

UNIVERSIDADE FEDERAL DE ITAJUBÁ - UNIFEI
PROGRAMA DE PÓS-GRADUAÇÃO EM
CIÊNCIA E TECNOLOGIA DA COMPUTAÇÃO

eCattle: Uma Plataforma para Rastreamento
e Gerenciamento de Registros Pecuários
Utilizando Contratos Inteligentes

Nilton Raimundo de Assis Júnior

Itajubá, 29 de agosto de 2024

**UNIVERSIDADE FEDERAL DE ITAJUBÁ - UNIFEI
PROGRAMA DE PÓS-GRADUAÇÃO EM
CIÊNCIA E TECNOLOGIA DA COMPUTAÇÃO**

Nilton Raimundo de Assis Júnior

**eCattle: Uma Plataforma para Rastreamento
e Gerenciamento de Registros Pecuários
Utilizando Contratos Inteligentes.**

Dissertação submetida ao Programa de Pós-Graduação em Ciência e Tecnologia da Computação como parte dos requisitos para obtenção do Título de Mestre em Ciência e Tecnologia da Computação.

Área de Concentração: Sistemas de Computação

Orientador: Prof. Dr. Bruno Guazzelli Batista

**29 de agosto de 2024
Itajubá**

UNIVERSIDADE FEDERAL DE ITAJUBÁ - UNIFEI
PROGRAMA DE PÓS-GRADUAÇÃO EM
CIÊNCIA E TECNOLOGIA DA COMPUTAÇÃO

eCattle: Uma Plataforma para Rastreamento
e Gerenciamento de Registros Pecuários
Utilizando Contratos Inteligentes.

Nilton Raimundo de Assis Júnior

Dissertação aprovada na banca de defesa realizada
em 25 de Julho de 2024, para obtenção do título
de **Mestre em Ciência e Tecnologia da Com-
putação.**

Banca Examinadora:

Prof. Dr. Edvard Martins de Oliveira
Prof. Dr. Maycon Leone Maciel Peixoto

Itajubá
2024

Nilton Raimundo de Assis Júnior

eCattle: Uma Plataforma para Rastreamento e Gerenciamento de Registros
Pecuários Utilizando Contratos Inteligentes/ Nilton Raimundo de Assis Júnior. –
Itajubá, 29 de agosto de 2024-

117 p. : il. (algumas color.) ; 30 cm.

Orientador: Prof. Dr. Bruno Guazzelli Batista

Dissertação (Mestrado)

Universidade Federal de Itajubá - UNIFEI

Programa de Pós-graduação em Ciência e Tecnologia da Computação, 29 de agosto
de 2024.

CDU 07:181:009.3

Agradecimentos

Primeiramente, agradeço a Deus, por ter me abençoado, guiado, e me dado forças, durante toda a minha trajetória de vida até aqui.

Aos meus pais e irmãos, que sempre estiveram presentes em minha vida, me orientando para sempre dar os passos certos e principalmente me incentivando e me dando todo o suporte necessário nos estudos. São pessoas fundamentais em minha vida.

A todos os professores que estiveram ligados à minha vida acadêmica, pois cada um deles obteve um papel fundamental na minha formação. Em especial, ao professor Doutor Bruno Guazzelli, que me orientou durante toda a minha trajetória junto ao Programa de Pós-Graduação em Ciência e Tecnologia da Computação da Universidade Federal de Itajubá.

E, por fim, agradeço a todos os amigos e pessoas que sempre estiveram por perto, me apoiando e torcendo por mim.

*"A força não provém da capacidade física. Provém de uma vontade indomável."
(Mahatma Gandhi)*

Resumo

A pecuária é vital para a economia global, mas enfrenta desafios na rastreabilidade, sustentabilidade e eficiência produtiva devido à falta de padronização e dificuldades no compartilhamento de dados. Com a crescente demanda por transparência, a tecnologia Blockchain surge como uma solução, destacando-se por sua descentralização, imutabilidade e transparência. Este estudo explora a aplicação da Blockchain na pecuária, desenvolvendo uma rede Blockchain e um sistema chamado eCattle para monitoramento e gestão de rebanhos. A pesquisa aborda os desafios dos sistemas atuais e apresenta um protótipo baseado em Blockchain para o setor. O estudo enfatiza o potencial da Blockchain para fortalecer a confiança e a sustentabilidade na cadeia de produção pecuária. Além do desenvolvimento do sistema, foram realizadas análises quantitativas que demonstraram a eficácia, a usabilidade, e a aceitação do eCattle entre profissionais do setor agropecuário. Adicionalmente, testes de desempenho revelaram a capacidade do sistema em lidar com múltiplas transações simultâneas. Os resultados indicaram que o sistema é funcional e apresenta potencial para ser implementado em larga escala, contribuindo para a melhoria da rastreabilidade e sustentabilidade na pecuária.

Palavras-chaves: Contratos Inteligentes, Rastreabilidade, Gerenciamento Pecuário.

Abstract

Livestock farming is vital to the global economy, but it faces challenges in traceability, sustainability, and production efficiency due to a lack of standardization and difficulties in data sharing. With the growing demand for transparency, Blockchain technology emerges as a solution, standing out for its decentralization, immutability, and transparency. This study explores the application of Blockchain in livestock farming by developing a Blockchain network and a system called eCattle for herd monitoring and management. The research addresses the challenges of current systems and presents a Blockchain-based prototype for the sector. The study emphasizes the potential of Blockchain to strengthen trust and sustainability in the livestock production chain. In addition to system development, quantitative analyses demonstrated the effectiveness, usability, and acceptance of eCattle among professionals in the agricultural sector. Additionally, performance tests revealed the system's ability to handle multiple simultaneous transactions. The results indicated that the system is functional and shows potential for large-scale implementation, contributing to improved traceability and sustainability in livestock farming.

Key-words: Smart Contracts, Traceability, Livestock Management

Lista de ilustrações

Figura 1 – Ambiente pecuário inteligente. Fonte: [1]	19
Figura 2 – Fluxo da cadeia de alimentação entre o produtor e o consumidor final. Fonte: [2]	20
Figura 3 – Diretrizes para rastreabilidade da cadeia de suprimentos alimentícios. Adaptado de [3].	21
Figura 4 – Cadeia de blocos interligados. Fonte: Autor.	26
Figura 5 – Arquitetura do sistema eCattle. Fonte: Autor.	34
Figura 6 – Modelo C4 para o Contexto. Fonte: Autor.	35
Figura 7 – Modelo C4 para o <i>Container</i> . Fonte: Autor.	36
Figura 8 – Modelo C4 para os Componentes. Fonte: Autor.	37
Figura 9 – Arquitetura Cliente-Servidor. Fonte: Autor.	38
Figura 10 – Diagrama de Fluxo do Algoritmo <i>Raft</i> . Fonte: Autor.	41
Figura 11 – Diagrama de sequência para a integração da API RESTful com a rede Blockchain. Fonte: Autor.	43
Figura 12 – Requisição da API para a Blockchain. Fonte: Autor.	44
Figura 13 – Diagrama de casos de uso UML do sistema proposto. Fonte: Autor.	52
Figura 14 – Diagrama Entidade-Relacionamento. Fonte: Autor.	53
Figura 15 – Diagrama de Classes. Fonte: Autor.	55
Figura 16 – Tela Inicial. Fonte: Autor.	62
Figura 17 – Fazendas Cadastradas. Fonte: Autor.	63
Figura 18 – Gerenciamento de Animais. Fonte: Autor.	64
Figura 19 – Gerenciamento de Funcionários. Fonte: Autor.	65
Figura 20 – Gerenciamento de Transações. Fonte: Autor.	66
Figura 21 – Interação entre <i>peers</i> e <i>orderer</i> . Fonte: Autor.	72
Figura 22 – Validação e confirmação de blocos de transações. Fonte: Autor.	73
Figura 23 – Consultas ao <i>ledger</i> . Fonte: Autor.	74
Figura 24 – Médias e Desvios-Padrão dos Dados Coletados. Fonte: Autor.	82
Figura 25 – Resultados do Teste T. Fonte: Autor.	83
Figura 26 – Latência das Transações no Primeiro Experimento com 15GB de RAM e 6 vCPUs. Fonte: Autor.	89
Figura 27 – Latência das Transações no Segundo Experimento com 4GB de RAM e 1 vCPU. Fonte: Autor.	90
Figura 28 – Uso de CPU durante o Teste de Carga de Ambos Níveis. Fonte: Autor.	91
Figura 29 – Uso de Memória durante o Teste de Carga de Ambos Níveis. Fonte: Autor.	92
Figura 30 – Comparação Entre Métricas e Recursos Computacionais. Fonte: Autor.	94

Figura 31 – Formação profissional. Fonte: Autor.	103
Figura 32 – Experiência no setor agropecuário. Fonte: Autor.	103
Figura 33 – Decisões relacionadas à gestão do rebanho. Fonte: Autor.	104
Figura 34 – Entende claramente o propósito e funcionamento do sistema eCattle. Fonte: Autor.	104
Figura 35 – Funcionalidades e dados apresentados pelo sistema eCattle. Fonte: Autor.	105
Figura 36 – Experiência inicial com o eCattle. Fonte: Autor.	105
Figura 37 – Dificuldades de uso ou navegação. Fonte: Autor.	106
Figura 38 – Entendimentos para a gestão do rebanho. Fonte: Autor.	106
Figura 39 – Oportunidades de melhoria na gestão do rebanho. Fonte: Autor.	107
Figura 40 – O sistema eCattle atende às necessidades do usuário. Fonte: Autor.	107
Figura 41 – As funcionalidades do eCattle são importantes para a gestão do rebanho. Fonte: Autor.	108
Figura 42 – Essencial para o dia-a-dia do usuário na gestão do rebanho. Fonte: Autor.	108
Figura 43 – Potencial de auxiliar significativamente nas tarefas do usuário relacionadas à gestão do rebanho. Fonte: Autor.	109
Figura 44 – A visualização de dados no sistema eCattle é clara e útil. Fonte: Autor.	109
Figura 45 – A visualização de dados não são confusas ou menos úteis no sistema eCattle. Fonte: Autor.	110

Lista de tabelas

Tabela 1 – Trabalhos relacionados	29
Tabela 2 – Requisitos Funcionais	50
Tabela 3 – Requisitos Não Funcionais	51
Tabela 4 – <i>Docker Containers</i> em Execução na Rede Blockchain	75
Tabela 5 – Configuração Geral da VM e Sistema Blockchain	86
Tabela 6 – Fatores e níveis para os experimentos	86
Tabela 7 – Resultados do Apache Benchmark para os Níveis 1 e 2	89

Lista de Siglas

- API** Application Programming Interface
- CPU** Central Processing Unit
- CSS** Cascading Style Sheets
- DER** Diagrama Entidade-Relacionamento
- GPS** Global Positioning System
- HATEOAS** Hypermedia as the Engine of Application State
- HTML** HyperText Markup Language
- HTTP** Hypertext Transfer Protocol
- IoT** Internet of Things
- JSON** JavaScript Object Notation
- MVC** Model-View-Controller
- MSP** Membership Service Provider
- PDF** Portable Document Format
- PHP** Hypertext Preprocessor
- PoS** Proof of Stake
- PoW** Proof of Work
- RAM** Random-Access Memory
- REST** Representational State Transfer
- RFID** Radio Frequency Identification
- RPM** Rotações por Minuto
- SGBD** Sistema de Gerenciamento de Banco de Dados
- UML** Unified Modeling Language
- VM** Virtual Machine
- XML** Extensible Markup Language

Sumário

1	INTRODUÇÃO	16
1.1	Contextualização	16
1.2	Motivação	16
1.3	Objetivos	17
1.4	Estrutura do Documento	18
2	REVISÃO BIBLIOGRÁFICA	19
2.1	Pecuária e rastreabilidade: importância e desafios	19
2.1.1	Desafios da rastreabilidade na pecuária	22
2.1.2	Tecnologias atuais de rastreamento de rebanho	23
2.2	Tecnologia Blockchain	24
2.2.1	Funcionamento da Blockchain	26
2.2.2	Trabalhos relacionados	27
2.3	Linguagens de Programação e <i>Frameworks</i>	30
2.4	API RESTful	31
2.5	Considerações Finais	32
3	ECATTLE	33
3.1	Considerações Iniciais	33
3.2	Arquitetura do sistema	33
3.3	Descrição da plataforma web	38
3.3.1	<i>Backend</i>	38
3.3.2	<i>Frontend</i>	39
3.4	Descrição da Blockchain	39
3.4.1	Estrutura dos blocos e consenso	40
3.4.2	Algoritmo de consenso selecionado	40
3.4.3	Configuração e implantação da rede Blockchain	41
3.4.4	Contratos Inteligentes e <i>chaincode</i>	41
3.5	Integração entre a plataforma web e a Blockchain	42
3.5.1	Arquitetura da API RESTful	43
3.5.2	Comunicação entre Aplicação web e Blockchain	44
3.6	Gerenciamento de Identidades com Carteiras	45
3.7	Banco de dados híbrido	45
3.7.1	Estrutura do banco de dados local e entidades	46
3.7.2	Integração entre o SGBD local e a Blockchain	47
3.8	Considerações Finais	47

4	DESENVOLVIMENTO DA PLATAFORMA WEB	48
4.1	Considerações Iniciais	48
4.2	Arquitetura de Desenvolvimento	48
4.3	Requisitos	49
4.4	Modelagem	49
4.4.1	Casos de Uso	49
4.4.2	Banco de Dados	52
4.4.3	Diagrama de Classes	54
4.5	Implementação Técnica do Sistema eCattle	56
4.5.1	Implementação do Painel Inicial	57
4.5.2	Implementação do Módulo de Gerenciamento de Fazendas	58
4.5.3	Implementação do Módulo de Gerenciamento de Animais	58
4.5.4	Implementação do Módulo de Gerenciamento de Funcionários	60
4.5.5	Implementação do Módulo de Gerenciamento de Transações	60
4.6	Visualização de Dados na Plataforma web	61
4.6.1	Visualização da Tela Inicial	62
4.6.2	Visualização do Gerenciamento de Fazendas	63
4.6.3	Visualização do Gerenciamento de Animais	63
4.6.4	Visualização do Gerenciamento de Funcionários	64
4.6.5	Visualização do Gerenciamento de Transações	65
4.7	Controle de acesso e segurança	66
4.7.1	Autenticação e Gerenciamento de Sessão	66
4.7.2	Autorização e Controle de Acesso	66
4.7.3	Criptografia e Segurança de Dados	67
4.7.4	Medidas Adicionais de Segurança	67
4.8	Considerações Finais	67
5	DESENVOLVIMENTO DA REDE BLOCKCHAIN	69
5.1	Considerações Iniciais	69
5.1.1	Arquitetura Blockchain	69
5.1.2	Sistema web e Rede Blockchain para Gerenciamento de Transações de Gado	69
5.2	Arquitetura da Rede Blockchain eCattle	70
5.2.1	<i>Dockers</i> da Rede Blockchain	74
5.2.2	Integração entre a plataforma web e a rede Blockchain	75
5.3	Considerações Finais	76
6	AValiação DO SISTEMA ECATTLE	78
6.1	Considerações Iniciais	78
6.2	Metodologia de Avaliação do Sistema eCattle	78
6.3	Discussão da Amostra	79

6.4	Avaliação do Sistema eCattle	80
6.4.1	Interpretação dos Resultados	84
6.5	Análise de Comportamento do Sistema eCattle	85
6.5.1	Configuração de Hardware e Virtualização	86
6.5.2	Teste com <i>Script</i> Personalizado	87
6.5.3	Apache Benchmark (ab) para Testes de Carga	88
6.5.4	Análise do Uso de Memória e CPU para Diferentes Configurações de Recursos	89
6.5.5	Interpretação dos Resultados	93
6.6	Implicações para Aprimoramento do Sistema eCattle	94
6.7	Considerações Finais	95
7	CONCLUSÃO	97
7.1	Considerações Finais	97
7.2	Desafios	98
7.3	Publicações e Submissões	99
	 APÊNDICES	 101
	 APÊNDICE A – ROTEIRO DA AVALIAÇÃO PARA O SISTEMA ECATTLE	 102
	 REFERÊNCIAS	 111

1 Introdução

1.1 Contextualização

A pecuária consiste em uma atividade econômica fundamental em muitos países, desempenhando um papel crucial na geração de empregos e no desenvolvimento regional. A produção de carne, leite e derivados, além de couro e lã, contribui significativamente para a economia e a nutrição global [4]. A demanda por produtos de origem animal tem aumentado devido ao crescimento populacional, à urbanização e ao aumento da renda, especialmente nos países em desenvolvimento [4, 5].

A indústria pecuária também é uma fonte importante de subsistência para milhões de pequenos agricultores e produtores rurais, que dependem da criação de animais para sua sobrevivência e renda [6, 7]. Além disso, a pecuária pode contribuir para o desenvolvimento regional e a conservação de ecossistemas, especialmente em áreas com condições climáticas e de solo menos favoráveis à agricultura [4].

No entanto, a pecuária também enfrenta desafios relacionados à sustentabilidade, ao bem-estar animal, à saúde pública e ao meio ambiente, como emissões de gases de efeito estufa, desmatamento, uso de recursos naturais e disseminação de doenças zoonóticas [8, 9]. Esses desafios têm levado a uma crescente demanda por práticas pecuárias mais sustentáveis e sistemas de rastreamento eficientes para garantir a qualidade e a segurança dos produtos de origem animal [4, 10].

1.2 Motivação

A rastreabilidade dos rebanhos é um elemento fundamental para a indústria pecuária, pois permite monitorar a movimentação e a saúde dos animais, garantir a qualidade dos produtos de origem animal e assegurar a segurança alimentar [11, 12]. Além disso, sistemas eficientes de rastreamento de animais são essenciais para prevenir a disseminação de doenças, como a febre aftosa e a Encefalopatia Espongiforme Bovina (EEB), popularmente conhecida como “doença da vaca louca”, que podem ter consequências devastadoras para a pecuária e a saúde humana [13, 14].

No entanto, a rastreabilidade de animais enfrenta diversos desafios, como a falta de padronização nas práticas de identificação e registro, o uso de tecnologias e sistemas de rastreamento ineficientes e a dificuldade de compartilhar e acessar informações entre os diferentes atores da cadeia produtiva [15, 16]. Esses desafios podem resultar em lacunas na rastreabilidade, o que aumenta os riscos para a saúde animal e humana, e podem

dificultar a investigação e a resposta a surtos de doenças ou a *recalls* de produtos [17].

A rastreabilidade também é importante para combater a fraude e o roubo de animais, que são problemas persistentes em muitas regiões [18]. A falta de sistemas de rastreamento eficazes e transparentes pode facilitar práticas ilegais e prejudicar a confiança dos consumidores nos produtos de origem animal [19].

Como resultado desses desafios na rastreabilidade, há uma crescente necessidade de desenvolver e implementar sistemas de rastreamento de rebanhos mais eficientes e seguros, que possam garantir a procedência e qualidade dos produtos de origem animal. A tecnologia Blockchain, com suas características de descentralização, imutabilidade, segurança e transparência, tem o potencial de oferecer uma solução inovadora para superar esses desafios e melhorar a rastreabilidade na indústria pecuária [12].

1.3 Objetivos

Em face ao exposto, o objetivo deste trabalho consiste em investigar o potencial da tecnologia Blockchain para melhorar a rastreabilidade, a segurança e a qualidade na indústria pecuária, desenvolvendo um sistema web chamado eCattle para o cadastro e rastreamento de rebanhos de gado, com a aplicação dessa tecnologia para gerenciar os dados relacionados às atividades de gado, permitindo que as informações sobre um determinado animal sejam rastreadas desde o seu nascimento até o final de sua vida. O objetivo deste trabalho não consiste na aplicação de técnicas de rastreabilidade em campo, como o uso da tecnologia *Global Positioning System* (GPS) ou *tags*, mas sim nas características providas pela tecnologia Blockchain, tais como descentralização, imutabilidade, segurança e transparência. A ideia de se utilizar Blockchain no sistema eCattle tem por objetivo garantir que todos os eventos e transações relacionados aos animais sejam registrados de forma segura e inalterável, proporcionando uma rastreabilidade completa e confiável ao longo de todo o ciclo de vida do gado.

Considerando o objetivo principal deste trabalho, os seguintes objetivos específicos foram definidos:

1. Analisar os desafios e limitações dos sistemas de rastreamento de animais atuais, identificando as áreas em que a tecnologia Blockchain pode contribuir para superar essas limitações [3, 16, 12].
2. Estudar as características e os princípios fundamentais da tecnologia Blockchain, como descentralização, imutabilidade, segurança e transparência, explorando como esses elementos podem ser aplicados no contexto do rastreamento de rebanhos [12, 20].

3. Propor uma arquitetura e um modelo de dados para um sistema web de rastreamento de rebanhos de gado baseado em Blockchain, considerando os requisitos e as necessidades específicas da indústria pecuária [20, 21].
4. Desenvolver e implementar um protótipo do sistema web proposto, utilizando ferramentas e tecnologias apropriadas para a criação de uma solução Blockchain escalável, que atenda às necessidades de rastreabilidade pecuária [20, 22].
5. Avaliar a usabilidade, eficácia e aceitação do sistema eCattle entre profissionais do setor agropecuário, aplicando métodos quantitativos, incluindo questionários baseados em uma escala de Likert para coletar dados dos usuários, seguidos de análises estatísticas detalhadas para quantificar o grau de satisfação dos usuários e identificar áreas de melhorias. Além disso, realizar testes de análises de desempenho.

1.4 Estrutura do Documento

Além desta introdução, esta dissertação está estruturada em mais seis capítulos, que são brevemente descritos a seguir.

- **Capítulo 2 - Revisão Bibliográfica:** Aborda a importância e desafios da rastreabilidade na pecuária, as tecnologias atuais de rastreamento e a tecnologia Blockchain, incluindo seu funcionamento e trabalhos relacionados.
- **Capítulo 3 - eCattle:** Detalha a arquitetura do sistema eCattle, incluindo a descrição da plataforma web, a Blockchain utilizada e o banco de dados híbrido.
- **Capítulo 4 - Desenvolvimento da Plataforma Web:** Descreve a arquitetura de desenvolvimento, os requisitos funcionais e não funcionais, a modelagem, a implementação técnica do sistema e as visualizações de dados.
- **Capítulo 5 - Desenvolvimento da Rede Blockchain:** Apresenta a arquitetura da rede Blockchain, os detalhes sobre a integração com o sistema eCattle e considerações sobre segurança e controle de acesso.
- **Capítulo 6 - Avaliação do Sistema eCattle:** Detalha a metodologia de avaliação utilizada e os resultados obtidos, com análise dos dados coletados, destacando a percepção dos usuários sobre o sistema.
- **Capítulo 7 - Conclusão:** Traz as considerações finais, os desafios enfrentados ao longo do desenvolvimento deste trabalho, publicações, e também a perspectiva de trabalhos futuros.
- **Apêndice:** Inclui o roteiro da avaliação para o sistema eCattle.

2 Revisão Bibliográfica

2.1 Pecuária e rastreabilidade: importância e desafios

A pecuária desempenha um papel fundamental na economia mundial e na segurança alimentar, fornecendo alimentos ricos em proteínas e nutrientes essenciais para a população global [23, 24]. No entanto, a indústria enfrenta desafios importantes relacionados à rastreabilidade, segurança alimentar e sustentabilidade [25, 12]. A rastreabilidade é crucial para garantir a qualidade e a segurança dos produtos de origem animal, bem como para prevenir a disseminação de doenças e combater práticas ilegais [13, 26, 18]. Além disso, promove um ambiente pecuário mais inteligente, permitindo um maior controle do rebanho por parte do produtor rural, conforme ilustrado na Figura 1.



Figura 1 – Ambiente pecuário inteligente. Fonte: [1]

A rastreabilidade é a capacidade de rastrear e identificar animais e produtos de origem animal ao longo de toda a cadeia produtiva, desde a produção até o consumo [3].

A Figura 2 ilustra o fluxo da cadeia de alimentação entre o produtor e o consumidor final, no qual a rastreabilidade pode ser aplicada. Em cada etapa, diferentes tipos de rastreabilidade podem ser implementados: no nível do produtor, a rastreabilidade pode incluir informações sobre a origem dos animais, métodos de criação e dados de saúde; no processador, a rastreabilidade se concentra na transformação dos produtos e controle de qualidade; no distribuidor, o foco é na logística e no transporte seguro dos produtos; no varejo, a rastreabilidade garante a procedência e segurança dos produtos oferecidos aos consumidores [2]. Este trabalho explora especificamente a rastreabilidade provida pela tecnologia Blockchain, que oferece imutabilidade e transparência das informações desde o produtor até o processador, com ênfase na integração de dados ao longo de toda a cadeia produtiva, até o abate do animal.



Figura 2 – Fluxo da cadeia de alimentação entre o produtor e o consumidor final. Fonte: [2]

Dessa forma, a rastreabilidade é essencial na pecuária pelos seguintes motivos:

1. Segurança fitossanitária: permite identificar rapidamente a origem de problemas de segurança alimentar, como contaminação por patógenos ou resíduos químicos, e tomar medidas apropriadas para proteger a saúde pública [25, 3].
2. Controle e prevenção de doenças: ela é fundamental para monitorar e controlar a propagação de doenças animais, como a febre aftosa ou a doença da "vaca louca" [13]. Um sistema eficiente de rastreamento pode facilitar a rápida identificação e eliminação de animais infectados e prevenir a disseminação de doenças [26].
3. Confiança do consumidor: aumenta a confiança do consumidor ao fornecer informações precisas e transparentes sobre a origem, a qualidade e a segurança dos produtos de origem animal [16, 18].
4. Combate à fraude e roubo de animais: ajuda a combater práticas ilegais, como fraude e roubo de animais, ao tornar mais fácil identificar e rastrear animais e produtos de origem animal ao longo da cadeia produtiva [15].

Há uma série de diretrizes motivadoras ou direcionadoras para a capacidade de rastreamento na cadeia de abastecimento alimentar. Essas diretrizes reforçam a rastreabilidade como uma ferramenta para responder às perguntas de “quem (ator/produto), o que (informações do ator/produto), quando (tempo), onde (localização) e por que (causa/razões)” em relação à segurança, qualidade e visibilidade alimentar.

A Figura 3 ilustra as diretrizes, cada um representado por um círculo ao redor do núcleo central. O círculo de “Legislação” aborda os requisitos de segurança e qualidade

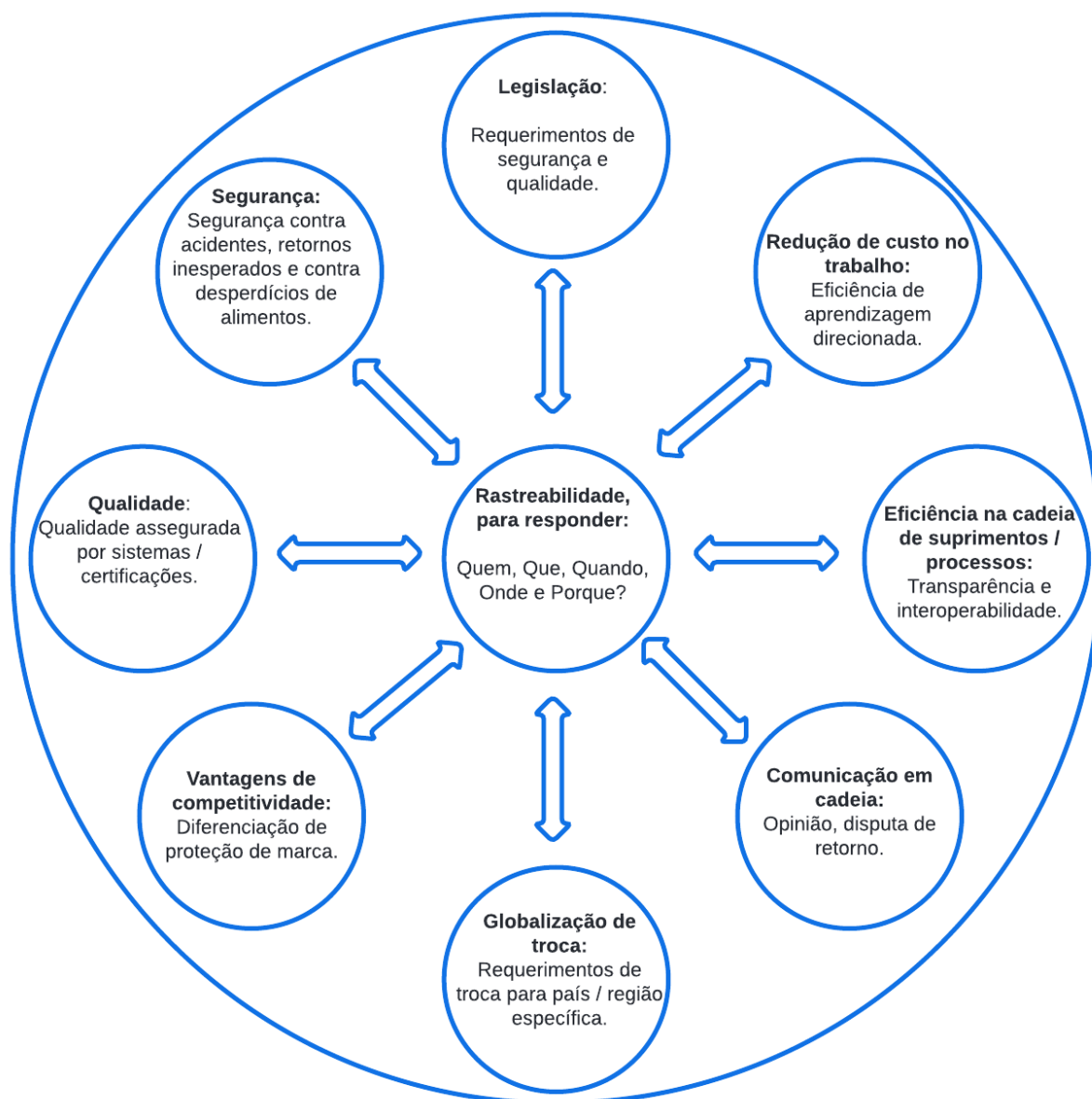


Figura 3 – Diretrizes para rastreabilidade da cadeia de suprimentos alimentícios. Adaptado de [3].

que são impostos por regulamentações legais, destacando como a rastreabilidade ajuda a cumprir essas exigências ao permitir o monitoramento detalhado de cada etapa do processo produtivo. Em relação à “Redução de custo no trabalho”, a rastreabilidade promove a eficiência e o aprendizado direcionado, resultando na redução dos custos operacionais, pois permite identificar rapidamente problemas e áreas de desperdício, melhorando a eficiência geral. O círculo de “Eficiência na cadeia de suprimentos/processos” enfatiza a transparência e a interoperabilidade que a rastreabilidade oferece, permitindo uma gestão mais eficaz da cadeia de suprimentos e processos produtivos, facilitando a coordenação entre diferentes partes envolvidas e melhorando a eficiência. “Comunicação em cadeia” é um aspecto que melhora a comunicação ao longo da cadeia de produção, possibilitando uma troca de informações mais clara e rápida, o que ajuda a resolver disputas e a fornecer

retornos de opinião mais precisos. O círculo “Globalização de troca” destaca a importância da rastreabilidade para atender aos requisitos específicos de troca entre diferentes países ou regiões, garantindo que os produtos estejam em conformidade com as normas e regulamentações locais e internacionais. “Vantagens de competitividade” mostra como a rastreabilidade pode diferenciar uma marca no mercado, protegendo-a e aumentando sua competitividade, pois produtos com rastreabilidade clara são percebidos como mais confiáveis e de maior qualidade pelos consumidores. O círculo de “Qualidade” enfatiza que a rastreabilidade assegura a qualidade dos produtos através de sistemas e certificações, permitindo monitorar e manter padrões de qualidade ao longo de toda a cadeia produtiva. Finalmente, o aspecto de “Segurança” aborda como a rastreabilidade contribui para a segurança dos alimentos, protegendo contra acidentes, retornos inesperados e desperdícios, permitindo uma resposta rápida e eficaz em caso de problemas, minimizando riscos e perdas [3].

2.1.1 Desafios da rastreabilidade na pecuária

É sabido que a pecuária fornece uma fonte significativa de proteínas e nutrientes essenciais para a população. No entanto, a indústria enfrenta vários desafios relacionados à rastreabilidade, o que é crucial para garantir a qualidade e a segurança dos produtos de origem animal, bem como para prevenir a disseminação de doenças e combater práticas ilegais [23, 25].

Desafios na rastreabilidade da pecuária incluem:

1. Padronização e interoperabilidade:

- Falta de padronização nos sistemas de rastreamento e identificação animal.
- Dificuldades na integração e compartilhamento de informações entre diferentes atores da cadeia produtiva [3].

2. Centralização e vulnerabilidade:

- Sistemas de rastreamento centralizados, suscetíveis a falhas e ataques.
- Possibilidade de adulteração e manipulação de dados, comprometendo a confiabilidade e eficácia dos sistemas [20].

3. Acesso e compartilhamento de dados:

- Dificuldades no acesso e troca de informações entre partes interessadas.
- Falta de interoperabilidade e lacunas na rastreabilidade, aumentando os riscos à saúde animal e segurança alimentar [3].

4. Custos e complexidade de implementação:

- Custos elevados e complexidade na implementação de sistemas de rastreamento eficazes.
- Barreiras significativas para a adoção, especialmente por pequenos produtores e atores em países em desenvolvimento [15, 16].

5. Adoção de tecnologia e resistência à mudança:

- Relutância em adotar novas tecnologias e métodos de rastreamento por parte de alguns atores da indústria.
- Desafios na educação e conscientização sobre a importância da rastreabilidade e as tecnologias disponíveis [27].

6. Legislação e regulamentação:

- Diferenças nas leis e regulamentações de rastreabilidade entre países e regiões.
- Dificuldades na aplicação e fiscalização das normas existentes, resultando em práticas inadequadas e ineficientes [28].

Diante desses desafios, é importante buscar soluções eficientes para melhorar a rastreabilidade na indústria pecuária.

2.1.2 Tecnologias atuais de rastreamento de rebanho

As tecnologias de rastreamento de rebanho desempenham um papel fundamental na gestão e monitoramento da pecuária, permitindo aos produtores acompanhar a saúde, a localização e o desempenho dos animais. Algumas das principais tecnologias atuais de rastreamento de rebanho incluem:

1. Identificação de animais:

- Identificação visual: *tags* de orelha, tatuagens e marcas a quente [29].
- Identificação eletrônica: *microchips*, *transponders* e *tags* de orelha eletrônicas que utilizam a tecnologia *Radio Frequency Identification* (RFID) [30].

2. Sistemas de localização e monitoramento:

- *Global Positioning System* (GPS): rastreamento de animais e veículos em um curto intervalo de tempo, possibilitando o monitoramento da localização dos rebanhos e a identificação de padrões de movimento [31].
- *Geofencing*: criação de áreas geográficas virtuais, permitindo o monitoramento e a gestão dos animais dentro de limites predefinidos [32].

3. Monitoramento da saúde e do desempenho:

- Sensores biométricos: coleta de dados sobre a saúde, o bem-estar e o desempenho dos animais, como frequência cardíaca, temperatura corporal e atividade [33].
- Sistemas automatizados de ordenha: coleta de dados sobre a produção de leite e a saúde das vacas durante o processo de ordenha [33].
- Balanças eletrônicas: monitoramento do peso e crescimento dos animais, permitindo a detecção precoce de problemas de saúde e ajustes na alimentação e manejo [34].

4. Sistemas de informação e bancos de dados:

- Bancos de dados centralizados e sistemas de informação agrícola: armazenamento e gerenciamento de informações sobre animais, como identificação, genealogia, registros de saúde e de produção [35].
- Sistemas de rastreabilidade baseados na nuvem: permitindo o acesso e compartilhamento de informações entre os atores da cadeia produtiva [36].

Embora essas tecnologias tenham melhorado significativamente a capacidade de rastrear e gerenciar rebanhos de gado, elas ainda enfrentam limitações em termos de padronização, interoperabilidade, segurança e custo. A tecnologia Blockchain é uma abordagem emergente que tem o potencial de abordar essas limitações e oferecer soluções mais eficazes e confiáveis para a rastreabilidade na indústria pecuária [12].

2.2 Tecnologia Blockchain

A Blockchain é uma tecnologia de registro distribuído que permite armazenar e compartilhar informações de forma descentralizada, segura e transparente [37]. É composta por uma série de blocos interligados, cada um contendo um conjunto de transações ou registros de dados. Uma vez que um bloco é adicionado à cadeia, suas informações se tornam imutáveis, garantindo a confiabilidade e a integridade dos dados armazenados [38].

A Blockchain tem um enorme potencial para transformar diferentes setores, incluindo cadeias de suprimentos, finanças, governança, propriedade intelectual e muito mais [39]. No contexto da pecuária, a tecnologia pode ser usada para melhorar a rastreabilidade do gado, garantir a qualidade e a segurança dos produtos e aumentar a confiança dos consumidores na origem e na autenticidade dos produtos de origem animal.

A seguir estão listados os principais conceitos teóricos e técnicos relacionados à tecnologia Blockchain:

1. Registro distribuído: A Blockchain é uma forma de registro distribuído (ou *ledger*) que armazena informações de forma descentralizada em uma rede de computadores (ou nós). Isso permite que a rede mantenha o registro e o consenso dos dados, sem a necessidade de uma autoridade centralizada [39].
2. Criptografia: A Blockchain utiliza a criptografia para proteger e garantir a integridade dos dados. Cada bloco na cadeia contém uma lista de transações e um identificador único chamado “*hash*”. Esses *hashes* são gerados usando funções criptográficas de *hash*, como SHA-256 [37]. Essas funções transformam os dados do bloco em uma sequência alfanumérica fixa, de modo que qualquer alteração nos dados resulte em um *hash* diferente [38]. A criptografia e o consenso distribuído tornam a Blockchain altamente resistente a ataques e fraudes [40].
3. Imutabilidade e consenso: Os blocos na Blockchain são vinculados uns aos outros através de ponteiros criptográficos, que contêm o *hash* do bloco anterior. Isso cria uma cadeia de blocos que se torna praticamente imutável, uma vez que qualquer tentativa de alterar um bloco exigiria a recriação de todos os blocos subsequentes [37]. Além disso, a rede utiliza mecanismos de consenso, como *Proof of Work* (PoW) ou *Proof of Stake* (PoS), para garantir que todos os nós concordem com o conteúdo e a ordem dos blocos, permitindo a integridade dos dados [40].
4. Transparência e rastreabilidade: A Blockchain é geralmente pública e transparente, permitindo que todos os participantes da rede vejam e verifiquem as transações e os dados armazenados na cadeia [41]. Isso facilita a rastreabilidade e a auditoria de informações, como a origem e o histórico de um produto ou a autenticidade de uma transação.
5. Contratos inteligentes (*Smart Contracts*): A Blockchain também pode suportar contratos inteligentes, que são programas autônomos que executam ações predefinidas quando certas condições são atendidas [42]. Esses contratos podem ser usados para automatizar processos e facilitar a cooperação entre diferentes partes sem a necessidade de intermediários.

Essas referências teóricas cobrem aspectos importantes dos conceitos de Blockchain, incluindo registro distribuído, criptografia, imutabilidade, consenso, transparência e contratos inteligentes. Elas ajudam a entender como a Blockchain funciona, suas principais vantagens e aplicações potenciais em diversos setores, incluindo rastreabilidade na pecuária e cadeia de suprimentos.

2.2.1 Funcionamento da Blockchain

O funcionamento da Blockchain é baseado em uma série de princípios que garantem a segurança, a descentralização e a imutabilidade dos dados armazenados [39]. A rede Blockchain começa com um bloco inicial, chamado de bloco gênese, que serve como ponto de partida para a cadeia [37]. Os nós na rede mantêm cópias sincronizadas dessa cadeia e participam do processo de adição de novos blocos. Quando os usuários da rede realizam transações, estas são transmitidas para os nós da rede [37], podendo ser transações financeiras, como transferências de criptomoeda, ou envolver dados, como registros de propriedade ou rastreabilidade de produtos.

Os nós coletam transações não confirmadas e as agrupam em um bloco candidato [37], contendo um conjunto de transações, o *hash* do bloco anterior e um valor chamado “*nonce*” [38]. Para adicionar um novo bloco à cadeia, os nós competem para resolver um problema matemático complexo, conhecido como PoW [37] ou PoS em outras implementações [40]. O primeiro nó que resolve o problema tem o direito de propor o novo bloco para ser adicionado à cadeia. Outros nós na rede verificam a solução e as transações no bloco proposto [37]. Se o bloco for válido e a solução aceita, os nós concordam em adicionar o novo bloco à cadeia, processo conhecido como consenso [40]. Após o consenso, a cadeia é atualizada em todos os nós da rede para incluir o novo bloco [37], com as transações no bloco sendo consideradas confirmadas e a PoW redefinida para o próximo bloco.

Este processo se repete continuamente, criando uma cadeia crescente de blocos interligados e protegidos por criptografia [39]. A combinação de registro distribuído, criptografia, consenso e incentivos garante a segurança, a descentralização e a imutabilidade dos dados armazenados na Blockchain [41]. A Figura 4 representa um exemplo de cadeia de blocos interligados. Seguindo esse processo, a Blockchain constrói uma cadeia de registros (blocos) praticamente imutáveis e transparentes, permitindo que todos os participantes da rede acessem e verifiquem as informações armazenadas [41].

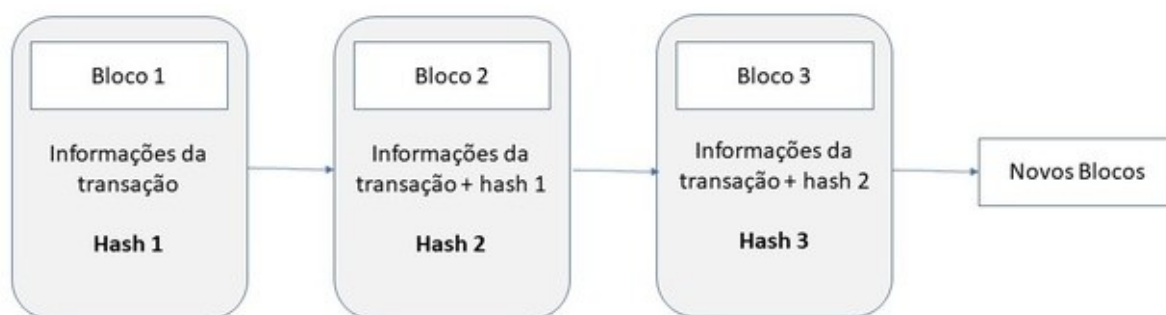


Figura 4 – Cadeia de blocos interligados. Fonte: Autor.

2.2.2 Trabalhos relacionados

Apesar de seu papel crucial na nutrição global, a indústria de pecuária enfrenta diversos desafios em rastreabilidade no acompanhamento de animais desde o nascimento até o abate [11]. Esses desafios incluem a falta de padronização e interoperabilidade nos sistemas de rastreamento, vulnerabilidades dos sistemas centralizados, dificuldades no acesso e compartilhamento de dados, custos elevados e complexidade na implementação de sistemas eficazes, resistência à adoção de novas tecnologias e variações nas leis e regulamentos de rastreabilidade [18]. O presente trabalho se concentra na aplicação da tecnologia Blockchain para superar esses desafios, fornecendo rastreamento através da imutabilidade dos dados e a transparência garantida pela descentralização, permitindo auditorias confiáveis e seguras ao longo de toda a cadeia de produção pecuária.

As tecnologias atuais de rastreamento de rebanho, que são essenciais para a gestão da pecuária, incluem identificação visual e eletrônica, GPS e *Geofencing* para rastreamento de localização, além de monitoramento de saúde e desempenho através de sensores biométricos, sistemas de ordenha automatizados e balanças eletrônicas. Apesar desses avanços, os desafios como padronização, interoperabilidade, segurança e custos ainda persistem [12, 20, 3].

A falta de padronização e interoperabilidade nos sistemas de rastreamento de gado dificulta significativamente a eficiência na detecção de doenças e a garantia de segurança alimentar, aumentando os riscos para a saúde animal e humana [12]. Essa lacuna também facilita práticas ilegais, como o comércio de animais doentes ou não regulamentados, e, economicamente, impede a otimização da cadeia de suprimentos de gado [43]. Além disso, esses desafios de rastreabilidade minam a confiança dos consumidores, especialmente em uma era em que sustentabilidade e bem-estar animal são preocupações crescentes.

A tecnologia Blockchain é vista como uma solução promissora para superar muitas das limitações dos sistemas de rastreamento atuais, oferecendo uma alternativa eficiente e confiável para a rastreabilidade na indústria de pecuária [44].

No setor agrícola, especialmente no gerenciamento de pecuária, as aplicações de Blockchain incluem a rastreabilidade de cadeias de suprimentos de gado e laticínios, certificações de bem-estar animal, contratos inteligentes para transações financeiras e gestão de registros veterinários, demonstrando sua capacidade de melhorar a transparência e a confiança na origem de produtos animais [45, 46].

Projetos como BeefLedger e Ambrosus focam na rastreabilidade das cadeias de suprimentos de carne bovina e laticínios, respectivamente, enquanto Vion Food Group e AgriChain concentram-se na certificação de bem-estar animal e facilitação de transações financeiras através de contratos inteligentes. O Vetsource usa Blockchain para gestão de registros veterinários [47, 48, 49, 50, 51].

Além disso, existem diversos artigos disponíveis na literatura que propõem soluções para a área agrícola utilizando Blockchain. No entanto, nenhum dos trabalhos analisados aplica contratos inteligentes à gestão da cadeia de produção da pecuária. A Tabela 1 ilustra a versatilidade da tecnologia Blockchain em várias aplicações agrícolas, posicionando este projeto como uma abordagem para lidar com desafios específicos na gestão da cadeia de produção de gado.

A principal diferença entre o sistema proposto neste trabalho, chamado eCattle, e os trabalhos analisados está na aplicação do Blockchain para gestão e rastreamento de rebanhos de gado.

Como exemplo, o sistema eCattle utiliza a tecnologia Blockchain para gerenciar transações de gado e rastrear informações de rebanhos, enquanto que Patil et al. (2023) [52] enfocam serviços de aluguel e empréstimo na agricultura. A principal diferença é que o eCattle se concentra em garantir a segurança e a transparência no rastreamento de gado, aplicando Blockchain no contexto pecuário.

Mao et al. (2019) [53] abordam o comércio automático de alimentos, enquanto o sistema eCattle lida especificamente com gerenciamento e rastreamento de rebanhos de gado usando Blockchain, indicando a versatilidade da tecnologia Blockchain em várias partes da cadeia de suprimentos de alimentos.

Ramprasath et al. (2023) [54] enfatizam a segurança nas transações de dados de coiteira, enquanto o sistema eCattle usa Blockchain para proteger informações relacionadas ao gado e rastrear transações, garantindo segurança e confiabilidade na gestão pecuária.

Vanditha et al. (2023) [55] exploram a gestão da cadeia de suprimentos agrícolas, enquanto o sistema eCattle aplica Blockchain para gerenciar transações e rastreabilidade de rebanhos, destacando o potencial da Blockchain para melhorar a eficiência e a segurança na cadeia de suprimentos de gado.

Alshehri (2023) [56] combina Blockchain com IoT em pecuária inteligente, relacionado ao uso da Blockchain no sistema eCattle para gerenciamento de rebanhos de gado. Ambos mostram a inovação tecnológica na agricultura, mas o eCattle diferencia-se por focar em rastreabilidade e segurança na gestão pecuária.

Jegadeesan et al. (2023) [57] propõem um esquema seguro para agricultura inteligente, enquanto o sistema eCattle se concentra na gestão de transações de gado usando Blockchain para garantir segurança e eficiência operacional, evidenciando a flexibilidade da tecnologia Blockchain.

Rambhia et al. (2021) [58] apresentam uma solução baseada em Blockchain para agricultura, alinhada ao sistema eCattle em busca de melhorias na gestão da cadeia de suprimentos. No entanto, o eCattle distingue-se ao aplicar Blockchain para rastreabilidade e gerenciamento de rebanhos de gado.

Tabela 1 – Trabalhos relacionados

Artigo	Objetivo
Patil et al. (2023) [52]	Desenvolve um sistema de aluguel e empréstimo de equipamentos agrícolas utilizando a tecnologia Blockchain, destacando como essa tecnologia pode ser aplicada para facilitar serviços na agricultura.
Mao et al. (2019) [53]	Propõe um sistema de comércio automático de alimentos usando Blockchain em um consórcio, mostrando como a tecnologia pode ser usada para transações automatizadas na cadeia alimentar.
Ramprasath et al. (2023) [54]	Investiga a segurança das transações de dados na colheita usando Blockchain, indicando métodos para proteger informações durante o processo de colheita agrícola.
Vanditha et al. (2023) [55]	Explora um sistema de gestão da cadeia de suprimentos agrícolas usando Blockchain, destacando a importância de rastreabilidade e coordenação entre diferentes etapas da cadeia.
Alshehri (2023) [56]	Apresenta um <i>framework</i> de Blockchain assistido por Internet das Coisas (<i>Internet of Things</i> - IoT) para pecuária inteligente, demonstrando a aplicação do Blockchain na pecuária com integração de tecnologias emergentes.
Jegadeesan et al. (2023) [57]	Propõe um esquema seguro para agricultura inteligente usando Blockchain, enfatizando a importância da segurança e da confiabilidade dos dados no contexto da agricultura inteligente.
Rambhia et al. (2021) [58]	Introduz uma solução de Blockchain para gestão da cadeia de suprimentos na agricultura, destacando como a tecnologia pode ser usada para rastrear e gerenciar produtos agrícolas.
Lin et al. (2018) [59]	Desenvolve um sistema de rastreabilidade na cadeia de suprimentos de alimentos usando Blockchain, mostrando como a tecnologia pode aumentar a transparência e a segurança no setor alimentício.
Balekundri et al. (2023) [60]	Aborda a gestão da cadeia de suprimentos agrícolas com Blockchain, enfatizando o potencial para melhorar a coordenação entre partes interessadas e facilitar o acesso a financiamentos.
Adeniyi et al. (2023) [61]	Explora o uso de contratos inteligentes em negócios agrícolas usando Blockchain, indicando como essa tecnologia pode ser usada para automatizar contratos e transações no setor agrícola.

Lin et al. (2018) [59] abordam a rastreabilidade na cadeia de suprimentos de alimentos, enquanto o sistema eCattle busca rastreabilidade e segurança para rebanhos de

gado utilizando Blockchain, enfatizando a contribuição da tecnologia para a transparência na cadeia alimentícia.

Balekundri et al. (2023) [60] destacam os benefícios da Blockchain na gestão da cadeia de suprimentos, enquanto o sistema eCattle se concentra em transações de gado, demonstrando como a Blockchain pode melhorar a eficiência e a segurança na gestão pecuária.

Adeniyi et al. (2023) [61] exploram contratos inteligentes nos negócios agrícolas, enquanto o eCattle integra uma plataforma web com Blockchain para automatizar contratos e transações de gado, demonstrando a adaptabilidade da Blockchain para diversos contextos agrícolas.

2.3 Linguagens de Programação e *Frameworks*

O *HyperText Markup Language* (HTML), *Cascading Style Sheets* (CSS) e *JavaScript* são as linguagens fundamentais para o desenvolvimento de interfaces de usuário web. HTML é a linguagem padrão para criar estruturas de páginas web [62]. CSS é usada para estilizar e formatar a aparência das páginas web [63]. *JavaScript* é uma linguagem de programação que permite a criação de interações dinâmicas e conteúdo web [64].

Já o *Hypertext Preprocessor* (PHP) é uma linguagem de programação de código aberto amplamente utilizada para desenvolvimento web [65]. PHP é especialmente adequado para a criação de aplicações web dinâmicas e é compatível com uma ampla gama de sistemas de gerenciamento de banco de dados e servidores web. A linguagem possui muitos recursos e bibliotecas disponíveis que facilitam o desenvolvimento rápido e eficiente de sistemas web.

O *Bootstrap* é um *framework* de código aberto para desenvolvimento de aplicações web responsivas e *mobile-first* [66]. Ele é baseado em HTML, CSS e *JavaScript* e oferece uma ampla gama de componentes pré-construídos, como botões, formulários, navegação e tipografia, que facilitam a criação de interfaces de usuário atraentes e funcionais. *Bootstrap* também possui uma grande comunidade de desenvolvedores e uma vasta quantidade de recursos e exemplos disponíveis.

Já o *CodeIgniter* é um *framework* de desenvolvimento de aplicações web em PHP, baseado no padrão de arquitetura *Model-View-Controller* (MVC) [67]. *CodeIgniter* oferece uma série de bibliotecas e ferramentas que simplificam o desenvolvimento de aplicações web, incluindo gerenciamento de banco de dados, validação de formulários, sessões e suporte a autenticação. Além disso, o *framework* tem uma grande comunidade e uma ampla gama de recursos disponíveis para auxiliar no desenvolvimento.

O padrão MVC é um *design* amplamente utilizado para desenvolver aplicativos da

web, pois promove a separação de preocupações e facilita a manutenção e a escalabilidade do código [68]. O padrão MVC é definido por:

- *Model*: Representa a lógica de negócios e a camada de acesso aos dados. Os *Models* interagem com a Blockchain e gerenciam a leitura e a escrita de dados, além de aplicar regras de negócios.
- *View*: Responsável pela apresentação dos dados e pela interface do usuário. As *Vistas* exibem informações recuperadas dos *Models* e permitem a interação do usuário com o sistema.
- *Controller*: Atua como intermediário entre o *Model* e a *View*, gerenciando a comunicação e coordenando as ações entre eles. Os *Controllers* também lidam com as solicitações do usuário e gerenciam a autenticação e a autorização.

2.4 API RESTful

As *Applications Programming Interfaces* (APIs), que seguem os princípios da arquitetura *Representational State Transfer* (REST), são um modelo amplamente adotado para o desenvolvimento de APIs RESTful na web. REST se baseia em um conjunto de princípios projetados para utilizar de maneira eficaz as características do protocolo *Hypertext Transfer Protocol* (HTTP) [69]. Dentre os princípios fundamentais das APIs RESTful destacam-se:

1. Métodos HTTP: As APIs RESTful empregam métodos HTTP padrão de forma sistemática — *GET* para recuperação de dados, *POST* para criação de recursos, *PUT* para atualização de recursos e *DELETE* para a remoção de recursos [70].
2. *Statelessness*: A comunicação entre cliente e servidor é *stateless*, ou seja, cada requisição HTTP deve ser autocontida, carregando todas as informações necessárias para o servidor entender e responder à requisição [69].
3. *Cacheable*: As respostas das APIs devem ser definidas como *Cacheables* ou não, o que facilita a reutilização de respostas anteriormente armazenadas, aumentando a eficiência da rede [71].
4. Sistema de Camadas: A arquitetura pode ser estruturada em múltiplas camadas, permitindo o uso de balanceadores de carga e caches para melhorar a escalabilidade e desempenho do sistema [69].
5. Interface Uniforme: A uniformidade da interface entre os componentes do sistema simplifica a arquitetura, facilitando a interação e evolução independentes das suas partes [69].

6. Uso de Hiper Mídias: Segundo o princípio de *Hypermedia as the Engine of Application State* (HATEOAS), as respostas de uma API RESTful devem incluir *hiperlinks* para outros recursos disponíveis, permitindo uma navegação dinâmica entre eles [70].

As APIs RESTful são conhecidas por sua simplicidade e flexibilidade, podendo manipular diversos formatos de dados como *JavaScript Object Notation* (JSON), *Extensible Markup Language* (XML), entre outros. Esta característica as torna adequadas para ambientes de grande escala, sendo independentes de plataforma e linguagem, o que permite sua implementação e consumo em qualquer sistema que suporte HTTP [71]. Além disso, as APIs RESTful são particularmente úteis em aplicações que requerem integração web, como serviços de *backend* para aplicações móveis ou web, sistemas empresariais, e plataformas para a Internet das Coisas (*Internet of Things* - IoT) [72].

2.5 Considerações Finais

Neste capítulo, foram apresentados os principais desafios na rastreabilidade na indústria de pecuária e como a tecnologia Blockchain pode ser utilizada para superar essas dificuldades. Foram discutidos os conceitos fundamentais da Blockchain, incluindo sua descentralização, imutabilidade e segurança, além de detalhar o funcionamento dessa tecnologia, desde a inicialização da rede até a atualização da cadeia com novos blocos.

Também foi enfatizado como a Blockchain pode fornecer uma nova solução para o rastreamento de animais, garantindo a integridade e a transparência dos dados ao longo de toda a cadeia de produção pecuária. Também foram discutidas as vantagens da utilização de contratos inteligentes para automatizar processos e melhorar a cooperação entre diferentes partes interessadas.

Além disso, foi apresentado um breve referencial teórico sobre as linguagens de programação e *frameworks* utilizados no desenvolvimento do Sistema eCattle.

No Capítulo 3, será apresentada a metodologia de desenvolvimento do eCattle, detalhando as etapas e abordagens utilizadas para a criação e implementação do sistema proposto. O desenvolvimento do sistema proposto segue a metodologia descrita na próxima seção.

3 eCattle

3.1 Considerações Iniciais

Neste capítulo, é apresentado o sistema eCattle, um sistema proposto para aprimorar a rastreabilidade, a segurança e a gestão de rebanhos na indústria pecuária utilizando a tecnologia Blockchain. O propósito do eCattle é fornecer uma plataforma transparente para o cadastro, monitoramento e rastreamento de rebanhos de gado, garantindo a integridade e a imutabilidade dos dados através das características inerentes da Blockchain.

O eCattle foi projetado para atender às necessidades específicas dos produtores rurais, veterinários, técnicos agrícolas e demais profissionais do setor agropecuário, facilitando a gestão de informações e melhorando a eficiência e a transparência dos processos envolvidos na cadeia produtiva.

A seguir, é detalhado a implementação do sistema eCattle, incluindo as linguagens de programação e os *frameworks* utilizados. O sistema foi desenvolvido utilizando PHP para a criação do *backend*, *JavaScript*, HTML e CSS para o *frontend*. O *framework CodeIgniter 4* foi escolhido para o desenvolvimento do *backend*, devido à sua simplicidade e eficiência na criação de aplicações web. Para a camada de persistência, foi utilizado o banco de dados *MariaDB*, e a tecnologia Blockchain foi implementada utilizando a plataforma *Hyperledger Fabric*, que oferece escalabilidade, segurança e suporte a contratos inteligentes.

Com essa abordagem, o eCattle integra boas práticas em desenvolvimento de sistemas web e tecnologias Blockchain, proporcionando uma solução eficiente para a rastreabilidade de rebanhos na indústria pecuária.

3.2 Arquitetura do sistema

A arquitetura do sistema refere-se à sua estrutura geral, incluindo os componentes e as interações entre eles. A arquitetura é responsável principalmente pela organização e a comunicação entre as partes do sistema, sendo fundamental para garantir a escalabilidade, a manutenibilidade e a eficiência do sistema. A Figura 5 apresenta a arquitetura como um todo, dividida em camadas, e ilustrando as tecnologias adotadas no desenvolvimento do sistema eCattle. Nessa Figura, as setas em verde ilustram o fluxo de autenticação. Após a autenticação, os proprietários e funcionários se comunicam diretamente com seus controladores, sem passar novamente pela autenticação.

Para o desenvolvimento do projeto arquitetural, foi escolhido o modelo MVC [68].

A fim de ter uma abordagem visual para modelar a arquitetura de sistemas de software, diagramas na notação C4 foram modelados [73].

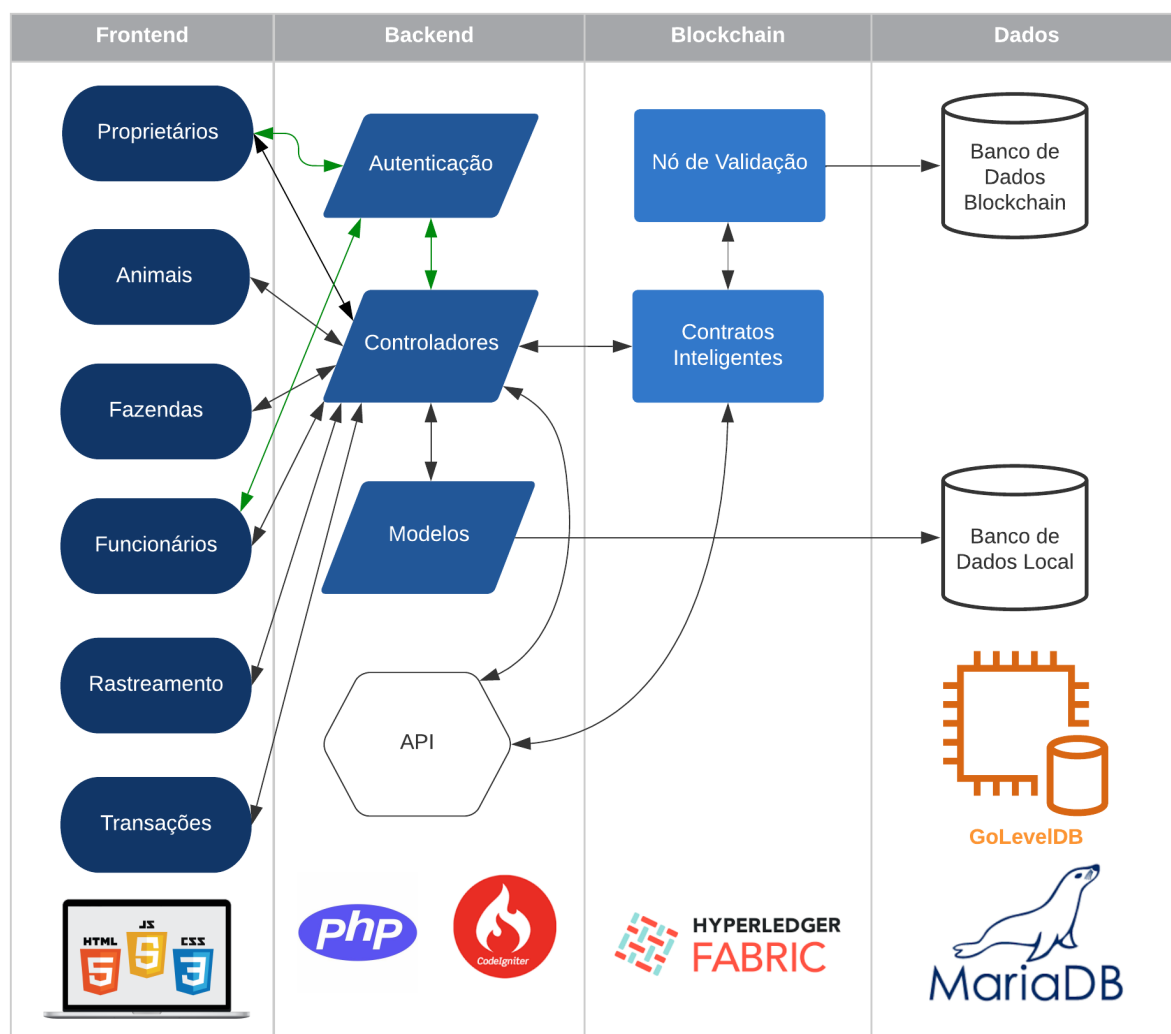


Figura 5 – Arquitetura do sistema eCattle. Fonte: Autor.

A notação C4 é uma abordagem visual para modelar a arquitetura de sistemas de software, desenvolvida por Simon Brown [73]. Ela fornece uma maneira eficaz e padronizada de representar a arquitetura de sistemas em diferentes níveis de abstração, facilitando a comunicação entre os membros da equipe e a compreensão geral do projeto. A notação C4 se baseia em quatro níveis de detalhamento: Contexto, *Container*, Componente e Código.

Neste projeto, a notação C4 é utilizada para descrever a arquitetura do sistema em diversos níveis, mostrando como os elementos do sistema interagem e se relacionam com os atores externos. Os diagramas C4 propostos para o projeto estão alinhados com o Diagrama Entidade-Relacionamento (DER) e o Diagrama de Classes, apresentados no Capítulo 4, respectivamente nas Seções 4.4.2 e 4.4.3, garantindo que a arquitetura seja coerente e consistente.

O Diagrama de Contexto apresenta uma visão geral do sistema eCattle, ilustrando como ele se relaciona com os atores externos (Clientes), como o Proprietário e o Funcionário, usuários do sistema. Este diagrama destaca o propósito e as responsabilidades gerais do sistema, que incluem o gerenciamento de rebanho, a extração de relatórios e o acompanhamento dos animais. Na Figura 6, pode-se observar uma representação visual desse diagrama, que é essencial para entender o escopo e as interações externas do sistema.

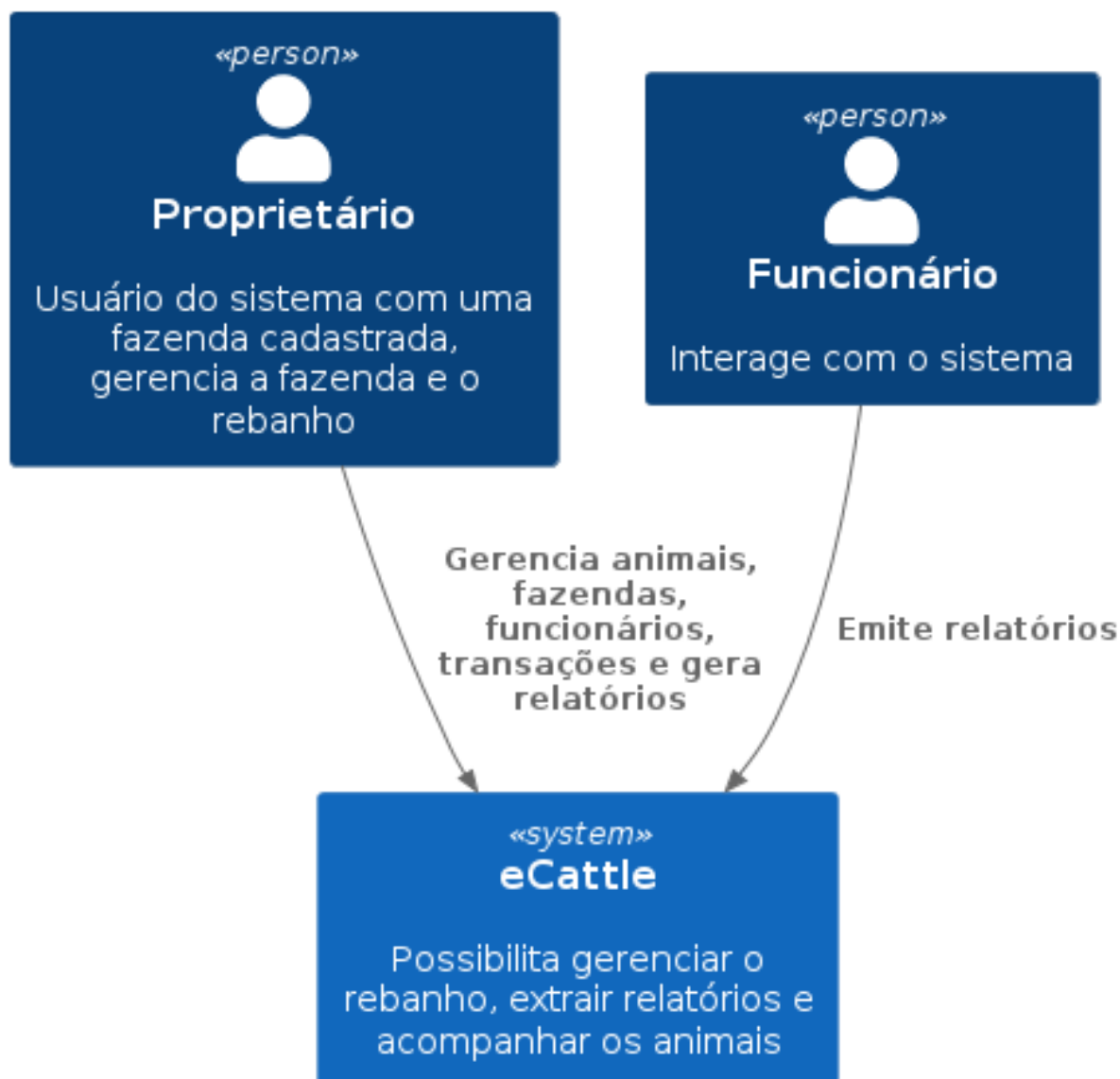


Figura 6 – Modelo C4 para o Contexto. Fonte: Autor.

O Diagrama de *Container* é um passo adiante na decomposição do sistema eCattle, dividindo-o em diferentes *containers*, como *View*, *Controller*, *Model*, e os bancos de dados *MySQL* e *Blockchain*. Este diagrama detalha as tecnologias utilizadas em cada *container* e suas interações, garantindo modularidade e organização. Ele é crucial para entender como as diferentes partes do sistema colaboram para fornecer funcionalidades integradas e como a arquitetura do sistema é estruturada para suportar escalabilidade e manutenção. A Figura 7 ilustra essa divisão em *containers* e as interações entre eles.

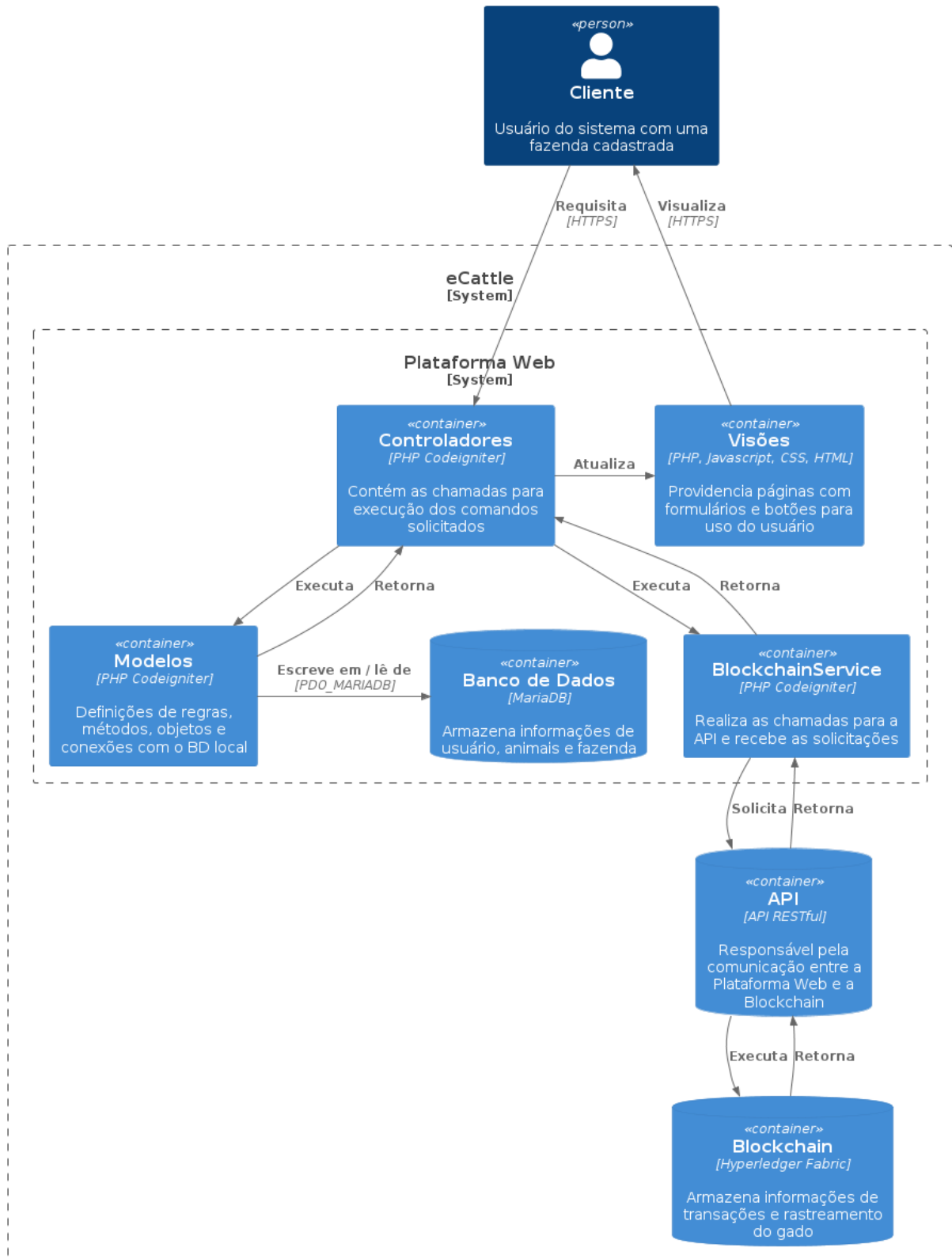


Figura 7 – Modelo C4 para o *Container*. Fonte: Autor.

Por fim, o Diagrama de Componente aprofunda ainda mais a estrutura do sistema ao detalhar os principais componentes de cada *container* e suas responsabilidades. Este diagrama mostra como os componentes interagem entre si e com os atores externos. Ele

é fundamental para compreender a lógica interna do sistema e como cada componente contribui para a realização das operações desejadas. A Figura 8 apresenta uma visão detalhada dos componentes, facilitando a análise de dependências e a identificação de possíveis pontos de melhoria ou expansão.

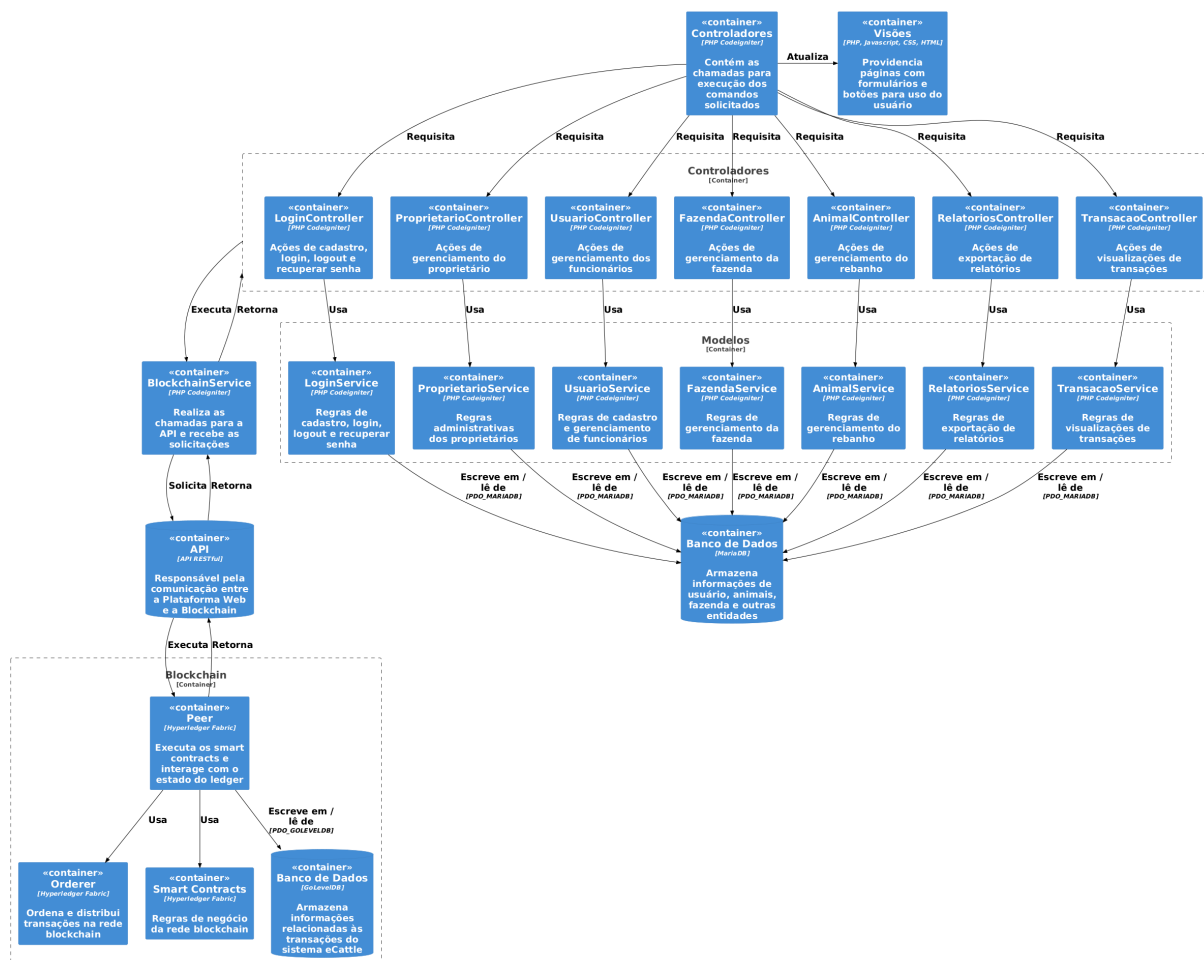


Figura 8 – Modelo C4 para os Componentes. Fonte: Autor.

Esses diagramas, juntos, fornecem uma compreensão completa e detalhada da arquitetura do sistema eCattle, desde uma visão macro das suas interações externas até os detalhes específicos de cada componente interno. Eles são ferramentas essenciais para garantir que o sistema seja bem projetado, fácil de manter e escalável, atendendo às necessidades dos seus usuários e adaptando-se às evoluções tecnológicas.

Para desenvolver o sistema eCattle, foi seguido uma abordagem estruturada e modular para garantir a escalabilidade e a manutenibilidade do sistema. A arquitetura Cliente-Servidor foi escolhida para modelar a comunicação entre o cliente e a aplicação web. Também foi considerado outros dispositivos de rede, tais como sensores.

Nesse modelo, o cliente não compartilha nenhum de seus recursos com o servidor, ele solicita alguma função, sendo ele, o cliente, responsável por iniciar a comunicação com o servidor, enquanto o mesmo aguarda requisições de entrada. Na Figura 9 é apresentado

um modelo da arquitetura Cliente-Servidor.

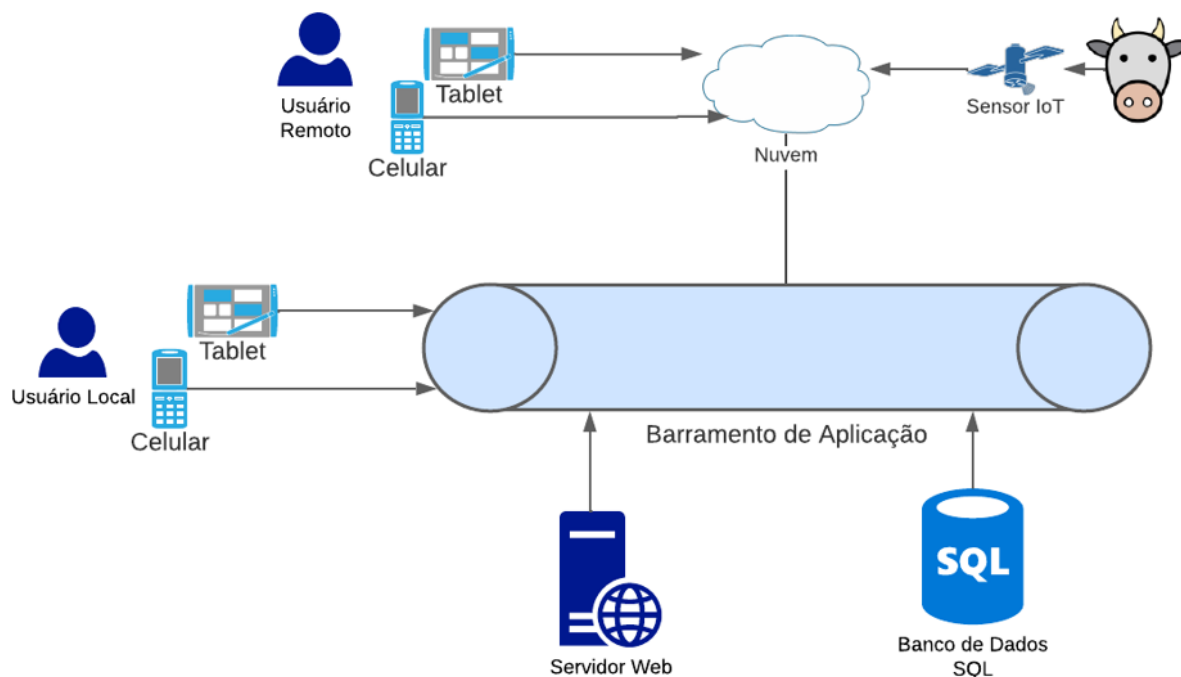


Figura 9 – Arquitetura Cliente-Servidor. Fonte: Autor.

É importante salientar que a arquitetura Cliente-Servidor é aplicada para a comunicação entre os clientes e a aplicação web. A Blockchain é uma tecnologia distribuída, e não segue o modelo tradicional Cliente-Servidor. No modelo cliente-servidor, um servidor centralizado fornece serviços a múltiplos clientes. Em contraste, a Blockchain é baseada em uma rede de nós (ou *peers*) distribuídos, onde cada nó mantém uma cópia completa do registro (*ledger*) e participa no processo de validação de transações através de algoritmos de consenso. Este modelo distribuído oferece vantagens como maior resistência a falhas, transparência e imutabilidade dos dados [37]. A comunicação entre a aplicação web e a rede Blockchain está descrita na seção 3.5

3.3 Descrição da plataforma web

O desenvolvimento da plataforma web foi dividido em algumas camadas, descritas nas subseções a seguir.

3.3.1 Backend

O *backend* é a parte do sistema responsável por processar a lógica do aplicativo, gerenciar o banco de dados e as comunicações com a Blockchain. No contexto deste projeto, o *backend* foi desenvolvido utilizando a linguagem de programação PHP e o *framework* CodeIgniter 4.

3.3.2 Frontend

O *frontend* é a parte do sistema que interage diretamente com o usuário e é responsável pela apresentação e interação dos dados. O desenvolvimento do *frontend* envolve o uso de tecnologias como HTML, CSS e *JavaScript*. Neste projeto, foi utilizado o *framework Bootstrap* para facilitar a criação de uma interface responsiva e agradável.

3.4 Descrição da Blockchain

A implementação da Blockchain neste projeto tem como objetivo fornecer um mecanismo seguro e descentralizado para registrar e rastrear as transações e informações relacionadas aos animais e aos proprietários. A Blockchain selecionada para este projeto foi a *Hyperledger Fabric*, uma plataforma de Blockchain de código aberto desenvolvida para aplicações empresariais [74] [75]. A seguir são apresentadas algumas razões para a escolha do *Hyperledger Fabric*:

1. Permissão e privacidade: O *Hyperledger Fabric* é um protocolo de Blockchain permissionado, o que significa que cada participante da rede precisa de permissão para ingressar. Isso é útil para o rastreamento de rebanhos de gado, onde os participantes podem ser produtores, reguladores e outras partes interessadas que necessitam de permissões específicas e acesso controlado às informações.
2. Desempenho e escalabilidade: O *Hyperledger Fabric* possui um mecanismo de consenso modular e altamente escalável, adequado para lidar com grandes volumes de transações e participantes. Isso é importante no contexto da pecuária, onde a quantidade de dados e as interações entre os participantes podem ser bastante significativas.
3. Modularidade e flexibilidade: O *Hyperledger Fabric* permite a personalização de vários componentes, como o algoritmo de consenso e o gerenciamento de identidades. Essa modularidade facilita a adaptação do protocolo para atender às necessidades específicas do projeto, como a garantia de rastreabilidade eficiente e segura do gado.
4. Suporte a contratos inteligentes: O *Hyperledger Fabric* suporta o desenvolvimento e a execução de contratos inteligentes (chamados de *chaincode*) em várias linguagens de programação, como Go, *JavaScript* e Java. Isso permite a implementação de lógica de negócios complexa, como o gerenciamento de propriedade e rastreamento de animais, diretamente na Blockchain.
5. Interoperabilidade com sistemas existentes: O *Hyperledger Fabric* pode ser facilmente integrado com sistemas existentes através de APIs RESTful, o que facilita a comunicação entre o sistema web proposto e a Blockchain.

Em resumo, o *Hyperledger Fabric* é uma escolha apropriada para este projeto devido à sua natureza permissionada, desempenho, escalabilidade, flexibilidade e suporte a contratos inteligentes.

3.4.1 Estrutura dos blocos e consenso

A estrutura dos blocos no *Hyperledger Fabric* inclui um cabeçalho, que contém informações sobre o bloco, como o *hash* do bloco anterior e o número do bloco, e uma lista de transações. Cada transação é representada por um conjunto de ações e informações associadas. O consenso na *Hyperledger Fabric* é alcançado por meio de um protocolo de ordem de transações que garante a consistência e a integridade das informações armazenadas na Blockchain [75].

3.4.2 Algoritmo de consenso selecionado

O algoritmo de consenso selecionado para este projeto foi o *Raft*. Desenvolvido por Diego Ongaro e John Ousterhout na Universidade de Stanford, o algoritmo *Raft* é um protocolo de consenso distribuído de replicação de *log* baseado em liderança [76]. Foi projetado para garantir a consistência e a tolerância a falhas em sistemas distribuídos, onde múltiplos servidores ou nós cooperam para manter um estado replicado e consistente [76].

De maneira simplificada, uma implementação de *Raft* elege um líder que, a partir de então, é responsável por tomar todas as decisões relativas ao estado do banco de dados. Isso evita a necessidade de comunicação extra entre as réplicas durante leituras e gravações individuais. Cada nó do sistema rastreia o líder atual e encaminha as solicitações para ele.

O *Raft* é estruturado em torno do conceito de um *log* replicado. Quando o líder recebe uma solicitação, ele primeiro armazena uma entrada correspondente em seu *log* local durável. Em seguida, este *log* local é replicado para todos os seguidores, ou réplicas. Assim que a maioria das réplicas confirma que o *log* foi persistido, o líder aplica a entrada e instrui as réplicas a fazerem o mesmo. Em caso de falha do líder, uma réplica com o *log* mais atualizado assume a liderança.

Além de definir como o grupo toma decisões, o *Raft* também especifica o protocolo para adicionar novos membros e remover membros do grupo. Esta funcionalidade torna o *Raft* uma escolha natural para gerenciar mudanças de topologia em sistemas distribuídos, proporcionando uma maneira robusta e eficiente de manter a consistência e a confiabilidade do sistema. A Figura 10 representa um diagrama de fluxo do algoritmo *Raft*.

Entre as vantagens do *Raft*, destaca-se a facilidade de entendimento proporcionada

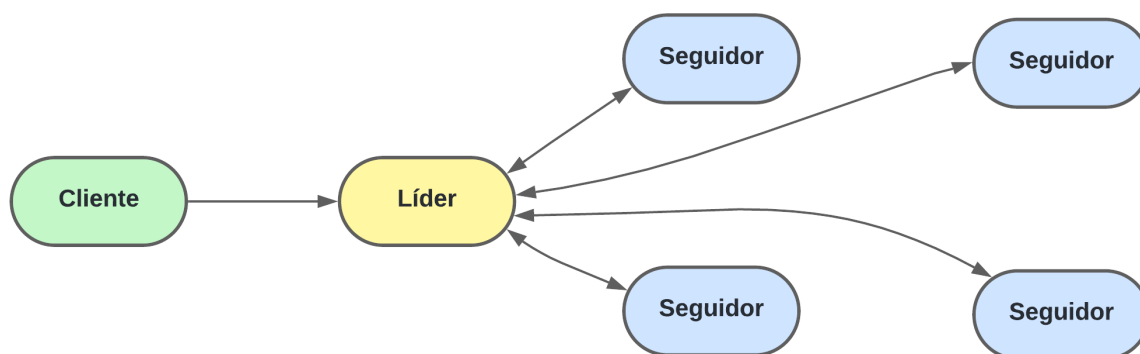


Figura 10 – Diagrama de Fluxo do Algoritmo *Raft*. Fonte: Autor.

pela sua estrutura modular, com separação clara entre eleição de líder, replicação de log e segurança. Além disso, o *Raft* garante consistência e tolerância a falhas, assegurando que todos os servidores mantenham um estado consistente, mesmo na presença de falhas de servidor. O *Raft* também proporciona baixa latência e alta disponibilidade em ambientes distribuídos, tornando-o uma escolha popular para a implementação de sistemas distribuídos que dependem de um estado replicado e consistente.

3.4.3 Configuração e implantação da rede Blockchain

A configuração e implantação da rede Blockchain no *Hyperledger Fabric* envolve a definição de organizações, *peers*, *orderers* e *channels*. Além disso, é necessário configurar e instalar o ambiente de desenvolvimento, incluindo o *Hyperledger Fabric* SDK, o *Docker* e o *Composer*. A configuração e implantação da rede envolvem também a definição de políticas de acesso e de consenso, a criação e instalação de *chaincodes*, e a criação de *channels* de comunicação entre as organizações [74].

3.4.4 Contratos Inteligentes e *chaincode*

Os Contratos Inteligentes (ou *smart contracts*) são programas que executam a lógica de negócios definida pelos desenvolvedores e são responsáveis por validar e executar transações na rede *Hyperledger Fabric*. O *chaincode* é a implementação específica do *smart contract* na plataforma *Hyperledger Fabric* e é escrito em uma linguagem de programação suportada, como *Go*, *JavaScript* ou *Java* [74]. Neste projeto, os Contratos Inteligentes são responsáveis pela validação das informações inseridas e pela execução das operações relacionadas aos animais e aos proprietários, e foram desenvolvidos na linguagem *JavaScript*.

3.5 Integração entre a plataforma web e a Blockchain

A integração entre a aplicação web e a Blockchain é uma parte crucial do projeto, pois permite que os usuários do sistema eCattle interajam com a Blockchain por meio da interface web, garantindo a segurança e a integridade das informações armazenadas na rede.

É importante enfatizar que o sistema eCattle contempla apenas uma organização dentro da rede Blockchain desenvolvida. Devido à característica distribuída da rede Blockchain, vários outros sistemas podem ser desenvolvidos e integrados à rede proposta por este projeto, através da ingresso de suas organizações à rede.

A integração entre o *backend* da plataforma web e a rede Blockchain foi realizada por meio de uma API RESTful, desenvolvida para este fim e descrita na subseção 3.5.1, permitindo a comunicação bidirecional entre as duas partes. Essa integração possibilita a realização de consultas e transações de forma transparente e segura, garantindo que as informações armazenadas estejam sempre atualizadas e sejam acessíveis apenas por usuários autorizados. Além disso, essa API tem a característica de chamar funções do *chaincode* e executar transações, sendo usada para consultar e modificar os dados armazenados na Blockchain, além de invocar os Contratos Inteligentes para executar a lógica de negócios definida para o sistema eCattle. Isso permite um melhor aproveitamento das vantagens de descentralização, imutabilidade e transparência proporcionadas pela tecnologia Blockchain, melhorando a confiabilidade e a rastreabilidade das informações inerentes aos rebanhos de gado [75].

O processo de integração envolveu várias etapas, detalhadas a seguir:

1. Design da API: A API RESTful foi projetada para expor *endpoints* para todas as operações necessárias, incluindo a criação, leitura, atualização e exclusão de registros relacionados a animais, fazendas, proprietários e transações.
2. Implementação da API: Utilizando o *framework* CodeIgniter 4, os *endpoints* da API foram implementados para lidar com solicitações HTTP, validar entradas e formatar respostas.
3. Interação com a Blockchain: Cada endpoint da API foi conectado à rede Hyperledger Fabric através do Fabric SDK. Isso envolveu a invocação de funções de *chaincode*, a submissão de transações e a consulta ao ledger.
4. Medidas de Segurança: Para garantir a transmissão segura de dados, a API utilizou HTTPS, e mecanismos de autenticação foram implementados para verificar a identidade dos usuários e autorizar ações.

A Figura 11 ilustra a sequência de interações entre a plataforma web, a API e a rede Blockchain eCattle durante um processo típico de transação.

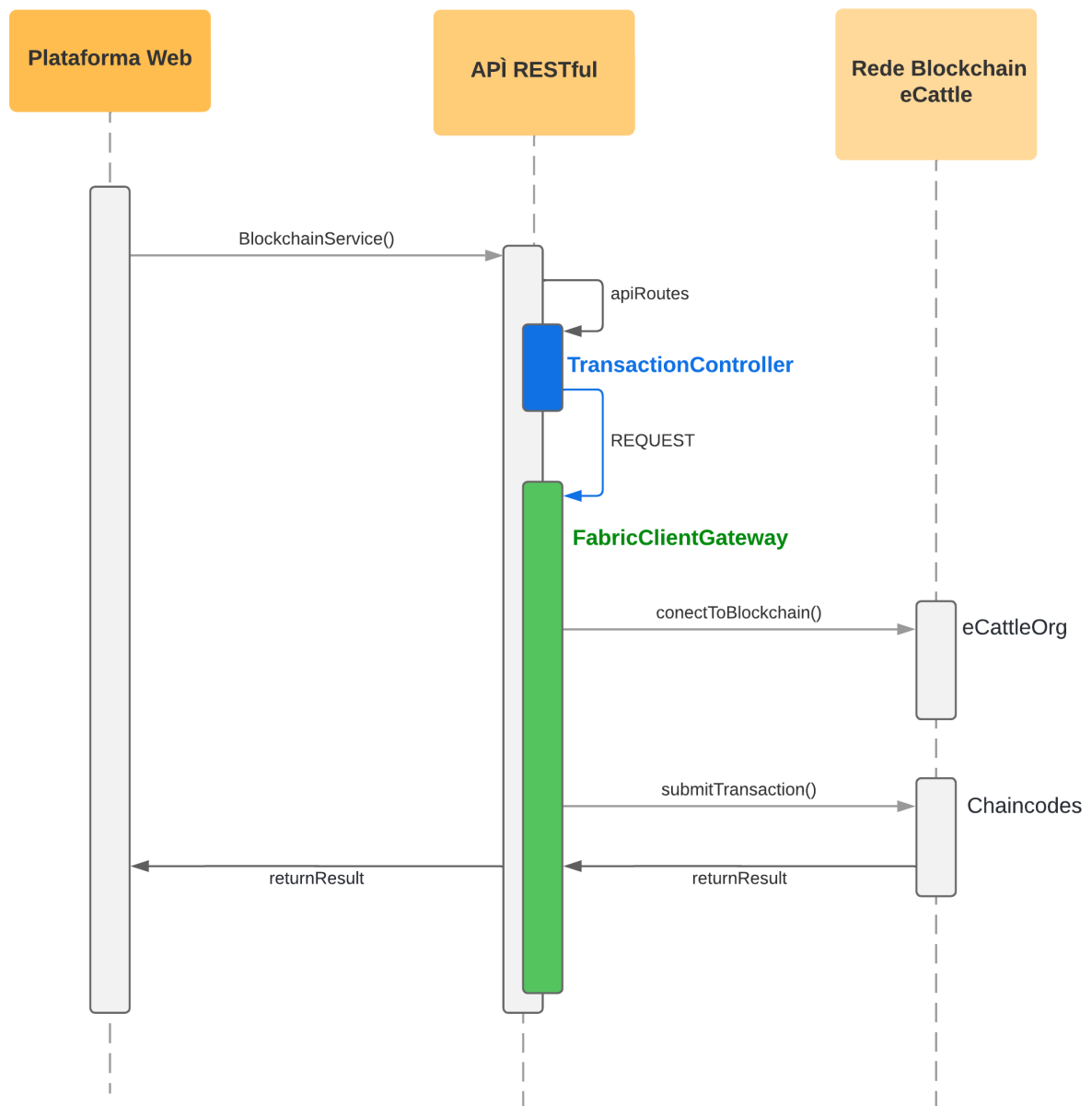


Figura 11 – Diagrama de sequência para a integração da API RESTful com a rede Blockchain. Fonte: Autor.

Esta seção explora ainda a configuração das carteiras de identidade, a arquitetura técnica, os componentes envolvidos na comunicação, a implementação dos Contratos Inteligentes, e a escolha do algoritmo de consenso *Raft*, essenciais para garantir a integridade, segurança e transparência das operações.

3.5.1 Arquitetura da API RESTful

A API RESTful, desenvolvida para intermediar a comunicação entre o sistema eCattle e a Blockchain, foi construída com Node.js e expressa a funcionalidade dos *end-*

points que correspondem às operações do *chaincode*. O *FabricClientGateway* desempenha um papel crucial, gerenciando as conexões de rede e invocando os Contratos Inteligentes apropriados para executar as transações na Blockchain. A robustez do sistema é garantida através de tratamento de erros e logs detalhados que capturam e reportam falhas tanto na aplicação quanto na comunicação com a Blockchain. A Figura 12 ilustra a comunicação entre a API e a rede Blockchain, com o retorno de sucesso da rede contendo o código *hash* gerado pela transação na Blockchain.

```
Body da requisição recebida: {
  tipo: 'carne',
  idade: '60',
  sexo: 'F',
  raca: 'Nelore',
  data_nascimento: '2024-03-08',
  peso_atual: '200',
  fazenda_atual_id: '27',
  tag: '51',
  fazenda_origem_id: '27',
  animalId: 78
}
Resultado da transação: {"success":true,"message":"Animal 78 criado com sucesso.,"animalId":78,"hash_id":"3851864062c83a7cd91581b5917f36027830eedf5fd6861c790cbc4135aede46"}
```

Figura 12 – Requisição da API para a Blockchain. Fonte: Autor.

3.5.2 Comunicação entre Aplicação web e Blockchain

A plataforma eCattle utiliza o *BlockchainService*, uma biblioteca PHP desenvolvida para facilitar a interação com a Blockchain. A biblioteca gerencia as requisições à API RESTful, que é projetada especificamente para atuar como uma ponte entre a aplicação web e a rede Blockchain. As requisições são realizadas através do método HTTP apropriado, onde cada chamada é autenticada e validada antes de ser processada pela Blockchain, garantindo assim a segurança e a integridade dos dados. O Código 3.1 ilustra o processo de encapsulamento das requisições:

Código 3.1 – Função para Solicitações à Blockchain.

```
1 private function requestBlockchain($method, $url, $data) {
2     try {
3         $response = $this->client->request($method, $url, ['json' =>
4             $data]);
5         $result = json_decode($response->getBody(), true);
6         if (!$result['success']) {
7             throw new \RuntimeException($result['message']);
8         }
9         return $result;
10    } catch (\Exception $e) {
11        throw new \RuntimeException('Erro: ' . $e->getMessage());
12    }
```

Este método de tratamento de erros assegura que qualquer falha na comunicação seja prontamente identificada e tratada, minimizando o risco de inconsistências de dados.

3.6 Gerenciamento de Identidades com Carteiras

No contexto de Blockchain, uma carteira digital é usada para armazenar e gerenciar as credenciais dos usuários, facilitando a execução segura de transações. No sistema eCattle, é utilizado o módulo *Wallets* do Fabric SDK para criar e gerenciar carteiras de identidade. Cada organização da rede Blockchain deve possuir sua própria carteira digital, contendo os certificados e chaves privadas para realização das transações.

A configuração inicial das identidades é feita por meio de um *script*, configurado na API RESTful, que prepara a carteira do administrador da organização eCattle, como mostrado no trecho de Código 3.2.

Código 3.2 – *Script* de Configuração de Identidade de Administrador.

```
1 const { Wallets } = require('fabric-network');
2 const fs = require('fs');
3 const path = require('path');
4
5 async function setupAdminIdentity() {
6   const wallet = await Wallets.newFileSystemWallet('./wallet');
7   const adminExists = await wallet.get('admin.ecattleorg');
8   if (!adminExists) {
9     const identity = {
10      credentials: {
11        certificate: fs.readFileSync('Admin@ecattleorg.ecattle.com-cert.
12        pem', 'utf8'),
13        privateKey: fs.readFileSync('Admin@ecattleorg.ecattle.com-key.
14        pem', 'utf8'),
15      },
16      mspId: 'ecattleorgMSP',
17      type: 'X.509',
18    };
19    await wallet.put('admin.ecattleorg', identity);
20  }
21 }
```

Este processo envolve carregar as credenciais do administrador, que incluem um certificado e uma chave privada gerados para a organização eCattleOrg, salvando-os na carteira. Isso habilita o administrador da organização a autenticar e interagir com a rede Blockchain de maneira segura.

3.7 Banco de dados híbrido

O projeto proposto utiliza um banco de dados híbrido, combinando um Sistema de Gerenciamento de Banco de Dados (SGBD) local com outro banco de dados gerenciado pela rede Blockchain. Essa abordagem permite que o sistema aproveite as vantagens

dos dois tipos de armazenamento de dados, garantindo segurança e imutabilidade para informações críticas e maior flexibilidade para dados menos sensíveis.

Informações pertinentes ao rebanho de gado são armazenadas tanto no banco local quanto na rede Blockchain, enquanto informações como senhas de usuários e alterações de senha, são armazenadas somente no banco de dados local [77].

Para o SGBD local, o *MySQL* foi escolhido devido à sua popularidade, robustez e ampla compatibilidade com diferentes linguagens de programação e plataformas, incluindo PHP, que é utilizado no desenvolvimento do sistema web. O *MySQL* é um sistema de gerenciamento de banco de dados relacional de código aberto, amplamente utilizado em aplicações web e conhecido por sua eficiência e facilidade de uso [78].

Para a rede Blockchain, o banco de dados escolhido foi o *GoLevelDB*. Além de ser a escolha padrão para o armazenamento de dados no *Hyperledger Fabric*, é uma implementação leve de um armazenamento de dados de chave-valor que é adequado para pequenas quantidades de dados que requerem acesso rápido, tanto para leitura quanto para gravação. Ele é integrado diretamente no *peer* do *Fabric* e não oferece suporte a consultas ricas, sendo otimizado para cenários onde a performance da leitura e gravação são críticas [74]. Suas principais vantagens são:

1. Aplicações que não requerem consultas complexas aos dados armazenados.
2. Cenários onde a performance de leitura e escrita é uma prioridade.
3. Quando a simplicidade da implementação e a manutenção são importantes.

3.7.1 Estrutura do banco de dados local e entidades

A estrutura do banco de dados é baseada no modelo Entidade-Relacionamento (ER), que representa as entidades envolvidas no sistema e as relações entre elas. As principais entidades do banco de dados incluem *Animal*, *Fazenda*, *condicao_saude*, *Rastreamento*, *Proprietario* e *Transacao*. Cada entidade possui atributos específicos e está relacionada a outras entidades por meio de chaves estrangeiras. O diagrama poderá ser visto em detalhes no Capítulo 4, Seção 4.4.2.

Por exemplo, a entidade *Animal* possui atributos como idade, sexo, raça e peso atual, além de relacionamentos com as entidades *Fazenda* e *condicao_saude*. A entidade *Fazenda* inclui atributos como nome, localização e relação com a entidade *Proprietario*. A integridade referencial e as restrições de chave são aplicadas para garantir a consistência dos dados armazenados no banco de dados.

3.7.2 Integração entre o SGBD local e a Blockchain

A integração entre o SGBD local e a plataforma Blockchain é uma parte crucial do projeto, pois permite a combinação das vantagens de ambos os tipos de armazenamento de dados. As informações críticas, como transações de gado, são armazenadas na Blockchain, garantindo a imutabilidade e a segurança dos dados. Por outro lado, as informações mais voláteis como informações de funcionários e alterações de senhas dos usuários, são armazenadas no SGBD local, proporcionando maior flexibilidade e eficiência no gerenciamento dos dados.

Os *Controllers* da aplicação web são responsáveis por intermediar a comunicação entre o banco de dados *MySQL* e a Blockchain, garantindo que as informações sejam mantidas sincronizadas e consistentes entre os dois sistemas. Além disso, os *Models* são responsáveis por executar consultas e atualizações nos dados armazenados no SGBD local, utilizando comandos SQL e APIs específicas para o *MySQL*, de acordo com as ações executadas pelos usuários do sistema.

A combinação de um SGBD local com a plataforma Blockchain permite a criação de uma solução híbrida, segura e flexível para gerenciar os dados do sistema de rastreamento de gado. A utilização da API RESTful para a comunicação entre o SGBD local e a Blockchain garante uma integração eficiente e consistente dos dados armazenados em ambas as plataformas. Além disso, a aplicação web é responsável por manter a sincronia e consistência entre os dois sistemas, garantindo um gerenciamento eficaz dos dados e proporcionando uma experiência de usuário mais completa e confiável.

3.8 Considerações Finais

Neste capítulo, foram abordados os aspectos essenciais do sistema eCattle, incluindo sua arquitetura, funcionamento e os diagramas que ilustram suas partes e interações. Foi detalhado como o sistema foi projetado para melhorar a rastreabilidade e gestão de rebanhos na indústria pecuária, utilizando a tecnologia Blockchain para garantir segurança, transparência e imutabilidade dos dados.

O Capítulo 4 discutirá em detalhes a implementação da plataforma web do sistema eCattle, abordando os requisitos e modelagem, bem como os desafios enfrentados durante o desenvolvimento. Será apresentada a metodologia de desenvolvimento adotada e como cada componente da arquitetura foi implementado para garantir uma solução escalável.

4 Desenvolvimento da Plataforma web

4.1 Considerações Iniciais

O desenvolvimento de plataformas web robustas e eficientes exige a seleção de um *framework* de desenvolvimento que ofereça tanto flexibilidade quanto poderosas ferramentas de otimização de código. Neste contexto, o *CodeIgniter 4* se destaca devido às suas características de desempenho, facilidade de uso e capacidade de escalabilidade.

CodeIgniter 4 foi escolhido para o desenvolvimento desta plataforma devido à sua arquitetura leve e a capacidade de oferecer um alto desempenho em aplicações web de pequena a grande escala. Diferente de outros *frameworks* que podem apresentar uma curva de aprendizado íngreme e uma sobrecarga de recursos, o *CodeIgniter 4* proporciona um equilíbrio ideal entre funcionalidade e simplicidade.

O objetivo principal ao escolher *CodeIgniter 4* para o desenvolvimento desta plataforma foi criar uma solução web que não apenas atenda às necessidades operacionais específicas, mas que também seja sustentável, segura e escalável. A implementação foi focada em aproveitar as capacidades nativas do *framework* para maximizar a eficiência operacional e a qualidade do código.

Ao longo deste capítulo, será discutido em detalhes as fases de desenvolvimento da plataforma, incluindo o planejamento arquitetônico e a implementação de funcionalidades específicas.

4.2 Arquitetura de Desenvolvimento

A arquitetura de desenvolvimento foi dividida em três camadas principais: *frontend*, *backend* e banco de dados. O *frontend* é responsável pela interação com o usuário e pela apresentação dos dados, sendo desenvolvido utilizando as tecnologias HTML, CSS e *JavaScript*. O *backend*, desenvolvido em PHP, gerencia a lógica do negócio e a comunicação entre o *frontend* e o banco de dados. O banco de dados híbrido é composto por um SGBD local (*MySQL*) e uma plataforma Blockchain (*Hyperledger Fabric*), garantindo a integridade e a segurança dos dados armazenados.

O sistema também possui funcionalidades específicas para o gerenciamento de rebanhos, como cadastro e atualização de informações dos animais, além de análise de dados e relatórios personalizados. Essas funcionalidades permitem aos usuários acompanhar o desempenho do rebanho através das visualizações do dados e informações inseridas previamente por eles.

O controle de acesso e a segurança são aspectos cruciais do sistema, e foram implementados utilizando técnicas e tecnologias de autenticação e autorização, garantindo que apenas usuários autorizados possam acessar e modificar informações sensíveis. Além disso, o sistema foi projetado para ser facilmente integrado a sistemas externos, como aplicativos móveis e outros *softwares* de gerenciamento de fazendas.

O sistema foi projetado para validação e autenticação de produtores rurais, de acordo com suas inscrições estaduais. A estrutura modular e a arquitetura baseada em componentes facilitam futuras implementações de novas funcionalidades, bem como a correção de problemas e a melhoria do desempenho do sistema.

Em resumo, o sistema proposto oferece uma solução para o gerenciamento, rastreamento e realização de transações de gado, combinando a segurança e a transparência da tecnologia Blockchain com a facilidade de uso e a flexibilidade de uma aplicação web moderna. O projeto foi desenvolvido com foco na usabilidade, escalabilidade e integração, possibilitando que o sistema possa crescer e evoluir junto com as necessidades dos usuários e do mercado.

4.3 Requisitos

Os requisitos do sistema proposto são divididos em requisitos funcionais e não funcionais. Os requisitos funcionais descrevem as funcionalidades e recursos que o sistema deve oferecer, enquanto os requisitos não funcionais estão relacionados a aspectos de desempenho, usabilidade, segurança e outros atributos de qualidade do sistema.

Os requisitos funcionais do sistema podem ser verificados na Tabela 2. Já os requisitos não funcionais estão descritos na Tabela 3.

4.4 Modelagem

A primeira fase do projeto foi a definição dos artefatos de modelagem do sistema, nessa etapa foram criados os diagramas *Unified Modeling Language* (UML) de Caso de Uso, Entidade Relacionamento, Classes, Pacotes e os Protótipos de Tela.

4.4.1 Casos de Uso

O primeiro diagrama a ser criado foi o diagrama de caso de uso, que representa graficamente as necessidades do usuário. Este diagrama fornece uma visão sobre a interação entre usuários e o sistema. Ele descreve os requisitos funcionais do sistema, oferecendo uma descrição clara e consistente do que o sistema deve fazer e servindo de base para a modelagem do projeto ao longo de seu desenvolvimento. Na Figura 13, pode-se observar o

Tabela 2 – Requisitos Funcionais

Requisitos Funcionais	Descrição
Cadastro e gerenciamento de usuários	O sistema deve permitir o cadastro de novos usuários e a manutenção de suas informações pessoais, incluindo nome, e-mail e senha.
Autenticação e autorização	O sistema deve oferecer mecanismos de autenticação e autorização para garantir que apenas usuários autorizados tenham acesso às funcionalidades e informações pertinentes.
Cadastro e gerenciamento de rebanhos	O sistema deve permitir o registro e a manutenção de informações detalhadas sobre os animais do rebanho, como identificação, espécie, raça, idade, peso, sexo e histórico de saúde.
Rastreamento e monitoramento do gado	O sistema deve integrar-se à rede Blockchain para armazenar e rastrear informações relevantes do gado ao longo de sua vida, incluindo movimentações, tratamentos médicos, eventos de reprodução e abate, quando aplicável.
Visualização e análise de dados	O sistema deve oferecer interfaces gráficas para a visualização e análise dos dados armazenados, permitindo aos usuários monitorar o desempenho do rebanho, identificar tendências e tomar decisões informadas.
Controle de acesso e segurança	O sistema deve garantir a segurança e a privacidade das informações armazenadas, implementando mecanismos de controle de acesso e criptografia.
Integração com sistemas externos	O sistema deve ser capaz de se integrar a outros sistemas e plataformas, como sistemas de gestão agropecuária e sistemas de monitoramento ambiental, para coletar e compartilhar informações relevantes.

diagrama de caso de uso gerado, onde os usuários são representados por atores dispostos na parte externa do retângulo que representa o sistema, enquanto as funcionalidades do sistema são descritas através dos balões (casos de uso).

A seguir, é apresentada de forma detalhada a descrição dos casos de uso e atores do diagrama, destacando suas funcionalidades e responsabilidades.

No gerenciamento de animais, o sistema permite que o proprietário ou administrador cadastre um novo animal, incluindo informações como tipo, idade, sexo, raça, data de nascimento e peso atual. Além disso, é possível atualizar informações de animais já cadastrados, consultar informações de um animal específico ou uma lista de animais cadastrados e, como ação estendida, deletar um animal do sistema, uma ação que deve ser usada com cautela.

No gerenciamento de fazendas, o sistema permite que o proprietário ou administrador cadastre uma nova fazenda, com detalhes como nome e localização. Também é possível atualizar informações de fazendas já cadastradas e consultar informações de uma

Tabela 3 – Requisitos Não Funcionais

Requisitos Não Funcionais	Descrição
Desempenho	O sistema deve ser capaz de processar e responder às solicitações dos usuários de forma rápida e eficiente, garantindo uma experiência satisfatória para o usuário.
Escalabilidade	O sistema deve ser projetado para suportar um aumento no número de usuários e volume de dados, garantindo que o desempenho e a qualidade do serviço não sejam comprometidos.
Usabilidade	As interfaces do sistema devem ser intuitivas e fáceis de usar, permitindo que os usuários realizem suas tarefas de forma eficiente e sem dificuldades.
Disponibilidade	O sistema deve estar disponível para acesso e uso a qualquer momento, minimizando interrupções e tempos de inatividade.
Segurança	O sistema deve garantir a proteção das informações armazenadas contra acessos não autorizados, ataques e vazamento de dados.
Portabilidade	O sistema deve ser compatível com diversos dispositivos e navegadores, permitindo que os usuários acessem e utilizem o sistema em diferentes plataformas e ambientes.
Manutenibilidade	O sistema deve ser projetado de forma modular e com código claro e documentado, facilitando a manutenção, atualizações e correções de erros.
Conformidade regulatória	O sistema deve atender aos requisitos legais e regulatórios aplicáveis ao setor agropecuário, garantindo a conformidade e evitando penalidades ou infrações.

fazenda específica ou uma lista de fazendas cadastradas.

Para o gerenciamento de proprietários, o sistema oferece funcionalidades para cadastrar novos proprietários, incluindo dados como nome, endereço, telefone, CPF e e-mail. Os administradores podem também atualizar informações de proprietários cadastrados e consultar dados específicos ou listas de proprietários.

O monitoramento da condição de saúde dos animais permite que usuários e administradores registrem novas condições de saúde, com informações sobre diagnóstico, tratamento e observações, como por exemplo infestações por Berne. É possível também consultar o histórico de condições de saúde registradas para um animal específico.

No que se refere ao gerenciamento de transações utilizando a Blockchain, o sistema permite que o proprietário ou administrador registre novas transações, incluindo detalhes sobre o animal, vendedor e comprador. O histórico de transações pode ser consultado, e a autenticidade de transações registradas pode ser validada.

Por fim, o sistema permite a visualização e análise de dados gerais. Usuários podem

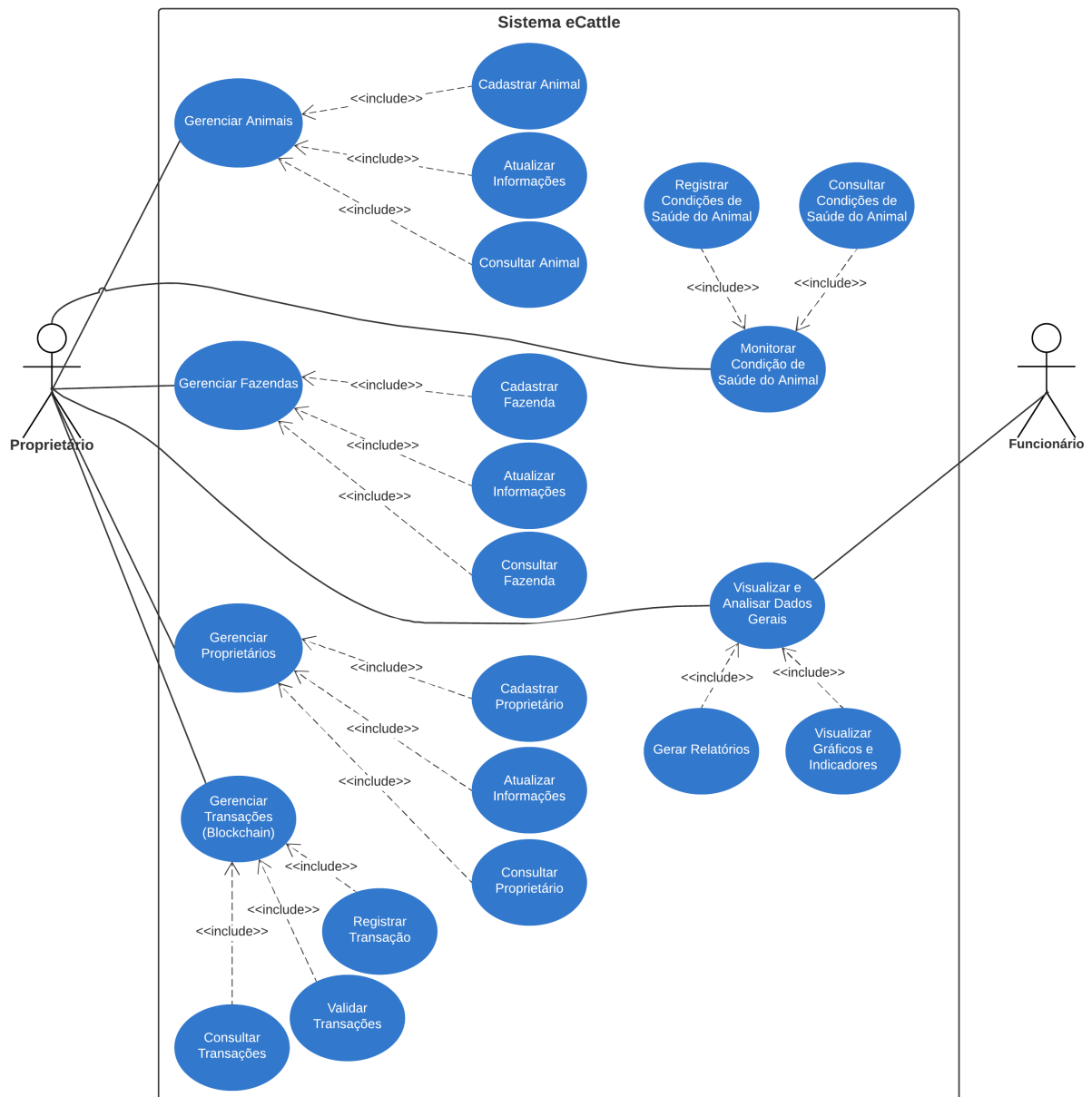


Figura 13 – Diagrama de casos de uso UML do sistema proposto. Fonte: Autor.

gerar relatórios com informações agregadas sobre animais, fazendas e proprietários, bem como visualizar gráficos e indicadores para analisar dados e identificar tendências ou padrões.

4.4.2 Banco de Dados

Um diagrama de entidade-relacionamento (DER) foi utilizado para ilustrar o modelo de dados do sistema, mostrando as entidades envolvidas e seus relacionamentos.

A Figura 14 representa o DER do sistema, destacando a integração com a Blockchain. As entidades marcadas como 'Integração com a Blockchain' precisam ser validadas antes de serem adicionadas no banco de dados, garantindo a integridade e segurança dos

dados. Isso significa que cada transação envolvendo essas entidades deve ser verificada na Blockchain antes de sua confirmação. O campo *hash_id* em cada uma dessas entidades resgata sempre o *hash* da transação do último estado do *ledger* na Blockchain, correspondente à operação realizada. Esse *hash* serve como uma prova imutável de que a transação foi registrada e validada na Blockchain, proporcionando um alto nível de confiança e rastreabilidade para todas as operações do sistema.

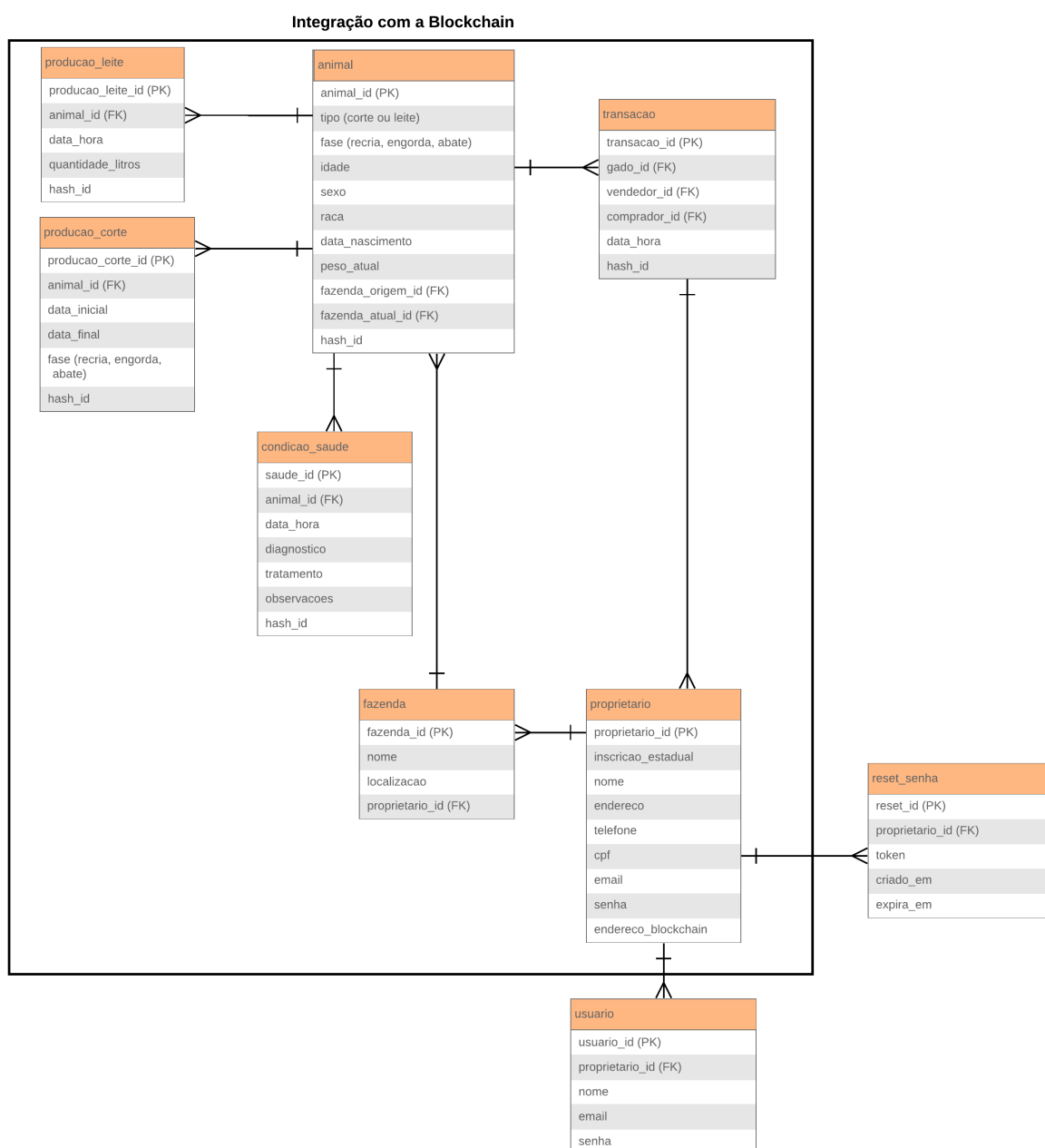


Figura 14 – Diagrama Entidade-Relacionamento. Fonte: Autor.

A seguir, estão descritas detalhadamente as relações entre as entidades do banco de dados. Um animal está associado a uma fazenda de origem e a uma fazenda atual, sendo que a fazenda de origem é onde o animal nasceu e a fazenda atual é onde ele está atualmente. Cada fazenda pode ter vários animais, configurando uma relação do

tipo 1:N entre a entidade fazenda e a entidade animal, utilizando as chaves estrangeiras *fazenda_origem_id* e *fazenda_atual_id*.

Uma fazenda pertence a um proprietário e cada proprietário pode possuir várias fazendas. Esta relação é do tipo 1:N entre a entidade proprietário e a entidade fazenda, utilizando a chave estrangeira *proprietario_id*. Um animal pode ter várias condições de saúde registradas ao longo do tempo, estabelecendo uma relação do tipo 1:N entre a entidade animal e a entidade *condicao_saude*, utilizando a chave estrangeira *animal_id*.

Uma transação representa a venda de um animal, envolvendo um vendedor e um comprador. Esta relação é do tipo 1:N entre a entidade animal e a entidade transacao, utilizando a chave estrangeira *gado_id*. Além disso, a entidade transacao também está relacionada à entidade proprietário por meio das chaves estrangeiras *vendedor_id* e *comprador_id*, representando os proprietários envolvidos na transação. Um proprietário pode vender ou comprar animais, e cada transação de compra ou venda envolve dois proprietários, um vendedor e um comprador. A entidade transacao armazena informações sobre essas transações, estabelecendo uma relação do tipo 1:N entre a entidade proprietário e a entidade transacao, com as relações estabelecidas por meio das chaves estrangeiras *vendedor_id* e *comprador_id* na entidade transacao.

Se um animal for destinado à produção de leite, ele pode ter vários registros de produção de leite, configurando uma relação do tipo 1:N entre a entidade animal e a entidade *producao_leite*, utilizando a chave estrangeira *animal_id*. Se um animal fizer parte da cadeia de corte, ele pode ter vários registros de eventos de fases, como recria, engorda e abate, estabelecendo uma relação do tipo 1:N entre a entidade animal e a entidade *producao_corte*, utilizando a chave estrangeira *animal_id*.

O acesso ao sistema é definido por um usuário, que pode ser administrador ou não do sistema. Um usuário do tipo proprietário (administrador) estará associado a um único proprietário, e um proprietário poderá ter vários usuários associados a ele. Esta relação é do tipo 1:N entre a entidade proprietário e a entidade usuario, utilizando a chave estrangeira *proprietario_id*. Caso um proprietário perca a senha, poderá ser solicitada a redefinição da mesma. A entidade *reset_senha* armazena informações de reset de senhas, estabelecendo uma relação do tipo 1:N entre a entidade proprietário e a entidade *reset_senha*, utilizando a chave estrangeira *proprietario_id*.

4.4.3 Diagrama de Classes

O diagrama de classes é uma ferramenta de modelagem visual utilizada na linguagem UML para representar o projeto de um sistema orientado a objetos [79]. Ele ilustra as classes envolvidas no sistema, seus atributos, métodos e os relacionamentos entre elas. Um diagrama de classes fornece uma base sólida para entender o funcionamento e a or-

ganização de um sistema e é fundamental durante a fase de desenvolvimento de software. Na Figura 15 está representado o diagrama de classes proposto para este projeto.

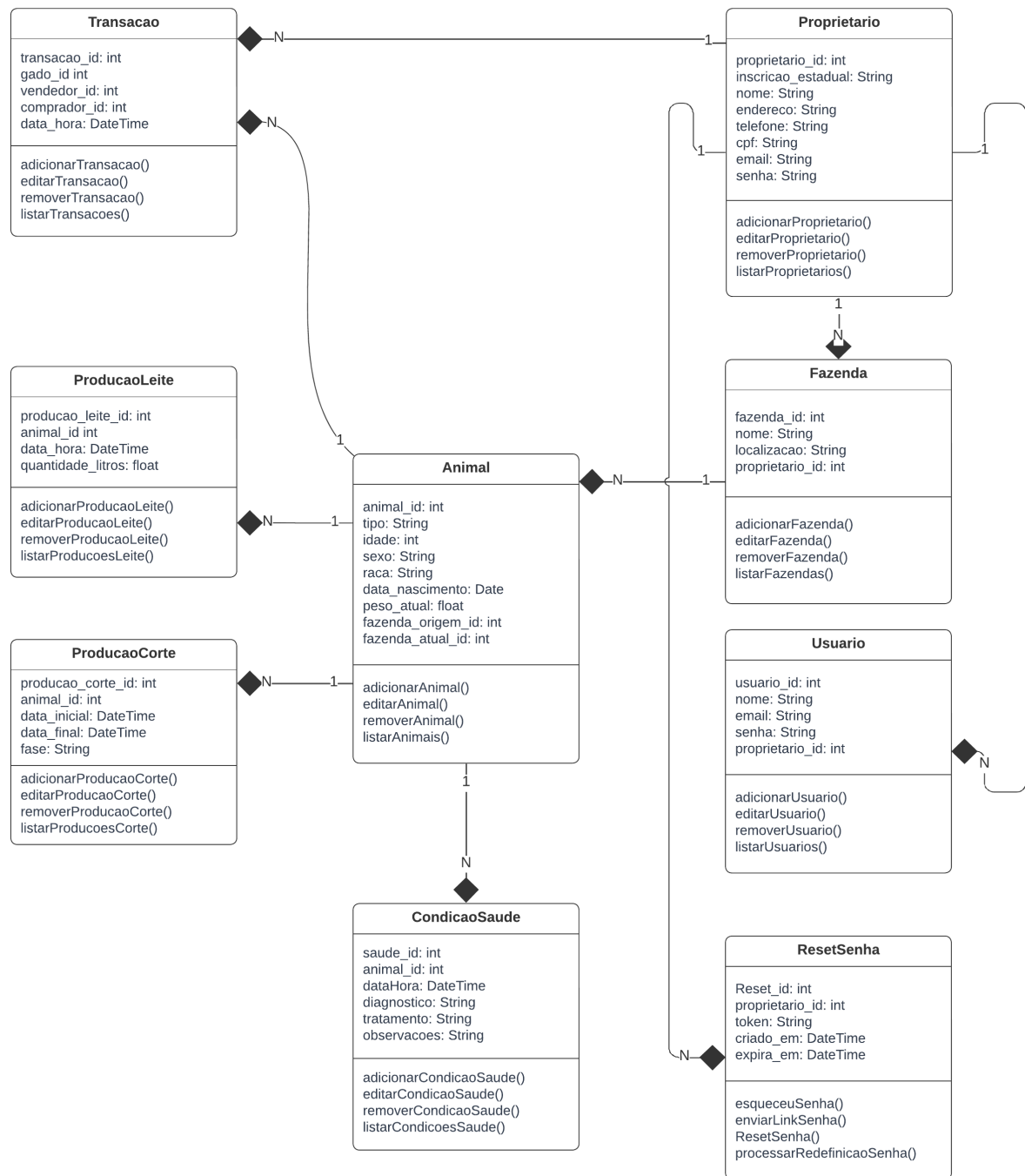


Figura 15 – Diagrama de Classes. Fonte: Autor.

Com base no diagrama de classes, o fluxo de interação entre as entidades pode ser descrito da seguinte forma: um Proprietário é responsável por uma ou mais Fazendas, e cada Fazenda deve ter pelo menos um Proprietário. Um Proprietário pode criar, editar e excluir suas Fazendas, além de consultar informações sobre elas. Uma Fazenda pode conter muitos Animais, e cada Animal deve pertencer a uma Fazenda. O Proprietário ou um Usuário com acesso à Fazenda pode adicionar, editar e remover Animais, bem

como consultar informações sobre eles. Cada Animal pode ter múltiplas instâncias de *condicao_saude*, Rastreamento, *producao_leite*, *producao_corte* e *evento_animal*. Esses registros são criados e gerenciados pelos usuários do sistema que têm acesso aos Animais.

Para cada instância de *condicao_saude*, os usuários podem registrar informações de saúde dos Animais, como diagnóstico, tratamento e observações. Essas informações podem ser consultadas, editadas e removidas pelos usuários autorizados. Se um Animal for do tipo “leite”, os usuários podem registrar informações sobre *producao_leite*, como data e hora e quantidade de litros produzidos. Essas informações podem ser consultadas, editadas e removidas pelos usuários autorizados. Se um Animal for do tipo “corte”, os usuários podem registrar informações sobre *producao_corte*, como a fase de produção (recria, engorda ou abate), data inicial e data final. Essas informações também podem ser consultadas, editadas e removidas pelos usuários autorizados.

A entidade *Transacao* armazena informações sobre transações comerciais envolvendo Animais, como venda e compra. Esses dados são armazenados na Blockchain e podem ser consultados pelos usuários autorizados. As operações de criação, edição e remoção são limitadas pelas permissões e regras da Blockchain. A entidade *Usuario* representa os usuários do sistema, com exceção dos proprietários, tendo acesso limitado. Cada usuário possui um nome, e-mail, senha e tipo de usuário. Eles podem criar, editar e remover suas próprias informações e gerenciar suas preferências. Por fim, a entidade *ResetSenha* representa a possibilidade de um proprietário solicitar a redefinição de sua senha, conforme achar necessário.

O diagrama de classes além de ilustrar a estrutura do sistema web, demonstrando as relações entre as entidades e as operações que os usuários podem realizar, ajuda a compreender o fluxo de informações e interações entre as entidades e a lógica de negócios subjacente. Isso facilita a implementação e a manutenção do sistema, bem como a comunicação com os usuários e outras partes interessadas.

4.5 Implementação Técnica do Sistema eCattle

Esta seção descreve em detalhes o funcionamento dos módulos que compõem o sistema, destacando as práticas de desenvolvimento que asseguram a segurança, integridade e usabilidade dos dados. Esta parte é vital para entender como cada componente do software contribui para a gestão de fazendas e recursos agropecuários, facilitando a escalabilidade, manutenção e futuras atualizações do sistema. Este conhecimento é essencial para desenvolvedores, administradores e *stakeholders* que dependem do eCattle para decisões operacionais e estratégicas.

4.5.1 Implementação do Painel Inicial

A implementação do Painel Inicial do sistema eCattle serve como a interface central para visualização de dados e acesso rápido às funcionalidades mais utilizadas do sistema. Este módulo proporciona uma visão geral das operações agropecuárias gerenciadas pelo sistema.

A estrutura do *Controller* é composta por várias construções e dependências importantes, incluindo *Auth*, que gerencia a autenticação e identificação do usuário, e diversos modelos como **AnimalModel**, **FazendaModel**, **UserModel**, **SaudeAnimalModel**, **TransacaoModel**, **ProducaoCorteModel** e **ProducaoLeiteModel**. Esses componentes fornecem acesso e gerenciamento de dados específicos de animais, fazendas, usuários, saúde dos animais, transações e produção de corte e leite.

O Painel Inicial oferece várias funcionalidades, como a visualização de dados agregados (total de animais, fazendas, usuários e alertas de saúde), o gerenciamento de transações (com visualizações em gráficos das transações mensais e por categoria) e a geração de relatórios em *Portable Document Format* (PDF), facilitando a documentação e análise fora do sistema.

Detalhes técnicos e o fluxo de dados incluem a função `index()`, que carrega e exibe a página principal do painel inicial, buscando dados como o número de animais, fazendas e usuários, além de preparar os dados para os gráficos de transações. As funcionalidades `gerarRelatorioPDF()` e `gerarRelatorioFuncionario()` permitem a geração de relatórios detalhados sobre todos os aspectos gerenciados pelo usuário, adaptados para visualização em PDF. A função `indexFuncionario()` apresenta uma versão do painel inicial adaptada para funcionários, com visualizações e permissões ajustadas conforme o nível de acesso do usuário.

Os benefícios da implementação incluem eficiência operacional, centralizando informações essenciais e permitindo uma rápida avaliação das condições atuais e tomada de decisão mais informada. A flexibilidade é garantida pelo suporte a diferentes níveis de acesso e personalização do painel conforme o tipo de usuário, seja proprietário ou funcionário, assegurando que as informações sejam relevantes e adequadas. A integração e segurança são aprimoradas pela consistente atualização e integração de dados de diferentes módulos em um único ponto de acesso, aumentando a segurança e integridade dos dados manipulados.

Este módulo não apenas simplifica a interação do usuário com o sistema eCattle, mas também serve como um ponto importante para a monitoração e gerenciamento eficaz das operações agropecuárias, evidenciando a robustez e a abrangência do sistema em fornecer soluções tecnológicas avançadas para o setor.

4.5.2 Implementação do Módulo de Gerenciamento de Fazendas

A implementação do Módulo de Gerenciamento de Fazendas define o controle e gestão das propriedades dentro do sistema eCattle. Este módulo proporciona funcionalidades para a criação, atualização e monitoramento de fazendas, integrando operações locais com tecnologias de Blockchain para garantir segurança e transparência.

A estrutura do *Controller* de fazendas inclui o **FazendaModel**, que gerencia todas as operações de banco de dados relacionadas às fazendas, o **ProprietarioModel**, utilizado para operações relacionadas aos proprietários das fazendas, e o **BlockchainService**, que facilita a integração com a Blockchain, garantindo operações seguras e rastreáveis.

As funcionalidades do módulo de fazendas incluem a listagem de fazendas (*index*), que verifica a autenticação do usuário e lista todas as fazendas associadas ao proprietário logado, exibindo-as através da *View fazenda_index*. A criação de fazendas (*create e store*) envolve a exibição de um formulário para a adição de novas fazendas e a validação e registro de novas fazendas no banco de dados e na Blockchain, revertendo a operação em caso de falhas para manter a consistência dos dados. A atualização de fazendas (*edit e update*) carrega o formulário preenchido com os dados existentes da fazenda para edição e, após a validação, atualiza os dados da fazenda no banco de dados local e na Blockchain, assegurando a consistência entre as plataformas. A remoção de fazendas (*delete*) remove registros de fazendas do banco de dados e redireciona para a lista de fazendas.

A integração com a Blockchain ocorre nas funções de *store* e *update*, onde cada fazenda registrada ou atualizada tem seu registro correspondente criado ou modificado na Blockchain, fornecendo uma camada adicional de segurança e rastreabilidade para as operações agrícolas.

Os benefícios da implementação incluem segurança e transparência, com transações imutáveis garantidas pela Blockchain, oferecendo um alto nível de segurança e transparência para os registros das fazendas. A flexibilidade e escalabilidade do módulo facilitam a expansão e personalização conforme as necessidades dos usuários e a evolução do mercado. A usabilidade e acesso a informações são aprimorados por interfaces intuitivas e funções automatizadas, permitindo que os proprietários e gestores de fazendas gerenciem suas propriedades com eficácia e eficiência.

Este módulo não apenas simplifica a gestão de fazendas dentro do sistema eCattle, mas também estabelece uma base para futuras expansões e melhorias, alinhando tecnologia de ponta com as necessidades práticas do setor agropecuário.

4.5.3 Implementação do Módulo de Gerenciamento de Animais

O módulo de gerenciamento de animais é um componente fundamental do sistema eCattle, responsável pela criação, atualização e monitoramento dos registros dos animais.

Este módulo segue uma estrutura clara e modular, facilitando a manutenção e expansão futura do sistema.

A estrutura do *Controller* de animais é composta por diversas dependências essenciais. O **AnimalModel** é encarregado das operações de banco de dados relativas aos animais, enquanto o **FazendaModel** é utilizado para gerenciar informações relacionadas às fazendas. O **BlockchainService** assegura a integração com a Blockchain para garantir transparência e segurança nas transações. Além disso, os *Models* **ProducaoCorteModel** e **ProducaoLeiteModel** são responsáveis pela gestão de dados específicos de produção de corte e leite, e o **SaudeAnimalModel** administra os registros de saúde dos animais.

As funcionalidades principais do módulo incluem a listagem de animais, onde a função *index* compila uma lista de todos os animais sob a propriedade do usuário logado, enriquecendo cada entrada com detalhes adicionais como a fazenda de origem e informações de saúde ou produção atuais. A criação de animais é gerida pelas funções *create* e *store*. A função *create* apresenta um formulário para a inserção de novos animais, preenchido com informações relevantes como as fazendas disponíveis. Já a função *store* processa o formulário, validando e inserindo dados no banco e registrando-os na Blockchain para garantir a consistência e segurança.

A atualização de animais é gerida pelas funções *edit* e *update*. A função *edit* carrega o formulário de edição preenchido com dados existentes de um animal específico, enquanto a função *update* atualiza as informações do animal nos registros do banco de dados e na Blockchain, garantindo a integridade dos dados através da consistência transacional. Além disso, a gestão de eventos de saúde e produção é realizada pelas funções *addEvento*, *eventoCorte* e *eventoLeite*, permitindo a adição e gestão de eventos relacionados à saúde e à produção dos animais, como vacinações ou registros de produção leiteira, cada um com sua respectiva lógica de validação e integração Blockchain.

A integração com a Blockchain é vital para a transparência e segurança das operações. O *BlockchainService* registra cada novo animal e atualizações no *ledger* distribuído, o que é essencial no contexto agropecuário para a rastreabilidade proposta pelo sistema.

Os benefícios da implementação deste módulo são múltiplos. A segurança e transparência são asseguradas pelo uso da Blockchain, que registra todas as transações de maneira segura e permanente. A flexibilidade e escalabilidade são garantidas pelas funcionalidades detalhadas e a modularidade do código, permitindo adaptar o sistema às necessidades emergentes do setor. A usabilidade é aprimorada através de interfaces intuitivas e um processo claro de listagem e edição, melhorando significativamente a interação do usuário com o sistema.

Essa implementação não apenas atende às necessidades operacionais correntes, mas também estabelece uma base sólida para futuras expansões e melhorias no sistema

eCattle, destacando-se como uma solução eficaz para a gestão moderna de fazendas.

4.5.4 Implementação do Módulo de Gerenciamento de Funcionários

O Módulo de Gerenciamento de Funcionários do sistema eCattle é essencial para administrar as permissões e acessos dos usuários envolvidos nas operações agropecuárias. Este módulo oferece funcionalidades completas para o gerenciamento de contas de usuário, desde a criação até a atualização de perfis.

A estrutura do *Controller* de usuários é composta por várias dependências cruciais. O **UserModel** é responsável por todas as operações de banco de dados relacionadas aos usuários, enquanto o **Auth** é utilizado para autenticação e verificação do estado de login do usuário.

As funcionalidades principais do módulo incluem a listagem de funcionários, onde a função *index* lista todos os funcionários associados ao proprietário logado, mostrando-os na *View usuario_index*. A criação de funcionários é gerida pelas funções *create* e *store*. A função *create* exibe um formulário para registrar novos funcionários, enquanto a função *store* valida os dados de entrada e registra novos funcionários no banco de dados, garantindo a conformidade com as normas de segurança através de senhas criptografadas.

A edição de funcionários é gerida pelas funções *edit* e *update*. A função *edit* carrega um formulário preenchido com dados existentes para edição, enquanto a função *update* atualiza os dados do funcionário após a validação, assegurando que as informações estejam corretas e seguras. Além disso, o gerenciamento de perfil é realizado pelas funções *perfil* e *atualizarPerfil*, permitindo ao usuário visualizar e editar seu próprio perfil, e atualizar as informações do perfil do usuário logado após a validação dos dados fornecidos.

Os benefícios da implementação deste módulo são múltiplos. O controle de acesso gerencia quem pode acessar diferentes níveis de informações e funcionalidades dentro do sistema. A flexibilidade e segurança oferecidas garantem um gerenciamento de usuários eficaz, mantendo padrões elevados de segurança. A usabilidade é aprimorada através de interfaces intuitivas e processos de validação de entrada, assegurando uma gestão eficaz e segura dos usuários.

Este módulo não só facilita a administração de pessoal e segurança dentro do sistema eCattle, mas também ajuda a manter a integridade e a confidencialidade das operações e dados agrícolas.

4.5.5 Implementação do Módulo de Gerenciamento de Transações

O módulo de gerenciamento de transações é responsável pela gestão financeira e operacional dentro do sistema eCattle, facilitando a transferência de gado entre proprietários. Este módulo garante uma base sólida para transações seguras e rastreáveis.

A estrutura do *Controller* de transações é composta por várias dependências importantes. O **TransacaoModel** gerencia todas as operações de banco de dados relacionadas às transações. O **FazendaModel** é utilizado para acessar informações sobre fazendas, fundamentais para as transações. O **ProprietarioModel** fornece acesso aos dados dos proprietários envolvidos nas transações, e o **AnimalModel** é necessário para gerenciar informações sobre os animais que são objetos das transações.

As funcionalidades principais do módulo incluem a listagem de transações, onde a função *index* exibe todas as transações associadas ao proprietário logado, facilitando a visualização e gestão financeira. A criação de transações é gerida pelas funções *create* e *store*. A função *create* exibe um formulário que permite ao usuário iniciar uma nova transação, preenchendo detalhes como o comprador, o animal em questão e o valor da transação. A função *store* valida e processa os dados da transação, inserindo-os no banco de dados e registrando-os na Blockchain. Esta função garante que todas as informações sejam consistentes e seguras, revertendo operações em caso de falha.

Além disso, a atualização de informações de transações permite a edição de transações já registradas, ajustando detalhes conforme necessário e garantindo que todas as alterações sejam refletidas de forma segura na Blockchain.

A integração com a Blockchain, através do *BlockchainService*, é essencial para garantir a integridade e a transparência das transações dentro do sistema eCattle. Cada transação é registrada em um ledger distribuído, fornecendo uma prova inalterável de todas as operações realizadas.

Os benefícios da implementação deste módulo são múltiplos. A segurança e transparência são garantidas pela integração com a Blockchain, o que assegura que todas as transações sejam rastreáveis e seguras, aumentando a confiança entre os usuários. A flexibilidade é oferecida pela estrutura modular do código, permitindo adaptar facilmente o sistema às necessidades dos usuários e a regulamentos emergentes. A usabilidade é aprimorada através de interfaces claras e a facilidade de realizar e gerenciar transações, tornando o sistema eCattle uma ferramenta valiosa para os proprietários de gado.

Esta implementação não só melhora a eficiência operacional, como também proporciona um ambiente de negócios seguro e confiável para os usuários do sistema eCattle, promovendo práticas comerciais éticas e transparentes no setor agropecuário.

4.6 Visualização de Dados na Plataforma web

A visualização de dados na plataforma web eCattle, proporciona aos usuários acesso fácil e intuitivo às informações necessárias para a gestão das operações agropecuárias. Cada módulo foi desenvolvido com o intuito de fornecer uma interface clara e

funcional para diferentes aspectos do gerenciamento de fazendas, animais, funcionários e transações.

4.6.1 Visualização da Tela Inicial

A tela inicial do sistema eCattle desempenha um papel fundamental como ponto de partida para a navegação e gerenciamento geral das operações agrícolas. Ela oferece uma visão geral rápida e acessível das principais métricas e funcionalidades do sistema, facilitando o acesso às informações mais importantes com eficiência e conveniência. A Figura 16 representa a tela de visualização da página inicial, sendo um ponto de entrada no sistema eCattle para os gestores agropecuários.

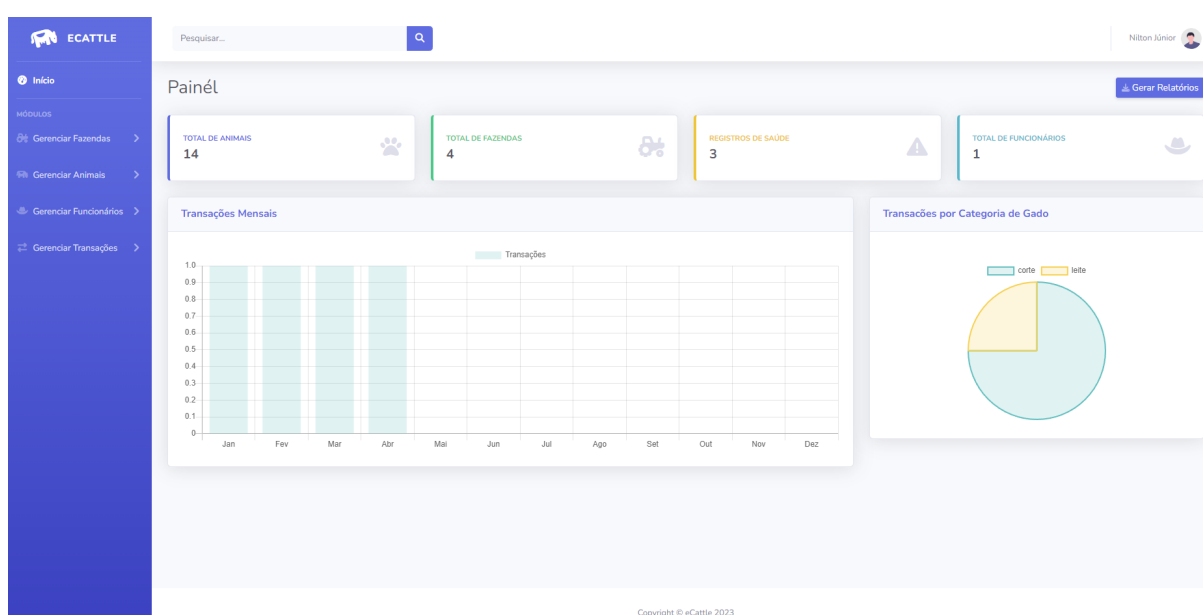


Figura 16 – Tela Inicial. Fonte: Autor.

A tela inicial exibe um painel de resumo que inclui informações como o total de animais, fazendas, e alertas de saúde, permitindo aos usuários obter um panorama instantâneo da situação atual de suas operações. Uma barra de pesquisa integrada permite aos usuários realizar pesquisas diretas no banco de dados do sistema, facilitando a localização rápida de informações específicas sobre animais, fazendas ou transações. *Links* rápidos para funcionalidades frequentemente utilizadas, como adição de novos registros ou geração de relatórios, estão disponíveis para garantir que as tarefas comuns sejam realizadas de maneira ágil. Além disso, a tela inicial oferece acesso a relatórios detalhados e análises de tendências, que são essenciais para a tomada de decisão baseada em dados, incluindo visualizações de dados e gráficos que destacam padrões de produção, saúde animal, e transações financeiras. A integração com outras partes do sistema proporciona uma transição suave para o gerenciamento detalhado de fazendas, funcionários, e transações, entre outros.

4.6.2 Visualização do Gerenciamento de Fazendas

O módulo de gerenciamento de fazendas no sistema eCattle tem o papel de administrar as propriedades e operações agrícolas. Ele fornece uma interface simplificada para o cadastro e a gestão das fazendas, integrando diversas funcionalidades essenciais que são acessíveis através de uma interface sistemática. A Figura 17 ilustra a tela de visualização das fazendas cadastradas pertencentes ao proprietário logado no sistema.

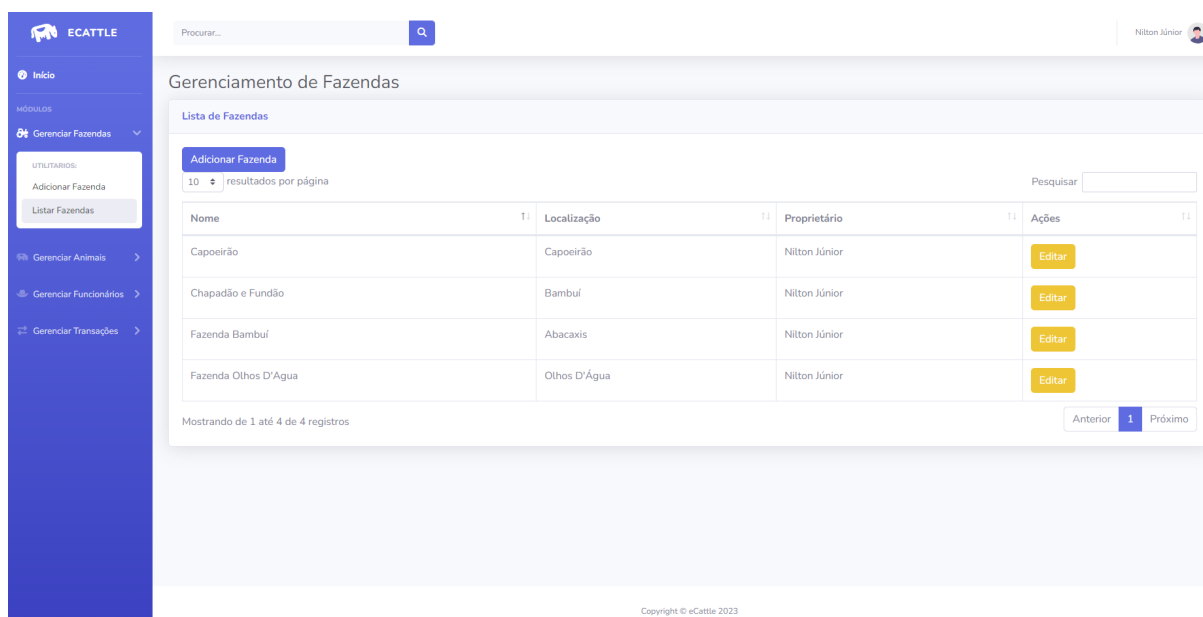


Figura 17 – Fazendas Cadastradas. Fonte: Autor.

O módulo permite aos usuários adicionar novas fazendas ao sistema através de um formulário detalhado que solicita informações como nome da fazenda, localização e detalhes do proprietário. Após o cadastro, os usuários podem visualizar uma lista de todas as fazendas registradas, podendo filtrar e pesquisar para acessar rapidamente informações específicas. Os usuários também podem editar informações de fazendas existentes para refletir quaisquer mudanças nos dados operacionais ou de propriedade. Este módulo interage de forma integrada com outros módulos do sistema, como o de gerenciamento de animais e transações, proporcionando uma visão unificada das operações agrícolas. A implementação eficaz do módulo de gerenciamento de fazendas ajuda os gestores a manter o controle sobre suas propriedades.

4.6.3 Visualização do Gerenciamento de Animais

O módulo de gerenciamento de animais no sistema eCattle é essencial para permitir que os usuários executem o cadastro e a administração dos animais. Este módulo proporciona uma série de funcionalidades que são acessadas através de uma interface organizada na barra lateral do sistema. A Figura 18 ilustra a tela de visualização de animais deste módulo.

Gerenciamento de Animais

Lista de Animais

Adicionar Animal

10 resultados por página

Pesquisar

Identificação	Tipo	Idade (dias)	Sexo	Raça	Data de Nascimento	Peso Atual (Kg)	Fazenda Origem	Fazenda Atual	Alerta	Alterar
1	corde	584	M	Nelore	2022-08-04	500	Chapadão e Fundão	Chapadão e Fundão	Engorda	Editar
2	corde	586	F	Nelore	2022-08-31	400	Chapadão e Fundão	Chapadão e Fundão		Editar
3	corde	546	M	Nelore	2022-09-11	300	Chapadão e Fundão	Chapadão e Fundão		Editar
3	leite	720	F	Nelore	2020-01-01	700	Fazenda do Servílio 2	Chapadão e Fundão		Editar
4	corde	530	F	Nelore	2022-10-26	300	Chapadão e Fundão	Chapadão e Fundão		Editar
4	corde	720	M	Nelore	2023-01-01	700	Fazenda do Servílio 1	Chapadão e Fundão	Abatido	Editar
4	corde	1	F	Nelore	2024-01-01	10000	Fazenda do Servílio 1	Chapadão e Fundão		Editar
5	corde	529	M	Nelore	2022-10-27	330	Chapadão e Fundão	Chapadão e Fundão		Editar
6	corde	345	F	Nelore	2023-04-29	300	Chapadão e Fundão	Chapadão e Fundão		Editar
7	corde	338	F	Nelore	2023-05-06	270	Chapadão e Fundão	Chapadão e Fundão		Editar

Mostrando de 1 até 10 de 14 registros

Anterior 1 2 Próximo

Figura 18 – Gerenciamento de Animais. Fonte: Autor.

A funcionalidade de adicionar novos animais é central para o módulo, permitindo aos usuários inserir detalhes significativos através de um formulário que solicita informações como espécie, raça, data de nascimento e identificação. Após o cadastro, os registros dos animais podem ser acessados e gerenciados através de uma visualização que lista todos os animais registrados, permitindo aos usuários aplicar filtros e realizar buscas específicas. Os usuários também têm a capacidade de atualizar informações de registros existentes e registrar eventos específicos que afetam os animais, como tratamentos médicos e vacinações. O módulo de gerenciamento de animais está integrado com outros componentes do sistema, como gerenciamento de fazendas e transações, permitindo uma visão holística das operações agropecuárias. A implementação desse módulo no sistema eCattle representa uma melhoria significativa na capacidade dos gestores agropecuários de monitorar e controlar a produtividade e saúde do rebanho.

4.6.4 Visualização do Gerenciamento de Funcionários

O módulo de gerenciamento de funcionários no sistema eCattle gerencia o pessoal envolvido nas operações agrícolas. Este módulo fornece uma interface simples através da qual os gestores podem acessar e gerenciar informações relacionadas aos funcionários. A Figura 19 apresenta a tela dos funcionários cadastrados no sistema.

Este módulo permite o registro de novos funcionários no sistema, coletando informações como nome, e-mail e senha durante o processo de cadastro. O sistema apresenta uma lista de todos os funcionários cadastrados, permitindo que os gestores visualizem detalhes específicos e realizem atualizações conforme necessário. A funcionalidade de atualização de informações permite editar dados como e-mail e senha, assegurando que os

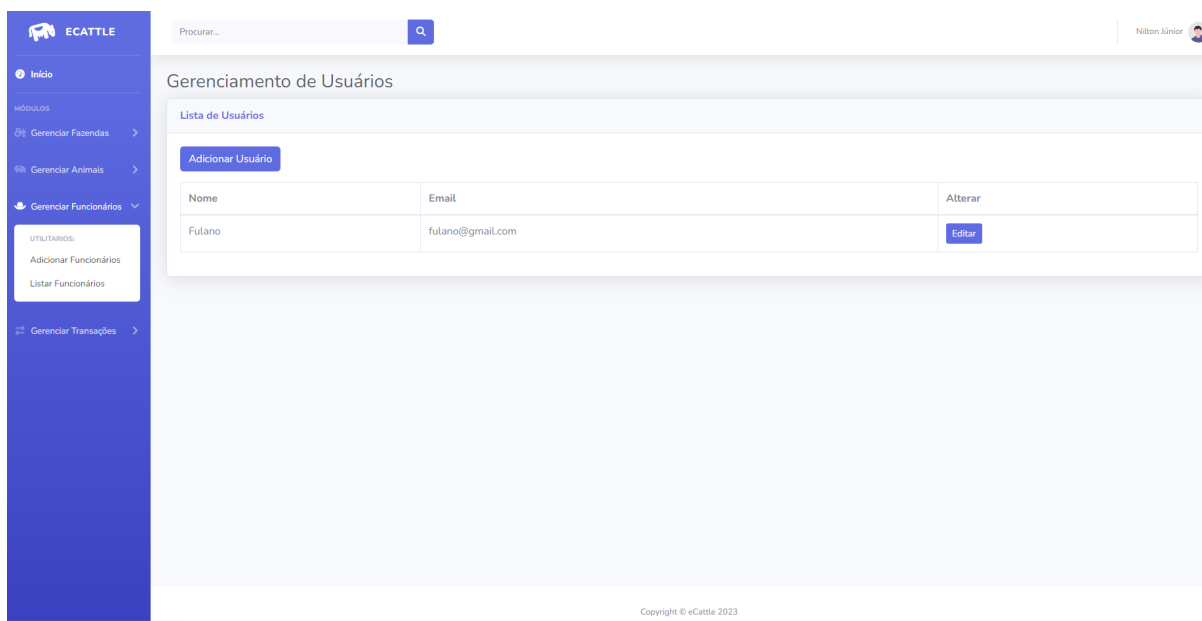


Figura 19 – Gerenciamento de Funcionários. Fonte: Autor.

registros estejam sempre atualizados. O módulo de gerenciamento de funcionários está integrado com outras funcionalidades do sistema, como gerenciamento de fazendas e animais, permitindo uma operação coesa e uma visão unificada das atividades agrícolas e de gestão de pessoal. Implementar um módulo dedicado ao gerenciamento de funcionários apoia a otimização das operações agrícolas, assegurando que os recursos humanos sejam geridos de forma eficaz.

4.6.5 Visualização do Gerenciamento de Transações

O módulo de gerenciamento de transações no sistema eCattle desempenha um papel fundamental na administração das operações comerciais relacionadas à compra e venda de animais. Este módulo oferece uma interface para o registro e acompanhamento de transações, garantindo a integridade dos dados e a transparência nas atividades financeiras. A Figura 20 apresenta a tela das transações cadastradas no sistema.

O módulo permite aos usuários cadastrar novas transações de compra ou venda de animais, coletando informações críticas como identificação do animal, detalhes do comprador e vendedor, data, hora e valor da transação. Os usuários podem acessar um registro completo das transações realizadas, incluindo detalhes históricos que facilitam o rastreamento financeiro e a gestão de inventário. Funcionalidades de filtragem e busca permitem localizar transações específicas rapidamente, aumentando a eficiência na gestão de dados. O sistema também permite que os usuários atualizem ou modifiquem transações após o registro inicial, garantindo que as informações se mantenham atualizadas em caso de mudanças ou correções necessárias. A eficaz gestão de transações por meio deste módulo contribui diretamente para a sustentabilidade das operações agrícolas, assegurando que

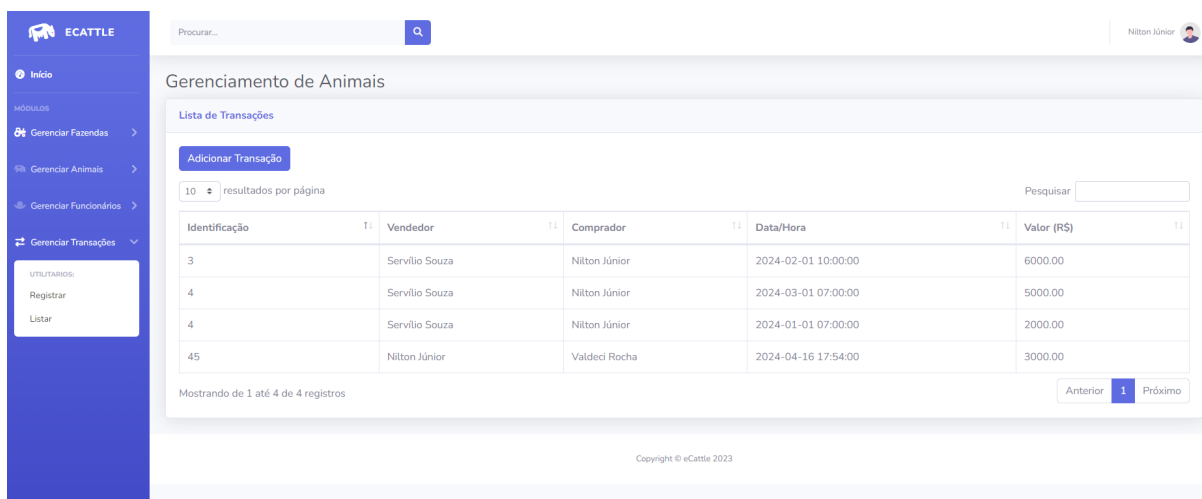


Figura 20 – Gerenciamento de Transações. Fonte: Autor.

todos os aspectos financeiros sejam geridos com transparência e eficiência.

4.7 Controle de acesso e segurança

Nesta seção são abordados os mecanismos de controle de acesso e segurança implementados para garantir a proteção das informações e a privacidade dos usuários. Os métodos de autenticação do sistema e o controle de acessos entre Proprietários e Funcionários são essenciais para a integridade operacional do sistema.

4.7.1 Autenticação e Gerenciamento de Sessão

A autenticação é gerida pelo *Controller AuthController*, utilizando práticas padrão de segurança disponibilizadas pelas bibliotecas integradas do *CodeIgniter 4*, validando credenciais de usuários através dos registros disponíveis no banco de dados. A sessão é iniciada após a validação bem-sucedida, armazenando informações essenciais do usuário, como seu ID, tipo de usuário (Proprietário ou Funcionário), e nome. Isso não apenas personaliza a experiência do usuário, mas também facilita o controle de acesso durante a sessão ativa.

4.7.2 Autorização e Controle de Acesso

Após autenticação, o sistema impõe controles de acesso diferenciados, delimitando as permissões e funcionalidades acessíveis a Proprietários e Funcionários. O *UsuarioController* gere essas funcionalidades específicas, como a criação e atualização de dados de funcionários, garantindo que cada usuário acesse somente as operações pertinentes ao seu papel.

4.7.3 Criptografia e Segurança de Dados

Para a segurança das credenciais, senhas são criptografadas utilizando o algoritmo BCRYPT antes de sua armazenagem no banco de dados [80]. Este procedimento assegura que as senhas não sejam armazenadas ou transmitidas em texto claro, protegendo-as contra acessos não autorizados mesmo em caso de violações de dados.

4.7.4 Medidas Adicionais de Segurança

O sistema implementa validações de e-mail durante o registro de novos usuários e processos de redefinição de senha. A redefinição de senha envolve a geração de tokens seguros e temporários, enviados ao usuário via e-mail, que permitem a redefinição da senha dentro de um intervalo de tempo definido para garantir a segurança.

Esses mecanismos compõem uma estrutura de segurança que protege os dados dos usuários contra acessos não autorizados, mantendo a integridade e a confidencialidade das informações. A implementação de tais controles é vital para manter a confiança dos usuários e para a sustentabilidade das operações agrícolas em um ambiente digital.

4.8 Considerações Finais

Ao longo deste capítulo, foi delineado o desenvolvimento do sistema web voltado para o rastreamento de gado. A escolha do *framework CodeIgniter*, conhecido por sua agilidade e capacidade de desenvolvimento rápido, demonstrou ser fundamental para a estruturação e funcionamento fluido do sistema. Combinado com a flexibilidade e estética do *Bootstrap*, o resultado foi uma interface responsiva, que atende a uma variedade de dispositivos e tamanhos de tela, garantindo que os usuários tenham uma experiência uniforme e eficaz.

O desenvolvimento de um sistema web conforme o proposto, possui várias vantagens inerentes. Primeiramente, ele centraliza informações, tornando o rastreamento e gerenciamento do gado mais prático e seguro, e minimizando erros que podem ocorrer em sistemas manuais ou fragmentados. A digitalização e organização dos dados também facilitam análises mais profundas, auxiliando na tomada de decisões informadas no setor agropecuário.

Além disso, a integração de tecnologias web modernas, como *CodeIgniter* e *Bootstrap*, não apenas melhora a performance do sistema, mas também simplifica a manutenção e futuras atualizações. Esta escolha tecnológica confirma um compromisso com a escalabilidade e adaptabilidade, garantindo que o sistema possa evoluir juntamente com as crescentes demandas do setor.

Em resumo, o desenvolvimento do sistema web para o rastreamento de gado não só responde a uma necessidade prática imediata, mas também sinaliza um passo em direção à modernização e otimização da gestão agropecuária. O investimento em tal plataforma reforça a ideia de que a tecnologia, quando aplicada de maneira criteriosa e estratégica, pode trazer inovações significativas e soluções duradouras para desafios tradicionais.

A seguir, o Capítulo 5 abordará o desenvolvimento da rede Blockchain, que é fundamental para garantir a segurança, transparência e rastreabilidade das operações dentro do sistema eCattle. Tal capítulo detalhará a arquitetura da rede Blockchain implementada, os protocolos de consenso utilizados e como a integração com a plataforma web foi realizada para oferecer uma solução completa para a gestão agropecuária.

5 Desenvolvimento da Rede Blockchain

5.1 Considerações Iniciais

Com o advento da tecnologia digital e a necessidade crescente de transparência e segurança, a arquitetura Blockchain emergiu como uma solução descentralizada. A Blockchain é uma estrutura de dados em cadeia composta por blocos de registros, cada um contendo uma lista de transações. Cada bloco está conectado ao anterior através de uma função criptográfica, formando uma cadeia imutável de registros [37].

5.1.1 Arquitetura Blockchain

A arquitetura Blockchain opera em uma rede descentralizada de nós, onde cada nó possui uma cópia completa da cadeia. O consenso é alcançado através de protocolos específicos, como PoW ou PoS, garantindo a integridade e a validação das transações [81]. Uma das principais vantagens dessa arquitetura é a resistência a falhas, fraude e manipulação, uma vez que alterar um bloco exigiria a alteração de todos os blocos subsequentes na cadeia.

5.1.2 Sistema web e Rede Blockchain para Gerenciamento de Transações de Gado

O setor agropecuário, em particular o mercado de gado, enfrenta desafios significativos em termos de rastreamento, transparência e eficiência nas transações. Para abordar essas questões, o presente projeto desenvolveu uma Plataforma web e uma Rede Blockchain para gerenciar transações de gado. Este sistema visa fornecer uma solução transparente e segura para o gerenciamento de transações, bem como um mecanismo de acompanhamento de eventos relacionados aos animais.

O sistema integra tecnologia Blockchain para registrar e validar todas as transações de compra e venda de gado, garantindo uma trilha de auditoria imutável e transparente. Além disso, o gerenciamento de eventos relacionados aos animais, como vacinação, nascimento e morte, também é capturado na Blockchain, fornecendo uma visão completa e precisa do histórico de cada animal.

5.2 Arquitetura da Rede Blockchain eCattle

A rede Blockchain eCattle é composta por duas organizações, uma Autoridade Certificadora, um nó de ordem e dois nós de *peers*, sendo um *peer* para cada organização. Ambos componentes estão ingressados em *channel* chamado eCattle Channel.

As organizações desempenham papéis vitais na rede Blockchain, assegurando sua funcionalidade, governança, segurança e ampla adoção. Elas operam os *peers*, desenvolvendo e validando transações que garantem a integridade do *ledger* descentralizado. Na governança, as organizações definem regras e políticas, resolvendo disputas e mantendo a ordem na rede.

Além disso, as organizações garantem a conformidade regulatória, assegurando que as operações do eCattle estejam alinhadas com as exigências definidas pelas regras de negócio, promovendo transparência e confiança. Como usuários finais, as organizações utilizam a Blockchain para rastrear e verificar transações de gado, melhorando a rastreabilidade e a segurança dos dados. No caso do sistema eCattle a organização eCattleOrg é responsável pelo envio das transações à rede Blockchain, enquanto a Org2 atua apenas validando e aceitando as transações conforme as regras previamente estabelecidas.

O *channel* desempenha um papel crucial ao fornecer um meio de comunicação e compartilhamento de dados seguro e privado entre as organizações participantes da rede. O compartilhamento e acesso das transações relacionadas ao sistema eCattle, ocorrem apenas nesse canal. Isso garante que as informações sensíveis e específicas, como dados de transações de animais, fazendas e proprietários, sejam acessíveis apenas para os membros autorizados do canal. Além disso, o uso de canais melhora a eficiência e a escalabilidade da rede, pois isola o tráfego e as operações de um grupo de participantes das demais atividades na rede Blockchain.

Os serviços da rede são definidos e gerenciados por meio do *Docker Compose*, através de um arquivo de configuração contendo informações de diretórios dos arquivos e regras da rede Blockchain [82]. Após a execução do *Docker Compose*, é criado um *container Docker* para cada um dos *peers* e *orderer* da rede. A rede opera dentro de um único *channel* chamado *eCattle Channel*. O arquivo *configtx.yaml* define organizações, políticas e capacidades. A geração de certificados é gerida pelo *crypto-config.yaml*. Políticas e certificados garantem a segurança e governança da rede.

O gerenciamento de identidade e controle de acesso são componentes fundamentais para garantir a integridade, transparência e segurança das transações. No sistema, certificados digitais emitidos por uma autoridade de certificação específica são usados para autenticar e identificar os nós de *peers* na rede Blockchain. Esses certificados são como uma espécie de carteira de identidade digital, fornecendo informações verificáveis e confiáveis sobre cada nó. No contexto deste projeto, os nós, como *Peer 1* e *Peer 2*, são

validados e associados às organizações eCattleOrg e Org2, respectivamente, por meio de dois *Membership Service Provider* (MSP), conhecidos como *eCattleOrg.MSP* e *Org2.MSP*.

Os MSPs são os mecanismos essenciais que definem o papel e os privilégios de cada nó dentro das organizações. Eles especificam as operações que um nó pode realizar e os dados que ele pode acessar ou modificar dentro da Blockchain. Cada nó de par na rede é único para uma organização, neste caso, eCattleOrg e Org2, sendo portanto, vinculado ao seu próprio MSP.

O funcionamento desse sistema de identidade é extensível a todas as entidades que interagem com a rede Blockchain do sistema eCattle. Isso inclui não apenas os nós de *peers*, mas também aplicativos, usuários finais e administradores, todos os quais devem ter uma identidade digital e um MSP associado para interagir de forma segura e eficaz com a rede.

O recurso MSP fornece uma camada de segurança e governança, permitindo que o sistema eCattle opere com a confiabilidade e eficiência necessárias para gerenciar transações complexas e dados sensíveis relacionados à gestão de animais, fazendas e proprietários. Em resumo, o MSP serve como a espinha dorsal que conecta a identidade digital à função organizacional específica, garantindo uma operação segura e eficaz de toda a rede eCattle.

Quatro Contratos Inteligentes foram desenvolvidos, os quais são responsáveis pela gestão de animais, fazendas, proprietários e transações. Esses contratos são importantes para o manuseio das entidades dentro do sistema.

- ***AnimalContract***: este contrato lida com a criação e manipulação de animais em uma fazenda e inclui várias classes para produção e saúde animal.
- ***FarmContract***: este contrato lida com a criação e manipulação de fazendas.
- ***OwnerContract***: este contrato lida com a criação e manipulação de proprietários de fazendas.
- ***TransactContract***: este contrato lida com a criação e manipulação de transações relacionadas à pecuária, como a venda de um animal.

O mecanismo pelo qual a plataforma web e os *peers* interagem entre si para garantir que o registro de todos os *peers* seja mantido consistente é mediado por nós especiais chamados *orderer*. A Figura 21 apresenta um fluxograma de como a interação entre os *peers* e o *orderer* do sistema opera no eCattle.

A interação entre os *peers* e o *orderer* é crucial para manter a consistência do *ledger*. A atualização do *ledger* ocorre através de um processo de três fases:

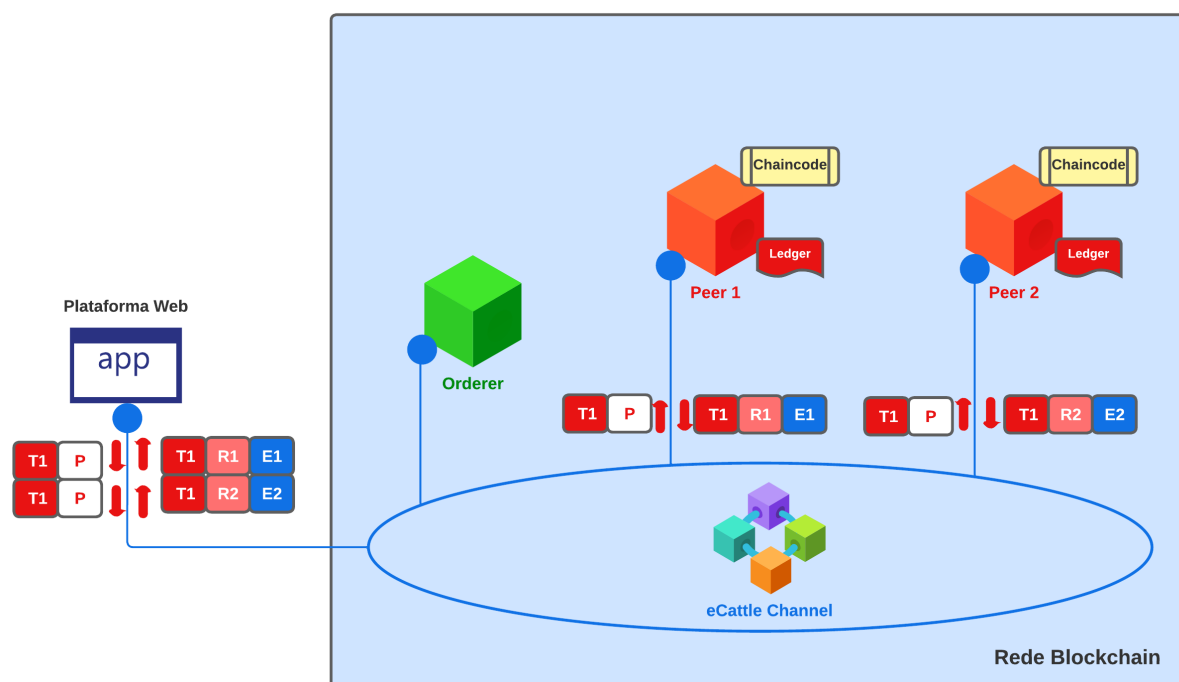


Figura 21 – Interação entre *peers* e *orderer*. Fonte: Autor.

1. **Proposta:** A plataforma web gera uma proposta de transação (P) a partir da transação T1 e a envia para um subconjunto de *peers* endossantes (*Peer 1* e *Peer 2*). Esses *peers* avaliam a proposta executando os *chaincodes* (neste caso, um dos 4 Contratos Inteligentes desenvolvidos), mas não a aplica ao *ledger*; eles apenas a assinam e retornam as respostas R1 e R2, endossadas por E1 e E2. A decisão sobre quais *peers* endossarão a transação é guiada por uma política de endosso vinculada ao *channel* eCattle.
2. **Ordenação e Empacotamento de Transações em Blocos:** Nesta fase, o *orderer* é responsável por receber as transações endossadas e as organiza em blocos. Este processo é vital para manter a ordem e a consistência entre os *peers*.
3. **Validação e Confirmação:** Os blocos são distribuídos de volta aos *peers* através do *orderer*. Cada par valida as transações nos blocos para garantir que elas são consistentemente endossadas pela organização eCattleOrg antes de serem confirmadas no *ledger*, como pode ser observado na Figura 22.

O sistema eCattle integra a tecnologia Blockchain para registrar e validar todas as transações de compra e venda de gado, garantindo um rastro de auditoria imutável e transparente. Além disso, a gestão de eventos relacionados aos animais, como vacinação, nascimento e morte, também é capturada no Blockchain, fornecendo uma visão completa e precisa da história de cada animal.

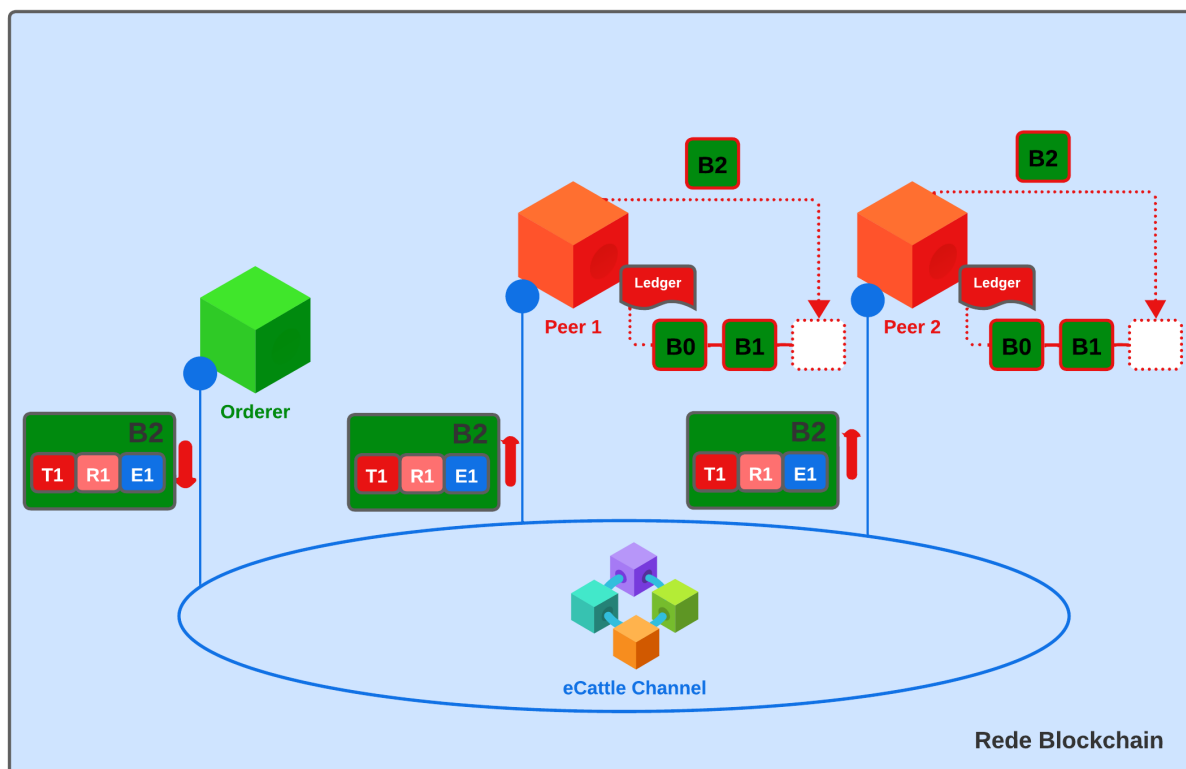


Figura 22 – Validação e confirmação de blocos de transações. Fonte: Autor.

As regras de negócio definem a lógica subjacente e as restrições do sistema, incluindo a classificação dos animais como 'carne' ou 'leite'; a associação de cada animal a um proprietário e a duas fazendas (origem e atual); rastreamento do estágio e da saúde dos animais; gestão de eventos específicos para tipos de animais; e gestão de transferência de animais ou transações de compra e venda.

Para garantir a integridade e a consistência dos dados, várias validações foram implementadas:

- **Validação da Fazenda:** Garante que a fazenda atual pertence ao mesmo proprietário do animal.
- **Validação do Tipo de Evento:** Verifica se o evento corresponde ao tipo de animal.
- **Validação do Estágio Animal:** Assegura que o estágio atual está em conformidade com os estágios permitidos para aquele tipo de animal.
- **Validação do Proprietário:** Confirma que o proprietário que realiza a operação está logado.
- **Validação do Abate:** Checa se o animal não foi abatido antes da atualização ou adição de um evento.

- **Validação da Saúde:** Certifica que a condição de saúde pertence ao proprietário logado e que a data não está no futuro.

Após a modelagem do Sistema eCattle e a criação da rede Blockchain, a integração entre eles foi realizada.

5.2.1 Dockers da Rede Blockchain

Dentro da rede Blockchain *Hyperledger Fabric* implementada para o sistema eCattle, os *containers Docker* são responsáveis por hospedar instâncias de *chaincodes* específicos para cada tipo de transação ou entidade gerenciada pela plataforma. Cada *container Docker* listado representa uma instância específica que é executada isoladamente para tratar as operações de negócio codificadas.

Esses *containers* são gerenciados pelo *Docker* e configurados para interagir com o restante da infraestrutura da Blockchain, incluindo *peers* e *orderers*, através de portas de rede específicas. Por exemplo, *containers* como *peer0.ecattleorg.ecattle.com* servem como nós da rede que hospedam os *chaincodes* e gerenciam o estado da Blockchain e o *ledger* de transações. A comunicação entre esses *containers* e o aplicativo cliente é intermediada por APIs RESTful e *gateways* que facilitam a execução de chamadas de *chaincode*, validação de transações e consenso na rede.

A arquitetura de *containers Docker* contribui significativamente para a flexibilidade, escalabilidade e resiliência do sistema Blockchain, permitindo uma distribuição eficaz dos recursos computacionais e uma gestão eficiente do ambiente de execução dos *chaincodes*. A Tabela 4 ilustra todos os *containers* da rede em execução.

Os *chaincodes*, como *farmchaincode*, *ownerchaincode*, *animalchaincode*, e *transactionchaincode*, responsáveis por encapsular os Contratos Inteligentes, são carregados em *containers Docker* específicos, e são inicializados dinamicamente pela rede *Hyperledger Fabric* conforme a necessidade de execução das transações. Essa abordagem modular e isolada garante que cada aspecto da lógica de negócio seja mantido seguro, encapsulado e otimizado para performance, além de permitir a atualização e manutenção independentes de cada *chaincode* sem afetar o restante do sistema. Esses *chaincodes* são instalados e gerenciados nos *peers* de acordo com as regras de negócios propostas. A Figura 23 ilustra consultas ao *ledger* da rede, destinadas à cada um dos *chaincodes* instalados no *peer* da organização eCattleOrg.

```
root@5ff6fc9a4f4:/opt/gopath/src/github.com/hyperledger/fabric/peer# peer chaincode query -c ecattlechannel -n animalchaincode -c '{"Args":["queryAnimal", "78"]}'
{"animalId": "78", "tipo": "corte", "idade": "60", "sexo": "F", "raca": "Nelore", "data nascimento": "2024-03-08", "peso atual": "200", "fazenda origem": "27", "fazenda atual": "27", "tag": "51", "hash_id": "3851864862c3a7cd915815817f36027830eed5f6d801c7982c41838e8e90"}
root@5ff6fc9a4f4:/opt/gopath/src/github.com/hyperledger/fabric/peer# peer chaincode query -c ecattlechannel -n farmchaincode -c '{"Args":["queryFarma", "27"]}'
{"farmaId": "27", "nome": "Chapadão e Fundão", "localizacao": "Bambuí", "proprietario_id": "14", "hash_id": "8702cbce4e8e97027c446f70542a1902ba820abc5550909e6f57c734b41a7"}
root@5ff6fc9a4f4:/opt/gopath/src/github.com/hyperledger/fabric/peer# peer chaincode query -c ecattlechannel -n ownerchaincode -c '{"Args":["queryOwner", "14"]}'
{"tipo": "proprietario", "proprietarioId": "14", "assincaesId": "08410099a-08-60", "nome": "Hilton Junior", "endereco": "Rua Manoel Bento", "telefone": "37999495937", "cpf": "69940518676", "email": "hiltonraja@gmail.com", "hash_id": "4173cd978ae514add937730e4f2345e55e8b2923094210b1195ec5341d57"}
root@5ff6fc9a4f4:/opt/gopath/src/github.com/hyperledger/fabric/peer# peer chaincode query -c ecattlechannel -n transactionchaincode -c '{"Args":["queryTransaction", "18"]}'
{"orderId": "14", "vendedorId": "14", "compradorId": "14", "data": "2024-02-03T10:00", "valor": "6000", "docType": "transaction", "hash_id": "ecb20b30e1ecce3b0e9c2b44c2ae31917ad115c3917a3d76d78dabb2448c"}
root@5ff6fc9a4f4:/opt/gopath/src/github.com/hyperledger/fabric/peer#
```

Figura 23 – Consultas ao *ledger*. Fonte: Autor.

Tabela 4 – *Docker Containers* em Execução na Rede Blockchain

Containers ID	Imagens	Nomes
63e2fb6bb453	dev-peer0.ecattleorg.ecattle.com-farmchaincode_2.0.0	dev-peer0.ecattleorg.ecattle.com-farmchaincode_2.0.0
a48c9a865d8e1	dev-peer0.ecattleorg.ecattle.com-ownerchaincode_2.0.0	dev-peer0.ecattleorg.ecattle.com-ownerchaincode_2.0.0
742a9c0e52a	dev-peer0.ecattleorg.ecattle.com-ownerchaincode_2.0.0	dev-peer0.ecattleorg.ecattle.com-ownerchaincode_2.0.0
7da38862fb9de	dev-peer0.ecattleorg.ecattle.com-ownerchaincode_2.0.0	dev-peer0.ecattleorg.ecattle.com-ownerchaincode_2.0.0
8eb9b0bc127	dev-peer0.org2.ecattle.com-farmchaincode_2.0.0	dev-peer0.org2.ecattle.com-farmchaincode_2.0.0 "
91ee845898a9	dev-peer0.org2.ecattle.com-farmchaincode_2.0.0	dev-peer0.org2.ecattle.com-farmchaincode_2.0.0
82d5cfbf	dev-peer0.ecattleorg.ecattle.com-animalchaincode_2.0.0	dev-peer0.ecattleorg.ecattle.com-animalchaincode_2.0.0
41c5f1ff44	dev-peer0.ecattleorg.ecattle.com-animalchaincode_2.0.0	dev-peer0.ecattleorg.ecattle.com-animalchaincode_2.0.0
841c0dc19b7b4	dev-peer0.ecattleorg.ecattle.com-transactionchaincode_2.0.0	dev-peer0.ecattleorg.ecattle.com-transactionchaincode_2.0.0
c4fe9ed05d1ff144	dev-peer0.ecattleorg.ecattle.com-transactionchaincode_2.0.0	dev-peer0.ecattleorg.ecattle.com-transactionchaincode_2.0.0
841c0dc19b7b4	dev-peer0.org2.ecattle.com-transactionchaincode_2.0.0	dev-peer0.org2.ecattle.com-transactionchaincode_2.0.0
ce9ed05d1ff144	dev-peer0.org2.ecattle.com-transactionchaincode_2.0.0	dev-peer0.org2.ecattle.com-transactionchaincode_2.0.0
33a5379de7d3	hyperledger/fabric-tools:latest	cli
17d4474e	hyperledger/fabric-peer:latest	peer0.ecattleorg.ecattle.com
6adbc19a3d9	hyperledger/fabric-peer:latest	peer0.org2.ecattle.com
8bd430eb2e3	hyperledger/fabric-orderer:latest	orderer.ecattle.com

5.2.2 Integração entre a plataforma web e a rede Blockchain

A integração entre a plataforma web e a rede Blockchain foi concebida para combinar a versatilidade e acessibilidade de um sistema web com a segurança e imutabilidade

do Blockchain. Essa integração permite que os usuários do sistema beneficiem-se de um registro transparente e inalterável de transações e dados relacionados ao gado, garantindo maior confiança e eficiência nas operações.

A integração foi planejada para ser bidirecional. O sistema web se comunica com a rede Blockchain baseada no *Hyperledger Fabric* através de uma API dedicada. Isso assegura que qualquer atividade ou transação registrada no sistema web seja simultaneamente gravada na Blockchain, proporcionando uma camada adicional de verificação e segurança.

Os benefícios dessa integração são múltiplos. Primeiramente, a segurança é significativamente reforçada. A natureza descentralizada do Blockchain garante que os dados sejam à prova de adulteração, o que é crucial para a integridade das informações. Em termos de transparência, todas as transações são registradas de forma aberta e acessível, permitindo auditorias mais eficientes e uma rastreabilidade superior. Além disso, a integração reduz a possibilidade de inconsistências de dados, já que as informações permanecem atualizadas e sincronizadas entre o sistema web e a rede Blockchain. Outro benefício notável é a automação de processos; com a utilização de Contratos Inteligentes, diversas atividades e decisões podem ser automatizadas com base em critérios predefinidos, otimizando operações e economizando tempo.

No entanto, a integração com a rede Blockchain também apresenta desafios e considerações importantes. A latência é um fator a ser levado em conta, pois o processo de consenso do Blockchain pode introduzir pequenos atrasos durante o registro de transações. A manutenção é outro aspecto crucial, pois é necessário garantir que as atualizações do sistema web e da rede Blockchain sejam compatíveis, exigindo um monitoramento contínuo. Finalmente, os custos associados à manutenção de uma rede Blockchain, embora justificados pelos benefícios, devem ser avaliados e gerenciados.

Em resumo, a integração do sistema eCattle com a rede Blockchain oferece uma solução para a gestão agropecuária, combinando segurança, transparência e eficiência operacional.

5.3 Considerações Finais

Ao longo deste capítulo, foi apresentado a estrutura da rede Blockchain desenvolvida para rastrear o gado usando o *Hyperledger Fabric*. A natureza detalhada da arquitetura, aliada à robustez dos contratos inteligentes, exemplifica o compromisso em fornecer uma solução eficaz e confiável para o setor agropecuário. A integração de componentes técnicos até a definição de regras de negócio, ressalta a importância da precisão e confiabilidade em um sistema desse porte.

O *Hyperledger Fabric*, como uma plataforma de Blockchain empresarial, provou ser

uma escolha acertada, oferecendo segurança, transparência e rastreabilidade, elementos essenciais para um sistema de rastreamento eficaz. Conforme o avanço, fica evidente que o sucesso do sistema web não se baseia apenas em tecnologia de ponta, mas também na cuidadosa consideração de práticas e demandas do setor agropecuário, culminando em uma solução que verdadeiramente serve seu propósito e seus usuários.

A integração de contratos inteligentes, políticas e certificados fornece uma base sólida para uma gestão eficiente de gado, oferecendo uma solução de rastreamento confiável e facilmente auditável.

No Capítulo 6, será apresentada uma avaliação detalhada do sistema eCattle, onde serão discutidos os resultados obtidos através de testes de usabilidade, eficácia e aceitação do sistema entre os profissionais do setor agropecuário, proporcionando *insights* para futuras melhorias e validação do sistema proposto.

6 Avaliação do Sistema eCattle

6.1 Considerações Iniciais

O processo de avaliação é fundamental na abordagem *Design Science Research* (DSR), pois permite testar, analisar e validar artefatos desenvolvidos para resolver problemas específicos. A avaliação é parte crucial para garantir que os artefatos sejam úteis, eficazes e aceitos pelos usuários finais [83]. No caso do sistema eCattle, a avaliação foi projetada para medir sua usabilidade, eficácia e aceitação entre profissionais do setor agropecuário, como veterinários, técnicos agrícolas, agrônomos e produtores rurais com conhecimentos em computação.

A metodologia DSR enfatiza a criação de artefatos para resolver problemas do mundo real, ao mesmo tempo em que contribui para o conhecimento científico [84]. Ela envolve um ciclo de construção, avaliação e *feedback* para aprimorar os artefatos ao longo do tempo [85]. Portanto, o objetivo da avaliação foi entender como o sistema eCattle se comporta em cenários práticos e identificar áreas de melhoria para aumentar sua aceitação e usabilidade.

Para avaliar o sistema eCattle, foi utilizada uma escala de *Likert* para medir a percepção dos usuários em relação à usabilidade, eficácia e aceitação do sistema. A escala de *Likert* é uma ferramenta amplamente usada em pesquisas para medir opiniões e atitudes, permitindo aos respondentes indicar seu grau de concordância ou discordância em uma escala de vários pontos [86]. Essa abordagem fornece dados quantitativos para análise, facilitando a interpretação dos resultados e a identificação de tendências.

O uso de uma escala de *Likert* na avaliação do sistema eCattle permitiu uma análise detalhada da percepção dos usuários. Os resultados obtidos com essa ferramenta forneceram *insights* que orientaram melhorias e adaptações do sistema [87].

Com base nos resultados obtidos através da avaliação do sistema eCattle, foi realizada a implementação de melhorias e testes adicionais para garantir que o sistema atendesse as necessidades dos profissionais do setor.

6.2 Metodologia de Avaliação do Sistema eCattle

A pesquisa seguiu o ciclo de avaliação descrito por Peffers et al. [85], que inclui as etapas de problematização, construção do artefato, avaliação e comunicação dos resultados. Para avaliar o sistema eCattle, foram planejados questionários para coleta de dados. O *design* da pesquisa foi concebido para fornecer *insights* sobre a usabilidade e eficácia

do sistema em um ambiente real, além de identificar possíveis áreas de melhoria.

Os participantes da avaliação foram selecionados de forma a representar um amplo espectro de profissionais do setor agropecuário. O grupo composto por cinco pessoas incluiu veterinários, técnicos agrícolas, agrônomos e produtores rurais com conhecimentos em computação. A escolha desses perfis foi baseada na premissa de que eles representariam os principais usuários do sistema eCattle e poderiam fornecer *feedback* relevante para aprimorar o artefato [84].

Para a coleta de dados, foram utilizados questionários estruturados, que permitiram aos participantes expressar suas opiniões de maneira flexível, ao mesmo tempo em que forneceram dados quantificáveis [88]. Na Apêndice A, é apresentado o roteiro de avaliação para o sistema eCattle, onde é possível verificar as questões utilizadas além de um resumo compilado para cada uma delas.

Os questionários estruturados incluíram perguntas baseadas em uma escala de 5 variáveis, permitindo medir a percepção dos usuários em relação à usabilidade e eficácia do sistema. A escala de Likert é uma ferramenta valiosa para medir atitudes e percepções, oferecendo uma maneira simples e eficaz de obter dados quantitativos para análise [86]. A aplicação da escala de Likert na metodologia de avaliação permitiu uma análise detalhada da percepção dos usuários e facilitou a identificação de padrões e tendências [87]. Para seleção por parte dos usuários, pôde-se escolher entre as seguintes alternativas, considerando uma escala de 1 à 5:

- Discordo totalmente. (1)
- Discordo. (2)
- Neutro. (3)
- Concordo. (4)
- Concordo totalmente. (5)

6.3 Discussão da Amostra

A escolha da amostragem de apenas cinco usuários foi motivada por vários fatores, incluindo a limitação de tempo e a dificuldade em encontrar usuários com o perfil específico necessário para testar o sistema. O ramo da pecuária no Brasil apresenta particularidades, como por exemplo falta de utilização de tecnologias digitais e pouca instrução tecnológica por parte dos produtores rurais, o que dificultam ainda mais a obtenção de participantes adequados para a avaliação de novas tecnologias. A tecnologia da informação ainda é pouco aplicada no setor agropecuário brasileiro, resultando em baixa acessibilidade a

meios de comunicação digitais e pouca familiaridade com sistemas tecnológicos. Esta situação é reflexo de uma cultura local que ainda está em processo de adaptação às novas ferramentas e tecnologias digitais.

Adicionalmente, é importante considerar que o setor pecuário no Brasil apresenta pouca tecnologia aplicada e, conseqüentemente, há uma baixa acessibilidade a meios de comunicação e sistemas informatizados por parte dos produtores rurais, que geralmente não têm familiaridade com o uso de tecnologias avançadas. Esta realidade aumenta a dificuldade de se encontrar um número maior de usuários qualificados para testar o sistema em um período de tempo limitado.

Os perfis dos usuários foram criteriosamente selecionados para garantir uma representatividade adequada do setor. Entre eles estavam um engenheiro da computação mestre em engenharia elétrica, uma licenciada em letras, um administrador mestre em sustentabilidade ambiental, uma veterinária e um veterinário. Destes, três são produtores de gado de corte (engenheiro da computação, licenciada em letras e um veterinário) e um é produtor de gado de leite (um veterinário). Além disso, dois deles são funcionários do Instituto Mineiro de Agropecuária (IMA), proporcionando uma perspectiva prática e regulatória valiosa para a avaliação do sistema. Um deles, apesar de não ser produtor, atua diretamente com registros e fiscalizações de animais, leilões de gado, entre outras atividades (administrador).

A diversidade de perfis profissionais dos usuários garantiu que o *feedback* coletado abordasse diferentes aspectos e necessidades do setor agropecuário. A inclusão de funcionários do IMA, por exemplo, trouxe uma visão importante sobre a conformidade regulatória e os desafios operacionais enfrentados no dia a dia. Já a inclusão de um profissional da área de computação, proporcionou uma análise mais crítica à nível de sistema. Os veterinários representam também uma parcela importante responsável pela vivência diária no setor pecuário.

A amostra, apesar de pequena, foi cuidadosamente selecionada para garantir a representatividade de diferentes perfis profissionais do setor agropecuário. A diversidade de perfis garante que o *feedback* coletado aborde diferentes aspectos e necessidades desse setor.

6.4 Avaliação do Sistema eCattle

A execução da avaliação do sistema eCattle seguiu uma abordagem detalhada e orientada à metodologia Design Science Research (DSR), envolvendo a participação ativa dos usuários que testaram o sistema e um plano estruturado para coleta de dados. Para garantir a compreensão do sistema, foi criado um vídeo tutorial de dez minutos que explica em detalhes o propósito do projeto e como utilizar o sistema eCattle. Além disso,

o vídeo deixou claro o objetivo do sistema eCattle ao integrar a aplicação web com a rede Blockchain, expondo os benefícios desta tecnologia. O vídeo foi enviado por e-mail para os usuários potenciais, permitindo-lhes assistir ao conteúdo conforme sua conveniência. O vídeo pode ser visualizado através do *Link* ¹.

Após o envio do vídeo tutorial, foi solicitado aos usuários que fornecessem seu *feedback*, preenchendo um formulário do Google Forms especialmente criado para esta avaliação. O uso do Google Forms permitiu uma coleta de dados simplificada e facilitou a análise posterior, alinhando-se com a abordagem DSR, que busca utilizar ferramentas eficientes para coleta de dados [85].

Os participantes foram encorajados a utilizar o sistema eCattle em seus próprios ambientes de trabalho, proporcionando um cenário mais realista para avaliação. A duração das interações com o sistema variou de acordo com a disponibilidade de cada participante, com o objetivo de simular o uso prático no setor agropecuário, sendo em média quinze dias. A abordagem flexível para o ambiente e a duração das interações está alinhada com as recomendações da DSR, que enfatiza a avaliação do artefato em cenários reais para obter *feedback* genuíno dos usuários [84].

Os dados coletados incluíram as respostas ao questionário enviado por e-mail, que utilizou uma escala de Likert para avaliar a usabilidade e eficácia do sistema, seguindo uma abordagem quantitativa para análise de dados [86]. As respostas fornecidas pelos usuários permitiram a quantificação do grau de satisfação com diversos aspectos do sistema eCattle.

Os usuários demonstraram uma compreensão clara do propósito e funcionamento do sistema, com a maioria expressando um alto grau de satisfação quanto à clareza das funcionalidades e dados apresentados. Conforme pode ser observado na Figura 24, a facilidade de uso inicial e a intuitividade do sistema foram em média bem avaliadas, com a maioria dos usuários não encontrando dificuldades significativas de uso ou navegação.

Os dados coletados dos questionários e das notas do formulário foram organizados e submetidos a análises estatísticas detalhadas, incluindo cálculo de médias, desvios-padrão e Teste T, para quantificar o grau de satisfação dos usuários com diversos aspectos do sistema [86, 87, 89, 90]. Estas análises ajudaram a estabelecer uma compreensão clara da aceitação geral do sistema pelos usuários.

A Figura 24 ilustra a avaliação média de diferentes aspectos do sistema eCattle, destacando a percepção dos usuários em relação a várias funcionalidades e experiências de uso. Os dados coletados, representados através de barras horizontais em azul, demonstram a pontuação média atribuída a cada quesito avaliado, com um indicativo de variabilidade por meio dos desvios padrão, representados pelas linhas horizontais nas extremidades das barras.

¹ Apresentação do sistema eCattle e tutorial de utilização - <https://youtu.be/dTrtGZd-_3o>

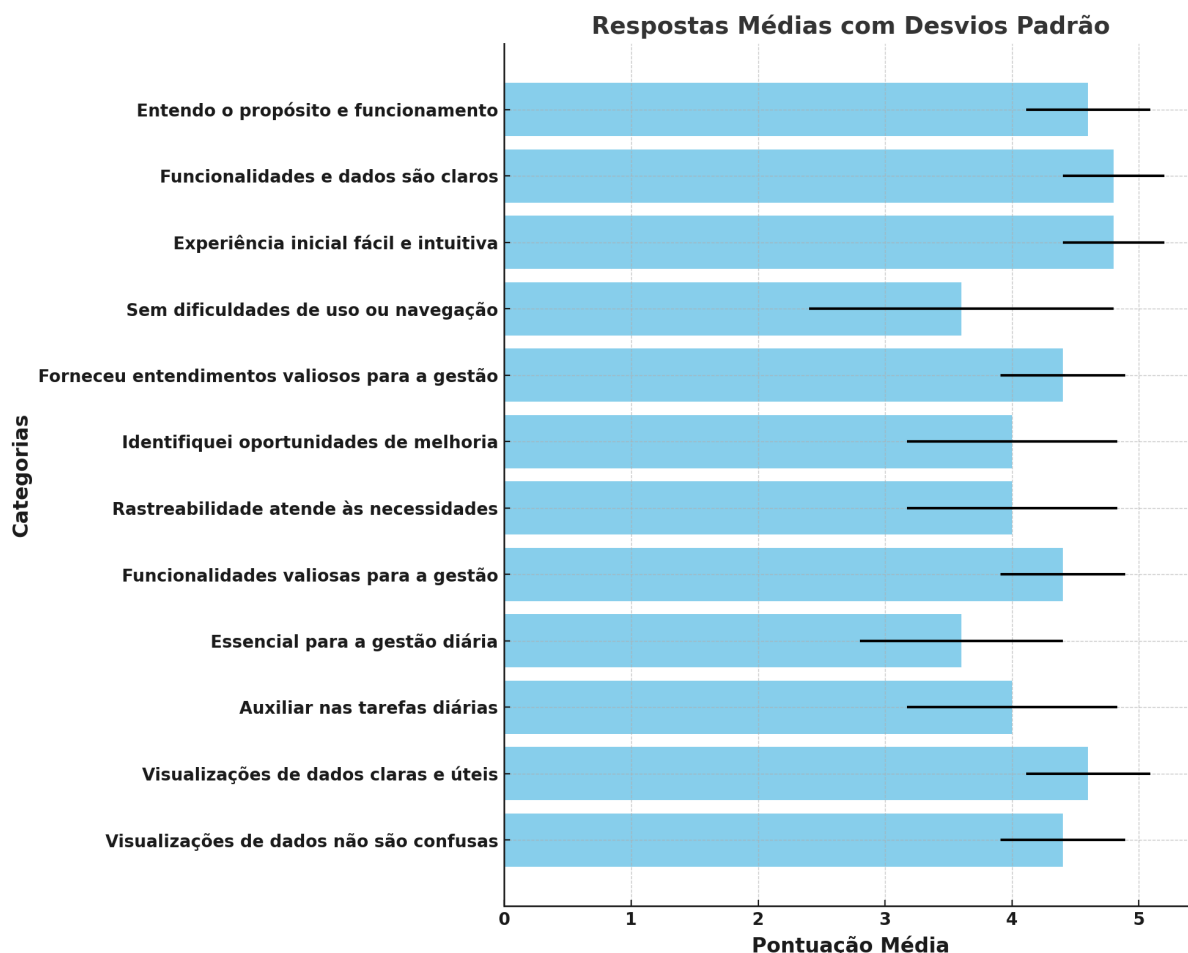


Figura 24 – Médias e Desvios-Padrão dos Dados Coletados. Fonte: Autor.

A Figura 25 ilustra os resultados do Teste T para comparar as médias das diferentes categorias de avaliação do sistema eCattle. Este teste foi escolhido devido à sua capacidade de avaliar se a média das respostas dos usuários é significativamente diferente de um valor neutro (neste caso, 3 na escala de Likert), ajudando a determinar a percepção geral dos usuários sobre o sistema.

O Teste T é uma técnica estatística amplamente utilizada para determinar se há uma diferença significativa entre a média de um grupo e um valor específico [91]. No contexto da avaliação do sistema eCattle, o Teste T foi utilizado para comparar a média das respostas dos usuários com o valor neutro de 3, considerado o ponto médio da escala de Likert utilizada na pesquisa.

A escolha do Teste T se justifica por algumas razões:

- **Simplicidade e Eficácia:** O Teste T é direto e eficaz para testar a diferença entre a média de um grupo e um valor específico [89].
- **Adequação à Escala de Likert:** A escala de Likert é ordinal e, quando a distribuição dos dados se aproxima de uma distribuição normal (ou seja, a maioria dos dados se

concentra em torno da média, com uma simetria aproximada em ambos os lados), o Teste T é apropriado para análises comparativas [87].

- Pequenas Amostras: O Teste T pode ser utilizado com pequenas amostras, desde que as premissas de normalidade sejam atendidas [90].
- Objetividade: Fornece uma medida clara e objetiva da diferença entre a percepção dos usuários e o valor neutro [90].
- Sensibilidade: É sensível o suficiente para detectar pequenas diferenças nas médias das respostas [90].
- Aplicabilidade: Adequado para dados coletados em uma escala de Likert, comum em pesquisas de satisfação e avaliação [87].

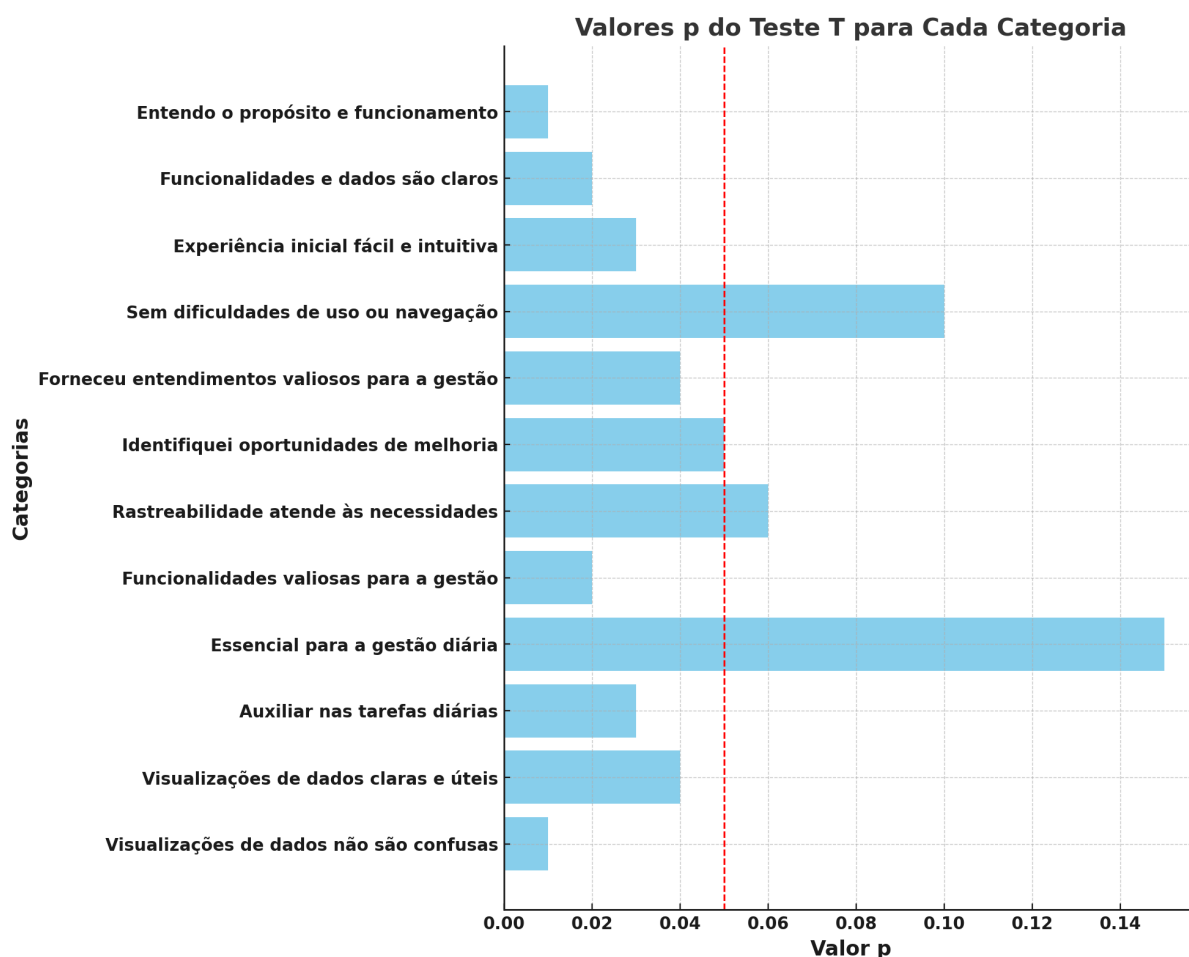


Figura 25 – Resultados do Teste T. Fonte: Autor.

Para calcular os p-valores apresentados no gráfico do Teste T para cada categoria de avaliação, o procedimento envolveu comparar a média das respostas de cada categoria com um valor específico, neste caso, 3, considerado um ponto neutro na escala de avaliação. O processo detalhado foi o seguinte:

1. Formulação das Hipóteses:

- Hipótese nula (H_0): A média das respostas de uma categoria é igual a 3 (média neutra).
- Hipótese alternativa (H_1): A média das respostas de uma categoria é diferente de 3.

2. Calcular a estatística T utilizando a fórmula:

$$T = \frac{\bar{X} - \mu}{\frac{s}{\sqrt{n}}}$$

onde:

- \bar{X} é a média das respostas da categoria.
- μ é o valor com o qual se está comparando a média (neste caso, 3).
- s é o desvio padrão das respostas da categoria.
- n é o número de observações na categoria.

3. Determinação do p-valor:

O p-valor foi então determinado como a probabilidade de obter um valor da estatística T que seja tão ou mais distante de zero quanto o valor observado, assumindo que a hipótese nula seja verdadeira. Em outras palavras, o p-valor mede a probabilidade de observar os dados ou algo mais extremo, dado que a hipótese nula é verdadeira.

6.4.1 Interpretação dos Resultados

A avaliação revelou que as categorias “Funcionalidades e dados são claros”, “Experiência inicial fácil e intuitiva”, e “Auxiliar nas tarefas diárias” receberam as maiores pontuações médias, todas acima de 4,5 em uma escala até 5, indicando uma alta satisfação dos usuários com esses aspectos. Isso sugere que o sistema é percebido como claro e intuitivo, facilitando a adoção e a continuidade de seu uso no cotidiano.

Por outro lado, o quesito “Sem dificuldades de uso ou navegação” apresentou uma pontuação média mais baixa, com maior dispersão nos dados, o que indica uma variabilidade maior nas respostas. Este resultado evidenciou as sugestões para melhorias, sendo necessárias para garantir uma experiência de uso mais uniforme entre os usuários.

A categoria “Essencial para a gestão diária” também apresentou uma pontuação mediana e maior variabilidade, destacando que, embora alguns usuários considerem o sistema crucial para suas atividades diárias, outros podem não perceber a mesma essencialidade.

No Teste T, os p-valores calculados para cada categoria indicaram a probabilidade de que as médias observadas fossem devidas ao acaso. Um p-valor menor que 0,05 indica que há uma diferença estatisticamente significativa entre a média das respostas da categoria e o valor neutro (3), significando que é improvável que a diferença observada seja devida ao acaso, ou seja, é improvável que a diferença seja resultado de variabilidade aleatória nos dados. No contexto do gráfico fornecido, p-valores pequenos indicam que a média das respostas é significativamente diferente de 3, sugerindo concordância ou discordância significativa dos usuários com as declarações afirmativas avaliadas. Por outro lado, p-valores altos indicam que não há diferença significativa entre a média das respostas e o ponto neutro, sugerindo variabilidade nas respostas dos usuários.

Esses p-valores foram críticos para avaliar a aceitação e eficácia do sistema eCattle. Eles ajudaram a identificar quais aspectos do sistema são bem recebidos pelos usuários e quais podem necessitar de melhorias. Ao comparar as médias das respostas com um valor neutro, os p-valores forneceram uma medida objetiva de como os usuários perceberam diferentes funcionalidades e a usabilidade do sistema.

Os gráficos não apenas refletem a eficácia e a aceitação do sistema nas tarefas cotidianas dos usuários, mas também destacam áreas para desenvolvimento contínuo e aprimoramento, conforme percebido na variação das pontuações. Estas visualizações fornecem uma ferramenta valiosa para entender melhor as necessidades dos usuários e orientar as futuras melhorias no sistema.

Avaliando a qualidade do *feedback* recebido e a consistência nas respostas, foi possível validar o sistema eCattle de maneira satisfatória. A análise dos resultados indicou pouca variação na maioria das respostas, sugerindo que, apesar do número reduzido da amostra, as percepções dos usuários foram consistentes. Este fato reforça a validade dos resultados obtidos e demonstra que a amostra utilizada, embora pequena, foi considerada suficiente para fornecer *insights* sobre a eficácia e a usabilidade do sistema eCattle.

6.5 Análise de Comportamento do Sistema eCattle

Nesta seção são apresentados testes realizados para avaliar do sistema eCattle em diferentes cenários de carga, focando em latência, uso de *Central Processing Unit* (CPU) e memória *Random-Access Memory* (RAM). O objetivo desses testes consiste em entender como o sistema se comporta sob condições reais de operação e simulações de alta demanda. Foi avaliado o sistema eCattle em situações de baixa carga, utilizando testes simples de transações sequenciais, e em condições de carga pesada, com múltiplas requisições simultâneas.

Essa análise é importante para garantir que o sistema eCattle seja capaz de atender às demandas de usuários finais de forma eficiente, sem comprometer a integridade da rede

Blockchain. Através desses testes, buscou-se identificar possíveis gargalos e limitações que possam impactar a capacidade do sistema de lidar com transações em escala e determinar áreas onde melhorias no código e na infraestrutura são necessárias.

6.5.1 Configuração de Hardware e Virtualização

O sistema eCattle foi configurado em uma *Virtual Machine* (VM) hospedada em um ambiente VMware 6.5. A VM utiliza o sistema operacional Debian 11 e foi configurada com recursos específicos para suportar a execução dos serviços, bem como os nós da rede Blockchain. A rede utilizada foi Gigabit Ethernet, fornecendo uma conexão de alta velocidade para a comunicação entre os nós da Blockchain. A configuração geral do sistema eCattle é apresentada na Tabela 5.

Tabela 5 – Configuração Geral da VM e Sistema Blockchain

Parâmetro	Especificação
Plataforma de Virtualização	VMware 6.5
Sistema Operacional	Debian 11
Nós Docker	3
Função dos Nós Docker	Executar a rede Blockchain

Para realizar experimentos de execução do sistema, a fim de analisar variáveis como latência, uso de memória RAM e taxa de processamento de CPU, foram considerados diferentes fatores e níveis de recursos para suportar a execução dos serviços, bem como os nós da rede Blockchain. Foram consideradas variações na quantidade de núcleos virtuais (vCPUs) e de memória RAM. A configuração detalhada dos recursos alocados é apresentada na Tabela 6.

Tabela 6 – Fatores e níveis para os experimentos

Fatores	Níveis
Memória RAM	4 GB e 15 GB
vCPUs	1 e 6
Armazenamento Físico	250 GB
Velocidade do Disco	7,2K Rotações por Minuto (RPM)
Nós Docker	3

Dentro desta VM são executados três nós *Docker*² principais (*orderer.ecattle.com*, *peer0.ecattleorg.ecattle.com* e *peer0.org2.ecattle.com*) que rodam a rede Blockchain. Estes nós são virtualizados e devem ser considerados nas análises de latência do sistema, uma vez que introduzem uma camada adicional de virtualização que pode impactar o desempenho

² <https://www.docker.com/>

geral. A virtualização em ambientes de nuvem pode levar a uma sobrecarga considerável, especialmente no caso de VMs compartilhando recursos físicos limitados [92].

O uso de virtualização, tanto a nível de VM quanto de contêineres *Docker*, é amplamente adotado em sistemas modernos devido à sua flexibilidade e escalabilidade. No entanto, essa abordagem pode causar uma sobrecarga de virtualização que impacta o desempenho geral do sistema, principalmente em termos de latência [93]. Essa sobrecarga ocorre devido ao gerenciamento adicional de recursos e à emulação de hardware, o que gera atrasos adicionais em operações de entrada/saída, CPU e memória [94, 95].

Especificamente, a execução de múltiplos nós *Docker* dentro de uma VM pode aumentar a latência devido ao compartilhamento de recursos físicos entre contêineres e a VM subjacente. Cada nó da rede Blockchain rodando dentro de contêineres é, essencialmente, uma entidade isolada que compete por recursos com outros processos, incluindo o próprio sistema operacional e outros contêineres. Isso pode resultar em aumentos na latência de processamento e transmissão de dados, uma vez que as requisições passam por camadas adicionais de gerenciamento de recursos.

Além disso, a velocidade do disco de 7,2K RPM também pode limitar o desempenho de entrada/saída, o que pode ser particularmente notável em cenários de alta concorrência, onde múltiplas transações estão tentando ser processadas simultaneamente. Embora os discos de 7,2K RPM sejam suficientes para muitas operações, eles podem não ser capazes de acompanhar a alta demanda de transações em uma rede Blockchain, onde o desempenho de leitura e escrita é crítico para o processamento eficiente de transações.

6.5.2 Teste com *Script* Personalizado

O objetivo deste experimento é avaliar a latência do sistema eCattle para uma determinada quantidade de transações executadas de forma sequencial. O método utilizado envolve o uso de um *script* PHP personalizado que envia requisições ao sistema e mede o tempo de resposta de cada transação individualmente. O foco do teste é medir a latência de forma detalhada e específica para cada requisição, apresentando o tempo de resposta de cada transação, o que auxilia na identificação de possíveis problemas em operações específicas. A execução do teste foi realizada manualmente, envolvendo um total de 10 simulações, ambas com 1000 transações para cadastro de animais, onde cada transação foi registrada e analisada individualmente.

Utilizar 10 simulações com 1000 transações cada para o teste foi uma escolha estratégica baseada na análise de estabilidade do sistema eCattle. Durante experimentos prévios, com 10 e 100 transações, o sistema demonstrou um comportamento estável, sem grandes variações no tempo de resposta. Com esses resultados preliminares, decidiu-se aumentar o volume de transações para 1000, a fim de confirmar a consistência do desem-

penho e garantir que o sistema pudesse lidar com um número maior de operações sem comprometer a latência ou a capacidade de processamento.

Ao alcançar 1000 transações para cada simulação, o sistema continuou a apresentar estabilidade, com tempos de resposta previsíveis e sem sinais de degradação de desempenho. Dado que o objetivo era validar o comportamento sob uma carga razoável e que o sistema se manteve estável nesse patamar, não houve necessidade de aumentar ainda mais a quantidade de transações. Testes com volumes maiores não seriam tão relevantes, uma vez que o sistema já demonstrou capacidade de lidar eficientemente com o número de transações simuladas, o que sugere que ele está preparado para operar de forma confiável dentro desse escopo sem sobrecargas significativas.

Os resultados do teste de desempenho com 10 simulações e 1000 transações enviadas em cada simulação, considerando uma máquina virtual com 15GB de RAM e 6 vCPUs, indicaram uma latência média de 2290,69 ms por transação, conforme representado no gráfico da Figura 26. Esse valor reflete o tempo necessário para que uma transação seja processada desde o envio através da plataforma web até a confirmação na rede Blockchain. A análise dessas simulações revelou uma variação mínima de latência entre as execuções, o que sugere que o sistema possui um comportamento previsível em termos de tempo de resposta.

Considerando o segundo experimento com uma máquina virtual com 4GB de RAM e 1 vCPU, mantendo 10 simulações com 1000 transações cada, pode-se observar um tempo de resposta maior, com latência média de 5391,12 ms por transação, conforme representado no gráfico da Figura 27. Neste caso, pode-se observar que a redução dos recursos computacionais alocados para a VM tem relação direta com o aumento no tempo gasto para atender as requisições.

6.5.3 Apache Benchmark (ab) para Testes de Carga

O objetivo deste teste é avaliar como o sistema eCattle lida com um grande número de requisições simultâneas, medindo a latência sob alta concorrência. O método utilizado envolve a ferramenta de *benchmarking* Apache Benchmark [96], que é capaz de enviar múltiplas requisições simultâneas ao sistema. O foco do teste é analisar a escalabilidade e a robustez do sistema, onde muitos usuários tentam acessar o sistema ao mesmo tempo. Ele mede a latência média, com ênfase em como o sistema se comporta ao ser submetido a uma quantidade específica de requisições para determinado *endpoint* na rede Blockchain. A execução é automatizada e capaz de enviar requisições simultâneas com diferentes níveis de concorrência, fornecendo um conjunto completo de métricas, incluindo tempo de conexão e tempo de processamento.

Os resultados do Apache Benchmark para os diferentes níveis são apresentados na

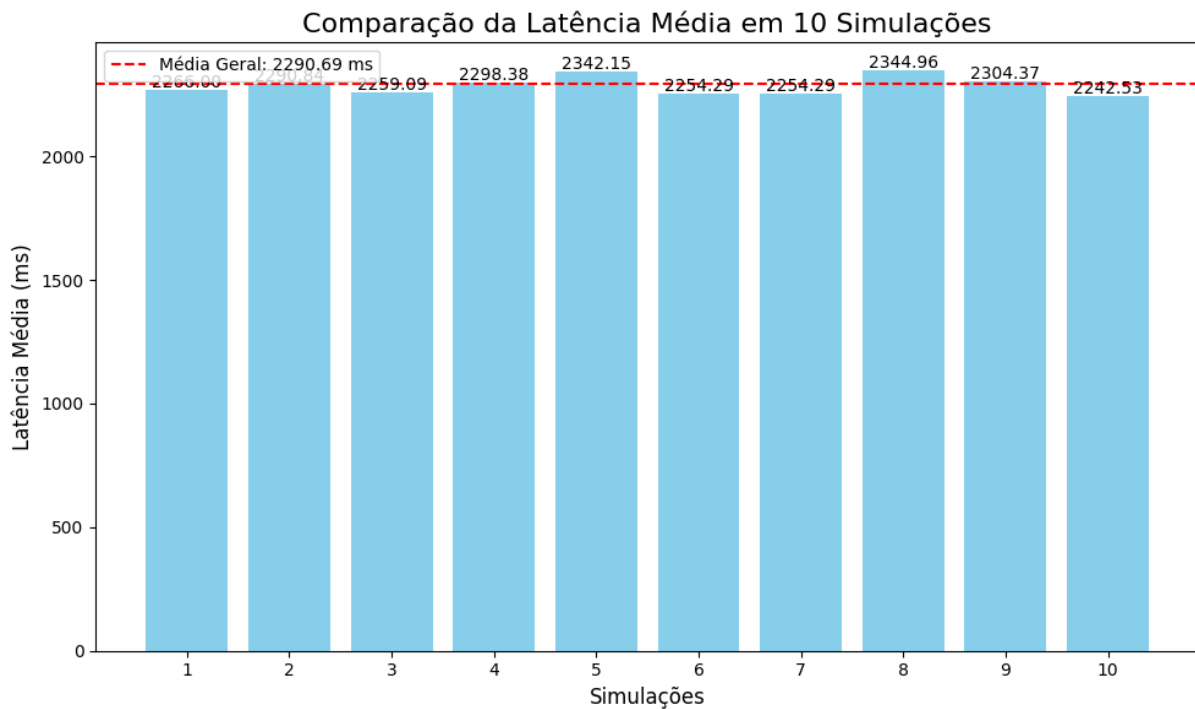


Figura 26 – Latência das Transações no Primeiro Experimento com 15GB de RAM e 6 vCPUs. Fonte: Autor.

Tabela 7.

Tabela 7 – Resultados do Apache Benchmark para os Níveis 1 e 2

Métrica	Nível 1	Nível 2
Total de Requisições Simultâneas	1000	1000
Requisições Concluídas com Sucesso	1000	1000
Latência Média	2267,68 ms	5361,50 ms
Tamanho Médio da Resposta	189 bytes	189 bytes
Tempo Total do Teste	2267,68 s	5361,50 s
Transferência Total	189.000 bytes	189.000 bytes
Tamanho Total do Corpo Enviado	188.000 bytes	188.000 bytes
Transferência de HTML	94.500 bytes	94.500 bytes
Nível de Concorrência	1000	1000

Os resultados da Tabela 7 mostram que, durante o teste de carga com 1000 requisições simultâneas, o sistema eCattle conseguiu processar todas as requisições com sucesso. Esses resultados indicam que o sistema é capaz de lidar com múltiplas requisições simultâneas.

6.5.4 Análise do Uso de Memória e CPU para Diferentes Configurações de Recursos

Nos experimentos conduzidos, foram analisados dois níveis de fatores para o desempenho do sistema eCattle: o primeiro nível utilizando 6 vCPUs e 15GB de RAM, e o

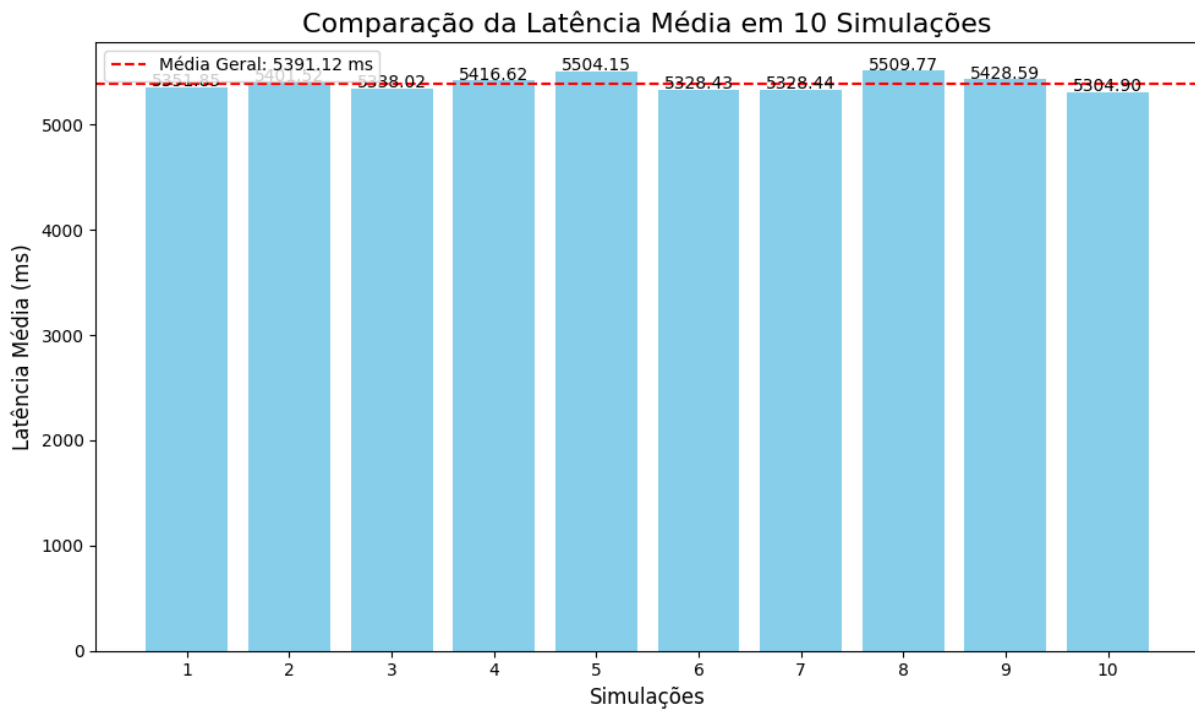


Figura 27 – Latência das Transações no Segundo Experimento com 4GB de RAM e 1 vCPU. Fonte: Autor.

segundo nível com 1 vCPU e 4GB de RAM. Para cada nível, foram coletados dados sobre o uso de CPU e memória ao longo do tempo durante a execução das simulações de carga e desempenho, utilizando a ferramenta *vmstat* [97] durante os testes. Essas medições são essenciais para avaliar como o sistema responde a diferentes alocações de recursos.

Na Figura 28, é apresentado o gráfico de uso de CPU correspondente às configurações dos níveis 1 e 2. Esse gráfico não apenas ilustra o comportamento médio de uso da CPU, mas também destaca a variação no uso, representada pelo desvio padrão, e a precisão das medições, indicada pelo intervalo de confiança de 95%. Essas informações fornecem uma visão detalhada de como o sistema utiliza os recursos de processamento ao longo do tempo, permitindo uma avaliação mais precisa da capacidade de resposta do sistema sob diferentes configurações de hardware.

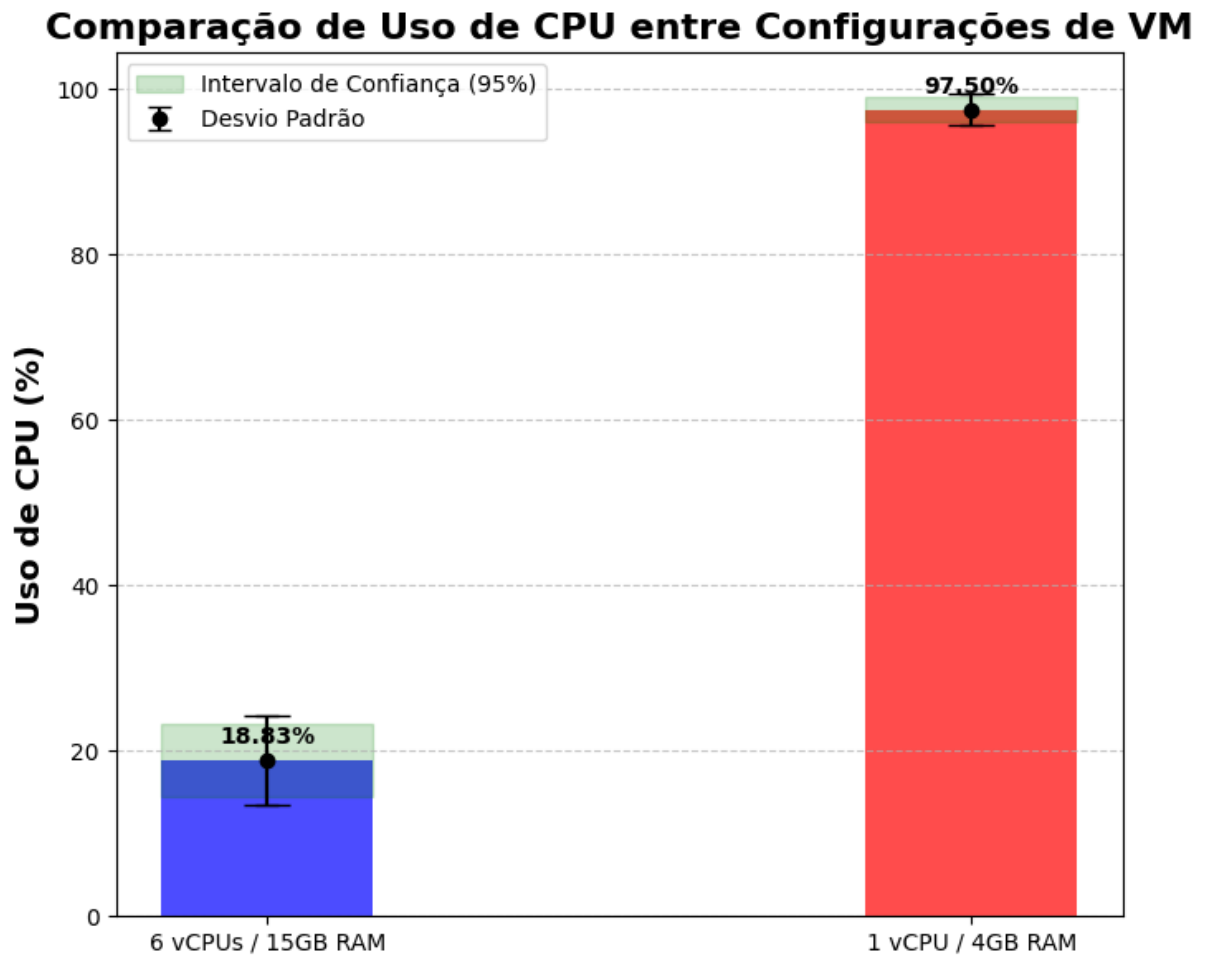


Figura 28 – Uso de CPU durante o Teste de Carga de Ambos Níveis. Fonte: Autor.

Já na Figura 29, são apresentados os gráficos de uso de memória correspondentes às configurações de 6 vCPUs e 15GB de RAM e 1 vCPU e 4GB de RAM. Assim como no gráfico de CPU, o gráfico de uso de memória também inclui as medidas de média, desvio padrão e intervalo de confiança, proporcionando uma melhor análise sobre a eficiência do uso de memória nas duas configurações.

Comparação de Uso de Memória entre Configurações de VM

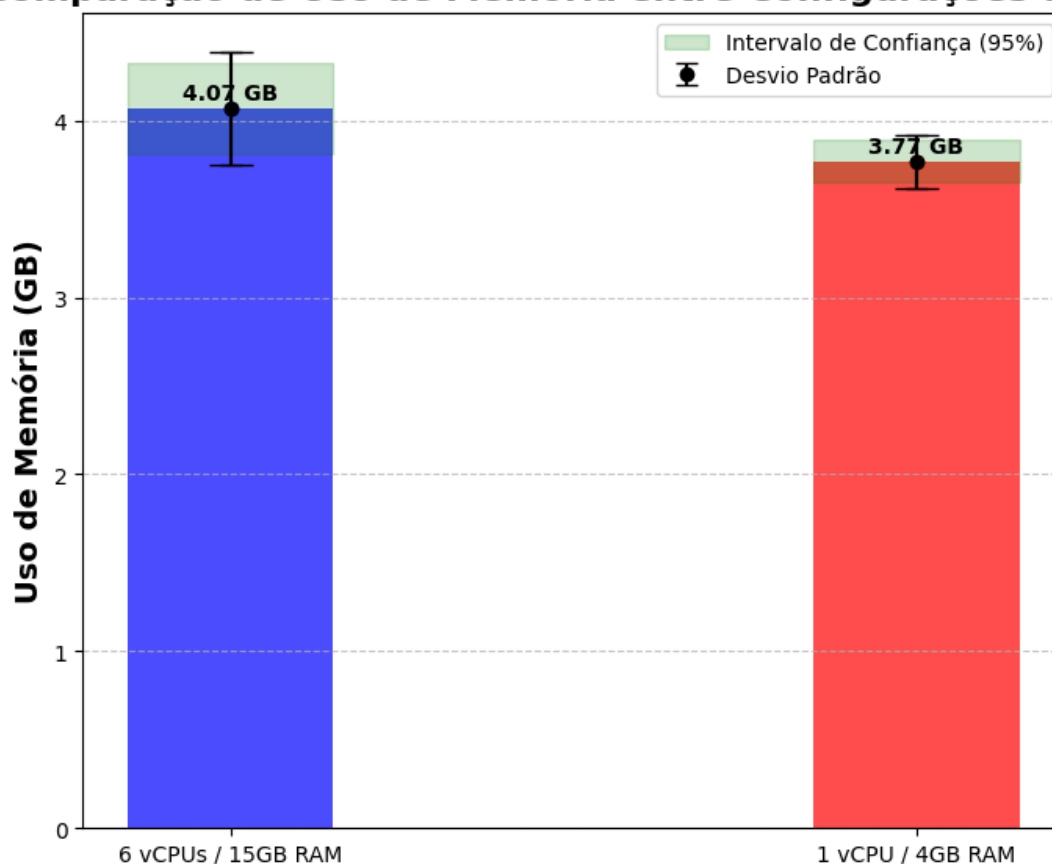


Figura 29 – Uso de Memória durante o Teste de Carga de Ambos Níveis. Fonte: Autor.

Os gráficos apresentam um comportamento distinto em termos de uso de recursos durante as simulações de carga com diferentes configurações de CPU e memória.

Para a simulação com 6 vCPUs e 15GB de RAM, observou-se que o uso da CPU permaneceu na média de 18,83%, com picos de alta utilização acima de 20%. A CPU livre variou entre 90% e 70%, indicando que o sistema ainda tinha margem de capacidade, sem nenhuma sobrecarga significativa de recursos observada. O desvio padrão e o intervalo de confiança mostraram que, apesar de algumas variações no uso de CPU, o sistema manteve-se estável dentro de uma faixa previsível de desempenho.

Já na simulação com 1 vCPU e 4GB de RAM, a CPU permaneceu próxima a 100% durante a maior parte do tempo, com uma média de utilização geral de 97,50%. Nesse cenário, a CPU livre foi praticamente inexistente, com picos de uso total em muitos momentos, sugerindo que o sistema estava sobrecarregado e sem capacidade adicional de processamento para lidar com mais transações ou operações. O pequeno desvio padrão e o estreito intervalo de confiança para esta configuração indicam que o sistema estava consistentemente operando próximo ao seu limite máximo, sem margem para acomodar cargas adicionais.

Em termos de memória, na configuração de 15GB de RAM, a memória utilizada

manteve-se em torno de 4GB durante boa parte da simulação, com a média geral de utilização em 4,07GB. Isso indica que o sistema ainda tinha uma quantidade considerável de memória disponível para suportar operações adicionais, sem esgotar os recursos de forma crítica. O desvio padrão e o intervalo de confiança nessa configuração foram baixos, sugerindo um uso estável e previsível da memória.

No entanto, na simulação com 4GB de RAM, a memória utilizada ficou na média de consumo de 3,77GB. A memória usada atingiu picos próximos de 4GB durante parte da simulação, o que indica que o sistema estava operando no limite de sua capacidade de memória, com pouca ou nenhuma margem para absorver mais operações. O desvio padrão foi relativamente pequeno, e o intervalo de confiança indicou que o sistema estava operando de maneira consistente próxima ao seu limite máximo de memória.

Esses resultados sugerem que, enquanto o sistema pode operar de forma estável com a configuração mais robusta (6 vCPUs e 15GB de RAM), ele enfrenta limitações significativas quando os recursos são reduzidos para 1vCPU e 4GB de RAM, especialmente em situações de alta demanda de processamento e memória. As medidas de média, desvio padrão e intervalo de confiança fornecem uma análise quantitativa adicional que corrobora essas conclusões, indicando que o sistema, na configuração mais restrita, opera no limite de suas capacidades com pouca margem para variações adicionais na carga de trabalho.

6.5.5 Interpretação dos Resultados

Nesta abordagem, buscou-se equilibrar o esforço computacional e a precisão dos resultados, analisando diferentes tipos de configuração computacional para verificar o desempenho do sistema. O ambiente foi configurado de forma isolada, com apenas as aplicações vitais em execução, garantindo que as medições refletissem o comportamento exclusivo do sistema eCattle sob diferentes cenários de carga.

Os resultados das simulações sugerem que, com a configuração mais robusta (6 vCPUs e 15GB de RAM), o sistema manteve uma boa estabilidade. A CPU teve uma utilização média de 20%, chegando a um pico de 30%, com uma margem confortável de capacidade disponível. O uso da memória permaneceu elevado, com uma quantidade significativa de memória livre (entre 9GB e 10GB), o que indica que o sistema ainda tinha muitos recursos disponíveis para atender a mais operações sem risco de sobrecarga.

Por outro lado, a simulação com 1 vCPU e 4GB de RAM apresentou resultados mais desafiadores. A CPU operou próxima de 100% durante a maior parte do tempo, sem quase nenhuma margem livre, o que sugere que o sistema estava sobrecarregado e enfrentava dificuldades para lidar com a demanda. Além disso, a memória livre ficou perto de zero durante a maior parte da simulação, com o sistema operando praticamente

no limite de sua capacidade de memória. A Figura 30 ilustra o comparativo das diferentes métricas e recursos computacionais analisadas entre os dois fatores de níveis.

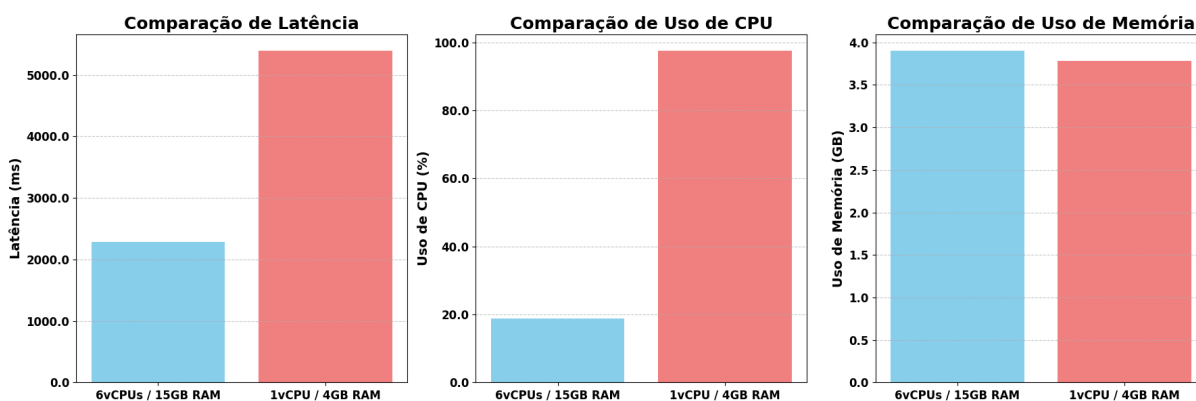


Figura 30 – Compara o Entre M tricas e Recursos Computacionais. Fonte: Autor.

Esses resultados indicam que, enquanto o sistema pode operar de forma est vel e eficiente em um ambiente com recursos mais elevados, ele apresenta limita es significativas em cen rios onde os recursos de CPU e mem ria s o restritos. A sobrecarga observada na configura o com 1 vCPU e 4GB de RAM destaca a necessidade de ajustes e otimiza es tanto no c digo quanto na infraestrutura subjacente.

Esses resultados tamb m refor am a import ncia de otimizar a infraestrutura para evitar gargalos de desempenho. Em estudos futuros, testes com um n mero maior de requisi es e simula es poder o fornecer uma avalia o mais detalhada do comportamento do sistema sob cargas ainda mais intensas. Al m disso, estrat gias de escalonamento din mico de recursos, tanto no n vel de CPU quanto de mem ria, podem ser cruciais para garantir que o sistema eCattle continue operando de maneira eficiente, mesmo em cen rios de alta demanda.

6.6 Implica es para Aprimoramento do Sistema eCattle

Os *insights* obtidos a partir dos testes de an lise de comportamento, juntamente com os *feedbacks* dos usu rios, forneceram dire es claras para o futuro desenvolvimento e aprimoramento do sistema eCattle. Em particular, a necessidade de melhorar a escalabilidade e reduzir a lat ncia das transa es ser  um foco principal nas pr ximas fases de desenvolvimento. Para alcan ar uma ado o mais ampla, especialmente em cen rios de grande escala, o sistema precisar  evoluir para suportar uma maior quantidade de transa es simult neas, sem comprometer a seguran a ou a integridade dos dados registrados na Blockchain.

Al m disso, os *feedbacks* dos usu rios indicaram a import ncia de continuar a aprimorar a interface do sistema e melhorar a tradu o de termos para aumentar a satisfa o

do usuário. Estes aspectos, alinhados às práticas recomendadas de *design* de interação, reforçam a necessidade de adaptar o sistema às linguagens e fluxos de trabalho locais, visando aumentar a adesão do usuário [98]. Também foi destacada a necessidade de um módulo que permita a alteração do idioma do sistema eCattle para uma versão em Inglês, o que garantirá que o sistema não seja restrito apenas ao público brasileiro, ampliando seu alcance global.

Adicionalmente, estudos anteriores que investigaram o desempenho de sistemas baseados em Blockchain também identificaram a latência como fatores críticos de desempenho [99, 94]. Nesses estudos, foi destacado que a virtualização e a sobrecarga do ambiente de execução podem introduzir atrasos significativos, impactando negativamente a capacidade do sistema de processar grandes volumes de transações em tempo real. É importante considerar a eficiência do gerenciamento de recursos, especialmente em ambientes virtualizados, onde o provisionamento dinâmico pode ajudar a mitigar os impactos da sobrecarga [93].

6.7 Considerações Finais

A avaliação do sistema eCattle, utilizando a metodologia *Design Science Research* (DSR), forneceu *insights* valiosos sobre a usabilidade, eficácia e aceitação do sistema entre profissionais do setor agropecuário. Através da aplicação de uma escala de *Likert* para medir as respostas dos usuários, foi possível quantificar a percepção dos usuários em relação a vários aspectos do sistema. Para tal, foram realizadas análises de médias, desvios-padrão e a técnica estatística do Teste T, permitindo uma compreensão mais aprofundada da aceitação do sistema eCattle.

Os *insights* obtidos a partir da análise das respostas do questionário indicam uma aceitação positiva do sistema eCattle. A maioria dos usuários reportou que o sistema é intuitivo e que as funcionalidades são adequadas às suas necessidades diárias, destacando a facilidade de uso e a eficiência nas operações cotidianas. Isso sugere que o sistema atende bem às expectativas dos usuários e é percebido como uma ferramenta útil e eficaz no gerenciamento de atividades agropecuárias.

No entanto, a análise de comportamento do sistema revelou áreas significativas para melhoria, especialmente em termos de escalabilidade e eficiência sob cargas intensas de trabalho. Os testes de comportamento, realizados com ferramentas como o Apache Benchmark, indicaram que, embora o sistema seja capaz de lidar com múltiplas requisições simultâneas, os valores de latência podem destacar a necessidade de otimizações adicionais na arquitetura do sistema. Essas melhorias são cruciais para garantir que o eCattle possa ser utilizado em cenários de larga escala, onde o volume de transações é elevado.

Este estudo contribui para o campo de sistemas de informação em agropecuária,

demonstrando a aplicabilidade e a importância da metodologia DSR na avaliação e no aprimoramento de sistemas tecnológicos. Através desta abordagem, foi possível alinhar o desenvolvimento do eCattle às necessidades reais dos usuários, resultando em um sistema que não apenas atende aos requisitos funcionais, mas também é bem aceito pelos seus usuários finais.

Ademais, o estudo destaca a importância de considerar as condições reais de uso durante o desenvolvimento de sistemas tecnológicos, especialmente em setores como a agropecuária, onde a robustez e a eficiência são cruciais [83]. A análise detalhada do desempenho, juntamente com o *feedback* contínuo dos usuários, fornece uma base sólida para o aprimoramento contínuo do sistema, assegurando que ele permaneça relevante e eficaz à medida que as necessidades dos usuários evoluem.

Finalmente, os resultados sugerem que o eCattle tem potencial para se expandir e se adaptar a um mercado global, desde que sejam implementadas melhorias que abordem as limitações identificadas. O alinhamento contínuo entre o desenvolvimento tecnológico e as necessidades dos usuários finais será fundamental para o sucesso e a longevidade do sistema no competitivo ambiente da agropecuária digital.

7 Conclusão

7.1 Considerações Finais

Esta pesquisa teve como objetivo principal desenvolver e implementar um sistema de gerenciamento de gado baseado na tecnologia Blockchain, visando melhorar a rastreabilidade e a segurança das informações na indústria pecuária. Os resultados alcançados demonstraram que o sistema não só atende a essas necessidades, mas também proporciona uma plataforma confiável e transparente para os produtores, gerando dados seguros e imutáveis que suportam a tomada de decisão.

A implementação deste sistema traz significativas contribuições para a indústria pecuária, introduzindo um modelo de transparência e eficiência que pode ser estendido para outras áreas. A capacidade de rastrear e validar cada etapa do ciclo de vida do gado oferece uma garantia adicional contra fraudes e erros, beneficiando não apenas os fazendeiros, mas também os reguladores e consumidores. Essa transparência é crucial para o aumento da confiança no mercado e pode potencialmente abrir novos mercados internacionais para produtos certificados.

Este projeto poderá servir como base para futuras pesquisas e aplicações em áreas relacionadas, como a integração com dispositivos IoT para rastreamento em campo e a otimização da cadeia produtiva.

Além disso, pode ser necessário garantir a validade e rastreabilidade do gado em campo, tendo como finalidade evitar a substituição ou extravio dos animais. Para lidar com essas preocupações, algumas soluções podem ser implementadas em trabalhos futuros, sendo elas:

- Uso de dispositivos IoT: a integração de dispositivos IoT, como dispositivos de rastreamento GPS e sensores de saúde, pode ser considerada para garantir uma rastreabilidade mais precisa e monitoramento em tempo real do gado. Esses dispositivos podem ser implantados nos animais para rastrear suas localizações e condições de saúde em tempo real, além de ajudar a evitar a substituição ou extravio de animais.
- Identificação única do gado: Assegurar que cada animal tenha uma identificação única, como RFID, pode dificultar a substituição ou o extravio dos animais. Isso pode ser feito através de etiquetas eletrônicas, chips ou outros dispositivos de identificação que sejam difíceis de remover ou adulterar.
- Controles de acesso e segurança: Implementar controles de acesso e segurança rigorosos no sistema para evitar acessos não autorizados e manipulação de dados.

Em resumo, o sistema desenvolvido nesta pesquisa representa um passo significativo em direção a uma gestão pecuária mais moderna e informatizada. As fundações estabelecidas aqui servem como um ponto de partida para futuras inovações que poderão transformar ainda mais a indústria, tornando-a mais eficiente, segura e sustentável.

7.2 Desafios

Apesar das conquistas alcançadas com o desenvolvimento e a implementação do sistema eCattle, diversos desafios foram identificados ao longo do projeto.

Inicialmente, pensou-se em trabalhar com a rastreabilidade dos animais utilizando dispositivos de rastreamento avançados instalados diretamente nos animais e tecnologias IoT. No entanto, essa abordagem foi descartada devido à falta de infraestrutura adequada nas propriedades rurais e alto custo de implementação. Muitas áreas enfrentam problemas de conectividade, o que impede a comunicação contínua e confiável entre os dispositivos IoT e a rede Blockchain. A solução encontrada foi focar na rastreabilidade baseada nos dados providos pela Blockchain, sem a necessidade de rastreamento através de sensores instalados no gado, o que ainda assim proporciona uma segurança e imutabilidade dos dados.

Um dos principais desafios atuais por parte dos produtores rurais é a resistência à adoção de novas tecnologias no setor agropecuário. Muitos produtores ainda estão acostumados com métodos tradicionais de gerenciamento e podem hesitar em adotar sistemas baseados em Blockchain ou IoT. Para superar essa resistência, é essencial investir em programas de treinamento e demonstrações práticas que mostrem os benefícios tangíveis da tecnologia.

A integração de dispositivos IoT com a Blockchain também apresenta desafios técnicos, especialmente em áreas rurais com conectividade limitada. A implementação bem-sucedida exigirá infraestrutura robusta e soluções inovadoras para garantir que os dispositivos possam se comunicar de forma confiável com a rede Blockchain, independentemente das condições ambientais.

Outro desafio significativo é a escalabilidade do sistema. À medida que o número de animais e transações aumenta, é crucial garantir que o sistema eCattle possa lidar com essa expansão sem comprometer o desempenho ou a segurança. Isso exigirá otimizações contínuas do software e possivelmente a adoção de novas tecnologias de escalabilidade para Blockchain.

A questão da privacidade dos dados também deve ser cuidadosamente considerada. Embora a transparência seja uma das principais vantagens da Blockchain, é importante equilibrá-la com a necessidade de proteger informações sensíveis dos produtores. Imple-

mentar protocolos de privacidade será fundamental para garantir a confiança dos usuários no sistema.

Finalmente, o custo de implementação e manutenção de um sistema baseado em Blockchain e IoT pode ser proibitivo para pequenos produtores. Desenvolver modelos de negócio sustentáveis e buscar apoio de políticas públicas e subsídios serão passos importantes para tornar a tecnologia acessível a todos os segmentos do setor agropecuário.

7.3 Publicações e Submissões

Durante o desenvolvimento do sistema eCattle, outros trabalhos foram documentados e submetidos para congressos acadêmicos, com o intuito de compartilhar conhecimentos e contribuir para a comunidade científica.

O artigo intitulado “*AgroShare: a system for sharing and trading agricultural resources*”, foi publicado no Simpósio Brasileiro de Sistemas de Informação (SBSI) de 2024. Este artigo foi uma colaboração entre Francis E. R. da Silva, Nilton R. Assis Júnior e Bruno G. Batista da Universidade Federal de Itajubá e está disponível no *Link*¹.

O AgroShare aborda a problemática enfrentada por pequenos e médios produtores rurais no Brasil, especialmente no que se refere à dificuldade de acesso a recursos essenciais como maquinário e mão de obra especializada. A solução proposta no artigo é a criação de um sistema de compartilhamento e negociação de recursos agrícolas, similar ao modelo de negócios utilizado por plataformas como Uber e Airbnb, mas adaptado ao contexto rural brasileiro. Utilizando tecnologias como React Native e Firebase, o sistema AgroShare foi modelado e desenvolvido para facilitar a interação entre produtores rurais, permitindo o compartilhamento de recursos de forma eficiente e acessível.

Os resultados do estudo demonstraram que o AgroShare é uma ferramenta eficaz para a digitalização das práticas agrícolas, promovendo a conectividade e a colaboração entre os produtores. A avaliação realizada com os usuários finais indicou uma aceitação positiva do sistema, com destaque para sua usabilidade e utilidade prática no contexto agropecuário. O AgroShare foi desenvolvido inicialmente para dispositivos móveis com o Sistema Operacional (SO) Android, mas poderá ser expandido no futuro para outros SOs, tal como iOS.

Além deste artigo já publicado no SBSI, outros resultados do eCattle foram submetidos. Entre eles, um versão submetida ao congresso *Symposium On Applied Computing* (SAC), foi aprovada como *Short Paper*, porém os autores preferiram não prosseguir com a publicação devido à aspectos financeiros.

¹ *AgroShare: a system for sharing and trading agricultural resources*. <<https://dl.acm.org/doi/10.1145/3658271.3658290>>

Outra versão, intitulada “*eCattle: A Platform for Traceability and Management of Livestock Transactions using Smart Contracts*”, foi submetida ao congresso *International Conference on Information, Intelligence, Systems and Applications* (IISA) de 2024, aprovada como *Full Paper* e apresentada no dia 18 de julho de 2024. Até o momento da entrega desta versão final da dissertação, ainda não foi efetivada a publicação do trabalho.

Além dos artigos já submetidos, está sendo modelado um artigo final do eCattle para ser submetido ao *The International Journal for the Computer and Telecommunications Industry*.

Todas essas submissões refletem a relevância e o impacto das soluções desenvolvidas no âmbito deste projeto, bem como o compromisso em disseminar o conhecimento e promover inovações no setor agropecuário.

Apêndices

APÊNDICE A – Roteiro da Avaliação para o Sistema eCattle

Preparação da Avaliação

Breve introdução aos tópicos:

- Explicação sobre o sistema eCattle e sua funcionalidade de rastreabilidade e gestão de transações de gado usando a tecnologia blockchain atrelada à contratos inteligentes.
- Apresentação do objetivo do eCattle e como ele pretende resolver problemas específicos no setor agropecuário.
- Esclarecimento do objetivo e estrutura da avaliação, incluindo a importância da participação do entrevistado.

Informações Iniciais e Consentimento

Você concorda em participar desta avaliação do sistema eCattle, entendendo que suas respostas serão usadas para melhorar o sistema? As respostas serão mantidas confidenciais.

Perfil do Usuário

Informações Demográficas e Profissionais

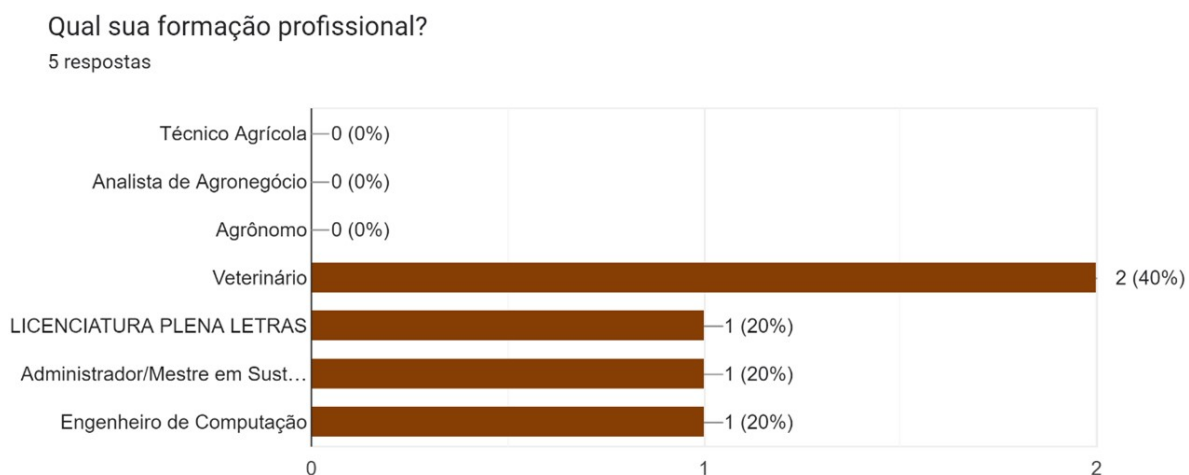


Figura 31 – Formação profissional. Fonte: Autor.

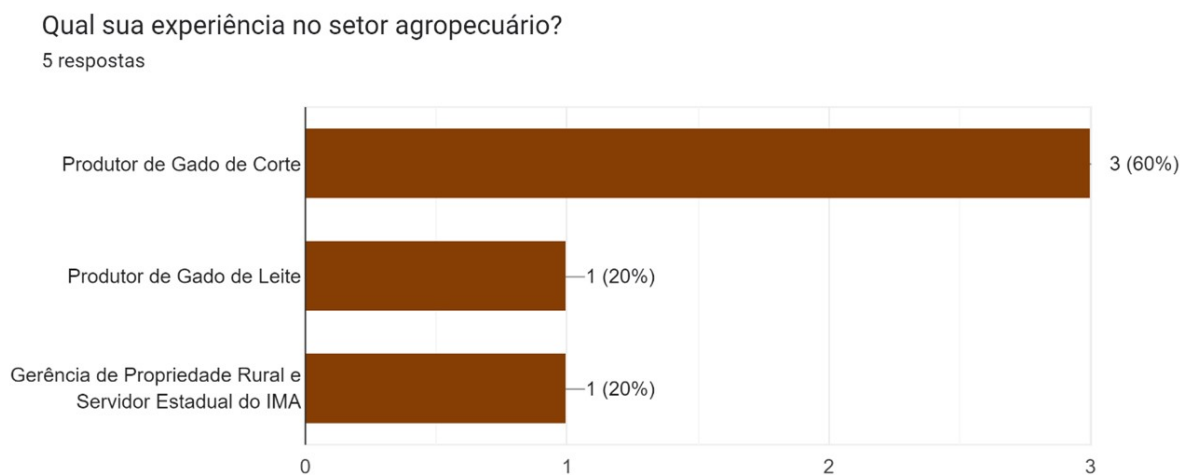


Figura 32 – Experiência no setor agropecuário. Fonte: Autor.

Você costuma tomar decisões relacionadas à gestão do seu rebanho?

5 respostas

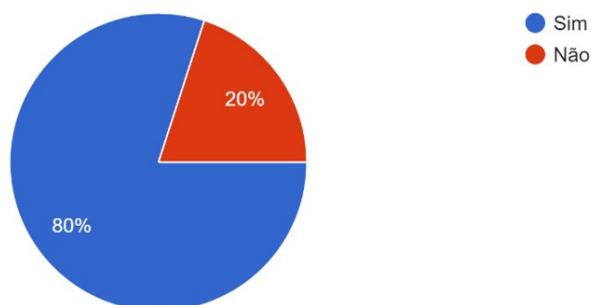


Figura 33 – Decisões relacionadas à gestão do rebanho. Fonte: Autor.

Questões

Compreensão Geral

Entendo claramente o propósito e funcionamento do sistema eCattle.

5 respostas

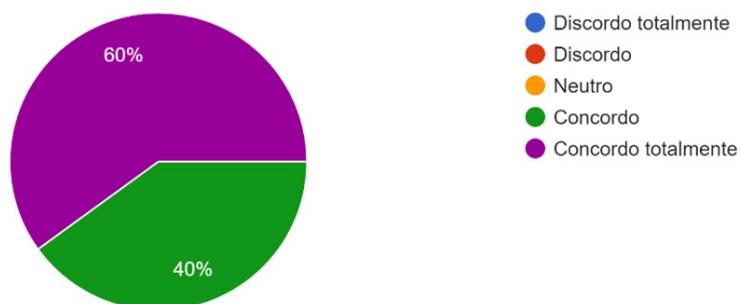


Figura 34 – Entende claramente o propósito e funcionamento do sistema eCattle. Fonte: Autor.

Todas as funcionalidades e dados apresentados pelo sistema eCattle foram claros para mim.
5 respostas

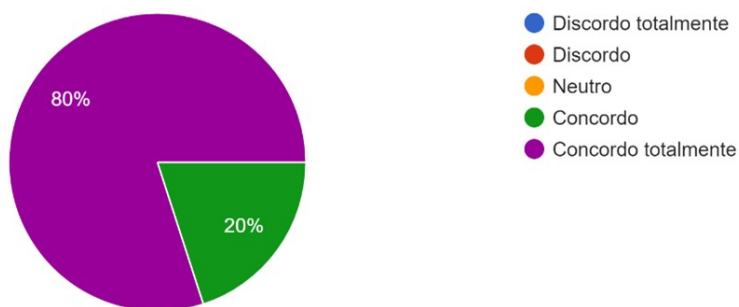


Figura 35 – Funcionalidades e dados apresentados pelo sistema eCattle. Fonte: Autor.

Usabilidade e Intuitividade

Minha experiência inicial com o eCattle foi fácil e intuitiva.
5 respostas

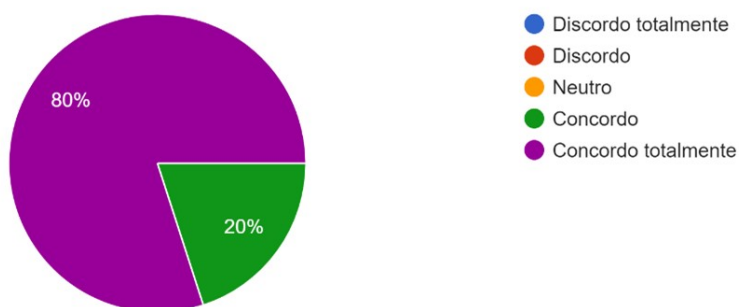


Figura 36 – Experiência inicial com o eCattle. Fonte: Autor.

Não encontrei dificuldades de uso ou navegação no sistema eCattle.

5 respostas

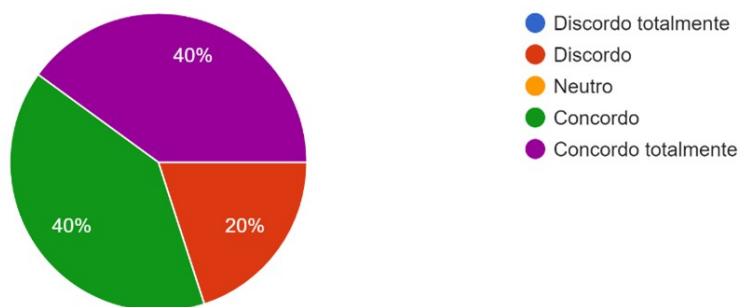


Figura 37 – Dificuldades de uso ou navegação. Fonte: Autor.

Avaliação e Melhorias do Rebanho

O sistema eCattle forneceu entendimentos valiosos para a gestão do meu rebanho.

5 respostas

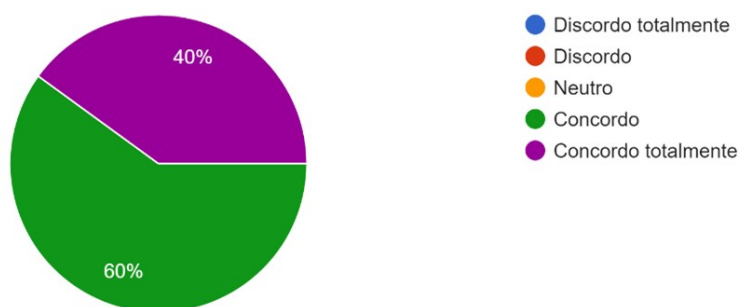


Figura 38 – Entendimentos para a gestão do rebanho. Fonte: Autor.

Identifiquei oportunidades de melhoria na gestão do rebanho a partir dos dados fornecidos pelo sistema eCattle.

5 respostas

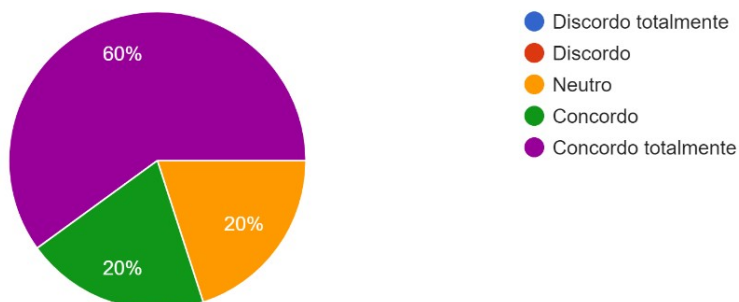


Figura 39 – Oportunidades de melhoria na gestão do rebanho. Fonte: Autor.

Percepções Específicas

As informações sobre a rastreabilidade e transações de gado dentro do sistema eCattle atendem às minhas necessidades.

5 respostas

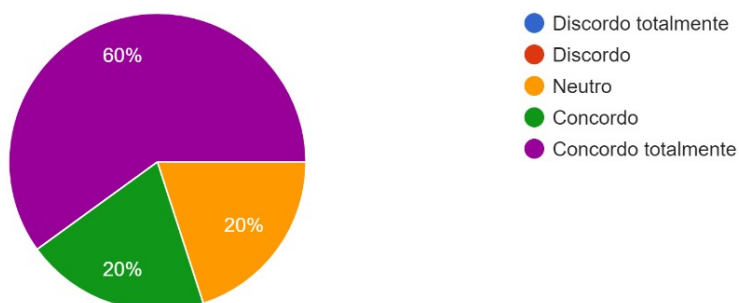


Figura 40 – O sistema eCattle atende às necessidades do usuário. Fonte: Autor.

Considero as funcionalidades e dados apresentados pelo eCattle particularmente valiosos para a gestão do rebanho.

5 respostas



Figura 41 – As funcionalidades do eCattle são importantes para a gestão do rebanho. Fonte: Autor.

Utilidade Potencial

Ter acesso ao sistema eCattle é essencial para o meu dia-a-dia na gestão do rebanho.

5 respostas

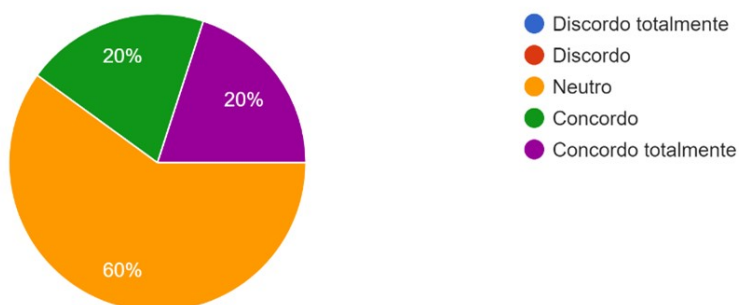


Figura 42 – Essencial para o dia-a-dia do usuário na gestão do rebanho. Fonte: Autor.

O eCattle tem o potencial de auxiliar significativamente nas minhas tarefas diárias relacionadas à gestão do rebanho.

5 respostas

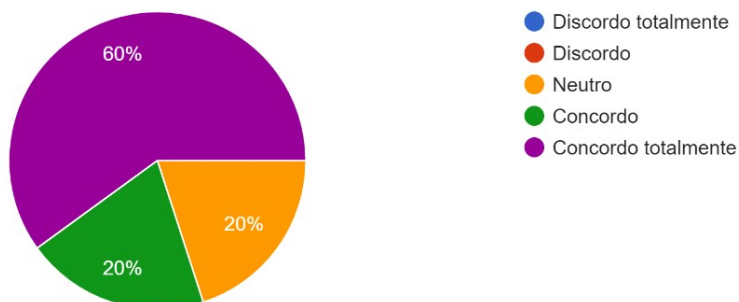


Figura 43 – Potencial de auxiliar significativamente nas tarefas do usuário relacionadas à gestão do rebanho. Fonte: Autor.

Feedback Visual e Dados

As visualizações de dados no eCattle chamaram minha atenção pela clareza e utilidade.

5 respostas

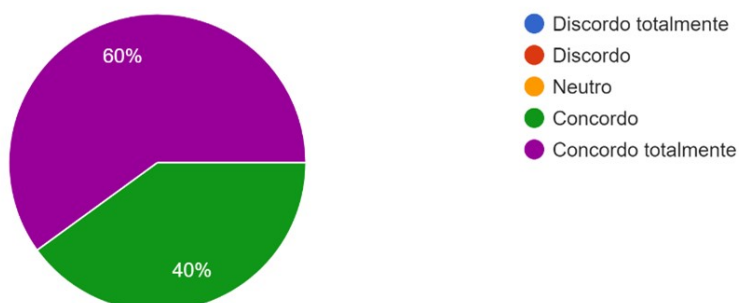


Figura 44 – A visualização de dados no sistema eCattle é clara e útil. Fonte: Autor.

Não encontrei visualizações de dados confusas ou menos úteis no sistema eCattle.

5 respostas

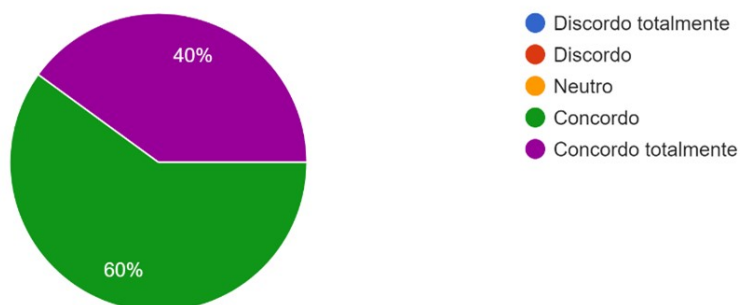


Figura 45 – A visualização de dados não são confusas ou menos úteis no sistema eCattle.
Fonte: Autor.

Considerações Finais

Feedback Adicional

- Existe alguma sugestão ou comentário adicional que você gostaria de fazer sobre o sistema eCattle?
- Há alguém que você recomendaria para participar deste processo de avaliação?

Referências

- 1 NXP Semiconductors. *Smart Farming: Harvesting Data to Increase Yields and Sustainability*. 2024. Accessed: 2024-05-25. Disponível em: <<https://www.nxp.com/company/blog/smart-farming-harvesting-data:BL-SMART-FARMING>>. 9, 19
- 2 MAGISTECH. *Do Produto para Trás: Por que Apostar na Rastreabilidade?* 2024. Accessed: 2024-01-15. Disponível em: <<https://site.magistech.com.br/do-produto-para-tras-por-que-apostar-na-rastreabilidade/>>. 9, 19, 20
- 3 AUNG, M. M.; CHANG, Y. S. Traceability in a food supply chain: Safety and quality perspectives. *Food control*, Elsevier, v. 39, p. 172–184, 2014. 9, 17, 19, 20, 21, 22, 27
- 4 FAO. The state of food security and nutrition in the world. *Web page of the Food and Agriculture Organization of the United Nations*, 2018. 16
- 5 DELGADO, C. et al. Livestock to 2020: The next food revolution. *Outlook on Agriculture*, SAGE Publications Sage UK: London, England, v. 30, n. 1, p. 27–29, 2001. 16
- 6 THORNTON, P. et al. *Mapping poverty and livestock in the developing world*. [S.l.]: ILRI (aka ILCA and ILRAD), 2002. v. 1. 16
- 7 THORNTON, P. K. et al. The impacts of climate change on livestock and livestock systems in developing countries: A review of what we know and what we need to know. *Agricultural systems*, Elsevier, v. 101, n. 3, p. 113–127, 2009. 16
- 8 STEINFELD, H. et al. *Livestock's long shadow: environmental issues and options*. [S.l.]: Food & Agriculture Org., 2006. 16
- 9 GODFRAY, H. C. J. et al. Meat consumption, health, and the environment. *Science*, American Association for the Advancement of Science, v. 361, n. 6399, p. eaam5324, 2018. 16
- 10 GARNETT, T. et al. Sustainable intensification in agriculture: premises and policies. *Science*, American Association for the Advancement of Science, v. 341, n. 6141, p. 33–34, 2013. 16
- 11 RADAS, S.; TEISL, M. F.; ROE, B. An open mind wants more: opinion strength and the desire for genetically modified food labeling policy. *Journal of Consumer Affairs*, Wiley Online Library, v. 42, n. 3, p. 335–361, 2008. 16, 27
- 12 MGONJA F. J., J. Z. L. X. . M. S. V. Enhancing food safety and integrity through blockchain technology: A review of the existing gaps, challenges, and opportunities. *Food Reviews International*, 2021. 16, 17, 19, 24, 27
- 13 SLINGENBERGH, J. et al. Ecological sources of zoonotic diseases. *Revue scientifique et technique-Office international des épizooties*, Paris: L'Office, 1982-, v. 23, n. 2, p. 467–484, 2004. 16, 19, 20

- 14 CHARLIER, J. et al. Measurement of antibodies to gastrointestinal nematodes and liver fluke in meat juice of beef cattle and associations with carcass parameters. *Veterinary parasitology*, Elsevier, v. 166, n. 3-4, p. 235–240, 2009. [16](#)
- 15 LIDDELL, S.; BAILEY, D. Market opportunities and threats to the us pork industry posed by traceability systems. *The International Food and Agribusiness Management Review*, Elsevier, v. 4, n. 3, p. 287–302, 2001. [16](#), [20](#), [23](#)
- 16 HOBBS, J. E. Information asymmetry and the role of traceability systems. *Agribusiness: An International Journal*, Wiley Online Library, v. 20, n. 4, p. 397–415, 2004. [16](#), [17](#), [20](#), [23](#)
- 17 SANDERSON, K.; HOBBS, J. E. Traceability and process verification in the canadian beef industry. *Department of Agricultural Economics, University of Saskatchewan*, 2006. [17](#)
- 18 MEEUSEN R., C. A. . P. E. Animal traceability for official control: An overview of the current situation and challenges in the european union. *EFSA Supporting Publications*, 2018. [17](#), [19](#), [20](#), [27](#)
- 19 OPARA, L. U.; MAZAUD, F. Food traceability from field to plate. *Outlook on agriculture*, SAGE Publications Sage UK: London, England, v. 30, n. 4, p. 239–247, 2001. [17](#)
- 20 TIAN, F. An agri-food supply chain traceability system for china based on rfid & blockchain technology. In: IEEE. *2016 13th international conference on service systems and service management (ICSSSM)*. [S.l.], 2016. p. 1–6. [17](#), [18](#), [22](#), [27](#)
- 21 KAMBLE, S.; GUNASEKARAN, A.; ARHA, H. Understanding the blockchain technology adoption in supply chains-indian context. *International Journal of Production Research*, Taylor & Francis, v. 57, n. 7, p. 2009–2033, 2019. [18](#)
- 22 GALVEZ, J. F.; MEJUTO, J. C.; SIMAL-GANDARA, J. Future challenges on the use of blockchain for food traceability analysis. *TrAC Trends in Analytical Chemistry*, Elsevier, v. 107, p. 222–232, 2018. [18](#)
- 23 THORNTON, P. K. Livestock production: recent trends, future prospects. *Philosophical Transactions of the Royal Society B: Biological Sciences*, The Royal Society, v. 365, n. 1554, p. 2853–2867, 2010. [19](#), [22](#)
- 24 FOOD; (FAO), A. O. of the U. N. *World Livestock 2011: Livestock in Food Security*. [S.l.]: fao Rome, 2011. [19](#)
- 25 POULIOT, S.; SUMNER, D. A. Traceability, liability, and incentives for food safety and quality. *American Journal of Agricultural Economics*, Wiley Online Library, v. 90, n. 1, p. 15–27, 2008. [19](#), [20](#), [22](#)
- 26 RUSHTON, J.; GILBERT, W. The economics of animal health: direct and indirect costs of animal disease outbreaks. *Working Document*, v. 84, 2016. [19](#), [20](#)
- 27 TRIENEKENS, J. H. et al. Transparency in complex dynamic food supply chains. *Advanced Engineering Informatics*, Elsevier, v. 26, n. 1, p. 55–65, 2012. [23](#)

- 28 GOLAN, E. H. et al. *Traceability in the US food supply: economic theory and industry studies*. [S.l.], 2004. 23
- 29 DÜRR J. W., . G. A. R. Definition and integration of animal identification systems for total herd recording in dairy cattle. *Livestock Science*, v. 111, n. 1-2, p. 104–116, 2007. 23
- 30 GRAY A. W., . B. M. Rfid and traceability in agriculture: The case of the cattle industry. *New Name 2008: Agricultural and Applied Economics Association*, 2005. 23
- 31 KALOXYLOS, A. et al. Farm management systems and the future internet era. *Computers and electronics in agriculture*, Elsevier, v. 89, p. 130–144, 2012. 23
- 32 EGEA M. Á., P. M. D. . M. J. M. A web-based system for monitoring and controlling precision livestock farming facilities. *Computers and Electronics in Agriculture*, v. 90, p. 1–10, 2013. 23
- 33 MOTTRAM, T. Animal board invited review: precision livestock farming for dairy cows with a focus on oestrus detection. *Animal*, Elsevier, v. 10, n. 10, p. 1575–1584, 2016. 24
- 34 MOTTRAM, T. Livestock precision farming: A review. *European Society of Agricultural Engineers*, p. 1–15, 2011. 24
- 35 CAO, S. et al. Strengthening consumer trust in beef supply chain traceability with a blockchain-based human-machine reconcile mechanism. *Computers and electronics in agriculture*, Elsevier, v. 180, p. 105886, 2021. 24
- 36 TAYLOR, N.; RUSHTON, J. *A value chain approach to animal diseases risk management: technical foundations and practical framework for field application*. [S.l.]: FAO, 2011. v. 4. 24
- 37 NAKAMOTO, S. Bitcoin: A peer-to-peer electronic cash system. *Decentralized business review*, p. 21260, 2008. 24, 25, 26, 38, 69
- 38 MERKLE, R. C. A digital signature based on a conventional encryption function. In: SPRINGER. *Advances in Cryptology—CRYPTO’87: Proceedings 7*. [S.l.], 1988. p. 369–378. 24, 25, 26
- 39 ZOHAR, A. Bitcoin: under the hood. *Communications of the ACM*, ACM New York, NY, USA, v. 58, n. 9, p. 104–113, 2015. 24, 25, 26
- 40 BONNEAU, J. et al. Sok: Research perspectives and challenges for bitcoin and cryptocurrencies. In: IEEE. *2015 IEEE symposium on security and privacy*. [S.l.], 2015. p. 104–121. 25, 26
- 41 CROSBY, M. et al. Blockchain technology: Beyond bitcoin. *Applied Innovation*, v. 2, n. 6-10, p. 71, 2016. 25, 26
- 42 CHRISTIDIS, K.; DEVETSIKIOTIS, M. Blockchains and smart contracts for the internet of things. *Ieee Access*, Ieee, v. 4, p. 2292–2303, 2016. 25

- 43 HÄSLER, B. et al. The economic value of one health in relation to the mitigation of zoonotic disease risks. *One Health: The Human-Animal-Environment Interfaces in Emerging Infectious Diseases: The Concept and Examples of a One Health Approach*, Springer, p. 127–151, 2013. 27
- 44 TSANG, Y. P. et al. Blockchain-driven iot for food traceability with an integrated consensus mechanism. *IEEE access*, IEEE, v. 7, p. 129000–129017, 2019. 27
- 45 LENG, K. et al. Research on agricultural supply chain system with double chain architecture based on blockchain technology. *Future Generation Computer Systems*, Elsevier, v. 86, p. 641–649, 2018. 27
- 46 HASAN, H. et al. Smart contract-based approach for efficient shipment management. *Computers & industrial engineering*, Elsevier, v. 136, p. 149–159, 2019. 27
- 47 ZHANG, C.; GONG, Y.; BROWN, S. Case analysis: Beefledger. In: *Blockchain Applications in Food Supply Chain Management: Case Studies and Implications*. [S.l.]: Springer, 2023. p. 99–139. 27
- 48 KAMPAN, K.; TSUSAKA, T. W.; ANAL, A. K. Adoption of blockchain technology for enhanced traceability of livestock-based products. *Sustainability*, MDPI, v. 14, n. 20, p. 13148, 2022. 27
- 49 STUDENT, M.; DIEPENHORST, J. J. Proof of concept: Demand driven supply through farming techniques on carcass quality control points. 2021. 27
- 50 GALLO, P. et al. Agrichain: Blockchain syntactic and semantic validation for reducing information asymmetry in agri-food. *Proceedings http://ceur-ws.org ISSN*, v. 1613, p. 0073, 2022. 27
- 51 VETSOURCE. Home delivery for veterinary practices. 2021. 27
- 52 PATIL, R. et al. Rental and loan system in agriculture using blockchain technology. *Journal of Scientific Research and Technology*, p. 1–14, 2023. 28, 29
- 53 MAO, D. et al. Novel automatic food trading system using consortium blockchain. *Arabian Journal for Science and Engineering*, Springer, v. 44, p. 3439–3455, 2019. 28, 29
- 54 RAMPRASATH, J. et al. Secured data transaction for agriculture harvesting using blockchain technology. In: IEEE. *2023 2nd International Conference on Vision Towards Emerging Trends in Communication and Networking Technologies (ViTECoN)*. [S.l.], 2023. p. 1–4. 28, 29
- 55 VANDITHA, M. et al. Agricultural supply chain management system using blockchain. In: IEEE. *2023 International Conference on Recent Trends in Electronics and Communication (ICRTEC)*. [S.l.], 2023. p. 1–4. 28, 29
- 56 ALSHEHRI, M. Blockchain-assisted internet of things framework in smart livestock farming. *Internet of Things*, Elsevier, v. 22, p. 100739, 2023. 28, 29
- 57 JEGADEESAN, S. et al. Blockchain based lightweight and secure aggregation scheme for smart farming. In: IEEE. *2023 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS)*. [S.l.], 2023. p. 1266–1271. 28, 29

- 58 RAMBHIA, V. et al. Agrichain: A blockchain-based food supply chain management system. In: SPRINGER. *International Conference on Blockchain*. [S.l.], 2021. p. 3–15. [28](#), [29](#)
- 59 LIN, J. et al. Blockchain and iot based food traceability for smart agriculture. In: *Proceedings of the 3rd international conference on crowd science and engineering*. [S.l.: s.n.], 2018. p. 1–6. [29](#)
- 60 BALEKUNDRI, O.; TIGADI, R.; JAYAKKANAVAR, A. Farmsupply: Food supply chain management using blockchain technology. In: IEEE. *2023 IEEE 3rd International Conference on Technology, Engineering, Management for Societal impact using Marketing, Entrepreneurship and Talent (TEMSMET)*. [S.l.], 2023. p. 1–6. [29](#), [30](#)
- 61 ADENIYI, J. K. et al. Agrolend: A blockchain implementation approach of smart contract farming platform. In: IEEE. *2023 International Conference on Science, Engineering and Business for Sustainable Development Goals (SEB-SDG)*. [S.l.], 2023. v. 1, p. 1–7. [29](#), [30](#)
- 62 RAGGETT D., L. H. A. . J. I. Html 4.01 specification. 1999. [30](#)
- 63 BOS B., L. H. W. L. C. . J. I. Cascading style sheets, level 2: Css2 specification. 1999. [30](#)
- 64 FLANAGAN, D. *JavaScript: The Definitive Guide*, O'Reilly. [S.l.]: London, 2006. [30](#)
- 65 LERDORF, R. et al. *Programming Php*. [S.l.]: "O'Reilly Media, Inc.", 2002. [30](#)
- 66 OTTO M., . T. J. Bootstrap. 2011. [30](#)
- 67 GRIFFITHS, A. *CodeIgniter 1.7 Professional Development*. [S.l.]: Packt Pub., 2010. [30](#)
- 68 LEFF, A.; RAYFIELD, J. T. Web-application development using the model/view/controller design pattern. In: IEEE. *Proceedings fifth iee international enterprise distributed object computing conference*. [S.l.], 2001. p. 118–127. [31](#), [33](#)
- 69 FIELDING, R. T. *Architectural styles and the design of network-based software architectures*. [S.l.]: University of California, Irvine, 2000. [31](#)
- 70 FIELDING, R. T.; TAYLOR, R. N. Principled design of the modern web architecture. *ACM Transactions on Internet Technology (TOIT)*, ACM New York, NY, USA, v. 2, n. 2, p. 115–150, 2002. [31](#), [32](#)
- 71 RICHARDSON, L.; RUBY, S. *RESTful web services*. [S.l.]: "O'Reilly Media, Inc.", 2008. [31](#), [32](#)
- 72 PAUTASSO, C.; ZIMMERMANN, O.; LEYMANN, F. Restful web services vs."big"web services: making the right architectural decision. In: *Proceedings of the 17th international conference on World Wide Web*. [S.l.: s.n.], 2008. p. 805–814. [32](#)
- 73 BROWN, S. *The Art of Visualising Software Architecture*. [S.l.]: Leanpub, 2015. [34](#)
- 74 FABRIC, H. Hyperledger fabric documentation. 2023. [39](#), [41](#), [46](#)

- 75 ANDROULAKI, E. et al. Hyperledger fabric: a distributed operating system for permissioned blockchains. In: *Proceedings of the thirteenth EuroSys conference*. [S.l.: s.n.], 2018. p. 1–15. [39](#), [40](#), [42](#)
- 76 ONGARO, D.; OUSTERHOUT, J. In search of an understandable consensus algorithm. In: *2014 {USENIX} Annual Technical Conference ({USENIX}{ATC} 14)*. [S.l.: s.n.], 2014. p. 305–319. [40](#)
- 77 LUO, L. et al. On the security and data integrity of low-cost sensor networks for air quality monitoring. *Sensors*, MDPI, v. 18, n. 12, p. 4451, 2018. [46](#)
- 78 ORACLE. *MySQL: The world's most popular open source database*. [S.l.: s.n.], 2023. [<https://www.mysql.com/>](https://www.mysql.com/). [46](#)
- 79 BOOCH, G. *The unified modeling language user guide*. [S.l.]: Pearson Education India, 2005. [54](#)
- 80 PROVOS, N.; MAZIERES, D. A future-adaptable password scheme. In: *USENIX Annual Technical Conference, FREENIX Track*. [S.l.: s.n.], 1999. v. 1999, p. 81–91. [67](#)
- 81 BUTERIN, V. et al. A next-generation smart contract and decentralized application platform. *white paper*, v. 3, n. 37, p. 2–1, 2014. [69](#)
- 82 REIS, D. et al. Developing docker and docker-compose specifications: A developers' survey. *Ieee Access*, IEEE, v. 10, p. 2318–2329, 2021. [70](#)
- 83 HEVNER, A. R. et al. Design science in information systems research. *MIS quarterly*, JSTOR, p. 75–105, 2004. [78](#), [96](#)
- 84 HEVNER, A.; CHATTERJEE, S. *Design research in information systems: theory and practice*. [S.l.]: Springer Science & Business Media, 2010. v. 22. [78](#), [79](#), [81](#)
- 85 PEFFERS, K. et al. A design science research methodology for information systems research. *Journal of management information systems*, Taylor & Francis, v. 24, n. 3, p. 45–77, 2007. [78](#), [81](#)
- 86 LIKERT, R. A technique for the measurement of attitudes. *Archives of psychology*, 1932. [78](#), [79](#), [81](#)
- 87 JAMIESON, S. Likert scales: How to (ab) use them? *Medical education*, Blackwell, v. 38, n. 12, p. 1217–1218, 2004. [78](#), [79](#), [81](#), [83](#)
- 88 CRESWELL, J. W.; CRESWELL, J. D. *Research design: Qualitative, quantitative, and mixed methods approaches*. [S.l.]: Sage publications, 2017. [79](#)
- 89 COHEN, J. *Statistical power analysis for the behavioral sciences*. [S.l.]: Routledge, 2013. [81](#), [82](#)
- 90 MONTGOMERY, D. C. *Design and analysis of experiments*. [S.l.]: John wiley & sons, 2017. [81](#), [83](#)
- 91 STUDENT. The probable error of a mean. *Biometrika*, JSTOR, p. 1–25, 1908. [82](#)

- 92 CHERKASOVA, L.; GUPTA, D.; VAHDAT, A. Comparison of the three cpu schedulers in xen. *Performance Evaluation Review*, ACM ASSOCIATION FOR COMPUTING MACHINERY, v. 35, n. 2, p. 42, 2007. 87
- 93 CALHEIROS, R. N. et al. Cloudsim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. *Software: Practice and experience*, Wiley Online Library, v. 41, n. 1, p. 23–50, 2011. 87, 95
- 94 BASHIR, I. *Mastering blockchain*. [S.l.]: Packt Publishing Ltd, 2017. 87, 95
- 95 BATISTA, B. G. et al. Performance evaluation of resource management in cloud computing environments. *PloS one*, Public Library of Science San Francisco, CA USA, v. 10, n. 11, p. e0141914, 2015. 87
- 96 Apache Software Foundation. *Apache Benchmark*. 2024. Online; accessed 14-August-2024. <<https://httpd.apache.org/docs/2.4/programs/ab.html>>. 88
- 97 Linux Foundation. *vmstat: Report virtual memory statistics*. [S.l.], 2024. <<https://linux.die.net/man/8/vmstat>>. 90
- 98 SHNEIDERMAN, B.; PLAISANT, C. *Designing the user interface: strategies for effective human-computer interaction*. [S.l.]: Pearson Education India, 2010. 95
- 99 ZYSKIND, G.; NATHAN, O. et al. Decentralizing privacy: Using blockchain to protect personal data. In: IEEE. *2015 IEEE security and privacy workshops*. [S.l.], 2015. p. 180–184. 95