A Proposed Hybrid Approach for Developing Healthcare Information Systems

Nesma El-Sokkary
Department of Radiation Engineering, National Center for Radiation Research and Technology (NCRRT)
Egyptian Atomic Energy Authority
Cairo, Egypt

Walaa Hussein El-Masry
Information Technology department, Faculty of Computers and Artificial Intelligence, Cairo University
Cairo - Egypt

Nagy Ramadan Darwish
Information System & Information Technology department, Faculty of Graduate Studies for Statistical Research, Cairo University, Cairo, Egypt

ABSTRACT

Healthcare software development has emerged as a promise to rework the paper-based system into a digitized structure to make sure effective care, patient safety and decreasing medical errors. Healthcare considered complex process and do not leave errors occur therefore need particular methods for medical software development. A healthcare solution needs for compliance with laws and regulations, be more safer, stable, high performance and high availability. This is often not a neighborhood where you'll release a poor-quality product and continuously improve it – you would like your product to be trusted and used from day one. Agile is an iterative process that creates the merchandise usable from the first stages, enabling rapid and high-quality development which will respond quickly to changing requirements. The Waterfall model may be a good model for managing an outsized scale problem also as it's safer. This paper tackled the common software development methodologies and debates the benefits and drawbacks of every then how we adjust them to healthcare requirements. During this paper was proposed a hybrid software development methodology (Hybrid Scrum-Fall Model) for the healthcare system development. The proposed model combines the agile development methodology and therefore the waterfall model to make solutions that are characterized by quality and deliver more value to the end user.

General Terms

Software Engineering, Agile, Waterfall, Healthcare

Keywords

Healthcare, Systems Development Lifecycle, SDLC, Agile, Scrum, Waterfall, Hybrid

1. INTRODUCTION

In recent years, new technologies have transformed into healthcare development to improve healthcare delivery and services, access stakeholder goals and lead to software solutions face the challenges in the healthcare domain.

This industry deals with the most serious matter there – human health, so in most cases, there's no space for failing and learning through errors. A healthcare solution needs to work the way it was expected and perform all the functions from the very beginning. It requires compliance with laws and regulations, be more secure, stable, high performance and high availability. This is not an area where you can release a poor-quality product and continuously improve it – you need your product to be trusted and used from day one.

The traditional software development methodologies are characterized by sequential series of steps, defining and documenting requirements. The traditional methods are also based on a predictive approach where all activities are strictly based on the initial requirement analysis and planning. Software change requirements in the traditional methods must pass through established change management process. One of the key features of agile approaches to software development is the adoption of adaptive methods. There is no detailed planning in the initial phase and the scope of tasks is primarily restricted to only the current functionalities of the specified software. Therefore, the agile methods rely on the early client involvement, incremental and iterative approach whereby the software development life cycle activities are iteratively visited based on customer feedback and desires [1]. The agile approach involves frequent testing and constant open communication with clients and closed team collaboration but involves minimal documentation and reduced risks of major faults in ongoing activities [2]. This paper introduces literature review about the healthcare system development in section 2, outlines some of related works in section 3 ,then tackles a brief review about software development methodologies in section 4, and then introduces the Hybrid Scrum- Fall model in section 5.

2. HEALTHCARE SYSTEM DEVELOPMENT

Healthcare systems are often characterized as particularly complex systems located during a dynamic environment, reacting with a huge diversity of heterogeneous actors, interests and factors, often with different and changed demands. These include local, national and regional political forces, also as academic, commercial, non-governmental and patient stakeholders. Social and demographic changes, fast medical technological developments, professional concerns, also as changes publicly public demand for care and new theories about health care management and organization all affect healthcare systems [3]. Healthcare software development has all of the challenges of conventional project software development, with the addition of multiple critical and non-negotiable factors. These factors raise the complexity of software development be used within the medical domain. Software package is taken into account a medical device; there'll be important regulatory requirements to be taken into account. The development team will need to define these requirements in advance and they'll be hugely based on two key questions: Is the software product considered to be a medical device OR is it a part of a medical device? Are you storing protected health information (PHI)?

The development team will need to determine all of the regulatory requirements that must abide to, and then develop an approach and a set of processes that will meet those requirements before any development occur. In any case of whether you use an Agile or Waterfall approach for software development, the regulatory requirements will not change. Once defined what the regulatory requirements are, they can be documented as non-functional user stories. Regularity requirements and patient privacy requirements mean that role-based security for any software developed for healthcare must be completely determined and tested. Now, although it's ideal when developing healthcare solutions to define all product requirements upfront, healthcare software products are typically complex; they will include integration with other systems and are often mission critical. Their complexity adds the problem of having the ability to define all the wants before any development occurring this example is where Agile becomes very useful [4]. There are a lot of challenges that developers face during the development of the healthcare projects. The major challenges in healthcare application development:

1- Confidentiality of Data: Every application concerning to healthcare or not, consists and gathering data from the users. In the case of healthcare applications, they generally include data which is more sensitive. For example, information monitoring, health readings, diagnosis for each person. The Health Insurance Portability and Accountability Act (HIPPA) guaranty a set of guidelines that ensures privacy and security of information. Security of patient's data and information is an evident requirement of a healthcare system and thus the system should have subject with HIPPA. It is important to a plan that depicts the critical procedures and steps to safeguard the patient's information. How the data will be collected, what data is required for the system to efficiently function, and limit of data to main information only [5].

2. Managing the Big Data: within the healthcare sector, big data brings within the advantage of predicting epidemics, improving the standard of life, and developing medicines for curing disease. There are increasing in data elicited in medical, and healthcare fields and therefore the growing availability of data at the central location which will be utilized in other organizations associated with the healthcare system like insurance business and pharmaceutical firms which making healthcare organizations and every one organization related it have an enormous amount of knowledge that must be organized and manage in order that it are often utilize. Although this data consider because the key to extend the efficient performance the services within the healthcare field, huge data achieve within the advantage of predicting pestilences, improving the quality of life, and developing medicines for curing disease. But the safety and privacy issues are increasing Health organization are unable to benefit from it [6].

3. Handling Market Volatility/Trends: Every day, we acknowledge novel trends in the healthcare field that lead to the volatility, uncertainty, complexity, and ambiguity in the majority of healthcare sectors. This continuous volatile is contributed by change in living practices, the diversity of devices that are being introduced, new technologies, ideas, and compliances. Healthcare transformation is producing revolutionary, so Health sector leaders have to deal with all scenarios. Thus, keeping pace with the trends and preserving the healthy performance at the same time is primary to stay ahead in the field [7, 8].

4. Integration with Existing IT System: For transforming the healthcare system to a different system, it is important to be done within or across the organizational boundaries. For example: A remote monitoring system should send information to an electronic medical record system, which can be then recognized by the electronic health record system. Following interoperability, It can be known as the ability of the health system to work in cooperation this chain of processing data, the healthcare system is not only efficient in working but it's cost effective as well [9].

3. RELATED WORK

The healthcare industry has evolved to present intelligent e-healthcare systems to support the service delivery as well as to achieve customer satisfaction. Recent research has explored the possibilities to build healthcare systems able Improving health services and facing challenges and changes in the environment and produce a suitable software solution .therefore Software development is constructed using a diversity of methods and models. In many previous studies the researchers they proposed software development methodologies for a healthcare field. For presenting innovative and fresh software development technique, it is assumed that it will enhance the efficiency and productivity of software designers, testers. Kadurin, Vladimir. They reviewed the significant and common methodologies used in software development with the highlight of the differences between them and present the Advantages and disadvantages for each methodology and how can benefit from them in different situations. Also proposed a new software development methodology that combines agile with a more structured waterfall like methodology. The new approach includes 4 distinct phases that are executed sequentially but have iterative patterns internally Analysis Plan, Architecture, Development, Validation and Verification. In order to benefit of both waterfall and agile approaches it specifically suits the requirements of the complex healthcare domain [4]. 209Lalband, Neelu, and D. Kavitha. They submitted a summarized discussion about different software development lifecycle models follow in the software industry is given. Although there are many development methods, they only focused on the common and basic methods in this field. And make a comparison between these methods and select agile methodology as best suited for developing the healthcare system. Also, their survey included a discussion on the expansion and transformation of the healthcare field In addition to challenges of intelligent healthcare. The issues that must be taken into account when developing the healthcare system [10].

Abbas, Raja Manzar, et al. the major goal of healthcare software development is to provide patients the best service possible by collecting and interpreting precise information. They focused on an initial definition of trustworthiness attributes from literature. Then they attempted to search about factors and reasons that determines trustworthiness in healthcare software solutions. Also there was proposed trustworthiness healthcare software model to backing healthcare technology stakeholders, that can be applied to appreciate the level of trustworthiness connected with healthcare software solutions [11]. Jabbar, Talha, et al. in the first they presented and discussed the challenges faced by developers of the healthcare system over the development process. secondly they proposed a SCRUM based model that aid software development team to build healthcare systems, utilized with knowledge management techniques for achieving the promote and maintain for quality standards required [12]. McCaffery, Fergal, et al. highlighted the issues in traditional medical device software development in the healthcare sector. And discussed the benefits and challenges that are likely to be observed when implementing waterfall/V-Model in medical device software development lifecycle. These models have been widely adopted in medical device software development, the authors tried to focus on the importance use of agile methodology in the medical device field to promote practitioners to evaluate the benefits of performing agile software development. Secondly, they discussed the challenges and focused on agile practices that have been successfully taken in the medical device software industry [13].

Table 1. Related work objectives

authors	objective
Kadurin, Vladimir,2021	Proposed a new software development methodology that combines agile with a more structured waterfall like methodology.
Lalband, Neelu, and D. Kavitha.2019	Survey included a discussion on the expansion and transformation of the healthcare field In addition to challenges of intelligent healthcare.
Abbas, Raja Manzar, et al.,2017	Proposed trustworthiness healthcare software model to backing healthcare technology stakeholders.
Jabbar, Talha, et al.,2019	Proposed a SCRUM based model that aid software development team to build healthcare systems.
McCaffery, Fergal, et al.,2016	Discussed the benefits and challenges that are likely to be observed when implementing waterfall/V-Model in medical device software development lifecycle.

4. SOTWARE DEVELOPMENT METHODOLOGIES

Software development is achieved using a variety of methods. These methods referred to as software development methodologies. Software development methodologies are acknowledged as blueprints that are used to plan, build, organize and control all development processes of a software [4]. These development methodologies are diverse, each has their own advantages and drawbacks and have gone through advancements throughout the years. Software engineering methodology can be defined as a bunch of techniques and tools utilized in the development of software. The Methodology should describe each phase of the software-development life cycle. Software methodology normally does not specify the details. These details usually left to the organization's needs and requirements [14].

Software development is usually done through several stages and these stages are [15]:

1. Problem identification and analysis.
2. Market Enquiries (Research)
3. Requirements extraction and collection.
4. Design.
5. Implementation.
6. Testing.
7. Deployment.
8. Maintenance and the fixing of bugs.

These stages are usually used as steps within the development methodologies. Some methodologies used these steps in a linear or consecutive manner and other methodologies use some of these steps or additional steps added to the ones above. In this paper, a discussion will be held about software development methodologies, their strengths and weakness. The software methodologies which will be covered in this paper are:

1. Waterfall model
2. Agile development methodologies

4.1 Waterfall Model

The waterfall development methodology has been a well-known and commonly used development method Since its definition in 1970 by Winston W. Royce [16]. This methodology uses the planned approach and is also known as the "linear-sequential life cycle model" [4]. The sequential phases in Waterfall model are as shown in (see Figure 1: Waterfall Model).

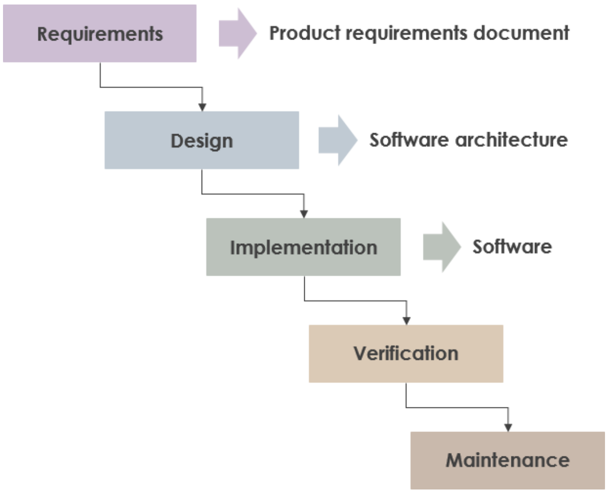

Figure 1: Waterfall Model

- **Requirement Gathering and analysis** – gather all possible requirements of the system to be developed.
- **System Design** − the requirement specifications from the first phase are studied during this phase and thus the system design is ready. The system design helps in defining overall system architecture also as specifying hardware and system requirements.
- **Implementation** − with inputs from the system design, the system is first developed in small programs called units, which are integrated in the subsequent phase.
- **Integration and Testing** − All the units developed within the implementation phase are integrated into a system after testing of every unit.
- **Deployment of system** − Once the functional and non-functional testing is done; the product is deployed within the customer environment or released into the market.
- **Maintenance** − There are some issues that come up within the client environment. To fix those issues, patches are released. Maintenance is accomplished to deliver these changes within the customer environment.

Some of the main advantages of the Waterfall Model are as follows [17]:

- Simple, easy to understand and use
- Easy to manage.
- Phases are processed and completed one at a time.
- Define stages clearly.
- Well understood well documented.
- Easy to rearrange tasks.

The main disadvantages of the Waterfall Model are as follows [17]:

- No working software is produced until late during the life cycle.
- High amounts of risk and uncertainty.
- Not a perfect model for object-oriented and complex projects.
- Poor model for long and ongoing projects.
- Not suitable for the projects where requirements are at a moderate to high risk of adjusting.
- It is difficult to measure progress within stages.
- Cannot accommodate changing requirements.

4.2 Agile Development Methodologies

Agile is that the ability to create and respond to change. It is a way of handling with, and ultimately succeeding in, an uncertain and turbulent environment. Agile methods were designed major to backing business application development where the system requirements usually changed quickly through the development process[18]. Agile software development is basically an iterative approach to project management and software development that focuses on incremental specification, design, and implementation, while requiring full integration of testing and development that helps teams deliver value to their customers faster and with fewer headaches [19]. Requirements, plans, and results are evaluated continuously so teams have a natural mechanism for responding to change quickly through collaboration between self-organizing, cross-functional teams. Agile methods have transformed and brought unique changes in software development practice by strongly emphasizing team collaboration, customer involvement, and change tolerance [20].

It promotes adaptive planning, evolutionary development and delivery, a time-boxed iterative approach, and encourages rapid and elastic response to change [21]. Each time-boxed iteration within the agile framework includes planning, requirement analysis and design, coding, and testing.

The techniques of agile software development have become popular during the last few years. For traditional Project Management, the agile methodology is widely used in software development to increase the quality of a project as well as to enhance customer satisfaction [22]. The traditional techniques (Waterfall, Unified Process, prototyping Model and Spiral Model) aren't capable for software development nowadays because new requirements are happening within the market. The new software development techniques include **XP, Scrum, Crystal, FDD, DSDM,** and **ASD** [23].

The aim of agile methodology is to help software teams to think differently, work iterations. In order to obtain this aim, agile manifesto was proposed with four core agile values leading to twelve principles. The composition of agile values is as follows efficiently, deliver on time, keep learning and re-learning from previous [24]. The Agile Manifesto highlights four values [25]:

1. Individuals and interactions over processes and tools.
2. Working software over comprehensive documentation.
3. Customer collaboration over contract negotiation.
4. Responding to change over following a plan.

An agile software development process always starts by defining the users and documenting a vision statement on a scope of problems, opportunities, and values to be addressed. The product owner captures this vision and works with a multidisciplinary team (or teams) to deliver on this vision. The roles in that process are: User, Product Owner, and Software development team. The main advantages that the agile-based software development achieved [26] [27] are:

- Improved communication and coordination among team members.
- The quick releases.
- The flexibility of design.
- More reasonable process.
- Flexible enough to update the life cycle depending on changing requirements from users
- Customer satisfaction is more thanks due to quick incremental releases
- Customer feedback after every release produces a high quality product.
- Saves time due to brief documentation
- Less risky due to frequent customer feedback
- Quickly find and fix bugs

The main challenges of the agile-based software development are:

- Customer ought to be clear about the product and its features
- Difficult for new developers in team to understand the brief documentation
- Waste of time and resources and efforts too if the release isn't up to the customer expectation

Various types of agile methods exist. These methods share the common ideas and values but with delicate characteristics between them [2].

5. HYBRID SCRUM-FALL MODEL

Hybrid Scrum-Fall model is formed based on an integration of the Waterfall and Agile software development paradigms taken into consideration the critical key points for the success of the healthcare system. The principal goal of IT application in healthcare is the quality improvement of medical care. Therefore, the healthcare system has to be characterized by a high reliable system, stable, secure, stable, high performance and high availability.

Hybrid Scrum-Fall model can be useful in healthcare software projects because it uses the advantages of both waterfall and agile models using Scrum. Some of new steps are involved in the proposed model as well as some of these steps are merged together. For large software projects require a long-term development time, this model is suitable for such projects [28].

Hybrid Scrum-Fall model combines the major characteristics of Waterfall model and Agile. Such that, waterfall can work well with large scale domain such as health care systems, achieve high security, steps occur in sequence, easy in implementation, usage and management as well as defining and documenting requirements clearly [29]. Agile working methods are iterative, collaborative, time-bounded, easy to change as well as it can be seen as a consequence of the need for faster, more output-focused product development and is now part of strategies for managing the healthcare digital enterprise transformation. As an agile working method Scrum plays a central role in healthcare development. (see **Error! Reference source not found.**) shows an overall view of the proposed development model.

Analysis Healthcare domain: the healthcare development team tries to understand the general area where the system is applied. As well as identifying the relevant objects of a healthcare application domain. Development team recognizes the domain of healthcare.

Requirements Elicitation: This is the most important phase as its main purpose is to bridge the healthcare-specific knowledge with the technical execution of the end product. The product development team analyzes the requirements based on the participation of healthcare specialist, and fully understands the problems. In healthcare domain accuracy and high quality are extremely important as it involves human lives and health. This phase will also set up the proper communication channels for healthcare personal and development team in order to resolve any future issues that may appear. The deliverable of this phase will be a detailed document of all requirements that is written in a clear enough manner that is easily understandable by non-medical people. This phase includes no building. The team attempts to ask all the questions and secure all the answers they need to build the product requirement [4]. At this phase the team should:

1. Determine the product vision and functional requirements.
2. Do feature brainstorming, priorities, and work planning.
3. Understand the details of the specific healthcare customer problem.
4. Understand how systems interact and contribute to overall business goals.
5. Understand the needs and constraints of system stakeholders.
6. Understand the specific needs of people who require system support in their work.

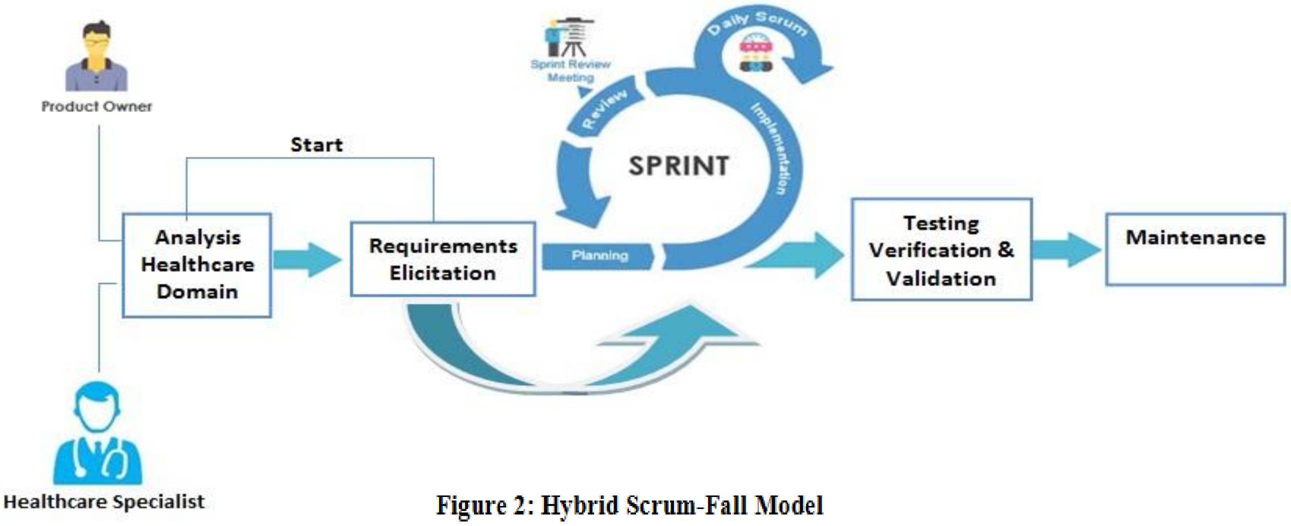

Figure 2: Hybrid Scrum-Fall Model

Sprint: Scrum is empirical based agile framework, structured on the basis of control theory of empiricism under the canopy of agility, which helps in addressing complicated processes and increased productivity and quality of software product development. scrum approach is based on the precepts and values of the agile statement ,which promote a various style for managing software development tasks, cooperation over contract negotiation, and responding to changes fo overall plan [30]. The sprint duration is very significant so that the user stories are as small as possible. The typical average sprint period lasts about 2 weeks. If the sprint period is smaller than the feature is that more user feedback can be received and the errors and bugs can be handled and addressed in early time as possible. If the sprint period is longer then it permits the developer to work totally. The next phase is to do the sprint backlog creation for which the scrum team must choose the important customer stories and make them into smaller tasks. They necessity to plan on how to obtain the task accomplished. The actual user stories are shifted as small tasks in the sprint backlog where the actual work starts. In this step the scrum meetings are important as it is done to track the work status and who is doing what status. The tasks completed are to be investigated as a working product with full life cycle testing. Every sprint that is finished must be presented and explained to the customer for his approval and his standpoint on the complete solution [31].

Testing – Verification & Validation: in this phase the validation and verification is performed by a joint team of developers, quality assurance engineers, end users and healthcare staff. This ensures that all aspects of the requirements will be reviewed - both from technical and medical side.

Maintenance: The final phase of the Scrum-fall model is maintenance. This is to make sure that the software is operating at optimum levels of performance. This can be concerned with bugs in the software, security issues or to address issues that have only come to light when the new system was implemented. Activities in this phase include obtaining maintenance requests from the end users, transforming these requests into changes, designing the changes and finally implementing said changes development and management.

6. CONCLUSION

Healthcare Software development organizations are seeking to optimize their Software Engineering Management practices. Scrum is trending agile methodologies for software project

This research introduced a hybrid software development model – (Hybrid Scrum-Fall model). Scrum-fall model is the integration of the waterfall model and scrum in the form of a hybrid framework to overcome the limitations of waterfall and

Scrum; and empower the strength of the healthcare software developer. Which has a great strength compared to stand alone framework and capabilities to answer the challenges of software development based on involvement of external stakeholder – healthcare specialist. It includes 5 distinct phases: Analysis healthcare domain, Requirements Elicitation, Sprint, Validation and Verification and Maintenance. This brings the benefits of both waterfall and agile approaches. It specifically suits the needs of the complex healthcare domain. Future research in this area can be done on the nature and characteristics of healthcare systems and the collaboration between healthcare specialists and software engineers within the timeframe of healthcare project duration.

7. FUTURE WORK

For future work, we intend to operate and test our proposed model on the real case study. Another useful approach would be replacing the currently used methods for software development process by proposed framework, which in turn will give more accurate solution and better performance for the whole system.

8. REFERENCES

[1] Y. B. Leau, W. K. Loo, W. Y. Tham, and S. F. Tan, "Software development life cycle AGILE vs traditional approaches."

[2] P. K. Yeng, S. D. Wolthusen, and B. Yang, "COMPARATIVE ANALYSIS OF SOFTWARE DEVELOPMENT METHODOLOGIES FOR SECURITY REQUIREMENT ANALYSIS: TOWARDS HEALTHCARE SECURITY PRACTICE," 2020.

[3] S. Tolf, M. E. Nyström, C. Tishelman, M. Brommels, and J. Hansson, "Agile, a guiding principle for health care improvement?," *International journal of health care quality assurance,* 2015.

[4] V. Kadurin, "A new proposed software development methodology for healthcare industry," in *AIP Conference Proceedings*, 2021, vol. 2333, no. 1, p. 030010: AIP Publishing LLC.

[5] K. Abouelmehdi, A. Beni-Hessane, and H. Khaloufi, "Big healthcare data: preserving security and privacy," *Journal of Big Data,* vol. 5, no. 1, pp. 1-18, 2018.

[6] K. Abouelmehdi, A. Beni-Hssane, H. Khaloufi, and M. Saadi, "Big data security and privacy in healthcare: a review," *Procedia Computer Science,* vol. 113, pp. 73-80, 2017.

[7] P. Turner, "The ecology of healthcare," in *Leadership in Healthcare*: Springer, 2019, pp. 17-43.

[8] A. Kumari, S. Tanwar, S. Tyagi, and N. Kumar, "Fog computing for Healthcare 4.0 environment: Opportunities and challenges," *Computers & Electrical Engineering,* vol. 72, pp. 1-13, 2018.

[9] X. Liang, J. Zhao, S. Shetty, J. Liu, and D. Li, "Integrating blockchain for data sharing and collaboration in mobile healthcare applications," in *2017 IEEE 28th annual international symposium on personal, indoor, and mobile radio communications (PIMRC)*, 2017, pp. 1-5: IEEE.

[10] N. Lalband and D. Kavitha, "Software engineering for smart healthcare applications," *International Journal of Innovative Technology and Exploring Engineering,* vol. 8, pp. 325-331, 2019.

[11] R. M. Abbas, N. Carroll, I. Richardson, and S. Beecham, "The Need for Trustworthiness Models in Healthcare Software Solutions," in *HEALTHINF*, 2017, vol. 5, pp. 451-456.

[12] T. Jabbar, Y. Hafeez, A. A. Kiani, N. Anwar, and J. Javaid, "Use of Knowledge Management and SCRUM techniques to increase the reusability in software development," in *2019 13th International Conference on Mathematics, Actuarial Science, Computer Science and Statistics (MACS)*, 2019, pp. 1-5: IEEE.

[13] F. McCaffery, K. Trektere, and O. Ozcan-Top, "Agile–Is it Suitable for Medical Device Software Development?," in *International Conference on Software Process Improvement and Capability Determination*, 2016, pp. 417-422: Springer.

[14] A. Al-Dahmash and S. El-Masri, "A new proposed software engineering methodologyfor healthcare applications development," *Int. J. Mod. Eng. Res,* vol. 3, no. 3, pp. 1566-1570, 2013.

[15] C. Jones, A guide to selecting software measures and metrics. CRC Press, 2017.

[16] B. Davis, Agile practices for waterfall projects: Shifting Processes for Competitive Advantage. J. Ross Publishing, 2012.

[17] S. Shylesh, "A study of software development life cycle process models," in *National Conference on Reinventing Opportunities in Management, IT, and Social Sciences*, 2017, pp. 534-541.

[18] D. Mohamed and N. R. Darwish, "Extracting CRM requirements–Waterfall or Agile: A comparative study," International Research Journal of Advanced Engineering and Science, vol. 4, pp. 1-5, 2019.

[19] F. Lonetti and E. Marchetti, "Emerging software testing technologies," in *Advances in computers*, vol. 108: Elsevier, 2018, pp. 91-143.

[20] Ö. Uludag, M. Kleehaus, C. Caprano, and F. Matthes, "Identifying and structuring challenges in large-scale agile development based on a structured literature review," in *2018 IEEE 22nd International Enterprise Distributed Object Computing Conference (EDOC)*, 2018, pp. 191-197: IEEE.

[21] S. Singh, I. J. I. J. o. u.-a. e.-S. Chana, Science, and Technology, "Introducing agility in cloud based software development through ASD," vol. 6, no. 5, pp. 191-202, 2013.

[22] S. A. Butt, "Study of agile methodology with the cloud," *Pacific Science Review B: Humanities and Social Sciences,* vol. 2, no. 1, pp. 22-28, 2016.

[23] P. Mani and S. Deebitha, "Analysis of agile software development utilising cloud computing capabilities," ed: Nr, 2014.

[24] A. Manifesto, "Manifesto for agile software development, 2001," *URL: http://agilemanifesto. org,* 2010.

[25] P. Rodríguez, M. Mäntylä, M. Oivo, L. E. Lwakatare, P. Seppänen, and P. Kuvaja, "Advances in using agile and lean processes for software development," in *Advances in Computers*, vol. 113: Elsevier, 2019, pp. 135-224.

[26] S. Al-Saqqa, S. Sawalha, and H. AbdelNabi, "Agile Software Development: Methodologies and Trends," *International Journal of Interactive Mobile Technologies*, vol. 14, no. 11, 2020.

[27] D. Jagli and S. Yeddu, "CloudSDLC: Cloud Software Development Life Cycle," *International Journal of Computer Applications*, vol. 168, no. 8, pp. 6-10, 2017.

[28] M. S. J. D. S. J. Gharajeh, "Waterative model: An integration of the waterfall and iterative software development paradigms," vol. 10, pp. 75-81, 2019.

[29] A. Lawal and R. C. J. B. J. O. G. S. Ogbu, "A COMPARATIVE ANALYSIS OF AGILE AND WATERFALL SOFTWARE DEVELOPMENT METHODOLOGIES," vol. 11, no. 2, pp. 1-2, 2021.

[30] N. R. Darwish and S. Megahed, "Requirements engineering in scrum framework," International Journal of Computer Applications, vol. 149, pp. 24-29, 2016.

[31] K. Schwaber and J. Sutherland, "The scrum guide. Scrum. org. 28.08. 2017," ed, 2016.