de Motoristas"

UCS 7.1

Pesquisa de Motoristas

<i>Fluxo</i>	BF1
<i>Teste de Aceitação</i>	Listagem de serviços exibida
<i>Condições de teste</i>	Existir pelo menos um registo

A funcionalidade de pesquisar Motoristas.

Tabela 39 - Slice 7.2 "Adicionar Motoristas"

UCS 7.2

Adicionar Motorista

<i>Fluxo</i>	BF2 + BF3 + BF4 + A1
<i>Teste de Aceitação</i>	Motorista adicionado com sucesso
<i>Condições de teste</i>	Dados válidos

Representa-se acima, a ação de adicionar um novo Motorista ao sistema.

Tabela 40 - Slice 7.3 "Modificar Motorista"

UCS 7.3

Modificar Motorista

<i>Fluxo</i>	UCS 7.1 + BF2 + BF3 + BF4 + A1
<i>Teste de Aceitação</i>	Motorista modificado com sucesso
<i>Condições de teste</i>	Dados válidos, Existir Registo

Acima representa a utilidade de modificar a informação de um Motorista conhecido ao sistema.

A plataforma não deverá permitir a remoção de Motoristas, apenas desativar os mesmos.

Tabela 41 - Slice 7.4 " Desativar Motorista "

UCS 7.4

Desativar Motorista

<i>Fluxo</i>	UCS 7.1 + BF2 + BF3 + BF4
<i>Teste de Aceitação</i>	Motorista marcado como desativo
<i>Condições de teste</i>	Existir Registo

3.2.1.8. Caso de Utilização - Gerir Utilizadores

Neste ponto, aborda-se o caso de uso “Gerir Utilizadores”, este sendo o responsável por toda a gestão de todos os utilizadores da plataforma. Começa-se por definir abstratamente o caso de utilização.

Tabela 42 - Fluxo "Gerir Utilizadores"

Caso de utilização

Gerir Utilizadores

<i>Descrição</i>	Administração total sobre os utilizadores da plataforma
<i>Fluxo Básico</i>	<ol style="list-style-type: none"> 1. Utilizador seleciona a operação desejada 2. Utilizador efetua a operação 3. Sistema valida a operação 4. Sistema apresenta notificação sobre o sucesso da ação
<i>Fluxo alternativo</i>	<ol style="list-style-type: none"> A1. Operação não permitida A2. Serviço de autenticação offline A3. Utilizador inválido

Tabela 43 - Slice 8.1 "Listar Utilizadores"

UCS 8.1 *Listar Utilizadores*

<i>Fluxo</i>	BF1
<i>Teste de Aceitação</i>	Listagem de utilizadores exibida
<i>Condições de teste</i>	Existir pelo menos um registo

A *slice* anterior define a funcionalidade de listar todos os utilizadores presentes na plataforma.

Tabela 44 - Slice 8.2 "Alterar grupo de utilizador"

UCS 8.2 *Alterar grupo de utilizador*

<i>Fluxo</i>	UCS 8.1 + BF2 + BF3 + BF4
<i>Teste de Aceitação</i>	Grupo de utilizador modificado
<i>Condições de teste</i>	Existir um utilizador

A funcionalidade definida anteriormente, modifica o grupo a que o utilizador pertence, ou seja, alterar o nível de acesso do mesmo perante a plataforma.

Tabela 45 - Slice 8.3 "Adicionar novo utilizador"

UCS 8.3 *Adicionar novo utilizador*

<i>Fluxo</i>	UCS 8.1 + BF2 + BF3 + BF4 + A2 + A3
<i>Teste de Aceitação</i>	Novo utilizador adicionado com sucesso
<i>Condições de teste</i>	Existir informação sobre o utilizador no serviço de autenticação da CMF

A *slice* definida, na tabela anterior, tem como objetivo adicionar um novo utilizador perante a plataforma. Para tal ser bem-sucedido, o utilizador terá de existir perante o sistema de autenticação da CMF.

Tabela 46 - Slice 8.4 "Bloquear utilizador"

UCS 8.4

Bloquear acesso ao utilizador

<i>Fluxo</i>	UCS 8.1 + BF2 + BF3 + BF4 + A2 + A3
<i>Teste de Aceitação</i>	Utilizador não consegue se autenticar perante a plataforma
<i>Condições de teste</i>	Existir um utilizador

A qualquer momento, será possível remover o acesso de um determinado utilizador à plataforma.

Tabela 47 - Slice 8.5 "Remover conta de utilizador"

UCS 8.5

Remover conta de utilizador

<i>Fluxo</i>	UCS 8.1 + BF2 + BF3 + BF4 + A1 + A2
<i>Teste de Aceitação</i>	Utilizador removido da plataforma
<i>Condições de teste</i>	Existir um utilizador

A plataforma deve permitir remover uma conta de utilizador, caso esta, não tenha registos associados.

3.2.1.9. Caso de Utilização - Gerir Circuitos

Apresenta-se o caso “Gerir Circuitos”.

Tabela 48 - Fluxo "Gerir Circuitos"

<i>Caso de utilização</i>	<i>Gerir Circuitos</i>
<i>Descrição</i>	O utilizador pode efetuar operações de CRUD sobre a entidade de Circuito
<i>Fluxo Básico</i>	<ol style="list-style-type: none"> 1. Utilizador seleciona a operação desejada 2. Utilizador efetua a operação 3. Sistema valida a operação 4. Sistema apresenta notificação sobre o sucesso da ação
<i>Fluxo alternativo</i>	A1. Dados inválidos
<i>Requisitos Especiais</i>	S1. Adicionar relatórios de circuito

De seguida apresenta-se as subfuncionalidades necessárias a realizar a tarefa previamente descrita.

Tabela 49 - Slice 9.1 "Pesquisa de Circuitos"

<i>UCS 9.1</i>	<i>Pesquisa de Circuitos</i>
<i>Fluxo</i>	BF1
<i>Teste de Aceitação</i>	Listagem de serviços exibida
<i>Condições de teste</i>	Existir pelo menos um registo

A funcionalidade de pesquisar Circuitos.

Tabela 50 - Slice 9.2 "Adicionar Circuitos"

UCS 9.2

Adicionar Circuito

<i>Fluxo</i>	BF2 + BF3 + BF4 + A1
<i>Teste de Aceitação</i>	Circuito adicionado com sucesso
<i>Condições de teste</i>	Dados válidos

Representa-se acima, a ação de adicionar um novo Circuito ao sistema.

Tabela 51 - Slice 9.3 "Modificar Circuito"

UCS 9.3

Modificar Circuito

<i>Fluxo</i>	UCS 9.1 + BF2 + BF3 + BF4 + A1
<i>Teste de Aceitação</i>	Circuito modificado com sucesso
<i>Condições de teste</i>	Dados válidos, Existir Registo

Acima representa a utilidade de modificar a informação de um Motorista conhecido ao sistema.

A plataforma não deverá permitir a remoção de Circuitos, apenas desativar os mesmos.

Tabela 52 - Slice 9.4 "Desativar Circuitos "

UCS 9.4

Desativar Circuitos

<i>Fluxo</i>	UCS 9.1 + BF2 + BF3 + BF4
<i>Teste de Aceitação</i>	Circuito marcado como desativo
<i>Condições de teste</i>	Existir Registo

UCS 9.5

Adicionar Relatório de remoção do Circuitos

<i>Fluxo</i>	S1
<i>Teste de Aceitação</i>	Relatório adicionado com sucesso
<i>Condições de teste</i>	Existir Movimentos associados ao relatório preenchido

3.2.1.10. Caso de Utilização - Visualizar registo de atividade do sistema

Explora-se nesta secção, o caso de uso “Visualizar registo de atividade do sistema”.

Tabela 53 - Fluxo "Visualizar registo de atividade do sistema"

*Caso de utilização**Visualizar Estatísticas*

<i>Descrição</i>	O Sistema deve apresentar o registo de atividade do sistema
<i>Fluxo Básico</i>	<ol style="list-style-type: none"> 1. Utilizador seleciona a entidade que pretende 2. O sistema pesquisa com base dos critérios selecionados 3. O sistema apresenta o registo de atividade ao utilizador

Define-se agora as *slices* provenientes do caso de utilização anterior, originando assim funcionalidades necessárias para cumprir o objetivo inicial e seus respetivos testes de aceitação.

- | | |
|---------------------------------------|---|
| 2. <i>Gerir Registo de Movimentos</i> | <ol style="list-style-type: none">1. Pesquisa de Registos2. Adicionar Registos3. Modificar Registos4. Desativar Registos |
| 3. <i>Gerir Veículos</i> | <ol style="list-style-type: none">1. Pesquisa de Registos2. Adicionar Registos3. Modificar Registos4. Desativar Registos |
| 4. <i>Gerir Fornecedores</i> | <ol style="list-style-type: none">1. Pesquisa de Registos2. Adicionar Registos3. Modificar Registos4. Desativar Registos |
| 5. <i>Gerir Resíduos</i> | <ol style="list-style-type: none">1. Pesquisa de Registos2. Adicionar Registos3. Modificar Registos4. Desativar Registos |
| 6. <i>Gerir Serviços</i> | <ol style="list-style-type: none">1. Pesquisa de Registos2. Adicionar Registos3. Modificar Registos4. Desativar Registos |
| 7. <i>Gerir Motoristas</i> | <ol style="list-style-type: none">1. Pesquisa de Registos2. Adicionar Registos3. Modificar Registos4. Desativar Registos |

8. <i>Gerir Utilizadores</i>	<ol style="list-style-type: none"> 1. Listar utilizadores 2. Alterar grupo de utilizador 3. Adicionar novo utilizador 4. Bloquear utilizador 5. Remover Utilizador
9. <i>Gerir Circuitos</i>	<ol style="list-style-type: none"> 1. Pesquisa de Registos 2. Adicionar Registos 3. Modificar Registos 4. Desativar Registos 5. Adicionar Relatório de remoção do Circuitos
10. <i>Visualizar Registo de atividade</i>	<ol style="list-style-type: none"> 1. Pesquisar de registo de atividade 2. Visualizar Registo de atividade

3.3. Requisitos Não-funcionais

Pode-se referir que um requisito funcional é um requisito a qual diz respeito uma preocupação funcional do sistema. Um requisito não funcional (RNF) é considerado um atributo ou até mesmo uma restrição do sistema em questão [7].

O principal atributo de qualidade é o de modificabilidade, visto que o projeto poderá evoluir ou até mesmo serem pedidas alterações, a facilidade de modificação foi considerada a principal *architecture driver* [8]. Abordaremos adiante que arquitetura que foi escolhida com base no atributo de qualidade descrito.

Outro atributo de qualidade importante para a plataforma é o de Usabilidade. A usabilidade representa o grau de facilidade em que o utilizador realiza a tarefa desejada, este atributo é considerado uma das formas mais simples de aumentar a qualidade do sistema [8].

Um outro requisito não-funcional por parte da plataforma, é o de manter um registo de atividade da mesma (*audit trail*).

3.4. Restrições tecnológicas

Após a plataforma ter sido entregue ao Departamento de Ambiente, o DSI torna-se responsável por qualquer tipo de modificação ou até extensão da mesma. Assim sendo, as tecnologias foram impostas pelo DSI. O desenvolvimento da plataforma foi realizado inteiramente em PHP e sua base de dados em MySQL.

No capítulo seguinte, abordaremos a revisão tecnológica tal como possíveis alternativas à mesma, caso não existissem tais restrições.

3.5. Conclusão

Neste capítulo definiu-se o mapa de stakeholders, requisitos funcionais, requisitos não funcionais e restrições tecnológicas.

Após definidos os casos de utilização recorreu-se à metodologia Use Case 2.0, especificando assim mais detalhadamente as funcionalidades necessárias.

Abordou-se os requisitos não funcionais e restrições tecnológicas, sendo estes os que impulsionam a decisão de uma arquitetura para a plataforma web.

Com base da especificação produzida neste capítulo, é possível então seguir com a implementação da plataforma web.

Capítulo 4

Revisão tecnológica

Este capítulo aborda o estudo realizado sobre as possíveis tecnologias para a realização da plataforma em questão.

Foi efetuada uma pesquisa a possíveis tecnologias alternativas às impostas pela instituição, investigando assim numa reflexão crítica, a existência de melhores alternativas.

4.1. Tecnologias server-side

Iremos abordar algumas das tecnologias de *Backend* possíveis, considerando-se apenas as tecnologias mais populares, para realizar o projeto [9].

4.1.1. JAVA

Considerada uma das linguagens de programação mais populares, sendo *open source*, apesar de não ser a linguagem de programação para desenvolvimento web mais utilizada, é das linguagens de eleição quando se trata de websites com grandes volumes de tráfego [9].

4.1.2. Javascript

Além de ser uma linguagem *scripting* interpretada pelo cliente, recentemente com o desenvolvimento de NodeJs [10], passou a estar presente também no lado do servidor.

Uma das principais vantagens para além da sua elevada performance, é unificar o processo de desenvolvimento numa única linguagem de programação, i.e., Javascript no lado do servidor tal como no lado do cliente.

Pode-se comparar a performance do NodeJS com soluções web populares, na seguinte figura.

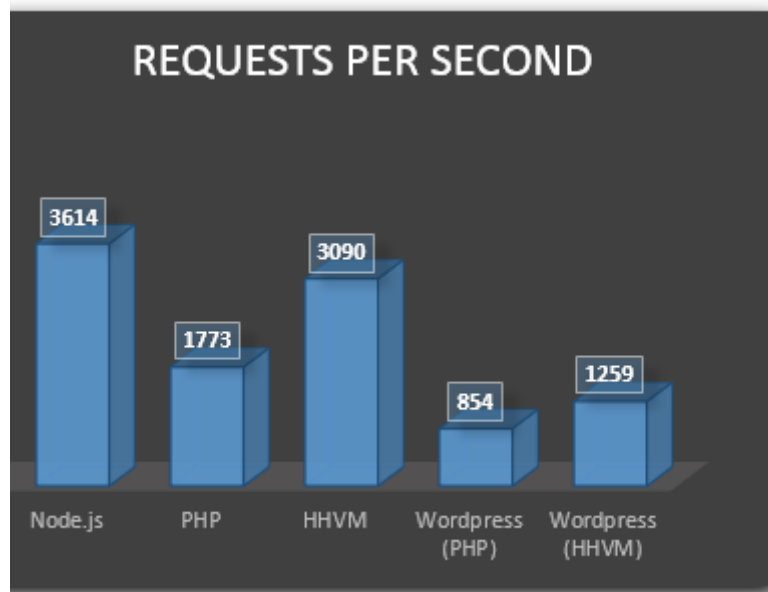


Figura 12 - Comparação Pedidos por segundo (adaptado de [9])

4.1.3. PHP

PHP trata-se de uma linguagem de script *open source*, muito utilizada em desenvolvimento web. Capaz de gerar conteúdo dinâmico, o código é interpretado no lado do servidor, gerando a página web para o cliente.

A instalação do PHP é possível na maioria dos sistemas operativos tornando-se versátil. É a linguagem de programação mais utilizada em *Backend* dos websites atuais, como apresentado na figura 13.

Server Side Programming Language



Figura 13 - Linguagens de servidor mais utilizadas (de [9])

Sendo esta a linguagem de programação definida pela organização envolvida neste trabalho, utilizou-se então uma *framework* PHP para o desenvolvimento da plataforma.

4.1.3.1. Frameworks PHP

Utilizando uma boa *framework* é possível aumentar a segurança, facilidade de manutenção e extensão da plataforma, tal como o reduzir o tempo de desenvolvimento.

As principais vantagens de utilizar uma *framework* podem ser consideradas [11] :

1. Eficiência - funcionalidades que poderiam levar horas a desenvolver, podem ser feitas em apenas alguns minutos utilizando funcionalidades pré-definidas da mesma;
2. Segurança - sendo uma *framework* popular, pode-se contar com uma grande equipa de desenvolvedores e utilizadores da mesma. Ao encontrar uma vulnerabilidade esta é rapidamente corrigida;
3. Suporte - uma boa *framework*, por norma, inclui uma documentação detalhada tal como uma grande comunidade *online* caso haja necessidade de ajuda.

Por outro lado as principais desvantagens na utilização de uma *framework* são [11]:

1. Aprendizagem - normalmente aprende-se a *framework* em si e não a linguagem por de trás da mesma;

2. Restrição - não é possível modificar o funcionamento base da mesma, ao utilizar a *framework* temos de aceitar as suas limitações e modo de funcionamento.

Apresenta-se de seguida um quadro comparativo entre as seis *frameworks* mais populares. Abordou-se apenas as *frameworks* mais populares devido a existir maior suporte por parte da comunidade, tal como existência de *plugins* às mesmas.






	 Yii2	 Zend	 Code Igniter	 Symfony 2	 CakePHP	 Phalcon
Adequado para Projectos	Qualquer Dimensão	Médios e Grandes	Qualquer Dimensão	Grandes	Pequenos e Médios	Qualquer Dimensão
Conhecimentos necessários	PHP, POO	PHP, POO	PHP, Bases POO	PHP, POO, ORM, Cosola	PHP, POO	PHP
Facilidade de Aprendizagem	Fácil	Difícil	Fácil	Difícil	Fácil	Fácil
Sistema Templating	Prado	Smarty	PHP	Twig	Smarty	Volt
Documentação	Excelente	Boa	Excelente	Compreensivo	Pobre	Excelente
Cache de pedidos	✓	✗	✓	✓	✓	✓

Figura 14 - Comparação de Frameworks PHP (adaptado de [12])

Optou-se pela *framework* Symfony2, pelas seguintes razões [13]:

- Utiliza ORM(*object-relational mapping*), o que torna a interação com a base de dados semelhante a utilizar Objetos em POO (programação orientada a objetos);
- Estável e com uma comunidade ativa, sendo esta melhorada e lançadas novas atualizações constantemente;
- Elevada segurança;
- Existência de uma grande variedade de componentes;

Já estar familiarizado com a *framework*, reduzindo assim o tempo de desenvolvimento e consequentemente o *time-to-market*.

4.2. Bases de dados

Consideramos como base de dados, uma coleção de dados seja esta de que tipo for.

No caso específico de Engenharia de Software, existem inúmeras opções no que toca à escolha da base de dados. Iremos abordar os dois tipos de base de dados mais comuns, i.e. relacionais e não-relacionais.

4.2.1. Relacionais

Bases de dados relacionais são bases de dados digitais, organizadas por um modelo relacional entre os dados (ligações tabulares) [14]. Este modelo organiza os dados numa ou mais tabelas, com uma chave primária identificando cada linha.

Os Softwares responsáveis por gerir e manter uma base de dados relacionais são chamados de Sistemas Gestores de Base de Dados (SGBD).

A linguagem *Structured Query Language* (SQL), é a linguagem desenvolvida para gerir os dados presentes nas SGBD. Estas bases de dados, são de momento o tipo mais utilizado para guardar informação [15].

4.2.2. Não-Relacionais

Existem vários tipos de bases de dados não-relacionais, estas assumem diferentes tipos de abordagem (tais como, grafos, orientadas-objeto ,etc.) [16].

Estas bases de dados, como o nome indica, não relacionam dados entre si, o que as torna extremamente flexíveis, com maior simplicidade de desenho e facilmente escaláveis [17].

Empresas como o Amazon, Google e Facebook estão a adotar este tipo de base de dados devido à sua elevada performance [18].

4.2.3. Relacionais vs. Não-Relacionais

Demonstra-se na figura 15, um quadro comparativo entre os dois tipos de bases de dados.

	SQL	NoSQL
Modelo	Relacional	Não-Relacional
	Guarda informação em Tabelas	Guarda informação de várias formas (documentos, grafos etc.)
Dados	Adequado quando todas as linhas da tabela têm as mesmas propriedades	Oferece flexibilidade, nem todo o objeto precisa guardar as mesmas propriedades
	Adicionar novas propriedades, pode requerer alterar o esquema da BD e preencher dados	Facilmente adiciona-se novas propriedades
	Adequado para dados estruturados	Adequado para dados semi-estruturados
Esquema	Rígido	Dinâmico e flexível
Expansível	Vertical	Horizontal

Figura 15 - Sql vs. NoSQL (adaptado de [19])

4.3. Conclusão

Neste capítulo, foi elaborada uma pequena pesquisa a possíveis escolhas tecnológicas para o desenvolvimento da aplicação.

Foi efetuada uma análise a possíveis tecnologias *backend* para implementação da plataforma. Sendo a restrição utilizar PHP no desenvolvimento da mesma por parte do DSI, decidiu-se utilizar uma *framework* de modo a auxiliar este processo.

Em relação às bases de dados, analisaram-se as relacionais e não-relacionais, e foi efetuada uma comparação entre ambas. Foi decidido

pelo DSI que a base de dados a utilizar seria relacional utilizando o SGBD MySQL.

Com base das restrições tecnológicas por parte da instituição, perde-se maioritariamente performance. Utilizando a tecnologia PHP invés de Javascript(NodeJS) perde-se performance e unificação de uma única linguagem de programação no lado do cliente e servidor. A escolha de uma base de dados relacional também afeta a performance e dificulta a alteração da estrutura da mesma caso exista necessidade para tal.

Capítulo 5

Desenho e Implementação

Este capítulo aborda, com base nos requisitos, a fase de planeamento, desenho tal como a fase de desenvolvimento da plataforma web.

Demonstra-se o esquema relacional da base de dados necessária para suportar o armazenamento de dados necessário para o desenvolvimento da plataforma.

Por fim apresenta-se a arquitetura da plataforma utilizada, e seus respetivos componentes.

5.1. Planeamento

Recorreu-se a um processo de desenvolvimento ágil, SCRUM, na fase de desenvolvimento.

Foi utilizada a filosofia dos modelos ágeis, mantendo assim constante contato com o Departamento de Ambiente. Sempre que uma nova funcionalidade tivesse sido implementada e testada, era então feito um *delivery*, neste caso, marcada uma reunião para testarem o Software.

O sistema foi então planeado, para ser desenvolvido incrementalmente devido a dependências lógicas, como exemplo, não poderíamos prosseguir com o desenvolvimento das estatísticas sem os movimentos estarem completos.

Decidiu-se manter a fase de testes, durante todo o processo de desenvolvimento, assim já podendo dar como terminado a implementação e teste de uma determinada funcionalidade.

Segue-se na seguinte figura o diagrama de Gantt que define o plano de desenvolvimento seguido.

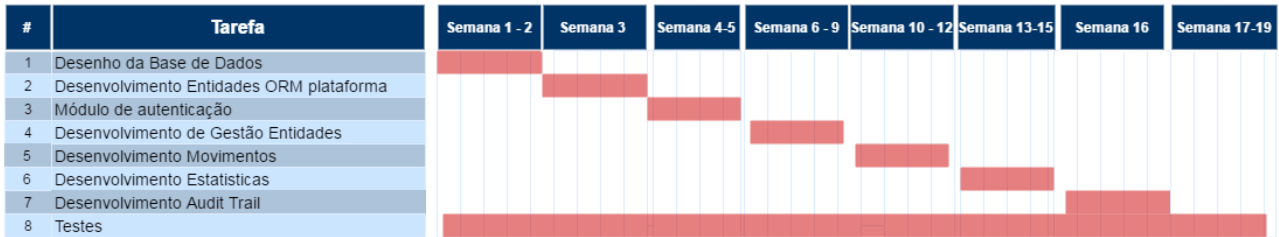


Figura 16 - Planeamento

5.2. Modelo de dados

Neste subcapítulo é demonstrado o modelo de dados utilizado para suportar a lógica de negócio apresentada.

Recorreu-se ao MySQL Workbench para o desenho da base de dados em questão. Apresenta-se uma pequena descrição a cada tabela e sua respetiva finalidade no contexto da aplicação.

Feito o desenho do diagrama ER (entidade-relação), este é utilizado como guia de implementação das respetivas Entidades ORM (*Object - Relational mapping*, i.e., converteu-se as tabelas apresentadas na figura 17 diretamente para ORM do Symfony2), assim fazendo um mapeamento direto entre o objeto e a tabela da base de dados. A figura 17 apresenta apenas as tabelas necessárias à aplicação, para a base de dados completa com as tabelas de *audit* ver Anexo A.

Após definidas as entidades ORM, toda a mudança necessária na base de dados é então efetuada na própria aplicação. É demonstrado como tal se realiza na secção 5.3.1.1.

5.2.1. Diagrama ER - Aplicação

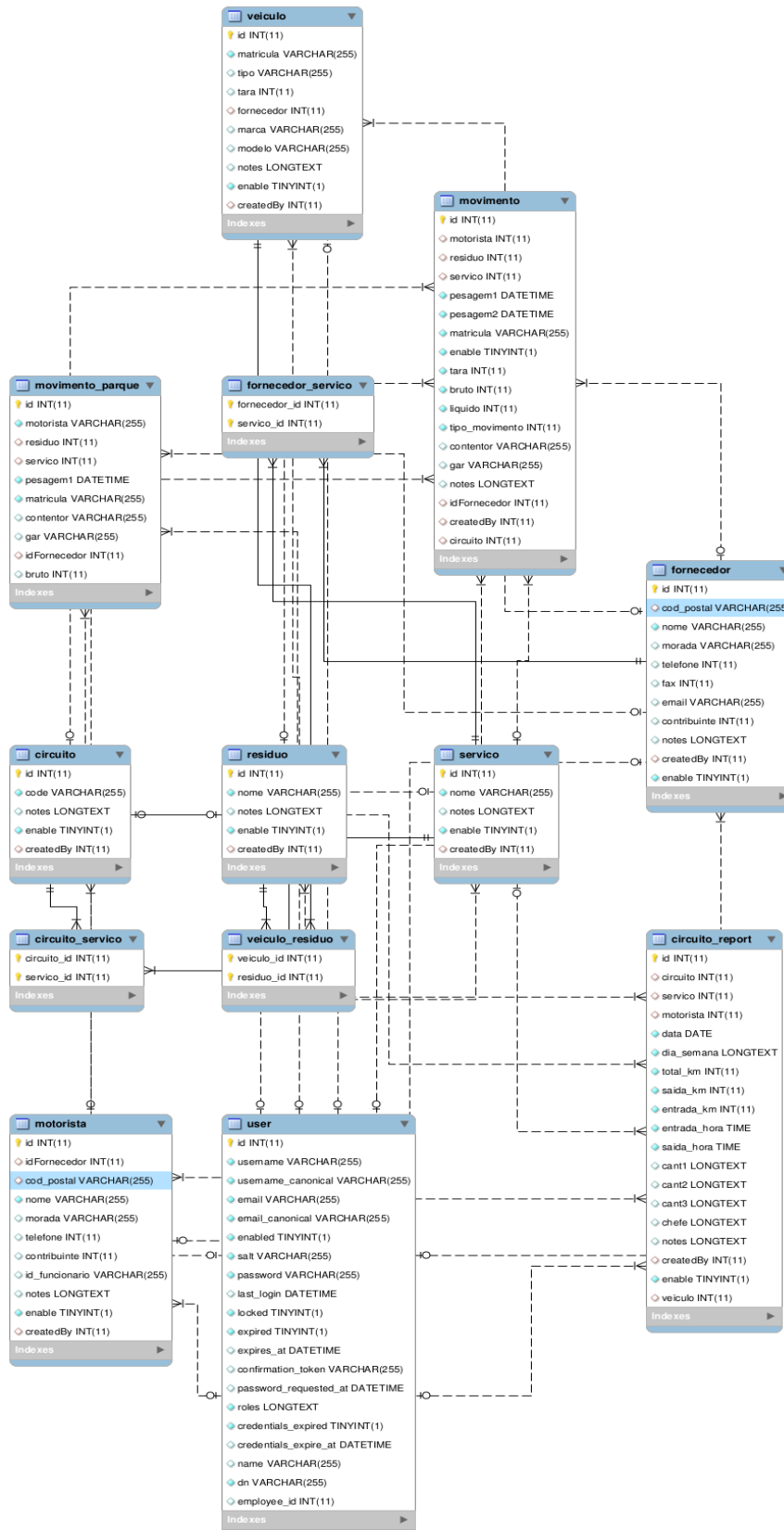


Figura 17 - Diagrama ER - Aplicação

5.2.2. Descrição das tabelas

Passamos agora a uma pequena descrição sobre a finalidade de cada uma das tabelas apresentadas anteriormente.

Tabela 57 - Descrição Tabelas ER

<i>Tabela</i>	<i>Descrição</i>
<i>User</i>	Contas dos utilizadores da plataforma
<i>Circuito</i>	Circuitos de remoção de lixo
<i>Circuito_report</i>	Relatório de remoção do Circuitos
<i>Servico</i>	Serviços que originam resíduos
<i>Circuito_servico</i>	Definição de quais os serviços podem ser associados a um circuito
<i>Movimento</i>	Movimento de entrada ou saída na estação
<i>Veículo</i>	Veículos conhecidos à CMF
<i>Veiculo_residuo</i>	Que tipo de resíduos um veículo pode transportar
<i>Fornecedor</i>	Fornecedores
<i>Fornecedor_servico</i>	Que tipo de serviços um fornecedor pode efetuar
<i>Residuo</i>	Resíduos
<i>Motorista</i>	Motoristas
<i>Movimento_parque</i>	Tabela auxiliar, para pesagem temporária, necessária para o novo software da balança

5.3. Plataforma Web

Nesta secção aborda-se o desenvolvimento da aplicação e começa-se por definir a arquitetura da aplicação.

5.3.1. Arquitetura

De forma a cumprir com o principal atributo de qualidade (i.e., facilidade de modificação), optou-se por uma arquitetura *Model-View-Controller* (MVC) [8].

Esta arquitetura é muito utilizada no desenvolvimento web, utilizando o princípio de separation of concerns [20], o que promove também a facilidade de modificação.

A arquitetura MVC é dividida em 3 camadas, Modelo, Vista e Controlador.

A *framework* utilizada para o desenvolvimento do projeto, utiliza esta arquitetura como base o que já facilita em termos de estruturação de código.

A figura 18 ilustra como essas três camadas do MVC comunicam entre si. Iremos abordar em mais detalhe cada uma das camadas nos próximos subcapítulos.

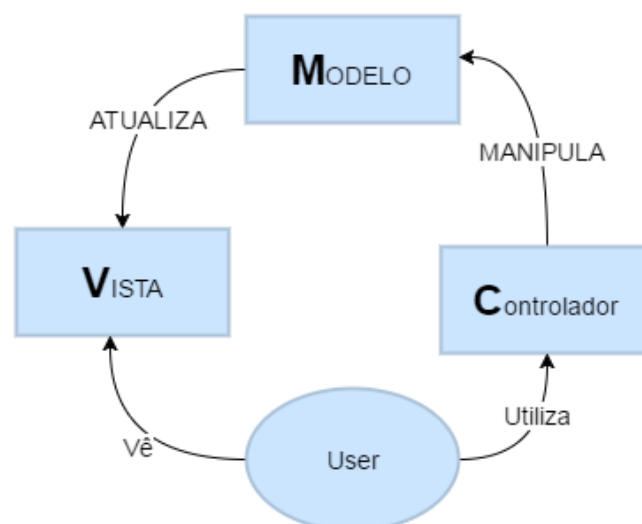


Figura 18 - Arquitetura MVC

5.3.1.1. Modelo

A camada de Modelo está associada aos dados e ao armazenamento de informação, sendo esta responsável pela manipulação de dados. A *framework* em questão utiliza ORM para manipulação de dados.

Partindo do diagrama ER anteriormente referido, demonstra-se então um pequeno exemplo de como é feita a conversão de uma tabela para formato ORM (doctrine2). Considerando como exemplo a tabela Veículo, esta contém dois tipos de ligações, 1 para N e de N para N.

```
<?php
/**
 * Veiculo
 *
 * @ORM\Table()
 * @ORM\Entity
 */
class Veiculo{
    /**
     * @ORM\Id
     * @ORM\Column(type="integer")
     * @ORM\GeneratedValue(strategy="AUTO")
     */
    private $id;

    /** @ORM\Column(type="string",nullable=false,unique=true) */
    private $matricula;

    /**
     * @ORM\ManyToOne(targetEntity="AppBundle\Entity\Fornecedor", inverse-
     dBy="veiculos")
     * @ORM\JoinColumn(name="fornecedor", referencedColumnName="id")
     */
    private $fornecedor;

    /**
     * @ORM\ManyToMany(targetEntity="AppBundle\Entity\Residuo")
     * @ORM\JoinTable(name="veiculo_residuo",
     *      joinColumns={@ORM\JoinColumn(name="veiculo_id", referencedColumnName="id")},
     *      inverseJoinColumns={@ORM\JoinColumn(name="residuo_id", referenced-
     ColumnName="id")})
     */
    private $residuo;

    /**
     * @ORM\ManyToOne(targetEntity="LDAPCMF\AuthBundle\Entity\User")
     * @ORM\JoinColumn(name="createdBy", referencedColumnName="id")
     */
    private $createdBy;

    /** @ORM\Column(type="boolean") */
    private $enable = true;
}
```

Figura 19 - Tabela Veiculo ORM

Ao definir a tabela no formato acima descrito ORM Doctrine2, e implementado os seus *getters* e *setters*, passamos a trabalhar sobre uma abstração de objetos, ao invés do uso de SQL sempre que é necessário utilizar uma operação sobre a tabela.

```
$ php app/console doctrine:schema:update --force
```

Figura 20 - Doctrine Update

Após a implementação do mesmo, basta executar o comando acima descrito na figura 20 sendo automaticamente criada a tabela “veiculo” tal como a tabela “veiculo_residuo”.

Todas as tabelas referidas, foram implementadas usando este método, apresenta-se na seguinte figura as Entidades ORM implementadas.

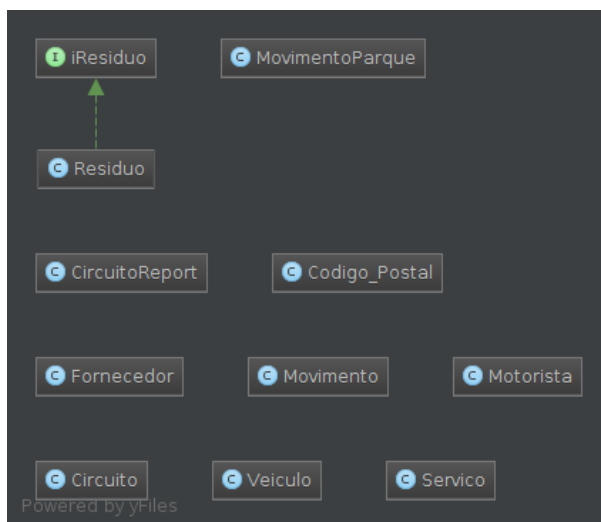


Figura 21 - ORM Diagrama

5.3.1.2. Controlador

Esta camada separa a camada de Modelo da camada Vista e é responsável por controlar o fluxo de dados do Modelo e por atualizar a Vista sempre que há mudanças nos dados.

Apresenta-se na seguinte figura o diagrama de Controladores.

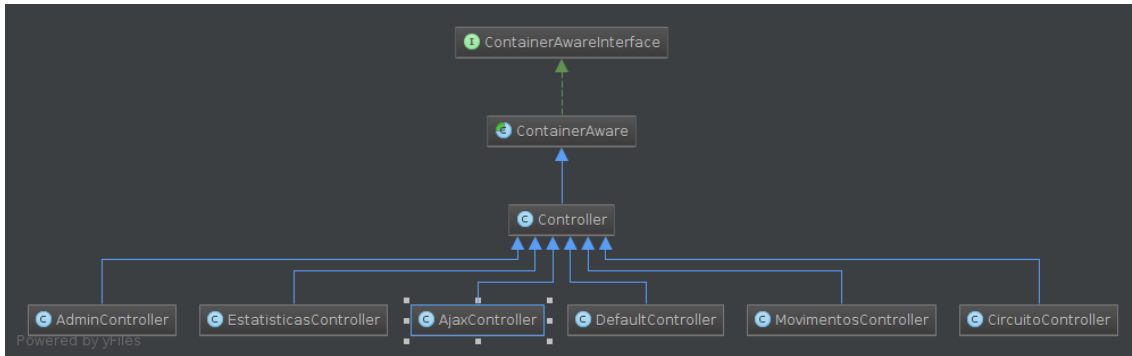


Figura 22 - Diagrama de Controladores

As classes de controladores apresentadas na figura anterior estão organizadas por tipo de funcionalidade, i.e., *AdminController* está responsável pelas ações de administração, tal como é apresentado no capítulo 3. Assim sendo cada controlador apresenta um conjunto de funcionalidades únicas associadas ao tipo de tarefa a que este pertence.

Foi necessário a criação de um controlador extra para pedidos Ajax, sempre que exista necessidade de efetuar um pedido Ajax este é o responsável por tal.

5.3.1.3. Vista

Camada de apresentação ao utilizador, apenas contém lógica de apresentação. As *views* são os ficheiros HTML, JavaScript com o qual o utilizador interage.

A *framework* utiliza um *template engine*, chamado de Twig, utilizado para o desenvolvimento de todas as vistas da aplicação.

Utilizando um sistema de *template*, consegue-se remover qualquer tipo de código PHP no HTML, as três principais vantagens de utilizar o twig são [21]:

1. Velocidade, twig compila os templates para php otimizado;
2. Segurança, twig contem um modo *sandbox*, que avalia todo o código de *template*;
3. Flexibilidade, twig contem vários *parsers*.

5.3.2. Plugins e Extras

De forma a otimizar o processo de desenvolvimento, foram introduzidos alguns *plugins* na plataforma, conhecidos como *bundles*, além dos já presentes no próprio Symfony2.

5.3.2.1. Plugins Backend

Utilizaram-se os seguintes *plugins* em Backend:

- FOSUserBundle: adiciona uma *mini-framework* para gestão de utilizadores, desde a registo, recuperar password e gerir permissões;
- FR3DLdapBundle: implementa toda a lógica de interação com servidores LDAP, sendo este o método de gestão de utilizadores utilizado pela CMF;
- IvoryCKEditorBundle: integração do CKEditor(editor de texto, estilo MS Word);
- SimpleThingsEntityAuditBundle: sistema de *audit trail*, registando assim todas as mudanças nas entidades selecionadas;
- KnpSnappyBundle: conversão de HTML para PDF;
- JMSSerializerBundle: serializador de dados, como por exemplo objetos de Entidades para formato JSON.

5.3.2.2. Plugins Frontend

Utilizou-se um *package manager* para instalação e controlo de dependências de *plugins* no lado do cliente, bower [22].

Os *plugins* utilizados são majoritariamente de assistência a manipulação de DOM e melhorias visais à aplicação, os principais *plugins* em questão são:

- jQuery: biblioteca Javascript, que facilita a manipulação de elementos DOM, Ajax com uma extensa documentação. É a biblioteca Javascript mais utilizada no mundo [23];
- Twitter Bootstrap: *framework* popular de HTML, CSS e Javascript utilizada no desenvolvimento de projetos web sendo o principal foco torna-los escaláveis ao tamanho da janela em questão;
- HighCharts: biblioteca Javascript responsável por criação de diversos tipos de gráficos;
- Moment.js: biblioteca Javascript de manipulação, validação, *parsing* e visualização de datas.

Adicionalmente, foi utilizado um *template* para o aspeto visual da aplicação AdminLTE, sendo este utilizado por várias aplicações internas da CMF mantendo assim coerência entre o aspeto das aplicações.

5.3.3. Diagrama de componentes

O principal objetivo do diagrama de componentes é demonstrar a relação das estruturas que interagem com o sistema [24].

Na figura 23, é possível observar como estão ligadas as componentes ao sistema desenvolvido.

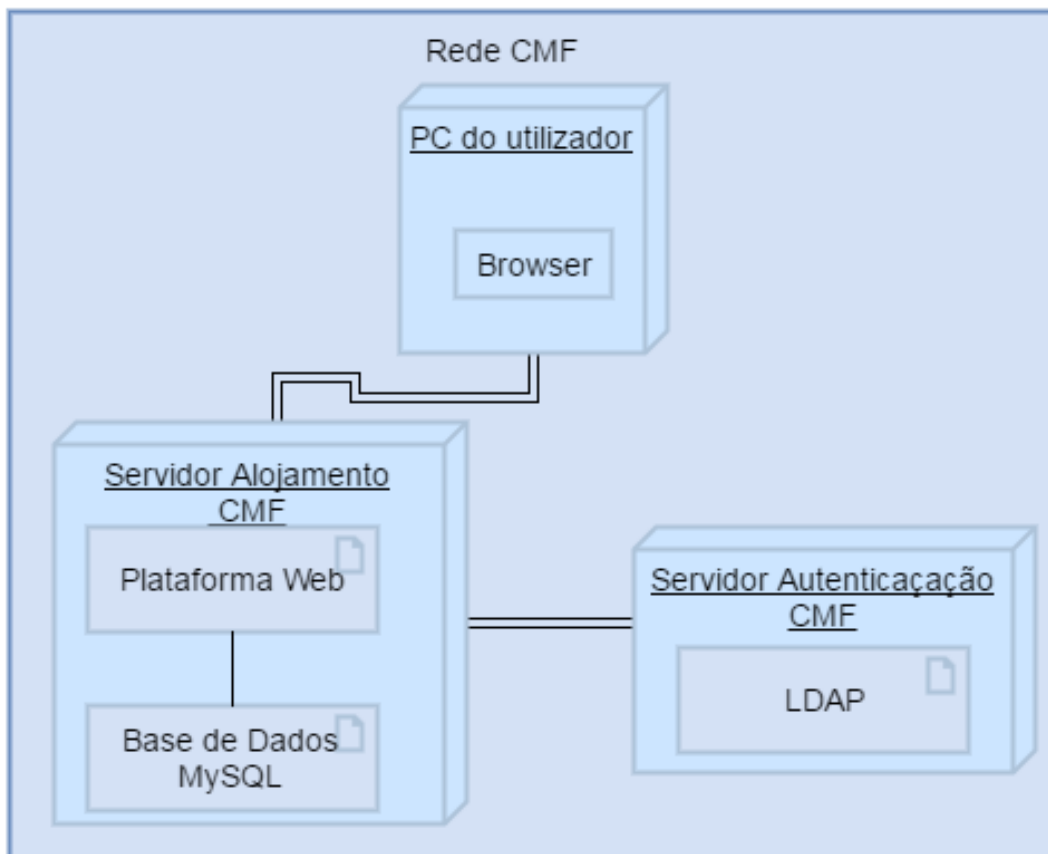


Figura 23 - Diagrama de Componentes

Toda a infraestrutura encontra-se na rede interna da CMF, sendo esta inacessível externamente, uma vez ligado à rede é possível o utilizador aceder à plataforma web utilizando o web *browser* do seu computador. A plataforma web e a base de dados da mesma, encontram-se hospedadas num dos servidores da CMF.

A plataforma web comunica com o servidor de LDAP de forma a validar todo o tipo de autenticação por parte da mesma.

5.3.4. Aplicação

Nesta secção apresenta-se uma pequena amostra da aplicação desenvolvida para o Departamento de Ambiente.

O utilizador ao ligar-se ao endereço da plataforma esta requer que este se identifique. Uma vez autenticado é apresentada a plataforma com as respetivas funcionalidades a que o utilizador tem acesso. A seguinte figura ilustra a autenticação da aplicação.



The image shows a login interface for 'FUNCHAL Resíduos Sólidos Urbanos'. At the top, there is a logo consisting of a stylized location pin above three wavy lines, with the text 'FUNCHAL' and 'Resíduos Sólidos Urbanos' below it. The main content area is a white box with a light gray border. It contains the heading 'Autentique-se perante a aplicação'. Below this, there are two input fields: 'Conta CMF' with a user icon and 'Password' with a lock icon. At the bottom left, there is a checkbox labeled 'Manter Sessão iniciada'. At the bottom right, there is a blue button labeled 'Sign In'.

Figura 24 - Menu de autenticação

Após autenticado corretamente perante a aplicação, e consequentemente ao serviço LDAP da CMF, o utilizador é então reencaminhado para a página principal da plataforma web, sendo esta vista como um género de *dashboard*, como é visualizado na figura 25.

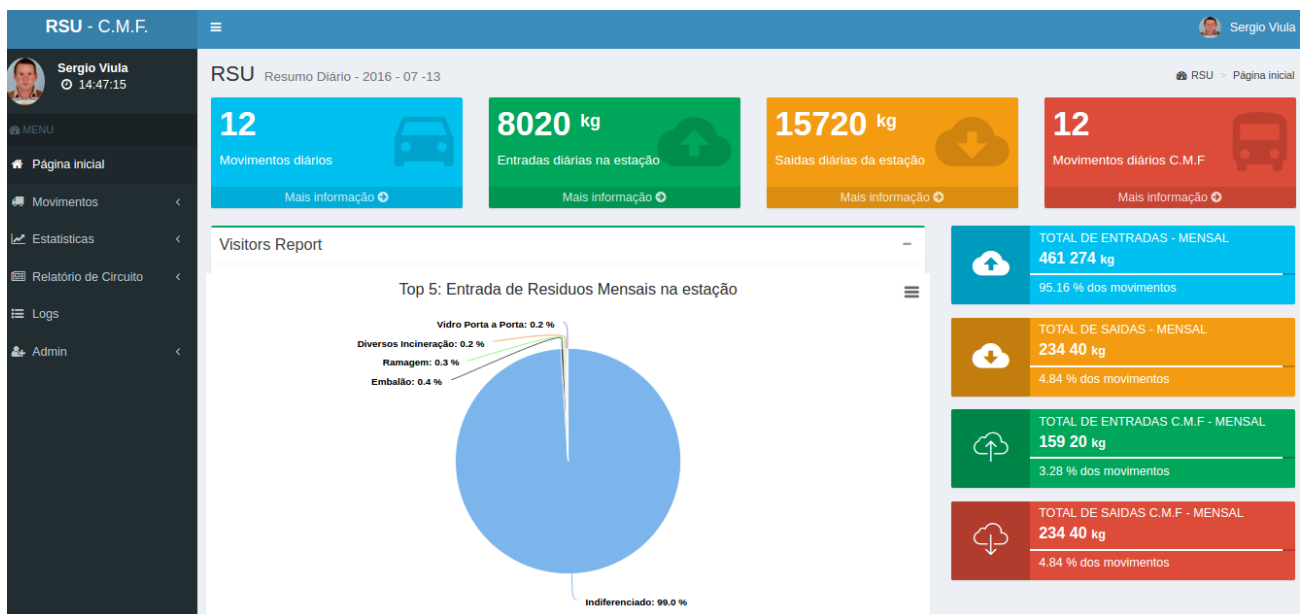


Figura 25 - Página inicial plataforma web

Uma vez na página inicial o utilizador pode então selecionar as ações que deseja realizar utilizando o menu.

De forma a limitar o tamanho do documento apresenta-se de seguida somente as partes principais da plataforma web.

Acedendo ao painel de *Admin* é possível adicionar entidades à plataforma, como definido no capítulo 3. Considerando como exemplo a entidade de Fornecedor, segue-se o formulário necessário para adicionar um novo fornecedor à plataforma.

Informação sobre o Fornecedor

Ativo

Nome do Fornecedor

Morada do Fornecedor

Número de contribuinte

Código Postal

Telefone do Fornecedor

Fax do Fornecedor

Email do Fornecedor

Serviços permitidos

Banco Alimentar C M C Lobos Capa Cartão Capa Indiferenciado

Caracterização Carnaval Casa Saúde São João Deus Cemitério

Figura 26 - Adicionar Fornecedor

Os registos de movimentos da estação são maioritariamente inseridos pelo Software da balança, é possível na plataforma efetuar operações de CRUD sobre os mesmos. Segue-se uma pequena amostra da listagem de movimentos do dia 13 de julho de 2016 na seguinte figura.

Listagem

Entre as Datas

Mostrar registos Procurar Por Matrícula:

Talão	Nº Talão	Motorista	Residuo	Serviço	Matricula	1º Pesagem	Peso Liquido	Fluxo	Verificado	
	6704	Raúl Camacho	Metais	R H Manhã	18-99-VR	2016-07-13 10:28:00	580 Kg	Entrada	●	
	6703	José Manuel Fernandes	Embalão	R H Manhã	62-EG-63	2016-07-13 10:27:36	720 Kg	Entrada	●	
	6702	António Serrão	Papel Porta a Porta	R H Manhã	26-AO-86	2016-07-13 10:11:30	280 Kg	Entrada	●	

Figura 27 - Listagem de movimentos

De forma a melhorar a usabilidade da plataforma foram adicionadas funcionalidades na própria listagem, sendo então possível visualizar mais detalhadamente a informação do registo, exportar o talão em formato pdf e editar o registo.

O utilizador ao clicar no ícone de pré-visualização presente na coluna “Nº Talão” é apresentado um *popup* com a informação mais detalhada sobre o registo como é possível conferir na figura 28.

The screenshot shows a window titled 'Movimento' with a close button (X) in the top right corner. The main content area has a green header bar with the text 'Talão Número: 6704'. Below this, the information is organized into two columns:

Fornecedor: CM Funchal	Resíduo: Metais
Serviço: R H Manhã	Matricula: 18-99-VR
Motorista: 10390 - Raúl Camacho	Peso Líquido: 580 Kg
1ª Pesagem: 2016-07-13 10:07	2ª Pesagem: 2016-07-13 10:07
Fluxo: Entrada	Circuito:
Contentor: 0	Gar: 0
Criado Por: semivi	

At the bottom of the window, there are two buttons: 'Fechar' (Close) on the left and 'Editar' (Edit) on the right.

Figura 28 - Popup informação de registo

Caso o utilizador deseje exportar o talão do registo é possível fazê-lo também a partir da listagem, esta informação é então exportada em formato pdf como demonstra a figura 29.

Câmara Municipal do Funchal
Praça do Município
9004-512 Funchal
Nº Cont: 505565285
Tel: 291 211 000
cmf@cm-funchal.pt

Talão de Pesagem: 6704

Fornecedor: CM Funchal	Matricula: 18-99-VR
Morada: Rua Mestre Sidónio nº18	
Telefone: 291743431	
Contribuinte: 511217315	

Resíduo: Metais	Data/Hora: 2016-07-13 10:28:00	Tara: 3050 kg
Serviço: R H Manhã	Movimento: Entrada	Bruto: 3630 Kg
Ref. Contentor: 0	Gar: 0	Líquido: 580 kg
Observações:		

Figura 29 - Talão de registo em formato pdf

Em termos de estatísticas apresenta-se o gráfico de resumo mensal do mês de junho de 2016, sendo possível selecionar os fornecedores que deseja visualizar informação, como é demonstrado na seguinte figura.

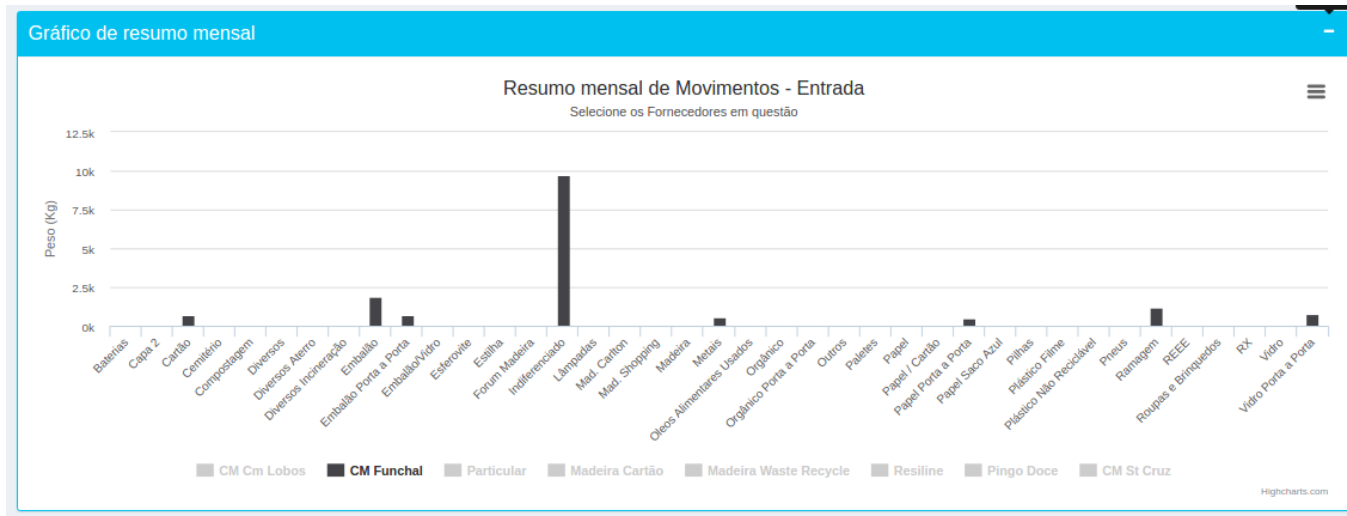


Figura 30 - Gráfico resumo mensal

Apresenta-se também a tabela de resumo anual de saídas de viaturas da estação com contentor aberto, sendo esta visível na figura 31.

Tabela de resumo - Contentor Aberto

Excel PDF Procurar:

Mês	Peso (kg)	Nº Saídas
Jan	0 kg	0
Fev	0 kg	0
Mar	0 kg	0
Abr	0 kg	0
Mai	0 kg	0
Jun	809550 kg	135
Jul	10840 kg	2
Ago	0 kg	0
Set	0 kg	0
Out	0 kg	0
Nov	0 kg	0
Dez	0 kg	0
-	-	-
Sub-Total	820390 kg	137
Mês	Peso (kg)	Nº Saídas

Mostrando de 1 até 14 de 14 registos

Figura 31 - Resumo anual contentor aberto

5.4. Conclusão

Neste capítulo abordou-se o desenho e implementação da plataforma desenvolvida, assim como o processo e planeamento adotados, o modelo de dados e o desenvolvimento da plataforma.

Utilizou-se o processo de desenvolvimento ágil SCRUM, assim definindo os artefactos a desenvolver em ordem temporal recorrendo a um diagrama de Gantt. Após um *sprint* estar concluído este era revisto tanto pelo DSI como pelo Departamento de Ambiente.

Elaborou-se um diagrama de entidade-relação de forma a armazenar toda a informação necessária à plataforma. Este diagrama foi utilizado como guia de implementação das correspondentes entidades ORM.

No ato de desenvolvimento da plataforma, optou-se por uma arquitetura MVC, assim promovendo a facilidade de modificação, pois o produto será gerido e estendido pelo DSI no termino do estágio. Procedeu-se de seguida à descrição das camadas respetivas do MVC (i.e., Modelo, Vista e Controlador).

Por fim é apresentado o diagrama de componentes, sendo possível visualizar a estrutura física em que a plataforma de encontra e compreender o fluxo de interação entra os vários componentes.

Capítulo 6

Testes de usabilidade

No capítulo anterior foi descrito a fase de implementação da plataforma. Além dos testes conduzidos durante toda a fase de desenvolvimento, foram conduzidos testes de usabilidade no término do projeto.

Os testes de usabilidade visam melhorar a facilidade de uso de um determinado produto. Todos os testes de usabilidade partilham estas cinco características [25] :

1. Objetivo principal é melhorar a usabilidade de um produto;
2. Os participantes são utilizadores reais;
3. Os participantes realizam tarefas reais;
4. Observar e registar o que os participantes fazem e dizem;
5. Analisar os dados, encontrar problemas de usabilidade e corrigir os mesmos.

6.1. Método de teste

Existem vários métodos de testes de usabilidade [26] (como por exemplo, *expert review*, *A/B testing*, etc.).

Devido à indisponibilidade do Departamento de Ambiente na fase de testes, foi acordado então com o DSI que poderia utilizar o método de *Hallway Testing*.

O método *Hallway testing* é um método rápido e de custo reduzido de testes de usabilidade. Este método recruta utilizadores aleatórios/a passar no corredor (daí o seu nome *hallway*). Segundo pessoas experientes em testes de usabilidade, este método revela até 95% dos problemas de usabilidade de um produto [27].

6.2. Preparação dos testes

A preparação dos testes, pode ser considerado o passo mais importante de um bom teste de usabilidade. Previamente a qualquer tipo de recrutamento de participantes definiu-se uma lista de tarefas.

Começou-se por definir numa única frase as tarefas desejadas para o teste, tal como se segue:

1. Crie um fornecedor;
2. Crie um registo de movimento;
3. Modifique um registo movimento;
4. Adicione um utilizador à plataforma;
5. Exporte a tabela de estatísticas do mês atual de fornecedores e resíduos transportados.

Estas foram consideradas as tarefas principais da plataforma, daí terem sido as escolhidas para realizar o teste.

Uma estratégia para envolver o utilizador é a criação de cenários para cada tarefa [28]. Seguindo esta estratégia, criou-se então um cenário a cada tarefa acima descrita.

6.2.1. Tarefa 1 – crie um fornecedor

Cenário: imagine que está a trabalhar no Departamento de Ambiente, e aparece uma pessoa que deseja registar a sua empresa como fornecedor de resíduos. Com base no desejo do indivíduo, você aceita os documentos com informação sobre o mesmo e cria um novo fornecedor na plataforma do RSU.

6.2.2. Tarefa 2 – crie um registo de movimento

Cenário: imagine que está a trabalhar no Departamento de Ambiente, e por razões desconhecidas o Software de registo de movimentos da estação encontra-se com problemas. Perante isto, o operador da estação aponta todos os registos numa folha de papel. Recebendo esta informação, tente inserir um registo de movimento na plataforma do RSU.

6.2.3. Tarefa 3 - modifique um registo movimento

Cenário: imagine que está a trabalhar no Departamento de Ambiente, e foi-lhe comunicado que um registo de movimento está com dados inválidos. Modifique então o tipo de resíduo de um movimento.

6.2.4. Tarefa 4 – adicione um utilizador à plataforma

Cenário: imagine que é o chefe do Departamento de Ambiente, e foi contratado recentemente um novo colaborador para a estação. Este novo colaborador irá ser responsável por verificar os dados da Plataforma web tal como imprimir talões de movimentos. Com base nisto, adicione um novo utilizador à plataforma com o nome de utilizador “jhchab”.

6.2.5. Tarefa 5 - exporte a tabela de estatísticas do mês atual, fornecedores - resíduos

Cenário: imagine que trabalha no Departamento de Ambiente, e o seu chefe pede-lhe para imprimir a tabela de resumo mensal dos movimentos da estação, mais concretamente, a quantidade de resíduos transportada por cada fornecedor. Exporte a tabela de resumo mensal para formato pdf para mais tarde poder imprimir.

6.3. Participantes

O recrutamento de participantes foi feito na CMF. Foram recrutadas 5 pessoas para realizarem os testes definidos.

Segundo Nielsen e Landauer (1993) cinco utilizadores são suficientes para descobrir 85% dos problemas de usabilidade [29]. Segundo o gráfico apresentado na figura 32, é possível visualizar a distribuição entre o número de participantes e os problemas de usabilidade encontrados.

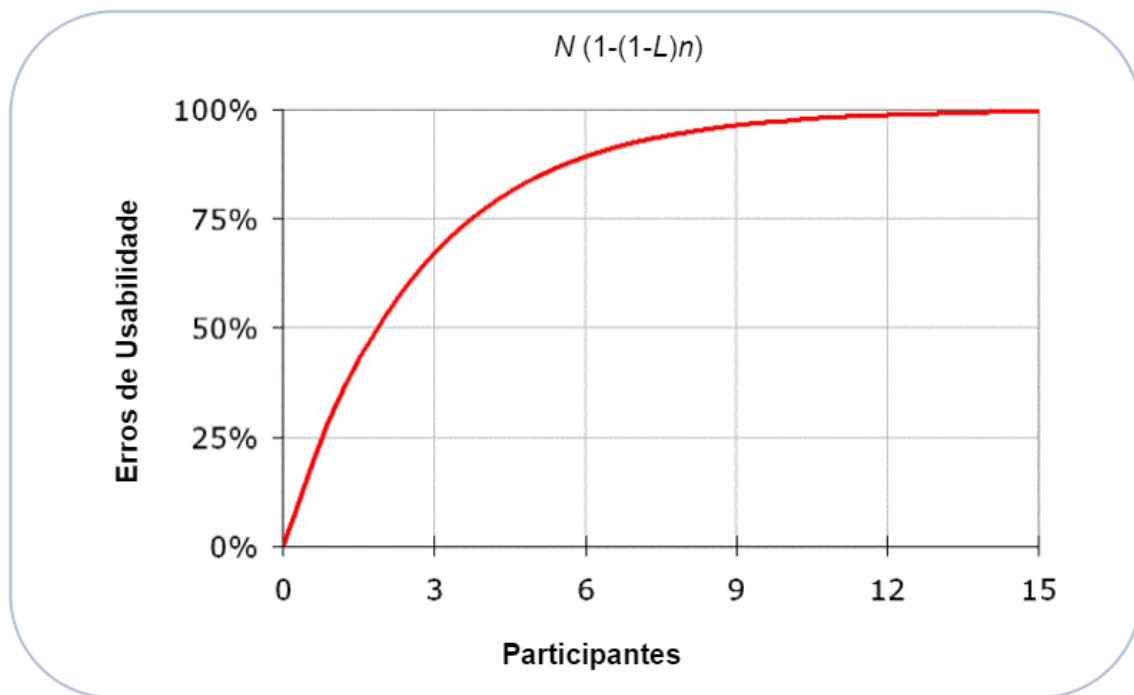


Figura 32 – Gráfico de erros de usabilidade (adaptado de [30])

Com base na relação tempo/benefício seguiu-se então a regra de Nielsen dos cinco utilizadores para realizarem o teste.

Dos 5 participantes recrutados, 2 deles do sexo feminino com idades entre 30-40 anos e 3 do sexo masculino com idades entre 34-45 anos. Todos os participantes pertencem à CMF, sendo dois deles ex-membros do de Departamento de Ambiente.

6.4. Resultados

Os testes de usabilidade foram avaliados em duas vertentes, Medidas Qualitativas e Medidas Quantitativas [25].

Em termos de dados qualitativos considerou-se os seguintes pontos:

- Comentários dos participantes enquanto efetuavam as tarefas;
- Ações dos participantes enquanto realizavam cada tarefa;
- Comentários e perguntas adicionais por parte dos participantes.

Relativamente a dados quantitativos foi medido o tempo necessário para completar cada tarefa.

Apresenta-se de seguida os resultados para cada uma das tarefas realizadas.

6.4.1. Resultado tarefa 1 – crie um fornecedor

Quando foi solicitado a criação de um fornecedor, os participantes ao olhar o menu, não encontrando a opção de “Fornecedor”, dirigiram-se para a opção de *Admin*. Após selecionar a opção, muito rapidamente selecionaram Fornecedor->Adicionar.

Todos os participantes conseguiram rapidamente completar esta tarefa, sem qualquer problema ou até mesmo comentários.

6.4.2. Resultado tarefa 2 – crie um registo de movimento

Ao realizar esta tarefa, os utilizadores digiram-se rapidamente à opção de Movimentos e acederam ao formulário de registo.

Um dos utilizadores ficou confuso ao realizar a tarefa, porque certos campos se encontravam bloqueados. A razão dos mesmos estarem bloqueados, é obrigar o utilizador selecionar apenas situações definidas pelas restrições da plataforma (por exemplo, apenas pode selecionar um serviço após ter selecionado um fornecedor, pois os fornecedores têm uma lista própria de serviços que podem fornecer).

Uma possível melhoria seria a utilização de um *wizard*, substituindo assim o formulário atual para utilizadores não familiarizados com o domínio do problema. Esta mudança poderia aumentar o tempo de realizar a tarefa, logo decidiu-se manter o formulário inicial.

6.4.3. Resultado tarefa 3 – modifique um registo de movimento

Os utilizadores não demonstraram qualquer tipo de problema ao abrir a listagem de movimentos. Contudo, demoraram um pouco a identificar o botão de Editar modificando assim o movimento.

Foi adicionado um *tooltip* ao botão de Editar assim adicionando uma legenda visível ao botão indicando a sua funcionalidade. Foi seguido este método nos ícones de pré-visualização rápida de movimento e de exportação pdf.

6.4.4. Resultado tarefa 4 – adicione um utilizador à plataforma

Esta tarefa realizou-se muito rapidamente, talvez devido os participantes já estarem familiarizados com o menu de *admin*. Esta tarefa foi concluída facilmente por todos os utilizadores.

6.4.5. Resultado tarefa 5 – exporte tabela de estatísticas

Os participantes concluíram esta tarefa muito rapidamente. Dirigiram-se diretamente ao menu de Estatísticas e selecionaram a opção correta. Uma vez na página de estatísticas clicaram facilmente no botão de PDF embebido na tabela.

6.5. Conclusão

Neste capítulo abordou-se os testes de usabilidade realizados à plataforma.

Abordou-se primeiro o método de teste utilizado para realizar os mesmos. Decidiu-se utilizar o método de *Hallway testing*, devido à indisponibilidade da chefia do Departamento de Ambiente. Segundo pessoas experientes em testes de usabilidade este método revela até 95% de problemas de usabilidade num produto [27].

Após definido o tipo de teste a realizar, foram então planeadas as tarefas que os participantes viriam a executar durante o teste. Começou-se então por definir as principais tarefas da plataforma, segundo os requisitos do Departamento de Ambiente. A cada tarefa foi escrito um cenário de modo ao participante compreender o contexto em que está inserido.

Uma vez definidas as tarefas e elaborados os cenários, foram recrutados participantes para o teste.

Com base dos resultados observados, foi possível analisar a usabilidade da plataforma. O principal problema identificado foi o de modificação de registos, pois apenas existia um ícone, sem legenda associada, o que tornava complicado a edição de registos para quem utiliza a plataforma pela primeira vez.

Acredita-se que a plataforma respeita de forma razoável o requisito não-funcional definido no capítulo 3, i.e., o atributo de qualidade usabilidade, no entanto, para poder garantir uma boa usabilidade está-se ciente da necessidade da realização de estudo mais aprofundada

Capítulo 7

Conclusão

A plataforma web foi realizada pela necessidade do Departamento de Ambiente da CMF. Estudou-se o funcionamento da estação antes de realizar qualquer tipo de especificação, compreendo assim primeiro o domínio do problema e possíveis melhorias ao mesmo. Após concluída essa análise, passou-se à especificação onde foram identificados os *stakeholders* e requisitos necessários. Passou-se de seguida à fase de pesquisa sobre possíveis tecnologias para realizar a plataforma, seguida da fase de desenho e implementação da mesma. Após concluída a implementação, realizaram-se testes de usabilidade melhorando assim aspetos da mesma. Concluído o projeto, faz-se então uma análise de todo o trabalho realizado e indicam-se perspetivas futuras acerca do mesmo.

7.1. Trabalho realizado

O trabalho realizado durante o estágio teve como objetivo a criação de uma plataforma web, tendo este a supervisão do Dr. João Miguel Figueira Gomes orientador da CMF e do Prof. Doutor José Luís Cardoso da Silva orientador da Universidade da Madeira. O projeto surgiu de a necessidade do Departamento de Ambiente necessitar de uma plataforma web que permitisse gerir os movimentos da estação.

Na fase inicial do estágio foi feita uma análise ao funcionamento da estação compreendendo assim o domínio do problema. Estudou-se cuidadosamente o fluxo de toda a estação desde o registo do movimento até às estatísticas elaboradas.

Nesta fase levantaram-se algumas questões sobre o Software de registo da balança, o que levou ao desenvolvimento de um novo Software da balança por parte do DSI.

Uma vez compreendido o domínio do problema, e após algumas reuniões foi então seguida a fase de especificação. Nesta fase identificaram-se os principais *stakeholders* do projeto criando assim um mapa de utilizadores. Após a análise dos stakeholders do projeto seguiu-se a especificação dos requisitos funcionais, requisitos não-funcionais e restrições tecnológicas.

Foram definidas pelo DSI as tecnologias nas quais o projeto teria de ser realizado, sendo estas PHP e MySQL. Por motivos de autoanálise efetuou-se um estudo às possíveis tecnologias para a realização do projeto, e uma comparação entre as mesmas. Com base na restrição tecnológica definida pelo DSI em que a plataforma teria de ser desenvolvida em PHP, elaborou-se uma análise às *frameworks* mais populares. Foi então decidido utilizar a *framework* Symfony2. Após definidas as tecnologias passou-se então à fase de desenho e implementação da plataforma.

A solução desenvolvida utiliza a arquitetura MVC, tornando-a de fácil manutenção e extensão futura.

Após o desenvolvimento da plataforma web, foram realizados testes de usabilidade tentando assim encontrar melhorias e possíveis problemas de usabilidade com a mesma. Foi possível identificar um problema de usabilidade e melhorá-lo facilmente.

Concluído todo o processo de desenvolvimento e de testes, consegue-se dar por cumprido todos os objetivos inicialmente propostos. A plataforma satisfaz todos os requisitos do Departamento de Ambiente, e encontra-se em modo de produção para os membros do mesmo.

7.2. Trabalho futuro

Como referido ao longo deste documento, as funcionalidades desejadas para a plataforma foram implementadas na totalidade. Esta encontra-se preparada para fácil modificação devido a arquitetura MVC tal como fácil extensão.

Em termos de trabalho futuro para a plataforma, poderiam ser adicionadas novos tipos de estatísticas caso haja necessidade futura.

Devido ao longo número de registos, seria melhor opção optar por uma base de dados Não-Relacional, como por exemplo MongoDB, prevenindo assim possíveis problemas de performance no acesso à base de dados.

Uma funcionalidade extra, embora não necessária, seria o envio mensal de resumos da estação aos chefes de serviço, criando assim uma *newsletter* à plataforma.

Referências

- [1] I. Jacobson, I. Spence, e B. Kerr, «Use-case 2.0», *Commun ACM*, vol. 59, n. 5, pp. 61–69, Abr. 2016.
- [2] W. Maalej, Z. Kurtanović, e A. Felfernig, «What stakeholders need to know about requirements», em *2014 IEEE 4th International Workshop on Empirical Requirements Engineering (EmpiRE)*, 2014, pp. 64–71.
- [3] M. Glinz e R. J. Wieringa, «Guest Editors' Introduction: Stakeholders in Requirements Engineering», *IEEE Softw.*, vol. 24, n. 2, pp. 18–20, Mar. 2007.
- [4] M. E. Shin e G.-J. Ahn, «UML-based representation of role-based access control», em *IEEE 9th International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, 2000. (WET ICE 2000). Proceedings*, 2000, pp. 195–200.
- [5] R. Biddle, J. Noble, e E. Tempero, «Essential Use Cases and Responsibility in Object-oriented Development», em *Proceedings of the Twenty-fifth Australasian Conference on Computer Science - Volume 4*, Darlinghurst, Australia, Australia, 2002, pp. 7–16.
- [6] D. Kulak e E. Guiney, *Use Cases: Requirements in Context*. Addison-Wesley, 2012.
- [7] M. Glinz, «On Non-Functional Requirements», em *15th IEEE International Requirements Engineering Conference (RE 2007)*, 2007, pp. 21–26.
- [8] L. Bass, P. Clements, e R. Kazman, *Software Architecture in Practice*, 3 edition. Upper Saddle River, NJ: Addison-Wesley Professional, 2012.
- [9] «Top 5 Programming Languages Used In Web Development : Programming and Development Blog | Stone River eLearning». [Em linha]. Disponível em: <http://blog.stoneriverelearning.com/top-5-programming-languages-used-in-web-development/>. [Acedido: 04-Ago-2016].
- [10] «Node.js». [Em linha]. Disponível em: <https://nodejs.org/en/>. [Acedido: 09-Set-2016].
- [11] «Advantages and Disadvantages of Frameworks», | *Vizteams*, 25-Nov-2014. [Em linha]. Disponível em: <http://www.vizteams.com/blog/advantages-and-disadvantages-of-frameworks/>. [Acedido: 04-Ago-2016].
- [12] S. Technologies, «Comparison of the top 6 PHP frameworks», *Suyati Technologies*, 14-Abr-2015. [Em linha]. Disponível em: <http://suyati.com/comparison-of-the-top-6-php-frameworks/>. [Acedido: 16-Ago-2016].
- [13] A. Monus, «10 PHP Frameworks For Developers – Best Of», *HKDC*. [Em linha]. Disponível em: <http://www.hongkiat.com/blog/best-php-frameworks/>. [Acedido: 04-Ago-2016].
- [14] «A relational model of data for large shared data banks». [Em linha]. Disponível em: <http://dl.acm.org/citation.cfm?id=362685>. [Acedido: 05-Ago-2016].
- [15] «DB-Engines Ranking - popularity ranking of database management systems». [Em linha]. Disponível em: <http://db-engines.com/en/ranking>. [Acedido: 05-Ago-2016].
- [16] «IEEE Xplore Abstract - Will NoSQL Databases Live Up to Their Promise?» [Em linha]. Disponível em: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5410700. [Acedido: 05-Ago-2016].
- [17] «Amazon DynamoDB – a Fast and Scalable NoSQL Database Service Designed for Internet Scale Applications - All Things Distributed». [Em linha]. Disponível em:

- <http://www.allthingsdistributed.com/2012/01/amazon-dynamodb.html>. [Acedido: 05-Ago-2016].
- [18] M. G. Institute, J. Manyika, e M. Chui, *Big data: The next frontier for innovation, competition, and productivity*. Lexington, KY: McKinsey Global Institute, 2011.
- [19] «NoSQL vs SQL». [Em linha]. Disponível em: <https://azure.microsoft.com/en-us/documentation/articles/documentdb-nosql-vs-sql/>. [Acedido: 05-Ago-2016].
- [20] M. Aksit, «Separation and Composition of Concerns in the Object-oriented Model», *ACM Comput Surv*, vol. 28, n. 4es, Dez. 1996.
- [21] «Introduction - Documentation - Twig - The flexible, fast, and secure PHP template engine». [Em linha]. Disponível em: <http://twig.sensiolabs.org/doc/intro.html>. [Acedido: 08-Ago-2016].
- [22] «Bower». [Em linha]. Disponível em: <https://bower.io/>. [Acedido: 08-Ago-2016].
- [23] Rami Sayar, «Top JavaScript Frameworks, Libraries and Tools and When to Use Them», *SitePoint*, 23-Nov-2015. [Em linha]. Disponível em: <https://www.sitepoint.com/top-javascript-frameworks-libraries-tools-use/>. [Acedido: 11-Ago-2016].
- [24] Miles e Hamilton, *Learning UML 2.0*, 1 edition. Beijing ; Sebastopol, CA: O'Reilly Media, 2006.
- [25] J. S. Dumas e J. Redish, *A Practical Guide to Usability Testing*. Intellect Books, 1999.
- [26] D. M. Y. Ivory, «Usability Testing Methods», em *Automated Web Site Evaluation*, Springer Netherlands, 2003, pp. 23–37.
- [27] «The Joel Test: 12 Steps to Better Code - Joel on Software». [Em linha]. Disponível em: <http://www.joelonsoftware.com/articles/fog0000000043.html>. [Acedido: 18-Ago-2016].
- [28] Stefan Rössler, «How to Write Better Tasks to Improve Your Usability Testing», *Userbrain Blog*, 17-Abr-2015. [Em linha]. Disponível em: <https://userbrain.net/blog/write-better-tasks-to-improve-usability-testing>. [Acedido: 19-Ago-2016].
- [29] J. Nielsen e T. K. Landauer, «A Mathematical Model of the Finding of Usability Problems», em *Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems*, New York, NY, USA, 1993, pp. 206–213.
- [30] J. Cao, «A Guide to User Testing», *The Next Web*, 27-Abr-2015. [Em linha]. Disponível em: <http://thenextweb.com/creativity/2015/04/27/user-testing-explained/>. [Acedido: 19-Ago-2016].

Anexo A – Base dados ER

