JOURNAL OF INFORMATION AND COMMUNICATION TECHNOLOGY
https://e-journal.uum.edu.my/index.php/jict

How to cite this article:
Abd Wahab, A., Ibrahim, H., & SarkinTudu, S. M. (2023). Committer assessment practice in blockchain project: A systematic literature review. *Journal of Information and Communication Technology, 22*(4), 675-706. https://doi.org/10.32890/jict2023.22.4.6

Committer Assessment Practice in Blockchain Project: A Systematic Literature Review

*[1]Alawiyah Abd Wahab, [2]Huda Haji Ibrahim &
[3]Shehu Malami SarkinTudu
[1]School of Computing, Universiti Utara Malaysia, Malaysia
[2&3]Institute for Advanced and Smart Digital Opportunity, School of Computing, Universiti Utara Malaysia, Malaysia
[3]Department of Computer Science,
Sokoto State University, Nigeria

*[1]alawiyah@uum.edu.my
[2]huda753@uum.edu.my
[3]Shehu.studu@ssu.edu.ng
*Corresponding author

Received: 12/2/2023 Revised: 17/10/2023 Accepted: 17/10/2023 Published: 25/10/2023

ABSTRACT

As Blockchain projects gain popularity among developers, the number of patched codes rapidly increases. With such growth, it is difficult for the few committers to maintain it in a timely manner. Subsequently, the community is always in search of new committers. This highlights the imperative importance of committer assessment decisions towards the success of Blockchain. However, the practices come with risks whereby new committers may harm the project. For example, a new

committer may initiate a hard fork that splits a project. Numerous systematic literature reviews have investigated developer turnover's impact on open-source software (OSS) projects. These studies mainly focused on aspects such as community participation, engagement, and motivation. However, previous reviews often overlooked committer assessment practices, particularly in the context of Blockchain projects. Although Blockchain operates as OSS, its distinct attributes, such as decentralisation and cryptography, justify the need for a dedicated review. Therefore, the objectives of this review are to 1) identify committer assessment practices, 2) identify problems in committer assessment, 3) identify existing factors in committer assessment, and 4) suggest some possible research topics. These goals were achieved through a systematic review of literature published between 2010 and 2022. The findings suggest that previous assessment models are useful but mainly focus on technical factors. The results also indicate that studies focusing on behavioural tendencies, which influence human activities, have so far been neglected. Finally, the paper concludes by charting potential open research opportunities.

Keywords: Social factors, Committer, Off-chain governance, Technical factors, Open-Source Software.

INTRODUCTION

Blockchain technology demonstrates the essential characteristics of digital information systems (IS) artefacts established for cryptocurrency (Hansun et al., 2022). Since an anonymous individual or group of developers presented Bitcoin in 2008, there has been a rising interest in Blockchain projects (Bosu et al., 2019), with the peer-to-peer networks of Bitcoin and Ethereum emerging as the two most well-known Blockchain initiatives (Biais et al., 2019). Additionally, more global economic sectors are becoming interested in Blockchain projects (Yli-Huumo et al., 2016), resulting in dramatic growth across the market capitalisation of cryptocurrencies (CoinMarketCap, 2019). For example, Bitcoin's market capitalisation increased from USD 611 million in 2014 to USD 1 trillion in 2021, with the price of 1 Bitcoin rising from USD 472.48 to USD 61 thousand in 2021 and was predicted to reach USD 70 thousand by the end of 2022 (CoinMarketCap, 2019). Other applications for the technology are now being explored in other cases (Carillo et al., 2017).

The initial Blockchain project established a global decentralised currency and payment system that may be acceptable globally without the need for a central authority (Hansun et al., 2022). This radical financial innovation, foundationally based on the concept of open-source software (OSS), triggered the emergence of Decentralized Autonomous Organization (DAO) as a business model (Mehar et al., 2019). DAO refers to a virtual entity comprising many individual actors, such as miners, users, and developers, that respond to a different set of rules or governances (Mehar et al., 2019). For example, in the case of transactions, miners secure the network by providing a confirmation and validation system based on their computing resources, in which they are governed by written rules embedded in the protocol software (Sarkintudu et al., 2019). The protocol software development process is usually governed and led by a community of developers (Anwar et al., 2019). Bitcoin, for instance, is a public Blockchain project whose source code is frequently available for modification, which links it to the OSS concepts discussed in the literature (O'Neil et al., 2021).

According to O'Neil et al. (2021), OSS is a software developed by a diverse group of individuals that can be used, modified, tested, and redistributed by contributors. Blockchain open-source communities rely on developer contributions (Chua & Zhang, 2020; De Filippi & Loveluck, 2016). As Blockchain projects continue to draw the interest of a large number of developers, there is growing concern about their viability and turnover (Hom et al., 2017). During the evolution of any OSS project, including Blockchain, developers typically change their role in the project (De Filippi & Loveluck, 2016). This phenomenon led to two categories of developers: (a) peripheral developers, whose role is to contribute code but cannot commit to the project directly, and (b) committers, who have permission to not only contribute but also commit code to the project directly (Amit & Feitelson, 2021). Therefore, committers conduct most of the maintenance tasks and constitute important key actors in the evolution of Blockchain software projects (Barraood et al., 2022).

However, as the popularity of Blockchain projects grows, the number of contributed patches and maintenance tasks, such as coding, bug-fixing, reporting, decoupling, testing, and documentation, increases at an extremely fast pace (Barraood et al., 2022). The rapid growth of contributed patches and the demand for maintenance make it difficult

to manage these projects in a timely manner (Barraood et al., 2022). Unlike traditional OSS, Blockchain developers are autonomous and lack institutional authority to regulate developer turnover behaviour, which puts a project's long-term sustainability at risk (Wang et al., 2020). Despite being an OSS project, managing the large influx of contributions and maintenance tasks of the Blockchain project code ultimately relies on a small number of committers (Coelho et al., 2020). This continues to create uncertainty about whether the understanding of this technology will spread enough to attract sufficient numbers of committers to evolve (Wang et al., 2020). However, thousands of developers are interested in becoming committers for a large-scale project like Blockchain (Wang et al., 2020).

It is difficult to identify a potential developer who has a high tendency to stay on board with the project (Wang et al., 2020). In general practice, a developer is assessed and promoted to a committer role through a comprehensive appraisal of the developer's activities, such as bug fixing and reporting as well as code testing (Wang et al., 2020). However, some developers often desert/leave a project soon after they are recommended for a committer role (Coelho et al., 2020). As a result, there is a blockchain expertise gap that is difficult to manage in a project (Coelho et al., 2017). To avoid the loss of Blockchain talents, project leaders (existing committers) must identify promising developers who may stay with the project indefinitely (Wang et al., 2020). Nevertheless, this process comes with prominent risks as the project leaders have no way of knowing whether the recommended developer will be actively involved in the project for a long time (Carillo et al., 2017).

Several systematic literature review studies have been conducted on the effects of developer turnover towards OSS projects, including those within the Blockchain domain (Hom et al., 2017; Steinmacher et al., 2014). Most of these reviews focused on developers community participation, engagement, and motivation (Wang et al., 2020; Kaur et al., 2020). Furthermore, earlier reviews did not take into account the committer assessment practices. Even if there are review studies on committer assessment practices, they are limited to traditional OSS projects (Carillo et al., 2017; Wang et al., 2020). Although Blockchain operates as an OSS project, it faces knowledge deficiency stemming from the departure of existing committers from the regular

occurrence of hard forks (Reiff, 2019). Hence, Blockchain projects constantly need to increase the number of committers to successfully evolve (Garagol & Nilsson, 2018). Moreover, the Blockchain domain warrants a dedicated review due to its unique technical attributes like decentralisation and cryptography.

Hence, it is necessary to examine and review the studies published in the developer community, specifically focusing on committer role assessment and the empirical evidence they have provided to date. Based on the authors' awareness, there has not been a systematic literature review conducted in this domain thus far. This lack of prior research inspired the authors to undertake a systematic literature review to gain insights into the current state of research concerning community dynamics and to discern the range of activities associated with community involvement and interaction. Therefore, the purpose of this research is to identify and collect the most recent IS and related discipline research on the evolution of the software developer role, as well as to identify critical knowledge gaps that have not been addressed in committer assessment practice within Blockchain and OSS projects.

The Systematic Literature Review (SLR) methodology was employed to address these concerns. The main contribution of this review is to aggregate the current assessment practices, challenges, and potential open research evidenced by different studies. It also identifies challenges and future research opportunities on committer assessment practices in Blockchain. Furthermore, the study can assist researchers who are interested in Blockchain off-chain governance (Anwar et al., 2019), particularly community management issues. The present study is based on the following research questions:

RQ1: What are the current committer assessment practices?
RQ2: What are the technical factors used in committer assessment practice?
RQ3: What are the problems/challenges in committer assessment?
RQ4: What are the potential future research opportunities?

The remaining parts of this paper are structured into five sections: a review of Blockchain governance challenges, Blockchain off-chain and committer assessment practice, methodology, findings, potential future research opportunities, and the conclusion.

BLOCKCHAIN GOVERNANCE CHALLENGES: ON-CHAIN AND OFF-CHAIN

OSS governance has been defined differently to suit the purposes of its respective authors (Anwar et al., 2019). De Filippi and Loveluck (2016) argue that OSS governance comprises the direction, control, and coordination of collaborating independent individuals. The OSS governance principles are widely adopted in Blockchain projects to coordinate changes committed to fit the system (Chen et al., 2021). Moreover, the public Blockchain has brought vital innovations, such as self-governability, which is based on a decentralised consensus protocol (Leewis et al., 2021). The protocol uses cryptography and economic incentives to persuade community members to protect the network (Mustafa et al., 2022). The technical capabilities of Blockchain are frequently claimed as superior in many respects to those of traditional OSS, especially in predicting its future growth (Lindman, 2021). However, technical supremacy alone is insufficient to turn technology into a viable business (Reijers et al., 2018). Thus, a sufficient presence of human factors is also essential in Blockchain (Mäenpää et al., 2018).

As a result, two alternative governance frameworks have emerged to coordinate the Blockchain community (Song & Kim, 2018). The first type of governance is known as on-chain governance (Reijers et al., 2018). It consists of a set of rules that are directly encoded via a protocol in a decentralised peer-to-peer network. In contrast, off-chain governance is a set of rules not encoded within the system, but that can be accomplished via community consensus (Reijers et al., 2018). Even if Blockchain projects do away with the requirement for potentially dishonest third parties (Hansun et al., 2022), developers, investors, and users are still faced with a governance crisis in the off-chain (De Filippi & Loveluck, 2016) where conflicting interests pose a risk to the Blockchain project as a whole (Bosu et al., 2019). Such a problem reveals the limitations of excessive dependence on technical means (on-chain) to handle social coordination and economic exchange difficulties in software projects (Reijers et al., 2018). Consequently, uncertainties in the off-chain structure can become vulnerable to the project's on-chain governance (Anwar et al., 2019).

For example, in the Bitcoin Foundation, project leaders, developers, and committers are all volunteers rather than funded or employed

(Anwar et al., 2019). As a result, the project lacks an ultimate decision-maker who can monitor their actions (Song & Kim, 2018). The ability of a committer to contribute to the project is frequently dependent on his or her own desire (Dong & Götz, 2021). The decision to upgrade the protocol, known as a "fork", is intrinsically decentralised and makes it difficult to achieve consensus (Reijers et al., 2018). Even though it can be initiated and proposed by any developer, agreement from the whole community is needed to succeed, leading to a governance crisis and a split in the Blockchain project and community (De Filippi & Loveluck, 2016). Following the split due to the hard fork, the community of existing committers (project leaders) will look for new committers (Cheng et al., 2017). The current committers will then ask the individual developers to demonstrate their skills (Garagol & Nilsson, 2018).

In addition, the current committers will also assess what they could contribute to the project management (Eichhorn, 2014). Without certain required skill levels, the developers may not be considered committers by the Blockchain community (Garagol & Nilsson, 2018). However, there was no requirement to become a committer in the early days of Blockchain (Chua & Zhang, 2020), and the condition was only established as the communities expanded and tasks were distributed (Wang et al., 2020). To become a committer, a developer has to complete many levels of tasks and exhibit required technical skills (Garagol & Nilsson, 2018). As part of the OSS community management, Reijers et al. (2018) emphasised the importance of finding a solid developer for the committer role. Nonetheless, the governance of a Blockchain environment will be extremely complicated because the central authority for selecting potential committers is undefined (Reijers et al., 2018).

Despite being an OSS project, the development and maintenance of the Blockchain code ultimately rely on a small number of committers (Bosu et al., 2019). The minimal number ensures that the Blockchain projects' code remains of excellent quality (Barraood et al., 2022). As the popularity of the Blockchain project grows, so does the number of patches contributed by a pool of developers (Bosu et al., 2019). The increasing quantity of provided patches makes it challenging for a small group of committers to keep track of them promptly (Amit & Feitelson, 2021). Furthermore, members' disputes frequently lead to a division in the Blockchain community (Garagol & Nilsson,

2018). Such a crisis exposes the limitations of off-chain governance in resolving social coordination and economic exchange (Reijers et al., 2018). For example, some committers can implement fresh ideas when decision-making is controversial by starting a new competing cryptocurrency (Sarkintudu et al., 2019). Bitcoin (BTC) variants such as Bitcoin Cash (BCH) and Bitcoin Gold (BTG), which were forked from the original Bitcoin, are currently available on the market (De Filippi & Loveluck, 2016).

Similarly, the Ethereum Classic (ETC) was created as a result of a fork in the Ethereum open-source project (Sarkintudu et al., 2019). This evidence exhibits a project split, which has resulted in the loss of committers and issues dealing with the enormous influx of developer contributions and requests (Al-Amri et al., 2019). Therefore, it is crucial for the number of committers to grow for a Blockchain project to succeed (Garagol & Nilsson, 2018).

Blockchain Off-chain Governance

Table 1 shows the OSS governance theory (Anwar et al., 2019). A study by Garagol and Nilsson (2018) argued that these governance mechanisms are expressed within Blockchain communities. The first mechanism is called modularity, which addresses the issue of coordination with large numbers of developers by dividing the project into several modules that may be worked on concurrently and in coordination. Role division is the second mechanism, where the individual developer is assigned a role as the project tasks grow larger. The most important decision within a project is code acceptance. The third mechanism is the delegation of decision-making. The majority of traditional OSS projects have a single project leader who is in charge of the project's overall strategy. In a Blockchain environment, the decision to accept a contribution is frequently made by vote. Training and indoctrination are the fourth mechanism that comes into play as communities grow and the division of roles is imposed. This mechanism sets the stage for the circumstances of role evolution, which are the focus of this study. The fifth mechanism is formalisation. As developers are geographically diverse, it is necessary to have communication tools that enable developers to, among other things, discuss, report bugs, and track Blockchain versions. This mechanism becomes critical when developers working on the same project are spread out across the globe and must communicate effectively.

Table 1

OSS Governance Mechanisms

Governance Mechanisms	Criteria	Traditional OSS	Blockchain	Sources
Modularity	The project is divided into modules.	Yes	Yes	Anwar et al. (2019)
Role division	Individual committers are divided into different roles.	Yes	Yes	Shah (2006)
Decision-making delegation	Decision is made on direction of the project.	Centralise	Decentralise	O'Neil et al. (2021), Chua and Zhang (2020)
Training & Indoctrination	Certain assessment requirements must be met to become a committer.	Yes	Yes	Anwar et al. (2019)
Formalisation	Formal communication tools are available.	Yes	Yes	O'Neil et al. (2021), Chua and Zhang (2020)
Autocracy/ democracy	Community chooses its leader.	Autocracy	Democracy	De Filippi and Loveluck (2016)
Exploratory outcome	Community-driven governance, with contributors supporting each other.	Yes	Yes	Garagol and Nilsson (2018)

Autocracy and democracy are the sixth mechanisms. In this regard, self-appointed leadership is best described as autocracy (e.g., Linus Torvalds, the founder of Linux), while a community-elected leader is known as democracy (Yang et al., 2022). For instance, some Blockchain projects can have project leaders who are elected by the community. Zhou and Mockus (2017) argue that project leaders should be elected by the community in a decentralised ecosystem. Finally, the seventh

mechanism is an exploratory outcome, which has been propounded to be suitable for Blockchain projects as it emphasises initiative-based progress within an informal, decentralised community. Unlike autocratic OSS models, all these mechanisms are democratic, with no single leader making decisions.

The technological novelty of Blockchain and the fundamental advances in design, development, and deployment have aroused the developer community's interest (Bosu et al., 2019). However, selecting a viable committer from a pool of open-source developers worldwide might be difficult (Garagol & Nilsson, 2018). In addition, committer assessment comes with uncontrollable risks, as a new committer may initiate an unnecessary hard fork that could split the project (De Filippi & Loveluck, 2016). Such a fork raises significant cyber-security concerns, creates incentive conflict, has a negative impact on the value of cryptocurrency, and renders previous investments obsolete (Alzahrani & Daim, 2019). As a result, off-chain governance decisions based on committer assessment are critical for the successful evolution of OSS, including Blockchain (Garagol & Nilsson, 2018).

Committer Assessment Practice Overview

Committer assessment is essential for the quality and long-term evolution of open-source projects (Mehar et al., 2019). Many empirical studies on the committer's assessment process have been published (Cheng et al., 2017; Garagol & Nilsson, 2018). For instance, Cheng et al. (2017) provided empirical evidence on the importance of community-centric roles where developers evolve their careers based on the number of contributions to a given number of projects. Their results revealed that developers could be promoted to the committer role based on subjective readiness and project environment that could significantly influence the developers' chance to evolve into a committer. The process of recruiting committers differs based on the type of project. The voting process for GNOME, for example, is formal and similar to that of Apache, whereas Mozilla and NetBeans take a more informal approach. The developer could petition the project leaders if they were unsatisfied with the assessment. The project leaders also have the right to accept or reject a developer. The assessment approaches for both projects were primarily concerned

with the developer's activities, such as the amount of code, which were collected from the project archives.

Aljemabi and Wang (2018) analysed the evolution of developer social networks (DSNs) on OSS projects collected from GitHub. The analysis was based on four perspectives: social network analysis, developer social network as an ecosystem, community evolution patterns, and the core-periphery structure. The findings revealed that DSNs include many peripheral developers and only a few core developers. Furthermore, around 10% of developers often changed roles from peripheral developers to committers over time. In identifying a potential future committer, the paper not only focused on the frequency of contributions but also the extent of communication among developers. This has been supported by prior findings advocating the notion that developers' expertise increases in software projects, suggesting that developers' productivity in terms of the number of activities completed per month can improve project success (Brasil-Silva & Siqueira, 2022; Li et al., 2021; Reboucas et al., 2017; Sharif et al., 2015).

Zhou and Mockus (2017) analysed the chances of becoming a long-term contributor and a new committer. The findings showed that developers mainly differed in their capacity and willingness to contribute (e.g., the number of reported issues and types of tasks from the period they started contributing). By constructing a committer promotion model, Garagol and Nilsson (2018) explored new committers' assessment of the Ethereum Blockchain using an established OSS governance mechanism. The study found that the Ethereum community will look for a new committer due to the hard fork split. Existing Ethereum committers employed GitHub to demand that a particular developer demonstrate their talents. Individual developers' contributions to the project will also be considered by the community. However, the developers will not be considered if they do not have specific knowledge and skills. Based on the review, all committer assessment models proposed by previous studies are helpful, although the majority focuses on aspects of developers' activities to identify good committers among thousands of developers using data from the project's archived repositories. Additionally, some studies used developer capacity and willingness to contribute to a project.

METHOD

Blockchain technology is undergoing rapid expansion and has led to an increase in research activity within the field. However, prior research in software engineering (SE) and computer science (CS) has predominantly focused on addressing technical challenges related to on-chain governance in Blockchain projects, specifically concerning aspects such as transaction validation via the technical protocol. Conversely, there has been limited exploration of off-chain governance aspects within Blockchain OSS communities, including governance topics like committer assessment practices. The socio-technical governance of Blockchain projects, in contrast to traditional OSS projects, does not have a single owner who can freely decide who to become a committer in the project (Lindman et al., 2017). Hence, it is crucial to categorise articles from the fields of IS, SE, and CS to gain diverse perspectives on the subject.

A literature review is an in-depth process that forms the foundation for all studies based on previous discoveries. SLR synthesises empirical data to address a given research question in a simple and repeatable method, seeking to incorporate all published information on the subject and evaluate the evidence's validity. Prior works must be used to increase knowledge. Researchers must first recognise the location of the knowledge boundary before they can push it. The extent of a corpus of existing research is then acknowledged, and significant literature is analysed to identify any gaps that need further investigation. This study followed the guidelines for conducting SLR provided by Kitchenham and Charters (2007). The review process started by developing a simple protocol that outlined the research questions, search strategy, selection criteria, quality assessment criteria, and data extraction strategy. The following sections summarise the review process conducted in this study.

Search Process

A systematic search for developer assessment and job promotion practices based on the IS point of view and the linked domain was undertaken to discover relevant publications to include in the review (Sarkintudu et al., 2019). It involved investigating leading databases in IS and its related areas, such as MIS Quarterly, ACM, IEEE, EJIS,

IEEE Computer, ACM Computing Surveys, Journal of the AIS, AIS Electronic Library, Scopus, and Web of Science.

The specific search terms used were "Bitcoin", "Blockchain project", "Open-source software", "Blockchain developer", "core developer", "peripheral developer", "committer", "developer contributions", "GitHub", "developer assessment", "developer productivity", "developer evolution", "developer participation", "developer role", and "developer promotion". Initially, individual terms were employed to search for articles. Subsequently, Boolean searches were conducted in the second round using search strings such as "developer assessment" OR "developer promotion" OR "developer evolution", which resulted in 419 primary sources. Another search was conducted by combining the above keywords with "systematic literature review", and it resulted in 12 review articles from OSS projects. Thus, the total number of articles was 431.

Selection Process

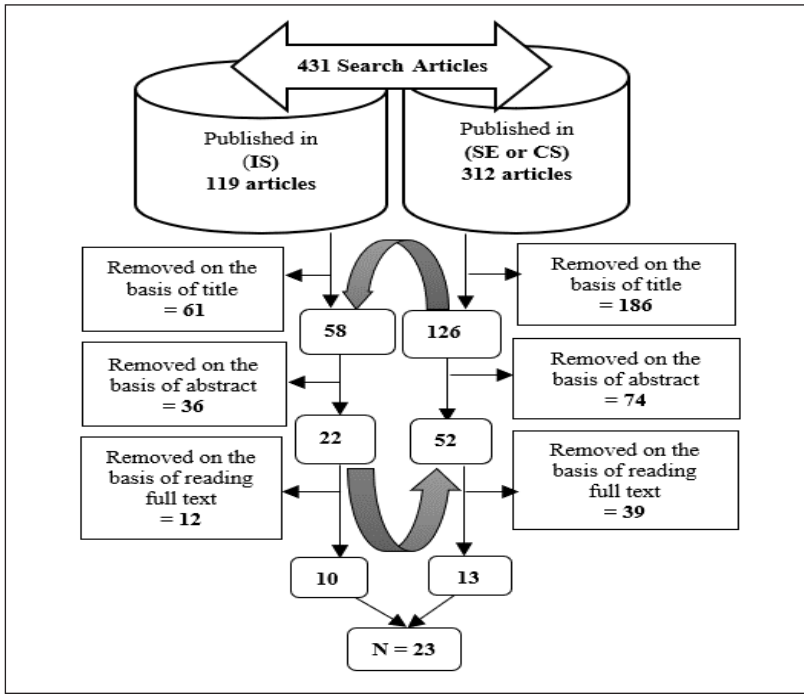
During this stage, the articles were selected based on five selection criteria: 1) IS senior scholar's journals from 2010 to 2022, such as MIS Quarterly, ACM, JMIS, EJIS, AIS Electronic Library, Scopus, and Web of Science (WoS); 2) Higher ranked IS chapters' conference proceedings, such as PACIS, ECIS, ICIS; 3) other related IS databases, such as IEEE, ACM; 4) Articles must be written in English; and 5) Titles or abstract included these phrases: "developer role assessment", "developer induction", and "developer role migration". As shown in Figure 1, only 15 articles met the first criterion, while 173, 75, and 168 articles met the second, third, and fourth criteria, respectively. The selection process began by removing duplicates, comparing the titles and abstracts of all search results, and controlling the number of citations of the individual articles. Systematic forward and backward citations, publication impact factor comparisons, and full-text scrutiny were conducted to ensure that only relevant articles were selected. The preliminary stage of the analysis included checking for redundant data.

The abstracts were then evaluated against the inclusion criteria. The methodology and discussion sections of each article were read and reviewed to determine its relevancy. Step 1 resulted in 431 articles

(AIS e-Library, 45; IEEE, 230; Taylor & Francis, 45; Springer, 79; and Elsevier, 32). In Step 2, 61 articles published in IS and 186 articles published in SE/CS were removed as they were considered irrelevant to this study. In Step 3, the abstracts of the 184 remaining articles were evaluated to identify any studies that should not be included, resulting in the removal of another 36 articles from IS and 74 articles from SE/CS. In Step 4, the introduction of each article was checked for quality and comprehensiveness against the inclusion criteria, resulting in further elimination of 51 articles. The remaining 23 articles (10 articles from IS and 13 articles from SE/CS) that fully met the inclusion criteria were used for further analysis (see Figure 1).

Figure 1

Search and Selection Process



Quality Assessment

Two approaches were used to determine the quality of the papers: a) based on reputable databases and b) using the quality criteria.

Table 2 shows the selected articles that appeared in higher-ranking IS journals and conference proceedings. Articles that appeared in higher-ranking IS journals and conference proceedings and were indexed in Scopus or Web of Science were considered to be of high quality. Despite the availability of different definitions and quality perspectives, quality researchers agree that quality assessment aims to determine the trustworthiness and relevance of each article against the research question or topic under investigation (Mengist et al., 2020; Mohammed, 2021). Hence, the present study used the definitions of quality assessment by Dybå and Dingsøyr (2008) and García-Peñalvo (2016) to evaluate the reliability, validity, and overall methodological rigour of individual research articles included in the review.

Table 2

Quality of Articles Based on Databases

Databases	Authors	No of Articles	Quality of Articles
AIS e-library	Bosu et al. (2019), Cheng et al. (2017)	2	Scopus & WoS
Tailor & Francis	CoinMarketCap (2019), Carillo et al. (2017), Anwar et al. (2019), Garagol and Nilsson (2018)	4	Scopus & WoS
IEEE	Chua and Zhang (2020), Hom et al. (2017), Reiff (2019), Mehar et al. (2019), Zhou and Mockus (2017), Fujita et al. (2014), Tsay et al. (2014)	10	Scopus & WoS
Springer	Amit and Feitelson (2021), Carillo et al. (2017), Anwar et al. (2019)	3	Scopus
Elsevier	Dong and Götz (2021), Bosu et al. (2019), Constantino et al. (2020), Sinha et al. (2015)	4	Scopus & WoS

In the second approach, we explored three major quality concerns suggested by Dybå and Dingsøyr (2008) and García-Peñalvo (2016) that must be considered when evaluating research findings: (1) Rigor: Did the study use essential research methods and conducted them thoroughly and appropriately? (2) Credibility: Are the presented results insightful? (3) Relevance: How helpful are the results to the scientific community and the software industry?

All three authors independently evaluated each of the 23 articles based on the above-mentioned quality criteria. Rigour was assessed through the clarity of objectives, methodology, and conclusions. If these elements were transparent and well-defined, a score of "1" was assigned, indicating a high level of rigour. If they were only partially apparent, a score of "0.5" was given, signifying moderate rigour. If any of these aspects were unclear, a score of "0" is provided, indicating low rigour. Meanwhile, relevance was indirectly assessed by aligning the articles with the review's goals. Articles that were closely aligned with the objectives received higher scores. Finally, credible quality was determined by evaluating how well the articles adhered to clear objectives, proper data collection methods, and transparent analysis procedures. Articles that demonstrated these qualities were considered more credible and received higher scores.

Table 3 shows the quality assessment findings, where 6 out of the 23 articles received excellent ratings while only two received low ratings.

Table 3

Number of Articles Based on their Quality Score

Quality (Score) of Selected Articles					
Quality status	Poor	Fair	Good	Very Good	Excellent
Quality score	< 15%	15%-45%	46%-65%	66% -85%	> 86%
Number of articles	2	4	7	4	6

Data Extraction

The reference management platform, Mendeley (http://www.mendeley.com/) was utilised to keep track of each article's references. The research disciplines, methodologies, and theories employed were among the data retrieved.

RESULTS AND DISCUSSION

Current Committer Assessment Practice

Based on the review, several technical and social factors were used for committer assessment. Articles like Zhou and Mockus (2017) and

Ihara et al. (2014) analysed developer technical activities to track and recommend new committers. It was discovered that the technical factors commonly used in the committer assessment practice were the number of patches created and submitted by developers. The majority of the studies (Garagol & Nilsson, 2018; Ihara et al., 2014; Long, 2006) used archive records to objectively measure developer technical activities. For example, Fujita et al. (2014) found that the committer assessment practice in the Apache OSS project is administered by the project management committee (PMC), which selects potential committers using a voting procedure that requires majority agreement. The productivity of Apache developers is measured in terms of the number of tasks completed each month in the project archive, such as the number of patches developed and defects reported by a single developer.

As a result, developers with a history of patch submissions in the project repository are frequently recommended to committers by the Python and PostgreSQL OSS projects. Other studies (Bosu et al., 2019; Cheng et al., 2017; Zhou & Mockus, 2017; Ihara et al., 2014) investigated the relationship between various technical and social factors and the possibility of accepted contribution to finding more possible future committers. The developer's efforts were assessed subjectively. For example, existing committers evaluate developers on GitHub and request that individuals exhibit their abilities (Garagol & Nilsson, 2018). They also looked at how a developer interacted with other project members over time (i.e., developer communication). As shown in Table 4, eight key performance indicators were identified in the literature based on the review of the committer assessment process. The review revealed that the committer assessment process differs based on the OSS communities. Some projects primarily focus on technical contributions, while others integrate both the technical and social activities of developers to assess new committers to the project (Zhou & Mockus, 2017).

Table 4

Technical and Social Factors Identified in the Literature

Factors	Descriptions	Measure	No of Articles
Patch creation	The total number of patches a developer has submitted	Objective	17

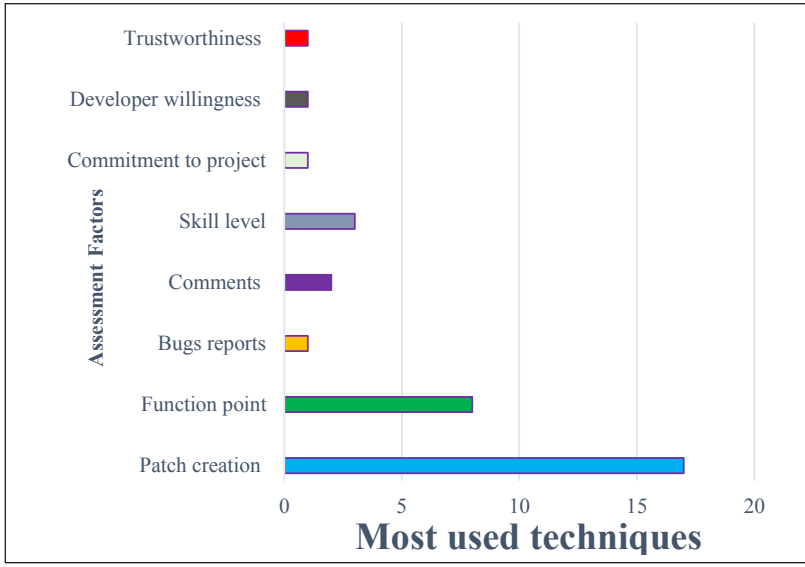
(continued)

Factors	Descriptions	Measure	No of Articles
Function point	The sum of function point	Objective	8
Bugs reports	The sum of bugs reported by the developer	Objective	1
Comments	Comments that a developer participate	Objective	2
Skill level	The skill level of a developer for a specific project	Objective	3
Commitment to project	The extent of sustained technical participation in the project	Subjective	1
Developer willingness	The ability of a developer to stay in a project for an extended period	Subjective	1
Trustworthiness	The extent of building individual trust in the community	Subjective	1

Based on the literature, the committer assessment practices commonly used by most OSS projects, including Blockchain, are the number of patches created by the developer and the function point that they added or modified in the project, as shown in Figure 2. Meanwhile, demonstrating skill level is the only appraisal technique to have been used by Blockchain projects, as reported by Garagol and Nilsson (2018) in the Ethereum project.

Figure 2

The Most Common Committer Assessment Practices



Only one article, authored by Garagol and Nilsson in 2018, focused on the Ethereum Blockchain project. In Blockchain, the existing committers (project leaders) promote developers based on their skill level demonstrated on the GitHub platform (Constantino et al., 2020). Potential developers will be asked to demonstrate their skills, which will be used as a basis for the existing committers to recommend the developer for the committer role (Garagol & Nilsson, 2018). It is interesting to note that some publications were found to demonstrate non-technical aspects, such as commitment to the project, developer personality, and trustworthiness, that influence the committer assessment practice in traditional OSS projects (Sinha et al., 2015). In Blockchain, governing decisions on who will become a new committer can be difficult (Garagol & Nilsson, 2018). Traditional OSS projects usually have a formal, centralised leader who decides and governs who joins the project as a committer (Aksoy-Yurdagul et al., 2021). In contrast, Blockchain eliminated the need for centralised decision-making (Al-Amri et al., 2019). The technology aims to create a self-sustaining, self-governing system that is exceptionally decentralised, democratic, and free from central control in managing transactions and project development coordination.

Technical Factors for Committer Assessment Practice

Based on extensive reading and re-reading of the extracted data, the findings show that the articles frequently employed technical factors, as shown in Table 5. Among the technical factors are patch creation, function points, bug reports, and technical comments shared in the forum. For the technical factors, 14 out of 23, or 60% of the studies, proposed using the technical factors in committer assessment practice. The review indicates that most of the articles used objective metrics to measure developer contribution efforts using data from the project archive available in the GitHub dataset (Gousios et al., 2018). For instance, the number of code patches submitted and accepted in the project over a given time frame, the total amount of lines of code in the OSS project that were submitted and accepted (Trinkenreich et al., 2022), the number of function point submitted by the developer (Chatzipetrou et al., 2018), and developer skill demonstration (Garagol & Nilsson, 2018).

Table 5

Technical Factors and Their Metrics Identified in the Literature

Technical Factors	Descriptions	Metric	Authors	OSS	Blockchain
Patch creation	The total number of patches a developer has submitted	Objective	Lee et al. (2018), Liu et al. (2019)	Eclipse & Mozilla	NIL
Function point	The sum of function point	Objective	Mahmood et al. (2022)	Gnome	NIL
Bugs reports	The sum of bugs reported by the developer	Objective	Marcilio et al. (2020)	Mozilla & Postgress	NIL
Comments	Comments that a developer participate	Objective	Wessel et al. (2022), Canfora et al. (2018)	Python & Apache	NIL
Skill level	The extent of the skill level of a developer to a specific project	Subjective	Garagol and Nilsson (2018)	NIL	Ethereum

The results of this SLR show that the majority of the current research on committer assessment practice focused on traditional OSS projects, while Blockchain projects received limited attention.

Problems/Challenges in Committer Assessment Practice

This study shows that research about committer assessment is primarily focused on traditional OSS projects, while Blockchain projects receive limited attention. It is argued that findings from traditional OSS studies are not directly applicable to the Blockchain context (Lindman, 2021). This is because many Blockchain technical characteristics differ from those of the traditional OSS project (Bosu et al., 2019). Although the authors found only one article that focused on the Blockchain project, studies on committer assessment practices lack rigour in reporting. Furthermore, the study by Garagol and Nilsson (2018) focused on the Ethereum Blockchain project and revealed that the committer assessment practice is based on the community's request for the developer to demonstrate their skills. However, the authors believe that future research will concentrate on other Blockchain projects as well as traditional OSS projects.

The current study also discovered several articles that investigated the potential for Blockchain projects other than cryptocurrencies, such as smart contracts and the Internet of Things (IoT), in the context of Blockchain projects. This kind of research will have a significant impact in the future and may be more intriguing than Blockchain projects using well-known cryptocurrencies. Finally, many articles did not indicate the basis for their research. IS scholars are expected to use existing theories and methods to better understand this exciting domain in the context of Blockchain. Although some researchers provide their research findings with technological contribution as one of the most prevalent factors, this might be an exciting way to combine technical and behavioural characteristics for committer assessment. This study provides significant evidence for industry players, such as cyber-security managers, cryptocurrency investors, and academic communities. IS researchers should focus their efforts on aspects of developer behaviour that are worth further investigation.

The gaps in committer assessment research in Blockchain projects resulted from several factors. Firstly, there is a bias towards traditional OSS projects due to their longer history and broader adoption

compared to the relatively new and specialised nature of Blockchain projects. Secondly, Blockchain's unique technical characteristics, such as decentralisation and cryptography, necessitate separate research. Additionally, the nascent state of the Blockchain industry, rapid technological changes, and limited reporting rigour contribute to the research gaps. Addressing these gaps is crucial as Blockchain technology diversifies into applications beyond cryptocurrencies, like smart contracts and the IoT. Establishing a comprehensive theoretical and methodological framework can benefit academia and industry as well as aiding cybersecurity managers and cryptocurrency investors.

Potential Future Research Opportunities

Overall, it can be stated that research about committer assessment practice in Blockchain projects has a strong alignment with the traditional OSS project development community because Blockchain projects exemplify OSS development models, as noted by Lindman (2021). However, they deviate from OSS projects due to their decentralised nature and the challenges associated with Blockchain project upgrades. These unique characteristics constitute the primary factors contributing to the committer assessment practice challenges faced by the Blockchain project community. Most of the studies are within the perspective of software engineering and computer science. For example, Bosu et al. (2019) highlighted the continuous increase of technical factors in committer assessment practice. Furthermore, the Blockchain community sought new committers by assessing developer technical skills and potential contributions in GitHub. Insufficient skills of individual developers were viewed as potentially harmful to the project rather than promotion to a committer role. Nevertheless, a complete understanding of committer assessment practices in Blockchain projects has not been reached yet. Furthermore, there is no coverage on the impact of unique characteristics of Blockchain in the committer assessment practice. The impact of these unique characteristics would provide clear insight towards understanding the phenomenon in the context of the Blockchain project. More investigation on committer assessment practice should also be done in the field of IS.

Although mentioned in one of the papers (i.e., Garagol & Nilsson (2018)), the focus of the reviewed papers on committer assessment still lies on analysing the technical code repositories only without

social influences (Cheng et al., 2017; Ihara et al., 2014). In Blockchain projects, the considerable autonomy granted to developers typically means that project leaders have limited formal authority to thoroughly assess developers' actions in extending functions and addressing bugs. Moreover, potential committers often depart the project once they achieve committer status. This could be a promising avenue for future research exploration. A summary of potential future research areas is shown in Table 6.

Table 6

Potential Future Research Area Opportunities Identified in the Literature

S/N	Research Area Opportunities	Description
1	A complete understanding of committer assessment practices in Blockchain projects has not been reached.	1. Develop a new criteria or model for the assessment of new committers. 2. Enhance the different Blockchain project committers' assessment practices. 3. Explore the barriers, challenges, and opportunities of committers' assessment practice in Blockchain projects. 4. Investigate the effectiveness of different committer assessment practices for the security, quality, and governance of Blockchain projects.
2	Most papers predominantly analysed individual developer contributions to code repositories, neglecting social influences in the committer assessment practice.	1. Develop a model of social and technical factors that can be used to solve complex problems of committer assessment practices.

(continued)

S/N	Research Area Opportunities	Description
		2. Investigate the effectiveness of integration of social and technical factors of committer assessment practices. 3. Assess the long-term effects of not considering social factors in committer assessment for the success of Blockchain projects.
3	Potential committers often depart from the project once they achieve a committer status.	1. Explore the reasons why potential committers often depart from the project once they achieve committer status. 2. Develop new criteria or supportive and rewarding models to reduce the number of committers who leave the project. 3. Investigate the implications of a high turnover rate of committers for the success of Blockchain projects.

CONCLUSION

This paper presented a systematic literature review that was conducted to learn more about committer assessment in OSS and Blockchain. A total of 23 studies published in the last decade (2010–2022) were identified as crucial to achieving the goal of this review. Analysis of the results revealed technical factors as the most famous and widely used key performance indicators for committer assessment practice, neglecting behavioural tendencies that influence developer activities in voluntary-driven OSS projects. Furthermore, numerous studies have been conducted in traditional OSS projects, while Blockchain has received little attention. The evidence also shows that behavioural tendencies, which affect a contributor's technical contributions, are

based on well-established traditional OSS. Therefore, researchers, particularly from the IS, should pay attention to public Blockchain projects from a community management perspective. Furthermore, existing approaches to examining specific Blockchain projects should be reinforced by IS researchers by bringing proven IS theories and methods that have effectively provided generalisable findings in related traditional OSS research.

Moreover, results from software ecosystem governance research may be helpful in committer assessment practice by examining developers' behavioural tendencies toward hard forks. This is because hard forks not only led to the loss of the committer but also triggered many catastrophic events that allowed some users to spend their digital assets twice. These events have severe consequences for investors and cyber-security managers. However, there is limited research on minimising the negative impact of hard forks on project evolution. IS and software engineering researchers can identify these crucial behavioural issues by understanding the committer's conduct who maintained the Blockchain project. Furthermore, a research agenda on Blockchain can offer further helpful directions for academics working in this area by outlining how to manage the Blockchain community.

ACKNOWLEDGMENT

This research was supported by the Ministry of Higher Education (MoHE) Malaysia through the Fundamental Research Grant Scheme (FRGS/1/2020/ICT03/UUM/02/4).

REFERENCES

Aksoy-Yurdagul, D., Rullani, F., & Lamastra, C. R. (2021). Designing shared spaces for firm-community collaborations for innovation: Formal policies and coordination in open source projects. *Creativity and Innovation Management*, *30*(1), 164–181. https://doi.org/https://doi.org/10.1111/caim.12426

Al-Amri, R., Zakaria, N. H., Habbal, A., & Hassan, S. (2019). Cryptocurrency adoption: current stage, opportunities, and open challenges. *International Journal of Advanced Computer Research*, *9*(44), 293–307. https://doi.org/10.19101/ijacr.pid43

Aljemabi, M. A., & Wang, Z. (2018). Empirical study on the evolution of developer social networks. *IEEE Access*, *6*, 51049–51060. https://doi.org/10.1109/ACCESS.2018.2868427

Alzahrani, S., & Daim, T. U. (2019). Analysis of the cryptocurrency adoption decision: Literature review. *PICMET 2019 - Portland International Conference on Management of Engineering and Technology: Technology Management in the World of Intelligent Systems, Proceedings*, 1–11. https://doi.org/10.23919/PICMET.2019.8893819

Amit, I., & Feitelson, D. G. (2021). Corrective commit probability: A measure of the effort invested in bug fixing. *Software Quality Journal*, *29*(4), 817–861. https://doi.org/10.1007/s11219-021-09564-z

Anwar, R., Rehman, M., Wang, K. S., Hashmani, M. A., & Shamim, A. (2019). Investigation of knowledge sharing behavior in global software development organizations using social cognitive theory. *IEEE Access*, *7*, 71286–71298. https://doi.org/10.1109/ACCESS.2019.2912657

Barraood, S. O., Mohd, H., & Baharom, F. (2022). An initial investigation of the effect of quality factors on agile test case quality through experts' review. *Cogent Engineering*, *9*(1), 1–27. https://doi.org/10.1080/23311916.2022.2082121

Biais, B., Bisière, C., Bouvard, M., & Casamatta, C. (2019). The Blockchain Folk Theorem. *Review of Financial Studies*, *32*(5), 1662–1715. https://doi.org/10.1093/rfs/hhy095

Bosu, A., Iqbal, A., Shahriyar, R., & Chakraborty, P. (2019). Understanding the motivations, challenges and needs of blockchain software developers: A survey. *Empirical Software Engineering*, *24*(4), 2636-2673. https://doi.org/10.1007/s10664-019-09708-7

Brasil-Silva, R., & Siqueira, F. L. (2022). Metrics to quantify software developer experience: A systematic mapping. In *Proceedings of the ACM Symposium on Applied Computing* (Vol. 1, Issue 1). Association for Computing Machinery. https://doi.org/10.1145/3477314.3507304

Canfora, G., Di Penta, M., Oliveto, R., & Panichella, S. (2018). Who is going to mentor newcomers in open source projects? In *Proceedings of the ACM SIGSOFT 20th International Symposium on the Foundations of Software Engineering, FSE 2012*, 1–11. https://doi.org/10.1145/2393596.2393647

Carillo, K., Huff, S., & Chawner, B. (2017). What makes a good contributor? Understanding contributor behavior within large Free/Open Source Software projects – A socialization perspective. *Journal of Strategic Information Systems*, *26*(4), 322–359. https://doi.org/10.1016/j.jsis.2017.03.001

Chakraborty, P., Shahriyar, R., Iqbal, A., & Bosu, A. (2018). Understanding the software development practices of blockchain projects. *Proceedings of the 12th ACM/ IEEE International Symposium on Empirical Software Engineering and Measurement 2018 Oct 11*, 1–10. https://doi.org/10.1145/3239235.3240298

Chatzipetrou, P., Šmite, D., & van Solingen, R. (2018). When and who leaves matters: emerging results from an empirical study of employee turnover. In *Proceedings of the 12th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement* (pp. 1-4). https://doi.org/10.1145/3239235.3267431

Chen, Y., Pereira, I., & Patel, P. C. (2021). Decentralized governance of digital platforms. *Journal of Management*, *47*(5), 1305–1337. https://doi.org/10.1177/0149206320916755

Cheng, C., Li, B., Li, Z. Y., Zhao, Y. Q., & Liao, F. L. (2017). Developer role evolution in open source software ecosystem: An explanatory study on GNOME. *Journal of Computer Science and Technology*, *32*(2), 396–414. https://doi.org/10.1007/s11390-017-1728-9

Chua, B. B., & Zhang, Y. (2020). Applying a systematic literature review and content analysis method to analyse open source developers' forking motivation interpretation, categories and consequences. *Australasian Journal of Information Systems*, *24*(16), 1–19. https://doi.org/10.3127/ajis.v24i0.1714

Coelho, J., & Valente, M. T. (2017). Why modern open source projects fail. In *Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering*, 186–196. https://doi.org/10.1145/3106237.3106246

Coelho, J., Valente, M. T., Milen, L., & Silva, L. L. (2020). Is this gitHub project maintained? Measuring the level of maintenance activity of open-source projects. *Information and Software Technology*, *122,* 106274. https://doi.org/10.1016/j.infsof.2020.106274

CoinMarketCap. (2019). *Historical data for Bitcoin*. https://coinmarketcap.com/currencies/bitcoin/historical-data/

Constantino, K., Zhou, S., Souza, M., Figueiredo, E., & Kästner, C. (2020). Understanding collaborative software development: An interview study. In *Proceedings of the 2020 ACM/IEEE 15th International Conference on Global Software Engineering,* 55–65. https://doi.org/10.1145/3372787.3390442

De Filippi, P., & Loveluck, B. (2016). The invisible politics of bitcoin: Governance crisis of a decentralised infrastructure. *Internet Policy Review*, *5*(3), 1–28. https://doi.org/10.14763/2016.3.427

Dong, J. Q., & Götz, S. J. (2021). Project leaders as boundary spanners in open source software development: A resource dependence perspective. *Information Systems Journal*, *31*(5), 672–694. https://doi.org/10.1111/isj.12313

Dybå, T., & Dingsøyr, T. (2008). Empirical studies of agile software development: A systematic review. *Information and Software Technology*, *50*(9–10), 833–859. https://doi.org/10.1016/j.infsof.2008.01.006

Eichhorn, B. R. (2014). The impact of user involvement on information system projects [Cleveland State University]. In *ProQuest Dissertations and Theses*. https://search.proquest.com/docview/1651232265?accountid=188395

Fujita, S., Ohira, M., Ihara, A., & Matsumoto, K. (2014). An analysis of committers toward improving the patch review process in OSS development. In *Supplementary Proceedings of the International Symposium on Software Reliability Engineering*, 369–374.

Garagol, D., & Nilsson, O. (2018). *Public blockchain communities a study on how governance mechanisms are expressed within blockchain communities* (Master's Dissertation). University of Gothenburg.

García-Peñalvo, F. J. (2016). Technological Ecosystems. IEEE *Revista Iberoamericana de Tecnologias Del Aprendizaje*, *11*(1), 31–32. https://doi.org/10.1109/RITA.2016.2518458

Gousios, G., Storey, M. A., & Bacchelli, A. (2018). Work practices and challenges in pull-based development: The contributor's perspective. In *Proceedings of the International Conference on Software Engineering*, *14-22-May-*, 285–296. https://doi.org/10.1145/2884781.2884826

Hansun, S., Wicaksana, A., & Khaliq, A. Q. M. (2022). Multivariate cryptocurrency prediction: Comparative analysis of three recurrent neural networks approaches. *Journal of Big Data*, *9*(1), 1–5. https://doi.org/10.1186/s40537-022-00601-7

Hom, P. W., Lee, T. W., Shaw, J. D., & Hausknecht, J. P. (2017). One hundred years of employee turnover theory and research. *Journal of Applied Psychology*, *102*(3), 530–545. https://doi.org/530-545, doi: 10.1037/apl0000103

Ihara, A., Kamei, Y., Ohira, M., Hassan, A. E., Ubayashi, N., & Matsumoto, K. I. (2014). Early identification of future committers in open source software projects. In *Proceedings of the International Conference on Quality Software*, 47–56. https://doi.org/10.1109/QSIC.2014.30

Kaur, R., Kaur Chahal, K., & Saini, M. (2020). Understanding community participation and engagement in open source software projects: A systematic mapping study. *Journal of King Saud University - Computer and Information Sciences*, *34*(7), 4607–4625. https://doi.org/10.1016/j.jksuci.2020.10.020

Kitchenham, B., & Charters, S. M. (2007). *Guidelines for performing systematic literature reviews in software engineering*.

Lee, A., Carver, J. C., & Bosu, A. (2018). Understanding the impressions, motivations, and barriers of one time code contributors to FLOSS projects: A survey. *IEEE/ACM 39th International Conference on Software Engineering (ICSE) 2017*, 187–197. https://doi.org/10.1109/ICSE.2017.25

Leewis, S., Smit, K., & van Meerten, J. (2021). An explorative dive into decision rights and governance of blockchain: A literature review and empirical study. *Pacific Asia Journal of the Association for Information Systems*, *13*(3), 25–56. https://doi.org/10.17705/1pais.13302

Li, Z., Yu, Y., Wang, T., Yin, G., Li, S., & Wang, H. (2021). Are you still working on this an empirical study on pull request abandonment. *IEEE Transactions on Software Engineering*, *48*(6), 1–17. https://doi.org/10.1109/TSE.2021.3053403

Lindman, J. (2021). What open source software research can teach us about public blockchain (s)?—Lessons for practitioners and future research. *Frontiers in Human Dynamics*, *3*(October), 1–7. https://doi.org/10.3389/fhumd.2021.642556

Lindman, J., Rossi, M., & Tuunainen, V. K. (2017). Opportunities and risks of blockchain technologies in payments - A research agenda. *Proceedings of the Annual Hawaii International Conference on System Sciences*, *2017*, 1533–1542.

Liu, K., Koyuncu, A., Kim, D., & Bissyande, T. F. (2019). AVATAR: Fixing semantic bugs with fix patterns of static analysis violations. In *Proceedings of the 2019 IEEE 26th International*

Conference on Software Analysis, Evolution, and Reengineering, 456–467. https://doi.org/10.1109/SANER.2019.8667970

Long, J. (2006). Understanding the role of core developers in open source development. *Journal of Information, Information Technology, and Organizations (Years 1-3)*, *1*, 075–085. https://doi.org/10.28945/148

Mäenpää, H., Mäkinen, S., Kilamo, T., Mikkonen, T., Männistö, T., & Ritala, P. (2018). Organizing for openness: Six models for developer involvement in hybrid OSS projects. *Journal of Internet Services and Applications*, *9*(1), 1–14. https://doi.org/10.1186/s13174-018-0088-1

Mahmood, Y., Kama, N., Azmi, A., Khan, A. S., & Ali, M. (2022). Software effort estimation accuracy prediction of machine learning techniques: A systematic performance evaluation. *Software - Practice and Experience*, *52*(1), 39–65. https://doi.org/10.1002/spe.3009

Marcilio, D., Furia, C. A., Bonifácio, R., & Pinto, G. (2020). SpongeBugs: Automatically generating fix suggestions in response to static code analysis warnings. *Journal of Systems and Software*, *168*(2020), 110671. https://doi.org/10.1016/j.jss.2020.110671

Mehar, M., Shier, C., Giambattista, A., Gong, E., Fletcher, G., Sanayhie, R., Kim, H., & Laskowski, M. (2019). Understanding a revolutionary and flawed grand experiment in blockchain: The DAO attack. *Journal of Cases on Information Technology*, *21*(1), 10–13. https://doi.org/https:\\10.4018/JCIT.2019010102

Mengist, W., Soromessa, T., & Legese, G. (2020). Method for conducting systematic literature review and meta-analysis for environmental science research. *MethodsX*, *7*, 100777. https://doi.org/10.1016/j.mex.2019.100777

Mustafa, A. S., Al-Mashhadani, M. A., Jasim, S. A., Shantaf, A. M., & Hamdi, M. M. (2022). Blockchain in fifth-generation network and beyond: A survey. *Bulletin of Electrical Engineering and Informatics*, *11*(3), 1399–1408. https://doi.org/10.11591/eei.v11i3.3209

O'Neil, M., Muselli, L., Raissi, M., & Zacchiroli, S. (2021). 'Open source has won and lost the war': Legitimising commercial–communal hybridisation in a FOSS project. *New Media and Society*, *23*(5), 1157–1180. https://doi.org/10.1177/1461444820907022

Porru, S., Pinna, A., Marchesi, M., & Tonelli, R. (2017). Blockchain-oriented software engineering: Challenges and new directions. *Proceedings - 2017 IEEE/ACM 39th International Conference on Software Engineering Companion, ICSE-C 2017*, 169–171. https://doi.org/10.1109/ICSE-C.2017.142

Reboucas, M., Santos, R. O., Pinto, G., & Castor, F. (2017). How does contributors' involvement influence the build status of an open-source software project? *IEEE International Working Conference on Mining Software Repositories*, 475–478. https://doi.org/10.1109/MSR.2017.32

Reiff, N. (2019). *A History of Bitcoin Hard Forks*. https://www.investopedia.com/tech/history-bitcoin-hard-forks/

Reijers, W., Wuisman, I., Mannan, M., De Filippi, P., Wray, C., Rae-looi, V., Vélez, A. C., Orgad, L. (2018). Now the code runs itself : On-chain and off-chain governance of blockchain technology. *Topoi*, *37*(17), 821–831. https://doi.org/10.1007/s11245-018-9626-5

Sarkintudu, S. M., Ibrahim, H. H., & Abd Wahab, A. (2019). Cryptocurrency platform ecosystem: A systematic literature review from information systems perspective. *International Journal of Advanced Computer Research*, *9*(44), 308–315. https://doi.org/10.19101/ijacr.pid97

Shah, S. K. (2006). Motivation, governance, and the viability of hybrid forms in open source software development. *Management Science*, *52*(7), 1000–1014. https://doi.org/10.1287/mnsc.1060.0553

Sharif, K. Y., English, M., Ali, N., Exton, C., Collins, J. J., & Buckley, J. (2015). An empirically-based characterization and quantification of information seeking through mailing lists during open source developers' software evolution. *Information and Software Technology*, *57*(1), 77–94. https://doi.org/10.1016/j.infsof.2014.09.003

Sinha, V. S., Mani, S., & Sinha, S. (2015). Entering the circle of trust: Developer initiation as committers in open-source projects. *In Proceedings of the 8th Working Conference on Mining Software Repositories*, 133–142. https://doi.org/10.1145/1985441.1985462

Song, J., & Kim, C. (2018). What is needed for the sustainable success of OSS projects: Efficiency analysis of commit production process via Git. *Sustainability*, *10*(9). https://doi.org/10.3390/su10093001

Steinmacher, I., Graciotto Silva, M. A., & Gerosa, M. A. (2014). Barriers faced by newcomers to open source projects: A systematic review. In *International Conference on Open Source Systems*. 153–163. https://doi.org/10.1007/978-3-642-55128-4_21

Trinkenreich, B., Wiese, I., Sarma, A., Gerosa, M., & Steinmacher, I. (2022). Women's participation in open source software: A survey of the literature. *ACM Transactions on Software Engineering and Methodology*, *31*(4), 81:2-81:37. https://doi.org/10.1145/3510460

Tsay, J., Dabbish, L., & Herbsleb, J. (2014). Influence of social and technical factors for evaluating contribution in GitHub. In *Proceedings of the 36th international conference on Software engineering*, 356–366. https://doi.org/10.1145/2568225.2568315

Wang, Z., Feng, Y., Wang, Y., Jones, J. A., & Redmiles, D. (2020). Unveiling elite developers' activities in open source projects. *ACM Transactions on Software Engineering and Methodology, 29*(3), 1–35. https://doi.org/10.1145/3387111

Wessel, M., Serebrenik, A., Wiese, I., Steinmacher, I., & Gerosa, M. A. (2022). Quality gatekeepers: Investigating the effects of code review bots on pull request activities. *Empirical Software Engineering, 27*(5), 108. https://doi.org/10.1007/s10664-022-10130-9

Yang, N., Ferreira, I., Serebrenik, A., & Adams, B. (2022). Why do projects join the Apache software foundation? *Proceedings - International Conference on Software Engineering*, *1*(1), 161–171. https://doi.org/10.1109/ICSE-SEIS55304.2022.9794012

Yli-Huumo, J., Ko, D., Choi, S., Park, S., & Smolander, K. (2016). Where is current research on Blockchain technology? - A systematic review. *PLoS ONE*, *11*(10), 1–28. https://doi.org/10.1371/journal.pone.0163477

Zhou, M., & Mockus, A. (2017). Who will stay in the FLOSS community? Modeling participant's initial behavior. *IEEE Transactions on Software Engineering*, *41*(1), 82–99. https://doi.org/10.1109/TSE.2014.2349496