



Politécnico de Leiria  
Escola Superior de Tecnologia e Gestão  
Departamento de Engenharia Informática  
Mestrado em Eng.<sup>a</sup> Informática – Computação Móvel

**AGRO PRECISION – A FRAMEWORK FOR  
PRECISION AGRICULTURE ON AN IOT  
BLOCKCHAIN**

**BERNARDO JOSÉ RIBEIRO FIGUEIREDO**

Leiria, Março de 2023





Politécnico de Leiria  
Escola Superior de Tecnologia e Gestão  
Departamento de Engenharia Informática  
Mestrado em Eng.<sup>a</sup> Informática – Computação Móvel

**AGRO PRECISION – A FRAMEWORK FOR  
PRECISION AGRICULTURE ON AN IOT  
BLOCKCHAIN**

**BERNARDO JOSÉ RIBEIRO FIGUEIREDO**

Projeto realizado sob a orientação de Catarina I. Reis, PhD, Professora da Escola Superior de Tecnologia e Gestão do Politécnico de Leiria.

Leiria, Março de 2023



## AGRADECIMENTOS

---

Quero expressar a minha sincera gratidão à minha orientadora de tese, Catarina Reis pela sua orientação, apoio e paciência ao longo deste projeto. Sem a sua experiência esta tese não teria sido possível.

Gostaria também de agradecer aos membros do Departamento de Informática do Politécnico de Leiria pelas suas contribuições para a minha educação e investigação desta área. O trabalho ao longo do curso e as oportunidades de investigação proporcionadas pelo departamento foram fundamentais para o desenvolvimento dos meus conhecimentos e competências.

Além disso, quero estender o meu apreço à comunidade Hedera e aos apaixonados pelas tecnologias descentralizadas, que têm sido fundamentais para a minha investigação e compreensão do potencial da tecnologia Blockchain no campo da Agro-Precision. Os vossos feedback e desenvolvimento da tecnologia têm sido inestimáveis para delinear a direção da minha investigação e reforçar a importância de explorar as vastas aplicações da tecnologia blockchain.

Também quero reconhecer o apoio inabalável da minha namorada, Beatriz, que tem sido uma fonte constante de inspiração, encorajamento ao longo desta jornada. A sua paciência, compreensão e encorajamento foram inestimáveis, e eu não o teria conseguido sem ela.

Finalmente, quero expressar o meu mais profundo apreço à minha família pelo amor incondicional, apoio e orientação ao longo da minha vida académica e pessoal. Os sacrifícios e encorajamento demonstrado têm sido fundamentais para moldar quem sou hoje, e estarei sempre grato pelo apoio incondicional.



## RESUMO

---

O conceito de Blockchain e DLT (Distributed Ledger Technology) atualmente é aplicado, de forma, a resolver diversos problemas ao nível de gestão de dados, registo de transações ou aplicar conceitos de confiança e confidencialidade em sistemas pouco fidedignos. Deste modo, a tecnologia permite a introdução de soluções simples e completas para a resolução de problemas nos setores financeiros, agrícolas entre outros. A evolução da tecnologia DLT permitiu o aparecimento de conceitos como Smart Agriculture e Agro-Precision, em que ambos pretendem melhorar o setor agricultura aumentando a eficiência e produtividade dos processos de registo, gestão e rastreamento dos produtos agrícolas. Sendo que aumentar o nível de sustentabilidade das práticas agrícolas é um objetivo comum em cada um dos conceitos. Os conceitos surgem para resolver um problema identificado nas cadeias de abastecimento do setor alimentar (food supply chain) que, atualmente as cadeias de abastecimento do sector agrícola são demasiado complexas e apresentam uma falta de rastreabilidade dos seus produtos, desde a produção até ao consumidor final. Além disso, problemas como a existência de múltiplas entidades diferentes de organizações distintas, vão aplicar processos de tratamento de dados ineficientes que aumentam uma falta de transparência do ciclo de vida do produto. O acesso limitado à informação de forma atempada vai gerar problemas de desinformação que pode ter um grande impacto nas operações do dia a dia de todas as entidades envolvidas numa cadeia de abastecimentos. O projeto Agro-Precision, foca no desenvolvimento de uma plataforma web denominada de COW que representa o desenvolvimento de uma solução útil que resolva os problemas descritos, através da aplicação da tecnologia DLT Hedera. A plataforma pretende melhorar a eficiência e transparência da gestão de uma cadeia de abastecimento de bovinos, permitindo apresentar a mesma informação em tempo real para toda a cadeia, aperfeiçoando a rastreabilidade do ciclo de vida de um bovino. Assim sendo, a implementação de tecnologias DLT permite a adição de características como imutabilidade, precisão e resiliência, promovendo a transparência nas cadeias de abastecimento.

**Palavras-chave:** Agro-Precision, Blockchain, DLT, Smart Agriculture, Cadeia de Abastecimento, Rastreabilidade, Transparência, Hedera.



## ABSTRACT

---

The concept of Blockchain and DLT (Distributed Ledger Technology) is currently applied to solve multiple problems in terms of data management, transaction registration or apply concepts of trust and confidentiality in unreliable systems. In this way, the technology allows the introduction of simple and complete solutions to solve problems in the financial, agricultural, and other sectors. The evolution of DLT technology has allowed the emergence of concepts such as Smart Agriculture and Agro-Precision, both of which aim to improve the agricultural sector by increasing the efficiency and productivity of agricultural product registration, management, and tracking processes. Increasing the level of sustainability of agricultural practices is a common goal in each concept. The previous concepts arise to solve distinct problems identified in the food supply chains like, agricultural sector is too complex and lack of traceability of their products, from production to the final consumer. In addition, problems such as the existence of multiple different entities from different organisations will apply inefficient data processing processes that increase a lack of transparency of the product life cycle. Limited access to information in a timely manner will generate problems of misinformation that can have a major impact on the day-to-day operations of all entities involved in a supply chain. The Agro-Precision project focuses on the development of a web platform called COW that represents the development of a useful solution that solves the problems described, through the application of DLT Hedera technology. The platform aims to improve the efficiency and transparency of the management of a cattle supply chain, allowing to present the same information in real time for the whole chain, improving the traceability of the life cycle of a bovine. Therefore, the implementation of DLT technologies allows the addition of features such as immutability, accuracy, and resilience, promoting transparency in supply chains.

**Keywords:** Agro-Precision, Blockchain, DLT, Smart Agriculture, Supply Chain, Traceability, Transparency, Hedera.



## INDICE

---

1	Introdução	1
1.1	Contexto . . . . .	3
1.2	Problemas e Motivação . . . . .	4
1.3	Objetivos . . . . .	6
1.4	Estrutura do Documento . . . . .	7
2	Estado de Arte	9
2.1	DLT & Blockchain . . . . .	10
2.2	Agricultura de Precisão . . . . .	12
2.3	Exploração Tecnológica . . . . .	17
2.3.1	Helium . . . . .	17
2.3.2	IOTA . . . . .	19
2.3.3	Hedera . . . . .	20
3	Arquitetura	25
3.1	Perfis de utilização . . . . .	25
3.2	Fluxos de Ações . . . . .	26
3.3	Arquitetura Geral . . . . .	33
3.3.1	Modelo de Dados . . . . .	35
3.3.2	API . . . . .	36
3.3.3	Smart Contracts . . . . .	39
3.3.4	Frontend . . . . .	41
4	Desenvolvimento	43
4.1	Autenticação e Registo de Utilizador . . . . .	43
4.2	Bovino . . . . .	44
4.3	Campo . . . . .	44
4.4	Consultas . . . . .	45
4.5	Leilão . . . . .	46
4.6	Dashboard e Perfil . . . . .	47
4.7	Testes . . . . .	48
5	Conclusões	49
5.1	Resultados . . . . .	50
5.2	Limitações . . . . .	51

INDICE

5.3 Trabalho Futuro . . . . .	51
Bibliografia	53
Apêndices	
A Anexo A	59
B Anexo B	65
C Anexo C	71
Declaração	73

## LISTA DE FIGURAS

---

Figura 1	Representação das distintas fases do projeto . . . . .	5
Figura 2	Representação dos conceitos Permissioned Blockchain vs Blockchain vs DLT . . . . .	11
Figura 3	Caracterização das soluções FSC & DLT . . . . .	14
Figura 4	Lista de Requisitos Funcionais . . . . .	27
Figura 5	Diagrama de Sequência - Farmer/Buyer - Criar Bovino . . .	28
Figura 6	Diagrama de Sequência - Farmer - Atualizar Campo Agrícola	30
Figura 7	Diagrama de Sequência - Veterinário - Criar Consulta . . . .	32
Figura 8	Diagrama de Sequência - Comprador - Listar Leilões . . . .	33
Figura 9	Representação da Arquitetura Geral . . . . .	34
Figura 10	Modelo de Domínio UML . . . . .	36
Figura 11	Modelo de Entidade e Relação . . . . .	37
Figura 12	Representação da comunicação dos contratos na rede Hedera	40
Figura 13	Detalhe do Bovino - Agricultor . . . . .	44
Figura 14	Listagem de Campos Agrícolas - Agricultor . . . . .	45
Figura 15	Lista de Marcações de Consultas - Veterinário . . . . .	46
Figura 16	Detalhe do perfil de utilizador . . . . .	47
Figura 17	COW API - Swagger . . . . .	65
Figura 18	COW API - Rotas . . . . .	66
Figura 19	Landing Page . . . . .	67
Figura 20	Página de Autenticação . . . . .	67
Figura 21	Página de Registo . . . . .	68
Figura 22	Criação de NFT do Bovino . . . . .	68
Figura 23	Detalhe NFT Bovino - Parte 2 . . . . .	69
Figura 24	Detalhe NFT Bovino - Parte 3 . . . . .	69
Figura 25	Lista de Bovinos . . . . .	70
Figura 26	Listagem de Marcações - Farmer . . . . .	70
Figura 27	Representação dos Testes de Usabilidade . . . . .	72



## ACRÓNIMOS E SIGLAS

---

ABFT Asynchronous Byzantine Fault Tolerance.  
ABI Application Binary Interface.  
API Application Programming Interface.  
ATTIA Australian Tea Tree Industry Association.

BD Base de Dados.  
BNB Binance Coin.

CDBC Central Bank Digital Currency.

DAG Directed Acyclic Graph.  
DAGS Directed Acyclic Graphs.  
DApps Decentralized Applications.  
DEFI Decentralized finance.  
DLT Distributed Ledger Technology.  
DTO Data Transfer Object.

FSCs Food Supply Chains.

GTIN Global Trade Identification Numbers.

HCS Hedera Consensus Service.  
HFS Hedera File Service.

IA Inteligência Artificial.  
IDE Integrated Development Interface.  
IoT Internet of Things.  
IT Information Technology.

JWT JSON Web Token.

M2M Machine-to-Machine.

MER Modelo de Entidade e Relação.

NFT Non-fundgible Token.

PLC Product Life Cycle.

POC Proof of Coverage.

POS Proof of Stake.

POW Proof of Work.

QR Quick Response.

RFID Radio Frequency Identification.

SDK Software Development Kit.

SPA Single Page Application.

UI User Interface.

UML Unified Modeling Language.

## INTRODUÇÃO

---

Cada vez mais, é notório a evolução gradual da tecnológica, também gradualmente, as empresas dedicam esforço e determinação no tratamento e recolha de dados associados a um utilizador ou uma ação realizada pelo próprio. Cada registo de informação que o utilizador expõe, qualquer operação que o próprio pratique, dependendo do setor a transação vai ser frequentemente utilizada de modo, a se retirar um dado. As empresas que apresentam produtos baseados em registos ou transações, caso permitam a tomada de decisão após uma análise de dados, e realizem um tratamento e recolha de dados têm por consequência um desenvolvimento dos produtos e melhoramento experiência do utilizador elevado consoante os métodos aplicados. No setor agrícola onde existem diversas entidades, múltiplos produtos, e os negócios que são gerados dentro do setor são imensuráveis, é necessário incluir transparência, confiança, rastreabilidade e auditoria aos dados transacionados, aplicando múltiplos processos de segurança, que impõe a veracidade da informação, permitindo o desenvolvimento de um sistema escalável [28, 21]. As características evidenciadas anteriormente são conceitos base da tecnologia DLT e Blockchain, que atualmente apresentam e disponibilizam diversos tipos de soluções.

A tecnologia Blockchain nos últimos anos tem apresentado avanços significativos e muito do trabalho desenvolvido advém do interesse nas criptomoedas e o valor monetário que representam. A tecnologia é identificada como um livro-razão digital (uma fonte de veracidade) que permite o registo seguro e transparente de dados. Uma das principais características que apresenta é a descentralização dos dados que é mantida geralmente por uma rede de computadores, e não por uma única entidade. Os dados de uma Blockchain estão organizados em blocos, que estão ligados entre si numa cadeia cronológica, com cada bloco contendo um hash criptográfico do bloco anterior [17]. Esta funcionalidade de imutabilidade torna os dados presentes numa cadeia de blocos tolerantes à manipulação e resistentes à adulteração de dados mal-intencionada, pois qualquer alteração a um bloco exigiria a alteração de todos os blocos subsequentes na cadeia, o que é computacionalmente inviável (ou pelo menos, muito difícil).

Atualmente, empresas como a BNB Smart Chain (empresa criada pela Binance) apresentam sistemas que melhoram e aumentam a escalabilidade dos serviços de uma Blockchain. Desta forma funcionalidades como a capacidade das transações, tempos de execução de transações, entre outras, apresentam novas vantagens em

relação aos serviços tradicionais. Um fator crucial para a aplicação da tecnologia Blockchain nos diversos setores é a forma de governação existente, proporcionando que as decisões e informações sejam partilhadas de forma transparente e inclusiva. Os progressos no desenvolvimento de características como escalabilidade, interoperabilidade, privacidade, sustentabilidade e governação, são conceitos que permitem alargar os casos de uso [5].

A acompanhar esta evolução, o setor agrícola apresenta consideráveis evoluções na área da agricultura de precisão, através da inclusão de tecnologias que pretendem otimizar a eficiência dos processos, e aplicação de recursos naturais reduzindo o desperdício de recursos [17]. Estes avanços verificam-se devido a ser um setor muito lucrativo, que influencia inúmeros negócios e múltiplas entidades. Estas características surgem devido à complexidade do inerente do setor, devido à existência de uma rede de intermediários vasta, onde cada intermediário possui um processo próprio com regras específicas, e um mecanismo privado de partilha dados. Este tipo de abordagem apresenta diversos problemas, desde a falta de transparência e inexistência de rastreabilidade dos produtos agrícolas [8]. Além disso, questões como a privacidade dos dados, falta de processos de gestão e ausência de alertas, implicam um acesso limitado e tardio à informação real, conclusão essa que tem um grande impacto em toda a cadeia [31].

De modo a enfrentar os diversos desafios existentes no setor agrícola, é necessário encontrar e executar uma solução que apresente uma abordagem heterogénea, simples, que aplique as vantagens das tecnologias atuais[38]. Encontrar um modelo inovador, que seja possível de ser adotado nas diversas cadeias de fornecimento agrícola é um dos objetivos iniciais, pretendendo desenvolver e apresentar um sistema transparente, eficiente e sustentável.

O objetivo de aplicar a tecnologia Blockchain ao setor agrícola, pretendendo criar uma solução que foque no conceito Agro-Precision é algo extremamente complexo devido à variedade de entidades existentes no setor. As tecnologias DLT (Distributed Ledger Technology) que apresentam uma rede pública descentralizada como o Hedera é uma das possíveis soluções. A tecnologia Hedera inclui um algoritmo de consenso denominado de Hashgraph e disponibiliza uma plataforma de acesso rápido, simples, segura e escalável. Este algoritmo tem como objetivo ser superior ao nível da eficiência e segurança, em relação à Blockchain tradicional. As tecnologias deste formato centram-se no processamento de transações o mais rápido, mantendo o elevado nível de segurança na solução.

## 1.1 CONTEXTO

Devido à complexidade do setor agrícola, a existência de problemas de transferência de dados, falta de processos heterogêneos entre entidades de uma cadeia de fornecimento, é notório que estes problemas prejudicam qualquer tipo de desenvolvimento analítico nas cadeias de fornecimento agrícola. É também necessário ter em conta que a sociedade atual, se preocupa cada vez mais com a sua saúde, os alimentos que comem, a forma como os alimentos são produzidos, avaliados, transportados, entre outras preocupações. O número de pessoas que se sensibilizam e pretendem possuir um estilo de vida mais saudável é considerável, portanto é natural existir uma maior preocupação sobre o ciclo de vida de determinado produto agrícola, quer seja uma semente ou um bovino.

Tendo em conta os problemas identificados e a dificuldade inerente das supply chains de partilhar e difundir a informação perante todas as entidades. O projeto Agro-Precision tem como foco principal as cadeias de fornecimento bovino, sendo necessário perceber o ciclo de vida bovino, quais as entidades que estão diretamente associadas a este processo. Sendo necessário desenvolver uma plataforma digital que permita a gestão de bovino, permitindo a partilha de informação de uma forma segura e transparente para toda a cadeia. Além disso, a adição de funcionalidades que permitam definir patamares de organização para os diversos tipos de entidades existentes numa cadeia de alimentação é uma inevitabilidade.

Tendo em conta os problemas identificados e a dificuldade inerente das supply chains de partilhar e difundir a informação perante todas as entidades. O projeto AGRO-PRECISION tem como foco principal as cadeias de fornecimento bovino, sendo necessário perceber o ciclo de vida bovino, quais as entidades que estão diretamente associadas a este processo. Sendo necessário desenvolver uma plataforma digital que permita a gestão de bovino, permitindo a partilha de informação de uma forma segura e transparente para toda a cadeia. Além disso, a adição de funcionalidades que permitam definir patamares de organização para os diversos tipos de entidades existentes numa cadeia de alimentação é uma inevitabilidade.

O projeto Agro-Precision apresenta através da plataforma COW, a oportunidade de lidar com uma tecnologia recente que introduza a inovação da tecnologia Blockchain, apresentando diferenças significativas das Blockchains tradicionais, como Ethereum, Ripple, Dash, Hyperledger Fabric ou Hyperledger Sawtooth. Sendo que a Blockchain aplicada ao projeto deve ser vista como a única fonte de verdade.

O projeto foi desenvolvido em 3 etapas/fases:

- **Exploração e Desenvolvimento de Protótipos:** A fase inicial focou-se na exploração de tecnologias DLT, através da execução de uma análise das

vantagens e desvantagens de cada. A investigação prosseguiu, de modo a entender a viabilidade de aplicar a tecnologia no setor agrícola e quais as tecnologias inseridas em soluções no setor. A preocupação de integrar uma nova tecnologia implicou a necessidade de desenvolvimento de múltiplos projetos com as tecnologias distintas seguintes: IOTA, Helium e Hedera.

- **Exploração e Procura de Casos de Uso:** Existiu uma fase de investigação sobre os casos de uso existentes no setor, e como as aplicações existentes no mercado são aplicadas e de que forma solucionam os problemas. Os resultados da investigação deram origem a um artigo científico focado nas soluções existentes no mercado e como a plataforma COW soluciona os problemas da cadeia de fornecimento dos bovinos.
- **Desenvolvimento da plataforma COW:** A última fase centrou-se no desenvolvimento da prova de conceito, através da aplicação da tecnologia Hedera no setor Agrícola.

A figura 1 representa no formato do diagrama BPMN (Business Process Modeling Notation), as diferentes fases do projeto. Primeiramente foi realizada uma análise e definição dos objetivos iniciais do projeto, permitindo perceber o foco do projeto e as entidades existentes numa cadeia de fornecimento bovino. Permitindo imediatamente a um levantamento de soluções, através uma pesquisa intensiva sobre o estado de arte existente na área do Agro-Precision, IoT (Internet of Things) e Blockchain. O levantamento de soluções permitiu identificar diversos tipos de tecnologias DLT sendo necessário uma avaliação prática das tecnologias. Este levantamento de soluções e análise do estado de arte permitiu a elaboração de um artigo científico acerca da plataforma. Após o desenvolvimento da plataforma e aplicação dos padrões das diferentes tecnologias, surge a elaboração do presente relatório.

É importante dar um destacar, que a análise inicialmente realizada ao setor revelou que não existe qualquer tipo de informação específica sobre modelos de dados, casos de uso, soluções ou quaisquer provas de conceito que incluam os conceitos de Agro-Precision, IoT e DLT.

## 1.2 PROBLEMAS E MOTIVAÇÃO

A existência de diversas entidades que apresentam uma responsabilidade no setor agrícola, é considerado um dos principais problemas para a falta de rastreabilidade no setor. As cadeias de abastecimento, apresentam dificuldades como a falta de transparência, rastreabilidade e confiança no sistema [15]. A existência de uma estrutura complexa, com múltiplos intermediários que estão distribuídos de forma dispersa e apresentam processos distintos na transmissão da informação resulta

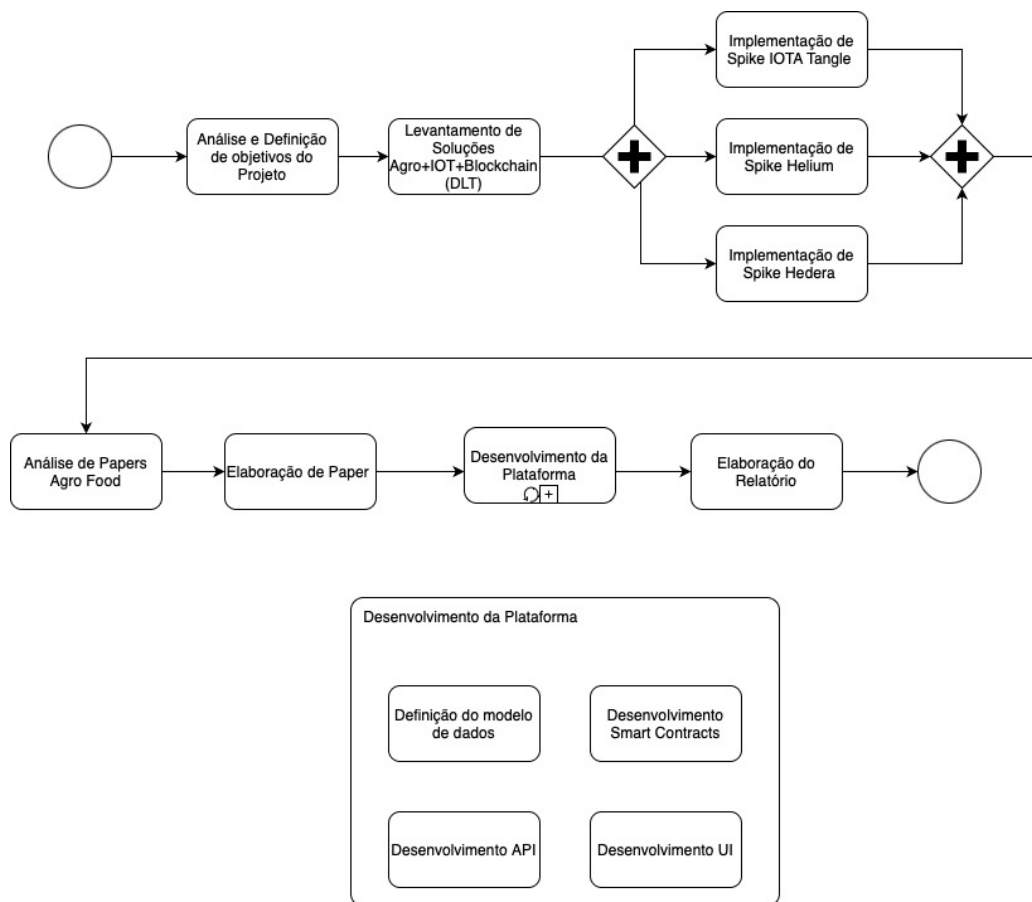


Figura 1: Representação das distintas fases do projeto

numa partilha de dados incompleta e defeituosa [14]. As dificuldades das cadeias de fornecimento são evidentes, e a complexidade de desenvolver métodos de rastreio que acompanhem os produtos desde a exploração agrícola até ao consumidor final (*seed to table*) é um processo extremamente complexo.

É necessário também ter em conta que a falta de transparência pode levar a problemas de segurança alimentar, problemas de controlo de qualidade, e fraude. Por exemplo a redução do prazo de validade dos alimentos devido a graves atrasos no transporte ou partilha de informação desatualizada, e inclusiva. Estas podem também fazer aumentar os custos para os consumidores.

É importante referir que as cadeias de fornecimento agrícola são conhecidas por provocar um impacto negativo no ambiente, devido à desflorestação que determinados processos envolvem, emissões de gases com efeito de estufa, e ainda o uso excessivo de água potável. Assim, torna-se necessário introduzir práticas mais eficientes, sustentáveis e amigas do ambiente. Por último é de referir que questões como a falta de rastreabilidade dificultam a identificação da fonte de doenças de origem alimentar, tornado o processo de conter o surto e prevenir incidentes futuros mais complicados.

A existência de produtos de má qualidade pode levar à insatisfação do consumidor e reduzir as vendas, o que pode ter impacto na rentabilidade de toda a cadeia de abastecimento. Assegurar o cumprimento de regras pode ser um desafio e acrescentar custos significativos à cadeia de abastecimento. O facto de a cadeia de abastecimento apresentar múltiplos intervenientes diferentes, desde agricultores, distribuidores e retalhistas implica a necessidade de coordenação entre todos. A falta de coordenação entre estes intervenientes pode levar a ineficiências, desperdícios e má comunicação.

### 1.3 OBJETIVOS

O objetivo inicial é desenvolver uma plataforma genérica, escalável, que permita introduzir conceitos de transparência, rastreabilidade e eficiência [7] entre todos os utilizadores da plataforma que representam os múltiplos intervenientes de uma cadeia de abastecimento. A plataforma tem como foco as cadeias de abastecimento dos bovinos, sendo o motivo principal para o nome da plataforma ser COW.

A solução que a plataforma COW pretende é estes problemas de falta de transparência na comunicação de uma cadeia de abastecimento, de forma a assegurar a sustentabilidade e rentabilidade do setor, bem como satisfazer a procura crescente de alimentos seguros e sustentáveis. Deste modo, aplicando tecnologias DLT como Blockchain, é possível aumentar a transparência, rastreabilidade e eficiência de toda a cadeia. A plataforma, pretende que todos os intervenientes na cadeia de

abastecimento, acedam a um registo partilhado e imutável de todas as transações e operações de produtos. Assim aumentando a transparência do sistema, aumentando o número de mecanismos que previnem fraude, contrafação, entre outros problemas.

A plataforma tem como objetivo fornecer informações precisas e em tempo real sobre os bovinos desde a criação, transporte, alterações de localização, compra e venda de bovinos e ainda permitir registos de marcações e consultas. Desta forma, os diversos intervenientes conseguem tomar decisões informadas e otimizar as operações agrícolas.

A plataforma disponibiliza diversos mecanismos que permitem a rastreabilidade dos produtos, desde a exploração até ao consumidor final. Estas funcionalidades aumentam a segurança alimentar e controlo de qualidade [19, 5, 9]. Através do registo detalhado da origem do produto, locais de passagem, e alterações de dono, permite aos interessados, de uma forma rápida e simples identificar e isolar potenciais problemas. Deve-se ter em conta que os processos enumerados na plataforma são automáticos reduzindo a quantidade de *bottlenecks* na transferência de informação.

A aplicação de *smart contracts* permite a execução automática de transações e desencadear ações pré-definidas reduzindo a necessidade da intervenção humana. A possibilidade de automaticamente realizar transações e desencadear ações baseadas em regras pré-definidas, reduz a necessidade de intervenção manual, sendo um mecanismo necessário para alcançar eficiência nas cadeias de abastecimento.

Além disso, a plataforma pretende garantir a segurança e qualidade dos produtos agrícolas utilizados, enaltecendo a importância dos conceitos da saúde e satisfação do consumidor final como uma das prioridades. Quando os consumidores confiam que os alimentos que compram são seguros e de alta qualidade, tendem comprá-los novamente, promovendo toda a cadeia de abastecimento.

#### 1.4 ESTRUTURA DO DOCUMENTO

O presente relatório está dividido em cinco capítulos, sendo que, no segundo é realizado um enquadramento do projeto com o trabalho relacionado, explorando os conceitos base, fundamentos nos quais o projeto se fundamenta e metodologia de investigação. É também abordado o caso de estudo sobre o qual o projeto foi desenvolvido.

No terceiro capítulo é descrita a arquitetura utilizada para o desenvolvimento da plataforma, sendo realizado uma descrição detalhada de cada um dos componentes.

No capítulo quatro é descrita a implementação da plataforma COW, explorando funcionalidades e a justificação das decisões tomadas.

## INTRODUÇÃO

O relatório termina com o quinto capítulo, a conclusão, onde é realizado um balanço final do desenvolvimento do projeto, bem como a concretização dos seus objetivos. Neste capítulo, é ainda efetuada uma reflexão crítica do trabalho desenvolvido e indicado o trabalho futuro a realizar.

ESTADO DE ARTE

---

Embora as soluções DLT mais conhecidas sejam focadas nas moedas criptográficas, é essencial lembrar que elementos e conceitos base desta tecnologia, são pretendidos em diferentes setores. Ao abordar características como o armazenamento de dados distribuído, partilha de dados entre entidades distintas, segurança e privacidade, são de extrema importância para setores como finanças, agricultura, cuidados de saúde e logística [33]. A tecnologia Blockchain aplicada nas FSCs (food supply chains) potência a partilha de dados verdadeiros e em tempo real para toda a cadeia [30, 24].

A adoção da Bitcoin tornou a tecnologia DLT e Blockchain, um tema mais discutido na indústria e no meio académico. A aplicação de técnicas criptográficas, transações, e bens digitais protegidos contra manipulação e roubo [23]. É inevitável compreender que os desenvolvimentos executados no setor da agricultura, gerou conceitos como a Agro-Precision. O conceito de *Smart Farming* surge com uma abordagem alternativa às diretrizes dos campos agrícolas na era das alterações climáticas, no auge da Indústria 4.0. O conceito tem como foco permitir o desenvolvimento de sistemas de gestão de segurança alimentar, através do uso combinado de conceitos avançados como IoT, Big Data, e Cloud Computing para controlar os campos, analisar o crescimento das culturas, entre outros parâmetros, de modo a fornecer informação precisa para auxiliar qualquer entidade à tomada de decisões.

Uma cadeia de abastecimento alimentar tem como base ser um sistema seguro, prático e sustentável, englobando todos os interessados. A ideia é desenvolver um sistema focado no rastreio e autenticação da informação ao longo de todo o ciclo de vida [17, 31, 38, 35]. É fundamental identificar e abordar fontes de contaminação, contribuindo para uma gestão sustentável nas cadeias agroalimentares. A falta de transparência e rastreabilidade nas cadeias de abastecimento alimentar suscita preocupações entre consumidores e partes interessadas sobre a credibilidade da informação alimentar, qualidade alimentar e segurança [19, 5, 9]. A falta de registos, falta de digitalização, ausência de normas e processos de partilha de informação são alguns dos desafios mais críticos, que podem ser enfrentados com tecnologias inovadoras, tais como IoT, DLT e Blockchain.

A rastreabilidade do percurso dos alimentos e a credibilidade da informação alimentar são aspetos críticos das cadeias de abastecimento agrícola e alimentar. Os consumidores e os participantes da cadeia alimentar precisam de ser totalmente informados sobre o PLC (Product Life Cycle) do produto [39, 22] para assegurar que

os produtos são seguros, sustentáveis e de qualidade. Contudo, os sistemas atuais de rastreabilidade alimentar não são eficazes para criar mecanismos de confiança entre os participantes na cadeia de rastreabilidade. A gestão da informação agroalimentar necessita de ser realizada de forma segura e eficaz.

Para responder a estes desafios, é explorado o conceito das tecnologias DLT e Blockchain, através da análise de soluções e a exploração das características de cada.

## 2.1 DLT & BLOCKCHAIN

O conceito DLT significa simplesmente uma base de dados descentralizada, gerida por várias entidades, onde cada participante tem uma cópia idêntica do estado da base de dados, sendo atualizado de forma síncrona. Caso num sistema distribuído as entidades não confiam completamente em todos os intervenientes denominado de *trustless environment*, é necessário aplicar um mecanismo de consenso que valide o conteúdo antes de ser partilhado. Por outras palavras, as atualizações do estado não são efetuadas por uma autoridade central, mas, por consenso entre as partes, de acordo com um conjunto de regras ou procedimentos aceites por todos.

As características da tecnologia DLT, são a resiliência, integridade, anonimato, descentralização e controlo autónomo, que pretendem promover a adoção desta tecnologia em vários setores [4]. A primeira aplicação real em Blockchain surge em 2009, logo após Satoshi Nakamoto ter publicado um *white paper* expondo o funcionamento da primeira moeda criptográfica a Bitcoin [23]. Desde então, a tecnologia tem sido aplicada a outros domínios como o Agrícola, Eleições, Economia, Logística, entre outros.

A descentralização de um sistema DLT é considerada uma das suas principais características, cujo significado é a ausência de uma autoridade central [30]. Isto torna o sistema mais resistente a ataques e adulteração, uma vez que não existe um ponto de falha único que possa ser atacado ou manipulado.

A tecnologia Blockchain sendo uma forma de um distributed ledger design, surge devido ao facto de uma rede distribuída se basear na criptografia e algoritmos para realizar as funções básicas. A tecnologia utiliza um par de chaves pública e privada que permite tanto o hashing, como a assinatura dos dados dentro dos blocos [36]. Os utilizadores confiam numa rede distribuída peer-to-peer, que valida as transações antes de serem inseridas. Quando uma transação é registada, é também assinada de forma criptográfica, numa abordagem sequencial.

A figura 2 representa a dimensão do conceito entre as tecnologias DLT e Blockchain. Sendo possível de identificar que a tecnologia de Blockchain é um típico específico de

DLT. Logo todas as Blockchains são DLT, mas nem todas as DLTs são Blockchains. Numa blockchain, as transações são agrupadas em blocos e adicionadas à cadeia numa ordem linear e cronológica. Em que cada bloco está ligado ao bloco anterior na cadeia, originando uma cadeia de blocos.

Além disso, a tecnologia DLT e Blockchain aplicam diferentes mecanismos de consenso para validar transações. O consenso numa Blockchain é tipicamente alcançado através de mecanismos como PoW (Proof of Work) ou PoS (Proof of Stake) [36], onde os participantes na rede competem para resolver problemas criptográficos para validar transações. O consenso numa DLT pode ser atingido através de sistema votação ou baseados na reputação, por exemplo Hasgraph e sistemas DAG (Directed Acyclic Graphs).

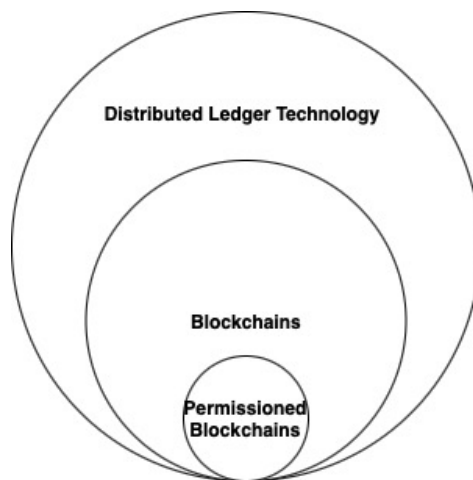


Figura 2: Representação dos conceitos Permissioned Blockchain vs Blockchain vs DLT

De seguida são apresentadas as principais características que uma implementação da tecnologia Blockchain apresenta em soluções no setor agrícola:

1. **Rede descentralizada, resiliente e *trustless*:** Uma Blockchain não apresenta um módulo centralizado, portanto, a remoção ou perda de qualquer nó da rede não afeta o funcionamento de todo o sistema, esta característica proporciona uma excelente robustez [25] e tolerância a falhas. Os dados são partilhados entre os participantes, validados na rede através de assinaturas digitais e mecanismos de *hashing*, sem necessidade de existir uma autoridade centralizada e de confiança mútua [4].
2. ***Smart Contracts* e rastreabilidade:** As transações numa Blockchain podem ser melhoradas através do uso smart contracts. A rede Ethereum [2], apresenta um ambiente dedicado ao desenvolvimento com smart contracts. Esta aplicação permitiu a adição de regras comerciais podem implantadas na Blockchain, permitindo aos participantes rastrear processos comerciais e validar regras contratuais. A estipulação de regras de negócio pode-se comportar de forma automática sem a validação ou intromissão de um elemento da rede [4].

3. **Transparência e pseudo anonimato:** No caso de Blockchains públicas, cada transação é visível por todos os nós, e cada nó participante pode ser anônimo. Quanto aos utilizadores finais identificados com carteiras digitais, também podem ser anónimos, mas devido à assinatura das transações, a *distributed ledger* mantém um registo rastreável de cada transação do utilizador.
4. **Mecanismo de consenso:** O mecanismo de consenso é a forma como todas as entidades determinam a validade de um registo. No exemplo da Bitcoin, isto é conseguido através de uma prova criptográfica, minimizando a possibilidade de fraude de dados no processo de rastreabilidade.
5. **Manipulação de dados:** O problema da duplicação de registos, é resolvido pela Blockchain, a Bitcoin indica que a *ledger* deve ser a única fonte de verdade, à prova de adulteração. Como esperado, a imutabilidade dos dados é uma característica extremamente importante em diversos setores que requerem rastreabilidade como a base para o seu funcionamento.

Contudo, o progresso da tecnologia DLT e Blockchain, também apresenta algumas limitações e desafios técnicos que especificam inúmeros condicionamentos à adoção da tecnologia. Alguns dos desafios que impactam diretamente a adoção da tecnologia são:

- **Privacidade:** Cada nó da rede mantém o histórico das transações da rede.
- **Custos e Latência:** O processamento da Blockchain estipula que todo o histórico de transações é replicado em todos os nós, sendo computacionalmente caro.
- **Segurança:** A Blockchain utiliza criptografia do tipo chave privada e pública para a autenticação e execução de transações. Se um nó perder ou publicar involuntariamente a sua chave privada, o sistema não tem mecanismo adicional de segurança.
- **Flexibilidade:** As características de garantir a integridade das transações, pode funcionar como uma barreira à utilização de casos que exijam alterações constantes nas transações.
- **Governança:** A arquitetura descentralizada da Blockchain pode ser uma limitação para o controlo e administração global para organizações baseadas na supervisão.

## 2.2 AGRICULTURA DE PRECISÃO

Com o intuito de avaliar a qualidade do produto passo de um FSC, é decisivo nas distintas fases implementar práticas de rastreabilidade adequadas. A implementação

de sistemas automáticos de recolha de dados é dispendiosa e as distintas entidades tornam difícil a sua implementação. Contudo, a globalização do comércio alimentar obriga os intervenientes nas cadeias de abastecimento, por exemplo, agricultores, fabricantes, retalhistas e distribuidores, a adotar normas de rastreabilidade [26, 37].

As operações da cadeia de abastecimento, podem ser divididas nas seguintes fases:

- **Fase 1 - Agricultura:** É o primeiro passo no processo de rastreabilidade agroalimentar. As informações a rastrear podem ser desde as condições do ambiente agrícola (por exemplo, qualidade do solo, água, temperatura e qualidade da humidade), pessoal agrícola, data, hora, origem, irrigação, fertilização e pesticidas [6, 10]. Os dispositivos IoT podem ajudar na recolha e transmissão dos dados de rastreabilidade necessários no processo comercial.
- **Fase 2 - Colheita:** Os agricultores atualmente armazenam as informações sobre a colheita, incluindo data, hora e escalas de peso [17, 24]. Após a colheita, a sua maioria são transportados para as fábricas de processamento [4, 6].
- **Fase 3 - Processamento:** A informação desta fase inclui as condições de processamento, equipamento, tempo, transformações, informação sobre embalagens, método de desinfeção, operadores, e informação sobre etiquetas do produto final [14].
- **Fase 4 - Logística e Armazenamento:** O armazenamento deve ser eficaz e o microambiente necessita de ser monitorizado (temperatura, RH, O<sub>2</sub> e CO<sub>2</sub>) através de sensores IoT com o propósito de obter informações relevantes sobre logística e armazenagem [17].
- **Fase 5 - Consumir:** A informação apresentada ao consumidor deve incluir nome do produto, tempo de venda, prazo de validade, preço, entre outros. Os consumidores devem facilmente verificar o histórico dos produtos antes de comprar.

Devemos ter em conta que, a adição de sensores numa FSC animal permite monitorizar e recolher as informações sobre a logística, ambiente de armazenagem, transportes utilizados, parâmetros ambientais, local de armazenagem, tempo de armazenagem e processo do operador. Desta forma, ao armazenar na Blockchain toda a informação é partilhada de forma completa e transparente.

Sendo que o setor agrícola é imenso, é possível presumir a existência de projetos reais e projetos de investigação, permitindo avaliar os métodos utilizados e quais as tecnologias DLT que permitem o desenvolvimento de uma FSC.

O desafio de aplicar um sistema escalável nas FSCs com a tecnologia Blockchain é algo que possui uma complexidade inerente e que deve ser realizado com precaução [17, 31, 38, 35]. O desafio surge na velocidade das transações e quantidade de dados

a analisar. Além disso, devem ser consideradas as interações importantes para a cadeia, de modo a reduzir as interações com a Blockchain, devido aos possíveis custos e tempo das operações.

Considerando esta informação, após uma profunda análise de diversas soluções, as mais difundidas são Hyperledger Sawtooth [12, 11], Hyperledger Fabric [12], Ethereum [12], Multichain [27], Solana [27], R3 Corda [27], e Quorum [27]. A figura 3 apresenta das soluções encontradas.

PROJECT	REAL USE CASE?	TECHNOLOGY	IOT?
<b>Geora</b>			
Tea Tree Oil			Yes
Cotton Finance	Yes	Geora API	NA
P.N.G Pigs			Yes
<b>Agridigital</b>			
Scheimer Farms			
Martin Farms			
Gnabro Pastoral Company	Yes	NA	Yes
GoolGowi			
<b>Tuna Tracking</b>	Yes	NA	Yes
<b>Ambosus - GestOlive</b>	Yes	NA	Yes
<b>Connecting Food</b>			
Herta			
Axéreal	Yes	Connecting Food Apps	NA
General Mills			
<b>Yara - Portato Farmer</b>	Yes	Yara Solution	Yes
<b>Ripe.io - Limonik</b>	Yes	Ripe.io API	Yes
<b>EIT Food</b>			
IFishCan	Yes		Yes
SAIFE	No	EIT Food API & AI	Yes
<b>AgriBlockIoT 1</b>	No	Hyperledger Sawtooth	Yes
<b>BB Soybean Trace</b>	No	Ethereum EVM	Yes
<b>Ingredia</b>	NA	Ingredia Apps	No

Figura 3: Caracterização das soluções FSC & DLT

A Geora apresenta três projetos que destacam diferentes tipos de processos agrícolas [18]. As soluções apresentadas focam na rastreabilidade da indústria australiana de óleo de árvore de chá, permitindo utilizar uma Blockchain para permitir a rastreabilidade e gestão dos certificados de cada produto [3]. Os restantes projetos utilizam *smart contracts* no registo de informação sobre o desenvolvimento de porcos e produção de algodão, integrando uma aplicação móvel e etiquetas RFID que permitem aos agricultores criar registos digitais rastreáveis e verificar nos seus produtos.

A AgriDigital é uma empresa dedicada à digitalização e cultivo de cadeias de abastecimento de cereais para que todas as entidades envolvidas como agricultores, comerciantes, e consumidores possam operar com confiança [16].

O projeto acadêmico AgriBlockIoT, é uma solução de rastreabilidade totalmente descentralizada, que permite a gestão da cadeia de abastecimento Agri-Food, que integra dispositivos IoT que produzem e consomem dados digitais ao longo da cadeia. A solução aplicada em Hyperledger Sawtooth, tem como objetivo recriar uma estrutura automatizada que permita o registo e armazenamento dos dados relacionados com o crescimento de alimentos, por exemplo, batata e trigo [13]. A proposta promove um cenário de rastreabilidade certificada dos alimentos ao longo de toda a cadeia.

O projeto o BB Soybean Trace foi desenvolvido com o objetivo de demonstrar a utilização de Blockchain e *smart contracts* de Ethereum para rastrear, seguir e permitir uma integração sem descontinuidades das transações comerciais e fluxos de trabalho na cadeia de abastecimento. A solução proposta concentrar-se na utilização de *smart contracts* executados de forma automática na rede pública Ethereum. As principais partes interessadas na blockchain incluem a Seed Company, Seed Company, The Farmer, Grain Elevator, Grain Processor, Distributor, Retailer, o End Consumer, e a Blockchain que utiliza a EVM para executar o *smart contract*.

O sistema Tuna Tracking tem como foco melhorar a transferência de informação na cadeia de abastecimento de atum, garantindo a certificação e normas são cumpridas por todos os elementos da cadeia. O sistema é composto por seis módulos: registo, normas, produção, fabrico, etiquetagem e interface. Estes seis programas são registados na mesma Blockchain, proporcionando um ambiente de consistência dentro do sistema.

O GestOlive tem como alvo a produção de azeite em França. Os intervenientes desta cadeia de fornecimento são o produtor de azeitonas, o primeiro processador no lagar, a embalagem na fábrica, o fornecedor, e o retalhista. A primeira etapa do processo é quando o agricultor transporta as azeitonas para o lagar, e é também quando a contrafação pode ocorrer. A adulteração pode ser mais difícil de detetar, e não existe um método único para detetar todos os tipos de contaminação. O sistema GestOlive, tem como objetivo abordar tais problemas através da recolha de informação, introduzida manualmente pelo produtor. O sistema não segue o produto depois de este sair do moinho.

A versão alfa do Ambrosus utiliza uma abordagem *hardware-in-place* para analisar os níveis de compostos fenólicos e ácido oleico. Além disso, as etiquetas RFID são utilizadas para identificar de forma única os artigos do produto, o identificador único é associado a informações sobre o processo e o transporte, armazenadas numa

Blockchain [20]. Os consumidores podem aceder a informação armazenada na cadeia através de códigos QR.

A Connecting Food é uma empresa cujo principal objetivo é apresentar uma plataforma de confiança alimentar baseada na tecnologia Blockchain, conseguindo uma rastreabilidade de ponta a ponta [23]. A apresentação de projetos como Herta, Axereal, e General Mills têm em comum a base o conceito Connecting Food, mas cada projeto necessita e aplica funcionalidades distintas [29]. O LiveAudit no projeto Herta tem como objetivo fornecer em tempo real uma garantia de qualidade e análises de fornecedores para operações do agronegócio em tempo real. Com esta abordagem, é possível ter uma solução de auditoria digital contínua que verifica proactivamente a conformidade de cada lote. Os projetos Axereal o Food Twin Generator convertem determinado esquema de dados alimentares em um padrão de forma a ser aplicado em *digital twins* de forma automática.

O projeto Ingredia pretende auxiliar fabricantes de alimentos e bebidas a tirar o máximo partido dos ingredientes lácteos, contribuindo para a inovação dos produtos e para o aumento da produtividade nas fábricas, apresentando uma abordagem de produção responsável. A Ingredia é um serviço de rastreabilidade e auditoria em tempo real, armazenando a informação na Blockchain. No final do processo, o produto será 100% rastreável e auditável através de um código QR.

A análise anterior pretende fornecer uma contribuição para a comunidade científica, alinhando as atuais aplicações DLT/Blockchain e IoT na indústria agroalimentar. Descreve as iniciativas, projetos, e como é possível fazer evoluir a agricultura e conceber um sistema que se concentre na complexa realidade que a maioria dos agricultores suporta atualmente. É evidente que a maioria das empresas não partilha qualquer informação sobre as tecnologias aplicadas ou detalhes em relação à implementação técnica dos projetos. A informação que está disponível ao público tem uma perspetiva comercial e de marketing.

É notório que 80% dos projetos reais aplicam sensores IoT para a medição de dados fundamentais aos negócios. A maioria dos projetos concentra-se no rastreio e compreensão do ciclo nas fases distintas do produto, mas, em alguns casos, o foco está na eficiência e na redução dos custos para os agricultores. Finalmente, é importante identificar e propor um atributo que identifica a qualidade ou certificação dos produtos. A forma típica de apresentar um certificado de qualidade é através da utilização de etiquetas RFID ou Códigos QR. Em particular o projeto Tea Tree Oil fornece um código especial ATTIA (Australian Tea Tree Industry Association), e o projeto Soybean Trace fornece um GTIN (Global Trade Identification Numbers) ao produto.

É necessário que as equipas de desenvolvimento considerem a explicar os objetivos e o propósito das soluções, já as empresas devem apresentar os seus projetos e apro-

fundar os detalhes técnicos do desenvolvimento. Neste momento apenas os projetos acadêmicos explicam e abordam as tecnologias e técnicas utilizadas. É no armazenamento da informação que circula do campo agrícola até ao utilizador/consumidor final que o real valor está representado.

## 2.3 EXPLORAÇÃO TECNOLÓGICA

Com o intuito de entender que tipo de tecnologia DLT se enquadrava mais com a solução pretendida, foi necessário realizar uma fase de estudo sobre as tecnologias existentes, entendendo quais as mais atualizadas e quais é que apresentam um desenvolvimento contínuo. Cada tecnologia seria avaliada na facilidade de desenvolvimento, documentação existente, comunidade de apoio da tecnologia, vantagens e desvantagem inerentes da tecnologia.

A exploração das tecnologias a explorar realizou-se no início de 2022 e todas as informações relativas a cada tecnologia focam-se nesta data temporal. Através de uma procura extensa na comunidade Blockchain selecionadas as seguintes tecnologias.

### 2.3.1 *Helium*

A primeira tecnologia explorada foi o Helium apresentando uma rede descentralizada baseada em Blockchain, que permite dispositivos comunicarem entre si, sem a necessidade de redes móveis ou WiFi. Aplica um mecanismo de consenso único chamado PoC (Proof-of-Coverage), que recompensa os nós da rede por fornecerem cobertura sem fios a dispositivos na rede [40].

Os dispositivos da rede utilizam ondas de rádio para comunicar com os nós próximos, que depois reencaminham os dados para outros nós até chegarem ao seu destino. Os nós que fornecem cobertura ganham a moeda criptográfica nativa do Helium, HNT, que pode ser utilizada para pagar o acesso à rede ou trocada por outras moedas criptográficas ou moedas Fiat.

A rede Helium suporta uma variedade de aplicações, incluindo localização de ativos, infraestrutura de cidades inteligentes, e até mesmo a ligação de dispositivos IoT como localizadores em animais de estimação e sensores. Por ser descentralizada, a rede é mais segura e resiliente do que as redes sem fios tradicionais, e pode operar em áreas com cobertura limitada [40].

A Blockchain de Helium disponibiliza os seguintes recursos:

1. **Nodes:** Os *nodes* são a base principal da rede de Helium, são dispositivos físicos que comunicam com outros *nodes* e dispositivos através de ondas de rádio.
2. **Proof of Coverage:** O mecanismo de consenso utilizado pela Blockchain do Helium é o PoC, que fornece cobertura sem fios a dispositivos próximos com o fim de ganhar recompensas [40].
3. **Hotspots:** São nós especializados para fornecer cobertura sem fios na rede de Helium, podem ser adquiridos a terceiros ou construídos através de hardware e software *open source*.
4. **Moeda criptográfica HNT:** A moeda é utilizada para pagar o acesso à rede e serviços.
5. **Taxas de dados:** As taxas são utilizadas para pagar a transmissão de dados na rede de Helium.

Os recursos trabalham em conjunto para criar uma rede sem fios descentralizada que é mais segura, resiliente e rentável do que as redes sem fios tradicionais. O Helium fornece uma plataforma flexível e versátil para a construção de aplicações sem fios descentralizadas que podem melhorar a eficiência, segurança, e privacidade numa variedade de indústrias e casos de utilização.

O mecanismo de consenso de PoC utilizado pela Helium incentiva os nós a fornecer cobertura sem fios pois utilizam baixa taxas de energia e podem funcionar durante meses ou mesmo anos com uma única carga de bateria, tornando-a ideal para IoT e outras aplicações de baixa energia.

A Blockchain de Helium apresenta suporte para as seguintes soluções e serviços descentralizados [34, 20]:

- **Localização de ativos:** O Helium pode ser utilizado para construir um sistema descentralizado de localização de bens que utiliza sensores sem fios para rastrear a localização e condição de bens tais como contentores de transporte, veículos e equipamento. Este tipo de serviços são uteis para setores de logística e transporte.
- **Infraestruturas urbanas inteligentes:** A tecnologia permite construir uma rede sem fios descentralizada para a gestão de tráfego, segurança pública, e monitorização do ambiente.

A realização do protótipo com a tecnologia, permitiu verificar desvantagens tais como a falta de documentação, inexistência de blogs/fóruns com informação acerca da tecnologia ou até o tratamento de erros. Além disso, a existência de múltiplos componentes na estrutura, adiciona demasiada complexidade para a infraestrutura pretendida para a plataforma COW.

### 2.3.2 IOTA

A tecnologia IOTA é uma tecnologia baseada em DLT concebida com o foco no ecossistema IoT. A tecnologia aplica um grau de inovação na estrutura de dados Directed Acyclic Graph, denominada de Tangle [40]. Na Tangle, cada transação está ligada a duas transações anteriores, formando uma estrutura semelhante a uma malha em vez de uma cadeia linear. Este tipo de processo permite uma maior escalabilidade e processamento de transações mais rápidos, uma vez que cada nova transação confirma múltiplas transações anteriores.

A tecnologia IOTA apresenta uma característica única, sendo o seu modelo *feeless*. Contraditando outras moedas criptográficas que requerem taxas de transação para incentivar os miners, as transações IOTA não implicam quaisquer taxas [32, 40]. Em vez disso, os utilizadores são obrigados a verificar duas transações anteriores no Tangle para validar a sua própria transação.

A tecnologia foi concebida para permitir transações *machine-to-machine* e realizar micro pagamentos entre dispositivos IoT. A sua natureza escalável e *feeless* torna-a adequada para gerir grandes volumes de transações em tempo real. A IOTA também tem aplicações para além do IoT, tais como gestão da cadeia de abastecimentos, identidade digital, e mercados de dados descentralizados (*marketplaces*) [32, 40].

Uma das características apresentadas é a Identidade Digital, sendo que pode ser utilizada para criar um sistema descentralizado, onde os utilizadores possuem e controlam os seus próprios dados de identificação. Este tipo de característica auxilia a prevenir casos de roubo de identidade, assegurando a sua autenticidade e impedir a fraude. Concluído, a natureza única da IOTA apresenta ser uma tecnologia promissora a uma vasta gama de indústrias.

A realização de um protótipo permitiu realizar uma análise das funcionalidades da rede, identificar as vantagens e desvantagens da tecnologia.

#### Vantagens:

- **Escalabilidade:** A estrutura Tangle é altamente escalável, uma vez que cada nova transação confirma múltiplas transações anteriores [32].
- **Inexistência de taxas:** A IOTA não requer taxas de transação, o que a torna uma solução rentável para micro pagamentos e outras transações de pequeno valor.
- **Transações rápidas:** As transações IOTA podem ser processadas rapidamente, uma vez que não precisam de ser validadas por um intermediário centralizado [32].

- **Eficiência energética:** A IOTA não depende de um mecanismo de consenso, sendo mais eficiente do ponto de vista energético.

**Desvantagens:**

- **Complexidade:** A estrutura Tangle da IOTA é mais complexa do que a tecnologia tradicional Blockchain, o que pode dificultar a adesão e compreensão dos utilizadores. .
- **Segurança:** A IOTA apresenta questões de segurança, relacionadas com a utilização de chaves privadas e o risco de ataques de duplicação de gastos, o que pode ser um fator negativo na adoção da tecnologia [32].
- **Ecossistema Limitado:** Em comparação com plataformas de blockchain mais estabelecidas como o Ethereum, a IOTA tem um ecossistema de desenvolvimento mais pequeno, com menos ferramentas e escassos recursos disponíveis para o desenvolvimento de aplicações [40].

Analisando as vantagens e desvantagens da tecnologia, tomou-se a decisão que não seria a tecnologia a utilizar no projeto, devido à existência de múltiplos artigos científicos acerca da fraca adoção da tecnologia que referem um conjunto de limitações de segurança.

### 2.3.3 Hedera

A última tecnologia explorada foi o Hedera Hashgraph, tendo em conta que é uma tecnologia DLT que apresenta um algoritmo de consenso único para alcançar alta eficiência na execução de transações de forma simples e segura.

O Hedera Hashgraph foi concebido para ser uma plataforma escalável e segura para o desenvolvimento de dApps (aplicações descentralizadas), e soluções baseadas em Blockchain. A tecnologia possui o algoritmo de consenso Hashgraph, que é uma variante ao protocolo tradicional utilizado em redes peer-to-peer . O algoritmo de consenso Hashgraph foi concebido para alcançar uma alta eficiência através da utilização de uma estrutura de dados DAG. [20, 1]. O Hashgraph apresenta um modelo de governação único, concebido para equilibrar as necessidades da comunidade com as exigências dos utilizadores da empresa. O modelo de governação baseia-se num conselho de 39 organizações globais, que são responsáveis pela gestão da rede e pela tomada de decisões relacionadas com atualizações, políticas e outras questões-chave.

A tecnologia inclui diversos serviços como o Consensus Service, Token Service, e Smart Contract Service. Estes serviços disponibilizam aos programadores diversos

recursos para implementar em soluções aplicações, de forma rápida, segura e com baixas taxas de transação.

A moeda criptográfica da Hedera chama-se HBAR, que pode ser utilizada para pagar taxas de transação e os acessos aos recursos da rede. O símbolo HBAR é também utilizado para apostar na rede, o que permite aos utilizadores ganharem recompensas por participarem em consensos [20, 28, 21].

A tecnologia Hedera tem potenciais aplicações em várias indústrias, incluindo finanças, gestão das cadeias de fornecimento, e cuidados de saúde. A plataforma oferece aos programadores diversas ferramentas incluindo diversos SDKs (Software Development Kits), permitindo ainda a integração de *smart contract* Solidity.

A tecnologia Hedera apresenta diversos atributos e características:

1. **Alto desempenho:** É altamente escalável e encarrega-se de lidar com milhares de transações por segundo, devido ao algoritmo de consenso que tem por base a tecnologia DAG [1].
2. **Segurança:** O algoritmo de consenso Hashgraph implementa um nível de segurança elevado através da utilização do aBFT (Asynchronous Byzantine Fault Tolerance), sendo um algoritmo robusto e resistente a vários tipos de ataques.
3. **Governança:** A rede é gerida por um conselho composto pelas principais organizações e empresas de todo o mundo [28], garantindo que a rede é gerida de forma justa e transparente, promovendo a adoção da tecnologia.
4. **Equidade:** O Hedera foi concebido para ser justo e equitativo, sem mecanismos de mining, em vez disso, todos os nodes da rede possuem a mesma hipótese de serem selecionados para validar as transações.
5. **Taxas de transação baixas:** São baixas e previsíveis.
6. **Interoperabilidade:** A tecnologia é interoperável com outras Blockchains e tecnologias DLT, com o propósito de facilitar as transações entre cadeias e aumentar a sua utilidade global [21].
7. **Developer-friendly:** A Hedera fornece aos programadores diversas ferramentas e recursos para os ajudar a construir aplicações descentralizadas, diversos SDKs e uma comunidade de programadores no Discord disposta a ajudar.

A tecnologia Hedera apresenta diversos tipos específicos de serviços:

- **File Service:** A Hedera fornece um serviço de armazenamento de ficheiros chamado Hedera File Service (HFS), que permite aos utilizadores armazenar e recuperar ficheiros de forma segura e eficiente na rede. O serviço permite carregar ficheiros para a rede e recuperá-los usando um único ID de ficheiro. Os ficheiros são armazenados de forma descentralizada em múltiplos nós da

rede, assegurando que estão altamente disponíveis e resistentes à perda de dados. O ficheiro é dividido em chunks menores e distribuído por vários nós na rede, utilizando o algoritmo de consenso do HCS.

O serviço de ficheiros também fornece uma série de características integradas para apoiar a criação de smart contracts, adição de outros ficheiros ao mesmo, e de definir controlos de acesso e permissões nos ficheiros. Além disso, o serviço de ficheiros Hedera suporta múltiplos formatos de ficheiros, incluindo texto, imagens, áudio e vídeo. Isto torna-o uma plataforma versátil para o armazenamento e partilha de uma vasta gama de tipos de ficheiros e dados.

- **Hedera Consensus Service:** O Hedera Consensus Service utiliza o algoritmo de consenso do Hashgraph, para validar e ordenar os eventos e apresentar transparência na história dos eventos ao longo do tempo sem exigir um histórico persistente de transações. Como resultado, o serviço de consenso proporciona um consenso rápido, justo e seguro a um custo mais baixo do que qualquer outra rede.

Uma das principais características do HCS (Hedera Consensus Service) é o seu apoio à linguagem de programação Solidity, que permite desenvolver smart contracts. Este processo facilita aos programadores a migração dos smart contracts existentes da rede Ethereum para a Hedera, ou a utilização das bibliotecas e ferramentas de código existentes [20].

- **Token Service:** O Hedera Token Service permite aos programadores criar e gerir tokens personalizados na rede Hedera. Estes tokens podem representar qualquer bem, tal como uma moeda criptográfica, pontos de fidelidade, ou acesso a um serviço específico. O HTS (Hedera Token Service) foi concebido para ser rápido e eficiente, com baixas taxas de transação e elevada eficiência [21].

Uma das principais características do HTS é o seu apoio a trocas atómicas, que permitem aos utilizadores trocar um token por outro sem a necessidade de um intermediário ou troca [20]. A Hedera fornece a capacidade de emitir tokens numa rede pública distribuída globalmente sem comprometer o desempenho. Os programadores podem agora definir e emitir tokens diretamente para a rede principal Hedera. Os tokens herdam muitas das características da própria HBAR, incluindo o consenso aBFT, e a finalidade em questão de segundos sem risco de bifurcação. .

- **Audit Tools:** A Hedera fornece ferramentas de análise e monitorização em tempo real para a rede, o que permite aos utilizadores acompanhar e analisar a atividade de transação, o desempenho da rede, e muito mais. Através do Mirror Node é possível realizar diversos tipos de auditoria aos dados na rede. O Mirror

Node fornece uma visão em tempo real da atividade da rede, permitindo aos programadores monitorizar transações, verificar chamadas, e outros eventos à medida que ocorrem [21]. Isto pode ser útil para fins de auditoria, pois permite aos utilizadores verificar se os dados na rede são exatos e atualizados.

O Mirror Node também pode ser utilizado para construir aplicações e serviços que requerem acesso em tempo real a dados na rede [28].

Além disso, é importante indicar que a Hedera opera três redes primárias, cada uma possui um objetivo diferente. [20, 1].

1. **Hedera Mainnet:** A Hedera Mainnet é a rede principal para a plataforma Hedera Hashgraph. É uma rede pública aberta a qualquer pessoa e concebida para casos de uso comercial. O principal objetivo da Mainnet Hedera é fornecer uma plataforma rápida, segura e fiável para a construção e implementação de aplicações descentralizadas e outras soluções baseadas em blockchain. A Mainnet suporta a moeda criptográfica HBAR.
2. **Hedera Testnet:** A Testnet é uma rede separada que foi concebida para os programadores e outros utilizadores testarem as suas aplicações e experimentarem novas funcionalidades antes de as implantarem na Hedera Mainnet. A Testnet é uma versão simulada da Hedera Mainnet e proporciona aos programadores um ambiente de teste onde podem testar o seu código e aplicações sem utilizar fundos ou recursos reais. A Testnet é também utilizada pela Hedera para fins de teste e desenvolvimento interno.
3. **Hedera Previewnet:** O objetivo da Previewnet é permitir aos programadores testar e experimentar novas características e atualizações num ambiente controlado, sem o risco de afetar o ambiente de produção da Hedera Mainnet. A Previewnet é muitas vezes utilizada para testar novas versões dos SDKs, APIs e outras ferramentas e serviços da Hedera.

Em conclusão, a tecnologia Hedera é a tecnologia indicada para o desenvolvimento da plataforma COW. O algoritmo de consenso permite transações rápidas e seguras, existe uma facilidade inerente de integrar a Hedera Network através do uso da Hedera API (Application Programming Interface), simplificando a infraestrutura de desenvolvimento da plataforma.

A facilidade de introduzir *smart contracts* na rede Hedera e integrar o seu funcionamento numa API foi um dos fatores que promoveram a adoção da tecnologia. Em comparação com IOTA e Helium, a Hedera tem um ecossistema mais estabelecido, com parcerias e integração de grandes empresas que garante a estabilidade do sistema. O facto de existir constantes desenvolvimentos e uma comunidade pronta a auxiliar qualquer questão facilitou o desenvolvimento do protótipo e do projeto.

Ainda que, o IOTA e Helium apresentem benefícios únicos, não são os mais adequados para o sistema pretendido devido à complexidade que possuem, e sérias questões com a escalabilidade, interoperabilidade e falta de documentação.

Independente de existirem inúmeras tecnologias que pretendem garantir a gestão de uma cadeia de abastecimento, tecnologias como a Hedera são reduzidos. A Hedera apresentou funcionalidades e componentes suficientes para enfrentar os desafios nesta indústria complexa.

## ARQUITETURA

---

Esta secção, pretende explorar cada componente, a arquitetura definida, bem como os protocolos de comunicação e modelos de dados aplicados. Abordaremos também os desafios encontrados durante a fase de decisão e definição do sistema e as estratégias para mitigar os riscos. O sistema desenvolvido consiste em múltiplos componentes, que inclui uma interface de utilizador, uma API Restful para tratar da lógica de negócio, a rede distribuída fornecida pela Hedera para manter a consistência de dados entre todos os intervenientes, e uma base de dados.

A arquitetura do sistema inclui padrões de design e práticas auxiliares no desenvolvimento dos diversos componentes da plataforma. O frontend apresenta uma interface ao utilizador de fácil compreensão para realizar qualquer tipo de análise de informações da cadeia de abastecimento. A API Restful tem como propósito realizar a comunicação entre o frontend, a rede Hedera e a base de dados. O Hedera é a tecnologia definida que permite aplicar as características de uma Blockchain descentralizada fornecendo uma plataforma segura e escalável para o registo e armazenamento de dados. A base de dados tem como objetivo manter um estado atual da rede Hedera, permitindo acesso a dados sem a aplicação de taxas, é necessário identificar que cada utilizador mantém o estado atual da rede de forma privada.

Deve se ter em conta, que a necessidade de uma base de dados surge no sentido de minimizar o custo das operações de acesso aos dados, reduzindo o tempo e a necessidade de aceder à rede Hedera. Desta forma, a base de dados tem como função armazenar qualquer tipo de registo realizado na plataforma, para que facilmente seja acedido, estes tipos de decisões têm em conta a necessidade de os utilizadores acederem com elevada frequência a diferentes informações da cadeia de abastecimento.

### 3.1 PERFIS DE UTILIZAÇÃO

A plataforma COW, permite o registo e utilização de 3 tipos de perfis distintos:

1. **Agricultor:** Um utilizador do tipo *Farmer*, tem acesso a qualquer operação relacionada com a gestão de campos agrícolas, desde a criação, atualização, listagem e verificação do detalhe de cada. Este tipo de utilizador pode ainda realizar uma gestão granular de bovinos, desde a criação, atualização, verifi-

cação genética do bovino, verificar histórico do bovino, compra e venda de novos bovinos, marcação de consultas, entre outras operações analíticas.

2. **Veterinário:** No caso de o utilizador ser do tipo *Veterinary*, é responsável por gerir todas as marcações de consultas que recebe, podendo aceitar ou rejeitar marcações. Caso o utilizador aceite marcações, automaticamente será gerado uma nova consulta que possui diferentes estados e aplica um custo final para o dono do bovino. Existe ainda a possibilidade de criar e atualizar consultas sem marcações prévias.
3. **Comprador:** O utilizador do tipo *Buyer*, tem acesso a todas as funcionalidades de gestão granular dos bovinos, mas todos os seus bovinos são guardados num único local (carteira digital). Este utilizador consegue ainda criar, atualizar leilões e participar em leilões.

Inicialmente a decisão de adicionar três tipos de perfis distintos advém da necessidade de representar uma cadeia de abastecimento real e complexa, sendo possível adicionar múltiplos utilizadores que identificam diferentes fases do ciclo de vida do bovino numa cadeia de abastecimento. A adição do tipo de utilizador veterinário surge numa iniciativa de demonstração o progresso e crescimento do bovino, bem como os cuidados de saúde que ocorrem.

A figura 4 representa a lista de todos os requisitos funcionais de cada um tipo de utilizador. Desta forma, é possível perceber quais as funcionalidades que cada tipo de utilizador possui. A figura 4 permite identificar as diferenças de requisitos funcionais dos utilizadores do tipo *Farmer*, *Veterinary* e *Buyer*.

### 3.2 FLUXOS DE AÇÕES

A secção seguinte, permite a análise de diferentes diagramas de fluxo que ilustram algumas das funcionalidades e requisitos disponíveis nos diferentes tipos de utilizadores. O objetivo destes diagramas é fornecer uma compreensão clara e abrangente de como o sistema funciona e como os utilizadores interagem com os componentes da plataforma.

O diagrama da figura 5 representa a sequência de passos que permitem da criação de um *smart contract* de um bovino, e a transação sequencial que é o *mint* permitindo mapear um bovino real para um bovino digital.

Inicialmente, o utilizador do tipo *Farmer* vai aceder à plataforma, preenchendo com sucesso as suas credenciais. De seguida o utilizador vai navegar para o formulário de criação de um bovino. O utilizador preenche todas as informações obrigatórias que identificam o bovino, desde o número de série, peso, altura, idade, local atual, progenitores entre outras informações. O objetivo do bovino digital é representar o

Requisitos Plataforma	
Tipo de Utilizador	Descrição
<b>Farmer</b>	Criar e atualizar qualquer informação de um bovino
	Verificar lista de bovinos que possui
	Verificar informação detalhada sobre cada bovino ( <i>localização, peso, altura, género, idade, pedigree, genealogia, histórico médico, histórico de localizações, histórico de donos</i> )
	Criar e atualizar qualquer informação de um campo
	Verificar lista de campos que possui
	Verificar informação detalhada sobre cada campo ( <i>localização, descrição, endereço, limite</i> )
	Criar e atualizar qualquer informação de uma marcação de consulta
	Verificar lista de marcações de consultas
	Verificar lista de consultas
	Verificar informação detalhada sobre cada consulta ( <i>motivo, custo, observação, estado</i> )
	Criar leilão para venda de qualquer bovino
	Verificar lista de leilões ( <i>a começar, em progresso, terminado</i> )
	Verificar lista de licitações de um leilão
	Verificar gráficos de análise com informação descritiva acerca da cadeia de abastecimento
	Atualizar informação detalha sobre o perfil de utilizador ( <i>foto, nome, data de nascimento, palavra-passe</i> )
<b>Buyer</b>	Criar e atualizar qualquer informação de um bovino
	Verificar lista de bovinos que possui
	Verificar informação detalhada sobre cada bovino ( <i>localização, peso, altura, género, idade, pedigree, genealogia, histórico médico, histórico de localizações, histórico de donos</i> )
	Verificar lista de leilões ( <i>a começar, em progresso, terminado</i> )
	Verificar lista de licitações de um leilão
	Verificar gráficos de análise com informação descritiva acerca da cadeia de abastecimento
	Atualizar informação detalha sobre o perfil de utilizador ( <i>foto, nome, data de nascimento, palavra-passe</i> )
<b>Veterinary</b>	Criar e atualizar qualquer informação de uma consulta
	Aceitar ou Rejeitar marcações de consulta
	Verificar gráficos de análise com informação descritiva acerca da cadeia de abastecimento
	Atualizar informação detalha sobre o perfil de utilizador ( <i>foto, nome, data de nascimento, palavra-passe</i> )

Figura 4: Lista de Requisitos Funcionais

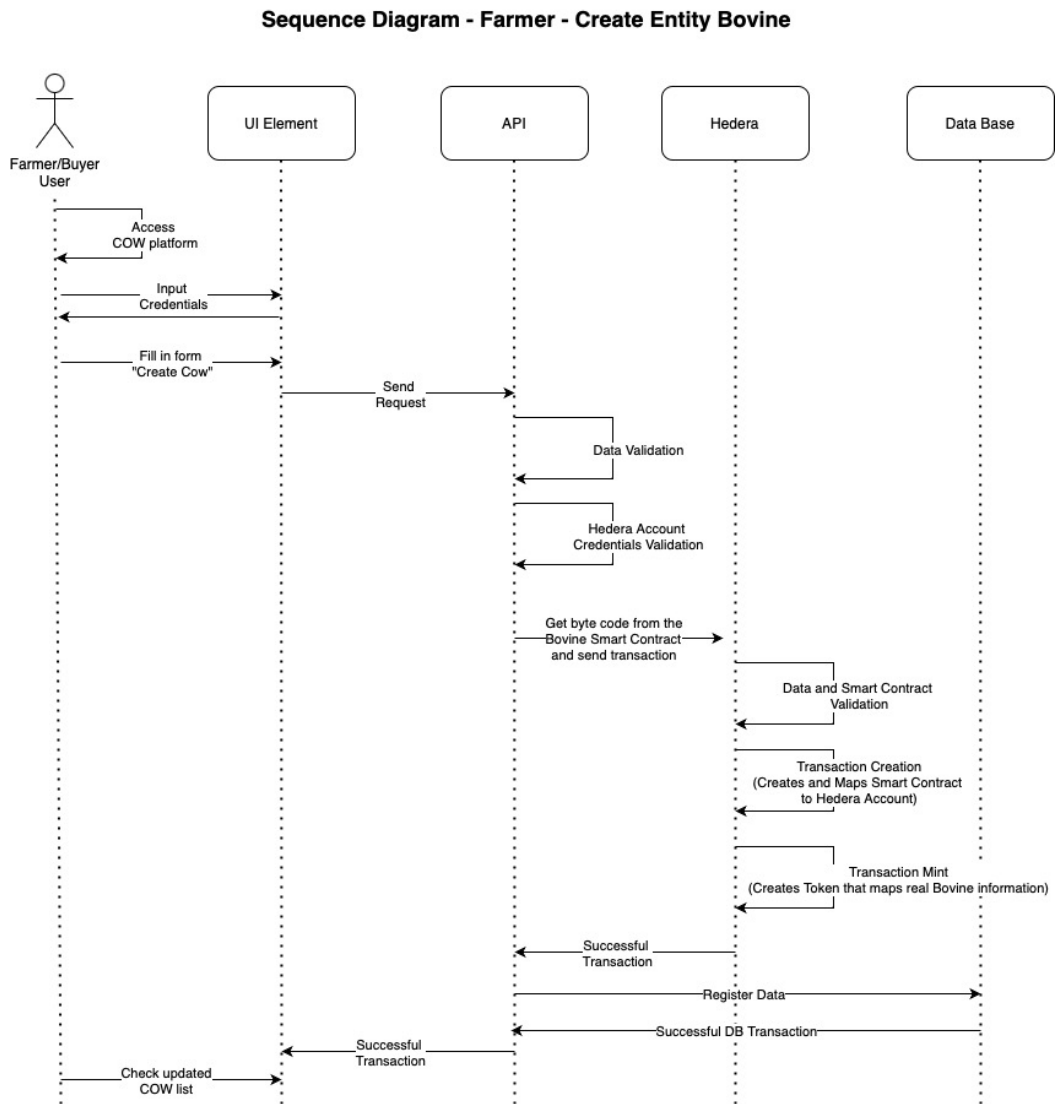


Figura 5: Diagrama de Sequência - Farmer/Buyer - Criar Bovino

bovino real ao pormenor. O formulário é validado pelas componentes do frontend e caso os dados inseridos passem a validação do frontend com sucesso, um pedido POST é executado para a API. A API na receção do pedido, valida o tipo de operação, valida os dados rececionados e se toda a informação obrigatória se encontrar presente no pedido, surge a próxima validação. Esta validação surge ao nível das credenciais da conta Hedera do utilizador, obtendo de seguida o *byte code* do *smart contract* do bovino.

No caso de o utilizador insira o seu primeiro bovino da plataforma COW, a API irá utilizar o *byte code* do *smart contract* do bovino, para executar uma transação de validação dos dados e de seguida uma transação para criar um contrato associado ao utilizador. Seguindo o fluxo e caso todas as operações sejam realizadas com sucesso, é realizado uma chamada ao smart contract (*mint*) que permite criar um NFT (Non-fungible Token) criando o bovino digital. Estas transações englobam a chamada ao construtor e de seguida a função *safeMint* presente no standard ERC-721 aplicado pelo *smart contract*. A Hedera valida as transações executadas, e regista no Mirror Node Hedera, este processo permite a validação das transações. Caso o utilizador já possua mais do que um bovino na rede, o processo passa a chamar diretamente a função de *mint*, sem existir a necessidade de criar um *smart contract*.

Caso o utilizador já possua mais do que um bovino na rede, o processo passa a chamar diretamente a função de *mint*, sem existir a necessidade de criar um smart contract.

Após o sucesso das transações anteriores é realizado o armazenamento do novo estado na base de dados do utilizador, e caso todas as ações sejam realizadas com sucesso, então a API responde ao pedido permitindo a componente gráfica atualizar e apresentar a lista de bovinos atualizada. A base de dados permite um acesso instantâneo aos dados, não existindo a necessidade de aplicar custos ou qualquer tipo de taxas nos acessos a dados.

O diagrama da figura 6 representa a sequência de passos necessários para atualizar o objeto que representa um campo agrícola.

A sequência inicia com um utilizador do tipo *Farmer* a aceder à plataforma e preenche com sucesso as suas credenciais. De seguida o utilizador vai navegar até à lista de campos agrícolas, selecionando o campo específico para atualizar. Após validar o detalhe do campo agrícola, surge o formulário de atualização de um campo previamente preenchido. Após a atualização dos atributos referentes com o campo agrícola, como nome, morada, localização geográfica, e identificação dos bovinos presentes no local, caso toda a informação tenha sido colocada convenientemente, o frontend irá realizar uma validação dos dados antes de um pedido ser criado para a API. Após a validação do formulário pelas componentes do frontend e caso os dados

**Sequence Diagram - Farmer - Update Entity Field**

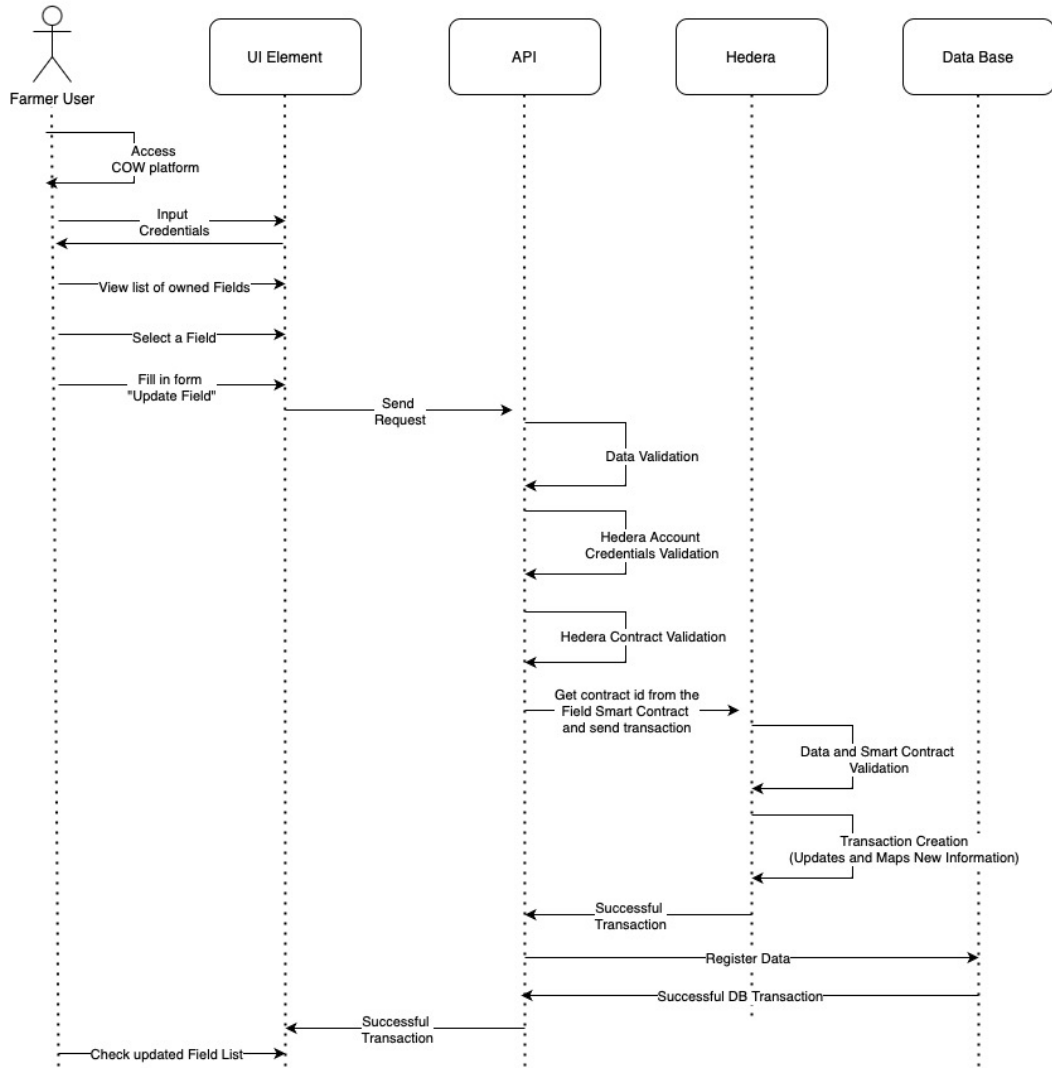


Figura 6: Diagrama de Sequência - Farmer - Atualizar Campo Agrícola

inseridos passem a validação do frontend com sucesso, um pedido PUT é executado para a API. A API na recepção do pedido, valida o tipo de operação, valida os dados rececionados e se toda a informação obrigatória se encontrar presente no pedido, surge a próxima validação.

Esta validação surge ao nível das credenciais da conta Hedera do utilizador, e validação do identificador do contrato na rede Hedera. Após o sucesso das transações anteriores é executado uma transação de validação de dados e de atualização da informação presente no smart contract do campo agrícola definido pelo utilizador.

Deve-se ter em conta que qualquer alteração na lista de bovinos identificados no terreno, iram originar operações sucessivas nos NFT dos bovinos, e como todas as ações com o Hedera são sincronas os acessos à rede deve ser rápido e simples. Após o sucesso das transações anteriores é realizado o armazenamento do novo estado na base de dados do utilizador, e caso todas as ações sejam realizadas com sucesso, então a API responde ao pedido permitindo a componente gráfica atualizar e apresentar a informação atualizada do campo agrícola.

O diagrama da figura 7 representa um utilizador do tipo veterinário a aceitar uma marcação de consulta e as transações sequenciais desta operação.

A sequência inicia com um utilizador do tipo *Veterinary* a aceder à plataforma e preenche com sucesso as suas credenciais. De seguida o utilizador vai navegar até à lista de marcações de consultas, selecionando a marcação que deseja confirmar. Após validar o detalhe da marcação que possui informações do bovino, data e hora e o motivo da marcação o frontend irá realizar uma validação dos dados antes de um pedido ser criado para a API. Após a validação do formulário pelas componentes do frontend e caso os dados inseridos passem a validação do frontend com sucesso, um pedido POST é executado para a API. A API na recepção do pedido, valida o tipo de operação, valida os dados rececionados e se toda a informação obrigatória se encontrar presente no pedido, surge a atualização da marcação na base de dados.

De seguida, surge uma validação surge ao nível das credenciais da conta Hedera do utilizador, e obtém-se o *byte code* do *smart contract* das consultas na rede Hedera. Após o sucesso das transações anteriores é executado uma transação de validação de dados e de registo da informação inserida no *smart contract* da consulta, que fica associado ao veterinário.

Após o sucesso das transações anteriores é realizado o armazenamento do novo estado na base de dados do utilizador, e caso todas as ações sejam realizadas com sucesso, então a API responde ao pedido permitindo a componente gráfica atualizar e apresentar a informação atualizada do campo agrícola. O dono do bovino, recebe a informação que a sua marcação foi atualizada e no momento em que abrir a aplicação irá se obter o novo estado presente na cadeia de abastecimento.

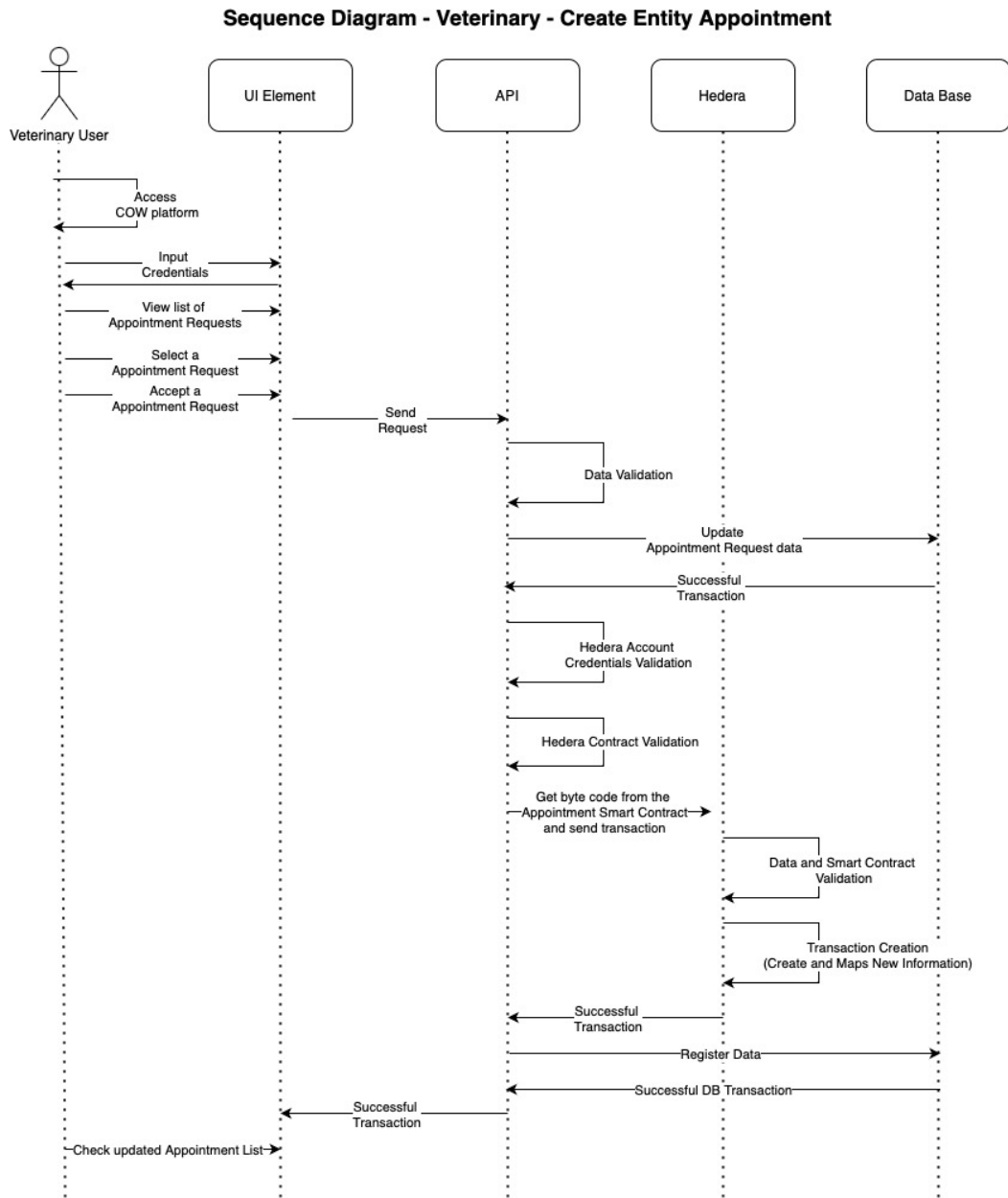


Figura 7: Diagrama de Sequência - Veterinário - Criar Consulta

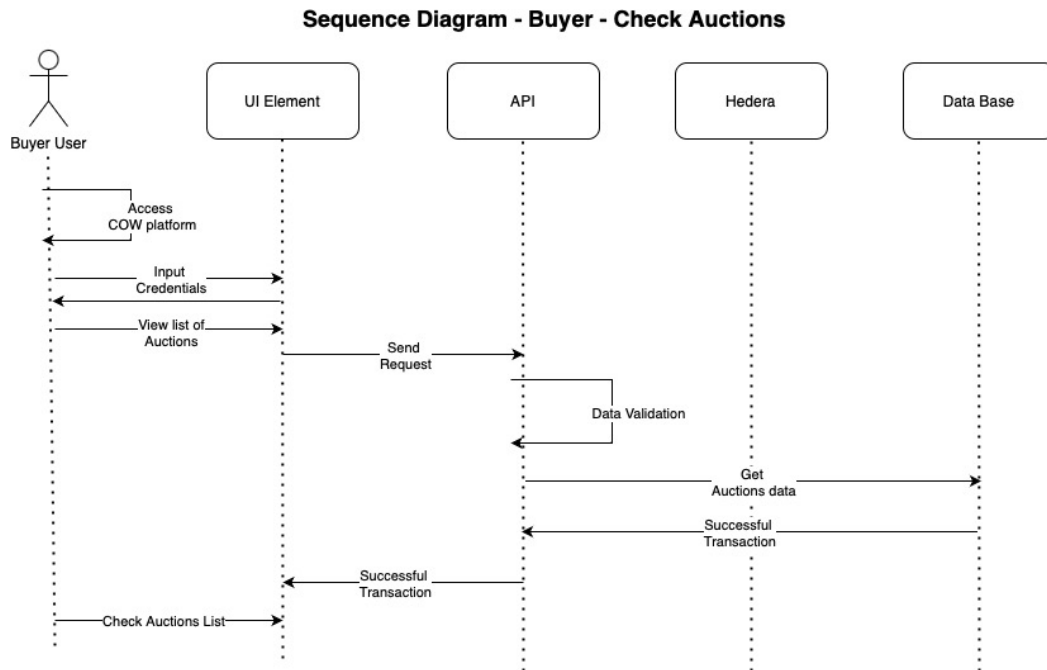


Figura 8: Diagrama de Sequência - Comprador - Listar Leilões

O último fluxo é representado pela figura 8 e pretende representar um simples listar de todos os leilões públicos.

A sequência inicia com um utilizador do tipo Buyer a aceder à plataforma e preenche com sucesso as suas credenciais. De seguida o utilizador vai navegar até à lista de leilões. Um pedido POST é executado para a API. A API na receção do pedido, valida o tipo de operação, e realiza um query à base de dados do utilizador, após a obtenção dos dados os mesmo são mapeados para objetos DTOs (Data Transfer Object).

O frontend ao receber esta resposta apresenta a informação do estado atual da base de dados ao utilizador. Este tipo de ações, têm como intuito diminuir o custo que envolve obter dados de uma rede distribuída.

### 3.3 ARQUITETURA GERAL

A arquitetura da plataforma COW está dividida em cinco grandes componentes:

- **Modelo de dados:** Inclui a estrutura de dados apresentada pela base de dados.
- **Smart Contracts:** Apresenta o modelo dos dados presentes na rede Hedera e necessários para o bom funcionamento da *supply chain*. Os contratos vão conter a informação real da entidade que mapeiam. O smart contract do bovino inclui o standard ERC-721 que permite gerar NFTs na rede Hedera e

ainda um conjunto de operações automáticas. Os *smart contracts* contêm a informação verídica da cadeia.

- **Hedera Hashgraph:** É a rede distribuída que mantém o registo de todas as transações realizadas na plataforma, qualquer tipo de ação está devidamente regista na Previewnet da rede Hedera. O acesso ao Mirror Node da Hedera permite realizar qualquer tipo de auditoria aos dados e validação de funções. Deste modo, é possível gerar um mecanismo que mantém a informação atualizada por toda a cadeia.
- **API:** Inclui toda a lógica de negócio, métodos de acesso aos modelos de dados, ações com a rede Hedera, o byte code de todas as entidades da plataforma e disponibilizando um canal de comunicação direto com o frontend.
- **Frontend:** É a plataforma web que apresenta a informação ao utilizador e comunica com a API. A plataforma COW é o suporte para que cada entidade aceda à cadeia de abastecimento.

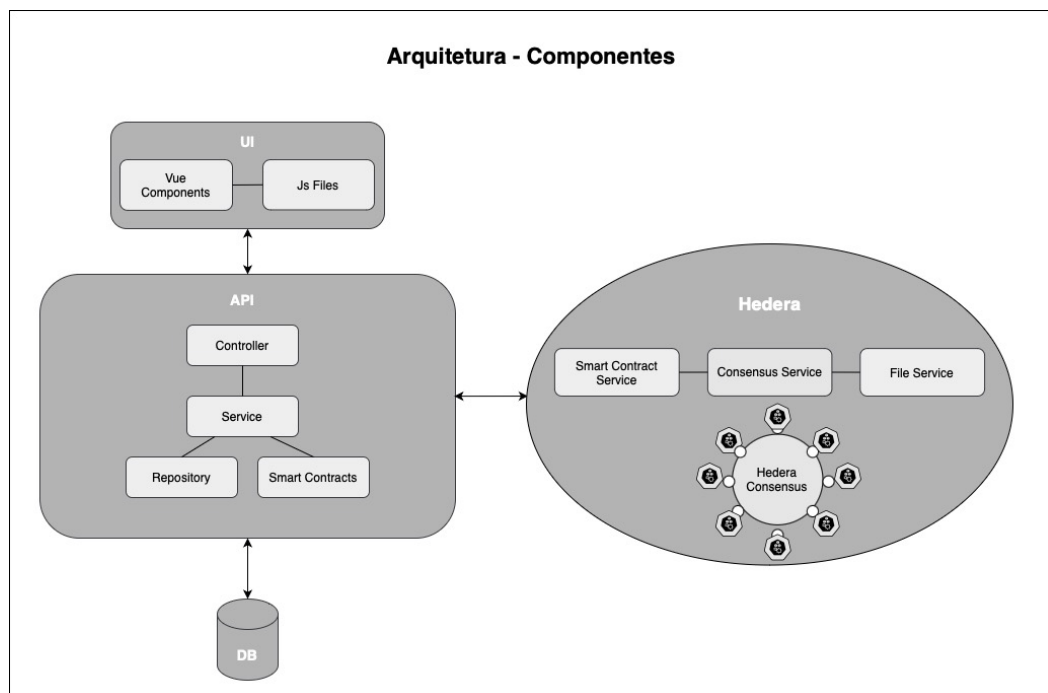


Figura 9: Representação da Arquitetura Geral

A figura 9 representa o ambiente de coexistência dos cinco componentes e como cada componente interage com os restantes. O frontend representado na figura por UI (User Interface) possui os diferentes componentes Vue e ficheiros JavaScript necessários para a execução do código. A API segue o modelo de 3 camadas, apresentando o Controlador, Serviço e Repositório. Além disso, possui a componente dos *Smart Contracts*, devido aos *byte codes* das entidades. A base de dados que comunica diretamente com a API e mantém o estado atual dos dados presentes na rede Hedera. E por último a rede Hedera, que possui diversos serviços e o algoritmo

de consenso Hashgraph, abordados no capítulo 2. É a rede Hedera que armazena os *smart contract* e funciona como a fonte da verdade para todos os utilizadores da plataforma.

### 3.3.1 Modelo de Dados

O modelo de dados é a representação conceptual dos dados, que descreve as relações entre os diferentes dados e as regras que regem a forma como os dados são armazenados, organizados, e acedidos dentro de uma base de dados.

O modelo de dados da plataforma COW, tem como foco representar uma cadeia de abastecimento de bovinos, de modo, a recriar um caso de estudo real e diversificado. O modelo inclui entidades desde a produção de bovinos, detalhes do ciclo de vida do bovino, mudanças de terreno, estado de saúde do bovino, entre outros pormenores necessários para alcançar o conceito *seed to table*.

Além disso, no contexto de uma cadeia de abastecimento de bovinos, surge o conceito de leilões e licitação com o intuito de apresentar um mecanismo facilitador da compra e venda de bovinos entre diferentes entidades.

Para a plataforma COW, definiram-se as seguintes entidades:

1. **Bovine:** Inclui dados sobre os bovinos, tais como raça, idade, peso, registos médicos, e número de identificação único. Esta entidade é especificamente para bovinos devido ao propósito inicial, mas facilmente permite a representação de qualquer tipo de animal, alargando o caso de estudo.
2. **Field:** Representa os campos agrícolas onde os bovinos são criados, inclui dados como a localização do campo, proprietário, informações de contacto, e registos de conformidade.
3. **Field History:** Entidade auxiliar dos registos de alterações de localizações dos bovinos, é a entidade auxiliar que mantém um registo de todas as transações ao nível da localização de bovinos.
4. **Appointment:** É a entidade que representa os detalhes das consultas, desde a data e hora da marcação, o motivo da consulta, custo e veterinário.
5. **Appointments Request:** A entidade de marcação de consulta fornece mecanismos para os utilizadores solicitarem marcações com antecedência, permitindo um planeamento eficiente.
6. **Auction** Identifica os detalhes num leilão, tais como data e hora, o bovino em leilão e a licitação vencedora.

7. **Bid:** Identifica as licitações sobre os bovinos em leilão, o licitante, o montante da oferta, e a data e hora da oferta. A entidade fornece uma forma de rastrear as ofertas individuais de bovinos durante o leilão. Assim sendo, permitindo o desenvolvimento de um mecanismo de participação em diferentes leilões. O processo de licitação permite auxiliar e ajuda a assegurar um processo justo e transparente.
8. **User:** Representa o utilizador na plataforma, possui nome, data de nascimento, tipo de utilizador, credenciais de autenticação, saldo e foto de perfil.
9. **User Type:** Representa os diferentes tipos de utilizador existentes na plataforma. Neste momento, estão definidos 3 tipos de utilizador *Farmer*, *Buyer* e *Veterinary*.

A figura 9 representa de forma gráfica as relações existentes entre cada entidade presente no modelo de domínio. De modo, a clarificar as relações utilizou-se um modelo de domínio baseado em UML (Unified Modeling Language).

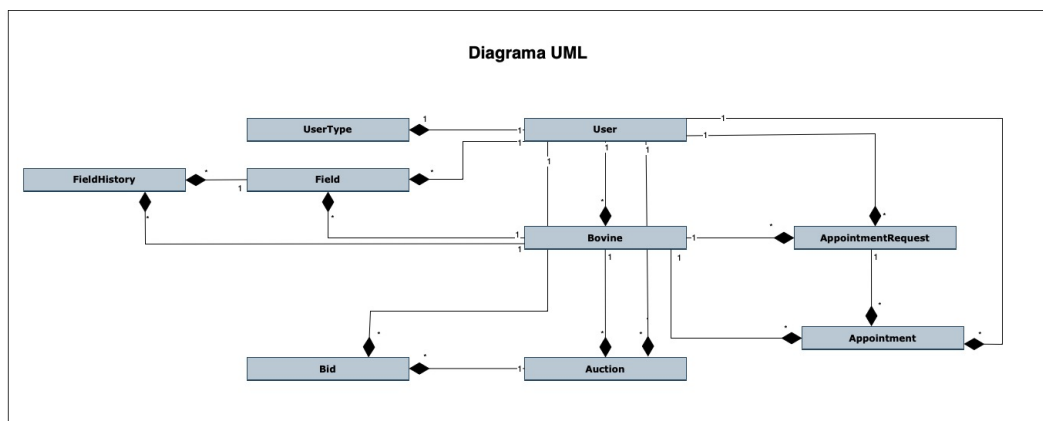


Figura 10: Modelo de Domínio UML

Através de uma revisão e análise, o modelo de domínio foi transposto para um MER (Modelo de Entidade e Relação apresentado na figura 11. Este modelo descreve o relacionamento das entidades da base de dados, isto é, permite explicar a estrutura lógica e relações que existem nas tabelas da base de dados.

### 3.3.2 API

Tendo em conta a necessidade de comunicação entre os diferentes componentes frontend, base de dados, rede Hedera e smart contracts. A API é um componente essencial para o funcionamento de toda lógica da plataforma COW, é a partir da API que é possível unir todos os componentes.

A importância elevada do componente levou à tomada a decisão de desenvolver uma Restful API, sendo uma estrutura que permite o desenvolvimento simples de

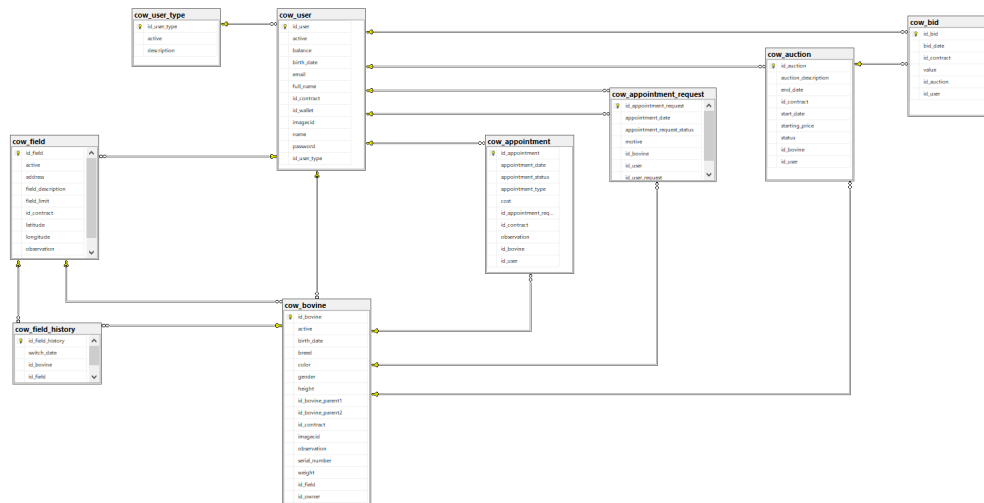


Figura 11: Modelo de Entidade e Relação

serviços web, permite escalar o sistema consoante a necessidade. O conceito REST significa Representational State Transfer, sendo a forma como os dados são trocados entre um cliente e um servidor.

A aplicação do Spring Boot numa API Rest facilitou a configuração do projeto base, aplicando logo no início serviços de segurança através do suporte do OAuth 2.0 e de JSON Web Tokens (JWT). O conjunto destas características simplificaram o desenvolvimento da solução sendo que tarefas como configuração injeção de dependências e integração da base de dados.

A decisão de utilizar a framework de Spring Boot foi uma escolha evidente devido às suas características de escalabilidade do sistema através de funcionalidade de *load balancing* e *caching*. Além disso, apresenta ainda processos de autenticação e autorização importantes para o tratamento de dados. É importante de referir que a equipa de desenvolvimento se sentia bastante confortável com a tecnologia, sem outro dos fatores de decisão.

Um dos pontos de definição da API é a integração com a rede Hedera, sendo que a plataforma Hedera garante o acesso a uma *distributed ledger* pública, permitindo o desenvolvimento de aplicações descentralizadas. A tecnologia Hedera inclui diversos SDKs que facilmente integram qualquer tipo de API. Neste caso, a opção de utilizar o SDK de Java tornou-se simples, sendo que é o SDK com a maior quantidade de serviços estáveis.

A API está dividida em diferentes camadas cada uma com diferentes responsabilidades. A camada inicial é a camada de controlo, responsável por receber os pedidos do frontend, realizar qualquer tipo de validação do objeto do pedido, invocar a lógica de negócio e devolver no final uma resposta. Os ficheiros da camada de controlo apresentam a denominação de Controller, possuem anotações do tipo @RestController-

ler e `@RequestMapping`. Cada um dos pedidos apresenta o tipo de pedido, objeto que recebe e resposta que é enviada como é possível de verificar no Anexo A.

A 2ª camada é camada de serviços e possui a lógica de negócio, executa as operações pedidas pelo controlador. É responsável por exercer as regras do negócio e interagir com os dados, além de os enviar para serem persistidos também manipula. É nesta camada que acontecem as chamadas à rede Hedera, os smart contracts são chamados e construídos. Além disso, é esta camada que envia os dados mais acedidos para uma base de dados local. Um serviço é denominado no projeto da API por Service, possui anotações como `@Service`, a injeção da camada de dados é realizada a partir de `@Autowired`. No Anexo A está representado uma função de serviço.

A execução de chamadas à rede Hedera surge com a chamada de transações do tipo do `TransactionResponse` e envolvem sempre a inicialização de um *client*. Este *client* é inicializado no arranque da API através do ficheiro de configuração `.env`. O ficheiro `.env` apresenta a configuração das credenciais do Hedera, que necessita de ser previamente realizada no Hedera Portal (o README do projeto contém indicações da configuração inicial).

É importante indicar que qualquer transação executada com o Hedera, requer um custo em dois valores HBAR e GAS, consoante o tamanho da transação e complexidade do mesmo implicam as taxas iram variar. Contudo, as taxas aplicadas pelo Hedera são bastante baixas sendo possível de verificar o custo de cada transação.

Tendo em conta que a plataforma COW está atualmente a executar transações para a Testnet da rede Hedera, quaisquer taxas aplicadas na execução de transações são suportadas por contas teste que a Hedera disponível.

Por último temos a camada de acesso aos dados que representa o acesso e armazenamento de dados, representado na API como `Repository`. Os repositórios estendem do `JPARespository` para a gestão de dados, incluindo as queries de acesso aos dados, representado no Anexo A.

A API inclui ainda algumas bibliotecas importantes para o desenvolvimento de um serviço escalável como:

- **GRPC-Netty:** Permite o desenvolvimento de servidores distribuídos utilizando gRPC. A biblioteca gRPC-Netty é baseada na estrutura Netty, que é uma estrutura de rede de alto desempenho que permite aos programadores desenvolver aplicações escaláveis.
- **Lombok:** Reduz o código necessário a escrever em aplicações Java. A biblioteca fornece um conjunto de anotações que podem ser utilizadas para gerar código Java automaticamente durante a compilação, tais como *getters* e *setters*,

construtores, e ainda métodos como *hashCode*, *toString*. Através do Lombok, é possível escrever código Java mais conciso e legível.

- **JSON Web Token:** Fornece um conjunto de funcionalidades para a gerar e verificar de JWTs (JSON Web Token) em aplicações Java. Os JWT neste caso são utilizados para implementar autenticação e autorização, bem como para partilhar informação entre micro serviços e outros sistemas distribuídos.

A comunicação entre componentes é realizada através da Restful API, através de pedidos HTTP. A API está documentada com o auxílio do Swagger 17 através de anotações, documentar todos os endpoints da aplicação de forma automática. Para além disso, esta ferramenta dá-nos a opção de fazer pedidos à API graficamente, o que facilitou o desenvolvimento e testes da aplicação.

A figura 18 apresenta todas as rotas da API, que estão documentadas através do Swagger. A tabela representa os diferentes tipos de métodos existentes para cada tipo de controlador, indicando ainda o propósito de cada rota.

### 3.3.3 *Smart Contracts*

É um dos componentes cruciais da plataforma COW, pois sem este tipo de objetos não seria possível armazenar a verdade dos dados na rede Hedera. É a partir do byte code presente na API que é possível gerar um NFT para a entidade bovino, e smart contracts para as restantes entidades da plataforma.

Através do IDE Remix Ethereum, foi possível desenvolver os *smart contracts* necessários para cada uma das entidades referidas no modelo de dados, assim é possível todos os elementos da cadeia terem acesso aos dados verdadeiros. Cada contrato é escrito em Solidity que é a linguagem dos contratos para a Blockchain de Ethereum, o IDE auxilia a escrever, testar e dar *deploy* a cada um dos contratos para diferentes tipos de Blockchains.

A entidade bovina, na rede Hedera possui um *smart contract* que representa o bovino através de um NFT, como evidenciado no Anexo A este *smart contract* implementa dois padrões do OpenZeppelin (ERC-721 e Ownable). O OpenZeppelin é uma biblioteca que contém diversos *standards* com diversos tipos de contratos que permitem desenvolver diferentes *tokens*. O *standard* ERC-721 permite definir um conjunto de regras para o smart contract, criando *tokens* únicos que representam qualquer bem material virtualmente.

O processo de testes e *deploy* do IDE, permite criar um NFT realizando múltiplos pedidos e transações ao contrato, desta forma, é possível perceber as taxas de cada tipo de transação e quais as configurações mais eficientes.

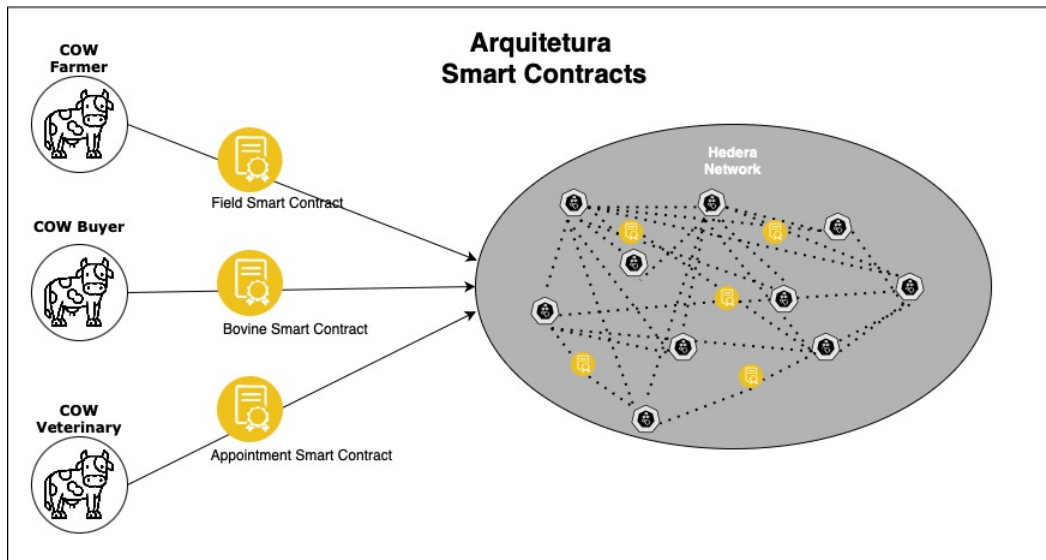


Figura 12: Representação da comunicação dos contratos na rede Hedera

A figura 12 representa como diferentes utilizadores, manipulam diferentes entidades entre a plataforma COW e Hedera. A figura representa ainda que os contratos registados com sucesso, ficam armazenados na rede Hedera sendo esta a fonte de verdade para qualquer elemento da cadeia.

Os smart contracts depois de verificado através de múltiplos testes no Remix, foram compilados e armazenados na API no formato ABI, estes ficheiros possuem o *byte code* do contrato. É a partir deste *byte code* que a API regista um *smart contract* na rede Hedera. O processo é simples e o tempo de *deploy* varia consoante o tamanho e complexidade do contrato.

No contexto de uma supply chain, a existência de smart contracts apresenta diversos vantagens:

1. **Transparência:** Os *smart contracts* permitem rastrear qualquer transação do bovino através do registo e atualização das propriedades como o estado de saúde do bovino. Assim todas as entidades de a cadeia de abastecimento visualizam a mesma informação, reduzindo o risco de fraude e evoluindo a confiança global no sistema.
2. **Eficácia:** Os *smart contracts* conseguem automatizar muitos dos processos de gestão da cadeia de abastecimento, tais como a registo do bovino, rastreio de movimentos dos bovinos, identificação de antigos donos, tornando a cadeia de abastecimento mais eficiente.
3. **Segurança:** Os *smart contracts* são imutáveis, o que significa que uma vez colocados na Blockchain, não podem ser alterados ou eliminados. Assim sendo, fornecem uma forma segura e fiável de registar dados e rastrear qualquer informação acerca da cadeia de abastecimento.

4. **Rastreabilidade:** Os *smart contracts* proporcionam rastreabilidade de *end-to-end* desde a exploração agrícola até ao consumidor. Assim sendo, existe uma maior simplicidade e facilidade em identificar a origem de problemas ou questões na cadeia de abastecimento, tais como surtos de doenças ou preocupações com a segurança alimentar, e permitir uma resposta rápida para prevenir ou mitigar o seu impacto.

Cada contrato foi compilado na versão `pragma solidity ^0.8.9` sendo esta a versão atual que a Hedera especifica para compilar qualquer tipo de contrato. O contrato do bovino é diferente importa dois standards ERC-721 e Ownable. O ERC-721 é o responsável de tornar o contrato em algo transferível e único. O Ownable aplica características de segurança para o controlo de acesso às funcionalidades.

Para garantir segurança sobre os *smart contracts* todas as variáveis são privadas e o contrato envolve um `totalSupply` que vai aumentando dependendo no número de NFT que são gerados.

O standard ERC-721 introduz a função de `mint` que permite criar um NFT e indicar quem é o dono do NFT, sendo neste caso quem executa o código. A função executa um `safeMint` que é uma extensão da função `mint` que garante uma camada extra de segurança, validando quem é o *owner* e o *tokenId* inserido. Este tipo de técnicas previne ataques maliciosos.

Os restantes contratos aplicam os standards de um smart contract, funcionam como um repositório de dados imutável que possui toda a informação de determinado objeto. Qualquer tipo de alteração de dados é verificada, sendo criando um registo das alterações.

Os restantes contratos aplicam os standards do Solidity e funcionam como um repositório de dados imutável que possui toda a informação de determinado objeto.

#### 3.3.4 *Frontend*

A componente visual deste projeto tem como base a framework Vue.js, desenvolvida em Vue, JavaScript, HTML e CSS. A framework Vue.js é utilizada para desenvolver SPAs (Single-Page Applications). O desenvolvimento segue uma arquitetura baseada em componentes, o que significa que se pode criar componentes reutilizáveis e desenvolver interfaces de utilizador complexas facilmente.

A componente visual da plataforma COW centra-se em uma única plataforma desenvolvida através da framework Vue.js. Desta forma, é possível indicar que a plataforma COW é uma SPA.

O projeto inclui ferramentas e plugins que facilitaram o desenvolvimento, como o Vue CLI para gestão do projeto, Vuex para gestão dos estados da página, e Vue Router para gestão de rotas de aplicação.

O Bulma é uma framework CSS que fornece um conjunto de classes CSS pré-construídas para componentes UI, permitindo criar rapidamente componentes de UI visualmente apelativos sem ter de escrever código CSS personalizado. A framework apresenta ainda componentes reativos e classes que facilitam a criação de layouts que funcionam corretamente em diferentes dispositivos.

O Buefy é uma biblioteca de componentes UI para Vue.js que é construída em cima da Bulma. A framework fornece um conjunto de componentes UI pré-construídos, tais como botões, formulários, e notificações, que são fáceis de usar e personalizáveis. Os componentes Buefy são concebidos para funcionar corretamente uns com os outros e com a estrutura Bulma, proporcionando um design consistente e coeso ao longo de toda a aplicação.

A biblioteca Axios apresenta uma API simples e consistente para fazer pedidos HTTP, com métodos HTTP comuns tais como GET, POST, PUT e DELETE. O Axios contém ainda suporte integrado para o tratamento de erros, incluindo códigos de erro HTTP e erros de rede, tornando mais fácil o tratamento de cenários de erro na aplicação.

## DESENVOLVIMENTO

---

Em relação às funcionalidades, foi definido implementar uma solução focada na gestão das cadeias de abastecimento de bovinos como prova de conceito. Os objetivos focam em melhorar e desenvolver mecanismos para facilitar a gestão, orientação e coordenação de bovinos.

### 4.1 AUTENTICAÇÃO E REGISTO DE UTILIZADOR

A plataforma desenvolvida tem em vista o tratamento de informação das cadeias de abastecimento bovino, sendo necessário um sistema de autenticação e registo de um utilizador. Assim sendo, a partir da página inicial o utilizador consegue realizar o seu registo e autenticação.

A autenticação é realizada a partir de um e-mail e palavra-passe, caso as credenciais sejam validas é apresentado o dashboard inicial da plataforma COW (figura 19 e 20 presentes no anexo B). O registo de uma conta é simples através da inserção de nome, e-mail, palavra-passe e o tipo de utilizador que se pretende adicionar.

O formulário de registo apresenta um mecanismo de regras sobre o nível de segurança da palavra-passe, com o mínimo de oito caracteres, um caracter especial, um caracter capitular e um número (figura 21). Além de validações genéricas como do email, sendo que cada erro é apresentado ao utilizador com uma mensagem específica.

O processo de registo de um utilizador, executa imediatamente na API o início de uma transação para o Hedera, sendo criado um *smart contract* do tipo utilizador e enviado para rede Hedera, visto ser necessário armazenar mais informações do utilizador além dos atributos disponíveis num utilizador da Hedera. A rede Hedera funciona como mecanismo de armazenamento dos contratos da plataforma armazenando o *smart contract* e registando um novo utilizador na rede.

Após o registo, os diferentes perfis de utilizador apresentam acesso a um dashboard com todas as funcionalidades que o utilizador tem acesso e ainda gráficos e informações analistas sobre a cadeia de abastecimento.

## 4.2 BOVINO

A entidade bovina surge com diversas funcionalidades, mas apenas utilizadores do tipo agricultor e comprador tem os privilégios de gerir de forma granular um bovino. Deste, criar (figura 22), atualizar, associar bovino a um campo (figura 25), criar marcações de consulta e listar consultas do bovino (figura 26 presente no anexo B).

O detalhe do bovino representado nas figuras 13, 23, 24 indicam diversas características desenvolvidas desde a apresentação da genealogia do bovino, registo de consultas, registo de alterações de localização e registo de donos.

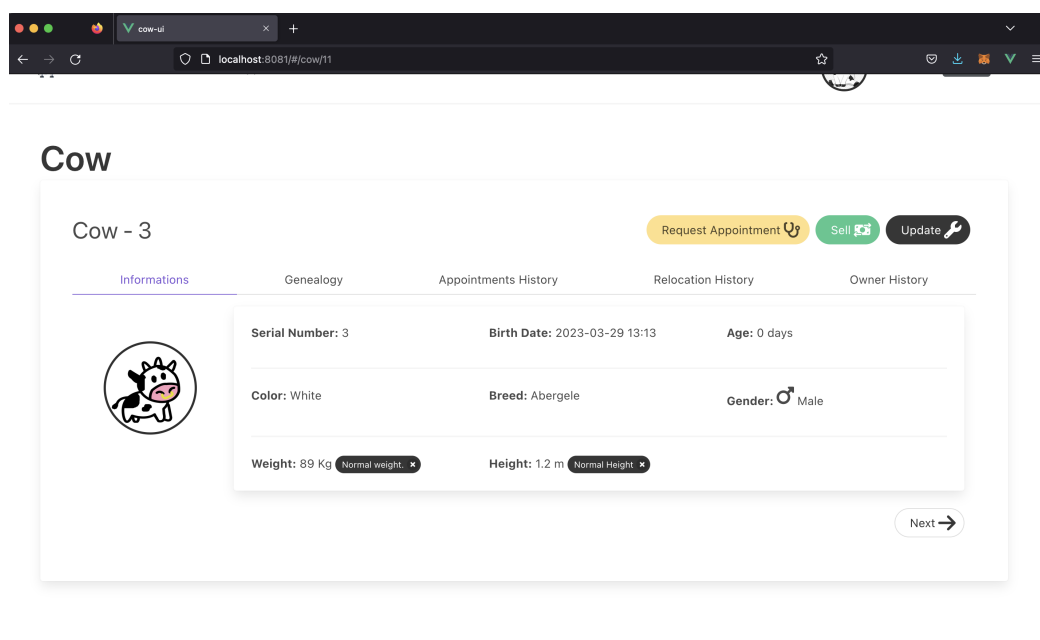


Figura 13: Detalhe do Bovino - Agricultor

O formulário de criação e atualização de um bovino necessita de informações como imagem, número de série, data de nascimento, altura, peso, raça, cor, localização atual e género, sendo possível ainda definir o primeiro nível de ascendência. Os atributos como ascendência, cor e raça, são os atributos utilizados para as estatísticas visualizadas no dashboard.

A lista de bovinos permite verificar todos os NFTs de bovinos registados pelo utilizador ou comprados em leilão presentes na rede Hedera. O pedido é realizado à base de dados sendo que mantém o estado atual da rede Hedera.

## 4.3 CAMPO

A entidade campo surge com o propósito de adicionar um mecanismo de localização e identificação dos diferentes locais que um bovino pode estar. O formulário de

identificação de um campo é preenchido com uma descrição, endereço, observações, coordenadas GPS e limite de bovinos que o campo pode conter. Funcionalidades com atualização de campos e alteração da localização de múltiplos bovinos podem ser acedidas pela página de atualização de campo.

É permitido alterar todos os campos do formulário do campo, tendo em conta as alterações que possam existir num campo, ou até aceitar caso o utilizador tenha previamente realizado algum erro ao inserir. Qualquer tipo de atualização deve ser tratado como uma atualização devido aos acessos e funcionalidades que são executadas nos contratos presentes na rede Hedera.

A lista de campos representada na figura 14 é populada por dados presentes na rede Hedera, através da validação de todos os *smart contracts* criados pelo utilizador. É necessário ter em conta que o acesso a esta informação implica taxas e um custo, sendo que qualquer ação de listagem acede sempre à base de dados local. Desta forma, apenas é necessário atualizar periodicamente um mecanismo de atualização do estado.

O método de atualização e sincronização de dados é realizado através da notificação enviada a cada utilizador sobre o estado alterado de determinado registo. Assim sendo, todas as aplicações são notificadas para obter o novo estado representado na rede Hedera.

Field Name	Address	Max Occupation	Current Occupation	Action
Field Leiria	2027 Rua Hudson	10	30%	Open
Field Porto	1695 Rua Long	5	80%	Open
Field Lisboa	1686 Burwell Heights Road	37	2%	Open

Figura 14: Listagem de Campos Agrícolas - Agricultor

#### 4.4 CONSULTAS

Ao registar consultas que são realizadas, para manter um registo elaborado de consultas foi implementado um mecanismo de marcações de consultas, onde é

identificado um veterinário da plataforma e ainda a data que se pretende marcar a consulta.

O formulário de marcação de consultas é simples, sendo necessário inserir um motivo para a marcação, data da marcação, identificação do bovino e o veterinário selecionado. O utilizador selecionado, recebe imediatamente as marcações na sua plataforma (figura 15) sendo implementado um mecanismo de aceitação. Uma marcação no estado de Aceite, inicia um mecanismo de registo de consultas.

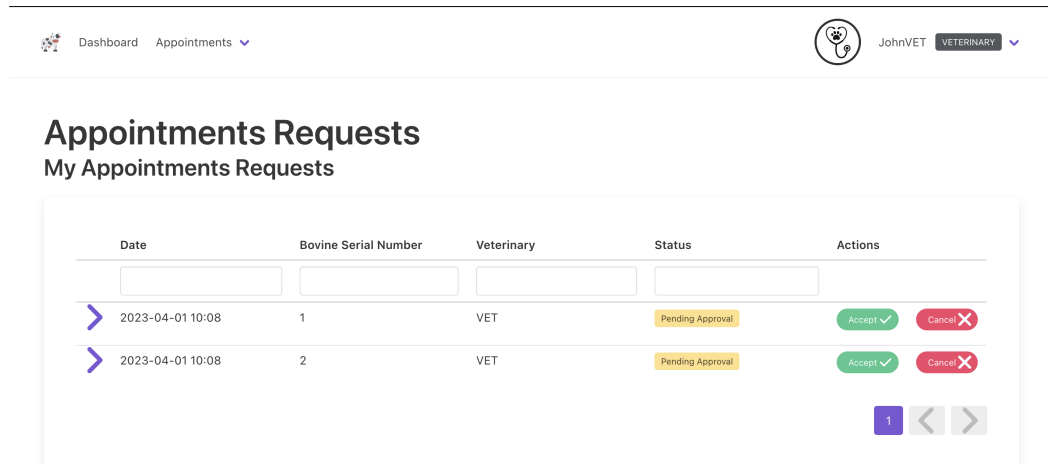


Figura 15: Lista de Marcações de Consultas - Veterinário

A criação da consulta implica o registo de um novo *smart contract* e o registo da transação que permite auditar qualquer consulta. As consultas marcadas são informações importantes para apresentar gráficos no dashboard e as estáticas apresentadas.

As consultas apresentar 3 estados diferentes *Em Espera*, *Pagamento Pendente* e *Concluída*. O veterinário consegue visualizar as consultas já realizadas, através da lista de consultas, sendo apresentado as agendadas e concluídas.

#### 4.5 LEILÃO

Uma vertente do tipo de utilizador comprador, é a compra e venda de bovinos, este tipo de vendas são realizadas entre agricultores, outros comprados do setor alimentar. Os leilões são públicos entre todos os utilizadores do tipo agricultor e comprador, sendo apresentado uma lista de todos os leilões que estão prestes a começar, já começaram e já foram terminados.

O registo de um leilão é realizado através do preenchimento de um simples formulário, com os seguintes dados: descrição, data de início e fim, bovino para venda e preço inicial. Sendo possível atualizar diversas informações do leilão.

O processo do leilão envolve a criação de licitações representadas pela entidade Bid, que por sua vez são identificadas em smart contracts, cada smart contract possui os atributos base de uma licitação, este processo de registo de transações permite criar diferentes tipos de auditoria.

#### 4.6 DASHBOARD E PERFIL

O dashboard apresenta diferentes tipos de estatísticas e permite o acesso ao perfil do utilizador. Os painéis de análise apresentam informações cruciais dos bovinos, leilões e consultas, através de gráficos e dados analíticos dos registos presentes na rede Hedera.

O perfil do utilizador, representado na figura 16 apresenta informações do utilizador como nome completo, nome, data de nascimento, email, tipo de utilizador e saldo atual. A atualização da palavra-passe é permitida através de um processo identico ao de registo na plataforma sendo necessário uma confirmação da operação. Existe ainda a possibilidade de atualizar a imagem de perfil, este mecanismo está diretamente ligado à Web3 Media Platform Pinata.

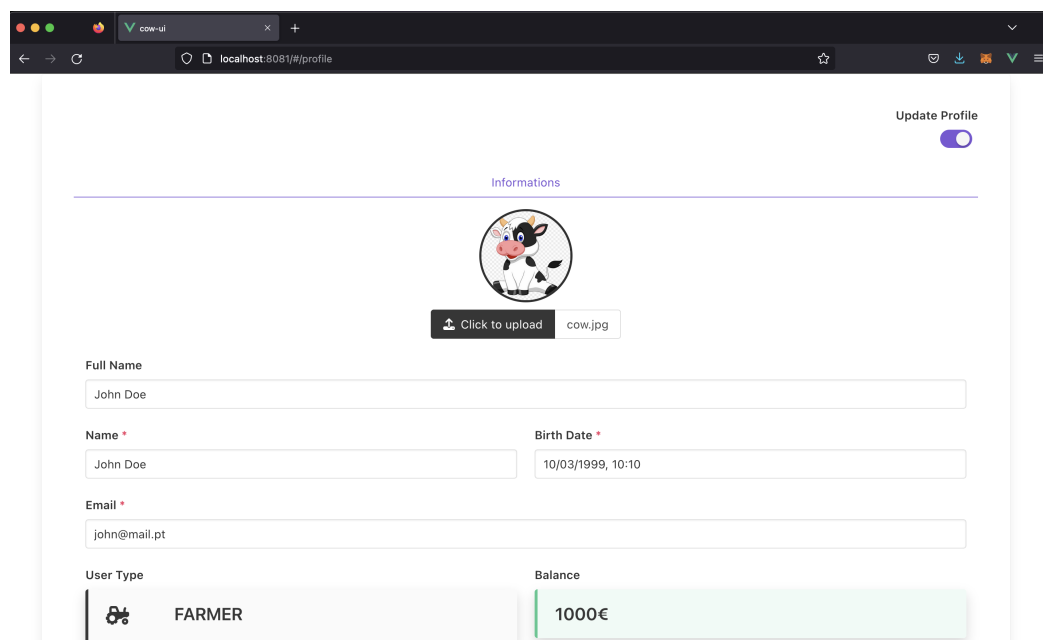


Figura 16: Detalhe do perfil de utilizador

O Pinata é uma plataforma de *media* Web3 que permite aos utilizadores armazenar, gerir e distribuir o seu conteúdo digital numa Blockchain. Permite os utilizadores criar e armazenar os seus bens digitais, tais como imagens, vídeos e música, em

redes descentralizadas como IPFS (InterPlanetary File System) e Ethereum. Isto assegura que o conteúdo é seguro, à prova de adulteração, e acessível a qualquer pessoa com ligação à Internet.

#### 4.7 TESTES

Esta seção, pretende apresentar os testes realizados na plataforma. Tendo em conta que todos os testes realizados aos smart contracts foram realizados a partir do IDE Remix, e os testes às componentes API e frontend basearam-se em diversos utilizadores, experimentarem a aplicação, é possível visualizar o formato dos testes de usabilidade no Anexo C.

A experiência de permitir diferentes utilizadores a utilizar a plataforma, foi extremamente positiva permitindo receber feedback que permitiu uma evolução contínua da plataforma ao longo do desenvolvimento. Esta seção pretende realçar os apresenta uma análise do trabalho desenvolvido e os resultados obtidos.

Durante a realização dos testes de usabilidade, os participantes foram convidados a executar diversas tarefas a partir da plataforma. Desta forma, permitiu validar as funcionalidades da plataforma COW e as suas diferentes áreas (bovinos, agrícolas e gestão de perfil). No decorrer da execução dos testes foi realizado uma análise das interações dos participantes com a aplicação.

Cada teste indica o nome do participante, área de experiência, entre outras informações. Os testes tinham o tempo estimado de 30 minutos, em que cada tarefa apresentava o seu objetivo, duração do participante, se seguiu o caminho esperado, reações do participante, erros apresentados pela plataforma e observações da ação do utilizador. Os testes foram executados por três participantes inexperientes com a aplicação, sempre com uma duração idêntica numa tentativa de realizar cinco tarefas.

Após a conclusão dos testes, os dados recolhidos serviram para identificar áreas da plataforma que precisavam de ser melhoradas. O feedback apresentado nos testes, permitiu alterações à interface do utilizador, modificações do fluxo de tarefas, e até melhorias de funcionalidades.

## CONCLUSÕES

---

Este último capítulo tem como objetivo fazer uma síntese do trabalho realizado, apresentado os objetivos e principais conclusões, e por fim, indicar possíveis evoluções e melhoramentos do projeto.

O projeto apresentava como principal objetivo o desenvolvimento de uma plataforma com respostas para a resolução dos problemas de gestão das cadeias de abastecimento. A plataforma COW, desenvolvida ao longo do processo pretendia a adição de diferentes entidades na cadeia, apresentando um sistema descentralizado, que permite a partilha de transações, de forma transparente.

Com esta premissa, após explorar e compreender o estado de arte atual das cadeias de abastecimento e o estado de desenvolvimento das tecnologias DLT e Blockchain, estavam tomadas as decisões necessárias para desenvolver um sistema prático e como uma arquitetura que permita alcançar a escalabilidade necessária nestes sistemas.

Inicialmente, após a análise do estado de arte das cadeias de abastecimento e rever a falta de informação técnica presente nos projetos, estava encontrado o primeiro problema na área. A exploração permitiu a identificação de um problema presente na área que é a falta de informação, este é um pormenor importante que tem de ser alterado, a informação dos projetos reais tem de evoluir. O facto de apenas projetos académicos partilharem as tecnologias aplicadas, métodos de trabalho e decisões técnicas para aplicar a lógica de negócio dificulta a adoção das tecnologias DLT e Blockchain.

A seleção de utilizar o Hedera Hashgraph foi essencial para o progresso e desenvolvimento do projeto. A tecnologia apresentou os conceitos aplicados e desenvolvidos numa API, permitindo a conceção de aplicações descentralizadas, e soluções baseadas em Blockchain. O mecanismo de consenso Hashgraph verificou-se uma vantagem no desenvolvimento da plataforma, permitindo equilibrar as necessidades de cada tipo de utilizador, mantendo as exigências necessárias para a gestão da cadeia de abastecimento dos bovinos.

O desenvolvimento da plataforma revelou-se um sucesso devido à quantidade de trabalho e funcionalidade apresentadas. A existência de múltiplos tipos de utilizador é um ponto chave da plataforma permitindo a criação de diversas entidades da cadeia. A evolução das funcionalidades em torno da entidade bovina, também foi

um dos pontos chave do desenvolvimento, no sentido de sempre focar no ciclo de vida de um bovino. A identificação de casos reais cujo indicam que informações como localização, saúde, compra e venda de bovinos seriam das funcionalidades básicas para qualquer cadeia de abastecimento permitiu um desenvolvimento mais informado e focado.

A plataforma foi desenvolvida para ser genérica facilitando a sua escalabilidade e alteração de caso de estudo, permitindo a introdução de qualquer caso de uso real. Deste modo, a plataforma foi desenvolvida permitindo a sua evolução e adição de novas funcionalidades, novos tipos de utilizador, e outras lógicas de negócio.

A decisão de adicionar o conceito NFT foi uma evolução final da plataforma e dos *smart contract*. O desenvolvimento de *tokens*, potencia o sistema de compras e vendas de bovinos, criando *tokens* únicos que representam bovino real no meio digital.

## 5.1 RESULTADOS

A componente da API, Frontend e Smart contracts está disponível em repositórios públicos no GitHub com o intuito de partilhar o conhecimento obtido do projeto e também os mecanismos de utilização da rede Hedera.

- Repositório cow-api (<https://github.com/bernardo-rf/cow-api>)
- Repositório cow-ui (<https://github.com/bernardo-rf/cow-ui>)
- Repositório cow-smart-contracts ([https://github.com/bernardo-rf/cow-smart\\_contracts](https://github.com/bernardo-rf/cow-smart_contracts))

Além disso, é importante relembrar o desenvolvimento de um artigo científico denominado de *COW: A Proof-of-Concept DLT Platform For The Agri-Food Supply Chain*, submetido e aceite na *IEEE International Conference on Artificial Intelligence, Blockchain, and Internet of Things, (AIBThings)*. O artigo apresenta uma análise do estado de arte da área Agro-Precision, informação técnica e detalhada do desenvolvimento da plataforma COW e decisões de implementação. O artigo científico surge com o objetivo de expor mais informação sobre a adição de tecnologias DLT em *supply chains*.

## 5.2 LIMITAÇÕES

É importante identificar algumas das limitações do projeto, inicialmente a falta de informação técnica dos casos de uso reais, e a complexidade inerente das cadeias de abastecimento, que dificultam o desenvolvimento.

É necessário indicar que uma das componentes iniciais do projeto era a adição de sensores IoT em bovinos reais, mas rapidamente foi identificado que a entidade externa que iria desenvolver estes componentes e serviços que não se encontrava disponível, devido a atrasos internos. Tendo em conta estas informações, a decisão foi de avançar com o projeto numa vertente de introduzir diferentes entidades na cadeia de abastecimento com diferentes funcionalidades além do agricultor.

Por último, é necessário evidenciar que seria do interesse comum apresentar e desenvolver mais testes de usabilidade e evoluir para a adição de testes unitários, mas devido a limitações de tempo não foram possíveis de desenvolver. Pretendo prosseguir no desenvolvimento e adicionar no futuro do projeto.

## 5.3 TRABALHO FUTURO

Sendo que o projeto foca num caso de uso específico como as cadeias de abastecimento bovino, a agregação a um sistema em produção, com o apoio de empresas reais acrescentaria muito valor ao projeto. Este tipo de planeamento será acautelado no futuro permitindo a plataforma evoluir com novas funcionalidades e novas regras de negócio.

A integração de sensores IoT e serviços de análise na plataforma é visto com uma funcionalidade que necessitam de ser adicionadas, com o objetivo de evoluir o projeto COW. Esta evolução permitiria a digitalização da informação recolhida, podendo desta forma ser inserida e analisada em tempo real. Permitindo assim aumentar a qualidade das decisões tomadas a partir da informação recolhida.

Além disso, adicionar novas funcionalidades a cada tipo de utilizador é um desejo para o futuro, permitindo adicionar novos intermediários na cadeia, adicionando novas funcionalidades focadas na nutrição e qualidade dos bovinos.

Como indicado anteriormente, a adição de testes unitários no frontend e API é uma prioridade e será o primeiro passo para diminuir o risco de falha e auxiliar a adoção da plataforma.



## BIBLIOGRAFIA

---

- [1] M. Aleksa, B. Maric e M. Knezevic. “Performance of Hedera Hashgraph consensus algorithm”. Em: *Proceedings of the 27th Telecommunications Forum (TELFOR)*. 2019, pp. 1–4. DOI: 10.1109/TELFOR49367.2019.8978723.
- [2] Merlinda Andoni et al. “Blockchain technology in the energy sector: A systematic review of challenges and opportunities”. Em: *Renewable and Sustainable Energy Reviews* 100 (2019), pp. 143–174.
- [3] Australian Tea Tree Industry Association. *Generating value with manufactures and protecting producers and consumers: Using Blockchain in the Australian Tea Tree Oil Industry*. [https://www.attia.org.au/images/Articles/Blockchain\\_ATTII\\_Summary\\_3\\_Feb\\_2020.pdf](https://www.attia.org.au/images/Articles/Blockchain_ATTII_Summary_3_Feb_2020.pdf). [Accessed:[Accessed: 07-Apr-2022]]. 2020.
- [4] Arshdeep Bahga e Vijay K. Madiseti. “Blockchain Platform for Industrial Internet of Things”. Em: *Journal of Software Engineering and Applications* 9 (out. de 2016), 10.4236/jsea.2016.910036. DOI: 10.4236/jsea.2016.910036.
- [5] Mandrita Banerjee, Junghee Lee e Kim-Kwang Raymond Choo. “A blockchain future for internet of things security: a position paper”. Em: *Digital Communications and Networks* 4 (2018), pp. 149–160.
- [6] Ali Bastas e Kapila Liyanage. “Sustainable supply chain quality management: A systematic review”. Em: *Journal of Cleaner Production* 181 (2018), pp. 726–744.
- [7] Kay Behnke e MFWH Anton Janssen. “Boundary conditions for traceability in food supply chains using blockchain technology”. Em: *Computers in Industry* 52 (2020). DOI: 10.1016/j.compind.2019.103180.
- [8] Omar Bermeo-Almeida et al. “Blockchain in Agriculture: A Systematic Literature Review”. Em: *International Conference on Human-Computer Interaction*. 2018, pp. 42–52. DOI: 10.1007/978-3-030-00940-3\_4.
- [9] Nikola Bozic, Guy Pujolle e Stefano Secci. “A tutorial on blockchain and applications to secure network control-planes”. Em: *2016 3rd Smart Cloud Networks & Systems (SCNS)*. 2016, pp. 1–8. DOI: 10.1109/SCNS.2016.7870552.
- [10] M. P. Caro et al. “Blockchain-based traceability in Agri-Food supply chain management: A practical implementation”. Em: (2018), pp. 1–4. DOI: 10.1109/IOT-TUSCANY.2018.8373021.

- [11] M. P. Caro et al. “Blockchain-based traceability in Agri-Food supply chain management: A practical implementation”. Em: (2018), pp. 1–4. DOI: 10.1109/IOT-TUSCANY.2018.8373021.
- [12] G. R. Chandra, I. A. Liaqat e B. Sharma. “Blockchain Redefining: The Halal Food Sector”. Em: (2019), pp. 349–354. DOI: 10.1109/AICAI.2019.8701321.
- [13] Huanhuan Feng et al. “Modeling and evaluation on WSN-enabled and knowledge-based HACCP quality control for frozen shellfish cold chain”. Em: 98 (2018), pp. 348–358.
- [14] Huanhuan Feng et al. “Modeling and evaluation on WSN-enabled and knowledge-based HACCP quality control for frozen shellfish cold chain”. Em: 98 (2019), pp. 38–358.
- [15] Tiago M Fernández-Caramés e Paula Fraga-Lamas. “A Review on the Use of Blockchain for the Internet of Things”. Em: *IEEE Access* 6 (2018), pp. 32979–33001. DOI: 10.1109/ACCESS.2018.2842685.
- [16] M. Ferreira et al. “Blockchain: A Tale of Two Applications”. Em: *Applied Sciences* 8.9 (set. de 2018), p. 1506. DOI: 10.3390/app8091506. URL: <http://dx.doi.org/10.3390/app8091506>.
- [17] Juan F Galvez, JC Mejuto e J Simal-Gandara. “Future challenges on the use of blockchain for food traceability analysis”. Em: *Trends in Food Science & Technology* 71 (2018), pp. 246–257.
- [18] Geora. *Building the future of Agriculture*. <https://www.geora.io/>. [Accessed: 07-Apr-2022]. n.d.
- [19] Muneeb Ul Hassan, Mubashir Husain Rehmani e Jinjun Chen. “Privacy preservation in blockchain based IoT systems: Integration issues, prospects, challenges, and future research directions”. Em: *Journal of Network and Computer Applications* 137 (2019), pp. 20–48.
- [20] Hedera Hashgraph LLC. *Hedera Hashgraph*. <https://www.hedera.com/>. Accessed: March 31, 2023. 2021.
- [21] A. Helium e B. Researcher. “Exploring the Benefits of Helium”. Em: *Journal of Wireless Communications* 10.3 (2022), pp. 25–35. DOI: 10.1007/s11235-022-01234-z. URL: <https://link.springer.com/article/10.1007/s11235-022-01234-z>.
- [22] Petri Helo e Yuqiuge Hao. “Blockchains in operations and supply chains: A model and reference implementation”. Em: *International Journal of Production Economics* 136 (2019), pp. 242–251.

- [23] Pedro roseiro Javier Dominguez. “Blockchain: A brief review of AgriFood Supply Chain Solutions and Opportunities”. Em: *ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal* 9 (2020), pp. 95–106. DOI: 10.14201/ADCAIJ20209495106.
- [24] Minhaj Ahmad Khan e Khaled Salah. “IoT security: Review, blockchain solutions, and open challenges”. Em: *Future Generation Computer Systems* 82 (2018), pp. 395–411.
- [25] Suleman Khan. “IoT with BlockChain: A Futuristic Approach in Agriculture and Food Supply Chain”. Em: *Journal of Smart Agriculture* 2021 (jun. de 2021), pp. 1–14.
- [26] Henry M. Kim e Marek Laskowski. *Agriculture on the Blockchain: Sustainable Solutions for Food, Farmers, and Financing*. 2017.
- [27] W. Lin et al. “Blockchain Technology in Current Agricultural Systems: From Techniques to Applications”. Em: 8 (2020), pp. 143920–143937. DOI: 10.1109/ACCESS.2020.3014522.
- [28] A. Mahmoud e A. Al-Fares. *An Overview of the Hedera Hashgraph Consensus Algorithm*. Rel. téc. KAUST-CS18-01. King Abdullah University of Science e Technology, mai. de 2018. URL: <https://hdl.handle.net/10754/627501>.
- [29] M. Oliveira et al. “Immunity passport ledger”. Em: *Innovations in Bio-Inspired Computing and Applications*. 2022, pp. 527–536.
- [30] Svein Ølnes, Jolien Ubacht e Marijn Janssen. “Blockchain in government: Benefits and implications of distributed ledger technology for information sharing”. Em: 34 (2017), pp. 355–364.
- [31] Petter Olsen e Melania Borit. “The components of a food traceability system”. Em: *Food Control* 77 (2018), pp. 143–149.
- [32] Ionut Emanuel Pirlea et al. “Security Analysis of IOTA Tangle”. Em: *2020 21st International Symposium on Signals, Circuits and Systems (ISSCS)*. 2020, pp. 1–4. DOI: 10.1109/ISSCS51271.2020.9164764.
- [33] Randy Priem. “Distributed ledger technology for securities clearing and settlement: benefits, risks, and regulatory implications”. Em: (2020). DOI: 10.1186/s40854-019-0169-6.
- [34] Timo Schmid, Daniel Seidel e Dietrich Gronau. “Blockchain Technology in Logistics - An Explorative Study on Potentials and Challenges”. Em: *Proceedings of the 51st Hawaii International Conference on System Sciences*. 2018, pp. 2463–2472. DOI: 10.24251/HICSS.2018.308.
- [35] Shengnan Sun, Xinpeng Wang e Yan Zhang. “Sustainable Traceability in the Food Supply Chain: The Impact of Consumer Willingness to Pay”. Em: *Sustainability* 9.5 (2017), p. 774.

- [36] Feng Tian. “A supply chain traceability system for food safety based on HACCP, blockchain & Internet of things”. Em: *2017 International Conference on Service Systems and Service Management*. 2017, pp. 1–6. DOI: 10.1109/ICSSSM.2017.7996119.
- [37] Frank Yiannas. “A New Era of Food Transparency Powered by Blockchain”. Em: *Forbes* 12 (2018), pp. 46–56.
- [38] Guoqing Zhao, Shaofeng Liu e Carmen Lopez. “A Literature Review on Risk Sources and Resilience Factors in Agri-Food Supply Chains”. Em: *International Journal of Production Economics* 183 (2017), pp. 739–754.
- [39] Guoqing Zhao et al. “Blockchain technology in agri-food value chain management: A synthesis of applications, challenges and future research directions”. Em: *Computers and Electronics in Agriculture* 109 (2019), pp. 83–99.
- [40] Aviv Zohar. “IOTA - A Cryptocurrency for the Internet-of-Things”. Em: *Proceedings of the 7th ACM Conference on Data and Application Security and Privacy*. 2017, pp. 189–198. DOI: 10.1145/3098954.3098966.

## APÊNDICES





## ANEXO A

---

```
/*
 *
 * @Copyright 2023 Bernardo Figueiredo, POLITÉCNICO DE LEIRIA.
 *
 */

// SPDX-License-Identifier: UNLICENSED
pragma solidity ^0.8.9;

import "@openzeppelin/contracts/token/ERC721/ERC721.sol";
import "@openzeppelin/contracts/access/Ownable.sol";

contract Bovine is ERC721, Ownable{

    string private idOwner;

    uint256 private idField;

    uint256 private serialNumber;

    uint256 private birthDate;

    bool private active;

    string private observation;

    uint256 private idBovineParentOne;

    uint256 private idBovineParentTwo;

    bool private gender;

    uint256 private totalSupply = 0;
```

```

constructor(
    string memory _idOwner,
    uint256 _idField,
    uint256 _serialNumber,
    uint256 _birthDate,
    bool _active,
    string memory _observation,
    uint256 _idBovineParentOne,
    uint256 _idBovineParentTwo,
    bool _gender ) ERC721('CBT', 'CBT') {
    mint( _idOwner,
        _idField,
        _serialNumber,
        _birthDate,
        _active,
        _observation,
        _idBovineParentOne,
        _idBovineParentTwo,
        _gender);
}

function mint public (
    string memory _idOwner,
    uint256 _idField,
    uint256 _serialNumber,
    uint256 _birthDate,
    bool _active,
    string memory _observation,
    uint256 _idBovineParentOne,
    uint256 _idBovineParentTwo,
    bool _gender) {
    idOwner = _idOwner;
    idField = _idField;
    serialNumber = _serialNumber;
    birthDate = _birthDate;
    active = _active;
    observation = _observation;
    idBovineParentOne = _idBovineParentOne;
    idBovineParentTwo = _idBovineParentTwo;
    gender = _gender;
}

```

```

        uint256 tokenId = totalSupply + 1;
        _safeMint(msg.sender, tokenId);
    }

function setUpdate(
    uint256 _idField,
    uint256 _serialNumber,
    uint256 _birthDate,
    bool _active,
    uint256 _idBovineParentOne,
    uint256 _idBovineParentTwo,
    bool _gender) public {
    require(ownerOf(_tokenId) == msg.sender);
    idField = _idField;
    serialNumber = _serialNumber;
    birthDate = _birthDate;
    active = _active;
    idBovineParentOne = _idBovineParentOne;
    idBovineParentTwo = _idBovineParentTwo;
    gender = _gender;
}

function setIDOwner(string memory _idOwner) public {
    require(ownerOf(_tokenId) == msg.sender);
    idOwner = _idOwner;
}

function getIDOwner() external view returns (string memory) {
    require(ownerOf(_tokenId) == msg.sender);
    return idOwner;
}

...

function setActive(bool _active) public {
    require(ownerOf(_tokenId) == msg.sender);
    active = _active;
}

function getActive() external view returns (bool) {

```

```

        require(ownerOf(_tokenId) == msg.sender);
        return active;
    }

    ...

    function setObservation(string memory _observation) public {
        require(ownerOf(_tokenId) == msg.sender);
        observation = _observation;
    }

    function getObservation() external view returns (string memory) {
        require(ownerOf(_tokenId) == msg.sender);
        return observation;
    }

    ...
}

/*
 *
 * @Copyright 2023 Politécnico de Leiria, @bernardo-rf.
 *
 */
-- Endpoint disponibilizado pelo BovineController

...

@GetMapping("/")
@ApiOperation("Get bovines")
public ResponseEntity<List<BovineDTO>> getBovinesActive() {
    List<Bovine> bovines = bovineService.getBovinesActive();
    return ResponseEntity.ok(bovineMapper.mapSourceListToTargetList(bovines));
}

/*
 *
 * @Copyright 2023 Politécnico de Leiria, @bernardo-rf.
 *
 */

```

```

-- Função registo de um ficheiro na rede Hedera disponibilizado pelo BovineService
...

private HederaReceipt getBovineDeployReceipt(Client client, FileId fileId) throws Time
    try {
        return buildBovineDeployReceipt(client, fileId);
    } catch (ReceiptStatusException e) {
        validateGas(e);
        throw new ErrorCodeException(ErrorCode.BOVINE_DEPLOY_FAILED);
    } catch (PrecheckStatusException e) {
        throw new ErrorCodeException(validateErrorCode(e, ErrorCode.BOVINE_DEPLOY_F
    }
}

/*
 *
 * @Copyright 2023 Politécnico de Leiria, @bernardo-rf.
 *
 */

-- Função acesso a determinado bovino BovineRepository
...

public interface BovineRepository extends JpaRepository<Bovine, Long> {

    @Query("SELECT b FROM Bovine b WHERE b.id = :id and b.active = true ORDER BY b.id A
    Bovine getBovineById(long id);

```



## ANEXO B

---

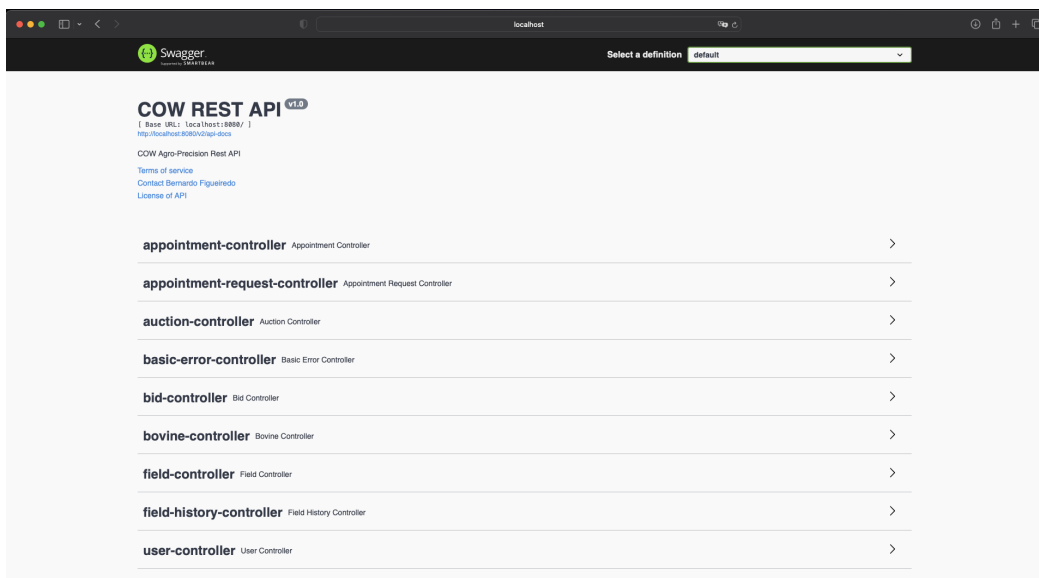


Figura 17: COW API - Swagger

Rotas - REST API			
Tipo	Método	Rota	Descrição
Appointment	POST	/api/appointments/	Create a appointment
	GET	/api/appointments/{appointmentId}	Get a appointment by id
	PUT	/api/appointments/{appointmentId}	Update a appointment
	DELETE	/api/appointments/{appointmentId}	Delete a appointment
	PUT	/api/appointments/{appointmentId}/status	Update appointment status
	GET	/api/appointments/bovine/{bovineId}	Get all appointments of a bovine
	GET	/api/appointments/bovines/{userWallet}	Get all appointments of all owned cows
Appointment Request	GET	/api/appointments/user/{userId}	Get all appointments by user id
	POST	/api/appointmentsRequest/	Create a appointment request
	GET	/api/appointmentsRequest/{appointmentRequestId}	Get a appointment request by id
	PUT	/api/appointmentsRequest/{appointmentRequestId}	Update a appointment request
	DELETE	/api/appointmentsRequest/{appointmentRequestId}	Delete a appointment request
	PUT	/api/appointmentsRequest/{appointmentRequestId}/status	Update appointment request status
	GET	/api/appointmentsRequest/bovine/{bovineId}	Get all appointments request of a bovine
Auction	GET	/api/appointmentsRequest/user/{userId}	Get all appointments request by user
	GET	/api/appointmentsRequest/userRequest/{userRequestId}	Get all appointments request by user request
	GET	/api/auctions/	Get all auctions
	POST	/api/auctions/	Create a auction
	GET	/api/auctions/{auctionId}	Get auction by auction id
	PUT	/api/auctions/{auctionId}	Update a auction
	PUT	/api/auctions/{auctionId}/status	Update a auction status
Bid	GET	/api/bids/	Get all bids
	POST	/api/bids/	Get bid by bid id
Bovine	GET	/api/bids/{bidId}	Create a bid
	POST	/api/bovines/	Create a bovine
	GET	/api/bovines/	Get all bovines
	GET	/api/bovines/{bovineId}	Get a bovine by id
	PUT	/api/bovines/{bovineId}	Update a bovine
	DELETE	/api/bovines/{bovineId}	Delete a bovine
	GET	/api/bovines/{userWallet}/own	Get all appointments of a bovine
Field	GET	/api/bovines/{userWallet}/own/auction	Get all appointments of all owned cows
	POST	/api/bovines/genealogy/{bovineId}	Get all appointments by user
	POST	/api/fields/	Create a field
	GET	/api/fields/{fieldId}	Get a field by id
	PUT	/api/fields/{fieldId}	Update a field
	DELETE	/api/fields/{fieldId}	Delete a field
	GET	/api/fields/{userWallet}/full-info	Get all fields owned by user wallet full info
Field History	GET	/api/fields/{userWallet}/not-occupied	Get all fields owned by user wallet not occupied
	GET	/api/fields/{fieldId}/bovines	Get field bovines by field id
	GET	/api/fields/{fieldId}/bovines/not-in	Get field bovines not in by field
	POST	/api/historyFields/	Create a field history record
	GET	/api/historyFields/{bovineId}	Get all history fields records by bovine id
	PUT	/api/historyFields/{fieldHistoryId}	Update a field history record
	User	POST	/api/users/
GET		/api/users/	Get all users
GET		/api/users/{userId}	Get user by user id
PUT		/api/users/{userWallet}	Update user by userWallet
POST		/api/users/auth	Authenticate user
GET		/api/users/email	Get user by email
GET		/api/users/veterinary	Get all users veterinaries
User Type	GET	/api/users/wallet	Get user by wallet
	POST	/api/userTypes/	Create a user type
	GET	/api/userTypes/	Get all user types
	PUT	/api/userTypes/{userTypeId}	Get user type by user type id

Figura 18: COW API - Rotas

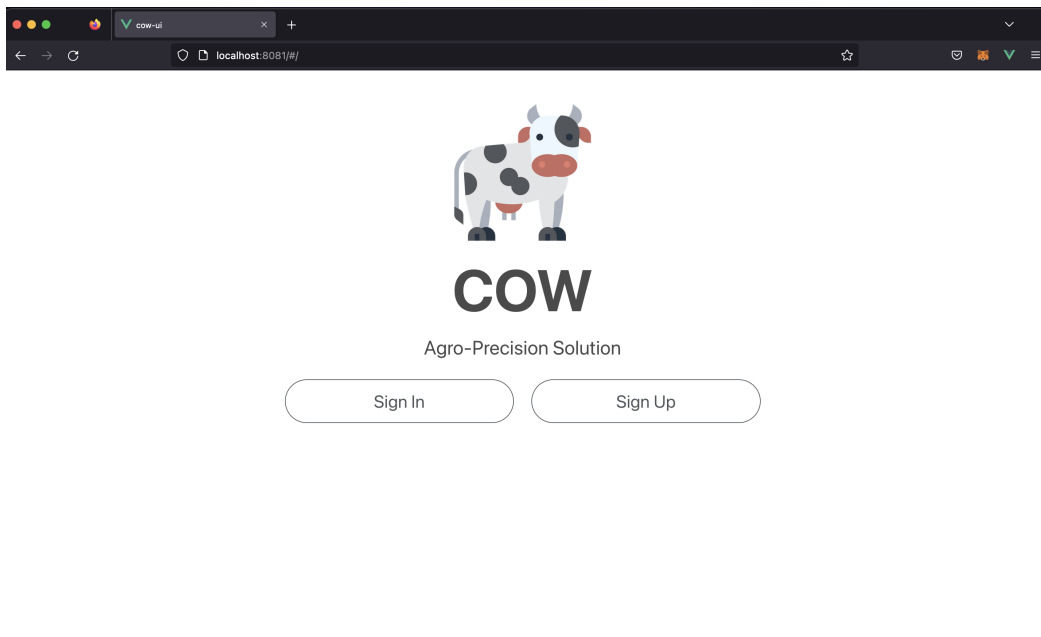


Figura 19: Landing Page

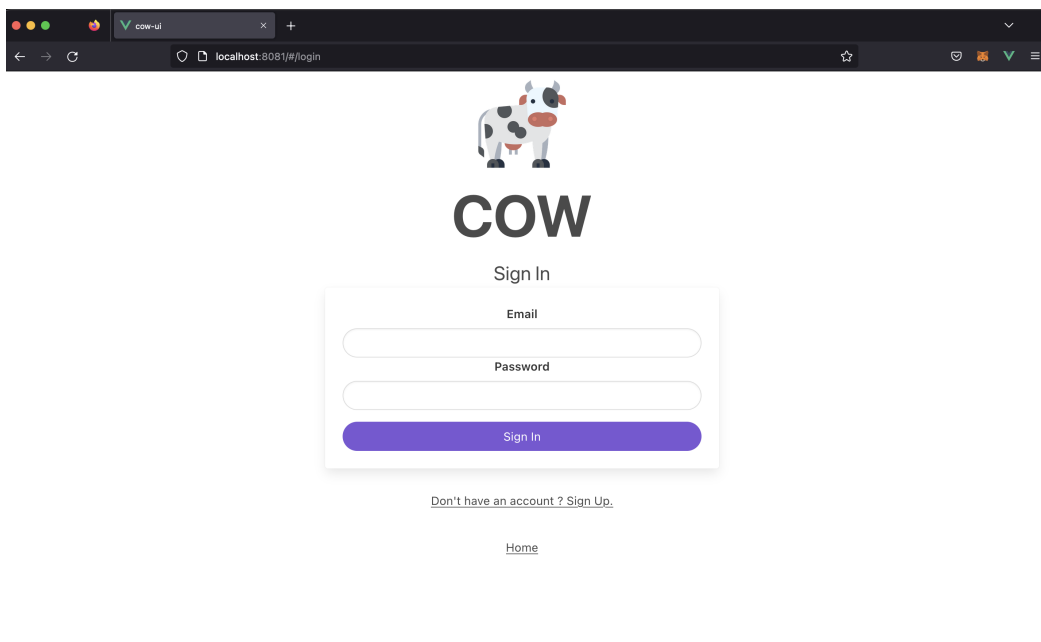


Figura 20: Página de Autenticação

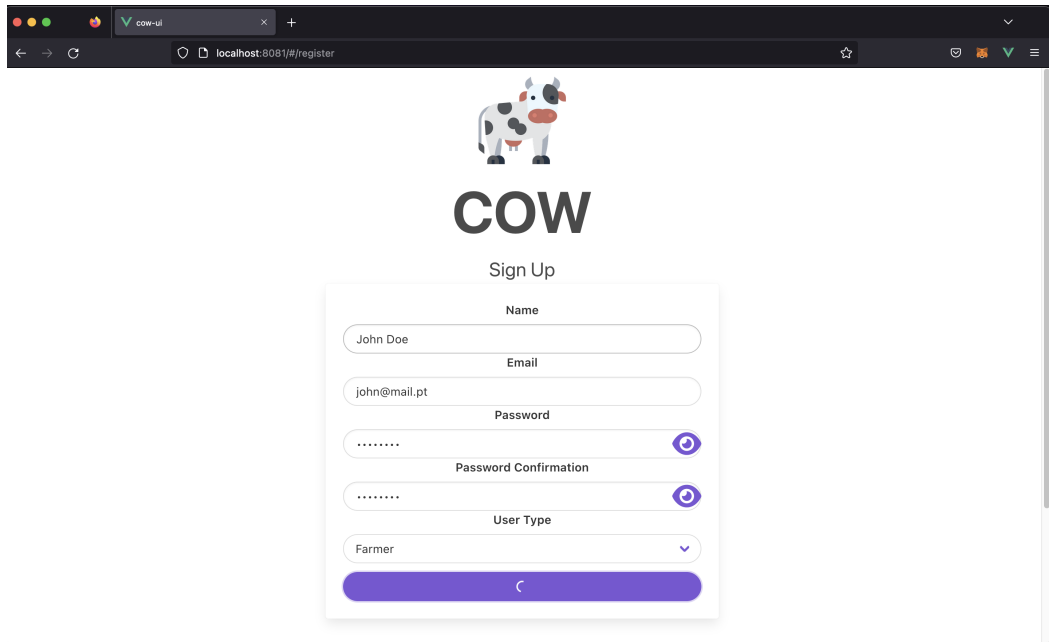


Figura 21: Página de Registo

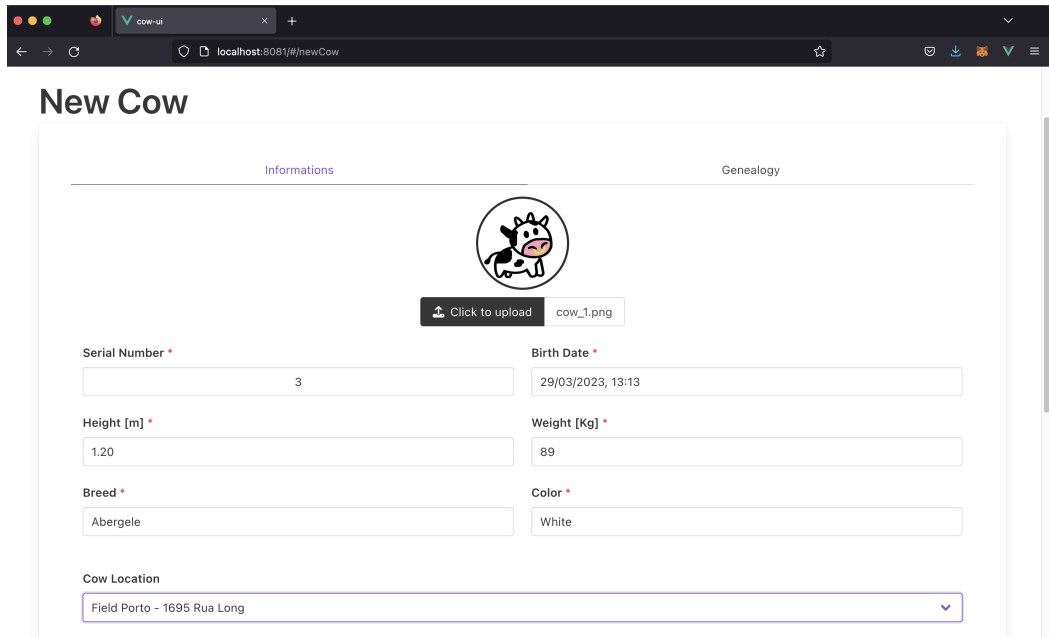


Figura 22: Criação de NFT do Bovino

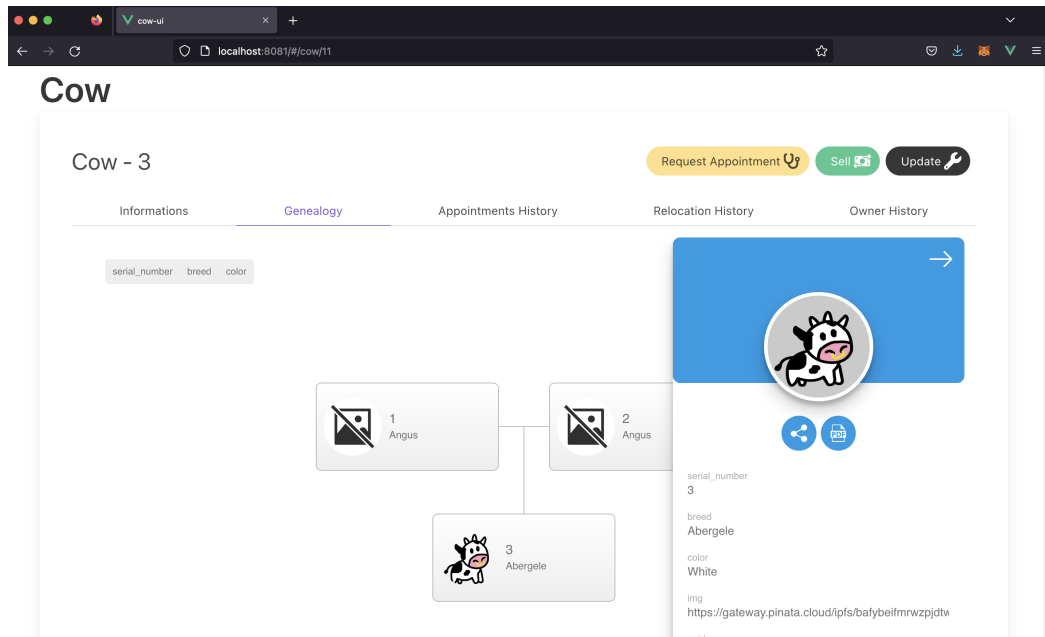


Figura 23: Detalhe NFT Bovino - Parte 2

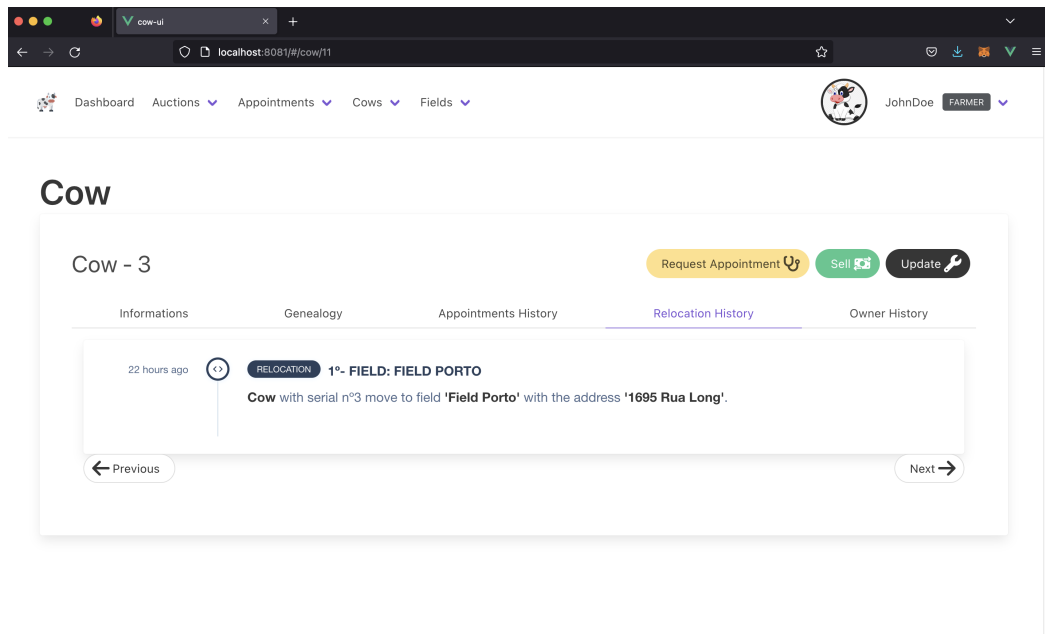


Figura 24: Detalhe NFT Bovino - Parte 3

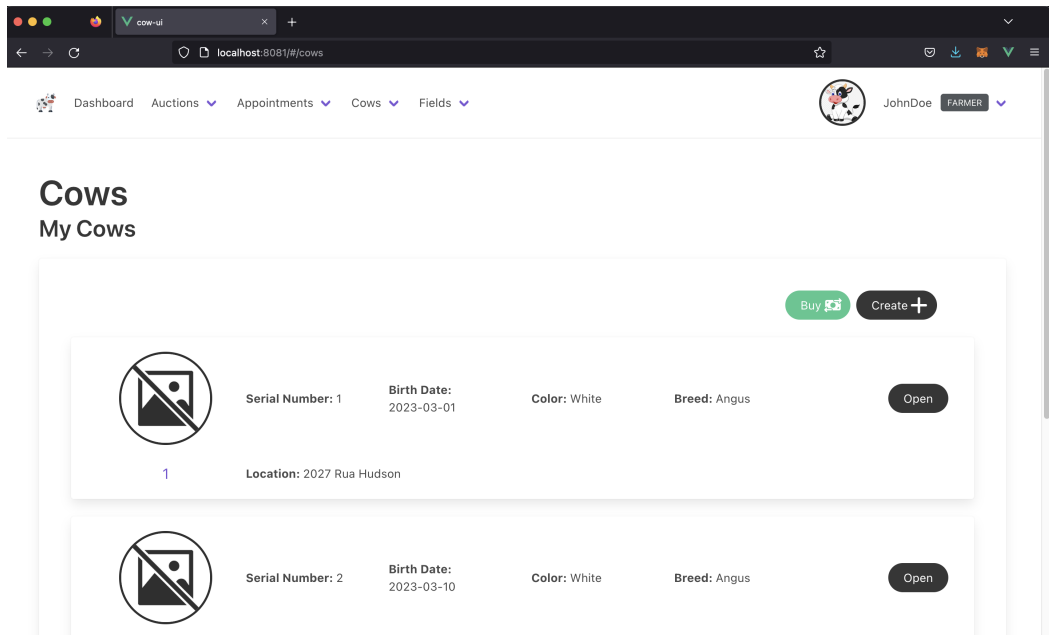


Figura 25: Lista de Bovinos

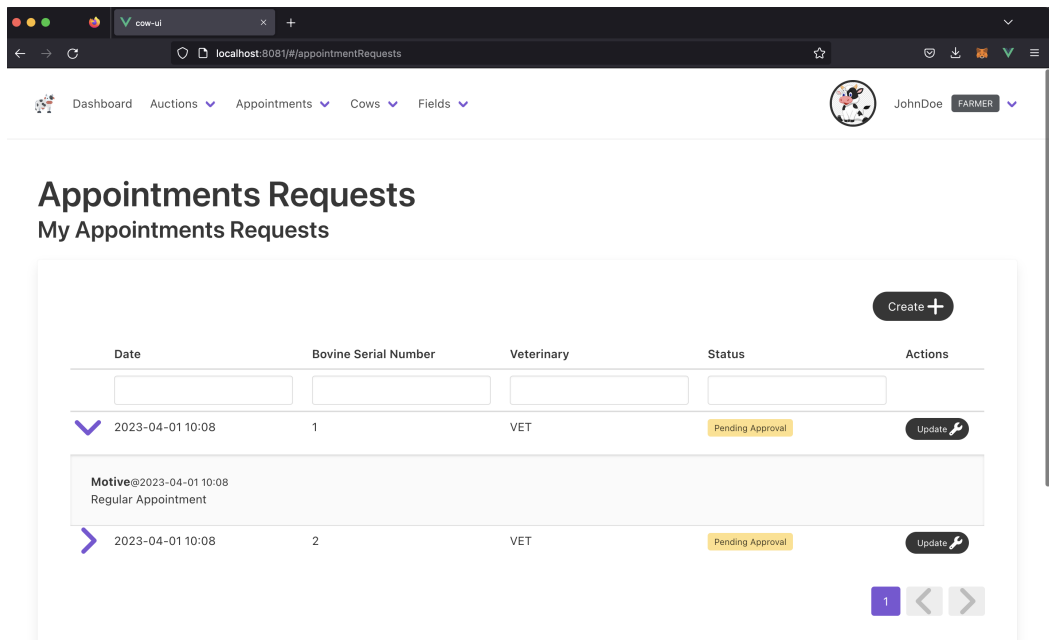


Figura 26: Listagem de Marcações - Farmer

# C

## ANEXO C

---

## Record Sheet for the Observer - 1 of 3 (3)

Nome:	Beatriz Santos			Área de Experiência:	Nutrição
Gender:	F	Data de Nascimento:	N/D	Número de Telemóvel:	N/D

Tempo Inicial:	21:32	Tempo Final:	21:58
----------------	-------	--------------	-------

<b>Tarefa #1</b> Registo de um utilizador do tipo Farmer. Consegue-me indicar como se autenticar na aplicação?	
<b>Duração</b> Quanto tempo demorou o utilizador a concluir a tarefa?	2 min
<b>Expected Path</b> Seguiu o processo esperado?	Sim
<b>[Expected Path]</b> Página Inicial – Clicar botão de Registo – Preencher formulário na página de Registo - Clicar Registrar Utilizador – Entrar na aplicação - Terminar sessão – Selecionar botão de autenticação na página inicial - Preencher formulário com credenciais válidas na página de autenticação - Clicar botão de autenticação na página de autenticação - Ver utilizador autenticado dentro da plataforma.	
<b>Reação</b> Quais foram os comentários ou expressões do utilizador?	Formulários simples; Navegação de páginas rápida; Processo de registo mais lento do que o esperado.
<b>Erros</b> Quantos cliques o utilizador realizou em sítios errados?	0
<b>User está PERDIDO</b> O que o utilizador esperou a aplicação indicar?  R: Utilizador não se sentiu perdido. Esperava apenas que o registo fosse mais rápido. Após explicar o sistema por detrás da aplicação entendeu o motivo.	

Figura 27: Representação dos Testes de Usabilidade

## DECLARAÇÃO

---

Declaro, sob compromisso de honra, que o trabalho apresentado nesta dissertação, com o título “*AGRO PRECISION – A FRAMEWORK FOR PRECISION AGRICULTURE ON AN IOT BLOCKCHAIN*”, é original e foi realizado por Bernardo José Ribeiro Figueiredo (2202295) sob orientação de Catarina I. Reis.

*Leiria, Março de 2023*



---

Bernardo José Ribeiro Figueiredo