
Criação de um *Framework* para
desenvolvimento de aplicações
descentralizadas baseadas em redes
Blockchain

Frank Castilio Pinheiro de Alencar

SERVIÇO DE PÓS-GRADUAÇÃO DA FACOM-UFMS

Data de Depósito:

Assinatura: _____

Criação de um *Framework* para desenvolvimento de aplicações descentralizadas baseadas em redes *Blockchain*

Frank Castílio Pinheiro de Alencar

Orientador: *Prof. Dr. Dionísio Machado Leite Filho*

Coorientador: *Prof. Dr. Carlos Henrique Gomes Ferreira*

Dissertação apresentada à Faculdade de Computação da Universidade Federal de Mato Grosso do Sul como parte dos requisitos necessários à obtenção do título de Mestre em Computação Aplicada.

**Campo Grande - MS
2024**

[Dedicatória]
Aos meus pais,
Margareti Castilio e Reinaldo Costa,

Ao meu filho,
Otávio,

À minha companheira, Celly Nataly.

Agradecimentos

Agradeço a minha esposa Celly, pelo apoio e compreensão.

Agradeço ao professor orientador Dionísio Filho e ao professor coorientador Carlos Ferreira.

À Faculdade de Computação da UFMS e ao corpo docente do Programa de Pós-Graduação em Computação Aplicada, por proporcionarem uma educação de qualidade.

Ao Instituto Federal de Mato Grosso do Sul (IFMS) por conceder sete meses de afastamento para conclusão do programa de pós-graduação.

Abstract

Based on Blockchain networks, decentralized applications (DApps) are expanding. Research shows that these applications have been created without following methodological criteria for software development. Therefore, it is important to systematize their construction so that developers can focus on their research problem. The lack of clear guidelines for creating DApps is a challenge and hinders the entry of new researchers and developers into this field. Research attempting to fill this gap often addresses specific issues in the DApp development process. Thus, this work relied on a systematic literature review (RSL) and created a framework to assist in DApp development, specifically in decision-making about which decentralized platform to use to host the application, conceptualizing and discussing the technologies used in each stage of this process. The RSL revealed the absence of a similar structure to the one developed in this work, the framework provides guidance for selecting the decentralized platform according to the DApp's requirements.

Keywords: DApp, decentralized application, systematization, framework, Blockchain.

Resumo

Baseadas em redes *Blockchain* as aplicações descentralizadas (DApp) estão em expansão. Pesquisas demonstram que estas aplicações vêm sendo criadas sem seguirem critérios metodológicos para desenvolvimento de *software*, desse modo, é importante que se sistematize sua construção para que o desenvolvedor possa focar em seu problema de pesquisa. A falta de diretrizes claras para a criação de DApps é um desafio, e dificulta a entrada de novos pesquisadores e desenvolvedores nesse campo. As pesquisas que tentam preencher essa lacuna geralmente abordam questões específicas do processo de desenvolvimento de DApps. Assim, este trabalho baseou-se em uma revisão sistemática da literatura (RSL) e criou um *framework* para auxiliar no desenvolvimento de DApps, mais especificamente na tomada de decisão sobre qual plataforma descentralizada utilizar para hospedar sua aplicação, conceituando e debatendo a respeito das tecnologias utilizadas em cada etapa deste processo. A RSL revelou a ausência de uma estrutura semelhante à que foi desenvolvida neste trabalho, o *framework* fornece orientações para seleção da plataforma descentralizada de acordo com os requisitos da DApp.

Palavras-chave: DApp, aplicação descentralizada, sistematização, *framework*, *Blockchain*.

Sumário

Sumário	x
Lista de Figuras	xii
Lista de Tabelas	xiii
Lista de Algoritmos	xiv
Lista de Abreviaturas	xiv
1 Introdução	1
1.1 Contextualização	1
1.2 Objetivos	3
1.2.1 Objetivo Geral	3
1.2.2 Objetivos Específicos	3
1.3 Estrutura do Documento	3
2 Fundamentação Teórica	5
2.1 Sistemas Distribuídos	5
2.2 <i>Peer-to-Peer</i>	6
2.3 <i>Blockchain</i>	7
2.3.1 <i>Blockchain</i> Pública e Privada	8
2.3.2 <i>Smart Contracts</i>	8
2.3.3 Algoritmos de Consenso	10
2.3.4 <i>Decentralized Application (DApp)</i>	12
2.3.5 Infraestrutura da <i>Blockchain</i>	15
2.4 Plataformas <i>Blockchain</i>	16
2.4.1 <i>Ethereum</i>	19
2.5 Interoperabilidade e Migração de Contratos Inteligentes	21
2.6 <i>Revisão Sistemática da Literatura (RSL)</i>	23
2.7 Considerações	24
3 Trabalhos Relacionados	26
3.1 Considerações	27

4	Revisão Sistemática da Literatura	29
4.1	Planejamento	29
4.2	Bases Científicas	30
4.3	Strings de busca	30
4.4	Critérios de inclusão e exclusão	32
4.5	Seleção das pesquisas	33
4.5.1	Ferramentas Utilizadas	34
4.6	Resultados	34
4.6.1	Discussão	34
4.7	Considerações	37
5	Resultados	38
5.1	Aplicação <i>Web</i> Clássica x DApp	38
5.2	Framework	40
5.3	Avaliação do Framework	43
5.3.1	Automatização do Framework	43
5.4	Execução da Avaliação	49
5.4.1	Perguntas	50
5.4.2	Perfil dos participantes	51
5.4.3	Sistemas	52
5.4.4	Resultados	53
5.5	Considerações	55
6	Conclusões	56
6.1	Dificuldades Encontradas	56
6.2	Principais Resultados e Contribuições	57
6.3	Trabalhos Futuros	58
	Referências	67

Lista de Figuras

2.1	Arquitetura <i>Peer-to-Peer</i> Um serviço fornecido por múltiplos servidores (Coulouris et al., 2013).	6
2.2	A história progressiva da <i>Blockchain</i> - era 1.0 e 2.0 (Lin et al., 2022b).	7
2.3	O princípio de funcionamento do algoritmo PoW (Xiong et al., 2022).	11
2.4	Comparação entre as aplicações. (a) Aplicação centralizada. (b) Aplicação descentralizada (Yang et al., 2019).	13
2.5	Arquitetura de uma DApp na rede Oracle (Baranwal, 2019).	13
2.6	Arquitetura de uma aplicação descentralizada baseada na plataforma Ethereum (Marchesi et al., 2020b).	15
2.7	A infraestrutura de uma <i>Blockchain</i> (Lin et al., 2022b)	16
2.8	Diagrama de uso de gas (Smith, 2023)	21
2.9	O processo de revisão (Elaborada pelo autor (2023))	24
4.1	Processo de seleção das pesquisas (elaborado pelo autor (2023)).	33
5.1	Tecnologias para desenvolvimento <i>web</i> clássico (elaborado pelo autor)	39
5.2	Plataforma/Tecnologias para desenvolvimento de DApp (elaborado pelo autor)	40
5.3	Diagrama para DApps em Geral (elaborado pelo autor (2023)).	41
5.4	Diagrama para DApps para Moedas Digitais (elaborado pelo autor).	43
5.5	Página principal da aplicação (elaborado pelo autor).	44
5.6	Segunda página da aplicação (elaborado pelo autor).	45
5.7	Terceira página da aplicação (elaborado pelo autor).	45
5.8	Pergunta sobre BFT (elaborado pelo autor).	46
5.9	Explicação sobre BFT (elaborado pelo autor).	47
5.10	Pergunta sobre rede permissionada (elaborado pelo autor).	47
5.11	Pergunta sobre linguagem de programação (elaborado pelo autor).	48
5.12	Pergunta sobre token (elaborado pelo autor).	48

5.13 Sugestão de plataforma (elaborado pelo autor).	49
5.14 Página de avaliação da aplicação (elaborado pelo autor).	50
5.15 Idade e Gênero do estudantes da turma 1 - graduação (elaborado pelo autor).	51
5.16 Idade e Gênero do estudantes da turma 2 - especialização (elaborado pelo autor).	51
5.17 Formação dos estudantes da turma 2 - especialização (elaborado pelo autor).	52
5.18 Resultado da avaliação com a turma 1 - graduação (elaborado pelo autor).	53
5.19 Resultado da avaliação com a turma 2 - especialização (elaborado pelo autor).	54

Lista de Tabelas

2.1	Várias plataformas de contratos inteligentes: prós e contras. (Taherdoost, 2023).	17
2.2	Plataformas de contratos inteligentes Sharma et al. (2023).	18
2.3	Principais diferenças entre as Plataformas (Khan et al., 2023).	18
2.4	Lacunas Das Soluções De Interoperabilidade (Monika e Bhatia, 2020)	23
3.1	Comparativo entre as pesquisas e alinhamento com os objetivos específicos desta dissertação (elaborado pelo autor (2023))	28
4.1	Bases de dados selecionadas (elaborado pelo autor (2023))	30
4.2	Bases e Strings de busca (elaborado pelo autor (2023))	31
4.3	Resultado das buscas (elaborado pelo autor (2023))	32
4.4	Comparativo entre as pesquisas relacionadas na RSL (elaborado pelo autor (2023))	37

Lista de Algoritmos

2.1 Máquina de venda automática (Wackerow, 2022)	9
--	---

Lista de Abreviaturas

ABCDE *Agile Block Chain DApp Engineering*

App *Application*

BPMN *Business Process Model and Notation*

BFT *Byzantine Fault Tolerance*

CSS *Cascading Style Sheets*

DADS *Decentralized (Mobile) Applications Deployment System*

DApp *Decentralized Application*

DASD *Distributed Agile Software Development*

DOM *Document Object Model*

DoS *Denial of service*

DPoS *Delegated Proof of Stake*

EVM *Ethereum Virtual Machine*

HTML *HyperText Markup Language*

IoT *Internet of Things*

IPFS *InterPlanetary File System*

JS *JavaScript*

JSON *JavaScript Object Notation*

LPoS *Leased Proof of Stake*

P2P *Peer to Peer*

PBFT *Practical Byzantine Fault-Tolerance*

PHP *PHP: Hypertext Preprocessor*

PoB *Proof of Burn*

PoW *Proof of Work*

PoWeight *Proof of Weight*

PoS *Proof of Stake*

RSL *Revisão Sistemática da Literatura*

SRV *State-reverting Vulnerability*

TPS *Transações por segundo*

TRC *Tracker Radar Collector*

SBFT *Simplified Byzantine Fault Tolerance*

SPAN *Sending and Posting Advertisement in Mass*

UI *User Interface*

UML *Unified Modeling Language*

Introdução

1.1 Contextualização

Executando em uma rede *Blockchain*, a *Decentralized Application* (DApp) é uma aplicação web que opera de maneira autônoma em cada uma das máquinas onde está instalada. Sua estrutura, com distinção entre as camadas de *back-end* e *front-end*, assemelha-se à de uma aplicação web convencional (Tas e Tanriover, 2019). A principal distinção reside no papel desempenhado pelo contrato inteligente, que age como intermediário entre a aplicação e a rede, realizando as transações entre comprador e vendedor sempre que os critérios pré-estabelecidos no contrato são atendidos (Tas e Tanriover, 2019).

O advento Dapps teve início na versão 2.0 da rede *Blockchain* e foi impulsionada pela melhoria dos contratos inteligentes e a necessidade de aprimorar o desempenho das transações realizadas. As funcionalidades e os serviços oferecidos por essas aplicações são variados, e a plataforma de desenvolvimento mais proeminente atualmente é a *Ethereum* (Lin et al., 2022b).

Muito se fala acerca de criação de aplicações descentralizadas, entretanto, raramente se evidencia de maneira clara e fundamentada o processo de desenvolvimento de software, isto é, os critérios ponderados em cada fase de decisão, bem como as alternativas tecnológicas viáveis ao longo da criação da DApp (Marchesi et al., 2020b).

Conforme observado por Marchesi et al. (2020b), em virtude da inovação da tecnologia *Blockchain* e com o propósito de explorar áreas inéditas, verificou-se um desenvolvimento apressado de aplicações descentralizadas e contratos inteligentes, nos quais os seus desenvolvedores deixaram de empregar boas

práticas. De acordo com os autores, devido à ausência de critérios metodológicos na sua concepção, um considerável número de aplicações acabou por ser suscetível à exploração por parte de usuários maliciosos.

Atualmente, muitas pesquisas estão relacionadas a compreensão do comportamento da *Blockchain* em uma rede 5G e seus impactos na criação de aplicações descentralizadas (Yue et al., 2021) e outros trabalhos como os de Gandhi e Salvi (2019) e Wang et al. (2022). Pesquisas que demonstram com mais detalhes a criação de DApps, o fazem utilizando a plataforma de desenvolvimento de aplicações descentralizadas da Ethereum (Bathen e Jadav, 2022), (Reilly et al., 2019), (Dhanvardini et al., 2023) e (Lallai et al., 2020).

A rede Ethereum hospeda uma variedade de aplicações devido à sua robustez e pioneirismo (Marchesi et al., 2020b). Contudo, ao direcionarem seus esforços para uma plataforma específica, os estudos muitas vezes omitem a discussão de pontos relevantes que emergem durante o desenvolvimento de aplicações web. Tal abordagem dificulta a compreensão para aqueles que estão ingressando na área (Marchesi et al., 2020b).

O desafio reside na escassez de estudos que ofereçam orientação abrangente, abarcando desde a análise até a publicação das DApps, o que dificulta a introdução de novos pesquisadores e desenvolvedores na área. Pesquisas, como as conduzidas por (Renu e Banik, 2021), (Shakila e Sultana, 2021) e (Lin et al., 2022a) que se propuseram a preencher essa lacuna, apresentam deficiências ao construir suas aplicações descentralizadas, direcionadas para uma plataforma específica, ou deixam de justificar de maneira clara as escolhas feitas ao longo desse processo de criação.

Diante da necessidade de estabelecer uma padronização no processo de desenvolvimento de DApps, o presente trabalho desenvolveu um *framework* de decisão concebido para orientar e facilitar o processo de criação de DApps. Este *framework* é baseado no resultado de uma revisão sistemática da literatura (RSL), essa RSL identifica e conceitua as tecnologias mais prevalentes, fundamenta as decisões adotadas, aborda os desafios inerentes a esse tipo de desenvolvimento, cataloga as ferramentas empregadas e detalha suas contribuições para o processo de criação da aplicação descentralizada.

Assim, destacam-se, como contribuições deste trabalho, a concepção do *framework*, juntamente com a exposição fundamentada dos elementos utilizados na criação da aplicação. A relação das ferramentas empregadas e todos os aspectos propostos por este estudo desempenha um papel crucial ao oferecer suporte a indivíduos externos a este domínio, mas que almejam criar uma aplicação baseada em redes *Blockchain*.

De fato, as contribuições alcançadas neste trabalho englobaram o acompanhamento minucioso do estado da arte, a formalização do processo de de-

envolvimento de DApps por meio da criação do *framework* de decisão e uma análise das tecnologias passíveis de serem empregadas ao longo desse trajeto, destacando suas vantagens, desvantagens e limitações.

O trabalho de Wei et al. (2018) traz um *survey* citando alguns pontos fracos em aplicações descentralizadas e em sua conclusão afirmam que DApp é o futuro das *Blockchains*. O que demonstra que esta dissertação está no caminho certo ao buscar essa formalização no desenvolvimento destas aplicações.

A previsão é que as DApps se configurem como o futuro do desenvolvimento de aplicações, contudo, carece-se atualmente de uma formalização abrangente em sua construção. Este trabalho visou suprir essa lacuna ao fornecer um *framework* que não apenas orienta o desenvolvimento, mas também promove uma discussão informada sobre as tecnologias pertinentes.

1.2 Objetivos

Nesta seção é apresentada a finalidade da pesquisa, separada por objetivo geral e objetivos específicos.

1.2.1 Objetivo Geral

O objetivo geral deste trabalho é a criação de um *framework* de decisão para auxiliar no desenvolvimento de DApps, facilitando o processo de criação de aplicações distribuídas com suporte à redes *Blockchain*, indicando a plataforma mais adequada para hospedar a aplicação que será desenvolvida.

1.2.2 Objetivos Específicos

Este trabalho propõe os seguintes objetivos específicos:

- Conceituação das tecnologias mais recorrentes;
- Extração das características essenciais para o desenvolvimento de DApp;
- Extração das plataformas mais utilizadas;
- Criação do *framework* para auxiliar na escolha da plataforma descentralizada.

1.3 Estrutura do Documento

Este trabalho está estruturado da seguinte forma: no **Capítulo 2** é apresentado o referencial teórico com conceitos fundamentais para o tema da pesquisa, assuntos como: *Blockchain* e Aplicações descentralizadas. No **Capítulo**

3 são abordados alguns estudos relacionados ao desenvolvimento de aplicações descentralizadas. O **Capítulo 4** apresenta a revisão sistemática da literatura. O **Capítulo 5** traz os resultados da pesquisa. Por fim no **Capítulo 6** são apresentadas as conclusões.

Fundamentação Teórica

Neste Capítulo são apresentados os conceitos de Sistemas Distribuídos e arquitetura Peer-to-Peer (P2P) que são os conceitos chaves na área de DApps. Na Seção 2.3 é apresentada a *Blockchain* e as tecnologias que a compõem, encerrando a seção de *Blockchain* com uma ilustração de sua estrutura. A seção 2.4 aborda as várias plataformas e suas individualidades. Na seção 2.5 são apresentados os temas interoperabilidade e migração de código, por fim, a Seção 2.6 apresenta a metodologia de revisão sistemática da literatura utilizada no contexto deste trabalho.

2.1 *Sistemas Distribuídos*

Conforme Tanenbaum e Steen (2007), sistemas distribuídos podem ser conceituados como um conjunto de computadores independentes que se apresentam aos usuários finais como um sistema único e coeso. Estes computadores interagem por meio de uma rede de computadores, coordenando suas ações e compartilhando recursos do sistema, como hardware, software e dados. O propósito fundamental de um sistema distribuído reside em proporcionar aos usuários acesso transparente e confiável a recursos compartilhados, independentemente da localização física destes recursos ou dos próprios usuários.

Coulouris et al. (2013) definem os sistemas distribuídos como junção de dois ou mais computadores que podem até estar geograficamente distantes, mas se comunicam, compartilham seus recursos e são interligados por meio da rede.

Sem dúvida, entre as diversas convergências nas definições dos autores,

destaca-se a centralidade da rede. Nesse contexto distribuído, em que a conexão entre sistemas é efetuada pela rede, observam-se duas formas predominantes de organização topológica e de comunicação entre processos: o modelo cliente/servidor e o modelo ponto a ponto (*Peer-to-Peer*) - P2P (Tanenbaum e Steen, 2007) (Coulouris et al., 2013).

No domínio do problema proposto, a organização topológica e de comunicação de processos utilizada é a P2P (Marchesi et al., 2020b). De acordo com Marchesi et al. (2020b), as aplicações que utilizam a *Blockchain* como base, normalmente são P2P devido a natureza da rede, pois, são aplicações distribuídas e que não possuem um ente centralizador.

2.2 Peer-to-Peer

Conforme destacado por Tanenbaum e Steen (2007), a rede P2P é uma arquitetura que sustenta um modelo de distribuição cliente e servidor, permitindo que cada computador (nó) na rede desempenhe ambos os papéis. Em outras palavras, cada nó possui seus próprios recursos e os utiliza para realizar suas operações. A Figura 2.1 ilustra a distinção entre uma rede P2P e um sistema cliente/servidor.

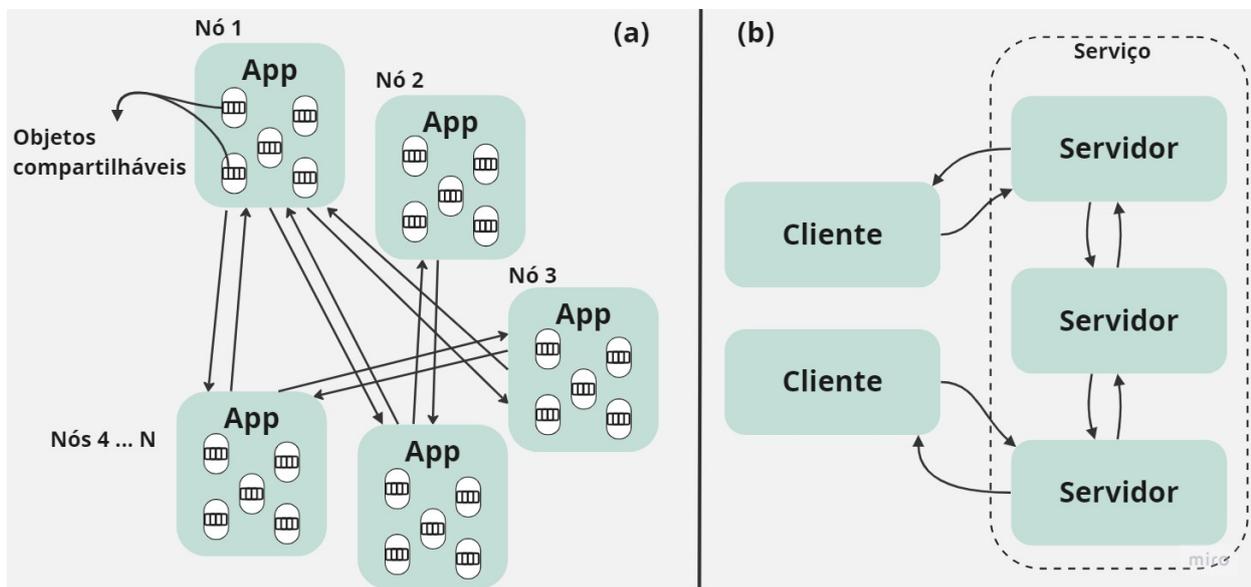


Figura 2.1: Arquitetura *Peer-to-Peer* | Um serviço fornecido por múltiplos servidores (Coulouris et al., 2013).

Na Figura 2.1 (a) é exibida uma ilustração sobre a arquitetura P2P onde é demonstrado que os nós na rede detêm vários pares de processos que estão sendo executados em diferentes computadores e se comunicando cada um a sua maneira seguindo os padrões estabelecidos pela aplicação (Coulouris et al., 2013). A Figura 2.1 (b) é uma arquitetura cliente servidor com múltiplos servidores.

As redes P2P possuem um caracter de ausência de ponto centralizador. O mesmo conceito é utilizado por redes *Blockchain*. Isso se deve ao fato de redes *Blockchain* serem desenvolvidas sobre redes P2P.

Fazendo uso desta arquitetura descentralizada, a *Blockchain* nasceu e será abordada na próxima subseção.

2.3 Blockchain

Concebida com o intuito de amparar a descentralização de moeda digital e suas transações, sua principal tarefa era realizar com segurança atividades que tinham como propósito a transação de moeda digital, o *BitCoin* (Lin et al., 2022b).

Apesar dessa tecnologia estar associada a criptomoedas e consequentemente *Bitcoins*, ela vai muito além (Nofer et al., 2017). Segundo Tijan et al. (2019), *Blockchain* é transparente, descentralizada e segura, pois compartilha com os nós da rede as transações que ocorrem entre seus usuários de forma que todos da rede tenham uma cópia deste registro. Para que essa transação seja efetivada, nós que compõem esta grande rede distribuída precisam validar esta ação. A primeira geração de *Blockchain* é sumarizada na Figura 2.2 (a).

Era 1.0 Blockchain (a)	(b) Era 2.0 Blockchain
<p>Princípios de desenvolvimento: Resolver a descentralização de moeda digital e de seus meios de pagamento.</p> <p>Principais características: contrato básico, criptomoeda digital, função única e alta segurança.</p> <p>Aplicação representativa: Bitcoin.</p>	<p>Princípios de desenvolvimento: Expansão do escopo de aplicação da Blockchain, reduzir custos de confiança e colaboração e melhorar seu desempenho.</p> <p>Principais características: contrato inteligente, mecanismo de consenso, suporte à criptografia de informações, DApp descentralizada e múltiplas áreas de aplicação.</p> <p>Aplicação representativa: 1. Blockchains públicas: Ethereum, TRON, NEO, IOST, etc. 2. Blockchains privadas: Hyperledger, ANTCHAIN, R3 Corda e FISCO BCOS, etc.</p>

Figura 2.2: A história progressiva da *Blockchain* - era 1.0 e 2.0 (Lin et al., 2022b).

A *Blockchain* ganhou vida com a proposição de Nakamoto (2008) e observa-se na Figura 2.2 (a), que sua principal funcionalidade e utilização durante toda a primeira fase foi voltada para realizar com segurança a descentralização do *Bitcoin*.

Pouco tempo após sua concepção a *Blockchain* estava sendo utilizada e financiada pela indústria de Internet das Coisas ou *Internet of Things* (IoT), computação em nuvem e cadeia de suprimentos que perceberam seu grande potencial (Lin et al., 2022b). Com a evolução dos contratos digitais a tecnologia poderia ir além de apenas realizar o controle de transações financeiras e essa evolução se deu por meio dos contratos inteligentes (*Smart Contracts*) (Szabo, 1997).

A Figura 2.2 (b) exibe a evolução da *Blockchain* e as tecnologias agregadas a ela em sua versão 2.0. Sua popularização após a criação dos contratos inteligentes, e a expansão de seu escopo de aplicação ao utilizar algoritmos de consenso e criptografia. Com essa expansão, surgiram as *Blockchains* públicas e privadas.

2.3.1 *Blockchain Pública e Privada*

Com o intuito de suprir demandas diferentes, ambas tecnologias se utilizam das mesmas premissas, são elas: rede descentralizada, transação sem interferência de terceiros e a propagação para todos os nós pertencentes a rede (Yakubu et al., 2023).

Yang et al. (2020) define em sua pesquisa que uma *Blockchain* privada tem uma alta taxa de processamento e, conseqüentemente, o número de transações finalizadas por ela é maior que em redes públicas. As redes privadas exigem que os nós que a integram estejam autenticados na rede para poderem utilizar o sistema, facilitando o consenso na rede.

Em uma *Blockchain* pública existe um número maior de nós e cada conjunto de nós que realizam uma transação precisa ser averiguado antes de ter sua transação concretizada. Posteriormente, o estado atual deve ser enviado para todos os nós pertencentes a rede, aumentando o tempo de processamento e, com isso, diminuindo o número de transações efetivadas em um espaço de tempo (Yakubu et al., 2023).

Outra característica importante é que em uma *on-chain* (rede pública) existe uma rede mais descentralizada e cada transação ocorrida pode ser verificada por aqueles que compõem a rede. Uma *off-chain* (rede privada) é menos descentralizada e apenas os nós de confiança da rede podem verificar e validar as transações (Yang et al., 2020).

2.3.2 *Smart Contracts*

Szabo (1997) demonstra em sua pesquisa que *smart contracts* permitem que relações entre redes de computadores possam ser seguras e o fazem utilizando protocolos, criptografia e outros mecanismos de segurança.

Os contratos inteligentes projetados por Szabo (1997) tem o objetivo de reduzir os custos mentais e computacionais facilitando as etapas de um contrato sem a necessidade de terceiros e realizando uma transação rastreável e irreversível.

Chen et al. (2023) cita que contratos inteligentes são programas que se executam na *Blockchain* e que atualmente sua aplicação não se limita aos domínios financeiros. Os autores enfatizam que a elaboração destes contratos é uma tarefa difícil, demandando dos desenvolvedores um profundo entendimento do funcionamento da rede descentralizada que o hospedará.

Os contratos inteligentes são *scripts* e um exemplo é apresentado no Algoritmo 2.1.

```
1 /pragma solidity 0.8.7;
2
3 contract VendingMachine {
4
5     // Declaracao e estado das variaveis do contrato
6     address public owner;
7     mapping (address => uint) public cupcakeBalances;
8
9     // Quando o contrato "VendingMachine" e implementado:
10    // 1. define-se o endereco de implantacao como sendo o
11    //    proprietario do contrato
12    // 2. definir o saldo de cupcake do contrato inteligente
13    //    implementado para 100
14    constructor() {
15        owner = msg.sender;
16        cupcakeBalances[address(this)] = 100;
17    }
18
19    // Permitir que o proprietario aumente o saldo de cupcakes do
20    //    contrato inteligente
21    function refill(uint amount) public {
22        require(msg.sender == owner, "Apenas o proprietario pode
23            aumentar.");
24        cupcakeBalances[address(this)] += amount;
25    }
26
27    // Permitir que qualquer pessoa compre cupcakes
28    function purchase(uint amount) public payable {
29        require(msg.value >= amount * 1 ether, "Necessario pagar pelo
30            menos 1 ETH por cupcake");
31        require(cupcakeBalances[address(this)] >= amount, "Nao ha
```

```

    cupcakes suficientes em estoque para completar esta compra
    ");
27     cupcakeBalances[address(this)] -= amount;
28     cupcakeBalances[msg.sender] += amount;
29 }
30 }
31 }

```

Algoritmo 2.1: Máquina de venda automática (Wackerow, 2022)

O Algoritmo 2.1 é um exemplo disponibilizado no site da plataforma *Ethereum* (Wackerow, 2022) e se trata da criação de uma loja de venda de *cupcakes*, onde inicialmente são definidas as variáveis do contrato e duas funções. Uma função realiza a reposição de estoque de *cupcakes*, reposição esta que apenas o dono do contrato pode fazer. A outra função (*purchase*) é para efetivar a transação, momento em que o contrato inteligente exige que o comprador tenha pelo menos 1 ether¹ e que no estoque tenha a quantidade solicitada.

Contrato inteligente é um modelo de contrato que utiliza uma rede *Blockchain* para interligar contratante e contratado, promovendo segurança e confiabilidade nas transações financeiras (Giancaspro, 2017). Não havendo necessidade da chancela de uma organização centralizadora, basta que as partes envolvidas satisfaçam os critérios pré-estabelecidos no contrato. Então, ele será concretizado e o registro desta transação adicionado à *Blockchain* após validação pelos nós que fazem parte da rede.

A validação é uma ação muito importante e ocorre por consenso dentro da rede *Blockchain*, seja ela pública ou privada (Giancaspro, 2017).

2.3.3 Algoritmos de Consenso

Os algoritmos de consenso são uma camada importante dentro da *Blockchain*. Estes algoritmos conectam os nós pertencentes a rede ao criar e atualizar os dados manipulados por ela. Após garantir que os dados são únicos e íntegros, eles são compartilhados com toda rede para sua validação (Xiong et al., 2022).

O processo de validação se dá por meio de um sistema que inicialmente foi criado para evitar ataques por envio e postagem de anúncios em massa (SPAM). O *Proof of Work* (Pow) hoje possui versões diferentes para satisfazer as necessidades das mais variadas plataformas descentralizadas. A prova de trabalho (PoW) funciona recompensando com tokens àquelas redes que dispenderam seu poder computacional para validar uma transação. Esse algoritmo é o coração do *Bitcoin* e de outras criptomoedas (Back, 2002).

¹<https://ethereum.org/pt/eth/>

A Figura 2.3 apresenta o funcionamento desse algoritmo que é parte da base para a arquitetura *Blockchain*.

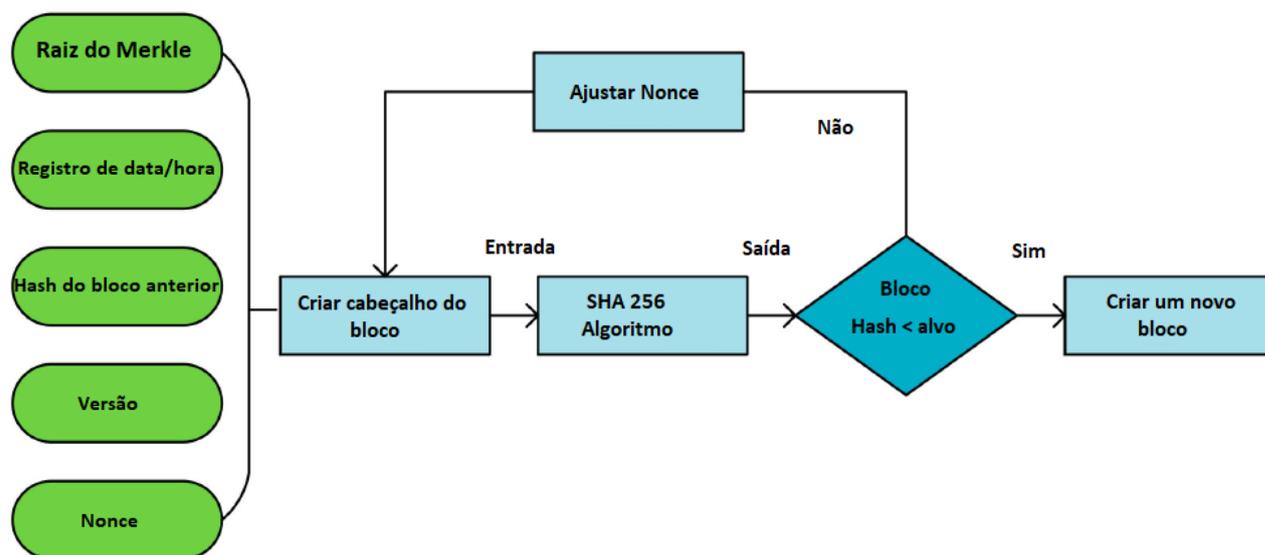


Figura 2.3: O princípio de funcionamento do algoritmo PoW (Xiong et al., 2022).

A Figura 2.3 demonstra a trajetória para se realizar a criação de um novo bloco na *Blockchain*. Para isso, é preciso ter o *hash* do primeiro bloco da árvore *Merkle*, data e hora da criação do bloco, valor do *hash* do bloco anterior, a versão do algoritmo e o poder computacional do nó (Xiong et al., 2022).

Outro tema relevante no âmbito dos algoritmos de consenso é o conceito de Tolerância a Falhas Bizantinas (BFT), do qual outros algoritmos derivam. Seu principal objetivo é permitir que o sistema sobreviva e continue a funcionar apesar de algumas máquinas apresentarem falhas arbitrárias (Muratov et al., 2018).

Alguns exemplos dos diferentes algoritmos de consenso são:

Practical Byzantine Fault-Tolerance (PBFT): é um dos algoritmos de consenso mais antigos e busca assegurar a estabilidade do sistema distribuído por meio de uma organização onde existem o nó mestre e os outros nós. Após selecionado pelo sistema o mestre deve processar as informações enviadas pelos nós e validá-las (Xiong et al., 2022).

Proof of Stake (Pos): segundo Yang et al. (2020) a escolha do vencedor se dá de forma aleatória e depende da quantidade de ativos (aquele com maior participação, ou seja, mais tokens) que ele possui.

Delegated Proof of Stake (DPOS): segundo Xiong et al. (2022) o algoritmo de prova de participação delegada, agrega vantagens encontradas nos algoritmos PoW e PoS. Ele atribui o poder de escolha dos nós que processarão a transação a cargo de uma votação onde a quantidade de votos que cada nó possui é sua quantidade de moedas.

Existem outros algoritmos de consenso, como: **Proof of Weight** (PoWeight); **Leased Proof of Stake** (LPoS); **Proof of Burn** (PoB); **Simplified Byzantine Fault Tolerance** (SBFT) (Yang et al., 2020).

O token desempenha um papel crucial nesse mecanismo como já mencionado nesta seção. O token é concedido como uma forma de recompensa aos nós pelo esforço computacional dispensado, além de ser determinante na seleção do nó mestre (Yang et al., 2020).

Token

Segundo Chen (2018) a tecnologia *Blockchain* impulsionou o empreendedorismo e a inovação ao viabilizar a representação de ativos escassos, por meio de *Tokens*, proporcionando assim a participação de novas pessoas e a captação de recursos.

O *Token* não se limita à representação de moedas, abarcando, de maneira mais ampla, uma variedade de ativos, tais como títulos, propriedades e pontos de fidelidade, entre outros. E podem ser transferidos entre as partes envolvidas na transação sem a interferência de um órgão centralizados (Chen, 2018).

Segundo (Xu et al., 2019), ao contrário de uma moeda digital, é importante observar que o *Token* não é nativo da *Blockchain*. Ele opera como uma evidência criptografada de direitos digitais, sendo capaz de representar uma diversidade de ativos, como identidade, diplomas acadêmicos, ingressos para eventos e chaves. Esta versatilidade demonstra o poder intrínseco do token ao capacitar a digitalização de uma ampla gama de direitos existentes na sociedade Xu et al. (2019).

Com a integração de todas essas tecnologias, como blockchain, algoritmos de consenso, contratos inteligentes e tokens, emergem as aplicações descentralizadas.

2.3.4 Decentralized Application (DApp)

Baseada em uma rede *BlockChain* e fazendo uso de contratos inteligentes, as aplicações descentralizadas são *softwares* de código aberto, transparentes e auditados com facilidade. Por ser descentralizada, seus dados e validações necessárias são providos pelos nós que compõem a rede (Yang et al., 2019).

Na Figura 2.4 é exibida a diferença entre uma aplicação centralizada e descentralizada.

A Figura 2.4(a) demonstra o modelo comum de criação de *software* utilizando um servidor que deve prover acesso, dados e funcionalidades para aqueles que queiram utilizar a aplicação. A Figura 2.4(b) exhibe uma aplicação descentralizada onde seus usuários utilizando o sistema podem ser servidores e em outro momento clientes (Yang et al., 2019).

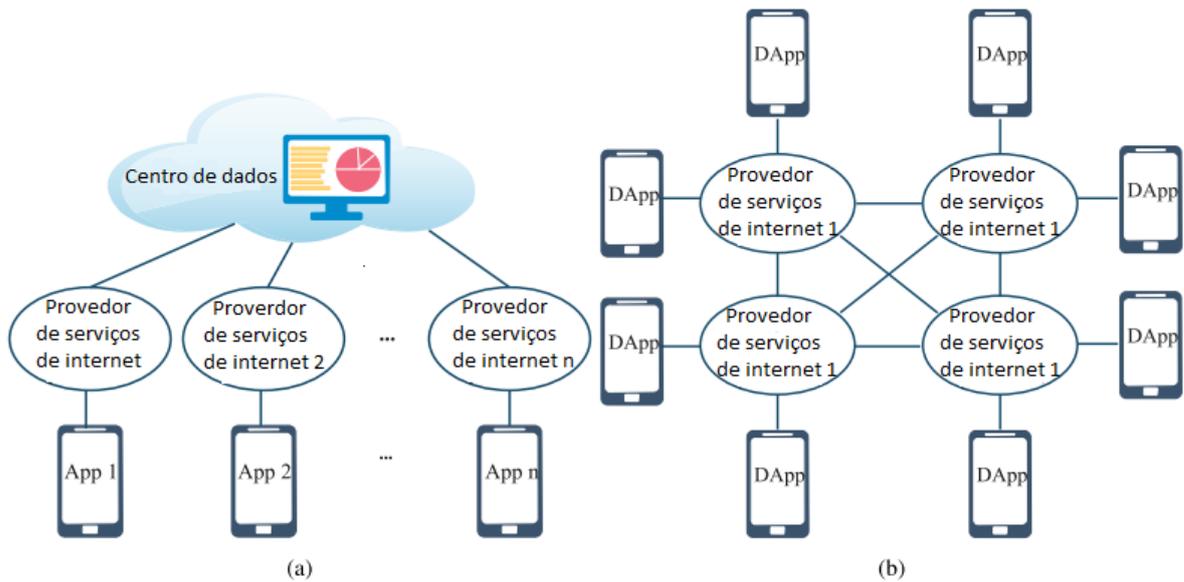


Figura 2.4: Comparação entre as aplicações. (a) Aplicação centralizada. (b) Aplicação descentralizada (Yang et al., 2019).

Segundo Marchesi et al. (2020b), contratos inteligentes rodam sobre uma rede *Blockchain* e sua comunicação fica restrita dentro da rede. E eles não podem trocar mensagens de maneira direta com nenhuma outra aplicação externa. Portanto, aplicações descentralizadas baseadas em redes *Blockchain* podem ser divididas em dois grupos, aquelas que executam apenas dentro da rede e as que se utilizam de interfaces com as quais é permitida a interação com usuários e dispositivos externos à rede.

A Figura 2.5 demonstra de forma simplificada a estrutura de uma DApp que utiliza a *Blockchain* da Oracle².

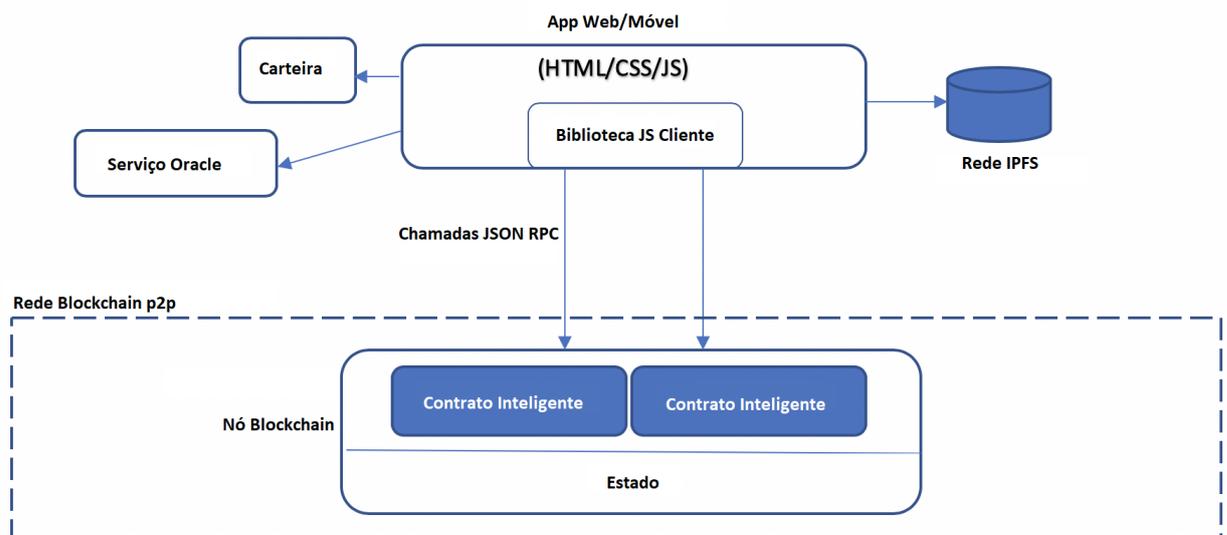


Figura 2.5: Arquitetura de uma DApp na rede Oracle (Baranwal, 2019).

Na Figura 2.5 observa-se algumas tecnologias que muitas vezes são utili-

²<https://www.oracle.com/blockchain/>

zadas para o desenvolvimento de DApps, mas que não são debatidas dentro das pesquisas, ou seja, o autor apenas cita a utilização de uma técnica ou tecnologia sem se preocupar em discorrer brevemente a respeito de outras alternativas que poderiam ser utilizadas para aquele propósito.

A Figura 2.5 demonstra que fazendo uso de serviços oferecidos pela *Oracle* utilizando Linguagem de marcação de hipertexto (HTML), Folhas de estilo em cascata (CSS) e JavaScript (JS) pode-se criar uma interface *Web* ou *mobile* para interação com o usuário. Essa interface de usuário ou *User Interface* (UI) interage com um sistema de arquivos e faz chamadas à *Blockchain* utilizando *JSON*. A Figura 2.5 apresenta alguns nomes importantes para a rede, mas que ainda não foram conceituados. São eles:

- **JSON (Notação de objetos JavaScript):** O JSON atua auxiliando a comunicação entre aplicações diferentes que apesar de não se “conhecerem” necessitam estabelecer uma conexão para realizar troca de mensagens (Bassett, 2015);
- **UI (Interface de Usuário):** É o meio pelo qual o usuário utiliza as funções disponíveis em uma aplicação (Lowdermilk, 2013);
- **Wallet:** Cada usuário possui sua chave que é utilizada como forma de assinatura de autenticidade em suas transações e esta chave precisa ficar armazenada em segurança dentro de uma carteira digital (*wallet*) (Baranwal, 2019);
- **IPFS:** segundo Benet (2014) o *InterPlanetary File System* (IPFS) é um sistema de arquivos distribuído para uma rede P2P, com o propósito de armazenar e compartilhar arquivos, onde todos os nós da rede são iguais e apenas aqueles que aceitarem, armazenarão os dados.

A Figura 2.6 traz a arquitetura para construção de uma DApp para a plataforma *Ethereum*.

A Figura 2.6 demonstra a arquitetura de aplicações descentralizadas desenvolvidas na plataforma da *Ethereum*. A esquerda em “*Ethereum Blockchain*” é exibido em destaque ao centro a composição de um *peer*: contrato inteligente, armazenamento e a máquina virtual da *Ethereum* (EVM).

O lado direito da Figura 2.6 exhibe o “sistema de aplicação” que é onde ficam armazenados dados que normalmente são utilizados em execução de cálculos comerciais e não podem ficar na *Blockchain*. “Interface de usuário que possibilita a interação do usuário com a DApp”, é uma aplicação que permite o acesso externo de clientes por meio de uma interface mais amigável e acessada pelo *browser*.

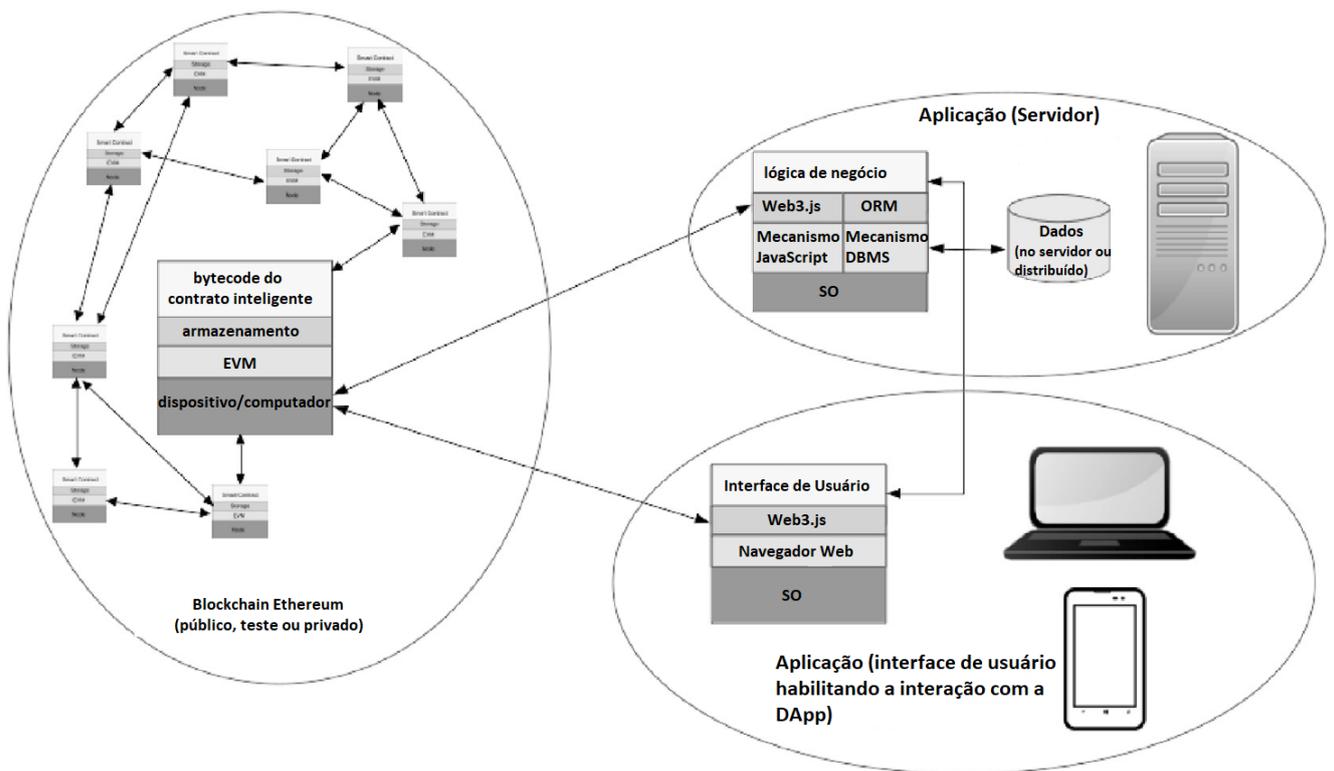


Figura 2.6: Arquitetura de uma aplicação descentralizada baseada na plataforma Ethereum (Marchesi et al., 2020b).

2.3.5 Infraestrutura da Blockchain

Após todo o exposto neste capítulo a Figura 2.7 traz um resumo que pode unir e diagramar todos os conceitos vistos, montando assim a infraestrutura de uma *Blockchain*.

A Figura 2.7 exhibe a infraestrutura de uma *Blockchain*, na camada mais baixa temos a de rede e suas principais características: a encriptação, armazenamento de dados e a segurança das transações.

A *Network* tem como seu objetivo principal proporcionar a troca de informação entre os nós pertencentes a rede, disseminando e verificando os dados que são trocados.

Na camada de consenso são utilizados diferentes tipos de algoritmos como os vistos nesta seção. A camada de incentivo é onde se implementa como serão recompensados aqueles que ajudam a *Blockchain* a garantir sua segurança.

Camada de contrato é onde ficam *scripts*, algoritmos e os contratos inteligentes. Por fim a camada de aplicação é onde residem as interfaces de usuário.

Por este ser um tema relativamente novo, que envolve conceitos densos de áreas diferentes e com o intuito de realizar uma pesquisa embasada e ampla. É importante fazer um bom levantamento do atual estado da arte e para isso foi adotado um método científico e sistemático.

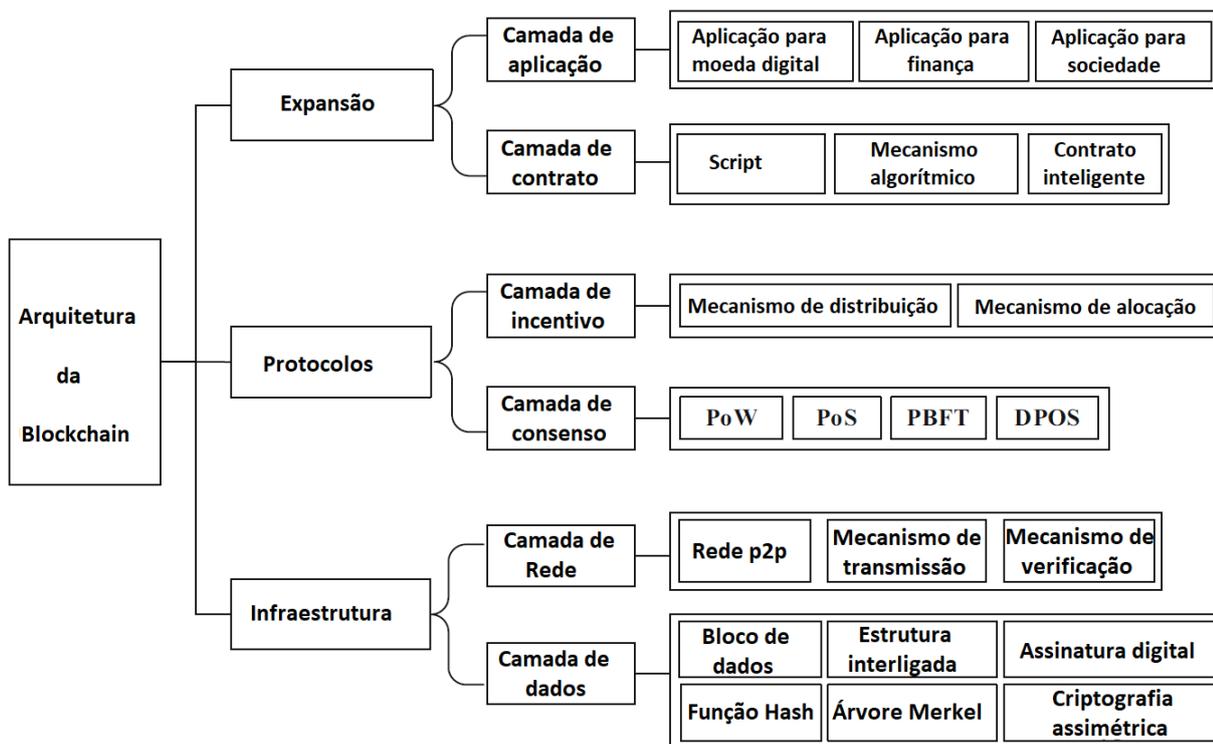


Figura 2.7: A infraestrutura de uma *Blockchain* (Lin et al., 2022b)

Um dos grandes responsáveis pelo sucesso da blockchain. Os contratos inteligentes são outro avanço fundamental, agindo como acordos autoexecutáveis armazenados na rede (Giancaspro, 2017).

Apresentados os conceitos fundamentais que permeiam o desenvolvimento de DApps, é necessário aprofundar-se na área com o propósito de conceber um *framework* eficaz que verdadeiramente beneficie os desenvolvedores. Nesse sentido, a próxima seção explorará conceitos e relacionará as plataformas descentralizadas mais proeminentes.

2.4 Plataformas Blockchain

Taherdoost (2023) realizou uma revisão da literatura sobre o avanço da blockchain e dos contratos inteligentes, em sua revisão o autor analisou 252 artigos publicados entre os anos de 2018 e 2022 e em resposta a uma de suas questões de pesquisa “o quão significantes são os contratos inteligentes para a tecnologia de Blockchain?”, demonstrou a importância desta escolha.

Taherdoost (2023) cita que “são necessários programadores especializados em Blockchain para ajudar os clientes a identificarem a melhor plataforma e o melhor método para desenvolver e implementar os contratos inteligentes de acordo com as necessidades da sua organização”. O autor criou a Tabela 2.1 com vantagens e desvantagens das 6 plataformas mais citadas em sua revisão.

Tabela 2.1: Várias plataformas de contratos inteligentes: prós e contras. (Taherdoost, 2023).

Plataforma	Vantagens	Desvantagens
<i>Ethereum</i>	Acesso a vários recursos Regras claras para os programadores Linguagem de programação própria Token padrão Ethereum Configuração gratuita	Contratos podem ser pirateados devido a codificação de má qualidade Mais caro do que outras plataformas Problemas de segurança com o código Rede sobrecarregada
<i>Hyperledger Fabric</i>	Habilitação de plug-in Desempenho confiável Codificação de contratos em várias linguagens Adesão com permissão Gratuito e de código aberto	Sem sistema de token
<i>NEM</i>	Desempenho excepcional Escalabilidade Linguagem de prog. independente da plataforma Simples de utilizar	Emprega codificação non-blockchain, tornando-o menos descentralizado Ferramentas menos acessíveis Base de programadores menor
<i>STELLAR</i>	Excelente desempenho Plataforma simples Muito respeitado no sector Mais barato que o Ethereum	Inadequado para o desenvolvimento de contratos inteligentes sofisticados
<i>Waves</i>	Adequado para vendas colectivas A criação de tokens requer conhecimentos básicos	Plataforma não versátil Base de utilizadores pequena
<i>Corda</i>	Proteção da privacidade a longo prazo Suporte para nós de regulamentação e supervisão Uma vasta gama de compatibilidades industriais Possibilidade de execução contratual realista Suporte a vários mecanismos de consenso	Verificado apenas por nós de confiança Sem criptomoeda nativa

Na Tabela 2.1, a plataforma da Ethereum está em primeiro lugar e tem como principais vantagens a possibilidade do programador acessar vários recursos e sua robusta linguagem própria para escrita de contratos inteligentes, por oferecer mais, a plataforma tem como desvantagem o valor cobrado, sobrecarga na rede devido a grande quantidade de aplicações hospedadas e, devido a novidade no desenvolvimento de DApps, sua complexidade e falta de experiência dos programadores acabam criando aplicações com codificação de má qualidade, com problemas de segurança o que aumenta suas chances de serem pirateados. *Hyperledger Fabric*, a segunda plataforma mais utilizada, além de disponibilizar componentes e ter um bom desempenho, permite que o desenvolvedor crie seu contrato inteligente utilizando linguagens já consolidadas no mercado.

Taherdoost (2023) faz uma lista com as 50 pesquisas analisadas por ele, que mais tem citações (a primeira com 404 e a última 23 citações) e as áreas mais abordadas são nesta ordem: saúde, estudo de potencial da tecnologia, cadeia de fornecimento, autenticidade, privacidade e segurança, energia, direitos e compartilhamento de dados e pagamento.

Sharma et al. (2023) escreveram uma revisão sobre os desafios e as aplicações de contratos inteligentes e plataformas. Foram analisados 183 trabalhos e os autores descrevem cada uma das plataformas mencionadas em sua pesquisa e criam uma tabela em que as cinco primeiras plataformas são as mesmas citadas no trabalho de Taherdoost, como pode-se observar na Tabela

2.2.

Tabela 2.2: Plataformas de contratos inteligentes Sharma et al. (2023).

Plataforma	Token	Consenso	Tipo de Rede	Linguagem	Aplicação
Ethereum	ETH	PoW e PoS	Ambos	Solidity e Viper	Geral
Hyperledger Fabric	não	PBFT	Privada	Java e Golang	Geral
Nem	XEM	PoA	Ambos	Java	Moeda Digital
Corda	Não	Raft	Privado	Kotlin	Moeda Digital
Stellar	Lumen(XLM)	Stellar Consenso	Ambos	Python, Javascript Golang e PHP	Moeda Digital
Waves	Waves	LPoS	Público	RIDE	Moeda Digital
Cardano	ADA	Ouroboros, PoS	Público	Plutus	Moeda Digital
Neo	NEO e GAS	dBFT	Público	Java, Python e C#	Geral
EOS	EOS	DPoS	Público	C++	Geral
RSK	RBTC	BFT-DPoS	Público	Solidity	Geral
Tendermint	Não	BFT	Privado	Qualquer Linguagem	Geral
Quorum	Não	Raft e IBFT	Privado	Solidity	Geral

A Tabela 2.2 traz muitas informações de maneira resumida e indica se a plataforma dispõe de um *token* nativo e qual é seu nome, qual algoritmo de consenso é adotado, se é uma rede que necessita de um cadastro prévio e quais são as linguagens suportadas para criação dos contratos inteligentes.

Na investigação conduzida por Sharma et al. (2023), são apresentadas análises mais detalhadas sobre as plataformas mencionadas na Tabela 2.2, bem como os tópicos abordados nas pesquisas resumidas pelos autores. As áreas de interesse mais frequentemente citadas compreendem: seguros, cadeia de abastecimento, Internet das Coisas (IoT), cuidados de saúde, multimídia, computação em nuvem, gestão de identidade e gestão de registros.

Khan et al. (2023) realizaram uma revisão crítica das principais plataformas para desenvolvimento de aplicações descentralizadas e resumidamente seu trabalho trouxe mais informações sobre estas tecnologias e foram sintetizadas na Tabela 2.3.

Tabela 2.3: Principais diferenças entre as Plataformas (Khan et al., 2023).

Plataforma	Início	Foco no setor	Suporte	Capacidade de processamento
Ethereum	2015	Criptomoeda	Comunidade Ethereum	+ - 100s
Hyperledger Fabric	2017	Cadeia de Suprimentos	Fundação Linux	> 200 tps
MultiChain	2015	Setor financeiro	Coin Sciences	< 100 tps
Quorum	2016	Setor financeiro	JPMorgan Chase	< 100 tps
Corda	2015	Setor financeiro	R3 Consortium	< 100 tps

Na Tabela 2.3 é possível visualizar informações sobre sua potencialidade em número de transações por segundo (tps), que a organização mantém o funcionamento da rede e, outro ponto importante, é a área da indústria em que a plataforma está focando no momento.

A plataforma da *Ethereum* é a que se destaca entre suas concorrentes, ela foi lançada em 2015 e opera como um alicerce para aplicações descentrali-

zadas de propósito geral. A *Ethereum* oferece a qualquer indivíduo a possibilidade de acessar as aplicações que estão em execução nela e, igualmente, construir suas próprias aplicações descentralizadas. A rede disponibiliza vários serviços como: financeiros, jogos, redes sociais e outras aplicações (Ethereum, 2023b).

2.4.1 *Ethereum*

Segundo o próprio site da plataforma (Ethereum, 2023b), a *Ethereum* é controlada pelos nós que a compõem e para fazer parte da rede basta utilizar seu *software* que segue o protocolo da plataforma.

Parizi et al. (2018) identificam em sua pesquisa que a plataforma Ethereum e a linguagem *Solidity* são as mais utilizadas por novos programadores de contratos inteligentes e ressalta que a linguagem é estaticamente tipada, possui uma sintaxe parecida com a do *JavaScript* e tem grande influência do C++, *Python* e *JavaScript*.

Segundo o site ³ da Ethereum, a Linguagem de Programação é:

- *De alto nível;*
- *Orientada à objetos;*
- *Destinada para implementação de contratos inteligentes;*
- *Influenciada pelo C++;*
- *Tipada estaticamente.*

O autor Wackerow (2023), conclui que a plataforma conta com duas linguagens: *Solidity* e *Vyper*. O *Vyper* é melhor para desenvolvedores acostumados a desenvolver utilizando a linguagem *Python*, dispõe de um número menor de funcionalidades e seu código é fácil para ser auditado. *Solidity* tem muitas ferramentas disponíveis, tem uma grande comunidade de desenvolvedores e dispõe de muitos tutoriais e ferramentas para auxiliar na aprendizagem.

A EVM que parte fundamental na arquitetura de uma DApp baseada na plataforma da Ethereum e pode ser vista na Figura 2.6. Segundo Rakita (2023) “Esta tecnologia existe como uma entidade única mantida por milhares de computadores ligados que executam um cliente Ethereum e é o ambiente em que todas as contas Ethereum e contratos inteligentes vivem”.

A máquina virtual da Ethereum é adotada por outras plataformas. Westerkamp (2019) relaciona em sua pesquisa plataformas *Blockchain* que utilizam a EVM:

³(Wackerow, 2023)

- *Hyperledger Fabric*⁴, *Burrow*⁵ e *Sawtooth*⁶;
- *Quorum*⁷;
- *Qtum*⁸;
- *Counterparty*⁹.

A *Ethereum* (Solidity, 2023) se utiliza de um mecanismo para mensurar o esforço despendido para realizar as tarefas solicitadas durante o uso das DApps e a nomeia como 'Gas'. O 'Gas' funciona como uma recompensa ao trabalho de mineradores, que são o coração da rede, eles se utilizam de seu grande poder computacional e processam as transações e produzem os blocos que serão adicionadas a cadeia (Cook, 2023), e de investidores que são usuários da rede que por manterem suas criptomoedas nela, podem participar do processo de validação das transações (Ethereum, 2023a).

Para cada transação realizada pela rede é cobrada quantidade de gas que é calculada de acordo com esforço despendido para aquela ação. Essas transações desencadeiam uma série de verificações, cálculos e chamadas à rede e cada uma destas pequenas e específicas operações tem seu custo (Solidity, 2023). Se durante a execução de uma transação a quantidade de gas do autor se esgotar, um processo de *rollback* das alterações é desencadeado, mas o gas utilizado com estas operações não são reembolsados, tornando a rede invulnerável a um possível erro de loop infinito (Solidity, 2023).

A Figura 2.8 exibe momentos durante a execução de uma tarefa solicitada pelo cliente.

Analisando a Figura 2.8 é possível perceber o primeiro passo que é avaliação do consumo de gas da transação. Ao acessar o contrato inteligente e a máquina virtual da *Ethereum* são consumidas as primeiras unidades de gas. Durante a realização das operações necessárias para a realização da transação, dados são movidos entre a pilha, a memória e o local de armazenamento (ações que consomem mais gas). Finalizando a transação, é enviada uma mensagem com o *status* da transação e são cobrados mais gas (Smith, 2023).

Numerosas pesquisas abordam o tópico do consumo de gas. Wang et al. (2020) realizaram uma revisão sistemática sobre segurança dos contratos que executam na rede *Ethereum* e como eles podem ser explorados. Os autores citam mecanismos que ajudam o desenvolvedor a prevenir alguns ataques

⁴<https://github.com/hyperledger/fabric-chaincode-evm>

⁵<https://github.com/hyperledger-archives/burrow>

⁶<https://www.hyperledger.org/projects/sawtooth>

⁷<https://www.jpmorgan.com/global/Quorum>

⁸<https://qtum.org/en>

⁹<https://counterparty.io/docs/faq-smartcontracts/>

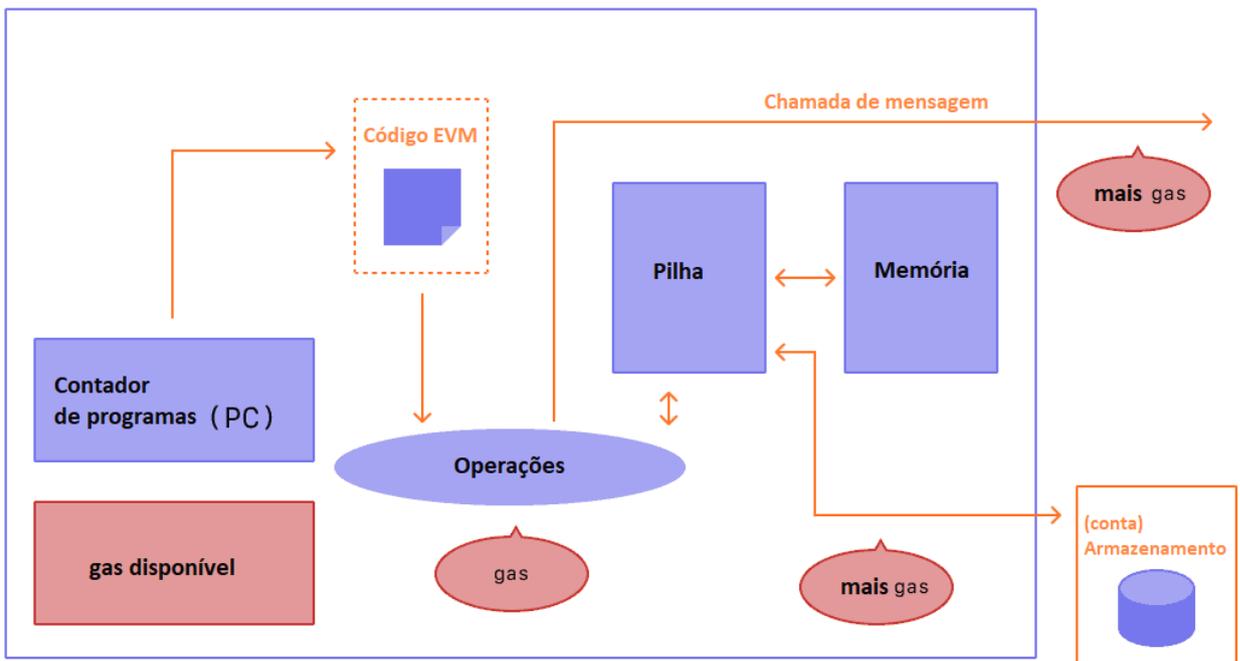


Figura 2.8: Diagrama de uso de gas (Smith, 2023)

de *hackers*, mas focam no tipo de ataque chamado como Negação de Serviço (*Denial of service* (DoS)).

Os pesquisadores Liao et al. (2023) apresentam um *framework* capaz de identificar a vulnerabilidade de reversão de estado (*State-reverting Vulnerability* (SRV)) e demonstra resultados da aplicação deste *framework* em contratos reais que estão em execução.

Já Qian et al. (2023) realizaram uma investigação sobre dois tipo de ataques denominados “números randômicos” e “reversão de transação” e criam um algoritmo eficaz na detecção e tratamento destes ataques.

Marchesi et al. (2020a) citam vários tipos de vulnerabilidades que frequentemente são exploradas. Explicam sobre o consumo de ‘gas’, e quanto esses ataques são nocivos e fazem a proposição de 24 padrões de projeto divididos em 5 categorias. Estes padrões podem auxiliar os desenvolvedores de contratos inteligentes a fazê-lo de uma forma mais segura.

Diante da compreensão da complexidade ao se desenvolver uma DApp, assim como, das singularidades das plataformas e tecnologias que adotam. A próxima seção abordará temas cruciais, tais como, interoperabilidade de aplicações e migração de código.

2.5 Interoperabilidade e Migração de Contratos Inteligentes

Westerkamp (2019) aborda em sua pesquisa um tema muito interessante

que é a portabilidade de contratos inteligentes. Nele o pesquisador criou um conceito a ser seguido para realizar a migração de um contrato inteligente com maior facilidade entre as plataformas descentralizadas que suportam a máquina virtual da *Ethereum*. O trabalho trata formas diferentes de realizar esta migração de acordo com o tamanho da aplicação a ser migrada.

A pesquisa de Fajge et al. (2021) traz uma solução interessante para quem está começando a desenvolver DApps. Os autores criaram um *framework* que automaticamente realiza a migração de uma aplicação centralizada criada com a linguagem Java, para aplicações descentralizadas baseadas na plataforma da *Ethereum*, com a linguagem *Solidity*.

Monika e Bhatia (2020) trabalharam para identificar soluções utilizadas para realizar a interoperabilidade de contratos inteligentes entre diferentes plataformas e categorizar esses achados. Os autores nortearam sua pesquisa buscando respostas para as seguintes problemáticas:

- Interação entre sistemas legados e uma plataforma *Blockchain*;
- Interação entre plataformas *Blockchain* diferentes;
- Interação entre contratos inteligentes.

Os achados foram categorizados sob os seguintes títulos:

- Sistemas notariais;
- *Sidechains*;
- Atomicidade de transações entre usuários de diferentes blockchains;
- Criação de pontes e rotas entre diferentes.

Os autores conseguiram encontrar mais de cinco soluções para cada um desses grupos de problema e elencaram lacunas que foram sumarizadas na Tabela 2.4.

Na Tabela 2.4 são citados os problemas comuns em um desenvolvimento DApp que precisa desta interoperabilidade, e após a pesquisa os autores apresentam as lacunas existentes.

A pesquisa de Nissl et al. (2021) propõe um método para realizar chamadas entre contratos inteligentes que estão sendo executados em redes diferentes. Os autores realizaram testes que obtiveram êxito em realizar essa colaboração entre contratos que estão hospedados em diferentes plataformas. Para trabalhos futuros se propuseram a estudar uma forma de reduzir o custo com estas chamadas que são efetuadas.

Khan et al. (2021) citam em sua pesquisa a busca por preencher a lacuna de interoperabilidade entre contratos inteligentes. Os autores categorizam

Tabela 2.4: Lacunas Das Soluções De Interoperabilidade (Monika e Bhatia, 2020)

Interoperabilidade Tipo de solução	Lacuna
Sistemas notariais	Ponto único de falha, altamente complexo O estado trocado entre dois ledgers é armazenado em três lugares Pressupõe que a maioria dos participantes são honestos Tende a centralização devido à presença de TTP (protocolo)
<i>Sidechain</i>	Implementa a transferência de ativos com relação de 1 para 1 Centra-se em sistemas homogêneos de <i>Blockchains</i> A pegada bidirecional federada sofre de centralização política Custo computacional mais elevado e altamente complexo
Contratos Inteligentes	Pouco explorado para implementar as soluções interoperáveis O compartilhamento não está disponível
Soluções de Ponte	Poucas soluções de troca de ativos; Apenas transferência de ativos Custo computacional elevado
Roteador de <i>Blockchains</i>	As soluções propostas ainda não estão implementadas A estrutura de nós da plataforma precisa de ser alterada para que possa atuar como um Roteador.

tipos diferentes de contratos e redes, e detalham outras pesquisas que se propuseram a preencher essa lacuna. Em um dos cenários explorados pelos autores, eles demonstram uma solução utilizando transações criptografadas entre as *blockchains* e como gerenciador desta transação um contrato inteligente hospedado em outra rede. Ao final de seu trabalho os autores concluem que ainda existe pouco material que investigue uma solução para esse problema, mas citam possíveis soluções que utilizam invocação de contratos remotos por virtualização.

O capítulo apresentou até o momento, teorias que permeiam o desenvolvimento de DApps. Estes temas constituem os fundamentos essenciais para o entendimento das aplicações descentralizadas. A próxima seção encerra esse capítulo conceituando RSL.

2.6 Revisão Sistemática da Literatura (RSL)

Segundo Kitchenham e Charters (2007), a RSL é um estudo secundário que, por meio de um protocolo bem definido, auxilia o pesquisador a realizar sua busca, análise e interpretação dos estudos primários que servirão de base para o seu estudo. É uma investigação utilizada quando se tem a intenção de resumir as pesquisas existentes e, com isso, identificar possíveis lacunas que possam ser exploradas por futuros trabalhos.

A RSL está fortemente associada a etapa de apropriação do conhecimento sobre o tema escolhido para sua pesquisa, buscar na literatura o que está sendo estudado, quais são os pontos que podem ser explorados (lacunas) e auxilia na compreensão e refinamento das questões de pesquisa (Almeida et al.,

2020).

Na Figura 2.9 é demonstrado o processo que deve ser seguido para se realizar uma revisão sistemática da literatura, conforme o trabalho de Kitchenham e Charters (2007).

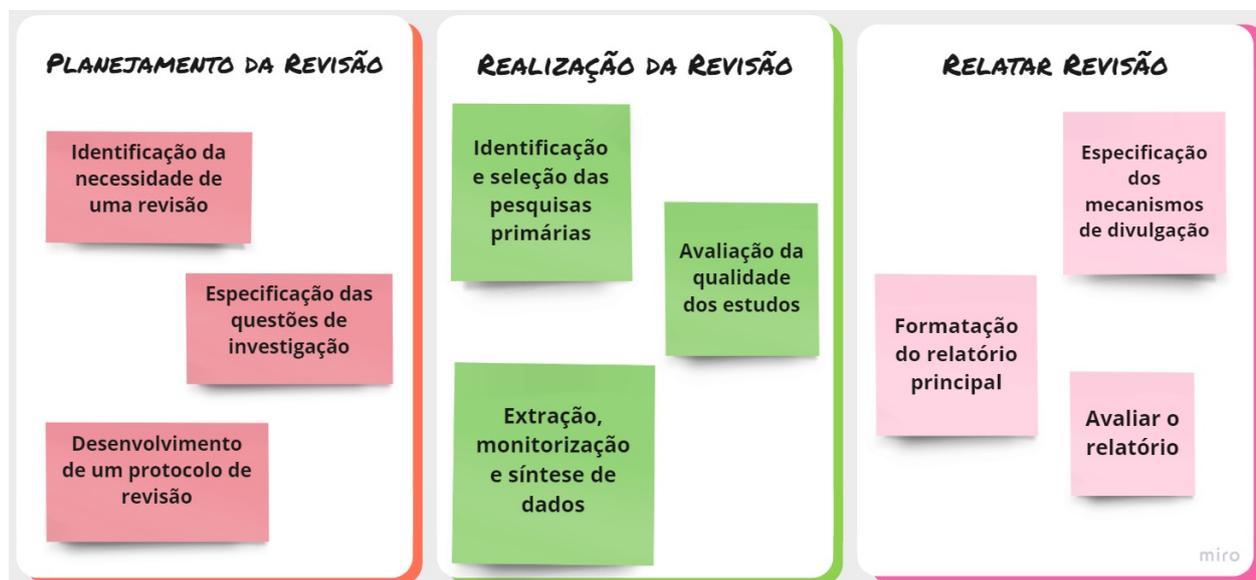


Figura 2.9: O processo de revisão (Elaborada pelo autor (2023))

A Figura 2.9 traz resumidamente o processo de RSL, onde primeiramente deve-se fazer o planejamento da revisão identificando entre os diversos tipos de revisões em qual a sua se encaixa. O segundo passo é especificar as questões de pesquisa para então desenvolver o protocolo. Fazem parte desse passo a escolha das palavras-chave e das bases científicas (Kitchenham e Charters, 2007).

Na fase de realização da revisão, as palavras-chave são aplicadas nos repositórios selecionados, de onde se obtém as pesquisas primárias para que no próximo passo elas possam ser avaliadas segundo sua “qualidade” (o quanto respondem as questões de pesquisa). Os dois últimos passos desta fase são: extração dos dados que possam responder as questões de pesquisa e a realização da síntese deles (Almeida et al., 2020).

Com os dados sintetizados, sua exibição se dá por meio de tabelas, gráficos e figuras para melhor compreensão do leitor. Na fase de relatar revisão, conclui-se respondendo as questões de pesquisa embasado na investigação conduzida, redigindo um trabalho que deve ser avaliado e publicado (Almeida et al., 2020).

2.7 Considerações

Para alcançar o principal objetivo deste trabalho, se faz necessário um forte embasamento teórico, e esse capítulo apresentou teorias que permeiam o de-

envolvimento de DApps, abordando tópicos como o tipo de rede empregado, um breve histórico sobre a evolução da tecnologia *blockchain*, algoritmos de consenso e categorias de plataformas.

A subseção que trata de plataformas *Blockchain* demonstra que este é o tema mais importante para quem deseja criar uma DApp, pois a escolha da plataforma descentralizada é o ponto de partida do desenvolvimento da aplicação e esta escolha direciona para a utilização de determinadas tecnologias.

Posteriormente, foram apresentadas tecnologias mais intrinsecamente ligadas ao desenvolvimento de Aplicações Descentralizadas, a plataforma *Ethereum*, a linguagem de programação *Solidity*, a Máquina Virtual *Ethereum* (EVM) e o conceito de Gas. A EVM foi uma excelente ideia da empresa, pois proporciona uma solução que viabiliza a migração facilitada de uma DApp para outras plataformas. Quanto ao consumo de Gas, trata-se de um ponto crucial que demanda a atenção do desenvolvedor, instigando a criação de contratos inteligentes seguros e a minimização das possibilidades de ocorrência de erros.

Com todo o exposto na penúltima subseção, fica claro que a interoperabilidade e migração são assuntos importantes, mas que ainda não estão consolidadas em uma solução amplamente utilizada. O que existe são soluções diferentes que devem ser pesquisadas conforme a necessidade da aplicação. Portanto, é extremamente importante o papel deste trabalho que cria um *framework* para auxiliar nesta tomada de decisão tão importante que é a escolha da plataforma descentralizada.

O próximo capítulo apresentará uma abordagem baseada em pesquisas primárias e secundárias relacionadas ao desenvolvimento de DApps, focando especialmente em contratos inteligentes e *frameworks* concebidos com o intuito de facilitar o desenvolvimento dessas aplicações.

Trabalhos Relacionados

Neste capítulo são apresentados os trabalhos que abordam a temática de desenvolvimento de aplicações descentralizadas baseadas em redes *Blockchain*. São pesquisas realizadas de maneira exploratória, utilizando palavras-chave como blockchain e DApp, e sem restrição temporal. As pesquisas relacionadas neste capítulo buscam a formalização de alguma parte do processo de desenvolvimento de DApps e foram o ponto de partida para a realização da RSL.

A pesquisa de Marchesi et al. (2020b) cita aplicações que foram desenvolvidas negligenciando práticas metodológicas adequadas, resultando na perda de bilhões de moedas digitais devido a vulnerabilidades resultantes desse desenvolvimento apressado. A pesquisa tem como objetivo principal a criação de um processo baseado na metodologia Scrum para auxiliar na análise, desenvolvimento, teste e integração entre contratos inteligentes e aplicações fora da rede descentralizada. Os autores concluíram salientando a baixa utilização de práticas sólidas de engenharia de *software* e as características peculiares que devem ser tratadas com as ferramentas utilizadas no desenvolvimento. Apesar disso, eles reforçam e demonstram a necessidade de se embasar sua criação em práticas consolidadas pela Engenharia de *Software* e processos sistemáticos, justamente onde este trabalho se propõe atuar, com a criação de um *framework* para auxiliar no desenvolvimento de DApps.

Já o trabalho de Johnson et al. (2019) propôs a criação de um tutorial, formalizando todo o processo de desenvolvimento de uma DApp destinada ao compartilhamento de dados médicos. No entanto, o autor mencionou apenas três plataformas de *Blockchain* sem explicar a razão para essa escolha específica. Esta dissertação, por sua vez, busca fornecer uma visão mais abrangente

das opções disponíveis para o desenvolvimento de DApps, visando orientar e facilitar o trabalho dos desenvolvedores nessa área.

Outros trabalhos, como o de Tas e Tanriover (2019) e Rosa-Bilbao e Boubeta-Puig (2021), descrevem brevemente sua jornada no desenvolvimento de DApps, sem elencar ferramentas ou técnicas utilizadas. O estudo de Trojanowska et al. (2020) se preocupou em demonstrar a análise do sistema trazendo dois diagramas (componentes e caso de uso), nomeou algumas ferramentas, mas não explorou alternativas à plataforma adotada e todos esses três trabalhos concentraram suas pesquisas exclusivamente na plataforma *Ethereum*. Contrastando com essas abordagens, a presente dissertação se propõe a apresentar as plataformas mais amplamente utilizadas no mercado atual e orientar os desenvolvedores, considerando os requisitos do sistema e as opções de plataforma disponíveis.

Os estudos de Bathen e Jadav (2022), Renu e Banik (2021), Shakila e Sultana (2021) e Lin et al. (2022a) enriquecem a compreensão do processo de desenvolvimento de DApps, detalhando as tecnologias empregadas e ilustrando o fluxo de dados e a arquitetura de suas aplicações com o auxílio de numerosas imagens. Contudo, esses autores não exploraram alternativas às tecnologias que escolheram utilizar em suas pesquisas.

A Tabela 3.1. sumariza todo o exposto nesta subseção, exibindo uma relação entre as pesquisas citadas e seu alinhamento com os objetivos específicos desta dissertação.

A Tabela 3.1 demonstra que, apesar de as pesquisas citadas fazerem uso de *frameworks* para solucionar problemas específicos, nenhum deles aborda o que esta dissertação está propondo.

3.1 Considerações

Trabalhos que foram analisados e relacionados neste capítulo, atacam o tema DApp de formas diferentes e trazem a informação de que esta é uma área em grande expansão, porém, ainda com algumas lacunas. Esta dissertação foca na criação de um *framework* de decisão para auxiliar no desenvolvimento de DApp.

Para compreender o estado da arte e embasar a proposta deste trabalho, se mostrou necessária a realização de uma revisão sistemática da literatura, o protocolo de investigação e o resultado da RSL serão exibidos no próximo capítulo.

Tabela 3.1: Comparativo entre as pesquisas e alinhamento com os objetivos específicos desta dissertação (elaborado pelo autor (2023))

	Utiliza Framework para camada de interação com o usuário	Demonstra etapa de análise da aplicação	Conceitua as tecnologias utilizadas	Fundamenta as escolhas de tecnologia	Cita as ferramentas utilizadas	Direciona para uma plataforma específica
<i>Building a Secure Biomedical Data Sharing Decentralized App (DApp): Tutorial</i>	Não	Não	Sim	Não	Sim	Sim
<i>Building A Decentralized Application on the Ethereum Blockchain</i>	Não	Não	Parcial	Parcial	Não	Sim
<i>RectorDApp: Decentralized Application for Managing University Rector Elections</i>	Não	Não	Parcial	Não	Não	Sim
<i>Secure Decentralized Application Development of Blockchain-based Games</i>	Não	Sim	Não	Não	Sim	Sim
<i>Smart Contracts in the Cloud</i>	Não	Não	Não	Não	Não	Sim
<i>Implementation of a Secure Ride-Sharing DApp Using Smart Contracts on Ethereum Blockchain</i>	Não	Não	Sim	Não	Sim	Sim
<i>A Decentralized Marketplace Application based on Ethereum Smart Contract</i>	Não	Não	Não	Não	Não	Sim
<i>Design and Implementation of Blockchain-based College Education Integrity System</i>	Não	Não	Não	Não	Parcial	Sim

Revisão Sistemática da Literatura

Este capítulo descreve a revisão sistemática da literatura conduzida para a definição do *framework*. Neste capítulo é detalhado o protocolo que foi adotado, abordando as questões de pesquisa, a metodologia de busca nas bases de dados científicas, os resultados alcançados, bem como as respostas e conclusões derivadas desse processo.

Um mapeamento sistemático não foi utilizado por sua característica de considerar um número maior de trabalhos e os analisar de uma maneira menos aprofundada (Napoleão et al., 2017). Dado o caráter novo, intrigante e complexo do tema, optou-se por conduzir uma RSL a fim de realizar uma análise aprofundada das pesquisas disponíveis. Essa abordagem permitiu adquirir um entendimento mais abrangente do assunto, capacitando a formulação de um *framework* que possa, de maneira mais precisa, auxiliar os desenvolvedores.

4.1 Planejamento

O protocolo da revisão seguiu o modelo estabelecido por Kitchenham e Charters (2007). As questões de pesquisa definidas foram:

Q1: Existe uma definição clara (critérios) para construção de DApps?

Atuando de forma complementar a questão principal:

Q2: Em que etapa/camada do desenvolvimento de DApp este *framework* atua?

As questões de pesquisa (Q1 e Q2) foram formuladas com o propósito de destacar que, embora existam *frameworks* disponíveis para apoiar o desenvolvimento de DApps, nenhum deles aborda especificamente a fase de seleção da plataforma, como proposto no presente estudo.

4.2 Bases Científicas

Após a definição das questões essenciais, o próximo passo foi selecionar as bases de dados científicas apropriadas e definir a estratégia de busca. Cinco bases de dados internacionais foram escolhidas para esta pesquisa, conforme detalhado na Tabela 4.1.

Tabela 4.1: Bases de dados selecionadas (elaborado pelo autor (2023))

Bases	Link de acesso
IEEE Xplore	https://ieeexplore.ieee.org/Xplore/home.jsp
ACM Digital Library	https://dl.acm.org/
Elsevier ScienceDirect	https://www.sciencedirect.com/search
Springer Link	https://link.springer.com/
Scopus	https://www.scopus.com/

Com as questões de pesquisa estabelecidas e as bases relevantes selecionadas, a próxima fase envolve a definição das palavras-chave e a formulação das strings de busca específicas para cada base de dados.

4.3 Strings de busca

As questões de pesquisa apontam para a necessidade de estabelecer uma padronização. Para identificar pesquisas com objetivos semelhantes, foram selecionadas as seguintes palavras-chave:

- *Definition;*
- *Criteria;*
- *Roadmap;*
- *Formalization;*
- *Systematic;*
- *Framework;*
- *Build.*

Para complementar este conjunto de palavras relacionadas à formalização, adicionou-se a palavra “*Build*”. Além disso, os seguintes termos que refletem o foco central deste estudo foram incorporados:

- *DApp*;
- *Decentralized Application*;
- *Blockchain*.

Após a seleção das palavras-chave, foi necessário criar as strings de busca, as quais são adaptadas para cada base de dados. Após testes de composição das *strings* de busca, a Tabela 4.2 resume os principais resultados. Com base nesses resultados, critérios de inclusão e exclusão foram aplicados para refinar a seleção dos estudos.

Tabela 4.2: Bases e Strings de busca (elaborado pelo autor (2023))

Bases	String de busca
IEEE Xplore	((“Abstract”:"decentralized application" OR dapp AND blockchain) OR (“Document Title”:"decentralized application" OR dapp AND blockchain)) AND ((“Abstract”:"definition OR criteria OR roadmap OR formalization OR systematic OR framework OR build* AND blockchain) OR (“Document Title”:"definition OR criteria OR roadmap OR formalization OR systematic OR framework OR build* AND blockchain))
ACM Digital Library	((title:"decentralized application"OR title:dapp) AND title:blockchain) OR ((abstract:"decentralized application"OR abstract:dapp) AND abstract:blockchain)) AND ((title:build* OR title:criteria OR title: definition OR title:roadmap OR title: formalization OR title:systematic OR title:framework OR title:build) OR (abstract:build* OR abstract:criteria OR abstract:definition OR abstract:roadmap OR abstract: formalization OR abstract:systematic OR abstract:framework))
Elsevier ScienceDirect	("decentralized application"OR dapp) AND blockchain AND (Definition OR roadmap OR formalization OR systematic OR framework OR build) + ("decentralized application"OR dapp) AND blockchain AND (criteria)
Springer Link	((ti:(dapp) OR ti:(“decentralized application”) AND ti:(blockchain)) OR (ab:(dapp) OR ab:(“decentralized application”) AND ab:(blockchain))) AND ((ti:definition OR ti:criteria OR ti:roadmap OR ti:formalization OR ti:systematic OR ti:framework OR ti:build*) OR (ab:definition OR ab:criteria OR ab:roadmap OR ab:formalization OR ab:systematic OR ab:framework OR ab:build*))
Scopus	((("decentralized application"OR dapp) AND blockchain) AND (definition OR criteria OR roadmap OR formalization OR systematic OR framework OR build*))

Na Tabela 4.2 percebe-se que as bases da IEEE, ACM e Springer, necessitam que o pesquisador informe quando pretende fazer a busca das palavras no título e no resumo dos trabalhos. A base da Scopus também permite realizar

tal tipo de busca, mas não aparece na String, por esta opção estar disponível para ser selecionada no site do repositório. O mesmo acontece na base da ScienceDirect. Na Tabela 4.2 é possível observar que a string de busca está dividida, o motivo é que no site não é permitido realizar buscas com mais de nove critérios, então, houve a necessidade de dividi-la em duas.

A Tabela 4.3 sumariza os principais achados.

Tabela 4.3: Resultado das buscas (elaborado pelo autor (2023))

Bases	Quantidade de trabalhos
IEEE Xplore	69
ACM Digital Library	26
Elsevier ScienceDirect	22
Springer Link	312
Scopus	247

Notavelmente, como apresentado na na Tabela 4.3 bases *Scopus* e *Springer Link* apresentaram um volume expressivo de trabalhos relacionados ao tema. Até o momento, o único critério de inclusão/exclusão adotado foi a data de publicação dos trabalhos, restringindo a análise aos artigos publicados entre 2020 e 2023. Tal critério assegura a relevância e a contemporaneidade dos estudos incluídos. Os critérios adicionais de inclusão e exclusão, que são fundamentais para refinar ainda mais a seleção dos estudos, serão detalhados na próxima subseção.

4.4 Critérios de inclusão e exclusão

Os critérios de inclusão definidos para esta revisão foram:

- Trabalhos publicados entre 2020 e 2023;
- Escritos em inglês;
- Pesquisas focadas na padronização do processo de desenvolvimento de DApps.

Por outro lado, os critérios de exclusão incluíram:

- *Short-paper*;
- *Survey*;
- *Abstract*;
- *Chapter/Book*;

- Trabalhos fora do escopo.

A delimitação temporal para trabalhos publicados entre 2020 e 2023 foi uma decisão estratégica para assegurar a incorporação de informações atuais e relevantes no campo da tecnologia *blockchain*. Essa escolha visa refletir as inovações mais recentes e as discussões correntes no desenvolvimento de DApps, garantindo que nosso *framework* esteja alinhado com as tendências e desafios emergentes.

Os trabalhos classificados como “Fora do escopo” foram aqueles que não se concentraram especificamente na formalização do processo de desenvolvimento de DApps. Com os artigos selecionados e os critérios de inclusão e exclusão estabelecidos, foi possível restringir a seleção aos estudos mais relevantes para as questões de pesquisa.

4.5 Seleção das pesquisas

A avaliação individual dos trabalhos retornados seguiu o processo ilustrado na Figura 4.1. Esta figura apresenta o total inicial de pesquisas e a quantidade que restou após a aplicação dos critérios de exclusão.

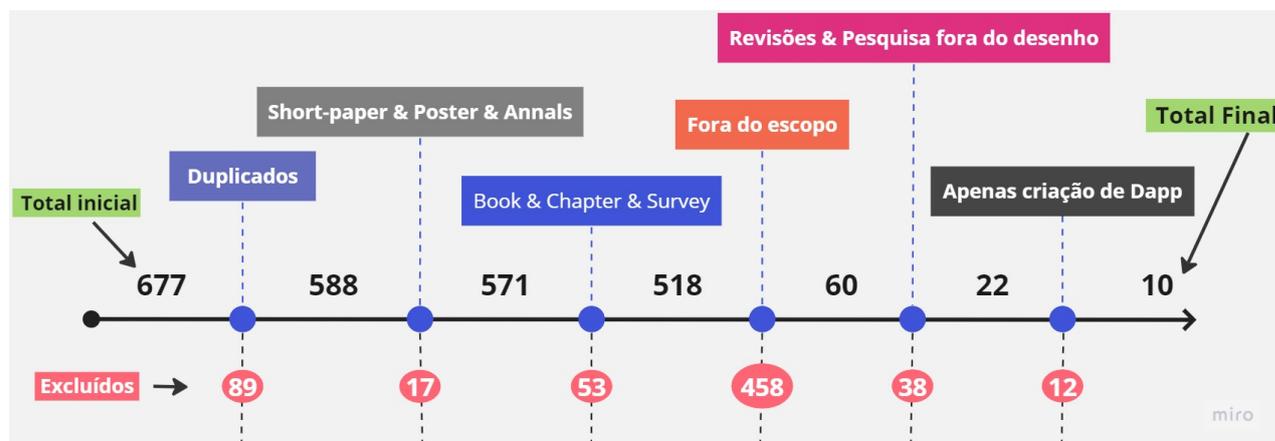


Figura 4.1: Processo de seleção das pesquisas (elaborado pelo autor (2023)).

Inicialmente, foram descartados os trabalhos duplicados, *short papers*, posters, livros, capítulos de livros e *surveys*, totalizando 159 exclusões. A eliminação de 458 trabalhos categorizados como “fora do escopo” exigiu a leitura dos respectivos resumos. Posteriormente, 38 trabalhos foram excluídos como “revisões e pesquisas fora do escopo” após análise dos resumos, introduções e conclusões. Estes trabalhos, apesar de serem estudos primários, não focavam na criação de DApps, mas em outras questões específicas do processo de desenvolvimento.

Os 12 trabalhos classificados como “apenas criação de DApp” foram lidos na íntegra, pois, embora não propusessem uma sistematização no desenvol-

vimento de DApps, forneciam informações valiosas sobre tecnologias e ferramentas utilizadas, contribuindo para o acúmulo de conhecimento na área.

Após a aplicação de todos os critérios, restaram 10 trabalhos, os quais se mostraram potencialmente capazes de responder às questões de pesquisa. A análise detalhada desses trabalhos será discutida na próxima seção.

4.5.1 Ferramentas Utilizadas

É de extra importância mencionar as ferramentas utilizadas durante este processo de condução da RSL.

O Zotero¹ é uma ferramenta de gerenciamento de referências bibliográficas que foi utilizada para converter o tipo de arquivo exportado nas bases científicas que foram consultadas, bem como na facilitação da organização e *download* de pesquisas.

Paralelamente, a ferramenta mais utilizada foi o Rayyan² que facilitou a gestão de várias etapas do processo, incluindo a importação dos estudos, a aplicação dos critérios de inclusão e exclusão, a visualização de resumos e outras informações relevantes, e a detecção de trabalhos duplicados. Ele é um sistema web de apoio à revisão sistemática, foi utilizado para organizar e analisar de maneira eficiente os estudos selecionados.

4.6 Resultados

Esta revisão delineou e aplicou critérios de inclusão e exclusão em estudos com o propósito de responder as questões de pesquisa, resultando na exclusão de vários trabalhos. Os estudos de Fan et al. (2021); Saygili et al. (2022); Choudhari et al. (2021) e muitos outros tratam de otimização, criação e configuração de infraestruturas de redes descentralizadas, ao passo que o escopo dessa dissertação se concentra na padronização e formalização dos procedimentos de desenvolvimento de DApps. Embora esses trabalhos ofereçam contribuições relevantes para o domínio das aplicações descentralizadas, sua ênfase na integração/otimização de redes não corresponde diretamente com o propósito desta pesquisa.

4.6.1 Discussão

Udokwu et al. avaliaram em um primeiro momento seis trabalhos que propuseram a criação de um *framework* para auxiliar no desenvolvimento de DApps e realizou entrevistas com 14 profissionais da área Udokwu et al.

¹<https://www.zotero.org/>

²<https://www.rayyan.ai/>

(2020). A segunda parte do estudo se concentra na síntese dos resultados obtidos e na proposta de um *framework* para o desenvolvimento de DApps. Apesar da relevância do estudo e da execução adequada do seu primeiro objetivo, os autores não conseguiram concluir a criação do *framework*. Na conclusão, eles afirmam que o objetivo da pesquisa era “identificar abordagens adequadas para a concepção e desenvolvimento de DApps”, o que diverge do resumo do trabalho. No entanto, os resultados da revisão bibliográfica são significativos e se alinham estreitamente com os desta pesquisa.

Os resultados da pesquisa conduzida por Udokwu et al. Udokwu et al. (2020) revelaram a existência de *frameworks*, métodos e técnicas que se concentravam na otimização e agilização de processos em diversos domínios e na engenharia de *software*, contudo, não foi encontrado nenhum *framework* que se alinhasse diretamente com a proposta delineada neste estudo.

Já a pesquisa de Nousias et al. Nousias et al. (2022) baseada na plataforma da *Ethereum*, e seus autores criaram um *framework* que auxilia o desenvolvimento de DApps normalizando e modelando processos utilizando a notação de *Business Process Model and Notation* (BPMN). Na contramão da pesquisa citada, o *framework* proposto nessa dissertação não foi delineado para utilização apenas em uma plataforma e sua atuação será durante a etapa de projeto do *software*.

De forma independente de plataforma, um segundo estudo de Udokwu et al. utiliza um *framework* proposto por eles chamado de *Decentralized Agent-Oriented Modelling* (DAOM) Udokwu et al. (2021), demonstrando seu uso através de diagramas de classes, componentes e casos de uso. Outro *framework* independente de plataforma, o Kaya, foi proposto por Wu et al. e visa facilitar os testes de DApps, gerando e executando automaticamente casos de teste e produzindo relatórios detalhados (Wu et al., 2020). Assim como as referências Udokwu et al. (2021) e (Wu et al., 2020), esta dissertação propõe um *framework* independente de plataforma, permitindo ao usuário a liberdade de selecionar aquela que melhor se adequa às suas necessidades.

Baseado na plataforma *Ethereum*, o *framework* Katena de Baresi et al. visa auxiliar os desenvolvedores a reduzir a quantidade de código em até 82,7%, facilitando a implantação e gestão de DApps Baresi et al. (2022). Já o o AgilePlus, proposto por Farooq et al. Farooq et al. (2022), introduz o conceito de *Distributed Agile Software Development* (DASD) e sugere um *framework* para verificação de etapas em um processo de compra na plataforma *Ethereum*.

Os *frameworks* HapiChain, DADS e Gifflar, desenvolvidos respectivamente por Kordestani et al. Kordestani et al. (2020), Altamimi et al. Altamimi et al. (2020) e Santiago et al. Santiago et al. (2021), são baseados na plataforma *Ethereum* e abordam aspectos distintos do desenvolvimento de DApps. O Ha-

piChain visa aprimorar os sistemas de saúde, facilitando o acesso remoto dos pacientes através de módulos de consulta e monitoramento. O DADS, por sua vez, foca em acelerar o processo de aprovação de aplicativos nas lojas de aplicativos da *Google* e *Apple*, otimizando a publicação de DApps. Já o Giffilar se concentra na criação de contratos inteligentes em tempo real, utilizando *TypeScript* para gerar o código necessário, facilitando a implementação de contratos inteligentes em aplicações.

Os já mencionados *frameworks* Baresi et al. (2022), Farooq et al. (2022), Kordestani et al. (2020), Altamimi et al. (2020) e Santiago et al. (2021), embora direcionados para a plataforma *Ethereum*, se concentram em diferentes estágios do desenvolvimento de DApps, resolvendo questões específicas, como a publicação de DApps, o processo de compra e a facilitação do acesso remoto. Por outro lado, o *framework* proposto neste estudo é concebido para ser utilizado por qualquer pessoa que esteja projetando sua aplicação, sem estar vinculada a uma rede descentralizada específica.

Frequentemente mencionado em diversas pesquisas, o *framework Agile Blockchain DApp Engineering* (ABCDE), desenvolvido por Marchesi et al. e baseado na plataforma *Ethereum* e na linguagem *Solidity* Marchesi et al. (2020b), atua na fase de análise de sistemas de DApps. Este *framework* propõe um fluxo de ação detalhado, visando otimizar sua utilização. Uma inovação chave introduzida pelos autores é a adaptação dos diagramas da Linguagem de Modelagem Unificada (UML) para se adequarem às especificidades do desenvolvimento de aplicações descentralizadas em redes *blockchain*.

Concluindo a RSL, as respostas para as questões de pesquisa inicialmente definidas foram:

Q1: Existe uma definição clara (critérios) para a construção de DApps?

A análise das pesquisas selecionadas revela que, atualmente, não existe um *framework* que aborde de forma abrangente a etapa inicial de desenvolvimento de DApps. Enquanto diversos *frameworks* se concentram em questões específicas e abarcam diferentes camadas (infraestrutura, protocolo e expansão), nenhum oferece diretrizes claras para a escolha da plataforma descentralizada mais apropriada de acordo com os requisitos do sistema.

Q2: Em que etapa/camada do desenvolvimento de DApp este *framework* atua?

Não foi identificada, nas pesquisas analisadas, uma proposta de sistematização do processo de desenvolvimento de DApps que seja semelhante ao que este trabalho se propõe a realizar. Conforme apresentado na Tabela 4.4, há uma lacuna significativa na literatura atual em relação à sistematização criteriosa e fundamentada do desenvolvimento de DApps.

Tabela 4.4: Comparativo entre as pesquisas relacionadas na RSL (elaborado pelo autor (2023))

Pesquisas	Etapa de atuação	Plataforma	Proposta	Sistematiza DApp
<i>ABCDE agile Blockchain DApp engineering</i>	Modelagem	Ethereum	Criação de <i>framework</i>	Não
<i>A Blockchain-Based Framework for Distributed Agile Software Development</i>	Validação de requisitos <i>smart contract</i>	Ethereum	Criação de <i>framework</i>	Não
<i>A declarative modelling framework for the deployment and management of blockchain applications</i>	Redução de linha de código <i>smart contract</i>	Ethereum	Criação de <i>framework</i>	Não
DADS: Decentralized (Mobile) Applications Deployment System Using Blockchain: Secured Decentralized Applications Store	Otimização de publicação de <i>smart contract</i>	Ethereum	Criação de <i>framework</i>	Não
<i>Evaluation of Approaches for Designing and Developing Decentralized Applications on Blockchain</i>	Modelagem e desenvolvimento	Independente de plataforma	Criação de <i>framework</i>	Não
<i>Giffjar: A Framework to Generate Smart Contracts on the Fly</i>	Geração de <i>smart contract</i>	Ethereum	Criação de <i>framework</i>	Não
<i>HapiChain: A Blockchain-based Framework for Patient-Centric Telemedicine</i>	Otimização de conexão remota	Ethereum	Criação de <i>framework</i>	Não
<i>Implementation and evaluation of the DAOM framework and support tool for designing blockchain decentralized applications</i>	Modelagem	Independente de plataforma	Avaliação de <i>framework</i>	Não
<i>Kaya: A Testing Framework for Blockchain-based Decentralized Applications</i>	Teste	Independente de plataforma	Criação de <i>framework</i>	Não
<i>Modelling the Development and Deployment of Decentralized Applications in Ethereum Blockchain: A BPMN-Based Approach</i>	Modelagem	Ethereum	Criação de <i>framework</i>	Não

4.7 Considerações

Neste capítulo, delineou-se o protocolo para a condução da revisão sistemática, detalhando o processo desde a fase inicial de busca por trabalhos até a aplicação dos critérios de inclusão e exclusão. Por fim, realizou-se uma análise minuciosa dos artigos, visando encontrar as respostas para as questões de pesquisa que foram levantadas.

A RSL demonstra que as camadas de infraestrutura e protocolo são bem exploradas, assim como os contratos inteligentes na camada de expansão. A subcamada de aplicação ainda é falha quanto a sistematização de seus procedimentos, pois normalmente as pesquisas focam no problema e não no processo. O próximo capítulo apresenta os resultados da dissertação.

Resultados

Neste Capítulo, serão apresentados os resultados deste trabalho, com ênfase nas distinções entre aplicações *web* clássicas e DApps. Ao final, será exposto o *framework* desenvolvido para auxiliar na tomada de decisões relacionadas à escolha da plataforma onde será executada a DApp e a avaliação que foi realizada.

5.1 Aplicação Web Clássica x DApp

O desenvolvimento de aplicações *web* de maneira clássica difere substancialmente do que é empregado em aplicações descentralizadas baseadas em redes *Blockchain*.

Diante desse cenário contrastante entre aplicação clássica e DApp, torna-se evidente que seria caótico e impraticável esperar que cada nó pertencente a uma *Blockchain* suporte DApps desenvolvidas em todas as diversas linguagens existentes no mercado, além de empregar distintas tecnologias, como *frameworks* e *plugins* que são comuns no desenvolvimento *web* clássico.

Por essa razão e outras, as DApps são frequentemente submetidas a restrições no que diz respeito às tecnologias que podem ser utilizadas em seu processo de desenvolvimento. Essa limitação representa uma resposta pragmática às complexidades inerentes à natureza descentralizada desse tipo de aplicação.

A Figura 5.1 apresenta, de maneira não exaustiva, as diversas possibilidades de escolha disponíveis para um desenvolvedor ao criar uma aplicação de forma clássica.

Banco de Dados Relacional e não Relacional	Desenvolvimento Back-end	Desenvolvimento Front-end
MariaDB MySQL SQLServer Elasticsearch PostgreSQL CouchDB MongoDB	Java PHP Python C# Go Ruby Node.js TypeScript	HTML JavaScript CSS React.js Angular.js Vue.js Node.js TypeScript

Figura 5.1: Tecnologias para desenvolvimento *web* clássica (elaborado pelo autor)

A Figura 5.1 ilustra a gama de combinações disponíveis para um desenvolvedor ao criar sua aplicação de maneira clássica. Ao optar por entre diversas linguagens de programação e bancos de dados, este tipo de desenvolvimento oferece uma notável flexibilidade. Ademais, destaca-se pela abundância de *plugins* disponíveis, tais como os relacionados à geolocalização, previsão do tempo, autenticação de login, exibição de anúncios, entre outros.

A facilidade com que esses *plugins* podem ser incorporados à aplicação constitui uma característica distintiva desse método de desenvolvimento, proporcionando ao desenvolvedor uma significativa possibilidade de personalização.

Por outro lado, uma DApp encontra suas opções substancialmente restritas pela escolha da plataforma descentralizada na qual ela será implantada. A seleção da rede determinará os elementos utilizados na criação da aplicação. A Figura 5.2 oferece uma visão concisa dessas opções.

A Figura 5.2 evidencia claramente a importância de selecionar a plataforma adequada para uma aplicação, uma vez que tal escolha restringirá o desenvolvimento da DApp às tecnologias suportadas pela rede escolhida. Antes de tomar essa decisão crucial, é imperativo realizar uma análise dos requisitos do sistema e das capacidades oferecidas pela plataforma. Nesse contexto, destaca-se a relevância do *framework* desenvolvido por este trabalho. As plataformas listadas na Figura 5.2 são aquelas que proporcionam um leque mais amplo de opções aos desenvolvedores, ressaltando ainda mais a importância de uma escolha criteriosa e alinhada aos objetivos da aplicação a ser desenvolvida.

Analisando a Figura 5.2, torna-se evidente a complexidade associada à eventual necessidade de migrar uma DApp para outra plataforma. Isso ocorre devido à adoção de linguagens distintas e à utilização de bibliotecas diversas para a execução de chamadas à rede em cada plataforma, demonstrando a

Plataformas	Banco de Dados Relacional e não Relacional	Desenvolvimento Back-end	Desenvolvimento Front-end
Ethereum	Blockchain Oracle IPFS	Solidity Vyper	HTML CSS JavaScript Dart TypeScript Node.js
Hyperledger	Blockchain Oracle IPFS	Java Go	HTML CSS JavaScript Node.js
Neo	Blockchain Oracle	Java Python C#	HTML CSS JavaScript Kotlin <small>miro</small>

Figura 5.2: Plataforma/Tecnologias para desenvolvimento de DApp (elaborado pelo autor)

importância de se escolher a plataforma correta para o desenvolvimento da DApp.

5.2 Framework

A escolha da plataforma é a parte mais importante do desenvolvimento, pois trata-se de uma tecnologia que vai ditar a utilização das demais que são necessárias para a criação da DApp.

Os estudos referenciados na RSL abordam um tema de elevada relevância para aqueles que almejam desenvolver uma DApp. A seleção da plataforma descentralizada representa o marco inicial no processo de desenvolvimento da aplicação, e as diretrizes empregadas no *framework* foram extraídas dos estudos mencionados na RSL.

Por esta área estar em constante evolução, o desenvolvimento de uma aplicação se reveste de desafios adicionais. Pessoas que manifestam interesse em ingressar neste domínio frequentemente se deparam com a escassez de material de apoio, os quais poderiam fornecer uma compreensão abrangente das opções disponíveis.

O *framework* foi criado com o intuito de auxiliar as pessoas que pretendem construir uma DApp, permitindo-lhes, tomar uma decisão pautada em uma pesquisa científica para fazer a escolha da plataforma que melhor se adequa ao escopo de sua aplicação.

Ele disponibiliza informações relativas à natureza da rede descentralizada, dados como a utilização de algoritmos específicos, transparência, acessibili-

dade, agilidade, *token* e linguagem adotada pela plataforma para desenvolvimento dos contratos inteligentes. É necessário responder "sim" ou "não" às perguntas e seguir o fluxo do diagrama, ao término, será oferecida a recomendação de uma ou mais plataformas adequadas para a finalidade desejada.

O *framework* foi dividido em dois diagramas, o primeiro, ilustrado pela Figura 5.3 demonstra o fluxo para quem deseja criar DApps de propósito geral e a Figura 5.4 é direcionado para moedas digitais.

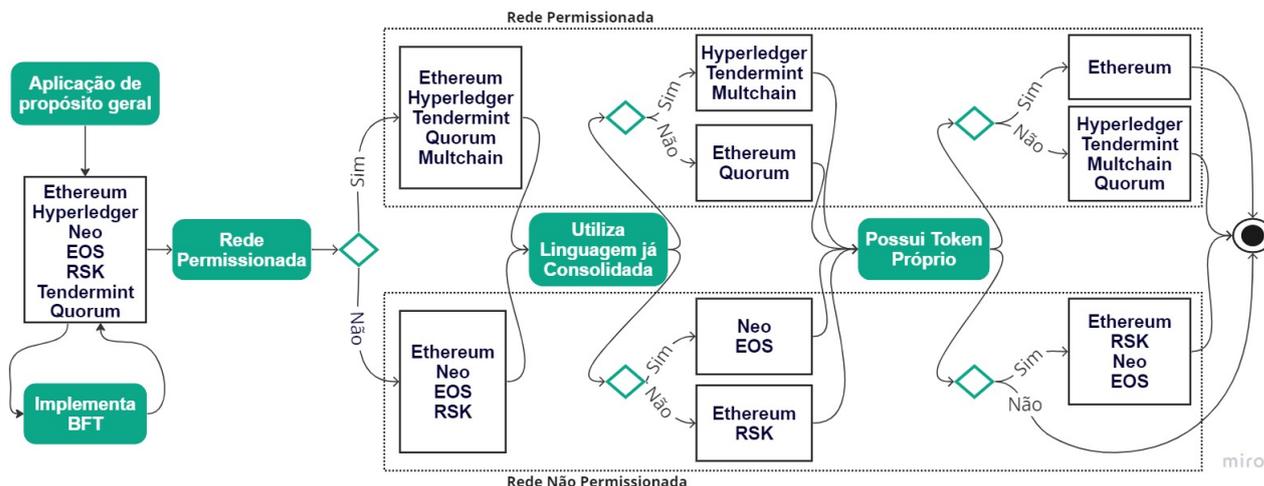


Figura 5.3: Diagrama para DApps em Geral (elaborado pelo autor (2023)).

A Figura 5.3 consiste na apresentação das plataformas sob consideração para a criação de uma DApp de propósito geral. A primeira pergunta é acerca da implementação do algoritmo de tolerância a falhas bizantinas (BFT) e tal implementação está presente em todas as redes analisadas. Este aspecto assume relevância significativa, uma vez que evidencia o comprometimento das plataformas em assegurar uma rede dotada de mecanismos capazes de prevenir seu colapso frente a falhas, sejam elas originadas no sistema e/ou ambiente de execução, ou desencadeadas por usuários maliciosos.

No segundo estágio, procede-se à classificação das plataformas em dois delineamentos distintos: redes permissionadas (privada) e não permissionadas (pública). Considerando às exigências do sistema em desenvolvimento, o programador pode demandar uma rede de natureza privada, devido à sua capacidade de conferir maior sigilo às transações ocorrentes na aplicação, aliada à restrição de acesso e à capacidade de identificação de um nó pertencente à rede que esteja acessando o sistema desenvolvido. Em contrapartida, a necessidade pode pender para uma plataforma que almeje alcançar um amplo número de usuários, proporcionando um acesso descomplicado e transações mais transparentes, assemelhando-se, assim, às características inerentes às redes públicas.

No terceiro estágio, caracteriza-se as plataformas com base na linguagem de programação adotada para o desenvolvimento de seus contratos intelligen-

tes. O desenvolvedor pode optar por redes que fazem uso de linguagens já estabelecidas no mercado, como Java e C++, visando à redução da curva de aprendizado. Alternativamente, pode manifestar indiferença quanto à necessidade de assimilar uma nova linguagem, desde que esta proporcione ferramentas mais avançadas e se alinhe de maneira mais eficaz com a proposta da DApp a ser desenvolvida.

A utilização de uma linguagem já consolidada no mercado é atraente, principalmente por atenuar a curva de aprendizado necessária para iniciar o desenvolvimento de DApps. Outro aspecto reside no amadurecimento da linguagem, que se traduz em uma menor incidência de atualizações de versão, mitigando, assim, a probabilidade de se realizar a refatoração do código para se alinhar às alterações eventualmente introduzidas.

Por fim, o critério consiste na presença de um *token* próprio na plataforma. Para aplicações que visam transacionar ativos digitais, a disponibilidade de um *token* próprio facilita o desenvolvimento da DApp. Cada invocação à *Blockchain* acarreta um ônus. Quando a plataforma em questão dispõe de seu próprio *token*, o processo de cobrança é facilitado

Após a conclusão do desenvolvimento da aplicação destinada à execução em uma plataforma específica, o proprietário da DApp pode proceder à sua migração em busca de um conjunto mais vantajoso de ferramentas disponibilizado por outra rede. Essa transição pode, inclusive, ser planejada, permitindo que o desenvolvedor opte inicialmente por uma plataforma pública visando a redução de custos, para posteriormente migrar para uma rede privada, buscando um ambiente mais condizente com as exigências de sua aplicação. A viabilidade de interoperar e migrar uma DApp é inquestionável, contudo, é um procedimento trabalhoso e suas soluções encontram-se em constante aprimoramento, conforme discutido no capítulo dedicado aos trabalhos relacionados.

A Figura 5.4 ilustra o fluxo a ser seguido no processo de tomada de decisão destinado a pessoas envolvidas no desenvolvimento de aplicações direcionadas a moedas digitais.

O fluxo de operações, disposto na Figura 5.4, é voltado para aplicações com foco em moedas digitais. A primeira pergunta a ser respondida é a respeito da necessidade ou não da aplicação rodar sobre uma rede permissionada, ou seja, uma plataforma em que os nós que a compõem precisam se identificar.

No segundo estágio, surge a indagação sobre a pertinência da plataforma descentralizada utilizar uma linguagem de programação para a elaboração de seus contratos inteligentes, dentre àquelas já consolidadas no cenário de desenvolvimento de software, como C#, Python e PHP.

No terceiro estágio, o foco recai sobre a implementação ou não do algoritmo de Tolerância a Falhas Bizantinas, uma medida destinada a conferir à

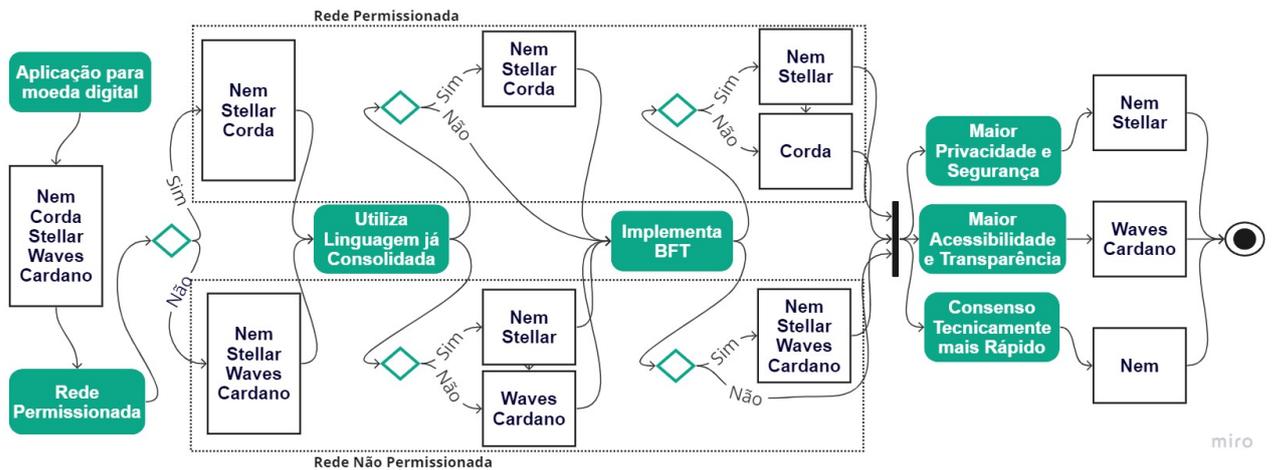


Figura 5.4: Diagrama para DApps para Moedas Digitais (elaborado pelo autor).

rede descentralizada a capacidade de manter suas operações mesmo diante de ataques lançados por usuários maliciosos ou de eventuais disfunções em alguns de seus nós.

Na última camada, são apresentados três estados para consideração do usuário. O primeiro estado propõe a plataforma que, em teoria, proporcione maior privacidade e segurança. Duas plataformas foram sugeridas devido à sua natureza de redes privadas e por implementarem o algoritmo de tolerância a falhas bizantinas.

O segundo estado caracteriza outras duas plataformas como sendo mais acessíveis e transparentes, uma vez que são redes públicas. O terceiro e último estado traz a classificação baseada no algoritmo de consenso mais rápido, sem a realização de testes de velocidade de transação na plataforma. A avaliação é fundamentada exclusivamente no tipo de algoritmo de consenso adotado pela rede.

5.3 Avaliação do Framework

A avaliação foi conduzida com o intuito de identificar potenciais discrepâncias antes da validação por pares. Com o objetivo de facilitar a adoção do *framework*, foi desenvolvida uma aplicação *web* utilizando a linguagem de programação PHP (PHP: *Hypertext Preprocessor*). A referida aplicação está hospedada no GitHub¹ e pode ser acessada através do [Link](#).

5.3.1 Automatização do Framework

Com o propósito de simplificar a condução da avaliação, o *framework* foi automatizado e será demonstrado nesta seção. A Figura 5.5 ilustra a interface

¹<https://github.com/>

inicial da aplicação.

Dev DApp [Principal](#) [Tutorial - dappuniversity](#) [Vídeo - Guia dev](#)

Aplicações Descentralizadas

Baseada em uma rede Blockchain e fazendo uso de contratos inteligentes, as aplicações descentralizadas são softwares de código aberto, transparentes e auditados com facilidade. Por ser descentralizada, seus dados e validações necessárias são providos pelos nós que compõem a rede (Yang et al., 2019).

Comece a programar agora!

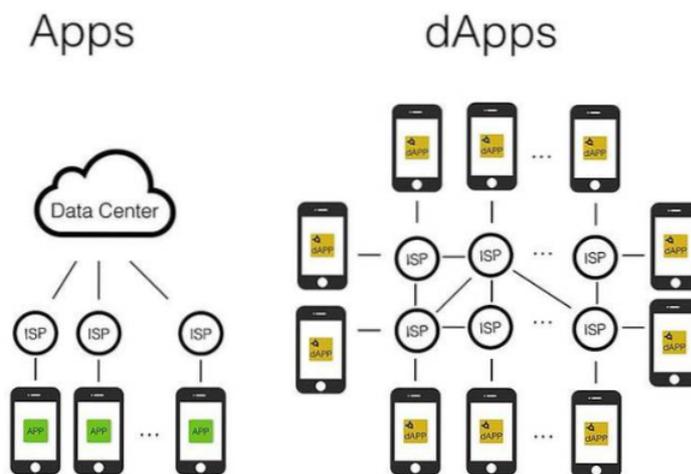


Figura 5.5: Página principal da aplicação (elaborado pelo autor).

Na Figura 5.5, é possível observar a interface principal da aplicação. Como este *framework* destina-se a pessoas iniciantes na área de desenvolvimento de DApps, a tela principal apresenta informações básicas e um menu com *links* para acesso a mais informações sobre o desenvolvimento de DApps.

A segunda tela, apresentada na Figura 5.6, exibe um breve parágrafo alertando o usuário sobre a importância de sua escolha de plataforma, visto que, a plataforma determinará as tecnologias que poderão ser utilizadas no desenvolvimento da DApp.

Na Figura 5.6, é apresentada uma tabela contendo as plataformas e suas características principais, as quais são consideradas no âmbito do *framework*.

Prosseguindo para a próxima tela, é exibido o diagrama completo que aborda aplicações de propósito geral. Esta página pode ser visualizada na Figura 5.7.

Na Figura 5.7, é apresentado o início do uso do *framework*, exibindo o diagrama completo, as plataformas consideradas e as etapas pelas quais o processo passará.

Aplicação Web Clássica x DApp

DApp encontra suas opções substancialmente restritas pela escolha da plataforma descentralizada na qual ela será implantada. A seleção da rede determinará os elementos utilizados na criação da aplicação.

[Voltar](#) [Avançar](#)

A tabela abaixo exibe as principais plataformas e suas características essenciais

Plataforma	Aplicação	Linguagem Contrato Inteligente	Rede
Hyperledger	Geral	Java, Golang e JavaScript	Privado
Tendermint	Geral	Qualquer linguagem	Privado
Multichain	Geral	JavaScript	Privado
Neo	Geral	C, VB.Net, F, Java, Kotlin, Python, Golang e JavaScript	Público
EOS	Geral	C++, Rust, Golang, AssemblyScript e Python	Público
Nem	Moeda	Java	Ambos
Stellar	Moeda	Rust	Ambos
Corda	Moeda	Kotlin e Java	Privado
Cardano	Moeda	Plutus	Público
Ethereum	Geral	Solidity e Vyper	Ambos
Quorum	Geral	Solidity, Vyper e Serpent	Privado
RSK	Geral	Solidity e Vyper	Público

Figura 5.6: Segunda página da aplicação (elaborado pelo autor).

Aplicações de propósito geral

Serão consideradas as seguintes plataformas:

- Ethereum, Hyperledger, Neo, EOS, RSK, Tendermint e Quorum.

As perguntas serão:

- Implementa BFT, Rede permissionada, Utiliza linguagem consolidada e Possui token próprio.

[Voltar](#) [Avançar](#)

A figura abaixo demonstra o todo do framework.

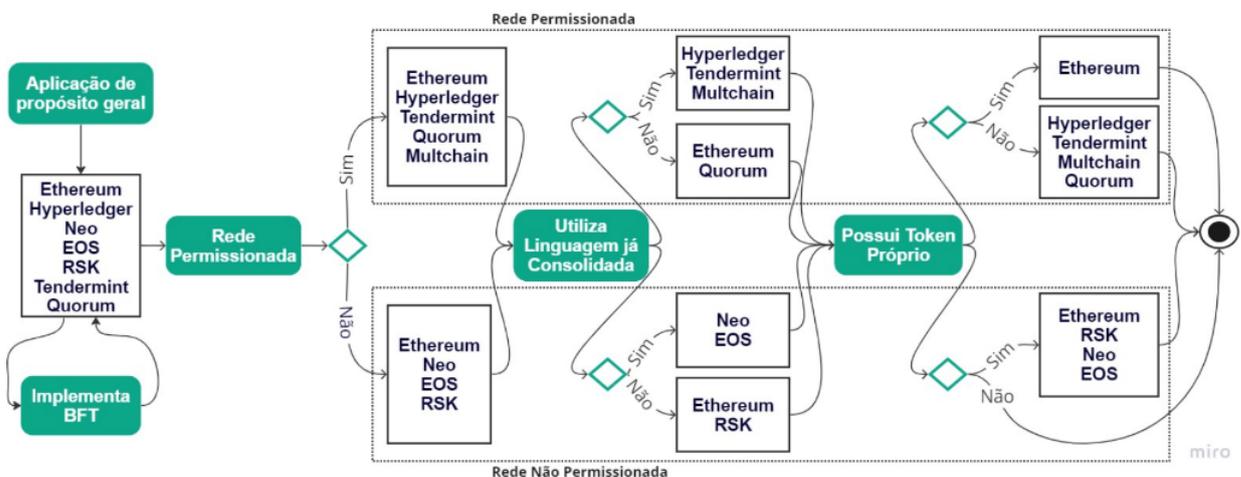


Figura 5.7: Terceira página da aplicação (elaborado pelo autor).

Para evitar a exibição de um número excessivo telas devido às diversas combinações possíveis pelo cliente, os próximos passos até a sugestão da plataforma foram demonstrados utilizando um problema como referência.

A avaliação do *framework* será conduzida com base na criação de uma aplicação que foi caracterizada e passou pelo fluxo estabelecido pelo *framework* proposto nesta dissertação.

O problema em questão consiste na criação de uma aplicação para calcular a média aritmética dos alunos de uma escola, sem a necessidade de armazenar informações pessoais dos estudantes e sem custos de hospedagem para a DApp.

Retomando o uso do *framework*, a tela seguinte apresenta a primeira pergunta relacionada à implementação do algoritmo BFT, conforme pode ser visualizado na Figura 5.8.

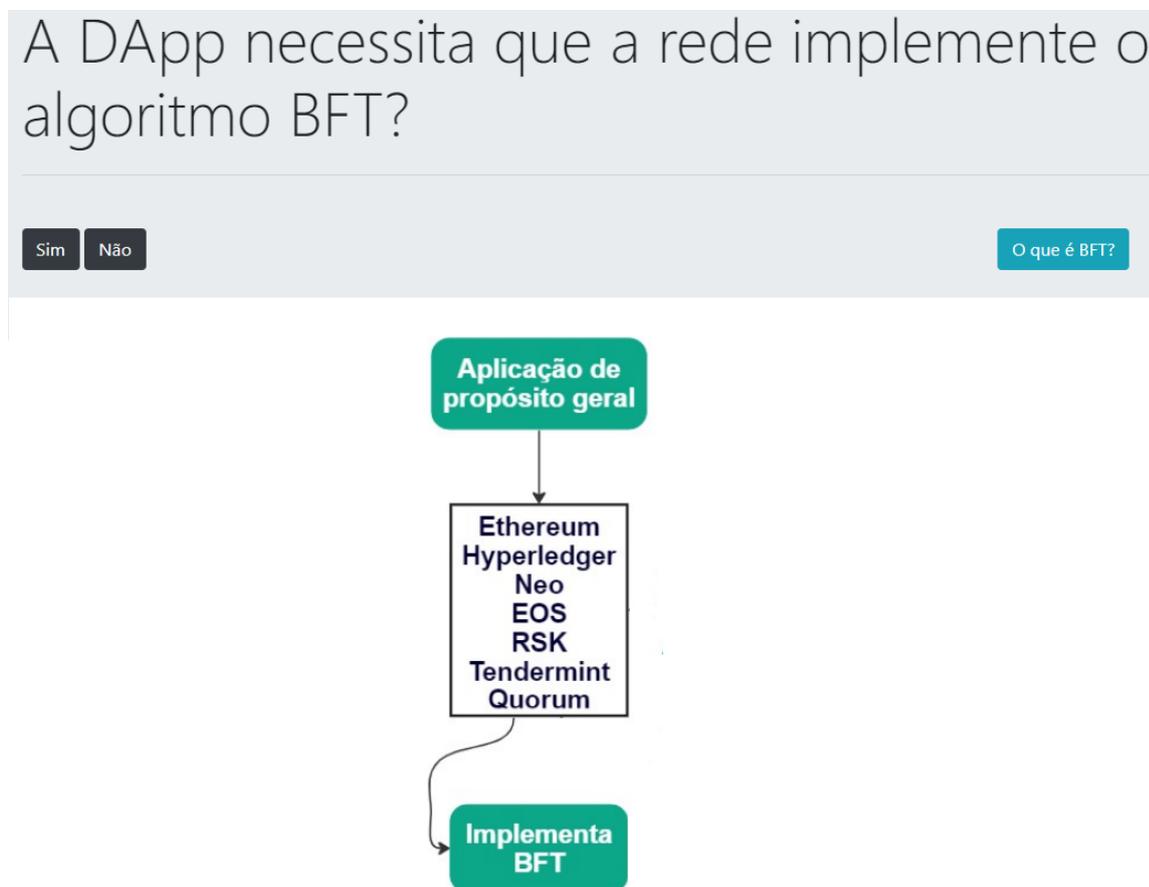


Figura 5.8: Pergunta sobre BFT (elaborado pelo autor).

Na Figura 5.8, são exibidas as plataformas que estão sendo consideradas, juntamente com a primeira pergunta sobre a necessidade de utilização do algoritmo BFT. Caso o usuário tenha dúvidas sobre esse algoritmo, ele pode clicar no botão "O que é BFT?", e uma janela será exibida para explicar do que se trata esse algoritmo, conforme ilustrado na Figura 5.9.

Após clicar no botão "sim", indicando a necessidade de implementar o al-

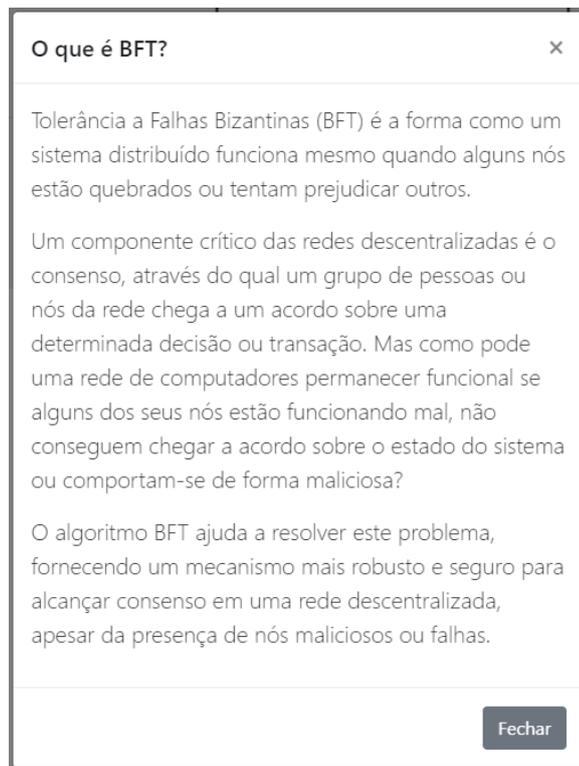


Figura 5.9: Explicação sobre BFT (elaborado pelo autor).

goritmo BFT, todas as plataformas continuam como sugestão de utilização, pois todas implementam o algoritmo. Em seguida, é exibida a segunda pergunta sobre a necessidade de utilizar uma rede permissionada ou não, como ilustrado na Figura 5.10.



Figura 5.10: Pergunta sobre rede permissionada (elaborado pelo autor).

Na Figura 5.10, assim como na página anterior, são fornecidos esclarecimentos sobre o tema da pergunta, neste caso, sobre o conceito de redes

permissionadas. Devido à opção da aplicação desenvolvida por não utilizar uma rede privada, a escolha recai sobre uma rede não permissionada.

A Figura 5.11 exibe a terceira pergunta, que aborda a necessidade de utilizar uma linguagem de programação consolidada ou não.

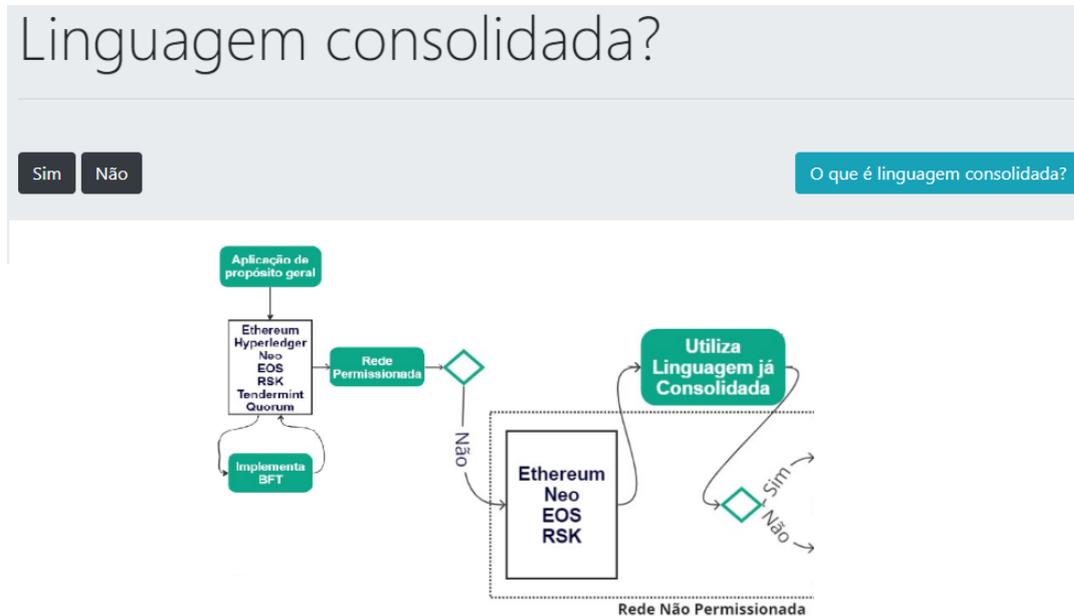


Figura 5.11: Pergunta sobre linguagem de programação (elaborado pelo autor).

Na Figura 5.11, há novamente um botão para fornecer informações adicionais sobre linguagens consolidadas. A parte inferior da página exibe o estágio atual do processo no diagrama. Como a aplicação não requer o uso de uma linguagem consolidada, a opção escolhida é "não".

A Figura 5.12 ilustra o novo estágio em que a aplicação se encontra após esta escolha.

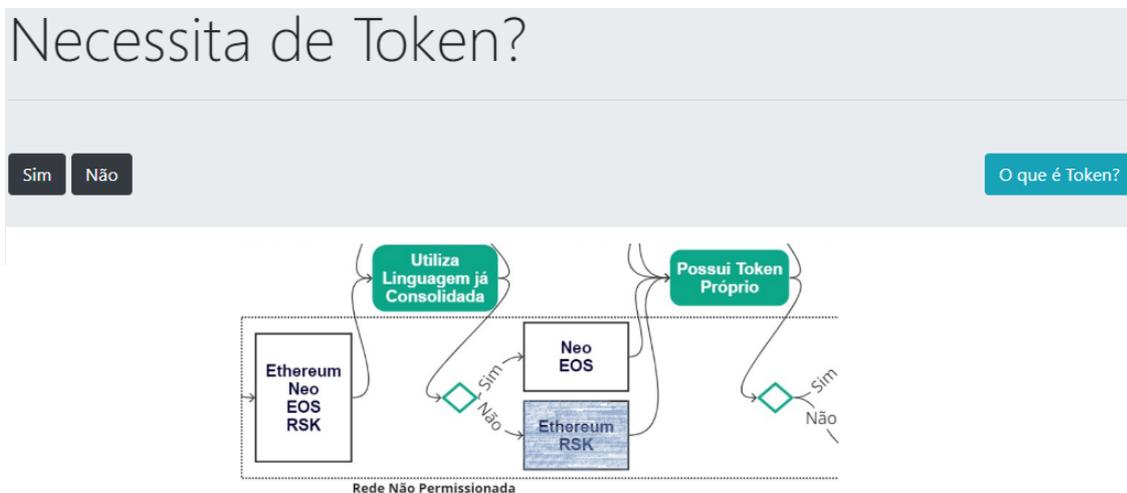


Figura 5.12: Pergunta sobre token (elaborado pelo autor).

A pergunta final, ilustrada pela Figura 5.12, indaga sobre a necessidade de

utilizar *tokens*. Dado que a aplicação desenvolvida não requer essa funcionalidade, a opção escolhida é "não".

Portanto, a Figura 5.13 exibe as sugestões de plataformas que podem ser utilizadas para a criação da aplicação, considerando as escolhas feitas ao longo do processo.

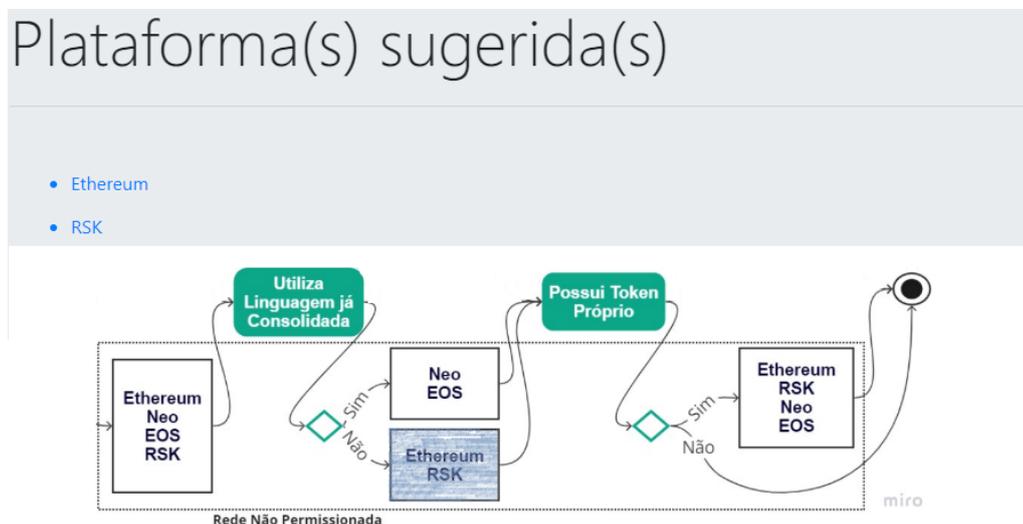


Figura 5.13: Sugestão de plataforma (elaborado pelo autor).

Na Figura 5.13, as plataformas sugeridas foram RSK e *Ethereum*. Devido ao fato de a *Ethereum* contar com um número maior de materiais de estudo, ela foi escolhida para o desenvolvimento da DApp.

A seção seguinte apresenta a perspectiva dos discentes matriculados em um curso de Tecnologia da Informação acerca dessa aplicação.

5.4 Execução da Avaliação

Com o propósito de conduzir a avaliação do *framework* delineado nesta dissertação, a aplicação descrita na seção anterior foi disponibilizada a estudantes de dois cursos diferentes, um deles é da área de Tecnologia da Informação e o outro é uma pós-graduação direcionada a docentes. Estes alunos utilizaram a aplicação e, ao término, avaliaram o *framework* mediante a um conjunto de perguntas específicas.

O método *Likert* foi selecionado para conduzir essa avaliação com os alunos. A escolha desse método se baseou em suas características principais, conforme discutido por Feijó et al. (2020), o método *Likert* é amplamente utilizado, de fácil compreensão e frequentemente empregado em pesquisas científicas.

5.4.1 Perguntas

As perguntas foram elaboradas de acordo com os princípios da escala *Likert*, e especifica cinco categorias que são frequentemente utilizadas com os termos extremos “discordo totalmente” e “concordo totalmente” (Feijó et al., 2020). Seguindo este padrão, as perguntas formuladas são:

- Quantidade de informações;
- Explicação dos conceitos;
- Facilidade de compreensão dos passos;
- Possibilidade de obtenção do produto.

A Figura 5.14, exibe o formulário que foi aplicado.

The screenshot shows a web page for 'Dev DApp' with navigation links for 'Principal', 'Tutorial - dappuniversity', and 'Vídeo - Guia dev'. The main heading is 'Qual o seu grau de satisfação com relação ao uso do framework?'. Below the heading, it states 'As respostas podem ser 1, 2, 3, 4 ou 5.' and provides instructions: '1 significa que está muito insatisfeito com o quesito.' and '5 significa que está muito satisfeito com o quesito.' The survey consists of four questions, each with five radio button options from 1 to 5. A blue 'Enviar' button is located at the bottom left of the form area.

	Perguntas	Muito Insatisfeito				Muito Satisfeito
1	Quantidade de informações	<input type="radio"/> 1	<input type="radio"/> 2	<input type="radio"/> 3	<input type="radio"/> 4	<input type="radio"/> 5
2	Explicação dos conceitos	<input type="radio"/> 1	<input type="radio"/> 2	<input type="radio"/> 3	<input type="radio"/> 4	<input type="radio"/> 5
3	Facilidade de compreensão dos passos	<input type="radio"/> 1	<input type="radio"/> 2	<input type="radio"/> 3	<input type="radio"/> 4	<input type="radio"/> 5
4	Possibilidade de obtenção do produto	<input type="radio"/> 1	<input type="radio"/> 2	<input type="radio"/> 3	<input type="radio"/> 4	<input type="radio"/> 5

Enviar

Figura 5.14: Página de avaliação da aplicação (elaborado pelo autor).

Na Figura 5.14, é possível visualizar as questões previamente mencionadas e sua aplicação conforme conceituado pela escala *Likert*. Após compreender a automatização do *framework* e as questões a serem abordadas, a próxima seção fornecerá informações sobre os estudantes que participaram dessa avaliação.

5.4.2 Perfil dos participantes

A primeira turma de participantes são alunos do sexto semestre do Instituto Federal de Mato Grosso do Sul (IFMS), campus Corumbá, e estão matriculados no curso de Tecnologia em Análise e Desenvolvimento de Sistemas.

A Figura 5.15, ilustra características pessoais dos discentes envolvidos nesta primeira etapa da avaliação.

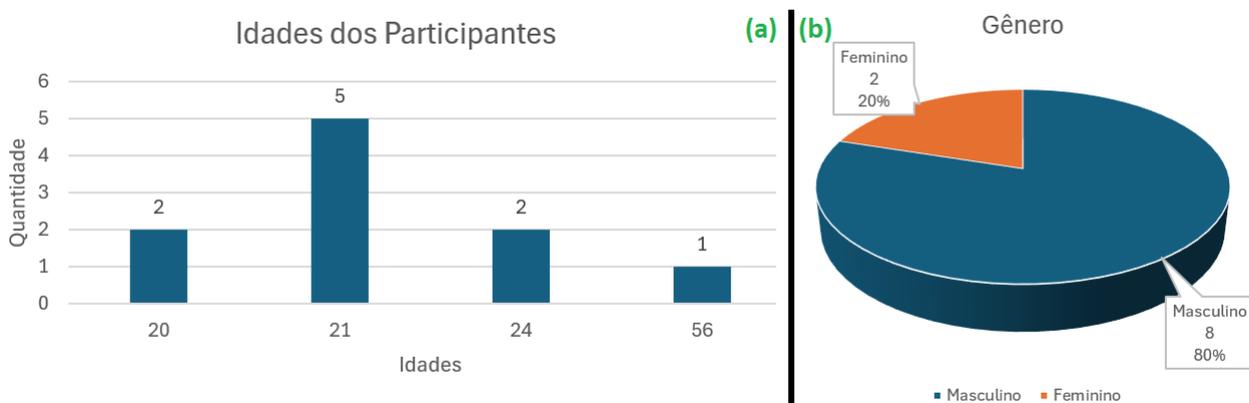


Figura 5.15: Idade e Gênero do estudantes da turma 1 - graduação (elaborado pelo autor).

A Figura 5.15 (a), exibe um gráfico indicando que 90% dos estudantes têm entre 20 e 24 anos. Já, a 5.15 (b) ilustra que 80% dos participantes são do sexo masculino. Todos os estudantes estão no último semestre do curso e possuem o ensino médio completo.

A segunda etapa de avaliação ocorreu com uma turma 12 alunos do curso de Especialização em Informática Aplicada à Educação, pertencentes também ao IFMS. Por ser um curso de especialização a faixa de idade é um pouco maior, assim como, existe uma diversificação na formação dos participantes. Essas características estão sintetizadas nas Figuras 5.16 e 5.17.

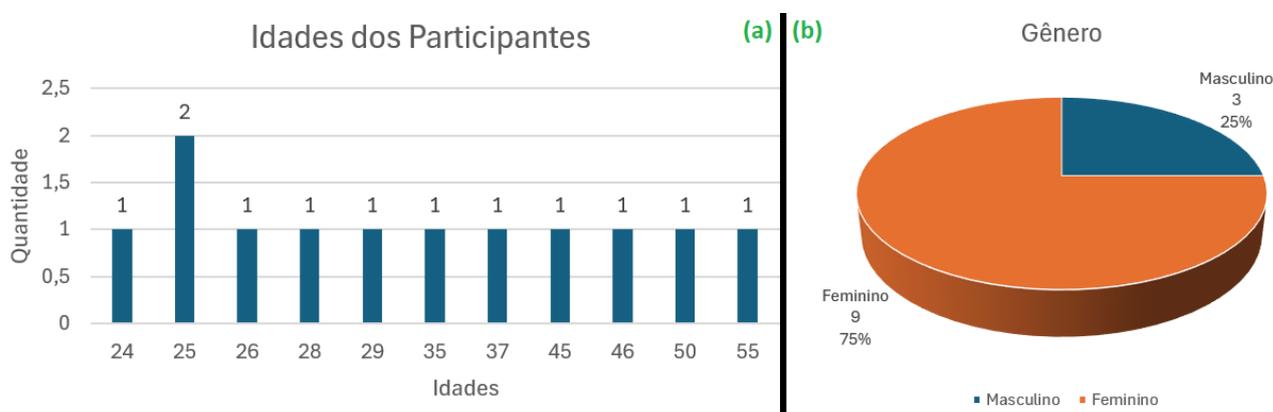


Figura 5.16: Idade e Gênero do estudantes da turma 2 - especialização (elaborado pelo autor).

Na Figura 5.16, percebe-se que as idades dos participantes são maiores que

a dos estudantes da turma 1, a média de idade dos discentes da especialização é de 32 anos, enquanto no curso de graduação é de 21 anos. Outro ponto contrastante é o gênero desses estudantes, a turma de especialização conta com 75% de pessoas do sexo feminino, enquanto na turma de graduação são apenas 20%.

A Figura 5.17, ilustra a formação desses participantes da turma de especialização.

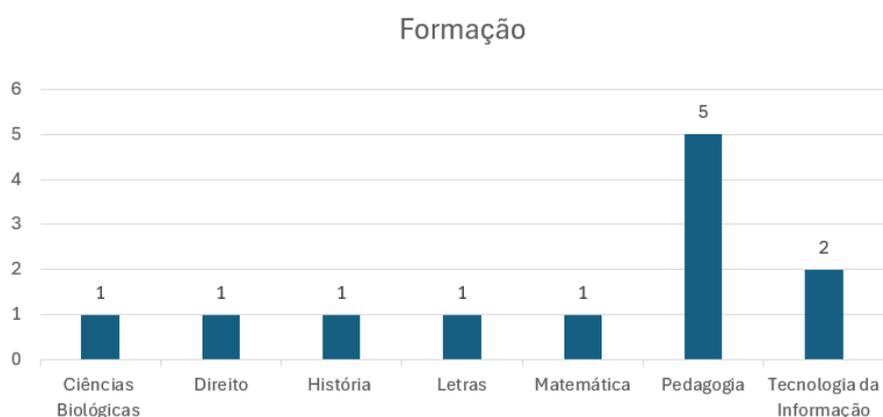


Figura 5.17: Formação dos estudantes da turma 2 - especialização (elaborado pelo autor).

A Figura 5.17, exhibe que os estudantes da especialização são formados em áreas diferentes, e a maior parte no curso de Pedagogia.

Os participantes desta avaliação foram apresentados a quatro alternativas de sistemas para que pudessem utilizar o framework.

5.4.3 Sistemas

Foram oferecidas aos discentes participantes desta avaliação do *framework* quatro aplicações, que foram brevemente especificadas e são de fácil compreensão. O intuito é direcionar o foco do estudante para a utilização do *framework*. As opções sugeridas foram:

- Controle de patrimônio: sistema com acesso por meio de senha. Baixa demanda por disponibilidade e processamento;
- Controle de informações sobre alunos: acesso por meio de senha. Alta demanda por disponibilidade e segurança. Baixa demanda de processamento;
- Sistema de troca de mensagens: acesso livre. Alta demanda por disponibilidade e processamento;

- Sistema de apresentação da instituição (servidores, turmas, cursos, pesquisas conduzidas etc.): acesso livre e simplificado. Alta demanda por disponibilidade.

Após a seleção do sistema, os participantes foram introduzidos ao *framework* e começaram a utilizá-lo. Os resultados obtidos serão detalhados na próxima seção.

5.4.4 Resultados

A avaliação do *framework* que foi conduzida com os dez participantes do curso de graduação estão resumidas na Figura 5.18.

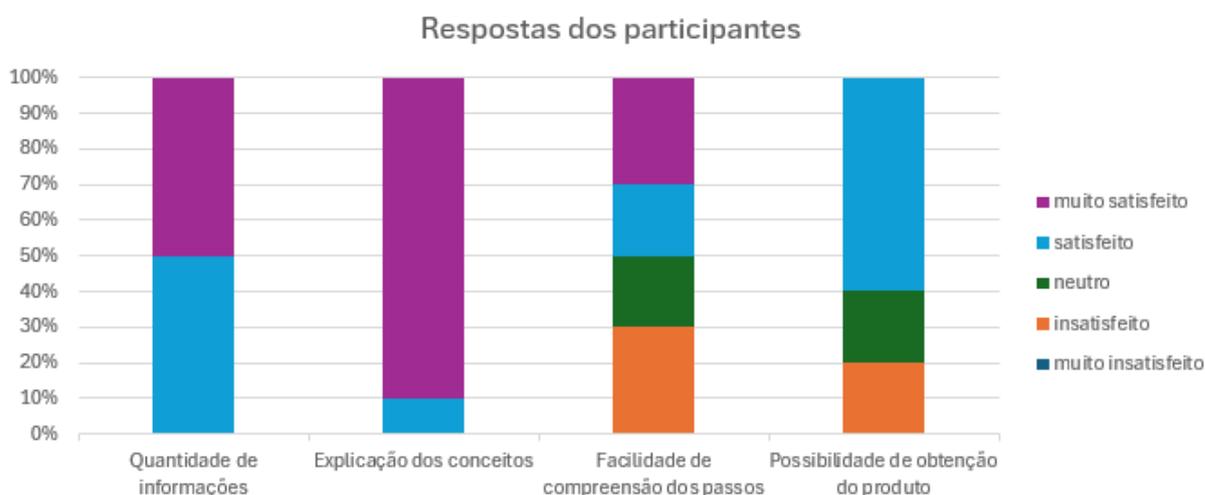


Figura 5.18: Resultado da avaliação com a turma 1 - graduação (elaborado pelo autor).

A Figura 5.18, exibe por meio de um gráfico, que os estudantes do curso de graduação apreciaram as explicações e a quantidade de informações apresentadas na página. No entanto, eles enfrentaram dificuldades para compreender o fluxo dos passos durante a utilização da aplicação. Contudo, uma proporção significativa dos participantes demonstrou confiança em sua capacidade de desenvolver a DApp em conformidade com os requisitos estabelecidos e as sugestões de plataformas.

As opiniões dos estudantes do curso de especialização, estão resumidos na Figura 5.19.

Os resultados da avaliação realizada pelos estudantes da especialização, conforme ilustrado na Figura 5.19, evidenciam que a quantidade de informações disponíveis e as explicações dos conceitos estão satisfatórias. Entretanto, o fluxo pelo qual o usuário passa até receber a sugestão das plataformas não foi um processo simples para metade dos participantes. Por fim, observou-se que a maioria dos estudantes, cerca de 80%, não se sentiu capaz de criar a

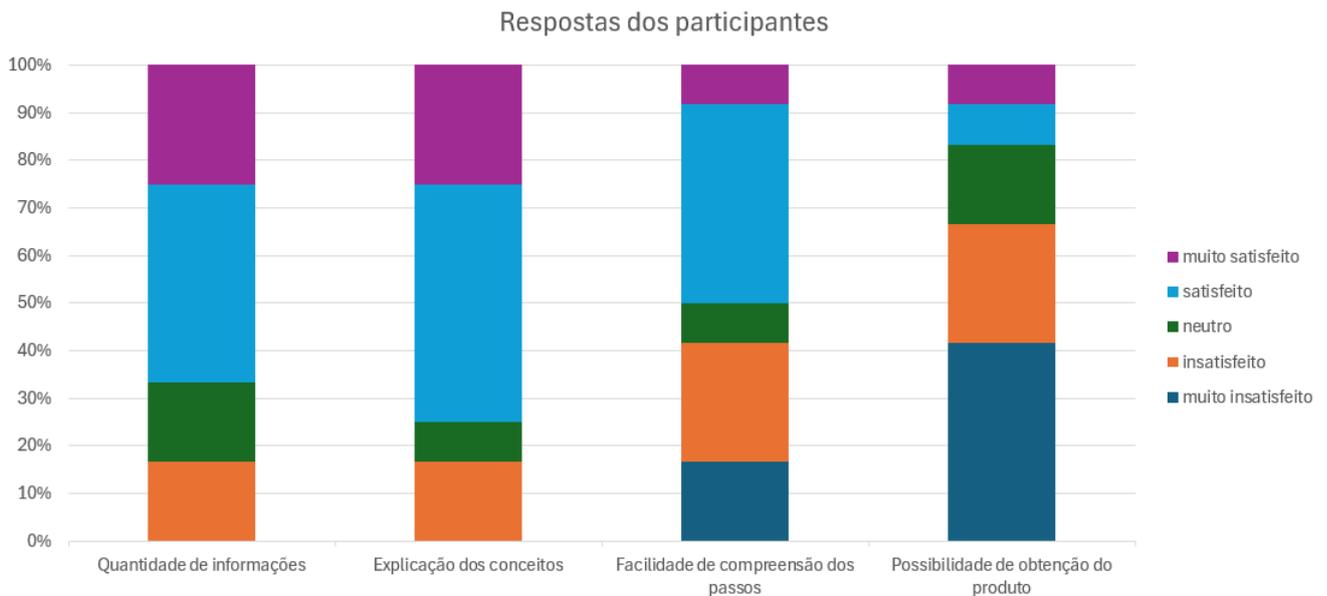


Figura 5.19: Resultado da avaliação com a turma 2 - especialização (elaborado pelo autor).

aplicação conforme os requisitos impostos e as plataformas sugeridas. Esse cenário pode ser atribuído à falta de formação na área de Tecnologia da Informação, o que destaca a necessidade de atenção especial a esse aspecto.

Após a realização da avaliação, foram identificadas ações a serem tomadas com o objetivo de aprimorar o *framework*, as quais incluem:

- Ampliar a quantidade de informações e explicações de conceitos. Haverá a disponibilização de links para páginas exclusivas com esse propósito, visando auxiliar os usuários que necessitam de um entendimento mais aprofundado. Esse suporte adicional será oferecido de forma a não sobrecarregar a página principal com informações excessivas.
- Em relação à facilidade de compreensão dos passos. Será realizada uma revisão do fluxo atual, com o intuito de simplificá-lo. Os passos serão divididos de forma mais clara e concisa, a fim de facilitar a compreensão por parte dos usuários do framework.
- Sobre a possibilidade de obtenção do produto. Ao final, juntamente com a sugestão da plataforma, serão disponibilizados links para guias e vídeos que auxiliarão na programação, além de mencionar os frameworks mais utilizados para desenvolvimento de DApps na referida plataforma. Essa medida visa fornecer um suporte mais abrangente aos usuários, aumentando suas chances de sucesso na obtenção do produto desejado.

5.5 Considerações

Neste capítulo, foram expostos os resultados alcançados, e é inquestionável que a revisão sistemática da literatura e os trabalhos relacionados desempenharam um papel fundamental na criação deste *framework*.

Inicialmente, foi realizada uma comparação entre o desenvolvimento clássico e o descentralizado, com o propósito de evidenciar a notável disparidade existente entre essas duas modalidades. Contudo, o destaque recai sobre o desenvolvimento de fluxos para aplicações que envolvem ou não moedas digitais, a concepção do *framework* proporciona orientação ao desenvolvedor no momento crucial da criação de DApp. O mencionado *framework* destaca, ademais, critérios significativos que serão empregados ao longo do processo de desenvolvimento da aplicação.

Por fim, a avaliação foi conduzida em diferentes turmas de estudantes com o objetivo de coletar opiniões que possam contribuir para o aprimoramento contínuo do *framework*.

Conclusões

O desenvolvimento de DApps revela-se promissor, embora esteja acompanhado por diversos desafios, tais como a compreensão de uma nova arquitetura de desenvolvimento, adoção de uma linguagem de programação específica e familiaridade com múltiplas tecnologias. Dessa forma, torna-se necessário abordar o desenvolvimento tanto de maneira conceitual quanto prático. Nesse contexto, este trabalho se propôs a elaborar um *framework* destinado a orientar a escolha da plataforma mais adequada para a criação de uma DApp, levando em consideração seus requisitos específicos.

O processo de RSL foi primordial para o desenvolvimento do *framework* e destacou uma lacuna significativa, a ausência de um guia abrangente para auxiliar pessoas não especializadas no desenvolvimento de DApps, principalmente na escolha da plataforma descentralizada adequada. Isso levou à proposta do *framework* de tomada de decisão apresentado neste trabalho.

Este *framework* não se destina apenas a enriquecer a pesquisa acadêmica; ele também oferece suporte prático à indústria. Ao fornecer um roteiro claro para a escolha de plataformas descentralizadas, facilita o processo de desenvolvimento de DApps, contribuindo para a expansão e inovação no campo.

6.1 Dificuldades Encontradas

A concepção de um *framework* destinado a facilitar a elaboração de uma DApp por si só constitui um desafio considerável. Isso se deve, em parte, à natureza recente dessa tecnologia e pela maioria dos trabalhos da área estarem redigidos em língua inglesa. Acrescente-se a isso o fato de tais recursos tornarem-se obsoletos com notável celeridade, dada a evolução constante do

domínio em questão. Essa representou a primeira dificuldade enfrentada.

No decorrer da pesquisa exploratória, ao identificar a perspectiva de originalidade e com o intuito de tonar a dissertação mais significativa, emergiu a ideia de realizar uma revisão sistemática da literatura, uma abordagem que confere maior credibilidade ao trabalho, embora se revista de notável complexidade. Trata-se de um processo metodológico extremamente criterioso, exigindo um investimento considerável de tempo para sua efetivação. A compreensão e implementação da RSL constituíram a segunda significativa dificuldade.

O mencionado processo de condução da RSL precisou de uma nova realização após a tentativa de submissão na trilha de desenho de pesquisa do Simpósio Brasileiro de Sistemas Colaborativos em 2023. Após a análise pela banca, algumas considerações foram feitas, ensejando a necessidade de uma segunda RSL para dirimir as lacunas identificadas.

Ao consolidar o conhecimento da área de maneira teórica, impõe-se a terceira grande dificuldade: a efetivação do desenvolvimento de uma aplicação descentralizada na prática. Como previamente delineado no trabalho e exemplificado, tal empreendimento concretizou-se sobre a infraestrutura da *Ethereum*, utilizando a linguagem *Solidity*. Este processo revelou-se um desafio substancial, atestando a marcada discrepância entre o desenvolvimento clássico e aquele voltado para DApps. Tal desenvolvimento impõe premissas, arquiteturas, tecnologias, linguagem e ferramentas inteiramente novas para a criação desse tipo particular de aplicação.

Os demais desafios residem nas atividades inerentes à vida acadêmica e serão elencados posteriormente na seção referente às contribuições.

6.2 Principais Resultados e Contribuições

O primeiro desdobramento consistiu na condução da RSL, por intermédio da qual se conhece o cenário atual do estado da arte na área e a identificação de uma lacuna suscetível de preenchimento. A mencionada revisão, longe de desempenhar somente o papel de base para a elaboração do *framework* proposto, revela-se também como uma valiosa fonte que pode ser aproveitada por outras pessoas desejosas de aprofundar seu entendimento na esfera do desenvolvimento de DApps.

O *framework* em questão representa uma contribuição subsequente e primordial, uma vez que se revela passível de utilização por indivíduos que almejam adentrar ao domínio do desenvolvimento de DApps, sejam eles pesquisadores ou profissionais vinculados à indústria de desenvolvimento de aplicações.

Outras contribuições dizem respeito à trajetória acadêmica durante o programa de pós-graduação, com destaque para:

- Participação no VI Escola Regional de Alto Desempenho do Centro-Oeste (ERAD-CO 2023);
- Apresentação da pesquisa no Integra UFMS 2023;
- Participação como avaliador de trabalhos no Integra UFMS 2023;
- Participação como avaliador de Artefatos no XVII Workshop de Trabalhos de Iniciação Científica e de Graduação (WTICG 2023) e recebimento de convite para fazer parte do Comitê Técnico de Artefatos (CTA) do WGRS24 (XXIV Workshop de Gerência e Operação de Redes e Serviços);
- Aceite de artigo no 20º Simpósio Brasileiro de Sistemas de Informação (SBSI);
- Submissão de artigo para avaliação no 42º Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC).

6.3 *Trabalhos Futuros*

O escopo deste trabalho, entretanto, é limitado a um conjunto específico de estados e plataformas descentralizadas.

Como direção para trabalhos futuros, propõe-se a expansão destes estados e a inclusão de uma gama mais ampla de redes descentralizadas. Enriquecer o framework com aspectos relacionados à interface do usuário. Isso inclui a consideração de abordagens de desenvolvimento descentralizadas, centralizadas e híbridas, bem como estratégias para a gestão de dados manipulados pelas aplicações. Tal expansão permitirá abordar as crescentes demandas e desafios da área de Sistemas Distribuídos, oferecendo soluções mais abrangentes e flexíveis para o desenvolvimento de DApps

A prioridade iminente é realizar a avaliação/validação do *framework* de uma maneira mais ampla, planeja-se realizar testes com desenvolvedores de DApps, onde eles aplicarão o *framework* em projetos reais para identificar pontos fortes, limitações e áreas de melhoria. Também se considera estudos de caso detalhados que demonstrem a aplicabilidade e os resultados do uso do *framework* em diferentes cenários de desenvolvimento de DApps.

Espera-se que o framework evolua para se adaptar às mudanças tecnológicas e às novas necessidades do mercado. Isso incluirá a adaptação às novas versões de blockchains existentes e a integração de plataformas emergentes, mantendo o framework atualizado e relevante. Esta evolução contínua

é essencial para garantir que o framework permaneça uma ferramenta valiosa tanto para acadêmicos quanto para profissionais no campo dinâmico das DApps.

Sucintamente alguns tópicos para trabalhos futuros são:

- Ampliação na quantidade de estados do diagrama;
- Ampliação na quantidade de plataformas elencadas;
- Abordar a interface de usuário;
- Criar uma DApp, para que as pessoas possam acessar o framework por meio do browser;
- Após escolha da plataforma:
 - Gerar um código básico de demonstração para o desenvolvedor ter o primeiro contato com a aplicação descentralizada;
 - Fazer a sugestão de frameworks mais específicos que podem ser utilizados para facilitar o desenvolvimento da aplicação;
 - Fazer a geração do código base para a aplicação, levando em consideração os requisitos do sistema e utilizando soluções que aumentem a segurança e diminuam o custo das transações.
- Utilizar inteligência artificial na geração do código base.

Referências Bibliográficas

Almeida, W., Furtado, F., Monteiro, L., Escobar, F., e Silva, S. (2020). Systematic review on the use of metrics for estimating the effort and cost of software applicable to the brazilian public sector. Citado nas páginas 23 e 24.

Altamimi, F., Asif, W., Rajarajan, M., M.S., O., K.-F., H., P., N., e D., C.-C. (2020). Dads: Decentralized (mobile) applications deployment system using blockchain: Secured decentralized applications store. *2020 International Conference on Computer, Information and Telecommunication Systems, CITS 2020*. Citado nas páginas 35 e 36.

Back, A. (2002). Hashcash - a denial of service countermeasure. Citado na página 10.

Baranwal, P. (2019). Decentralized applications - utilizing the power of blockchain technology. Disponível em <https://www.talentica.com/blogs/decentralized-applications-utilizing-the-power-of-blockchain-technology/>. Citado nas páginas xi, 13, e 14.

Baresi, L., Quattrocchi, G., Tamburri, D., Terracciano, L., e ACM Special Interest Group on Software Engineering (ACM SIGSOFT); et al.; Huawei; IEEE Computer Society (IEEE CS); IEEE Computer Society, T. C. o. S. E. T. U. o. M. (2022). A declarative modelling framework for the deployment and management of blockchain applications. *25th ACM/IEEE International Conference on Model Driven Engineering Languages and Systems, MODELS 2022*, páginas 311–321. Citado nas páginas 35 e 36.

Bassett, L. (2015). *Introdução ao JSON: Um Guia Para JSON que vai Direto ao Ponto*. Novatec. Citado na página 14.

Bathen, L. A. D. e Jadav, D. (2022). Smart contracts in the cloud. In Bathen, L., Saldamli, G., Sun, X., Austin, T. H., e Nelson, A. J., editors, *Silicon Valley*

- Cybersecurity Conference*, páginas 74–89, Cham. Springer Nature Switzerland. Citado nas páginas 2 e 27.
- Benet, J. (2014). Ipfs - content addressed, versioned, p2p file system. Citado na página 14.
- Chen, X., Liao, P., Kong, Q., Huang, Y., e Zhou, X. (2023). Studying differentiated code to support smart contract update. *Empirical Software Engineering*, 28(5). Cited by: 0. Citado na página 9.
- Chen, Y. (2018). Blockchain tokens and the potential democratization of entrepreneurship and innovation. *Business Horizons*, 61(4):567–575. Citado na página 12.
- Choudhari, S., Das, S. K., e Parasher, S. (2021). Interoperable blockchain solution for digital identity management. *2021 6th International Conference for Convergence in Technology (I2CT)*, páginas 1–6. RAYYAN-INCLUSION: "Frank"=>"Excluded
| RAYYAN-EXCLUSION-REASONS: wrong study. Citado na página 34.
- Cook, J. (2023). Mining. Disponível em <https://ethereum.org/en/developers/docs/consensus-mechanisms/pow/mining/>. Citado na página 20.
- Coulouris, G., Dollimore, J., Kindberg, T., e Blair, G. (2013). *Sistemas Distribuídos: Conceitos e Projetos*. Bookman. Citado nas páginas xi, 5, e 6.
- Dhanvardini, R., Martina, P., Vijay, R., Amirtharajan, R., e Pravinkumar, P. (2023). Development and integration of dapp with blockchain smart contract truffle framework for user interactive applications. In *2023 International Conference on Computer Communication and Informatics (ICCCI)*, páginas 1–6. Citado na página 2.
- Ethereum (2023a). Earn rewards while securing ethereum. Disponível em <https://ethereum.org/en/staking/>. Citado na página 20.
- Ethereum (2023b). The foundation for our digital future. Disponível em <https://ethereum.org/en/what-is-ethereum/>. Citado na página 19.
- Fajge, A. M., Thakur, S., Kumar, R., e Halder, R. (2021). An automated framework for migrating java applications to ethereum solidity applications. In *2021 3rd Conference on Blockchain Research & Applications for Innovative Networks and Services (BRAINS)*, páginas 1–3. Citado na página 22.

- Fan, W., Hong, H. J., Zhou, X., e Chang, S. Y. (2021). A generic blockchain framework to secure decentralized applications. *ICC 2021 - IEEE International Conference on Communications*, páginas 1–7. RAYYAN-INCLUSION: "Frank"=>"Excluded
| RAYYAN-EXCLUSION-REASONS: wrong study design. Citado na página 34.
- Farooq, M., Kalim, Z., Qureshi, J., Rasheed, S., e Abid, A. (2022). A blockchain-based framework for distributed agile software development. *IEEE Access*, 10:17977–17995. Citado nas páginas 35 e 36.
- Feijó, A., Vicente, E., e Petri, S. (2020). The use of likert scales in accounting research. 13:27. Citado nas páginas 49 e 50.
- Gandhi, G. M. e Salvi (2019). Artificial intelligence integrated blockchain for training autonomous cars. In *2019 Fifth International Conference on Science Technology Engineering and Mathematics (ICONSTEM)*, volume 1, páginas 157–161. Citado na página 2.
- Giancaspro, M. (2017). Is a smart contract really a smart idea? insights from a legal perspective. *Computer Law & Security Review*, 33(6):825–835. Citado nas páginas 10 e 16.
- Johnson, M., Jones, M., Shervey, M., Dudley, J. T., e Zimmerman, N. (2019). Building a secure biomedical data sharing decentralized app (dapp): Tutorial. *J Med Internet Res*, 21(10):e13601. Citado na página 26.
- Khan, D., Memon, M. M., Hashmani, M. A., Simpaio, F. T., Sales, A. C., e Santillan, N. Q. (2023). A critical review on blockchain frameworks for dapp. *International Journal of Technology Management and Information System*, 5(1):1–10. Citado nas páginas xiii e 18.
- Khan, S., Amin, M. B., Azar, A. T., e Aslam, S. (2021). Towards interoperable blockchains: A survey on the role of smart contracts in blockchain interoperability. *IEEE Access*, 9:116672–116691. Citado na página 22.
- Kitchenham, B. e Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering. Relatório técnico, Technical report, University of Durham. Citado nas páginas 23, 24, e 29.
- Kordestani, H., Barkaoui, K., e Zahran, W. (2020). Hapichain: A blockchain-based framework for patient-centric telemedicine. *2020 IEEE 8th International Conference on Serious Games and Applications for Health (SeGAH)*, páginas 1–6. Citado nas páginas 35 e 36.

- Lallai, G., Pinna, A., Marchesi, M., e Tonelli, R. (2020). Software engineering for dapp smart contracts managing workers contracts. In *DLT@ ITASEC*. Citado na página 2.
- Liao, Z., Hao, S., Nan, Y., e Zheng, Z. (2023). Smartstate: Detecting state-reverting vulnerabilities in smart contracts via fine-grained state-dependency analysis. In *Proceedings of the 32nd ACM SIGSOFT International Symposium on Software Testing and Analysis, ISSTA 2023*, pagina 980–991, New York, NY, USA. Association for Computing Machinery. Citado na página 21.
- Lin, S., Li, Z., Zhao, S., Zhao, H., Li, Y., e Wang, S. (2022a). Design and implementation of blockchain-based college education integrity system. In *2022 IEEE 5th International Conference on Information Systems and Computer Aided Education (ICISCAE)*, páginas 276–281. Citado nas páginas 2 e 27.
- Lin, S.-Y., Zhang, L., Li, J., e Sun, Y. (2022b). A survey of application research based on blockchain smart contract. *Wireless Networks*, 28(2):635–690. Citado nas páginas xi, 1, 7, 8, e 16.
- Lowdermilk, T. (2013). *Design Centrado no Usuário*. Novatec. Citado na página 14.
- Marchesi, L., Marchesi, M., Destefanis, G., Barabino, G., e Tigano, D. (2020a). Design patterns for gas optimization in ethereum. In *2020 IEEE International Workshop on Blockchain Oriented Software Engineering (IWBOSE)*, páginas 9–15. Citado na página 21.
- Marchesi, L., Marchesi, M., e Tonelli, R. (2020b). Abcde-agile block chain dapp engineering. *Blockchain: Research and Applications*, 1(1):100002. Citado nas páginas xi, 1, 2, 6, 13, 15, 26, e 36.
- Monika e Bhatia, R. (2020). Interoperability solutions for blockchain. In *2020 International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE)*, páginas 381–385. Citado nas páginas xiii, 22, e 23.
- Muratov, F., Lebedev, A., Iushkevich, N., Nasrulin, B., e Takemiya, M. (2018). Yac: Bft consensus algorithm for blockchain. *arXiv preprint arXiv:1809.00554*. Citado na página 11.
- Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system. *Decentralized business review*, pagina 21260. Citado na página 7.

- Napoleão, B., Felizardo, K. R., de Souza, É. F., e Vijaykumar, N. L. (2017). Practical similarities and differences between systematic literature reviews and systematic mappings: a tertiary study. In *SEKE*, volume 2017, páginas 85–90. Citado na página 29.
- Nissl, M., Sallinger, E., Schulte, S., e Borkowski, M. (2021). Towards cross-blockchain smart contracts. In *2021 IEEE International Conference on Decentralized Applications and Infrastructures (DAPPS)*, páginas 85–94. Citado na página 22.
- Nofer, M., Gomber, P., Hinz, O., e Schiereck, D. (2017). Blockchain. *Business & Information Systems Engineering*, 59. Citado na página 7.
- Nousias, N., Tsakalidis, G., Petridou, S., Vergidis, K., A.P., C. S. C., J., P., U., J., e D., K. (2022). Modelling the development and deployment of decentralized applications in ethereum blockchain: A bpmn-based approach. *8th International Conference on Decision Support System Technology, ICDSST 2022*, 447:55–67. Citado na página 35.
- Parizi, R. M., Amritraj, e Dehghantanha, A. (2018). Smart contract programming languages on blockchains: An empirical evaluation of usability and security. In Chen, S., Wang, H., e Zhang, L.-J., editors, *Blockchain – ICBC 2018*, páginas 75–91, Cham. Springer International Publishing. Citado na página 19.
- Qian, P., He, J., Lu, L., Wu, S., Lu, Z., Wu, L., Zhou, Y., e He, Q. (2023). Demystifying random number in ethereum smart contract: Taxonomy, vulnerability identification, and attack detection. *IEEE Transactions on Software Engineering*, 49(7):3793–3810. Citado na página 21.
- Rakita, D. (2023). Ethereum virtual machine (evm). Disponível em <https://ethereum.org/en/developers/docs/evm/>. Citado na página 19.
- Reilly, E., Maloney, M., Siegel, M., e Falco, G. (2019). An iot integrity-first communication protocol via an ethereum blockchain light client. In *2019 IEEE/ACM 1st International Workshop on Software Engineering Research Practices for the Internet of Things (SERP4IoT)*, páginas 53–56. Citado na página 2.
- Renu, S. e Banik, B. (2021). Implementation of a secure ride-sharing dapp using smart contracts on ethereum blockchain. 11(2):167–173. Citado nas páginas 2 e 27.
- Rosa-Bilbao, J. e Boubeta-Puig, J. (2021). Rectordapp: Decentralized application for managing university rector elections. In *2021 IEEE Interna-*

- tional Conference on Service-Oriented System Engineering (SOSE)*, páginas 161–165. Citado na página 27.
- Santiago, L., Abijaude, J. W., e Greve, F. (2021). Gifflar: A framework to generate smart contracts on the fly - proceedings of the 31st annual international conference on computer science and software engineering. pagina 214–219. Citado nas páginas 35 e 36.
- Saygili, M., Mert, I. E., e Tokdemir, O. B. (2022). A decentralized structure to reduce and resolve construction disputes in a hybrid blockchain network. *Automation in Construction*, 134:104056. RAYYAN-INCLUSION: "Frank"=>"Excluded
| RAYYAN-EXCLUSION-REASONS: wrong study. Citado na página 34.
- Shakila, U. K. e Sultana, S. (2021). A decentralized marketplace application based on ethereum smart contract. In *2021 24th International Conference on Computer and Information Technology (ICCIT)*, páginas 1–5. Citado nas páginas 2 e 27.
- Sharma, P., Jindal, R., e Borah, M. D. (2023). A review of smart contract-based platforms, applications, and challenges. *Cluster Computing*, 26(1):395–421. Citado nas páginas xiii, 17, e 18.
- Smith, C. (2023). Gas and fees. Disponível em <https://ethereum.org/en/developers/docs/gas/>. Citado nas páginas xi, 20, e 21.
- Solidity (2023). Introduction to smart contracts - gas. Disponível em <https://docs.soliditylang.org/en/v0.8.21/introduction-to-smart-contracts.html#gas>. Citado na página 20.
- Szabo, N. (1997). Formalizing and securing relationships on public networks. *First Monday*, 2(9). Citado nas páginas 8 e 9.
- Taherdoost, H. (2023). Smart contracts in blockchain technology: A critical review. *Information*, 14(2). Citado nas páginas xiii, 16, e 17.
- Tanenbaum, A. e Steen, M. (2007). *Sistemas Distribuídos: Princípios e Paradigmas*. Pearson. Citado nas páginas 5 e 6.
- Tas, R. e Tanriover, O. O. (2019). Building a decentralized application on the ethereum blockchain. In *2019 3rd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, páginas 1–4. Citado nas páginas 1 e 27.
- Tijan, E., Aksentijević, S., Ivanić, K., e Jardas, M. (2019). Blockchain technology implementation in logistics. *Sustainability*, 11(4). Citado na página 7.

- Trojanowska, N., Kedziora, M., Hanif, M., e Song, H. (2020). Secure decentralized application development of blockchain-based games. In *Secure Decentralized Application Development of Blockchain-based Games*, páginas 1–8. Citado na página 27.
- Udokwu, C., Anyanka, H., e Norta, A. (2020). Evaluation of approaches for designing and developing decentralized applications on blockchain - proceedings of the 4th international conference on algorithms, computing and systems. pagina 55–62. Citado nas páginas 34 e 35.
- Udokwu, C., Brandtner, P., Norta, A., Kormiltsyn, A., e Matulevičius, R. (2021). Implementation and evaluation of the daom framework and support tool for designing blockchain decentralized applications. *International Journal of Information Technology*, 13(6):2245–2263. Citado na página 35.
- Wackerow, P. (2022). Introduction to smart contracts. Disponível em <https://ethereum.org/pt/developers/docs/smart-contracts/>. Citado nas páginas xiv e 10.
- Wackerow, P. (2023). Smart contract languages. Disponível em <https://ethereum.org/en/developers/docs/smart-discretionary-contracts/languages/>. Citado na página 19.
- Wang, X., Ren, X., Qiu, C., Xiong, Z., Yao, H., e Leung, V. C. M. (2022). Integrating edge intelligence and blockchain: What, why, and how. *IEEE Communications Surveys Tutorials*, 24(4):2193–2229. Citado na página 2.
- Wang, Z., Jin, H., Dai, W., Choo, K.-K. R., e Zou, D. (2020). Ethereum smart contract security research: survey and future research opportunities. *Frontiers of Computer Science*, 15(2):152802. Citado na página 20.
- Wei, C., Zehua, W., Jason B., E., Zhen, H., Chen, F., e Victor C. M., L. (2018). Decentralized applications: The blockchain-empowered software system. 6:53019–53033. Citado na página 3.
- Westerkamp, M. (2019). Verifiable smart contract portability. *CoRR*, abs/1902.03868. Citado nas páginas 19 e 21.
- Wu, Z., Zhang, J., Gao, J., Li, Y., Li, Q., Guan, Z., e Chen, Z. (2020). Kaya: A testing framework for blockchain-based decentralized applications. *2020 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, páginas 826–829. Citado na página 35.
- Xiong, H., Chen, M., Wu, C., Zhao, Y., e Yi, W. (2022). Research on progress of blockchain consensus algorithm: A review on recent progress of blockchain

- consensus algorithms. *Future Internet*, 14(2). Citado nas páginas xi, 10, e 11.
- Xu, M., Chen, X., e Kou, G. (2019). A systematic review of blockchain. *Financial Innovation*, 5(1):27. Citado na página 12.
- Yakubu, B. M., Sabi'u, J., e Bhattarakosol, P. (2023). Blockchain-based privacy and security model for transactional data in large private networks. *Scientific Reports*, 13(1). Cited by: 0; All Open Access, Gold Open Access. Citado na página 8.
- Yang, R., Wakefield, R., Lyu, S., Jayasuriya, S., Han, F., Yi, X., Yang, X., Amarasinghe, G., e Chen, S. (2020). Public and private blockchain in construction business process and information integration. 118:103276. Citado nas páginas 8, 11, e 12.
- Yang, W., Aghasian, E., Garg, S., Herbert, D., Disiuta, L., e Kang, B. (2019). A survey on blockchain-based internet service architecture: Requirements, challenges, trends, and future. *IEEE Access*, 7:75845–75872. Citado nas páginas xi, 12, e 13.
- Yue, K., Zhang, Y., Chen, Y., Li, Y., Zhao, L., Rong, C., e Chen, L. (2021). A survey of decentralizing applications via blockchain: The 5g and beyond perspective. *IEEE Communications Surveys & Tutorials*, 23(4):2191–2217. Citado na página 2.