المنهجية البرمجية لتحسين عمليات تطوير أنظمة الرعاية الصحية

A Software Methodology to Improve Development Processes of Healthcare Systems

د.عبدالعزيز احمد ثوابة[1]
Dr.Abdulaziz Ahmed Thawaba [2]
أ.م.د.عزيز الأزهر الرملي[3]
Dr.Aziz ul Azhar Ramli[4]
أ.م.د محمد فرحان محمد فودزي[5]
Mohd. Farhan Md. Fudzee[4]

https://doi.org/10.54582/TSJ.2.2.22

(1) استاذ مساعد – هندسة برمجيات – كلية تكنولوجيا المعلومات وعلوم الحاسوب – جامعة إقليم سبأ – اليمن

(2) Dept. Computer Science, University of Saba Region, Marib, Yemen

(3) استاذ مشارك – عميد كلية علوم الحاسوب وتكنولوجيا المعلومات – جامعة تون حسين أون – ماليزيا

(4) Faculty of Computer Science and Information Technology,
Universiti Tun Hussein Onn Malaysia, Parit Raja, 86400 Batu Pahat,
Johor Darul Takzim, Malaysia

(5) استاذ مشارك – رئيس مركز الحاسوب – جامعة تون حسين أون – ماليزيا

عنوان المراسلة :

:azizth@usr.ac*, azizulr@uthm.edu.my, farhan@uthm.edu.my

الملخص

قد تكون منهجية تطوير البرمجيات لمشاريع الرعاية الصحية سببًا رئيسيًا لتأخر الجدول الزمني ، والإفراط في الميزانية ، وعدم الرضا عن المتطلبات ، وضعف الجودة. لذلك ، يجب أن تكون هناك منهجية لإدارة وتحسين عمليات تطوير البرمجيات كحل رئيسي لهذه المشاكل. تتطلب إدارة عملية تطوير البرامج دليلًا إرشاديًا لتطوير منتجات البرامج في الموعد المحدد ضمن الميزانية ، وتجنب المخاطر ، وتلبية المتطلبات. في السنوات الأخيرة ، تمت دراسة تحسين عملية البرامج على نطاق واسع في تطوير البرامج التقليدية أو السريعة ، وتم التعرف على نقاط القوة والضعف فيها. لقد تحدت المنهجيات السريعة الطرق التقليدية لتطوير نظام حاسم مثل الرعاية الصحية. الهدف الرئيسي من هذا البحث هو تحسين إنتاجية أنظمة الرعاية الصحية وتجنب المخاطر من خلال تحسين عمليات التنمية. اقترح البحث منهجية تطوير تسمى «التطوير السريع لنظام الرعاية الصحية (RHSD)». تعمل المنهجية المقترحة بسرعة وتعتني بالمخاطر أثناء عمليات التطوير من خلال تكثيف آلية الاختبار والتحقق. كما يسهل RHSD تطوير أنظمة الرعاية الصحية ضمن المتطلبات وتحقيق أعلى مستويات الجودة.

الكلمات المفتاحية: الصحة و نظام المعلومات (HIS) ؛ تحسين عملية البرمجيات (SPI) ؛ منهجيات تطوير البرمجيات؛ المجال: تطوير البرمجيات.

https://doi.org/10.54582/TSJ.2.2.22

ABSTRACT

Software development methodology for healthcare projects may be a major cause of late schedule, over-budgeting, dissatisfaction with requirements, and poor quality. Therefore, there should be a methodology for managing and improving software development processes as a major solution to these problems. Managing software development process requires a guideline for developing software products on schedule within budget, avoiding risks, and meeting requirements. In recent years, software process improvement has been extensively studied in traditional or rapid software development, and its strengths and weaknesses have been recognized. Rapidly methodologies have challenged the traditional ways of developing a critical system such as healthcare. The main objective of this research is to improve the productivity of healthcare systems and avoid risks by improving development processes. The research proposed a development methodology called "Rapid Healthcare System Development (RHSD)". The proposed methodology acts fast and takes care of risks during development processes by intensifying the testing and validation mechanism. RHSD also facilitates the development of health care systems within the required requirements and achieves the highest levels of quality.

Keywords: Health & Information System (HIS); Software Process Improvement (SPI); Software Development Methodologies; Area: Software development.

https://doi.org/10.54582/TSJ.2.2.22

INTRODUCTION

The software industry is constantly working to reduce the complexity of software products and focus more on the best, fastest, and cheapest development processes [1]. The software business has received assistance from new technologies, development tools, programming languages, and development methods [2]. Currently, software developers have modern development methods that keep pace with technological development and the rapid needs of the market [3]. Healthcare needs modern software and IT infrastructure more than any other industry around the world. The Healthcare system is a critical system in which simple mistakes may affect the patient's life [4]. Healthcare systems are transitioning from architectural practice centered around hospitals to programs based on patient-centered electronic medical records [5]. Nowadays, there is a strong demand for a modern quality healthcare system. A high-quality healthcare system relies heavily on the integration of data from administrative, clinical, and management sources [6]. Therefore, the software developer focuses on the quality of the processes used in the development of health systems to keep pace with the quality required for those types of critical systems7]]. However, there is a lack of clarity in software development methodologies that take into account the development characteristics of healthcare systems [8]. This paper is concerned with the analysis and evaluation of productivity improvements in various aspects of the health care system development and management processes. This research also presents a proposal for a development methodology for health care systems that take into account the characteristics of these systems and the speed of development processes to keep pace with the needs of the modern software industry. Throughout this paper, the theoretical background of Health and Information System (HIS), Software Process Improvement (SPI), and software development methodologies will be discussed in Section Two. The third section is discussing the contribution of research and highlights the proposed methodology for developing the healthcare system and finally the conclusion.

THEORETICAL BACKGROUND

1.1 Health & Information System (HIS)

A Healthcare Information System (HIS) is a critical system that processes, analyzes, and uses patient information to deliver health

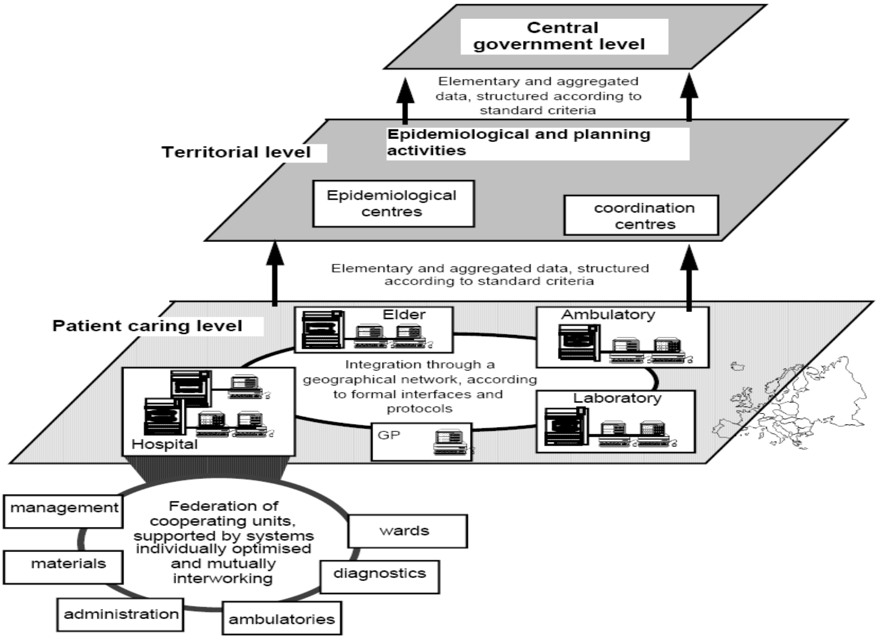
Central government level
Elementary and aggregated data, structured according to standard criteria
Territorial level
Epidemiological and planning activities
Epidemiological centres
coordination centres
Elementary and aggregated data, structured according to standard criteria
Patient caring level
Elder
Ambulatory
Integration through a geographical network, according to formal interfaces and protocols
Laboratory
Hospital
GP
management
Federation of cooperating units, supported by systems individually optimised and mutually interworking
wards
materials
diagnostics
administration
ambulatories

Figure 1: Structure of the Hospital Information System

1.2 Software Process Improvement (SPI)

Software Process Improvement (SPI) is a specific mechanism consisting of a set of interrelated resources and activities to improve development processes and improve product quality [12]. There are different groups of practitioners across the world who use different

https://doi.org/10.54582/TSJ.2.2.22

approaches to implement SPI. North American companies feel comfortable using CMM; Japanese companies prefer TQM and European companies use the ISO 9000 family to improve their regulatory capabilities. CMM is entirely designed for software development processes, while TQM and ISO 9000 standards are not, especially for software manufacturing [13]. The development process improvement needs to depend on the business needs and different approaches can be adopted depending on the organization's current practice and maturity. Thus, appropriate methods will be recommended and implemented in proportion to the organization's current practices and business objectives. It is not easy to improve the development process to determine the impact and predict the success of any improvement methods because this depends on the organization environment variables such as employee skills, the effectiveness of training, the efficiency of process implementation, and acceptance. The selection and successful implementation of SPI methods depend on many variables such as the maturity of the current process, skill base, organization, and business issues such as cost, risk, speed of implementation, etc. [14].

The Capability Maturity Model (CMM) is specifically designed to optimize development processes and support the maturity levels of different organization processes [15]. CMM defines and supports five levels of maturity. Where the initial level indicates that the organization does not have a stable software development environment. The second level demonstrates repeatability since the organization has established policies and standards for project management and defines the procedures for their implementation, but the processes may differ between different projects. At the third level (Defined Level), processes that are emphatically documented across the organization are used for software development and maintenance. The penultimate level (Managed Level) ensures that software products are of high quality and that management sets quantitative and quality goals for the product and process. The last level (Optimizing Level) the objectives of the organization are to focus primarily on continuous process improvement [16].

1.3 Software Development Methodologies

System development methodology in software engineering is a framework that is used to structure, plan, and control the information system development process [17]. Some software methodologies can be used to build healthcare systems and improve the quality of the end product [18]. Although some of these methodologies focus on the perspectives of users or developers, end systems sometimes fail to achieve satisfactory usability. Among the most important known methodologies in building systems are the following:

- Waterfall methodology is the first method for developing an organized system. It has come under attack in recent years for being too strict and unrealistic when it comes to meeting customer needs quickly, and the waterfall model is still widely used. The waterfall is credited with providing the theoretical basis for other process models because it is very similar to the "general" model of software development [19].
- Rational Unified Process (RUP) is a methodology for software development. Projects of all types and sizes have successfully used RUP as it varies from lightweight to meet the needs of small projects with short product cycles to more comprehensive operations that cater to the broader needs of large, possibly distributed project teams [20].
- Rapid Application Development (RAD) is a software framework that aims to enable developers to quickly create high-quality applications to meet new business requirements. RAD is designed to make the most of the powerful development programs that have

https://doi.org/10.54582/TSJ.2.2.22

been developed recently [21].

- Agile introduced several software development methodologies such as Scrum and Extreme Programming to meet critical system development requirements [22]. Extreme Programming (XP) is software development methodologies proposed to improve software quality and change the response to user requirements. XP is an area of software development based on the values of simplicity, communication, feedback, and courage. Extreme Programmers work together in pairs and as a group, with a simple design and painstakingly tested code, the design continues to be improved to keep it always well suited to current needs [23].

RESEARCH DISCUSSION

Through previous studies in the field of development methodologies for health care systems, we found that these systems need two levels; Regional level and level of patient care. We also found that the development of a reliable healthcare system should follow a structured and rapid development method, and the RAD methodology was somewhat appropriate for the development of these systems. RAD enhances the development process for critical systems and according to rapid market needs more than waterfall and/or agile methodologies according to previous research. However, RAD methodology still needs to improve product quality by increasing testing during all phases of the healthcare systems development. Therefore, the RAD phases must be redesigned with a mechanism that improves the testing and verification processes to reach a high-quality product.

3.1 Evaluation of Software Process Improvement

This section discusses software process implementation improvement measurement frameworks, as shown in Table 1, and there are three methods: CMM, TOM, and ISO 9000. The appropriate framework for measuring the quality of healthcare systems development

https://doi.org/10.54582/TSJ.2.2.22

processes is the CMM because it uses several levels of quality to measure the entire development processes as shows in Table 1.

Table 1 Measurement frameworks for improving implementation of SPI

Approaches	Description	Specification
CMM	CMM using comfortable in North American companies	designed entirely for software processes
TQM	Japanese companies	not especially for software manufacture
ISO 9000	European firms	

3.2 Evaluation of Software Development Methodologies

In this section, we studied the most development methodologies in software engineering that are suitable for developing healthcare systems. This section discusses development methodologies in software engineering appropriate for health care systems development as shown in Table 2. The waterfall approach can fit into the development of healthcare systems but it focuses on development processes and poorly manages time and changes, also do not involve the users in most of the development life cycle. The RAD methodology can guide the development of the health care system, but it needs to improve the mechanisms of testing and verification, especially when developing high-risk systems such as health care systems. This paper proposes to combine Waterfall and RAD into a new methodology that has the strengths of both and use it to develop a healthcare system.

https://doi.org/10.54582/TSJ.2.2.22

Table 2: Software development methodology

Method	Description	Phases	Methodology & Develop Healthcare System	
			Advantages	Disadvantages
Waterfall Model	It is a sequential design process, often used in development processes.	● Planning ● Analysis ● Design, ● Implementation ● Maintenance	1) Simple to implement 2) provides quality 3) provides a template into which methods of all phases can be placed	1) Doesn't focus on schedule and cost 2) Changing in the previous stage can cause big problems
Rational Unified Process (RUP)	RUP is an iterative development process	● Inception ● Elaboration ● Construction ● Transition	1) Training readily available 2) Reduced effort and time during the integration phase	1) Process too complex 2) Doesn't capture the sociological aspects of development processes.
Rapid Application Development (RAD)	RAD used minimal planning in favor of rapid prototyping.	● Requirements Planning ● User Design ● Construction ● Implementing	1) Adaptable to changes 2) Handle large projects 3) Realizes an overall reduction risk 4) Incorporates short development cycles	1) RAD is not compatible with all applications 2) RAD is not appropriate hen technical risks are high

https://doi.org/10.54582/TSJ.2.2.22

Extreme Programming (XP)	XP is intended to improve quality and responsiveness to changing customer requirements	• Fine-scale feedback • Continuous process • Shared understanding • Programmer welfare • Coding and Testing	1) Continuous testing and integration help to increase the quality 2) Enabling organizations to manage their software efforts in a better way	Cannot work with systems that have scalability issues

1.3RAPID HEALTHCARE SYSTEM DEVELOPMENT (RHSD)

The waterfall methodology was favored in the development of healthcare systems in earlier periods. Nowadays, rapid development methodology (RAD) methodology has become the most popular platform in healthcare system development for the fast delivery of the final product. The health care system is one of the critical systems that require deep analysis, many tests, and measurement of all parts of the development stages [24]. RAD is a rapid development method that may not be suitable for critical systems projects such as the healthcare system. The waterfall methodology or any of the traditional approaches may be suitable for healthcare systems development, but it takes a long time during the development process, which is not in line with the speed of the software industry and the needs of the market. Therefore, this research proposes a methodology called Rapid Healthcare System Development (RHSD), which is a combination of RAD and Waterfall methodologies. RHSD is a platform that can act quickly and pay attention to the details of the healthcare system. RHSD was created to fit the development of healthcare projects.

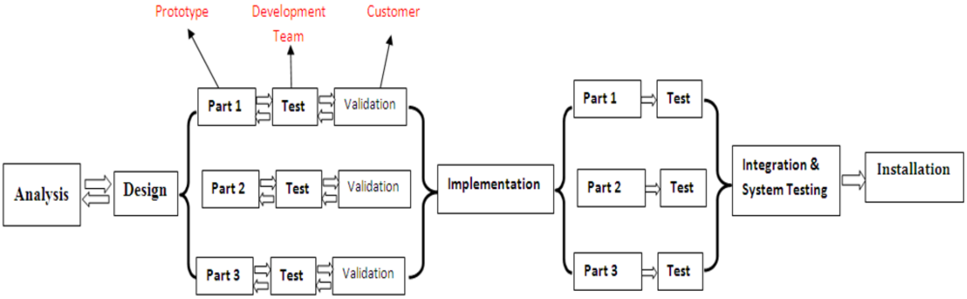

Figure 2 RHSD Methodology

Figure 2 shows the five development stages of RHSD. The task of developing an RHSD architecture is divided into several parts that are constantly tested and rapidly developed. The design, implementation, and integration phase involve many user and developer tests. RHSD uses the same techniques and computer tools that RAD uses, such as the CASE and JAD tools. The purpose of adapting JAD in RHSD development processes is to bring together the technical/creative team and the business community to extract consensus-based software requirements. The phases of RHSD development are:

1) Analysis phase: The RHSD analysis phase uses both traditional and modern requirement-gathering techniques as shown in Table 3. The two methods were used due to the sensitivity of the healthcare system which requires in-depth analysis.

Table 3 RHSD analysis techniques

Techniques/ Model	Type of Model	Description
Background Reading	Traditional	To know the organization and its business .objectives
Interviewing		To obtain depth understanding of the roles and requirements of managers, staff, and .customers
Observation		To enable the analyst to see the reality of a .situation
Document Analysis		Review of existing business documents and give a historical and formal view of system .requirements
Questionnaires		To obtain the views of a huge amount of people in a way that can be analyzed statistically.
Joint Application (Design (JAD	Modern	the purpose is to carry the business community, customers, and technical/creative team in a structured workshop to extract consensus of product requirements

1) Design Phase: The design phase begins with the required document that is delivered in the analysis phase and defines the requirements in the design. The RHSD design phase divides the system into more than two parts according to the system's sensitivity and size. Each part has an individual design in the form of a design prototype and then tested by the development team finally, the design prototypes must go to the customer for validation. However, if the testing team finds a lack of requirements or customer dissatisfaction during verification, the task will return frequently to the analysis phase to further investigate the requirements. The three stages that each part must go through; Prototyping, testing, and verification can work iteratively until the system part is completely ready for

implementation.

■ Prototypes: Develop a prototype according to the analysis and prototypes can contain more than one version. Upon completion of the development of the prototype should go for testing.

■ Testing: Each prototype must be tested by the development team several times.

■ Validation: When testing is completed by the developers, the customer must verify that the implementation matches the requirements.

3)Implementation phase: The development team builds the components of the system. The team should start the implementation process exactly as defined in the design phase and requirements gathered in the analysis phase with room for flexibility and innovation if needed. In the implementation, the development team deals with issues of coding, debugging, quality, and product performance.

4)Integration and Test Phase: The test data is migrated from the development environment to the package system environment. Inspections are carried out to ensure the correctness and completeness of the healthcare system product. The test kit verifies the product's compliance with quality standards and requirements. This phase finalizes with production reference data for users and associates them with their appropriate roles. The comprehensive integration testing plan includes all of the following:

- Regression Test: test one or more features of the product
- Internal Testing: check all internal components work well
- System Testing: Make sure the system can complete all scenarios
- Stress Testing: Run the system in an environment that is more stressful than the target environment

5)Installation Phase: installation is the organizational process of switching from the old information system to the new one. The greatest installation technology for the healthcare system is parallel installation. Old and new symbiosis are used in parallel to reduce the impact of error and high cost on system resources. The old system has been running parallel to the new system for some time; It provides a fallback if there are problems and the outputs of the two systems can be compared, so testing continues in the live environment.

The RHSD methodology provides documentation that focuses on the quality of the end product during the development phases of healthcare systems. RHSD will not move into the implementation phase unless the analysis and design phases are completed, which will reduce risks in the upcoming phases. Having the user in the analysis and design phase helps clarify and validate system requirements. RHSD uses a prototype at the design stage to provide early process functionality for optimal requirement definition. Finally, the RHSD methodology provides a formal specification embodied in an operational replica.

CONCLUSION

The main objective of this study is how to improve the productivity of healthcare systems by creating appropriate development processes that avoid risks and achieve a high-quality product. The software development activities within healthcare in the past have been notable for their propensity to go over budget, getting far schedule, and over-running project costs. However, some projects ended in complete failure after a lot of time, and countless resources were expended for their development. In addition, healthcare systems cannot waste their resources. Therefore, methodologies, practices, and procedures must be well planned and demonstrated, and best practices must be recognized and dynamically applied. To find solutions to improve healthcare systems development, this paper reviews and discusses frameworks for measuring implementation processes and appropriate development methodologies. This research proposes a rapid health care system development (RHSD)

https://doi.org/10.54582/TSJ.2.2.22

methodology. RHSD is a combination of RAD and Waterfall methodologies. It is a platform that can act quickly and take care of risks during development processes utilizing testing and verification mechanism. Healthcare systems need to ensure that the development methodology is operating according to standards. Therefore, future studies can focus on how to implement standards at all phases of development, as well as predicting the percentage of implementation compliance with standards at advanced stages of system development.

Acknowledgment

This research was supported by Ministry of Higher Education (MOHE) trough Fundamental Research Grant Scheme (FRGS/1/2019/ICT01/UTHM/02/2). We also want to thank to the Government of Malaysia which provide MyBrain15 programme for sponsoring this work under the self-funded research grant and L00022 from Ministry of Science, Technology and Innovation (MOSTI).

References

[1] O. Cico, L. Jaccheri, A. Nguyen-Duc, and H. Zhang, "Exploring the intersection between software industry and Software Engineering education-A systematic mapping of Software Engineering Trends," J. Syst. Softw., vol. 172, p. 110736, 2021.

[2] R. R. Althar and D. Samanta, "The realist approach for evaluation of computational intelligence in software engineering," Innov. Syst. Softw. Eng., vol. 17, no. 1, pp. 17–27, 2021.

[3] I. O. Biletska, A. F. Paladieva, H. D. Avchinnikova, and Y. Y. Kazak, "The use of modern technologies by foreign language teachers: developing digital skills," Linguist. Cult. Rev., vol. 5, no. S2, pp. 16–27, 2021.

[4] A. A. Thawaba, A. A. Ramli, M. F. M. Fudzee, and J. Wadata, "Characteristics for Performance Optimization of Safety-Critical System Development (SCSD)," J. Adv. Comput. Intell. Intell. Inform., vol. 24, no. 2, Art. no. 2, Mar. 2020, doi: 10.20965/jaciii.2020.p0232.

[5] T. A. Nguyen, D. Min, E. Choi, and J.-W. Lee, "Dependability and security quantification of an internet of medical things infrastructure based on cloud-fog-edge continuum for healthcare monitoring using hierarchical models," IEEE Internet Things J., vol. 8, no. 21, pp. 15704–15748, 2021.

[6] A. I. Newaz, A. K. Sikder, M. A. Rahman, and A. S. Uluagac, "A survey on security and privacy issues in modern healthcare systems: Attacks and defenses," ACM Trans. Comput. Healthc., vol. 2, no. 3, pp. 1–44, 2021.

[7] A. A. Thawaba, A. A. Ramli, and M. F. M. Fudzee, "PM-ISD Traceability Metrics Enhance Reliability Assessment for Safety-Critical Systems Development Processes: A Case Study of Oil and Gas Well Drilling Project." Accessed: Feb. 19, 2022. [Online].

Available: https://ieeexplore.ieee.org/abstract/document/9590906/

[8] A. Darwish, A. E. Hassanien, M. Elhoseny, A. K. Sangaiah, and K. Muhammad, “The impact of the hybrid platform of internet of things and cloud computing on healthcare systems: opportunities, challenges, and open problems,” J. Ambient Intell. Humaniz. Comput., vol. 10, no. 10, pp. 4151–4166, 2019.

[9] F. Y. Zulkifli and I. W. Mustika, “Development of monitoring and health service information system to support smart health on android platform,” in 2018 4th International Conference on Nano Electronics Research and Education (ICNERE), 2018, pp. 1–6.

[10] L. Faridah, F. R. Rinawan, N. Fauziah, W. Mayasari, A. Dwiartama, and K. Watanabe, “Evaluation of health information system (HIS) in the surveillance of dengue in Indonesia: lessons from case in Bandung, West Java,” Int. J. Environ. Res. Public. Health, vol. 17, no. 5, p. 1795, 2020.

[11] T. Rokicki, A. Perkowska, and M. Ratajczak, “Differentiation in healthcare financing in eu countries,” Sustainability, vol. 13, no. 1, p. 251, 2020.

[12] M. M. Alhammad and A. M. Moreno, “Challenges of gamification in software process improvement,” J. Softw. Evol. Process, vol. 32, no. 6, p. e2231, 2020.

[13] B. Boukhari, C. Boueri, R. Islambouli, Z. Sweidan, and R. A. Haraty, “SQA Models Comparative Analysis,” in 2020 IEEE International Conference on Industrial Technology (ICIT), 2020, pp. 1001–1006.

[14] S. L. Ramírez-Mora, H. Oktaba, and J. Patlán Pérez, “Group maturity, team efficiency, and team effectiveness in software development: a case study in a CMMI-DEV Level 5 organization,” J. Softw. Evol. Process, vol. 32, no. 4, p. e2232, 2020.

[15] L. Shen, X. Du, G. Cheng, and X. Wei, “Capability

maturity model (cmm) method for assessing the performance of low-carbon city practice," Environ. Impact Assess. Rev., vol. 87, p. 106549, 2021.

[16] E. Gökalp and V. Martinez, "Digital transformation maturity assessment: development of the digital transformation capability maturity model," Int. J. Prod. Res., pp. 1–21, 2021.

[17] V. V. Yahupov, V. Y. Kyva, and V. I. Zaselskiy, "The methodology of development of information and communication competence in teachers of the military education system applying the distance form of learning," 2020.

[18] J. A. Magnuson and B. E. Dixon, Public health informatics and information systems. Springer, 2020.

[19] N. Salnikov, "How software development methodologies affect dynamic capabilities under extreme contexts: a COVID-19 study on agile and waterfall methodologies," 2021.

[20] S. Shafiee, Y. Wautelet, L. Hvam, E. Sandrin, and C. Forza, "Scrum versus Rational Unified Process in facing the main challenges of product configuration systems development," J. Syst. Softw., vol. 170, p. 110732, 2020.

[21] M. Tabrani and H. Priyandaru, "Application of the Rapid Application Development Method to the BAZNAS Zakat Receipt Information System in Karawang," J. Teknol. Dan Open Source, vol. 4, no. 1, pp. 78–84, 2021.

[22] A. A. Thawaba, A. A. Ramli, Mohd. F. Md. Fudzee, and J. Wadata, "A Mechanism to Support Agile Frameworks Enhancing Reliability Assessment for SCS Development: A Case Study of Medical Surgery Departments," in Recent Advances on Soft Computing and Data Mining, Cham, 2020, pp. 66–76.

[23] I. Ahmad, R. I. Borman, J. Fakhrurozi, and G. G. Caksana, "Software Development Dengan Extreme Programming (XP) Pada Aplikasi Deteksi Kemiripan Judul Skripsi Berbasis Android,"

INOVTEK Polbeng-Seri Inform., vol. 5, no. 2, pp. 297–307, 2020.

[24] A. A. Thawaba, A. A. Ramli, and Mohd. F. Md. Fudzee, "WTM to Enhances Predictive Assessment of Systems Development Practices: A Case Study of Petroleum Drilling Project," in Intelligent Human Computer Interaction, Cham, 2022, pp. 584–596.