



ESTG



INSTITUTO POLITÉCNICO
DE VIANA DO CASTELO

PLATAFORMA DE RASTREABILIDADE PARA A CADEIA DE VALOR DO SETOR TÊXTIL E DO VESTUÁRIO

2023

PLATAFORMA DE RASTREABILIDADE PARA A CADEIA DE VALOR DO SETOR TÊXTIL E DO VESTUÁRIO

Ricardo Abreu Dias



INSTITUTO POLITÉCNICO
DE VIANA DO CASTELO

Ricardo Abreu Dias

PLATAFORMA DE RASTREABILIDADE PARA A CADEIA DE
VALOR DO SETOR TEXTIL E DO VESTUÁRIO

Nome do Curso de Mestrado
Engenharia Informática

Trabalho efectuado sob a orientação do
Professora Doutora Maria Estrela Ferreira Cruz
Professor Doutor António Miguel Rosado da Cruz

Julho de 2023

Resumo

O impacto ambiental do Setor Têxtil e do Vestuário tem vindo a crescer a um ritmo acelerado nas últimas décadas. A ausência de políticas ambientais e sociais no que diz respeito à confeção dos produtos têxteis associados à produção de produtos mais baratos e, por vezes, com menor qualidade, leva a um grande aumento do consumo gerando, no final, um enorme aumento do desperdício. Este aumento do consumo provoca um forte impacto ambiente global, não só porque é necessário aumentar o tratamento do lixo, mas também porque aumenta o consumo de matérias primas aumentando a degradação dos solos e o desperdício de água. Para além disso, a procura de mão de obra mais barata por parte das empresas para conseguirem vantagem competitiva, leva a que os produtos (intermédios e finais) percorram largas distancias aumentando assim o impacto ambiental final dos produtos.

Face a este problema, é necessário investigar e desenvolver soluções no sentido de minimizar esses impactos através da consciencialização dos consumidores, mas também, e de forma mais rígida, descobrir formas de controlar e monitorizar o desenvolvimento desses produtos e disponibilizar essa informação para todos os interessados. Assim, a criação de uma plataforma que permita acompanhar o movimento da produção têxtil e dos seus sub-produtos insere-se na progressão da descoberta de soluções para resolver os problemas que hoje encontramos neste setor. No trabalho aqui apresentado, propõe-se uma plataforma onde são registadas todos os lotes produzidos e as atividades que vão sendo executadas em todo o ciclo produtivo da indústria têxtil e vestuário, desde o cultivo das matérias-primas, a criação de fibras e tecidos, o corte, produção e acabamento até chegar ao consumidor final, não esquecendo todas as movimentações geográficas que os sub-produtos sofrem ao longo da cadeia. Para além da grande vantagem de serem armazenados todos os dados de rastreabilidade, estes são armazenados em blockchain o que torna a informação confiável, segura e imutável.

A recolha de toda esta informação possibilita o desenvolvimento algoritmos que, de forma mais ou menos exata, conseguem calcular os impactos sociais e ambientais. Estes algoritmos podem ser desenvolvidos de forma isolada (para um produto) e global e podem ser demonstrados, e os seus resultados apresentados a todos os intervenientes da cadeia de valor incluindo ao consumidor final. Esta informação é especialmente importante para que o consumidor final a possa ter em consideração aquando da renovação do seu guarda roupa e para que possa ser usada para aumentar a consciência ambiental e social.

A aplicação, desenvolvida neste projeto de investigação e aqui apresentada, permite visualizar através de um sistema interativo de grafos orientados e de grafos sobre um mapa geográfico toda a informação das atividades e lotes produzidos proveniente de fontes seguras sem qualquer possibilidade de manipulação de dados, contendo os indicadores mais importantes a nível de impactos ambiental e sociais.

Abstract

The environmental impact of the Textile and Clothing Sector has been growing at an accelerated pace in recent decades. The absence of environmental and social policies regarding the manufacturing of textile products associated with the production of cheaper products and, sometimes, with lower quality, leads to a large increase in consumption, generating, in the end, a huge increase in waste. This increase in consumption has a strong impact on the global environment, not only because it is necessary to increase waste treatment, but also because it increases the consumption of raw materials, increasing soil degradation and water waste. In addition, the search for cheaper labor by companies to achieve a competitive advantage means that products (intermediate and final) travel long distances, thus increasing the environmental impact of the products. With this being said, it is necessary to investigate and develop solutions to minimize these impacts through consumer awareness, but also, and more rigidly, to find ways to control and monitor the development of these products and make this information available to all interested parties. Thus, the creation of a platform that allows us to follow the movement of textile production and its by-products is part of the progression of finding solutions to solve the problems that we face in this sector today. In this project, a platform is proposed where all the batches produced and the activities that are carried out throughout the production cycle of the textile and clothing industry are registered, from the cultivation of raw materials, the creation of fibers and fabrics, the cutting, production and finishing until reaching the final consumer, not forgetting all the geographical movements that the by-products undergo along the chain. In addition to the great advantage of storing all traceability data, these are stored in blockchain which makes the information reliable, secure, and immutable. By collecting all this information, it is possible to develop algorithms that can, somewhat, calculate social and environmental impacts. These algorithms can be developed in isolation (for a product) and globally, and can be demonstrated, and their results presented to all members of the value chain, including the final consumer. This information is especially important, allowing the final consumer to take it into account when renewing their wardrobe, and so that it can be used to increase environmental and social awareness. The application, developed in this research project and presented here, allows viewing, through an interactive system of oriented graphs and graphs on a geographic map, all the information on the activities and batches produced from secure sources without any possibility of data manipulation, containing the most important indicators in terms of environmental and social impacts.

Agradecimentos

O desenvolvimento desta dissertação é o resultado de todo o meu percurso académico e da educação que me foi inculcada ao longo de toda a minha vida.

Assim sendo, o mérito é vindo em grande parte dos meus pais, aos quais agradeço pela oportunidade que me deram em me educarem para ser o homem que sou hoje, e o sucesso académico e pessoal que tenho vindo a conseguir ao longo dos anos.

Agradeço a minha irmã e restante família, incluso os que já partiram, por serem boas influências e me terem ajudado sempre que necessitei.

Tenho também um forte apreço por M. Estrela Cruz e A. Miguel Cruz, que tem desempenhado o papel de orientadores não só na dissertação mas também na minha carreira como investigador e em muitos outros projetos.

Ricardo Abreu Dias

Reconhecimentos

Este projeto de investigação foi desenvolvido no âmbito do projeto “StvGoDigital: Digitalização da Cadeia de Valor do Setor Têxtil e Vestuário”, especificamente no seu PPS1, com referência PPOCI-01-0247-FEDER-046086, financiado pelo Fundo Europeu de Desenvolvimento Regional (FEDER), através do Programa Operacional Competitividade e Internacionalização (COMPETE 2020).

“Quanto mais aumenta nosso conhecimento, mais evidente fica nossa ignorância.”

John F. Kennedy

Conteúdos

1	Introdução	1
1.1	Motivação	2
1.2	Objetivos	3
1.3	Metodologia	4
1.4	Estrutura da Dissertação	5
2	Conceitos e ferramentas	6
2.1	Introdução	6
2.2	Conceitos	6
2.2.1	Cadeia de valor	6
2.2.2	Cadeia de valor no Setor Têxtil e do Vestuário	7
2.2.3	Rastreabilidade	8
2.2.4	Sustentabilidade	8
2.2.5	Blockchain	9
2.3	Ferramentas	9
2.3.1	JavaScript	9
2.3.2	TypeScript	10
2.3.3	GIT	10
2.3.4	PL/pgSQL	10
2.3.5	JSON	10
2.3.6	UUID	11
2.3.7	Bcrypt	11
2.3.8	React	11
2.3.9	Nextjs	11
2.3.10	Material UI	12
2.3.11	Axios	12
2.3.12	NodeJs	12
2.3.13	Express	12
2.3.14	Sequelize	12
2.3.15	Jest	13
2.3.16	Supertest	13
2.3.17	i18next	13
2.3.18	Express-session	13
2.3.19	Express-validator	13
2.3.20	Leaflet	14
2.3.21	Redis	14

3	Levantamento do estado da arte	15
3.1	Introdução	15
3.2	Plataformas de rastreabilidade	15
3.3	Plataformas de rastreabilidade no STV	16
3.4	Discussão	17
4	Análise e concepção da plataforma	19
4.1	Introdução	19
4.2	Modelo de Casos de Uso	19
4.3	Arquitetura	20
4.4	Modelo de Dados	22
4.5	API Integração	23
4.5.1	Sequelize ORM	23
4.5.2	Traduções	26
4.5.3	Integração com Fablo API	27
4.5.4	Sistema de Cache com Redis	29
4.6	Processamento e demonstração da rastreabilidade	30
4.6.1	Algoritmo de rastreabilidade	30
4.6.2	Demonstração de grafos	35
4.7	Aplicação Web	36
4.7.1	Perfil de Consumidor Final	38
4.7.2	Perfis de Responsável e Membro Organização	38
4.7.3	Perfil de Administrador	39
5	Avaliação e Resultados	40
5.1	Introdução	40
5.2	Testes Unitários	40
5.3	Casos de Teste de Integração	42
5.4	Discussão	44
6	Conclusão e Trabalho Futuro	48
6.1	Introdução	48
6.2	Conclusão	48
6.3	Trabalho Futuro	49
	Referências	50

Lista de Figuras

4.1	Diagrama de casos de uso	20
4.2	Arquitetura do sistema	21
4.3	Modelo de dados On-chain	22
4.4	Modelo de dados Off-chain	23
4.5	Diagrama de funcionamento Express com Sequelize ORM	24
4.6	Diagrama de fluxo chamadas API com redis	29
4.7	Diagrama de fluxo do processamento dos dados de rastreabilidade	30
4.8	Diagrama de fluxo da função recursiva de rastreabilidade	32
4.9	Diagrama Estilo Representação Rastreabilidade no Mapa	34
4.10	Diagrama Estilo Representação Rastreabilidade	35
4.11	Representação rastreabilidade no mapa geográfico	37
4.12	Representação rastreabilidade no mapa livre	37
4.13	Gestão de dados On-chain na plataforma	38
4.14	Gestão de dados Off-chain na plataforma	39
5.1	Diagrama de fluxo dos testes	40
5.2	Chamada aos testes unitários	42
5.3	Diagrama de exemplo criação T-shirt na blockchain	43
5.4	Exemplo da T-shirt com grafos no mapa geográfico	43
5.5	Exemplo da T-shirt com grafos simples	44
5.6	Exemplo da T-shirt com grafos simples com pesquisa por ID	44
5.7	Resposta API dados rastreabilidade sem cache	46
5.8	Resposta API dados rastreabilidade com cache	46

Lista de Tabelas

3.1	Sumario das plataformas de rastreabilidade no Setor Têxtil e Vestuário (STV) . . .	18
-----	--	----

Lista de Excerto de Código

4.1	Definição de tabela no Sequelize	24
4.2	Definição de migração no Sequelize	25
4.3	Definição de seeder no Sequelize	26
4.4	Inicialização do i18next	26
4.5	Chamada de tradução requisição HTTP	27
4.6	Estrutura chamada a Fablo API	27
4.7	Chamada POST à Fablo API	28
4.8	Exemplo de remoção da cache ao adicionar nova atividade	29
4.9	Estrutura resposta informação de rastreabilidade	31
4.10	Função de sequenciação da rastreabilidade	31
4.11	Função recursiva criação lotes	31
4.12	Função de processamento dos lotes	33
4.13	Função de criação de arcos	34
4.14	Exemplo de resposta à requisição de nodos	36
4.15	Exemplo de resposta do Fablo API	36
5.1	Testes unitários na rota criação de utilizadores	41
5.2	Testes unitários na rota de listagem de utilizadores	41

Abreviaturas e Símbolos

API Application Programming Interface

BDD Base de dados

CAPSI Conferência da Associação Portuguesa de Sistemas de Informação

DDL Data Definition Language

DML Data Manipulation Language

IoT Internet of Things

MBA Modelos Baseados em Agentes

MUI Material UI

ORM Object-Relational Mapping

P2P Peer-To-Peer

SGBD Sistema Gestor de Bases de Dados

STV Setor Têxtil e Vestuário

SSG Server Statically Generated

SSR Server Side Rendered

URN Uniform Resource Names

UUID Universally Unique Identifier

Capítulo 1

Introdução

Este documento é uma dissertação de Engenharia Informática desenvolvido para a obtenção do grau de mestre pela Escola Superior de Tecnologia e Gestão do Instituto Politécnico de Viana do Castelo. A pesquisa e desenvolvimento do trabalho estão diretamente ligados ao projeto de investigação realizado no âmbito do STVgoDigital onde se pretende criar um sistema de rastreabilidade para a cadeia de valor do **STV**.

Neste capítulo pretende-se fazer uma contextualização sobre o **STV** e o tema da rastreabilidade de produtos, delinear os objetivos da investigação e do projeto em questão e finalizar com a descrição da estrutura do presente documento.

O **STV** é um dos maiores setores globais nos dias de hoje e tem vindo a degradar-se ao longos dos tempos devido à aceleração com que este se tem de reinventar constantemente para atrair as multidões de consumidores cegos pelo consumismo e tendências de moda rápida. Isto tem fortes impactos ambientais e sociais especialmente nos países menos desenvolvidos e com políticas menos defensoras das classes mais baixas da sociedade.

Os têxteis são materiais essenciais para os seres humanos em diversas áreas como as construções, a decoração, e a indústria automóvel, no entanto a indústria do vestuário é a que mais consome, totalizando cerca de 60% do uso dos têxteis. Isto deve-se em grande parte ao aumento do consumo relativo às tendências da moda que se alteram cada vez mais rápido tendo gerado um aumento de 400% neste setor nas últimas duas décadas [Jia et al., 2020]. Isto é especialmente verdade nos países mais desenvolvidos onde o poder de compra é maior e as pessoas se podem permitir a alterar o seu guarda-roupa diversas vezes por ano, o que se traduziu num aumento médio significativo de 7kg para 13kg de roupa por pessoa nos últimos vinte anos [Shirvanimoghaddam et al., 2020].

Com base nestes dados é naturalmente verdade que a poluição ambiental gerada por este setor tem vindo a aumentar consideravelmente, gerando em torno de 1.2 mil milhões de toneladas de gases de efeito de estufa por ano, ultrapassando o consumo de navios e aviões combinados [Watson, 2018]. Para além disso, é necessário contabilizar os desperdícios que são criados e enviados para aterros a céu aberto que geram em torno de 6500kg de dióxido de carbono equivalente por tonelada [Espinoza Pérez et al., 2022] e incineradoras. Tais valores totalizam cerca de 87%

do fim de vida dos produtos que o consumidor não deseja utilizar e que poderiam ser reciclados [Espinoza Pérez et al., 2022].

Além dos desperdícios e da poluição causada tanto a nível de produção como de descarte dos bens, é também importante notar que os produtos são maioritariamente produzidos em países menos desenvolvidos, onde a mão de obra é mais barata, e de seguida enviados para os países mais desenvolvidos, percorrendo assim milhares de quilómetros até chegarem ao seu destino final. Estes valores são difíceis de determinar devido à complexa lista de matérias primas que são necessários, assim como dos percursos de transporte das mesmas, no entanto existe uma estimativa que indica que os produtos exportados da China rondam os 0.16kg de dióxido de carbono equivalente por quilograma de têxtil [Palacios-Mateo et al., 2021].

A contaminação do ar e dos oceanos por micro fibras é também considerado um dos grandes poluentes e causadores de graves problemas especialmente nos oceanos, onde se estima que se concentrem entre 15 e 50 biliões de pequenas partículas de plástico [Barrett et al., 2020].

As tendências de moda cada vez mais rápidas tem resultado num desenvolvimento insustentável das práticas de produção no **STV** de modo a manter margens de lucro bastante elevadas [McNeill and Moore, 2015], o que leva a impactos não só ambientais, mas também sociais especialmente na forma de exploração da mão de obra barata nos países menos desenvolvidos que tem sistemas regulatórios mais precários e menor consciência ambiental [Napier and Sanguinetti, 2018].

A exploração da mão de obra infantil é um dos maiores problemas sociais especialmente na Ásia, onde aproximadamente 108 milhões de crianças trabalham de forma precária na agricultura, muitas vezes dedicando-se à produção de algodão para a criação de tecidos [Organization, 2018].

1.1 Motivação

Como já foi mencionado, são grandes os impactos causados pelo **STV** e a tendência é aumentar devido as grandes marcas que querem, a qualquer custo, causar ao consumidor uma falsa necessidade de comprar novas roupas e tecidos, causando uma sobreprodução que gera enormes danos no ambiente e na sociedade em geral.

Por esse mesmo motivo, é necessário combater de algum modo esse estilo de negócio e com o tempo conseguir que este se tornar mais sustentável e amigo do ambiente e das pessoas. Para isso, é necessário recorrer à tecnologia para quantificar e analisar os impactos dessas indústrias de modo a ser possível tomar medidas no sentido de diminuir grandes quantidades de desperdícios e produção de gases de efeito de estufa, que causam grandes danos ao planeta.

Deste modo surgiu a ideia de criar um sistema capaz de rastrear todos os produtos, desde a origem até ao seu destino final, onde se registam não só as suas produções, mas também todos os indicadores de sustentabilidade ambiental e social. Estes dados são registados diretamente pelos parceiros do projeto, que através da plataforma, tem um sistema de fácil acesso e sem grandes barreiras de aprendizagem.

Para além da funcionalidade de Backoffice, a plataforma está especialmente desenhada para o consumidor final poder rastrear o impacto de determinada peça ou lote de tecido ou vestuário.

Aqui, através de um código identificador, os clientes têm acesso à informação correta e fidedigna proveniente de informações disponibilizada pelos parceiros e inseridas na blockchain, que armazena essa informação de forma imutável. Isso é uma grande vantagem porque ajuda os clientes a ter uma maior consciência ambiental e social e repensar antes de decidir comprar novos produtos, que antes não tinha qualquer informação quando ao seu grave impacto no ambiente e na sociedade. Para além da informação disponibilizada aos consumidores, a rastreabilidade destes produtos ajuda também as empresas a trocar informações sobre as suas produções e origens e criar uma confiança mais sólida entre os parceiros.

1.2 Objetivos

Como visto na secção anteriormente o **STV** tem sérios problemas ambientais e sociais que necessitam soluções urgentes para a melhoria da qualidade de vida de todos e do planeta. Ainda é difícil conseguir chegar a certas conclusões e medidas sobre o setor devido à falta de meios para contabilizar realmente os impactos causados por esta indústria que emprega cerca de 12% do mercado global [Alves, 2023] à qual é necessário colocar um travão quando aos danos que está a causar. Deste modo são necessárias ferramentas que em conjunto sejam capazes de rastrear dados provenientes de diversas fontes e setores para que consigamos começar a rastrear com mais certezas os reais impactos que os diversos intervenientes da indústria estão a causar.

Assim surgiu o projeto STVgoDigital, financiado pela FEDER, com o objetivo de rastrear todas as atividades produtivas e logísticas desde a criação da matéria-prima até chegar ao consumidor final. Sendo este um projeto financiado e dividido em várias fases, diversas componentes já foram concebidas e demonstradas em outras fases, como [Alves, 2023] e revisto também em [Alves et al., 2022a, Alves et al., 2022b], onde já se decidiu e se desenvolveu soluções adequadas ao projeto, nomeadamente a utilização de blockchain para armazenar os dados referentes à rastreabilidade dos produtos. Com esta tecnologia, que está atualmente em moda devido ao crescimento das criptomoedas, somos capazes de guardar informação de forma autêntica e imutável, essencial para prevenir qualquer tipo de manipulação de dados, devido ao facto de existirem diversos intervenientes no sistema tornando assim essencial um mecanismo que proteja este tipo de possibilidade de falsificação dos dados.

No entanto o armazenamento desses dados por si só não é muito relevante se não tiverem um propósito final, pelo qual é necessária uma plataforma para a gestão e visualização dos mesmos, tanto na vertente administrativa como para a demonstração de resultados para o consumidor final.

Assim sendo, este projeto consiste na interligação dos dados de rastreabilidade armazenados na blockchain e dados provenientes de bases de dados relacionais, numa só plataforma web com a capacidade de visualização da rastreabilidade de produtos através de um sistema de grafos onde o utilizador pode visualizar a origem do seu vestuário através de um código identificativo inserido na peça. Além da interface com o utilizador desenvolvida como plataforma web, são necessárias diversas componentes, explicadas no capítulo 4, para a interligação de todos os dados.

Com a pretensão de desenvolver um sistema o mais completo possível é necessário ter atenção aos seguintes pontos:

- Desenvolvimento de uma Application Programming Interface (**API**) capaz de fornecer informação a sistemas terceiros e a diversos sistemas próprios distintos (Aplicação Web);
- Criar uma interface que forneça, de uma forma amigável, toda a informação de rastreabilidade do produto;
- A plataforma deve fornecer toda a informação sobre todos os indicadores de sustentabilidade (ambiental e social) do produto.

1.3 Metodologia

Este trabalho de investigação vai ser desenvolvido seguindo a metodologia de investigação denominada como Design Science Research (DSR). DSR é uma metodologia recente e muito utilizada para desenvolver projetos de investigação académicos no ramo da engenharia, em especial na área das Tecnologias de Informação [**Cruz and da Cruz, 2020a**]. Esta metodologia de investigação requer a criação de um artefacto útil e inovador que estenda o estado atual numa determinada área. Neste caso, pretende-se criar uma plataforma de rastreabilidade do **STV**.

Esta metodologia tem 6 etapas principais:

- Identificação do problema e motivação - o problema e a sua relevância foi apresentado na secção **1.1**
- Definição dos objetivos - Os objetivos são apresentados na secção **1.2**. O levantamento do estado da arte, para reconhecer até que ponto estamos a estender o estado atual da rastreabilidade no **STV**, está desenvolvido no capítulo **3**
- Design e desenvolvimento do artefacto - Pretende-se criar uma plataforma de rastreabilidade dos indicadores de sustentabilidade ambiental e social dos produtos têxteis e vestuário. A arquitetura e o modelo de casos de uso da plataforma proposta é detalhada no capítulo **4**.
- Demonstração - a demonstração pode ser feita através do uso da plataforma criada e da verificação de que esta satisfaz os requisitos especificados no capítulo **5**.
- Avaliação - a plataforma criada vai ser testada a vários níveis através de testes unitários, de integração e de sistema demonstrados em **5.2**. Tenciona-se testar especialmente a performance de todo o sistema produzido. A usabilidade da plataforma vai ser avaliada através de um inquérito feito aos utilizadores desta.
- Comunicação - pretende-se publicar um artigo numa conferência internacional onde se apresenta a plataforma e os resultados obtidos.

1.4 Estrutura da Dissertação

Para além da introdução, esta dissertação contém mais 5 capítulos.

- No capítulo 2 são apresentados os conceitos e ferramentas utilizados no desenvolvimento do projeto;
- No capítulo 3 é feito o levantamento do estado da artes e são apresentados diversos trabalhos relacionados;
- No capítulo 4 faz-se a análise e conceção da plataforma, onde são apresentados os modelos de caso de uso, modelo de dados e a arquitetura da solução. Neste capítulo também se detalha a API de integração, o algoritmo de rastreabilidade e a aplicação web desenvolvida;
- No capítulo 5 é feita uma avaliação e discussão do trabalho desenvolvido. Aqui apresenta-se um resumo dos testes unitários desenvolvidos, e uma demonstração do funcionamento do sistema através de um exemplo prático da criação de um artigo de vestuário;
- No capítulo 6 é concluído todo o trabalho desenvolvido e dissertadas novas propostas de trabalho futuro.

No final do documento são ainda apresentadas todas as citações bibliográficas recolhidas durante a extensa pesquisa para o projeto.

Capítulo 2

Conceitos e ferramentas

2.1 Introdução

O capítulo de conceitos e ferramentas tem como função dar ao leitor um entendimento de todos os elementos em que a dissertação se baseia gerando uma ideia base do que vai ser apresentado nos capítulos seguintes. Para além dos conceitos mais abstratos, são apresentadas todas as tecnologias utilizadas, nomeadamente as linguagens de programação, frameworks e livrarias.

2.2 Conceitos

Em qualquer área do conhecimento, é importante ter um bom entendimento dos conceitos fundamentais para uma compreensão completa e eficiente dos temas abordados. Para desenvolver soluções inovadoras e eficientes em diferentes setores da economia, é necessário ter uma base sólida em conceitos e ferramentas utilizadas na área.

Este capítulo tem como objetivo apresentar os conceitos fundamentais relacionados com diferentes áreas, incluindo temas como rastreabilidade, blockchain, cadeia de valor, sustentabilidade, entre outros. Serão discutidos os conceitos e ferramentas utilizados para garantir a qualidade e a segurança dos produtos, promover a transparência e a colaboração entre os intervenientes do **STV**, e como a tecnologia pode ajudar a promover a sustentabilidade ambiental e social.

Com isso, pretende-se contribuir para uma compreensão mais ampla e profunda dos conceitos fundamentais em diferentes áreas, expondo os conceitos base para o desenvolvimento de soluções cada vez mais inovadoras e eficientes.

2.2.1 Cadeia de valor

A cadeia de valor é um conceito amplamente utilizado na indústria para descrever todas as atividades interligadas necessárias para levar um produto ou serviço desde a conceção até a sua disposição final após o uso. Isso inclui a produção, serviços intermediários, entrega ao consumidor final e disposição final [**KAPLINSKY, 2004**].

Essa compreensão pode ajudar as empresas a identificar oportunidades de melhoria de eficiência e inovação em toda a cadeia de valor. Além disso, a gestão eficaz da cadeia de valor pode levar a benefícios como redução de custos, aumento da qualidade do produto ou serviço, redução do tempo de produção e aumento da satisfação do cliente. A cadeia de valor é frequentemente dividida em atividades primárias e atividades de suporte, cada uma das quais tem um papel fundamental na criação de valor para o produto ou serviço.

Para além disso, a cadeia de valor é uma ferramenta essencial para as organizações que desejam entender e gerir os seus impactos ambientais e sociais, pois com estas ferramentas é facilmente identificável os processos que mais dano fazem ao ambiente e a sociedade.

Dessa forma, entender a cadeia de valor é fundamental para que as empresas possam adotar práticas mais sustentáveis e responsáveis, visando minimizar os impactos negativos e maximizar os impactos positivos ao longo de toda a cadeia.

2.2.2 Cadeia de valor no Setor Têxtil e do Vestuário

A cadeia de valor do **STV** é uma das mais complexas devido a grande quantidade de intervenientes que são necessários para produzir um único produtos e também pelas enormes distâncias que os vários elementos necessários para cada peça tem de percorrer até ao destino final o que envolve uma enorme logística e custos ambientais.

No **STV** o ciclo de produção começa maioritariamente do outro lado do mundo, em países como a Índia e a China que lideram a plantação de algodão, e também onde são produzidas vastas quantidades de poliéster que representa mais de 50% de todas as fibras utilizadas no mundo [Smelik, 2023]. Depois de produzidas as fibras estas são enviadas para fábricas de manufatura dos tecidos e linhas, que enviam as suas produções para empresas situadas em países com leis laborais precárias onde são executadas as tarefas de corte, costura e acabamento dos produtos. Só no final de todo este processo, e de largas distâncias percorridas, é que os produtos são enviados para os países mais desenvolvidos, onde vão ser vendidos, e por vezes feitos alguns acabamentos e controlos de qualidade.

Para que tudo isso seja possível a distribuição e logística são fundamentais numa cadeia de valor tão extensa, pois garantem que os produtos sejam entregues aos clientes no prazo e de forma eficiente [Porter's, 1985] levando as modas até ao consumidor final a um ritmo totalmente acelerado.

Além disso, existem atividades secundárias que desempenham um papel importante na cadeia de valor, como marketing e vendas. Todas essas atividades juntas formam a cadeia de valor completa da indústria têxtil e de vestuário [Porter's, 1985].

Devido aos grandes consumos por parte do **STV** a análise deste setor é importante para as empresas conseguirem identificar as atividades que mais danos estão a causar para tentar mitigá-los, para melhorar assim a sua produção e torna-la mais sustentável.

2.2.3 Rastreabilidade

Rastreabilidade é a capacidade de rastrear a história, localização e uso de um produto ou item, desde o seu ponto de origem até o destino final. Esta começa desde a origem do menor componente do produto, onde a informação é inicialmente registada e depois vai sendo detalhada ao longo da sua jornada pela cadeia de valor, registando todos os eventos relevantes e/ou impactantes no produto, tais como mudança de localização, manipulações, controlos, testes, entre outros.

Na indústria, a rastreabilidade é usada para monitorizar e controlar a cadeia de valor e abastecimento, permitindo os intervenientes possam garantir que os produtos sejam fabricados de acordo com as normas de qualidade e segurança.

A rastreabilidade é especialmente importante em setores altamente regulamentados, como o setor de alimentos e medicamentos, onde a informação sobre a origem, qualidade e segurança dos produtos é essencial, e exigida por vários governos que requerem informações detalhadas sobre bens a serem comercializados. A rastreabilidade permite que os fabricantes possam garantir a segurança dos produtos e cumprir as normas regulatórias, além de proteger a saúde e a segurança dos consumidores [Alves. et al., 2021].

Para além disso, a rastreabilidade pode ajudar as empresas a identificar e corrigir problemas rapidamente, permitindo a redução dos impactos negativos dos mesmos sobre si mesmas e dos consumidores, tanto a nível monetário como social e ambiental.

A tecnologia tem sido um importante fator para melhorar a rastreabilidade na cadeia de valor, permitindo que as empresas possam coletar, armazenar, analisar e compartilhar informações de forma mais eficiente e segura [da Cruz and Cruz, 2020]. Com a crescente digitalização da cadeia de valor, surgiram várias soluções tecnológicas, incluindo sistemas de rastreabilidade por RFID, códigos de barras, sensores IoT, blockchain, entre outros .

Essas tecnologias têm permitido que as empresas possam recolher dados em tempo real e compartilhá-los com os parceiros de forma segura e confiável [Yu et al., 2020]. Além disso, as tecnologias de análise de dados permitem que as empresas possam identificar rapidamente problemas e oportunidades de melhoria na cadeia de valor, permitindo que elas possam agir de forma mais rápida e eficaz para melhorar a eficiência e a qualidade dos produtos.

Por exemplo, o uso de códigos de barras e sensores IoT pode ajudar a rastrear a localização e o status de um produto em tempo real, permitindo às empresas controlar a qualidade e a segurança dos produtos durante todo o processo de transporte e armazenamento. Já o uso de blockchain pode ajudar a garantir a segurança e a integridade das informações, permitindo que as empresas possam compartilhar informações com os parceiros de forma confiável e transparente.

2.2.4 Sustentabilidade

A sustentabilidade é um conceito que se preocupa com a capacidade de satisfazer as necessidades presentes sem comprometer as gerações futuras . Num mundo cada vez mais globalizado e interconectado, a sustentabilidade torna-se uma preocupação importante em todas as áreas da sociedade, incluindo nos negócios e na tecnologia [Torresi et al., 2010].

Em termos ambientais, a sustentabilidade visa minimizar o impacto negativo das atividades humanas no meio ambiente. Isso pode incluir a redução do consumo de energia e matérias-primas, a utilização de tecnologias mais limpas e a promoção de práticas de conservação e preservação do meio ambiente [AQUINO et al., 2016].

Em termos sociais, a sustentabilidade preocupa-se com as relações humanas e a justiça social. Isso pode incluir a promoção de condições de trabalho justas e seguras, o respeito aos direitos humanos e a melhoria das condições de vida das comunidades locais afetadas pelas atividades das empresas [Gomes, 2005].

A tecnologia pode ser uma ferramenta importante para promover a sustentabilidade. Através de soluções tecnológicas inovadoras, é possível promover a eficiência energética, o uso sustentável de recursos naturais [Serra and de Moraes, 2007] e a promoção de práticas mais justas e transparentes.

Em resumo, a sustentabilidade é uma preocupação importante em todas as áreas da sociedade, incluindo nos negócios e na tecnologia. Através da promoção de práticas mais sustentáveis, é possível garantir um futuro mais justo e equitativo para as gerações presentes e futuras.

2.2.5 Blockchain

Blockchain é uma tecnologia de registo distribuído que permite a criação de uma Base de dados (BDD) compartilhada, segura e transparente, que pode ser utilizada para rastrear a história de um produto, por exemplo, ao longo de toda a cadeia de valor do setor têxtil e do vestuário. Através da criação de uma rede descentralizada de computadores que validam e registam transações de forma consensual e permanente, formando blocos de informações interconectados, a tecnologia blockchain torna possível garantir a autenticidade e a integridade das informações relativas à produção, transporte e distribuição de um produto [Queiroz et al., 2020].

Com o uso da tecnologia blockchain, é possível aumentar a transparência e a confiança na cadeia de valor, permitindo que consumidores e outras partes interessadas acessem informações detalhadas sobre a origem dos materiais, os processos de produção e a logística envolvida no transporte e distribuição de um produto. Além disso, a tecnologia blockchain pode ajudar a combater a pirataria e a falsificação de produtos, garantindo que apenas produtos autênticos e certificados sejam comercializados [Rusinek and Zhang, 2018].

Em resumo, a tecnologia blockchain tem um grande potencial para melhorar a rastreabilidade na cadeia de valor do setor têxtil e do vestuário, garantindo a autenticidade e a transparência das informações ao longo de toda a cadeia, desde a matéria-prima até o produto final.

2.3 Ferramentas

2.3.1 JavaScript

A linguagem JavaScript, desenvolvida em meados da década de 1990 pela NetScape, surgiu com o objetivo de possibilitar a criação de scripts para os navegadores web. Com a grande versa-

tilidade oferecida, ganhou imensa popularidade e é hoje uma das linguagens mais utilizadas para o desenvolvimento de todo o tipo de soluções, estando os seus interpretadores em todo tipo de dispositivos, desde computadores até consolas de jogos [Flanagan, 2004].

2.3.2 TypeScript

O TypeScript é uma extensão do JavaScript desenvolvida com a pretensão de tornar o desenvolvimento de aplicações JavaScript em grande escala mais fácil. Embora todo programa TypeScript seja um programa JavaScript, o TypeScript oferece recursos adicionais, como um sistema de módulos, classes, interfaces e um sistema de tipos abrangente. A ideia do TypeScript é facilitar a transição dos programadores de JavaScript para esta nova linguagem, dando-lhes acesso a boas práticas e controlo de erros em tempo real durante do desenvolvimento das aplicações, evitando erros futuros por lapsos na hora de programar [Bierman et al., 2014].

2.3.3 GIT

O GIT é um sistema de controle de versão distribuído, amplamente utilizado para monitorizar as alterações em arquivos ao longo do tempo. Este foi projetado para lidar com projetos de qualquer dimensão, permitindo que vários intervenientes trabalhem no mesmo projeto de forma colaborativa, adicionando e gerindo todas as alterações feitas nos arquivos. Ele permite que sejam rastreadas as alterações, dando a possibilidade de reverter para versões anteriores, trabalhar em diferentes ramificações do projeto e juntar as versões feitas por diversas pessoas.

2.3.4 PL/pgSQL

A PL/pgSQL é uma linguagem imperativa utilizada no SGBD do PostgreSQL que executa comandos SQL através de uma linguagem de instruções imperativas e uso de funções, o que proporciona um maior controlo automático sobre o processamento de dados em comparação com as simples instruções SQL. A partir do PL/pgSQL é possível realizar cálculos complexos e criar novos tipos de dados de utilizador. Para além disso possui estruturas de controlo de repetição e condicionais, e também permite criar funções que podem ser chamadas em instruções SQL normais ou executadas em eventos de tipo trigger [CHÁVEZ, 2019].

2.3.5 JSON

O JavaScript Object Notation (JSON) é um formato leve baseado nos tipos de dados do JavaScript. Os documentos JSON são essencialmente dicionários compostos de pares chave-valor, onde o valor pode ser um documento JSON, permitindo um nível arbitrário de camadas. Além de dicionários simples, o JSON também suporta matrizes e tipos atômicos, como números e strings. Devido à sua simplicidade e facilidade de leitura por seres humanos e máquinas, o JSON está a tornar-se um dos formatos mais populares no intercâmbio de troca de dados na web, especialmente em API para enviar solicitações e respostas sobre o protocolo HTTP. Além disso, o formato JSON

é muito usado em SGBD construídos em torno do paradigma NoSQL ou em bases de dados de gráficos [Bourhis et al., 2020].

2.3.6 UUID

Uma Universally Unique Identifier (**UUID**) é um identificador único de tamanho fixo (128 bits) em relação ao espaço e ao tempo, podendo ser usada para diversas finalidades, desde a marcação de objetos com uma vida extremamente curta até a identificação confiável de objetos persistentes numa rede. Tendo esta particularidade são excelentes para serem utilizadas como Uniform Resource Names (**URN**) já que não tem a necessidade de um processo de registo permitindo um dos custos de criação mais baixos [Leach et al., 2005].

2.3.7 Bcrypt

Bcrypt é uma função de derivação de chave para senhas, baseada no cifra Blowfish. Além de incorporar um *salt* para proteger contra ataques de *rainbow table*, o BCRYPT é uma função adaptável: ao longo do tempo, a contagem de iterações pode ser aumentada para torná-la mais lenta, mantendo sua resistência a ataques de força bruta, mesmo com o aumento do poder de computação [Sriramya and Karthika, 2015].

2.3.8 React

React JS é uma biblioteca JavaScript de código aberto para construção de interfaces de utilizador. Foi criado pelo Facebook e é usado para desenvolver aplicações web de alta performance com uma experiência de utilizador dinâmica e responsiva. O React JS usa uma abordagem baseada em componentes, onde cada componente representa uma parte da interface do utilizador. Esses componentes podem ser reutilizados em diferentes partes das aplicações, o que ajuda a manter o código organizado e fácil de manter. Além disso utiliza o Virtual DOM, que é uma representação em memória do DOM real. Quando ocorre uma alteração no estado da aplicação, este compara a versão anterior do Virtual DOM com a nova e atualiza apenas os elementos que precisam ser alterados no DOM real, tornando-o mais eficiente e rápido do que outras bibliotecas.

2.3.9 Nextjs

O Next.js é um framework de desenvolvimento web de código aberto baseado em React que permite a criação de aplicações web Server Side Rendered (**SSR**) e/ou Server Statically Generated (**SSG**).

O **SSR** significa que as páginas são geradas no servidor e enviadas ao navegador como HTML completo. Com o **SSR**, o conteúdo da página é renderizado no servidor, o que pode melhorar o tempo de carregamento inicial e a indexação pelos motores de pesquisa.

Já o **SSG** é uma técnica que permite gerar páginas estáticas em tempo de construção, em vez de gerá-las sob demanda. Com o **SSG**, é possível criar aplicações mais rápidas e escaláveis, com a vantagem de ter um melhor desempenho em termos de SEO, segurança e economia de recursos.

2.3.10 Material UI

Material UI (**MUI**) é um framework CSS e um conjunto de Componentes React que implementa o design material do Google. Essa biblioteca oferece suporte a componentes como botões, diálogos, menus suspensos, botões de ícone, inputs, switches, barras de ferramentas e outros componentes de design material [Mahmud, 2020].

2.3.11 Axios

O Axios é uma biblioteca JavaScript baseada em promessas (*Promises*) que é usada para fazer solicitações HTTP, tanto do lado do cliente quanto do servidor. Ele é compatível com navegadores e Node.js e permite enviar solicitações HTTP com métodos como GET, POST, PUT, DELETE e outros. Além disso, o Axios suporta a inclusão de dados de cabeçalho (headers), o cancelamento de solicitações e a interceptação de solicitações e respostas para fins de autenticação, manipulação de erros e outras funcionalidades.

2.3.12 NodeJs

O Node.js é uma plataforma de desenvolvimento de software em JavaScript, desenvolvido para funcionar como ambiente de execução no lado do servidor, sendo amplamente utilizado para construir aplicações web, **APIs**, ferramentas de linha de comando e até mesmo aplicações *desktop*. Por possuir uma grande comunidade de programadores e bibliotecas de código aberto disponíveis, torna o desenvolvimento rápido e eficiente, com diversas integrações com outras tecnologias, como bases de dados, frameworks e ferramentas de automatização de tarefas.

2.3.13 Express

É um *framework* amplamente usado para criar aplicações web em Node.js. Ele fornece uma ampla gama de recursos, incluindo gestão de rotas, *middlewares*, manipulação de solicitações e respostas, suporte para vários tipos de dados, gestão de sessões entre outros. Para além de todas essas funcionalidades, é flexível para adição de novas ferramentas de terceiros como complemento.

2.3.14 Sequelize

O Sequelize é uma biblioteca Object-Relational Mapping (**ORM**) para Node.js que fornece uma maneira fácil de acessar bases de dados relacionais com JavaScript. A biblioteca suporta vários Sistema Gestor de Bases de Dados (**SGBD**), incluindo MySQL, PostgreSQL, SQLite e Microsoft SQL Server, e fornece uma **API** simples e intuitiva para criar, consultar e manipular dados em suas bases de dados relacionais. Os modelos de dados em JavaScript que representam

tabelas em sua **BDD** podem ser usados para executar consultas SQL, criar, atualizar ou excluir registros de **BDD** e muito mais.

2.3.15 Jest

Jest é um *framework* de testes em JavaScript especialmente projetado para testes unitários, mas também pode ser usado para testes de integração e testes de ponta a ponta. Ele fornece uma estrutura robusta e flexível para escrever, executar e analisar testes em JavaScript.

Uma das principais características do Jest é a sua capacidade de executar testes de forma assíncrona. Ele suporta *Promises*, funções assíncronas e *callbacks* para testes assíncronos e inclui recursos como temporizadores falsos para ajudar na escrita e execução de testes.

2.3.16 Supertest

Supertest é uma biblioteca Node.js usada para testar APIs HTTP de forma fácil e eficiente. Ele permite enviar solicitações HTTP simuladas numa aplicação Node.js sem precisar realmente iniciar o servidor. Isso possibilita testar as rotas da aplicação sem precisar executar um servidor real ou uma instância de banco de dados.

2.3.17 i18next

O i18next é uma biblioteca JavaScript para internacionalização (i18n), fornecendo suporte para traduções de texto em aplicações da web. Este usa uma abordagem de "chave-valor" para gerir as traduções, onde as chaves são usadas para identificar as frases ou palavras que precisam ser traduzidas e os valores das traduções em diferentes idiomas. Essas chaves e valores são armazenados em arquivos JSON ou YAML e são organizados em *namespaces* para simplificar a organização.

2.3.18 Express-session

Express-session é uma biblioteca de *middleware* para Express.js que fornece suporte para sessões do lado do servidor, permitindo armazenar dados de sessão do utilizador, como *cookies*, no servidor e gerir o estado dessa sessão durante as solicitações subsequentes.

2.3.19 Express-validator

O Express-validator é uma biblioteca de *middleware* para Express.js que fornece validação de entrada para formulários e solicitações de utilizadores. Este permite a validação para campos de formulário, retornando os erros como resposta em caso de falho. Fornece ainda recursos de segurança para evitar ataques de injeção de código malicioso.

2.3.20 Leaflet

O Leaflet é uma biblioteca JavaScript de código aberto amplamente utilizada para criar mapas interativos e responsivos em aplicações web. Ele foi projetado para ser fácil de usar e personalizável, permitindo que programadores e designers criem mapas com recursos avançados, como marcadores, *pop-ups*, camadas, rotas, entre outros. Além disso, possui uma ampla variedade de *plugins* e extensões disponíveis para personalização adicional.

2.3.21 Redis

O Redis é um sistema de armazenamento de dados em memória de alta performance. Ele é usado para armazenar, gerir e manipular informações, sendo amplamente utilizado em aplicações onde o desempenho e a escalabilidade são essenciais. Este permite armazenar diferentes tipos de dados, como strings, listas, conjuntos, *hashes* e muito mais. Além disso, o Redis oferece uma ampla gama de operações para manipular esses dados, como adicionar, remover, atualizar e consultar informações.

Capítulo 3

Levantamento do estado da arte

3.1 Introdução

A presente secção tem como objetivo apresentar uma revisão abrangente de projetos e soluções de plataformas de rastreabilidade existentes, tanto em contextos gerais como específicos do setor têxtil e do vestuário. Esta revisão destaca as melhores práticas e as soluções inovadoras que foram desenvolvidas para a implementação de sistemas de rastreabilidade, com especial ênfase em tecnologias e estratégias para a recolha, processamento e análise de dados, bem como as diferentes abordagens de integração de informações ao longo da cadeia de valor. Através desta revisão, será possível entender melhor as oportunidades e desafios que surgem na implementação de uma plataforma de rastreabilidade para o setor têxtil e do vestuário.

3.2 Plataformas de rastreabilidade

Em [Moreira, 2022] é apresentada uma plataforma para a rastreabilidade de aves caipira, que tem um valor comercial superior face as aves de origem industrial, e que carecem de sistemas com transparência e segurança para a sua comercialização, que atualmente são feitos por aplicações de comunicação, onde acontecem por vezes negócios fraudulentos, onde os animais carecem da qualidade vendida.

Em [Bendo, 2021] é descrito o desenvolvimento de uma plataforma para a rastreabilidade de Modelos Baseados em Agentes (MBA), que visam o estudo da simulação e complexidade dos indivíduos, importantes para o estudo das áreas de Geografia, sociologia e economia. Nesta pretendem criar uma plataforma capaz de transformar e categorizar cada uma das características dos MBA.

Em [Silva et al., 2019] é feita uma avaliação do uso de blockchain, com a rede Ethereum, na criação de uma plataforma de rastreabilidade para a industria agroalimentar e socioenergética. Aqui pretende-se rastrear os produtos através do identificador único do lote com a informação de qualidade e processamento da matéria prima, que são gravados na rede da blockchain.

A rastreabilidade é especialmente útil e está especialmente desenvolvida, na indústria alimentar, quer para garantir a origem dos produtos [Alves. et al., 2021], quer por razões de segurança de saúde pública, como é o caso das plataformas apresentadas em [Oliveira et al., 2021, Cruz and da Cruz, 2020b] criadas para rastrear peixe e seus derivados, da plataforma apresentadas em [Reffatti et al., 2021] que implementa a rastreabilidade das frutas e legumes, para implementar a rastreabilidade do azeite foi proposta uma plataforma em [Fernandes et al., 2022], em [Biswas et al., 2017] propõem-se rastrear o vinho, entre outras.

Existem também propostas de criação de plataformas que permitem rastrear contactos políticos [Dias et al., 2021], pegada de carbono de um produto [da Cruz et al., 2020], sistemas de donativos [Singh et al., 2020], produtos farmacêuticos [Panda and Satapathy, 2021], entre outros.

Em [Trigo et al., 2018] é apresentado um projeto de rastreabilidade de rebanhos de gado que diz economizar tempo e dar um maior controlo do negócio aos produtores. Neste é apresentada uma aplicação móvel, já disponível no mercado, para registar animais e rebanhos e posteriormente gerar diversos relatórios dos ciclos de produção.

Em [Lourenço, 2021] foi proposta e realizada uma plataforma de rastreabilidade para a cadeia de valor agroalimentar. Nesta foi desenvolvida um solução com recurso à blockchain para rastrear todo o processo produtivo dos produtos agroalimentares e dar ao consumidor final a possibilidade de verificar a sua rastreabilidade através de uma aplicação móvel.

3.3 Plataformas de rastreabilidade no STV

Nesta secção, é abordado o tema da rastreabilidade com especial ênfase a plataformas e estudos relacionados ao STV, de forma a ajudar a compreender melhor o que já existe em que ponto se pode contribuir com novas ideias.

No âmbito do STV, existem vários estudos e existem já várias plataformas de rastreabilidade. Algumas das quais são apresentadas em seguida.

Em [Rinaldi et al., 2022] os autores apresentam um relatório sobre como melhorar a rastreabilidade da STV e concluem que é preciso criar normas, definir padrões e formas de medição e a adoção de novas tecnologias que permitam a rastreabilidade [Rinaldi et al., 2022]. Os autores concluem também que para alcançar a transparência na cadeia de valor, é necessária a participação de todos, desde governos, consumidores e todas as empresas envolvidas nas cadeias de valor.

Em [Fu et al., 2018] os autores propõem um sistema de comércio de emissões baseado em blockchain para produtos da indústria de moda. O sistema proposto é capaz de mensurar o impacto ambiental de determinada peça de vestuário e sugere soluções para que os operadores da cadeia de valor compensem as emissões de carbono daquele produto.

Em [Yu et al., 2020], os autores propõem a criação de um sistema Peer-To-Peer (P2P) de forma a promover a economia partilhada no desenvolvimento de tecidos e vestuário, recorrendo ao uso da tecnologia blockchain. É referido que este tipo de sistemas ainda estão num estágio muito inicial e que precisam ser bem definidos no futuro, como forma potenciadora para o mundo colaborativo. No artigo é proposta uma hierarquia de serviço que inclui sistemas de produto-serviço,

configuração-serviço e recurso-serviço e é apresentado um estudo de caso que demonstra um melhor desempenho da manufatura partilhada sobre a produção em nuvem e a produção social.

Desenvolvido no âmbito do projeto STVgoDigital, onde este projeto também se insere, pretende avaliar o uso de blockchain para rastrear o Setor Têxtil e do Vestuário e estudar diversas tecnologias para o desenvolvimento de sistemas que promovam a economia circular e assim diminuir o impacto ambiental produzido por esta cadeia de valor tão complexa e fortemente poluidora. [Alves et al., 2022a]. No âmbito do mesmo projeto, foi também apresentada uma abordagem de rastreabilidade de indicadores de sustentabilidade recorrendo ao uso da tecnologia blockchain em [Alves et al., 2022b].

Nos dias atuais, a tecnologia blockchain tem sido usada em muitas das abordagens propostas para implementar rastreabilidade porque, teoricamente, satisfaz todas as necessidades da implementação de rastreabilidade [da Cruz and Cruz, 2020]. Desse modo abaixo podemos ver alguns exemplos do uso da mesma no setor da STV.

Em [ElMessiry and ElMessiry, 2018] é proposta uma solução blockchain para a melhoria da qualidade dos têxteis, quase em tempo real, onde as empresas identificadas podem adicionar informações sobre lotes defeituosos para serem identificados em todos os sistemas quanto antes.

No artigo [Rusinek and Zhang, 2018] os autores estudam uma solução para aumentar a rastreabilidade da cadeia de produção e em consequência uma melhoria da sustentabilidade da mesma. É ainda proposta a participação por parte do consumidor final para que exista uma economia circular e o produto não chegue ao fim de vida só porque o utilizador já não o pretende utilizar.

Em [Faridi et al., 2021] é proposto um sistema com recurso à blockchain e Internet of Things (IoT) para rastrear materiais utilizados no STV e dessa forma ajudar todas as partes interessadas a rastrear, melhorar e verificar a qualidade dos produtos ao longo da cadeia.

Em [Alves et al., 2023] os autores propõe a criação de uma aplicação descentralizada com gamificação para os consumidores tomarem melhores decisões quanto ao futuro do seu vestuário. Nela os produtos são guardados como gémeos digitais, em que cada artigo da plataforma corresponde a um artigo real, e que podem ser transacionados entre utilizadores, criando uma oportunidade de estender a vida útil dos produtos de vestuário. Certas atividades como a venda/aluguer de peças de vestuário dão recompensas e trunfeis digitais aos consumidores, o que se ajuda a criar bons comportamentos de consumo para essas pessoas. O artigo em questão usa dados provenientes da rede blockchain usada por este projeto, sendo uma extensão do mesmo.

3.4 Discussão

A rastreabilidade é um tema que tem vindo a ser debatido ao longo das últimas décadas mas ainda tem um longo caminho a percorrer em certos aspetos como a tecnologias, o consenso entre intervenientes e o tipo de informação a ser retirada. Apesar das grandes vantagens da rastreabilidade para a sociedade como um todo, ainda existe muita relutância por parte de empresas para adotá-la de forma pública, seja por questões de proteção da concorrência, segredos de negócio, ou até mesmo para ocultar falhas ou incumprimentos éticos ou legais. A questão da rastreabilidade é

por vezes muito complexa quando existem uma quantidade enorme de variáveis de informação a armazenar e como se devem processar para que essa informação tenha algum valor em termos de métricas. Para além destas questões, a escolha das tecnologias é fulcral pois esta pode decidir entre o sucesso e o insucesso de um projeto. O armazenamento dos dados é a variável mais importante e é também essa que mais consegui analisar no estudo efetuado previamente.

Na tabela 3.1 abaixo mostrada, são analisadas as referências bibliográficas das soluções de rastreabilidade no STV onde analisei o tipo de armazenamento, o uso de dispositivos IoT, as tecnologias utilizadas e a área de aplicação. Concluí que a grande maioria das propostas recorrem ao uso de blockchain para armazenar os dados, com algumas exceções que também são complementadas com bases de dados relacionais. Cerca de metade usam dispositivos IoT para recolher dados e armazena-los de alguma forma. Não são referenciadas muitas tecnologias interligadas com os projetos, falando maioritariamente todos sobre a parte do armazenamento de dados. Por fim, as áreas de aplicação encontram-se em todos os setores do STV de uma forma mais ou menos repartida.

Tabela 3.1: Sumario das plataformas de rastreabilidade no STV

Artigo	Armazenamento	Usa IoT	Outras Tecnologias	Área aplicação
[Rinaldi et al., 2022]	N/D	Não	N/D	Calçado
[Fu et al., 2018]	Blockchain	Não	N/D	Vestuário
[Yu et al., 2020]	Blockchain	Sim		Têxtil e Vestuário
[Alves et al., 2022a]	Blockchain	Sim	ERP e Cloud Computing	Têxtil e Vestuário
[Alves et al., 2022b]	Blockchain + BD Relacional	Sim	Aplicação Web, API	Têxtil e Vestuário
[Rusinek and Zhang, 2018]	Blockchain	Não	N/D	Têxtil
[ElMessiry and ElMessiry, 2018]	Blockchain + BD Relacional	Não	N/D	Vestuário
[Faridi et al., 2021]	Blockchain	Sim	N/D	Têxtil
[Alves et al., 2023]	Blockchain	Não	Gamificação, dApp	Vestuário

Capítulo 4

Análise e conceção da plataforma

4.1 Introdução

Este capítulo refere-se a todo o desenvolvimento da plataforma, incluindo todas as suas componentes, e também à análise de requisitos previamente feita. Após um estudo completo sobre projetos desenvolvidos na área da rastreabilidade, em especial no **STV** surgiu a necessidade de criar alguns modelos e a arquitetura de toda a solução antes de partir para o desenvolvimento dos programas.

Essencialmente foram feitos os modelos de casos de uso, modelos de dados e arquitetura do sistema que serviram como base, e elementos delimitadores do que a solução deveria ser. Após ter toda a estrutura planeada iniciou-se o desenvolvimento do servidor backend, onde foram feitas todas as conexões entre fontes de dados, e por fim feita a interface com o utilizador por meio de uma aplicação Web.

Nas próximas secções são detalhados todos os aspetos importantes do que foi desenvolvido nesta plataforma, desenvolvida especialmente para mostrar dados de rastreabilidade e sustentabilidade ambiental através de grafos orientados, provenientes das produções de empresas parceiras ao projeto, as quais fornecem as suas produções. Essas produções são armazenadas na rede blockchain após sofrerem diversos cálculos sobre os prejuízos ambientais e sociais que são posteriormente publicamente apresentados ao consumidor final. A plataforma tem essencialmente duas vertentes: demonstração de dados ao consumidor e *backoffice* para os parceiros do projeto.

4.2 Modelo de Casos de Uso

A aplicação foi desenhada para ser utilizada por diferentes tipos de utilizador, nomeadamente o Administrador, o Membro da Organização, o Responsável da Organização e o Consumidor Final.

O Administrador é o perfil com permissões máximas na plataforma e sobre os dados off-chain do projeto, desempenhando as seguintes tarefas: gerir as organizações e os seus membros, gerir as formulas de calculo dos indicadores de sustentabilidade, gerir outros dados da plataforma

armazenados off-chain e também tem a possibilidade de ver os dados públicos de rastreabilidade e sustentabilidade.

O Membro da Organização, que tem acesso ao backoffice, pode criar atividades de todos os tipos para a sua organização, visualizar os dados da mesma e ver os dados de rastreabilidade e sustentabilidade.

O Responsável da Organização herda as funções do membro da organização, com um extra de estar habilitado a gerir e criar novos utilizadores.

Por fim, o consumidor final, apenas pode visualizar os dados públicos disponibilizados na plataforma em forma de grafos simples e sobre um mapa geográfico, num panorama geral ou para um só lote, identificado através do seu identificador único.

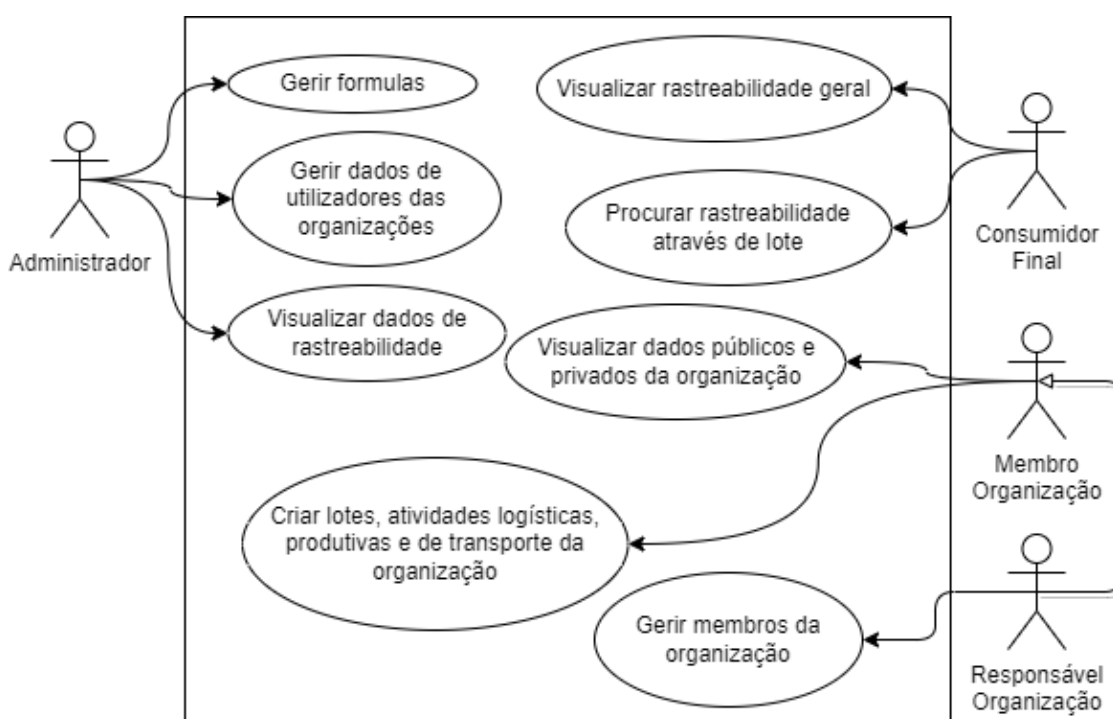


Figura 4.1: Diagrama de casos de uso

4.3 Arquitetura

A arquitetura do sistema é demonstrada na figura 4.2 onde são representados todos os componentes do sistema desenvolvido no âmbito do projeto já referido anteriormente.

Apesar a representação total, apresentada para dar uma ideia geral da solução, a parte demonstrada nesta dissertação não inclui do desenvolvimento da camada de blockchain, mas sim o uso da mesma, através de requisições à "Fablo API".

A camada de integração é a camada constituída por uma API desenvolvida em Node e Express com recurso à linguagem Typescript, é a responsável por recolher e inserir informação na rede blockchain e na BDD PostgreSQL, e fornecer através de endpoints informação para o exterior,

nomeadamente para a plataforma Web. Nessa camada são chamados diversos procedimentos que se responsabilizam por interligar os dados provenientes de ambas as bases de dados, calculando automaticamente, por exemplo, os indicadores de sustentabilidade de cada lote no momento da sua criação.

Por fim, a camada superior representa a interface com o utilizador, desenvolvida em React e NextJs, que é utilizada como *backoffice* para os parceiros do projeto e para a demonstração dos dados em forma de grafos representados num mapa para o utilizador final. Os diversos atores existentes na plataforma estão detalhadamente explicados na figura 4.1. Como o *backoffice* é uma componente que requer autenticação, foi desenvolvida uma autenticação com por email e palavra-passe que guarda uma sessão no servidor de backend e é usada para requisições posteriores. Para a interação com dados de leitura e escrita provenientes da BDD relacional esta medida é suficiente, no entanto, para ser possível adicionar dados à blockchain é necessário entrar com um operador/administrador da empresa parceira, sendo necessária uma autenticação extra, que é feita através de um módulo à parte que chama a API diretamente sem guardar informação sensível na BDD SQL.

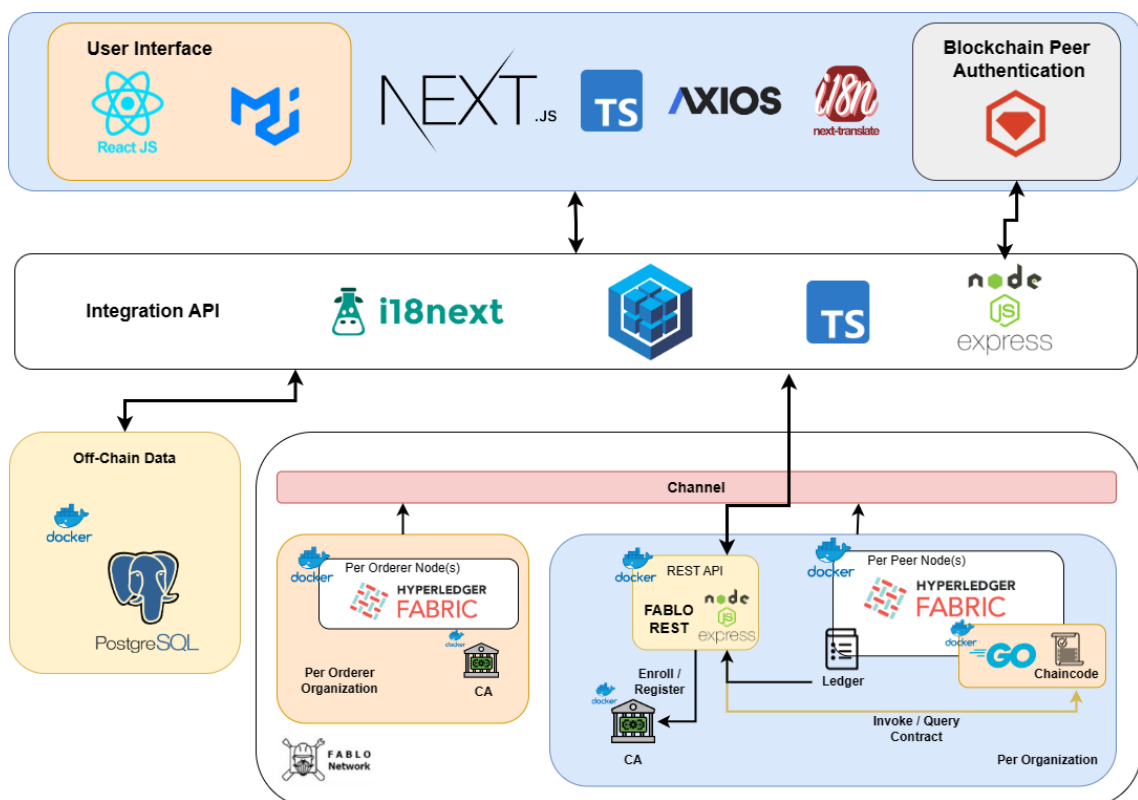


Figura 4.2: Arquitetura do sistema

4.4 Modelo de Dados

Nesta secção são demonstrados os modelos de dados detalhados de ambas as fontes de dados apresentadas. Os dados denominados de *on-chain* são os dados armazenados na blockchain, que estão representados na figura 4.3, e os dados *off-chain* estão armazenados na BDD relacional e representados na figura 4.4. Estes modelos foram desenhados em fases anteriores do projeto por vários centros de investigação pelo que só se desenvolveu nesta solução a parte off-chain da representação.

Apesar da representação estar feita através de modelos entidade-relacionamento as entidades descritas na figura 4.3 pertencem ao armazenamento na blockchain. Aqui são armazenadas as atividades de logística e produção, assim como os lotes. Além desses, são guardados os tipos de atividade, transporte e lote, e também a composição dos lotes.

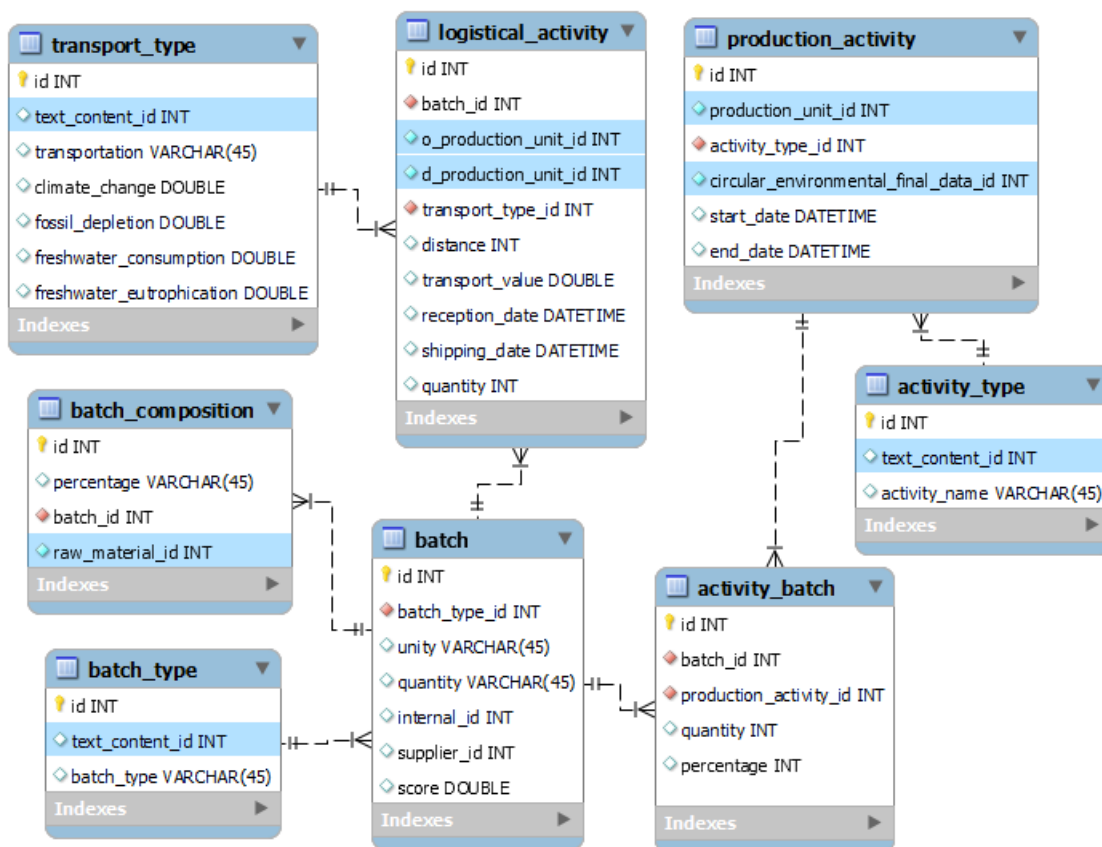


Figura 4.3: Modelo de dados On-chain

No armazenamento off-chain, representados na figura 4.4 são adicionados todos os dados necessários ao processamento dos indicadores de sustentabilidade e às empresas parceiras. Neste são armazenados na BDD PostgreSQL os dados de certificações, unidades de produção, indicadores socioeconómicos e de circularidade e diversas formulas de cálculo para os mesmos.

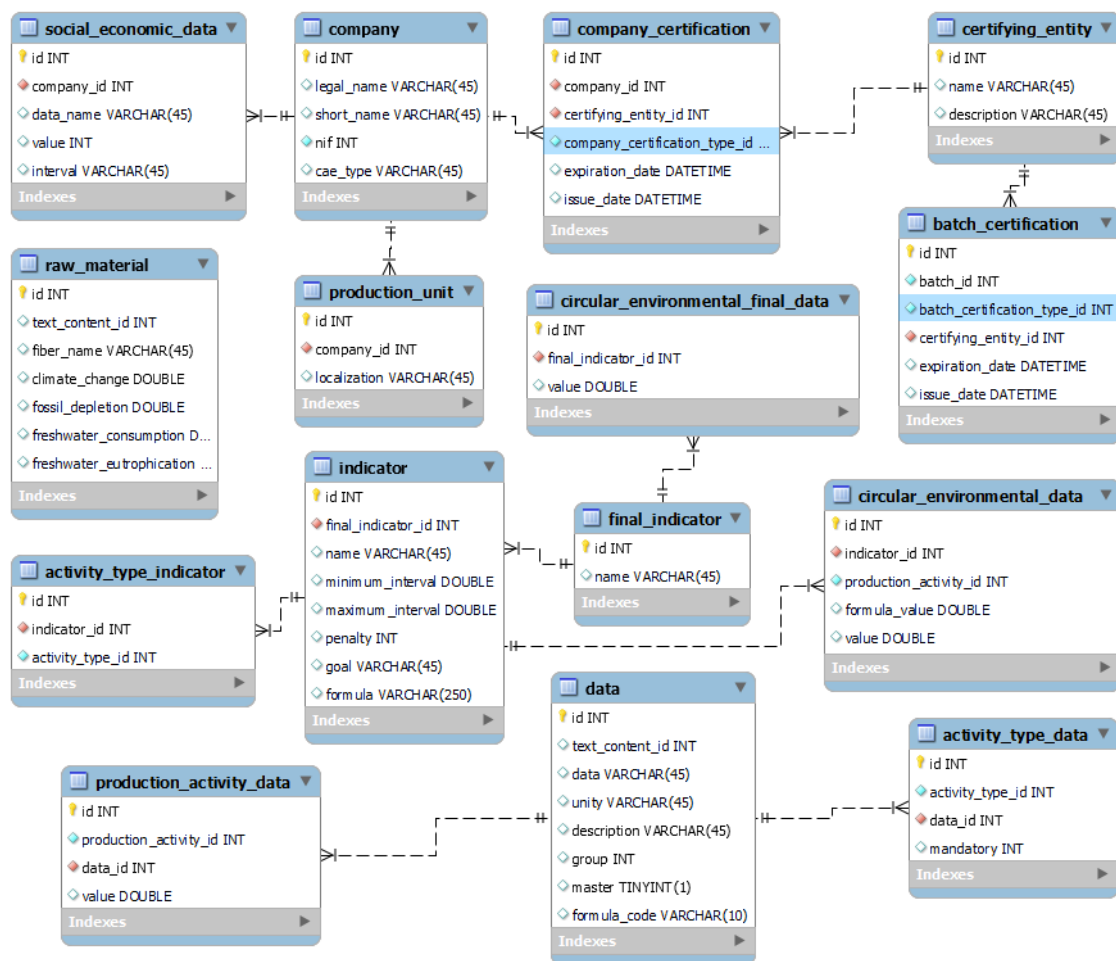


Figura 4.4: Modelo de dados Off-chain

4.5 API Integração

A **API** desenvolvida com Nodejs e Express localiza-se no centro da arquitetura geral, como apresentado na figura 4.2, para servir como conector entre as diversas componentes do projeto. Ela é responsável pela interligação e manipulação dos dados provenientes da **API** do Fablo, e da **BDD** relacional, em PostgreSQL, onde se aplicam diversas regras e controlos para dar uma interface segura e amigável para trabalhar com o exterior, em especial com a aplicação web. Sem a criação de um sistema intermédio a funcionar como ponte entre ambos os mundos *on-chain* e *off-chain* seria praticamente impossível desenvolver uma solução deste tipo pois não haveria uma comunicação eficaz entre as partes, muito menos um controlo sobre os dados para aí enviados.

4.5.1 Sequelize ORM

Como já explicado em 2.3.14 o Sequelize é uma ferramenta que fornece uma **API** para que seja possível comunicar facilmente com a **BDD** através do servidor javascript. Esta tem um papel muito importante neste projeto pois é responsável de fazer toda a comunicação base entre a fonte

de dados e o servidor backend e o armazenamento das sessões de utilizador. Esses dados vão ser usados por aplicações terceiras, que neste caso são a plataforma web e a Fablo API como se pode ver na figura 4.5.

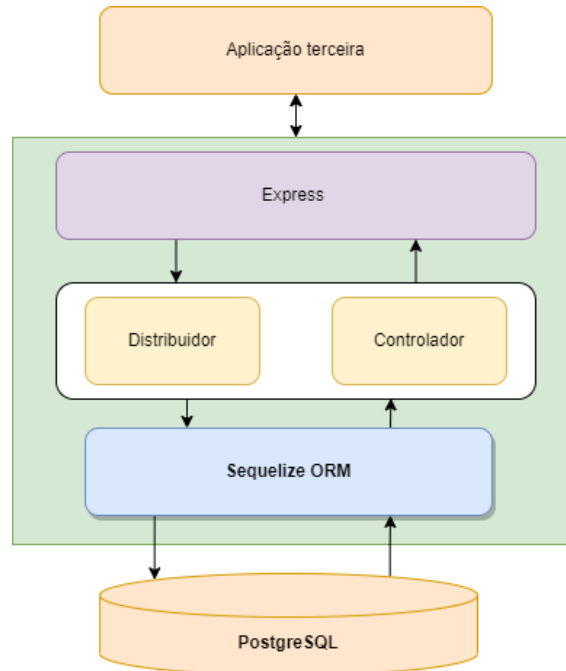


Figura 4.5: Diagrama de funcionamento Express com Sequelize ORM

De forma a ter um desenvolvimento mais controlado e centralizado, todos objetos e dados da **BDD** PostgreSQL foram desenvolvidos em javascript, no servidor Express, com a ajuda da **API** do Sequelize. Para cada tabela da **BDD** referida na figura 4.4 desenvolvi um modelo de definição de dados, similares ao excerto de código do listing 4.1, onde são definidos os tipos de dados, as chaves primárias e estrangeiras, e os comportamentos que esta deve ter, por exemplo, ao tentar eliminar um registo (linhas 21 à 23). Para além disso, o Sequelize permite fazer a validação dos dados do lado do servidor, sem ser necessário esperar uma resposta da **BDD** com um erro por tipo inconsistente de dados, como exemplificado nas linhas 11 à 19.

```
1 const User = db.define('User', {
2   id: {
3     type: DataTypes.UUID,
4     primaryKey: true,
5     defaultValue: DataTypes.UUIDV4
6   },
7   email: {
8     type: DataTypes.STRING,
9     allowNull: false,
10    unique: true,
11    validate: {
12      notEmpty: {
13        msg: "Error! Field {email} is required!"
14      },
15      isEmail: {
```

```
16         msg: "Error! Field {email} must be an email!"
17     },
18 }
19 },
20 }, {});
21 User.belongsTo(Company, {
22     foreignKey: { allowNull: true },
23 })
```

Listing 4.1: Definição de tabela no Sequelize

4.5.1.1 Migrações

As migrações feitas com o Sequelize são uma espécie de controlo de versão da **BDD**, que se pode comparar a ferramenta GIT usada para o controlo de versão de ficheiros. Com estas migrações é possível criar e navegar entre vários estados da **BDD** por ordem cronológica de desenvolvimento sem criar inconsistências nas tabelas ou chaves estrangeiras.

A criação de uma migração, que pode ser gerada automaticamente através da linha de comandos, consiste no desenvolvimento de duas funções (up e down) que são responsáveis por adicionar as modificações as tabelas (up) e por permitir reverter as mesmas (down). No excerto de código do listing 4.2, servindo de exemplo, criei uma migração para a criação de uma tabela (linhas 3 à 14), e a respetiva função de reversão (linhas 15 à 17) em que é feito a remoção da mesma.

```
1 module.exports = {
2   async up(queryInterface, Sequelize) {
3     await queryInterface.createTable('FinalIndicators', {
4       id: {
5         type: Sequelize.UUID,
6         primaryKey: true,
7         defaultValue: Sequelize.UUIDV4
8       },
9       createdAt: {
10        allowNull: false,
11        type: Sequelize.DATE
12      },
13    });
14  },
15  async down(queryInterface, Sequelize) {
16    await queryInterface.dropTable('FinalIndicators');
17  }
18 };
```

Listing 4.2: Definição de migração no Sequelize

Neste projeto apenas foram desenvolvidas algumas migrações, já após terem sido criados todos os modelos e tabelas de modo a investigar o seu funcionamento e enriquecer a pesquisa e o conhecimento próprio. De qualquer forma, as migrações são adaptáveis a projetos já existentes, podendo ser usadas em qualquer estado da **BDD**. Na secção de discussão 5.4 aborda-se mais sobre o tema nomeadamente alguns problemas de que carece.

4.5.1.2 Seeders

Os *Seeders* pertencem as migrações e funcionam de forma muito similar a estas, a diferença está essencialmente em que as migrações tratam da parte de operações SQL para Data Definition Language (DDL) e os seeders de Data Manipulation Language (DML). Assim sendo, esta componente responsabiliza-se por adicionar dados à base de dados de uma forma estruturada através de código javascript e que podem ser revertidas facilmente através da consola. Isto pode ser especialmente útil para a testar a aplicação adicionando vários dados de exemplo específicos e com a possibilidade de revertê-los mais tarde. Estes dados podem servir para testar por exemplo validações de campos ao inserir na BDD e ser usada a função up e down até os problemas estarem resolvidos e o seeder ser revertido. No excerto 4.3 desenvolvi o registo de um utilizador por defeito que é usado para testar a aplicação com a permissão de administrador, evitando a criação da mesma e alteração das permissões diretamente na base de dados.

```
1 module.exports = {
2
3   async up(queryInterface, Sequelize) {
4
5     await queryInterface.bulkInsert('Users', [
6       {
7         id: v4(),
8         email: 'admin@email.com',
9         name: 'Administrator',
10        password: await bcrypt.hashSync('admin', 10),
11        permission: 'ADMIN',
12        createdAt: new Date(),
13        updatedAt: new Date()
14      }
15    ],
16    async down(queryInterface, Sequelize) {
17      await queryInterface.bulkDelete('Users', null, {});
18    }
19  };
```

Listing 4.3: Definição de seeder no Sequelize

4.5.2 Traduções

De forma a tornar a aplicação o mais versátil possível foi decidido adicionar traduções às mensagens que são devolvidas para o utilizador. Dessa forma, é possível dar uma melhor experiência de utilização a utilizadores de várias línguas. O projeto foi desenvolvido com as mensagens em inglês como valores base com suporte a português.

A escolha do idioma é feito com a leitura do parâmetro "Accept-Language" contido nas requisições HTTP efetuadas ao servidor que gere qual o idioma que deve ser apresentado. No excerto 4.4 é inicializado o `i18next`, já mencionado em 2.3.17 que deteta o idioma fornecido pelo utilizador (linha 2) e caso não seja encontrada mostra o idioma por defeito (linha 4), neste caso o inglês.

```

1 i18next
2   .use(middleware.LanguageDetector)
3   .init({
4     fallbackLng: 'en',
5     resources: {
6       en: { translation: require('./locales/en/translation.json') },
7       pt: { translation: require('./locales/pt/translation.json') }
8     }
9   })

```

Listing 4.4: Inicialização do i18next

Todas as mensagens são armazenadas em ficheiros JSON para cada idioma (linhas 6 e 7) com a chave sendo o identificador da mensagem em todas as linguas e o valor a mensagem traduzida.

Após a configuração do i18next apenas é necessário chamar a função `req.t()` para enviar a tradução na resposta da requisição. No exemplo 4.5 é enviado na resposta da requisição a mensagem referente à chave indicada no idioma selecionado.

```

1   return res.status(200).json({ data: req.t("userAuthenticated") })

```

Listing 4.5: Chamada de tradução requisição HTTP

4.5.3 Integração com Fablo API

Como falado anteriormente em 4.3 a API central integra a Fablo API para a inserção e pedido de dados à blockchain. Aqui são feitas chamadas através de requisições HTTP a esta que se encontra num endereço IP distinto. Este endereço deve ser chamado por diferentes portas dependendo da empresa parceira que quer enviar a requisição. Para isso é necessário determinar previamente para onde devem ser feitas as requisições, que como demonstrado no excerto de código do listing 4.6, onde são determinadas pela empresa associada ao utilizador com sessão iniciada (linhas 10 e 11).

```

1 export default async function fabloRequest(req: Request, method: string, url: string, data:
  object | undefined = undefined): Promise<any> {
2   try {
3     const token = req.session.user && req.session.user.onChainToken
4     const PORTS: PortStructure[] = [ ... ]
5     const request = await axios({
6       headers: { authorization: token ? `Bearer ${token}` : undefined },
7       url: url, method: method,
8       baseUrl: "http://127.0.0.1:"
9         + PORTS.find(item => item.company === req.session.user.companyId)!.port,
10      data: data
11    })
12    return request;
13  } catch (error: any) { throw error; }
14 }

```

Listing 4.6: Estrutura chamada a Fablo API

Cada requisição segue uma estrutura definida na API externa a qual tem de ser cumprida sem exceção, visto que esta tem uma estrutura bastante rígida face a necessidade de enviar informações precisas para a blockchain.

Em 4.7 é demonstrado o envio de uma produção para a blockchain. São recebidos todos os parâmetros através do corpo da requisição da API de integração e depois inseridos de forma ordenada nos argumentos da requisição ao Fablo e o respetivo método.

De modo a ser mais fácil a interação com a API foi decidido desestruturar o *Array* enviado para o Fablo num objeto como chaves e valores (linhas 3 e 4), onde é possível ler e inserir facilmente os respetivos parâmetros. Dessa forma, ao testar as requisições no Postman, ou a enviar os pedidos através da plataforma web, tudo se torna mais fácil pois não é necessário seguir uma ordem de declaração dos parâmetros nem fazer a desestruturação dos objetos para texto, necessários nestas solicitações.

```
1 router.post('/insert', async (req: Request, res: Response<RequestResponse>) => {
2   try {
3     const { productionID, productionUnitInternalID, productionType, activityStartDate,
4         batchID, batchType, batchInternalID, supplierID, inputBatches, batchComposition
5         , quantity, finalScore, productionScore, ses } = req.body;
6
7     const data = {
8       method: "StvgdContract:CreateProduction",
9       args: [productionID, productionUnitInternalID, productionType,
10            activityStartDate, batchID, batchType, batchInternalID, supplierID, JSON.
11            stringify(inputBatches), JSON.stringify(batchComposition), quantity,
12            finalScore, productionScore, ses]
13     }
14
15     const request = await fabloChannelRequest(req, 'invoke', data)
16
17     return res.status(200).json({ data: request.data.response })
18   } catch (error: any) {
19     return res.status(400).json({ error: error })
20   }
21 });
```

Listing 4.7: Chamada POST à Fablo API

Para além da usabilidade mais eficaz deste método de chamadas a uma API, esta integração tem o objetivo de tornar possível o cálculo dos indicadores de sustentabilidade através de formulas armazenadas na BDD relacional, que atualmente não podem ser divulgadas ao público, aquando da requisição e, a validação dos parâmetros antes de os enviar para a blockchain, evitando assim o envio de informações erróneas que de outra forma estariam a sobrecarregar a blockchain com pedidos inúteis e a manter o utilizador em espera, visto que estes pedidos são mais lentos do que um simples pedido à API.

4.5.4 Sistema de Cache com Redis

Tendo em conta a lentidão de resposta das chamadas à Fablo API foi necessário procurar um método para reduzir esse tempo e melhorar a experiência do utilizador. A solução consiste num sistema de cache das respostas em que os dados recebidos e processados pela API são guardados em memória durante um determinado período de tempo. Enquanto existe informação armazenada em cache essa é automaticamente enviada como resposta as requisições, caso contrário é feito o processo normal de pesquisa e calculo dos arcos e nós, como demonstrado na figura 4.6.

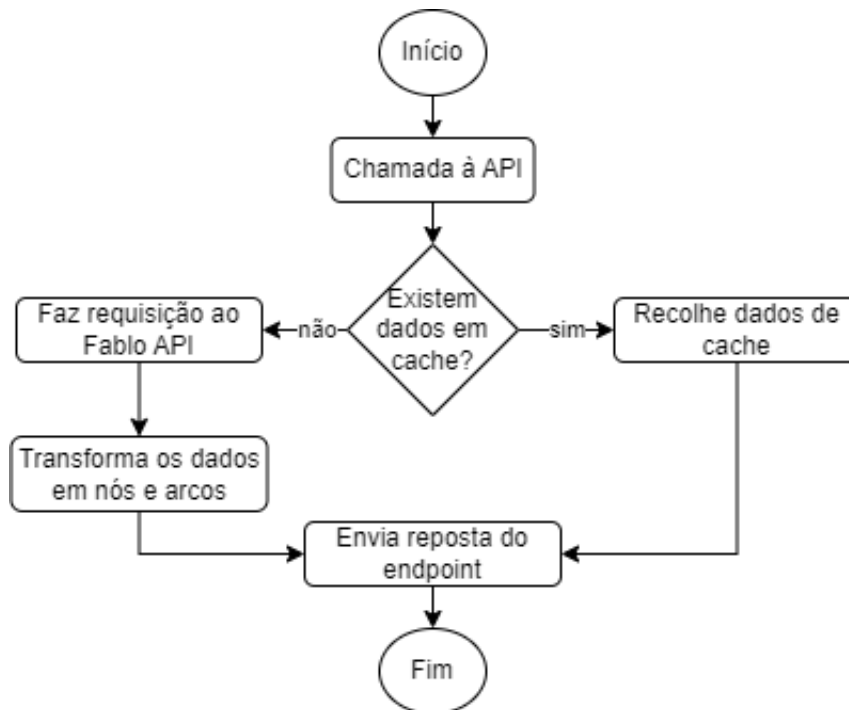


Figura 4.6: Diagrama de fluxo chamadas API com redis

A informação pode permanecer por um período longo em cache devido ao facto de as informações não serem alteradas frequentemente. Para além disso, sempre que é adicionada uma atividade à rede, o sistema de cache é limpo (linhas 4 e 5 do excerto de código do listing 4.8) o que obriga a atualização da rastreabilidade.

```

1 router.post('/insert', async (req: Request, res: Response<RequestResponse>) => {
2   try {
3     (...)
4     client.del('graphMode')
5     client.del('graphMapMode')
6
7     return res.status(200).json({ data: request.data.response })
8   } catch (error: any) {
9     return res.status(400).json({ error: error })
10  }
11 });
  
```

Listing 4.8: Exemplo de remoção da cache ao adicionar nova atividade

Esta solução faz com que a aplicação web consiga demonstrar os grafos num tempo muito reduzido que de outra forma tornava-se extremamente lento, criando grandes possibilidades do utilizador desistir da aplicação.

4.6 Processamento e demonstração da rastreabilidade

A representação dos dados de rastreabilidade e indicadores de sustentabilidade de forma a que os utilizadores da plataforma consigam entender facilmente requer diversos passos de recolha, processamento e estruturação de dados. Para isso foi necessário recolher dados da blockchain (através da Fablo API) para a API de integração, fazer o processo de estruturação dos mesmos através de funções recursivas, cujo diagrama de fluxo está representado na figura 4.8 e que são enviados para a plataforma web, onde são novamente alvo de algum processamento para demonstração nos componentes da mesma.

Na figura 4.7 é demonstrado em alto nível o fluxo da rastreabilidade desde a blockchain até à aplicação web.

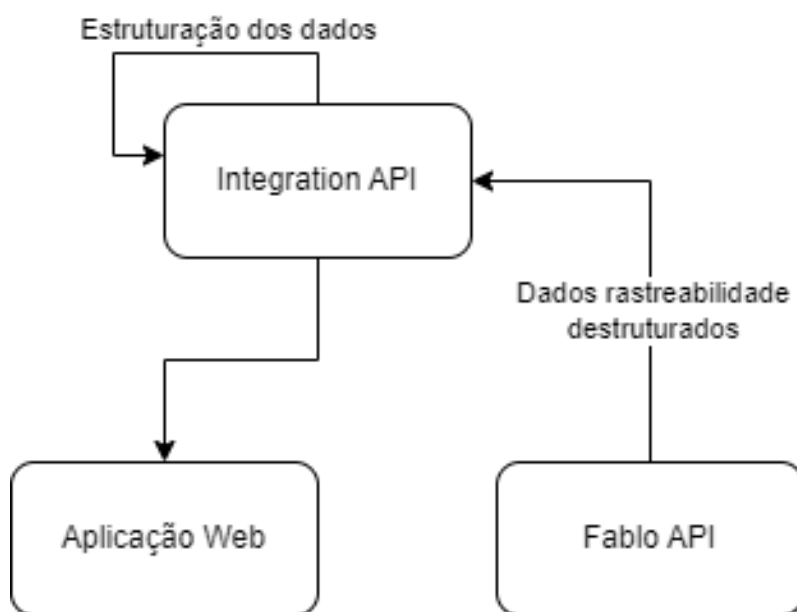


Figura 4.7: Diagrama de fluxo do processamento dos dados de rastreabilidade

4.6.1 Algoritmo de rastreabilidade

Perante a necessidade de demonstrar os dados em forma de grafos, tanto sobre um mapa como de forma simples, era imperativo a API de integração devolver os dados em forma de objeto com nós e arcos como apresentado no excerto 4.9. Considerando esse requerimento iniciou-se uma análise dos dados recolhidos da blockchain para definir uma estratégia para manipular-los e chegar a um resultado que todos os lotes e atividades eram armazenados num array, e a partir deles eram definidos os arcos de ligação (excerto 4.10).

```

1   data = {
2     nodes: [],
3     arcs: []
4   }

```

Listing 4.9: Estrutura resposta informação de rastreabilidade

O algoritmo de processamento de dados desenvolvido, dado o tipo de dados recolhidos na blockchain, é do estilo de árvore n-ária de pré-ordem que recursivamente vai adicionando todos os lotes e atividades em forma de nós e posteriormente são calculados os arcos através dos nós armazenados. Após recebida a informação através do endpoint da Fablo API é chamada a função demonstrada no excerto de código do listing 4.10 que se responsabiliza pela limpeza dos dados armazenados em consultas prévias, a criação dos nodos através dá função recursiva (listing 4.11) e depois o calculo dos arcos (listing 4.13). No final devolve os dados como demonstrados no excerto de código do listing 4.9.

```

1 export default function getTraceabilityMapData(data: any):
2   { nodes: Array<any>, arcs: Array<any> } {
3   nodes = arcs = [];
4   data.map((item: any) => GraphMapHandler(item))
5   calculateArcs(nodes)
6   return { nodes, arcs }
7 }

```

Listing 4.10: Função de sequenciação da rastreabilidade

No excerto de código do listing 4.11 é demonstrado como para cada interação em que a função é chamada, verifica-se qual é o tipo de documento (registo, lote, produção, transporte ou receção) e dependendo dessa variável é chamada a função que vai fazer a transformação dos dados e adicionar um novo elemento ao array de nós. Em caso desse elemento ter valores de rastreabilidade, é chamada novamente a função recursiva para prosseguir com a pesquisa em árvore até todos os nós serem adicionados.

```

1 function GraphMapHandler(info: any) {
2   try {
3     switch (info.docType) {
4       case 'rg': registrationProcedure(info); break;
5       case 'rc': receptionProcedure(info); break;
6       case 't': transportationProcedure(info); break;
7       case 'p': productionProcedure(info); break;
8       case 'b': batchProcedure(info); break;
9       default: throw new Error("GraphMapHandler: Invalid doc type: " + info.docType);
10    }
11    nodes = nodes.filter((node, index) => {
12      return index === nodes.findIndex((n) => n.ID === node.ID);
13    });
14  } catch (error) { throw error; }
15 }

```

Listing 4.11: Função recursiva criação lotes

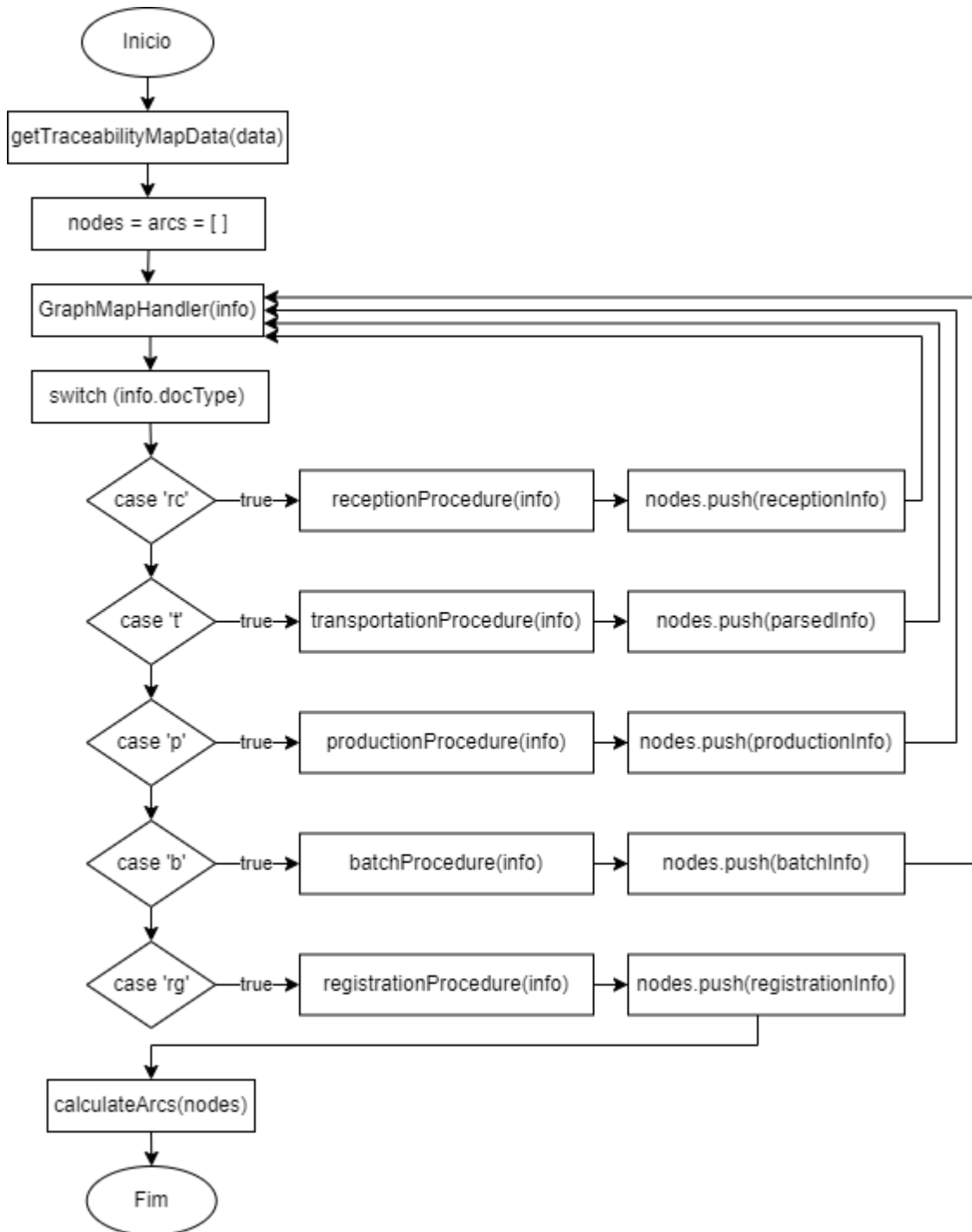


Figura 4.8: Diagrama de fluxo da função recursiva de rastreabilidade

No excerto de código do listing 4.12, tal como em todos os procedimentos contidos na função recursiva, é verificado se o tipo de documento é o correto (linhas 3 e 4), depois chamada a função recursiva caso hajam elementos de rastreabilidade (linha 8) e por fim criado um objeto que será adicionado como nó, que adiciona a localização geográfica e os dados de input e output necessários para a demonstração dos grafos.

```
1 function batchProcedure(info: BatchType): void {
2     try {
3         if (info.docType !== "b")
4             throw new Error("Unexpected document type! Expected value is 'b'.");
5
6         if (!info.traceability) return
7
8         GraphMapHandler(info.traceability[0])
9
10        let newObject: MapInfoInterface & BatchType = {
11            ...info,
12            mapInfo: {
13                coordinates: coordinatesHandler(info.latestOwner),
14                input: [],
15                output: []
16            }
17        }
18
19        newObject.mapInfo.input = []
20        newObject.mapInfo.output = []
21        newObject.traceability = []
22
23        setNodes(newObject)
24    } catch (error) {
25        throw error
26    }
27 }
```

Listing 4.12: Função de processamento dos lotes

Já no final, depois de todas as atividades e lotes terem sido adicionados ao array dos nós e a a recursividade ter terminado, para cada tipo de demonstração dos grafos, são chamadas funções distintas responsáveis por calcular todos os arcos entre atividades e entre atividades e lotes.

O excerto de código do listing 4.13 demonstra a função destinada à demonstração dos dados num mapa geográfico percorre todos os nós e de forma sequencial verifica diversas condições nos nós de output, input e a exceções para os nós de transporte (linhas 39 à 53), que não tem um output final, senão uma receção com o mesmo lote. Está demonstrado como são percorridos todos os nós de output em busca dos respetivos nós e caso coincidam é adicionado um arco entre si, sendo feito para os nós de input o mesmo processo. São também adicionados nós entre as atividades, verificando se o nó de output de um é igual ao nó de input do outro. Por fim, como exceção, são filtrados os nós de transporte que é necessário interligar com as receções as quais não tinham qualquer relação direta de rastreabilidade. Assim chega-se a um resultado final com todas as atividades interconectadas com a descrição dos lotes de entrada e saída, como se pode ver na

figura 4.9.

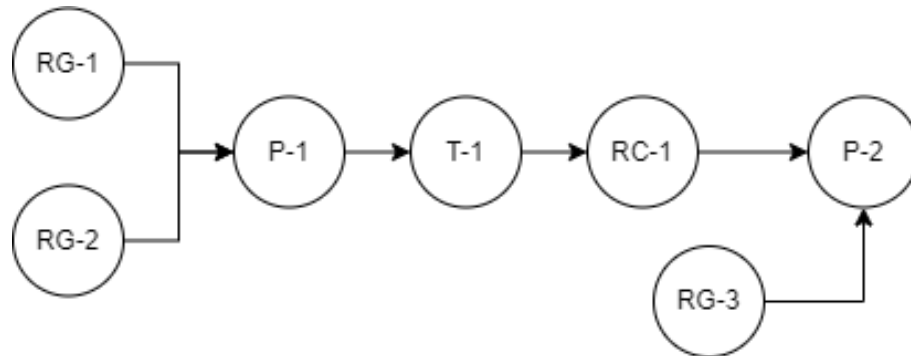


Figura 4.9: Diagrama Estilo Representação Rastreabilidade no Mapa

```

1
2 function calculateArcs(nodes: Array<any>): void {
3   nodes.forEach(node => {
4     node.mapInfo.output.forEach((element: Array<string>) => {
5       const initialNode = node.mapInfo.coordinates
6       const finalNode = nodes.find(filtered => filtered.ID === element).mapInfo.
          coordinates
7
8       arcs.push({
9         ID: node.ID + "-" + element,
10        activityConnection: false,
11        initialNode: { ID: node.ID, ...initialNode },
12        finalNode: { ID: element, ...finalNode }
13      })
14
15      const finalNodeAtivity = nodes.filter((filtered: any) => filtered.mapInfo.input
          .includes(element))[0]
16
17      if (finalNodeAtivity) {
18        arcs.push({
19          ID: element + "-" + node.ID,
20          activityConnection: true,
21          initialNode: { ID: node.ID, ...initialNode },
22          finalNode: { ID: finalNodeAtivity.ID, ...finalNodeAtivity.mapInfo.
              coordinates }
23        })
24      }
25    });
26
27    node.mapInfo.input.forEach((element: Array<string>) => {
28      const finalNode = node.mapInfo.coordinates
29      const initialNode = nodes.find(filtered => filtered.ID === element).mapInfo.
          coordinates
30
31      arcs.push({
32        ID: element + "-" + node.ID,
33        activityConnection: false,
34        initialNode: { ID: element, ...initialNode },
35        finalNode: { ID: node.ID, ...finalNode }

```

```

36     })
37   });
38
39   if (node.docType === 't') {
40     const initialNode = node.mapInfo.coordinates
41
42     let finalNode: any = nodes.filter((filteredDocType: any) => filteredDocType.
43       docType === 'rc')
44       .filter(filterInput => filterInput.mapInfo.input[0] === node.mapInfo.input
45         [0])[0]
46
47     if (finalNode) {
48       arcs.push({
49         ID: node.ID + "-" + finalNode.ID,
50         activityConnection: true,
51         initialNode: { ID: node.ID, ...initialNode },
52         finalNode: { ID: finalNode.ID, ...finalNode.mapInfo.coordinates }
53       })
54     }
55 }

```

Listing 4.13: Função de criação de arcos

Para a demonstração dos grafos num ambiente simples sem o mapa geográfico recorre-se, como indicado anteriormente, a outro algoritmo específico para esse fim. Aqui querem-se todas as ligações entre as atividades e os lotes consequentes, e também as ligações entre os transportes e as receções, que de outra forma não teriam ligação devido ao facto de ser o mesmo lote e ficar uma ligação duplicada. Na figura 4.10 é mostrado um exemplo de como se desenvolveu a ligação entre estes nós.

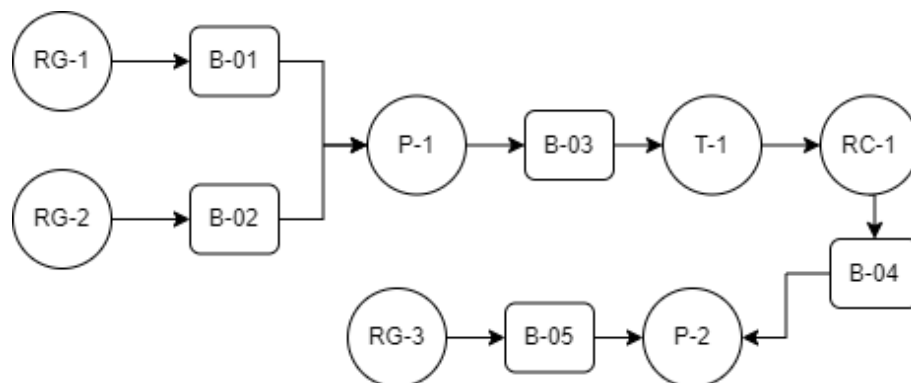


Figura 4.10: Diagrama Estilo Representação Rastreabilidade

4.6.2 Demonstração de grafos

Após o cálculo dos nós e arcos os dados necessários para criar os grafos são enviados como resposta da requisição. Em 4.14 são demonstrados os dados ainda desestruturados e em 4.15 é demonstrado um exemplo da resposta da API que devolve a informação já estruturada.

```

1 {data: [
2     {
3         docType: "b",
4         ID: "b-103",
5         [...],
6         traceability: [
7             {
8                 ID: "p-101",
9                 docType: "p",
10                inputBatches: {[...]}
11                traceability: [...]
12            },
13        ]
14    }
15 ]
16 }
```

Listing 4.14: Exemplo de resposta à requisição de nodos

```

1 {data:
2     {nodes: [{ID: "rg-41", activityDate: "2023-05-06T13:43:48.531Z", docType: "rg", ...},
3         ...],
4     {arcs: [{ID: "rg-41-b-41", activityConnection: false, ...}, {ID: "b-41-rg-41",
5         activityConnection: true, ...}, ...]
6 }
```

Listing 4.15: Exemplo de resposta do Fablo API

Com a resposta dada são aplicados algoritmos que dependendo do estilo de interface que se pretende, transformam os mesmos numa visualização fácil e agradável ao utilizador.

Na figura 4.11 está demonstrada uma parte do mapa geográfico em que são demonstrados os grafos. A representação consiste em nós com a informação da atividade e arcos que demonstram a ligação sequencial entre as mesmas. A informação de uma atividade é mostrada sempre que é feito um clique sobre o ícone da mesma o que faz surgir uma janela com a informação dessa atividade, assim como os lotes de entrada e saída da mesma.

Já na figura 4.12 a representação dos arcos e nós é baseada no esquema demonstrado na figura 4.10 em que há uma ligação entre as atividades e os nodos, de forma sequencial de modo a mostrar uma ordem na criação dos lotes.

A representação é interativa pelo que ao pressionar num dos nós é demonstrada a informação do nó e respeito impacto ambiental do mesmo.

4.7 Aplicação Web

A plataforma web é a ligação entre os utilizadores e o sistema. Aqui todos os intervenientes têm certas permissões e acessos a diferentes interfaces dependendo do grau de responsabilidade no ecossistema do projeto.

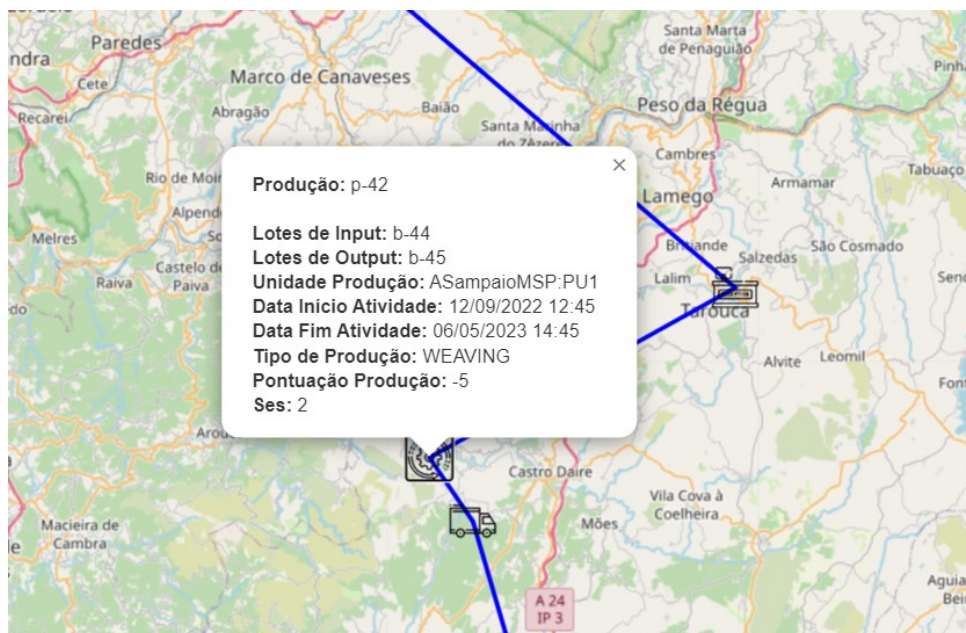


Figura 4.11: Representação rastreabilidade no mapa geográfico

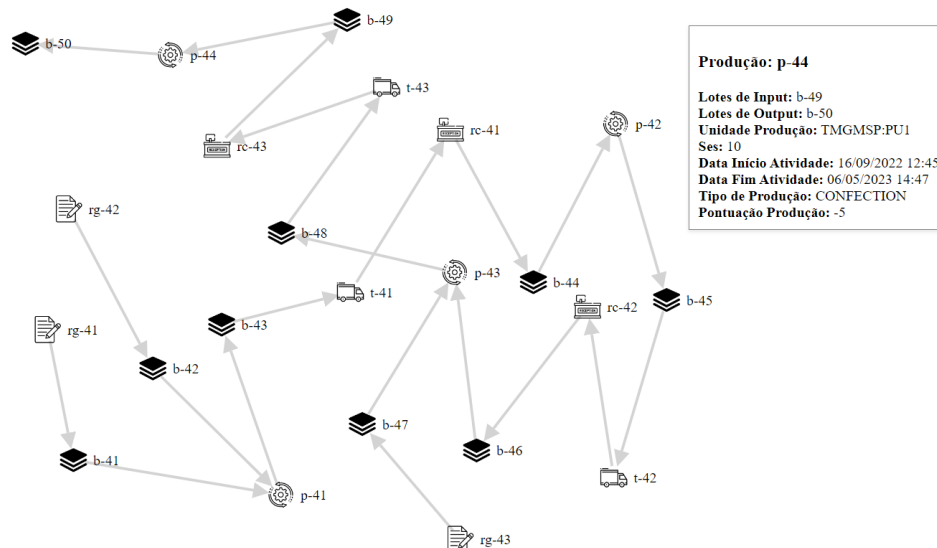


Figura 4.12: Representação rastreabilidade no mapa livre

De seguida irei apresentar todas as funcionalidades presentes na plataforma e quais perfis tem acesso as mesmas. Um controlo apropriados das permissões dos utilizadores é fundamental para o bom e seguro funcionamento da aplicação.

4.7.1 Perfil de Consumidor Final

Na aplicação o consumidor final tem acesso, sem necessidade de se autenticar com nenhum tipo de conta, aos dados de rastreabilidade e indicadores de sustentabilidade através de e grafos representados em cima de uma mapa geográfico e através de uma representação de grafos simples e interativos.

Para além de visualizar a representação de todos os nodos e arcos da rastreabilidade num todo, pode filtrar-los por lote onde só são apresentados os dados referentes a esse mesmo lote e à rastreabilidade que está para trás.

4.7.2 Perfis de Responsável e Membro Organização

No *smart contract* foi definido que existem dois perfis de utilizador em cada organização parceira: o administrador, e o operador. Assim sendo, na plataforma web reflete-se essa decisão e existem os mesmos dois agentes, aqui chamados de responsável e membro da organização.

Estes perfis são os responsáveis por adicionar dados de rastreabilidade a plataforma ao criarem atividades de registo, produção, receção e transporte através de um backoffice especificamente desenhado para esse propósito. Aí os membros e responsáveis das organizações acedem a diferentes formulários, demonstrados na figura 4.13, onde lhe é solicitado o preenchimento de diversos campos necessários para criar uma nova atividade. Esse registo de atividade é enviado para a API de integração que os recebe e perante essas informação faz o cálculo dos indicadores de sustentabilidade através das formulas armazenadas na BDD relacional.

Category	Field Name	Value
Registo	Registration ID *	rg-41
	Batch ID *	b-41
	Batch Internal ID *	b-41-iid
Receção	Quantity *	100
	Production Unit	PU1
Transporte	batchType	ORGANIC_COTTON
	Supplier ID *	suppl-01
Produção	Composition Name *	organic_cotton
	Composition Quantity *	100
	Composition Name 2 *	
	Composition Quantity 2 *	

Figura 4.13: Gestão de dados On-chain na plataforma

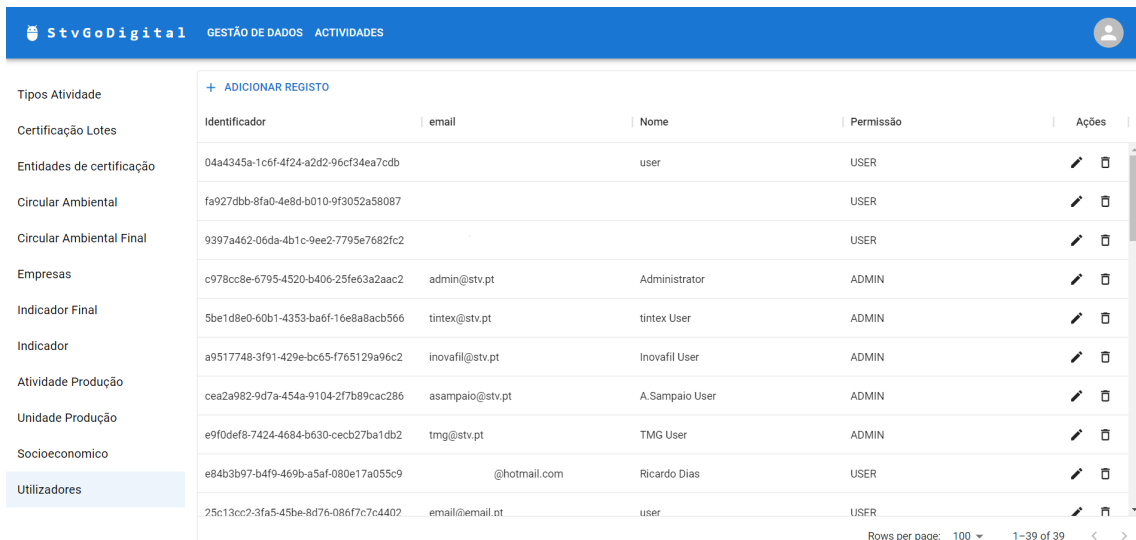
O Responsável da organização tem acesso às mesmas funcionalidades e ainda a uma funcionalidade extra que utiliza para criar novos membros que funcionam como operadores na blockchain.

Além da parte de backoffice, estes membros também podem pesquisar informações de rastreabilidade como os consumidores finais.

4.7.3 Perfil de Administrador

O Administrador tem acesso ao backoffice de administração onde pode gerir os dados guardados na base de dados off-chain do PostgreSQL que contém toda a informação das empresas, utilizadores, permissões, indicadores de sustentabilidade, entre outros.

Este perfil não tem acesso à criação de atividades pois não lhe pertence nenhuma empresa nem deve inserir dados na blockchain. Por outro lado, pode como todos os outros perfis, aceder a rastreabilidade das atividades e lotes. Na figura 4.14 é mostrado o painel de administração a que o administrador tem acesso. Aqui este pode adicionar, alterar e remover registos da base de dados caso cumpram as regras definidas na [API](#).



The screenshot shows the 'Gestão de dados Off-chain na plataforma' interface. The top navigation bar includes 'StvGoDigital', 'GESTÃO DE DADOS', and 'ACTIVIDADES'. A sidebar on the left lists various activity types, with 'Utilizadores' selected. The main content area displays a table with a '+ ADICIONAR REGISTO' button and a table of users.

Tipos Atividade	+ ADICIONAR REGISTO				
	Identificador	email	Nome	Permissão	Ações
Certificação Lotes	04a4345a-1c6f-4f24-a2d2-96cf34ea7cdb		user	USER	
Entidades de certificação	fa927dbb-8fa0-4e8d-b010-9f3052a58087			USER	
Circular Ambiental	9397a462-06da-4b1c-9ee2-7795e7682fc2			USER	
Circular Ambiental Final					
Empresas	c978cc8e-6795-4520-b406-25fe63a2aac2	admin@stv.pt	Administrator	ADMIN	
Indicador Final	5be1d9e0-60b1-4353-ba6f-16e8a8acb566	tintex@stv.pt	tintex User	ADMIN	
Indicador	a9517748-3f91-429e-bc65-f765129a96c2	inovafil@stv.pt	Inovafil User	ADMIN	
Atividade Produção	cea2a982-9d7a-454a-9104-2f7b89cac286	asampaio@stv.pt	A.Sampaio User	ADMIN	
Unidade Produção	e9f0def8-7424-4684-b630-cecb27ba1db2	tmg@stv.pt	TMG User	ADMIN	
Socioeconomico	e84b3b97-b4f9-469b-a5af-080e17a055c9	@hotmail.com	Ricardo Dias	USER	
Utilizadores	25e13cc2-3fa5-45be-8d76-086f7c7c4402	email@email.pt	user	USER	

Rows per page: 100 1-39 of 39

Figura 4.14: Gestão de dados Off-chain na plataforma

Capítulo 5

Avaliação e Resultados

5.1 Introdução

Este capítulo é dedicado à avaliação e testes aos resultados obtidos com a realização do projeto. Aqui vão ser demonstrados alguns dos testes feitos durante e no final do desenvolvimento da aplicação. Em primeiro demonstro os teste unitários realizados para testar a **API** e verificar se esta se comporta como é esperado para cada caso e exceção, e depois é feita uma avaliação a plataforma web, especificamente a demonstração dos grafos, onde é criado um caso de uso e verificada a sua correta demonstração. Por fim é feita uma discussão geral do que foi o desenvolvimento da aplicação, nomeadamente alguns problemas e soluções e o porquê de certas escolhas.

5.2 Testes Unitários

Os testes unitários são uma prática de desenvolvimento de software que visa testar unidades isoladas de código de forma independente. Uma unidade pode ser uma função, um método de uma classe, um componente, ou qualquer outra parte do código que possa ser testada isoladamente.

O objetivo dos testes unitários é garantir que cada unidade de código funcione corretamente de acordo com sua especificação. Isso ajuda a identificar e corrigir erros no código, melhorar a qualidade do software e facilitar a manutenção.

Neste projeto foram desenvolvidos testes unitários às requisições da API de integração com o objetivo de salvaguardar o aparecimento de erros nas respostas as requisições e dar uma maior tranquilidade na manutenção das mesmas visto que os cenários podem ser facilmente comprovados mesmo que hajam alterações.

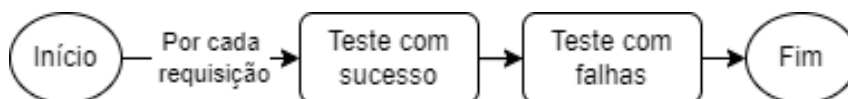


Figura 5.1: Diagrama de fluxo dos testes

A figura 5.1 demonstra através de um diagrama de fluxo simples o padrão de testes que foram realizados à API. Para cada requisição foram desenvolvidos testes com respostas de sucesso e para exemplos com falhas.

Os testes podem e devem ser feitos em blocos organizados para cada rota/funcionalidade que se pretende testar. Assim, como vemos nas figuras 5.1 e 5.2 estes são descritos numa função superior para cada rota e dentro desta são feitos diversos testes para verificar as mais variadas condicionantes que podem ocorrer.

```
1
2 const newUser = {
3   email: `${v4()}@stv.pt`, password: 'password', name: 'user'
4 }
5 const invalidUser = { email: `${v4()}@stv.pt` }
6
7 describe('POST /users/insert', () => {
8   it('should insert a new user', async () => {
9     const response = await request(app).post('/users/insert').send(newUser)
10    expect(response.status).toBe(201)
11    expect(response.body).toHaveProperty('data');
12    expect(response.headers['set-cookie']).toBeDefined();
13  });
14
15  it('should fail to insert a new user', async () => {
16    const response = await request(app).post('/users/insert').send(invalidUser)
17    expect(response.status).toBe(400)
18    expect(response.body).toHaveProperty('error');
19  });
20 })
```

Listing 5.1: Testes unitários na rota criação de utilizadores

```
1 describe("GET /users", () => {
2   it('should get all users', async () => {
3     const agent = request.agent(app);
4
5     await agent.post('/users/login')
6       .send({ email: 'admin@stv.pt', password: 'admin' })
7
8     const response = await agent.get('/users')
9     expect(response.status).toBe(200)
10    expect(response.body).toHaveProperty('data');
11  });
12
13  it('should get forbidden error', async () => {
14    const response = await request(app).get('/users')
15    expect(response.status).toBe(403)
16    expect(response.body).toHaveProperty('error');
17  });
18 })
```

Listing 5.2: Testes unitários na rota de listagem de utilizadores

Após a realização dos testes, e preferencialmente durante a escrita dos mesmos, estes são executados através da linha de comandos, que mostra uma lista detalhada de todos os testes executados e os totais dos testes com sucesso e com erros. Na figura 5.2 é feita uma demonstração de exemplo da verificação de alguns testes unitários.

```
PS C:\Github\fablo_bridge_api> npx jest --detectOpenHandles
PASS src/__tests__/user.test.js (5.427 s)
  POST /users/insert
    ✓ should insert a new user (598 ms)
    ✓ should fail to insert a new user (30 ms)
  POST users/login
    ✓ should login as a valid user (281 ms)
    ✓ should fail to login with invalid credentials (127 ms)
    ✓ should fail to login with missing parameters (62 ms)
  GET /users
    ✓ should get all users (179 ms)
    ✓ should get forbidden error (25 ms)

Test Suites: 1 passed, 1 total
Tests:       7 passed, 7 total
Snapshots:  0 total
Time:        5.656 s, estimated 7 s
Ran all test suites.
```

Figura 5.2: Chamada aos testes unitários

5.3 Casos de Teste de Integração

Para avaliar o resultado final do projeto recorreu-se à criação de um caso de uso específico em que são incluídos todos os tipos de atividades representados no *smart contract* registado na blockchain, e todos os parceiros do STVgoDigital. Este caso de uso consiste na manufatura de um lote de T-shirts pretas, desde o registo das matérias-primas até ao desenvolvimento final da camisola.

O caso de uso inicia-se pela autenticação do administrador de cada empresa para a criação dos respetivos operadores que vão inserir dados na blockchain. Após isso os operadores das empresas Inovafil, A. Sampaio, Tintex e TMG vão, conforme a ordem apresentada, adicionar atividades que por sua vez geram diversos lotes. De seguida demonstro um esquema do fluxo da criação deste exemplo onde são apresentados todos os passos das atividades e a referencia as empresas por onde passam através de secções graficamente divididas.

Na figura 5.3 vemos que a primeira empresa regista o algodão e o poliéster, produz um fio com ambas e envia para a empresa seguinte, a A. Sampaio. Esta recebe o lote de fio, produz o tecido com esse fio e envia-lo para a Tintex. A Tintex recebe o tecido, regista uma tinta preta e procede à tintura do tecido, de seguida envia-o para a TMG. Por fim a TMG recebe o tecido preto e produz a T-shirt preta.

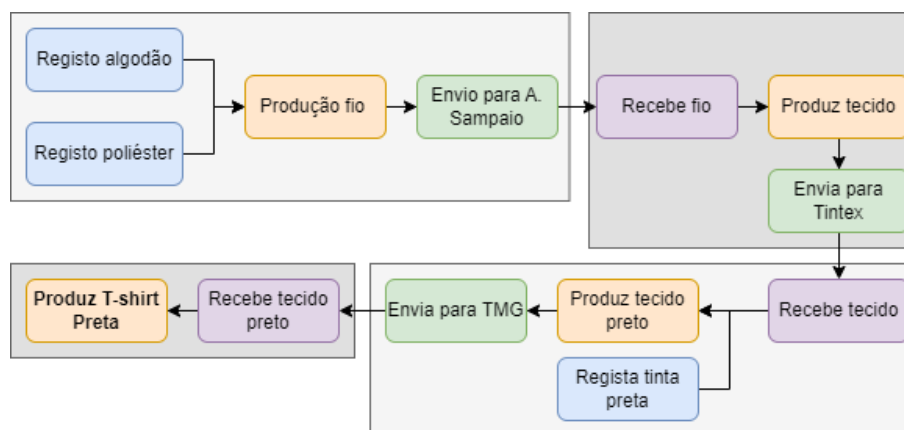


Figura 5.3: Diagrama de exemplo criação T-shirt na blockchain

Depois de registadas todas as atividades e respetivos lotes, já é possível visualizar no mapa a rede de grafos com as atividades. Para evitar a sobreposição os nodos estão distanciados, de forma aleatória, da coordenada central designada à unidade de produção. Na figura 5.4 vemos as atividades espalhadas pelo mapa de forma aleatória em volta dos quatro pontos referentes as unidades de produção e as respetivas ligações entre elas, onde cada lote de entrada e saída tem uma atividade respetiva.

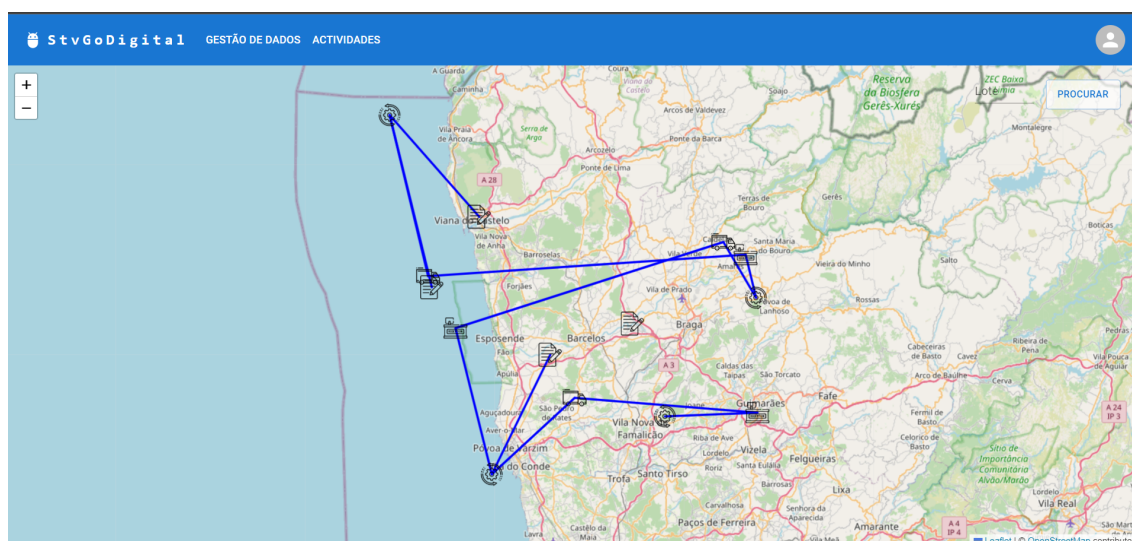


Figura 5.4: Exemplo da T-shirt com grafos no mapa geográfico

Para além do mapa também é possível ver a rastreabilidade dos lotes e atividades no mapa de grafos simples. Este, como apresentado nas figuras 5.5 e 5.6, mostra uma sequência de todos os lotes e atividades de forma detalhada, o que não acontece nos grafos com o mapa.

Na figura 5.6 vemos o resultado da pesquisa por lote, que produz um grafo com os dados do lote e a rastreabilidade anterior associada ao mesmo.

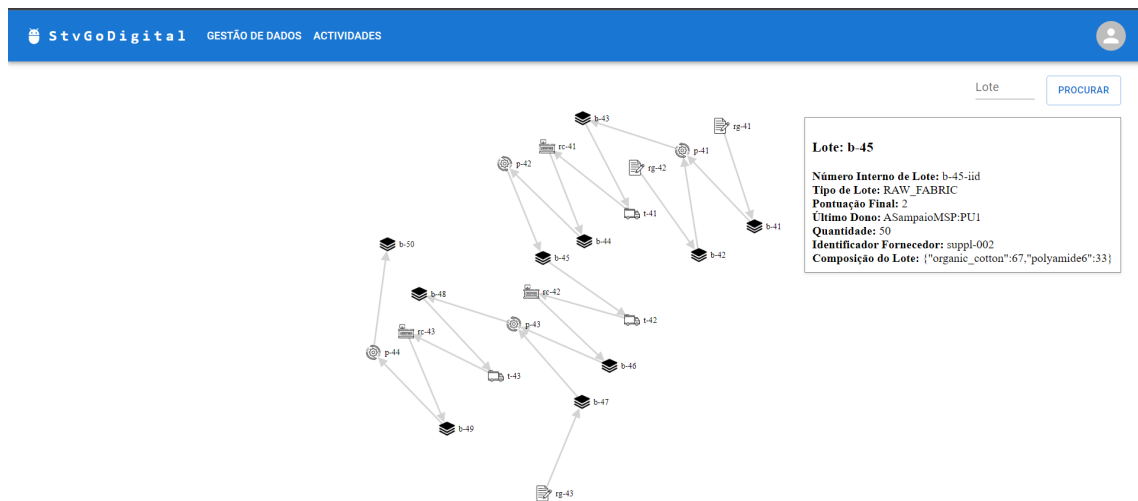


Figura 5.5: Exemplo da T-shirt com grafos simples

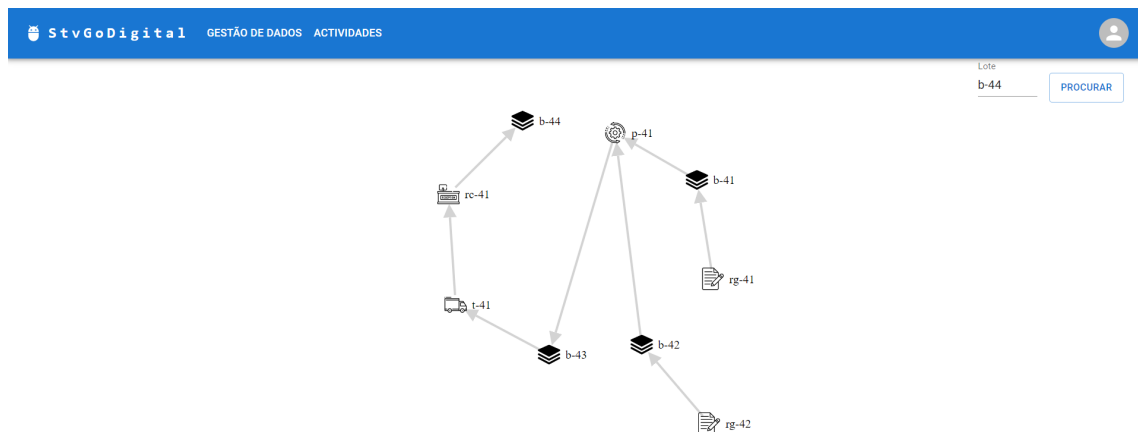


Figura 5.6: Exemplo da T-shirt com grafos simples com pesquisa por ID

5.4 Discussão

O desenvolvimento deste projeto surgiu com a forte necessidade que se tem vindo a sentir na redução do consumo e conseqüente impacto sobre os ecossistemas e a sociedade. Por esse facto é de extrema importância o desenvolvimento de soluções como esta que visam uma melhoria na sustentabilidade da cadeia de valor do **STV**. Com esta solução proporcionamos a todos que desejem consultar, informações fidedignas da origem dos produtos de vestuário que consomem e dos seus impactos a nível social e ecológico.

O projeto apresentado anteriormente a esta discussão está inteiramente documentado nos repositórios do GitHub: <https://github.com/rycardodias/stvgodigital> que contém a aplicação web e https://github.com/rycardodias/fablo_bridge_api que tem toda a parte de backend e ligações aos dados. Aqui é possível verificar todo o código desenvolvido para

a chegada à solução total apresentada no corpo desta dissertação.

Em primeiro será discutida a escolha das tecnologias, algumas já conhecidas e com casos de uso e outras totalmente novas. A escolha das tecnologias tem como princípio base que sejam de código aberto, pois possibilitam o seu uso livre de encargos e atualmente são igualmente versáteis como qualquer software pago.

De forma a uniformizar o desenvolvimento as *frameworks* usadas são programadas na linguagem de programação Javascript que é amplamente usada nos dias de hoje e com uma comunidade de desenvolvedores robusta, o que torna a solução de problemas mais fácil e rápida. Apesar da linguagem final ser essa, o desenvolvimento foi maioritariamente feito com Typescript que é um super conjunto do javascript adicionando-lhe produtividade e facilidade de manutenção do código.

Atualmente existe muita discussão no que toca à performance das aplicações e a busca pela linguagem mais rápida e moderna. Apesar disso, a escolha deve ser baseada em diversos fatores como custos de licenciamento, versatilidade e adaptabilidade das tecnologias e conhecimento das tecnologias por parte dos programadores, que são um custo fixo bastante elevado. Assim sendo, a escolha das frameworks centrou-se no conhecimento prévio das mesmas de forma a reduzir o tempo de aprendizagem e dar mais importância ao desenvolvimento de funcionalidades e um sistema robusto com código limpo.

A plataforma web desenvolvida com o NextJs e React facilitam imenso o desenvolvimento devido ao seu estilo de componentes reutilizáveis e de fácil escrita com suporte a Typescript e com a capacidade de armazenar as páginas pré-renderizadas no servidor o que torna a experiência do utilizador muito mais agradável e rápida. Além de ser excelente para o utilizador final, a barreira de aprendizagem desses conceitos é pequena e de rápida implementação.

O sistema de *backend*, centrado na **API** de integração, já consiste numa solução mais complexa e com diversas tecnologias e conceitos à mistura. Ao longo do projeto, e consoante os problemas foram surgindo, também novas ideias e soluções foram testadas e implementadas.

As migrações são um dos exemplos dessas ideias apresentadas acima. Estas ajudam a tem um desenvolvimento sequencial da estrutura da base de dados e a possibilidade de voltar atrás para implementar correções ou novas regras de negócio sem corromper a integridade da base de dados. Esse foi um dos conceitos explorados mais tarde e que serviu como material de aprendizagem que não foi implementado desde o início do desenvolvimento. Além de útil a ferramenta também carece de alguns melhoramentos pois, como falei anteriormente, é necessário serem desenvolvidos modelos para a utilização do ORM mas estes não são atualizados aquando do uso das migrações, o que faz com que possam existir automaticamente inconsistências entre os modelos e a estrutura atual da base de dados.

A necessidade de utilizar o Redis surgiu aquando do desenvolvimento da plataforma web, quando se verificou que os dados de rastreabilidade faziam com que a plataforma se torna-se lenta para o utilizador. Ao serem processados na blockchain tem um tempo de cálculo elevado que depois era acrescentado com a transformação dos dados para o formato de nós e arcos. Assim sendo foi necessário arranjar uma forma de reduzir o máximo possível a resposta da API com essa informação e surgiu a ideia do Redis como sistema de cache que passou a devolver os dados em

menos de 100 milissegundos ao invés de demorar vários segundos. Nas figuras 5.7 e 5.8 pode-se confirmar a diferença de tempo entre a primeira resposta, em que a cache ainda não foi adicionada e a segunda que já é chamada a cache diretamente sem ser necessário esperar nem sobrecarregar a blockchain.

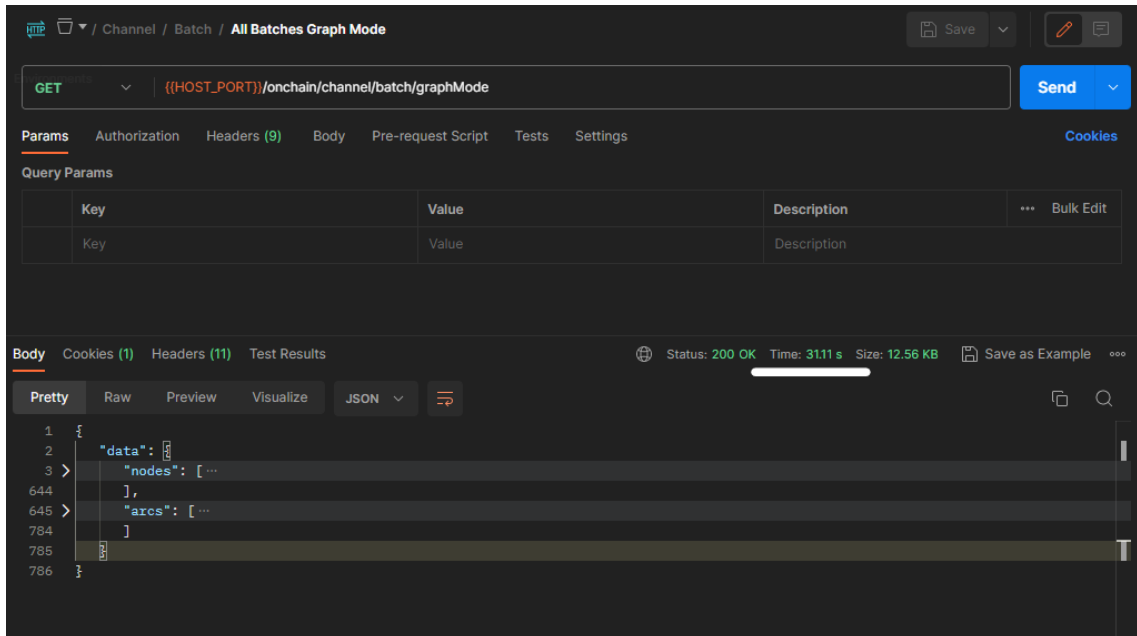


Figura 5.7: Resposta API dados rastreabilidade sem cache

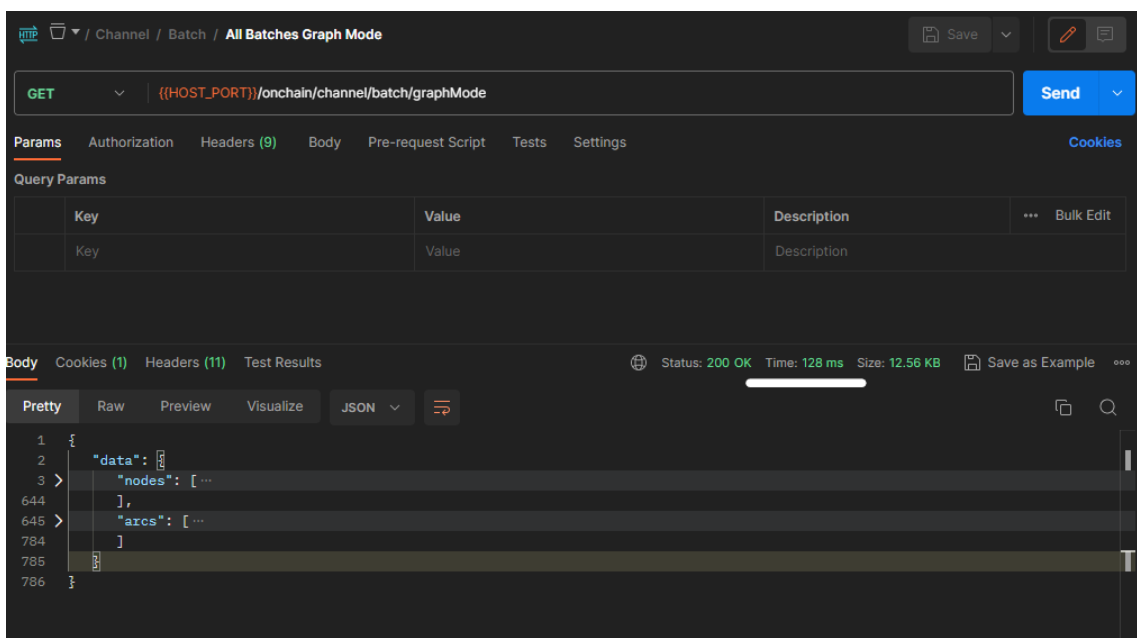


Figura 5.8: Resposta API dados rastreabilidade com cache

No final foram feitos os testes unitários e os testes de integração com um exemplo prático, demonstrados acima nas secções 5.2 e 5.3.

Os testes unitários poderiam ter sido desenvolvidos no início mesmo antes da criação das rotas da API, no entanto decidiu-se recorrer às validações do Typescript para controlar a uniformização das respostas em tempo de desenvolvimento. Como as validações nunca são demasiadas, pelo menos em quantidades razoáveis, decidiu-se criar no final os testes unitários para comprovar vários casos de resposta da API nas rotas mais essenciais, tanto para protegê-las de possíveis erros como para dar aso a investigação das ferramentas de teste Jest e Supertest.

Por fim foi feita a validação da solução como um todo, com um teste de integração de todas as componentes do sistema. Aqui foi criado um produto de raiz onde foram registados todos os passos do mesmo através da plataforma e finalmente foram surgindo os nós e arcos nos diagramas de rastreabilidade. No final temos os grafos finais, como mostrados na figura 5.4 em que aparece a rastreabilidade e todos os indicadores como se planeou inicialmente.

Capítulo 6

Conclusão e Trabalho Futuro

6.1 Introdução

O capítulo final destina-se a uma breve descrição de todo o processo de desenvolvimento do projeto, as dificuldades vivenciadas e superadas durante a sua realização e algum trabalho futuro que pode ser desenvolvido com a disponibilidade de mais tempo útil para esse fim.

Quer-se também referir os benefícios da solução como um todo para a sociedade e o ambiente, e como esta pode ser um ponto de partida para a construção de soluções mais robustas.

6.2 Conclusão

A criação de sistemas capazes de monitorizar e medir os impactos ambientais e sociais das atividades das organizações ao longo de uma cadeia de valor, ajuda a aumentar a consciência ambiental e social global. A presente dissertação demonstrou a análise e desenvolvimento de uma aplicação (backend API, bases de dados e portal web) de rastreabilidade dos indicadores de sustentabilidade social e ambiental na cadeia de valor do têxtil e vestuário. Esta permite que o consumidor final possa consultar os impactos do processo produtivo de determinado lote de um produto têxtil ou de vestuário, e assim ter uma melhor perceção do impacto ambiental e social das peças desse lote, permitindo-lhe responsabilizar socialmente o consumidor pelas suas escolhas, e aumentar a sua consciencialização ambiental e social. A plataforma aqui apresentada permite também verificar a autenticidade de origem de um lote de produtos desde a sua origem até ao consumidor final. Para além disso, as empresas parceiras do projeto, que se responsabilizam por adicionar os dados de rastreabilidade e sustentabilidade dos seus produtos, podem ver através dos dados recolhidos, as lacunas nos seus processos de produção e nos processos dos seus fornecedores, e efetuar melhoramentos de modo a colmatar os prejuízos ambientais e comerciais consequentes.

O desenvolvimento deste projeto é um progresso para a criação de uma economia circular da cadeia de valor do têxtil e vestuário que se baseia na reutilização e reciclagem de materiais e é uma das mais promissoras a nível ambiental. Para criar estes ecossistemas basta incluir empresas de

reciclagem e empresas de criação de fio a partir de materiais reciclados, na cadeia de valor, assim como as atividades correspondentes, para registo na plataforma.

O trabalho desenvolvido nesta dissertação, e aqui apresentado, deu origem a um artigo submetido para a 23ª Conferência da Associação Portuguesa de Sistemas de Informação (CAPSI) sobre este mesmo tema.

6.3 Trabalho Futuro

Partindo da premissa de que um projeto de software nunca tem um fim e apenas lhe é atribuído um por falta de tempo ou recursos, diversas melhorias e funcionalidades poderiam ser adicionadas ao projeto.

A componente talvez mais desejada por um consumidor final seria uma aplicação móvel exclusivamente dedicada para o consumidor final conseguir visualizar aos dados de rastreabilidade e indicadores de sustentabilidade de uma forma mais simples. Este componente reduziria bastante as barreiras de adesão ao projeto pelo facto de ser mais cómodo para o utilizador abrir uma aplicação no seu dispositivo móvel com a possibilidade de verificar de imediato as informações dos lotes e o impacto dos mesmos em qualquer lugar. Assim, este poderia analisar o impacto de determinada peça de vestuário mesmo antes de a adquirir.

Para além dessa componente completamente nova, algumas melhorias poderiam ser feitas tanto a nível de backend como da aplicação móvel. A nível da aplicação web, apesar da responsabilidade básica do MUI alguns componentes poderiam ser revistos e melhorados, os grafos no mapa geográfico carecem de setas de indicação nos arcos de ligação, que existindo dariam ao utilizador uma melhor perceção da rastreabilidade e por fim a interface de adição de atividades por parte dos membros e responsáveis da organização poderia ter um aspeto mais elegante. Já nos serviços de backend, algumas sugestões de melhoria são a adição de cache, como na pesquisa dos dados para os grafos, nos endpoints mais requisitados, e a implementação de serviços em Docker, que apenas foram feitos alguns testes de implementação, e que criariam um ambiente mais estável e de fácil configuração.

Referências

- [Alves. et al., 2021] Alves., L., Carvalhido., T., Cruz., E., and Rosado da Cruz., A. (2021). Using blockchain to trace pdo/pgi/tsg products. In *Proceedings of the 23rd International Conference on Enterprise Information Systems - Volume 2: ICEIS*, pages 368–376. INSTICC, SciTePress.
- [Alves et al., 2022a] Alves, L., Cruz, E. F., Lopes, S. I., Faria, P. M., and da Cruz, A. M. R. (2022a). Towards circular economy in the textiles and clothing value chain through blockchain technology and iot: A review. *Waste Management & Research*, 40(1):3–23. PMID: 34708680.
- [Alves et al., 2022b] Alves, L., Cruz, E. F., and Rosado da Cruz, A. M. (2022b). Tracing sustainability indicators in the textile and clothing value chain using blockchain technology. In *2022 17th Iberian Conference on Information Systems and Technologies (CISTI)*.
- [Alves et al., 2023] Alves, L., Faria, P. M., Cruz, E. F., Lopes, S. I., and Rosado da Cruz, A. M. (2023). Eco-gamification platform to promote consumersrsquo; engagement in the textile and clothing circular value chain. *Sustainability*, 15(6).
- [Alves, 2023] Alves, L. C. V. (2023). Smart contract and web dapp for tracing sustainability indicators in the textile and clothing value chain.
- [AQUINO et al., 2016] AQUINO, A. R. d., PALETTA, F. C., CAMELLO, T. C., MARTINS, T. P., and ALMEIDA, J. R. d. (2016). Sustentabilidade ambiental. *Rede Sirius*.
- [Barrett et al., 2020] Barrett, J., Chase, Z., Zhang, J., Holl, M. M. B., Willis, K., Williams, A., Hardesty, B. D., and Wilcox, C. (2020). Microplastic pollution in deep-sea sediments from the great australian bight. *Frontiers in Marine Science*, 7.
- [Bendo, 2021] Bendo, T. R. (2021). *Desenvolvimento de uma plataforma Web para a rastreabilidade de modelos baseados em agentes*. PhD thesis, Instituto Politécnico de Setúbal.
- [Bierman et al., 2014] Bierman, G., Abadi, M., and Torgersen, M. (2014). Understanding typescript. In Jones, R., editor, *ECOOP 2014 – Object-Oriented Programming*, pages 257–281, Berlin, Heidelberg. Springer Berlin Heidelberg.
- [Biswas et al., 2017] Biswas, K., Muthukkumarasamy, V., and Tan, W. L. (2017). Blockchain Based Wine Supply Chain Traceability System. *Proceedings of the 2017 Future Technologies Conference (FTC)*, pages 56–62.
- [Bourhis et al., 2020] Bourhis, P., Reutter, J. L., and Vrgoč, D. (2020). Json: Data model and query languages. *Information Systems*, 89:101478.
- [CHÁVEZ, 2019] CHÁVEZ, J. D. (2019). Fundamentos de programación en lenguaje pl/pgsql. *IEASS Editores*.

- [Cruz and da Cruz, 2020a] Cruz, E. F. and da Cruz, A. M. R. (2020a). Design science research for is/it projects: Focus on digital transformation. In *15th Iberian Conf. on Information Systems and Technologies (CISTI)*, pages 1–6.
- [Cruz and da Cruz, 2020b] Cruz, E. F. and da Cruz, A. M. R. (2020b). Using blockchain to implement traceability on fishery value chain. In *Proceedings of the 15th International Conference on Software Technologies (ICSOFT 2020)*, pages 501–508.
- [da Cruz and Cruz, 2020] da Cruz, A. M. R. and Cruz, E. F. (2020). Blockchain-based traceability platforms as a tool for sustainability. In *22st International Conference on Enterprise Information Systems (ICEIS 2020)*, volume 2, pages 330–337. SciTePress.
- [da Cruz et al., 2020] da Cruz, A. M. R., Santos, F., Mendes, P., and Cruz, E. F. (2020). Blockchain-based traceability of carbon footprint: A solidity smart contract for ethereum. In *22st International Conference on Enterprise Information Systems (ICEIS)*, volume 2, pages 258–268. SciTePress.
- [Dias et al., 2021] Dias, R., Cardoso, H., Cruz, E. F., and da Cruz, A. M. R. (2021). A blockchain-based platform for reliably tracing political contacts. In *2021 16th Iberian Conference on Information Systems and Technologies (CISTI)*, pages 1–6.
- [ElMessiry and ElMessiry, 2018] ElMessiry, M. and ElMessiry, A. (2018). Blockchain framework for textile supply chain management. In Chen, S., Wang, H., and Zhang, L.-J., editors, *Blockchain – ICBC 2018*, pages 213–227, Cham. Springer International Publishing.
- [Espinoza Pérez et al., 2022] Espinoza Pérez, L. A., Espinoza Pérez, A. T., and Óscar C. Vásquez (2022). Exploring an alternative to the chilean textile waste: A carbon footprint assessment of a textile recycling process. *Science of The Total Environment*, 830:154542.
- [Faridi et al., 2021] Faridi, M. S., Ali, S., Duan, G., and Wang, G. (2021). Blockchain and iot based textile manufacturing traceability system in industry 4.0. In Wang, G., Chen, B., Li, W., Di Pietro, R., Yan, X., and Han, H., editors, *Security, Privacy, and Anonymity in Computation, Communication, and Storage*, pages 331–344, Cham. Springer International Publishing.
- [Fernandes et al., 2022] Fernandes, M. A., Cruz, E. F., and Rosado da Cruz, A. M. (2022). Smart contract and web dapp for traceability in the olive oil production chain. In *2022 17th Iberian Conference on Information Systems and Technologies (CISTI)*.
- [Flanagan, 2004] Flanagan, D. (2004). *JavaScript: o guia definitivo*. Bookman Editora.
- [Fu et al., 2018] Fu, B., Shu, Z., and Liu, X. (2018). Blockchain enhanced emission trading framework in fashion apparel manufacturing industry. *Sustainability, MDPI, Open Access Journal*, 10(4):1–19.
- [Gomes, 2005] Gomes, I. (2005). Sustentabilidade social e ambiental na agricultura familiar. *Revista de biologia e ciências da terra*, 5(1):0.
- [Jia et al., 2020] Jia, F., Yin, S., Chen, L., and Chen, X. (2020). The circular economy in the textile and apparel industry: A systematic literature review. *Journal of Cleaner Production*, 259:120728.
- [KAPLINSKY, 2004] KAPLINSKY, R. (2004). Spreading the gains from globalization : What can be learned from value-chain analysis? *Problems of Economic Transition*, 47(2):74–115.

- [Leach et al., 2005] Leach, P., Mealling, M., and Salz, R. (2005). A universally unique identifier (uuid) urn namespace. Technical report, NON WORKING GROUP.
- [Lourenço, 2021] Lourenço, J. P. (2021). *Desenvolvimento de uma solução integrada de rastreabilidade para a cadeia de valor agroalimentar*. PhD thesis, Escola Superior de Tecnologia e de Gestão de Bragança.
- [Mahmud, 2020] Mahmud, N. M. (2020). *Membuat Front End Website Menggunakan React App dan Material-UI*. Tesis, Politeknik Negeri Bengkalis, Bengkalis.
- [McNeill and Moore, 2015] McNeill, L. and Moore, R. (2015). Sustainable fashion consumption and the fast fashion conundrum: fashionable consumers and attitudes to sustainability in clothing choice. *International Journal of Consumer Studies*, 39(3):212–222.
- [Moreira, 2022] Moreira, A. E. (2022). *Plataforma para comercialização de aves caipiras com rastreabilidade*. B.S. thesis, Universidade Tecnológica Federal do Paraná.
- [Napier and Sanguineti, 2018] Napier, E. and Sanguineti, F. (2018). Fashion merchandisers' slash and burn dilemma: a consequence of over production and excessive waste? *Rutgers Business Review*, 3(2).
- [Oliveira et al., 2021] Oliveira, J., Lima, J. E., da Silva, D., Kuprych, V., Faria, P. M., Teixeira, C., Ferreira Cruz, E., and Rosado da Cruz, A. M. (2021). Traceability system for quality monitoring in the fishery and aquaculture value chain. *Journal of Agriculture and Food Research*, 5:100169.
- [Organization, 2018] Organization, I. L. (2018). Clear cotton. eliminating child labour and forced labour in the cotton, textile and garment value chains: An integrated approach. Technical report, International Labour Organization.
- [Palacios-Mateo et al., 2021] Palacios-Mateo, C., van der Meer, Y., and Seide, G. (2021). Analysis of the polyester clothing value chain to identify key intervention points for sustainability. *Environmental Sciences Europe*, 33(1):2.
- [Panda and Satapathy, 2021] Panda, S. K. and Satapathy, S. C. (2021). Drug traceability and transparency in medical supply chain using blockchain for easing the process and creating trust between stakeholders and consumers. *Personal and Ubiquitous Computing*, pages 1617–4917.
- [Porter's, 1985] Porter's, V. C. M. (1985). What is value chain. *E-Commer.*, pages 1–13.
- [Queiroz et al., 2020] Queiroz, M. M., Telles, R., and Bonilla, S. H. (2020). Blockchain and supply chain management integration: a systematic review of the literature. *Supply chain management: An international journal*, 25(2):241–254.
- [Reffatti et al., 2021] Reffatti, L., Gonçalves de Oliveira, G., and Shibata, M. (2021). Fruit traceability via mobile application. *Comunicata Scientiae*, 12:e3483.
- [Rinaldi et al., 2022] Rinaldi, F. R., Bernardino, C. D., Cram-Martos, V., and Pisani, M. T. (2022). Traceability and transparency: enhancing sustainability and circularity in garment and footwear. *Sustainability: Science, Practice and Policy*, 18(1):132–141.
- [Rusinek and Zhang, 2018] Rusinek, M. and Zhang, H. (2018). Blockchain for a traceable , circular textile supply chain : A requirements approach. In *Software Quality Professional*.

- [Serra and de Moraes, 2007] Serra, M. and de Moraes, G. I. (2007). Tecnologia e sustentabilidade ambiental: desafios e possibilidades para os países periféricos. *Revista Economia & Tecnologia*, 3(2).
- [Shirvanimoghaddam et al., 2020] Shirvanimoghaddam, K., Motamed, B., Ramakrishna, S., and Naebe, M. (2020). Death by waste: Fashion and textile circular economy case. *Science of The Total Environment*, 718:137317.
- [Silva et al., 2019] Silva, F., da SILVA, F., CASTRO, A. d., and Yano, I. (2019). Avaliação da técnica de blockchain na rastreabilidade na agroindústria a sucroenergética. *congresso CNPTIA*.
- [Singh et al., 2020] Singh, A., Rajak, R., Mistry, H., and Raut, P. (2020). Aid, charity and donation tracking system using blockchain. In *2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184)*, pages 457–462.
- [Smelik, 2023] Smelik, A. (2023). Polyester: A cultural history. *Fashion Practice*, 0(0):1–21.
- [Sriramya and Karthika, 2015] Sriramya, P. and Karthika, R. (2015). Providing password security by salted password hashing using bcrypt algorithm. *ARP journal of engineering and applied sciences*, 10(13):5551–5556.
- [Torresi et al., 2010] Torresi, S. I., Pardini, V. L., and Ferreira, V. F. (2010). O que é sustentabilidade?
- [Trigo et al., 2018] Trigo, I. A., Yada, M. M., da Silva Lourençano, L., and de Lima, Y. K. (2018). Uso de tecnologia na rastreabilidade do rebanho de corte. *Revista Interface Tecnológica*, 15(2):381–391.
- [Watson, 2018] Watson, L. (2018). The price of fast fashion. *Nature Climate Change*, 8(1):1.
- [Yu et al., 2020] Yu, C., Xu, X., Yu, S., Sang, Z., Yang, C., and Jiang, X. (2020). Shared manufacturing in the sharing economy: Concept, definition and service operations. *Computers & Industrial Engineering*, 146:106602.