Submission date: 16/04/2024
Camera ready submission date: 31/07/2024

1st round notification: 30/07/2024
Available online: 04/12/2024
Publication date: 04/12/2024

Section: Regular article

Bases de Dados Distribuídas para Aplicações Computacionais: Estudo e Seleção de Tecnologias de Registros Distribuídos

Distributed Databases for Computer Applications: Study and Selection of Distributed Ledger Technologies

Carlo Kleber da Silva Rodrigues[1]

[1]Centro de Matemática, Computação e Cognição (CMCC)
Universidade Federal do ABC (UFABC)
Santo André, São Paulo – Brasil

carlo.kleber@ufabc.edu.br

Abstract. *Computer applications on distributed databases are always present in the digital society. An important issue is that these databases be secure, auditable, transparent, and scalable. This article has two objectives: (i) to provide a theoretical baseline on Distributed Ledger Technologies (DLTs) for implementing distributed databases, and (ii) to propose a method for selecting the type of DLT platform for a target organization, called Método Ágil de Seleção (MAS). To this end, we initially investigate literature works and, subsequently, we derive the MAS. Furthermore, we demonstrate the applicability of MAS through a case study. Finally, general conclusions and future works close this article.*

Keywords. *Distributed Ledger Technologies; Selection framework; Applications.*

Resumo. *Aplicações computacionais de bases de dados distribuídas estão sempre presentes na sociedade digital. Uma importante questão é que essas bases de dados sejam seguras, auditáveis, transparentes e escaláveis. Este artigo possui dois objetivos: (i) prover um arcabouço teórico sobre Distributed Ledger Technologies (DLTs) para implementação de bases de dados distribuídas, e (ii) propor um método de seleção do tipo de plataforma DLT para uma organização alvo, denominado Método Ágil de Seleção (MAS). Para tanto, inicialmente realizamos um estudo de trabalhos da literatura e, na sequência, derivamos o MAS. Além disso, demonstramos a aplicabilidade do MAS por meio de um estudo de caso. Por fim, conclusões gerais e trabalhos futuros encerram este artigo.*

Palavras-Chave. *Tecnologias de Registros Distribuídos; Método de seleção; Aplicações.*

1. Introdução

Aplicações computacionais de bases de dados distribuídas estão cada vez mais presentes na pujante sociedade digital de forma ubíqua e pervasiva. Sob uma visão tecnológica,

Cite as: Rodrigues, C. K. S. (2024). Distributed Databases for Computer Applications: Study and Selection of Distributed Ledger Technologies (Bases de Dados Distribuídas para Aplicações Computacionais: Estudo e Seleção de Tecnologias de Registros Distribuídos). iSys: Revista Brasileira de Sistemas de Informação (Brazilian Journal of Information Systems), , Vol 17(1), 12:1-12:31. DOI: 10.5753/isys.2024.4384

essas aplicações são executadas em servidores na *nuvem*, os quais são acessados pelos clientes por meio da Internet. Comércio Eletrônico, Entretenimento, Logística, Assistência Médica, Transporte, Votação Eletrônica, Internet das Coisas (do inglês, *Internet of Things* – IoT) e Pagamento Eletrônico são exemplos de domínios que fomentam o desenvolvimento dessas aplicações para atender as mais variadas organizações públicas e privadas da sociedade digital, incluindo, e.g., universidades, agências governamentais, bancos, hospitais e empresas privadas (e.g., [Passos et al. 2024, Huang 2024, Jayakumari et al. 2024]).

No entanto, esse complexo ecossistema tecnológico, caracterizado especialmente por um significativo volume de informações, suscita uma importante questão: Que tecnologia usar para implementar e manter as bases de dados distribuídas dessas aplicações, com garantias de segurança, auditabilidade, transparência e escalabilidade? Para responder essa questão, indústria e academia têm enveredado pelo caminho das tecnologias de registros distribuídos (do inglês, *Distributed Ledger Technologies* – DLTs). De forma sucinta, a DLT é uma solução disruptiva que oferta a visão lógica de uma única base de dados por meio de réplicas físicas localizadas em um conjunto de nós processadores conectados sob topologia *peer-to-peer* (P2P), sendo cada nó detentor exclusivo de uma réplica [Fan et al. 2020a, Chowdhury et al. 2019, Hunhevicz and Hall 2020a].

Os dados armazenados em cada réplica se dividem em dois conjuntos. O primeiro se relaciona ao negócio da aplicação (e.g., dados bancários, prontuários médicos, contratos de compra e venda de imóveis, etc.), os quais constituem o banco de dados da aplicação. O segundo identifica as transações realizadas na base de dados, i.e., são os metadados das transações. Por exemplo, em uma transferência de dinheiro entre duas contas bancárias, as réplicas possuem os saldos finais das contas (i.e., primeiro conjunto), além de metadados da operação, incluindo, e.g., tipo de transação, valor do montante, hora, data, versão de protocolo, e números das contas bancárias (i.e., segundo conjunto) [Fan et al. 2020a, Chowdhury et al. 2019, Hunhevicz and Hall 2020a].

Cada nó processador da rede P2P deve analisar e validar os dados de sua réplica local, além de disseminar esses dados via *broadcast* para atualização das réplicas dos demais nós processadores da rede. A convergência em estado das réplicas ocorre de acordo com as regras estabelecidas por um algoritmo de consenso. Essa convergência é que provê a visão lógica de uma única base de dados, mencionada anteriormente. Todo esse procedimento é sustentado por criptografia e protocolos intrínsecos da DLT, cujas propriedades fundamentais estão resumidas na Tabela 1 [Chowdhury et al. 2019, Hunhevicz and Hall 2020b, Fan et al. 2020a].

Cronologicamente, as DLTs possuem quatro fases de evolução [Singh et al. 2022, Nunes et al. 2020, Li and Kassem 2021]. A primeira abrange DLTs usadas para construção de sistemas de criptomoedas, tendo o Bitcoin [Nakamoto 2008] como trabalho seminal. Na segunda, tem-se DLTs e contratos inteligentes (do inglês, *smart contracts*) combinados visando ao desenvolvimento de aplicações, o que resulta no condicionamento de transações a acordos predefinidos entre as partes envolvidas. A terceira propicia a implementação de aplicações descentralizadas (do inglês, *decentralized applications* – DApps) de variados domínios. Por fim, a quarta fase objetiva o emprego de DLTs considerando a integração de diferentes domínios em uma mesma arquitetura lógica, englobando

todas as possíveis aplicações dos usuários, e.g., criptomoedas, contratos inteligentes e DApps [Chowdhury et al. 2019, Hunhevicz and Hall 2020b, Fan et al. 2020a].

Tabela 1. Propriedades fundamentais

Propriedade	Definição
Consenso distribuído	O consenso é alcançado de forma participativa com relação ao estado da base de dados (i.e., valores armazenados), sem depender de uma terceira parte confiável ou entidade centralizada.
Imutabilidade	Os valores armazenados não podem ser alterados. Decorre então que as transações executadas e os efeitos produzidos devem ficar permanentemente registrados.
Rastreabilidade e não repúdio	Toda informação é armazenada por meio da execução de transações. Toda transação precisa ser digitalmente assinada por uma chave criptográfica pública, a qual garante a autenticidade da origem da informação. Combinando-se essa condição com a imutabilidade, tem-se um mecanismo efetivo de rastreabilidade e de não repúdio.
Controle e transparência	O estado da base de dados e todas as transações executadas podem ser verificadas pelos participantes do sistema. Isso é feito consultando os nós processadores e as réplicas locais. Essa condição assegura controle e transparência do sistema.

O contexto acima é a motivação para este artigo, cujo objetivo é duplo: (i) prover um arcabouço teórico robusto sobre DLTs usadas na implementação de bases de dados distribuídas, e (ii) propor um método de seleção do mais adequado tipo de plataforma DLT para uma organização (empresa, companhia, agência, etc.), denominado de *Método Ágil de Seleção* (MAS). Para tanto, fazemos inicialmente um estudo teórico sobre DLTs, englobando contratos inteligentes, trabalhos relacionados, categorização, arquitetura de projeto e algoritmos de consenso. Em seguida, derivamos o MAS com base no estudo teórico realizado, bem como demonstramos sua aplicabilidade por meio da realização de um estudo de caso simplificado.

Como principal contribuição, este trabalho de pesquisa provê subsídios teóricos e práticos essenciais para desenvolvedores e arquitetos na implementação de bases dados distribuídas usando DLTs, bem como se constitui em uma ferramenta de auxílio no processo de seleção do mais adequado tipo de plataforma DLT para uma organização.

O restante deste trabalho é organizado como segue. A Seção 2 discorre sobre contratos inteligentes. Na Seção 3, abordamos trabalhos relacionados. A Seção 4 traz a categorização de DLTs. Na Seção 5, explica-se a arquitetura de projeto de DLTs. A Seção 6 discute algoritmos de consenso. Na Seção 7, tem-se a derivação do *Método Ágil de Seleção* (MAS). A Seção 8 apresenta o estudo de caso. Por fim, conclusões gerais e trabalhos futuros constituem a Seção 9.

2. Contratos Inteligentes

Os contratos inteligentes representam um importante marco na evolução do emprego das DLTs, pois permitiram a flexibilização para além da concepção original de uso em sistemas de pagamentos eletrônicos (e.g., [Cohen 2003, Garcia and Kleinschmidt 2020, Nunes et al. 2020, Li and Kassem 2021, Lone and Naaz 2021]).

Formalmente, os contratos inteligentes são definidos como construtores de linguagem (do inglês, *language constructs*) que possibilitam a criação de aplicações que podem

ir muito além do simples processamento e armazenamento de transações. Por meio dos contratos inteligentes, o objetivo filosófico é o de desenvolver aplicações cujos funcionamentos imitam os contratos tradicionais em papel, no sentido de que as partes envolvidas precisam obedecer a regras previamente acordadas para terem suas transações executadas [Li and Kassem 2021, Nunes et al. 2020].

Sob uma visão tecnológica mais prática de implementação, os contratos inteligentes são estritamente códigos computacionais baseados em cláusulas *If/Then*. Entretanto, os mesmos diferem de programas computacionais comuns baseados em *If/Then* por terem uma execução automatizada sob um sistema baseado em DLT, o que garante as intrínsecas propriedades nativas, e.g., imutabilidade, segurança e resistência à censura (do inglês, *censorship resistance*). Além disso, os contratos inteligentes removem a necessidade da constituição de uma terceira parte *intermediadora* para fins de, por exemplo, verificação e validação das transações a serem executadas [Li and Kassem 2021, Nunes et al. 2020, Lone and Naaz 2021].

Os programas computacionais comuns, que rodam em servidores de uma terceira parte, estão limitados às funções daquele servidor, enquanto que os contratos inteligentes estão limitados apenas pela capacidade do programador e pelo ambiente tecnológico no qual eles estão imersos (e.g., IoT, velocidade de processamento da rede, etc.). Também, como os contratos inteligentes estão habilitados com base nos acordos em uma rede distribuída, eles garantem naturalmente mais confiança do que qualquer outra alternativa de terceira parte centralizada. Uma outra denominação usada para contrato inteligente é o termo *chaincodes*, tipicamente associado com os sistemas Hyperledgers, desenvolvidos pela Linux Foundation [Li and Kassem 2021, Sivanathan et al. 2017, Lone and Naaz 2021].

Para efeito de classificação, existem dois tipos de contratos inteligentes: *determinístico* e *não determinístico*. Os *determinísticos* usam apenas os dados existentes dentro do sistema baseado em DLT no qual eles operam. Os *não determinísticos* necessitam de dados externos para que sejam executados. A fonte de dados externos é denominada de *oráculo* (do inglês, *oracle*). Por exemplo, o *oráculo* pode ser um dispositivo de *hardware* (e.g., uma *tag* RFID para IoT) ou um serviço lógico remoto (e.g., um *website* com a previsão do tempo) [Li and Kassem 2021, Sivanathan et al. 2017, Lone and Naaz 2021].

Finalmente, uma extensão dos contratos inteligentes são as chamadas organizações autônomas descentralizadas (do inglês, *Decentralized Autonomous Organizations* - DAOs). Conceitualmente, uma DAO encerra uma coleção de contratos inteligentes, a qual representa uma organização completamente autônoma operando sem a interveniência humana [Li and Kassem 2021, Singh et al. 2022, Lone and Naaz 2021].

3. Trabalhos Relacionados

Esta seção discorre sobre obras recentes da literatura que se relacionam e contribuem direta ou indiretamente com o propósito desta pesquisa, provendo uma visão do estado da arte. Para fins de organização e facilidade de entendimento, tem-se a divisão desta seção em duas subseções, conforme visto a seguir.

3.1. Sistemas Baseados em DLTs

Defini-se *sistema baseado em* DLTs como um *sistema* ou uma *aplicação* computacional cuja base de dados é desenvolvida usando DLTs. Salvo dito diferente, doravante usam-se os termos *sistema* e *aplicação* como sinônimos para maior simplicidade de discussão.

Inúmeras propostas de sistemas baseados em DLTs podem ser encontradas na literatura. Isso confirma a importância dessa tecnologia para a modernização dos serviços na sociedade. A seguir discorremos sobre alguns trabalhos da literatura direcionados para diferentes domínios de negócio, exemplificando a diversidade dos serviços que podem ser oferecidos pelas DLTs.

Tabela 2. Sistemas baseados em DLTs

Referências	Domínio/Aplicação
[Kumar et al. 2020]	Vendas de produtos via comércio eletrônico
[Yik et al. 2021]	Qualidade em produtos farmacêuticos
[Rodrigues 2021b, Rodrigues 2022]	Sistema público de saúde no Brasil
[Lu et al. 2022]	Rastreabilidade de alimentos
[Zheng and Zhou 2023]	Logística para agricultura
[Moulahi et al. 2023]	Saúde e controle de diabetes
[Passos et al. 2024]	Monitoramento de serviços distribuídos
[Huang 2024]	Casa inteligente em áreas rurais
[Jayakumari et al. 2024]	Votação *online* e computação na nuvem

Em [Kumar et al. 2020], os autores propõem um sistema baseado em DLT para gestão conjunta de vários produtos vendidos por meio de comércio eletrônico, incluindo medicamentos, dispositivos eletrônicos, aparelhos de segurança, produtos alimentícios, dentre outros. De maneira geral, o sistema, denominado de PRODCHAIN, utiliza as características inerentes da DLT, e.g., imutabilidade, transparência e auditabilidade, para reduzir a complexidade de rastreamento dos produtos oferecidos. Além disso, é apresentado um novo mecanismo de consenso baseado em ranqueamento, denominado de *Prova de Realização*. Os experimentos são conduzidos na plataforma Ethereum. Os resultados são discutidos em termos de latência e desempenho, os quais comprovam a efetividade do sistema proposto. Enfim, a solução apresentada é benéfica para melhorar a rastreabilidade dos produtos, garantindo a sustentabilidade social e financeira do comércio eletrônico.

Em [Yik et al. 2021], os autores propõem um sistema baseado em DLT para aprimorar o controle de qualidade de produtos farmacêuticos da classe de fitoterápicos. A autenticação fitoterápica correta é de suma importância para a segurança e o melhor interesse dos consumidores. O uso de DLTs visa especialmente minimizar a manipulação dos dados a serem registrados, incluindo, e.g., informações sobre a plantação, as técnicas de cultivo, as fábricas envolvidas nos processos existentes, os laboratórios de testes, os distribuidores e os varejistas. Os autores afirmam que, ao registrar devidamente os parâmetros e dados essenciais para a qualidade dos produtos na fabricação e na cadeia de fornecimento, a rastreabilidade e a confiabilidade dos produtos podem ser garantidas.

Em [Rodrigues 2021b], o autor propõe um sistema baseado em DLT com o

propósito de gerenciamento de prontuários médicos eletrônicos do sistema público de saúde no Brasil, denominado de Sistema Único de Saúde (SUS). Esse sistema atende uma população de aproximadamente 214 milhões de cidadãos em 27 unidades federativas que abrangem mais de 8,5 milhões de km^2. A eficácia da solução proposta é demonstrada por meio de discussão conceitual e modelagem analítica. A principal contribuição desse trabalho é o fornecimento de subsídios teóricos e experimentos para orientar desenvolvimento de projetos reais. Em continuidade a esse trabalho, esse autor ainda realiza uma comparação entre sistemas baseados em DLT de mesmo propósito em [Rodrigues 2022], chegando-se à principal conclusão de que o critério de consenso do tipo *Vote-based* é mais eficiente do que do tipo *Computação intensiva*, e que os requisitos de integridade e confidencialidade podem ser satisfatoriamente atendidos independentemente do tipo de consenso escolhido.

Em [Lu et al. 2022], os autores propõem um sistema de rastreabilidade de alimentos baseado em DLT e IoT em resposta aos problemas de fácil adulteração de dados e rastreabilidade, inerentes a sistemas tradicionais. O novo sistema utiliza o armazenamento descentralizado e as características de alta segurança da tecnologia DLT para armazenar dados de rastreabilidade dos alimentos nos processos de produção, venda e transporte. Ao mesmo tempo, por meio de IoT, o sistema ainda garante a autenticidade e confiabilidade dos dados de origem da DLT. Os resultados experimentais mostram que o sistema oferece maior segurança, menor atraso nas transações e menor custo de comunicação, comparativamente a sistemas tradicionais.

Em [Zheng and Zhou 2023], os autores propõe um modelo logístico para produtos agrícolas baseado em DLT. Esse modelo resolve questões relacionadas à circulação dos produtos, incluindo incompatibilidades entre a oferta e a procura, a má gestão da segurança da qualidade e as flutuações nos preços finais. Além disso, o artigo realiza experimentos comparativos para demonstrar a eficácia do modelo, em que se constata que é possível alcançar pelo menos 40% de receita adicional. Por fim, os autores também sugerem a possibilidade do emprego da tecnologia DLT para realizar a visualização da informação e a rastreabilidade de todo o ciclo de vida dos produtos agrícolas.

Em [Moulahi et al. 2023], os autores utilizam as tecnologias *aprendizagem federada* [Sharma and Guleria 2023] e DLT para obter proteções robustas contra ataques cibernéticos no contexto da saúde. A razão de ser dessa pequisa se deve, em grande medida, à onipresença de dispositivos IoT nos mais variados setores da área médica, a qual é percebida como uma resposta viável à escassez de profissionais de saúde. No entanto, a capacidade da área médica de utilizar essa tecnologia pode ser limitada por regras que regem o compartilhamento de dados e questões de privacidade. Especificamente, o objetivo da solução proposta nessa pesquisa é construir um sistema de aprendizagem confiável baseado em DLT que possa prever pessoas que correm risco de desenvolver diabetes. As descobertas do estudo são consideradas bem satisfatórias, pois níveis de acurácia acima de 93,95% são alcançadas nos experimentos então realizados.

Em [Passos et al. 2024], os autores propõem um sistema descentralizado de monitoramento de recursos que integra os conceitos de DLT e contratos inteligentes para ambientes distribuídos. A motivação desse trabalho se deve ao crescente número de usuários e

dispositivos conectados aos ambientes de *nuvem*, *névoa* e *borda*, o que fomenta a criação de serviços nos mais variados domínios de negócio. Esses serviços são distribuídos em infraestruturas heterogêneas que, como premissa fundamental, exigem monitoramento em tempo real. Em acordo com os experimentos realizados, o sistema proposto consegue monitorar, armazenar e transmitir continuamente medições de desempenho operacional dos serviços de forma significativamente descentralizada, com garantias de imutabilidade, rastreabilidade e níveis de segurança elevados.

Em [Huang 2024], o autor propõe um sistema de casa inteligente baseado na tecnologia DLT, buscando alcançar uma gestão eficiente de energia e controle inteligente em um ambiente de iluminação verde em áreas rurais. Sob uma visão macro e considerando metas, vislumbra-se especialmente melhorar o desempenho e a segurança de sistemas anteriores para atender às necessidades de iluminação das áreas rurais e, em simultâneo, promover o desenvolvimento sustentável. A eficácia do novo sistema é confirmada por meio de experimentos baseados em simulação, cujos resultados finais demonstram que o sistema proposto atinge baixas latências, bem como apresenta melhoria de desempenho e da segurança quando comparado a soluções anteriores.

Por fim, em [Jayakumari et al. 2024], os autores propõem um sistema de votação *online* baseado em DLT e no conceito de computação na *nuvem*. São consideradas três fases de operação do sistema: a fase de registro, a fase de emissão de votos e a fase de contagem de votos. Um protocolo de autenticação baseado em carimbo (data/hora) com assinatura digital valida eleitores e candidatos durante as fases de registro e votação. As intervenções de terceiros são eliminadas usando contratos inteligentes, e as transações são protegidas pelo emprego de DLT. Em específico, o mecanismo de consenso *Practical Byzantine Fault Tolerance* (PBFT) é adotado para garantir que a votação não é modificada ou corrompida. O sistema é avaliado como confiável, flexível, transparente, seguro e econômico, com desempenho superior ao do sistema tradicional.

3.2. Métodos de Seleção

Define-se *plataforma* DLT como um sistema baseado em DLT que, além de atender a algum propósito de um domínio de negócio, pode também vir a servir de infraestrutura tecnológica base para que outros sistemas baseados em DLT sejam desenvolvidos e executados sobre ele. Ademais de terem seus próprios componentes, os sistemas então desenvolvidos sobre a plataforma podem herdar os recursos nativos da plataforma utilizada como, por exemplo, algoritmos de consenso, técnicas criptográficas, processos de verificação e validação, além das estruturas de dados [Chowdhury et al. 2019, Nasir et al. 2018, Hang and Kim 2019]

Neste contexto, os métodos de seleção propostos na literatura orientam os projetistas na decisão sobre o mais adequado tipo de plataforma DLT para uma organização alvo. As informações que são consideradas nessa decisão pertencem a dois grupos distintos. O primeiro grupo se refere a informações ligadas a conhecimentos de tecnologia da informação (TI), sendo estas usualmente definidas por analistas de TI. O segundo grupo é constituído por informações relacionadas ao negócio da organização, sendo estas definidas pela equipe de analistas de negócio. Vê-se, assim, que a tomada de decisão não é, portanto, uma tarefa simples, pois envolve tanto fatores objetivos e, de certa forma, facil-

mente mensuráveis (i.e., conhecimento de TI), como também fatores menos objetivos e mais dificilmente mensuráveis (i.e., conhecimento do negócio). A Tabela 3 lista cronologicamente alguns trabalhos recentes e importantes da literatura que propõem métodos de seleção, além do método MAS (Seção 7) que aparece na última linha. A partir dessa tabela, podemos então observar o que se segue.

Primeiro, a maioria dos métodos têm uma *Abordagem* denominada *sequencial*, i.e., percorre-se uma séria de questões sucessivas para se chegar a uma conclusão sobre o mais adequado tipo de plataforma DLT. No caso da abordagem ser em *estágios*, tem-se as questões aglutinadas em blocos executados consecutivamente. Nesse caso, as respostas obtidas em um dado estágio se constituem na entrada do estágio seguinte. Existe ainda um único método que tem a abordagem de *mapeamento*, cuja diretriz é o estabelecimento de uma adequada interdependência entre as informações que são levantadas. Dentre as três possíveis abordagens, aquela em *estágios* possui a mesma simplicidade que a *sequencial*, com a vantagem de permitir que o *estágio* seguinte possa aproveitar, em simultâneo, todas as informações levantadas nas questões de *estágios* anteriores. A abordagem de *mapeamento* resulta como a mais complexa dentre as três, pois busca estabelecer uma função de interdependência entre vários fatores de projeto.

Segundo, a *Entrada* de quase todos os métodos busca ter informações importantes no escopo de TI da aplicação, o que diz respeito naturalmente apenas à equipe de TI. A captura de informações sob a visão de analistas de negócio não é explicitamente indicada, subtendendo-se como uma atividade que pode ser executada, quando necessária, no projeto da plataforma DLT. A exceção está apenas no método MAS, que inclui explicitamente analistas de TI e analistas de negócio, como detalhado na Seção 7. Assim, o método MAS termina sendo o mais amplo na coleta de informações para a *Entrada*.

Terceiro, quanto à *Saída*, que constitui a decisão do mais adequado tipo de plataforma DLT, tem-se o seguinte. Como é de se esperar, nenhum dos métodos fornece uma resposta contundente indicando uma plataforma DLT específica que deve ser utilizada. Em vez disso, os métodos fornecem uma resposta que indica um tipo de plataforma DLT que pode ser usada. Em quase todos os métodos, essa resposta está inserida em uma lista de tipos de plataformas DLTs com base no critério *Forma de Participação* (Subseção 4.2). A única exceção é o método MAS que, além desse critério, também considera o critério *Estruturas de Dados* (Subseção 4.1). Assim, o método MAS acaba sendo o mais preciso dentre os métodos considerados, pois leva em conta dois critérios em vez de apenas um. Note que o termo *plataforma* é deliberadamente omitido na Tabela 3 para fins de maior simplicidade e objetividade do texto nela contido.

Do exposto, pode-se concluir que o método MAS oferece uma maior probabilidade de acerto do tipo de plataforma DLT, pois: possui *Entrada* mais completa; apresenta *Abordagem* simples e capaz de fazer uso de um conjunto mais abrangente de respostas como *Entrada* para um *estágio* seguinte; e apresenta um maior conjunto de alternativas de resposta na *Saída*, o que confere uma possível maior precisão de resposta para o projeto de plataforma DLT visando a uma organização alvo.

Tabela 3. Métodos de seleção de DLTs

Referências	Abordagem	Entrada	Saída
[Peck 2017]	Sequencial	Conjunto de sete questões relacionadas aos participantes, probabilidade de ataque, confiança, inclusão de terceira parte intermediadora, privacidade, e dados atualizáveis.	São três alternativas: base de dados tradicionais; DLT permissionada; e DLT pública.
[Turk and Klinc 2017]	Sequencial	Conjunto de oito questões relacionadas a possibilidade de uso de bases de dados tradicionais, confiança, alinhamento de interesses dos participantes, possibilidade de terceira parte intermediária, controle de funcionalidade e confiança, e tipo de consenso.	São quatro alternativas: base de dados tradicionais; DLT pública; DLT híbrida; e DLT privada.
[Xu et al. 2017]	Sequencial	Conjunto de três blocos de questões envolvendo a existência de uma autoridade confiável, a possibilidade de descentralização da autoridade, possibilidade de uso de base de dados tradicionais, tipo de consenso, estruturas de dados, e outros aspectos de projeto.	São duas alternativas apenas: DLT; e uso de bases de dados tradicionais.
[Mulligan et al. 2018]	Sequencial	Conjunto de onze questões relacionadas a possibilidade de uso de bases de dados tradicionais, limitações técnicas, relacionamento entre participantes, confiança, e controle de funcionalidade.	São cinco alternativas: base de dados tracionais; não pronto para DLT, mais pesquisa se faz necessária para uso de DLT; DLT privada; e DLT pública.
[Wessling et al. 2018]	Quatro estágios	Conjunto de três estágios em sequência envolvendo a identificação dos participantes, as relações de confiança entre os participantes, e as interações existentes entre os participantes.	Consiste em um quarto estágio em que é proposta uma arquitetura inicial de projeto com base nas informações obtidas nas três etapas anteriores.
[Wüst and Gervais 2018]	Sequencial	Conjunto de seis questões envolvendo tipo de base de dados, participantes conhecidos e de confiança, alinhamento de interesses dos participantes, necessidade de verificação pública.	São quadro alternativas: base de dados tradicional; DLT privada; DLT Híbrida ou Consórcio; DLT pública.
[Hunhevicz and Hall 2019]	Mapeamento	O mapeamento consiste em estabelecer uma função entre as aplicações, nível de confiança (com base em três questões), e propriedades fundamentais da DLT	São quadro alternativas: base de dados totalmente centralizada (tradicional); DLT privada; DLT híbrida ou consórcio; e DLT pública.
[Li et al. 2019]	Sequencial	Conjunto de 14 questões que consiste em uma combinação das questões apresentadas nos trabalhos de [Peck 2017, Mulligan et al. 2018]	As mesmas cinco alternativas apresentadas no trabalho de [Mulligan et al. 2018].
[Hunhevicz and Hall 2020b]	Três estágios	Conjunto de três estágios envolvendo questões sobre a necessidade de DLT (primeiro estágio), opções de projeto da DLT (segundo estágio), e avaliação dos requisitos da aplicação (terceiro estágio)	São cinco alternativas: base de dados tradicional; DLT pública; DLT híbrida (não permissionada) ou consórcio; DLT híbrida (permissionada) ou consórcio; e DLT privada.
MAS (Seção 7)	Seis estágios	Conjunto inicial de quatro estágios consecutivos envolvendo questões sobre a organização, domínio da aplicação, forma de participação, requisitos a serem atendidos. A seguir, tem-se mais um estágio envolvendo questões sobre o tipo de consenso. Neste método há entrevistas com especialistas de Tecnologia da Informação (TI) e analistas de negócio.	Consiste em um sexto estágio que fornece as seguintes alternativas: base de dados tradicional; DLT Blockchain; e DLT DAG. Sendo Blockchain ou DAG, ainda há as seguintes subcategorias em cada uma: pública; híbrida ou consórcio; privada. Há, portanto, um total de seis alternativas.

4. Categorização

Podemos considerar os seguintes critérios para uma categorização abrangente das plataformas DLTs: i) *Estruturas de Dados* e ii) *Forma de Participação* [Chowdhury et al. 2019, Hunhevicz and Hall 2020b, Fan et al. 2020a, Fernández-Caramés and Fraga-Lamas 2018]. A Tabela 4 informa a categorização usual de algumas plataformas bem conhecidas da academia e/ou indústria para fins de exemplificação. Esses dois critérios são explicados nas subseções que seguem.

Tabela 4. Plataformas DLTs

Plataforma	Estrutura de dados	Forma de participação
Bitcoin [Nakamoto 2008], Litecoin [Litecoins 2020]	Blockchain	Pública
EOS [EOS Network Foundation 2023]	Blockchain	Híbrida ou Consórcio
Ethereum [Ethereum 2023]	Blockchain	Pública
Hyperledger Fabric [Hyperledger Foundation 2023]	Blockchain	Privada
Hashgraph [Baird 2016]	DAG	Híbrida ou Consórcio
IOTA [Popov 2018]	DAG	Pública
Byteball [Anton Churyumov 2016]	DAG	Híbrida ou Consórcio
Nano [LeMahieu 2015]	DAG	Híbrida ou Consórcio
Radix (Tempo) [Panwar and Bhatnagar 2020]	AD HOC	Pública
Corda [Brown 2018]	AD HOC	Privada
Sovrin [Tobin and Reed 2016]	AD HOC	Híbrida ou Consórcio
LTO [LTO Network 2016], Holochain [Eric Harris-Braun 2018], Monet [Arrivets 2018]	AD HOC	Privada

4.1. Estruturas de Dados

Este primeiro critério observa a organização lógica das estruturas de dados da plataforma DLT. Como já mencionado na Seção 1, as informações armazenadas são tradicionalmente os metadados das transações (i.e., tipo, tamanho, instante de ocorrência, etc.) executadas pelos atores das aplicações (e.g., médicos, clientes, advogados, etc.), além dos estados finais da base de dados (e.g., saldo de conta bancária, nível de poluição do ar, resultado de exame, etc.), gerados quando da execução das transações.

No entanto, pesquisas mais recentes preconizam que apenas as transações (i.e., metadados) devem ser armazenadas localmente (i.e., nas réplicas) pelos nós processadores. Os estados da base de dados devem ficar armazenados em sistemas de memória externos às réplicas, mais exatamente na *nuvem*. Essa concepção alia a expectativa de maior segurança e eficiência na manipulação dos dados, devido ao uso de DLTs, com a escalabilidade sistêmica, garantida pela *nuvem* [Ismail and Materwala 2020, Cerchione et al. 2022, Rajadevi et al. 2022, Zaabar et al. 2021, Rodrigues 2022, Wu et al. 2022]. Salvo dito o contrário, deixa-se subentendida essa concepção deste ponto em diante.

Com base neste primeiro critério, podemos identificar as três categorias de plataformas DLTs a seguir.

1) Blockchain

A ideia geral é que as transações executadas sejam agrupadas e armazenadas em blocos de dados. Os blocos validados são então interligados pelos valores de seus *hashes*, resultando em uma lista encadeada de blocos [Wu et al. 2022, Hunhevicz and Hall 2020a, Rodrigues 2021a]. Sob modelagem matemática e com o termo Blockchain denotando a própria estrutura de dados resultante, tem-se formalmente o que se segue [Wu et al. 2022, Hunhevicz and Hall 2020a].

A Blockchain B consiste de um conjunto de blocos β, um conjunto de arestas ℓ, a regra de aresta γ, e um conjunto de transações ordenadas τ. Cada bloco $b \in \beta$ é conectado a outro por uma aresta em τ, e cada dois blocos são conectados entre si por uma aresta sob a regra γ: usualmente o bloco mais recente aponta para o valor de *hash* do bloco mais antigo. Cada bloco b contém um subconjunto ordenado de arestas $\tau_i \subset \tau$, de forma que $\forall\, \tau_j (j \neq i) \cap \tau_i = \emptyset$. Cada bloco b também contém um conjunto de itens gerais ω, que pode encerrar, e.g., raiz da árvore de Merkle das transações, *hash* do bloco anterior, tamanho do bloco, carimbo de tempo da transação, nível de dificuldade, valor de *nonce*, dentre outros parâmetros. Em verdade, os elementos de ω diferem conforme o projeto específico da aplicação-alvo [Wu et al. 2022, Hunhevicz and Hall 2020a].

A evolução da definição original da Blockchain B segue duas abordagens, que não são necessariamente excludentes entre si [Wu et al. 2022, Hunhevicz and Hall 2020a]. A primeira abordagem se concentra em modificações na estrutura do bloco b, especialmente considerando adições e/ou remoções de itens do conjunto ω, bem como ajuste dos valores considerados para esses itens [Eyal et al. 2016, Puthal et al. 2019, Tian et al. 2019]. A segunda abordagem, a qual abriga mais propostas da literatura, se volta para modificações no encadeamento dos blocos, conforme explicado a seguir.

A propriedade da imutabilidade (Tabela 1) pode ser vista como um problema para o sistema. É notório que essa propriedade protege a integridade das informações, mas também impede que conteúdos indesejados (e.g., inconsistentes ou desatualizados) sejam removidos da base de dados. Para então ter uma Blockchain passível de remoção/correção de dados, há propostas, e.g., para substituir o item relacionado ao *hash* do bloco anterior por um item relacionado à ordenação (posição) do bloco no encadeamento [Cheng et al. 2019], ou mesmo considerar duas arestas de conectividade entre os blocos: uma para o bloco original, e outra para o novo bloco a ser considerado como eventual substituto [Deuber et al. 2019].

Neste contexto, também há propostas voltadas para diminuir o custo da armazenagem de dados da Blockchain, uma vez que toda a cadeia de blocos (desde o primeiro bloco criado, i.e., o bloco gênese) precisa, por definição, ser armazenada pelos nós processadores da rede P2P. Para tratar essa questão, existe, e.g., a possibilidade de considerar a adição de blocos especiais no encadeamento (i.e., na cadeia de blocos), os quais serviriam como blocos de checagem ou referência [Hu et al. 2020]. Analogamente, há também propostas para tornar a busca por informações na cadeia mais eficiente, as quais se baseiam, e.g., na adição de outra cadeias em paralelo à cadeia original [Fan et al. 2020b, Lunardi et al. 2020, Nunes et al. 2020, Michelin et al. 2018].

2) DAG

A ideia geral é que as transações executadas na base de dados sejam armazenadas individualmente [Popov 2018, Živic et al. 2019] ou em blocos [Živić et al. 2020, Akhtar 2019] nos vértices de um grafo acíclico direcionado (do inglês, *Directed Acyclic Graph* - DAG). A interligação dos vértices do DAG é feita por apontamento direto, conforme procedimento adotado para validação das transações. Sob modelagem matemática e com o termo DAG denotando a própria estrutura de dados resultante, tem-se formalmente o que segue [Wu et al. 2022, Hunhevicz and Hall 2020a].

O DAG D consiste de um conjunto de vértices V e um conjunto de arestas E. Cada vértice de V pode ser uma transação $t_x \in \tau$, um bloco $b \in \beta$, ou um evento $e \in \epsilon$, conforme o tipo de DAG definido. Cada aresta de E é denotada por uma tupla $< u, v >$, com $u \neq v$, que representa uma relação entre os vértices u e v de V. A existência da tupla $< u, v >$ em D implica que u referencia diretamente v (ou aponta diretamente para v), o que é representado por meio da notação $u \rightarrow v$. Neste caso, diz-se que u confirma (verifica, testemunha) v.

O DAG D possui duas importantes propriedades: (i) existe apenas uma direção na estrutura (i.e., unidirecional); e (ii) não há ciclos (i.e., acíclico). Essas duas propriedades garantem que os vértices de V somente podem ser adicionados (i.e., não há remoção) e estão ordenados entre si, analogamente ao que ocorre na Blockchain. O DAG D pode resolver o problema da escalabilidade, que é o principal obstáculo para expandir o emprego da Blockchain para certos domínios de aplicação, especialmente o ecossistema IoT.

Em função da existência de blocos, os sistemas baseados em DAG podem ser de duas categorias: (i) *Transaction-based* DAG (TDAG) e *Block-based* DAG (BlockDAG) [Wu et al. 2022, Chowdhury et al. 2019, Hunhevicz and Hall 2020a]. Sob TDAG, não há blocos de dados. Neste caso, cada transação referencia duas ou mais transações anteriormente executadas. Sob BlockDAG, o bloco existe e cada bloco contém uma lista de blocos anteriores.

Alternativamente, os sistemas baseados em DAG podem estar ainda em três outras categorias [Wu et al. 2022, Chowdhury et al. 2019, Hunhevicz and Hall 2020a]: *Naive* DAG, *Parallel Chain-based* DAG e *Main Chain-based* DAG, como descrito a seguir. Sob *Naive* DAG, tem-se a arquitetura mais simples dentre as três. Cada vértice do DAG é uma transação, e cada transação referencia duas ou mais transações anteriores. Esta categoria se encaixa dentro da categoria TDAG, citada anteriormente. Sob *Parallel Chain-based* DAG, cada vértice armazena múltiplas cadeias (de blocos ou transações) individuais que interagem entre si por meio de apontamentos definidos pelo processo de validação considerado. Por fim, sob *Main Chain-based* DAG, podem existir cadeias individuas (de blocos ou de transações) no vértice, como em *Parallel Chain-based* DAG, mas eventualmente haverá convergência para uma cadeia principal. Estas duas últimas categorias podem estar inseridas tanto na categoria TDAG como em BlockDAG, citadas anteriormente.

3) AD HOC

Esta categoria de plataforma DLT pode ser considerada ainda incipiente, não tendo atingido um grau de maturidade avançado. Isso se confirma, principalmente, pelo número

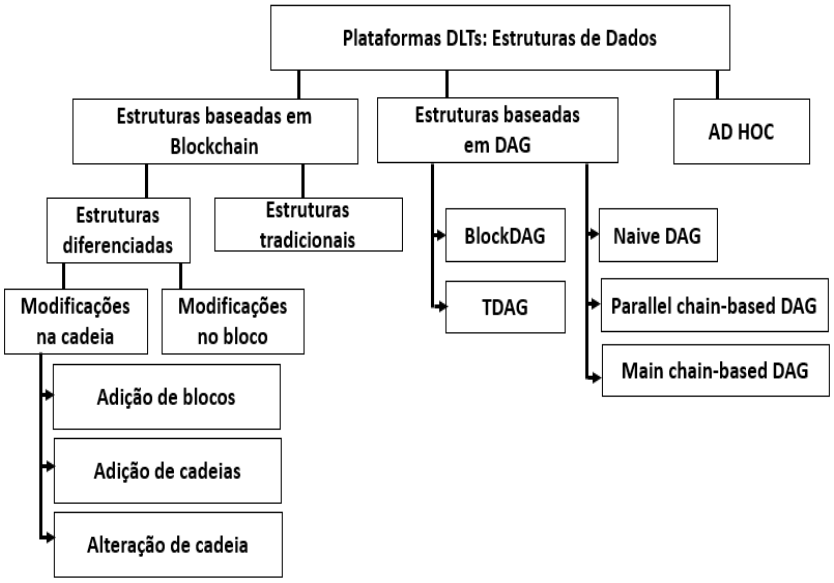

Figura 1. Categorias de plataformas DLTs sob critério Estruturas de Dados.

reduzido de publicações de trabalhos na literatura científica.

Por definição, esta categoria agrupa plataformas mais customizados para atender modelos de negócio de domínios de aplicação bem específicos. Neste contexto, os projetos dessas plataformas visam precipuamente mitigar deficiências das duas categorias de plataformas vistas anteriormente (i.e., Blockchain e DAG), com respeito especialmente à escalabilidade e à confidencialidade. Nessa categoria há uma flexibilidade com relação ao tipo de estrutura de dados a ser utilizado e à forma de operação da plataforma (e.g., atingimento de consenso entre nós e validação dos dados). Os projetos desta categoria abrem mão, portanto, de um caráter mais generalista em prol de melhor desempenho por meio de uma maior especificidade. Duas conhecidas plataformas desta categoria são brevemente comentadas a seguir para fins de mera exemplificação.

A plataforma Holochain [Eric Harris-Braun 2018, Zaman et al. 2022] tem como principal diferencial a característica de ser centrada no agente e não nos dados. Isso leva à condição de que cada nó possui e mantém a sua própria base de dados [Panwar and Bhatnagar 2020] usando um protocolo de consenso predefinido, não havendo, portanto, a necessidade de um protocolo de consenso global. Holochain tem sido particularmente considerada no desenvolvimento de aplicações móveis para o domínio da saúde no ecossistema IoT. Por sua vez, a plataforma Radix (Tempo) tem como característica diferenciadora a condição de que cada nó pode decidir manter apenas uma parte da base de dados global [Panwar and Bhatnagar 2020]. Radix (Tempo) tem sido especialmente considerada para o desenvolvimento de aplicações no domínio financeiro.

Doravante, para permitir maior facilidade na análise de DLTs, deliberadamente limitamos o escopo desta pesquisa às categorias Blockchain e DAG. A análise de DLTs pertencentes à categoria AD HOC é deixada como trabalhos futuros. Finalmente, para encerrar esta subseção, a Figura 1 traz uma síntese esquemática das categorias de plata-

formas DLTs discorridas nesta subseção.

4.2. Forma de Participação

Este segundo critério se traduz no controle e no acesso à plataforma e aos dados armazenados [Fan et al. 2020a, Hunhevicz and Hall 2020a, Wu et al. 2022, Ismail and Materwala 2019, Islam et al. 2023, Singh et al. 2022]. A Figura 2 traz as categorias de plataformas DLT que podemos identificar sob este segundo critério, as quais são explicadas a seguir.

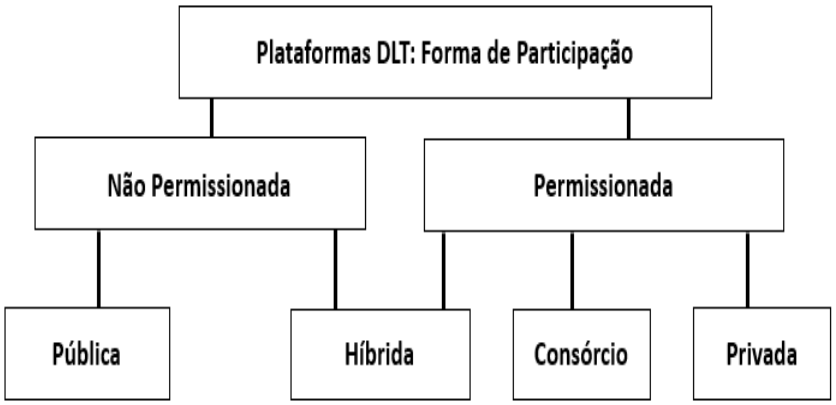

Figura 2. Categorias de plataformas DLTs sob critério Forma de Participação.

1) *Pública*

É *não permissionada*, pois não exige permissão para um nó da rede se tornar um participante, bastando apenas ter acesso à Internet. Qualquer participante pode acessar o histórico de transações e atualizar sua réplica local, bem como adicionar informações. Há um viés de descentralização, pois não há uma entidade ou grupo de entidades que realizam qualquer controle. A segurança da base de dados se baseia no uso de algoritmos de consenso que exigem significativo processamento computacional dos participantes (vide Seção 6), o que pode comprometer o desempenho final da plataforma. As transações são anônimas, mas transparentes, i.e., seus respectivos conteúdos podem ser vistos por qualquer participante. Exemplos de plataformas desta categoria incluem Bitcoin, Ethereum, Litecoin e NEO.

2) *Privada*

É *permissionada*, pois exige permissão para que um nó possa tornar-se participante. Opera em um ambiente restrito, ou seja, uma rede fechada. Está sob o controle de uma única entidade. Assim, tem-se um viés de centralização. A segurança é mais fácil de ser garantida que no caso da categoria *Pública*. Um consenso baseado em votação (vide Seção 6) é usualmente utilizado, eliminando a necessidade de alto poder de processamento computacional dos participantes, além de garantir um melhor desempenho final da plataforma. Esta categoria é mais apropriada para emprego em empresas privadas ou organizações que desejam operacionalizar estudos de caso bem específicos e inter-

nos. Exemplos de plataformas desta categoria incluem Multichain, Hyperledger Fabric, Hyperledger Sawtooth e Corda.

3) *Híbrida*

Pode ser vista como uma combinação das categorias *Pública* e *Privada*. Possui a condição de poder ser *Permissionada* ou *Não Permissionada*. Isso implica que o acesso aos dados armazenados é gerenciável. Por exemplo, um determinado subconjunto de dados pode ter acesso aberto, enquanto o restante é mantido sob a condição de confidencial. Além disso, transações e registros não são de livre acesso, mas a verificação pode ser feita publicamente se necessária, obedecendo a contratos inteligentes. Isso garante bons níveis de segurança e transparência. Existe o viés de centralização, pois a plataforma está sob o controle de uma única entidade que deseja tratar seus próprios estudos de caso. Exemplos desta categoria incluem Dragonchain e XinFin.

4) *Consórcio*

Esta categoria também é conhecida como *Federado*. Tem sua definição próxima ao da categoria *Híbrida*, pois consegue reunir em simultâneo características tanto da *Pública* quanto da *Privada*. A principal diferenciação está na condição de possuir mais de uma entidade como controladora, levando a um viés de semi-descentralização, além de ser exclusivamente *Permissionada*. Plataformas desta categoria incluem bancos, agências governamentais e outras grandes organizações como, por exemplo, Marco Polo, Energy Web Foundation e IBM Food Trust.

5. Arquitetura

Com base nos trabalhos de [Fan et al. 2020a, Dinh et al. 2017, Li and Kassem 2021], a Figura 3 traz uma arquitetura genérica de plataformas DLT. Cada uma das cinco camadas da arquitetura é brevemente explicada a seguir sob uma visão *top-down*.

1) *Camada de Aplicação/Apresentação*

Esta camada contém as aplicações que são utilizadas pelos usuários finais. Para exemplificar, citam-se os sistemas de criptomoedas, os contratos inteligentes e as DApps. Como esta camada é responsável pela apresentação dos resultados finais provenientes da plataforma DLT, existe um impacto sobre a mesma advindo das demais camadas inferiores da arquitetura.

2) *Camada de Execução*

Esta camada é responsável pela processamento e execução dos códigos computacionais que constituem as aplicações. Mais especificamente, nesta camada são executados os contratos inteligentes ou os códigos de máquina de baixo nível (i.e., *bytecodes*) em um ambiente de execução (do inglês, *runtime environment*), que está instalado em cada nó processador da rede.

As linguagens de programação e o ambiente de execução variam em função da plataforma DLT. Por exemplo, na Ethereum, há uma linguagem de máquina própria e uma máquina virtual desenvolvida para executar os contratos inteligentes, os quais são escritos na linguagem de alto nível denominada Solidity. Já na Hyperledger Fabric, os contratos

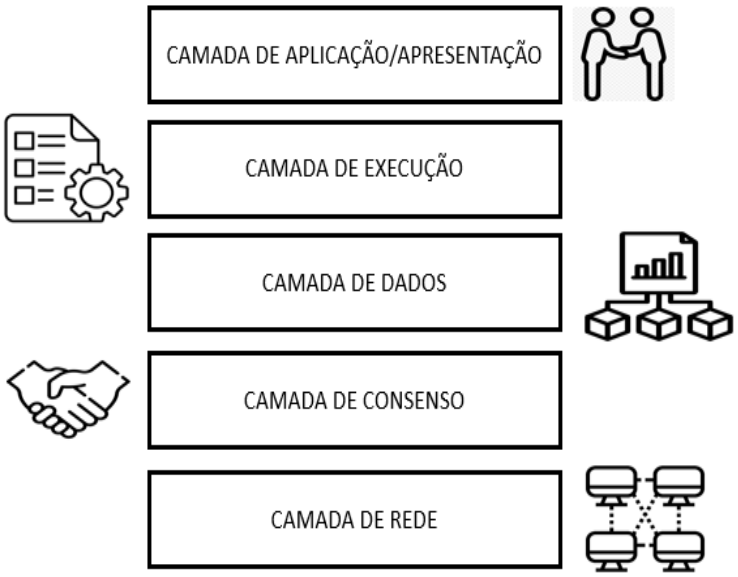

Figura 3. Arquitetura genérica de plataformas DLT.

inteligentes podem ser desenvolvidos usando diferentes linguagens de alto nível, e.g., Go, node.js e Java.

O ambiente de execução usado para executar os contratos e as transações precisa ser eficiente. É necessário garantir resultados determinísticos para evitar incertezas e inconsistências das transações em todos os nós da rede, sob pena de desperdício de recursos computacionais e deterioração de desempenho. Some-se a isso, a importância de uma adequada especificação de configuração de *hardware* (e.g., CPU e RAM) para garantir o desempenho sistêmico desejado.

3) *Camada de Dados*

Nesta camada estão os tópicos voltados para a manipulação dos dados armazenados na plataforma DLT, incluindo modelos das transações, padrões de armazenamento, funções de *hash*, algoritmos de encriptação, dentre outros. Há inúmeras alternativas e a escolha final se pauta nos requisitos que se espera atender do ponto de vista da implementação da aplicação final, a qual situa-se na primeira camada da arquitetura. Para efeito de exemplificação, a seguir tem-se uma sucinta discussão de alguns desses tópicos.

No caso de sistemas de criptomoedas, há dois modelos conhecidos de transação: *Unspent Transaction Output* (UTXO) e *Account*. Para UTXO, são considerados dois passos: (i) o possuidor do recurso assina a transação com a sua chave privada, e o destinatário é identificado a partir de sua chave pública; (ii) após uma busca para confirmação do possuidor da transação do lado do emissor, a transação é então efetivada, i.e., o montante é transferido de uma conta para outra. Para *Account*, observa-se uma possível maior eficiência, pois sua implementação permite que, atomicamente (i.e., em um só passo), as duas contas envolvidas na transação sejam atualizadas quando da transferência de um montante.

Para o caso de plataformas que usam blocos de transações, tem-se que esses blocos e aqueles contendo os estados de execução dos contratos inteligentes são interligados entre si, constituindo uma lista encadeada. Isso é feito colocando-se o *hash* do conteúdo do bloco anterior no cabeçalho do bloco corrente. Para o armazenamento dos conteúdos dos blocos, a depender da plataforma (e.g., Ethereum e HyperLedger Fabric), tem-se a alternativa de se fazê-lo em dois níveis e utilizar o paradigma NoSQL, e.g., LevelDB, CouchDB e RocksDB. Isso é feito sob a justificativa de prover uma maneira mais ágil e fácil de armazenar, consultar e comparar dados armazenados.

Por fim, citamos as alternativas de projeto relacionadas às funções de *hash* (e.g., SHA 256 v.s. SHA 128), aos algoritmos de encriptação (e.g., RSA v.s. ECC), e aos diferentes tamanhos de blocos de dados (e.g., 1 MB, 2 MB, etc.) que podem ser usados. Esses aspectos, em conjunto com aqueles mencionados anteriormente, devem ser adequadamente definidos na camada de *Dados*, pois impactam diretamente o desempenho do sistema.

4) *Camada de Consenso*

Nesta camada está o algoritmo de consenso utilizado na plataforma. Esse algoritmo define as regras que os nós processadores da rede P2P devem seguir para chegar a um acordo sobre o conteúdo a ser armazenado ou, em outras palavras, para confirmar as transações por meio de processos de verificação e validação [Bamakan et al. 2020, Singh et al. 2022, Islam et al. 2023, Ferdous et al. 2021, Wu et al. 2022, Ismail and Materwala 2019, Oyinloye et al. 2021, Nguyen and Kim 2018]. Devido à importância e variedade hoje existente de algoritmos de consenso, tem-se a discussão deliberadamente adiada para a Seção 6.

5) *Camada de Rede*

A caracterização da rede de conexão dos nós processadores da rede P2P ocorre nesta camada. Sobre a dimensão geográfica dessa rede, pode-se afirmar o seguinte. No caso da plataforma *Pública*, como o sistema Bitcoin, a rede tem dimensão significativa, com milhares de nós trabalhando para chegar a um consenso. No caso de plataformas das categorias *Privada*, *Híbrida* e *Consórcio*, a escala é usualmente mais modesta, podendo variar de alguns nós até centenas. Em qualquer dos casos, o requisito básico desta camada é sempre buscar garantir rapidez de processamento das transações e estabilidade sistêmica.

Sobre a entrada de novos nós processadores na rede, tem-se o seguinte. Quando um novo nó processador se junta ao sistema, esta camada é responsável por garantir que os nós antigos possam imediatamente reconhecer esse novo nó. Para tanto, todos os nós se comunicam entre si por meio de mensagens para haver sincronização e, assim, manter o estado consistente da base de dados. Além disso, o envio em *broadcast* de transações, a verificação de transações, o armazenamento de transações, e a propagação do estado global da base de dados também são de responsabilidade desta camada.

Quanto aos tipos de nós processadores da rede, existem usualmente duas possibilidades: *full* e *light*. Os nós *full* são responsáveis por todo o processamento das transações (verificação, confirmação, etc.) e execução das regras de consenso, enquanto que os nós

light se comportam majoritariamente como clientes, apenas submetendo transações à base de dados.

Ante o exposto, conclui-se então que, comparativamente às camadas superiores da arquitetura, esta camada é aquela que é mais especialmente afetada pelos aspectos físicos constituintes da conexão entre os nós processadores, incluindo, por exemplo, latência dos meios de conexão entre os nós, taxa de transmissão das placas de rede dos nós, capacidade de processamento computacional dos nós, topologia física (além de lógica) de conexão entre os nós e, ainda, taxa de perda de mensagens na rede.

6. Algoritmos de Consenso

Proof-of-Work (PoW) foi primeiro algoritmo de consenso para plataforma DLT [Bamakan et al. 2020, Singh et al. 2022, Islam et al. 2023]. Mais especificamente, PoW foi usado na plataforma DLT de categoria Blockchain que implementou a base de dados do sistema Bitcoin [Nakamoto 2008]. Sob este algoritmo, blocos de transações somente podem ser adicionados à base de dados quando existe uma prova de que houve a realização de um trabalho computacional significativo dentro de um certo intervalo de tempo.

A prova da realização de trabalho computacional do PoW existe quando nós *mineradores* (assim nomeados os nós processadores da rede) apresentam uma solução matemática para um difícil desafio criptográfico baseado em uma função de *hash*, sendo essa solução verificável por todos os demais nós *mineradores* da rede. O processo de criação e adição de blocos é chamado de *mineração* e seu resultado é a lista encadeada de blocos, mencionada na Subseção 4.1.

Em que pese sua apreciável robustez contra remoção e alteração de informações da base de dados, o algoritmo PoW possui as seguintes principais desvantagens: (i) uma exigência de alto poder de processamento computacional para solução do desafio criptográfico, levando a um alto consumo de energia elétrica; (ii) uma alta latência para confirmação das transações por toda a rede; e (iii) uma baixa escalabilidade do sistema.

Na busca de soluções alternativas ao PoW, surgiu uma ampla variedade de outros algoritmos de consenso ao longo do tempo, os quais podem ser agrupados em classes de acordo com suas principais características. Neste contexto, esta subseção discute então brevemente as principais classes de algoritmos de consenso, bem como cita alguns exemplos da literatura [Bamakan et al. 2020, Singh et al. 2022, Islam et al. 2023, Ismail and Materwala 2019, Machacek et al. 2021, Prisco et al. 2000, Wang et al. 2023].

Da Tabela 5, temos o que segue. São três classes de algoritmos: *Proof-of-X*, *Vote-based* (*Leader-based*) e DAG. Para cada uma dessas classes, são informadas filosofia de operação, objetivos fulcrais e subclasses. Por meio dessas informações, notam-se as mais importantes peculiaridades dos algoritmos em função da classe em que estão inseridos. Por exemplo, quando uma plataforma tem como maior prioridade a segurança dos dados (e.g., robustez contra alterações maliciosas), a classe *Proof-of-X* tende a ser a melhor opção. Por outro lado, quando a prioridade é o desempenho final (e.g., rapidez de processamento de transações), a melhor opção pode estar entre as classes *Vote-based* e DAG.

Em complemento à Tabela 5, tem-se a Figura 4. Nesta figura temos a citação de exemplos da literatura, considerando as subclasses identificadas anteriormente na tabela. É possível notar o grande número de possibilidades. A escolha de um algoritmo em detrimento do outro não é absoluta, devendo estar sempre pautada na observação estrita das prioridades que se desejam atender quando do desenvolvimento da plataforma. Por exemplo, se o consumo de energia elétrica é restrito na região geográfica da implantação da plataforma DLT, então os algoritmos da subclasse *Computação intensiva* são proibitivos. Por outro lado, se a rede de conexão dos nós processadores apresenta problemas de continuidade de serviço, ocasionando a inoperância de nós, então os algoritmos da subclasse *Crash Fault Tolerance* são alternativas bem recomendáveis.

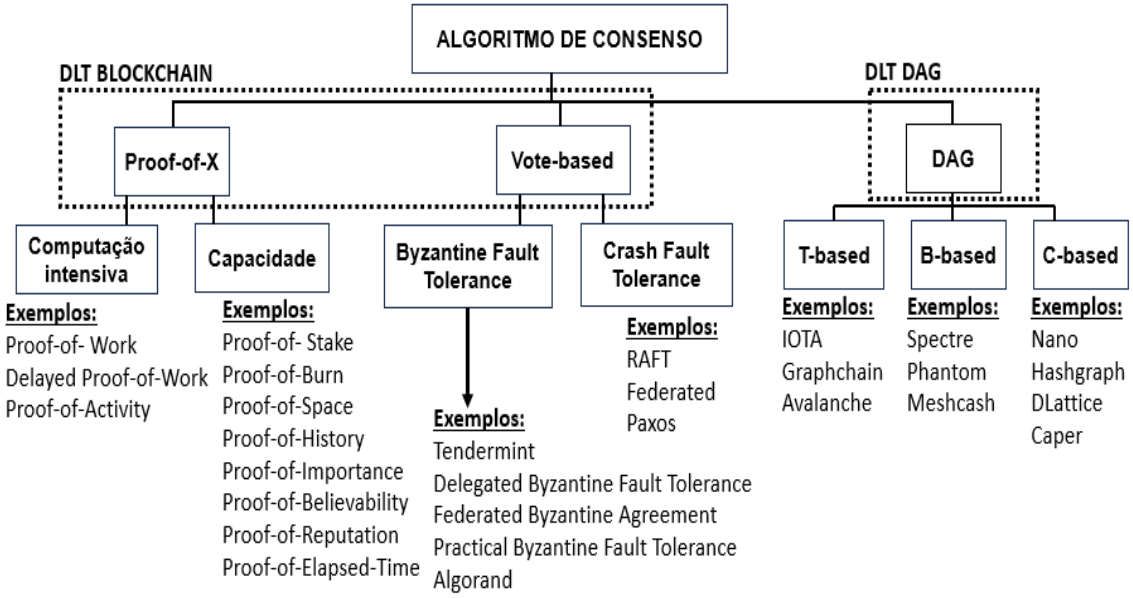

Figura 4. Algoritmos de consenso.

7. Método ágil de Seleção

O *Método Ágil de Seleção* (MAS) possui seis estágios a serem executados sob a condução da equipe de TI, conforme ilustrado na Figura 5. Os quatro primeiros estágios são executados em sequência. Ao término do quarto estágio, os dois últimos estágios são então executados em sequência. As saídas dos quatro primeiros estágios são encaminhadas como subsídios para o quinto estágio. A saída do quinto estágio é então encaminhada para o sexto estágio, onde ocorre a escolha do tipo de plataforma DLT. Para facilidade de entendimento e organização, as subseções a seguir detalham individualmente cada um dos seis estágios constituintes do MAS.

7.1. Organização

O MAS prevê em seu estágio inicial o conhecimento pleno da organização onde a plataforma DLT será implantada. Esse conhecimento é obtido por meio do estudo e análise de documentos técnicos e administrativos, bem como por meio de entrevistas com os analistas de negócio.

Tabela 5. Classes de algoritmos de consenso

Classe	Filosofia de operação	Objetivos fulcrais do projeto	Subclasses
Proof-of-X	Os nós da rede precisam apresentar algum tipo de prova, com base em um dado critério, para então poderem adicionar informação à base de dados. O foco é a aplicação em DLTs da categoria Blockchain.	Garantir alto nível de segurança, traduzido pela robustez contra remoção e alteração de informações registradas.	(i) *Computação intensiva*: o consenso depende da capacidade de processamento computacional dos nós (e.g., para resolução de um desafio criptográfico). Os nós que conseguem mais rapidamente apresentar a prova são os escolhidos, estabelecendo uma competição entre os nós da rede. Como principal desafio, está a diminuição do alto consumo de energia elétrica por parte dos nós. (ii) *Capacidade*: o consenso é baseado na prova de disponibilidade de algum recurso que os nós possuem (e.g., tempo, espaço, reputação). Os nós de maior capacidade têm mais chances de serem escolhidos para adicionar informação. O principal desafio é mitigar a tendência de que nós de maiores capacidades tenham sempre as maiores capacidades, redundando em um indevido monopólio.
Vote-based	Os nós que adicionam informação são escolhidos por meio de uma votação. O foco é a aplicação em DLTs da categoria de Blockchain. Deve-se tolerar falhas bizantinas, assumindo que há falhas de nós independentes na rede ou que alguns dos nós podem se comportar de forma maliciosa. A tolerância a falhas bizantinas (do inglês, *Bizantine Fault Tolerance* (BFT)) é a capacidade da rede de alcançar o consenso, apesar de alguns nós do sistema falharem (i.e., tornarem-se inoperantes) ou estarem se comportando maliciosamente.	(i) Mitigar o alto consumo de energia elétrica por parte dos nós que adicionam informação, inerente aos algoritmos *Proof-of-X* baseados em *computação intensiva*; (ii) Mitigar o problema de os nós poderem vir a ter suas capacidades cada vez mais aumentadas em seleções subsequentes, inerente aos algoritmos *Proof-of-X* baseados em *capacidade*; (iii) Aumentar a vazão (i.e., número de transações por unidade de tempo) observada nos algoritmos *Proof-of-X*.	(i) *Byzantine Fault Tolerance*: os algoritmos de consenso toleram tanto o caso de nós que falham (i.e., nós inoperantes) quanto o caso de nós que agem maliciosamente (i.e., nós não confiáveis). (ii) *Crash Fault Tolerance*: os algoritmos de consenso toleram apenas o caso de nós que falham (i.e., nós inoperantes).
DAG	O foco é a aplicação em DLTs da categoria DAG. Os algoritmos ainda não estão em um estágio de desenvolvimento avançado. Os algoritmos usualmente não têm denominação própria, mas são referidos pelo sistema onde são aplicados. Por exemplo, IOTA é a denominação do algoritmo de consenso utilizado no sistema de criptomoeda IOTA Como principal característica diferenciadora, não há escolha de um nó ou de um conjunto de nós para adicionar informação à base de dados. Todos os nós da rede podem ter essa função.	Descentralizar o poder de adição de informação, permitindo uma maior eficiência e uma maior escalabilidade do sistema. Múltiplas unidades de dados (transações ou blocos de transações) podem ser adicionadas simultaneamente.	(i) *T-based*: os algoritmos trabalham com unidades de dados constituídas apenas por transações individuais. A topologia resultante de interligação dessas unidades é um grafo que se expande progressivamente. (ii) *B-based*: os algoritmos trabalham com unidades de dados constituídas por blocos de individuais. A topologia resultante de interligação dessas unidades é também um grafo que se expande progressivamente. (iii) *C-based*: os algoritmos trabalham com unidades de dados constituídas apenas por transações individuais. Todavia, diferentemente do que ocorre na subclasse *T-based*, a topologia resultante de interligação é um grafo com um conjunto de múltiplas cadeias de unidades de dados em paralelo.

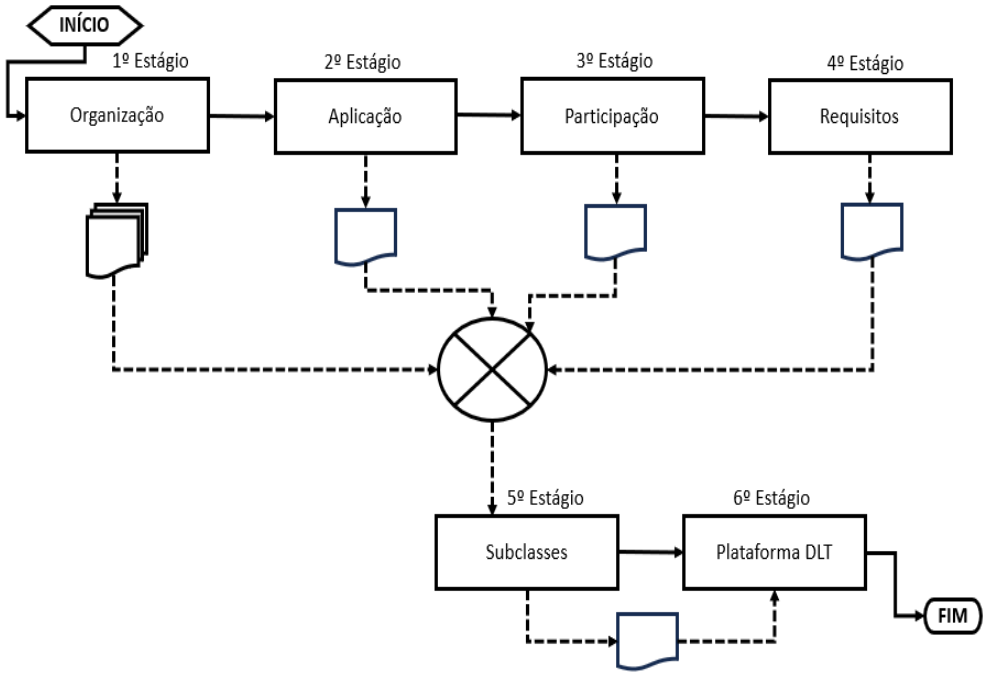

Figura 5. Metodologia Ágil de Seleção (MAS).

De forma breve, os documentos técnicos encerram a descrição da infraestrutura de TI, considerando aspectos de *hardware* e *software* da organização. Por sua vez, os documentos administrativos devem refletir o entendimento da organização e sua forma de operação. Ao fim deste estágio, ao menos as seguintes perguntas devem ser respondidas sobre a organização:

- 1) Qual é a definição (Missão, visão e valores)?
- 2) Qual é o domínio de negócio (Vide Tabela 6)?
- 3) Qual é o organograma funcional?
- 4) Quais são os processos organizacionais e gerenciais?
- 5) Quantos são os colaboradores e os clientes?
- 6) Quais são os recursos disponíveis para investimento?
- 7) Existe motivação (estudo de viabilidade) justificada (sob a luz de TI e de negócio) de que uma plataforma DLT é de fato recomendável? Em caso negativo, os demais estágios do MAS não são executados, e a recomendação é o uso de base de dados tradicional.

Este estágio existe porque a decisão da mais adequada plataforma depende da definição da organização e dos recursos nela disponíveis. Por exemplo, uma opção pela classe de algoritmos de consenso *Proof-of-X* pressupõe a necessidade de um custo econômico maior que o da classe DAG, ou seja, a organização deve ter condições de arcar com esse investimento.

7.2. Aplicação

Ao fim deste segundo estágio do MAS, deve haver um pleno entendimento da aplicação computacional a ser desenvolvida, considerando suas implicações especialmente relacio-

nadas às necessidades de TI. Ao menos, as seguintes perguntas devem ser respondidas:

- 1) Qual é a aplicação (sistema) computacional a ser desenvolvida (entradas, saídas, requisitos, etc.)?
- 2) Existe infraestrutura de TI suficiente para implantar qualquer uma das opções de subclasses de algoritmos de consenso? Se não, quais são as subclasses de algoritmos que podem ser consideradas como alternativas?

A importância deste estágio se revela pelo fato de ser a oportunidade de definir a infraestrutura de TI necessária para a aplicação alvo, o que afeta a decisão final acerca da subclasse de algoritmos de consenso e, consequentemente, da plataforma DLT a ser implantada.

Tabela 6. Entendimento sintético dos domínios

Domínio	Explicação
Comércio Eletrônico	Compreende qualquer tipo de negócio/transação comercial que implica a transferência de informação através da Internet.
Entretenimento	Compreende qualquer ação, evento ou atividade com o fim de entreter e suscitar o interesse de uma audiência.
Logística	Compreende uma especialidade da administração e engenharia responsável por prover recursos e informações para a execução de todas as atividades de uma organização.
Assistência Médica	Compreende o tratamento de doenças e a preservação da saúde por meio de serviços médicos, farmacêuticos, enfermagem e outras profissões relacionadas.
Transporte	Compreende toda atividade relacionada ao processo de movimento de pessoas e mercadorias entre locais, incluindo diversas características a nível de infraestrutura, operações comerciais, veículos, navios, aviões, dentre outros.
Votação Eletrônica	Compreende o processo baseado no emprego de uma urna eletrônica ou máquina de votação, a qual consiste em uma combinação de equipamentos mecânicos, eletromecânicos ou eletrônicos, usada para contagem de votos e para manter e produzir qualquer informação de trilha de auditoria.
Internet das Coisas (IoT)	Compreende um conceito que se refere à interconexão digital de objetos eletrônicos cotidianos com a Internet, em especial mais para a comunicação entre objetos do que entre objetos e pessoas.
Pagamento Eletrônico	Compreende um sistema de liquidação de bens e serviços pela Internet entre organizações financeiras, organizações comerciais e usuários da Internet.

7.3. Participação

Este estágio trata sobre o controle e o acesso à plataforma e à base de dados, cuja discussão está na Subseção 4.2. Como visto, a plataforma DLT pode ser categorizada como *Pública*, *Privada*, *Consórcio* e *Híbrida*. Dependendo da categoria, a plataforma pode ainda ser *Permissionada* ou *Não Permissionada*, como ilustrado na Figura 2.

A importância deste estágio se evidencia em virtude da necessidade de se definir de forma contundente o nível de segurança que se deseja ter para fins de controle e acesso à plataforma e às informações armazenadas na base de dados, o que naturalmente influencia na escolha da plataforma DLT.

7.4. Requisitos

Para a execução deste estágio, são considerados os sete requisitos operacionais definidos na Tabela 7 [Hunhevicz and Hall 2020b]. Ao seu término, deve haver a indicação dos valores dos *pesos* de cada um dos desses requisitos. Esses valores estão em um intervalo de 1 a 10, sendo determinados pelos especialistas de TI com base nas opiniões dos analistas de negócio.

Explica-se que o *peso* atribuído a um requisito reflete a importância relativa dele. Por exemplo, se o *peso* da *Eficiência* é 3 e os respectivos pesos de todos os demais requisitos é 1, então a *Eficiência* é três vezes mais importante que qualquer outro requisito. Enfim, a importância deste estágio se revela por garantir que a plataforma DLT atende os requisitos definidos, em conformidade com a visão de prioridade fornecida pela própria organização.

Tabela 7. Requisitos operacionais

Requisito	Explicação
Vazão	Taxa de transações efetivadas na base de dados por unidade de tempo.
Armazenamento	Capacidade de armazenamento da base de dados.
Interoperabilidade	As informações são compartilhadas entre as réplicas da base de dados, com garantias de consistência e integridade [Rodrigues and da Silva 2019].
Privacidade e Confidencialidade	Estes requisitos devem ser garantidos sob a luz da Lei Geral de Proteção de Dados Pessoais (LGPD) [Presidência da República do Brasil 2019].
Contratos inteligentes	Devem ser implementados na camada de aplicação (Figura 3) de forma a garantir o conjunto de procedimentos e regras que tem como objetivo manter a organização em linha com as normas vigentes, sejam elas externas ou internas (Seção 2).
Custo econômico	Refere-se ao custo econômico do projeto, implantação e operação da plataforma.
Integridade	Refere-se ao requisito da corretude das informações armazenadas na base de dados.
Disponibilidade	Refere-se ao requisito de as informações estarem sempre disponíveis para serem consultadas.

7.5. Subclasses

A discussão sobre as classes e subclasses dos algoritmos de consenso está na Seção 6. Ao término deste estágio, deve haver a identificação da subclasse dos algoritmos de consenso a ser considerada para a escolha da plataforma DLT. Para tanto, os os três passos a seguir devem ser executados.

1) Especialistas de negócios devem individualmente votar nos requisitos do estágio anterior. Ao final da votação, cada requisito contará com um certo número de votos totalizados;

2) Cada requisito tem sua prioridade dada pela fórmula: $P = v \times p$, onde v é o número de votos do requisito, e p é o peso definido do requisito no estágio anterior (Subseção 7.4). Ao final, cada requisito tem então uma prioridade individual calculada;

3) Os requisitos são então dispostos em ordem decrescente de P. Essa ordenação é então utilizada para se escolher a subclasse de algoritmo de consenso mais aderente sob a luz da Tabela 5.

7.6. Plataforma DLT

Este estágio trata sobre a escolha final do tipo mais adequado de plataforma DLT para uma organização alvo. Para tanto, os dois passos a seguir devem ser executados.

1) Com base na Figura 4 e conhecida a subclasse de algoritmos de consenso (Subseção 7.5), identificar a correspondente classe;

2) Sob o critério *Estruturas de Dados* e conhecida a classe de algoritmos de consenso, identificar então a categoria da plataforma DLT também com base na Figura 4.

Lembramos que a categorização da plataforma DLT, sob o critério *Forma de Participação*, é previamente definida no estágio da Subseção 7.3. Note que a execução dos dois passos acima tem seis alternativas de resposta para o tipo de plataforma DLT, conforme indicado na Tabela 3. Em complemento, para o projeto das estruturas de dados e arquitetura da plataforma, tem-se os subsídios apresentados na Subseção 4.1 (vide Figura 1) e na Seção 5 (vide Figura 3).

8. Estudo de Caso Simplificado

Esta seção apresenta um estudo de caso simplificado para essencialmente demonstrar a aplicabilidade do MAS. A organização considerada para efeito de demonstração é o sistema de saúde público do Brasil, denominado Sistema Único de Saúde (SUS) [Ministério da Sáude do Brasil 2024].

Para fins de maior objetividade e facilidade de entendimento, a realização deste estudo de caso é feita por meio de discussão sucinta de cada um dos estágios constituintes do MAS, conforme apresentado na Tabela 8. Dessa tabela, nota-se que o tipo de plataforma DLT indicada é: Blockchain privada. Adicionalmente, também se conclui que o algoritmo de consenso deve pertencer à classe *Vote-based* e à subclasse BFT.

Em que pese a simplificação do estudo de caso realizado, podemos constatar a simplicidade de aplicação do MAS para a seleção do tipo de plataforma DLT. Em especial, comparativamente a alguns outros métodos propostos na literatura (vide Subseção 3.2, Tabela 4), nota-se inclusive a maior completude da resposta final alcançada em virtude da adicional indicação de classe e de subclasse do algoritmo de consenso.

9. Conclusões Gerais e Trabalhos Futuros

Este artigo apresentou um abrangente arcabouço teórico sobre DLTs para implementação de bases de dados distribuídas de aplicações computacionais e, também, propôs um método de seleção do tipo de plataforma DLT para uma organização, denominado de *Método Ágil de Seleção* (MAS).

O arcabouço teórico dissertou sobre importantes tópicos relativos ao projeto de plataformas DLT, incluindo contratos inteligentes e algoritmos de consenso, além de uma criteriosa taxonomia. Ainda, para fins de demonstração da aplicabilidade do MAS, foi realizado um estudo simplificado considerando o Sistema Único de Saúde (SUS) do Brasil, o qual permitiu evidenciar sua simplicidade de aplicação e, em comparação a outros métodos da literatura, sua maior precisão e completude na resposta final.

Tabela 8. Estudo de caso simplificado

Estágio	Discussão	Resultado
1º	O SUS atende a uma população de aproximadamente 214 milhões de cidadãos em 27 unidades federativas que abrangem mais de 8,5 milhões de km^2. O propósito da plataforma DLT é permitir o gerenciamento de Prontuários Médicos Eletrônicos (PMEs) dessa população. O domínio é Assistência Médica. Por simplificação, não são feitas entrevistas. Assume-se a existência de estudo de viabilidade que ratifica a necessidade da implantação de plataforma DLT. As informações utilizadas nesta pesquisa se baseiam em documentos públicos disponíveis no próprio sítio da organização, em agências governamentais e em trabalhos acadêmicos (e.g., [Rodrigues 2021a, Cerchione et al. 2022]).	Organograma, processos, colaboradores e clientes podem ser identificados em documentos públicos. Os recursos disponíveis para investimento variam em acordo com a arrecadação federal do Brasil, sendo aqui, por simplificação, considerados irrestritos.
2º	Os PMEs devem ser armazenados em uma robusta base de dados, possibilitando o compartilhamento entre diferentes atores do SUS (e.g., clínicas, hospitais, etc.) e, também, entre o SUS e diferentes atores externos (e.g., laboratórios privados, centros de pesquisa, etc.).	Assume-se o pleno entendimento da aplicação, bem como assume-se a infraestrutura de TI existente suficiente para implantar qualquer uma das opções de subclasses de algoritmos de consenso.
3º	A confidencialidade e a privacidade devem ser implementadas sob a luz da Lei Geral de Proteção de Dados Pessoais (LGPD) [Presidência da República do Brasil 2019].	DLT *Privada* (Subseção 4.2, Figura 2).
4º	Por simplificação, não são realizadas entrevistas com analistas de negócio. Dessa forma, não há priorização (ranqueamento) para os sete requisitos operacionais definidos na Tabela 7.	Os requisitos definidos na Tabela 7 possuem a mesma prioridade.
5º	Por simplificação, não são realizadas entrevistas com analistas de negócio. Dessa forma, a ordenação resultante se baseia em documentos públicos disponíveis no próprio sítio da organização, em agências governamentais e em trabalhos acadêmicos (e.g., [Rodrigues 2021a, Cerchione et al. 2022]).	Assume-se a ordenação: privacidade e confidencialidade; integridade; vazão; interoperabilidade; disponibilidade; contratos inteligentes; armazenamento. Da Tabela 5 e com o requisito vazão precedendo o requisito disponibilidade, a subclasse indicada é *Byzantine Fault Tolerance*.
6º	Conhecida a subclasse de algoritmos de consenso, identifica-se a correspondente classe com base na Figura 4. Sob o critério *Estruturas de Dados* e conhecida a classe de algoritmos de consenso, identifica-se a categoria da plataforma também a partir da Figura 4.	A classe de algoritmos de consenso é *Vote-based*. A categoria da DLT é Blockchain. Lembramos que a mesma é *Privada*, conforme visto no 3º Estágio.

Como pesquisas futuras e ante as limitações desta pesquisa, podemos apontar os seguintes caminhos: (i) estudar a categoria de DLT denominada AD HOC para fins de sua inclusão no MAS, tornando-o mais completo; e (ii) aplicar o MAS em estudos de casos de variados domínios para fins de ratificação e/ou aperfeiçoamento metodológico.

Referências

[Akhtar 2019] Akhtar, Z. (2019). From Blockchain to Hashgraph: Distributed Ledger Technologies in the Wild. In *2019 International Conference on Electrical, Electronics and Computer Engineering (UPCON)*, pages 1–6.

[Anton Churyumov 2016] Anton Churyumov (2016). Byteball: A Decentralized System for Storage and Transfer of Value. [Online] Available at: https://obyte.org/Byteball.pdf. Accessed on: April 16th, 2024.

[Arrivets 2018] Arrivets, M. (2018). Monet: Mobile ad hoc blockchains. [Online] Available at: https://www.coinpare.io/whitepaper/monet.pdf. Accessed on: April 16th, 2024.

[Baird 2016] Baird, L. (2016). The Swirdls Hashgraph Consensus Algorithm: Fair, Fast, Byzantine Fault Tolerance. [Online] Available at: https://www.swirlds.com/downloads/SWIRLDS-TR-2016-01.pdf. Accessed on July 30th, 2024.

[Bamakan et al. 2020] Bamakan, S. M. H., Motavali, A., and Bondarti, A. B. (2020). A survey of blockchain consensus algorithms performance evaluation criteria. *Expert Systems with Applications*, 154:113385.

[Brown 2018] Brown, R. G. (2018). The Corda Platform: An Introduction. [Online] Available at: https://corda.net/content/corda-platform-whitepaper.pdf. Accessed on Jan. 26th, 2024.

[Cerchione et al. 2022] Cerchione, R., Centobelli, P., Riccio, E., Abbate, S., and Oropallo, E. (2022). Blockchain's coming to hospital to digitalize healthcare services: Designing a distributed electronic health record ecosystem. *Technovation*, page 102480.

[Cheng et al. 2019] Cheng, L., Liu, J., Su, C., Liang, K., Xu, G., and Wang, W. (2019). Polynomial-based modifiable blockchain structure for removing fraud transactions. *Future Generation Computer Systems*, 99:154–163.

[Chowdhury et al. 2019] Chowdhury, M. J. M., Ferdous, M. S., Biswas, K., Chowdhury, N., Kayes, A. S. M., Alazab, M., and Watters, P. (2019). A Comparative Analysis of Distributed Ledger Technology Platforms. *IEEE Access*, 7:167930–167943.

[Cohen 2003] Cohen, B. (2003). Incentives build robusteness in BitTorrent. In *First Workshop on Economics of Peer-to-Peer System*, Berkeley, USA.

[Deuber et al. 2019] Deuber, D., Magri, B., and Thyagarajan, S. A. K. (2019). Redactable Blockchain in the Permissionless Setting. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 124–138.

[Dinh et al. 2017] Dinh, T. T. A., Wang, J., Chen, G., Liu, R., Ooi, B. C., and Tan, K.-L. (2017). Blockbench: A framework for analyzing private blockchains. In *Proceedings of the 2017 ACM International Conference on Management of Data*, SIGMOD '17, page 1085–1100, Chicago, Illinois, USA.

[EOS Network Foundation 2023] EOS Network Foundation (2023). Introducing EOS. [Online]. Available at: https://eosnetwork.com/introducing-eos/. Accessed on Jan. 26th, 2024.

[Eric Harris-Braun 2018] Eric Harris-Braun, Nicolas Luck, A. B. (2018). Holochain: scalable agent-centric distributed computing. [Online] Available at: https://www.allcryptowhitepapers.com/holo-whitepaper/. Accessed on Feb. 23rd, 2024.

[Ethereum 2023] Ethereum (2023). Ethereum Platform. [Online] Available at: https://ethereum.org/pt-br/. Accessed on: Feb. 23rd, 2024.

[Eyal et al. 2016] Eyal, I., Gencer, A. E., Sirer, E. G., and Van Renesse, R. (2016). Bitcoin-NG: A Scalable Blockchain Protocol. In *Proceedings of the 13th Usenix Conference on Networked Systems Design and Implementation*, NSDI'16, page 45–59, USA. USENIX Association.

[Fan et al. 2020a] Fan, C., Ghaemi, S., Khazaei, H., and Musilek, P. (2020a). Performance evaluation of blockchain systems: A systematic survey. *IEEE Access*, 8:126927–126950.

[Fan et al. 2020b] Fan, Y., Zou, J., Liu, S., Yin, Q., Guan, X., Yuan, X., Wu, W., and Du, D. Z. (2020b). A blockchain-based data storage framework: A rotating multiple random masters and error-correcting approach. *Peer-to-Peer Networking and Applications*, 13:1486 – 1504. [Online]. Available at: https://api.semanticscholar.org/CorpusID:216292006. Accessed on: July 30th, 2024.

[Ferdous et al. 2021] Ferdous, M. S., Chowdhury, M. J. M., and Hoque, M. A. (2021). A survey of consensus algorithms in public blockchain systems for crypto-currencies. *Journal of Network and Computer Applications*, 182:103035.

[Fernández-Caramés and Fraga-Lamas 2018] Fernández-Caramés, T. M. and Fraga-Lamas, P. (2018). A Review on the Use of Blockchain for the Internet of Things. *IEEE Access*, 6:32979–33001.

[Garcia and Kleinschmidt 2020] Garcia, P. S. R. and Kleinschmidt, J. H. (2020). Sharing Health and Wellness Data with Blockchain and Smart Contracts. *IEEE Latin America Transactions*, 18(06):1026–1033.

[Hang and Kim 2019] Hang, L. and Kim, D. (2019). Design and Implementation of an Integrated IoT Blockchain Platform for Sensing Data Integrity. *Sensors*, 10:2228.

[Hu et al. 2020] Hu, Y., Kumar, S., and Popa, R. A. (2020). Ghostor: Toward a Secure Data-Sharing System from Decentralized Trust. In *Proceedings of the 17th Usenix Conference on Networked Systems Design and Implementation*, NSDI'20, page 851–878, USA. USENIX Association.

[Huang 2024] Huang, Y. (2024). Smart home system using blockchain technology in green lighting environment in rural areas. *Heliyon*, 10(4):e26620.

[Hunhevicz and Hall 2019] Hunhevicz, J. J. and Hall, D. M. (2019). Managing mistrust in construction using DLT: a review of use-case categories for technical decisions. *Proceedings of the 2019 European Conference on Computing in Construction*.

[Hunhevicz and Hall 2020a] Hunhevicz, J. J. and Hall, D. M. (2020a). Do you need a blockchain in construction? Use case categories and decision framework for DLT design options. *Advanced Engineering Informatics*, 45:101094.

[Hunhevicz and Hall 2020b] Hunhevicz, J. J. and Hall, D. M. (2020b). Do you need a blockchain in construction? Use case categories and decision framework for DLT design options. *Advanced Engineering Informatics*, 45:101094.

[Hyperledger Foundation 2023] Hyperledger Foundation (2023). Hyperledger Fabric Platform. [Online] Available at: https://www.hyperledger.org/use/fabric. Accessed on April 16th, 2024.

[Islam et al. 2023] Islam, S., Islam, M. J., Hossain, M., Noor, S., Kwak, K.-S., and Islam, S. M. R. (2023). A Survey on Consensus Algorithms in Blockchain-Based Applications: Architecture, Taxonomy, and Operational Issues. *IEEE Access*, 11:39066–39082.

[Ismail and Materwala 2019] Ismail, L. and Materwala, H. (2019). A Review of Blockchain Architecture and Consensus Protocols: Use Cases, Challenges, and Solutions. *Symmetry*, 11(10).

[Ismail and Materwala 2020] Ismail, L. and Materwala, H. (2020). Blockchain paradigm for healthcare: Performance evaluation. *Symmetry*, 12(8).

[Jayakumari et al. 2024] Jayakumari, B., Sheeba, S. L., Eapen, M., Anbarasi, J., Ravi, V., Suganya, A., and Jawahar, M. (2024). E-voting system using cloud-based hybrid blockchain technology. *Journal of Safety Science and Resilience*, 5(1):102–109.

[Kumar et al. 2020] Kumar, G., Saha, R., Buchanan, W. J., Geetha, G., Thomas, R., Rai, M. K., Kim, T.-H., and Alazab, M. (2020). Decentralized accessibility of e-commerce products through blockchain technology. *Sustainable Cities and Society*, 62:102361.

[LeMahieu 2015] LeMahieu, C. (2015). Nano: A Feeless Distributed Cryptocurrency Network. [Online] Available at: https://media.abnnewswire.net/media/en/docs/91948-whitepaper.pdf. Accessed on July 30th, 2024.

[Li et al. 2019] Li, J., Greenwood, D., and Kassem, M. (2019). Blockchain in the built environment and construction industry: A systematic review, conceptual models and practical use cases. *Automation in Construction*, 102:288–307.

[Li and Kassem 2021] Li, J. and Kassem, M. (2021). Applications of distributed ledger technology (DLT) and Blockchain-enabled smart contracts in construction. *Automation in Construction*, 132:103955.

[Litecoins 2020] Litecoins (2020). The Future of Money. [Online] Available at: https://litecoin.com/en/. Accessed on: April 5th, 2024.

[Lone and Naaz 2021] Lone, A. H. and Naaz, R. (2021). Applicability of Blockchain smart contracts in securing Internet and IoT: A systematic literature review. *Computer Science Review*, 39:100360.

[LTO Network 2016] LTO Network (2016). LTO.network: Blockchain for Decentralized Workflows. [Online]. Available at: https://ltonetwork.com/. Accessed on: Feb. 23rd, 2024.

[Lu et al. 2022] Lu, Y., Li, P., and Xu, H. (2022). A Food anti-counterfeiting traceability system based on Blockchain and Internet of Things. *Procedia Computer Science*, 199:629–636. The 8th International Conference on Information Technology and Quantitative Management (ITQM 2020 2021): Developing Global Digital Economy after COVID-19.

[Lunardi et al. 2020] Lunardi, R. C., Michelin, R. A., Neu, C. V., Nunes, H. C., Zorzo, A. F., and Kanhere, S. S. (2020). Impact of Consensus on Appendable-Block Blockchain for IoT. In *Proceedings of the 16th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, MobiQuitous '19, page 228–237, New York, NY, USA. Association for Computing Machinery.

[Machacek et al. 2021] Machacek, T., Biswal, M., and Misra, S. (2021). Proof of x: Experimental insights on blockchain consensus algorithms in energy markets. In *2021 IEEE*

Power Energy Society Innovative Smart Grid Technologies Conference (ISGT), pages 1–5.

[Michelin et al. 2018] Michelin, R. A., Dorri, A., Lunardi, R. C., Steger, M., Kanhere, S. S., Jurdak, R., and Zorzo, A. F. (2018). Speedychain: A framework for decoupling data from blockchain for smart cities. *CoRR*.

[Ministério da Sáude do Brasil 2024] Ministério da Sáude do Brasil (2024). Meu SUS Digital. [Online]. Available at: https://www.gov.br/saude/pt-br/composicao/seidigi/meususdigital. Accessed on: April 16th, 2024.

[Moulahi et al. 2023] Moulahi, W., Jdey, I., Moulahi, T., Alawida, M., and Alabdulatif, A. (2023). A blockchain-based federated learning mechanism for privacy preservation of healthcare IoT data. *Computers in Biology and Medicine*, 167:107630.

[Mulligan et al. 2018] Mulligan, C., Scott, J. Z., Warren, S., and Rangaswami, J. (2018). Blockchain Beyond the Hype: A Practical Framework for Business Leaders. In *White Paper, World Economy Forum*. [Online] Available at: https://www3.weforum.org/docs/48423_Whether_Blockchain_WP.pdf. Accessed on: April 16th, 2024.

[Nakamoto 2008] Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system. [Online]. Available at: https://bitcoin.org/bitcoin.pdf. Accessed on: July 31st, 2024.

[Nasir et al. 2018] Nasir, Q., Qasse, I. A., Talib, M. A., and Nassif, A. B. (2018). Performance Analysis of Hyperledger Fabric Platforms. *Security and Communication Networks*, 2018:3976093.

[Nguyen and Kim 2018] Nguyen, G.-T. and Kim, K. (2018). A Survey about Consensus Algorithms Used in Blockchain. *Journal of Information Processing Systems*, 14(1):101–128.

[Nunes et al. 2020] Nunes, H. C., Lunardi, R. C., Zorzo, A. F., Michelin, R. A., and Kanhere, S. S. (2020). Context-based smart contracts for appendable-block blockchains. In *2020 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*, pages 1–9.

[Oyinloye et al. 2021] Oyinloye, D. P., Teh, J. S., Jamil, N., and Alawida, M. (2021). Blockchain Consensus: An Overview of Alternative Protocols. *Symmetry*, 13(8).

[Panwar and Bhatnagar 2020] Panwar, A. and Bhatnagar, V. (2020). Distributed ledger technology (dlt): The beginning of a technological revolution for blockchain. In *2nd International Conference on Data, Engineering and Applications (IDEA)*, pages 1–5.

[Passos et al. 2024] Passos, R. B., Matteussi, K. J., Dos Anjos, J. C. S., and Geyer, C. F. R. (2024). Towards a Decentralized Blockchain-Based Resource Monitoring Solution For Distributed Environments. *Journal of Internet Services and Applications*, 15(1):1–13.

[Peck 2017] Peck, M. E. (2017). Blockchain world - Do you need a blockchain? This chart will tell you if the technology can solve your problem. *IEEE Spectrum*, 54(10):38–60.

[Popov 2018] Popov, S. Y. (2018). The Tangle - Version 1.4.3. [Online]. Available at: https://www.iota.org/foundation/research-papers. Accessed on Jan. 26th, 2024.

[Presidência da República do Brasil 2019] Presidência da República do Brasil (2019). Lei Geral de Proteção de Dados Pessoais (LGPD) - Lei Nº 13.853, de 8 de julho de 2019. [Online]. Available at: http://www4.planalto.gov.br/legislacao/. Accessed on: April 16th, 2024.

[Prisco et al. 2000] Prisco, R. D., Lampson, B., and Lynch, N. (2000). Revisiting the paxos algorithm. *Theoretical Computer Science*, 243(1):35–91.

[Puthal et al. 2019] Puthal, D., Mohanty, S. P., Nanda, P., Kougianos, E., and Das, G. (2019). Proof-of-Authentication for Scalable Blockchain in Resource-Constrained Distributed Systems. In *2019 IEEE International Conference on Consumer Electronics (ICCE)*, pages 1–5.

[Rajadevi et al. 2022] Rajadevi, R., Devi, E. R., Latha, R., Harshini, S., Ajay, K., and Abinash, M. (2022). Secured Storing and Sharing of Medical Records Based on Blockchain. In *2022 International Conference on Computer Communication and Informatics (ICCCI)*, pages 1–5.

[Rodrigues 2021a] Rodrigues, C. K. S. (2021a). Analyzing Blockchain integrated architectures for effective handling of IoT-ecosystem transactions. *Computer Networks*, 201:108610.

[Rodrigues 2021b] Rodrigues, C. K. S. (2021b). Blockchain-Based Platform for Managing Patients' Data in the Public Healthcare System of Brazil . *Revista de Sistemas e Computação (RSC)*, 11(3):63–72.

[Rodrigues 2022] Rodrigues, C. K. S. (2022). Comparative Analysis of Blockchain-Based Platforms for Managing Electronic Health Records in the Public Healthcare System of Brazil. *Revista de Informática Teórica e Aplicada*, 29(3):11–20.

[Rodrigues and da Silva 2019] Rodrigues, C. K. S. and da Silva, P. C. (2019). Uma Análise de Algoritmos de Consenso para Blockchain visando à Implementação de Sistemas de Informação Distribuídos Transparentes. *Revista de Sistemas e Computação*, 9(1):163–188.

[Sharma and Guleria 2023] Sharma, S. and Guleria, K. (2023). A comprehensive review on federated learning based models for healthcare applications. *Artificial Intelligence in Medicine*, 146:102691.

[Singh et al. 2022] Singh, A., Kumar, G., Saha, R., Conti, M., Alazab, M., and Thomas, R. (2022). A survey and taxonomy of consensus protocols for blockchains. *Journal of Systems Architecture*, 127:102503.

[Sivanathan et al. 2017] Sivanathan, A., Sherrat, D., Gharakheili, H. H., Radford, A., Wijenayake, C., Vishwanath, A., and Sivaraman, V. (2017). Characterizing and classifying IoT traffic in smart cities and campuses. In *2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 559–564.

[Tian et al. 2019] Tian, Z., Li, M., Qiu, M., Sun, Y., and Su, S. (2019). Block-def: A secure digital evidence framework using blockchain. *Information Sciences*, 491:151–165.

[Tobin and Reed 2016] Tobin, A. and Reed, D. (2016). The inevitable rise of self-sovereign identity. *The Sovrin Foundation*, 29(2016):18. [Online]. Avai-

lable at: https://sovrin.org/wp-content/uploads/2018/03/The-Inevitable-Rise-of-Self-Sovereign-Identity.pdf. Accessed on: July 31st, 2024.

[Turk and Klinc 2017] Turk, Z. and Klinc, R. (2017). Potentials of Blockchain Technology for Construction Management. *Procedia Engineering*, 196:638–645.

[Wang et al. 2023] Wang, Q., Yu, J., Chen, S., and Xiang, Y. (2023). SoK: DAG-based Blockchain Systems. *ACM Comput. Surv.*, 55(12).

[Wessling et al. 2018] Wessling, F., Ehmke, C., Hesenius, M., and Gruhn, V. (2018). How much blockchain do you need? towards a concept for building hybrid DApp architectures. In *Proceedings of the 1st International Workshop on Emerging Trends in Software Engineering for Blockchain*, WETSEB '18, page 44–47, New York, NY, USA. Association for Computing Machinery.

[Wu et al. 2022] Wu, H. Y., Yang, X., Yue, C., Paik, H.-Y., and Kanhere, S. S. (2022). Chain or DAG? Underlying data structures, architectures, topologies and consensus in distributed ledger technology: A review, taxonomy and research issues. *Journal of Systems Architecture*, 131:102720.

[Wüst and Gervais 2018] Wüst, K. and Gervais, A. (2018). Do you Need a Blockchain? In *2018 Crypto Valley Conference on Blockchain Technology (CVCBT)*, pages 45–54.

[Xu et al. 2017] Xu, X., Weber, I., Staples, M., Zhu, L., Bosch, J., Bass, L., Pautasso, C., and Rimba, P. (2017). A Taxonomy of Blockchain-Based Systems for Architecture Design. In *2017 IEEE International Conference on Software Architecture (ICSA)*, pages 243–252.

[Yik et al. 2021] Yik, M. H.-Y., Wong, V. C.-W. T., Wong, T.-H., and Shaw, P.-C. (2021). HerBChain, a blockchain-based informative platform for quality assurance and quality control of herbal products. *Journal of Traditional and Complementary Medicine*, 11(6):598–600.

[Zaabar et al. 2021] Zaabar, B., Cheikhrouhou, O., Jamil, F., Ammi, M., and Abid, M. (2021). HealthBlock: A secure blockchain-based healthcare data management system. *Computer Networks*, 200:108500.

[Zaman et al. 2022] Zaman, S., Khandaker, M. R. A., Khan, R. T., Tariq, F., and Wong, K.-K. (2022). Thinking Out of the Blocks: Holochain for Distributed Security in IoT Healthcare. *IEEE Access*, 10:37064–37081.

[Zheng and Zhou 2023] Zheng, F. and Zhou, X. (2023). Sustainable model of agricultural product logistics integration based on intelligent blockchain technology. *Sustainable Energy Technologies and Assessments*, 57:103258.

[Živic et al. 2019] Živic, N., Kadušić, E., and Kadušić, K. (2019). Directed Acyclic Graph as Tangle: an IoT Alternative to Blockchains. In *2019 27th Telecommunications Forum (TELFOR)*, pages 1–3.

[Živić et al. 2020] Živić, N., Kadušić, E., and Kadušić, K. (2020). Directed acyclic graph as hashgraph: an alternative dlt to blockchains and tangles. In *2020 19th International Symposium INFOTEH-JAHORINA (INFOTEH)*, pages 1–4.