



SmartRecycle - Aplicação para deteção de resíduos eletrónicos e dispositivo de recolha

CARLOS ANDRÉ MONTEIRO CARDOSO

novembro de 2021

POLITÉCNICO DO PORTO
INSTITUTO SUPERIOR DE ENGENHARIA DO PORTO

SmartRecycle - Aplicação para deteção de resíduos eletrónicos e dispositivo de recolha

Carlos André Monteiro Cardoso

Mestrado em Engenharia Electrotécnica e de Computadores
Área de Especialização em Automação e Sistemas



DEPARTAMENTO DE ENGENHARIA ELETROTÉCNICA
Instituto Superior de Engenharia do Porto

Novembro, 2021

Esta dissertação satisfaz, parcialmente, os requisitos que constam da Ficha de Unidade Curricular de Tese/Dissertação, do 2º ano, do Mestrado em Engenharia Electrotécnica e de Computadores, Área de Especialização em Automação e Sistemas.

Candidato: Carlos André Monteiro Cardoso, N.º 1141250,
1141250@isep.ipp.pt

Orientação Científica: Susana Nicola , sca@isep.ipp.pt

Empresa: Do iT Lean, Inc.

Orientador: Paula Graça, paula.graca@doitlean.com



DEPARTAMENTO DE ENGENHARIA ELETROTÉCNICA
Instituto Superior de Engenharia do Porto
Rua Dr. António Bernardino de Almeida, 431, 4200-072 Porto

Novembro, 2021

Agradecimentos

Gostaria nestas breves palavras expressar os meus mais profundos agradecimentos pelo apoio e incentivos recebidos.

À Doutora Susana Nicola por ter proporcionado o contacto com esta empresa e estágio e pela disponibilidade permanente no esclarecimento de dúvidas ou informações. Não posso deixar de referir também a forma como confiou nas minhas potencialidades e me encorajou a ir mais além.

À empresa Do iT Lean, na pessoa da Engenheira Paula Graça pela forma como me acolheu, prestando toda a informação e apoio necessário à concretização do estágio, permitindo-me desenvolver o trabalho de acordo com os meus interesses e ritmo próprio.

Ao Engenheiro Nuno Dâmaso, pela camaradagem e relação de entreaajuda. Esteve sempre presente para partilhar o seu conhecimento e apoiar o trabalho que foi desenvolvido. Mais do que um orientador técnico, vi-o como um colega com quem cooperei e com quem muito aprendi.

À minha família que me possibilitaram o percurso escolar que agora termino. Pelo incentivo e confiança constante, e pela sua colaboração na concretização desta etapa.

Resumo

Devido aos avanços tecnológicos e à transformação digital, o crescente uso e expansão dos dispositivos eletrônicos têm originado uma ampla gama de produtos e serviços inovadores num mundo cada vez mais exigente. No entanto, essa tendência aumenta as preocupações da sociedade com a sustentabilidade. A enorme quantidade de dispositivos não utilizados ou descartados, principalmente *smartphones*, causa uma grande quantidade de lixo eletrônico (*e-waste*) e com um baixo índice de reciclagem. Além disso, no fabrico desses aparelhos eletrônicos são essenciais os chamados “*minerais de conflito*” (estanho, tungstênio, tântalo e ouro), também conhecidos como 3TG. Esses recursos naturais, muito procurados, são extraídos em países politicamente instáveis, sendo a sua transação financiadora de conflitos armados e violação sistemática do direito internacional: corrupção, lavagem de dinheiro, trabalho forçado e outros abusos dos direitos humanos.

O presente trabalho de pesquisa reflete essas preocupações e está alinhado com a “*Agenda 2030 para o Desenvolvimento Sustentável*” das Nações Unidas, propondo um sistema de reciclagem de lixo eletrônico a ser disponibilizado em locais públicos, como escolas, empresas e organizações públicas, ao mesmo tempo que aumenta a consciencialização e incentiva a população a reciclar. Este trabalho consiste numa aplicação móvel e numa plataforma *web* (SmartRecycle App), além de um MVP (Minimum Viable Product) que permite a recolha de um dispositivo eletrónico num recetor (Delivery Point). Como resultado, do uso desta aplicação o autor do dispositivo reciclado recebe vouchers para obter descontos em comerciantes locais que aderem à plataforma *web*. Simultaneamente, esses comerciantes aderentes são promovidos como contribuintes ativos para uma economia mais sustentável.

A aplicação e o MVP foram alvo de avaliação quanto a sua usabilidade, através de questionários com escala de cinco níveis e contendo cada um deles cinco questões. Os resultados obtidos questionários situaram-se, na generalidade e num contínuo, nos dois níveis superiores e com uma percentagem significativa. Tal facto, permite constatar que os inquiridos consideraram a aplicação e o MVP um produto funcional e viável.

Palavras-Chave: Sustentabilidade, Economia Circular, Reciclagem, Desenvolvimento de Software, Outsistemas.

Abstract

Due to technological advances and digital transformation, the growing use and expansion of electronic devices have given rise to a wide range of innovative products and services in an increasingly demanding world. However, this trend increases society's concerns about sustainability. The huge amount of unused or discarded devices, especially smartphones, causes a large amount of electronic waste (e-waste) and there is still a low recycling rate. Furthermore, in the manufacture of these electronic devices, the so-called conflict minerals (tin, tungsten, tantalum and gold), also known as 3TG, are essential. These highly sought after natural resources are extracted in politically unstable countries, and their transaction finances armed conflicts and systematic violations of international law: corruption, money laundering, forced labor and other human rights abuses.

This research work reflects these concerns and is in line with the United Nations "*Agenda 2030 for Sustainable Development*", proposing an electronic waste recycling system to be made available in public places such as schools, businesses and organizations while raising awareness and encouraging people to recycle. This work consists of a mobile application and a web platform (SmartRecycle App), in addition to an MVP (Minimum Viable Product) that allows the collection of an electronic device in a receiver (Delivery Point). As a result, using this application the author of the recycled device receives vouchers to obtain discounts at local merchants who join the web platform. Simultaneously, these adherent merchants are promoted as active contributors to a more sustainable economy.

The application and the MVP were evaluated for their usability, through questionnaires with a five-level scale and each containing five questions. The results obtained from the questionnaires were located, in general and in a continuum, in the two upper levels and with a significant percentage. This fact shows that respondents considered the application and the MVP a functional and viable product.

Keywords: Sustainability, Circular Economy, Recycling, Software Development, Outsystems.

Índice

Lista de Figuras	vii
Lista de Tabelas	xi
Listagens	xiii
Lista de Acrónimos	xv
1 Introdução	1
1.1 Contextualização	1
1.1.1 Empresa de acolhimento	3
1.2 Questões de investigação e metodologia de investigação	5
1.3 Objetivos	5
1.4 Plano de Trabalho	6
1.5 Organização da Dissertação	6
2 Economia Circular	9
2.1 Desenvolvimento Sustentável	9
2.2 Transição de Paradigmas: Emergência do modelo de Economia Circular	13
2.3 Reciclagem (Resíduos de Equipamentos Elétricos e Eletrônicos) . . .	19
2.4 Aplicações existentes	23
3 Metodologias	25
3.1 Metodologia de investigação	25
3.2 Metodologia de trabalho	28
4 Desenvolvimento de Software	33
4.1 OnBoarding	33
4.2 Execução	37
4.2.1 Atores	41
4.2.2 Sistema SmartRecycle	42
4.2.3 Sistema <i>Backoffice</i>	56
4.2.4 Sistema <i>Backend</i>	61
4.2.5 Sistema <i>Delivery Point</i>	66

5	Testes e Avaliação da solução	75
5.1	Testes	75
5.2	Avaliação da solução	78
6	Conclusões	87
	Referências	91
A	Documentação das API's do Sistema Backend	95
B	Sistema Delivery Point	107
B.1	Desenho técnico do Delivery Point	108
B.2	Documentação da API do Delivery Point	110
C	Código do Delivery Point	111
C.1	Código linguagem Kivy para criação do UI	111
C.2	Código linguagem Python para o funcionamento do Delivery Point .	115

Lista de Figuras

1.1	Organização da empresa.	4
1.2	Gantt chart com a calendarização do projeto.	6
2.1	Objetivos de Desenvolvimento Sustentável: ONU agenda 2030 [1]. . .	13
2.2	Modelo de Economia Linear [2].	14
2.3	Modelo de Economia Circular [3].	16
2.4	Plano de Ação para a Economia Circular [4].	17
2.5	Modelo de Economia Circular [5].	19
2.6	Estratégias de economia circular [4].	20
2.7	Elementos existentes em REEE [6].	21
2.8	Aplicações de reciclagem de resíduos electrónicos na Google Play [7].	24
3.1	Estrutura de investigação de sistemas de informação [8].	27
3.2	Metodologia DSR figura adaptada do artigo [8].	28
3.3	Diferenças entre as duas metodologias [9].	29
3.4	Comparação entre metodologias [9].	30
3.5	Práticas de SCRUM [10].	31
3.6	Processo de SCRUM [11].	32
4.1	Full-stack Outsystems [12].	34
4.2	Componentes da plataforma Outsystems [13].	34
4.3	Camadas da aplicação [14].	36
4.4	Camadas da arquitetura [14].	36
4.5	Iteração da arquitectura [14].	37
4.6	Diagrama de arquitetura de Outsystems <i>Organize</i>	38
4.7	Diagrama de arquitetura do Outsystems (DAO).	39
4.8	Modelo de base de dados.	40
4.9	Tarefas realizadas em cada <i>sprint</i>	41
4.10	Diagrama de casos de uso da aplicação móvel.	48
4.11	<i>Wireframe</i> de apresentação da aplicação.	48
4.12	<i>Wireframe</i> de <i>login</i>	49
4.13	<i>Wireframe</i> de registo.	49
4.14	<i>Wireframe</i> de <i>Settings</i>	49
4.15	<i>Wireframe</i> de <i>Homepage</i> e informação da imagem analisada.	50

4.16	<i>Wireframe</i> do mapa de pontos de recolha.	50
4.17	<i>Wireframe</i> de <i>profile</i>	51
4.18	<i>Wireframe</i> de <i>partners</i> e <i>discounts</i>	51
4.19	<i>Wireframe</i> de resíduos aceites pela aplicação.	52
4.20	DS-1:Diagrama de sequência dos casos de uso 1,2 e 3.	53
4.21	DS-2:Diagrama de sequência do caso de uso 4.	53
4.22	DS-3:Diagrama de sequência dos casos de uso 5 e 6.	54
4.23	DS-4:Diagrama de sequência do caso de uso 7.	54
4.24	DS-5:Diagrama de sequência dos casos de uso 8 e 9.	54
4.25	DS-6:Diagrama de sequência dos casos de uso 10 e 11.	55
4.26	Diagrama de casos de uso do administrador.	58
4.27	<i>Wireframe</i> de <i>login</i> na página de Administrador.	58
4.28	<i>Wireframe</i> da página de Administrador.	59
4.29	<i>Wireframe</i> da página de Administrador para edição de informação.	60
4.30	DS-1:Diagrama de sequência dos casos de uso 1 e 2.	60
4.31	DS-2:Diagrama de sequência dos casos de uso 3,4 e 5.	61
4.32	DS-1:Diagrama de sequência do caso de uso 1.	63
4.33	DS-2:Diagrama de sequência do caso de uso 2.	63
4.34	Função de análise de imagem.	64
4.35	Função de comparação de resíduo aceites.	65
4.36	<i>Delivery Point</i> com identificação de componentes.	66
4.37	Diagrama de bloco do <i>Delivery Point</i> (idb).	67
4.38	Diagrama de casos de uso do <i>Delivery Point</i>	68
4.39	DS-1:Diagrama de sequência do caso de uso 1.	69
5.1	Resultado dos testes do reconhecimento de imagem com recurso a IA.	76
5.2	Resultado dos testes do reconhecimento de imagem com recurso a IA.	76
5.3	<i>Delivery Point</i> (MVP).	77
5.4	Questionário de usabilidade do SmartRecycle.	79
5.5	Gráfico 1 – Facilidade de utilização.	79
5.6	Gráfico 2 – Clareza da informação.	80
5.7	Gráfico 3 – Implementação de funcionalidades.	80
5.8	Gráfico 4 – Utilização num cenário real.	81
5.9	Gráfico 5 – Percentagem de pessoas que recomendariam a aplicação.	81
5.10	Questionário de usabilidade do <i>Delivery Point</i>	82
5.11	Gráfico 1 – Facilidade de utilização.	82
5.12	Gráfico 2 – Clareza do funcionamento.	83
5.13	Gráfico 3 – Clareza de informação.	83
5.14	Gráfico 4 – Útil nos tempos atuais.	84
5.15	Gráfico 5 – Percentagem de pessoas que usariam num cenário real.	84

A.1	Documentação do Sistema Backend (I).	96
A.2	Documentação do Sistema Backend (II).	97
A.3	Documentação do Sistema Backend (III).	98
A.4	Documentação do Sistema Backend (IV).	99
A.5	Documentação do Sistema Backend (V).	100
A.6	Documentação do Sistema Backend (VI).	101
A.7	Documentação do Sistema Backend (VII).	102
A.8	Documentação do Sistema Backend (VIII).	103
A.9	Documentação do Sistema Backend (IX).	104
A.10	Documentação do Sistema Backend (X).	105
B.1	Desenho técnico das partes do <i>Delivery Point</i> .	108
B.2	Desenho técnico do <i>Delivery Point</i> .	109
B.3	Documentação da API.	110

Lista de Tabelas

2.1	Reciclagem - Benefícios.	21
2.2	Efeito de elementos (usados em equipamentos eletrônicos) na saúde humana.	23
2.3	Carateristicas das aplicações existentes no mercado.	24
4.1	Tabela de requisitos da aplicação móvel SmartRecycle.	43
4.2	Cenário realizar login na aplicação.	43
4.3	Cenário criação de conta na aplicação.	44
4.4	Cenário eliminar conta na aplicação.	44
4.5	Cenário análise de resíduo.	45
4.6	Cenário mapa de pontos de recolha e navegação.	45
4.7	Cenário visualizar histórico de resíduos entregues.	45
4.8	Cenário aquisição de descontos.	46
4.9	Cenário visualizar resíduos aceites.	46
4.10	Cenário visualizar pegada ecológica.	46
4.11	Cenário visualizar descontos adquiridos.	47
4.12	Cenário visualizar resíduos analisados.	47
4.13	Matriz de rastreabilidade.	52
4.14	Matriz de rastreabilidade entre casos de uso e diagrama de sequência.	55
4.15	Tabela de requisitos de <i>Backoffice</i>	56
4.16	Cenário realizar autenticação.	56
4.17	Cenário análise manual.	56
4.18	Cenário edição de parceiros e descontos.	57
4.19	Cenário edição de pontos de recolha.	57
4.20	Cenário edição de resíduos aceites.	57
4.21	Matriz de rastreabilidade.	60
4.22	Matriz de rastreabilidade entre casos de uso e diagrama de sequência.	61
4.23	Tabela de requisitos de <i>Backend</i>	62
4.24	Cenário receber imagem para classificar.	62
4.25	Cenário utilizador obteve um desconto.	63
4.26	Matriz de rastreabilidade.	64
4.27	Lista de componentes eletrônicos usados no <i>Delivery Point</i>	66
4.28	Tabela de requisitos do <i>Delivery Point</i>	67
4.29	Cenário interação através de LCD.	68

4.30 Matriz de rastreabilidade.	69
4.31 Matriz de rastreabilidade entre casos de uso e diagrama de sequência.	70

Listagens

4.1	Código de leitura de QR <i>Code</i>	71
4.2	Formatação de dados para API.	72
4.3	Função de envio dos dados para API.	73
4.4	Código de medição de peso.	74
C.1	Código linguagem Kivy para criação do UI.	111
C.2	Código linguagem Python para o funcionamento do Delivery Point.	115

Lista de Acrónimos

API	<i>Application Programming Interface</i>
APP	<i>Application</i>
CCU	<i>Cenário de Caso de Uso</i>
CRUD	<i>Operações de Create, Read, Update e Delete</i>
DAO	<i>Diagrama de Arquitetura do Outsystems</i>
DEE	<i>Departamento de Engenharia Eletrotécnica</i>
DS	<i>Diagrama de Sequência</i>
DSR	<i>Design Science Research</i>
EUA	<i>Estados Unidos da América</i>
GEE	<i>Gases de Efeito de Estufa</i>
HTTP	<i>Hypertext Transfer Protocol</i>
IA	<i>Inteligência Artificial</i>
IDB	<i>Internal Block Diagram</i>
ISEP	<i>Instituto Superior de Engenharia do Porto</i>
IT	<i>Information Technology</i>
LCD	<i>Liquid Crystal Display</i>
MEEC	<i>Mestrado em Engenharia Eletrotécnica e de Computadores</i>
MVP	<i>Minimum Viable Product</i>
ONU	<i>Organização das Nações Unidas</i>
QR	<i>Quick Response</i>
REEE	<i>Resíduos de Equipamentos Eléctricos e Electrónicos</i>
SOA	<i>Service-Oriented Architecture</i>
TEDI	<i>Tese/Dissertação</i>
UI	<i>User interface</i>
URL	<i>Uniform Resource Locator</i>
UX	<i>User Experience</i>

Capítulo 1

Introdução

Neste capítulo é feita uma contextualização dos temas abordados e são definidos os objetivos propostos. De seguida, apresentar-se-á um diagrama de Gantt, que mapeia as tarefas desenvolvidas, assim como o tempo que estas ocuparam. Por fim, descrever-se-á a estrutura do relatório e resumir-se-á o conteúdo do mesmo.

Este relatório descreve o desenvolvimento do projeto no âmbito da unidade curricular de Tese/dissertação (TEDI), integrada no 2º ano do Mestrado de Engenharia Eletrotécnica e de Computadores (MEEC), no Departamento de Engenharia Eletrotécnica (DEE) do Instituto Superior de Engenharia do Porto (ISEP), bem como o estágio realizado na Do iT Lean na execução do mesmo.

1.1 Contextualização

Atualmente existe uma elevada preocupação com o meio ambiente, daí que a reciclagem assuma grande ênfase como passo para a redução de recursos usados. A preservação do meio ambiente está a tornar-se cada vez mais importante e a reciclagem é fundamental no combate de muitas das agravantes que contribuem para a sua degradação. Por esse motivo, a criação de uma aplicação que facilite a identificação de resíduos que tenham “*valor*” reciclável e consequentemente a atribuição desse valor em forma de descontos, representa uma estratégia inovadora e de fomento de comportamentos ecológicos. Esta dissertação emerge da preocupação evidente

na Agenda 2030 para o Desenvolvimento Sustentável das Nações Unidas [1]. O Desenvolvimento Sustentável surge da crescente preocupação com as necessidades de crescimento económico mundial (desenvolvimento) e, simultaneamente, com os limites biofísicos do planeta (sustentabilidade), conciliando objetivos sociais aparentemente conflitantes (proteção ambiental e crescimento económico).

O conceito de economia circular é um modelo económico que privilegia uma ação humana inspirada pelas leis da natureza, consciente, coordenada e orientada para a redução, reutilização e reciclagem. A mesma tem vindo a ganhar destaque no discurso da sustentabilidade. Assim, o presente projeto será desenvolvido tendo em conta o modelo teórico de economia circular e o conceito de sustentabilidade.

Em 2019, foram produzidas mundialmente cerca de 53,6 milhões de toneladas de lixo eletrónico, dos quais apenas 17.4% foram recolhidos e reciclados, o que se traduz num baixo valor recuperado face ao total possível. A Europa tem a maior taxa de reciclagem, com cerca de 42%. Portugal apresenta aproximadamente o mesmo valor considerados todos os tipos de resíduos. Infelizmente, a maior parte do lixo, não apenas eletrónico, continua a ser descartado, abrindo assim espaço para a intensificação da reciclagem, a nível nacional e mundialmente [6].

Os metais e minerais 3TG (*Tin, Tantalum, Tungsten e Gold*) são essenciais à economia global e constituem matérias-primas necessárias à produção industrial. A extração destes tipos de materiais tem apresentado um acentuado incremento, tendência que deverá manter nas próximas décadas, acompanhando o crescimento da população mundial e a transição para as energias limpas [15].

A extração destas matérias acarreta elevadas consequências na degradação da vida selvagem e ecossistemas. Porém, a realidade pode ser diferente se a indústria extratora assumir metas de Desenvolvimento Sustentável através de promoção de coesão social, inclusão e progresso económico dos países/população vulneráveis fornecedoras das matérias-primas. O relatório de Fevereiro de 2020, do International Resource Panel, elenca orientações para o setor extrator e introduz a “*Licença Operacional de Desenvolvimento Sustentável*”, mais holística, integrada e inclusiva, garantia de resultados económicos, sociais e ambientais positivos, defendendo negócios justos e distribuição equitativa dos lucros e responsabilidades dos parceiros [15].

A International Resource Panel sugere ainda incluir leis e regulamentações de extração no Plano de Desenvolvimento Estratégico, que consagrem os princípios de consulta, transparência e prestação de contas, bem como reconheçam e promovam os direitos das populações locais. Ademais, os denominados minerais-conflito usados

na manufactura de produtos IT alimentam guerras e abusos dos direitos humanos, em países instáveis e economicamente desfavorecidos: trabalho escravo, trabalho infantil, violência, corrupção e degradação ambiental. Cerca de 40% dos conflitos internos estão relacionados com estes recursos naturais [15]. Representa também, uma fonte de problemas graves de saúde pelos processos inseguros de mineração e da degradação do ambiente nestas comunidades.

A aplicação proposta, destina-se a todos os utilizadores que não realizam a reciclagem e até mesmo aos restantes que já fazem a separação de resíduos em casa, e contribui para o Objetivo 12 do Agenda 2030 para o Desenvolvimento Sustentável das Nações Unidas. Além disso, irá contribuir para minimizar o “*Conflito de Materiais*” através do rastreamento e da sensibilização dos 3TG, minimizando assim o comércio dos mesmos.

1.1.1 Empresa de acolhimento

A Do iT Lean é uma empresa global na área da tecnologia/informática, fundada e com sede em Portugal (Leiria). A empresa iniciou atividade em finais de 2009, por dois engenheiros experientes e entusiastas no uso da plataforma de baixo código Outsystem.

O ano de 2010 foi especial; a empresa consegue os dois primeiros clientes de grande importância, a Via Verde e Ministério dos Negócios Estrangeiros. No ano seguinte funde-se com a empresa Simetricircle de Leiria, facto que permitiu expandir o seu alcance para mercados em outros países, como Espanha, Reino Unido, Hong Kong, Brasil e, assim, para clientes internacionais.

Em 2014, a Outsystems reconheceu a Do iT Lean como parceiro mais valioso. Este facto consolidou a sua internacionalização. Dois anos depois apresentou um crescimento de 40%, com a receita a duplicar. Nesse mesmo ano, conseguiu o primeiro cliente remoto nos Estados Unidos da América, tendo obtido igualmente o primeiro Outsystems Innovation Award.

Durante 2017 torna-se uma empresa verdadeiramente global, ao entrar no mercado australiano. No mesmo ano, em resultado deste crescimento, a empresa instala um centro no Porto e aumenta o espaço de Leiria para responder às solicitações do mercado e de aumento de colaboradores.

No ano de 2018 vê o seu nível reconhecido com seis prémios Nacionais e Internacionais. Destacou-se, também, ao apresentar novos clientes, em novas latitudes, novos colaboradores e novos escritórios (Açores). O ano seguinte coincide com a criação do Do iT Lean, Inc. com sede em Austin (Texas), responsável pelas operações na América do Norte e com o prémio EBA Growth Strategy National Winner. Em 2020, apesar da situação pandémica, a empresa intensificou a sua ação e conseguiu

expandiu-se para a região Ásia Pacífico (Malásia/Índia).

A empresa Do iT Lean compreende, na sua estrutura vários sectores ou departamentos. Desde logo, o departamento executivo que integra os vários diretores executivos, a quem compete definir as políticas e orientações da empresa, conforme ilustrado na Figura 1.1.



Figura 1.1: Organização da empresa.

O departamento de gestão (*Management*) é responsável pela gestão do negócio em várias regiões e pela gestão de equipas de trabalho e de projetos de clientes. Ao departamento de equipa de projeto é atribuído o desenvolvimento de projetos *web* e *mobile*, de acordo com a necessidade do cliente. Ao departamento de apoio compete apoiar os demais departamentos, desde *marketing*, recursos humanos e finanças. O departamento de operações internas está incumbido de assegurar um bom produto final e de satisfazer as necessidades dos clientes, de disponibilizar suporte e manutenção e de treinar estagiários e recém-chegados a empresa para a sua cultura e processos.

A Do iT Lean é uma empresa focada em fornecer aplicações de *web* e aplicações móveis com *design* e funcionalidades actuais. Especializou-se no uso da plataforma de baixo código OutSystems, podendo assim reduzir custos de desenvolvimento e aumentar a velocidade de produção.

A empresa apoia-se numa abordagem ÁGIL e LEAN, abordagem essa que permite à empresa assegurar que as funcionalidades mais importantes são disponibilizadas ao cliente, primeiro e mais rapidamente. As funcionalidades de prioridade mais baixas são substituídas pelas ideias mais importantes, resultando numa aplicação simples, mas que satisfaz as necessidades do cliente [16].

1.2 Questões de investigação e metodologia de investigação

O projeto tem como intuito poder apresentar respostas a questões importantes da atualidade sobre a sustentabilidade do planeta e o modelo de economia. Assim, considerou-se como perguntas mais importantes:

- Como contribuir para o Objetivo 12 da Agenda de 2030 do Desenvolvimento Sustentável?
- Como contribuir para impulsionar a Economia Circular?
- Como sensibilizar para a minimização do comércio dos 3TG em regiões de conflito?

A metodologia utilizada para desenvolvimento deste projeto, teve por base o *design science research*, o que permite a utilização de diversas ferramentas como auxílio ao desenvolvimento do projeto. Esta metodologia de investigação tem vindo a cimentar-se na área dos sistemas de informação, uma vez que se define como abordagem epistemológica - metodológica focada no pensar e fazer.

1.3 Objetivos

A presente proposta visa, assim, o desenvolvimento de uma solução com os seguintes objetivos:

- Reconhecer dispositivos eletrónicos através da imagem, de forma a determinar o seu “*valor*” reciclável, ou seja, o seu valor como resíduo;
- Implementar um ponto de recolha inteligente simulado através de um micro-controlador (Arduíno, ESP32, Raspberry Pi, ou outros dispositivos adequados);
- Fomentar a entrega dos resíduos ao disponibilizar/entregar vales de pontos baseados no seu “*valor*” como resíduo;
- Incentivar a reciclagem, através, por exemplo, de descontos no comércio local (vales de descontos);

1.4 Plano de Trabalho

O Gantt chart ilustrado na Figura 1.2 resume o plano de trabalhos para este projeto.

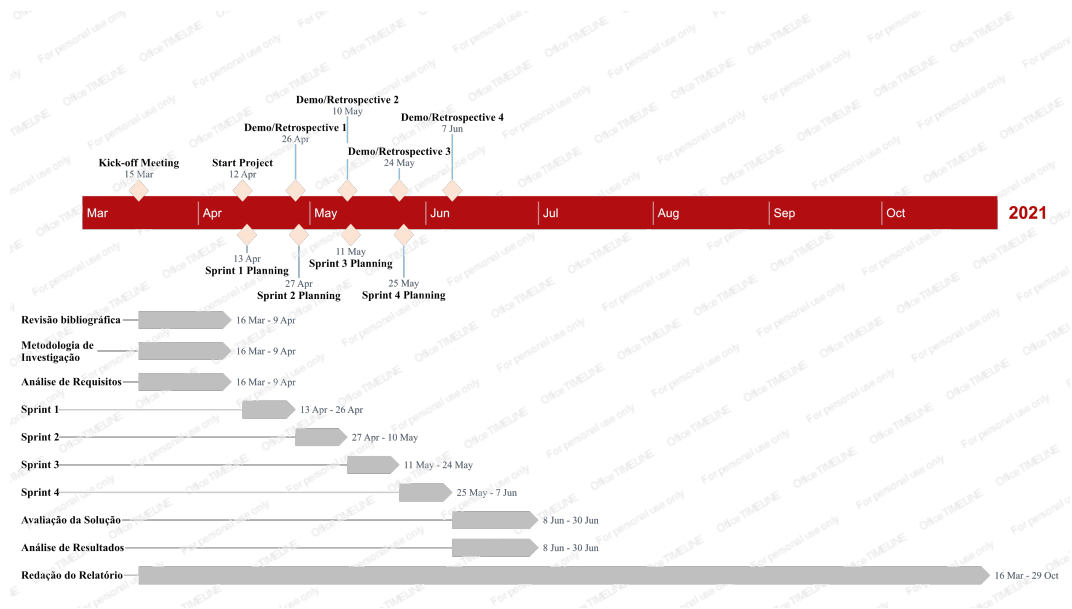


Figura 1.2: Gantt chart com a calendarização do projeto.

1.5 Organização da Dissertação

O presente relatório contém seis capítulos: Introdução, Economia Circular, Metodologias, Desenvolvimento de Software, Testes e Avaliação da solução e Conclusão.

No Capítulo 1, de carácter introdutório, é feita a contextualização do estágio, são apresentados os objetivos da empresa para o mesmo, bem como a calendarização das diferentes etapas.

O Capítulo 2, consiste na revisão bibliográfica para o desenvolvimento do projeto. É explorado o conceito de Desenvolvimento Sustentável e apresentado o modelo de Economia Circular como resposta às preocupações sociais e ambientais emergentes. É também veiculada informação sobre a Reciclagem e por fim, abordadas as aplicações existentes no mercado atualmente.

No Capítulo 3, é referida e explanada a metodologia de investigação *Design Science Research* e a metodologia Ágil de trabalho adotada durante a execução do projeto.

O Capítulo 4, compreende a apresentação da plataforma usada para o desenvolvimento do projeto e descrito o mesmo, através de tabelas de requisitos, casos de uso, matrizes de rastreabilidade e diagramas de sequência.

No Capítulo 5, é salientada a testagem dos requisitos e de que maneira esta implementação satisfaz os objetivos iniciais.

Por fim, no Capítulo 6, a dissertação é concluída com uma reflexão do trabalho realizado e são avançadas possíveis melhorias.

Capítulo 2

Economia Circular

O presente capítulo refere-se à revisão bibliográfica e a fundamentação teórica para o desenvolvimento do projeto concebido. Inicialmente é explorado o conceito de Desenvolvimento Sustentável, filosofia tão presente e premente face aos desafios que se colocam à gestão ambiental. De seguida, é apresentado e explorado o Modelo de Economia Circular, um novo paradigma económico, potenciador do desenvolvimento económico e social, mas simultaneamente conciliador com as preocupações ecológicas e a preservação do Planeta. Posteriormente, é feita uma incursão sobre o Ciclo dos 4 R's da Economia Circular, com especial relevo para o processo de Reciclagem. Na parte final são abordadas as aplicações de reciclagem de resíduos existentes atualmente no mercado e as suas funcionalidades mais usadas.

2.1 Desenvolvimento Sustentável

Perspetiva histórica

A preocupação com a sustentabilidade não é um fator surgido no presente, mas reporta-se a vivências e abordagens que emergiram nos séculos XVIII e XIX, com Matheus e Jevous, fruto de explosão populacional e da conseqüente escassez de recursos [17]. É, contudo, no século XIX que assume grande relevância face aos impactos ambientais resultantes do desenvolvimento industrial de parte significativa de países.

Assentes num modelo económico capitalista, as sociedades industrializadas promovem a massificação de produção e consumo. Este período traz grande prosperidade e desenvolvimento económico, mas, paralelamente, acarreta consequências ambientais, mudanças sociais e culturais e clivagens ao nível do desenvolvimento entre os países do hemisfério Norte e Sul [18].

Na 2^a metade do século XX (décadas de 60 e 70) constatou-se a necessidade de integrar as questões ambientais e o desenvolvimento social nas políticas económicas dos estados, [19], iniciando-se um percurso, ainda que irregular, na direção da conciliação de políticas económicas e de políticas ambientais - Política internacional do ambiente. Este modelo pressupõe um desenvolvimento sustentável, tendo em conta a defesa do ambiente e dos recursos materiais e a manutenção do desenvolvimento económico e social. Na perspetiva de Soromenho-Marques (2003) o percurso rumo à sustentabilidade Global divide-se em 4 ciclos distintos [20].

1º ciclo (1962-1973) – “*Génese*”:

Ciclo de expansão. Emergem as primeiras obras sobre o desenvolvimento económico no ambiente e sobre o despertar para a consciência ambiental e são criados, principalmente, na EUA, Japão e Europa, organismos referentes às políticas nacionais do ambiente. Em Portugal, surge em 1971 a Comissão Nacional do Ambiente. Em Estocolmo (1972), a Conferência das Nações Unidas sobre Ambiente humano debruçou-se sobre a necessidade de debater e resolver problemas de ordem ambiental que ocorreram com o desenvolvimento económico no pós-guerra.

2º ciclo (1973-1983) - “*Recuo*”

Fase caracterizada pela crise petrolífera e pela instabilidade económica. A recessão económica, os problemas laborais e crises sociais assumem o papel de charneira, remetendo para as preocupações ambientais. Em contraciclo (início da década de 80), vão aparecendo movimentos e até partidos ecologistas, com representação parlamentar.

3º ciclo (1983-1997): - “*Virtuoso*”

Ciclo que se define pelo assumir de uma posição e discurso ambiental cada vez mais internacional e que se consubstancia na legislação e criação de ministérios do ambiente nos países desenvolvidos. Em 1983, as Nações Unidas criam a Comissão Mundial para o Ambiente e Desenvolvimento e quatro anos mais tarde disponibilizam o relatório “*Nosso Futuro Comum*” ou o relatório Brundtland. Este constitui-se

como um marco inquestionável, não só pela definição do “*Desenvolvimento Sustentável*”, assimilada e integrada por governos, empresas e outras organizações (Gladwin et. al, 1995, in [17]), como também pela referência às necessidades a superar, por forma a alcançar um desenvolvimento sustentável, quer através da adoção de um modelo de desenvolvimento económico sem degradação e exaustão dos recursos, quer pela equidade de distribuição de recursos. Assim, “*Desenvolvimento Sustentável é o desenvolvimento que dá resposta às necessidades presentes, sem comprometer a capacidade das gerações futuras de poderem satisfazer as suas.*” [21].

Durante este ciclo sucedem-se convenções e conferências com a assinatura de protocolos que representam esforços assinaláveis de vários países na concretização do desenvolvimento sustentável. A convenção de Viena (1985), o protocolo de Montreal (1987), a conferências do Rio de Janeiro (1992) e o protocolo de Quioto (1997), são momentos de viragem no paradigma do desenvolvimento e na promoção de sustentabilidade do planeta.

4º ciclo (1997-2006) - “*Impasse*”:

Ciclo que se caracteriza pela desmobilização dos EUA em relação ao compromisso assumido no Protocolo de Quioto (redução de emissões de gases com efeito de estufa) e pela dissonância entre os discursos de política ambiental e a concretização das ações no terreno. Em 2000 são assinados pelos países participantes, os Objetivos do Milénio a serem cumpridos até 2015, com enfoque na erradicação da pobreza e na promoção do desenvolvimento humano, no Mundo.

Em 2002 realiza-se a Cimeira da Terra de Joanesburgo, de onde sai uma Declaração Política e um Plano de Ação, que inclui capítulos sobre a pobreza, água e saneamento, energia, saúde, educação, biodiversidade, recursos naturais, alterações climáticas, globalização, comércio internacional e ajuda ao desenvolvimento. No entanto, os documentos não são vinculativos e não incluem qualquer tipo de monitorização ou sanção [17].

Atualidade

Nos últimos anos, surgiu uma nova fase que se pauta pela crescente preocupação e sensibilidade para com as questões ambientais e o bem-estar humano, manifestada pela comunidade internacional. Muitas figuras públicas têm feito ouvir a sua voz num persistente alerta para a necessidade de AGIR e em tempo útil. A sensibilização da comunidade internacional tem vindo a alcançar um papel de relevo nesta mudança de orientações políticas, assente em informação especializada (relatórios e outros documentos) e no reconhecimento dos esforços desenvolvidos.

Mais recentemente a ativista ambiental Greta Thunberg alertou consciências e colocou as preocupações ambientais sob os holofotes da comunicação social e da comunidade internacional. Este facto foi ainda mais avolumado com os fenómenos climáticos que têm ocorrido de forma mais frequente, intensa e mundialmente, com desfechos dramáticos.

A definição de desenvolvimento sustentável emanada do Relatório de Brundtland congrega vários aspetos na sua formulação que o tornaram amplamente aceite e muito relevante na discussão do desenvolvimento sustentável. Desde logo, o facto de possibilitar a conciliação de objetivos aparentemente em conflito: a proteção ambiental e o crescimento económico.

A contextualização temporal do relatório, incide sobre uma época de crise ambiental e de focos de poluição, alvo de atenção por parte dos políticos e ambientalistas, reforçada por problemas emergentes (ex: Buraco do Ozono na Antártida ou o Desastre de Chernobyl) com implicações acentuadas. Porém, é também uma fase marcada pela necessidade de desenvolver económica e socialmente os países em desenvolvimento. O termo “*Desenvolvimento*” está associado às questões socioeconómicos, enquanto o termo “*Sustentabilidade*” se liga a objetivos ecológicos. O conceito de desenvolvimento sustentável, dirige a atenção para as necessidades dos mais pobres, mas também para os limites biofísicos do planeta que podem condicionar o crescimento económico, mesmo se este resultar de uma nova organização social e tecnológica [17]. O Desenvolvimento Sustentável pretende assegurar a melhoria da qualidade de vida das pessoas, contudo, sem comprometer os recursos naturais para além da capacidade do planeta. Assumem-se, assim, 3 grandes vetores, igualmente importantes, mas com o imperativo de integração e equidade dos vários domínios.

- Crescimento económico e equidade: proporcionado a todas as comunidades dos países;
- Qualidade ambiental e recursos: garantindo às gerações futuras a satisfação das suas necessidades, adotando ações que conciliem o desenvolvimento económico, com a gestão racional dos recursos naturais (conservação dos habitats, redução de poluição e do consumo dos recursos);
- Desenvolvimento social: proporcionado à população a satisfação das necessidades em termos de trabalho, alimentação, educação, saúde, energia, água e saneamento básico, bem como de outras necessidades (sociais e culturais).

Em 2015 as Nações Unidas definiram 17 objetivos de Desenvolvimento Sustentável, indicados na Figura 2.1, documento que integrou 150 objetivos específicos, bastante ambiciosos e que pressupõem uma mudança sistêmica e global, exigindo consequentemente a mobilização de múltiplos grupos de interesse.



Figura 2.1: Objetivos de Desenvolvimento Sustentável: ONU agenda 2030 [1].

Para responder efetivamente ao Objetivo 12 - Produção e Consumo Responsável foi adotado o Modelo de Economia Circular como paradigma e mecanismo passível de contribuir para o alcance das metas indicadas em outros objetivos de Desenvolvimento Sustentável.

2.2 Transição de Paradigmas: Emergência do modelo de Economia Circular

Na perspectiva de MacArthur (2013), a economia mundial assenta num modelo de consumo de recursos – Economia Linear, que consiste na extração das matérias-primas, que transforma e no final descarta, terminando, assim, o seu ciclo de vida [22]. O aumento do preço das matérias-primas, decorrente da escassez de recursos, o aumento da forma e acesso aos produtos, mas também a preocupação crescente com o estado ambiental, constituíram motivos significativos para a mudança de modelo económico.

O modelo de Economia Linear, exposto na Figura 2.2, consiste na sequência extração, produção, disposição [22].

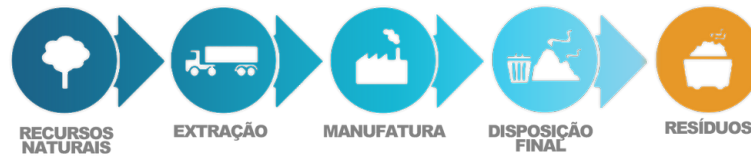


Figura 2.2: Modelo de Economia Linear [2].

As empresas extraem as matérias-primas, que transformam (processo de produção) num determinado produto, disponibilizado ao consumidor e que este, por seu turno, vai descartar quando o produto já não se revelar necessário ou terminar a sua função. Esta visão simplista tende a reduzir o ambiente a uma fonte inesgotável de recursos e a ignorar o impacto negativo no meio ambiente [23]. A manutenção do Modelo de Economia Linear e do sistema produtivo associado a este, faz prever o crescimento acentuado da necessidade de matérias-primas.

Segundo MacArthur (2013), em 2010 entraram no ciclo produtivo cerca de 65 mil milhões de toneladas de matérias-primas, valor que se previa chegar aos 82 mil milhões no espaço de uma década. Os valores apresentados mostram que este modelo não é exequível por muito mais tempo e que se impõem medidas, tendo em vista reduzir os gastos e proteger o meio ambiente. O aumento do consumo significa também mais emissões de gases com efeito de estufa, resíduos e poluentes, sendo que mais de metade de todos os poluentes produzidos a nível mundial se deve à produção de matérias básicas e 70% de todos os resíduos produzidos são emitidos ao gerar o produto.

“Do ponto de vista ambiental, estamos a viver a crédito” (João Pedro Matos Fernandes – Ministro do Ambiente 2017-2020) [4]. Segundo Matos Fernandes, a última vez que a sociedade consumiu dentro dos limites da capacidade de regeneração do ecossistema foi no início da década de 70. A partir desse momento o limite dessa capacidade verifica-se cada vez mais cedo. Daí que considere que se está a viver a crédito, situação que já está a atingir um nível de alarme. A economia linear é responsável por 65 mil milhões de toneladas de recursos extraídos todos os anos. Destes, apenas uns meros 7% são reciclados. Este modelo económico exige um permanente aumento de recursos, o que acarreta a diminuição acelerada das suas reservas, podendo este ser definido como o modelo do risco e da escassez [4].

A Comissão Europeia (2014) alertou para o facto do efeito do crescimento demográfico e o aumento de riqueza/poder de compra, conduzir ao incremento da procura de recursos, que inevitavelmente se tornam cada vez mais escassos e até mesmo inacessíveis face ao seu valor de mercado. A escassez e mesmo a rutura dos recursos naturais nunca entraram na equação de economia linear.

Este organismo deparou-se com a fragilidade em que a Europa se encontra ao depender maioritariamente de recursos importados, com apenas 9% de capacidade interna para 54 dos materiais essenciais, ficando refém de abastecimentos de tais recursos e de preços demasiado elevados. Por último, a preocupação com o meio ambiente é um ponto igualmente importante, pelo que a adoção do modelo de economia circular é encarada como uma forma de combater excessos, apostando-se na necessidade de sermos mais eficientes ao nível do consumo e da rentabilização dos recursos naturais.

Em Portugal, a indústria transformadora depende de 53% do seu volume de negócio para adquirir matérias-primas. No entanto, é possível reduzir este valor, o que representaria um aspeto muito favorável para a economia em geral. Porém, estas mudanças serão lentas e progressivas, tendo cada empresa de se reinventar e adotar uma nova estratégia para os seus produtos, processos e plano de negócio e assim garantir a sustentabilidade perante estas mudanças estruturais. Este processo não será fácil e haverá sectores em que poderão ficar a perder com as novas mudanças. Por esse motivo, é necessário apresentar legislação que incentive e ajude à transição.

Modelo de economia circular

Contrariamente à economia linear emerge agora uma nova ordem, um novo paradigma – a Economia Circular. A Economia Circular constitui o paradigma do sistema do futuro [24], abandonando o modelo vigente, fortemente ameaçado pela disponibilidade limitada de recursos naturais e da própria capacidade de assimilação da poluição produzida por parte do planeta. Este modelo, apesar de recente, tem assumido supremacia, não apenas na dimensão de rentabilidade, mas também da própria independência em relação a recursos e à oscilação de custos de produção [25]. A Economia Circular incentiva a inovação nas práticas de gestão e promove

novas oportunidades para as empresas/organizações na criação de valor, sempre em harmonia com o meio ambiente. Alicerçada na “*Sustentabilidade*”, esta nova abordagem representa possibilidades de crescimento sólido e de competitividade no mercado dinâmico e global [24].

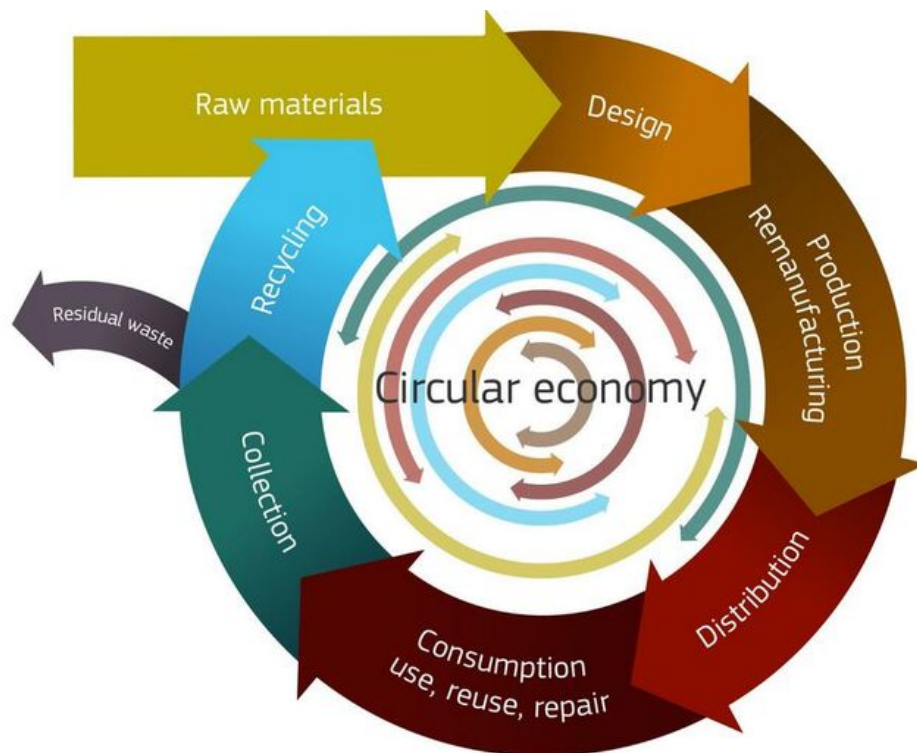


Figura 2.3: Modelo de Economia Circular [3].

A Economia Circular foca a sua dinâmica numa visão restauradora e regeneradora, quer pelo uso de energias não poluentes, quer pelo minimizar de desperdícios e resíduos, apresentado na Figura 2.3. Assim, assenta na ideia de criação de valor baseado na longevidade dos bens produzidos e em novas formas de consumo [26]. Na perspetiva de Stahel (1982) [27], a Economia Circular é um ciclo em espiral que minimiza materiais, fluxo de energia e deterioração do ambiental, sem comprometer o crescimento económico ou o progresso técnico e social [28].

As vantagens obtidas com a adoção deste modelo são imensas. Do ponto de vista ambiental será possível reduzir a emissão de gases de efeito de estufa, ao minimizar todos os recursos primários necessários, bem como atenuar a destruição de habitats, protegendo a biodiversidade.

Na Europa, em termos económicos e financeiros, os benefícios concretizam-se com a redução de importações de matérias-primas a altos preços, o que implicaria uma poupança de cerca de 600 mil milhões de euros, aspeto também benéfico para Portugal, caso que se traduziria num ganho anual bruto de 3.3 mil milhões [4].

A abordagem sustentável é, por conseguinte, mais do que a redução e minimização do impacte negativo em termos ambientais. Não se traduz apenas na gestão mais eficiente dos recursos, mas implica toda uma mudança no fluxo de produção. Na Economia Circular, estamos perante um modelo que otimiza o fluxo de bens, maximiza os recursos naturais e reduz, significativamente, a produção de resíduos. Permite, assim, maximizar o valor do produto.

Esta nova perspetiva estimula benefícios operacionais e estratégicos, oportunidades assinaláveis de inovação e *design* a nível de produtos, processos e modelos de negócio, criação de emprego e crescimento económico sustentável e integrador, apresentado na Figura 2.4. É um modelo de indústria regenerativa e não mais destrutiva e predatória [24].



Figura 2.4: Plano de Ação para a Economia Circular [4].

Este modelo económico de produção alicerça-se em princípios mobilizadores e agregadores de conhecimentos de várias áreas e apelam ao dinamismo e inovação no ciclo produtivo.

Princípios do Modelo de Economia Circular

- Conceber produtos, serviços e modelos de negócio que previnam a produção de resíduos e a poluição do sistema natural;
- Manter produtos e materiais em utilização, no seu valor económico e utilidade mais elevados, pelo máximo tempo possível;
- Fomentar a regeneração dos recursos materiais utilizados e dos sistemas naturais subjacentes.

A Economia Circular é, claramente, um tema recorrente na agenda internacional representando uma linha orientadora para alcançar objetivos e metas definidas de crescimento verde, de desenvolvimento sustentável, de mitigação e de adaptação às alterações climáticas. Portanto, constituem finalidades a perseguir por parte dos diferentes agentes económicos e entidades governamentais.

A Economia Circular promove o uso eficiente e produtividade de recursos, focando-se na desmaterialização, reutilização, reciclagem e recuperação das matérias-primas. Estes processos traduzem-se na retenção do valor da matéria por mais tempo e na redução dos custos da sua obtenção. A preservação, restauração ou reintrodução no ciclo produtivo da matéria-prima recuperada, traz vantagens económicas e também ambientais, quer pela diminuição da extração de matérias-primas, quer pela redução de resíduos e emissões.

Stahel (1982) elenca 4 possíveis ciclos na economia circular:

- **Reutilização** – Processo de recolocação do produto no mercado após limpeza e manutenção mínima, se necessária;
- **Reparação** – Processo que implica a entrega do produto em boas condições após a substituição dos componentes estragados ou em mau funcionamento;
- **Recondicionamento/Remanufactura** – Processo industrial que consiste nas etapas de desmontagem do produto usado, de limpeza das suas peças e da recuperação dos componentes funcionais, a serem usados na construção de um novo produto;
- **Reciclagem** – Processo de conversão de desperdício em matérias ou produtos de potencial utilidade. Reintrodução do produto no sistema produtivo, sob a forma de matéria-prima bruta.

2.3 Reciclagem (Resíduos de Equipamentos Elétricos e Eletrônicos)

A Economia Circular pressupõe o recurso a diversos processos, presentes na Figura 2.5, de forma a rentabilizar recursos, energia e processo produtivo: reutilização do produto por um período mais longo: reparação do produto estragado; remanufatura do que não pode ser reparado; e reciclagem do que não pode ser reutilizado, mas pode funcionar como matéria-prima [27].

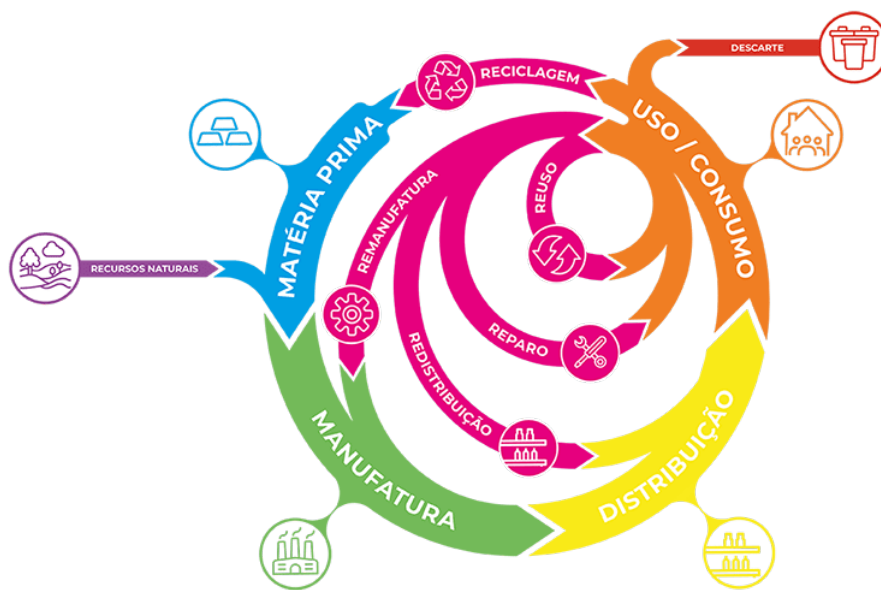


Figura 2.5: Modelo de Economia Circular [5].

Assim sendo, os materiais são preservados, restaurados ou reintroduzidos no sistema de modo cíclico, com vantagens económicas para fornecedores e utilizadores, e vantagens ambientais decorrentes de menor extração e importação de matérias-primas, redução na produção de resíduos e redução de emissões associadas [4]. Inversamente ao modelo de Economia Linear que comportava, com o fim de vida do produto, a perda de matérias/recursos, de energia, e do próprio custo da produção, o princípio implícito na Economia Circular remete para a ação eficiente, ao reduzir a intensidade material e energética do processo produtivo [4].

A circularidade da economia depende da implementação de estratégias de produção e utilização inteligente, expostas na Figura 2.6. Quanto mais “circular” a

economia, menor a extração de matérias-primas e menor a pressão ambiental. Paralelamente, maior é a exigência de ações inovadoras no *design* do produto, no modelo de receitas e em inovação social e institucional [4].

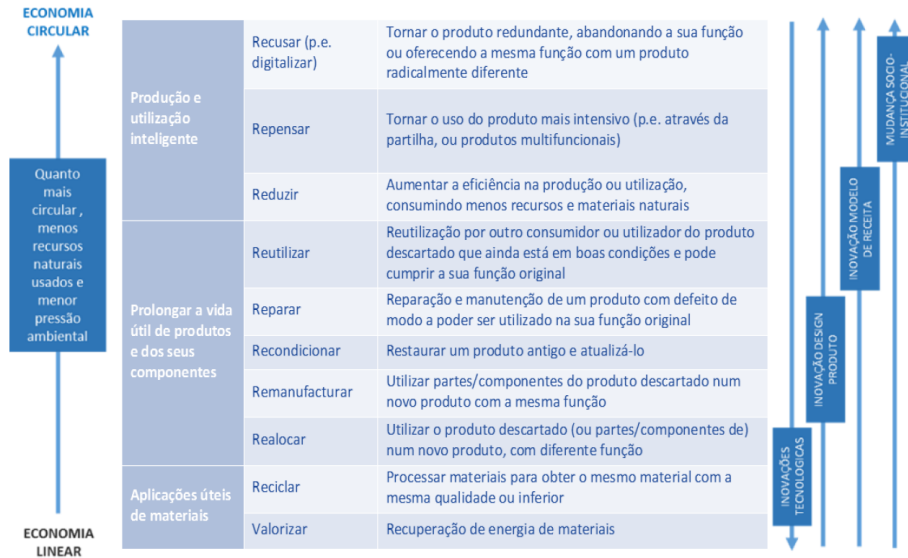


Figura 2.6: Estratégias de economia circular [4].

A reciclagem constitui-se como um componente da máxima dos 4 R's (Reutilização, Reparação, Recondicionamento e Reciclar) integrando-se também na hierarquia dos resíduos (Reduzir, Reutilizar e Reciclar), amplamente assimilada e integrada pelas sociedades e que se tem vindo a materializar na utilização universal dos ecopontos e da separação dos detritos. Deste modo, favorece a redução do consumo de matérias-primas, mas também de energia e da própria poluição da água, e do ar, com a emissão de gases de efeito de estufa.

A reciclagem consiste na conversão dos desperdícios em matérias ou produtos de potencial utilidade, ou seja, é um processo de transformação de materiais que podem voltar ao seu estado inicial e originar produtos iguais e com as mesmas características.

Tabela 2.1: Reciclagem - Benefícios.

Dimensão	Benefícios
Económico	Redução do valor despendido na aquisição das matérias-primas; Redução dos custos de produção.
Social	Melhoria da qualidade de vida das pessoas (melhorar condições ambientais); Reorganização do mercado de trabalho (novas profissões).
Ambiental	Redução da acumulação de resíduos (gases, resíduos sólidos, material perigosos) e da redução da contaminação de solos e água; Redução da extração de matérias-primas; Redução do consumo de energia.

No presente trabalho, a reciclagem foca-se no Resíduos de Equipamentos Elétricos e Eletrônicos – REEE. De acordo com a diretiva da união europeia (2002) referente à gestão de resíduos deste tipo de equipamentos, é definida a necessidade de recolha, reciclagem e valorização de produtos elétricos/eletrónicos. Tal diretiva, tornou-se lei em Portugal em 2003.

Nesta perspetiva, incentiva-se a recuperação de matérias-primas contidas em produtos já existentes, como por exemplo produtos eletrónicos, onde é possível encontrar até 60 elementos da tabela periódica, visíveis na Figura 2.7, sendo que muitos destes são recuperáveis. Elementos como ouro, prata, cobre, platina e paládio são muito utilizados, cobiçados e dispendiosos.

Source: Deubzer et al. 2019

Figura 2.7: Elementos existentes em REEE [6].

A United Nations University estima que se encontrem 55 biliões de euros em matérias-primas no lixo eletrónico. Muitos destes são matérias perigosas para a saúde humana e para o meio ambiente, o que torna extremamente importante e até mesmo imperativo o tratamento adequado destes materiais, de forma a reduzir os riscos de poluição e ao mesmo tempo assegurar a recuperação de materiais valiosos para a indústria. Estima-se que em 435 quilotoneladas recolhidas de telemóveis, no

ano de 2016, estejam guardados apenas 9.4 biliões de euros em matéria-prima não recuperadas, isto sem falar no possível valor se tivessem sido inseridos no mercado de aparelhos usados, ao prolongar a vida útil do aparelho [29].

Em 2019, foram produzidas mundialmente cerca de 53,6 milhões de toneladas de lixo eletrónico, dos quais apenas 17.4% foram recolhidos e reciclados, o que se traduz num baixo valor recuperado face ao total possível. A Europa tem a maior taxa de reciclagem, com cerca de 42%. Portugal apresenta aproximadamente o mesmo valor considerados todos os tipos de resíduos. Infelizmente, a maior parte do lixo, não apenas eletrónico, continua a ser descartado, abrindo assim espaço para a intensificação da reciclagem, a nível nacional e mundial [6].

A gestão dos Resíduos de Equipamentos Elétricos e Eletrónicos compreende várias etapas: recolha, desmontagem, pré-processamento e processamento. A recolha consiste na receção dos REEE junto do consumidor ou com recurso a sistemas de recolha ou ecopontos. Posteriormente, os equipamentos são desmontados e cada componente/material separado e classificado. No caso de substâncias tóxicas, estas são neutralizadas e as demais materiais são transformados em matérias-primas, sendo reintroduzidos no ciclo de produção.

O aumento do uso de produtos eletrónicos por parte da sociedade mundial foi vindo a crescer exponencialmente, criando cada vez mais equipamentos utilitários. Tal facto conduziu a um avolumar deste tipo de descarte – lixo eletrónico. Pela sua natureza e composição, tal equipamento não pode ser descartado de qualquer forma. Se o fizerem no lixo comum, os materiais pesados podem contaminar os solos e os lençóis freáticos e conseqüentemente os seres vivos. Se procedermos à sua queima, esta ação acarreta danos ambientais e risco para a saúde humana.

Por este motivo, estes resíduos devem ser reciclados, de forma segura e especializada. A sua manipulação e processamento obedecem a regras e procedimentos adequados e seguros, para que os trabalhadores envolvidos no seu manuseamento estejam protegidos dos seus efeitos nefastos (intoxicação/doenças graves).

Tabela 2.2: Efeito de elementos (usados em equipamentos eletrônicos) na saúde humana.

Elementos	Componentes	Efeito na saúde
Chumbo	Computador Smartphone Televisor	Danos no sistema nervoso e sanguíneo.
Mercurio	Computador Monitores LCD	Danos cerebrais e no fígado.
Cadmio	Computador Baterias	Envenenamento; Danos nos ossos, rins, pulmões e sistema nervoso.
Arsénico	Smartphone	Doenças de pele; Prejudica sistema nervoso; Pode causar cancro do pulmão.
Berílio	Computador Smartphone	Cancro do pulmão.

O crescimento significativo da produção, consumo e descarte de equipamentos eletrônicos e o crescente problema de contaminação daí decorrente, conduziu à elaboração de leis específicas, em vigor em diversos países do mundo. Contudo, a existência de legislação não impede más práticas e atropelos legislativos e ambientais. Assim, a criação de leis pode não ser suficiente para atingir este propósito. É fundamental haver um compromisso para atingir a sustentabilidade, sendo para isso necessário consciencializar e educar as populações.

2.4 Aplicações existentes

De maneira a desenvolver uma boa aplicação, foi realizada a análise de aplicações existentes já no mercado, apresentadas na Figura 2.8. Nessa pesquisa foram encontradas várias aplicações móveis, na área da reciclagem de resíduos.

Grande parte das aplicações são de âmbito local, a nível de cidade ou município, e com uma funcionalidade muito precisa, restrita e simples. Maioritariamente remetem para a notificação atualizada e recorrente do horário e eventual local (domicílio/ecopontos) de recolha dos resíduos. Constituem, da mesma forma, fonte de informação e sensibilização no que respeita ao ciclo dos 4 R's (Reutilização, Reparação, Recondicionamento e Reciclagem).

Estas aplicações apresentam também uma vertente de carácter educativo reforçando a educação para o ambiente através das orientações para a separação dos resíduos. Existem igualmente aplicações que se orientam para a promoção/viabilização da revenda /oferta de produtos. É o caso de produtos de vestuário ou produtos eletrónicos, ou no caso de recolha e oferta, produtos alimentares.

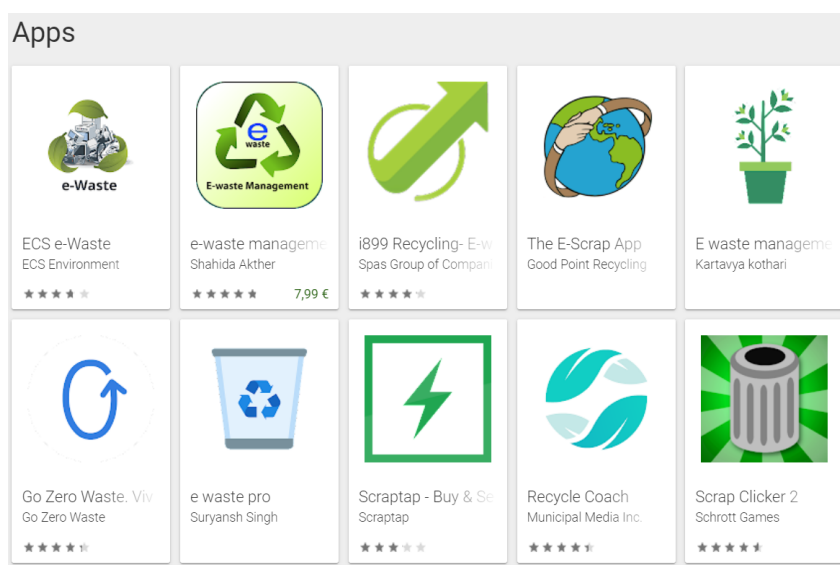


Figura 2.8: Aplicações de reciclagem de resíduos electrónicos na Google Play [7].

Tabela 2.3: Características das aplicações existentes no mercado.

Aplicações	Características						
	Agendamento	Mapa de entrega	Atribuição de pontos	Recompensa de descontos	Informação sobre resíduos	Ponto de recolha automático	Reconhecimento do resíduo IA
ECS e-Waste	✓	✓			✓		
e-waste management	✓	✓			✓		
i899 Recycling- E-waste Recycling		✓	✓				
The E-Scrap App	✓	✓					
E waste management		✓					
Scrap - Buy Sell Scrap		✓					
Recycle Coach	✓				✓		
SmartRecycle		✓	✓	✓	✓	✓	✓

Faz-se referência que a maioria das aplicações aludidas, ainda que se encontrem localizadas um pouco por todo o mundo, têm maior incidência nos USA, no Brasil, na China, na Índia e na Austrália.

No que se refere a aplicações móveis mais complexas e multifacetadas, que integram diversas etapas ou processos e gamificação, à semelhança do projeto desenvolvido, não se encontrou qualquer exemplo.

Capítulo 3

Metodologias

Neste terceiro capítulo vai ser descrita a metodologia de investigação usada e as metodologias de trabalho. Face às características do presente projeto adotou-se como metodologia de investigação o Modelo Design Science Research. Esta abordagem considera dois objetivos que se complementam e ocorrem em simultâneo: o desenvolvimento de um produto em resposta a um problema prático e a criação de conhecimentos científicos e técnicos. Optou-se pela metodologia Ágil, mais especificamente pela framework SCRUM, como ferramenta de trabalho. Esta framework é especialmente usada na realização de projetos complexos e que apresentem várias mudanças de objetivos e requisitos, pelo facto de proporcionar a avaliação do trabalho executado no final de cada sprint.

3.1 Metodologia de investigação

Modelo Design Science Research

Historicamente, a investigação científica apoiava-se em metodologias tradicionais que englobavam Experimento, Survey e Etnografia, processos não aplicáveis à conceção e construção de um produto. Tendo esta premissa em mente, a que abordagem e métodos recorrer na investigação do uso de um produto e das suas implicações práticas no contexto? O desenvolvimento e a investigação científica no uso de produtos

diverge da investigação de fenómenos naturais e sociais. No mundo atual, composto de realidades naturais e sociais e, de igual modo e cada vez mais, de realidades construídas - artificiais, a compreensão do artificial e a criação de conhecimento científico explicativo destas, requer novos paradigmas e ferramentas [30].

As ciências da engenharia e computação, por exemplo, caracterizam-se como ciências do projeto ou ciências do *design* ou ciências do artificial e apontam como objetivos principais produzir conhecimento resultante do próprio processo de concepção de produto/artefacto [30]. “as ciências naturais ocupam-se de como as coisas são [...], ao projeto interessa o que as coisas devem ser, a concepção de artefactos que realizem objectivos.” [31]. Assim, a ciência do *design* pressupõe a criação sistemática do conhecimento sobre e com o produto.

O *Design Science Research* é um paradigma que se define como abordagem epistemológica - metodológica para pensar e fazer e tem vindo a afirmar-se na área dos sistemas de informação. Este modelo de investigação apresenta dois objetivos essenciais e complementares:

- Desenvolver um produto para responder a um problema prático num contexto específico;
- Criar novos conhecimentos científicos e técnicos.

O conhecimento científico e técnico necessário à construção de um produto difere do conhecimento científico, uma vez que assenta no conhecimento sobre a arte de fazer. Na investigação - *design* (produto) verifica-se a articulação entre conhecimento técnico e o conhecimento científico, ou seja, entre produto e teoria.

Nesta perspetiva, os desenvolvimentos de novos produtos de sistema de informação sustentáveis integram-se no DSR. O paradigma DSR orienta a concepção, desenvolvimento, melhoria e implementação de produtos na resolução de problemas e desafios recorrentes. O desenvolvimento e avaliação deste tipo de produto recai no paradigma mencionado, o qual tem por propósito compreender o *design*/construção do produto e, conseqüentemente, avaliar a sua eficácia na resolução da situação/-problema social, organizacional ou tecnológico definido e na base da sua concepção. Representa, portanto, uma ajuda na concretização dos objetivos de informação de sistema verde. Assim sendo, contribui consistentemente para a sustentabilidade ambiental. Não obstante o potencial deste tipo de investigação no que se refere ao

sistema de informação verde, tem sido esquecida a relação entre o desenvolvimento do produto e os significativos impactos e contribuições no domínio ambiental.

O processo do DSR pode ser estruturado e definido de várias formas. Contudo, a estrutura, que é correntemente aceite e aplicada, foi desenvolvida por Hevner et al (2004) [32] e mais tarde atualizada pelo mesmo autor, facilitando apreender não só a natureza da DSR, mas também apropriar-nos do vocabulário descrito do modelo. A sua estrutura, demonstrada na Figura 3.1, consiste em 3 ciclos: Ciclo Relevância (ambiente/contexto), Ciclo Rigor (conhecimento base) e Ciclo *Design* (produto).

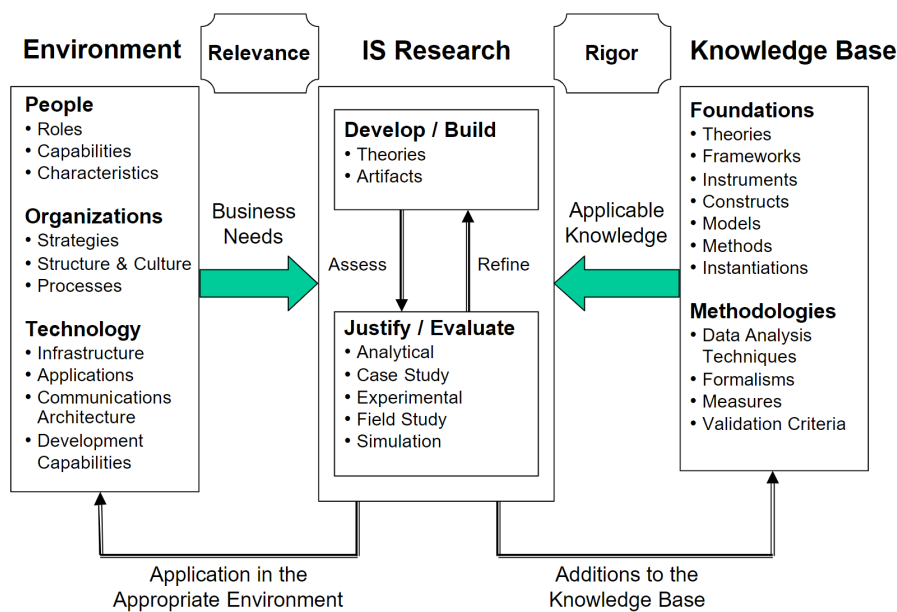


Figura 3.1: Estrutura de investigação de sistemas de informação [8].

O **Ciclo da Relevância** compreende a inter-relação entre atividades de *design* e a correspondente aplicação ambiental. Tal permite a emergência de novos produtos no contexto.

O **Ciclo do Rigor** liga atividades de *design* a conhecimentos e investigação existentes, integrando ou evoluindo esse conhecimento, ao adicionar os resultados do processo de investigação.

O **Ciclo do Design/Produto** consiste na interação entre a construção e a avaliação do produto a conceber, sendo que a avaliação do produto se pode concretizar de diferentes formas:

- Observação;
- Análise;
- Experimentação;
- Testagem;
- Descrição.

No que se refere ao presente trabalho de investigação, adotou-se o modelo da DSR de investigação por se apresentar como a metodologia mais adequada à conceção e construção de um produto, Figura 3.2.

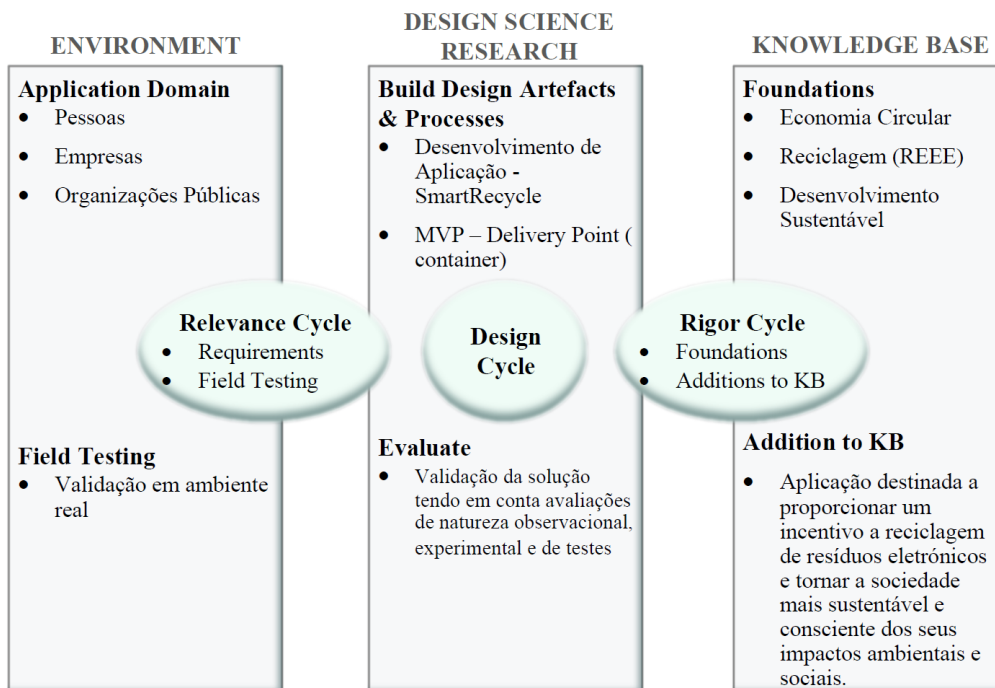


Figura 3.2: Metodologia DSR figura adaptada do artigo [8].

3.2 Metodologia de trabalho

Metodologia Ágil vs. Tradicionais

A escolha da metodologia do trabalho correta é o pilar de um bom funcionamento de um projeto. As principais metodologias no mercado dividem-se em ágeis

e tradicionais. Duas metodologias bastante similares mas com diferenças de funcionamento importantes, Figura 3.3. Embora ambas as abordagens tenham pontos positivos e negativos, fazer a escolha certa constitui um papel crucial. Os principais pontos a serem considerados nessa escolha são os seguintes:

- **Necessidade de negócios** – Impacto da implementação de requisitos especificados pelo cliente.
- **Percepção do cliente** – Perspetiva do cliente sobre o impacto nos negócios.
- **Cronograma do projeto** – Cronograma definido para a implementação do projeto em tempo real.

A metodologia tradicional de desenvolvimento de software baseia-se em fases pré-organizadas de desenvolvimento. Nessas, o fluxo de desenvolvimento é unidirecional: dos requisitos ao *design*, em seguida, ao desenvolvimento e depois aos testes e manutenção. Assim, cada fase tem produtos específicos e documentação detalhada que passou por um processo de revisão completo.

Esta abordagem é apropriada quando os requisitos são bem compreendidos, o que requer maior esforço na sua definição logo no início do projeto. Por outro lado, em sectores que apresentam rápidas mudanças, esta metodologia pode não ser a mais adequada, podendo não conseguir atingir os objetivos finais do projeto.

Ao contrário das abordagens tradicionais, as metodologias Ágil são precisas e amigáveis para o cliente. Os clientes têm a oportunidade de fazer modificações e ajustes contínuos.

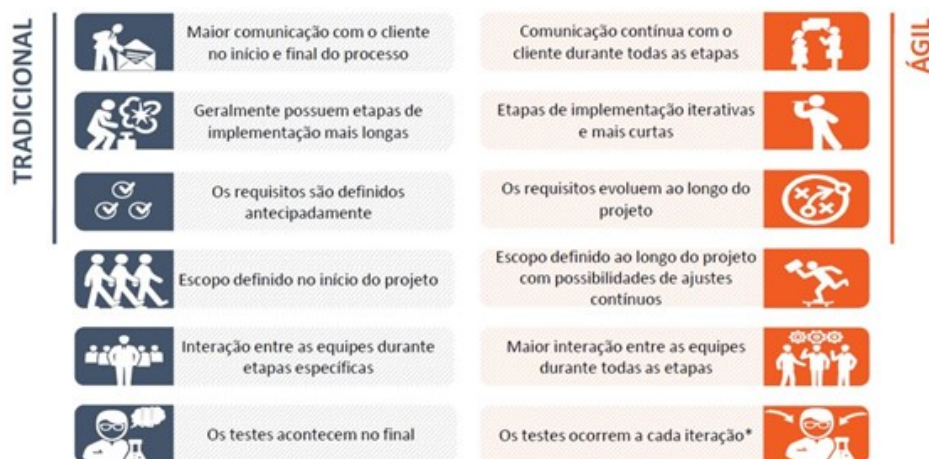


Figura 3.3: Diferenças entre as duas metodologias [9].

Destacam-se, de seguida, outras características da Metodologia Ágil, de grande relevância no desenvolvimento de trabalho e a sua comparação com a metodologia tradicional, na Figura 3.4, especialmente na área considerada:

- A visibilidade e transparência durante todo o processo de desenvolvimento;
- A adaptabilidade da metodologia ágil, possibilitando o planeamento iterativo e permitindo uma fácil adaptação no caso de mudança de requisitos;
- O planeamento e o *feedback* contínuo numa gestão ágil, promove a entrega de valor desde o início do projeto;
- Os riscos de produção de desperdício numa gestão ágil são minimizados.



Figura 3.4: Comparação entre metodologias [9].

Framework SCRUM

É um *framework* ágil que pode ser utilizado para gerir projetos complexos, tratando-se um método usado para fazer a organização, planeamento da gestão de projetos, inicialmente, relacionados à criação de softwares [33]. Porém, hoje em dia é possível aplicar a estratégia em diversos tipos de projeto. Esta *framework* requer um trabalho de entregas contínuas, evitando que problemas surgidos sejam apenas encontrados no final do processo. Consiste num conjunto de práticas e princípios a cumprir de forma a preservar a integridade do projeto, presentes na Figura 3.5.

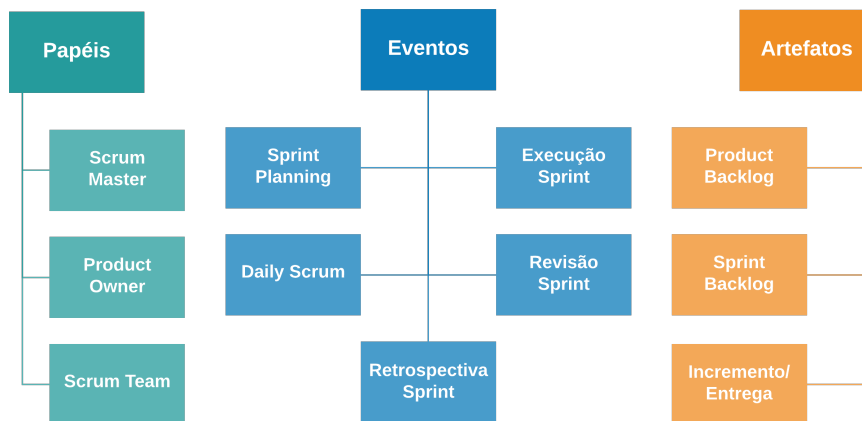


Figura 3.5: Práticas de SCRUM [10].

As atividades são divididas em sprints, ou seja, em etapas específicas que se referem ao conjunto de tarefas que deverão ser realizadas. O tamanho de cada sprint é relativo a cada empresa, sendo necessário adequar as tarefas num bloco de atividades, que podem demorar de uma a quatro semanas. Os membros da Scrum Team são autossuficientes, gerindo a responsabilidade, as suas tarefas, organizando-as da melhor forma e dispensando assim, a frequente supervisão.

Todavia, para que todo o processo funcione corretamente, existem três papéis/-funções importantes:

- **Scrum Master** – Responsável por facilitar o cumprimento dos princípios e práticas de *Scrum*, evitando interferências externas e protegendo de tudo o que possa atrapalhar o desenvolvimento.
- **Product Owner** – Responsável por determinar o entra e o que sai do projeto, as datas de entrega de cada tarefa e por coordenar a comunicação entre colaboradores.
- **Scrum Team** – Responsável pela formação das equipas e pelas tarefas que compõem um *sprint*. Para tal, deverá ter um bom pensamento estratégico, a fim de realizar uma engenharia de valor adequado, optando pelas tarefas mais importantes primeiro.

Ainda que pareça complexo, na *framework scrum* existem todos esses papéis para tornar o processo mais ágil, exposto na Figura 3.6. Desde as etapas que compõem o dia a dia até aos principais termos usados pelos participantes.

- ***Sprint Planning Meeting*** – Reunião rápida e objetiva destinada a organizar as tarefas a realizar no *sprint*.
- ***Product Sprint Backlog*** – Lista de todas as tarefas, ainda sem um cálculo de engenharia de valor, a serem desenvolvidas durante o projeto. Ficam registradas até que o Project Owner e a Scrum Team decidam se devem ou não serem feitas.
- ***Sprint Retrospective*** – Recapitulação do projeto com o intuito de identificar pontos fortes e possíveis pontos que precisam de melhorias ou de evolução.
- ***Daily Scrum*** – Reunião diária para o grupo verificar o que foi realizado no dia anterior e as tarefas a serem desenvolvidas no dia atual, assim como a existência de algum impedimento à execução do trabalho diário.
- ***Sprint Review Meeting*** – Reunião no final de cada *sprint* destinado a avaliar o progresso do projeto.

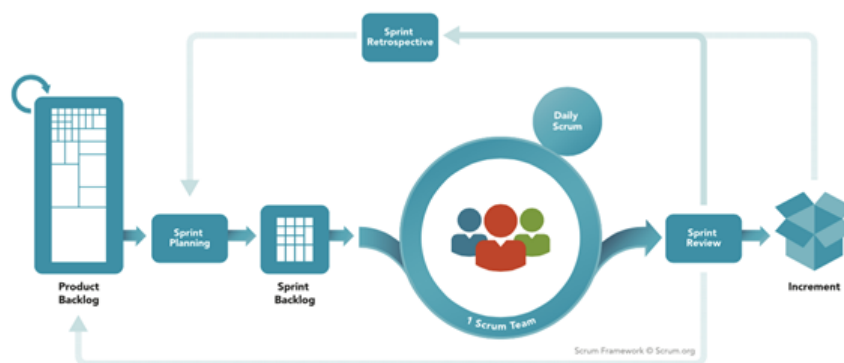


Figura 3.6: Processo de SCRUM [11].

Capítulo 4

Desenvolvimento de Software

Neste quarto capítulo vai ser apresentada a plataforma usada para o desenvolvimento do projeto e descrito o mesmo através de tabelas de requisitos, casos de uso, matrizes de rastreabilidade e diagramas de sequência.

4.1 OnBoarding

De seguida, será apresentada a plataforma Outsystems, realçando a sua arquitetura e componentes, mas também a sua funcionalidade, de forma pormenorizada e reveladora de todas as suas potencialidades, enquanto ferramenta de trabalho.

A plataforma de Outsystems assume-se como uma possibilidade de desenvolvimento rápido de aplicações para diversos dispositivos, usando para tal, de um ambiente visual de desenvolvimento *full-stack*, que pode ser observado na Figura 4.1.

Rapid Development and Integration		Quality Assurance	Deploy	Operate & Measure	Manage
Frontend Web Mobile		End-to-End Quality Assurance <ul style="list-style-type: none"> Impact Analysis Self Healing App Feedback Device Testing Testing Framework 	Deployment and Staging <ul style="list-style-type: none"> 1-Click Full Stack Deploy 24/7 Upgrades Process Upgrades Database Upgrades API Dependency Checking 	Client, Network and Server Analytics <ul style="list-style-type: none"> Frontend Analytics Logic and Workflow Analytics Database Analytics APIs and Extensions Analytics Processes and Schedulers Operation 	Manage Apps and Services <ul style="list-style-type: none"> Identity Management Users Roles Groups Apps Services Environments Servers Storage
Backend Logic Workflow Notifications					
Database Relational No SQL Caching					
Integrations Schedulers REST, SOAP SAP C#, JS, Java					
Infrastructure React C# / Net Java					

Figura 4.1: Full-stack Outsystems [12].

O desenvolvimento rápido de aplicações e até mesmo a automatização de tarefas manuais que são morosas, possibilitando aos programadores centrarem a sua função em tarefas mais inovadoras e de conceção, e executá-las de forma mais ágil e rápida. Deste modo, estamos a rentabilizar e a valorizar os conhecimentos e ação do programador, revelando-se, este método bem mais eficiente do que os tradicionais.

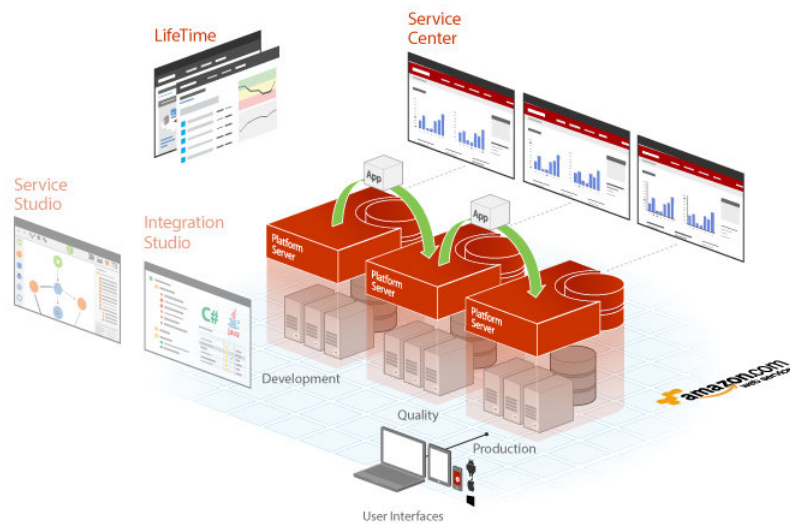


Figura 4.2: Componentes da plataforma Outsystems [13].

A plataforma OutSystems possui ferramentas, componentes e ambientes, presentes na Figura 4.2, que facilitam todo o processo de gestão de aplicações [34]:

- **Ambiente de desenvolvimento (*Service Studio*)** – É utilizado para o desenvolvimento da lógica da aplicação, interface, o fluxo do processo de negócio, políticas de segurança e o modelo de dados.
- ***Integration Studio*** – Ambiente de criação de componentes, que permitem à plataforma de OutSystems integrar sistemas e micros serviços. Estes componentes uma vez implantados podem ser reutilizados por todas as aplicações na plataforma.
- ***Platform Server*** – Núcleo da plataforma, onde são executadas as etapas necessárias para gerar, otimizar, compilar e implantar aplicações em um servidor de aplicações da *web* padrão ou empacotar em uma aplicação móvel nativa. O processo ocorre através de múltiplos serviços, começando pelo *Code Generator* gera código nativo em .NET, seguido pelos *Deployment Services*, que têm como função implantar a aplicação num servidor de aplicações *web*. Isto vai permitir que a aplicação seja consistentemente instalada em cada *front-end* da *server farm* (conjunto de computadores servidores). A Platform Server concilia, igualmente, de serviços de agendamento de tarefas e de gestão da aplicação no momento. Inclui também um *log service*, que monitoriza os erros e questões de *performance*.
- ***Mobile Apps Build Service*** – Serviço *Cloud* desenvolvido pela OutSystems que propicia o gerar de *aplicações mobile* para iOS e Android.
- ***Service Center*** – Ferramenta oferecida pela OutSystems de gestão via *web*, dirigida a controlar e monitorizar o funcionamento das aplicações (versões, utilizadores, erros, *e-mails*).
- ***Forge*** – Repositório de componentes para uso gratuito pelos utilizadores, criados e partilhados por membros da comunidade de OutSystems.

Para o desenvolvimento do projeto foi utilizado o processo de arquitetura de desenvolvimento de projeto concebida pelo OutSystems. Neste processo de *design* de arquitetura são utilizadas 3 etapas. A etapa *Disclose* é usada para a obtenção dos requisitos funcionais e não funcionais e descoberta dos casos de uso do utilizador, necessidades de integração e as expectativas de UX. Na etapa *Organize* são reagrupados os requisitos funcionais que estiverem conceptualmente relacionados e inseridos na sitio apropriado da *Architecture Canvas*, apresentado na Figura 4.3. Na última etapa, *Assemble*, são implementam os conceitos identificados.

A *Architecture Canvas* é uma ferramenta de arquitetura OutSystems para simplificar o projeto de Arquitetura Orientada a Serviços (SOA). Promove a abstração

de serviços reutilizáveis e o isolamento de módulos funcionais, demonstrado na Figura 4.4, proporcionando a manutenção das várias aplicações que reutilizam módulos comuns, durante o desenvolvimento do projeto. Diversas aplicações e módulos tem diferentes ciclos de vida e são desenvolvidos por equipas diferentes. Novas aplicações tem a tendência a evoluir rapidamente, enquanto módulos altamente reutilizados mudam muito mais lentamente. Logo, é benéfico realizar uma correta projeção da arquitetura da aplicação para diminuir ao mínimo as dependências e o impacto geral de alterações nos módulos que compõem a aplicação. Este procedimento torna o *design* de arquitetura Outsystems de mais baixo custo, mais fácil de manter e de progredir.

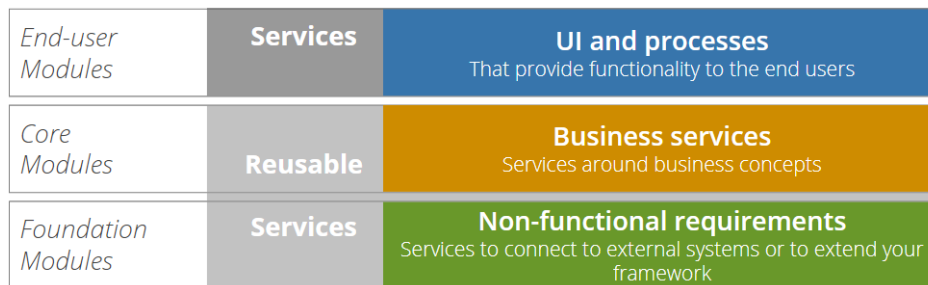


Figura 4.3: Camadas da aplicação [14].

O *Architecture Canvas* é usado em duas etapas diferentes do projeto de arquitetura:

1. *Identify Concepts*

Recolher requisitos funcionais, não funcionais e de integração e requisitos de arquitetura de forma estruturada e sistemática.

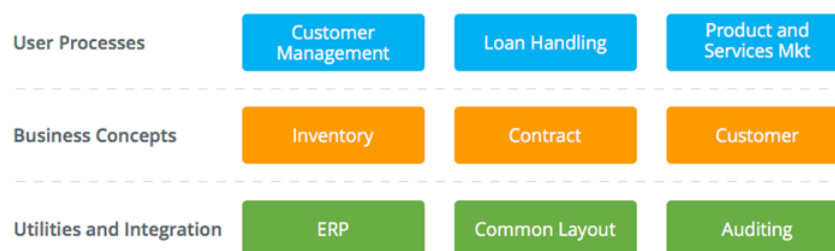


Figura 4.4: Camadas da arquitetura [14].

2. *Define Modules*

Desenvolver os módulos que implementam os conceitos identificados.

Este é um processo contínuo durante todo o desenvolvimento do projeto, não podendo ser um evento único. A arquitetura deve ser iterada, conforme demonstrado na Figura 4.5, percorrendo estas duas etapas, dependendo da evolução da própria solução e de novos conceitos e necessidades emergentes ao projeto.

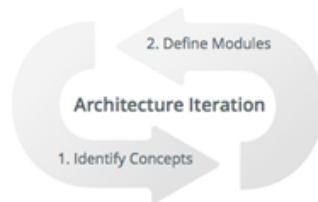


Figura 4.5: Iteração da arquitetura [14].

4.2 Execução

Numa fase inicial do desenvolvimento foi realizado o levantamento de requisitos funcionais e não funcionais para cada sistema distinto. Estes requisitos serão explanados de forma mais exaustiva posteriormente. Desses requisitos foi elaborada a arquitetura de Outsystems do projeto, começando pela etapa de *Organize* ilustrada na Figura 4.6.

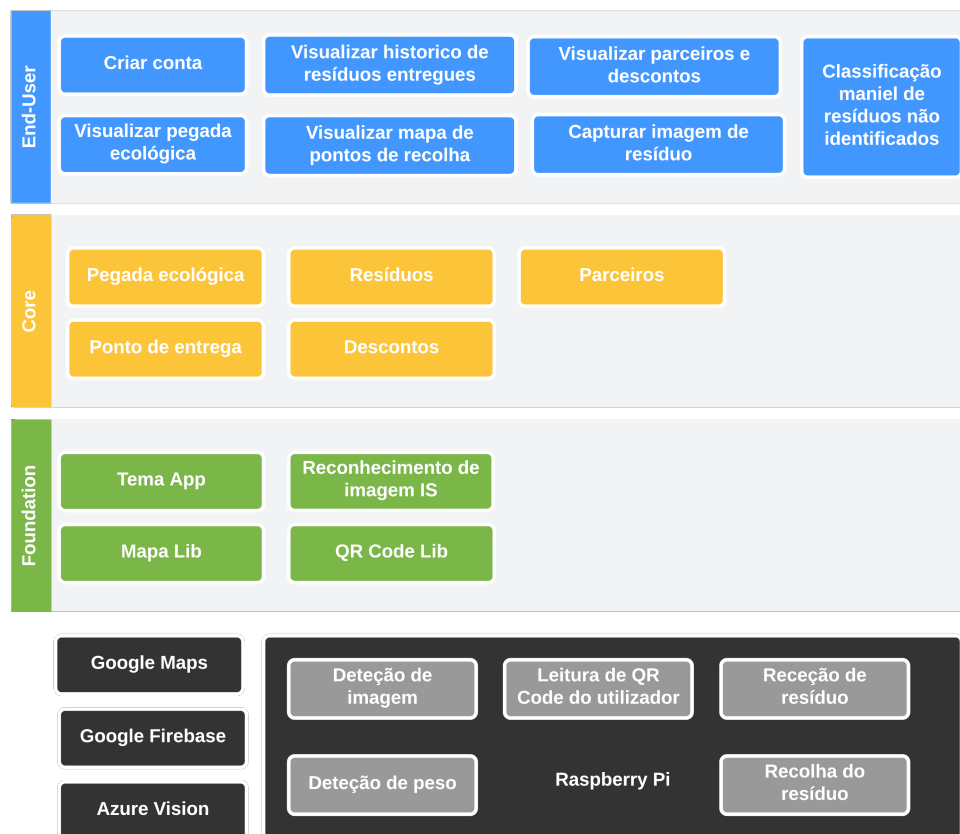


Figura 4.6: Diagrama de arquitetura de Outsystems *Organize*.

Por fim, foram agrupados todos os requisitos e diferenciados os sistemas que seriam desenvolvidos e as suas ligações de comunicação, tanto entre si como com os atores. Procedeu-se, também, a elaboração do diagrama de arquitetura final do projeto, exibido na Figura 4.7. Os diferentes sistemas desenvolvidos são aprofundados, de seguida no relatório.

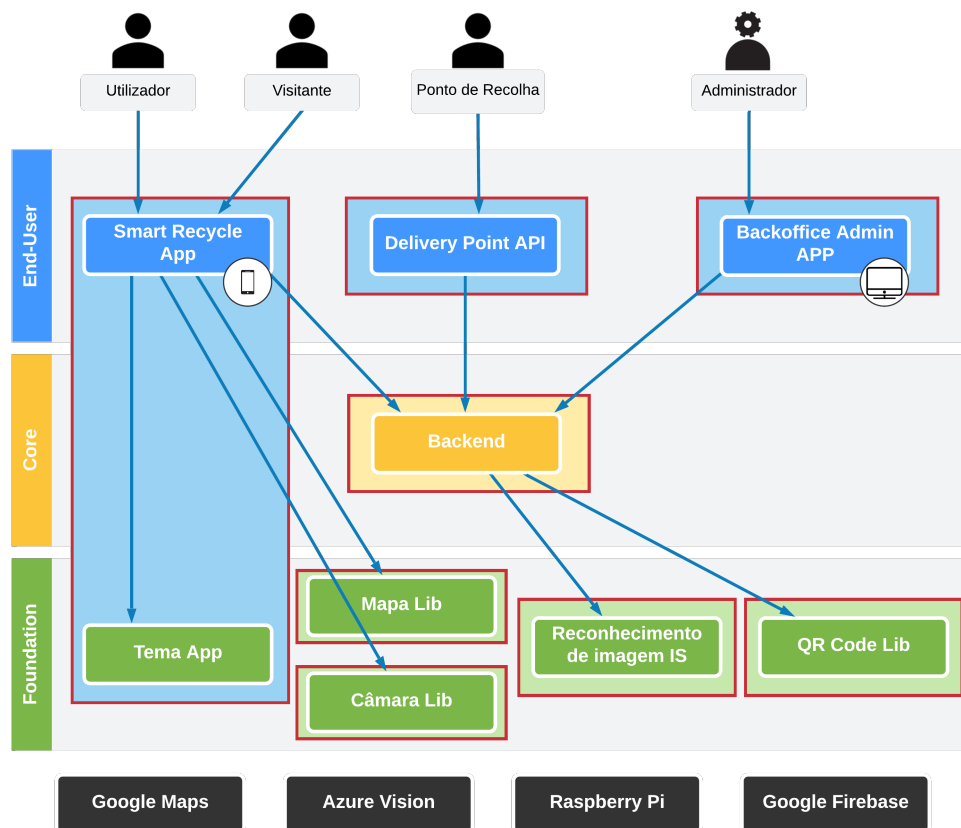


Figura 4.7: Diagrama de arquitetura do Outsistemas (DAO).

Após o levantamento feito, foi possível a construção do modelo de dados, apresentado na Figura 4.8, indispensável a implementação de todas as funcionalidades e satisfação dos casos de uso definidos. Porém, todo o desenvolvimento até este ponto, foi sendo revisto e alterado durante o período do desenvolvimento do projeto, seguindo a *framework* SCRUM.

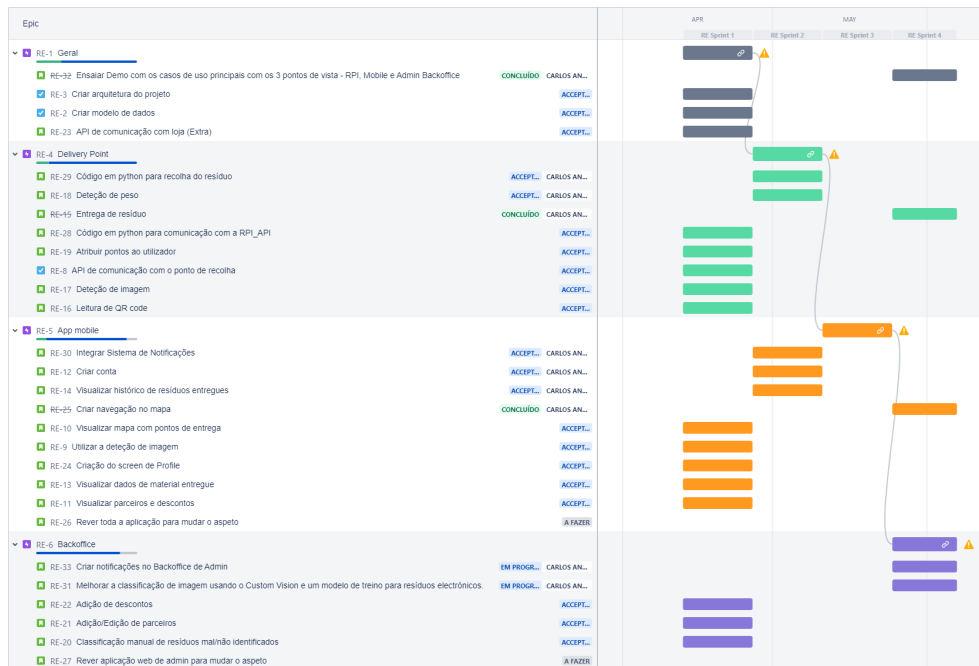


Figura 4.9: Tarefas realizadas em cada *sprint*.

4.2.1 Atores

Durante a realização do presente trabalho foram considerados como atores todos aqueles que por algum motivo terão, aquando da implementação final do sistema desenvolvido, alguma interação com o mesmo. Assim, foram considerados os seguintes atores:

- **Visitante**

Qualquer utilizador que interaja com a aplicação sem possuir qualquer tipo de permissões avançadas recebe a designação de visitante dado a sua interação com o sistema ser limitada. Tem a capacidade de utilizar a análise de resíduo, visualizar de parceiros e descontos, visualizar o mapa com os pontos de recolha e navegação, visualizar resíduos aceites e obter informação sobre o seu impacte ambiental. É também facultada a opção de se registar (caso o faça torna-se um utilizador).

- **Utilizador**

O sistema desenvolvido pretende facultar serviços ao utilizador, podendo este ser considerado o ator fundamental do sistema. Tem a capacidade de utilizar a análise de resíduo, visualizar parceiros e descontos, adquirir vales de descontos através de pontos, visualizar o mapa com os pontos de recolha e navegação,

visualizar pegada ecológica, visualização de resíduos aceites e informação sobre o seu impacte ambiental, histórico de resíduos entregues, visualizar resíduos analisados juntamente com QR *Code* a exibir para a entrega, aquisição de pontos na entrega de resíduos e possibilidade de apagar conta.

- **Administrador**

O administrador é o ator com o grau de permissões mais elevado. Tendo por responsabilidade a manutenção e configuração do sistema. Tendo a capacidade de atribuição de pontos aos utilizadores aquando da ocorrência de erros na entrega de resíduos (podendo atribuir ou não pontos aos utilizadores), introdução e edição de novos parceiros e descontos, introdução e edição de pontos de recolha de resíduos e adição e edição de novos resíduos aceites.

- ***Delivery Point***

O *Delivery Point* está encarregue de receber o resíduo do utilizador, disponibilizar a confirmação da entrega do mesmo e de atribuição de pontos pelo resíduo.

4.2.2 Sistema SmartRecycle

Este sistema corresponde à aplicação móvel, com o mesmo nome. Aqui serão expostos e demonstrados os requisitos da aplicação, os seus casos de uso, *wireframes* e, por fim, diagramas de sequência dos casos de uso da mesma.

Tabela 4.1: Tabela de requisitos da aplicação móvel SmartRecycle.

ID	Tipo	Descrição
1.F.SR	F	Deverá ter um sistema de autenticação.
1.1.F.SR	F	Deverá permitir o registo de um novo utilizador.
1.2.F.SR	F	Deverá possibilitar a eliminação de conta.
1.3.F.SR	F	Deverá possibilitar o acesso anónimo.
2.F.SR	F	Deverá possibilitar a visualização no mapa dos pontos de recolha.
3.F.SR	F	Deverá fornecer o histórico de resíduos entregues do utilizador.
4.F.SR	F	Deverá fornecer os descontos adquiridos pelo utilizador.
5.F.SR	F	Deverá fornecer informação sobre os resíduos aceites.
5.1.F.SR	F	Deverá fornecer informação sobre o impacto ambiental.
5.2.F.SR	F	Deverá fornecer informação sobre a pegada ecológica.
6.NF.SR	NF	Deverá incentivar a reciclagem através de um sistema de descontos.
7.F.SR	F	Deverá fornecer informação sobre os parceiros disponíveis.
7.1.NF.SR	NF	Deverá associar os comerciantes aderentes a uma imagem “verde” e “sustentável”.
7.2.NF.SR	NF	Deverá associar os comerciantes a uma plataforma colaborativa associada ao projeto.
8.F.SR	F	Deverá ter funcionamento <i>offline</i> .
9.F.SR	F	Deverá ter captura de imagens.
10.F.SR	F	Deverá notificar o utilizador de eventos no sistema.
11.NF.SR	NF	Deverá ter um tema de <i>User Interface</i> .
12.NF.SR	NF	Deverá comunicar com a <i>Backend</i> .
12.1.NF.SR	NF	Deverá usar API's de acordo com as especificações da <i>Backend</i> para comunicação.
12.2.NF.SR	NF	Deverá conhecer o modelo de dados.
13.F.SR	F	Deverá fornecer capacidade de adquirir descontos.
14.F.SR	F	Deverá fornecer os resíduos analisados na aplicação.

Cenários

De seguida, são apresentados os diversos cenários do sistema, onde estão representadas as diferentes interações dos atores com o sistema.

Tabela 4.2: Cenário realizar login na aplicação.

ID e Nome	CCU-1 (Cenário de Caso de Uso): <i>Login</i> na aplicação
Sumário	Um utilizador registado pretende realizar <i>login</i> na aplicação com uma conta já existente.
Ator	Utilizador registado
Cenário	<ol style="list-style-type: none"> O utilizador abre a app; Escolhe a opção de <i>login</i> no ecrã de apresentação da aplicação; Preenche os campos apresentados.
Cenário Alternativo	<ol style="list-style-type: none"> O utilizador abre a app; Escolhe o <i>Profile</i> no menu de navegação; Carrega no botão “Already have an account?”; Preenche os campos apresentados.

Tabela 4.3: Cenário criação de conta na aplicação.

ID e Nome	CCU-2: Registrar na aplicação
Sumário	Um utilizador não registado cria conta na aplicação para usufruir de todas as funcionalidades.
Ator	Visitante
Cenário	<ol style="list-style-type: none"> 1. O utilizador abre a app; 2. Escolhe a opção de criar conta no ecrã de apresentação da aplicação; 3. Preenche os campos apresentados; 4. Carrega no botão “Registo”; 5. Redirecionado para o ecrã de <i>login</i>.
Cenário Alternativo	<ol style="list-style-type: none"> 1. O utilizador abre a app; 2. Escolhe o <i>Profile</i> no menu de navegação; 3. Preenche os campos apresentados.

Tabela 4.4: Cenário eliminar conta na aplicação.

ID e Nome	CCU-3: Eliminar conta na aplicação
Sumário	Um utilizador registado pretende eliminar a sua conta na aplicação.
Ator	Utilizador registado
Cenário	<ol style="list-style-type: none"> 1. O utilizador abre a app; 2. Escolhe o <i>Profile</i> no menu de navegação; 3. Carregar no botão de <i>Settings</i>; 4. Carrega no botão “Delete your account”; 5. Realiza a confirmação no <i>pop-up</i> exibido.

Tabela 4.5: Cenário análise de resíduo.

ID e Nome	CCU-4: Análise de resíduo pelo visitante
Sumário	Um utilizador utiliza a aplicação SmartRecycle para analisar um resíduo.
Ator	Visitante, Utilizador registado
Cenário	<ol style="list-style-type: none"> 1. O utilizador abre a app; 2. Carrega no botão de <i>scan</i>; 3. Utilizada a classificação de imagem; 4. Apresentação de informação sobre o resíduo analisado.
Cenário Alternativo	<ol style="list-style-type: none"> 1. O utilizador abre a app; 2. Carrega no botão de <i>scan</i>; 3. Utilizada a classificação de imagem; 4. Receção de notificação de resíduo analisado; 5. Apresentação de informação sobre o resíduo analisado; 6. Apresentação do QR <i>Code</i> gerado; 7. Entrada de dados na tabela <i>Analised</i> da base de dados; 8. Guardar QR <i>Code</i> em <i>LocalStorage</i>.

Tabela 4.6: Cenário mapa de pontos de recolha e navegação.

ID e Nome	CCU-5: Mapa de pontos de recolha
Sumário	Um utilizador não registado utiliza a aplicação SmartRecycle para visualizar a localização dos pontos de recolha.
Ator	Visitante, Utilizador registado
Cenário	<ol style="list-style-type: none"> 1. O utilizador abre a app; 2. Escolher o <i>Map</i> no menu de navegação; 3. Visualizar a localização dos pontos de recolha.

Tabela 4.7: Cenário visualizar histórico de resíduos entregues.

ID e Nome	CCU-6: Histórico de resíduos entregues
Sumário	Um utilizador registado pretende rever os resíduos entregues.
Ator	Utilizador registado
Cenário	<ol style="list-style-type: none"> 1. O utilizador abre a app; 2. Escolhe o <i>Profile</i> no menu de navegação; 3. Carrega no botão “Waste Delivered” 4. Apresentada a lista de resíduos e informação sobre cada um.

Tabela 4.8: Cenário aquisição de descontos.

ID e Nome	CCU-7: Aquisição de descontos
Sumário	Um utilizador registado utiliza a aplicação SmartRecycle para gastar os pontos obtidos para obter descontos em diversos parceiros.
Ator	Utilizador registado
Cenário	<ol style="list-style-type: none"> 1. O utilizador abre a app; 2. Escolhe o Partners no menu de navegação; 3. Apresentação de lista de parceiros e informação sobre cada um; 4. Carrega num parceiro; 5. Escolher o desconto pretendido; 6. Carrega no botão “Get Discount”; 7. Apresentação de notificação de sucesso; 8. Gerar QR Code do desconto; 9. Entrada de dados na tabela UserDiscounts da base de dados; 10. Guardar QR Code em <i>LocalStorage</i>.

Tabela 4.9: Cenário visualizar resíduos aceites.

ID e Nome	CCU-8: Visualizar resíduos aceites
Sumário	Um utilizador registado utiliza a aplicação SmartRecycle pretende saber que tipos de resíduos são aceites pela aplicação.
Ator	Utilizador registado
Cenário	<ol style="list-style-type: none"> 1. O utilizador abre a app; 2. Escolhe o Waste no menu de navegação; 3. Apresentação de lista de resíduos; 4. Carrega num resíduo; 5. Apresentação informação sobre o mesmo.

Tabela 4.10: Cenário visualizar pegada ecológica.

ID e Nome	CCU-9: Visualizar pegada ecológica
Sumário	Um utilizador registado utiliza a aplicação SmartRecycle pretende saber a quantidade de resíduos que entregou.
Ator	Utilizador registado
Cenário	<ol style="list-style-type: none"> 1. O utilizador abre a app; 2. Escolhe o <i>Profile</i> no menu de navegação; 3. Apresentada as quantidades de resíduos e materiais entregues.

Tabela 4.11: Cenário visualizar descontos adquiridos.

ID e Nome	CCU-10: Visualizar os descontos adquiridos
Sumário	Um utilizador registado utiliza a aplicação SmartRecycle pretende visualizar e usar os descontos adquiridos
Ator	Utilizador registado
Cenário	<ol style="list-style-type: none"> 1. O utilizador abre a app; 2. Escolhe o <i>Profile</i> no menu de navegação; 3. Carrega no botão “Discounts to use”; 4. Apresentação da lista de descontos adquiridos; 5. Carrega num desconto; 6. Apresentação do QR <i>Code</i> do desconto.

Tabela 4.12: Cenário visualizar resíduos analisados.

ID e Nome	CCU-11: Visualizar os resíduos analisados
Sumário	Um utilizador registado utiliza a aplicação SmartRecycle pretende visualizar os resíduos previamente analisados
Ator	Utilizador registado
Cenário	<ol style="list-style-type: none"> 1. O utilizador abre a app; 2. Escolhe o <i>Profile</i> no menu de navegação; 3. Carrega no botão “Waste Analised”; 4. Apresentação da lista de resíduos previamente analisados; 5. Carrega num resíduo; 6. Apresentação do QR <i>Code</i> do resíduo.

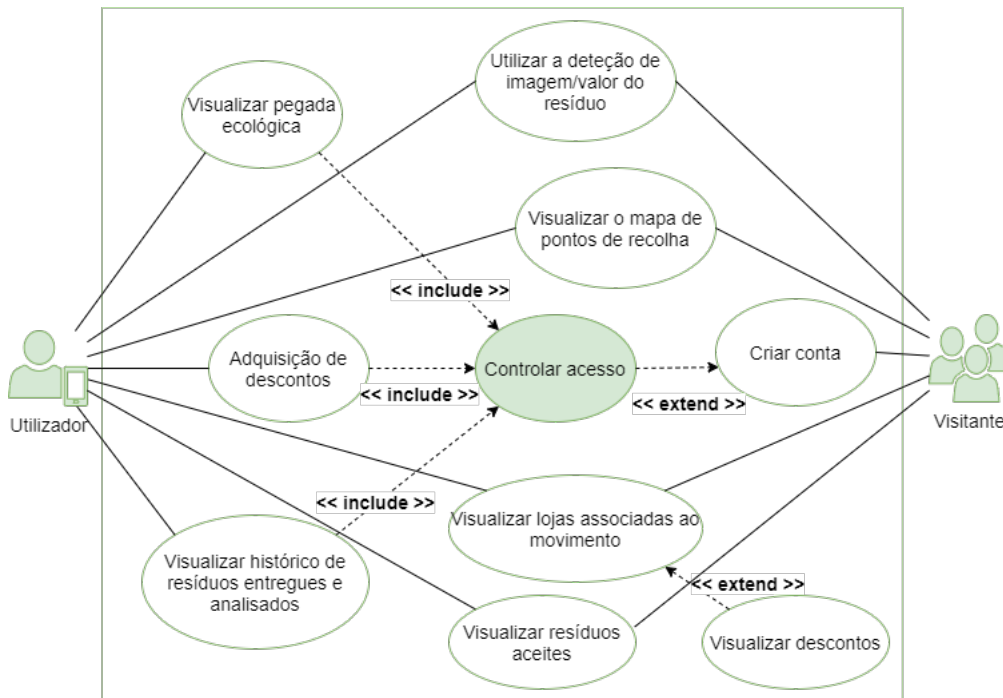


Figura 4.10: Diagrama de casos de uso da aplicação móvel.

Wireframes

Através dos requisitos da aplicação levantados, foi possível realizar os desenhos dos diferentes *Wireframes*, tendo sido utilizada a ferramenta Balsamiq. São apresentados os que melhor respondem aos requisitos.

Como Visitante e Utilizador, ao aceder a aplicação pela primeira vez.

Detalhes: na primeira vez, esboçado na Figura 4.11, para aceder à aplicação, após a sua instalação, é apresentada informação sobre a mesma e o seu funcionamento.

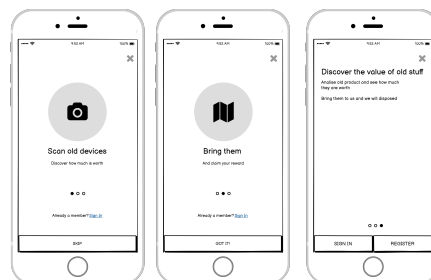


Figura 4.11: *Wireframe* de apresentação da aplicação.

CCU1. Como Utilizador, quero fazer *login* na aplicação para aceder a todas as funcionalidades disponibilizadas.

Detalhes: o *login*, esboçado na Figura 4.12, é realizado através de uma conta *e-mail*, caso já a tenha registado anteriormente. Existe a hipótese de ser redirecionado para o ecrã de registo.

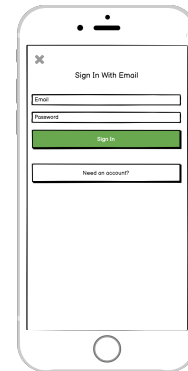


Figura 4.12: Wireframe de *login*.

CCU2. Como Visitante, quero registar-me na aplicação para aceder a todas as funcionalidades disponibilizadas.

Detalhes: o registo, esboçado na Figura 4.13, é realizado através de uma conta email. Este é feito quando se acede à aplicação pela primeira vez, porém existe a possibilidade de realizar o *login*, caso já possua uma conta.

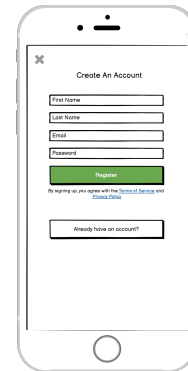


Figura 4.13: Wireframe de registo.

CCU3. Como Utilizador, quero realizar o fecho da minha conta na aplicação.

Detalhes: para eliminar conta, esboçado na Figura 4.14, é realizada através do ecrã de *settings*, encontrado no menu de *profile*. Igualmente, que contém as definições de notificações.

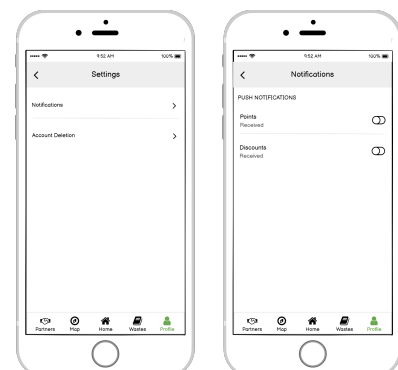


Figura 4.14: Wireframe de *Settings*.

CCU4. Como Visitante e Utilizador, quero realizar à análise de um resíduo.

Detalhes: para analisar, esboçado na Figura 4.15, basta tirar uma fotografia ao resíduo e de seguida é recebida a notificação de conclusão da análise, sendo redirecionado para o ecrã de informação do resíduo.

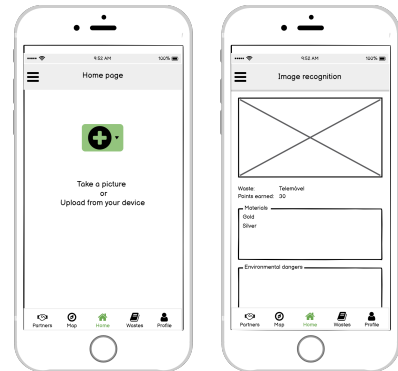


Figura 4.15: Wireframe de *Homepage* e informação da imagem analisada.

CCU5. Como Visitante e Utilizador, quero realizar a entrega de um resíduo.

Detalhes: para entrega, esboçado na Figura 4.16, utiliza-se o mapa integrado na aplicação, para escolher e navegar para o *Delivery Point* de resíduos mais próximo.

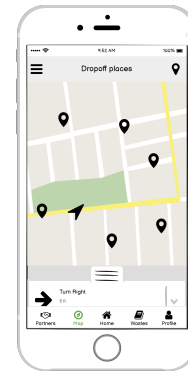


Figura 4.16: Wireframe do mapa de pontos de recolha.

CCU6. Como Utilizador, quero realizar a visualização do histórico de resíduos entregues.

CCU9. Como Utilizador, quero visualizar a pegada ecológica.

CCU11. Como Utilizador, quero visualizar os resíduos analisados para entrega.

Detalhes: para visualizar os resíduos entregues, esboçado na Figura 4.17, escolhe-se a opção de *Waste Delivered* presente no menu de *profile*. No ecrã de *profile* são apresentadas as métricas de quantidades de resíduos entregues e os seus materiais mais importantes. Por último, é possível visualizar os resíduos analisados e os seus respetivos QR Code, a apresentar no *Delivery Point*.

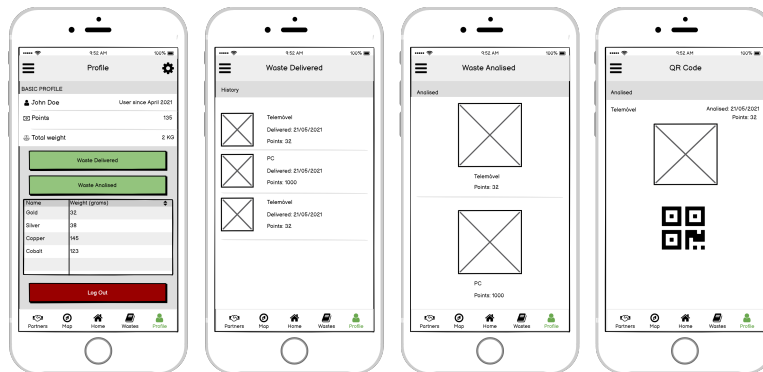


Figura 4.17: Wireframe de profile.

CCU7. Como Utilizador, quero adquirir descontos.

CCU10. Como Utilizador, quero visualizar os descontos já adquiridos para uso.

Detalhes: para aquisição dos descontos, esboçado na Figura 4.18, é necessário navegar para o ecrã de *partners*, escolher um deles e visualizar os descontos disponibilizados. No mesmo ecrã é possível visualizar os descontos já obtidos e o seu *QR Code* para uso.

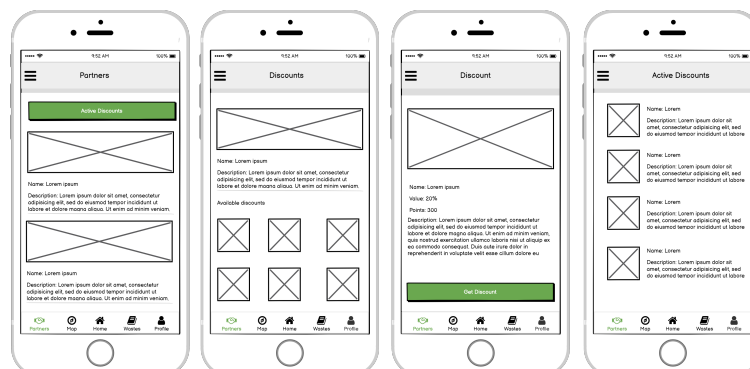


Figura 4.18: Wireframe de partners e discounts.

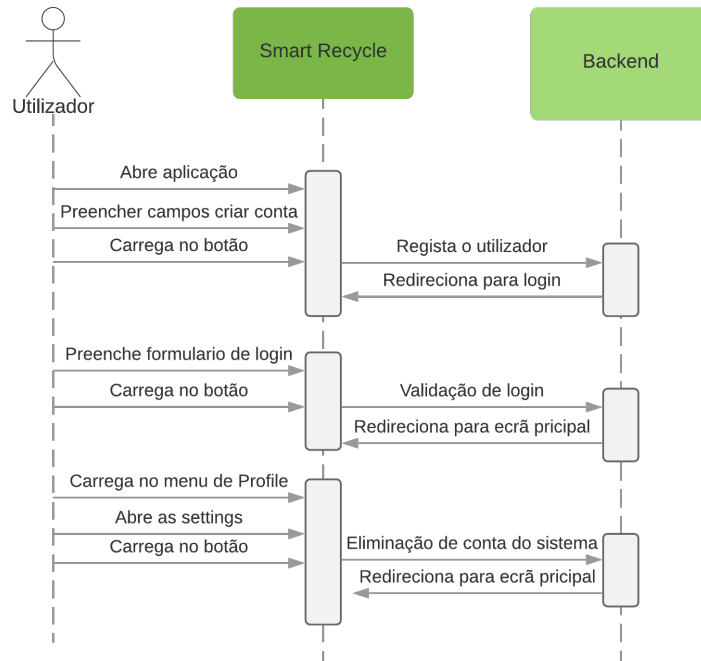


Figura 4.20: DS-1:Diagrama de seqüência dos casos de uso 1,2 e 3.

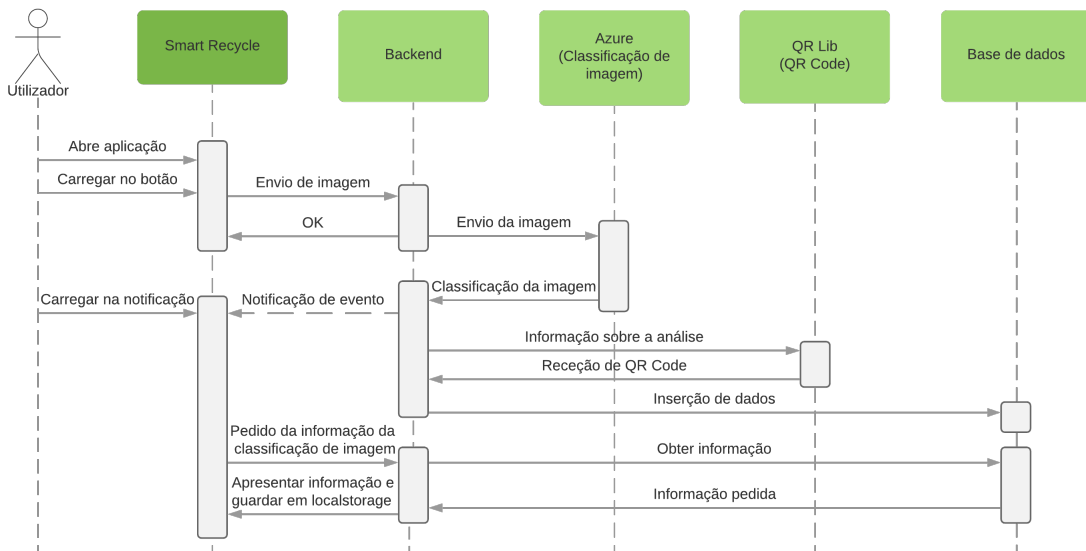


Figura 4.21: DS-2:Diagrama de seqüência do caso de uso 4.

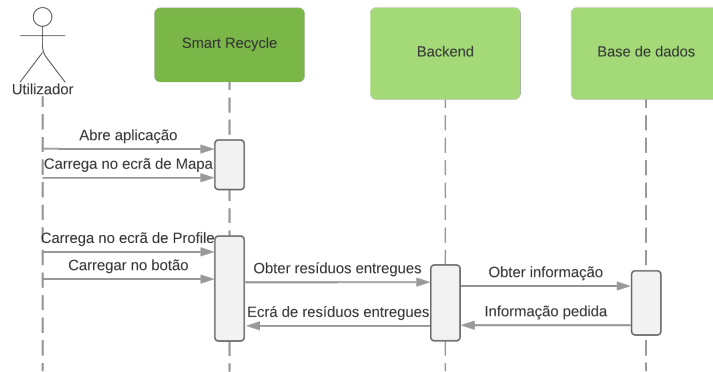


Figura 4.22: DS-3:Diagrama de sequência dos casos de uso 5 e 6.

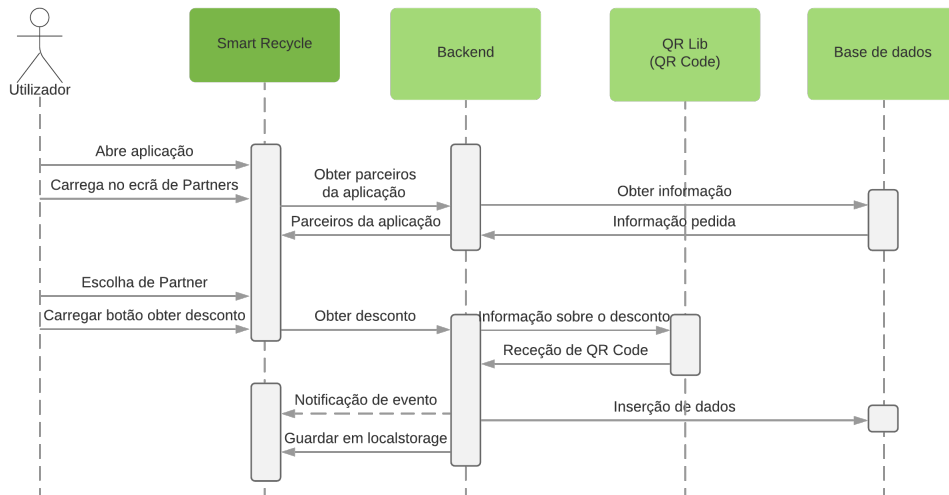


Figura 4.23: DS-4:Diagrama de sequência do caso de uso 7.

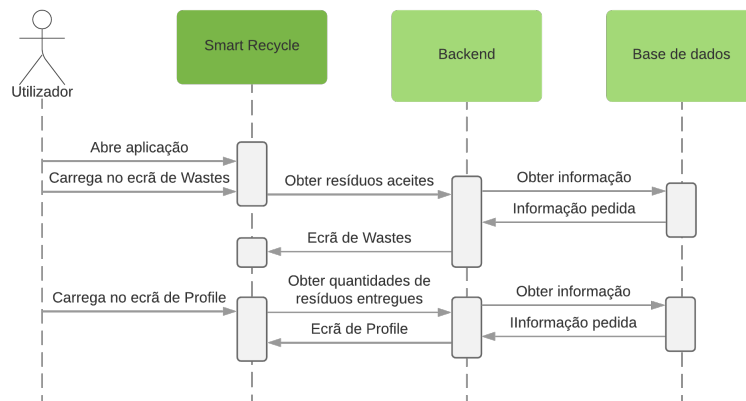


Figura 4.24: DS-5:Diagrama de sequência dos casos de uso 8 e 9.

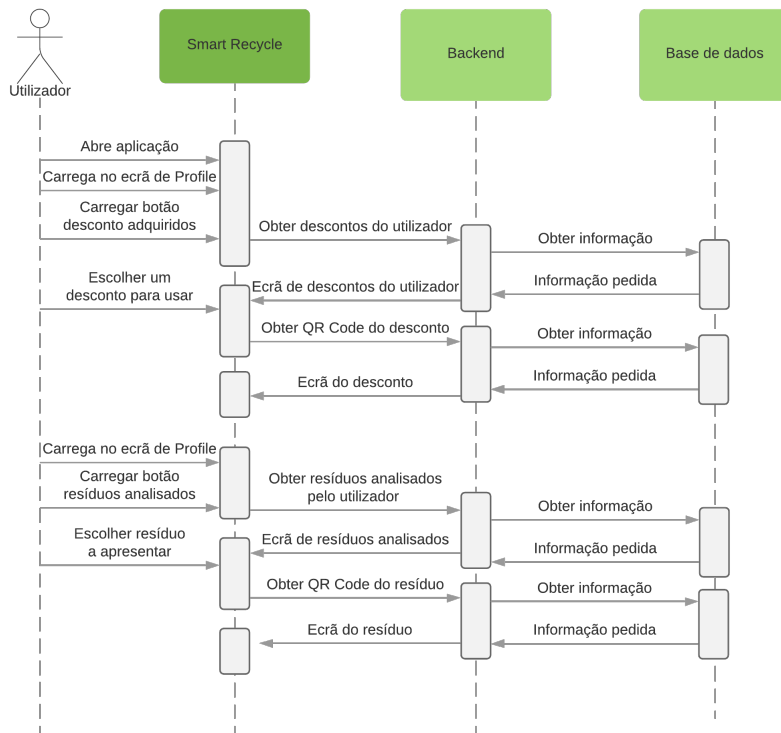


Figura 4.25: DS-6:Diagrama de sequência dos casos de uso 10 e 11.

Tabela 4.14: Matriz de rastreabilidade entre casos de uso e diagrama de sequência.

CU.ID	DS-1	DS-2	DS-3	DS-4	DS-5	DS-6
CCU-1	✓					
CCU-2	✓					
CCU-3	✓					
CCU-4		✓				
CCU-5			✓			
CCU-6			✓			
CCU-7				✓		
CCU-8					✓	
CCU-9					✓	
CCU-10						✓
CCU-11						✓

4.2.3 Sistema *Backoffice*

Este sistema foi desenvolvido para ser usado pelo administrador de toda a aplicação, sendo ele responsável pela análise manual de resíduos entregues mal identificados, pela edição de informação sobre os parceiros da aplicação e correspondentes descontos disponíveis, assim como pela edição de novos pontos de recolha de resíduos e de novos resíduos aceites pela aplicação.

Tabela 4.15: Tabela de requisitos de *Backoffice*.

ID	Tipo	Descrição
1.F.BO	F	Deverá ter um mecanismo de autenticação.
2.F.BO	F	Deverá ter análise manual de resíduos não identificados pelo <i>Delivery Point</i> .
3.F.BO	F	Deverá permitir a edição de novos parceiros e descontos.
4.F.BO	F	Deverá permitir a edição de pontos de recolha.
5.F.BO	F	Deverá permitir a edição de resíduos.
6.F.BO	F	Deverá notificar o administrador de eventos no sistema.

Cenários

Apresentam-se os diversos cenários do sistema, onde estão representadas as diferentes interações dos atores com o sistema.

Tabela 4.16: Cenário realizar autenticação.

ID e Nome	CCU-1: Login em <i>Backoffice</i>
Sumário	O administrador do sistema pretende realiza <i>login</i> no <i>Backoffice</i> da aplicação.
Ator	Administrador
Cenário	<ol style="list-style-type: none"> 1. O administrador acede a página <i>web</i>; 2. Preenche os campos apresentados.

Tabela 4.17: Cenário análise manual.

ID e Nome	CCU-2: Análise manual <i>Backoffice</i>
Sumário	O administrador do sistema realiza a avaliação dos resíduos não identificados pelo <i>Delivery Point</i> .
Ator	Administrador
Cenário	<ol style="list-style-type: none"> 1. O administrador acede a página <i>web</i>; 2. Recebe notificação de novos resíduos não identificados; 3. Apresentação da lista de resíduos para analisar; 4. Possibilidade de aceitar/rejeitar o resíduo.

Tabela 4.18: Cenário edição de parceiros e descontos.

ID e Nome	CCU-3: Parceiros e descontos no <i>Backoffice</i>
Sumário	O administrador do sistema edita a informação de parceiros e de descontos.
Ator	Administrador
Cenário	<ol style="list-style-type: none"> 1. O administrador acede a página <i>web</i>; 2. Escolhe o <i>Partners tab</i> no menu de navegação; 3. Escolhe um parceiro para editar a sua informação; 4. Escolhe o <i>Discount tab</i> no menu de navegação; 5. Adiciona um novo desconto associado ao parceiro.

Tabela 4.19: Cenário edição de pontos de recolha.

ID e Nome	CCU-4: Pontos de recolha no <i>Backoffice</i>
Sumário	O administrador do sistema edita a informação dos pontos de recolha.
Ator	Administrador
Cenário	<ol style="list-style-type: none"> 1. O administrador acede a página <i>web</i>; 2. Escolhe o <i>Delivery Point tab</i> no menu de navegação; 3. Escolhe um <i>Delivery Point</i> para editar a sua informação.

Tabela 4.20: Cenário edição de resíduos aceites.

ID e Nome	CCU-5: Resíduos no <i>Backoffice</i>
Sumário	O administrador do sistema edita a informação de resíduos aceites.
Ator	Administrador
Cenário	<ol style="list-style-type: none"> 1. O administrador acede a página <i>web</i>; 2. Escolhe o <i>Wastes tab</i> no menu de navegação; 3. Escolhe um resíduo para editar a sua informação.

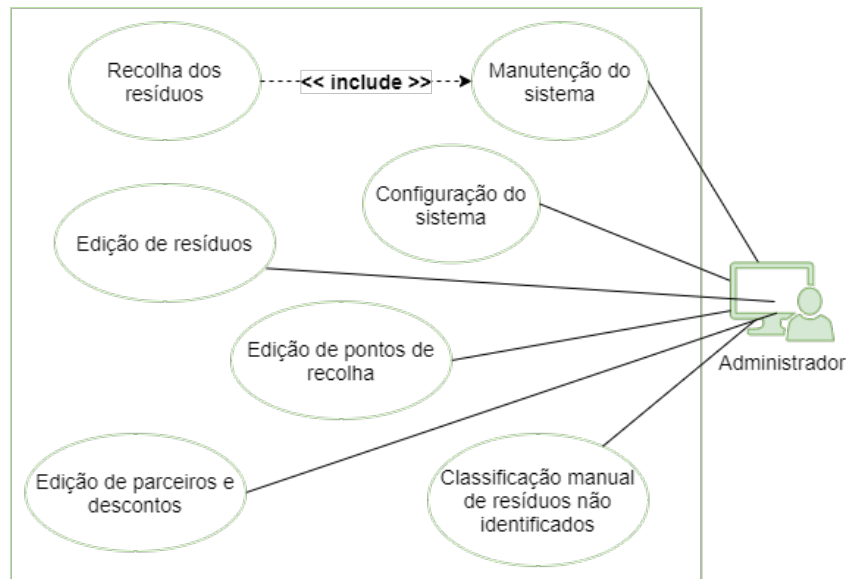


Figura 4.26: Diagrama de casos de uso do administrador.

Wireframes

Através dos requisitos da aplicação levantados, foi possível realizar o desenho do *Wireframes*. Destes, são apresentados os que melhor respondem aos requisitos.

CCU1. Como Administrador, quero fazer *login* na página de administrador para aceder as funcionalidades disponíveis.

Detalhes: o *login*, esboçada na Figura 4.27, é realizado através de uma conta de e-mail autorizada.

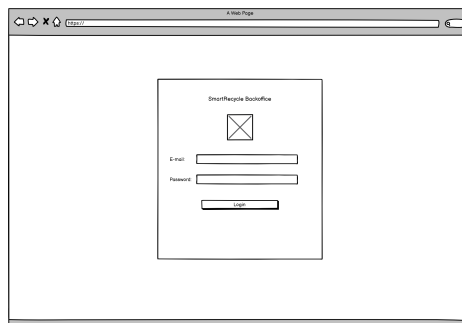


Figura 4.27: *Wireframe* de *login* na página de Administrador.

CCU2. Como Administrador, quero realizar a análise de resíduos mal identificados.

Detalhes: a análise, esboçada na Figura 4.28, é apresentada no ecrã principal do *Backoffice*, exibindo todos os resíduos que foram mal identificados pelo reconhecimento de imagem ou que são diferentes do resíduo analisados inicialmente pelo utilizador e como informação adicional. O Administrador pode ampliar a imagem e escolher da lista os que devem ser aceites e os recusados, com correspondência à atribuição ou não dos devidos pontos ao Utilizador.

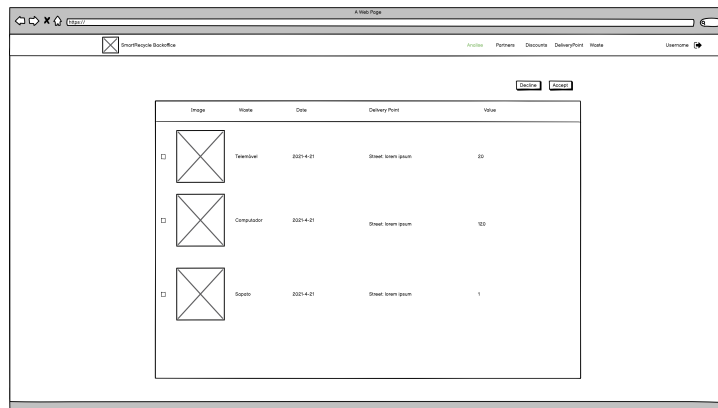


Figura 4.28: *Wireframe* da página de Administrador.

CCU3. Como Administrador, quero realizar a edição de parceiros e descontos associados a aplicação.

CCU4. Como Administrador, quero realizar a edição de pontos de recolha disponíveis.

CCU5. Como Administrador, quero realizar a edição da informação de resíduos aceites pela aplicação.

Detalhes: formulário de edição, esboçada na Figura 4.29, é utilizado para a inserção/edição de informação.

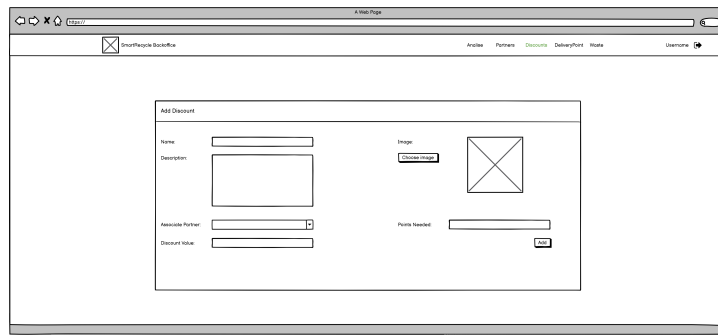


Figura 4.29: Wireframe da página de Administrador para edição de informação.

Tabela 4.21: Matriz de rastreabilidade.

Req.ID	CCU-1	CCU-2	CCU-3	CCU-4	CCU-5
1.F.BO	✓				
2.F.BO		✓			
3.F.BO			✓		
4.F.BO				✓	
5.F.BO					✓
6.F.BO		✓			

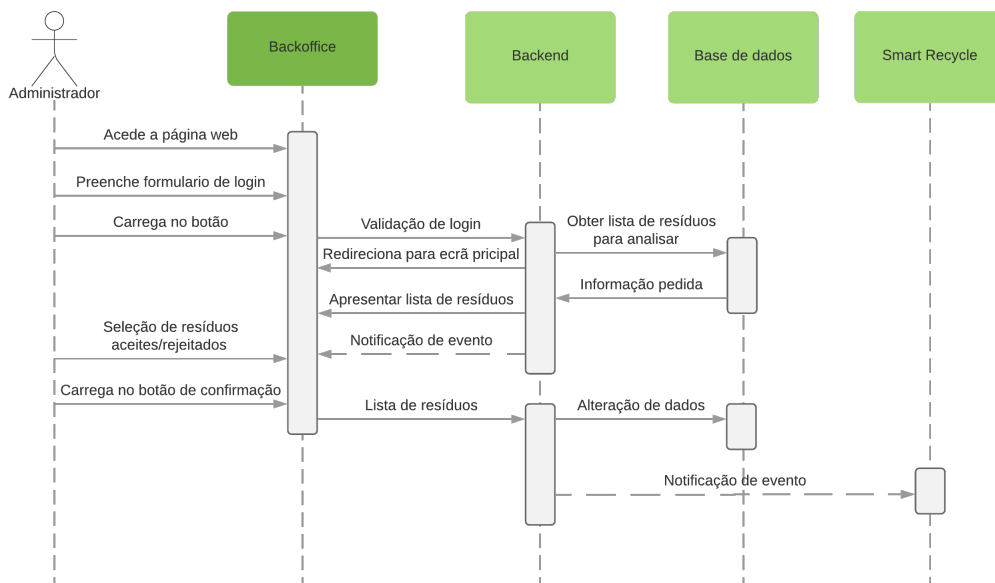


Figura 4.30: DS-1:Diagrama de sequência dos casos de uso 1 e 2.

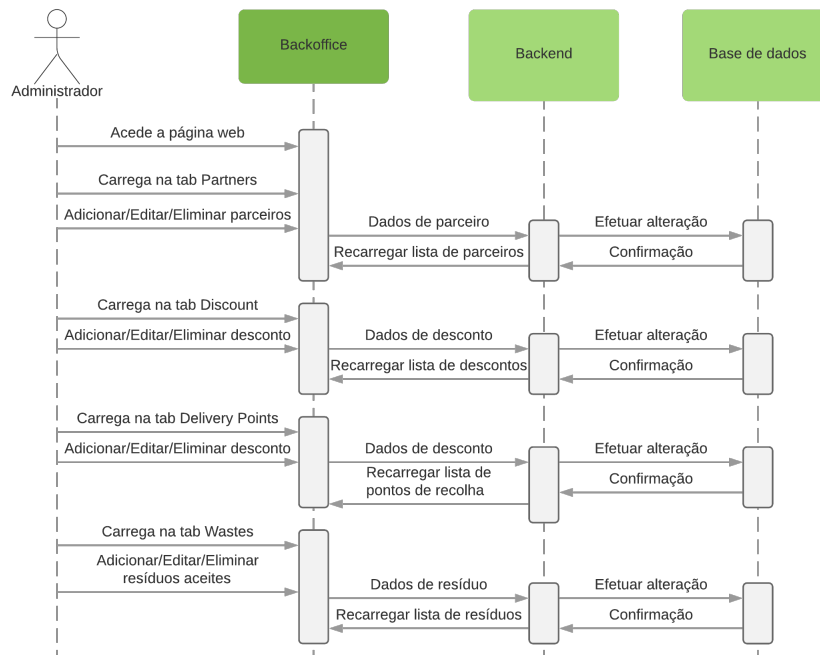


Figura 4.31: DS-2:Diagrama de seqüência dos casos de uso 3,4 e 5.

Tabela 4.22: Matriz de rastreabilidade entre casos de uso e diagrama de seqüência.

CU.ID	DS-1	DS-2
CCU-1	✓	
CCU-2	✓	
CCU-3		✓
CCU-4		✓
CCU-5		✓

4.2.4 Sistema *Backend*

Este sistema está encarregue da integração e interação de todos os outros, estando neste toda a lógica de funcionamentos do projeto. Foi gerada a documentação das API's do sistema *Backend* em Outsystems, presente no Anexo A.

Tabela 4.23: Tabela de requisitos de *Backend*.

ID	Tipo	Descrição
1.F.BE	F	Deverá classificar uma imagem de um resíduo electrónico.
2.F.BE	F	Deverá emitir um <i>QR Code</i> .
3.NF.BE	NF	Deverá implementar um modelo de dados de suporte ao sistema.
3.1.NF.BE	NF	Deverá fornecer APIs de comunicação com o modelo de dados.
4.NF.BE	NF	Deverá fornecer um <i>Hub</i> de eventos para notificações.
5.F.BE	F	Deverá atribuir pontos ao utilizador.
6.F.BE	F	Deverá notificar os diferentes <i>Frontends</i> de alterações no sistema.
7.F.BE	F	Deverá remover pontos ao utilizador.

Cenários

Neste tópico serão apresentados os diversos cenários do sistema, onde estão representadas as diferentes interações dos atores com o sistema.

Tabela 4.24: Cenário receber imagem para classificar.

ID e Nome	CCU-1: Análise de imagem
Sumário	A <i>backend</i> irá receber uma imagem para classificar vinda da aplicação Smart Recycle quando um utilizador captura uma imagem e irá receber uma imagem do <i>Delivery Point</i> aquando da entrega do resíduo.
Ator	SmartRecycle, <i>Delivery Point</i>
Cenário	<ol style="list-style-type: none"> 1. Receção de um binário para análise de imagem; 2. Envio do binário para análise no Vision; 3. Gerar <i>QR Code</i> associado ao resíduo analisado; 4. Entrada de dados na tabela <i>Analised</i> da base de dados; 5. Emitir notificação para o utilizador.
Cenário Alternativo 1	<ol style="list-style-type: none"> 1. Receção de um binário para análise de imagem; 2. Entrada de dados na tabela <i>Delivered</i> e <i>DeliveryPoint</i> da base de dados; 3. Emitir notificação para o utilizador; 4. Atribuir pontos pelo resíduo entregue.
Cenário Alternativo 2	<ol style="list-style-type: none"> 1. Receção de um binário para análise de imagem; 2. Entrada de dados nas tabelas <i>Error</i> e <i>DeliveryPoint</i> da base de dados; 3. Emitir notificação para o utilizador; 4. Emitir notificação para o administrador.

Tabela 4.25: Cenário utilizador obteve um desconto.

ID e Nome	CCU-2: Obter desconto
Sumário	A <i>backend</i> irá realizar a obtenção de um desconto.
Ator	SmartRecycle
Cenário	<ol style="list-style-type: none"> 1. Recebe o pedido; 2. Gerar QR Code associado ao desconto obtido; 3. Entrada de dados na tabela UserDiscounts da base de dados; 4. Remoção de pontos do utilizador.

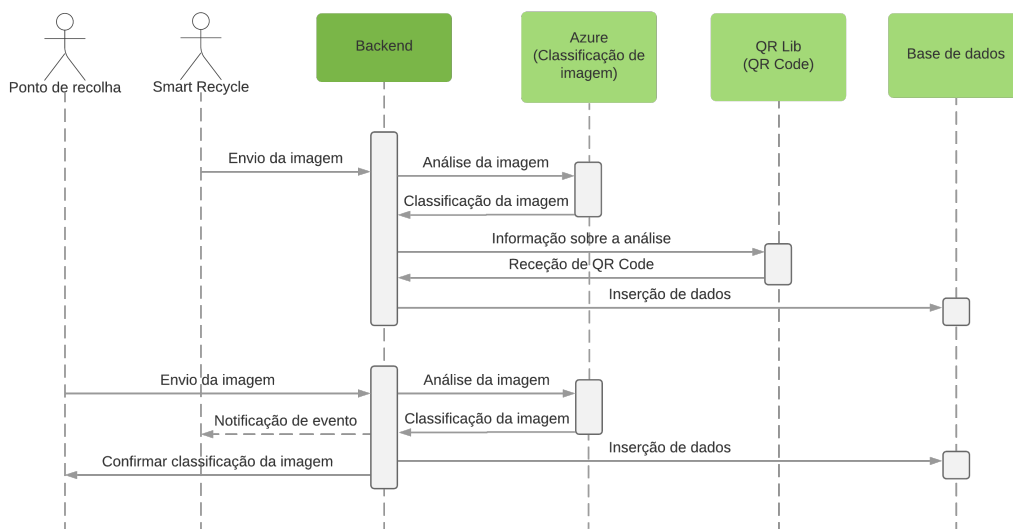


Figura 4.32: DS-1:Diagrama de sequência do caso de uso 1.

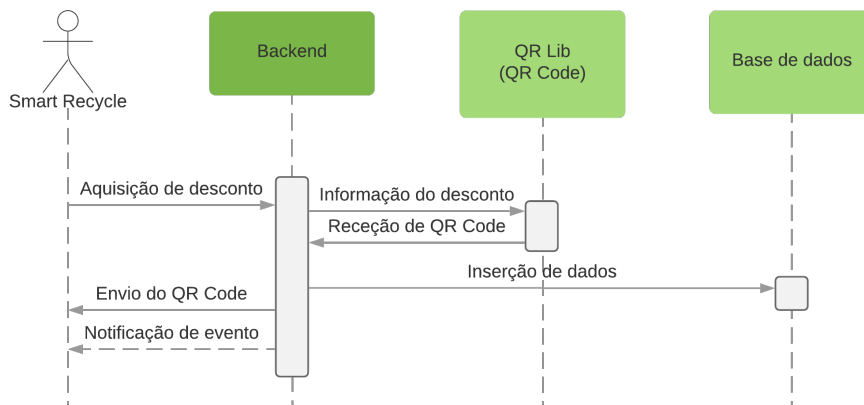


Figura 4.33: DS-2:Diagrama de sequência do caso de uso 2.

Tabela 4.26: Matriz de rastreabilidade.

Req.ID	CCU-1	CCU-2	DAO
1.F.BE	✓		
2.F.BE	✓	✓	
3.NF.BE			✓
3.1.NF.BE			✓
4.NF.BE			✓
5.F.BE	✓		
6.F.BE	✓		
7.F.BE		✓	

Foi criado um serviço em Outsystems que contém o módulo AzureCognitive-ServicesConnector. Este proporciona funcionalidades de inteligência artificial que são usadas para realizar a detecção de dispositivos numa imagem, apresentada na Figura 4.35. O serviço foi criado para poder estar separado de toda a aplicação, dando mais versatilidade para futuras alterações, uma vez que não está fortemente interligado com o resto da aplicação.



Figura 4.34: Função de análise de imagem.

Seguidamente foi criada a função que realiza a confirmação de que o resíduo

analisado se encontra presente na base de dados de resíduos aceites e se a confiança da análise pelos serviços da Azure é superior a 80%. Esta função é utilizada igualmente no *Delivery Point* para testar se o resíduo entregue coincide com o analisado na aplicação. No caso da confiança ser inferior a 80% e superior a 60%, o resíduo é aceite. Porém é marcado como possível erro para que o administrador confirmar manualmente.

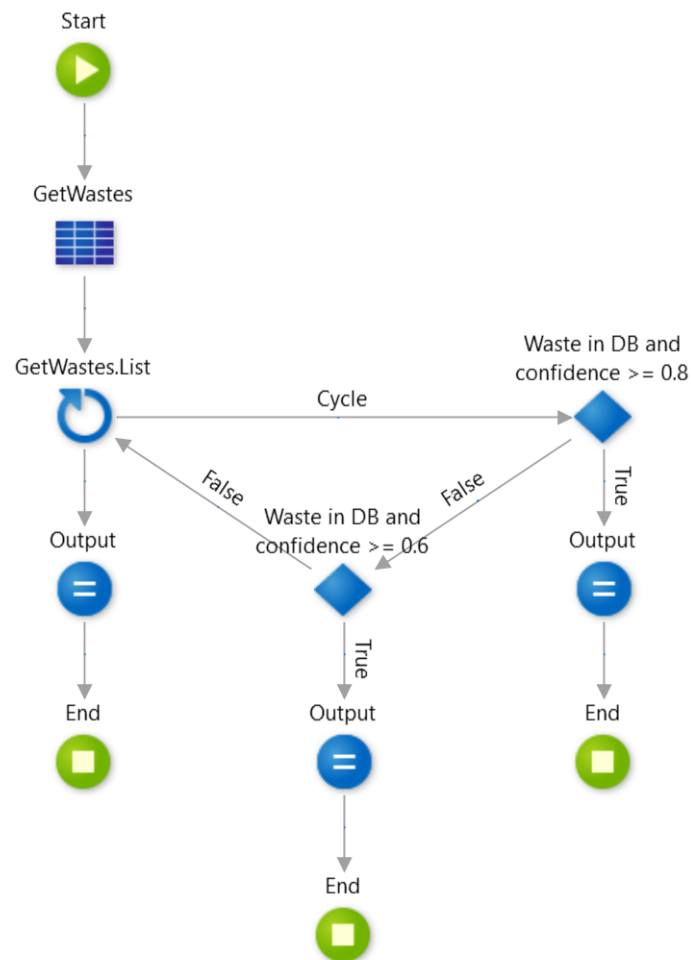


Figura 4.35: Função de comparação de resíduo aceites.

4.2.5 Sistema *Delivery Point*

Este sistema corresponde ao produto viável mínimo (MVP) desenvolvido que proporciona interação e realiza a recolha dos resíduos entregues pelo utilizador. Para a construção do *Delivery Point* (MVP - minimum viable product) foi elaborado um desenho tridimensional, com recurso a ferramenta FreeCAD, cujo o desenho técnico se apresenta em Anexo B.1. Posteriormente, foi construído o *Delivery Point* funcional em madeira. Foi também gerada a documentação da API do sistema *Delivery Point* em Outsystems, presente no Anexo B.2.

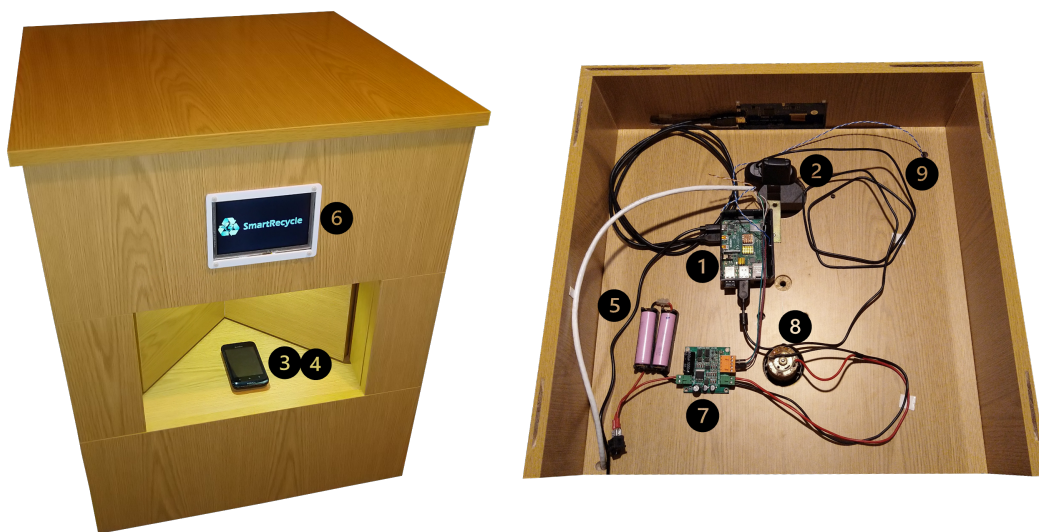
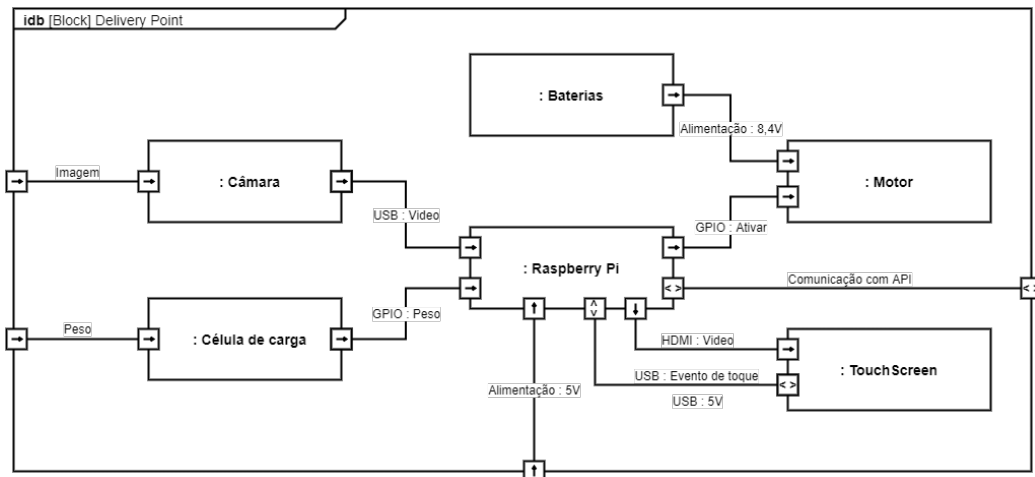


Figura 4.36: *Delivery Point* com identificação de componentes.

Tabela 4.27: Lista de componentes eletrónicos usados no *Delivery Point*.

Lista de componentes	
Processamento	1 - Raspberry Pi 4B
Câmara	2 - Logitech HD Webcam C270
Célula de carga	3 - Tal220
Módulo de conversão da célula	4 - HX711
Baterias	5 - 2x18650 3,7V 3500mAh (ICR18650) - Samsung
LCD	6 - 5" Resistive Touch Screen LCD (B), 800×480,Low Power Consumption
Driver de motor	7 - 15A DC Motor Driver SKU DRI0042
Motor	8 - Motor DC 12V
Sensores	9 - Interruptor magnético reed switch

A numeração indicada na tabela tem correspondência na Figura 4.36.

Figura 4.37: Diagrama de bloco do *Delivery Point* (idb).Tabela 4.28: Tabela de requisitos do *Delivery Point*.

ID	Tipo	Descrição
1.NF.PR	NF	Deverá estar equipado com controlador Raspberry Pi.
2.NF.PR	NF	Deverá estar capacitado para receber um resíduo.
3.NF.PR	NF	Deverá estar equipado com uma célula de carga.
4.NF.PR	NF	Deverá estar equipado com uma câmara.
5.NF.PR	NF	Deverá estar equipado com iluminação auxiliar.
6.NF.PR	NF	Deverá estar equipado com um motor.
7.NF.PR	NF	Deverá estar equipado com um LCD <i>TouchScreen</i> .
8.NF.PR	NF	Deverá comunicar com a <i>Backend</i> .
8.1.NF.PR	NF	Deverá usar API's de acordo com as especificações da <i>Backend</i> para comunicar.
8.2.NF.PR	NF	Deverá ter um identificador único de comunicação.
9.F.PR	F	Deverá possibilitar a interação com o <i>Delivery Point</i> através do LCD.
10.F.PR	F	Deverá fazer a leitura de <i>QR Codes</i> .
11.F.PR	F	Deverá fazer a leitura de imagem capturada do resíduo.
12.NF.PR	NF	Deverá ser uma caixa com 50x50x60cm.
13.NF.PR	NF	Deverá ter uma plataforma triangular de 46,6x22,9cm.
14.NF.PR	NF	Deverá ter uma fonte de alimentação de 5V para o Raspberry Pi.
15.NF.PR	NF	Deverá ter bateria de 8,4V para alimentar o motor.

Cenários

Neste tópico serão apresentados os diversos cenários do sistema, onde estão representadas as diferentes interações dos atores com o sistema.

Tabela 4.29: Cenário interação através de LCD.

ID e Nome	CCU-1: Interação com o <i>Delivery Point</i>
Sumário	O utilizador irá interagir com o <i>Delivery Point</i> através do <i>touchscreen</i> do lcd.
Ator	Visitante, Utilizador registado
Cenário	1. O utilizador poderá visualizar vídeo informativo da aplicação e o seu funcionamento.
Cenário Alternativo	1. O utilizador registado carrega no botão para entrega de resíduo; 2. Apresenta o QR Code do resíduo a entregar; 3. Entrega o resíduo na plataforma; 4. Fecha o compartimento de entrega; 5. Carrega no botão “Deliver”; 6. Recebe informação sobre a entrega no LCD.

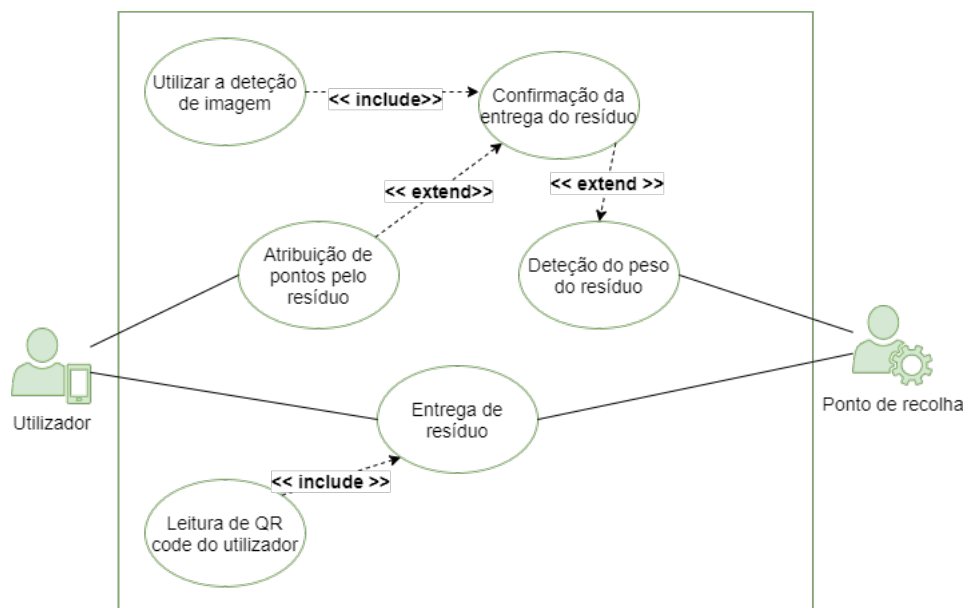
Figura 4.38: Diagrama de casos de uso do *Delivery Point*.

Tabela 4.30: Matriz de rastreabilidade.

Req.ID	CCU-1	DAO	idb
1.NF.PR			✓
2.NF.PR			✓
3.NF.PR			✓
4.NF.PR			✓
5.NF.PR			✓
6.NF.PR			✓
7.NF.PR			✓
8.NF.PR		✓	
8.1.NF.PR		✓	
8.2.NF.PR		✓	
9.F.PR	✓		
10.F.PR	✓		
11.F.PR	✓		
12.NF.PR			✓
13.NF.PR			✓
14.NF.PR			✓
15.NF.PR			✓

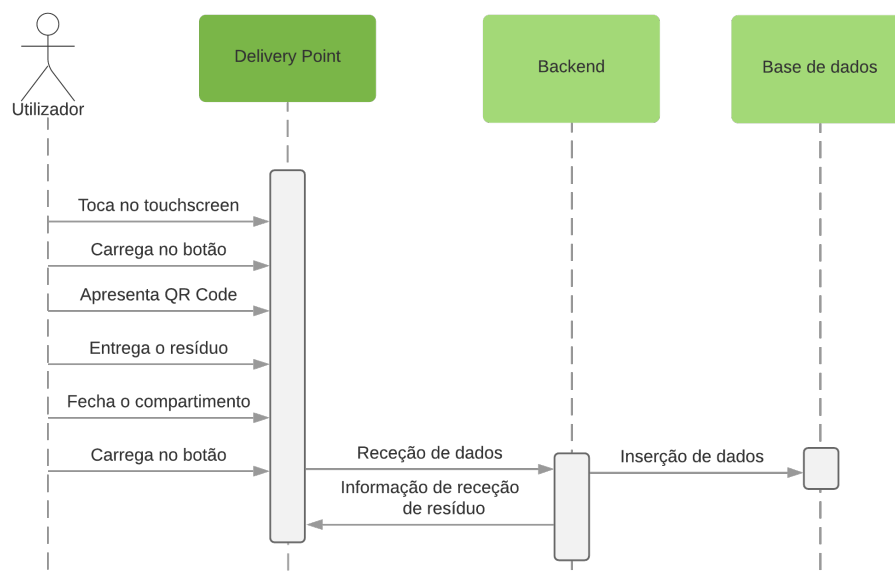


Figura 4.39: DS-1:Diagrama de sequência do caso de uso 1.

Tabela 4.31: Matriz de rastreabilidade entre casos de uso e diagrama de sequência.

CU.ID	DS-1
CCU-1	✓

Para realizar a leitura do *QR Code* no *Delivery Point*, foi escolhida a biblioteca OpenCV que proporciona a capacidade de desenvolver aplicações de visão computacional em tempo real. Focando-se essencialmente no processamento de imagem, captura e análise de vídeo, incluindo recursos extras como detecção de objeto.

Para decodificar o *QR Code* usou-se a biblioteca pyzbar, capaz de realizar a decodificação de *QR Codes* e *barcodes* e com um desempenho bastante bom e boa precisão. Por último, foi usada a biblioteca json para converter os dados oriundos do *QR Code* em formato json, tornando-se mais fácil de trabalhar com os mesmo no programa.

```
1 from pyzbar.pyzbar import decode
2 import datetime
3 import json
4 import cv2
5 # Configura o para a camera
6 cam = cv2.VideoCapture(0)
7 cam.set(3, 1280)
8 cam.set(4, 720)
9 while camera:
10     success, frame = cam.read()
11     if not success:
12         x = datetime.datetime.now()
13         day = x.strftime("%d_%m_%Y_%H_%M")
14         erro = "Erro abrir camera no dia {}\n".format(day)
15         f = open("log.txt", "a")
16         f.write(erro)
17         f.close()
18         # Desligar camera
19         cam.release()
20         camera = False
21     else:
22         for code in decode(frame):
23             dataRaw = code.data.decode("utf-8")
24             dataDump = json.dumps(dataRaw)
25             data = json.loads(dataDump)
26             if data != "":
27                 QrSuccess = True
28                 # Desligar camera
29                 cam.release()
30                 camera = False
```

Listagem 4.1: Código de leitura de QR Code.

Para obter a imagem do resíduo, utiliza-se o método `imwrite` da biblioteca OpenCV, bastando fornecer o nome a atribuir à imagem e o frame do vídeo. O nome é criado desta forma sempre que é apresentado um novo resíduo, ficando identificado pelo dia e hora, criando um nome diferente para cada imagem.

Para efeito de envio de dados para a API criada em Outsystems, estes são todos agrupados em formato json, o que facilitando a obtenção dos mesmos na API. Finalmente, é preciso converter a imagem para base64, transformando dados binários em texto, e garantindo assim que os dados se mantêm inalterados. É somente necessário em Outsystems voltar a converter para dados binários. Posteriormente é realizada

a sua classificação da imagem obtida, através de inteligência artificial (AzureCognitiveService), com o intuito de confirmar a recessão de um resíduo anteriormente analisado.

```
1 x = datetime.datetime.now()
2 day = x.strftime("%d_%m_%Y_%H_%M")
3 #Captura da imagem
4 cv2.imwrite(img_name, frame)
5 time.sleep(1)
6 #Desligar a camera
7 cam.release()
8 #Encode a imagem em base64 para envio
9 with open(img_name, "rb") as image:
10     encode_string = base64.b64encode(image.read())
11 #Formatar dados a enviar para a API
12 data = {
13     "RPI_ID": RPI_ID,
14     "User": data["Analised"]["UserId"],
15     "Waste": data["Analised"]["WasteId"],
16     "AnalisedID": data["AnalisedID"],
17     "Weight": int(weight),
18     "Image": encode_string.decode("utf-8"),
19 }
20 response = send(data)
```

Listagem 4.2: Formatação de dados para API.

Esta função que realiza o envio dos dados usando a biblioteca requests, esta abstrai as complexidades de fazer solicitações HTTP em Python por trás de uma API simples de usar. Indica-se, unicamente o URL da API com que se quer comunicar, os dados e por fim os cabeçalhos de HTTP, aguardando a resposta da API.

```
1 #Funcao de envio de dados para a API criada em Outsystems
2 def send(data):
3     response = ""
4     while response == "":
5         response = requests.post(URL_API, data=json.dumps(
6             data), headers=headers)
7     return response
8
9 #API URL
10 URL_API = "https://edulab01.outsystemsenterprise.com/
11     RPI_API/rest/Image/POST"
12
13 #ID do Delivery Point e chave de autentica o
14 RPI_ID = "1"
15 API_KEY = "traga-bolas"
16
17 #Definicao dos header de envio
18 content_type = "application/json"
19 headers = {"content-type": content_type, "API_KEY": API_KEY
20     }
```

Listagem 4.3: Função de envio dos dados para API.

De forma a confirmar que o resíduo foi entregue no *Delivery Point*, usou-se uma célula de carga, com a finalidade de medir o peso do objeto entregue e também despoletar a captura da imagem do resíduo. Desta maneira, facilita a contabilização do peso total que o *Delivery Point*, sendo que este emite um alerta para o administrador do sistema quando alcança o limite.

```
1 # Funcao para obtencao do peso
2 def weight():
3     i = 0
4     Count = 0
5     DT = 15 # Pino de data do HX711
6     SCK = 14 # Pino de Clock
7     time.sleep(0.001)
8     GPIO.setup(DT, GPIO.OUT)
9     GPIO.output(DT, 1)
10    GPIO.output(SCK, 0)
11    GPIO.setup(DT, GPIO.IN)
12    while GPIO.input(DT) == 1:
13        i = 0
14        for i in range(24):
15            GPIO.output(SCK, 1)
16            Count = Count << 1
17            GPIO.output(SCK, 0)
18            time.sleep(0.001)
19            if GPIO.input(DT) == 0:
20                Count = Count + 1
21                # print Count
22            GPIO.output(SCK, 1)
23            Count = Count ^ 0x800000
24            time.sleep(0.001)
25            GPIO.output(SCK, 0)
26    return Count
```

Listagem 4.4: Código de medição de peso.

Na construção da interface do LCD para interação do utilizador com o *Delivery Point*, foi usada a *framework* Kivy, desenvolvida em Python. É uma *framework* utilizada para o desenvolvimento de aplicações multiplataforma, seja para *desktop* (Windows, Linux e MacOS) ou *mobile* (Android e iOS).

Foi considerada e usada esta *framework* por ser compatível com o sistema operativo Linux, usado no Raspberry Pi e porque todo o código para o funcionamento do *Delivery Point* ter sido desenvolvido em Python.

Encontram-se presentes em Anexo C.1 o código para a criação das interfaces e no Anexo C.2 o código do funcionamento do *Delivery Point*.

Capítulo 5

Testes e Avaliação da solução

Este capítulo, onde é realizada a análise de resultados, constitui um momento de avaliação da investigação, tendencialmente direcionada para a constatação/confirmação das hipóteses definidas. No caso presente, dado o seu carácter prático, e por que estamos perante o desenvolvimento de uma aplicação móvel e um produto viável mínimo (MVP), os resultados são refletidos na confirmação do funcionamento efetivo do produto concebido.

5.1 Testes

Aplicação SmartRecyle

De forma a testar o funcionamento da aplicação móvel foram realizados testes funcionais no final de cada *sprint*, sendo possível verificar a ausência de qualquer requisito. Na testagem, no final do desenvolvimento do projeto, obteve-se um produto 100% funcional, sem qualquer falha de requisitos. Dado que o propósito central da aplicação é o reconhecimento de imagem e a sua classificação, apresentado na Figura 5.1 e na Figura 5.2, foram realizados múltiplos testes desta função durante todo o projeto, certificando-se que estava sempre a funcionar corretamente, mesmo durante alterações profundas de funcionalidades da aplicação.

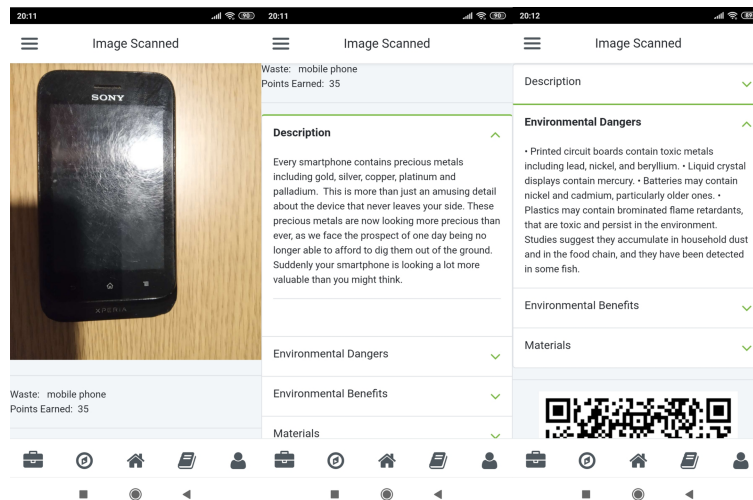


Figura 5.1: Resultado dos testes do reconhecimento de imagem com recurso a IA.

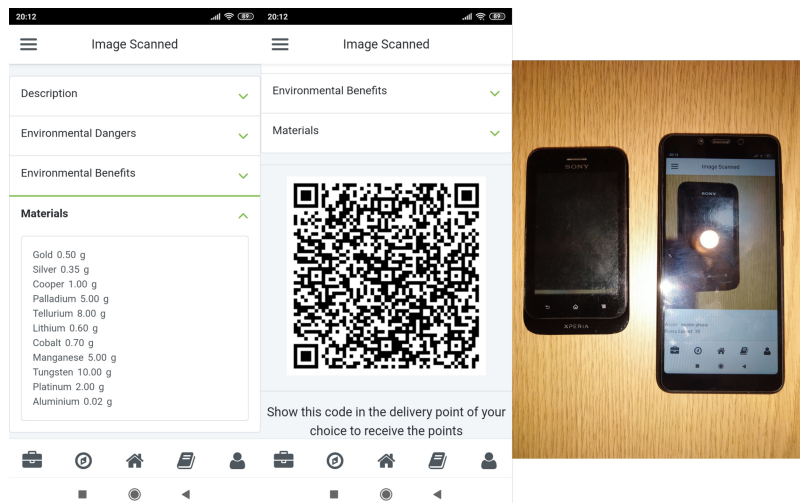


Figura 5.2: Resultado dos testes do reconhecimento de imagem com recurso a IA.

Por fim, confirmou-se que todos os requisitos tinham sido satisfeitos, resultando numa aplicação totalmente funcional.

Aplicação *Backoffice*

Os testes realizados a este sistema foram efetuados também no final de cada *sprint* e no final de todo o desenvolvimento, tendo sido atestado que o administrador tinha a habilidade de satisfazer todos os requisitos propostos.

Aplicação *Backend*

A *Backend*, é um sistema central para o funcionamento da aplicação móvel e do *Backoffice* e sendo fulcral para a comunicação de todos os sistemas, mereceu as atenções e foi alvo da realização de mais testes funcionais durante o desenvolvimento. Efetuou-se principalmente testes às operações de CRUD e à devida inserção de dados na base de dados.

Por fim, foram realizados alguns testes de desempenho, centrados na transmissão de dados entre os diferentes sistemas, com principal foco no envio de ficheiros de imagem vindos do sistema *Delivery Point*, por ser necessário o armazenamento e a conversão dos dados em formato Base64.

Sistema *Delivery Point*

O *Delivery Point* (MVP - minimum viable product), apresentado na Figura 5.3, foi desenhado e concebido em paralelo com a aplicação móvel com o intuito da experimentação de todo o processo de entrega de resíduos, juntando, deste modo, a vertente de eletrónica com a vertente de informática (programação). No *Delivery Point* é possível visualizar um vídeo informativo sobre economia circular, quando o MVP está em modo de espera (não está em uso), e visualizar a explicação do próprio processo de entrega. Ou seja, disponibiliza os procedimentos, passo a passo a realizar aquando da entrega de um resíduo.



Figura 5.3: *Delivery Point* (MVP).

5.2 Avaliação da solução

No final do desenvolvimento do projeto foi concebido, através do **Google Forms**, um questionário de usabilidade da aplicação SmartRecycle e do Delivery Point, exibidos na Figura 5.4 e na Figura 5.10 respetivamente, que foi posteriormente disponibilizado a algumas pessoas, entre elas alguns colaboradores da empresa. Pretendeu-se assim, obter uma avaliação da usabilidade, viabilidade e pertinência do produto final desenvolvido.

Na formulação do inquérito adotou-se a escala de Likert, uma vez que se tratar de uma escala psicométrica usada em questionários/inquéritos de pesquisa de opinião. A resposta a este tipo de inquérito revela o nível de concordância do inquirido em relação a uma afirmação, ou seja, o que se deseja medir é o nível de concordância ou não concordância.

Normalmente são usados 5 níveis de resposta, que no caso presente, se traduzem em:

- 1 – Discordo completamente;
- 2 – Discordo;
- 3 – Não concordo nem discordo;
- 4 – Concordo;
- 5 – Concordo completamente.

The image shows a web-based questionnaire titled "Análise de Usabilidade SmartRecycle". It contains five Likert scale questions, each with a 5-point rating from 1 (Discordo Completamente) to 5 (Concordo Completamente). The questions are:

- Considero a utilização da aplicação fácil.
- Considero a informação apresentada clara.
- Considero as várias funções do sistema bem integradas.
- Acho que utilizaria a aplicação num cenário real.
- Recomendo esta aplicação.

At the bottom of the form, there are two buttons: "Submeter" (Submit) and "Limpar formulário" (Clear form).

Figura 5.4: Questionário de usabilidade do SmartRecycle.

Os resultados obtidos no questionário, correspondente a 30 inquiridos, encontram-se indicados de seguida.

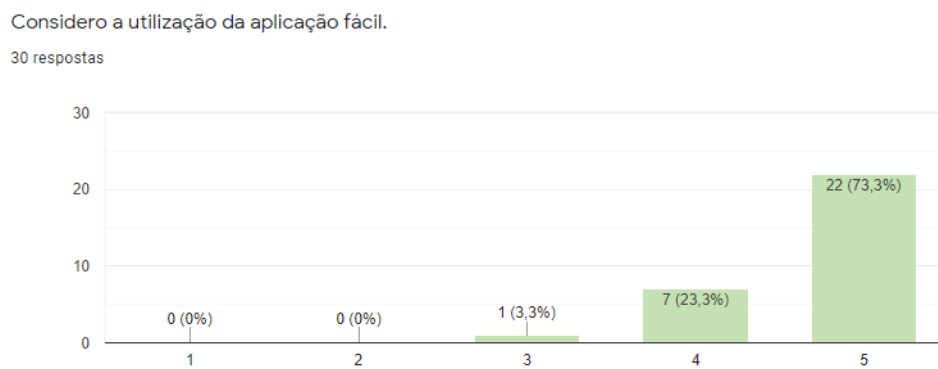


Figura 5.5: Gráfico 1 – Facilidade de utilização.

Em relação a facilidade de utilização da aplicação (Figura 5.5), a maioria dos inquiridos considerou-a muito fácil, sendo que sete pessoas a considerou fácil e houve uma pessoa não a qualificou nem como fácil nem difícil.

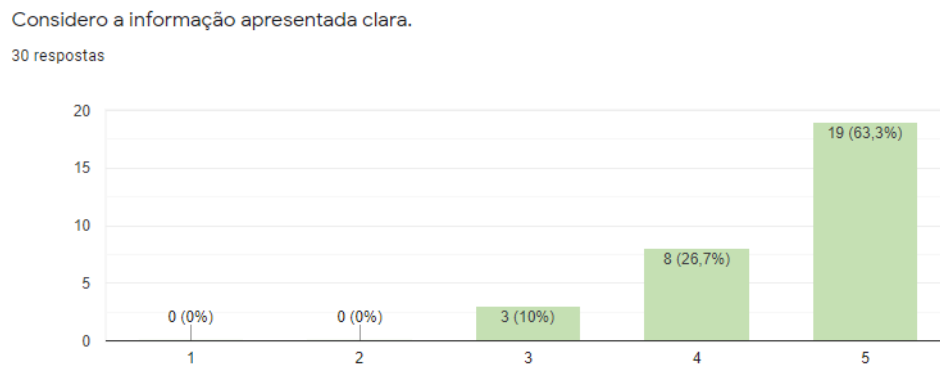


Figura 5.6: Gráfico 2 – Clareza da informação.

Quanto a clareza da informação apresentada na aplicação (Figura 5.6), a grande parte dos inquiridos considerou-a muito clara, oito pessoas consideraram-na clara e houve ainda três pessoas que não a qualificaram nem como clara nem confusa.

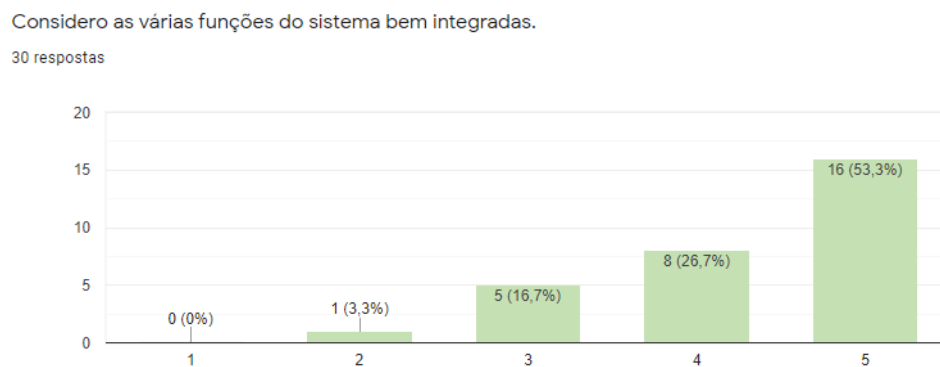


Figura 5.7: Gráfico 3 – Implementação de funcionalidades.

No que respeita à integração das diversas funcionalidades da aplicação (Figura 5.7), o maior número dos inquiridos considerou-a muito bem integradas. Consideraram-na bem integradas oito pessoas e houve cinco pessoas que não a qualificou nem como bem integradas nem mal integradas. Constatou-se que uma pessoa respondeu no sentido de estarem mal integradas.

Acho que utilizaria a aplicação num cenário real.

30 respostas

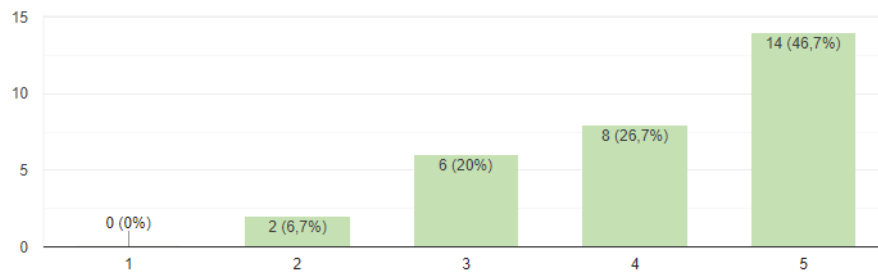


Figura 5.8: Gráfico 4 – Utilização num cenário real.

Relativamente ao uso da aplicação em situação concreta e real (Figura 5.8), a grande maioria respondeu que utilizaria ou utilizaria muito o produto (20), sendo que seis referem não ter um comportamento definido do uso do produto e ainda dois que não estão recetivos ao uso do produto.

Recomendo esta aplicação.

30 respostas

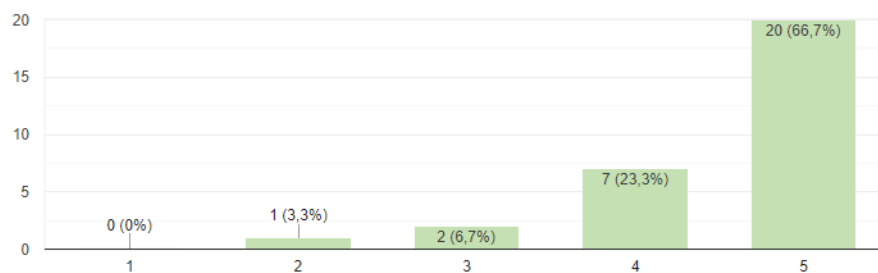


Figura 5.9: Gráfico 5 – Percentagem de pessoas que recomendariam a aplicação.

No que concerne a recomendação da própria aplicação (Figura 5.9), a grande maioria recomendaria ou recomendaria significativamente a aplicação (27). Obteve-se duas respostas sem opinião definida e ainda uma que não recomendaria a aplicação.

Análise de Usabilidade Delivery Point

Inicie sessão no Google para guardar o seu progresso. Saiba mais

A sua utilização é fácil.

1 2 3 4 5

Discordo Completamente Concordo Completamente

O seu funcionamento é claro.

1 2 3 4 5

Discordo Completamente Concordo Completamente

A informação apresentada é clara.

1 2 3 4 5

Discordo Completamente Concordo Completamente

O seu uso é útil nos tempos atuais.

1 2 3 4 5

Discordo Completamente Concordo Completamente

Pondero usar este Delivery Point num cenário real.

1 2 3 4 5

Discordo Completamente Concordo Completamente

Submeter
Limpar formulário

Figura 5.10: Questionário de usabilidade do *Delivery Point*.

De seguida são apresentados os resultados obtidos no questionário relativos ao *Delivery Point*, correspondente a 14 inquiridos. Este número mais diminuto de inquirido prende com a localização do MVP (domicílio) e que, face a situação pandémica, assissibilidade ao mesmo foi restrita.

A sua utilização é fácil.

14 respostas

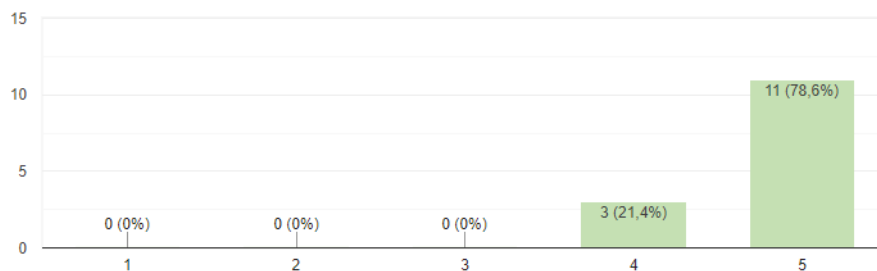


Figura 5.11: Gráfico 1 – Facilidade de utilização.

Em relação a facilidade de utilização do *Delivery Point* (Figura 5.11), a maioria dos inquiridos considerou-a muito fácil, sendo que três pessoas a consideraram fácil.

O seu funcionamento é claro.

14 respostas

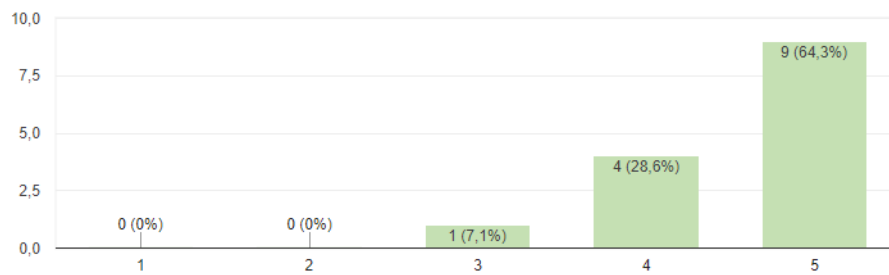


Figura 5.12: Gráfico 2 – Clareza do funcionamento.

No que respeita à implementação das diversas funcionalidades do *Delivery Point* (Figura 5.12), o maior número dos inquiridos considerou-a muito bem implementadas. Obtendo-se uma resposta que qualificou nem como bem implementadas nem mal implementadas.

A informação apresentada é clara.

14 respostas

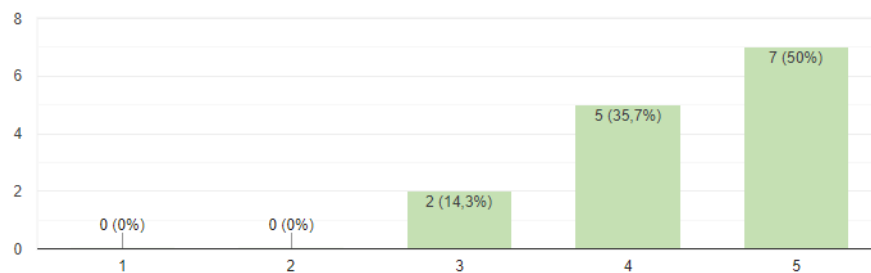


Figura 5.13: Gráfico 3 – Clareza de informação.

Quanto a clareza da informação apresentada no LCD do *Delivery Point* (Figura 5.13), sete dos inquiridos considerou-a muito clara, cinco pessoas consideraram-na clara e houve ainda duas pessoas que não a qualificaram nem como clara nem confusa.

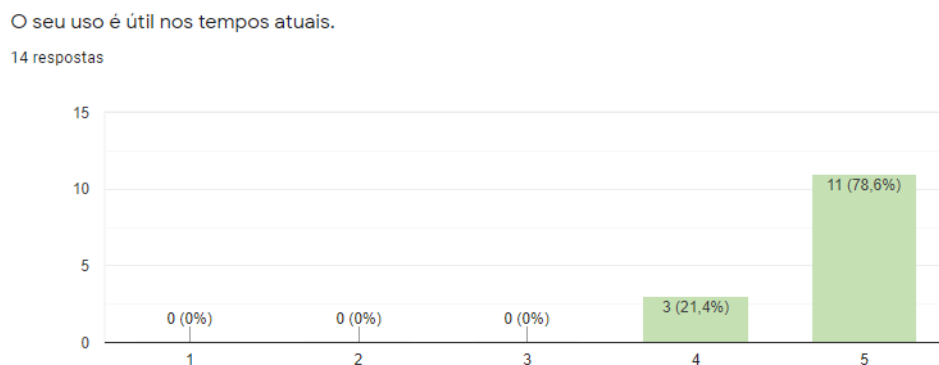


Figura 5.14: Gráfico 4 – Útil nos tempos atuais.

No que respeita a importância do uso do *Delivery Point* nos tempos atuais (Figura 5.14), como forma de incentivo a reciclagem, obtiveram-se três respostas que a consideraram útil, sendo que as restantes se posicionaram no muito útil.

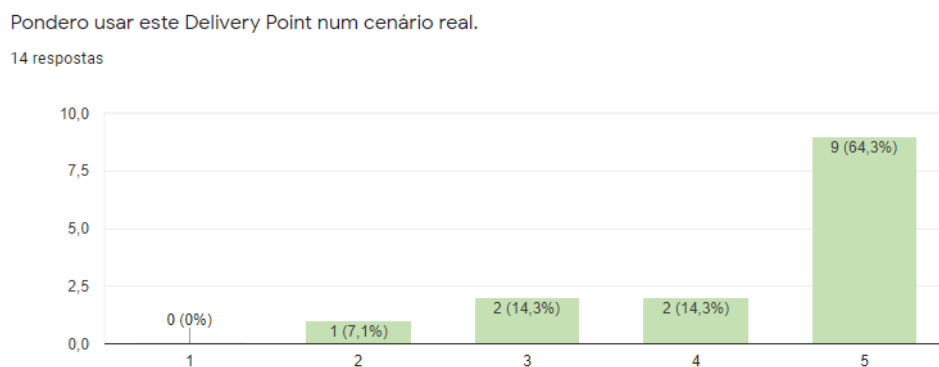


Figura 5.15: Gráfico 5 – Percentagem de pessoas que usariam num cenário real.

Relativamente ao uso do *Delivery Point* em num cenário real (Figura 5.15), a grande maioria respondeu que utilizaria ou utilizaria muito o produto (11). Dois referem não ter um comportamento definido relativo ao uso do produto e um não está receptivo ao mesmo.

Através da análise dos resultados relativos ao questionário de usabilidade do SmartRecycle constatou-se que a grande maioria dos inquiridos direccionou as suas respostas para os dois níveis de concordância, com maior destaque no nível de concordância completa. Tal facto, permite concluir que a generalidade dos inquiridos considera a aplicação de fácil utilização, detentora de uma informação clara e acessível, cujas funcionalidades se encontram bem definidas e implementadas. Salienta-se, igualmente, o facto de terem considerado a aplicação passível de ser utilizada em

situação concreta e real, motivo pelo qual a recomendam a outros utilizadores.

No que se refere ao questionário de usabilidade do *Delivery Point*, os dados obtidos centraram-se, uma vez mais, nos dois níveis de concordância, salientando-se o último nível, concordância completa por apresentar o valor mais elevados ao longo de todo o questionário. Depreende-se, assim, que em relação ao *Delivery Point* (MVP - minimum viable product) a maioria dos inquiridos considerou-o de fácil utilização, que é potenciada e complementada pela informação disponibilizada no LCD, que de forma e clara e assertiva orientações de utilização e informação sobre a reciclagem. Finalmente, menciona-se o facto da maioria dos inquiridos considera o produto útil e importante como forma e incentivo a reciclagem, daí que manifestem o seu interesse no seu uso em contexto real.

Capítulo 6

Conclusões

A conclusão é o momento final do processo de investigação e representa a súmula das várias fases de investigação: objetivos, passando pela fundamentação teórica, a conceção, elaboração e realização do projeto/produto, finalizando com a confirmação da sua eficácia. É o resumir de todo um processo de múltiplas etapas e sucessivas reflexões e reajustamentos, com o intuito de contribuir para o avanço tecnológico e científico. A relevância da conclusão define-se pela retrospectiva crítica e pela construção científica e inovação tecnológica.

Na base deste trabalho de investigação está a crescente preocupação com a sustentabilidade do planeta, facto que tem vindo a assumir uma relevância central nas políticas económicas e ambientais. Tornou-se, na viragem deste século, um tema recorrente e transversal a todas as sociedades, que clamam por mudanças de paradigma, de modo a assegurar os recursos naturais e a biodiversidade.

A racionalidade económica e ambiental é uma premissa que não pode estar ausente das preocupações dos decisores políticos e sociais. Em resposta ao desenvolvimento descontrolado e ilimitado, emerge agora uma abordagem que preserva um desenvolvimento equilibrado e sustentável. Esta baseia-se e acentua duas noções extremamente preponderantes no presente, mas principalmente para a vida das futuras gerações: por um lado, a satisfação das necessidades básicas a toda a população mundial; por outro lado, o proporcionar do desenvolvimento respeitando os limites dos recursos naturais e prevenindo os danos ambientais, ou seja, promover o

desenvolvimento sustentável.

A Economia Circular é o modelo de e para o futuro, agregador do desenvolvimento económico e das preocupações ambientais. Trata-se de uma nova forma de ver as práticas económicas, inspirada no funcionamento da própria natureza. O desafio atual é integrar a inovação de produtos e serviços, sistemas produtivos e modelo de negócio, num novo paradigma de sustentabilidade, com o intuito de gerar crescimento económico, mas também bem-estar social e preservação ambiental.

A procura excessiva e desenfreada de matérias-primas, que são cada vez mais escassas, por parte das economias mais desenvolvidas, que, por sua vez, controlam o seu mercado, tem provocado o aumento exponencial do preço desses recursos naturais e, conseqüentemente comprometido a rentabilidade económica das empresas.

Inovação e *design* de produtos e sistemas é a chave de toda esta transformação social. Pressupõe o princípio do “*fechar o ciclo de vida*” dos produtos, possibilitando, assim, a redução do consumo das matérias-primas, de energia e de água. Esta abordagem de desenvolvimento económico, “*Promove o desenvolvimento de novas relações entre as empresas, que passam a ser simultaneamente consumidoras e fornecedoras de materiais que são reincorporados no ciclo produtivo*”, de acordo com Ellen MacArthur Foundation [22].

A circularidade deste novo modelo produtivo e económico assenta não só na redução da extração de metais e minerais, mas compreende, igualmente, a recuperação destes materiais e a sua reintrodução no sistema produtivo. Desta forma, reduz o impacto da atividade económica no bem-estar social e ambiental. Através do processo de reciclagem, os componentes ou materiais são recuperados, permitindo assim, constituírem novamente matérias-primas. Assim sendo, a Reciclagem adquire um destaque assinalável, alargando o seu âmbito, quer em termos de matéria-primas, quer em termos de processos. Indubitavelmente, tudo o que nos permita reciclar matérias-primas, constitui um ganho considerável na corrida para a sustentabilidade.

Tendo presente estas novas teorias, abordagens e processos considerou-se pertinente e marcante conceber um produto que promovesse a reciclagem de resíduos eletrónicos (ex: *Smartphones, Tables,...*), uma vez que é escassa, no que se refere a este tipo de resíduos. Pensou-se num produto que possibilitasse e incentivasse a recolha dos aparelhos eletrónicos, com vista à reciclagem dos seus componentes.

O projeto estruturou-se de forma a materializar os objetivos definidos, tendo por base um problema prático e atual. Nesse sentido concebeu-se e contruiu-se um MVP

que consiste num ponto de recolha de resíduos eletrónicos de possível colocação em locais estratégicos (*shopping*, lojas de material eletrónico, organizações oficiais, e outros).

O MVP é um ponto de recolha autónomo e inteligente ao qual se encontra associado uma aplicação móvel criada, passível de orientar a ação antes, durante e após a própria recolha do resíduo. No processo de testagem confirmou-se a funcionalidade e eficácia do produto desenvolvido no reconhecimento de dispositivos eletrónicos através da imagem, de forma a determinar o seu “*valor*” reciclável, ou seja, o seu valor como resíduo.

O valor reciclável do resíduo eletrónico entregue/recolhido é convertido em pontos. Estes, por sua vez podem ser trocados por vales de descontos a serem usados no comércio local, aderente à iniciativa, e assim ficar associado a uma imagem “*verde*” e “*sustentável*” e beneficiar de publicidade numa plataforma colaborativa. Tais vantagens servem como mecanismo de fomento de entrega de resíduos e de incentivo à reciclagem deste tipo de material, assim como ao desenvolvimento do próprio comércio local.

A aplicação e o MVP foram alvo de avaliação quanto a sua usabilidade, através de questionários com escala de cinco níveis e contendo cada um deles cinco itens/-questões. No caso da aplicação SmartRecycle os itens do questionário remetem para a facilidade de utilização, a clareza da informação disponibilizada, a implementação das funcionalidades, a sua implementação em contexto real e por fim, e a sua recomendação por parte de utilizadores. Em relação ao MVP os itens do questionário focaram-se na facilidade de utilização deste *Delivery Point*, na clareza dos procedimentos de utilização, na clareza e relevância da informação disponibilizada, e também a utilidade e pertinência do seu uso na atualidade e em contexto real.

Os resultados obtidos nos questionários situaram-se, na sua generalidade, e ao longo de todo o questionário, nos dois níveis superiores (concordo e concordo completamente), sempre com uma percentagem igual ou superior a 70%. Mais ainda, realça-se o facto do nível cinco (concordo completamente) apresentar valores de grande destaque. Tendo por base a análise dos resultados infere-se que os inquiridos consideraram a aplicação SmartRecycle e o *Delivery Point* de fácil uso, de grande viabilidade em contexto real e bastante pertinentes nos tempos atuais.

Em suma, constatou-se que o projeto elaborado cumpriu os objetivos definidos respondendo, deste maneira, aos problemas apontados, pelo que se considera um produto funcional.

É entendido que o SmartRecycle apresenta grande potencial de implementação e de distribuição global, com a finalidade de resolver ou minimizar os problemas identificados. Pelas suas características e eficácia revelada, o projeto constitui uma considerável possibilidade de negócio.

Referências

- [1] “Objetivos de desenvolvimento sustentável.” Available at <https://unric.org/pt/objetivos-de-desenvolvimento-sustentavel/>, Feb. 2021. (Last accessed in 27/04/2021). [Citado nas páginas vii, 2 e 13]
- [2] FpM, “Economia circular.” Available at <https://www.financaspormulheres.com/post/economia-circular>, July 2021. (Last accessed in 30/07/2021). [Citado nas páginas vii e 14]
- [3] “Economia circular - o que é a economia circular?.” Available at <https://eco.nomia.pt/pt/economia-circular/estrategias>. (Last accessed in 18/05/2021). [Citado nas páginas vii e 16]
- [4] “Plano de ação para a economia circular em portugal: 2017-2020.” Available at <https://eco.nomia.pt/pt/economia-circular/principios>. (Last accessed in 27/04/2021). [Citado nas páginas vii, 14, 17, 19 e 20]
- [5] p. F. Andrade, “Economia circular e sustentável - aterra.” Available at <https://aterraambiental.com/economia-circular/>, Jan. 2021. (Last accessed in 10/04/2021). [Citado nas páginas vii e 19]
- [6] V. Forti, C. P. Balde, R. Kuehr, and G. Bel, “The global e-waste monitor 2020: Quantities, flows and the circular economy potential,” July 2020. (Last accessed in 20/06/2021). [Citado nas páginas vii, 2, 21 e 22]
- [7] “E-waste - android apps on google play.” Available at <https://play.google.com/store/search?q=e-waste&c=apps>. (Last accessed in 10/08/2021). [Citado nas páginas vii e 24]
- [8] A. Hevner and S. Chatterjee, “Design science research in information systems,” in *Design research in information systems*, pp. 9–22, Springer, 2010. [Citado nas páginas vii, 27 e 28]
- [9] L. Matsumota, “Três diferenças da gestão tradicional e ágil.” Available at <https://leonardo-matsumota.com/2019/07/29/tres-diferencas-da-gestao-tradicional-e-agil/>, July 2019. (Last accessed in 7/05/2021). [Citado nas páginas vii, 29 e 30]

- [10] “Scrum: A metodologia ágil explicada de forma definitiva.” Available at <https://mindmaster.com.br/scrum/>, Mar. 2021. (Last accessed in 18/05/2021). [Citado nas páginas vii e 31]
- [11] “Scrum - framework de desenvolvimento de produtos, abordagem ágil.” Available at <https://www.scrumportugal.pt/scrum/>, July 2021. (Last accessed in 12/08/2021). [Citado nas páginas vii e 32]
- [12] “Developing with outsystems: Evaluation guide.” Available at <https://www.outsystems.com/evaluation-guide/developing-with-outsystems/>. (Last accessed in 19/07/2021). [Citado nas páginas vii e 34]
- [13] “Outsystems enterprise apaas: Core products.” Available at <https://www.outsystems.com/enterprise-apaas/>. (Last accessed in 19/07/2021). [Citado nas páginas vii e 34]
- [14] “The architecture canvas.” Available at https://success.outsystems.com/Support/Enterprise_Customers/Maintenance_and_Operations/Designing_the_Architecture_of_Your_OutSystems_Applications/The_Architecture_Canvas, Jul 2021. (Last accessed in 19/07/2021). [Citado nas páginas vii, 36 e 37]
- [15] “How minerals and metals companies can help achieve 2030 agenda for sustainable development.” Available at <https://www.unep.org/news-and-stories/story/how-minerals-and-metals-companies-can-help-achieve-2030-agenda-sustainable>. (Last accessed in 21/08/2021). [Citado nas páginas 2 e 3]
- [16] “Technology you can lean on.” Available at <https://www.doitlean.com/>. (Last accessed in 2/03/2021). [Citado na página 4]
- [17] J. Simão, “Sustentabilidade e desenvolvimento: conceitos e políticas emergentes: roteiro de estudo 2017/18,” *Electron. Lett.*, 2017. [Citado nas páginas 9, 11 e 12]
- [18] A. Maddison, “The world economy: Historical statistics oecd,” 2003. [Citado na página 10]
- [19] S. Dresner, *The principles of sustainability*. Routledge, 2012. [Citado na página 10]
- [20] V. Soromenho-Marques, “As idades da política internacional de ambiente,” in *Fórum Ambiente*, no. 85, pp. 5–6, 2003. [Citado na página 10]
- [21] U. WCED, “Our common future—the brundtland report,” *Report of the World Commission on Environment and Development*, 1987. [Citado na página 11]

- [22] E. MacArthur *et al.*, “Towards the circular economy,” *Journal of Industrial Ecology*, vol. 2, no. 15, pp. 23–44, 2013. [Citado nas páginas 13, 14 e 88]
- [23] D. W. Pearce, R. K. Turner, and R. K. Turner, *Economics of natural resources and the environment*. Johns Hopkins University Press, 1990. [Citado na página 14]
- [24] A. Leitão, “Economia circular: uma nova filosofia de gestão para o séc. xxi,” *Portuguese Journal of Finance, Management and Accounting*, vol. 1, no. 2, pp. 150–171, 2015. [Citado nas páginas 15, 16 e 17]
- [25] E. M. Foundation, M. C. for Business, and Environment, *Growth within: a circular economy vision for a competitive Europe*. Ellen MacArthur Foundation, 2015. [Citado na página 15]
- [26] U. G. Schulte, “New business models for a radical change in resource efficiency,” *Environmental Innovation and Societal Transitions*, vol. 9, pp. 43–47, 2013. [Citado na página 16]
- [27] W. Stahel, “The product life factor. an inquiry into the nature of sustainable societies: The role of the private sector,” *Houston Area Research Center*, pp. 72–105, 1982. [Citado nas páginas 16 e 19]
- [28] L. M. C. R. C. d. Cruz, *DEstudo exploratório da relação entre Economia Circular e Economia da Partilha: motivações dos clientes*. PhD thesis, Católica Porto Business School, Mar. 2017. [Citado na página 16]
- [29] C. Baldé, V. Forti, V. Gray, R. Kuehr, and P. Stegmann, “New business models for a radical change in resource efficiency,” *United Nation University (UNU), International Telecommunication Union (ITU) and International Solid Waste Association (ISWA), Bonn/Geneva/Vienna*, pp. 1–109, 2017. [Citado na página 22]
- [30] M. Pimentel, D. Filippo, and T. M. Santos, “Design science research: pesquisa científica atrelada ao design de artefatos,” *RE@ D-Revista de Educação a Distância e eLearning*, vol. 3, no. 1, pp. 37–61, 2020. [Citado na página 26]
- [31] H. Simon, “As ciências do artificial, 2ª edição,” *Arménio Amado*, 1981. [Citado na página 26]
- [32] A. R. Hevner, S. T. March, J. Park, and S. Ram, “Design science in information systems research,” *MIS quarterly*, pp. 75–105, 2004. [Citado na página 27]
- [33] “Scrum: Saiba tudo sobre este framework ágil.” Available at <https://k21.global/blog/o-que-e-scrum>. (Last accessed in 18/05/2021). [Citado na página 30]

- [34] “Outsystems development and management tools: Evaluation guide.” Available at <https://www.outsystems.com/evaluation-guide/development-and-management-tools/>. (Last accessed in 19/07/2021). [Citado na página 35]
- [35] Atlassian, “Jira: Issue amp; project tracking software.” Available at <https://www.atlassian.com/software/jira>. (Last accessed in 25/07/2021). [Citado na página 40]

Anexo A

Documentação das API's do Sistema Backend

Backend API

Implementation

Data Model

Structures

Accepted

No Description

Attributes

Attribute	Description	Type	Length	Decimals	Default	Mandatory
WasteID	Waste Identifier					
Accepted		Boolean				✓

Analise

No Description

Attributes

Attribute	Description	Type	Length	Decimals	Default	Mandatory
AnalisedID	Analised Identifier					
Analised	Analised					

Confirm

No Description

Attributes

Attribute	Description	Type	Length	Decimals	Default	Mandatory
Id	Integer					
Value	Boolean					✓

DataDiscount

No Description

Attributes

Attribute	Description	Type	Length	Decimals	Default	Mandatory
Userid	User Identifier					
DiscountId	UserDiscounts Identifier					
Used	Boolean					

Deliverer

No Description

Attributes

Figura A.1: Documentação do Sistema Backend (I).

Attribute	Description	Type	Length	Decimals	Default	Mandatory
Date		Date				✓
Image		Binary Data				✓
WasteId		Waste Identifier				✓
DeliveryPointId		DeliveryPoint Identifier				✓
UserId		User Identifier				✓
AnalisedId		Analised Identifier				✓

Select

Supports tables with multiple checkboxes and allows **selecting** them in bulk.

Attributes

Attribute	Description	Type	Length	Decimals	Default	Mandatory
Value		Boolean				

Actions

Service Actions

DeliveryPoint
No Description

AddDeliveryPoint_SA
No Description

Input parameters

Name	Description	Data Type	Mandatory
Info		DeliveryPoint	✓

Output parameters

Name	Description	Data Type
Confirm		Confirm

DeliveryPointDelete_SA_NotInUse
No Description

Input parameters

Name	Description	Data Type	Mandatory
DeliveryPointID		DeliveryPoint Identifier	✓

Output parameters

Name	Description	Data Type
Confirm		Confirm

DeliveryPointUpdate_SA
No Description

Input parameters

Name	Description	Data Type	Mandatory
Info		DeliveryPoint	✓

Figura A.2: Documentação do Sistema Backend (II).

Output parameters

Name	Description	Data Type
Confirm		Confirm

User
No Description

AnaliseDelete_SA
No Description

Input parameters

Name	Description	Data Type	Mandatory
AnalisedId	Analised Identifier		✓

Output parameters

Name	Description	Data Type
Confirm		Confirm

DeleteUserDiscount_SA
No Description

Input parameters

Name	Description	Data Type	Mandatory
DiscountId	UserDiscounts Identifier		✓
UserId	User Identifier		✓

Output parameters

Name	Description	Data Type
Confirm		Confirm

GiveUserPoints_SA_NotInUse
No Description

Input parameters

Name	Description	Data Type	Mandatory
Data	UserPoints		✓

Output parameters

Name	Description	Data Type
Confirm		Confirm

Register_SA
Register the user

Input parameters

Name	Description	Data Type	Mandatory
UserData	User		✓

Output parameters

Name	Description	Data Type
Confirm		Confirm

Figura A.3: Documentação do Sistema Backend (III).

UserDelete_SA
User cancels the account

Input parameters

Name	Description	DataType	Mandatory
UserId	User Identifier		✓

Output parameters

Name	Description	DataType
Confirm	Confirm	

UserDiscount_SA
Creates a entry on the UserDiscount table and in the QrDiscount table

Input parameters

Name	Description	DataType	Mandatory
UserId	User Identifier		✓
DiscountId	Discounts Identifier		✓
Points	Integer		✓

Output parameters

Name	Description	DataType
Confirm	Confirm	

Waste
Crud for waste table

AddMaterials_SA
No Description

Input parameters

Name	Description	DataType	Mandatory
DataMaterials	Materials		✓

Output parameters

Name	Description	DataType
Confirm	Confirm	

AddWaste_SA
Add waste to the database by the administrator

Input parameters

Name	Description	DataType	Mandatory
DataWaste	Waste		✓

Output parameters

Name	Description	DataType
Confirm	Confirm	

AddWasteDetails_SA

Figura A.4: Documentação do Sistema Backend (IV).

No Description

Input parameters

Name	Description	Data Type	Mandatory
DataWasteDetails		WasteDetails	✓

Output parameters

Name	Description	Data Type
Confirm		Confirm

🔍 **FindWaste_SA_NotInUse**
Find out is any of the tags from the AzureVision is in the table Waste

Input parameters

Name	Description	Data Type	Mandatory
Name		Text	✓
Confidence		Decimal	✓

Output parameters

Name	Description	Data Type
Accepted		Accepted

🔍 **MaterialsDelete_SA_NotInUse**
No Description

Input parameters

Name	Description	Data Type	Mandatory
MaterialsId		Materials Identifier	✓

Output parameters

Name	Description	Data Type
Confirm		Confirm

🔍 **MaterialsUpdate_SA**
No Description

Input parameters

Name	Description	Data Type	Mandatory
DataMaterials		Materials	✓

Output parameters

Name	Description	Data Type
Confirm		Confirm

🔍 **WasteDelete_SA_NotInUse**
Deletion of waste product by the administrator

Input parameters

Name	Description	Data Type	Mandatory
Wasteld		Waste Identifier	✓

Output parameters

Name	Description	Data Type

Figura A.5: Documentação do Sistema Backend (V).

Name	Description	Data Type
Confirm		Confirm

🔍 WasteDetailsDelete_SA_NotInUse
No Description

Input parameters

Name	Description	Data Type	Mandatory
WasteDetailsId	WasteDetails Identifier		✔

Output parameters

Name	Description	Data Type
Confirm		Confirm

🔍 WasteDetailsUpdate_SA
No Description

Input parameters

Name	Description	Data Type	Mandatory
DataWasteDetails	WasteDetails		✔

Output parameters

Name	Description	Data Type
Confirm		Confirm

🔍 WasteErrorDelete_SA_NotInUse
No Description

Input parameters

Name	Description	Data Type	Mandatory
ErrorId	Error Identifier		✔

Output parameters

Name	Description	Data Type
Confirm		Confirm

🔍 WasteUpdate_SA
Update a waste values and materials by the administrator

Input parameters

Name	Description	Data Type	Mandatory
DataWaste	Waste		✔

Output parameters

Name	Description	Data Type
Confirm		Confirm

📁 Admin
No Description

Figura A.6: Documentação do Sistema Backend (VI).

AdminGivePoints_SA No Description			
Input parameters			
Name	Description	DataType	Mandatory
Data	DeliveryPoint, Error, Select, User, Waste Record List		
Output parameters			
Name	Description	DataType	Mandatory
Confirm	Confirm		
AdminRemoveDeliveryPoint_SA No Description			
Input parameters			
Name	Description	DataType	Mandatory
Data	DeliveryPoint, Select Record List		
Output parameters			
Name	Description	DataType	Mandatory
Confirm	Confirm		
AdminRemoveError_SA No Description			
Input parameters			
Name	Description	DataType	Mandatory
Data	DeliveryPoint, Error, Select, User, Waste Record List		
Output parameters			
Name	Description	DataType	Mandatory
Confirm	Confirm		
AdminRemoveWaste_SA No Description			
Input parameters			
Name	Description	DataType	Mandatory
WasteList	Select, Waste Record List		
MaterialsList	Materials, Select Record List		
WasteDetailsList	Select, WasteDetails Record List		
Output parameters			
Name	Description	DataType	Mandatory
Confirm	Confirm		
AdminRemoveWaste_SA_NotInUse No Description			
Input parameters			
Name	Description	DataType	Mandatory
WasteList	Select, Waste Record List		
MaterialsList	Materials, Select Record List		

Figura A.7: Documentação do Sistema Backend (VII).

Name	Description	Data Type	Mandatory
WasteDetailsList	Select, WasteDetails	Record List	

Output parameters

Name	Description	Data Type
Confirm		Confirm

Partners
No Description

AddDiscounts_SA
No Description

Input parameters

Name	Description	Data Type	Mandatory
Discount		Discounts	✓

Output parameters

Name	Description	Data Type
Confirm		Confirm

AddPartners_SA
Add partners to the app and the discounts to that partner by the administrator

Input parameters

Name	Description	Data Type	Mandatory
Partner		Partners	✓

Output parameters

Name	Description	Data Type
Confirm		Confirm

DeleteDiscount_SA
Delete a discount from the Discount table

Input parameters

Name	Description	Data Type	Mandatory
DiscountId		Discounts Identifier	✓

Output parameters

Name	Description	Data Type
Confirm		Confirm

DeletePartner_SA
Delete a partner from the Partners table

Input parameters

Name	Description	Data Type	Mandatory
PartnerId		Partners Identifier	✓

Output parameters

Figura A.8: Documentação do Sistema Backend (VIII).

Name	Description	Data Type
Confirm		Confirm

UpdateDiscount_SA
Update the Discount table

Input parameters

Name	Description	Data Type	Mandatory
Discount		Discounts	✓

Output parameters

Name	Description	Data Type
Confirm		Confirm

UpdatePartner_SA
Update a Partner

Input parameters

Name	Description	Data Type	Mandatory
Partner		Partners	✓

Output parameters

Name	Description	Data Type
Confirm		Confirm

Utility
No Description

AnaliseImageAndAnaliseAdd_SA
No Description

Input parameters

Name	Description	Data Type	Mandatory
Image		Binary Data	✓

Output parameters

Name	Description	Data Type
Confirm		Confirm

AnaliseNotRegister_SA
No Description

Input parameters

Name	Description	Data Type	Mandatory
Image		Binary Data	✓

Output parameters

Name	Description	Data Type
Confirm		Confirm
WasteID		Waste Identifier

Figura A.9: Documentação do Sistema Backend (IX).

GenerateToken_SA
No Description

Output parameters

Name	Description	DataType
↳ FirebaseToken		Text
↳ FirebaseProjectId		Text

TmplImageDelete_SA
No Description

Input parameters

Name	Description	DataType	Mandatory
↳ TmplImageID	TmplImage Identifier		✔

Output parameters

Name	Description	DataType
↳ Confirm	Confirm	

Referenced Service Actions

Name	Owner	Description
↳ Notification_Admin	EventsHub	

Figura A.10: Documentação do Sistema Backend (X).

Anexo B

Sistema Delivery Point

B.1 Desenho técnico do Delivery Point

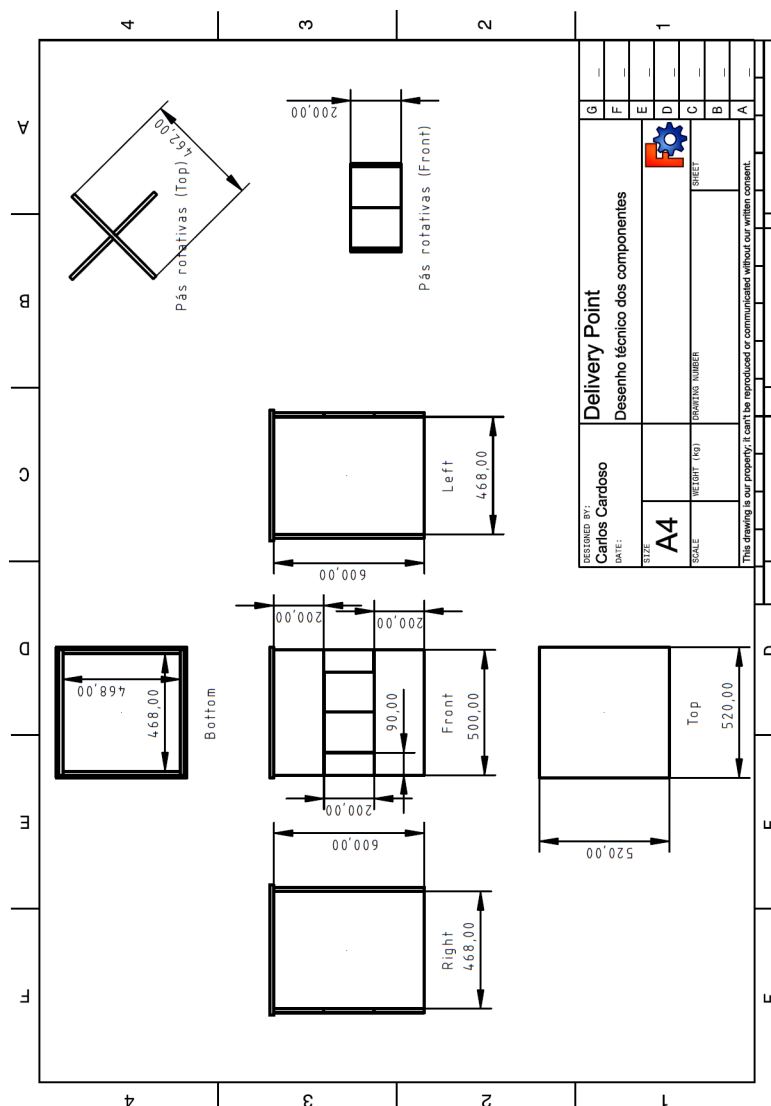


Figura B.1: Desenho técnico das partes do *Delivery Point*.

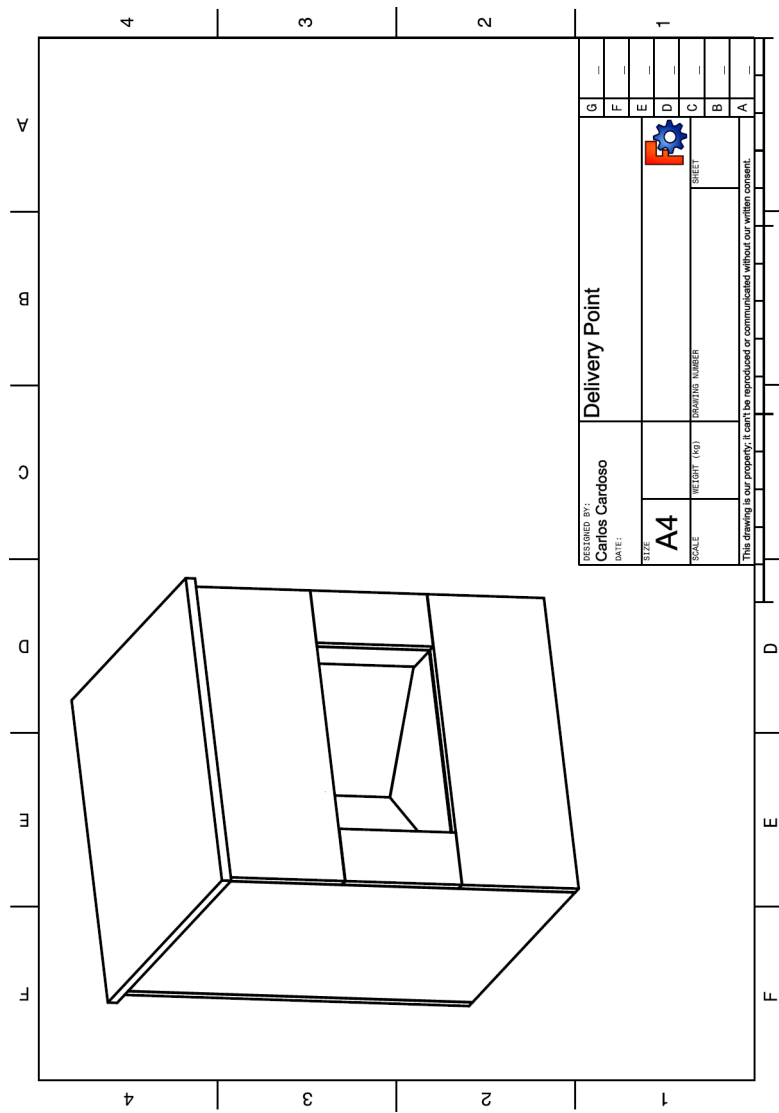



Figura B.2: Desenho técnico do *Delivery Point*.

B.2 Documentação da API do Delivery Point

 outsystems
REST API Documentation

Image

Receção da imagem vinda do Raspberry pi para envio para Azure Vision

[API DEFINITION URL: https://edulab01.outsystemsenterprise.com/RPI_API/rest/Image/swagger.json]

List Operations | Expand Operations

POST /Post

Request URL

`https://edulab01.outsystemsenterprise.com/RPI_API/rest/Image/Post`

Parameters

Parameter	Description	Parameter Type	Data Type (Model Example)
Request		body	<pre>{ "RPI_ID": 1234567891234567, "User": 0, "Waste": 1234567891234567, "AnalisedID": 1234567891234567, "Weight": 0, "Image": "string" }</pre>

Response Messages

HTTP Status Code	Description	Content-Type	Data Type (Model Example)
200		text/plain	<code>true</code>

[BASE URL: /RPI_API/rest/Image]

Built with OutSystems

Figura B.3: Documentação da API.

Anexo C

Código do Delivery Point

C.1 Código linguagem Kivy para criação do UI

```
1 #:kivy 1.0
2 <NewLabel@Label>:
3
4     halign: "center"
5     color:1,1,1,1
6     canvas.before:
7         Color:
8             rgba: 0.2,0.2, 0.2,1
9         Rectangle:
10            pos: self.pos
11            size: self.size
12
13 <ButtonClass@Button>:
14
15     font_size: "22sp"
16     background_normal: ''
17     background_color: 0.92549,0.92549,0.92549, 1
18     color:0,0,0,1
19     canvas.before:
20         Color:
21             rgba: 0,956862745, 0,9764705882, 0,988235294, 0.8
22         RoundedRectangle:
23             pos: self.pos
```

```

24         size: self.size
25
26 #----- Player -----
27
28 <Player>:
29     name: "Player"
30     video: videoId #easier to access this way
31
32     Video:
33         id: videoId
34         source: "movie.mp4"
35         state: "play"
36         options: {'eos': 'loop'}
37         on_touch_down:
38             root.on_stop()
39             app.root.current = "QRCode"
40
41 #----- QRCode -----
42
43 <QRCode>:
44     name: "QRCode"
45
46     FloatLayout:
47         NewLabel:
48             text: 'Present your QR Code for scan'
49             font_size: "22sp"
50             size_hint: (1, 0.17)
51             pos_hint: {'x': 0, 'y': 0.8}
52
53         Image:
54             id: qrimage
55             source: 'QR.png'
56             size_hint: (1.0, 0.5)
57             pos_hint: {'x': 0, 'y': 0.25}
58
59         ButtonClass:
60             id: but_1
61             text: 'Ready to deliver'
62             font_size: "22sp"
63             pos_hint: {'x': 0.33, 'y': 0.05}
64             size: 275, 75
65             size_hint: None, None
66             on_press: root.qr_click()
67
68         Button:
69             id: but_2
70             text: 'Cancel'
71             font_size: "22sp"
72             pos_hint: {'x': 0.85, 'y': 0.05}

```

```
73         size: 100, 50
74         size_hint: None, None
75         on_press:
76             root.cancel()
77             app.root.current = "Player"
78
79 #----- Photo -----
80
81 <Photo>:
82     name: "Photo"
83
84     FloatLayout:
85         NewLabel:
86             text: 'Place the waste'
87             font_size: "22sp"
88             size_hint: (1, 0.17)
89             pos_hint: {'x': 0, 'y': 0.8}
90
91         Image:
92             id: imageView
93             source: 'Photo.png'
94             size_hint: (1.0, 0.5)
95             pos_hint: {'x': 0, 'y': 0.25}
96
97         ButtonClass:
98             id: but_1
99             text: 'Ready to deliver'
100            font_size: "22sp"
101            pos_hint: {'x': 0.3, 'y': 0.05}
102            size: 275, 75
103            size_hint: None, None
104            on_press: root.photo_click()
105
106         Button:
107             id: but_2
108             text: 'Cancel'
109             font_size: "22sp"
110             pos_hint: {'x': 0.85, 'y': 0.05}
111             size: 100, 50
112             size_hint: None, None
113             on_press:
114                 root.cancel()
115                 app.root.current = "Player"
116
117 #----- Error -----
118
119 <Error>:
120     name: "Error"
121
```

```
122     FloatLayout :
123         NewLabel:
124             text: 'Delivery Point not working'
125             font_size: "24sp"
126             size_hint: (1.0, 0.5)
127             pos_hint: {'x': 0, 'y': 0.3}
128
129 #----- Wrong Waste -----
130
131 <Wrong>:
132     name: "Wrong"
133
134     FloatLayout :
135         NewLabel:
136             text: 'Waste delivered not the same as the analised'
137             font_size: "24sp"
138             size_hint: (1.0, 0.5)
139             pos_hint: {'x': 0, 'y': 0.3}
140
141 #----- Already delivered -----
142
143 <SameQR>:
144     name: "SameQR"
145
146     FloatLayout :
147         NewLabel:
148             text: 'QR code already scanned'
149             font_size: "24sp"
150             size_hint: (1.0, 0.5)
151             pos_hint: {'x': 0, 'y': 0.3}
152
153 #----- Thank you -----
154
155 <Thank>:
156     name: "Thank"
157
158     FloatLayout :
159         NewLabel:
160             text: 'Thank you for the delivery'
161             font_size: "24sp"
162             size_hint: (1.0, 0.5)
163             pos_hint: {'x': 0, 'y': 0.3}
```

Listagem C.1: Código linguagem Kivy para criação do UI.

C.2 Código linguagem Python para o funcionamento do Delivery Point

```
1 from logging import root
2 from kivy.app import App
3 from kivy.logger import Logger
4 from kivy.uix.screenmanager import ScreenManager, Screen
5 from kivy.lang import Builder
6 from kivy.properties import StringProperty
7 from kivy.clock import Clock
8 from kivy.uix.image import Image
9 from kivy.graphics.texture import Texture
10
11 import RPi.GPIO as GPIO
12 import time
13 from pyzbar.pyzbar import decode
14 import datetime
15 import requests
16 import json
17 import cv2
18 import base64
19 from io import BytesIO
20
21 from kivy.core.window import Window
22
23 Window.clearcolor = (0.466666, 0.7372, 0.24705, 0.8)
24 Window.size = (800, 480)
25
26 # Funcao para confirmar se ja foi analisado o QR code
27 def check_qr_code(file_name, string):
28     with open(file_name, "r") as read:
29         for line in read:
30             if string in line:
31                 return True
32     return False
33
34
35 # Funcao para obtencao do peso
36 def weight():
37     i = 0
38     Count = 0
39     DT = 15 # Data from the HX711
40     SCK = 14 # Clock pin
41     time.sleep(0.001)
42     GPIO.setup(DT, GPIO.OUT)
43     GPIO.output(DT, 1)
44     GPIO.output(SCK, 0)
```

```

45     GPIO.setup(DT, GPIO.IN)
46
47     while GPIO.input(DT) == 1:
48         i = 0
49         for i in range(24):
50             GPIO.output(SCK, 1)
51             Count = Count << 1
52
53             GPIO.output(SCK, 0)
54             time.sleep(0.001)
55             if GPIO.input(DT) == 0:
56                 Count = Count + 1
57
58         GPIO.output(SCK, 1)
59         Count = Count ^ 0x800000
60         time.sleep(0.001)
61         GPIO.output(SCK, 0)
62         if Count < 0:
63             Count = App.get_running_app().OldCount
64         else:
65             App.get_running_app().OldCount = Count
66         return Count
67
68
69 # Funcao de envio de dados para a API criada em Outsystems
70 def send(data):
71     App.get_running_app().response = ""
72     while App.get_running_app().response == "":
73         App.get_running_app().response = requests.post(
74             App.get_running_app().URL_API,
75             data=json.dumps(data),
76             headers=App.get_running_app().headers,
77         )
78     return App.get_running_app().response
79
80
81 # ----- Screens -----
82
83 class WindowManager(ScreenManager):
84     pass
85
86
87 class Player(Screen):
88     def __init__(self, **kwargs):
89         super(Player, self).__init__(**kwargs)
90         self.bind(on_touch_down=self.on_stop)
91
92     def on_stop(self, *args):
93         self.video.state = "stop"

```

```
94
95
96 class QRCode(Screen):
97     def __init__(self, **kwargs):
98         super(QRCode, self).__init__(**kwargs)
99
100    def on_enter(self):
101        cam = cv2.VideoCapture(0)
102        cam.set(cv2.CAP_PROP_FOURCC, cv2.VideoWriter_fourcc("M", "J",
103        , "P", "G"))
104        cam.set(3, 1280)
105        cam.set(4, 720)
106        self.capture = cam
107
108    def qr_click(self):
109        Clock.schedule_interval(self.update, 1.0 / 30)
110        self.ids.but_1.disabled = True
111
112    def cancel(self):
113        self.capture.release()
114        Clock.unschedule(self.update)
115        screen1.add_widget(Player())
116        self.parent.current = "Player"
117        self.ids.but_1.disabled = False
118
119    def error(self):
120        self.capture.release()
121        Clock.unschedule(self.update)
122        self.parent.current = "Error"
123        self.ids.but_1.disabled = False
124
125    def sameQR(self):
126        self.capture.release()
127        Clock.unschedule(self.update)
128        self.parent.current = "SameQR"
129        self.ids.but_1.disabled = False
130
131    def photo(self):
132        self.capture.release()
133        Clock.unschedule(self.update)
134        self.parent.current = "Photo"
135        self.ids.but_1.disabled = False
136
137    def update(self, dt):
138        if App.get_running_app().QrSuccess == False:
139            success, frame = self.capture.read()
140
141            if not success:
```

```

142         x = datetime.datetime.now()
143         day = x.strftime("%d_%m_%Y_%H_%M")
144         erro = "Erro a abrir camara no dia {}".format(day)
145         f = open("log.txt", "a")
146         f.write(erro)
147         f.close()
148         # Desligar camara e LED's
149         self.error()
150
151     else:
152         if check_qr_code("data.txt", App.get_running_app().
dataRaw):
153             # QR code ja utilizado
154             self.sameQR()
155
156         elif (
157             App.get_running_app().data == ""
158             and App.get_running_app().camera == True
159         ):
160
161             for code in decode(frame):
162                 App.get_running_app().dataRaw = code.data.
decode("utf-8")
163                 App.get_running_app().dataDump = json.dumps(
164                     App.get_running_app().dataRaw
165                 )
166                 App.get_running_app().data = json.loads(
167                     App.get_running_app().dataDump
168                 )
169
170                 if App.get_running_app().data != "":
171                     App.get_running_app().QrSuccess = True
172                     # Desligar camara e LED's
173                     self.photo()
174
175
176 class Photo(Screen):
177     def __init__(self, **kwargs):
178         super(Photo, self).__init__(**kwargs)
179
180     # Enter when first time load the screen
181     def on_enter(self):
182         if App.get_running_app().QrSuccess == False:
183             self.redirect()
184         else:
185             # Criar instante da camara
186             cam = cv2.VideoCapture(0)
187             cam.set(cv2.CAP_PROP_FOURCC, cv2.VideoWriter_fourcc("M",
"J", "P", "G"))

```

```
188         cam.set(3, 1280)
189         cam.set(4, 720)
190         self.capture = cam
191
192     def photo_click(self):
193         Clock.schedule_interval(self.update, 1.0 / 30)
194         self.ids.but_1.disabled = True
195
196     def redirect(self):
197         screen1.add_widget(Player())
198         self.parent.current = "Player"
199         self.ids.but_1.disabled = False
200
201     def cancel(self):
202         self.capture.release()
203         Clock.unschedule(self.update)
204         screen1.add_widget(Player())
205         self.parent.current = "Player"
206         self.ids.but_1.disabled = False
207
208     def wrong(self):
209         self.capture.release()
210         Clock.unschedule(self.update)
211         self.parent.current = "Wrong"
212         self.ids.but_1.disabled = False
213
214     def thank(self):
215         self.capture.release()
216         Clock.unschedule(self.update)
217         self.parent.current = "Thank"
218         self.ids.but_1.disabled = False
219
220     def update(self, dt):
221         success, frame = self.capture.read()
222
223         if App.get_running_app().flag == False:
224             if not success:
225                 x = datetime.datetime.now()
226                 day = x.strftime("%d_%m_%Y_%H_%M")
227                 erro = "Erro a abrir camera no dia {}".format(day
228             )
229
230                 f = open("log.txt", "a")
231                 f.write(erro)
232                 f.close()
233                 # Desligar camera e LED's
234                 self.cancel()
235
236         count = weight()
237         Peso = (count - App.get_running_app().sample) / -240
```

```

236         if Peso > 0 and Peso > 20:
237
238             time.sleep(5)
239
240             if App.get_running_app().flag == False:
241
242                 x = datetime.datetime.now()
243                 day = x.strftime("%d %m %Y %H %M")
244                 img_name = "frame_{}.jpg".format(day)
245
246                 # Captura da imagem
247                 cv2.imwrite(img_name, frame)
248
249                 time.sleep(1)
250                 # Desligar camara e LED's
251                 self.capture.release()
252                 GPIO.output(LED_output, GPIO.LOW)
253
254                 # Codificar a imagem em base64 para envio
255                 with open(img_name, "rb") as image:
256                     encode_string = base64.b64encode(image.read())
257
258                 json_data = json.loads(App.get_running_app().data)
259
260                 # Formatar dados a enviar para a API
261                 data = {
262                     "RPI_ID": App.get_running_app().RPI_ID,
263                     "User": json_data["Analised"]["UserId"],
264                     "Waste": json_data["Analised"]["WasteId"],
265                     "AnalisedID": json_data["AnalisedID"],
266                     "Weight": int(Peso),
267                     "Image": encode_string.decode("utf-8"),
268                 }
269
270                 App.get_running_app().response = send(data)
271
272                 App.get_running_app().flag = True
273
274                 if App.get_running_app().response.text == "true":
275                     if GPIO.input(App.get_running_app().S_motor) == GPIO
276                     .LOW:
277                         GPIO.output(App.get_running_app().in1, GPIO.LOW)
278                         GPIO.output(App.get_running_app().in2, GPIO.HIGH
279                     )
280
281                     if App.get_running_app().n < 100:
282                         App.get_running_app().n = App.
283                     get_running_app().n + 1
284                     else:

```

```
281             App.get_running_app().speed.ChangeDutyCycle
282         (3)
283             elif GPIO.input(App.get_running_app().S_motor) ==
GPIO.HIGH:
284                 GPIO.output(App.get_running_app().in1, GPIO.LOW)
285                 GPIO.output(App.get_running_app().in2, GPIO.LOW)
286                 f = open("data.txt", "a")
287                 f.write(App.get_running_app().dataRaw)
288                 f.close()
289                 self.cancel()
290             elif App.get_running_app().response.text == "false":
291                 self.wrong()
292
293         elif Peso >= 0 and Peso < 20:
294             # Desligar o motor
295             GPIO.output(App.get_running_app().in1, GPIO.LOW)
296             GPIO.output(App.get_running_app().in2, GPIO.LOW)
297             f = open("data.txt", "a")
298             f.write(App.get_running_app().dataRaw)
299             f.close()
300             self.thank()
301
302     class Error(Screen):
303         def __init__(self, **kwargs):
304             super(Error, self).__init__(**kwargs)
305
306         def on_enter(self):
307             Clock.schedule_interval(self.update, 5)
308
309         def redirect(self):
310             Clock.unschedule(self.update)
311             screen1.add_widget(Player())
312             self.parent.current = "Player"
313
314         def update(self, dt):
315             self.redirect()
316
317
318     class Wrong(Screen):
319         def __init__(self, **kwargs):
320             super(Wrong, self).__init__(**kwargs)
321
322         def on_enter(self):
323             Clock.schedule_interval(self.update, 5)
324
325         def redirect(self):
326             Clock.unschedule(self.update)
327             screen1.add_widget(Player())
```

```
328         self.parent.current = "Player"
329
330     def update(self, dt):
331         self.redirect()
332
333
334 class SameQR(Screen):
335     def __init__(self, **kwargs):
336         super(SameQR, self).__init__(**kwargs)
337
338     def on_enter(self):
339         Clock.schedule_interval(self.update, 5)
340
341     def redirect(self):
342         Clock.unschedule(self.update)
343         screen1.add_widget(Player())
344         self.parent.current = "Player"
345
346     def update(self, dt):
347         self.redirect()
348
349 class Thank(Screen):
350     def __init__(self, **kwargs):
351         super(Thank, self).__init__(**kwargs)
352
353     def on_enter(self):
354         Clock.schedule_interval(self.update, 5)
355
356     def redirect(self):
357         Clock.unschedule(self.update)
358         screen1.add_widget(Player())
359         self.parent.current = "Player"
360
361     def update(self, dt):
362         self.redirect()
363
364
365 # KV file
366 kv = Builder.load_file("main.kv")
367
368
369 # Screens usados
370 sm = ScreenManager()
371
372 screen1 = Screen(name="Player")
373 screen1.add_widget(Player())
374 sm.add_widget(screen1)
375
376 sm.add_widget(QRCode(name="QRCode"))
```

```
377
378 sm.add_widget(Photo(name="Photo"))
379
380 sm.add_widget(Error(name="Error"))
381
382 sm.add_widget(Wrong(name="Wrong"))
383
384 sm.add_widget(SameQR(name="SameQR"))
385
386 sm.add_widget(Thank(name="Thank"))
387
388 sm.current = "Player"
389
390
391 class DeliveryPointApp(App):
392
393     # Configuracao do funcionamento dos pinos do raspberry pi
394     GPIO.setwarnings(False)
395     GPIO.setmode(GPIO.BCM)
396
397     # Pinos do motor
398     in1 = 24
399     in2 = 23
400     en = 25
401
402     # Pinos dos sensores
403     S_motor = 26
404     S_porta = 2
405
406     # Variaveis
407     sample = 0
408     n = 0
409     response = ""
410     dataRaw = "?"
411     camera = True
412     data = ""
413     QrSuccess = False
414     flag = False
415     sample = weight()
416     OldCount = 0
417
418     GPIO.setup(S_motor, GPIO.IN, pull_up_down=GPIO.PUD_DOWN)
419
420     # Outputs GPIO's
421     GPIO.setup(14, GPIO.OUT)
422     GPIO.setup(in1, GPIO.OUT)
423     GPIO.setup(in2, GPIO.OUT)
424     GPIO.setup(en, GPIO.OUT)
425     GPIO.output(in1, GPIO.LOW)
```

```
426 GPIO.output(in2, GPIO.LOW)
427 Frequencia de 1KHz
428 speed = GPIO.PWM(en, 5000)
429 speed.start(7)
430
431 # URL da API
432 URL_API = "https://edulab01.outsystemsenterprise.com/RPI_API/
rest/Image/Post"
433
434 # ID do Delivery Point e chave de autentica o
435 RPI_ID = "1"
436 API_KEY = "traga-bolas"
437
438 # Definicao dos header de envio
439 content_type = "application/json"
440 headers = {"content-type": content_type, "API_KEY": API_KEY}
441
442 def build(self):
443     return sm
444
445
446 if __name__ == "__main__":
447     DeliveryPointApp().run()
```

Listagem C.2: Código linguagem Python para o funcionamento do Delivery Point.