BlockPS: A Blockchain Platform Selection Methodology

No Author Given

No Institute Given

Abstract. Blockchain technology has gained significant interest across various industries including finance, business applications, Smart Cities, and the Internet of Things (IoT). As organizations increasingly recognize blockchain's strategic importance, interest in its adoption has surged, viewing it as a critical growth element. However, challenges such as integration with existing systems, security concerns, regulatory uncertainties, and a lack of expertise hinder its adoption. A major barrier is the reluctance to replace or adapt current systems, leading to technical debt. With an increasing amount of blockchain platforms and cryptocurrencies, selecting the right platform is daunting and requires a comprehensive evaluation of functionality, adaptability, and compatibility. Existing methodologies help decide when to use blockchain but do not address platform selection. In response, we propose **BlockPS** – a Blockchain Platform Selection methodology designed to streamline the distributed application development process. BlockPS integrates seamlessly with existing software development processes, aiding in blockchain adoption while minimizing technical debt in just 3 easy to follow steps.

Keywords: Blockchain · Methodology · Blockchain Platforms.

1 Introduction

A decade after the emergence of blockchain technology, experts have hailed it as the technology of the future. Industries ranging from finance (including cryptocurrencies and FinTech) to business applications, Smart Cities, and the Internet of Things (IoT) have embraced blockchain. This broad interest underscores its potential for creating everyday applications [22].

As organizations recognize the strategic importance of blockchain, their interest in its adoption has surged. Shahid and Jungpil [31] indicate a shift in organizational attitudes, with blockchain now seen as a critical element for growth. However, adopting new technology is always fraught with barriers. Pawczuk and Massey [25] highlight challenges such as integrating blockchain with existing systems, security threats, and concerns over sensitive information. Additional hurdles include regulatory uncertainties and a lack of skills and understanding within organizations [15].

One major barrier is the reluctance to replace or adapt existing systems that already meet current needs [31]. Despite the resources and interest in blockchain, organizations often struggle with the transition, leading to technical debt that varies based on numerous factors. With over 1,000 blockchain platforms and more than 20,000 active cryptocurrencies, selecting the right platform for a particular system is daunting [27].

Choosing a blockchain platform requires evaluating its functionality, adaptability, and compatibility with existing software. Most software developers and architects lack expertise in this domain, often requiring them to seek external advice or self-educate, which increases costs and extends delivery times. Thus, the decision-making process for selecting a blockchain platform becomes increasingly complex as the number of decision-makers, alternatives, and criteria grow – a common scenario in this context. Therefore, a structured decision methodology is essential to guide and organize knowledge for selecting the right blockchain platform, reducing costs and development time.

Although existing methodologies help organizations decide *when* to use blockchain [19–21,33], none address the challenge of choosing a specific platform when the decision to use blockchain has already been made. These methodologies must also mitigate disadvantages and adapt to existing systems.

Given this context, we propose the development of a methodology applicable to pre-existing systems identified for blockchain integration. This methodology aims to facilitate the correct adoption of blockchain technology without incurring significant technical debt for software development teams.

In this paper, we present **BlockPS** – a Blockchain Platform Selection methodology designed to improve the distributed application development process. This methodology can be integrated with existing software development processes without adding complexity or time. It will aid in the adoption of blockchain technology by enhancing systems and organizations that have already identified this need. Additionally, BlockPS will contribute to a better understanding of blockchain technology among developers and decision-makers.

To achieve this goal, Section 2 presents the theoretical framework supporting this work, including the operation of blockchain and existing methodologies for blockchain adoption. Section 3 details the BlockPS methodology, explaining its 3 steps. Section 4 illustrates the application of BlockPS with an example. Section 5 evaluates BlockPS, demonstrating its effectiveness in building blockchain-based systems. Finally, Section 6 presents conclusions and future work.

2 Background

2.1 Blockchain

In 2008, Satoshi Nakamoto published an article titled “Bitcoin: A Peer-to-Peer Electronic Cash System” [29], describing a decentralized system for transferring electronic money online without relying on financial institutions or third parties. This was the first step in creating a decentralized technology operating on a consensus protocol, leading to the development of the first cryptocurrency, Bitcoin, and subsequently giving rise to the technology known as blockchain.

Blockchain is defined as a distributed database within a peer-to-peer network. This database consists of "blocks" that record transactions performed by the network nodes. Each block is cryptographically linked to the previous one via its hash, making it computationally difficult and practically impossible to corrupt the current block or alter the entire database. Any attempt to change the current block would result in a mismatch with the previous block's information [16]. These blocks are verified and confirmed without a central authority, adhering to a consensus protocol that ensures decision-making processes are secure, crash-tolerant, and resistant to collusion [23].

Information to be stored on the blockchain is formatted as a digitally signed transaction. This transaction is included in a block along with other transactions and is then sent to all nodes in the network. At this point, the nodes execute the chosen consensus protocol to validate the transaction. Nodes participating in this protocol are typically rewarded, often in the form of cryptocurrency. Once the block is validated, it is added to the blockchain and subsequently distributed to all nodes in the network, ensuring they update their copies of the blockchain accordingly [16].

Blockchain offers an application-agnostic solution where control is transferred from individuals to a digital system that uses cryptographic primitives to guarantee trust, security, and integrity. This significantly reduces the likelihood of attacks and interventions by malicious third parties. The operation of blockchain leverages elements from cryptography and distributed systems to deliver several highly desirable properties [24]:

Pseudo-anonymity. Users identify themselves with public keys (pseudonyms), rather than their real identities.

Transparency. The stored information is accessible to any network user at any time, as everyone holds a copy.

Trust. Blockchain enables asset transactions between unknown parties without mutual trust. By distributing a database among several network nodes and updating it through consensus, the validity of transactions in an untrusted environment is ensured.

Immutability. Due to the use of hash functions, information aggregated in a block becomes unmodifiable after a certain time.

Integrity. Information remains complete and accurate throughout its lifecycle, safeguarded against unauthorized modifications through hash functions and digital signatures.

Non-Repudiation. A sender cannot deny having sent information, ensured through the use of digital signatures.

Depending on how blockchain elements are used, we can categorize blockchains into four types. *Public Blockchains*, such as Bitcoin and Ethereum, are accessible to anyone with an Internet connection and are completely decentralized, allowing any individual to participate in transaction validation and verification. Transparency is a key feature, with all transactions visible to any participant, and security is maintained through robust consensus mechanisms like Proof of Work

(PoW) or Proof of Stake (PoS). In contrast, *Private Blockchains* are restricted to a specific group of participants and controlled by a central entity or consortium. Access and participation are limited, offering greater control over data and transactions, and are primarily used by organizations needing additional privacy, security, and efficiency in managing internal transactions. Examples include Hyperledger Fabric and Corda. *Permissioned Blockchains* combine elements of public and private blockchains. They can be public or private in terms of visibility but require permissions for participation, with participants needing authorization to validate transactions and add blocks. This type of blockchain is commonly used in enterprise environments where a balance between transparency and control is needed, providing security and efficiency while restricting access to authorized users, as seen with Ripple (XRP Ledger) and Quorum. Lastly, *Permissionless Blockchains* allow anyone to participate without authorization, being inherently open and decentralized to facilitate broad and democratic participation. Users can read, write, and audit transactions without restriction, which is essential for applications where transparency and inclusivity are a priority. Bitcoin and Ethereum are also examples of permissionless networks, where any user can join and contribute to the consensus process [35].

Another element that differentiates blockchain types is the consensus protocols they use. Consensus protocols coordinate nodes in distributed and decentralized environments, increasing fault tolerance, system robustness and efficiency. These protocols are generally categorized into two types. The first type is *proof-based consensus protocols*, where nodes joining the network must solve a cryptographic problem to gain the right to add a block. These protocols are predominantly used in public blockchains. The second type is *voting-based consensus protocols*, which are optimal for private blockchains with identified nodes. Unlike proof-based protocols, designed for environments with high node turnover, voting-based protocols require nodes to exchange results within the network before adding a block. In this system, a participant must receive approval from at least x participants, where x is a pre-defined threshold to add a block [30].

3 Related Work

Currently, several methodologies assist users in determining whether blockchain is suitable for their system development. Examples include works such as [17, 26, 32–34], where the authors employ a series of critical questions to help end users assess if blockchain is the appropriate technical solution for a particular application scenario.

However, once users have evaluated blockchain's suitability, there is limited guidance available to assist them in selecting the most appropriate platform.

Some works integrate blockchain into the development process, such as the framework proposed by Farooq *et al.* [19]. They introduce AgilePlus, a blockchain-based framework that executes smart contracts on a private Ethereum blockchain. Their experimental results show that AgilePlus enhances transparency, communication, coordination, traceability, and security, while addressing trust issues for both customers and developers in Distributed Agile Software Development (DASD).

Conversely, Farshidi *et al.* [20] argue that the blockchain platform selection process can be modeled as a multi-criteria decision-making (MCDM) problem. This involves evaluating various alternatives based on a defined set of decision criteria. They propose designing and implementing a decision support system (DSS) to aid decision-makers in addressing technology selection challenges in software production.

Most existing research, however, focuses on general aspects of software development, such as software architecture and functional requirements. In contrast, our methodology emphasizes non-functional software engineering requirements, including security, latency, technical debt, deployment processes, and the economic context of the blockchain networks to be used.

4 BlockPS: Blockchain Platform Selection Methodology

BlockPS or Blockchain Platform Selection Methodology guides institutions or companies in the process of changing from a centralized technology to a decentralized one based on blockchain. It aims to guide institutions and companies that have already decided to use blockchain in this adoption process in an informed manner. BlockPS consists of three steps: (1) Selection of Blockchain Type, (2) Blockchain Requirements Analysis and (3) Blockchain Platform Selection. Those are described in detail below.

4.1 Step 1. Selection of Blockchain Type

In this step, the decision model presented in Fig. 1 is used. This model involves answering a set of straightforward questions based on the requirements of the system to be built.

The first question aims to determine whether a feasibility analysis for blockchain implementation has already been conducted. If not, such an analysis should be performed using methods outlined in other studies [17,26,32–34]. The second question assesses whether participant registration or authentication is necessary before they can participate in the system. If registration is required, a *Permissioned Blockchain* is recommended. If not, a *Public (Permissionless) Blockchain*, which does not require any form of authentication, would be the ideal choice. The third question helps to determine the appropriate type of Permissioned Blockchain: whether it should be open to any participant (*Public*) or restricted to specific participants (*Private*). Ultimately, this process identifies the most suitable type of blockchain technology for a system's development.

4.2 Step 2. Blockchain Requirements Analysis

In this step, the most relevant characteristics of blockchain technology are organized and discussed to provide end users with a clear understanding of its

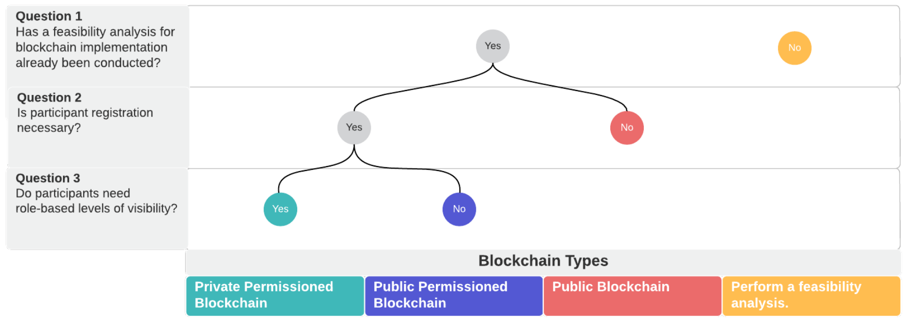

Fig. 1. Step 1. Selection of blockchain type

technical language, specifications, and particularities. To achieve this, we constructed a relational map that includes aspects of software engineering as well as the business needs of companies.

The relational map, shown in Fig. 2, places the blockchain consensus protocol at the center, highlighting it as the primary element from which blockchain properties arise. Two critical characteristics of the consensus protocol are transaction throughput (TPS) and transaction latency (S). These factors significantly impact the response time of systems built on blockchain platforms and must be analyzed first.

The map categorizes blockchain properties into three main areas (in blue):

1. Software Architecture. This category refers to the interconnected structural elements that provide the supporting framework for the entire development structure.
2. Technical Debt. This category focuses on business needs and costs associated with implementing blockchain technology, including factors such as popularity and the programming language used, which can affect development costs.
3. Security. This category addresses how the technology ensures the availability, confidentiality, and integrity of information and activities within a blockchain, as well as the scalability of the platform.

Based on this relational map, we also created a guide to help end users understand how these concepts are related and provide relevant references for further learning. This guide is available in both English and Spanish and can be found in here [1].

4.3 Step 3. Blockchain Platform Selection

At this stage, the end user already knows which type of blockchain to use and understands the important characteristics of blockchain in detail. This knowledge

[1] https://tinyurl.com/BlockPS-Step2

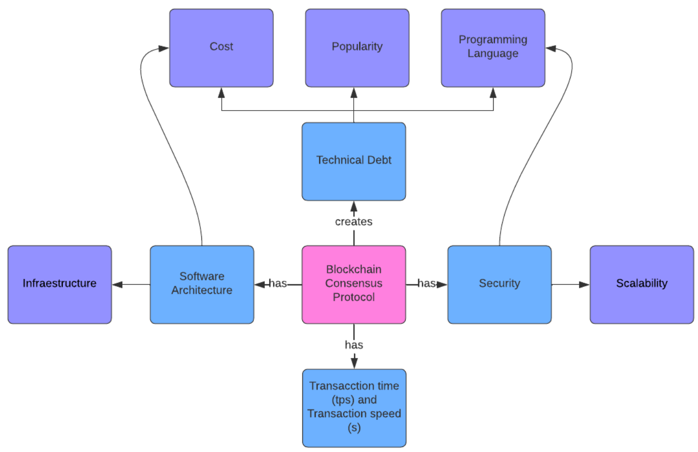

Fig. 2. Step 2. Blockchain Requirements Analysis

will help to make the next decision. A decision matrix for each blockchain type is then presented, highlighting the most relevant data for each platform. These decision matrices are based on the relational map outlined in the previous step.

If the output from Step 1 was a *Public Blockchain*, then the decision matrix shown in Fig. 3 will be used. This matrix presents nine blockchain platforms: Ethereum 2.0 [6], Algorand [1], Polkadot [10], Solana [13], Tezos [14], Ripple [12], Avalanche [2], Cardano [4], and EOSIO [5]. All these platforms support application development using smart contracts and were selected to provide a balance in popularity and token price. Each platform is evaluated based on a set of criteria discussed in Step 2.

The first criterion is the consensus protocol used by each platform, which influences transaction latency and throughput—both key performance measures. In terms of cost, the token price at the time of review and the SIMETRI digital asset classification [28] are considered, offering investment metrics based on in-depth fundamental analysis. For technical debt, factors such as the programming language used to develop smart contracts, the founding year, and the number of GitHub repositories are included. These elements provide insight into the platform's popularity, longevity, and stability in the market.

Regarding security, the matrix evaluates the security level, past security incidents, and whether bug bounties are conducted, as reported by Certified [18]. Additionally, information on audits performed, provided by Hacken [8], is included.

With this comprehensive evaluation, the end user can now select one or two platforms to develop the desired system, weighing the advantages and disadvantages of each.

	Ethereum 2.0	Algorand	Polkadot	Solana	Tezos	Ripple	Avalanche	Cardano	EOSIO
Consensus Protocol	PoS	PPoS	NPoS	DPoS / PoH	LPoS	XRP Ledger	Snowman BFT	Ouroboros	DPoS
Transaction latency (S)	12 S	3.6 S	6-12 S	0.4 s	10 s	3-5 s	1 s	20 s	0.5 s
Ttransaction throughput (TPS)	15 TPS	1200TPS	100,000TPS	710,000TPS	1000TPS	1500TPS	5000TPS	250 TPS	1M TPS
Technical Debt									
Token Price	$1,571.76	$0.3796	$5.64	$13.48	$1.01	$0.39	$17.31	$0.4057	$1.14
SIMETRI	B+	B-	B-	B	B-	D	B	C	B-
Programming Languages	Solidity, Python, JavaScript, Go, Rust	TEAL, PyTeal, JavaScript, Go	Rust	Rust	TypeScript	Python, TypeScript	Solidity, TypeScript, Go, Rust	Haskell	JavaScript
Founded	2015	2017	2020	2018	2014	2012	2020	2015	2017
Github repositories	278r	69r	122r	109r	13r	73r	66r	550r	119r
Security									
Security level	D	C	DD	BBB	BBB	DD	B	A	DDD
Audited	NO	NO	NO	YES	YES	NO	YES	YES	NO
Security Incidents	YES	NO	YES	YES	NO	NO	YES	NO	YES
Bug Bounty	NO	NO	YES	YES	YES	YES	YES	YES	YES

Fig. 3. Step 3. Blockchain Platform Selection - Public Platforms

If the output from Step 1 was a *Permissioned Blockchain*, then the decision matrix shown in Fig. 4 will be used. This matrix presents four blockchain platforms: two private (Hyperledger Fabric [9] and R3 CORDA [11]) and two public (BESU [3] and go-ethereum [7]).

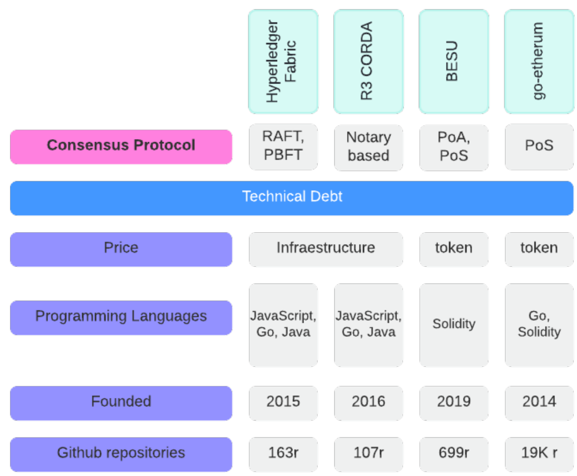

Fig. 4. Step 3. Blockchain Platform Selection - Permissioned Platforms

This matrix contains fewer elements than the previous one due to the nature of Permissioned Blockchains, where trust is based on the authentication of participants. As a consequence, the consensus protocols used are voting-based, eliminating the need for tokens. A distinction is made between *Private Permissioned Blockchains*, where costs arise from the infrastructure companies must set up to host private nodes for consensus, and *Public Permissioned Blockchains*, where tokens can be used to enable certain authenticated users to manage the consensus process. When tokens are involved, the elements discussed in Fig 3 should be considered. In this context, two types of blockchains are presented: BESU and go-ethereum, both of which use the open-source code of Ethereum. For assessing technical debt, factors such as the programming language used for developing smart contracts, the founding year, and the number of GitHub repositories are included. These factors provide insight into the platform's popularity, longevity, and stability.

5 BlockPS Application Example

In this section, we will demonstrate the application of the BlockPS methodology using an example. An educational institution has conducted a cost-benefit and feasibility analysis for using blockchain to manage student certificates. The analysis concluded that employing blockchain technology for issuing, storing, and verifying certificates would not only be faster and cheaper but also more reliable and verifiable.

The analysis yielded the following set of requirements:

- Only nodes that are part of the institution can participate in the verification and consensus of certificates.
- The information querying, such as accessing certificates, should be available to anyone with internet access.
- It is desirable to use JavaScript or Java, as the development team has prior experience with these programming languages.
- The speed of transactions is not a critical factor since the process will be inherently faster than the current one.
- The platform must be stable, ensuring consistent and reliable operation at all times. Platform stability is essential to maintain continuity of operations, minimize disruptions, and preserve the integrity of the data and services offered.

Using the requirements outlined above, we will apply the BlockPS methodology. In the first step, we follow the flow of questions and answers, which leads to the selection of a *permissioned public blockchain*. This process is illustrated in Fig. 5.

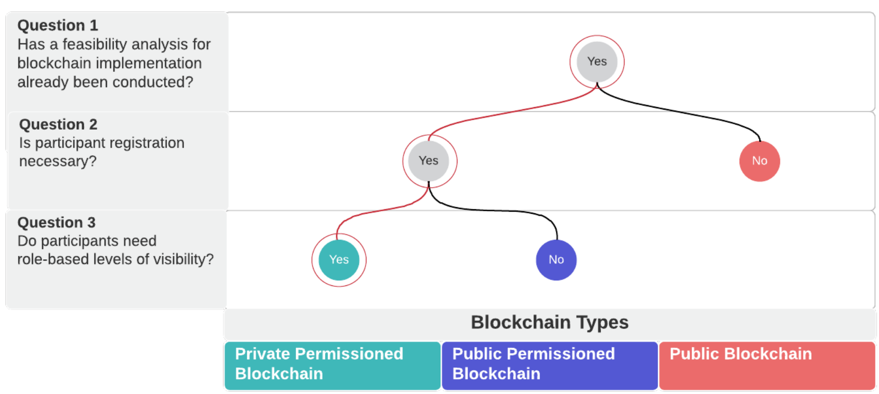

Fig. 5. Step 1. Example

The next step involves reviewing the document mentioned in Step 2. Finally, in Step 3, we select the values that are most desirable for our specific case. Based on these values, both Hyperledger Fabric and R3 Corda emerge as good options. However, the deciding factor is the number of repositories each platform has, which can indicate the extent of documentation and support available. Consequently, we decide to use Hyperledger Fabric. These can be seen in Fig. 6.

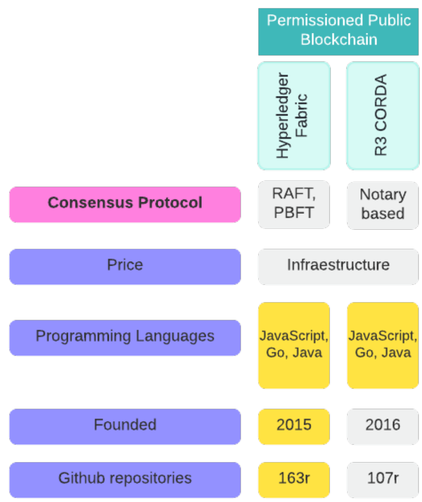

Fig. 6. Step 3. Example

6 Evaluation

To identify the right technology for implementing a system, organizations typically perform task such as benchmarking, prototyping and proof of concept (PoC), feasibility studies, consulting and expert opinions, literature reviews, and case studies. These processes can be time-consuming, often extending beyond what a company or institution is willing to invest. With BlockPS, however, it is possible to make an informed decision more efficiently. By following the three outlined steps, organizations can select a technology that aligns with their needs. Additionally, both the functional and non-functional requirements of the system are considered during the selection process, ensuring that the chosen technology best meets these criteria.

7 Conclusions

Although blockchain adoption presents significant challenges, such as adapting to existing systems and security concerns, its implementation can bring substantial benefits to organizations that manage to overcome these barriers, hence the BlockPS methodology is presented. It consists of 3 steps: (1) Choice of blockchain technology, where the user is supported with a flowchart to choose the type of blockchain that best suits their business needs, (2) Analysis of blockchain requirements, where it seeks to raise awareness of the software engineering aspects of equivalent correspondence with Blockchain development technologies, with the objective that the user understands key aspects that stand out from this technology and, finally, (3) Platform choice, where with the previous step and a decision matrix the user can choose one of the many existing blockchain platforms to start its development.

For the evaluation of the BlockPS methodology we used a use case that allowed us to validate and evaluate if this methodology is efficient and effective. Thus we were able to observe that the choice of a platform takes little time. It is worth mentioning that the evaluation has been carried out with 5 other use cases that are not presented here due to lack of space.

As future work, we propose the evaluation with real companies that can decide if the chosen platform meets their needs.

Acknowledgments. This study was funded by (TO BE ADDED)

References

1. Algorand. https://algorand.co/
2. Avalanche. https://www.avax.network/
3. Besu. https://www.hyperledger.org/projects/besu
4. Cardano. https://cardano.org/
5. Eosio. https://eos.io/
6. Ethereum 2.0. https://ethereum.org/en/staking/
7. go-etherum. https://geth.ethereum.org/
8. Hacken, audited projects. https://hacken.io/audits/
9. Hyperledger fabric. https://www.hyperledger.org/projects/fabric
10. Polkadot. https://polkadot.network/
11. R3 corda. https://r3.com/products/corda/
12. Ripple. https://ripple.com/
13. Solana. https://solana.com/
14. Tezos. https://tezos.com/
15. Alashaikh, L.: Blockchain-based software systems: Taxonomy development pp. 491–498 (2021). https://doi.org/10.1109/Blockchain53845.2021.00075
16. Aldeco-Pérez, R., Rajsbaum, S.: ¿Qué es Blockchain? Academia Mexicana de Computación **04** (Dec 2022)
17. Carlos Solís and Elizabeth Pérez Cortés and Humberto Cervantes: ¿usar o no blockchain para mi sistema? https://sg.com.mx/revista/57/usar-o-no-blockchain, accessed: 2023-11-21
18. Dyma Budorin CEO, H.G.: Cryptocurrency rankings by security. https://cer.live/cryptocurrency-security-rating, accessed: 2024-1-10
19. Farooq, M.S., Ahmed, M., Emran, M.: A survey on blockchain acquainted software requirements engineering: Model, opportunities, challenges, and future directions. IEEE Access **10**, 48193–48228 (2022). https://doi.org/10.1109/ACCESS.2022.3171408

20. Farshidi, S., Jansen, S., España, S., Verkleij, J.: Decision support for blockchain platform selection: Three industry case studies. IEEE Transactions on Engineering Management **67**(4), 1109–1128 (2020). https://doi.org/10.1109/TEM.2019.2956897
21. Hileman, G., Rauchs, M.: 2017 global blockchain benchmarking study. SSRN Electronic Journal (9 2017). https://doi.org/10.2139/SSRN.3040224, https://papers.ssrn.com/abstract=3040224
22. Huang, H., Kong, W., Zhou, S., Zheng, Z., Guo, S.: A survey of state-of-the-art on blockchains: Theories, modelings, and tools. ACM Comput. Surv. **54**(2) (mar 2021). https://doi.org/10.1145/3441692
23. Ismail, L., Materwala, H.: Article a review of blockchain architecture and consensus protocols: Use cases, challenges, and solutions. Symmetry **11**, 1198 (2019), https://api.semanticscholar.org/CorpusID:234776749
24. Kaur, S., Chaturvedi, S., Sharma, A., Kar, J.: A research survey on applications of consensus protocols in blockchain. Security and Communication Networks **2021**(1), 6693731 (2021). https://doi.org/10.1155/2021/6693731
25. Linda Pawczuk, Jonathan Holdowsky, R.M.: Delloite's 2020 global blockchain survey: A new age of digital assets (2020)
26. Lo, S.K., Xu, X., Chiam, Y.K., Lu, Q.: Evaluating suitability of applying blockchain. In: 22nd International Conference on Engineering of Complex Computer Systems (ICECCS). pp. 158–161 (2017). https://doi.org/10.1109/ICECCS.2017.26
27. McGovern, T.: How many blockchains are there in 2024? (Jun 2023), https://earthweb.com/how-many-blockchains-are-there/
28. Moos, M.: Simetri digital asset ratings. https://simetri.cryptobriefing.com/ratings/
29. Nakamoto, S.: Bitcoin: A Peer-to-Peer Electronic Cash System. Tech. rep. (2018), www.bitcoin.org
30. Pahlajani, S., Kshirsagar, A., Pachghare, V.: Survey on Private Blockchain Consensus Algorithms. In: 2019 1st International Conference on Innovations in Information and Communication Technology (ICIICT). pp. 1–6 (2019). https://doi.org/10.1109/ICIICT1.2019.8741353
31. Shahid, M.N., Jungpil, H.: A cross-disciplinary review of blockchain research trends and methodologies: Topic modeling approach. Proceedings of the Annual Hawaii International Conference on System Sciences **2020-January**, 4053–4059 (2020). https://doi.org/10.24251/HICSS.2020.495
32. Sultanik, E., Myers, M.: Do you really need a blockchain? (2022), https://www.trailofbits.com
33. Wust, K., Gervais, A.: Do you need a Blockchain? pp. 45–54. "Institute of Electrical and Electronics Engineers Inc." (11 2018). https://doi.org/10.1109/CVCBT.2018.00011
34. Xu, X., Weber, I., Staples, M.: Design Process for Applications on Blockchain, pp. 93–111. Springer International Publishing, Cham (2019). https://doi.org/10.1007/978-3-030-03035-3_6
35. Yaga, D., Mell, P., Roby, N., Scarfone, K.: Blockchain Technology Overview. Tech. Rep. NIST Interagency/Internal Report (NISTIR) 8202, National Institute of Standards and Technology, Gaithersburg, MD (Oct 2018). https://doi.org/10.6028/NIST.IR.8202