



ISSN: 2814-1709

CTICTR 4(1): 103 - 115 (June 2025)

Received: 08-03-2025

Accepted: 07-06-2025

<https://doi.org/10.61867/pcub.v3i1a.202>

ELICITING USER REQUIREMENTS FOR SENSITIVE DATA PROTECTION MODEL FOR ONLINE SOCIAL NETWORKS

Idepefo Felix Olutokunbo

Aderibigbe Stephen Ojo

Awoniyi Amos Tolulope

¹felixidepefo@gmail.com

School of Computing, Babcock University, Ilishan-Remo, Ogun State, Nigeria.

Eliciting User Requirements for Sensitive Data Protection Model for Online Social Networks

Idepefo Felix Olutokunbo¹, Aderibigbe Stephen Ojo², Awoniyi Amos Tolulope³

¹ Computer Science Department, Babcock University, Ilishan-Remo, Ogun State

² Computer Science Department, Lagos State University of Science & Technology, Ikorodu, Lagos State

³ Department of Software Engineering, Babcock University, Ilishan-Remo, Ogun State

¹ felixidepefo@gmail.com

^{1,3} *School of Computing, Babcock University, Ilishan-Remo, Ogun State, Nigeria.*

Abstract

Online Social Networks (OSNs) have become a significant communication platform where users invest considerable time to share personal information and lifestyles. However, the widespread use of OSNs is often matched by serious privacy concerns, as evidenced by various recent scandals and data breaches that have shown their vulnerabilities. Identifying proper requirements for OSNs during the initial stages of software development will help mitigate the issues associated with requirements determination, which have been recognized as a primary factor in the failure of software projects. It will also alleviate the adverse consequences of incomplete requirements, such as project delays, cancellations, and the release of substandard software products. This paper aims to outline both the functional and non-functional requirements of the Sensitive Data Protection Models, which leverage Blockchain technology in conjunction with cryptographic methods, hashing algorithms, and consensus mechanisms to protect users' sensitive data. The proposed model shall guarantee that users maintain complete control over their data while ensuring privacy, trust, and data availability among untrustworthy peers on OSNs. This research is ongoing, and as part of future work, the prototype model is being developed for deployment within a real social network setting.

Keywords: Requirement Engineering, Sensitive Data Protection Model, Online Social Network, Blockchain Technology, Data breaches.

1.0 Introduction

The Internet has completely evolved in recent years and has become an inevitable part of human lives, offering the opportunity to interact digitally with billions of users worldwide [1], [2]. The current globalization has consequently made the internet more social, and OSNs platforms such as Facebook, Twitter, LinkedIn, and Instagram have become part of the real life of people [2]. People use OSNs to share their personal information as a daily activity. Today, social media users are about three (3) billion worldwide [3]. At the edge of these technology expansions, it seems that social media has become a platform for individuals or organizations to create, share, and exchange information and ideas [4]. OSNs are therefore huge in scale and have been predicted to keep growing both in the number of users and the amount of data uploaded and shared by users [5]. For instance, about 40 items of personal information are exposed when a user fills in information on the Facebook apps and this

provides Facebook enough information about who those user is and the people they are identified with [6].

The numerous cases of privacy disclosures and other data breaches make it clear that proper privacy preservation measures are needed [5]. Additionally, many of OSNs are still built on a centralized architecture, which gives the service provider (like Facebook) control over users' private and potentially sensitive information about their lifestyle, behaviors, and interactions [7]. The recent Cambridge Analytical data breach, in which 87 million users' personal information was stolen from the Facebook site for a political advertising campaign, is a prime example of privacy disclosure by OSNs service providers [8], [3]. Other issues of contemporary online social platforms include censorship, surveillance, and information leaks [9]. Facebook has been banned in some countries, such as China, Tunisia, Iran, and so on, only to mention a specific case, because of their online censorship and surveillance of instant communications [3]. To address these issues, several proposals have suggested the decentralization of social data so that control will entirely be in the hands of the owners of the data. However, this approach (though it provides some solutions) has not been able to provide adequate solutions [10],[11]. Therefore, it is essential to create a Sensitive Data Protection Model (SDPM) that integrates Blockchain technology with cryptographic methods, the hashing algorithm, and a consensus mechanism to ensure privacy, foster trust, and empower users regarding their data in environments with untrusted peers on Online Social Networks (OSNs).

Determining the functional and non-functional needs of the model is the first step in creating an SDPM that will accomplish the aforementioned objective. Requirements engineering (RE) is the process of identifying requirements, and it is used to capture software requirements [4]. Identification, modeling, communication, and documentation of a system's needs as well as the context in which it will be utilized are the focus of requirement engineering [12]. In requirements engineering (RE), the process is not just a checklist but a deliberate set of actions that guide the creation, verification, and refinement of a system's requirements specification [12]. Getting these requirements right is one of the hardest and most important parts of the software development lifecycle [13]. When requirements are poorly managed, projects often run into trouble as delays, cancellations, or incomplete products are common outcomes [12], [13]. Requirements are usually split into two main groups: FR and non NFR [16]. Functional requirements describe the operations or tasks a system should perform, without being restricted by physical limitations [17]. They focus on what the system must do, how it should react to different inputs, and the way it is expected to behave under certain conditions [4], [16].

2.0 Literature Review

Requirements engineering (RE) is widely regarded as the most critical stage of the software development life cycle (SDLC) [17]. Well-defined and precise requirements are fundamental to the successful design and implementation of information technology (IT) projects [18]. RE serves to convert vague and incomplete requirements and user needs into thorough, clear, and formal specifications [19]. The primary objective of RE is to ensure end-user satisfaction while minimizing cost and time [20]. Requirements can be categorized into FRs and NFRs [14]. The activities involved in RE include elicitation, analysis and negotiation, documentation, and validation [18]. Numerous studies in the literature highlight the significance of FRs and NFRs throughout the software development process. A qualitative study by [21] explored and detailed the most vital FRs and NFRs of the knowledge sharing system (KSS) employed in Malaysian public academic institutions. Semi-structured interviews were used to gather data, and fifteen distinguished academicians were interviewed. To extract the needs from the qualitative data, the content analysis method was employed. The study found twelve FRs and NFRs that academicians think should be in the KSS since they are end users. Sedelmaier and Landes [22] developed a thorough requirements engineering competency

profile by combining qualitative and quantitative research methodologies. In addition to improving future requirements engineers' education in handling the problems presented by digitalization, the resulting skill profile advances the understanding of requirements engineering competencies.

Curcio et al. [11] conducted a comprehensive examination aimed at mapping the field of requirements engineering within an agile context to highlight key topics that have been investigated while pinpointing gaps for future research development. For their review, they selected, analyzed, and evaluated 104 publications from 2001 to 2017. The results indicate a significant need for concentrated attention on under-explored aspects such as requirement elicitation, FRs and NFRs, sources of requirements, change management, measurement of requirements, software requirements tools, and comparative studies. In the research presented in [23], a framework was developed to assess the importance of project management in the contexts of requirement engineering (RE) and requirement change management (RCM). The researchers collected data through surveys and interviews, utilizing Likert scale statistical methods to analyze the responses. The findings indicated that implementing a phase-oriented specialized project management approach in both frameworks results in enhanced RE and RCM activities, contributing to a higher quality product and improved project success rates. Additionally, the study highlighted the significance of thorough and accurate requirements gathering and management within the software development lifecycle (SDLC). Lindoerfer and Mansmann [24] introduced an evidence-based checklist for compiling requirements for patient registry software systems (CIPROS). By employing a systematic literature review to compile a thorough collection of technical concepts, conducting a qualitative content analysis to establish a catalog of relevant criteria, and creating a checklist to develop a minimal evaluation standard, they successfully formulated the evidence-based CIPROS checklist. The findings suggest that the CIPROS checklist enhances the clarity of patient registry software system descriptions while aiding developers in identifying their system requirements and appraising existing systems.

An investigation into the requirements engineering process in software development outsourcing was carried out in [25] to identify and rank frequently encountered issues. Three questionnaire surveys were organized with the assistance of experienced software development outsourcing practitioners, who also conducted an extensive review of existing literature to determine the challenges faced in practice. The study employed the 50% rule, the cut off value method, and the Delphi procedure. Through category wise ranking and overall rating, the researchers classified forty-three commonly occurring concerns within each area. The findings indicate that the identification and prioritization of issues aid in the creation of a proactive software project management strategy that helps address software development outsourcing failures and achieve the anticipated advantages of software development outsourcing. Wu *et al.* [4] identify the pertinent FRs and NFRs for knowledge sharing (KS), investigating the connections between the requirements and the perception of social media as a tool for KS support. The needs on KS were gathered by a survey of software requirements practitioners, hypothesis and research model were assessed using the partial least squares (PLS) approach. Functional needs and perceived social media to promote requirements knowledge sharing (RKS) were shown to be positively and strongly correlated, while NFRs and perceived SM to support RKS were found to be weakly correlated. Hidellaarachchi *et al.* [26] carefully assessed original studies that looked at how various human elements impact the RE process. A thorough survey of the literature turned up 474 preliminary primary research studies. In the end, they were condensed to 74 outstanding, relevant primary studies. The findings indicate that it is advantageous to look into several human factors at once, since this shows the connections between them and how they together affect the RE process.

Gathering requirements is a fundamental phase in the software engineering process, as highlighted in [27] and [28]. The method for gathering requirements in Business Intelligence (BI) projects was detailed in [29]. The proposed procedure outlines various steps and tasks, their interconnections, a range of input and output documents, and the most effective methods for gathering requirements in BI initiatives. By reducing the disparity between what the development team produces and what the users expect, the research aims to ensure that requirements for BI projects are more relevant to the concerns of stakeholders. Furthermore, several studies have indicated that the requirements elicitation process frequently overlooks NFRs [29], [30]. In [30], citizen initiated platforms were framed as NFRs to explore how contextual factors in the relationship between citizens and government can be better understood by implementers. Following an extensive literature review, a survey involving 938 potential users was conducted to identify contextual factors influencing citizen to government (C2G) adoption. To formalize these factors as NFRs, a soft goal interdependence graph (SIG) was employed, and logistic regression was applied for data analysis. The results showed that the most influential factors were “citizens’ concerns regarding city conditions” and “the influence of users’ perceived contributions.” Natsiavas et al. [31] sought to establish a basis for developing a secure and interoperable toolkit for cross border health data exchange within the European Union (EU). Their methodology consisted of four key components: a gap analysis study, the creation of user scenarios with a primary emphasis on cross-border health data exchanges in the project's three pilot countries, a user requirement gathering phase that included a threat analysis of the business processes associated with the user scenarios, and discussions and surveys with key stakeholders. The results indicate a current absence of a comprehensive security strategy, the lack of established sustainability plans to adapt to evolving frameworks aligned with contemporary standards, and that full compliance with information security standards is not consistently achieved.

Unexpected relationships between components of the system-to-be are frequently the cause of missing requirements, which are some of the main reasons software fails [32], [33]. Yousef and Almarabeh [33] developed a requirements elicitation framework that constructs the system’s CRUD (Create, Read, Update, Delete) matrix from an organization’s business process models. This matrix captures all potential system needs by offering every relationship between the system’s elements and operations. In order to guarantee that all comprehensive questions are asked during interviews, analysts can use the produced relationships between entities and functions as a guide. Additionally, the proposed approach enhances the completeness of requirements to a larger degree. Lu and Liang [29] employed four classification methods and three machine learning algorithms (namely, Naive Bayes, J48, and Bagging) to automatically categorize user reviews collected from two popular applications (iBooks and WhatsApp) into four types of NFRs (reliability, usability, portability, and performance), FRs, and other categories. The findings indicate that better classification of NFRs from user evaluations can be achieved through improved user reviews. To address socio-political challenges in RE, [34] presented a value-based requirements engineering (VBRE) paradigm that supplements the current examination of NFRs. In addition to considering stakeholders’ possible emotional responses to system change, the framework directs the elicitation of stakeholders’ values and reasons behind sociopolitical issues in systems development. By outlining many ways that knowledge can be applied directly or as a memory aid, it also supports both beginner and expert practice. A framework for identifying and analyzing 38 peer-reviewed publications that present methodologies (such as methods, processes, and modeling styles) that support any phase of the requirements engineering (RE) process for the robotic systems domain was proposed by [35] using a systematic mapping study (SMS). All RE phases were addressed by one or more approaches, modeling styles, and processes, according to the outcomes of the chosen studies.

Putera *et al.* [36] conducted a study to determine the requirements analysis for developing e-library apps utilizing the Technical Operational Economic (TOE) and Mandatory Desirable Inessential (MDI)

approaches. FRs and NFRs analysis served as the foundation for this study's examination of the requirements for the online library information system. Interviews with knowledgeable stakeholders and observation were used to gather data. Twenty-three (23) FRs and four (4) NFRs for the e-library applications were found by the study. Calvo *et al.* [37] combined Software Engineering (SE) and Human Computer Interaction (HCI) methodologies to define criteria aimed at enhancing interactions for individuals who face challenges with the flow and rhythm of discourse in chat environments. The study applied nine SE techniques alongside eighteen HCI techniques. The results indicated that integrating SE and HCI approaches plays a vital role in the requirements engineering process, as it enables the accurate identification of both user and system requirements for creating accessible chat applications on mobile devices within learning environments. Similarly, Kaur and Verma [38] investigated the importance of non-functional requirements (NFRs) in the context of online banking systems. Questionnaires were distributed to online banking users, and responses were collected from active customers. The Total Weightage Score System (TWS) statistical method was used to analyze the data. The analysis revealed that security emerged as the most critical concern for customers, followed by usability, performance, and availability. Insights from the reviewed literature highlight that accurately determining both functional requirements (FRs) and NFRs is essential for developing software products that meet the needs and expectations of stakeholders and end users.

3.0 Sensitive Data Protection Model for Online Social Networks

A decentralized application (DApp) called the SDPM will be used to guarantee the confidentiality, availability, and trust of sensitive data in OSNs. Along with cryptographic approaches, hashing algorithms, and consensus mechanisms, SDPM leverages blockchain technology to ensure data availability, privacy, and trust among untrusted peers on OSNs. It consists of an encryption module, a client application (user interface), a Hyperledger Indy Blockchain that houses the chaincode, hashes of the encrypted user data, and the Plenum Byzantine Fault Tolerant (PBFT) consensus algorithm, with a MongoDB local database. Mathematically, the SDPM (*sdpm*) will be represented as a 6-tuples which is defined as shown in Equation (3.1) as follows;

$$sdpm = \{ur, d, ea, ld, fra, bt\} \quad (3.1)$$

Where

ur = User

d = Social media sensitive data

ea = Encryption module algorithms

ld = Local MongoDB database

fra = Algorithm for Friend recommendation

bt = Hyperledger Indy framework Blockchain

The Blockchain (*bt*) is a 3-tuple as shown in Equation 3.4 as follows;

$$bt = \{ccode, cm, hed^1\} \quad (3.2)$$

where

ccode = Chaincode

cm = consensus mechanism

hed = Hashed of the encrypted data

4.0 Data Collections

Data (Requirements) collection was carried out by reviewing and analyzing over seventy (70) published journals that cover recent research works on Online Social media platforms that were built on Centralized architectures, Federated servers, peer-to-peer architectures, software requirements and Blockchain technologies. Current published papers on encryption algorithms, hashing algorithms, privacy, availability, trust, and Blockchain technologies applications were also reviewed and analysed.

Twenty-five Software developers who are experienced in social media platform development were interviewed to gather their views on their expectations of requirements that a Sensitive Data Protection Model (SDPM) should possess. Based on an understanding of the research problem, the interview questions were created. The MoSCoW prioritizing approaches [39] were used to prioritize the requirement lists, and the Mandatory Desirable Inessential (MDI) method was used to rank them [36], [40]. Analysts and stakeholders use the MoSCoW technique, which is an acronym made up of the initial letter of each of the four priority categories (Must have, Should have, Could have, and Will not have), to collaboratively prioritize requirements. Twenty-two functional and twenty-six (26) NFRs were selected for the development of the model. The selected FRs and NFRs are shown in the next sections.

5.0 Functional Requirements for the Sensitive Data Protection Models

FRs are declarations of services that outline the functions of the system, including their inputs and outputs, how the system is expected to respond to specific inputs, and the anticipated behavior of the system in certain scenarios [16]. The functional requirements for the SPDM are illustrated in Table 1.1.

6.0 Non-functional Requirements for the Sensitive Data Protection Models

NFRs refer to requirements that do not directly relate to the specific services provided by the system to its users. These types of requirements typically define or limit attributes of the overall system. The non-functional requirements for the SPDM are presented in Table 1.2.

7.0 Expected Contribution to Knowledge

The research work contributed to knowledge by determining the FRs and NFRs of a sensitive data protection model (SPDM) for OSN that ensures privacy, trust, and availability. The purpose of careful determination of these requirements is to ensure users' privacy is not breached, unlike what is obtainable in centralized and decentralized OSNs. The study also provides suitable metrics and requirements that will help researchers resolve the many bottlenecks in the formulation of models to protect data.

8.0 Conclusion

Identifying FRs and Non-Functional Requirements NFRs of the model is the initial phase in developing an SDPM. RE involves capturing the software needs. The SDPM aims to offer secure solutions by embedding trust within the network itself, enabling identity owners to maintain control over their personal information and manage access to their data while ensuring its availability and integrity. This goal can be accomplished through early identification of requirements. It will contribute to the establishment of a fully decentralized and secure online social network that provides high-quality services at little operational cost, even when functioning on unreliable, insecure, and sometimes malicious user devices. The research also seeks to address existing gaps by leveraging Blockchain technology alongside cryptographic techniques, hashing functions, and consensus protocols to guarantee data availability, privacy, and trust among untrusted participants in OSNs. Looking ahead, a prototype model could be developed for implementation in a real social network setting as part of this ongoing research initiative.

Table 1.1: List of functional requirements of the SPDM

FRID	Requirement Descriptions
FR1	The system must allow users to register and create a profile
FR2	The system must allow only registered users to log in and log out
FR3	The system should allow registered user to edit their profile
FR4	The system should allow registered users to set profile picture
FR5	The system shall allow registered user to reset their password using the “Forget Password”
FR6	The system should allow registered users to search for new friends
FR7	The system should allow registered users to create/reply/edit/delete/share post(s)
FR8	The system should allow registered users to follow/unfollow a friend
FR9	The system could allow registered users to view other users' profile
FR10	The system could allow a registered user to assign visibility permission to a friend ✓ Private: Only the user and friends can see their posts. ✓ Public: Everyone can see the posts theoretically if they visit the user's profile, but only followers (both approved and unapproved) see the user's posts in their feed. ✓ Approved-Followers: Only those followers, whose follow requests have been approved by the followee can see the followee's posts.
FR11	The system should allow a registered user to like/dislike a post The system must allow registered users to choose a node role (i.e., validator or observer node)
FR12	The system should allow the registered user to accept a friend
FR13	The system should allow a registered user to send a friend request
FR14	The system should allow a registered user to search for a friend
FR15	The system should allow registered users to generate and send a private key for other friends to view their posts
FR16	The system could allow registered users to upload files (video, audio, pictures, pdf)
FR17	The system could allow a guest user to search and view basic information about a registered user and send a message to them
FR18	
FR19	The system should allow the DApps to display a friend list using the friend recommendation algorithm
FR20	The system could the DApps to send a confirmation to the user's email to validate a newly registered user.
FR21	The system must allow validator nodes to vote to elect a leader using a consensus protocol in the Hyperledger Indy Blockchain

* FRID – Functional requirement ID

Table 1.2: List of Non-functional requirements

Types of Non-functional Requirements	NFRID	Description of non-functional requirements
Usability Requirements	NFR1	The system must be available online at all times to users
	NFR2	The system should be presentable on low-resolution devices (mobiles and tablets).
	NFR3	The system must be easy to learn and usable by both sophisticated and novice users
	NFR4	The system must respond to user's activities quickly
	NFR5	The system must provide the conveniences of usage
Performance Requirements	NFR6	The should shall be accessible to all registered participants
	NFR7	The system shall be fast in operation
	NFR8	The system must have a standard and friendly Graphical User Interface (GUI) that allows data entry, editing, and deleting of data during processing with ease
	NFR9	The system shall allow reliable storage of information
	NFR10	Posts shall be placed in the right category for quick response by users
	NFR11	The system shall be easy to maintain and upgradable
	NFR12	The system shall be able to work with relevant hardware devices
Operational Requirements	NFR13	The system shall not be prone to crashing and errors.
	NFR14	The system shall be able to handle and cater for multiple users
	NFR15	The system shall allow registered users to has access to posts according to their visibility permission
Security Requirements	NFR16	The system shall ensure sensitive information is hidden from non-users and other unauthorized users
	NFR17	The system shall allow Password to be case sensitive.

Portability Requirements	NFR18	The system shall allow only registered users to use the system
	NFR19	If anyone sends a message, the user should be able to know if it is a guest user or a friend in the network.
	NFR20	The system should ensure the protection of the personal information of users.
	NFR21	The system should be compatible with all operating systems and hardware
	NFR22	The system should operate on demand
Space Requirements	NFR23	The system should be compatible across browsers
	NFR24	The system should not consume much space
Ethical requirements	NFR25	The system should comply with quality assurance and other regulatory standards
	NFR26	The system should allow the removal of malicious nodes

* NFRID

References

- [1] J. Ahmed, "A Privacy Protection Model for Online Social Networks," in *SW4LAW+ DC@ JURIX*, 2014.
- [2] K. Graffi and N. Masinde, "LibreSocial: A Peer-to-Peer Framework for Online Social Networks," *arXiv preprint arXiv:2001.02962*, 2020.
- [3] B. Guidi, "When Blockchain meets Online Social Networks," *Pervasive and Mobile Computing*, vol. 62, p. 101131, 2020.
- [4] L. Wu, N. C. Pa, R. Abdullah, W. N. Ab Rahman, and M. Tee, "Exploring functional and non-functional requirements of social media on knowledge sharing," *Journal of Theoretical and Applied Information Technology*, vol. 93, no. 2, p. 595, 2016.
- [5] J. M. Such and M. Rovatsos, "Privacy policy negotiation in social media," *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, vol. 11, no. 1, p. 4, 2016.
- [6] N. C. Rathore and S. Tripathy, "InfoRest: Restricting Privacy Leakage to Online Social Network App," *arXiv preprint arXiv:1905.06403*, 2019.
- [7] L. Bahri, B. Carminati, and E. Ferrari, "Decentralized privacy preserving services for online social networks," *Online Social Networks and Media*, vol. 6, pp. 18–25, 2018.

- [8] C. Cadwalladr and E. Graham-Harrison, "How Cambridge Analytica turned Facebook 'likes' into a lucrative political tool," *The Guardian*, Mar. 17, 2018. [Online]. Available: <https://www.theguardian.com/>. [Accessed: Apr. 28, 2025].
- [9] A. De Salve, P. Mori, and L. Ricci, "A survey on privacy in decentralised online social networks," *Computer Science Review*, vol. 27, pp. 154-176, 2018.
- [10] N. Elisa, L. Yang, F. Chao, and Y. Cao, "A framework of a blockchain-based secure and privacy-preserving E-government system," *Wireless Networks*, pp. 1–11, 2018.
- [11] F. O. Idepefo, B. Akhigbe, I. Aderibigbe, S. Ojo, and S. B. Afolabi, "Towards an architecture-based ensemble methods for online social network sensitive data privacy protection," *International Journal of Recent Contributions from Engineering, Science & IT (IJES)*, vol. 9, p. 33, 2021, doi: 10.3991/ijes.v9i1.20819.
- [12] K. Curcio, T. Navarro, A. Malucelli, and S. Reinehr, "Requirements engineering: A systematic mapping study in agile software development," *Journal of Systems and Software*, vol. 139, pp. 32–50, 2018.
- [13] M. H. Ferreira, A. Carvalho de Oliveira Junior, E. Dias Canedo, R. A. Dias Kosloski, R. Á. Paldês, and E. C. Oliveira, "Design thinking: Challenges for software requirements elicitation," *Information*, vol. 10, no. 12, p. 371, 2019.
- [14] Z. Kurtanović and W. Maalej, "Automatically classifying functional and non-functional requirements using supervised machine learning," in *2017 IEEE 25th International Requirements Engineering Conference (RE)*, pp. 490–495, 2017.
- [15] S. Mughal, A. Abbas, N. Ahmad, and S. U. Khan, "A Social Network Based Process to Minimize In-Group Biasedness During Requirement Engineering," *IEEE Access*, vol. 6, pp. 66870–66885, 2018.
- [16] I. Sommerville, *Software Engineering GE*, Pearson Australia Pty Limited, 2016.
- [17] X. Franch, C. Palomares, C. Quer, P. Chatzipetrou, and T. Gorschek, "The state-of-practice in requirements specification: an extended interview study at 12 companies," *Requirements Engineering*, vol. 28, no. 3, pp. 377-409, 2023.
- [18] J. White, S. Hays, Q. Fu, J. Spencer-Smith, and D. C. Schmidt, "ChatGPT prompt patterns for improving code quality, refactoring, requirements elicitation, and software design," in *Generative AI for Effective Software Development*, pp. 71-108, Cham: Springer Nature Switzerland, 2024.
- [19] P. Malcher, E. Silva, D. Viana, and R. Santos, "What do we know about requirements management in software ecosystems?," *Requirements Engineering*, vol. 28, no. 4, pp. 567–593, 2023.
- [20] S. Pargaonkar, "A comprehensive research analysis of software development life cycle (SDLC) agile & waterfall model advantages, disadvantages, and application suitability in software quality engineering," *Int. J. Sci. Res. Publ. (IJSRP)*, vol. 13, no. 08, pp. 345-358, 2023.
- [21] S. Alsaleh and H. Haron, "The Most Important Functional and Non-Functional Requirements of Knowledge Sharing System at Public Academic Institutions: A Case Study," *Lecture Notes on Software Engineering*, vol. 4, no. 2, p. 157, 2016.

- [22] Y. Sedelmaier and D. Landes, "How can we find out what makes a good requirements engineer in the age of digitalization?," in *2017 IEEE Global Engineering Education Conference (EDUCON)*, pp. 230–238, 2017.
- [23] M. Shafiq *et al.*, "Effect of project management in requirements engineering and requirements change management processes for global software development," *IEEE Access*, vol. 6, pp. 25747–25763, 2018.
- [24] D. Lindoerfer and U. Mansmann, "Data for the elaboration of the CIPROS checklist with items for a patient registry software system: Examples and explanations," *Data in Brief*, vol. 14, pp. 494–497, 2017.
- [25] J. Iqbal, R. B. Ahmad, M. Khan, S. Alyahya, M. H. Nizam Nasir, A. Akhunzada, and M. Shoaib, "Requirements engineering issues causing software development outsourcing failure," *PloS one*, vol. 15, no. 4, p. e0229785, 2020.
- [26] D. Hidellaarachchi, J. Grundy, R. Hoda, and K. Madampe, "The effects of human aspects on the requirements engineering process: A systematic literature review," *IEEE Trans. Softw. Eng.*, vol. 48, no. 6, pp. 2105–2127, 2021.
- [27] D. A. Menéndez and P. C. da Silva, "A requirement elicitation process for BI projects," *Lecture Notes on Software Engineering*, vol. 4, no. 1, p. 20, 2016.
- [28] D. Lindoerfer and U. Mansmann, "Enhancing requirements engineering for Patient Registry Software Systems with evidence-based components," *Journal of Biomedical Informatics*, vol. 71, pp. 147–153, 2017.
- [29] M. Lu and P. Liang, "Automatic classification of non-functional requirements from augmented app user reviews," in *Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering*, pp. 344–353, 2017.
- [30] A. M. Nascimento, D. S. da Silveira, J. S. Dornelas, and J. Araújo, "Exploring contextual factors in citizen-initiated platforms to non-functional requirements elicitation," *Transforming Government: People, Process and Policy*, 2020.
- [31] P. Natsiavas, J. Rasmussen, M. Voss-Knude, K. Votis, L. Coppolino, P. Campegiani, ... and M. Nalin, "Comprehensive user requirements engineering methodology for secure and interoperable health data exchange," *BMC Medical Informatics and Decision Making*, vol. 18, no. 1, p. 85, 2018.
- [32] B. Aysolmaz, H. Leopold, H. A. Reijers, and O. Demirörs, "A semi-automated approach for generating natural language requirements documents based on business process models," *Information and Software Technology*, vol. 93, pp. 14–29, 2018.
- [33] R. Yousef and T. Almarabeh, "An enhanced requirements elicitation framework based on business process models," *Scientific Research and Essays*, vol. 10, no. 7, pp. 279–286, 2015.
- [34] S. Thew and A. Sutcliffe, "Value-based requirements engineering: method and experience," *Requirements Engineering*, vol. 23, no. 4, pp. 443–464, 2018.
- [35] D. Albuquerque, J. Castro, S. Ribeiro, and T. Heineck, "Requirements Engineering for Robotic System: A Systematic Mapping Study," in *WER*, 2017.

- [36] A. R. Putera, S. Riyanto, and M. Arianto, "Requirement analysis of e-library application using Mandatory Desirable Inessential (MDI) and Technical Operational Economic (TOE) method," in *Journal of Physics: Conference Series*, vol. 1381, no. 1, p. 012068, 2019.
- [37] R. Calvo, A. Iglesias, and L. Moreno, "User-Centered Requirement Engineering for Accessible Chats in m-Learning," *Journal of Universal Computer Science*, vol. 20, no. 7, pp. 964–985, 2014.
- [38] H. Kaur and G. N. Verma, "A Case Study upon Non-functional Requirements of Online Banking System," *International Journal of Computer Applications Technology and Research*, vol. 4, pp. 220–225, 2015.
- [39] A. Hudaib, R. Masadeh, M. H. Qasem, and A. Alzaqebah, "Requirement'prioritization techniques comparison," *Modern Applied Science*, vol. 12, no. 2, p. 62, 2018.
- [40] M. Dabbagh and S. P. Lee, "An approach for integrating the prioritization of functional and nonfunctional requirements," *The Scientific World Journal*, vol. 2014, 2014.