IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 14, 2022, accepted April 26, 2022, date of publication April 29, 2022, date of current version May 9, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3171408

A Survey on Blockchain Acquainted Software Requirements Engineering: Model, Opportunities, Challenges, and Future Directions

MUHAMMAD SHOAIB FAROOQ[1], MISHAAL AHMED[1], AND MUHAMMAD EMRAN[2]
[1]Department of Computer Science, University of Management and Technology (UMT), Lahore 54770, Pakistan
[2]Department of Computer Science, King Abdul Aziz University, Jeddah 22254, Saudi Arabia

Corresponding author: Muhammad Shoaib Farooq (shoaib.farooq@umt.edu.pk)

ABSTRACT Requirements are the basis of software development practices. Ambiguities in requirements lead a project to a point of failure or penalize it with a high budget and time for defect traceability. The ever-growing demand for advanced computing systems has increased the complexity of Software Requirements Engineering (SRE) practices. Blockchain systems require specialized SRE practices as the issues of Requirement Traceability (RT), developer/client confidentiality, and Requirement Negotiation (RN) typically exist in conventional approaches, which require more improvement. Moreover, blockchain technology incorporates the capacity to function as an infrastructure for the SRE framework providing transparency, security, and reliability. Even though the significance of studying blockchain in the context of SRE is evident, it is still in its infancy. None of the previous studies surveyed this domain to the best of our knowledge. We aim to summarize the scholarly contributions of blockchain acquainted SRE from 2015 to 2021 and to provide academia and practitioners with in-depth knowledge about this domain. In this article, we have provided a novel comprehensive review of the aspects of blockchain-acquainted SRE practices. We have presented SRE-based quality improvement factors and outlined the need for blockchain technology in this domain. Furthermore, we have classified SRE practices based on blockchain engineering. In addition, we have proposed a generic SRE model built on blockchain infrastructure along with its workflows. Similarly, we have provided implementation guidelines for the future development guidance of SRE applications built on blockchain technology. Finally, we have presented the current research challenges and provided future directions based on blockchain acquainted SRE.

INDEX TERMS Software requirements engineering (SRE), software engineering (SE), requirements negotiation (RN), requirements traceability (RT), requirements validation (RV) software development life cycle (SDLC), software requirements specification (SRS), decentralized applications (DApps).

I. INTRODUCTION

The foundation of software systems relies on the accuracy of captured requirements. The requirements are captured during the initial phase of the Software Development Life Cycle (SDLC) [1]. Each requirement undergoes through rigorous Software Requirements Engineering (SRE) process [2]. The SRE process occurs before the actual development in which, the requirements are defined, examined, and developed, to propose software solutions in a documented form. Consequently, it helps in understanding and interpreting the beliefs, goals, and needs of the project stakeholders to guarantee that the problem being stated is clear, complete, and concise while ensuring the solution is reasonable, effective, and correct [3]. In the course of SRE practices, the requirements team negotiates with the stakeholders and designs the Software Requirements Specification (SRS) documentation (consisting of a finalized set of requirements) for the development team, through which the actual software application is engineered [4].

Generally, software defects occur as a result of deficient requirements [5]. Deficient requirements are ambiguous, incomplete, faulty, and imprecise. These requirements create unsuccessful or ineffective projects, making it crucial for practitioners to detect defective requirements during the early stages of SDLC. As the software project proceeds

The associate editor coordinating the review of this manuscript and approving it for publication was Yang Liu.

towards maturity, tracing defects becomes increasingly difficult, imposing a lot of budget and time penalties to accommodate changes or fix defects. Sometimes the expense of the change is unbearable so reconstructing the project from scratch becomes more feasible than resolving the defects [6]. Hence, SRE practitioners carefully elicit, examine, and specify the requirements by following the most precise approach.

Current digital transformation as a result of rapid innovation in computing technologies has increased the demand for new and reliable SRE practices [7]. It has become more crucial for software companies to cope with technological advancement, and to retain competitiveness by employing state-of-the-art SRE practices. Emerging technologies are complex and require definitive SRE practices. Since the conventional SRE practices create hindrance for practitioners to efficiently produce reliable requirements for advanced systems [8].

Conventional SRE practices are outdated and demand improvement [9]. To address the quality issues, contemporary software companies employ a mix of SRE practices such as agile-based or specialized methods which assists them in effectively delivering the projects [10]. Multiple tools and techniques are utilized to capture reliable requirements based on these methods [11]. However, the issues such as the client-developer gap, Requirements Negotiation (RN), and Requirements Traceability (RT) factors are generally found in contemporary SRE practices. In addition, some other concerns such as confidentiality, requirement inconsistency, ineffective SRE tools, stakeholder conflicts, and collaboration also exist [12].

Reliable requirements are corrective, negotiated, interpreted, and produce effective software solutions [13]. A successful software solution is accepted by all stakeholders, has reliable requirements, completed within predefined costs and promptly [14]. Modern clients are anticipative of software companies having transparent, trustable, and secure SDLC processes, producing successful quality-based software solutions. [15]. Most SRE practices are designed to address specific problems, making them difficult to be applicable in every case [16]. Hence, software companies employ the most suitable SRE practices per their standards and domain preferences.

Modern technologies are developed through specially designed SRE practices. Recently, blockchain technology has emerged as a promising solution in every computing domain [17]. Blockchain is fundamentally a digital ledger of records that are distributed and replicated over the complete web of computer systems present on a blockchain. Each block in the chain encapsulate records and whenever new data transpires on the blockchain, the record of that data is appended to each user's ledger. The decentralized database administered by the users is known as DLT (Distributed Ledger Technology). The features of blockchain technology offer transparency, security, a consensus mechanism, immutability, and a decentralized system, addressing the problems currently found in centralized systems [18]. Current SRE approaches utilized by software companies are based on confidentiality, manual work, centralization, and slow processes to develop software systems. Nevertheless, blockchain technology shows a promising potential to function as an infrastructure for SRE practices, and support practitioners in gathering, validating, and specifying the requirements in a decentralized, secure, trustable, and transparent manner [19]. In addition, SRE built on blockchain infrastructure can be very cost-effective and can automate SRE practices and enhance reliability.

The significance of the studies based on blockchain acquainted SRE is evident, however, more studies are required in this domain. We have used the term "acquainted" which means SRE is accustomed to or familiar with blockchain technology. This term incorporates SRE's familiarity with blockchain from each dimension whether it is to build blockchain systems or use blockchain as an infrastructure for SRE to build software systems. Most of the previous studies presented state-of-art SRE practices for the development of blockchain systems whereas some presented the concept of SRE built on blockchain infrastructure. Nevertheless, none of the previous studies provided a generic SRE model and a comprehensive survey in this domain. The primary aim of this research paper is to highlight the domain of SRE embraced by blockchain technology and cover the study gap in this domain. This research paper aims to summarize the scholarly contributions from 2015 to 2021 in the domain of blockchain acquainted SRE. Moreover, this research can work as a guideline for early career researchers or industry experts to find gaps and solutions for developing blockchain frameworks based on SRE using the guidelines presented in this article. In addition, this research covers all the dimensions of blockchain-related SRE so it provides a comprehensive background on up-to-date trends in blockchain technology based on Software Requirement Engineering practices. To this end, we have outlined the quality factors of improvement found in conventional SRE practices. Consequently, we have explained the significance of blockchain technology in addressing these quality factors. Moreover, we have provided the classification of blockchain acquainted SRE practices through a detailed state-of-art review. Furthermore, we have proposed a state-of-art SRE model built on blockchain infrastructure. In addition, we have provided implementation guidelines, research challenges, and future directions. The novelty of this work is that while reviewing the previous practices we have not only focused on blockchain SRE practices but we have also strongly involved each aspect to provide a brief comprehension of this domain. The discussion of the research reveals that previously practices are inflexible imposing domain, application, and automation constraints. In addition, we have further outlined that the SRE framework built on blockchain infrastructure can effectively produce reliable requirements, however, more applications and proofs-of-concept are required in this domain for further comprehension.

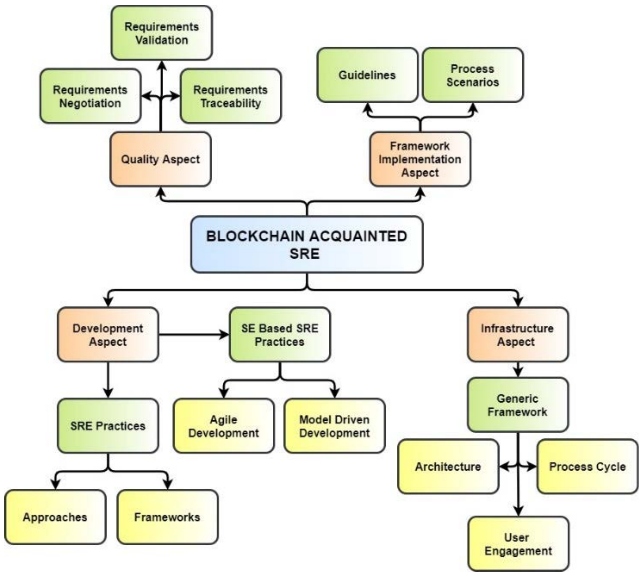

FIGURE 1. Taxonomy of Blockchain Acquainted SRE.

To carry out this research, we searched for previous conference and journal publications in top-rated scholarly databases such as Google Scholar, IEEE Xplore, Scopus, Science Direct, and JSTOR. The focus of the search in the scholarly database remained mainly related to two keywords namely Software Requirement Engineering and Blockchain Technology. In addition, keywords like Software Engineering, Software Development Life Cycle, Requirements Elicitation, Requirements Validation, Requirements Management, Blockchain Development, Agile Development, and Software Systems were also employed along with Blockchain. After downloading the relevant publications from the selected scholarly databases, a successive screening procedure was then employed to filter the downloaded publications. In the first phase of the screening process, topic refinement was carried out while visiting the websites of the scholarly databases to eliminate redundancy in the publications. Similarly, in the second phase of the screening process, the screened publications were further examined for their abstracts and titles from the context of blockchain and Software Requirements Engineering. Finally, in the last phase of the screening process, the screened articles found to be most relevant were captured from the scholarly databases and reviewed after checking against inclusion and exclusion criteria

The taxonomy of blockchain acquainted SRE is shown in Fig.1. There are four major aspects of this domain namely quality, development, infrastructure, and implementation.

As the topic of this research is entirely new, the search for publications was carried out irrelevant to time constraints. Hence, the search for publications remained in the process until December 2021. We primarily focused on conference articles and academic journals to look for publications and make sure that the coverage and quality of the scientific knowledge of Software Requirements Engineering and blockchain technology remain sublime. At the same time, the publications neglecting the aspects of SRE based on blockchain technology were discarded. In addition, the screening process was executed by all of the authors of this study using a back-to-back approach. After capturing the publications, the results were contrasted, and the authors of this study discussed the results with differences until a consensus was reached. Only a few articles were captured precisely to be previously published in the context of blockchain technology and requirement engineering. Though there were many papers captured from the scholarly databases, however, most of them were rejected due to filtration which significantly improved the accuracy of the literature. Based on the review mechanism, we also synthesized the collected articles by observing the application domain, type of publication, and year of publication. In addition, the application level of the captured publication has also been observed.

A. CONTRIBUTION OF THE SURVEY AND COMPARISON WITH RELATED SURVEYS

In recent research, only a few surveys proposed solutions to integrate blockchain in SE (Software Engineering) and SRE practices. For instance, [20] characterize BBS (Blockchain-Based Systems) engineering using a state-of-art survey. The study presents a generalized survey of the BBSs from an SE perspective, whereas we provide a comprehensive survey on the aspects of blockchain acquainted SRE. Moreover, the study classifies SE practices while this survey categorizes SRE practices. Furthermore, the study did not cover problematic or improvement factors, while we provide a detailed comprehension of this aspect. Similarly, [21] conducted a Systematic Literature Review (SLR) to identify and address SRE challenges during COVID-19 through the application of blockchain technology. The SLR outlined SRE problematic factors, however, it did not classify blockchain acquainted SRE practices, while we have covered both aspects. In addition, the SLR only examined the strengths of integrating blockchain with the SRE while we propose model, workflow, and implementation guidelines based on a proof-of-concept. Hence, this study contributes a novel state-of-art survey based on blockchain acquainted SRE practices. The survey incorporates the conception of different aspects of the blockchain-acquainted SRE ecosystem. The survey also presents a detailed review of recent research efforts carried out in different domains of blockchain acquainted SRE, proving to have a substantial impact. The comparison of this survey with other related surveys is shown in Table 1.

Previous studies did not cover a comprehensive survey in this domain. Hence, this survey contributes a comprehensive discussion on the recent advances in the context of blockchain acquainted SRE, fulfilling the comprehension gap currently found in this domain. The focus of the survey is based on SRE, blockchain technology, and decentralizing the SRE practices using blockchain technology. In addition, it covers the roles of the entities involved in the SRE infrastructure when integrated with blockchain, the tradeoffs in selecting

TABLE 1. Comparison with related surveys.

Blockchain Acquainted SRE Survey Contributions	[20]	[21]	This Survey
Discussed SRE Improvement Factors	×	✓	✓
Presented the Need for Blockchain-Based SE or SRE	✓	✓	✓
Classified Blockchain Acquainted SRE Practices	×	×	✓
Presented Blockchain SRE Practices Based on SLDC Models	×	×	✓
Proposed Blockchain Acquainted SRE Model	×	×	✓
Presented Implementation Guidelines	×	×	✓
Challenges and Future Directions	✓	✓	✓

appropriate blockchain consensus algorithms for different application scenarios, recent research efforts made towards resolving the key issues found under this domain, architectural and implementation guideline, and open research directions for future work.

The summary of the contributions based on this survey is as follows:

- Highlights the quality improvement factors of SRE practices and the need for a generic model in this domain based on blockchain infrastructure.
- Performs feature analysis to signify the applicability of blockchain technology in SRE practices.
- Classifies the quality improvement factors found in SRE practices and existing blockchain acquainted SRE approaches and frameworks.
- Presents SRE architecture and workflows built on blockchain infrastructure along with implementation guidelines.
- Outlines research challenges and future directions based on blockchain acquainted SRE.

B. STRUCTURE OF THE SURVEY

The structure of this survey is shown in Fig.2. There are eight sections to this survey. Observing the structure of the survey, Section I is based on the introduction in which introductory literature and survey structure has been discussed while Section II of the research paper provides literature on the significance of blockchain SRE and pertinent taxonomies. Moreover, Section III of this research presents the feature analysis of blockchain technology and the SRE framework. Similarly, Section IV presents blockchain-based SRE practices. Furthermore, Section V presents a generic SRE model driven by blockchain technology.

In addition, Section VI presents guidelines that are required to implement a blockchain-driven SRE model. We have presented tools and technologies to implement a blockchain-driven SRE system. Furthermore, Section VII outlines the research challenges and future directions in this domain. Finally, Section VIII presents the conclusions of this research study along with the limitations.

II. BLOCKCHAIN ACQUAINTED SRE MOTIVATION AND QUALITY IMPROVEMENT FACTORS

Over the years, a tremendous amount of developments in SRE have been carried out to cope with the ever-growing demands of modern computing technologies. Numerous scholars have proposed plentiful SRE practices to produce reliable requirements by addressing the quality issues to efficiently address the stated problem in form of software documentation (known as a blueprint) processed through distinct frameworks and approaches. The blueprint is also known as a software requirement specification (SRS) which contains the final set of requirements for the system under development. The SRS consists of two essential requirement categories namely functional and non-functional [22]. Functional requirements are based on features while non-functional requirements are based on outside scope. The advancement in technologies demands new robust and rigorous SRE practices to specify functional and non-functional requirements. The previous SRE studies focused on improving requirements gathering, requirement analysis, and specification, modeling of new tools, and highlighting the need for newer approaches or frameworks.

For instance, [23] outlined that RN in cloud computing demands more efficiency for improved applications. Moreover, [24] offered a mobile application for RN practices but highlighted the need for automatic RT. Furthermore, [25] presented an SRE model for big data applications to improve requirement correctness in safety-critical systems however a more robust method is required. Similarly, [26] presented an interest-based learning negotiation system to increase the interest of stakeholders and learners but highlighted the need for a more effective methodology. Likewise, [27] presented a mobile speech translator model by combining machine translation and speech recognition to address language issues found among stakeholders however its integration into SRE practices is challenging. In addition, [28] and [29] outlined that new scenarios are required for further consideration of the resource requirements of systems based on blockchain and the need for specialized approaches for their design and development of smart contracts. The studies further highlighted the need for new functions with design patterns for the development of blockchain implementation frameworks. Since the standardization of the Ethereum-based programmable interfaces, now businesses have to adopt such interfaces and data exchange formats as basic requirements [30]. According to [28], the application of blockchain improves the integrity of SLDC. Recently, [19] outlined the concept of blockchain-driven SRE to resolve contemporary development practices issues focusing on RT to enhance trust, transparency, and security. However, the study did not share any modeling guidelines or implementation details. In short, each study based on SRE in advanced computing systems demanded the need for a more reliable, trustable, and secure SRE platform. Hence, the state of SRE in the context of advanced technologies specifically blockchain offers

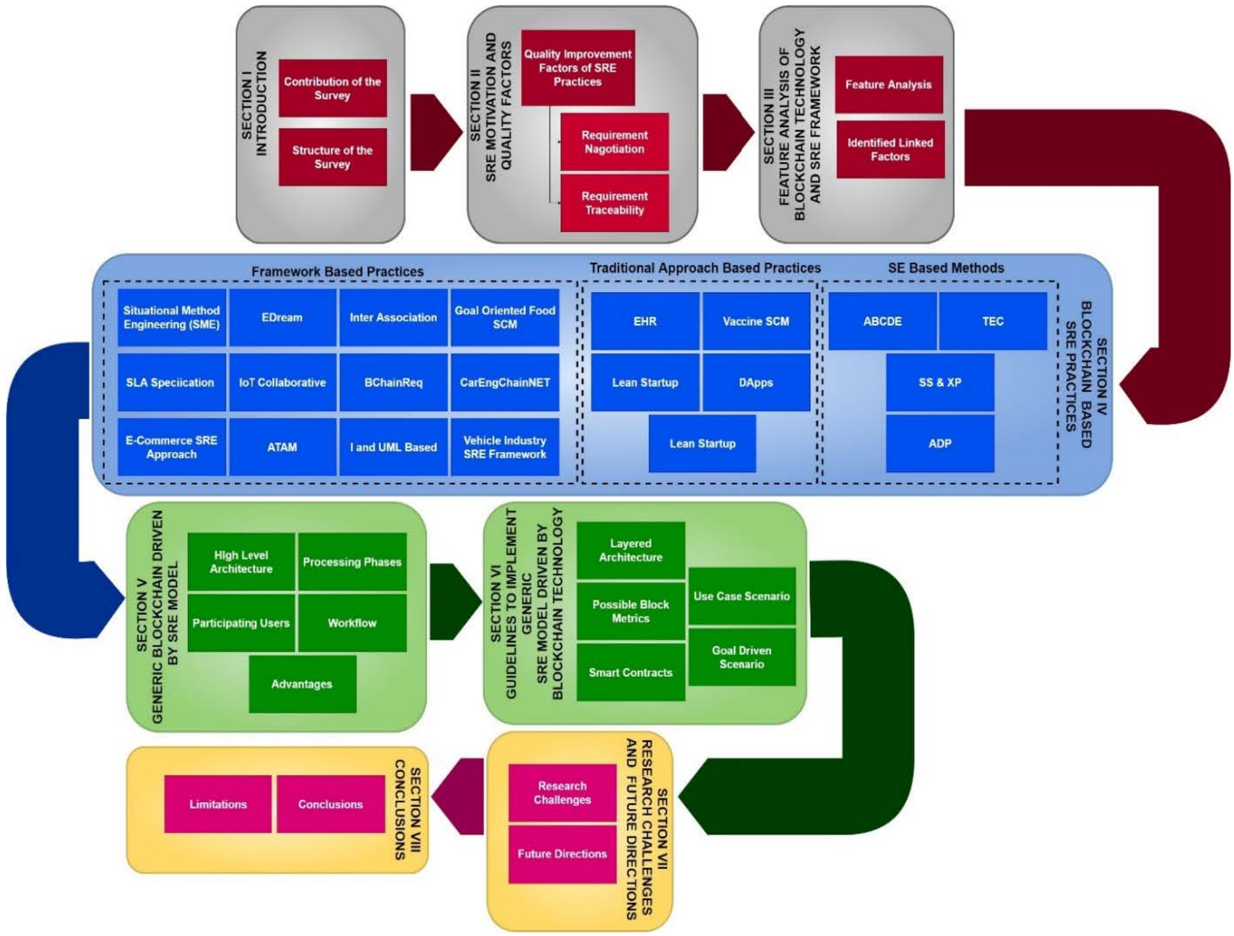

FIGURE 2. Structure of the Survey.

academia to explore this domain more comprehensively and empirically. The overview of a general SRE process (along with its goals and phase-wise outcomes) is shown in Fig.3.

In general, there are six phases of SRE and each phase plays a vital role in the development of SRS. Though previous studies improved the quality of SRE practices, however, each addressed specific domain-related problems [31]. The previous studies highlighted the factors related to Requirements Negotiation (RN), Requirements Traceability (RT), and Requirements Validation (RV) as most critical, requiring improvement in terms of transparency, quality, automation, and reliability [32]. The classification of SRE quality improvement factors is shown in Fig.4.

The RN aspect encapsulates several more factors like negotiation, quality requirements, and stakeholder relations. While the negotiation aspect is further divided into time and decision conflict factors. Similarly, the RV aspect incorporates the factor of unexpected delays which occur due to manual techniques employed for SRE validation practices and inconsistency in currently available tools. The lack of efficiency in the RV factor produces errors resulting in delays and demands for automation. Likewise, the RT aspect encapsulates the factors of trace links precision and architectural

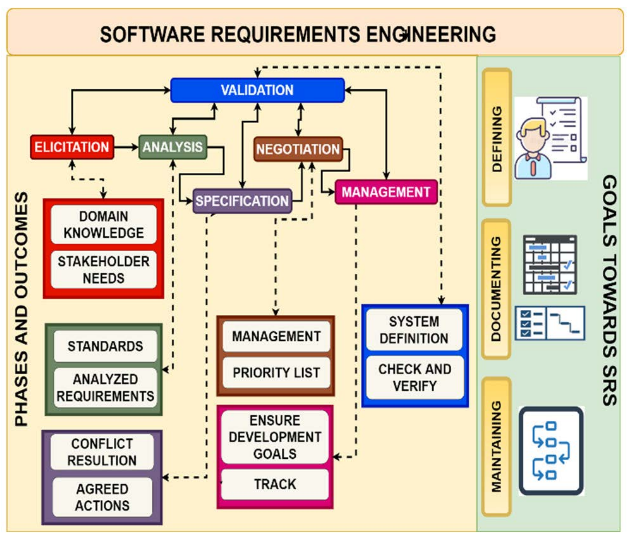

FIGURE 3. Overview of SRE in Software Development.

complexity which put constraints on tool integration and traceability of requirements. Hence, it is essential to briefly understand the in-depth comprehension of these aspects by

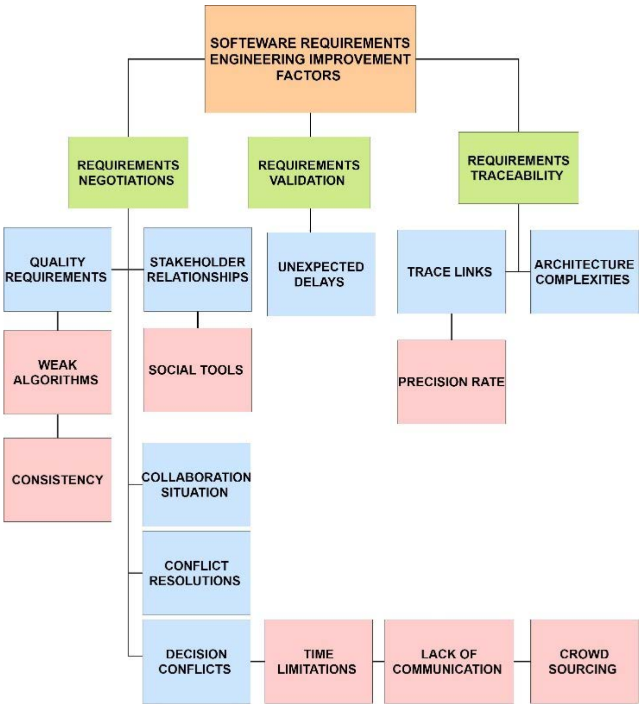

FIGURE 4. Quality Factors of Improvement in SRE Practices.

reviewing the problematic factors that come under these aspects. A brief discussion on these aspects is discussed in the next sub-headings.

A. REQUIREMENTS NEGOTIATION (RN)

The RN is considered the most difficult phase of SRE practices. In the RN process, the involvement of all the stakeholders and their agreement based on negotiation is required. The RN process takes place after the initial requirements are specified and analyzed to satisfy the needs of each stakeholder group through discrete negotiation practices, producing a refined set of requirements. The RN process varies from company to company due to diversified standards and domain considerations. In general, it is the effectiveness of RN that determines the overall success of a software project. The impact of any ineffective decision or ambiguity in RN produces drastic results, leading the project toward failure [33]. Hence, the understanding of problematic factors associated with RN requiring improvement is essential. These factors are shown in Table 2.

The underlying RN factors are linked with various stakeholder groups as shown in Fig.5 [48].

As the RN process revolves around different stakeholder groups making it significant to understand their needs and roles from the SRE perspective. The portrayed stakeholder groups are generally found in SRE practice. A successful software project is aligned with the needs of each stakeholder group [49]. Hence, it makes it crucial for SRE practitioners to identify and specify their needs. The conventional

TABLE 2. Requirements negotiation problem factors.

Factor	Problem
Consistency	The requirements of a software system originate from diverse sources and most tools and techniques lack precision. As a result, it becomes difficult to synchronize the specified requirements [34].
Time	The negotiation and conflicts between stakeholders generally require an excessive amount of time [35]. In most cases, unexpected delays occur penalizing the project in various ways.
Social Tools	The use of social networks and discussion forums is commonly employed for negotiation, but it has become obsolete and requires the integration of new technologies to work as an infrastructure for SRE practices [36] [37]
Stakeholders Relationships	Rapid innovations are enhancing the complexity of SRE practices. Power relationships remained effective for resolving conflicts, yet stakeholder conflicts still suppress the power relationships leading to ineffective decision making [38].
Negotiation	It is often difficult to devise requirements to satisfy the need of each stakeholder. RN requires an excessive amount of time and budget to select and employ tools such as passive support tools (emails, chats, discussion forums), active facilitated tools (used for analysis formulation, and problem-solving to reach agreement), and pro-active tools (administer actions of stakeholders). All these tools are employed to resolve conflicts and improve collaboration [39]. However, these tools lack quality attributed which disrupts the expected time and budget of a project.
Conflict Resolution (CR)	This approach focuses on resolving conflicts and reaching a unified agreement. CR employs qualitative (voting, compromise, agreement, and applying variants to solution parameters) and quantitative (overruling, Facts consideration, Plus Minus Interesting, Decision Matrix) techniques. These techniques are quite helpful however, improvements in these approaches are required to enhance the RN process [40].
Collaboration Situation	This approach is used to interconnect and manage diverse stakeholders using location and time(same time same location, same time different location, different time different location). Collaboration is either asynchronous(different time different location) or synchronous(same time same location). These approaches are helpful however, they are vulnerable to pandemic situations and stakeholder attitudes [41].
Conflicts in Decision Making	It is difficult to satisfy the diversified stakeholders of a project. Hence, conflict occurrence remains high and if the final agreement fails, stakeholders lose interest and excessive time is required to resolve the conflicts [42].
Communication Gap	The communication gap between stakeholders is a common problem, especially for remote participants. Generally, it is difficult to gather stakeholders into a single location or time. Even if they do, it is difficult for some stakeholders to explain their decisions [43].
Social Language	If stakeholders include foreigners, a translator is required for communication, costing an extra budget. In general, language differences can lead to bad decision-making [44].
Crowd Sourcing	Crowdsourcing enables the stakeholders to share their perceptions with a large group of online networks. Nevertheless, it is a complex task as it demands synchronization of location, time, and understanding between participants. In general,

TABLE 2. ***(Continued.)*** **Requirements negotiation problem factors.**

	stakeholders fail to understand each other's viewpoints [45].
Quality of Requirements	Requirments quality occurs due to an ineffective negotiation approach. Most projects fail due to ambiguity in requirements [46].
Weak Algorithms	Algorithms employed in negotiation processes are ineffective in verifying and prioritizing requirements [47].

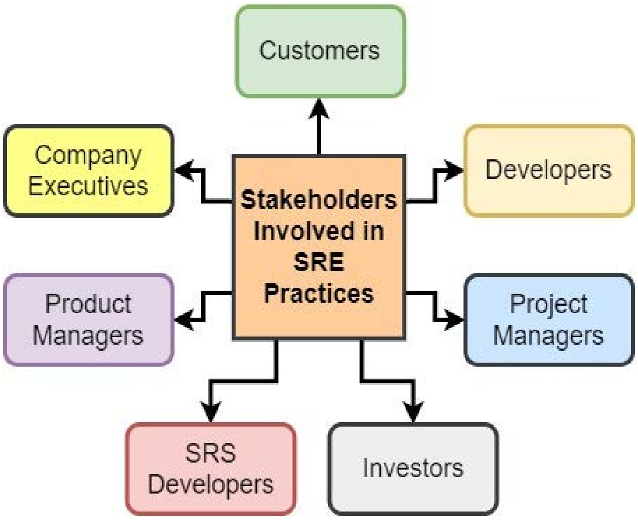

FIGURE 5. **Stakeholders Involved in SRE Practices.**

SRE practices incorporate the discussed RN factors that are linked with the portrayed stakeholders. Moreover, it is considerably difficult to satisfy the needs of each stakeholder. Hence, SRE practitioners utilize specialized management and negotiation models, tools, and approaches to deal with the project requirements covering the quality gaps. Although most software companies utilize conventional SRE practices to specific requirements based on negotiation, however large-scale or safety-critical systems include more stakeholders and demands for specialized SRE practices [50].

B. REQUIREMENTS VALIDATION (RV)

The RV process in SRE practices is associated with each phase. SRE practitioners employ RV methodologies from the initial to the final phase to minimize the risk of uncertainty and eliminate errors. The RV methodologies or tests performed to attain valid requirements are very difficult due to the lack of automated tools, especially for advanced computing systems. It requires careful examination based on expertise and domain knowledge. Any requirement error left during the RV phase leads a project towards uncertainty. The tools employed for validation resolve defects however, errors in negotiation may keep the stakeholders unsatisfied [34]. In general, due to manual methodologies and other stakeholders' considerations, RV is commonly performed informally based on peer review or ad hoc basis.

The most commonly used RV techniques are based on conventional methods such as walkthroughs, inspections, and expert reviews [51]. To err is the nature of humans and the conventional SRE methods produce unexpected project delays due to ineffective analysis tools, lack of knowledge, and diversification in SRE practices. These manual methodologies produce unexpected delays and incorporate errors.

C. REQUIREMENTS TRACEABILITY (RT)

The RT aspect based on SRE practices demands improvement in key areas namely architectural complexity, and automation of traceability links. Numerous studies proposed solutions to automate the RT process however, full automation is still not achieved yet due to a lack of trust in existing tools and SRE practices. In most cases, human intervention is essentially required to trace the links that are generated by traceability tools. In general, Information Retrieval has been applied to the majority of the proposed tools and approaches as the nature of the requirements is textual [52]. Nevertheless, each proposed method incorporates low accuracy and automation issues.

1) AUTOMATION OF TRACEABILITY LINKS

The most popular published Information Retrieval-based frameworks are latent semantic indexing [53] and the vector space model [54]. However, these methodologies a exhibit low precision rate. For instance, [55] outlined that about twenty to fifty percent of the proposed SRE traceability tools and approaches have a low precision rate. Nevertheless, in recent years, various scholars have proposed studies to enhance the precision rate. For instance, [56] recommended the inclusion of term-based relevance feedback in the creation of trace links. Similarly, [57] presented FORTA, a feature-oriented tool to automate requirements traceability. The results of their examination outlined higher precision and recall levels. Likewise, [58] presented a tool referred to as RETRO, enhanced with a dynamic thesaurus showing improved precision results. In addition, [59] reviewed various types of RN and RT tools and outlined that frameworks, such as DOORS, are more popular than pure RT tools. Nevertheless, none of the proposed tools supports the automatic link detection of the requirements. The limitations in each study presented approaches leading to low-level adoption of automated RT in SRE practices. The factor of weak algorithms in these computations approaches makes it difficult for the practitioners to successfully implement the fully automatic traceability detection, especially for large-scale or advanced projects.

2) ARCHITECTURAL COMPLEXITY

The architectural complexity is another factor contributing to the inefficiency of previously proposed RT methodologies. The complexity of architecture in existing RT tools requires improvement in the key areas of system integration and weak

TABLE 3. Requirements traceability empirical studies.

Ref.	Approach and Highlights
[60]	• Conducted SLR on issues based on 25 companies • Identified RT sub-issues • Highlighted that a single solution to all issues is infeasible
[61]	• Conducted case study on a global software company • Identified traceability adoption decision factors • Outlined that stakeholders are generally unaware of traceability practices
[62]	• Conducted a survey on decision-making factors for the selection of RT tools. • Provided a case study on an international software company • Contended that client awareness, process flow, company goals, stakeholder preferences, and return on investment are found to be significant factors. • Proved that senior management is generally unaware of RT tools being employed.
[63]	• Surveyed the issues of RT in SE Model-Driven approaches • Outlined the challenges of Social Science and Engineering disciplines
[64]	• Conducted an SLR to present the challenges of quality of support tools, stakeholder perceptions of tools, and change management • Outlined that weak algorithms pose a great challenge for the integration of RT tools

algorithms. The previous studies pointed out the issues of integrating the RT tools in existing SRE practices as a result of weak algorithms which impose infrastructural constraints.

As shown in Table 3, several empirical studies contributed to the identification of RT tools in the context of integration problems in SRE practices and pointed out the need for more improved applications. According to the reviewed studies, the themes of previously identified factors are based on: Manual Work; Diverse Artifacts, and Tools. While the themes of the presented solutions are based on: Automation; and Integration of Tools. The studies further outline that most of the solutions are not feasible for integration and automation due to the constraints imposed by the characteristics of the mechanization domain. These constraints are Architectural Complexity; Distributed Development; and Critical Security-Based Frameworks. Furthermore, the standardization of tracing links exchange and tracing activities is yet to be employed for safety-critical security-based frameworks. Moreover, in the context of traceability, tracing activities as still taken as an overhead in these frameworks.

III. FEATURE ANALYSIS OF BLOCKCHAIN TECHNOLOGY AND SRE FRAMEWORK

Blockchain technology is generally known as append-only data storage having decentralized, immutable, Peer-to-Peer (P2P), and transparent architecture [65].

Blockchain technology stores the data in a permanent fashion. In the blockchain network, the stored records or transactions are found in the form of immutable data blocks as shown in Fig.6. Every block in the chain of the blockchain

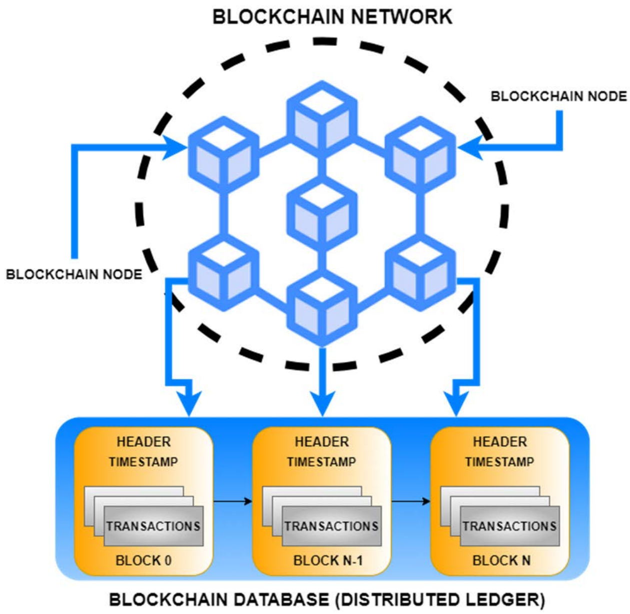

FIGURE 6. Overview of Blockchain Technology.

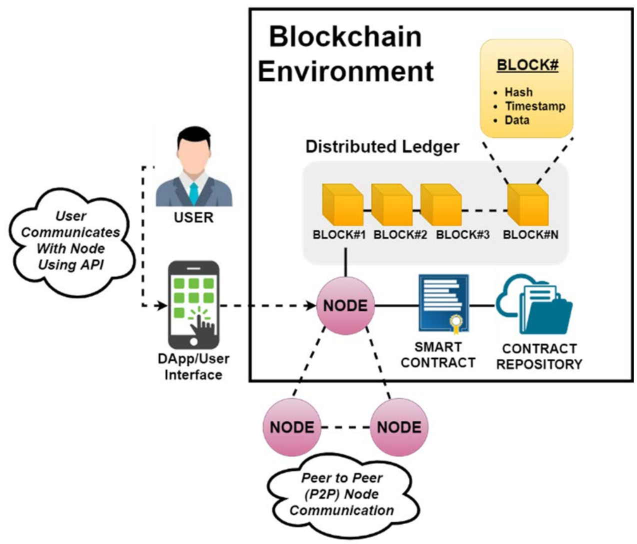

FIGURE 7. Blockchain Environment.

network comprises hash values that are computed through hashing algorithms such as SHA256 or consensus algorithms such as PoW (Proof of Work), PoS (Proof of Stake), or PoA (Proof of Authority) [66]. Each block in the chain carries the hash value of the previous block which is also known as the parent block. The first block of the chain is known as the genesis block. To access a blockchain environment, a user node has to communicate with the network using user-level interfaces commonly known as API as shown in Fig.7. The commonly used APIs to access blockchain networks include DApps (Decentralized Applications) and web portals [67]. Blockchain offers P2P communication between user nodes. A user node can participate and execute transactions based on smart contracts in the blockchain environment, after being registered to the network [68].

Blockchain technology-enabled applications have enhanced the infrastructure of a wide range of domains [69]. Similarly, the implementation of blockchain infrastructure in Software Development practices shows a promising direction toward the development of new applications in this domain [28]. Likewise, the blockchain-enabled applications based on SRE can provide a Requirement Management System (RMS). The RMS based on blockchain shows a promising direction to manage trustworthy traceability of requirements by enhancing security and transparency. To examine how effectively blockchain technology contributes to the SRE process, the employment of a feature analysis approach is essential to produce predictions. Feature analysis visualizes the characteristic connections of blockchain technology and the necessities of the SRE framework.

A. FEATURE ANALYSIS

The overview of feature analysis is shown in Fig.8. The inner-circle contains the attributes of blockchain technology while the outer circle comprises the provisions of the SRE framework. The features of the blockchain technology are shown inside the pink-colored circle with outward arrows while the necessities of the SRE framework are shown inside the blue circle with inward arrows. The outward arrows of the blockchain features and the inward arrows of the SRE framework necessities show that these characteristics are satisfying each other needs. The characteristics include external features such as crowdfunding and incentives which are also promising for an SRE framework.

As shown in Table 4, the characteristics of blockchain technology are contrasted with the necessities of the SRE framework.

Notably, the concept of SRE built on blockchain infrastructure significantly addresses the SRE problem factors and dispenses a secure and transparent environment. The detailed aspects of the analysis are discussed in the next sub-headings.

1) TRAIL OF REQUIREMENTS

The distributed ledger facilitates blockchain participants in providing the history of records [70]. An SRE framework-built blockchain infrastructure comprising high-level application features for users while allowing data transmission can potentially assist as a solution and enable the software companies to resolve requirement transactions by publishing the information for stakeholders' reusability and sharing. Also, it provides the capacity to construct the path of requirements by sequentially displaying the proceeding of objects leading towards software features. In addition, timestamping all the initial requirements from beginning to feature deployment permits to visibly record the information with connected blocks in the chain and consequently recreate the required feature to work on reproducibility and traceability.

TABLE 4. Feature analysis of blockchain driven SRE framework.

Blockchain Driven Software Requirement Engineering Framework Attributes	Decentralization	Timestamping	Access & Governance	Consensus Mechanism	Immutability	Cryptographic Hashing
Collaborative Environment	✓	×	✓	✓	×	✓
No Censorship	✓	×	×	✓	✓	×
Open Data	×	×	✓	×	×	✓
Identity Management	✓	×	✓	✓	×	✓
Extensible System	×	×	×	✓	×	×
Incentives for Collaboration	✓	×	✓	✓	×	✓
Equality of All Participants	✓	×	✓	✓	×	×
Stakeholder's Involvement	×	✓	✓	×	×	✓
System Metrics	✓	×	✓	✓	×	×
Data Sharing	✓	×	✓	✓	×	✓
Trail of Requirements	✓	✓	×	✓	✓	✓
Transparency	×	×	✓	×	×	✓

2) FEEDBACK OF STAKEHOLDERS

Blockchain technology employs consensus mechanisms to approve transactions and can work as an infrastructure for digital feedback systems to ensure the anonymity of the participants [71]. SRE framework built on blockchain assures productive input of the stakeholders for the given set of requirements. Moreover, it dispenses financial enticements utilizing tokens as an award for commitments and agreements based on a consensus algorithm. Furthermore, it offers the likelihood to produce new types of evaluations and enticements for SRE practices specifically the RN process.

3) OPEN DATA ACCESS

The blockchain network provides open access to the data available on the distributed ledger, it further enables the network participants to select the accessibility options for their data which depends upon the nature of the blockchain type as shown in Fig.9 [72].

In the SRE framework built on blockchain infrastructure, the stakeholders' nodes can have open access according to their rights. In a public blockchain, the data of a user is visible. While in private or consortium blockchain, only privileged users have the right to control the accessibility of their data. In such cases, some SRE documents can be made confidential such as financial statements. Nevertheless, private or hybrid blockchains offer participants to store data in a restricted manner [73]. For instance, if the SRE platform is based on a private or hybrid blockchain type, the client participation, if required at any point, can be requested by the development team or software company so that the user can grant access.

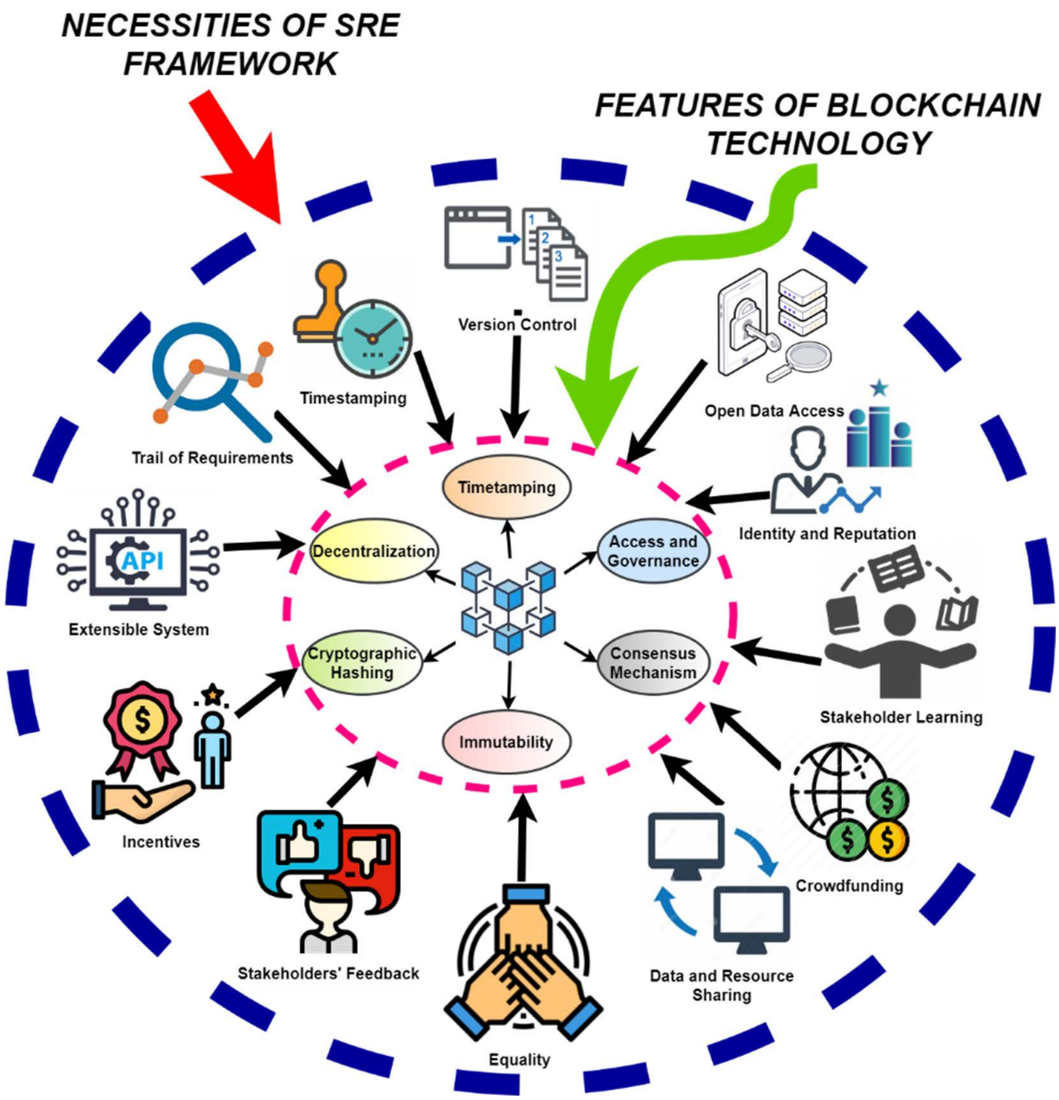

FIGURE 8. Fostering SRE Framework Using Blockchain Technology.

4) IDENTITY AND REPUTATION

To precisely compute the standings of stakeholders, a notoriety framework is fundamental to facilitate in producing enticements for blockchain members as affirmation for their contributions. As a repository of records, the blockchain is suitable to function as a catalog for storing appropriate participant information [74]. The scoring feature quantifies the quality and effect of the recorded information. Moreover, the ranking of the members is definite to empower participants in auditing and scoring the requirements. Furthermore, blockchain guarantees that no middle command administers the information, thus, scorings are made logically and autonomously by the members of the blockchain through their collective feedback.

5) EXTENSIBLE SYSTEM

An SRE application built on blockchain offers extensibility and addresses the integration problem found in conventional SRE practices. The likelihood to enhance a blockchain infrastructure is acknowledgeable and comparable to different frameworks; for instance, APIs empower the integration of applications within an environment. Hence, it becomes more feasible to speak with outside stakeholders through the integration of DApps or web portals for information communication and utilization of available services [75]. The scope of utilization is based on situations that can be consistently extended. An extra explanation would be the speed of growing innovations which makes it essential to incorporate flexibility and effortlessly expand prevailing frameworks, evading the demand of extra time and cost.

6) INCENTIVES

The consensus mechanisms based on blockchain technology are considerably significant for the RN aspect. These mechanisms enable the approved participants of the blockchain network to acknowledge each transaction before its execution. Moreover, the incentives for utilizing a blockchain network inspire individuals' involvement [76]. Furthermore, the safety

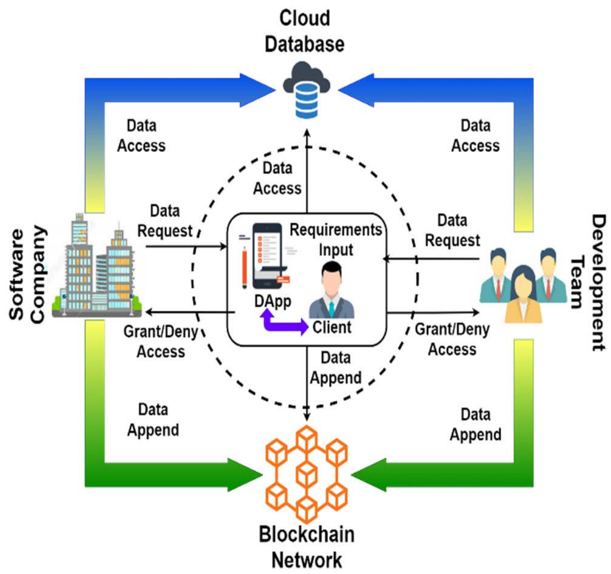

FIGURE 9. Data Access in Blockchain.

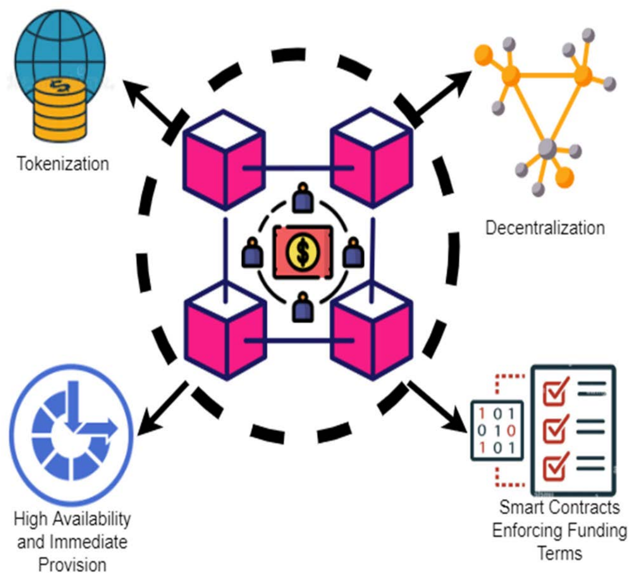

FIGURE 10. Blockchain Crowdfunding.

provided by the consensus approaches and the trial of requirements ensures immutability which assists in recognizing the contributions of the stakeholders through rewards and enables them to take interest and participate effectively.

7) EQUALITY

The decentralized property of blockchain is promising for SRE practices. It provides the facility to administer the discrete roles of the participants [77]. For instance, to form a committee that collaboratively decides the progression of SRE documents as the project proceeds. Moreover, it eases the coordination of the capacities and administrations into the existing work processes and outside interfaces without genuine exertion or expenses. Appropriate documentation, refined organization plan, and simple APIs are crucial to facilitate the mix of subsystems. Furthermore, the agreement model is an applicable element since to some degree it characterizes how many assets are expected to take interest in the organization [78]. In addition, the clients eventually utilize their recognizable programming to deal with their tasks and information with the likelihood of profiting from the features of a blockchain framework on an equal basis.

8) DATA SHARING

In the context of information sharing, a blockchain infrastructure ensures that there is no weak link because of its decentralized trademark [79]. There is no potential information misfortune and the distributed ledger guarantees accessibility as long as the association with it exists. Whenever a participant enters information into the blockchain, an agreement algorithm approves the approaching transaction to minimize risks of security vulnerabilities and information redundancy. The data in a blockchain is repetitively stored across all nodes of the network. From the viewpoint of data administration, information originators have the option to limit admittance to their data. Hence, information gets encoded in the blockchain, in a way that it is not permissible until its proprietor makes it open to different participants which is beneficial for SRE. In addition, off-chain storages like a conventional dataset or an IPFS are also connectable and usable using APIs or web portals.

9) CROWDFUNDING

In recent years, crowdfunding has become a developing economy and acquired a lot of prevalence through advanced platforms. The approaches exhibiting crowdfunding can acquire funding or assets in return for the acknowledgment of promising outcomes of an initiative. In the context of software development, some software projects require crowdfunding which is also beneficial for SRE. Blockchain-based crowdfunding has gained attention from scholars due to its impactful features [80]. Crowdfunding can be very significant for SRE practices in gaining the interest of investors. The typical features of blockchain crowdfunding are displayed in Fig.10.

Blockchain technology offers an agreement-controlled financial token framework, permitting participants to support projects through funding. The hashed aliases and concerning identities provide opportunities for abstract participation. As an expansion, management chains can serve to oversee crowdfunding projects, for instance, to disperse assets in complex subprojects, perform voting, execute programmed orders, or computerized activities [81].

10) STAKEHOLDER LEARNING

Blockchain technology offers data transparency in a standardized and ethical way [82]. It can support SRE stakeholders to stay consistently and changelessly connected to their information regardless it is an idea, initial requirement, complete feature, or a finished software project. Blockchain enables the stakeholders to know each bit of knowledge about the project. The immutable characteristic of blockchain guarantees that the trail of requirements stays unaffected. In case the stored data requires to be modified, for instance, if the SRS has to be remodified, it is likely to add new versions while the old documents get archived or checked. Non-specialist participants can likewise take part to contribute information to the project, particularly in big data assortments comprising basic data. Moreover, members may utilize advanced interfaces for estimating a wide range of properties for a given software project.

11) TIMESTAMPS AND VERSION CONTROL

The use of timestamping in blockchain transactions additionally complements and validates the values associated with time and the information is consistently accessible [83]. It provides the function of version control. Also, the reuse of collected SRS in SRE empowers practitioners to produce extra bits of knowledge through their review which is considered significant. Version control and timestamps are crucial for SRS documents. The process automation of SRS documents in SRE practices can be very effective in reducing the time taken to process a requirement or SRS.

12) TRANSPARENCY

In a blockchain network, the characteristics of data, tools, and source code are transparent for all participating nodes, enabling them to unequivocally understand what tools, instruments, and algorithms are doing and how the data is proceeding [82]. Transparency supplies confidence and additionally benefits all the members of the blockchain to coordinate in SRE practices or project development by recommending thoughts, modules, and updates for consistent improvement. It can likewise include prototypical programming, so experienced practitioners can provide their feedback to accomplish the most ideal arrangements.

13) STAKEHOLDER PARTICIPATION

Despite contributing towards SRS, stakeholders of a software project may contribute towards RN by sharing their unused system assets like the processing power of their systems for their interests. The distributed infrastructure of the blockchain technology supplies an ideal ground to proficiently assign assets and share them using an agreement system that upholds the reasonable distribution [84]. The clients may distribute their reviews of an underlying project, so other stakeholders' nodes with discrete roles can verify them. Such a type of participation strategy confirms the soundness of the SRE framework.

As shown in Table 5, using PoA with private or hybrid blockchain has the most potential to contribute to SRE practices.

TABLE 5. Blockchain consensus analysis based on stakeholder participation.

Cooperative Requirements	Type of Blockchain			Type of Consensus		
	Public	Private	Hybrid	PoW	PoS	PoA
Participation by Member Only	×	✓	✓	✓	✓	✓
Transparency Principle	✓	✓	✓	✓	✓	✓
Equality Principle	✓	✓	✓	×	×	✓
Togetherness Principle	✓	✓	✓	✓	✓	✓
Computing Efficiency	×	×	✓	×	×	✓

14) SYSTEM METRICS

All over the globe, quality metrics are crucial for software development practices and are considered an intrinsic characteristic of organizational policies. These metrics assist as an element for monetary bodies to choose from whom they share their resources such as to execute explicit software projects. Blockchain shows a promising potential to compute exact and dependable metrics for all primary stakeholders grounded on the giving and sharing of data through an open trustable foundation [85]. An SRE framework built on blockchain infrastructure can accomplish this through distributed architecture and the consensus algorithm by enabling the users of the network to take interest and participate in the estimation and checking of the core values of the infrastructure. Such a mechanism is fundamental for the subjective assurance of measurements in the finalized outputs. Though blockchain computes and approves the measurements yet it doesn't respond to the topic of what figures are pertinent and significant for an SRE ecosystem. Nevertheless, it is expandable to incorporate new opportunities.

In summary, this section demonstrated how the attributes of blockchain technology can satisfy the necessities of an SRE framework. As shown in Table 6, blockchain technology significantly addresses quality factors for the improvement of SRE practices.

The current innovative state of technology is well equipped for the acknowledgment of the SRE platform built on blockchain technology. However, an assortment of generalized and specialized tools with an appropriate agreement and administration framework, motivating force, and system metrics, is essential for its realization.

IV. BLOCKCHAIN-BASED SRE PRACTICES

Blockchain technology has changed the conventional hierarchy of a wide range of disciplines that were previously based on centralized systems by enhancing scalability, security, transparency, and trust factors [86]. Similarly, an acknowledgeable amount of work has been covered in the domain of blockchain acquainted SRE. Previous studies presented several SRE practices to engineer blockchain applications by fulfilling the quality gaps found in conventional software development methodologies. The blockchain acquainted SRE

TABLE 6. Identified quality factors based on feature analysis.

Feature	Linked Quality Factors
Trail of Requirements	• Traceability • Immutability
Open Data Access	• Accessibility • Security
Identity and Reputation	• Integrity • Reliability
Extensible System	• Interoperability • Scalability
Incentives	• Integrity • Productivity
Feedback of Stakeholders	• Integrity • Reliability
Equality	• Trust • Reliability
Data Sharing	• Accessibility
Crowdfunding	• Interoperability • Trust
Stakeholder Learning	• Accessibility
Timestamps and Version Control	• Security • Integrity • Traceability
Transparency	• Accessibility
Stakeholder Participation	• Integrity
System Metrics	• Security • Reliability

practices are classified into three categories namely frameworks, approaches, and SE-based methods as shown in Fig.11.

We have classified blockchain acquainted SRE practices into three categories. The first category comprises framework-based SRE practices. The '*Framework Based Practices*' category encapsulates specialized SRE systems for developing blockchain applications. These state-of-art frameworks-based practices enable the development of blockchain software systems rigorously and effectively using specialized processes. A total of ten framework-based practices have been identified and analyzed in this research. The second category comprises traditional approaches based on SRE practices for developing blockchain systems. The '*Traditional Approaches Based Practices*' category refers to the employment of traditional SRE tools and techniques for software development. These traditional SRE approaches are utilized in combinations with each other to capture reliable requirements for blockchain systems. We have identified and analyzed five different state-of-art approaches to developing blockchain systems. The third category consists of Software Engineering Models based on SRE practices. The '*Software Engineering Based Practices*' category encapsulates SRE methods based on Software Engineering methods for developing blockchain systems. The SRE methods employed in this category are dependent upon the Software Engineering model being employed. We have identified two Software Engineering based SRE practices that are used to develop blockchain systems namely architecture and agile based methodologies are employed.

A. FRAMEWORK BASED PRACTICES

The blockchain acquainted SRE frameworks are employed to develop blockchain system requirements through specialized mechanisms. These frameworks are novel methods that are specially designed to ensure reliability and trust factors to capture precise requirements for blockchain systems. There are a total number of ten SRE frameworks that have been presented by academia to build blockchain systems or use blockchain technology as an infrastructure to build software systems. In addition, each framework distinctively addresses the quality gaps found in SRE practices.

1) SITUATIONAL METHOD ENGINEERING (SME) FRAMEWORK

A recent study by [87] presented a blockchain use case development (BUD) framework based on the action research design approach (ADR) and situational method engineering (SME). The goal of the ADR approach is to address organizational complications through the production of innovative artifacts serving meaningful human purposes, while SME provides a structured approach towards the creation of a system. ADR consists of four phases and seven fundamental principles as shown in Fig.12.

The second phase of ADR encapsulates the BUD model. The BUD model is applicable in four distinct industrial cases namely banking, automotive, insurance, and construction. A goal orientation mechanism is employed while the BUD model aims to develop blockchain use cases, allowing the practitioners to follow a systematic methodology encompassing six stages. The model follows concrete principles and produces results that are superior to existing approaches. In stage one, the conceptual and technical basis of the blockchain system is recognized. In stage two, blockchain application scenarios are derived. In stage three, current blockchain trends are recognized. In stage four, detailed use cases are developed. In stage five, application scenarios are documented. Finally, in stage six, prototypes are developed and evaluated.

2) EDREAM FRAMEWORK

The use of blockchain technology in energy systems such as smart grids (SGs) has become prominent. The previous studies in the domain of blockchain energy systems overlooked the aspects of SRE practices. However, a recent study by [88] presented a blockchain flexibility framework for SGs that incorporate SRE aspects based on use case development. In the proposed framework, the flow of the SRE process is specially designed for the eDREAM project (blockchain-driven SG system) and it is based on an iterative approach to elicit and assess the requirements that are defined by stakeholders. The overview of the proposed SRE framework is shown in Fig.13.

The presented SRE process for eDREAM is driven by the needs of the stakeholders and based on several techniques such as outlining the requirements based on internal

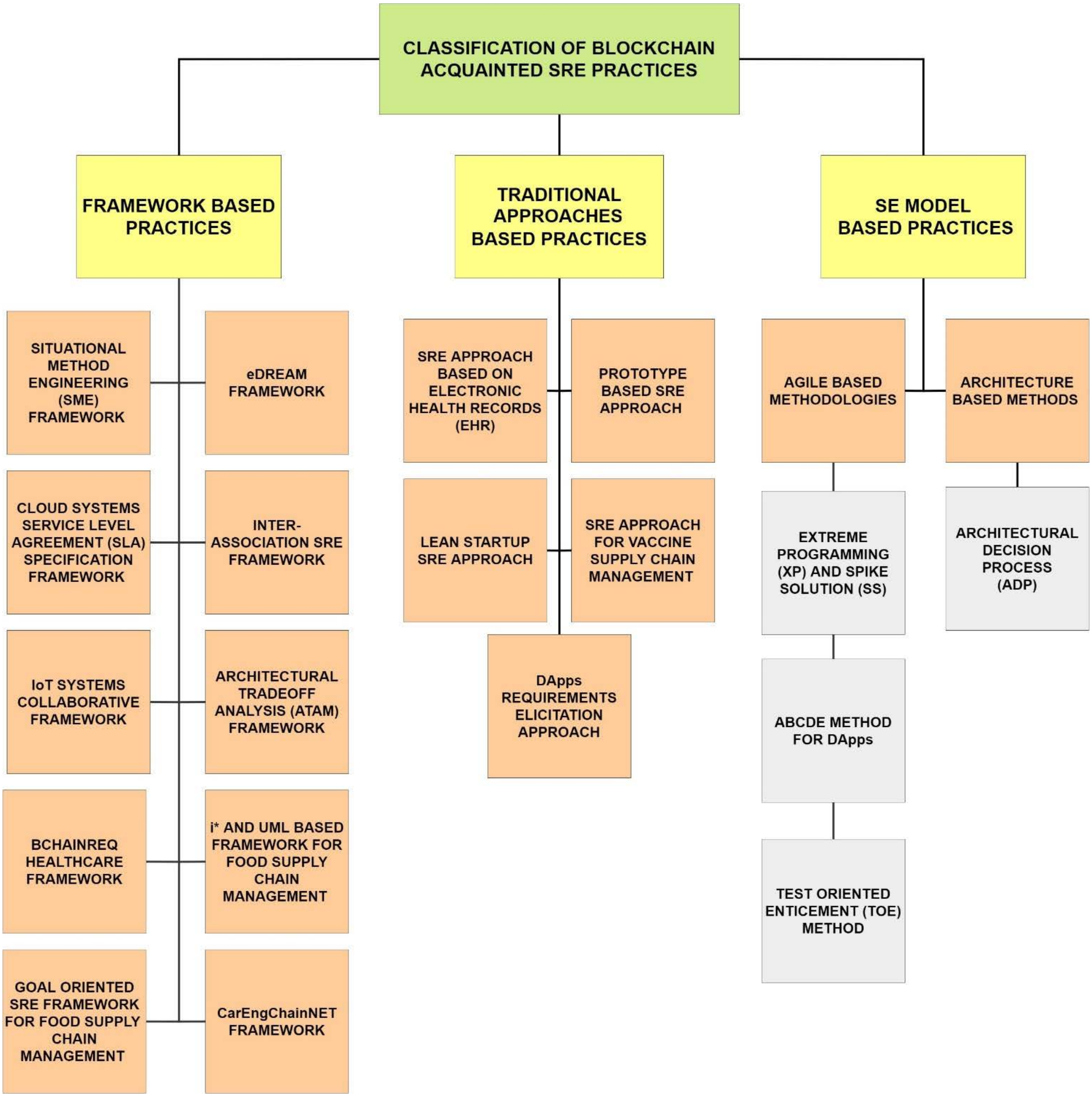

FIGURE 11. Classification of Blockchain Acquainted SRE Practices.

interrogations of the pilot studies executed by in-house experts to demonstrate a development plan and survey questionnaire for external stakeholders formalized through the study of existing scholarly literature, the inauguration of workshops, and organizing a public or conference consultations while specifically focusing on the concerning stakeholders. The study outlined that features offered by the proposed platform are perpetually contrasted with the requirements of the stakeholders. In this way, the stakeholders and end-users are involved in each phase of the SRE process to specify the eDREAM system and components. The study emphasizes employing a common method of requirement elicitation to offer a set of clear requirements for the developers of the system. The focus of the presented SRE framework is based on user scenarios and requirements involving stakeholders thereby ensuring the final output in terms of accessibility, usability, and functionality. The primary focus of eDREAM is based on scenario 1 as shown in Fig.14. It enables the

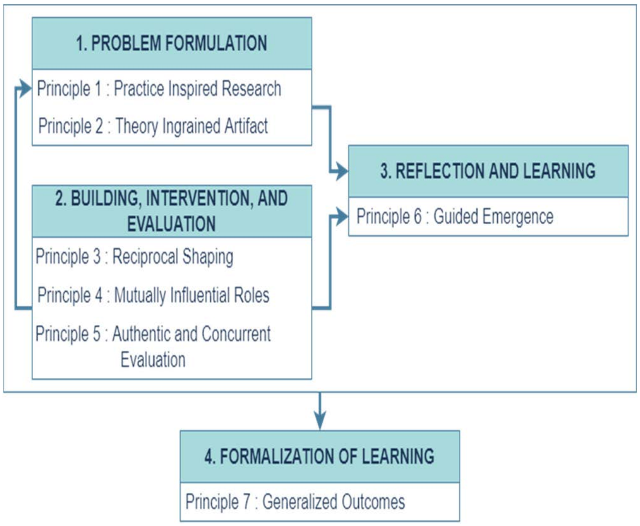

FIGURE 12. SME Process [87].

prosumers to offer load modulation and production based on smart contracts. A particular template based on a ''basic use case'' has been adopted for the definition of use cases [89]. The proposed framework leads to new business opportunities providing the aggregators the possibility to operate in a ''per-use'' model. Nevertheless, the model is only applicable for eDREAM.

3) CLOUD SYSTEMS SLA (SERVICE LEVEL AGREEMENT) SPECIFICATION FRAMEWORK

The use of blockchain technology in cloud systems is evident. A lot of studies have contributed to cloud computing to address the challenges found in this domain. However, the SRE aspects remain overlooked.

Nevertheless, [90] presented an SRE approach based on Service Level Agreement (SLA) specification for blockchain cloud systems. The study provides a state-of-art framework by illustrating how cloud-based blockchain systems are specified in smart contracts by using KAOS modeling while maintaining trust and security. It further outlined that requirements elicitation (RE) imparts a utilitarian operation to specify and design SLA-driven smart contracts. In addition, the proposed RE process for blockchain-based cloud systems is significant in terms of assuring that the resulting SLA specifications are complete, verified, consistent, validated, and accurate. The RE process is based on the KAOS model. The KAOS is a goal-driven SRE model based on four stages, as shown in Fig.15. In the first stage, SLA requirements are elicited. In the

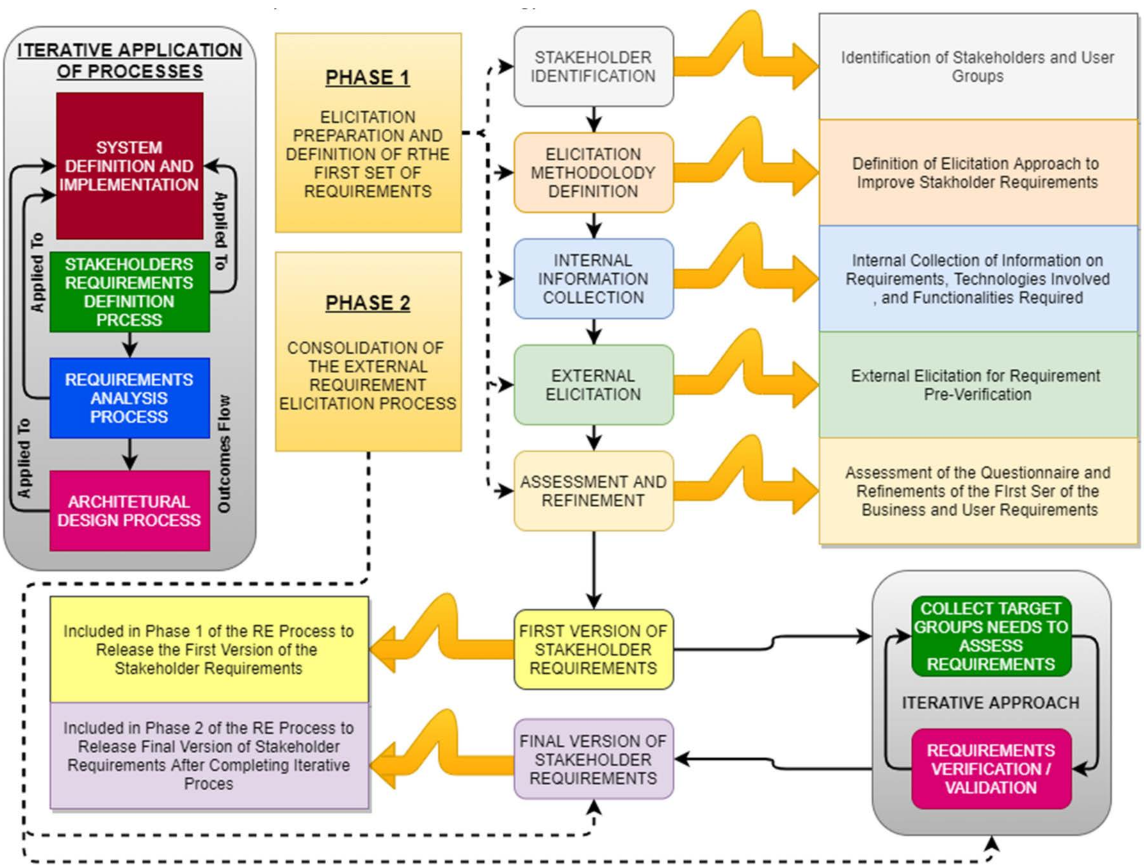

FIGURE 13. eDREAM SRE Process for Smart Grids.

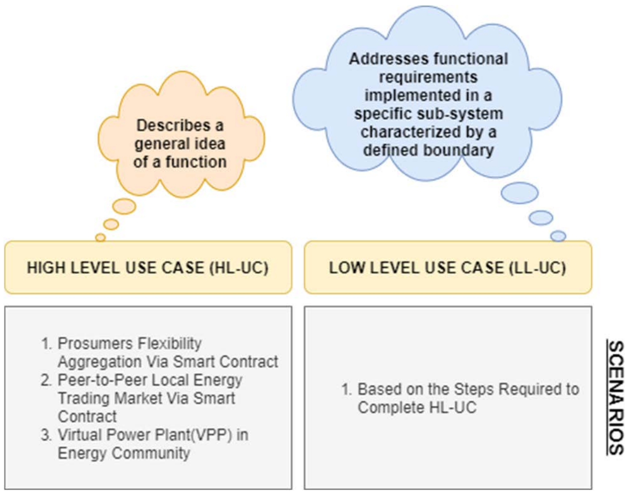

FIGURE 14. eDREAM Use Case Scenarios.

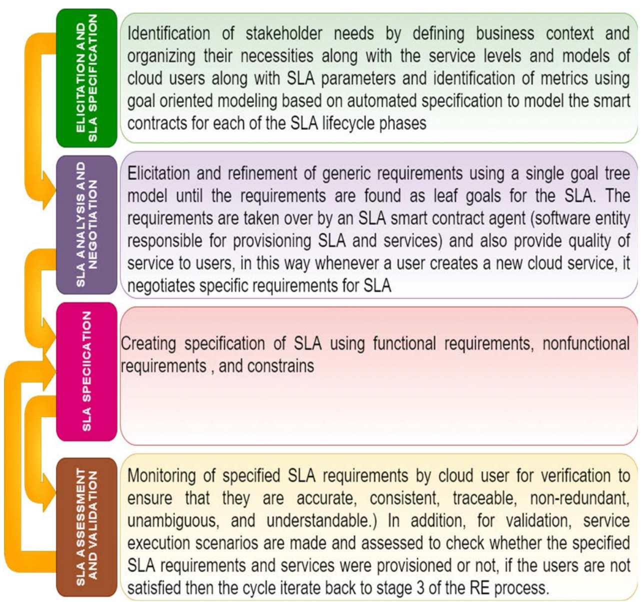

FIGURE 15. KAOS SRE Phases for SLA Specification.

second stage, requirements analysis (RA) and RN take place. In the third stage, the requirements are specified. In the fourth stage, the SLA specification is verified and validated. The third and fourth stages remain in a loop until the final set of requirements is obtained.

The study contributes a novel direction toward the quality and efficient engineering of blockchain-based cloud applications through KAOS modeling while maintaining trust and security. The detailed infrastructure of the proposed SRE framework to produce SLA specifications in blockchain cloud systems is shown in Fig.16. Notably, the smart contract agent remains intact until each phase while the SLA metrics and parameters are focused on each phase. The contribution of the presented framework is significant in the context of blockchain-based cloud computing applications.

4) INTER-ASSOCIATION SRE FRAMEWORK

The conventional SRE practices employed to resolve issues like RT and RN lack quality and require human involvement. For instance, a gap of trust exists in SRE practices and instruments. Moreover, the problems of integration caused by diversity in tools and practices also occur. Furthermore, these practices exhibit confidentiality constraints that impede the complete traceability across organizational scope and manual work. The studies based on blockchain-driven SRE are currently in infancy. However, the study by [19], presented the concept of the SRE framework based on blockchain to address the problems related to traceability and trustworthy management of software requirements for inter-association software projects. The proposed theory allows the stakeholders or participants to register requirements and metadata on the distributed ledger, and track the progress of their proposed information throughout the SLDC. The study further added that blockchain allows an auditable history of requirements that is verifiable and visible to authorized participants such as clients or investors. The study provided a good understanding of the theoretical concept however, its realization is yet to be executed.

5) IOT (INTERNET OF THINGS) SYSTEMS COLLABORATIVE FRAMEWORK

The use of IoT systems has increased rapidly. The IoT systems are rapidly evolving and integrating into new domains such as blockchain technology. The work done in blockchain IoT systems is evident, however, the focus on SRE aspects is currently trivial. Nevertheless, a recent study by [91] presented an SRE framework based on blockchain enable IoT (Internet of Things). The detailed architecture of the proposed framework is shown in Fig.17. The study has demonstrated the issues of IoT systems and the applicability of blockchain SRE to effectively engineer blockchain IoT applications. According to the analysis of the study, the key issues of contemporary IoT blockchain applications are caused by requirement ambiguities, communication gaps among stakeholders leading to poor requirement analysis, insufficient comprehension and understanding of SRE approaches, insufficient IoT domain knowledge, poor management of functional and nonfunctional requirement, and stakeholder conflicts in SRE practices due to variability in software development practices. Moreover, the proposed close collaborated SRE model provides a functional process for IoT-based blockchain applications.

There are three primary user groups of the proposed SRE framework for blockchain IoT systems namely IoT customers, IoT business personnel, and the IoT requirements team. The proposed framework enables the stakeholders to effectively understand terminologies related to IoT systems and to describe their needs based on the abstract system requirements. These abstract system requirements are examined and refined by the requirement analysts through elicitation, verification, validation, and mutual consensus.

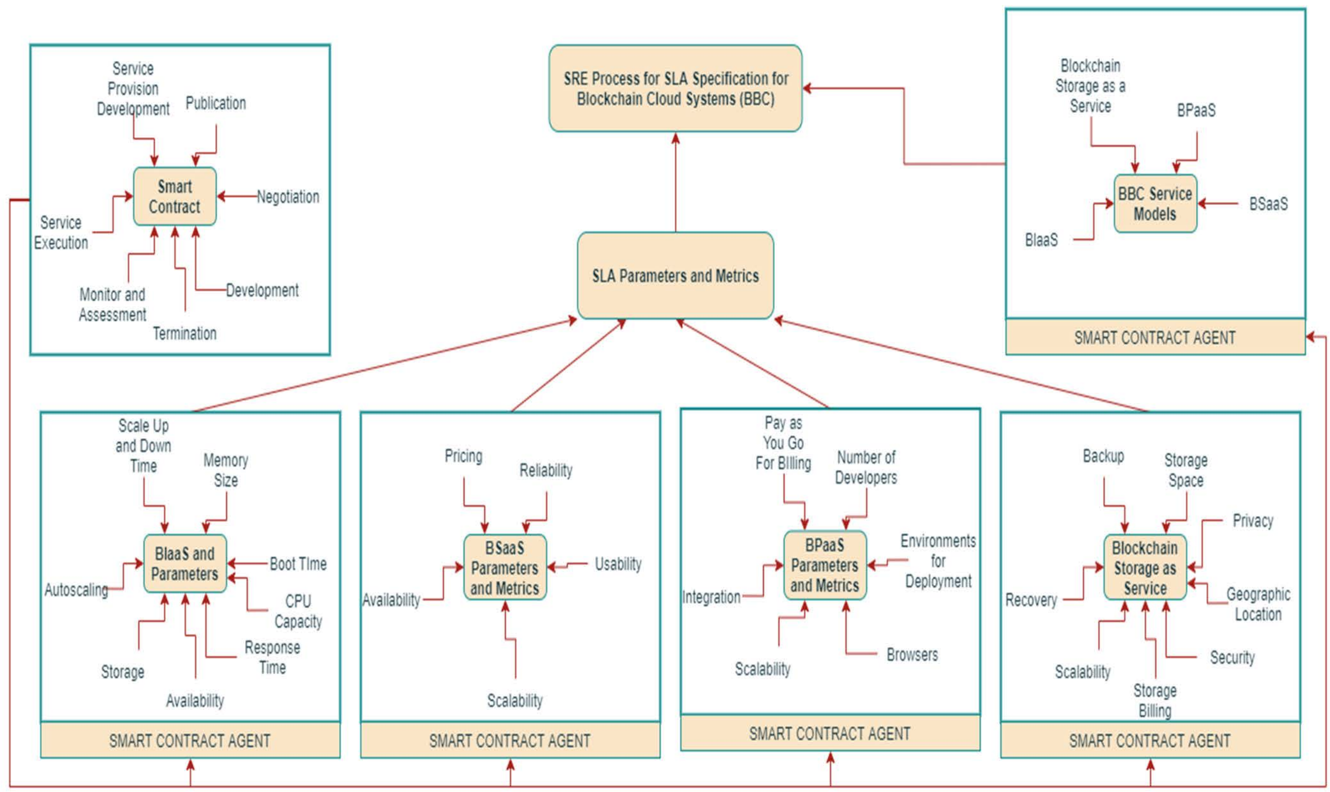

FIGURE 16. Infrastructure of SRE Framework for SLA Specification of Blockchain Cloud Systems [90].

After the specification verification, the requirements are added to the IoT-based blockchain network which is visible to every participant.

The use of a consensus mechanism in blockchain ensures mutual trust within nodes and is used for verifying requirements. It also helps in managing requirements. The contribution of the study remained significant however, the proposed model is constrained to blockchain IoT systems and not tested for other domains.

6) ARCHITECTURAL TRADEOFF ANALYSIS (ATAM) FRAMEWORK

The significance of the impact of blockchain on e-voting has captured the attention of scholars and practitioners. Numerous studies contributed to this domain however, the aspects of SRE remain neglected. Nevertheless, a recent study by [92] studied blockchain e-voting for national elections. The study employed the use of the architecture tradeoff analysis method (ATAM) and proposed a stakeholder-centric model which is based on promoting communal engagement to derive the quality attributes and associated risks of BANES (blockchain architecture for national e-voting system). The detailed architecture of the ATAM framework is shown in Fig.18.

Given the architectural perspective, the framework employs the use of elicited user-based requirements to deduce the scope of system capableness to fulfill the required quality characteristics based on critical analysis that involves stakeholders and domain experts. The ATAM framework is quite useful for system implementation guidance, project-oriented decision making, SPM (software project management), cost and benefit analysis, and improvement of system design. BANES qualified core characteristics based on ATAM deriving a secure voting system. In addition, the study further adds that voter security and validation are spotted as the most concerning aspects of blockchain e-voting systems and require significant attention from academia and practitioners.

7) GOAL-ORIENTED REQUIREMENT ENGINEERING (GORE) FOR FOOD SUPPLY CHAIN MANAGEMENT (SCM)

In recent years, the trend of food businesses has been oriented toward the adaptation of blockchain technology for Supply Chain Management (SCM). The increased use of blockchain technology in food SCM is evident. The growing demand for food SCM systems requires new specialized SRE practices. Recently, [93] proposed a goal-driven SRE approach to capture reliable requirements to develop blockchain-based food SCM. The approach aids in understanding the business goals in a detailed manner. The model focuses on recognizing system goals namely hard and soft goals for the blockchain-enabled framework. It acts as the preliminary step towards the formalization of requirement analysis. A case study based on blockchain-enabled food SCM has been employed by the

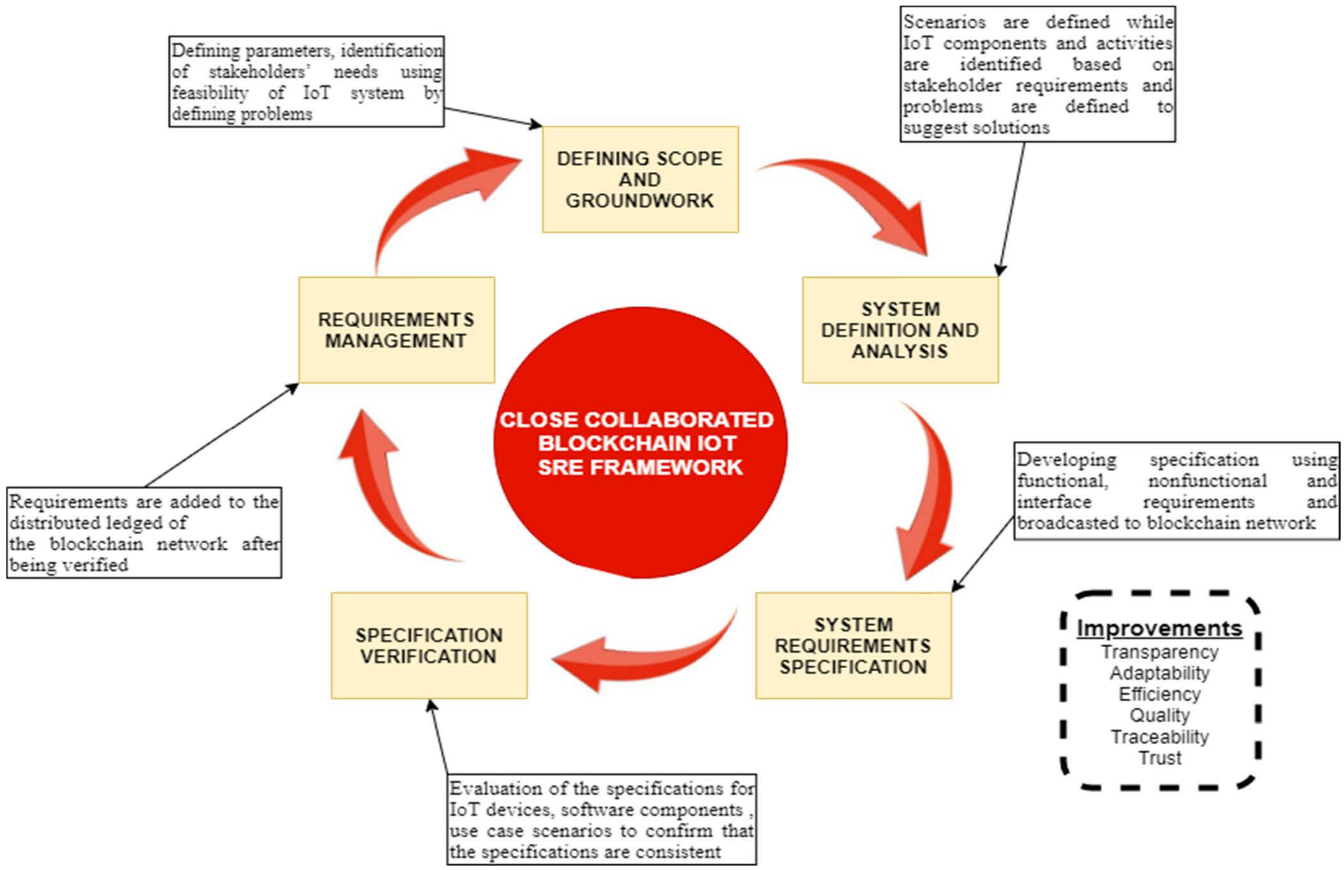

FIGURE 17. Close Collaborated SRE Framework for Blockchain IoT Systems.

study. The results of the study outlined that these goals are used by SRE to devise SRS and facilitate system design.

The proposed methodology is based on GRL (Goal Requirements Language) and Use Case Maps(UCM) based on URN (User Requirement Notation). URN focuses on modeling goals and justifying the decision rationale that finally forms a system and assists in modeling dynamic systems that can be adapted at run time. The use of GRL and UCM diagrams helps in various ways such as prioritizing the requirements, choosing alternatives or tradeoffs, finding completeness of requirements, envisioning the system scenarios, and resolving any conflicts earlier rather than later in SDLC The employed GORE model has various benefits, especially in the case of novel technologies where there is not much clarity on what can be the impact of its incorporation into existing business processes. The business process administration of SCM and integration of blockchain technology in it has been attempted in research. This is a conceptual contribution to how GORE can be used to include blockchain in food SCM.

8) CARENGCHAINNET FRAMEWORK

Over the years, the use of blockchain technology in the vehicle industry has gained popularity among scholars and practitioners. However, the previous studies have neglected SRE. Nevertheless, a recent study by [94] enlightened that contemporary car manufacturers face numerous problems based on inconsistent software versions, that occur due to incorrect handling of the software versions during vehicle lifecycle management. According to the discussions of the study, these issues that occur during vehicle production limit the capability of the companies to ensure integrity, traceability, and transparency. Consequently, to address the issues, the study presented CarEngChainNet, a novel blockchain-based platform for vehicle engineering which includes the SRE aspects and provides the features of producing new main-chains and sub-chains while facilitating immutable data management across the entire chain along with new approaches such as model-based systems engineering of the requirements and functional integration of software components in discrete areas of vehicle development. In this way, new transmission chains of vehicles with individually packaged software artifacts are securely transmitted from P2P into the vehicle. The proposed framework mainly focuses on requirements elicitation.

9) BCHAINREQ HEALTHCARE FRAMEWORK

The significance of blockchain technology in the healthcare domain is evident. Nevertheless, technical challenges

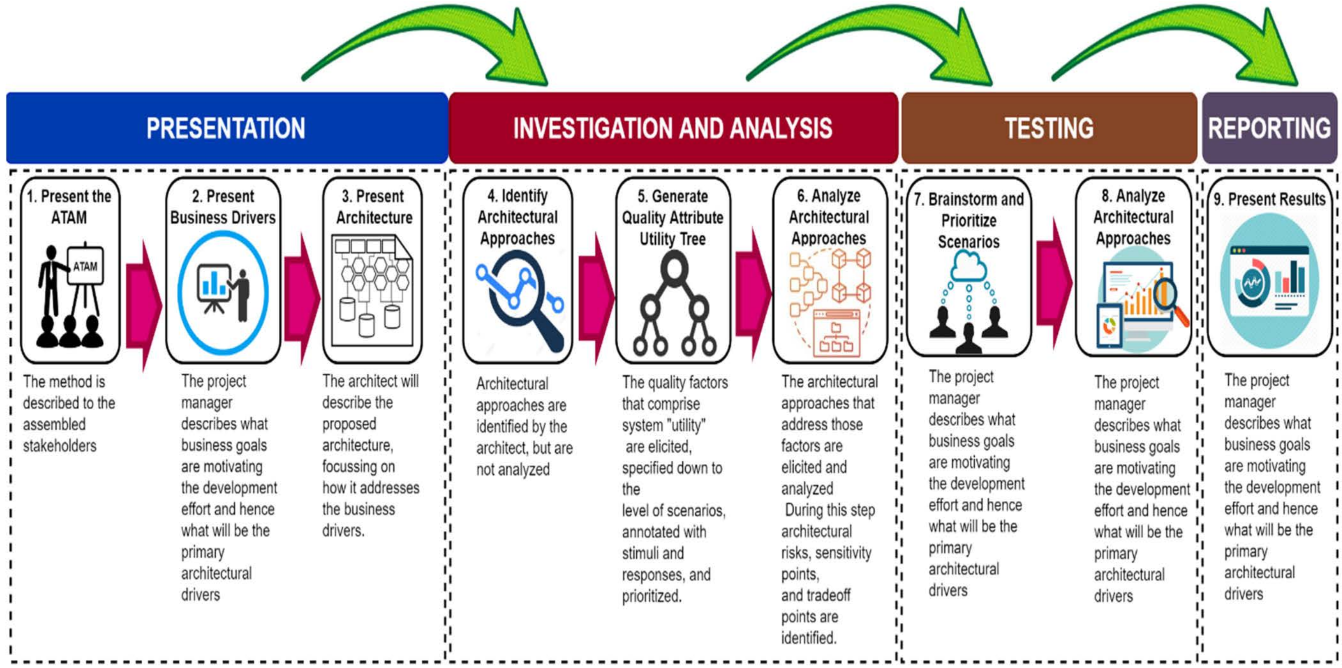

FIGURE 18. The ATAM Framework.

such as hacking and fraud impact the reliability and trust issues towards the blockchain in this domain. In the healthcare domain, the requirements elicitation is considered the most crucial phase of SRE during which the project requirements are identified based on negotiation involving every stakeholder, starting from documentation to the development stage. In general, the frequently employed requirements elicitation practices lack in considering trust elements. Studies published in blockchain healthcare systems neglected the aspects of SRE. However, a recent study by [95] presented an automated requirements elicitation framework that works as a catalog for trust requirements for blockchain healthcare applications. The goal of the proposed framework is based on refining the requirements captured through the acknowledgment of the community towards the healthcare domain, providing optimistic recommendations, and producing sustainable blockchain solutions. The proposed framework is referred to as 'BChainReq'. It is classified as a modeling tool to assist during the requirements analysis activity based on trust for implementing acknowledgeable blockchain healthcare solutions. The BChainReq supports requirement analysts to examine the trust requirements or trustworthiness level based on stakeholders as portrayed in Fig. 19. There are various phases incorporated in the BChainReq framework. Firstly, blockchain application requirements are elicited and returned in textual form elicited from stakeholders with requirement engineers. Secondly, these requirements are entered into BChainReq to validate trust requirements. Thirdly, the string matching algorithm is executed by contrasting trust factors and attributes with textual requirements that have been integrated with the BChainReq library by

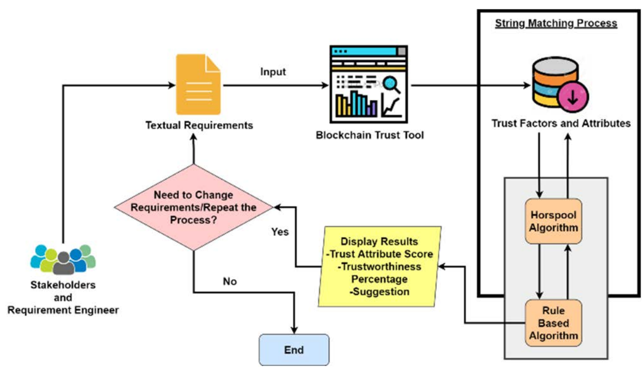

FIGURE 19. BChainReq Framework [94].

employing Horspool's algorithm. Fourthly, after the termination of the string matching phase, BChainReq proceeds to measure the trustworthiness percentage and trust factors level by using a rule-based algorithm. Fifthly, BChainReq proceeds to measure the trustworthiness percentage by using a rule-based algorithm. The output of the string matching phase is displayed as output to the requirement engineer and stakeholders. Sixthly, the suggestions of missing trust requirements in the textual requirements are displayed. Finally, after the analysis of the results, the stakeholders and requirements engineer terminates the process or append changes.

The proposed SRE framework effectively reduce time, human effort, cost, and human interference during the requirements elicitation phase in gathering quality trust requirements from stakeholders and assist novice requirements engineer to write precise and reliable blockchain requirements. Nevertheless, the proposed framework is constrained to the healthcare application domain.

10) I AND UML BASED FOOD SUPPLY CHAIN MANAGEMENT (SCM) FRAMEWORK

Similarly, in the context of blockchain food SCM, [96] illustrated the employment of existing SRE techniques namely software modeling and organization language to enhance the documentation of blockchain projects in this domain. The two common types of SRE frameworks namely UML and i* based on sequence diagrams and use cases are utilized to propose use case diagrams based on requirements. The study further recommended including privacy concepts based on the proposed graphical concepts to enhance SCM models. Furthermore, following the study, the regulations and policies on the blockchain are unprecedented and enhancement of the i* model for forthcoming contemporary laws seems to be highly promising.

The study illustrated the findings of a real-life case study referred to as 'Farm-to-Fork', providing a blockchain solution for the SCM of farm animals as shown in Fig.20. The study revealed that i* model assists to ameliorate the understanding of the goals, intentions, and social aspects of all the stakeholders in the network of the blockchain. It concentrates on the interdependencies between the actors in the blockchain which reflects the core of trust and mutual understanding in the network. The study recommended the use of (extended) i* representations and the aforementioned UML diagrams in a complementary way because of the discrete perspectives they provide to develop a blockchain. The results of the study outlined that i* fits during the early RE/OM phases of the SRE process to understand the 'why' of the SCM systems, while UML better fits the late RE/OM and design stages by offering concrete diagrams to understand the 'what' and 'how'.

The proposed framework concentrates on the reasoning or internal rationale of participants related to dependencies between actors. In addition to the interaction between the different SC participants, the supermarket's ability to specify the quality requirements for each stakeholder is also important and is therefore depicted with the SRE model. The SRE model focuses on the interdependencies between the supermarket, the blockchain, the smart contracts, and the consumer. The supermarket is an especially important node as the final product arrives here and is sold to the consumers. Hence, chicken meat must be of the best quality to sell to consumers.

Most benefits of blockchain adoption are experienced in this stage of the SCM. In addition, there is no more wastage because of higher quality and avoidance of contaminated products. The contaminated products can no longer get into the hands of consumers which limits health risks, and consumer awareness is higher because they can scan the QR code on the packaging of the chicken meat to check the history of the product. Given these four important actors (supermarket, blockchain, smart contracts, and consumer), the SR model can understand the 'why' of interdependencies.

The original i* extension to describe regulatory compliance was specifically targeted towards the SR type of models in i*. In Fig.20, the integration of the EU GDPR law in the SRE model has been presented. The overall aim of the regulation is to protect personal data. This can be achieved through guaranteeing the 'right to be forgotten', keeping data processing transparent, only recording data when necessary, keeping the data within the ER, and ensuring data integrity, security, and confidentiality.

B. TRADITIONAL APPROACH BASED PRACTICES

Blockchain acquainted SRE approaches utilize conventional techniques in a specialized manner to cover the quality factors and produce reliable requirements. These approaches eliminate possible vulnerabilities related to blockchain system development. There are a total number of five approaches presented by academia to develop blockchain systems. Each approach is designed for a specific domain.

1) ELECTRONIC HEALTH RECORDS (EHR) SRE APPROACH

Some blockchain projects are fully stakeholder or end-user-oriented and require advanced data security mechanisms such as in the case of medical patient data. The characteristics of blockchain include verification of transactional data based on a decentralized architecture which proves to be a promising solution for securing critical data. The contributions of the studies based on blockchain EHR are evident. In this context, [97] proposed an SRE approach in the scope of the research paradigm and design science based on blockchain EHR systems. The study introduced an SRE approach for blockchain EHR based on recognition of stakeholders and systematic elicitation of requirements. In the proposed approach, the identified requirements are produced through a review of the literature and semi-formal meetings with the medical professionals. Afterward, the developed framework is evaluated using workshops comprised of several members. Consequently, a five-layered architecture is produced based on identified groups and specified requirements. Finally, improvements in traceability, data security, and automation by smart contract are indicated.

2) LEAN STARTUP APPROACH

Business startups based on software development require consistency to gain a competitive position in the industry. A recent study by [98] proposed a blockchain product-oriented development approach based on Lean Startup. In the proposed approach peer review has been employed for the validation process based on inspection. The validation process has been defined in a general way, and it is applicable in different business domains that are supported by blockchain technology. The results of the study outline that the Lean Startup approach can address a lot of problems that are inherent in a software development process, such as requirement specification and analysis.

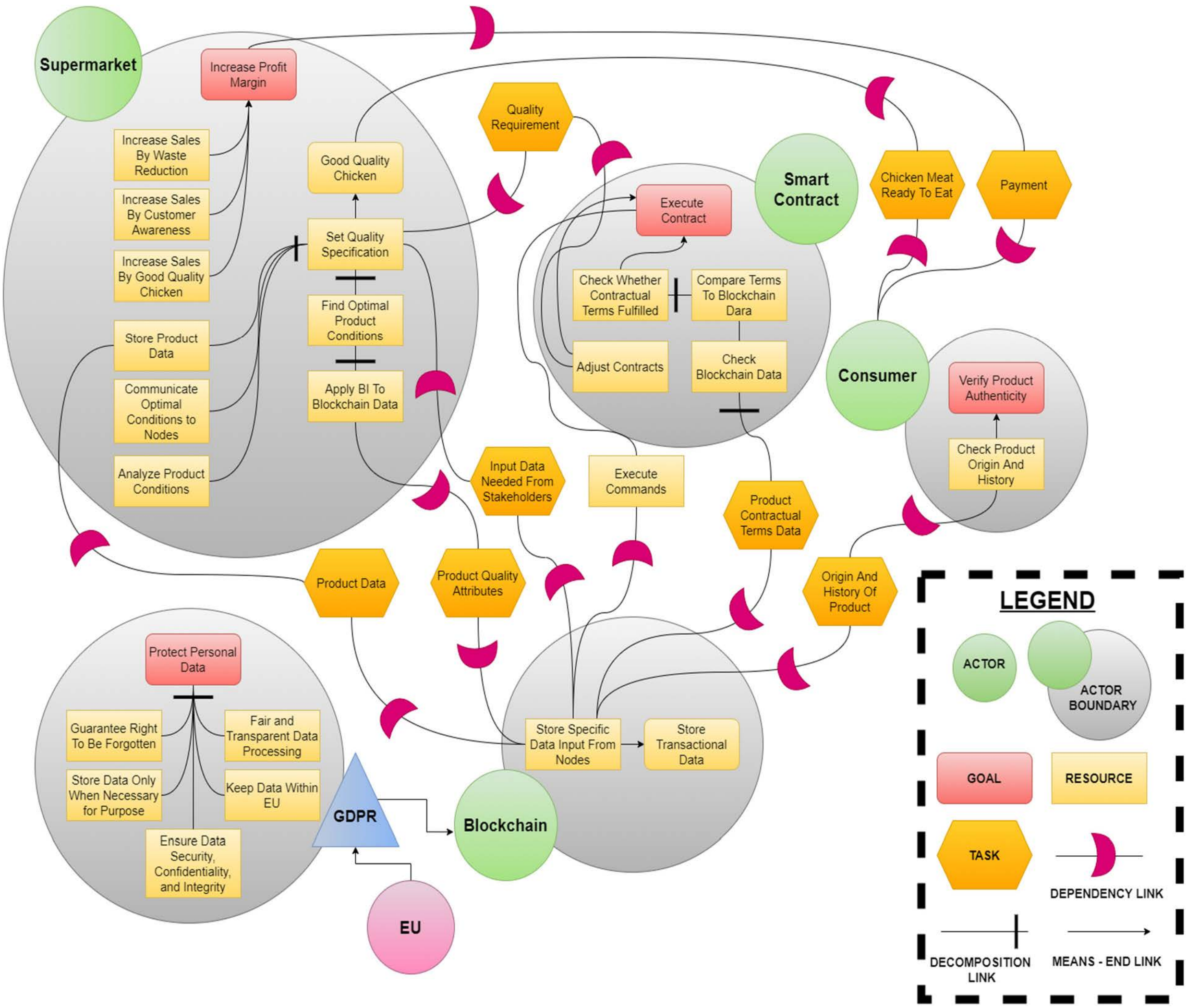

FIGURE 20. Strategic Rationale Diagram for Specification of Quality Requirements Based on Farm-2-Fork [96].

3) DECENTRALIZED APPLICATIONS (DAPPS) REQUIREMENT ELICITATION APPROACH

Nowadays the use of DApps has become prominent. However, the SRE aspects based on blockchain DApps are presently understudied. Nevertheless, a recent study by [99], presented requirement elicitation techniques for applications running on blockchain technology and highlighted preliminary results based on a case study. The study raised the concern about the insignificance of SE aspects of decentralized applications (DApps) as well, besides their emerging popularity. The study has bridged the gap by presenting the SRE aspects in this domain. The presented SRE approach is based on the collection, examination, and integration of DApp user reviews to produce the first set of user requirements for DApps. Through the employment of several examples as case studies, it has been signified in the study that DApp requirements have practical implications for both researchers and practitioners. The latter can use the results to guide them in the design of DApps, while the former may utilize their article as a first result to build upon or advance the field of SRE practices in the context of blockchain applications. The proposed approach utilized use case scenarios, inspection, and domain analysis to elicit the requirements.

4) PROTOTYPE-BASED APPROACH

A recent study by [100] implemented a prototype based on blockchain e-voting and evaluated various costs related to the scalability and efficiency of blockchain-based e-voting. The study provided some requirement elicitation techniques such as Focus Groups, Semi-Structured Interviews, and Scenario-based walkthroughs for system implementation. Nevertheless, the details of the type of requirements that the approach could handle were not discussed in the study as well as other aspects of SRE such as validation and verification. In addition, some studies [101] and [102] focused only on stakeholder engagement feedback in SRE using focus groups, semi-structured interviews, and walkthroughs.

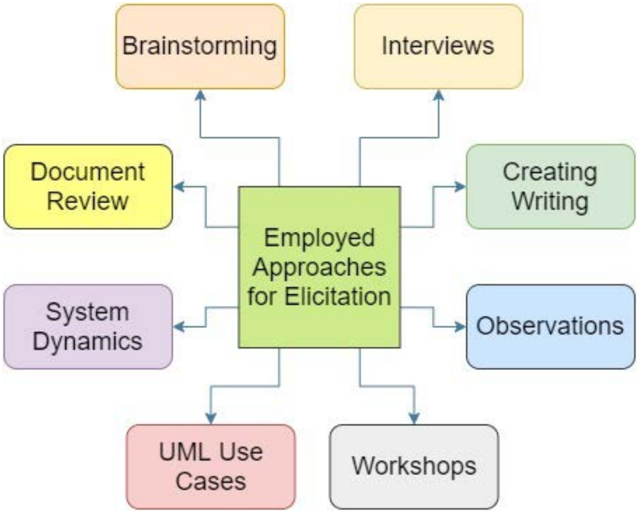

FIGURE 21. **Employed Elicitation Approaches.**

5) VACCINE SUPPLY CHAIN MANAGEMENT (SCM) SRE APPROACH

Since the outbreak of the COVID-19 pandemic, the domain of vaccine SCM has acquired great attention from scholars and practitioners. The previous studies published in vaccine SCM based on blockchain is evident, however, the SRE aspects in this domain remain understudied. The SCM systems based on medical applications process critical data and require complete, secure, and correct requirements to guarantee high quality to fulfill the needs of all the stakeholders to enhance system reliability, making it crucial to study this domain.

A recent study by [103] proposed mechanisms that are mandatory to be executed in developing or gathering complete and reliable requirements, producing a transparent, secure, and effective blockchain framework for vaccine SCM. The study proposed requirements elicitation tasks for mobile and web-based blockchain frameworks for vaccine SCM. The results of the study provided an understanding of the operation of the existing vaccine SCM and the requirements for mobile and web applications of the framework.

In addition, the requirements for the construction of desired vaccine SCM application were also captured. The study provided end-to-end visibility of vaccine SCM through mobile or web applications. The approaches utilized for requirements elicitation are shown in Fig.21. The requirement elicitation techniques proposed in the study to define the project scope are classical traditional techniques (interviews, questionnaires, and surveys), cognitive analytical techniques (card sorting, laddering, and repertory grids), modern and group elicitation techniques (brainstorming, joint application development (JAD), prototyping) and social analysis (ethnography, direct observation, passive observation). The employed requirements elicitation methods are used in designing distinct aspects of the project.

C. SOFTWARE ENGINEERING MODEL BASED PRACTICES

The Software Engineering driven SRE techniques are employed to develop blockchain applications. These SRE practices are based on SLDC models such as agile or architectural development models. These practices follow the SE approach being employed for software development. Moreover, these SRE practices are generic and utilized for all blockchain application domains. However, only a few studies have been published in this context that presented SRE techniques. We have presented two methods namely Agile and Architectural Software Engineering. These methods further include SRE techniques to develop software systems.

1) AGILE BASED DEVELOPMENT

The agile encapsulated frameworks primarily prioritize delivering a functioning component of the software applications and pay less attention to detailed software specifications such as detailed requirement descriptions, and architecture descriptions [104]. Such an approach significantly prolongs software transfer to the support team thereby making maintenance more complex and requiring more time to find defects due to unavailability of complete SRS or requirement description but fosters quick development.

a: EXTREME PROGRAMMING (XP) AND SPIKE SOLUTION (SS)

Blockchain-based application using agile prototyping has been discussed by [105]. Based on the discussion of the study, the use of Spike Solution in Extreme Programming (XP), is highlighted as a useful technique at the early stage of blockchain-based application development for the following purposes: Requirements elicitation, specification, and hence far from target blockchain platform; Forming a base target blockchain application architecture without dependency to a specific platform; Identifying uncertainties in system quality factors such as security, transaction execution performance, as well as the trade-off between both factors.

b: ABCDE for DApps

A recent study by [106] discussed the usefulness of user stories in blockchain-based agile development. It is based on SCRUM. User stories effectively support defining blockchain application features as seen from the stakeholder's viewpoint, to fetch early feedback to validate the required smart contract. ABCDE is a software development process employed to design blockchain applications. It is iterative and incremental and uses formal notations such as UML diagrams describing the design of the system.

c: TEST ORIENTED ENTICEMENT (TOE)

Recently, [107] presented a test-oriented enticement mechanism framework. The framework utilizes test-driven incentive mechanism that makes use of blockchain principles in which software developers are miners and testers are validators. The framework addressed the non-functional requirement and integrity of largescale agile software development practices. It furthermore perceived a byzantine problem with participants (developers) creating bugs. In which developers code while testers create blocks on each increment of the software module.

2) ARCHITECTURE-BASED DEVELOPMENT

Recently, the popularity of blockchain has significantly enhanced in reputation because of its attributes based on singularity, empowering the improvement of contemporary revolutionary decentralized systems. However, the difficulty of integrating the blockchain system into architectures and the plethora of possible choices hinder its implementation.

a: Architectural Decision Process (ADP)

The study by [108] discussed easing the implementation of blockchain in discrete firms, especially with the development of an automatic verdict approach to solve this difficulty wherein requirements are first-class citizens, an information base containing architectural patterns and chains refined over time, and an infrastructure originator capable of produce results into architectural stubs. The proposed process takes the input of functional and non-functional requirements and is referred to as strict requirements. Moreover, the study provided contemporary progression in the context of this decision process, by introducing the preliminary model that can select the most suitable blockchain between multiple selections and their proposed process-oriented benchmarking technique. Furthermore, the study presented preliminary results on this topic which is a decision process for blockchain technologies called the BLADE project which can recommend the most suitable blockchain from non-functional software-related requirements and user preferences. The decision process performances are expressed in wide intervals instead of fixed values.

D. SUMMARY OF PRACTICES

The discussed practices significantly contributed toward the improvement of various SRE factors as outlined in Table 7.

V. GENERIC SRE MODEL DRIVEN BY BLOCKCHAIN TECHNOLOGY

A generic SRE model built on blockchain infrastructure has been presented in this section. The proposed model achieves the following core objectives:

- To provide an abstraction of the SRE framework built on blockchain infrastructure.
- To facilitate efficient conveyance of model details between stakeholders.
- To offer a point of reference for system designers to extract system specifications.
- To deliver a documented model for future reference and collaboration.

A. HIGH-LEVEL ARCHITECTURE

The previous studies have proposed domain-specific blockchain acquainted SRE practices to capture accurate, consistent, and reliable requirements. A generic model having the capacity to proceed without domain constraints is currently essential. Based on the reviewed blockchain acquainted SRE practices, we formulate a generic high-level architecture based on five phases, dispensing an efficient requirement management system (RMS), as shown in Fig. 22. The components and process cycle for developing a reliable SRS for any domain are also portrayed. A good amount of time and expense to develop the final version of SRS. The use of blockchain infrastructure for the SRE platform can leverage stakeholders, the development team, and the software company.

The use of organizational standards, domain information, system information, stakeholder needs, and demands, rules, and regulations, verified and validated transactions, and system modeling are the inputs and outputs of the platform. The SRS is developed after being processed through various stages.

In stage one, initial requirements, standards, and domain scope is specified for the given software project. In stage two, an analysis of the preliminary set of requirements is performed. In stage three, the SRS is developed. In stage four, the SRS is verified. Lastly, in stage five, requirements are managed. In general, these stages are found in traditional SRE practices however, the use of blockchain as infrastructure is promising, enhancing trust, security, and reliability.

Additionally, the sub-components of each phase are outlined, depicting what type of tools, techniques, inputs, and outputs would be essential for each phase of the process and how every component aids in creating a reliable SRS provided that all the conditions of the blockchain framework are met. For instance, during the consensus of the network, if the majority of the stakeholders do not agree with a given set of requirements then they get rejected. The details of the inputs and outputs of the system are shown in Table 8.

B. PARTICIPATING USERS

An SRE platform built on blockchain infrastructure requires a close association between the users of the network as shown in Fig.23.The type of users for such a platform may include clients, stakeholders, developers, design engineers, requirement analysts, and blockchain engineers. Some additional users may also be incorporated, depending upon the nature of the project. The platform provides a secure, reliable, and transparent infrastructure for the communication of SRS between the users during the SLDC.

The platform facilitates users to elicit, design, discuss, verify, validate, and manage software requirements through mutual consensus of the participating nodes. The finalized set of captured requirements is added to the blockchain network. Mutual trust between the participants of the blockchain network is ensured through a consensus mechanism. In addition, it is also used to verify requirements. Furthermore, blockchain technology utilizes distributed ledger to maintain the synchronized copies of transactions among the participants of the blockchain network ensuring consistency, trust, traceability, and transparency among the participating nodes.

TABLE 7. Summary of reviewed approaches.

Practices	Focus		Pros and Cons		Outcomes		Stakeholder Relevance	
	SRE Focus	Domain Focus	Advantages	Drawbacks	Domain	SRE	Involves Stakeholders?	Involve stakeholders at all phases?
Traditional Approach Based Practices								
EHR	Requirements Elicitation and Specification	Healthcare Systems	Systematic Elicitation of Requirements	Manual Work is Required	EHR Framework Based on Five Layered Architecture is Produced	Requirements are Produced Though User Reviews, Meetings, and Professional Interviews	✓	✓
LEAN STARTUP	Requirements Specification, Validation, and Analysis	Blockchain Supported Systems	Addresses Inherent Problems of SLDC	Not Applicable for Large Businesses	Effective for Initial Startup Businesses Producing Blockchain Systems	Inspection Based Peer Review Employed for Validation	✓	×
DAPPS RE	Requirement Analysis by Examining User Reviews	Decentralize Systems Apps	Practical Implications for Practitioners	Lacks Requirement Management Aspects	Provides Guidelines on the Design of DApps	Produce First Set of User Requirements	✓	×
VACCINE SCM	Requirement Elicitation and Analysis	Vaccine Supply Chain	Provides End to End Visibility	Based on Conventional Techniques	Produces Vaccine SCM Prototypes	Capturing Requirements Based on Vaccine SCM Applications	✓	✓
PROTOTYPE BASED	Requirement Elicitation	E-Voting	Client Oriented Approach	Not Applicable for All Types of Requirements	Yields E-Voting Prototypes	Documenting Telecom Requirements as User Stories	✓	×
Framework Based Practices								
SME	Requirement Analysis using Scenarios	Banking, Automotive, Insurance, Construction	Strong System Technical Basis	Based on Conventional Techniques	Action Research Design Providing Domain Scenarios	Documentation of Blockchain Application Scenarios	✓	×
EDREAM	Requirement Elicitation and Assessment	Smart Grids	Stakeholder Oriented Method	Restricted to eDREAM Project	Provide Aggregators to Operate in Per-Use Model	Produce a Clear Set of Base Requirements	✓	×
CS SLA	Specification of SLA Smart Contracts	Cloud Systems	Use of KAOS to Maintain Security and Reliability	Restricted to Blockchain Cloud Systems	Quality and Efficient Engineering of Blockchain Cloud Systems	Ensure Accuracy, Consistency, Understandability, and Traceability of Requirements	✓	✓
COLAB IOT	Blockchain-Based Requirement Analysis	IoT Systems	Provides a Functional Process for IoT based Blockchain Applications	Restricted to Blockchain IoT Systems	Based on Blockchain Infrastructure	Improve Overall SRE Quality by Producing Reliable IoT Requirements	✓	✓
ATAM	Requirement Analysis Based on	E-Voting	Stakeholder Centric Approach	Manual Work is Essential for Each	Determines the Extent of System Potentialities	System Implementation Guidance and	✓	×

TABLE 7. *(Continued.)* **Summary of reviewed approaches.**

	Industrial Experts			Phase of the Process	to Satisfy Expected Quality Attributes	Decision Making		
INTER-ASOCIATION	Requirement Traceability	Inter-Organization Projects	Utilize Distributed Ledger to Manage Requirements	Proposed Hypothetical Solution	Provides Auditable History of Requirements	To capture traceable requirements	✓	×
BCHAINREQ	Automated Requirements Elicitation, Requirement Analysis	Healthcare	Requirements Based on Trust	Restricted to Healthcare Domain	Captures Requirements Based on Trust	Validate the Quality and Standard of Trust Requirements	✓	✓
GOAL-ORIENTED	Requirement Analysis	Food SCM	Goal-Oriented Approach	Does Not Allow Seamless Transition to Design Phase	Assists in the Selection of Better Alternatives, Reasoning about Business Goals, and Resolving Conflicts	System Design Guidance	✓	×
I* AND UML BASED	Requirement Analysis	Food SCM	Goal-Driven Approach	i* Model Lack Some Elements and Imposes Various Restrictions	i* Helps in Understanding the 'Why' of Business Processes While UML focus on 'What'	System Design Guidance	✓	×
CARENGCHAIN NET	Requirement Elicitation	Automotive	Assist in Vehicle Engineering	Restricted to Automotive Domain	Assist in Version Control of the Vehicles During Lifecycle	Produce New Transmission Chain for Vehicle Software Artifacts	✓	×
SE Model Based Methods								
SS and XP	Requirement Elicitation	Generic	Forms a Base Target Blockchain Without Dependency	Lacks in Providing a Detailed SRS	Recognizes Trade-offs and Uncertainties	Useful in Gathering Initial Requirements	✓	×
TOE	Requirement Elicitation	Generic	Utilizes Blockchain Principles to Produce Applications	Lesser Requirement Documentation Mechanisms	Provides a Transparent and Secure Software Development Process	Addresses Non-functional requirements and Integrity of Large-Scale Projects	✓	×
ABCDE	Requirement Elicitation	Generic DApps	Produces Blockchain Application Based on Ethereum and Solidity	Not Tested on Other DApp Development Environments	Explicit Activities to Design DApps Using Formal Diagrams	Modeling Based on UML Use Case and Sequence Diagrams using Stereotypes	✓	×
ADP	Requirements Elicitation	Generic	Infrastructure Originator Producing Results into Architectural Stubs	It is Difficult to Determine Blockchain Parameters Using Wide Intervals	Decision Process Helps in Generating Architectural Suggestions	Translates All Inputs and Architectural Patterns Into a Set of Numeric Values	✓	×

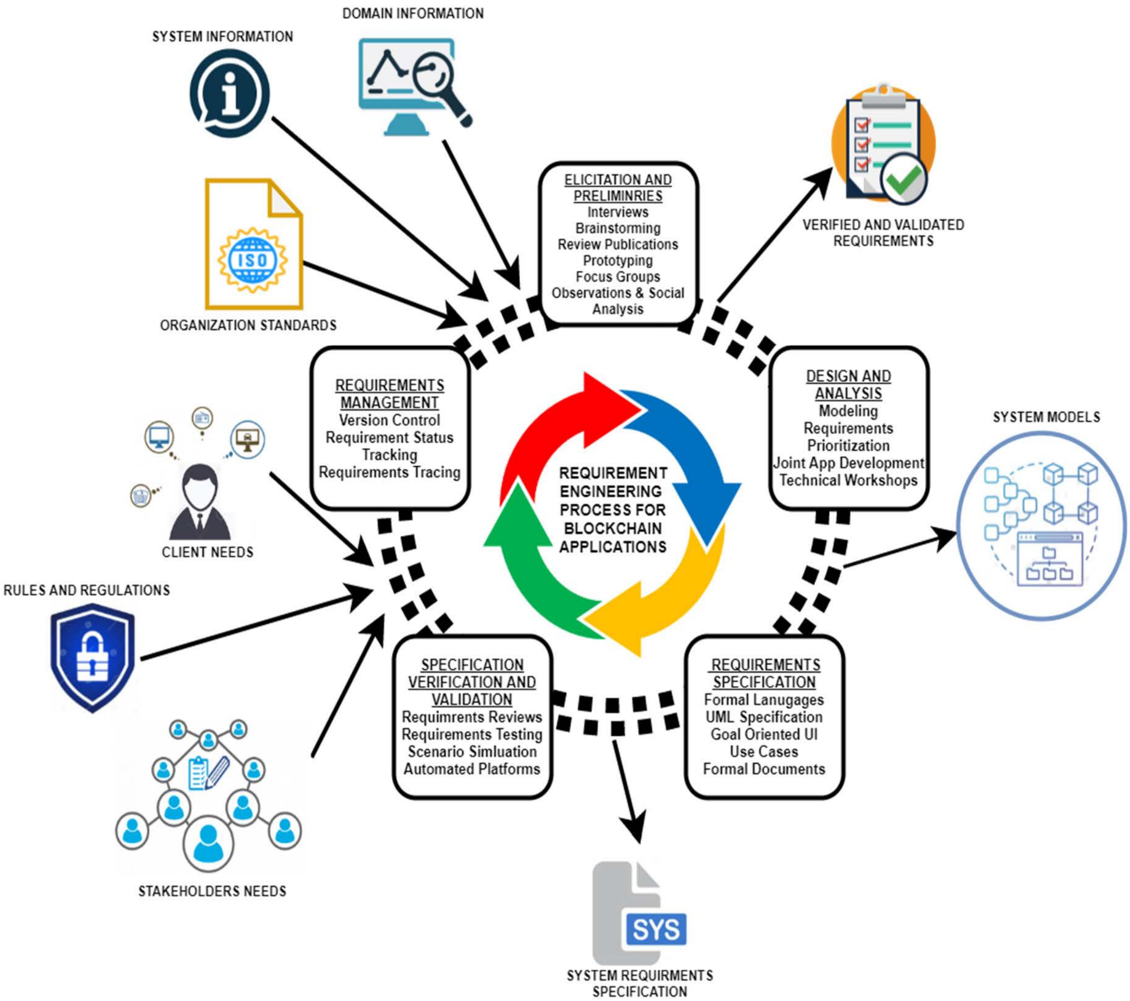

FIGURE 22. Proposed High-Level Architecture SRE Model Built on Blockchain Technology.

TABLE 8. Inputs and outputs of the phases for the proposed high-level architecture of blockchain-driven SRE model.

Phase	Input	Output
Definition of Scope and Groundwork	Domain Information System Information Organizational Standards	Verified and Validated Requirements
Defining and Examining the System	Verified and Validated Requirements	System Models
SRS Development	System Models	SRS
SRS Verification	SRS	Stakeholder Needs Client Needs Rules and Regulations
Requirements Management	Verified SRS	Requirement Tracking Version Control

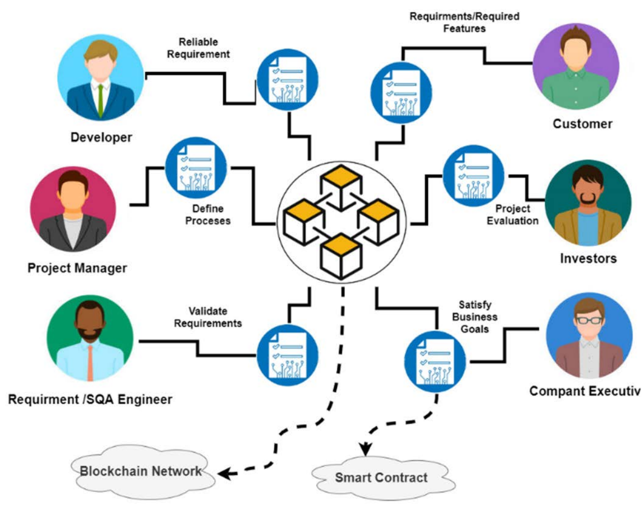

FIGURE 23. Participants of the Proposed Blockchain-Driven SRE Model.

The proposed platform addresses the communication gap between the stakeholders, transparency and traceability issues, and the problems related to requirement ambiguities. In addition, stakeholders generally lack knowledge about domain and SRE practices. The proposed SRE model facilitates educating the clients and stakeholders, facilitating them to easily describe their needs. Furthermore, the poor management of functional requirements and secure nonfunctional requirements is a key issue that makes traceability nearly impossible, especially after the deployment of the developed software system. The proposed SRE platform main-

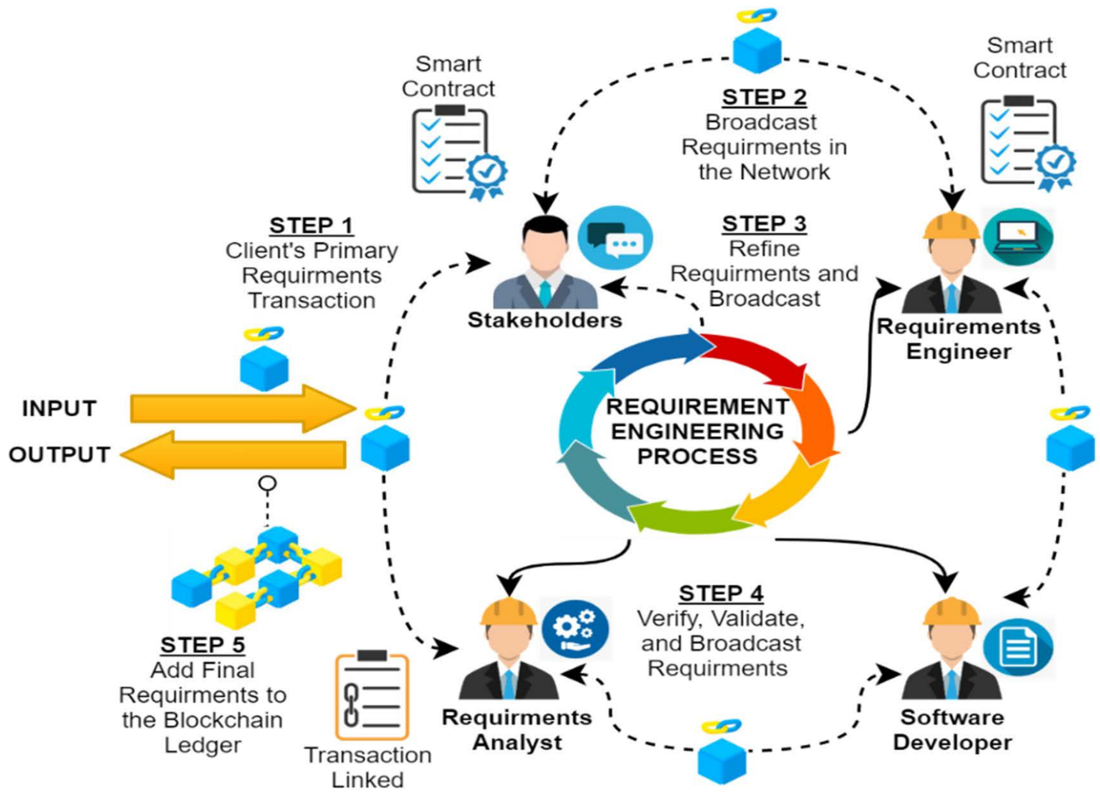

FIGURE 24. Process Steps for the Proposed Blockchain-Driven SRE Model.

tains traceability through the use of distributed ledger on the chain of records in an automated way. Similarly, the conflicts among stakeholders in SRE practices are another key issue that mostly occurs due to variability in SRE approaches. Nevertheless, in the proposed platform, participating users follow the generic approach within the network. Considering the advantages of the proposed platform, it has the potential to improve the domain of SRE practices by enhancing security, trust, traceability, efficiency, quality, adaptability, transparency, effectiveness, and efficiency.

C. PROCESSING PHASES

The detailed processing steps further provide insights into how accurate, reliable, traceable, and consistent requirements are captured.

The processing phases along with participants of the blockchain network are shown in Fig.24. Each user has specific roles and responsibilities toward the development of SRS. Currently, we considered four different participants. When a client specifies primary requirements to the requirement engineer, they are typically reverted in the form of SRS, illustrating to the client how the system operates. The overall process consists of five stages which are as follows: Elicitation and Preliminaries; Design and Analysis; Requirements Specification; Specification Verification and Validation; and Requirements Management. A detailed view of the process steps is discussed in the next sub-sections.

1) ELICITATION AND PRELIMINARIES

In this stage, the context, problems, preliminaries, and parameters of the project are defined. Also, the potential stakeholders and their needs are identified. Thus, the feasibility of the system is examined based on the application domain problems and needs of the stakeholders. Once the construction of the system is feasible, the application domain problems are defined in detail. In addition, the needs of the stakeholders are transformed into stakeholder requirements by identifying functions, constraints, and quality characteristics. Furthermore, the requirements of the stakeholders are evaluated to check whether they are affordable, feasible, complete, and verifiable. Finally, a clear agreement is taken from the stakeholders, which proves that a common understanding of the system has been achieved.

2) DESIGN AND ANALYSIS

In this stage, system scenarios, activities, and components are identified based on stakeholders' requirements and the defined problem, suggesting solutions. The Requirements engineer then refines the requirements and broadcasts them into the blockchain network. The broadcasted requirements are transparent to each participating node of the blockchain network.

3) REQUIREMENTS SPECIFICATION

In this stage, the requirements engineer specifies system scenarios, activities, and components. The solutions are suggested based on the defined requirements and stakeholders' problems. These solutions are elaborated as specifications of domain and software components including interface, functional, and nonfunctional requirements. These specifications are also referred to as the comprehensive set of domain requirements which are again broadcasted into the blockchain network for evaluation.

4) SPECIFICATION VERIFICATION AND VALIDATION
In this stage, the requirement analyst evaluates the specifications of system device components, software components, and use cases to verify that the specifications are correct, traceable, unambiguous, concise, possible to implement, feasible, and possible to test, complete, non-redundant, consistent and understandable.

5) REQUIREMENTS MANAGEMENT
In this stage, the requirements management takes place. The requirements after being verified and validated by the users through mutual consensus are added to the distributed ledger of the blockchain network in form of SRS. Moreover, each node gets an updated copy of the ledger.

D. WORKFLOW BASED ON PROCESSING PHASES
The workflow model based on the previously discussed processing stages is shown in Fig.25.

Initially, the client and stakeholder work on the scope and preliminaries to collect the domain information and generate initial requirements transactions for the underlying system. Secondly, the client broadcasts the initial requirement transaction in the network. Thirdly, the requirements engineer refines the requirements and broadcasts them again into the network. Fourthly, the decision box checks the verification of the requirements. If they are verified, then it leads to another decision box, which checks the validation of the requirements. In any case, if the requirements are not verified or validated, the requirements analyst performs verification or validation on the requirements and broadcasts them into the network. Nevertheless, if the requirements are verified and validated, then the system analyst adds the final requirements to the distributed ledger of the blockchain network, and the flow gets finished.

E. ADVANTAGES OF THE MODEL
The initial requirements of the clients are incomplete, ambiguous, inconsistent, and redundant. At first, the customer transmits initial requirements in the network consisting of a requirement analyst/engineer, blockchain engineer, and design specialist. Each participant in the network can view every requirement transaction and its processing. Also, verify transaction validity and transmit them using mining algorithms such as PoW or PoA in which miners contest among one another and validate the refined requirements on the blockchain network, and get rewarded. The first mine gets a financial incentive from the customer. Finally, the miner broadcasts the requirements into the distributed ledger. The

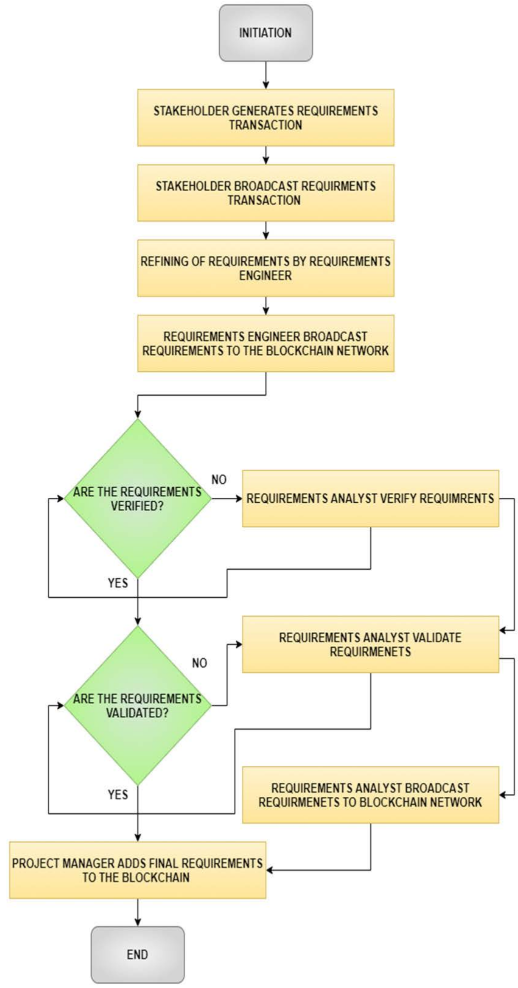

FIGURE 25. Workflow of Proposed Blockchain-Driven SRE Model.

proposed SRE model addresses several problems related to conventional SRE practices as shown in Fig.26.

VI. GUIDELINES TO IMPLEMENT GENERIC SRE MODEL DRIVEN BY BLOCKCHAIN TECHNOLOGY
This section provides the implementation guideline for the proposed SRE model. In particular, the overview of recommended tools and technologies has been presented.

A. BLOCK METRICS
The block structure for the proposed platform is shown in Fig.28. The presented block parameters are based on the characteristics of contemporary SRE practices employed in software companies. There are four possible major elements of the block namely Block ID, Header, Transaction Details, and Lock Time or Timestamp. The transaction elements further encapsulate 18 elements. TID is the transaction ID of the block, RID is the requirement ID of the block, RN is the name of the requirement, RT is the type of requirement being specified, RIN is the person in charge of the requirement specification, RDS is the description of the requirement, RVR is the version of the requirement, RAU is the author of the

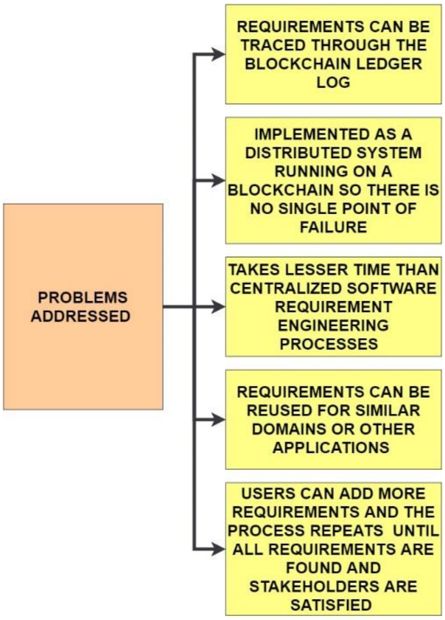

FIGURE 26. Problems Addressed by the Proposed Blockchain-Driven SRE Model.

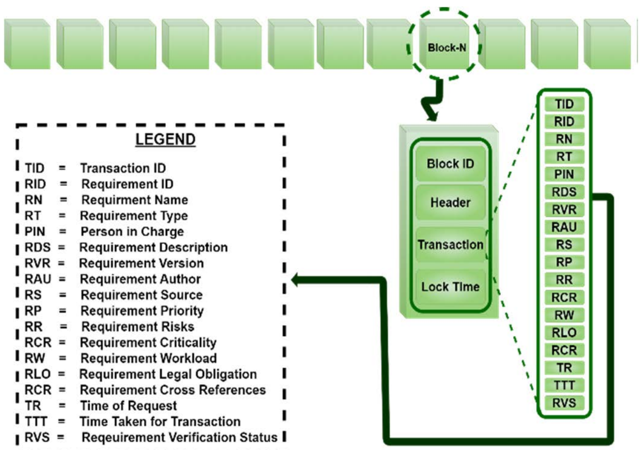

FIGURE 27. Possible Block Metrics to Implement Proposed Blockchain-Driven SRE Model.

requirement, RS is the source of the requirement, RP is the priority of the requirement which can be low, medium or high, RR stands for the risks related to the requirement being specified, RCR is the criticality of the requirement, RW is the workload required to process the requirement, RLO is the legal obligations associated with the requirement, RCR is the cross-references for the requirement, TR is the time of request for the requirement, TTT is the total time taken for the transaction, and RVs is the verification status for the given requirement.

B. LAYERED ARCHITECTURE

The layered architecture of the proposed model consists of eight levels as shown in Fig.27.

The in-depth information about every layer is in the next sub-sections.

FIGURE 28. Layered Architecture for the Proposed Blockchain-Driven SRE Model.

1) INTERFACE LAYER

This layer consists of high-level user-end applications such as Websites and DApps of the RMS. The primary objective of this layer is to provide the user-level interface to the clients, development team, the software company, and other stakeholders of a software project. Every user has to interact with the interface layer to start the SRE process.

2) APPLICATION LAYER

This layer encapsulates the digital records such as user profiles, metadata of transactions, requirement records, project information, and software company transactions. Moreover, this layer links the business logic in the form of a smart contract with the interface layers.

3) BUSINESS LOGIC LAYER

This layer comprises smart contracts and deals with the terms such as interaction criteria, regulations, and role scenarios. The business logic layer is taken as an active database of smart contracts including all the rules of contract invocation, execution, and communication.

4) TRUST LAYER

This layer encompasses security and trust details of formal verifications, consensus algorithms, and smart contracts. In addition, this layer deals with all sorts of consensus protocols for transactions, newly added block verification, and results of executions stored in the blockchain layer.

5) BLOCKCHAIN LAYER

This layer encapsulates the data or information regarding the status of blocks and distrusted nodes. It also stores the basic information of the distributed ledged and hashes of the transactions made by clients, development teams, and

TABLE 9. Blockchain technologies supporting proposed blockchain SRE system modeling.

Technology	Type
Blockchain Type	Public, Private, Hybrid
Consensus Algorithm	PoW, PoA, PoS
Platform	Ethereum, HyperLedger, Ganache
Cryptographic Security	SHA256
DApps	Geth, Ethereum, Solc, Truffle
Smart Contract	Solidity, Ethereum

software companies against their private and public key addresses.

6) TRANSACTION LAYER

This layer deals with the transactions that are triggered by the users or smart contracts of the RMS.

7) INFRASTRUCTURE LAYER

This layer comprises a P2P network to forward the transaction mapped on the Ethereum blockchain after its distribution and verification. In addition, it deals with distributed networking, communication, and verification mechanisms. Whenever a transaction is executed, it is broadcasted over the blockchain network. Every node verifies the transaction by the predefined parameters and finally, the transaction gets stored in the distributed ledger of the blockchain network.

8) ADMINISTRATION AND SECURITY LAYER

This is the most vital layer of the blockchain network. It protects the infrastructure of the blockchain. This layer stays connected in parallel with the blockchain system and exhibits various security algorithms and protocols to maintain system security. It also includes the roles of administrative users to sustain the integrity of the blockchain network. Every user such as client, development team, project manager, and software companies having online e-wallets can use the system through DApps or Web Portals providing user-level interfaces for RMS. Once the policies or rules are defined by the smart contracts, the trust layer incorporates consensus algorithms to sustain the credibility of the blockchain network. Moreover, recommend the Ethereum blockchain as the baseline technology with the PoW implementation to ensure the temporal property of the distributed network. Furthermore, the administration and security layers are integrated with all the other layers of the system to ensure authorized monitoring by the software company or regulatory bodies. Blockchain-based supporting technologies which are generally available are shown in Table 9.

The use of a private blockchain is recommended along with the use of PoS or PoW. It is also recommended to use HyperLedger as it is more famous than other technologies. Furthermore, the SHA256 hashing algorithm is recommended to be utilized along with development on either Truffle or Ethereum for DApps. Finally, we recommend developing smart contracts on Solidity based on Ethereum.

C. SMART CONTRACT

A smart contract is similar to a computer program that incorporates predefined instructions and acts like a finite state machine whenever a user is linked with a blockchain network [109]. These contracts enable permissions and restrict violations of data integrity. A smart contract once deployed in any blockchain network can never be tempered. It is similar to rules or policies which are once defined cannot be modified. In the context of the proposed SRE platform, it is recommended to employ a solidity compiler for the construction of smart contracts. Formulation of smart contracts based on requirement information transfer from client to requirement engineer, rewards mechanism, consensus mechanism, and DApps integration is essential.

1) SMART CONTRACT DEPLOYMENT

Tools such as Remix Compiler or Solidity are recommended to be employed to deploy smart contracts for the blockchain SRE model through byte code development [110].

2) POSSIBLE VULNERABILITIES OF SMART CONTRACT

There are various vulnerabilities of smart contracts. For instance, a small issue in coding can cause some significant problems in terms of information loss, money loss, and privacy leakage, hence high security is vital for the deployment of smart contracts [111]. The peer nodes of the blockchain network can also exploit bugs in smart contracts to gain profit or advantage from the system. Hence, it is necessary to eliminate the risk of vulnerabilities using efficient quality assurance standards.

D. SYSTEM SCENARIOS

A goal-oriented scenario has been presented in Fig.29. In the given goal-oriented scenario, we consider requirement analysts, project managers, and developers as primary actors. In this scenario, the goal of the requirements engineer is to produce reliable requirements using soft or hard goals. Soft goals are related to the negotiation process and hard goals are linked with the essential requirements. Essential requirements are necessities which are basic functions that are mandatory for the execution, implementation, and running of a software application.

Similarly, in the given workflow scenario, the requirement engineer is responsible to add transactions while the project manager can keep the requirements regulated following the standards of the software company. The developer must endorse the transactions and add transactions to the ledger which previously got processed by the client, project manager, and requirements engineer. Notably, in this workflow scenario, we did not use the client as an actor because we suppose that the requirements are already entered into the blockchain system by the client using mobile or web API. In addition, at any time, the clients can see the progress of the requirements using their API to check how, when, and who processed the requirements. The use of techniques like

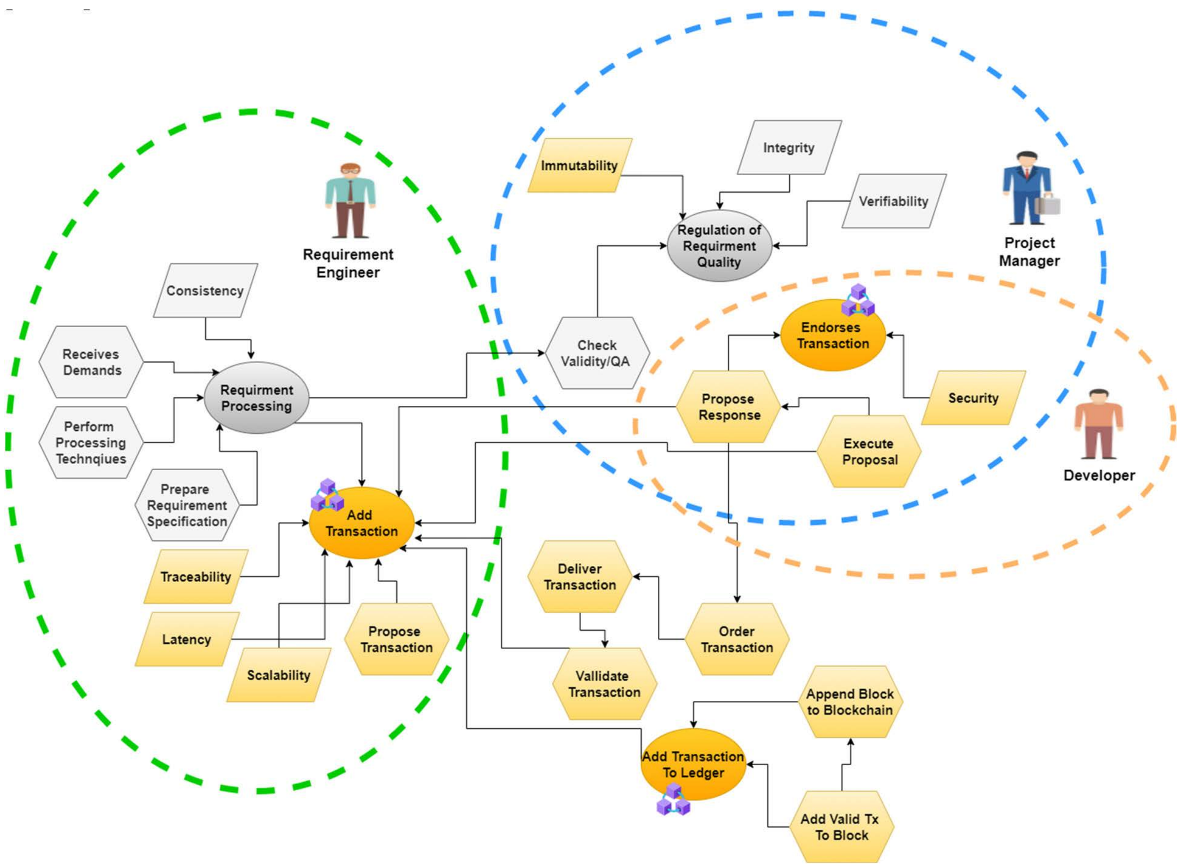

FIGURE 29. Scenario-Based SRE Processing Workflow Based on Goals for Proposed Blockchain-Driven SRE Model.

use case mapping and goal-oriented SRE supported prioritizing requirements, choosing alternatives, envisioning the scenarios, finding completeness of requirements, and resolving conflicts of the stakeholders.

The use case scenario for the given workflow is shown in Fig.30. The client requests to add the requirement transactions and other parties endorse which transaction should be verified and vice versa. If the transaction is not verified, it is still added to the ledger however the status of the blockchain network does not change. In this paper, only one scenario has been presented, however different alternative scenarios can also be modeled and implemented. The proposed scenario enables the practitioners to visualize scenarios, goals, and soft goals to devise reliable requirements satisfying all stakeholders. Moreover, the employment of a use case diagram for the scenario reveals that the system can trace requirements and help to produce reliable requirements thereby eliminating traceability issues.

The overall comparative results of the approaches are shown in Table 10. There are three primary sections of the table. The first factor includes the SRE phase coverage. The second factor incorporates domain coverage. Finally, the third factor encapsulates the quality factors coverage. We can see that the previously proposed SRE practices exhibit quality gaps. Some studies covered all the phases of the SRE however they were domain confined and lacked quality factors coverage. While some studies did not cover all the phases of SRE.

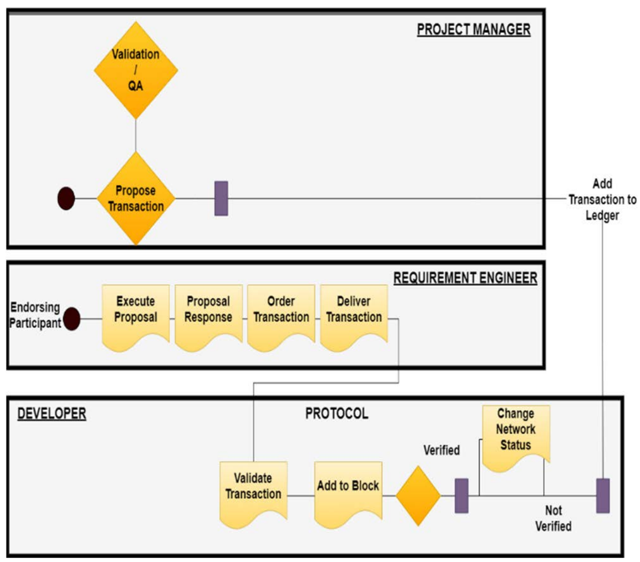

FIGURE 30. Use Case for the Proposed Blockchain-Driven SRE Model Scenario.

In short, all the previous studies lacked in covering all the points. Nevertheless, the SRE model proposed in this research article covers all the key factors and shows a promising solution for the future.

The characteristics based on features being offered by the proposed model are shown in Fig.31.

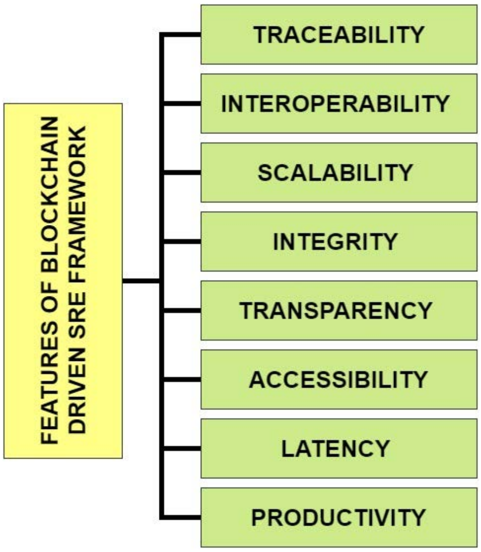

FIGURE 31. Characteristics based on features of the Proposed Blockchain-Driven SRE Model.

Notably, the proposed approach covers all the discussed aspects which shows a novel direction in the blockchain SRE domain. The portrayed characteristics are based on blockchain technology when incorporated are SRE practices which show significant effectiveness in enhancing the overall process reliability.

The current state-of-art SRE acquainted practices (frameworks, approaches, models, and tools) lack in various aspects. For instance, most of the practices do not use blockchain as an infrastructure for the SRE environment. While some practices lack in covering each aspect of the SRE method i.e. Elicitation, Analysis, Validation, Management, Specification. Moreover, most of the reviewed practices are domain-constrained and cannot be employed for other domains. Furthermore, most of the practices do not cover the problematic factors of Requirements Negotiation, Requirements Validation, and Requirements Traceability as identified in the research. In addition, manual work is required in most of the observed SRE practices. Nevertheless, the proposed blockchain-driven SRE model covers all the gaps found previously in observed practices. Early career academia and industry practitioners may use this model to develop new blockchain-driven SRE frameworks or enhance current state-of-art practices.

VII. RESEARCH CHALLENGES AND FUTURE DIRECTIONS

In this section, we present the discussion on research challenges and future directions of blockchain technology in the context of the SRE domain.

A. RESEARCH CHALLENGES

1) DIVERSIFIED ORGANIZATIONAL STANDARDS

All over the globe, the use of software development practices varies from company to company due to diversified standards and domains. A single blockchain SRE platform may not be suitable for every company. The implementation of the proposed generic SRE model is challenging for the practitioners, requiring them to have a firm understanding of the organizational goals, capacities, and standards so that they can carefully select the most appropriate blockchain type, consensus mechanism, and smart contracts to devise a flexible SRE platform that can be applied in any domain.

2) THE GAP OF REQUIREMENT ANALYSIS

The requirement analysis for blockchain applications does not significantly differ from centralized systems. Many software firms still employ traditional SRE practices for blockchain engineering as suggested by [112]–[114]. In the context of the process aspect, the discussion on requirements analysis has received much less attention from scholars. Previous studies overlooked how requirements contribute value to blockchain stakeholders leaving it as an unexplored area. It is challenging to understand which requirements constitute value to stakeholders exhibiting diverse goals and commitment levels if they are addressed by smart contracts. This is consistent with the case study findings of [115] who accentuated the complexity of blockchain-based applications for stakeholders by contending that they primarily care about the application being useful rather than what the underlying technology offers. The definition of smart contracts and solution architecture depends on the elicitation of true requirements as well as legal agreements which requires a tighter connection between the analysis and design phase to endure smart contract errors.

3) NEED FOR STAKEHOLDER ORIENTED APPROACHES

Blockchain application designing presents more kinds of stakeholders as opposed to other domains, ranging from core blockchain engineers to legitimate IT experts, who might play various parts inside the improvement lifecycle and be keen on being impacted by the lifecycle. Considering the common rules for blockchain stakeholders, a group-based viewpoint for application designing keeps on advancing in SE with an accentuation on jobs and distributed groups/development. The SRE practices based on stakeholders' orientation are insignificant and conventional techniques inhibit the communication gap. Hence, there is a dire need to design the SRE practices based on stakeholders.

4) LACK OF EMPIRICAL STUDIES

It is challenging for SRE practitioners to get a better understanding of the criteria and tools for selecting the requirements when fulfilled by blockchain applications to create added value for the stakeholders due to the lack of empirical studies in this domain. The studies published in blockchain SRE are evident, however, it is still in infancy. Moreover, studies are required in this domain to effectively understand the core elements and components residing within this domain.

B. FUTURE DIRECTIONS

The future directions of this research are listed below:

TABLE 10. Overall summary of the reviewed practices and comparison with proposed model.

Practices	SRE Phases Coverage					Domain Coverage		Quality Factors Coverage			Use Blockchain as Infrastructure?
	Elicitation	Analysis	Specification	Verification	Management	Cover all Domain in Use Cases?	Extensible for other Domains?	Cover RN factors?	Cover RT factors?	Cover RV factors?	
Approaches											
EHR	✓	✓	✓	✓	×	✓	×	✓	✓	✓	×
LEAN STARTUP	✓	×	×	✓	×	×	×	×	×	×	×
DAPPS RE	✓	✓	✓	×	×	×	×	✓	✓	✓	×
VACCINE SCM	✓	✓	✓	×	×	✓	×	✓	×	✓	×
PROTOTYPE BASED	✓	✓	✓	×	×	×	×	×	×	×	×
Frameworks											
SME	✓	✓	✓	✓	×	✓	×	✓	×	✓	×
EDREAM	✓	✓	✓	✓	✓	×	×	✓	×	✓	×
CS SLA	✓	✓	✓	✓	✓	✓	×	✓	×	✓	×
COLAB IOT	✓	✓	✓	✓	✓	✓	×	✓	✓	✓	✓
ATAM	✓	✓	✓	✓	✓	✓	×	✓	×	✓	
INTER-ASOCIATION	✓	×	✓	✓	✓	✓	×	✓	✓	✓	✓
BCHAINREQ	✓	✓	✓	×	×	✓	×	×	×	×	×
GOAL-ORIENTED	✓	✓	✓	✓	×	×	×	×	×	×	×
I AND UML BASED	✓	✓	✓	×	✓	✓	×	×	×	×	×
CARENGCHAIN NET	✓	×	×	×	✓	×	×	×	✓	×	×
SE Based Models											
SS and XP	✓	×	×	×	×	✓	✓	✓	×	✓	×
TOE	✓	×	×	×	×	✓	✓	✓	×	✓	×
ABCDE	✓	×	✓	×	×	✓	✓	✓	✓	✓	×
ADP	✓	×	×	×	×	✓	✓	×	×	×	×
Proposed Model	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

1. Further studies can conduct precise examinations across different companies to recognize when stakeholders are involved, what jobs are allotted to them, and how their impact is affected throughout the investigation, plan, execution, and activity of the SRE framework built on blockchain infrastructure. Such examinations can add value to the distinguishing blockchain practices concerning integration and project estimations.
2. In terms of the modeling aspect, several models are presented ranging from requirements to smart contract design elements, and down to smart contract code fragments. Managing the chain of these models as a means to understand logical traceability and bounded dependency plays a significant role to enable automated blockchain system design given requirements. Techniques related to the RT are partially captured in this literature which stimulates another possible future research.
3. Bearing in mind that blockchain development involves more types of technical and non-technical stakeholders than many conventional software systems, with diverse goals, different levels of engagement, incentives, stewardship, and cooperation, further studies can conduct technical analysis of the blockchain SRE framework.

VIII. CONCLUSION

In this article, we have employed a survey-based approach to understanding the current developments in blockchain-acquainted SRE practices. We have provided an in-depth comprehension of blockchain-acquainted SRE practices in two primary aspects. The first aspect comprises SRE practices employed during blockchain engineering. While the second aspect encapsulates the concept of the SRE framework built on blockchain infrastructure. We have highlighted the significance of blockchain acquainted SRE by outlining the problematic aspects of contemporary SRE practices by classifying and defining each problem factor. Major SRE problems are associated with RN, RV, and RT. Consequently, we have performed a feature analysis to illustrate how blockchain technology satisfies the necessities of an SRE framework. Furthermore, we have classified the SRE practices based on blockchain development. We have identified that each framework and approach is domain-oriented or formalized or specific scenarios, making them inflexible. Accordingly, we have proposed a generic SRE model built on blockchain infrastructure along with its workflows and scenarios. We have briefly presented the phases of the proposed generic model and outlined its effectiveness in the automation of SRE practices and coverage of the problematic quality improvement factors. In addition, we have outlined the implementation guidelines for the proposed generic SRE model driven by blockchain infrastructure. Finally, we have presented research challenges and future directions.

In summary, this article made contributions to the existing literature on blockchain acquainted SRE in three aspects. Firstly, the article presents the taxonomy of problematic quality factors of SRE practices, laying a beneficial foundation to drive the need for blockchain infrastructure in this domain. Secondly, the taxonomy of blockchain acquainted SRE practices outlines the need for a more generic approach. Thirdly, a generic SRE model built on blockchain infrastructure has been proposed. The proposed model provides new insights and opportunities for scholars and software developers to rethink and reexamine the existing SRE practices and consider the use of blockchain infrastructure. The presented workflows depict the significance of employing the model for SRE practices. In the future, this research can be extended by implementing a functional SRE framework built on blockchain infrastructure based on technologies like Ethereum or HyperLegder.

LIMITATIONS OF THE RESEARCH

There are two limitations of this article. Firstly, there might still be more quality attributes or challenges present in SRE practices. Future studies can extend the review to explore and discuss case studies of software companies in the context of blockchain acquainted SRE. Secondly, the domain of blockchain acquainted SRE is still in infancy, we only reviewed research publications based on journals and conferences and did not perform a market review to identify market trends. Further studies can conduct a more comprehensive study in combination with market development.

ACKNOWLEDGMENT

Mishaal Ahmed would like to thank the Almighty Allah who gave him the strength to carry out this research and for His countless blessings. Then, he is especially thankful to Dr. Muhammad Shoaib Farooq for supporting him throughout his research as his teacher and mentor. Also, he is thankful to his friends and colleagues for their moral support. Finally, he is thankful to his parents for all the support.

REFERENCES

[1] N. T. More, B. S. Sapre, and P. M. Chawan, "An insight into the importance of requirements engineering," *Int. J. Internet Comput.*, vol. 1, no. 2, pp. 34–36, 2011.

[2] U. Rani and R. Dhir, "Platform work and the COVID-19 pandemic," *Indian J. Labour Econ.*, vol. 63, no. 1, pp. 163–171, 2020.

[3] K. Villela, A. Hess, M. Koch, R. Falcao, E. C. Groen, J. Dörr, C. N. Valero, and A. Ebert, "Towards ubiquitous RE: A perspective on requirements engineering in the era of digital transformation," in *Proc. IEEE 26th Int. Requirements Eng. Conf. (RE)*, Aug. 2018, pp. 205–216.

[4] A. Van Lamsweerde, "Goal-oriented requirements engineering: A guided tour," in *Proc. 5th IEEE Int. Symp. Requirements Eng.*, London, U.K., Aug. 2001, pp. 249–262.

[5] S. El Ghazi El Houssaïni, I. Maskani, and J. Boutahar, "A security requirement engineering case study: Challenges and lessons learned," in *Intelligent Computing*, vol. 1, no. 1. Cham, Switzerland: Springer, 2021, pp. 761–783.

[6] G. George, K. Lakhani, and P. Puranam, "What has changed? The impact of COVID pandemic on the technology and innovation management research agenda," *J. Manage. Stud.*, vol. 1, no. 2, pp. 17–54, 2020.

[7] P. Neto, U. Mannan, E. de Almeida, N. Nagappan, D. Lo, P. Kochhar, C. Gao, and I. Ahmed, "A deep dive on the impact of COVID-19 in software development," 2020, pp. 1–23, vol. 70, no. 48, *arXiv:2008.07048*.

[8] L. S. Machado, C. Caldeira, M. G. Perin, and C. R. B. de Souza, "Gendered experiences of software engineers during the COVID-19 crisis," *IEEE Softw.*, vol. 38, no. 2, pp. 38–44, Mar. 2021.

[9] M. Garcia-Valls, P. Bellavista, and A. Gokhale, "Reliable software technologies and communication middleware: A perspective and evolution directions for cyber-physical systems, mobility, and cloud computing," *Future Gener. Comput. Syst.*, vol. 71, pp. 171–176, Jun. 2017.

[10] S. Abdullahi, M. Zayyad, N. Yusuf, L. Bagiwa, A. Nura, and Z. A. B. Dansambo, "Software requirements negotiation: A review on challenges," *Int. J. Innov. Comput.*, vol. 11, no. 1, pp. 1–6, 2021.

[11] G. Kumar and P. K. Bhatia, "Comparative analysis of software engineering models from traditional to modern methodologies," in *Proc. 4th Int. Conf. Adv. Comput. Commun. Technol.*, Chicago, IL, USA, Feb. 2014, pp. 189–196.

[12] H. H. Altarturi, K.-Y. Ng, M. I. H. Ninggal, A. S. A. Nazri, and A. A. A. Ghani, "A requirement engineering model for big data software," in *Proc. IEEE Conf. Big Data Anal. (ICBDA)*, Chicago, IL, USA, Nov. 2017, pp. 111–117.

[13] D. Beyer, S. Löwe, and P. Wendler, "Correction to: Reliable benchmarking: Requirements and solutions," *Int. J. Softw. Tools Technol. Transf.*, vol. 21, no. 1, pp. 1–29, 2019.

[14] P. Laplante, *Requirements Engineering for Software and Systems*. Boca Raton, FL, USA: Auerbach Publications, 2017.

[15] M. Yaseen and Z. Ali, "Success factors during requirements implementation in global software development: A systematic literature review," *Int. J. Comput. Sci. Softw. Eng.*, vol. 8, no. 3, pp. 56–68, 2019.

[16] O. Okesola, K. Okokpujie, P. Oyom, K. Olamide, A. Oludele, and K. Afolashade, "Structuring challenges in requirement engineering techniques," in *Proc. World Congr. Eng. (WCE)*, London, U.K., vol. 1, Jul. 2018, pp. 197–200.

[17] S. Bhardwaj and M. Kaushik, "Blockchain—Technology to drive the future," in *Smart Computing and Informatics*, vol. 78, no. 1. Singapore: Springer, 2018, pp. 263–271.

[18] S. Makridakis and K. Christodoulou, "Blockchain: Current challenges and future prospects/applications," *Future Internet*, vol. 11, no. 12, pp. 258–269, 2019.

[19] S. Demi, "Blockchain-oriented requirements engineering: A framework," in *Proc. IEEE 28th Int. Requirements Eng. Conf. (RE)*, Zurich, Switzerland, Aug. 2020, pp. 428–433.

[20] M. Fahmideh, J. Grundy, A. Ahmed, J. Shen, J. Yan, D. Mougouei, P. Wang, A. Ghose, A. Gunawardana, U. Aickelin, and B. Abedin, "Software engineering for blockchain based software systems: Foundations, survey, and future directions," 2021, *arXiv:2105.01881*.

[21] B. Shahzad, I. Javed, A. Shaikh, A. Sulaiman, A. Abro, and M. A. Memon, "Reliable requirements engineering practices for COVID-19 using blockchain," *Sustainability*, vol. 13, no. 12, p. 67485, 2021.

[22] T. Galli, F. Chiclana, and F. Siewe, "Software product quality models, developments, trends, and evaluation," *Social Netw. Comput. Sci.*, vol. 1, no. 3, pp. 1–24, May 2020.

[23] R. Baig, W. A. Khan, I. U. Haq, and I. M. Khan, "Agent-based SLA negotiation protocol for cloud computing," in *Proc. Int. Conf. Cloud Comput. Res. Innov. (ICCCRI)*, Singapore, Apr. 2017, pp. 33–37.
[24] N. Seyff, S. Betz, L. Duboc, C. Venters, C. Becker, R. Chitchyan, B. Penzenstadler, and M. Nöbauer, "Tailoring requirements negotiation to sustainability," in *Proc. IEEE 26th Int. Requirements Eng. Conf. (RE)*, Banff, AB, Canada, Aug. 2018, pp. 304–314.
[25] L. Martins and T. Gorschek, "Requirements engineering for safety-critical systems: Overview and challenges," *IEEE Softw.*, vol. 34, no. 4, pp. 49–57, Jul. 2017.
[26] X. Tao and Y. Miao, "Interest based learning activity negotiation," in *Proc. Int. Conf. Cyberworlds*, Hangzhou, China, Sep. 2008, pp. 58–64.
[27] F. Calefato, F. Lanubile, D. Romita, R. Prikladnicki, and J. H. S. Pinto, "Mobile speech translation for multilingual requirements meetings: A preliminary study," in *Proc. IEEE 9th Int. Conf. Global Softw. Eng.*, Shanghai, China, Aug. 2014, pp. 145–152.
[28] S. Porru, A. Pinna, M. Marchesi, and R. Tonelli, "Blockchain-oriented software engineering: Challenges and new directions," in *Proc. IEEE/ACM 39th Int. Conf. Softw. Eng. Companion (ICSE-C)*, May 2017, pp. 169–171.
[29] R. Norvill, B. Fiz, R. State, and A. Cullen, "Standardising smart contracts: Automatically inferring ERC standards," in *Proc. IEEE Int. Conf. Blockchain Cryptocurrency (ICBC)*, May 2019, pp. 192–195.
[30] H. Treiblmaier and C. Sillaber, "The impact of blockchain on e-commerce: A framework for salient research topics," *Electron. Commerce Res. Appl.*, vol. 48, Jul. 2021, Art. no. 101054.
[31] V. Pekar, M. Felderer, and R. Breu, "Requirements engineering: A systematic mapping study in agile software development," in *Proc. 9th Int. Conf. Quality Inf. Commun. Technol.*, Chicago, IL, USA, 2014, pp. 242–245.
[32] P. Baszuro and J. Swacha, "Requirement engineering as a software development process," in *Data-Centric Business and Applications: Towards Software Development*. Cham, Switzerland: Springer, 2020, pp. 21–39.
[33] M. Mehmood and I. Bb, "A review of requirement engineering process models," *J. Architectural Eng. Technol.*, vol. 7, no. 1, pp. 1–9, 2018.
[34] S. Maalem and N. Zarour, "Challenge of validation in requirements engineering," *J. Innov. Digit. Ecosyst.*, vol. 3, no. 1, pp. 15–21, Jun. 2016.
[35] S. Alla, *Role of Requirements Engineering in Software Project's Success*. Norfolk, VA, USA: Old Dominion Univ., 2017.
[36] S. Mughal, A. Abbas, N. Ahmad, and S. U. Khan, "A social network based process to minimize in-group biasedness during requirement engineering," *IEEE Access*, vol. 6, pp. 66870–66885, 2018.
[37] N. Kukreja, "Winbook: A social networking based framework for collaborative requirements elicitation and WinWin negotiations," in *Proc. 34th Int. Conf. Softw. Eng. (ICSE)*, Zurich, Switzerland, Jun. 2012, pp. 1610–1612.
[38] H. Yang and P. Liang, "Reasoning about stakeholder groups for requirements negotiation based on power relationships," in *Proc. 20th Asia–Pacific Softw. Eng. Conf. (APSEC)*, Bangkok, Thailand, vol. 1, Dec. 2013, pp. 247–254.
[39] A. Lenz and M. Schoop, "Decision problems in requirements negotiations–identifying the underlying structures," *Bus. Inf. Syst.*, vol. 29, no. 3, pp. 120–131, 2017.
[40] J. Liu, H. Chen, C. Chen, and T. Sheu, "Relationships among interpersonal conflict, requirements uncertainty, and software project performance," *Int. J. Project Manage.*, vol. 29, no. 5, pp. 547–556, 2011.
[41] M. Shafiq, J. Matthews, and S. Lockley, "A study of BIM collaboration requirements and available features in existing model collaboration systems," *J. Inf. Technol. Construction*, vol. 18, no. 1, pp. 148–161, 2013.
[42] M. Aldekhail, A. Chikh, and D. Ziani, "Software requirements conflict identification: Review and recommendations," *Int. J. Adv. Comput. Sci. Appl.*, vol. 7, no. 10, pp. 229–326, 2016.
[43] A. Zin and N. Pa, "Measuring communication gap in software requirements elicitation process," in *Proc. 8th WSEAS Int. Conf. Softw. Eng., Parallel Distrib. Syst.*, Cambridge, U.K., 2009, pp. 66–71.
[44] U. S. Shah and D. C. Jinwala, "Resolving ambiguities in natural language software requirements: A comprehensive survey," *ACM SIGSOFT Softw. Eng. Notes*, vol. 40, no. 5, pp. 1–7, Sep. 2015.
[45] K. B. Wilson, V. Bhakoo, and D. Samson, "Crowdsourcing: A contemporary form of project management with linkages to open innovation and novel operations," *Int. J. Oper. Prod. Manage.*, vol. 38, no. 6, pp. 1467–1494, May 2018.
[46] M. Talha, "Critical requirements engineering errors leads to fails software project," *Educ. Rev.*, vol. 2, no. 2, pp. 174–180, 2018.
[47] K. Curcio, T. Navarro, A. Malucelli, and S. Reinehr, "Requirements engineering: A systematic mapping study in agile software development," *J. Syst. Softw.*, vol. 139, no. 1, pp. 32–50, May 2018.
[48] P. Rodríguez, E. Mendes, and B. Turhan, "Key stakeholders' value propositions for feature selection in software-intensive products: An industrial case study," *IEEE Trans. Softw. Eng.*, vol. 46, no. 12, pp. 1340–1363, Dec. 2020.
[49] J. Awotunde, F. Ayo, R. Ogundokun, O. Matiluko, and E. Adeniyi, "Investigating the roles of effective communication among stakeholders in collaborative software development projects," in *Proc. Int. Conf. Comput. Sci. Appl.* Cambridge, U.K.: Springer, 2020, pp. 311–319.
[50] R. Hans and E. Mnkandla, "A model for assisting software project managers to treat project teams as key stakeholders: What do experts say?" *Proc. Comput. Sci.*, vol. 181, no. 1, pp. 1105–1113, 2021.
[51] F. B. Zainuddin, R. B. A. Arshah, R. B. Mohamad, R. B. Mokhtar, R. B. A. Hamid, and N. A. B. Ahmad, "Reviewing the challenge and practices of human factor involvement in requirement specification validation," *Adv. Sci. Lett.*, vol. 24, no. 10, pp. 7322–7327, Oct. 2018.
[52] B. Wang, R. Peng, Y. Li, H. Lai, and Z. Wang, "Requirements traceability technologies and technology transfer decision support: A systematic review," *J. Syst. Softw.*, vol. 146, pp. 59–79, Dec. 2018.
[53] J. W. W. Shao and P. Geng, "An improved approach to the recovery of traceability links between requirement documents and source codes based on latent semantic indexing," in *Proc. Int. Conf. Comput. Sci. Appl.* Berlin, Germany: Springer, 2013, pp. 547–557.
[54] Y. Udagawa, "An augmented vector space information retrieval for recovering requirements traceability," in *Proc. IEEE 11th Int. Conf. Data Mining Workshops*, Sydney, NSW, Australia, Dec. 2011, pp. 771–778.
[55] T. Dietrich, J. Cleland-Huang, and Y. Shin, "Learning effective query transformations for enhanced requirements trace retrieval," in *Proc. 28th IEEE/ACM Int. Conf. Automated Softw. Eng. (ASE)*, Chicago, IL, USA, Nov. 2013, pp. 586–591.
[56] W. Wang, A. Gupta, N. Niu, L. Da Xu, J.-R.-C. Cheng, and Z. Niu, "Automatically tracing dependability requirements via term-based relevance feedback," *IEEE Trans. Ind. Informat.*, vol. 14, no. 1, pp. 342–349, Jan. 2018.
[57] C. Ziftci and I. Krueger, "Tracing requirements to tests with high precision and recall," in *Proc. 26th IEEE/ACM Int. Conf. Automated Softw. Eng. (ASE)*, Nov. 2011, pp. 472–475.
[58] S. Pandanaboyana, S. Sridharan, J. Yannelli, and J. H. Hayes, "Requirements tracing on target (RETRO) enhanced with an automated thesaurus builder: An empirical study," in *Proc. 7th Int. Workshop Traceability Emerg. Forms Softw. Eng. (TEFSE)*, May 2013, pp. 61–67.
[59] M. Shahid, S. Ibrahim, and M. Mahrin, "An evaluation of requirements management and traceability tools," *World Acad. Sci., Eng. Technol.*, vol. 1, no. 1, pp. 1–6, 2011.
[60] B. Ramesh, "Factors influencing requirements traceability practice," *Commun. ACM*, vol. 41, no. 12, pp. 37–44, Dec. 1998.
[61] F. Blaauboer, K. Sikkel, and M. Aydin, "Deciding to adopt requirements traceability in practice," in *Proc. Int. Conf. Adv. Inf. Syst. Eng.* Berlin, Germany: Springer, 2007, pp. 294–308.
[62] A. Kannenberg and H. Saiedian, "Why software requirements traceability remains a challenge," *CrossTalk J. Defense Softw. Eng.*, vol. 22, no. 5, pp. 14–19, 2009.
[63] S. Winkler and J. von Pilgrim, "A survey of traceability in requirements engineering and model-driven development," *Softw. Syst. Model.*, vol. 9, no. 4, pp. 529–565, 2010.
[64] H. Tufail, M. F. Masood, B. Zeb, F. Azam, and M. W. Anwar, "A systematic review of requirement traceability techniques and tools," in *Proc. 2nd Int. Conf. Syst. Rel. Saf. (ICSRS)*, Dec. 2017, pp. 450–454.
[65] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "An overview of blockchain technology: Architecture, consensus, and future trends," in *Proc. IEEE Int. Congr. Big Data (BigData Congr.)*, Jun. 2017, pp. 557–564.
[66] J. Dattani and H. Sheth, "Overview of blockchain technology," *Asian J. Converg. Technol.*, vol. 5, no. 1, pp. 1–3, 2019.
[67] D. Maesa and P. Mori, "Blockchain 3.0 applications survey," *J. Parallel Distrib. Comput.*, vol. 1, no. 138, pp. 99–114, 2020.
[68] S. Wang, L. Ouyang, Y. Yuan, X. Ni, X. Han, and F.-Y. Wang, "Blockchain-enabled smart contracts: Architecture, applications, and future trends," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 49, no. 11, pp. 2266–2277, Nov. 2019.
[69] M. Xu, X. Chen, and G. Kou, "A systematic review of blockchain," *Financial Innov.*, vol. 5, no. 1, pp. 1–14, 2019.

[70] A. Sunyaev, "Distributed ledger technology," in *Internet Computing*. Cham, Switzerland: Springer, 2020, pp. 265–299.
[71] M. M. Rahman, M. M. H. Rifat, M. Y. Tanin, and N. Hossain, "A feedback system using blockchain technology," in *Proc. 3rd Int. Conf. Intell. Sustain. Syst. (ICISS)*, Dec. 2020, pp. 1114–1118.
[72] X. Liu, B. Farahani, and F. Firouzi, "Distributed ledger technology," in *Intelligent Internet of Things*. Cham, Switzerland: Springer, 2020, pp. 393–431.
[73] S. Sabry, N. Kaittan, and I. Majeed, "The road to the blockchain technology: Concept and types," *Periodicals Eng. Natural Sci.*, vol. 7, no. 4, pp. 1821–1832, 2019.
[74] R. Dennis and G. Owen, "Rep on the block: A next generation reputation system based on the blockchain," in *Proc. 10th Int. Conf. Internet Technol. Secured Trans. (ICITST)*, Dec. 2015, pp. 131–138.
[75] V. Dhillon, D. Metcalf, and M. Hooper, *Blockchain Enabled Applications*. Berkeley, CA, USA: Apress, 2017.
[76] D. Freund, "Economic incentives and blockchain security," *J. Securities Oper. Custody*, vol. 10, no. 1, pp. 67–76, 2018.
[77] W. Al-Saqaf and N. Seidler, "Blockchain technology for social impact: Opportunities and challenges ahead," *J. Cyber Policy*, vol. 2, no. 3, pp. 338–354, Sep. 2017.
[78] H. Ouattara, D. Ahmat, F. Ouédraogo, T. Bissyandé, and O. Sié, "Blockchain consensus protocols," in *Proc. Int. Conf. e-Infrastruct. e-Services Developing Countries*. Cham, Switzerland: Springer, 2017, pp. 304–314.
[79] A. Shrestha, J. Vassileva, and R. Deters, "A blockchain platform for user data sharing ensuring user control and incentives," *Frontiers Blockchain*, vol. 3, no. 1, p. 48, 2020.
[80] H. Baber, "Blockchain-based crowdfunding," in *Blockchain Technology for Industry 4.0*. Singapore: Springer, 2020, pp. 117–130.
[81] M. Saadat, S. Halim, H. Osman, R. Nassr, and M. Zuhairi, "Blockchain based crowdfunding systems," *Indonesian J. Electr. Eng. Comput. Sci.*, vol. 15, no. 1, pp. 409–413, 2019.
[82] E. Bertino, A. Kundu, and Z. Sura, "Data transparency with blockchain and AI ethics," *J. Data Inf. Qual.*, vol. 11, no. 4, pp. 1–8, Sep. 2019.
[83] T. Hyla and J. Pejaś, "Long-term verification of signatures based on a blockchain," *Comput. Electr. Eng.*, vol. 81, Jan. 2020, Art. no. 106523.
[84] D. Mingxiao, M. Xiaofeng, Z. Zhe, W. Xiangwei, and C. Qijun, "A review on consensus algorithm of blockchain," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Oct. 2017, pp. 2567–2572.
[85] G. Hileman and M. Rauchs, *Global Blockchain Benchmarking Study*. Rochester, NY, USA: Social Science Research Network, 2017.
[86] M. Friedlmaier, A. Tumasjan, and I. Welpe, "Disrupting industries with blockchain: The industry, venture capital funding, and regional distribution of blockchain ventures," in *Proc. 51st Annu. Hawaii Int. Conf. Syst. Sci. (HICSS)*, 2017, pp. 1–10.
[87] G. Fridgen, J. Lockl, S. Radszuwill, A. Rieger, A. Schweizer, and N. Urbach, "A solution in search of a problem: A method for the development of blockchain use cases," *AMCIS*, vol. 1, no. 1, pp. 1–11, 2018.
[88] L. D'Oriano, G. Mastandrea, G. Rana, G. Raveduto, V. Croce, M. Verber, and M. Bertoncini, "Decentralized blockchain flexibility system for smart grids: Requirements engineering and use cases," in *Proc. Int. IEEE Conf. Workshop Óbuda Elect. Power Eng. (CANDO-EPE)*, Nov. 2018, pp. 39–44.
[89] A. Cockburn, *Writing Effective Use Cases*. New Delhi, India: Pearson, 2001.
[90] I. Singh and S.-W. Lee, "RE_BBC: Requirements engineering in a blockchain-based cloud system: Its role in service-level agreement specification," *IEEE Softw.*, vol. 37, no. 5, pp. 7–12, Sep. 2020.
[91] I. Singh and S. Lee, "Requirement engineering and its role in a blockchain-enabled Internet of Things," in *Blockchain Technology for IoT Applications*. Singapore: Springer, 2021, pp. 1–15.
[92] O. Daramola and D. Thebus, "Architecture-centric evaluation of blockchain-based smart contract e-voting for national elections," *Informatics*, vol. 7, no. 16, pp. 1–16, 2020.
[93] S. Chawla, "Goal oriented requirements engineering for blockchain based food supply chain," *Int. J. Softw. Eng. Comput. Syst.*, vol. 6, no. 1, pp. 94–103, 2020.
[94] V. Salehi, "Integration of blockchain technologie in case of systems engineering and software engineering in an industrial context," *Proc. Des. Soc.*, vol. 1, pp. 1887–1896, Aug. 2021.
[95] S. Alzahari and M. Kamalrudin, "An approach to elicit trustworthiness requirements in blockchain technology," *J. Phys., Conf. Ser.*, vol. 1807, no. 1, Apr. 2021, Art. no. 012031.
[96] A. Vingerhoets, S. Heng, and Y. Wautelet, "Using i* and UML for blockchain oriented software engineering: Strengths, weaknesses, lacks and complementarity," *Complex Syst. Informat. Model. Quart.*, vol. 26, no. 1, pp. 26–45, 2021.
[97] J. F. C. Beinke and F. Teuteberg, "Towards a stakeholder-oriented blockchain-based architecture for electronic health records: Design science research study," *J. Med. Internet Res.*, vol. 21, no. 10, p. 13585, 2019.
[98] S. Almeida, A. Albuquerque, and A. Silva, "An approach to develop software that uses blockchain," in *Proc. Comput. Sci. Line Conf.* Cham, Switzerland: Springer, 2018, pp. 346–355.
[99] S. Bouraga, C. Burnay, I. Jureta, and S. Faulkner, "Requirements elicitation for applications running on a blockchain: Preliminary results," in *Proc. Int. Conf. Adv. Inf. Syst. Eng.* Cham, Switzerland: Springer, 2021, pp. 38–46.
[100] C. Braghin, S. Cimato, S. Cominesi, E. Damiani, and L. Mauri, "Towards blockchain-based E-voting systems," in *Proc. Int. Conf. Bus. Inf. Syst.* Cham, Switzerland: Springer, 2019, pp. 274–286.
[101] P. McCorry, S. Shahandashti, and F. Hao, "A smart contract for boardroom voting with maximum voter privacy," in *Proc. Int. Conf. Financial Cryptogr. Data Secur.* Cham, Switzerland: Springer, 2017, pp. 357–375.
[102] D. Pawade, A. Sakhapara, A. Badgujar, D. Adepu, and M. Andrade, "Secure online voting system using biometric and blockchain," *Data Manage., Anal. Innov.*, vol. 1, no. 1, pp. 93–110, 2020.
[103] A. N. Mindila, J. M. Wafula, H. A. M. Ratemo, C. Tabu, J. Charo, and C. Silali, "Requirements elicitation for a blockchain vaccine supply chain management web/mobile application," *Gates Open Res.*, vol. 3, p. 1420, Apr. 2019.
[104] I. Inayat, S. S. Salim, S. Marczak, M. Daneva, and S. Shamshirband, "A systematic literature review on agile requirements engineering practices and challenges," *Comput. Hum. Behav.*, vol. 51, pp. 915–929, Oct. 2015.
[105] M. Marchesi, L. Marchesi, and R. Tonelli, "An agile software engineering method to design blockchain applications," in *Proc. 14th Central Eastern Eur. Softw. Eng. Conf. Russia*, 2018, pp. 1–8.
[106] L. Marchesi, M. Marchesi, and R. Tonelli, "ABCDE—Agile block chain DApp engineering," *Blockchain, Res. Appl.*, vol. 1, nos. 1–2, Dec. 2020, Art. no. 100002.
[107] M. Yilmaz, S. Tasel, E. Tuzun, U. Gulec, R. V. O'Connor, and P. Clarke, "Applying blockchain to improve the integrity of the software development process," in *Proc. Eur. Conf. Softw. Process Improvement*. Cham, Switzerland: Springer, 2019, pp. 260–271.
[108] N. Six, "Decision process for blockchain architectures based on requirements," vol. 21, no. 1, pp. 89–98, 2021.
[109] L. Cong and Z. He, "Blockchain disruption and smart contracts," *Rev. Financial Stud.*, vol. 32, no. 5, pp. 1754–1797, 2019.
[110] I. Sergey and A. Hobor, "A concurrent perspective on smart contracts," in *Proc. Int. Conf. Financial Cryptogr. Data Secur.* Cham, Switzerland: Springer, 2017, pp. 478–493.
[111] A. Singh, R. Parizi, Q. Zhang, K. Choo, and A. Dehghantanha, "Blockchain smart contracts formalization: Approaches and challenges to address vulnerabilities," *Comput. Secur.*, vol. 88, no. 1, 2020, Art. no. 101654.
[112] M. Jurgelaitis, R. Butkienė, E. Vaičiukynas, V. Drungilas, and L. Čeponienė, "Modelling principles for blockchain-based implementation of business or scientific processes," in *Proc. CEUR Workshop, IVUS Int. Conf. Inf. Technol., Int. Conf. Inf. Technol.*, Kaunas, Lithuania, vol. 2470, Apr. 2019, pp. 43–47.
[113] P. Garamvölgyi, I. Kocsis, B. Gehl, and Klenik, "Towards model-driven engineering of smart contracts for cyber-physical systems," in *Proc. 48th Annu. IEEE/IFIP Int. Conf. Dependable Syst. Netw. Workshops (DSN-W)*, Jun. 2018, pp. 134–139.
[114] C. Hebert and F. Di Cerbo, "Secure blockchain in the enterprise: A methodology," *Pervas. Mobile Comput.*, vol. 59, no. 1, 2019, Art. no. 101038.
[115] W. Du, S. L. Pan, D. E. Leidner, and W. Ying, "Affordances, experimentation and actualization of FinTech: A blockchain implementation study," *J. Strategic Inf. Syst.*, vol. 28, no. 1, pp. 50–65, Mar. 2019.