Framework for Enhancing Requirements Engineering Processes: a Conceptual view of Health Information System

Ishaya Gambo
Obafemi Awolowo University
Computer Science &
Engineering Department
Ile-Ife, Nigeria

Abimbola Soriyan
Obafemi Awolowo University
Computer Science &
Engineering Department
Ile-Ife, Nigeria

Rhoda Ikono
Obafemi Awolowo University
Computer Science &
Engineering Department
Ile-Ife, Nigeria

ABSTRACT
Requirements engineering (RE) involves understanding the needs of stakeholders, the contexts in which the to-be-developed software will be used; analysing, negotiating, and documenting the stakeholders' requirements; and managing requirement evolution. This is achieved through a process called requirement engineering process. Therefore, the ability to improve software products in order to meet the needs of these stakeholders within today's challenging and fast-paced environment is a great concern in the software development industries. This paper is aimed at presenting an enhanced framework for reasoning about RE process in terms of skill enhancement within the scope of software engineering (SE) and information system development (ISD) using the health information system (HIS) as an example to conceptualize the idea. The paper justifies requirement engineering as a process like all other SE and ISD activities to be adapted to the needs of the processes, the products, the projects and the people doing the work in a conceptual framework for quality improvement. The paper conceptualized this by considering the university as the domain where information system, software system, and healthcare research is carried out for knowledge acquisition that proffers solutions to the software organisations where skills are enhanced. This is with the view to improve the customer organisations where the products, processes and services are used. The framework strive to suggest the enhancement of RE process skills to what the stakeholders wants, which serve to establish a solid foundation for the design and construction of what the stakeholders' gets. As such, a framework to enhance the ability to develop software products for the healthcare domain with the aim of meeting the needs of all system stakeholders and to ensure that the development process is going on the right tract is very important.

General Terms
Requirements Engineering, Information System Development, Software Engineering, Health Information System, MINPHIS.

Keywords
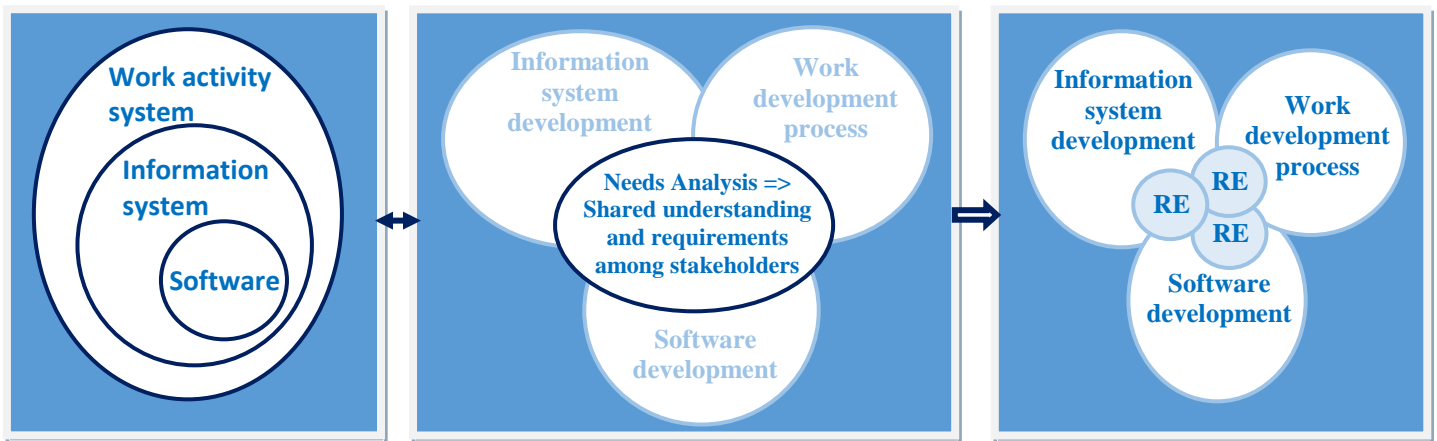Requirements, Requirements Engineering Process, Health Information System, MINPHIS.

1. INTRODUCTION
Every system needs requirement! Without it, the extent to which the system is expected to perform and achieve its purpose is impossible. In addition, errors in these requirements are the most expensive to fix when found during the development process of any system. Not only are requirements the most expensive, but also the most frequent problems. Before developing a system we need to understand the requirements the system should fulfill. This cuts across the various types of information system to be developed, where software development and application is key.

However, advancement in the information technology industry has engendered the requirements for software systems to become more and more motivated, useful and applicable. This is because, the software industry is seen as one of the fastest growing and most profitable industry due to its profound impact at the societal level. Despite many failures with some of these software systems that are well publicised and in use, the engineering of such systems has improved consistently over the past few decades. In particular, the success of a software system depends on how well it fits the needs of its users and its environment. Software requirements comprise these needs, and Requirement Engineering (RE) is the process by which the requirements are determined [1]. In essence, requirements drive almost every activity, task, and deliverable in the engineering process of software systems, specifically, in a software development project and Information Systems Development (ISD) practices.

Successful RE involves understanding the needs of users, customers, and other stakeholders; understanding the contexts in which the to-be-developed software will be used; modeling, analysing, negotiating, and documenting the stakeholders' requirements; and managing requirement evolution. Figure 1 shows that shared understanding among various stakeholders is the basis for requirements, and RE process is needed for producing the specific requirements in the development process. In this sense, a shared understanding of the target domain is require to serve as the basis for successful development. According to [2], the Needs Analysis is conducted jointly, producing descriptions and models that can serve as the starting-point of the different development activities of software applications, information systems, or work. These descriptions serves as the basis for analysis, negotiation, validation and then to documentation of requirements. This makes the RE a collaborative process and activity, which comes with challenges. In the work of [3], the focus of RE is changing from the software system to human activities that will be supported by it. This brings an interesting insight into workflow processes. [3] Opined that "most RE problems occurs at the point where humans come fully on the scene. In particular during elicitation, negotiation and elaboration of requirements in the intended organisational context of the system's use.

Fig 1: Shared understanding as the basis for requirements (adapted and modified from: [2])

In [4] RE was defined as "the branch of software engineering concerned with the real-world goals for functions of and constraints on software systems". Under this conception "information systems are seen as fulfilling a certain purpose in an organisation and requirements engineering helps in the conceptualisation of these purposeful systems" [5]. As observed in most ISD practices, RE in highly multidisciplinary projects involves lot of stakeholders with different backgrounds and perspective, for example in hospitals, we have the medical or healthcare professionals, the administrators at the managerial level, the technical staff etc. This makes the process knowledge-intensive [6] and human intensive [7] thereby demanding an extensive analysis and tradeoff to be performed. In essence, RE lays the foundation for successful software and system development projects regarding cost and quality [8]. The activities, which are performed as part of the RE process, aim at the discovery and specification of requirements that unambiguously reflect the purpose of a software system as well as the needs of all relevant stakeholders [9]. This paper is aimed at presenting an enhanced framework for reasoning about RE process within the scope of software engineering (SE) and ISD using the health information system (HIS) as an example to conceptualize the idea.

Holistically, the purpose of any information system like the health information system (HIS), is to support work activities, and assist the achievement of the goals of work. These activities includes capturing, storing, retrieving, managing or transmitting information related to the health of individuals or the activities of organisations that work within the health sector. These goals are majorly captured or embedded in the workflow system, which specifies the various processes and activities. The authors in [10] opined that "Information systems are developed through a process that includes activities for analysis, design, implementation, introduction, and sustained support, as well as process management". These development activities are majorly dependent on the requirement engineering processes that seeks to elicit individual stakeholder's requirements and needs and developing them into detailed, agreed requirements documented and specified in such a way that they can serve as the basis for all other system development activities. In this sense, Requirements Engineering (RE) is the activity that mediates the needs of the system users to the developers who,

in turn, design and build the solutions [11]. In the works of [12], [13], [14], [15], [16] and [17], it was established that most methods for requirements engineering are designed for supporting software development, which can also be applicable to ISD. As an example, the development of health information system is not only the concern of software developers and/or professionals, but also involve other stakeholders and/or end users. With this, it becomes necessary to state that the early phases of development is very crucial and critical to the overall success of the system in terms of analysis, design, implementation, deployment, maintenance and continuous use. This justifies the observation in [11] that 'cooperatively obtained shared understanding of the needs for information systems, work processes and software systems should be a basis for the development'. This paper justifies requirement engineering as a process like all other SE and ISD activities to be adapted to the needs of the processes, the products, the projects and the people doing the work in a conceptual framework for quality improvement. The paper conceptualized this by considering the university as the domain where information system, software, and healthcare research is carried out for knowledge acquisition that proffers solutions to the software organisations where skills are acquired, towards improving the customer organisations where the products, processes and services are used. The framework strive to suggest the enhancement of RE skills to what the stakeholders wants, which serve to establish a solid foundation for the design and construction of what the stakeholders' gets. Therefore, enhancing the RE process will engender quality improvements.

2. MOTIVATION
In practice, developing information systems and/or software systems can be so compelling and demanding that many developers want to jump right in before they have a clear understanding of what is needed. The argument has always been that things will become clear as they develop; that project and system stakeholders will be able to better understand need only after examining early iterations of the software; that things change so rapidly that requirements engineering is a waste of time; that the bottom line is producing a working program and that all else is secondary. This paper argues that each of these is flawed, and all can lead to a failed project if a process is not in place.

Consequently, in order to start quality control and assurance activities earlier, to have more quality control and assurance levels, to resolve early conflicting expectations of different stakeholders, requirement engineering process becomes inevitable. This is why [18] indicates that "requirements engineering process critically influences the success of software development process". Despite the significant varieties of software development processes and project management techniques, software projects may still fail due to vague definitions of project goals, ambiguous requirements, communication and coordination problems (e.g. conflicts that leads to omission in requirements and inconsistencies of requirements) among the stakeholders. Apparently, problems that associates with software project failure in software development processes are majorly related to requirements engineering areas like capturing, analysing, specifying, and managing requirements. In fact, when improving the software engineering process, the area, which may have the largest effect on the result, is RE. This is because, requirements are the first things produced, and projects are conducted and finalized in strict concordance with them. So, the ability to improve the software product in order to meet the needs of all stakeholders in information systems development is a great concern in the software industries. Therefore, requirements engineering process is required to provide the appropriate mechanism for quality assurance ([19], [20]) in the development process of health information system (HIS). As such, a framework to enhance the ability to develop software products for the healthcare domain with the aim of meeting the needs of all system stakeholders and to ensure that the development process is going on the right tract is very important.

3. HEALTH INFORMATION SYSTEM AND REQUIREMENT ENGINEERING PROCESS: THE NIGERIAN CASE

Health information system is a service system that comprises multiple types of organisation. An inside view of this reveals a multitude of different types of actors, information, and information systems that are highly regulated and governed by legislation. For the Nigerian healthcare system, HIS has gone through a well-defined manual structure and is being provided by collaborative teams that involve multiple healthcare providers at different levels in terms of delivery. In the work of [21], three tiers of healthcare delivery system implementation were identified to include: the federal, state and the local government. The federal government is responsible for tertiary healthcare delivery, the state governments is responsible for secondary care delivery (general hospitals), and the local government is responsible for primary healthcare. These three tiers or levels of healthcare delivery follow a well-structured manual information flow as shown in figure 2. For instance at the Primary Healthcare (PHC) level, [21] quoted the National Health Policy as having the most important objective to "Set up the nation's PHC system using the District Health System Approach". This approach presents the local communities as the centre of the PHC system, with several health related services at their disposal ranging from environmental services and traditional healthcare to the "modern" healthcare delivery system. Information flow through three record systems which [21] identified as:

- Home based records. These involves patients being issued with individual PHC cards to facilitate fast record searching and treatment whenever a visit is done to the health facility.

- Community based records. These are kept by voluntary health workers in the community.

- Institution based records: These are kept in the health facility and include records of activities carried out on daily, monthly and yearly basis.

These authors in [21] observed that the other tiers of healthcare delivery (i.e. secondary and tertiary) follow almost similar patterns except that community based records are not kept. Information flow follows a defined pattern from the community level up to the federal level. Presently, the Made in Nigeria Primary Healthcare Information System (MINPHIS) is one of the appropriate software package for African hospitals and health centers developed with the peculiar nature of Nigeria and Africa in mind.

In general, healthcare anywhere in the world is information-intensive, process and service oriented. This is because, vast amounts of information is created in the course of a treatment process and service demand on this information is crucial in the actualization of the entire system. As shown in figure 2 for the Nigerian Healthcare information flow, the cooperative work of healthcare professionals is dependent on the inclusion of in-deterministic and unpredictable workflows where information plays a significant or even critical role. Viewed from the perspective of software engineering and ISD practices, RE process will be concerned with how to elicit, capture, analyse, negotiate, elaborate, prioritise, validate, manage, and document these information during development. Evaluating value and/or cost of different requirements through analysis, resolving conflicts between requirements, deciding which to accept by setting priorities and identifying dependencies between requirements is still an open research issue that will improve the quality of healthcare systems. Analysing the requirements for such complex systems is difficult without a due process of discovery and inventions through appropriate skills. According to [22], the struggle with such complexity starts in the "frontend activities" which precede the actual development activities. However, there is a lack of guidance and methods supporting the early phases ([23], [22]).

In addition, the healthcare system is an increasingly complex information system that is used for supporting the healthcare organisations in many ways. The development of these system is not only the concern of software developers and/or professionals, but also involve other stakeholders and/or end users. Consequently, it is necessary to state that the early phases of development is very crucial and critical to the overall success of the system in terms of analysis, design, implementation, deployment, maintenance and continuous use. The author in [11] opined that 'cooperatively obtained shared understanding of the needs for information systems should be a basis for the development'. However, eliciting or gathering the right data and/or information within this collaborative and cooperative development environment is challenging, as there is no framework available to salvage the problem during requirement engineering process. This is because, ensuring and managing the various viewpoints, goals and expectations from the different stakeholders in the development process is key to the overall success of the system. For the healthcare system being a complex and multifaceted domain, requirements engineers are prone to face the challenge of communication with distributed stakeholders groups. Even if the communication is smooth, the challenge to understand the information as collected during elicitation and negotiation processes still surfaces. When this happens, the

effect will be seen during development as software developers might not be able to capture the exact intent, viewpoints, and

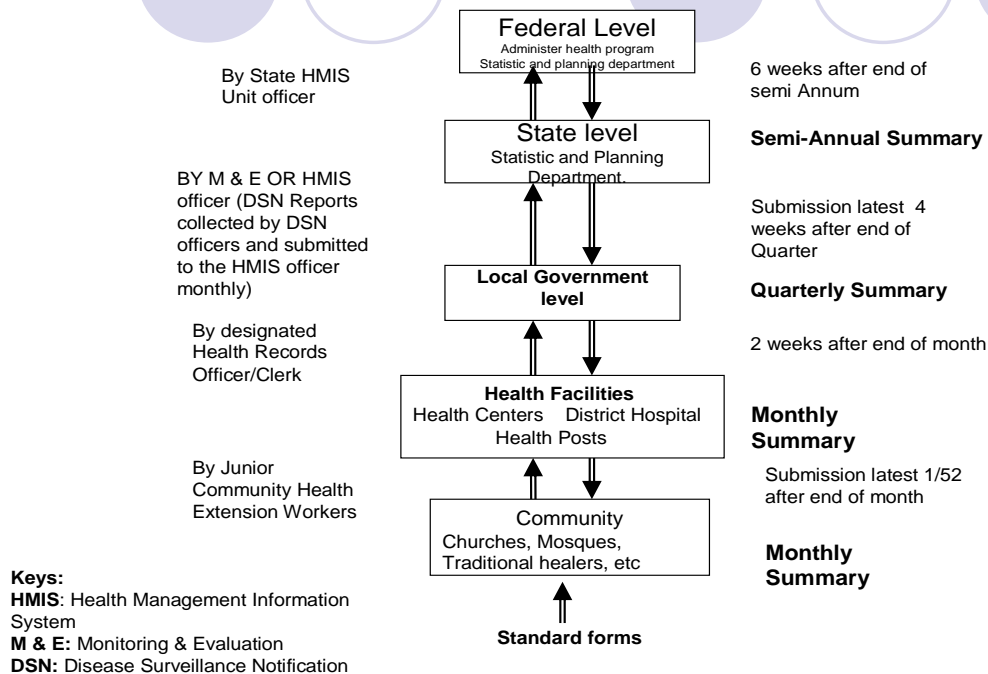goals of the users for the system.

Fig 2: The Nigerian Healthcare Information flow (Source: [24])

More importantly, creating and maintaining software is a design activity [25]. In the same way, creating and maintaining HIS is a design activity, which must be preceded by the requirement engineering activities. In the work of Neil Alexander Ernst [26], three elements involved in the design activity were identified. First is the domain assumptions under which the software will operates. This consists of the properties that are assumed to hold in the domain and are exploited in designing the software. The second is the requirements the software must meet for its users to accept it. This consists of the desired set of new properties of that domain. The third one is the specification that consists of detailed plan for implementing the software. These three elements are related: if the software is developed according to the specification, taking into account the domain assumptions, the software system will satisfy the requirements [27]. All of these elements are issues of concern at the requirement engineering phase. More to this is the fact that several surveys in literature indicate that around 50% of all software project problems are related to requirements. In addition to this, the typical outcome of these problems is an expectation gap where the difference between what developers build and what customers really need becomes an issue of concern. Specifically to the Nigerian case, one of the primary goal for the Made in Nigeria Primary Healthcare Information System (MINPHIS) development is to meet the needs of all stakeholders involve in the development process. In enhancing the ability to develop this product with the aim of meeting the needs of all system stakeholders, requirement engineering process becomes inevitable..

4. MADE IN NIGERIA PRIMARY HEALTHCARE INFORMATION SYSTEM (MINPHIS)

The authors in [28] reported that MIHPHIS is a software package developed within the INDEHELA projects (Methods for Informatics Development for Health in Africa) in Nigeria and has been utilized in a number of Nigerian Teaching and Specialist Hospitals. It is principally a hospital patient information system. The system was originally installed on a PC server with 3 dumb terminals back in 1991. The 2nd generation of the system implemented in 1998 was based on more powerful servers running Microsoft NT, Intersystems Caché, the VA Kernel and FileMan, and the FixIT software developed in Finland for the MUSTI system. It was upgraded and enhanced for potential national use [29]. The MINPHIS application keeps patient records and generates various reports for health management and research purposes. The reports include the patient status, medical history and admissions plus indicators like length of stay per patient, discharge summaries, mortality and morbidity data, and operations. The application can answer ad hoc queries from medical researchers (e.g. cases of cholera for a period per geographical location for specific age group or sex or both). It can also provide performance information relevant to particular health care professionals, such as the mortality rates for patients treated by a particular staff member. MINPHIS has spanned through a thorough Software Engineering and Information System Development Processes with clinical and patient information well taken care of via a wide range of reports that could aid health policy and decision makers. In an

action research, [24] conducted a study on Information Systems Development in Nigerian Software Companies with the aim of identifying factors that affect development practice, the development activities, the method(s) and tools employed in the Nigerian software industry, the type of education received and other relevant factors in such settings using MINPHIS as one of the real life software development experience. All the factors identified can be enhanced through RE processes in terms of skill enhancement in the software industry.

5. REQUIREMENTS ENGINEERING PROCESSES

In practice, a software is considered to be successful if it meets the objective for which it was made and to identify that objective we perform RE [9]. The process includes; Identification of stakeholders, gathering their needs and understanding them, documenting the specifications and analysing it for subsequent implementation [30]. RE can be seen as being made up of two phases. These phases includes; Requirement Development, which includes activities like, Requirement Elicitation, Analysis, Specification and Verification; and Requirement Management that deals with managing and controlling changing and evolving requirements. The work in [31] suggest that the aim of RE is to introduce engineering principles into the practice of traditional information systems analysis. Therefore, a systematic and disciplined process should be followed [32]. The RE process should thus consist of structured and repeatable activities.

However, RE process is naturally collaborative in nature where agreements among the stakeholders is very crucial and broad mastering is generally infeasible for a single person ([33], [34]). In most cases the problem of unclear descriptions of requirement always occur, which often leads to terminological misalignment between analyst and stakeholders, and then potentially wrong requirements [35] are elicited. As a result, negative consequences of these wrong requirements are felt during all downstream activities such as architecting, design, implementation, and testing. As observed in [36], Poor-quality requirements greatly increase development and sustainment costs and often cause major schedule overruns. More to this problem is the elicitation of partial information due to stakeholder's omissions [37]. As such, the RE process becomes valuable when followed in evolving a system.

6. PROPOSED FRAMEWORK: CONCEPTUAL MODEL

The proposed framework recognised the on-going larger research effort undertaken by the research network Informatics Development for Health in Africa (INDEHELA). The goal has been to understand and solve real-life problems for Information Systems Development (ISD) with particular interest in healthcare. It is on this premise that the RE process is introduced as shown in figure 3. The framework presents a conceptual model for reasoning about major issues in development practice and activities, the type of education received at the university level in engendering quality research in information systems, software and healthcare. This framework indicates that the difficulties in ISD that relates to the problems of unstable, constantly changing requirements, poor requirements specification, client behaviour, such as approval delays, requirements changes, and poor communication, especially in multi-cultural and multi-disciplinary environments like the healthcare domain can be

overcome by devising measures to improve systems quality and customers' satisfaction, ease compliance with standards and regulations, reduce project cost and delay, ensure better control of complex projects and improve team communication through Requirement Engineering Process (REP). At the software project level, there is need to establish a common understanding of the requirements to be addressed by the software product and to define the requirement in the process of analysing the desired product. All these put together can be tailored towards closing the gap in the industry and university level through Information System (IS) and Software Engineering (SE) education (research and development inclusive).

In the context of ISD and SE, many approaches to software project development process have been formulated. In spite of their differences, virtually all of them include the Requirements Engineering (RE) phase. This is because, inaccurate, inadequate, or misunderstood requirements are the most common causes of poor quality, cost overruns and late delivery of software systems [38]. Requirements problems are widely acknowledged to reduce the quality of software and to impact on the effectiveness of the software development process [39]. As observed in [40] and the Standish Group Report [41], Information System (IS) projects deliver 42 to 67% of their original requirements and 48% of Information Technology (IT) development relates to requirements engineering. Many ISD problems could be traced to RE ([42]; [43]; [44]; and [45] and estimates suggest incorrect requirements are expensive to correct during implementation ([40] and [46]). Consequently, RE is the most important activity of the software development life cycle (SDLC) and significantly affects the success or failure of information systems ([47; [48]; and [49]).

Within the premise of the framework, the University, Software organisations and Customer organisations are main entities through which the reasoning about RE process is introduced. This is with the view of improving quality during development across all the entities. The university as an entity is the domain where information systems, software and healthcare related research and development is needed. Basically, we want to know 'how research and development can close the gap in the Industry and university' for ISD, software systems and HIS as a whole in order to ensure quality improvement. To close this gap, we proposed that adequate skill in RE process is required. Such skill when acquired and enhanced, will proffer solution for software projects at the software organisational level. The skills at this level will help software organisation to establish a common understanding of the requirements to be addressed by the software product, and will also help in defining the requirements in the process of analysing the desired product. In between the software organisation and customers' organisation is the ISD. At this points adequate skills in RE process will help to address the problems of unstable, constantly changing requirements, poor requirements specification, client behavior, requirements changes, and poor communication, especially in multi-cultural and multi-disciplinary environments like the healthcare domain through research and development. At the customer organisation's level, the focus is on the services render by the citizens and communities. At this point, the emphasis is on the product use. In particular, the MINPHIS. Here we want to know how the RE process of MINPHIS development can be improved. Obviously, introducing the RE process within this framework of reasoning has great potential to engender quality improvement.
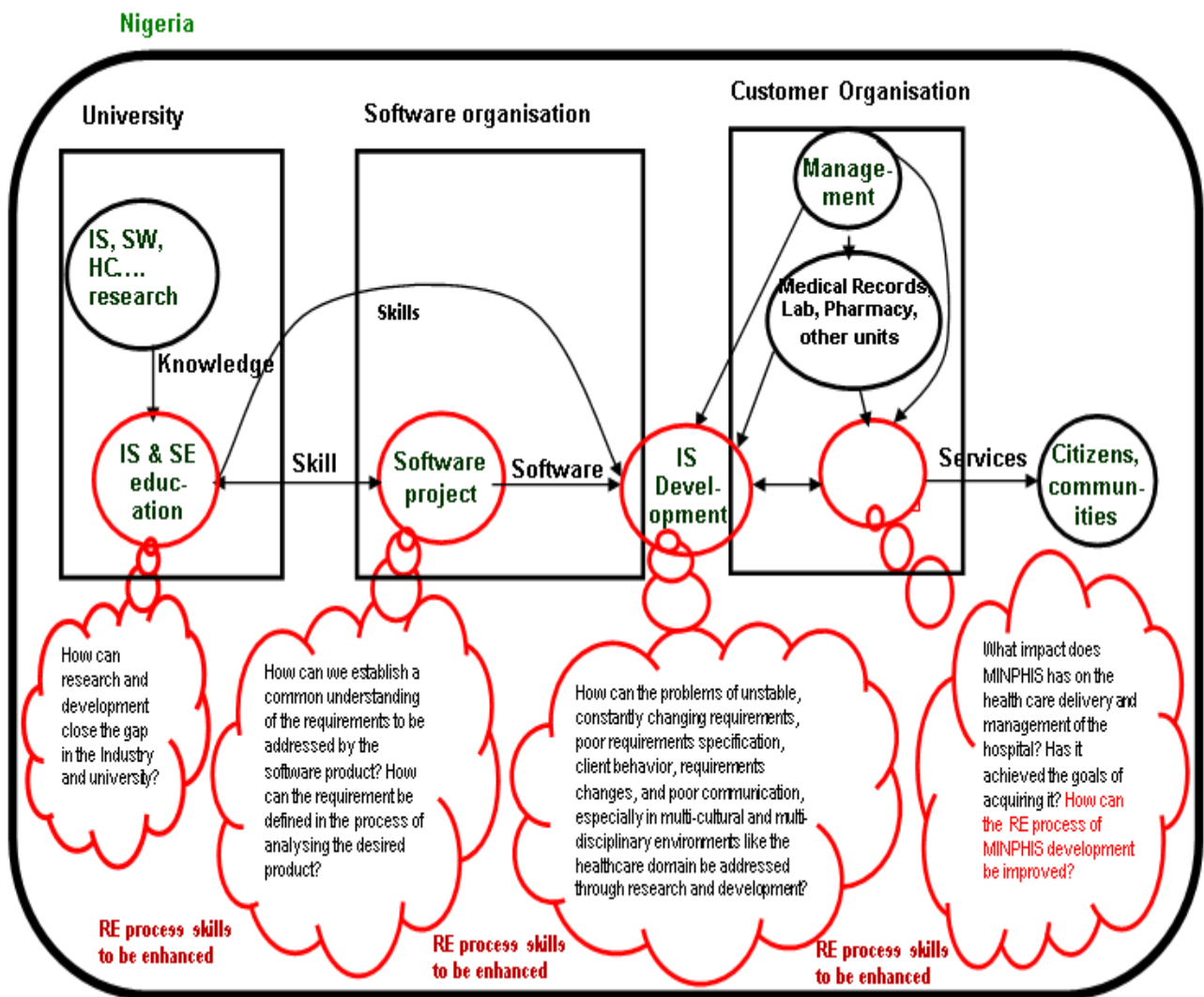
Fig 3: Conceptual framework showing the need for RE process skills enhancement (Modified from: [50])

7. CONCLUSION

Conclusively, the ability to identify problems and suggestions for improvements in the RE process opens up significant potential for increasing the success of information systems development and software projects. In order to improve RE processes, the current practices need to be examined within the scope of a defined ISD like the case with healthcare information system during development. Therefore, educating developers, managers and users about RE practices will enhanced skills in RE process for SE and ISD. In addition, understanding the different kinds of requirements and classifying users input into the appropriate categories during RE process is key to excellent software requirements for a software project. In particular, the ultimate goal of building a robust service-oriented system like the health information system, is to solve the problems of quality improvement. Specifically, understanding and analysing requirements of (healthcare) business processes will results in a requirements specification, which can be used for communication among stakeholders and may be part of a formal contract. It is therefore expedient to seek ways on improving the RE process in ISD. In this regard, a framework to enhance the ability to develop software products for the healthcare domain with the aim of meeting the needs of all system stakeholders and to ensure that the development process is going on the right tract is very important.

8. ACKNOWLEDGMENT

9. REFERENCES

[1] Cheng, B. H., & Atlee, J. M. (2009). Current and future research directions in requirements engineering. In Design Requirements Engineering: A Ten-Year Perspective (pp. 11-43). Springer Berlin Heidelberg.

[2] Luukkonen, I., Toivanen, M., Mursu, A., Saranto, K. and Korpela, M. (2013). Researching Activity-Driven

Approach for Information Systems Development. In Cruz-Cunha MM, Miranda IM, and Gonçalves P. (eds.) Handbook of Research on ICTs and Management Systems for Improving Efficiency in Healthcare and Social Care. IGI Global.

[3] Kavakli, E. (1999). Goal-driven requirements engineering: modeling and guidance (Doctoral dissertation, the University of Manchester).

[4] Zave, P. (1997). Classification of research efforts in requirements engineering. ACM Computing Surveys (CSUR), 29(4), 315-321.

[5] Rolland, C. (2006). From conceptual modeling to requirements engineering. In Conceptual Modeling-ER 2006 (pp. 5-11). Springer Berlin Heidelberg.

[6] Maalej, W., & Thurimella, A. K. (2009, September). Towards a research agenda for recommendation systems in requirements engineering. In Managing Requirements Knowledge (MARK), 2009 Second International Workshop on (pp. 32-39). IEEE.

[7] Castro-Herrera, C., & Cleland-Huang, J. (2010, May). Utilising recommender systems to support software requirements elicitation. In Proceedings of the 2nd International Workshop on Recommendation Systems for Software Engineering (pp. 6-10). ACM.

[8] Broy, M. (2006). Requirements engineering as a key to holistic software quality. In Computer and Information Sciences–ISCIS 2006 (pp. 24-34). Springer Berlin Heidelberg.

[9] Nuseibeh, B., & Easterbrook, S. (2000, May). Requirements engineering: a roadmap. In Proceedings of the Conference on the Future of Software Engineering (pp. 35-46). ACM.

[10] Korpela M, Soriyan HA, Olufokunbi KC. (2000) Activity analysis as a method for information systems development: general introduction and experiments from Nigeria and Finland. Scand J Inf Syst 12(1):191-210.

[11] Luukkonen, I. (2012): Activity-Driven Needs Analysis and Modeling in Information Systems Development. Publications of the University of Eastern Finland, Dissertations in Forestry and Natural Sciences, No 74. Academic Dissertation.

[12] Sommerville I., Sawyer P. and Viller S. (1997) Requirements Process Improvement through the Phased Introduction of Good Practice, Software Process-Improvement and Practice, 3, 19-34.

[13] Robertson, S. and Robertson, J. (1999). Mastering the Requirements Process. Addison-Wesley, Harlow.

[14] Bray, I.K. (2002). An Introduction to Requirements Engineering, Addison-Wessley.

[15] Yourdon, E. (1989). Modern Structured Analysis, Prentice-Hall, New Jersey, 1989.

[16] Rumbaugh, J., Jacobson, I. and Booch, G. (1999). The Unified Modelling Language Reference Manual, Addison-Wesley.

[17] Van Lamsweerde, A. (2000, June). Requirements engineering in the year 00: A research perspective. In Proceedings of the 22nd international conference on Software engineering (pp. 5-19). ACM.

[18] Dhirendra, P. (2013). Requirement Engineering Research, International Journal of Computer Science & Engineering Technology (IJCSET), Vol. 4 No. 04, pp. 447-450, ISSN: 2229-3345.

[19] Watson, A. (2008, September). Reflections on Requirements Engineering. In EDOC, pp. xxxiii – xxxiii, ISBN: 978-0-7695-3373-5, IEEE.

[20] Yen, J., & Tiao, W. A. (1997, January). A systematic tradeoff analysis for conflicting imprecise requirements. In Requirements Engineering, 1997, Proceedings of the Third IEEE International Symposium on (pp. 87-96). IEEE.

[21] Ijadunola KT, Bello MO, Korpela M, and Koskivirta M. (1998). A strategy for the use of information systems to support community-oriented primary healthcare in a deprived economy: A case study of Ife Central LGA, Nigeria. In: Richards B, ed. Current Perspectives in Healthcare Computing 1998, Conference Proceedings. Weybridge, BJHC Books, pp 3-9.

[22] Hannola, L. (2009). Challenges and means for the front end activities of software development. Academic dissertation, Lappeenranta University of Technology, Digipaino.

[23] Dorn, J., Grun, C., Werthner, H. and Zapletal, M. (2007). A Survey of B2B Methodologies and Technologies: From Business Models towards Deployment Artifacts. In Proceedings of the 40th Annual Hawaii International Conference on System Sciences (HICSS'07). p. 143. IEEE Computer Society Washington, DC, USA.

[24] Soriyan H.A.,Afolabi A.O.,Fatusi O., Mursu A.,Korpela M.J. Akinde A.D(2005). Development of a computer based Primary health care Information System: Towards a holistic system. In: Abiodun O. Bada and Adekunle Okunoye, eds. Enhancing Human Resource Development Through ICT: In Research-In-Progress Papers[CD-ROM]. Procceedings of the 8th International Working Conference of IFIP WG 9.4 26-28 May, 2005, Abuja, Nigeria

[25] Jarke, M., Loucopoulos, P., Lyytinen, K., Mylopoulos, J., & Robinson, W. (2010, January). The brave new world of design requirements: four key principles. In Advanced Information Systems Engineering (pp. 470-482). Springer Berlin Heidelberg.

[26] Ernst, N. A. (2012). Software Evolution: a Requirements Engineering Approach (Doctoral dissertation, University of Toronto).

[27] Zave, P., & Jackson, M. (1997). Four dark corners of requirements engineering. ACM Transactions on Software Engineering and Methodology (TOSEM), 6(1), 1-30.

[28] Gambo, I., Soriyan, A., & Achimugu, P. (2012). Software Performance Quality Evaluation of MINPHIS Architecture using ATAM. International Journal of Computer Applications, 46.

[29] Groen, P.; Soriyan A., and Nyamai-Kisia, C. (2005) "MINPHIS: Nigeria's experience using VistA, MUSTI, and other technologies in building their own national health information system", in the Virtual Medical World, a monthly news service for virtual medical community,http://www.hoise.com/vmw/05/articles/vmw/ LV-VM-10-05-19.html (Accessed 05/01/2006)

[30] Damian, D. E. H. (2003) "Challenges in Requirements Engineering," Requirements E, Springer, Springer, vol. 8, no. 3, pp. 149-160.

[31] Berry, D. M. and Lawrence, B. (1998): Requirements Engineering, IEEE Software, Vol. 15, No. 2, pp.26-29.

[32] Leite, J. C. (1987): A Survey on Requirements Analysis, Advanced Software Engineering Project Technical Report RTP-071, University of California at Irvine, Department of Information and Computer Science.

[33] Damian, D., Izquierdo, L. Singer, J. and Kwan, I. (2007). Awareness in the wild: Why communication breakdowns occur. In Second IEEE International Conference on Global Software Engineering, 2007. ICGSE 2007, pages 81-90.

[34] Kwan, I. and Damian, D. (2011). The hidden experts in software-engineering communication (NIER track). In Proceedings of the 33rd International Conference on Software Engineering, ICSE '11, pages 800-803, New York, NY, USA, ACM.

[35] McAllister, C. A. (2006). Requirements determination of information systems: User and developer perceptions of factors contributing to misunderstandings. ProQuest.

[36] Firesmith, D. (2007, May). Engineering Safety and Security Related Requirements for Software Intensive Systems. In ICSE Companion (p. 169).

[37] Sawyer, P., Gervasi, V., & Nuseibeh, B. (2011, August). Unknown knowns: Tacit knowledge in requirements engineering. In Requirements Engineering Conference (RE), 2011 19th IEEE International (pp. 329-329). IEEE.

[38] El Emam, K., & Madhavji, N. H. (1995, March). A field study of requirements engineering practices in information systems development. In Requirements Engineering, 1995., Proceedings of the Second IEEE International Symposium on (pp. 68-80). IEEE.

[39] Sommerville I. (1996) Software Engineering Fifth Edition, Addison-Wesley.

[40] Schrodl, H., and Wind, S. (2011). Requirements Engineering for Cloud Computing, Journal of Communications and Computers, Vol. 8, pp. 707-715.

[41] Standish Group, CHAOS Report, available online http://www.standishgroup.com, 2010.

[42] Rouibah, K., and Al-Rafee, S. (2009). Requirement engineering elicitation methods: A Kuwaiti empirical study about familiarity, usage and perceived value, Information Management & Computer Security, Vol. 17, No. 3, pp. 192-217.

[43] Aurum, A., & Wohlin, C. (Eds.). (2005). Engineering and managing software requirements. Springer Verlag, Berlin.

[44] Beecham, S., Hall, T., & Rainer, A. (2003). Software process improvement problems in twelve software companies: An empirical analysis. Empirical software engineering, 8(1), 7-42.

[45] Leffingwell, D. (1997). Calculating the Return on Investment from More Effective Requirements Management, American Programmer, Vol. 10, No. 4, pp. 13–16.

[46] Grady, R. (1999). An Economic Release Decision Model: Insights into Software Project Management. In Proceedings of the Applications of Software Measurement Conference, Orange Park, FL, pp.227–239.

[47] Gilb, T. (2010). What's Wrong with Requirements Specification? An Analysis of the Fundamental Failings of Conventional Thinking about Software Requirements, and Some Suggestions for Getting it Right. JSEA, 3(9), 827-838.

[48] Sadraei, E., Aurum, A., Beydoun, G., and Peach, B. (2007). A field study of the requirements practice in Australian software industry, Requirements Engineering, Vol. 12, pp.145-162.

[49] Hall, T., Beecham, S., and Rainer, A. (2002). Requirements problems in twelve software companies: an empirical analysis, IEEE Proceedings Software 149, pp. 153-160.

[50] Soriyan H. A. (2004) A conceptual Framework for Information System Development Methodology for Educational and Industrial Sectors in Nigeria. PhD Thesis. Obafemi Awolowo University, Ile-Ife Nigeria.