

Article

Reliable Requirements Engineering Practices for COVID-19 Using Blockchain

Basit Shahzad ^{1,*}, Iqra Javed ^{2,*}, Asadullah Shaikh ³, Adel Sulaiman ³, Ahsanullah Abro ⁴ and Muhammad Ali Memon ⁵

¹ Department of Software Engineering, Faculty of Engineering, NUML, Islamabad 44000, Pakistan

² Software Engineering and Social Computing Research Group (SESCRG), Islamabad 44000, Pakistan

³ College of Computer Science and Information Systems, Najran University, Najran 61441, Saudi Arabia; asshaikh@nu.edu.sa (A.S.); aaalsulaiman@nu.edu.sa (A.S.)

⁴ Department of Computer Science, Sukkur IBA University, Sukkur 79165, Pakistan; ahsanullah.abro@iba-suk.edu.pk

⁵ Department of Information Technology University of Sindh, Jamshoro 76090, Pakistan; muhammad.ali@usindh.edu.pk

* Correspondence: bshahzad@numl.edu.pk (B.S.); iqrajavedd9@gmail.com (I.J.)

Abstract: Improvement in the requirements for engineering practices is needed in areas such as requirement elicitation, validation, prioritization, and negotiations between stakeholders to create successful projects for COVID-19 (coronavirus disease 2019) software. Many algorithms and techniques are used to create quality software projects, but they still need more improvement to work effectively for global pandemic COVID-19 software. By improving the reliability of requirement engineering practices using blockchain-based technology, the software will be reliable and will make it easier for the users working in a lockdown situation because of COVID-19. Therefore, our purpose is to identify the factors for reliable software engineering practices using blockchain-oriented technology for COVID-19 software. A systematic literature review is conducted to identify challenges and offer solutions. Through using blockchain-based technology for requirement engineering practices, the requirements will be gathered accurately and validated, and the conflicts between stakeholders will also be solved. It will improve the quality and reliability of COVID-19 software projects, which will help society work effectively from home. Improvement in the quality and reliability of COVID-19 software will improve users' interest, and their working capacity will be increased.

Keywords: blockchain; COVID-19 software; reliability; requirement engineering



Citation: Shahzad, B.; Javed, I.; Shaikh, A.; Sulaiman, A.; Abro, A.; Ali Memon, M. Reliable Requirements Engineering Practices for COVID-19 Using Blockchain. *Sustainability* **2021**, *13*, 6748. <https://doi.org/10.3390/su13126748>

Academic Editor:
José-Luis Sierra-Rodríguez

Received: 25 March 2021
Accepted: 2 June 2021
Published: 15 June 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

As a result of the COVID-19 global pandemic, the whole world must work from home, because it will be helpful to prevent this disease from spreading. Unfortunately, we lack quality software projects, so it reduces the interest and working capacity of workers. Most of the software is slow because of workload, which irritates the users. There are many problems in their requirement engineering practices, which lead to the slow development of quality software projects. The competition level between organizations is increased with innovations in technology. All the organizations must cope with the increasing rate of change in technology for COVID-19 software projects. In the face of challenging work situations, software products play a fundamental role because successful software products work according to the requirement of stakeholders. So, the general purpose of this research is to identify the factors for the reliability of requirement engineering practices using blockchain-based technology. These are the factors that can be improved using the blockchain platform to create reliable COVID-19 software.

Requirement engineering is the process that defines, documents, and maintains the requirements to produce a successful software product. The main objective of requirement

engineering is to interpret and understand the needs, goals, and beliefs of stakeholders. There are many techniques and methods of requirement engineering processes to make it easier and faster [1]. However, correctly interpreted, gathered, and negotiated requirements help to complete projects successfully according to the needs of customers within a given time and estimated budget. This process of requirement elicitation, validation, negotiation, and prioritization is referred to as requirement negotiation [2]. If the quality of the requirement negotiation process is effective and reliable, then the software project will also be successful. Requirement elicitation is related to various ways used to gain knowledge about the project domain and requirements of stakeholders [3]. The sources of domain knowledge of requirement elicitation include customers, business manuals, and existing software of the same type standards and other stakeholders of the project [4]. The techniques used for requirement elicitation include interviews, brainstorming, task analysis, prototyping, etc. These techniques help with gathering and interpreting requirements in more organized and easy forms. To check that if these gathered requirements are according to customer needs or not, requirement validation tests are performed [5]. These requirements validation techniques ensure that the gathered requirements are according to the demands of stakeholders; if there is some error in gathered requirements, then it will help to correct them. Several techniques are used for requirement validation, but they are complex, time-consuming, and expensive such as walk-through, automated consistency analysis. Requirement negotiation is most important for e-business in which all business is conducted online. While work on negotiation automation is somewhat mature, it still needs to be approached systematically with a broader scope [6]. In e-commerce, the web-based server is used for conducting bargaining-type negotiations between stakeholders such as Yahoo's auction service. In e-commerce, companies and individuals order, buy, and sell products and get payment in an electronic way [7]. Requirement negotiation is the most crucial part of e-commerce, which needs more improvement for COVID-19 software projects.

As we know, the effective development of any project involves a software system based on the quality of any requirement engineering practice [8]. For designing and developing successful COVID-19 software projects, requirement engineering is primary and one of the most crucial parts because it addresses the design and development of problems for the suitable project for the customer as well as allows creating correct time and budget estimation and to fulfill the needs of the customer in this pandemic lockdown situation [9]. In addition, the consistency of requirements is a question that needs to be answered. While we need to accumulate requirements from multiple sources sometimes in parallel, their synchronization and consistency need to be ensured [10]. To improve the rate of successful COVID-19 software projects, the latest technology should be used. The use of the latest innovative technology in requirement engineering practices in all phases of the software development lifecycle (SDLC) will improve the performance of COVID-19 software projects [11].

Thus, blockchain technology is adopted in many organizations and applications because of its quality, security, and consistency [12]. So, to improve the quality and reliability of COVID-19 software projects and to reduce the failure rate of projects, blockchain-oriented requirement engineering is discussed to make requirement engineering processes more verified and easier [13]. Blockchain is a secured and distributed ledger technology that provides an immutable record of transactions. Blockchain is a decentralized online global database, so its ledger is shared among all the stakeholders [14]. Through the network, all the transactions are recorded in a ledger on a computer, and this ledger is shared with every computer around the world. Anyone can access these transactions and add transactions but cannot change the ledger after the transaction is added [15]. This blockchain is a collection of data, and by connecting one block after another in a chronological way, each piece of data is added to the blockchain. This series of chronologically connected blocks make a chain of blocks, and it is called blockchain [16]. The idea of blockchain is now spreading

widely all over the world. The concept of blockchain can now be applied to requirements for trustworthy record management [17].

Blockchain is allowing people to secure digital relationships that were impossible before because data are being disclosed, recorded, and secured differently. So, the uniqueness of blockchain is that it is secure, immutable, disintermediating, and less costly [18]. By using a blockchain platform in requirement engineering practices, we can create COVID-19 software that is more reliable and accessible, and it can help to gather consistent data or requirements more easily [19]. Blockchain technology integrates a series of technical systems with the following characteristics of decentralization, openness, autonomy, anonymity, security, and information that cannot be tampered with [20].

As we have discussed, the quality of the COVID-19 project involves that how well its requirements were gathered and fulfilled. However, the requirement elicitation is a complex and time-consuming task. Although many techniques and algorithms are used to make it easier and efficient, still, there is a lacking in the quality of projects. To improve the quality of COVID-19 projects, the requirements must be gathered accurately and completely. For this purpose, blockchain platforms can be used to make requirement engineering practices more efficient. So, we are identifying factors for the reliability of requirement engineering practices using blockchain technology.

By using the blockchain platform in requirement engineering practices, we can create requirement engineering processes that are more secure, and it can help gather data or requirements more easily and more consistently and correctly [21]. So, the stakeholders can easily contribute/gather authorized requirements more easily, which will reduce their time and efforts [22]. By introducing blockchain-oriented requirement engineering, the requirement engineering practices will be more efficient, with improved quality of COVID-19 software projects, and the failure rate of projects will be reduced. Since quality work is lacking in this area, there is a great potential to explore this area for future research and usage of blockchain technology, improving the quality of software requirements in the field of health care. The rest of the paper describes the research in further detail. Section 2 contains the literature review, Section 3 presents the problem statement, Section 4 lists the objectives, Section 5 outlines the research questions, Section 6 describes the methodology, Section 7 presents the discussion, and research paper is concluded in Section 8.

2. Literature Review

Problems in requirement engineering practices can be of many types; they can be incomplete, ambiguous, imprecise, informal, conflicting, and have unattainable requirements [23]. Every organization needs to fulfill all the functional and non-functional requirements of stakeholders because the quality in requirements produces quality products. Several techniques and algorithms are used for gathering correct requirements and developing projects successfully [24]. In a global pandemic, COVID-19 software projects need more improvement and reliability to give users ease and to increase their interest in work. Requirement engineering is the process of defining, documenting, and maintaining requirements to create successful software projects [25]. It is a process of gathering and defining services provided by the system. However, correctively interpreted and gathered requirements help to complete projects successfully according to the needs of the customer within a given time and estimated budget [26]. The quality of the requirement negotiation process results in the good quality of the product [27].

Requirement negotiation helps to solve conflicts between stakeholders and to come to an agreement. However, in requirement elicitation, there are many techniques to gather quality requirements [28]. However, most of the time, there is a problem in prioritizing which requirements are the most important and which requirements are not important for projects [29]. The requirement elicitation is related to various ways used to gain the knowledge about the project domain and requirements of stakeholders. In requirement elicitation, various sources of domain knowledge include customers, business manuals, and existing software of the same type, standards, and other stakeholders of the project [30].

The proficiency and reliability in the process of requirement negotiation are also very important for quality in the COVID-19 e-commerce business. While in some respects, work on negotiation automation is mature enough, it still needs to be approached systematically with a broader scope for improved COVID-19 software projects. An e-commerce web-based server is used for conducting bargaining-type negotiation between stakeholders such as Yahoo's auction service. In e-commerce, companies and individuals order, buy, and sell products and get payment through in an electronic way, which needs proficiency and sustainability in the requirement gathering and negotiation process.

The negotiation process is also used in cloud computing [31]. The negotiation of requirements in these servers needs more improvement and proficiency for improved COVID-19 software. In this new era, web applications and mobiles are used for the process of requirement negotiation [32]. Goal-oriented requirements engineering is used to add goals in requirement elicitation, validation, analysis, and requirement negotiation to add improvement in these processes [33]. An interest-based learning activity negotiation system is performed to note the interests of both developers and stakeholders, which will increase the interest of both learners and stakeholders [34]. To overcome language problems among all stakeholders who are remotely negotiating software requirements, machine translation and speech recognition are combined to generate a mobile speech translator [35]. Requirement negotiation is collaborating with the software ecosystem to make COVID-19 software more understandable [36]. Factors that need to be improved for the reliability of requirement engineering by using blockchain-based technology for COVID-19 software projects are discussed below in detail and are shown in a sequence in Figure 1. These factors are further divided into sub-factors. These factors include quality, social networks, conflicts in decision making, stakeholder relationships, negotiation process, time, validation, and consistency of requirements. All these factors are important for producing quality software requirements and reliable COVID-19 software projects.

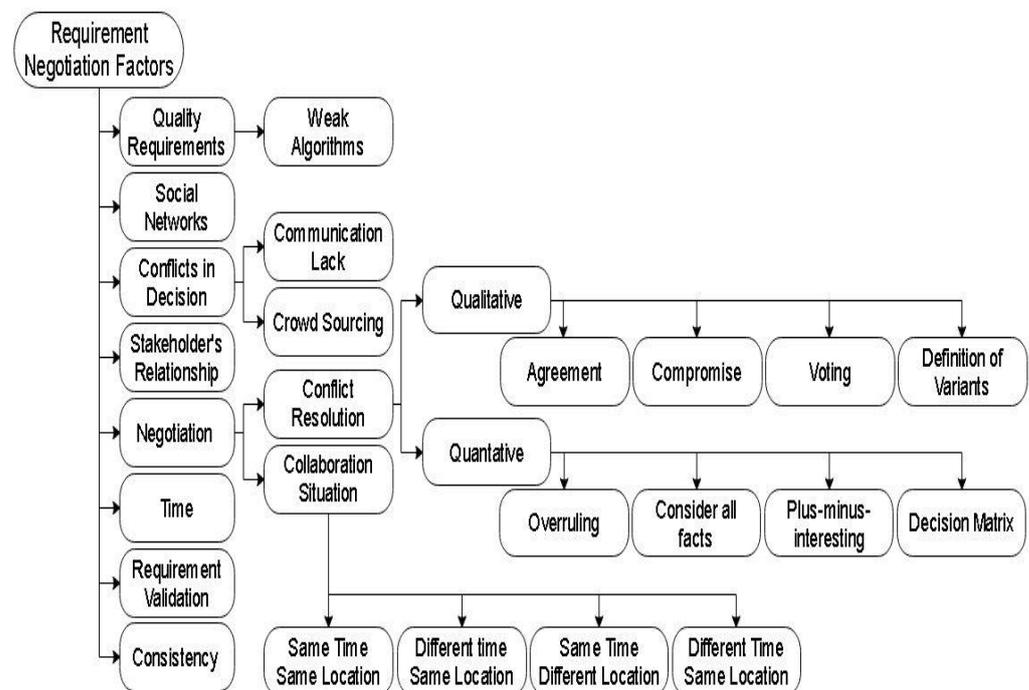


Figure 1. Factors for improvement in requirement negotiation.

2.1. Quality

Many expert systems fail because of a lack in quality of the requirement negotiation process [37]. For designing and developing successful projects, requirement negotiation is primary and one of the crucial parts, and it addresses designing and developing problems

for the suitable project for the stakeholders [38]. The software development life cycle is followed by almost every software development organization, but still, many projects fail due to some reasons. The rate of failed projects is still more than the rate of successful projects all over the world.

Weak Algorithms

In the requirement negotiation process, algorithms generate quality requirements by the elicitation, validation, prioritization, and negation of requirements, but these algorithms are weak for producing valid and prioritized requirements, so the projects lack quality. So, to improve the quality of projects, strong and efficient algorithms should be used.

2.2. Time

Before finalizing any decision, a discussion or meeting takes place, and all the stakeholders communicate and argue with each other on their decisions. Sometimes, there are conflicts between stakeholders, and solving these conflicts is a time-consuming process. Many other approaches to the negotiation process are time-consuming and difficult. To reduce the wastage of time on the approaches of requirement negotiation, it can be improved by adopting some new platforms.

2.3. Social Network

Social media plays an important role in the requirement negotiation process because of its huge amount of data coverage. To remove biasness and solve conflicts in requirement negotiation phases, social media serves as a communication interface. By using social media for requirement negotiation, many processes and activities are improved to give better results [39]. There is a bulk of information on the internet, but most of the data are located on websites, which tend to be unauthorized sources. So, the data should be authorized, and data gathering should be easy and the conflicts between stakeholders should be solved [40]. The implementation of non-technical user-friendly methods is a significant challenge for system definition and evaluation and decreases the failure rate of the system. The WinWin negotiation framework has improved these challenges until its fifth generation; even then, there are many challenges with no technical users in the latest version of the Wiki WinWin toolset [41]. For the requirement negotiation process, the arrival of social media made Gmail and Facebook tremendously popular [42]. Winbook is the recent version of the WinWin framework; it works on social networking models similarly to Facebook and Gmail. The use of Winbook in small-scale projects showed improvement in the requirement negotiation process, but it also has many challenges in requirement negotiation and software development. The establishment of requirement negotiation is not an easy process, because it is very difficult to identify conflicts and techniques for how these identified conflicts can be resolved. Then, finding the feasible alternatives is also a difficult step, and then giving the charge of negotiation is also a difficult decision-making process. It is also difficult to know how we can support the negotiation with tools and other means [43].

2.4. Conflicts in Decision Making

In the requirement negotiation process, all the stakeholders need to make some important decisions during this process. Although they have the same goal, because of different experiences and thoughts, they have conflicts regarding requirement selection. In software organizations, the negotiation stakeholders include the project managers, customers, developers, suppliers, product managers, and management of companies. They all have different experiences, so their conflicts among requirements are also high. Many tools and techniques are used in the requirement negotiation process to make the decision-making process easy, but it is a complex and time-consuming task. If the final decision fails to satisfy any of the stakeholders, then they will give less interest in the further decision-

making process, and this will lead to a poor decision-making activity [44]. So, the conflicts between stakeholders should be solved.

2.4.1. Lack of Communication

Lack of communication is a big problem in decision making in requirement negotiation. The requirement negotiation process should involve all of the stakeholder collaboration at the same location and same time. However, most of the time, the stakeholders communicate from a different location and different time, which is the reason for bad decision-making activities. Sometimes, stakeholders fail to explain or justify their decisions due to a lack of communication skills.

2.4.2. Crowdsourcing

Crowdsourcing is a model in which organizations or stakeholders share their services, ideas, or finances with a large group of internet users. Different organizations assign projects to the developers through the internet and the developers send back that project after completion and receive his or her payment. Requirement negotiation in crowdsourcing is a complex task because it needs the synchronizations of time, location, and understanding between stakeholders [45]. Most of the time, stakeholders fail to understand each other's point of view and have conflicts.

2.5. Power Relationship between Stakeholders

Due to continuous changes in technology, the complexities are increasing in requirement engineering, and power relationships between stakeholders are also effective for resolving conflicts among them. Different principles are used for solving this gap, but still, the power relationships are getting affected because of conflicts between stakeholders [46].

2.6. Consistency of Requirements

In addition, the consistency of requirements is a question that needs to be answered. While we need to accumulate requirements from multiple sources in parallel, their synchronization and consistency need to be ensured.

2.7. Requirement Validation

To check that if these gathered requirements are correct, complete, and according to the needs of stakeholders or not, requirement validation tests are performed. These requirement validation techniques ensure that the gathered requirements are according to the demands of stakeholders; if there are some errors in the gathered requirements, then it will help to correct them and protect the software project from defects after deployment. Requirement validation is the most important phase after requirement gathering because in requirement gathering, there is a big number of useless requirements. This increases the complexity of the system, so when gathering complete and correct requirements, a validation phase is essential. Several techniques are used for requirement validation, but because of errors in the requirement negotiation process, the gathered requirements are not fully according to the needs of all of the stakeholders.

2.8. Requirement Negotiation

The requirement negotiation is used to define mutually satisfactory requirements on which all the stakeholders are agreed. As a result of rapid changes in the market, the requirement negotiation in different phases of requirement engineering helps to make changes according to each stakeholder's updated requirements. Requirement negotiation is not used only once in a project; it should be repeated in later stages of the requirement engineering process [47]. Different tools are used for requirement negotiation; some of them are passive support, active facilitative support, and pro-active interventive support tools. Passive tools support all collaboration situations by providing the infrastructure for negotiation. All the parties express their views and ideas and collaborate through

email, multimedia rooms, chat, etc. Active facilitated support tools help the stakeholders come to an agreement; for instance, for mutual gain, it identifies the situation. This tool helps in evaluation, formulation, and problem-solving. Pro-active interventive support tools manage the actions of stakeholders. These tools critically analyze the actions of stakeholders and help them agree [48].

In requirement negotiation, there is a mutual interaction between all the stakeholders, and they negotiate on all the requirements, and the requirements on which all the stakeholders are agreed are used in creating a project. As a result of many requirements, it is difficult to negotiate on each requirement under a specific budget and time [49]. Two different dimensions of requirement negotiation are discussed, which helps to solve conflicts between stakeholders, the collaboration between stakeholders, and the tools that are used for solving problems. These dimensions are as follows.

1. Conflict resolution strategy

It is very important to resolve the conflicts between stakeholders because if one party was not convinced with the resolved conflict, then there will be a lack of interest in stakeholder engagement. This will lead to the failure of the project. When conflicts are resolved by all stakeholders, then the selected requirements will be authorized, the developed project will be according to the needs of stakeholders, and it will lead to the success of the project. There are many kinds of conflict resolution techniques:

The qualitative conflict resolution techniques are divided into four categories: agreement, compromise, voting, and definition of variants.

- Agreement: To negotiate a solution to the conflict, the stakeholders work together, which includes the discussions about views of each stakeholder to agree.
- Compromise: In this technique, all the stakeholders compromise on an acceptable solution that is generated after an analysis of various solutions.
- Voting: In this technique, the stakeholders simply vote in favor of the solution or against that solution, and the solution with high votes will be selected as the best solution.
- Definition of variants: In this technique, the preferred solution of stakeholders is implemented by applying their variants to the solution parameters. This is a qualitative technique [50].

The qualitative conflict resolution strategy is categorized into overruling, consider all facts, plus–minus–interesting, and decision matrix.

- Overruling: In this technique, the solution of the most senior stakeholders is taken as the resolution. This technique works as the hierarchy of organizations.
- Consider all facts: In this technique, all the facts about specific conflicted requirements are considered and then prioritized in a readiness to be used as an input to the “Plus–Minus–Interesting” technique.
- Plus–Minus–Interesting: All the positive and negative facts of solutions alternative are analyzed in this technique. Two categories, one for Plus and one for Minus, are developed to list the positives and negatives. When a repercussion is neither a positive nor a negative item, it is placed in the Interesting column.
- Decision Matrix: The decision matrix consists of a comparison matrix of all key criteria to resolve a requirement conflict. These key criteria need to be considered against each solution alternative. The comparison of information in the matrix highlights the best solution to conflicts.

2. Collaboration situation

In this dimension of requirement negotiation, all the stakeholders interconnect with each other for which their time and location are negotiated. The collaboration can be synchronous or asynchronous. In synchronization negotiation, stakeholders work together face to face, but in asynchronous negotiation, stakeholders collaborate from different

locations at different times, which is a difficult stage [1]. The collaboration situation is divided into four types:

- Same time same location: The stakeholders collaborate at the same time and location. The face-to-face communication helps them communicate with each other easily.
- Different time same location: The stakeholders collaborate face to face, but their timings are different from each other. Thus, all the stakeholders cannot collaborate at the same time.
- Same time different location: The stakeholders collaborate at the same time but from different locations. Due to the communication gap, they face many problems and the rate of conflict between them is high.
- Different time different location: The stakeholders collaborate in different times and different location; due to the communication gap and different timing, conflicts arises.

These requirement negotiation strategies are used to make the requirement elicitation and validation process easier and validated with many algorithms, but still, there is a need to improve these strategies, time, and budget with the innovative technology [51]. The time and communication problems exceed the specific budget during the requirement negotiation process and cause the failure rate of projects to increase. The use of the latest technology will increase the performance and help produce successful projects efficiently.

2.9. Solution

So here, blockchain-oriented requirement engineering is discussed to make the requirement engineering practices more verified, validated, reliable, and easy. Blockchain is a decentralized, secured, and distributed ledger technology that provides an immutable record of transactions. Blockchain is a decentralized online global database, so its ledger is shared among all computers [52]. Through the internet, all the transactions are registered on a machine, and this record is shared by every machine throughout the world. Anyone can access these activities and add transactions but cannot change the record after the transaction is added. This blockchain is a collection of data, and by connecting one block after another in a chronological way, each piece of data is added to the blockchain. This series of chronologically connected blocks makes a chain of blocks, and it is called blockchain [53]. The idea of blockchain innovation is spreading widely, because the concept can now be applied to any requirement for trustworthy records [54].

- It facilitates people's records with encryption and makes them secure and synchronized [55].
- Blockchain is allowing people to secure digital relationships that were impossible before because data is being disclosed, recorded, and secured differently [56].
- The main advantage of the Bitcoin-style blockchain systems is the immutability of data and decentralized authority [57].
- The decentralized feature of blockchain will help to create a decentralized ledger of requirements that will easily be accessible by stakeholders [58].
- The uniqueness of blockchain is that it is secure, immutable, and less costly [59].
- By using the blockchain platform in requirement negotiation, we can create requirement engineering practices that are more secure and accessible.
- It can help to gather and validate requirements easily for COVID-19 software projects [60].
- Blockchain technology integrates a series of technical systems with the following characteristics of decentralization, openness, autonomy, anonymity, security, and information that cannot be tempered [61].

As we have discussed before, the quality of the project depends on how well its requirements were gathered, fulfilled, or negotiated. However, the requirement negotiation is a difficult and time-consuming task [62]. To improve the quality of COVID-19 software projects, the requirements must be negotiated accurately and completely. For this

purpose, we can use blockchain platforms to make the requirement negotiation process more efficient [63].

The innovation of blockchain is a revolution in frameworks of record [64]. Thus, blockchain technology is adopted in many organizations and applications, and because of it, the quality, security, and speed quality of projects improved. So, to improve the quality and speed of the requirement negotiation process and to reduce the failure rate of COVID-19 software projects, blockchain-oriented requirement engineering is discussed to make requirement engineering practices more verified and easier [65]. Some applications of blockchain as a record are digital identity and tokenization [66]. Blockchain is the grouping of the assortment of advancements of the products. Cryptography, computer network, mathematics, etc. are used to create blockchain technology [67].

The objective of blockchain is to make a decentralized advanced framework by using algorithms and cryptography, which makes it secure. The use of blockchain is broad and applicable in all parts of society [68]. Platforms based on blockchain will help people gather information according to its type. If the record is from a public blockchain, then the user can read, write, or edit information, and if it is from a private blockchain, then the individual or organization that is in charge checks and secures important things such as selective reading, writing, or access privileges [69]. It works similar to a private property of anyone [70]. If information is from a consortium or federated blockchain rather than one in control, you have more than one in control [71]. In a consortium blockchain, a group or organization's agent people settle on choices together for benefit of the whole system [72]. We can use blockchain platforms to make the requirement negotiation process more efficient. In this process, requirements will be gathered from users, and those gathered requirements will be added in the requirement correctly, completed, and authenticated. In the second step, these user requirements will be validated by using the blockchain-based distributed ledger framework [73]. This distributed ledger will update the record efficiently. The most likely accepted requirement will be ranked as number one and then another lesser likely requirement will be ranked as number two and arranged according to their maturity level [74]. Through this voting process, all the requirements will be tagged as valid or invalid [75]. It will also distinguish between functional and non-functional requirements [76]. In the context of requirement engineering, blockchain can maintain a sequence of records, research, authorized information, transactions, negotiations, and other supporting activities and organizations [77], which as a result will increase the rate of successful projects. By using blockchain technology, the trust level on the selected functional requirements will be increased [78]. So, by using the blockchain platform in requirement negotiation, we can create requirement engineering practices that are more reliable and accessible, it can help gather requirements that are more synchronized, and the people from any place can easily gather authorized requirements more easily, which will reduce their time and efforts during the COVID-19 pandemic [79].

The studies in Table 1 show that the requirement negotiation process needs more improvement, and blockchain technology is better for implementation in the requirement negotiation process. The first five papers listed in the table describe and tell the importance of the requirement negotiation process and explain the methods used for the negotiation process; their limitation shows that the introduced methods are better suitable for small projects, but for large-scale projects, these methods should be improved. Six to nine papers are explaining the blockchain platforms and their implementation in multiple fields. In further listed papers, factors of requirement negotiation are discussed, and their limitations are given for further improvement, and blockchain platforms and their frameworks are described in detail. This helps to implement a blockchain-based framework for the requirement negotiation process. So, this research has identified factors for improvement in the requirement negotiation process using blockchain technology to make the requirement negotiation process efficient. Related studies that help in gathering and supporting information are listed in Table 1, which shows the advantages, limitations, and key factors, which are identified in related studies.

The related studies are given in Table 1 to show the understanding of requirement engineering and its processes. Requirement engineering techniques, practices, and methods are listed with their key factors, advantages, and limitations. The papers listed below outline many limitations, which include security issues regarding synchronization, low-level projects, human error, biasness, prioritization, time consumption, undefined requirements, validation, and elicitation of requirements in software.

Requirement negotiation-related studies and factors in requirement engineering are listed in Table 2, which outlines different requirement negotiation techniques and methods as well as their advantages and limitations. The limitations include centralization, applications not capable of handling a vast number of requirements, time consumption, synchronization, negotiation, trust issues, integration, the relationships between stakeholders, and conflicts between stakeholders. The most repeated challenges in techniques of the requirement negotiation process are time and budget increase, conflicts between stakeholders, communication gaps, voting process, and prioritization of requirements. These papers support research question 1. Factors and challenges in requirement negotiation are identified by literature review, which are listed in the below table. There are many risks and limitations in the methods of requirements engineering because the software success rate is lower than its production. These limitations and risks are also studied, and some are listed in Table 2. The risk identified from the papers listed below helps to improve the performance of a project and to give important aspects on which to focus.

Blockchain platforms are listed in Table 3. In this table, the advantages and limitations of blockchain platforms are listed. These platforms help to implement a framework for requirement negotiation. Blockchain platforms are very effective and secure to use. These platforms are adopted in different fields. These platforms are decentralized and effective and can help to speed up the process of the requirement negotiation process. The applications in which blockchain is used are studied and addressed in Table 3. These blockchain-based applications and blockchain methods help to give a better understanding of blockchain and help to develop a framework using the features of blockchain. Blockchain-based applications are more secure and efficient than other applications. Blockchain-based applications help to develop a framework for the requirement negotiation process.

Related studies that help in gathering and supporting information are in Tables 1 and 2, which show the advantages, limitations, and key factors that are identified in related studies [80]. The literature in Table 2 is related to the factors of requirement negotiation and its limitations, and it also includes information about the platforms, applications, framework, strengths, and weaknesses of blockchain technology. The limitations in the requirement negotiation process help ascertain the main factors to focus on in this research area and can be improved using the blockchain-based framework.

Many research papers are studied thoroughly about requirement negotiation and blockchain technology. Few papers discussed factors in the requirement negotiation process. Some papers show the gaps in the requirement negotiation process, and some papers help to find the improvements in the requirement negotiation process [81]. A list of factors of requirement negotiation is identified that need further improvement. The factors will be validated through the survey in which questions about factors will be asked from respondents.

There is a need for improvement in factors of the requirement negotiation process, so we have identified that these factors can be improved using blockchain-based technology. Many papers, which are listed in Table 3, are about blockchain technology in which different features, platforms, and applications of blockchain technology are briefly explained. Types and platforms of blockchain technology are also explained, which helps to introduce a framework for the improvement of requirement negotiation process factors. The introduced framework will be based on the decentralized feature of blockchain technology, and it will be more efficient and easier to use. The introduced framework will be validated using a survey by taking the responses of experts in blockchain and requirement negotiation fields.

Table 1. Bibliometric analysis of requirement engineering.

Ref.	Key Factors	Merits	Demerits
[1]	Need to improve requirement engineering, the negotiation process	Help in pinpointing requirement negotiation process	A human being is performing the role of moderator, high chance of human error, not secure
[8]	A module-based integrated model is proposed and demonstrated with UML modeling	Produce a quality product	Not synchronized, techniques are not applicable on all types of expert systems, a requirement not defined properly
[9]	“Customized requirement engineering practices followed by the software development organization of Pakistan”	“Help us to understand the gaps in the requirement engineering process of Pakistan’s organizations and other developing countries”	Not prioritize efficiently, time consuming
[11]	“Synthesizes the underlying concepts of GORE concerning coverage of requirement engineering activities”	Make requirements clear	Not for large projects, not synchronized, time consuming
[25]	How to model goals and information requirements for data warehouses, and how to derive a conceptual multidimensional model	“Helps to provides the required information to support the decision-making process”	Not synchronized, techniques are not able to give the accurate results from different expert systems, requirements not defined properly
[29]	Better modeling and analysis of problem domains, development of richer models for capturing and analyzing non-functional requirements	Helps to find out an overview of software system requirement engineering. Helps to find challenges in software engineering	Not prioritized efficiently, time consuming
[38]	Model implementation for successful software development and requirement management	Helps to produce quality software products by its model	Other elicitation techniques can also be used, requirement validation is not used
[23]	Propose a refinement calculus for requirements engineering (Care) for gathering accurate requirements	Helps to refine to create accurate unambiguous and complete requirements	Lacks proper semantics, not a cost-effective evaluation
[33]	Adding goals in the requirement negotiation process	To improve the quality of requirement negotiation	“Progress in RE activities are felt to be harder to measure, the benefits of using RE technologies are felt to be hard to measure, requirements documents are generally perceived as big, complex, outdated, and too far away from the executable products customers are paying for”

Table 2. Bibliometric analysis of requirement negotiation, risk factors, and challenges.

Ref.	Key Factors	Merits	Demerits
[4]	“The results reveal that the intuitive and quick PG technique is superior to [the] AHP technique”	Two techniques of requirement prioritization are described, which help in making requirement negotiation.	Other models are neglected, time consuming, changeable, centralized
[1]	“To motivate the need for negotiation in requirements engineering to introduce fundamental concepts and terminology, and to provide an overview about negotiation research”	“Presenting a general negotiation process, typical negotiation stages. Framework covering important dimensions of requirements negotiation comprising the conflict resolution strategy, the collaboration situation of the stakeholders, and the degree of negotiation tool support discussing and classifying existing negotiation tools using the general process and framework”	Not for large projects, not synchronized, time-consuming
[44]	“Identifies decision problem structures in software requirements negotiations using a literature-based research approach”	“Helps to select appropriate scenarios of decision problem structures in software requirements negotiations”	Not for large projects, not synchronized, time consuming
[50]	Explains requirement negotiation process and its dimensions	Helps to identify the purpose of doing requirement negotiation	Not synchronized, techniques cannot be used for every kind of system, a requirement not defined properly
[6]	To create requirement negotiation process systematic	Systematic requirements Prioritize requirements	Time-consuming
[7]	Introduces a replicable web-based framework	To create a bargaining-type negotiating process that is easier with the framework	Complex and time and efforts consuming
[74]	Concept requirement maturity is proposed to give a degree to requirements	Prioritize requirements by their given maturity level	We should use approach improvement. Stability analysis, large-scale trial usage extension
[75]	“Offer useful tools to aid stakeholders [in the] negotiation process”	Helps rank the requirements	“Option generation and negotiation planning, criteria exploration and objective assessment for option score and criteria weight, systematic post-analysis for agreement and graphical support”
[31]	“Propose a formal model of Service Level Agreement (SLA)-based negotiations”	“Allows agent-based multi-round SLA negotiation for making dynamic offers influenced by varying client requirements, different pricing models, and decision strategies”	Multiple rules, multiple providers, multiple rounds, complex, not dynamic, and customized

Table 2. Cont.

Ref.	Key Factors	Merits	Demerits
[76]	“Proposes a broker-based BS negotiation framework that can automatically determine the non-functional requirements”	Helps to determine non-functional requirements and create a difference between functional and non-functional requirements	“Integration of this work with a BS composition approach”
[78]	“The policies participating trust negotiation is modeled as a Negotiation Petri Net, and a trust negotiation MSC strategy is proposed”	To use for security requirements in which user can trust that they are the best	“Complete trust negotiation model, which includes credential and policy rules, the trust negotiation strategies, consistency detectors, etc., will be built for the project
[24]	“Easy WinWin has been successfully applied to real-client custom development projects”	Improved involvement and interaction Defined process, support for different collaboration scenarios, improved prioritization, and elaboration of issues	Not efficient with the latest technology
[27]	“Evaluated the relative effect of requirements hand-off, analysis, and negotiation on requirements and design volatility and requirements understanding”	Ease challenges in the collaboration between marketing and development	“Address the understanding of requirements communication by evaluating the product manager’s evolving requirements understanding”
[32]	“Research in distributed and mobile requirement negotiation, describe the Easy WinWin, identify issues in face-to-face negotiations”	“Overcome issues with distributed and mobile tools. Broadband face-to-face interactions [and the] distributed ARENA II tool provides sufficient support for distributed teams”	“Complete, correct, and consistent requirements, quality of the gathered requirements need to be evaluated”
[34]	“Proposed an interest-based learning activity negotiation system”	“Negotiation learning content/activity plays an important role in effective learning; learners will have more opportunities to learn in a personalized style and with adaptive content”	Needs adoption of other knowledge models such as cognitive maps, or Bayesian networks
[35]	To overcome language problems among stakeholders who are remotely negotiating, software requirements including machine translation and speech recognition are merged to generate a mobile speech translator	Overcome language gap	Results are not confirmed
[62]	An automated service negotiation framework is introduced in these parties as negotiators are involved, who negotiate to reach an agreement	Negotiators can choose different negotiation strategies according to their preference and their requirements and preference	Develop negotiation mechanism experimental service platform, which is called Safary,

Table 2. Cont.

Ref.	Key Factors	Merits	Demerits
[2]	Introduced negotiation spiral model	“Well-aligned requirements by all the stakeholders, improved system quality, a sound basis for resource estimation, and less resource wastage	“Negotiation process because of tacit individual and undeclared perspectives perceptions will lead a sub-optimal set of requirements”
[40]	Introduced Winbook with a social networking-based way of capturing and negotiating	Improve the requirement negotiation process	Time-consuming
[46]	“Make use of the basic principles of requirements negotiation and define reasoning rules to reason about stakeholder groups for requirements negotiation, based on an extended i* modeling framework”	Improve power relationships between stakeholders	Still, there are challenges that affect the power relationship between stakeholders
[43]	“Discourse is reframed as a requirements process, suggest using techniques of requirements engineering (RE) and Crowd RE, propose video as a medium for communicating problems”	“With the explicit goal to state software, hardware, and organizational requirements, getting stakeholders involved”	“Can be used directly for developing innovative software, which will speed up the implementation of a decision”
[48]	“Presenting software quality models as a framework to support the negotiation of both initial and rising requirements and the settlement of stakeholders’ concerns”	Approach supporting requirement negotiation between stakeholders and solving their conflicts	The approach did not present significant drawbacks
[36]	“How requirements negotiation collaborates to ecosystem’s health and success, defining negotiation strategies along Software Ecosystem evolution considering the Software Platform Management”	To develop improved quality software ecosystems	Not discussing the results after implementation
[21]	“An automated genetic-based approach for security negotiation among services is conducted”	Improving security, to get an acceptable security level for both sides	Securing only the web service and not covering all the conditions of web service
[41]	“Using an instrumented version of the USC WinWin groupware system”	To increase the quality of requirements	Not suitable for the latest technology-based negotiation
[47]	“Extended the WinWin Negotiation Model by incorporating sustainability concepts”	“Includes the ability to consider the impact of requirements on sustainability, [which] also allows practitioners to reflect on requirements and their effects on sustainability”	Further validate the initial findings and answer open questions
[10]	“A trust vector-based sensitive information protecting scheme is presented based on the existing schemes”	Recovers bugs, improved reliability	Not very efficient according to the advancement in technology

Table 2. Cont.

Ref.	Key Factors	Merits	Demerits
[42]	Collaborative requirements elicitation and management	WinWin negotiation framework is used to give better management in the requirement negotiation process on social media	“Could use with a relatively low learning curve and continually monitor the commitment status of the teams”
[30]	Investigate common risk factors and proposed the top ten lists of software risks	Helps in project planning and control purposes during the project execution	Repetitions in the model, not secure, complex
[37]	“Basic introduction of expert systems consisting of their composition, and basic characteristics and advantages”	“Helps to find suitable and best techniques during requirement engineering process, [which] is vital”	Techniques discussed are time-consuming, not secure
[39]	“Address the bias problem while identifying and prioritizing stakeholders”	“Identified the stakeholders and their requirements more efficaciously; prioritized the stakeholders significantly”	Not covering all the application areas
[77]	“Blockchain architecture as a new system solution to supply a reliable mechanism for secure and efficient medical record exchanges”	“Revolutionize the e-Health industry with greater efficiency by eliminating many of the intermediates”	Does not prioritize efficiently, time consuming
[51]	“Focus on the reasoning support, which is not fairly addressed by existing Semantic Wikis”	To help find requirements that are complex and different from all requirements	Cost and time effectiveness
[24]	Providing knowledge about handling risk factors occurring in software development	Strategies are provided to reduce risk factors	Strategies are provided for a comprehensive list of risk factors
[3]	Software project factors are identified, prioritized, and discussed in this research	The most important factors for software success are cost, time, and effort	Data are not gathered from latest projects success rate
[13]	Categories of risks according to their dimensions are identified, listed, and discussed	Helps to identify that budget, time, and complexity of process are the most important factors to improve	These identified risks can further be handled with the latest technologies
[12]	Factors are identified for both build versus buy software	Factors defined that need improvement in both build and buy software cases	Applicable on limited types of software
[22]	“Determines relative impact of risk factors on the software development activities”	“Relative impact ratio ensures determining the direct and indirect impacts of risk on all activities of software development”	“Risks factors can be better identified, and by using the avoidance strategies, they can be better avoided”
[26]	“A thorough handling and avoidance strategy is proposed for the identification of risk factors when the incremental model is used for software development”	“Advised strategies are expected to provide a helping hand for the avoidance or mitigation of a risk factor”	“Identified risk factors may grow in future and so can be the mitigation and avoidance strategies”
[49]	Proposes a Risk Reduction Model that minimizes risk and reduces the cost of development for large-scale projects without spending too many resources	Minimize risks, budget, resources, and time	Not applicable for small scale projects

Table 3. Bibliometric analysis of blockchain platforms and applications.

Ref.	Key Factors	Merits	Demerits
[14]	Design a high-performance blockchain platform, using technologies	Efficient connection provides higher consensus efficiency while guaranteeing the decentralization, provides higher efficiency	Repetitions in the model, not secure, complex
[53]	Discussed whether Blockchain platform can improve the performance of supply chain sustainability	Positive impacts of blockchain on three pillars of sustainability, governance, social equity and protection.	This article neither do not estimates correlations and do not finds casual inference between performance of performance of sustainability and blockchain based supply chain.
[73]	Using air-gapped wallet techniques	Most secure implementation possible, provides an upper bound on wallet security, can help the user identify the weakest parts of their air-gapped wallet scheme	Wallet analysis and quantification
[71]	Propose a compacted DAG-based blockchain protocol CoDAG that greatly improves the throughput and performance of blockchain	Improve the linear structure of traditional blockchain protocol	Tune the parameters for better performance; combine with hybrid consensus to improve the throughput further
[45]	Propose a novel hybrid blockchain crowdsourcing platform to achieve decentralization and privacy preservation	Ensure secure communication between their users and the workers, decentralization, and privacy preservation	Test the performance of the public chain as well as the interaction between the public chain and the subchains
[58]	Proposed system uses big data processing techniques	Provides a secure, robust, and scalable way to process SQL queries over any blockchain	Time delays
[16]	Proposed a blockchain for mobile communication	Secure communication	Techniques discussed are time-consuming, not secure
[79]	Discussing blockchain achievements and future work	Helps to find future work in blockchain	Other elicitation techniques can also be used, requirement validation is not used
[20]	Combining the transaction link of energy “Internet with blockchain technology”	“Can effectively solve the transaction friction, maintain the order of the market, and guarantee the legitimacy of the transaction”	Not covering all the application areas
[44]	“The principle of fast authentication with an APG-PBFT algorithm is present in this paper	Can reduce the authentication frequency	Not prioritized efficiently, time consuming
[59]	Identify the cost of verification and networking	Gives an overview of cost factors of blockchain	Repetitions in the model, not secure, complex
[64]	Presents blockchain and compares the various platforms on which blockchain can be implemented	Illustrates the use of blockchain applications for building the secure infrastructure of cloud computing	Techniques discussed are time consuming, not secure

Table 3. Cont.

Ref.	Key Factors	Merits	Demerits
[65]	Introduction and explanation of bitcoin, Brito and Castillo both support innovation, clarity	Helps to understand the process of Bitcoin and digital trading using Bitcoin	Other elicitation techniques can also be used, requirement validation is not used
[66]	“Private blockchain is chosen to solve the problem of mutual trust between users, load aggregators, and power grids in multi-level demand–response reliable communication”	Demand response automation problem will be solved easily	Not covering all the application areas
[68]	“New life cycle model of power internet financial products was proposed under the blockchain effects”	Helps to improve the quality of software projects	Not for large projects, not synchronized, time-consuming
[72]	“Propose computationally practical procedures for digital time-stamping”	“Infeasible for a user either to back-date or to forward date his document, even with the collusion of a time-stamping service”	Not synchronized, techniques are not working for every expert system, requirements not defined properly
[10]	Identification of future prospects challenges in blockchain technology	Helps to improve the quality of blockchain-based frameworks	Not identifying all the future prospects
[52]	“Propose a novel hybrid blockchain architecture that combines private and public blockchain to allow sensitive bids to be opened on a private blockchain”	“Only the auctioneer can learn the bids, and no one else, efficient in terms of run time and monetary cost”	“Create a framework for multiple blockchains hosting different applications and show how we can better preserve privacy when private, public, and consortium blockchains are combined”
[61]	“Proposes a privacy-aware PKI system based on permission BCs”	“The separation of user registration and authorization, and has the characteristics of anonymity and conditional traceability, to protect users’ identity privacy”	“Security can be further improved”
[54]	“Propose a native blockchain voting protocol for peers to vote over their existing blockchain network without the need of any trusted or third party”	“Facilitates decision making in a decentralized and secure manner, end-to-end privacy, and possesses desirable properties such as detect ability and correct ability against cheating”	“Need to perform formal security analysis, potential security attacks, such as cartel attack, should be tested against the design”
[63]	The framework is applied to malicious node detection where malicious agents are excluded from blockchain consensus	Detecting and excluding malicious nodes	Can use for consensus calculation, where agent contribution depends on their prior performances
[67]	“Propose a hybrid architecture of using both blockchain and edge nodes to impose attribute-based access control of EHR data”	“To execute smart contracts to impose ACL policy and to record legitimate access events into blockchain”	“Investigate novel consensus protocol designs for the proposed mechanism to achieve better performance”

Table 3. Cont.

Ref.	Key Factors	Merits	Demerits
[70]	“Construct a trust relationship model according to the equipment communication characteristics of intelligent manufacturing CPS system”	“Can make statistical analysis of communication data in the system, improve the concurrency of the system”	Can be further improved
[15]	“Using employment of blockchain in RE market and represent the facilities it can give to the RE market”	“Offer meaningful tools for a game-theoretic stable-priced market, process integrity, network reliability and longevity, faster transactions, and lower transaction costs”	The architecture proposed can be more flexible and simpler
[57]	“Proposed a hierarchical storage structure to store the majority of the blockchain in clouds, and maintain the most recently generated blocks in a blockchain overlay network”	“Maintain both blocks and transactions generated by the IoT networks”	“Work on the implementation of the proposed blockchain-based IoT architecture in more real IoT applications”
[69]	“Proposed an approach to increase the cost of a successful 51% double-spending attack on Proof-of-Work types of blockchain protocols”	“Utilizes the frequency rate of miners in history blocks and calculates the total Historical Weighted Difficulty to determine if a branch switch is needed, can improve smaller blockchain security drastically with easy integration”	Only applicable on Ethereum
[60]	Investigate the impact of a wide range of network latency configuration on blockchain security	Quantify blockchain security	Time spent
[56]	“Propose fast chain to scale the effective block rate of blockchain systems”	“Reduces the block propagation time through its informed neighbor selection policy, effective when nodes have different mining powers”	“Time can be reduced by more effective techniques”
[18]	“Proposed the scheme that the transactions and their history in the permissioned blockchain can be verified and concealed from the permission-less blockchain”	“Transactions and their history in a permissioned blockchain can be verified and concealed from a permission-less blockchain”	“Time consuming”
[19]	“Proposed a trust management framework for blockchain-based supply chain applications”	“Address the issue of trust associated with the quality of commodities and the entities logging data on the blockchain”	“Different network models will affect the average throughput and latency of the system”

Blockchain technology will be very helpful in the requirement negotiation process. Not only it will help developers to gather requirements from the user but also help research students and researchers to gather validated and prioritized requirements [82]. Furthermore, it will help society, because well prioritized, managed, and quality requirements will help develop quality products that will be improved, easy, efficient, and reliable for the users and will help society in efficiently fulfilling their tasks.

3. Problem Statement

Quality in requirements engineering practices creates quality software projects, so there is a need to improve the quality of requirement engineering practices for COVID-19 software projects. Requirement engineering uses many algorithms and processes to gather well-negotiated requirements. However, the requirement negotiation is a complex and time-consuming task. Although many techniques and algorithms are used to make it easy and efficient, but still, there is a lacking in the quality of COVID-19 software projects. Blockchain technology is being adopted in many fields and applications to improve the quality, security, reliability, and speed of projects. So, we can use blockchain platforms in requirement engineering practices to gather more accurate requirements and to create improved COVID-19 software projects. Blockchain-oriented requirement engineering will improve the quality and reliability of COVID-19 software projects. By using this technology in requirements engineering practices, the success rate of software projects related to COVID-19 is expected to improve.

4. Objective

The objective of this research has been studied with the help of the following sub-objectives:

1. To study the process of requirement elicitation, validation, and negotiation.
2. To study blockchain-based frameworks.
3. To study the strength and weaknesses of blockchain.

5. Research Questions

Requirement negotiation is a difficult and time-consuming process that includes requirement elicitation, validation, prioritization, and negotiation. This process of requirement negotiation can be improved by using a blockchain-based framework.

The following research questions are addressed in this research:

RQ1. What is the state of art in requirement negotiation?

RQ2. How blockchain can help improve the requirement engineering process?

6. Methodology

The purpose of requirement engineering is to gather quality requirements to develop quality software products. However, there is still a lack of quality in software development, due to which requirement negotiation is facing some challenges, and the success rate of the software is less since COVID-19. Mixed method research will be conducted to obtain validated results. First, a systematic literature review is conducted to gather related studies through which we have investigated the factors and challenges in the requirement negotiation process, and we have also identified platforms and applications of the blockchain technology. A systematic literature review includes a thorough study of the latest research papers and books to identify challenges in the requirement negotiation process. This shows that these factors need more improvement to create successful software projects for COVID-19. To validate these challenges of requirement negotiation both, quantitative and qualitative methods will be used. The quantitative method includes a survey in which a questionnaire will be distributed among the sample population. The sample population will be experienced in the field of software engineering and blockchain. In the qualitative method, a focus group meeting will be arranged in which seven to nine members will par-

ticipate. These members will be from the field of requirement engineering and blockchain technology.

A list of questions will be asked from both respondents of the survey and focus group. Half the questions will be about the challenges in the requirement negotiation process and other questions will be about the improvement in these challenges using blockchain technology. Respondents will give their reviews according to their knowledge and experience. The results of both methods (survey and focus group) will be stored separately.

After conducting the results from both qualitative and quantitative methods, a triangulation process will be applied to both the results of the qualitative and quantitative methods. The triangulation process will give the final result based on the dominant status. This will support the implementation of blockchain-based requirement engineering to create successful software for COVID-19. A structural view of the methodology is shown in Figure 2.

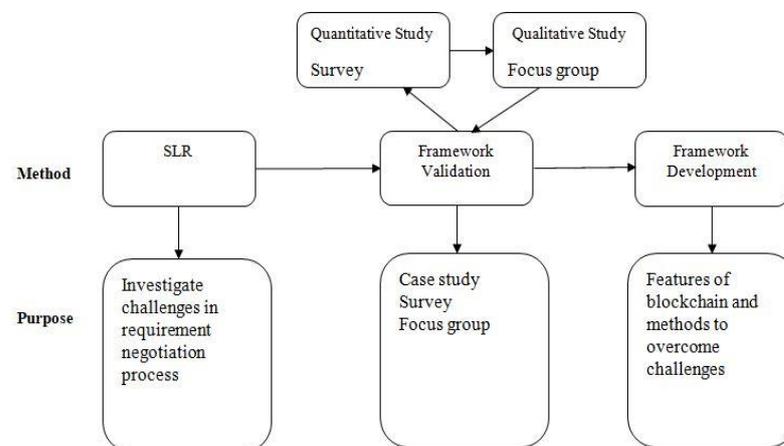


Figure 2. Methodology structural view.

7. Discussion

Factors for the improvement and reliability of requirement engineering practices for successful COVID-19 software projects are identified, and time consumption is a serious problem in requirement negotiation that can be improved using blockchain-based technology, because it offers speed in data processing. Most of the time, budgets are exceeded due to the use of all the invalid and valid requirements. The blockchain-based framework will help to automatically validate the requirements in less time. This will also help to complete the COVID-19 software projects under a specific budget, and all the invalid and unimportant requirements will be neglected, so the cost will be applied only on valid requirements. During the requirement negotiation process, conflicts between stakeholders are a common problem that takes more time, cost, and effort. Using blockchain in this process will automatically validate the requirements using its voting algorithms, and it will reduce the conflicts between stakeholders. Decision-making problems will also be solved because of the voting algorithms of a blockchain framework. The quality of software projects will improve because the blockchain framework will produce validated and prioritized requirements. Negotiation through social networking is still facing problems, and it lacks quality. Therefore, blockchain implementation will help all the stakeholders vote on the requirements at a specific time. It will decrease the time consumption and ambiguous requirements, and the voting of non-technical stakeholders will also be considered using specific voting. This will improve the negotiation using blockchain-oriented technology; meanwhile, inconsistent requirements also cause the failure of projects, so the consistency of requirements can be improved using the blockchain-based platform. The power relationships between stakeholders will be improved, requirement validation and prioritization will also be improved due to the voting algorithms in a blockchain-based

system. These factors can be improved using innovative blockchain-based technology for the reliability of COVID-19 software, which is now being adopted in every field of life. Through using this technology in requirement engineering practices, most of the listed factors will be improved, and the success rate of COVID-19 software projects will increase.

The blockchain-based framework will consist of three main components: a requirement pool, distributed ledger, and decentralized network of stakeholders. A decentralized feature of blockchain will be used, which will contain a decentralized ledger. This decentralized, shared ledger will store all the records of stakeholders. The records will be stored automatically as the stakeholders enter data. Once data are entered, the record will not be changed or tempered. The management will also be a part of the network that gives access to authorized stakeholders. Once the requirement negotiation process is done on a specific project, the management will restrict the access to the user or customer of the current project and give access to the customer of the new project. It will not be an open-source project. It will be secured, and the results of this framework will be validated by taking votes on requirements from stakeholders.

The use of the blockchain-based platform in requirement engineering will help to improve the methodological and operational process of requirement negotiation. The improvement in the requirement negotiation process will help the developers, management team, and stakeholders to complete the requirement negotiation process with improved quality and within a specific time frame and budget. The improved challenges of the requirement negotiation process are listed in Table 4.

Table 4. Blockchain methodological contribution in facilitating technology for COVID-19 software.

No.	Blockchain Methodological Contribution
1.	Help to gather correct complete requirements
2.	Requirement engineering processes will be secure
3.	Requirement engineering processes will be easily accessible
4.	Will reduce their time and efforts of developers with improved quality of requirements
5.	Will ensure consistency and synchronization of requirements
6.	The failure rate of projects will be reduced
7.	The rate of successful projects will be increased

The implementation of the blockchain-based framework will help researchers, scientists, academia, and students in many ways. The improvement in factors will help them fulfill their tasks on time and in an effective way. These improvements are given in Table 5.

Table 5. Blockchain research contribution in facilitating technology for COVID-19 software.

No.	Blockchain Research Contribution
1.	Researchers will easily negotiate for data elicitation and validation
2.	Students will further use blockchain for better performance of methods and algorithms used in software engineering
3.	The quality of the requirement negotiation process will be improved
4.	Help researchers to gather quality data.
5.	Requirements will be prioritized according to their importance and need
6.	Requirements will be validated

8. Conclusions

The global pandemic COVID-19 environment needs the whole world to work from home through different software. The quality of the project involves how well its require-

ments were gathered and fulfilled. Many techniques and algorithms are used to make the requirement negotiation process easier and efficient, but still, there is a lacking in the quality of projects. Requirement negotiation is a complex and prolonged task. To improve the quality of COVID-19 software projects, the requirements must be gathered accurately and completely. For this purpose, we can use blockchain platforms in requirement engineering practices to make the requirement negotiation process more efficient. The innovation of blockchain is a revolution in frameworks of record. The idea of blockchain innovation is spreading widely because the concept can now be applied to any requirement for trustworthy records. Some applications of blockchain are digital identity and tokenization. Therefore, the purpose of our research is to identify the factors that can be improved and reliable using blockchain-based technology in requirement engineering for COVID-19 software projects. Factors that are identified for improvement in requirement engineering practices using blockchain technology are quality, time, social networks, conflicts in decision making, stakeholder relationships, negotiation process, validation, and consistency of requirements. All these factors are important for producing quality software requirements for COVID-19. We can use blockchain platforms to make these factors of requirement engineering practices more reliable and effective. This will help to create successful projects that will reduce the rate of failure of projects of COVID-19.

Author Contributions: Conceptualization, B.S. and I.J.; methodology, A.S. (Asadullah Shaikh), A.S. (Adel Sulaiman) and A.A.; formal analysis, B.S, I.J. and A.S. (Asadullah Shaikh); software, validation, Writing—Original draft preparation, M.A.M. and A.A.; Writing—Review and editing, B.S., I.J., A.S. (Asadullah Shaikh) and A.A.; supervision, project administration, and funding acquisition, A.A. and A.S. (Adel Sulaiman). All authors have read and agreed to the published version of the manuscript.

Funding: The authors would like to acknowledge the support of the Deputy for Research and Innovation—Ministry of Education, Kingdom of Saudi Arabia for this research through a grant (NU/IFC/INT/01/008) under the institutional Funding Committee at Najran University, Kingdom of Saudi Arabia.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Grünbacher, P.; Seyff, N. Requirements Negotiation. In *Engineering and Managing Software Requirements*; Springer Science and Business Media LLC: Berlin/Heidelberg, Germany, 2006; pp. 143–162.
2. Ahmad, S. Negotiation in the Requirements Elicitation and Analysis Process. In Proceedings of the 19th Australian Conference on Software Engineering (aswec 2008), Perth, WA, Australia, 26–28 March 2008; pp. 683–689.
3. Shahzad, B.; Said, A.M. Identification and Quantitative Analysis of Project Success Factors for Large Scale Projects. *Int. J. Knowl. Soc. Res.* **2014**, *5*, 83–95. [[CrossRef](#)]
4. Siddiqui, S.; Beg, M.; Fatima, S. Effectiveness of requirement prioritization using analytical hierarchy process (AHP) and planning game (PG): A comparative study. *Int. J. Comput. Sci. Inf. Technol.* **2013**, *4*, 46–49.
5. Aurum, A.; Wohlin, C. *Engineering and Managing Software Requirements*; Springer: Berlin/Heidelberg, Germany, 2005.
6. Kim, J.B.; Segev, A. A framework for dynamic eBusiness negotiation processes. In Proceedings of the IEEE International Conference on E-Commerce, Newport Beach, CA, USA, 24–27 June 2003; pp. 84–91.
7. Su, S.Y.W.; Huang, C.; Hammer, J. A replicable Web-based negotiation server for e-commerce. In Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, Maui, HI, USA, 7 January 2000.
8. Saeed, M.S.; Sarwar, N.; Bilal, M. Efficient requirement engineering for small scale project by using UML. In Proceedings of the 2016 Sixth International Conference on Innovative Computing Technology (INTECH), Dublin, Ireland, 24–26 August 2016.
9. Riaz, M.Q.; Rehman, F.U.; Maqbool, B.; Butt, W.H. Customization of requirement engineering best practices for Pakistan software industry. In Proceedings of the 2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), Sukkur, Pakistan, 3–4 March 2018.

10. Lei, J.-Y.; Zhang, B.-C.; Fang, X.-H. Trust vector-based sensitive information protecting scheme in automatic trust negotiation. In Proceedings of the 2011 International Conference on Computer Science and Network Technology, Harbin, China, 24–26 December 2011; Volume 2, pp. 735–738.
11. Anwer, S.; Ikram, N. Goal Oriented Requirement Engineering: A Critical Study of Techniques. In Proceedings of the 2006 13th Asia Pacific Software Engineering Conference (APSEC'06), Bangalore, India, 6–8 December 2006; pp. 121–130.
12. Shahzad, B.; Abdullatif, A.M.; Ikram, N.; Mashkoo, A. Build Software or Buy: A Study on Developing Large Scale Software. *IEEE Access* **2017**, *5*, 24262–24274. [[CrossRef](#)]
13. Shehzad, B.; Awan, K.M.; Lali, M.; Aslam, W. Identification of patterns in failure of software projects. *J. Inf. Sci. Eng.* **2017**, *33*, 1465–1479.
14. Yu, S.; Lv, K.; Shao, Z.; Guo, Y.; Zou, J.; Zhang, B. A High Performance Blockchain Platform for Intelligent Devices. In Proceedings of the 2018 1st IEEE International Conference on Hot Information-Centric Networking (HotICN), Shenzhen, China, 15–17 August 2018; pp. 260–261. [[CrossRef](#)]
15. Latifi, S.; Zhang, Y.; Cheng, L.-C. Blockchain-Based Real Estate Market: One Method for Applying Blockchain Technology in Commercial Real Estate Market. In Proceedings of the 2019 IEEE International Conference on Blockchain (Blockchain), Atlanta, GA, USA, 14–17 July 2019; pp. 528–535.
16. Chen, L.; Li, Y.; Wen, H.; Lei, W.; Hou, W.; Chen, J. Block Chain Based Secure Scheme For Mobile Communication. In Proceedings of the 2018 IEEE Conference on Communications and Network Security (CNS), Beijing, China, 30 May–1 June 2018; pp. 1–2.
17. Makridakis, S.; Polemitis, A.; Giaglis, G.; Loucal, S. Blockchain current achievements and future prospects challenges combining AIBC. *Future Internet* **2019**, *11*, 258. [[CrossRef](#)]
18. Mitani, T.; Otsuka, A. Traceability in permissioned blockchain. In Proceedings of the 2019 IEEE International Conference on Blockchain (Block-chain), Atlanta, GA, USA, 14–17 July 2019; pp. 286–293. [[CrossRef](#)]
19. Malik, S.; Dedeoglu, V.; Kanhere, S.S.; Jurdak, R. TrustChain: Trust Management in Blockchain and IoT Supported Supply Chains. In Proceedings of the 2019 IEEE International Conference on Blockchain (Blockchain), Atlanta, GA, USA, 14–17 July 2019; pp. 184–193.
20. Peiyu, X.; Qian, Z.; Haining, W.; Haoyue, Z.; Chunyan, W. Exploration of Block chain Technology in Electric Power transaction. In Proceedings of the 2018 International Conference on Power System Technology (POWERCON), Guangzhou, China, 6–8 November 2018; Volume 2, pp. 729–733.
21. Abdelatey, A.; Elkawkagy, M.; El-Sisi, A.B.; Keshk, A. RGSS-negotiation: A genetic-based approach for web service security negotiation. In Proceedings of the 2016 11th International Conference on Computer Engineering & Systems (ICCES), Cairo, Egypt, 20–21 December 2016; pp. 53–58. [[CrossRef](#)]
22. Shahzad, B.; Afzal, T.; Irfan, R. Enhanced Risk Analysis-Relative Impact Factorization. In Proceedings of the 2005 International Conference on Information and Communication Technologies, Karachi, Pakistan, 27–28 August 2005.
23. Elrakaiby, Y.; Ferrari, A.; Mylopoulos, J. CaRE: A refinement calculus for requirements engineering based on argumentation semantics. In Proceedings of the 2018 IEEE 26th International Requirements Engineering Conference (RE), Banff, AB, USA, 20–24 August 2018; pp. 364–369. [[CrossRef](#)]
24. Gruenbacher, P. Collaborative requirements negotiation with EasyWinWin. In Proceedings of the 11th International Workshop on Database and Expert Systems Applications, London, UK, 4–8 September 2000; pp. 954–958.
25. Mazón, J.-N.; Pardo, J.; Trujillo, J. A model-driven goal-oriented requirement engineering approach for data ware-houses. *Adv. Concept. Model.* **2007**, *4802*, 255–264. [[CrossRef](#)]
26. Shahzad, B.; Ullah, I.; Khan, N. Software Risk Identification and Mitigation in Incremental Model. In Proceedings of the 2009 International Conference on Information and Multimedia Technology, Jeju, Korea, 16–18 December 2009; pp. 366–370.
27. Fricker, S.; Glinz, M. Comparison of Requirements Hand-off, Analysis, and Negotiation: Case Study. In Proceedings of the 2010 18th IEEE International Requirements Engineering Conference, Sydney, NSW, Australia, 27 September–1 October 2010; pp. 167–176.
28. Shahzad, B.; Al-Wakeel, S. Software risk management and avoidance strategy. In Proceedings of the 2009 International Conference on Machine Learning and Computing, Perth, Australia, 10–12 July 2009; Volume 3.
29. Nuseibeh, B.; Easterbrook, S.M. Requirements engineering: A roadmap. *Proc. Conf. Future Softw. Eng.* **2000**, *1*, 35–46.
30. Arnuphaptrairong, T. Top ten lists of software project risks: Evidence from the literature survey. *Int. Multi-Conference Eng. Comput. Sci.* **2011**, *1*, 732–737.
31. Baig, R.; Khan, W.A.; Haq, I.U.; Khan, I.M. Agent-Based SLA Negotiation Protocol for Cloud Computing. In Proceedings of the 2017 International Conference on Cloud Computing Research and Innovation (ICCCRI), Singapore, 11–12 April 2017; pp. 33–37.
32. Seyff, N.; Hoyer, C.; Krohler, E.; Grünbacher, P. Enhancing GSS-based Requirements Negotiation with Distributed and Mobile Tools. In Proceedings of the 14th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprise (WETICE'05), Linköping, Sweden, 13–15 June 2005; pp. 87–92.
33. van Lamsweerde, A. Goal-oriented requirements engineering: A roundtrip from research to practice. In Proceedings of the 12th IEEE International Requirements Engineering Conference, Kyoto, Japan, 10 September 2004; pp. 3–6. [[CrossRef](#)]
34. Tao, X.; Miao, Y. Interest Based Learning Activity Negotiation. In Proceedings of the 2008 International Conference on Cyber-worlds, Hangzhou, China, 22–24 September 2008; pp. 58–64.

35. Calefato, F.; Lanubile, F.; Romita, D.; Prikladnicki, R.; Pinto, J.H.S. Mobile speech translation for multilingual requirements meetings: A preliminary study. In Proceedings of the 2014 IEEE 9th International Conference on Global Software Engineering, Shanghai, China, 18–21 August 2014; pp. 145–152. [[CrossRef](#)]
36. Valenca, G. Requirements negotiation model: A social oriented approach for software ecosystems evolution. In Proceedings of the 2013 21st IEEE International Requirements Engineering Conference (RE), Rio de Janeiro-RJ, Brazil, 15–19 July 2013; pp. 393–396.
37. Agarwal, M.; Goel, S. Expert system and its requirement engineering process. In Proceedings of the International Conference on Recent Advances and Innovations in Engineering (ICRAIE-2014), Jaipur, India, 9–11 May 2014; pp. 1–4. [[CrossRef](#)]
38. Pandey, D.; Suman, U.; Ramani, A. An Effective Requirement Engineering Process Model for Software Development and Requirements Management. In Proceedings of the 2010 International Conference on Advances in Recent Technologies in Communication and Computing, Kottayam, India, 16–17 October 2010; pp. 287–291.
39. Mughal, S.; Abbas, A.; Ahmad, N.; Khan, S.U. A Social Network Based Process to Minimize In-Group Biasedness During Requirement Engineering. *IEEE Access* **2018**, *6*, 66870–66885. [[CrossRef](#)]
40. Kukreja, N.; Boehm, B. Process implications of social networking-based requirements negotiation tools. In Proceedings of the 2012 International Conference on Software and System Process (ICSSP), Zurich, Switzerland, 2–3 June 2012; pp. 68–72.
41. Boehm, B.; Egyed, A. Software requirements negotiation: Some lessons learned. In Proceedings of the 20th International Conference on Software Engineering, Kyoto, Japan, 6 August 2002; pp. 503–506.
42. Kukreja, N. Winbook: A social networking based framework for collaborative requirements elicitation and WinWin negotiations. In Proceedings of the 2012 34th International Conference on Software Engineering (ICSE), Zurich, 2012; pp. 1610–1612. [[CrossRef](#)]
43. Schneider, K.; Karras, O.; Finger, A.; Zibell, B. Reframing societal discourse as requirements negotiation: Vision statement. In Proceedings of the 2017 IEEE 25th International Requirements Engineering Conference Workshops (REW), Lisbon, Portugal, 4–8 September 2017; pp. 188–193. [[CrossRef](#)]
44. Lenz, A.; Schoop, M. Decision Problems in Requirements Negotiations—Identifying the Underlying Structures. *Bus. Inf. Syst.* **2017**, *293*, 120–131.
45. Zhu, S.; Hu, H.; Li, Y.; Li, W. Hybrid Blockchain Design for Privacy Preserving Crowdsourcing Platform. In Proceedings of the 2019 IEEE International Conference on Blockchain (Blockchain), Atlanta, GA, USA, 14–17 July 2019; pp. 26–33.
46. Yang, H.; Liang, P. Reasoning about Stakeholder Groups for Requirements Negotiation Based on Power Relationships. In Proceedings of the 2013 20th Asia-Pacific Software Engineering Conference (APSEC), Bangkok, Thailand, 2–5 December 2013; Volume 1, pp. 247–254.
47. Seyff, N.; Betz, S.; Duboc, L.; Venters, C.; Becker, C.; Chitchyan, R.; Penzenstadler, B.; Nobauer, M. Tailoring Requirements Negotiation to Sustainability. In Proceedings of the 2018 IEEE 26th International Requirements Engineering Conference (RE), Banff, AB, Canada, 20–24 August 2018; pp. 304–314.
48. Carvallo, J.P.; Franch, X. Requirements negotiation for multilayer system components. In Proceedings of the 2011 IEEE 19th International Requirements Engineering Conference, Washington, DC, USA, 29 August–2 September 2011; pp. 285–290.
49. Bakhsh, S.T.; Shahzad, B.; Tahir, S. Risk management approaches for large scale software development. *Inf. Sci. Eng.* **2017**, *33*, 1547–1560. [[CrossRef](#)]
50. Fernandes, J.M.; Machado, R.J. Requirements negotiation. *Requir. Eng. Proj.* **2016**. [[CrossRef](#)]
51. Liang, P.; Avgeriou, P.; Clerc, V. Requirements Reasoning for Distributed Requirements Analysis Using Semantic Wiki. In Proceedings of the 2009 Fourth IEEE International Conference on Global Software Engineering, Limerick, Ireland, 13–16 July 2009; pp. 388–393.
52. Desai, H.; Kantarcioglu, M.; Kagal, L. A Hybrid Blockchain Architecture for Privacy-Enabled and Accountable Auctions. In Proceedings of the 2019 IEEE International Conference on Blockchain (Blockchain), Atlanta, GA, USA, 14–17 July 2019; pp. 34–43.
53. Park, A.; Li, H. The Effect of Blockchain Technology on Supply Chain Sustainability Performances. *Sustainability* **2021**, *13*, 1726. [[CrossRef](#)]
54. Zhang, W.; Yuan, Y.; Hu, Y.; Huang, S.; Cao, S.; Chopra, A.; Huang, S. A Privacy-Preserving Voting Protocol on Blockchain. In Proceedings of the 2018 IEEE 11th International Conference on Cloud Computing (CLOUD), San Francisco, CA, USA, 2–7 July 2018; pp. 401–408.
55. Chen, Z.; Chen, S.; Xu, H.; Hu, B. A Security Authentication Scheme of 5G Ultra-Dense Network Based on Block Chain. *IEEE Access* **2018**, *6*, 55372–55379. [[CrossRef](#)]
56. Wang, K.; Kim, H.S. FastChain: Scaling Blockchain System with Informed Neighbor Selection. In Proceedings of the 2019 IEEE International Conference on Blockchain (Blockchain), Atlanta, GA, USA, 14–17 July 2019; pp. 376–383.
57. Wang, G.; Shi, Z.; Nixon, M.; Han, S. ChainSplitter: Towards Blockchain-Based Industrial IoT Architecture for Supporting Hierarchical Storage. In Proceedings of the 2019 IEEE International Conference on Blockchain (Blockchain), Atlanta, GA, USA, 14–17 July 2019; pp. 166–175.
58. Linoy, S.; Mahdikhani, H.; Ray, S.; Lu, R.; Stakhanova, N.; Ghorbani, A. Scalable Privacy-Preserving Query Processing over Ethereum Blockchain. In Proceedings of the 2019 IEEE International Conference on Blockchain (Blockchain), Atlanta, GA, USA, 14–17 July 2019; pp. 398–404.
59. Catalini, C.; Gans, J.S. Some simple economics of the blockchain. *Commun. ACM* **2020**, *63*, 80–90. [[CrossRef](#)]
60. Wan, L.; Eysers, D.; Zhang, H. Evaluating the Impact of Network Latency on the Safety of Blockchain Transactions. In Proceedings of the 2019 IEEE International Conference on Blockchain (Blockchain), Atlanta, GA, USA, 14–17 July 2019; pp. 194–201.

61. Wang, R.; He, J.; Liu, C.; Li, Q.; Tsai, W.-T.; Deng, E. A Privacy-Aware PKI System Based on Permissioned Blockchains. In Proceedings of the 2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS), Beijing, China, 23–25 November 2018; pp. 928–931.
62. Xu, T.; Li, T.; Liu, L.; Bryant, B.R. Negotiating Service Requirements among Strategic Actors. In Proceedings of the 2010 IEEE 34th Annual Computer Software and Applications Conference Workshops, Seoul, Korea, 19–23 July 2010; pp. 31–36.
63. Salman, T.; Jain, R.; Gupta, L. A reputation management framework for knowledge-based and probabilistic block-chains. In Proceedings of the 2019 IEEE International Conference on Blockchain (Blockchain), Atlanta, GA, USA, 14–17 July 2019; pp. 520–527. [[CrossRef](#)]
64. Sharma, S.G.; Ahuja, L.; Goyal, D.P. Building Secure Infrastructure for Cloud Computing Using Blockchain. In Proceedings of the 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 14–15 June 2018; pp. 1985–1988. [[CrossRef](#)]
65. Brito, J.; Castillo, A. Summary for policymakers. *Clim. Change* **2013**, *29*, 1–30. [[CrossRef](#)]
66. Cui, G.; Shi, K.; Qin, Y.; Liu, L.; Qi, B.; Li, B. Application of block chain in multi-level demand response reliable mechanism. In Proceedings of the 2017 3rd ICIM, Chengdu, China, 21–23 April 2017; pp. 337–341. [[CrossRef](#)]
67. Guo, H.; Li, W.; Nejad, M.; Shen, C.-C. Access control for electronic health records with hybrid blockchainedge architecture. In Proceedings of the 2019 IEEE International Conference on Blockchain (Blockchain), Atlanta, GA, USA, 14–17 July 2019; pp. 44–51. [[CrossRef](#)]
68. Nakasumi, M. Information sharing for supply chain management based on blockchain technology. In Proceedings of the IEEE 19th Conference on Business Informatics, Thessaloniki, Greece, 24–27 July 2017. [[CrossRef](#)]
69. Yang, X.; Chen, Y.; Chen, X. Effective Scheme against 51% Attack on Proof-of-Work Blockchain with History Weighted Information. In Proceedings of the 2019 IEEE International Conference on Blockchain (Blockchain), Atlanta, GA, USA, 14–17 July 2019; pp. 261–265.
70. Li, S.; Xiao, H.; Wang, H.; Wang, T.; Qiao, J.; Liu, S. Blockchain Dividing based on node community clustering in intelligent manufacturing CPS. In Proceedings of the 2019 IEEE International Conference on Blockchain (Blockchain), Atlanta, GA, USA, 14–17 July 2019; pp. 124–131. [[CrossRef](#)]
71. Yang, S.; Chen, Z.; Cui, L.; Xu, M.; Ming, Z.; Xu, K. CoDAG: An Efficient and Compacted DAG-Based Blockchain Protocol. In Proceedings of the 2019 IEEE International Conference on Blockchain (Blockchain), Atlanta, GA, USA, 14–17 July 2019; pp. 314–318.
72. Stuart, H.; Scott, S.W. How to time-stamp a digital document. *J. Cryptol.* **1991**, *3*, 12.
73. Davenport, A.; Shetty, S. Air Gapped Wallet Schemes and Private Key Leakage in Permissioned Blockchain Platforms. In Proceedings of the 2019 IEEE International Conference on Blockchain (Blockchain), Atlanta, GA, USA, 14–17 July 2019; pp. 541–545.
74. Peng, R.; Ye, Q.; Ye, M. A requirements maturity measurement approach based on SKLSE Wiki. In Proceedings of the 2010 IEEE 34th Annual Computer Software and Applications Conference Workshops, Seoul, Korea, 19–23 July 2010; pp. 251–254. [[CrossRef](#)]
75. In, H.; Olson, D.; Rodgers, T. A requirements negotiation model based on multi-criteria analysis. In Proceedings of the Fifth IEEE International Symposium on Requirements Engineering, Toronto, ON, Canada, 7 August 2000; pp. 312–313. [[CrossRef](#)]
76. Dubois, E.; Kritikos, K.; Kubicki, S. An Automatic Requirements Negotiation Approach for Business Services. In Proceedings of the 2011 IEEE Ninth European Conference on Web Services, Lugano, Switzerland, 14–16 September 2011; pp. 133–140.
77. Miller, T.W. Do family risk factors influence attention deficit hyperactivity disorder via disruption of neurocognitive functioning? *Diss. Abstr. Int. Sect. B Sci. Eng.* **2010**, *70*, 6560. [[CrossRef](#)]
78. He, Y.; Zhu, M.; Zheng, C. An efficient and minimum sensitivity cost negotiation strategy in automated trust negotiation. In Proceedings of the 2008 International Conference on Computer Science and Software Engineering, Wuhan, China, 12–14 December 2008; pp. 182–185. [[CrossRef](#)]
79. Makridakis, S.; Polemitis, A.; Giaglis, G.; Louca, S. *Blockchain: Current Achievements, Future Prospects/Challenges and Its Combination with AI*; Faculty University of Nicosia, Members of the Blockchain/AI Team: Cyprus, Nicosia, 2017; pp. 1–21.
80. Iqbal, N.; Sang, J. Fuzzy Logic Testing Approach for Measuring Software Completeness. *Symmetry* **2021**, *13*, 604. [[CrossRef](#)]
81. da Silva, A.; Savić, D. Linguistic Patterns and Linguistic Styles for Requirements Specification: Focus on Data Entities. *Appl. Sci.* **2021**, *11*, 4119. [[CrossRef](#)]
82. Yang, C.; Wu, L.; Yu, C.; Zhou, Y. A Phrase-Level User Requests Mining Approach in Mobile Application Reviews: Concept, Framework, and Operation. *Information* **2021**, *12*, 177. [[CrossRef](#)]