International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Do Non-Functional Requirements Matter? A Comparison of Traditional and Agile Approaches

Naina Handa

Submitted: 14/09/2023 **Revised**: 27/10/2023 **Accepted**: 07/11/2023

Abstract: Requirement Engineering is an essential part of the development of applications.However, NFRs are not properly considered in the initial phases of SDLC. NFRs are often hidden in FRs. In the paper, we are conducted a survey about the importance of NFRs in Traditional and Agile development process. The 52 respondents have answered the questionnaire and we are able to conclude that it is very important to consider the NFR. We have found that the 75% respondents consider the NFRs are very important and more than 25 % considered it as important.

Keywords: *Agile, Requirement Engineering, Questionnaire*

Introduction

Requirement Design is an integral part Requirements are a description of the services to be rendered by a software application and the constraints it will work under [1]. RE is the method of determining the services provided by the program by the client, and the restrictions under which it is to be established and operated. RE is the identification, review, documentation and validation of the system specifications to be created. This sets out what to do but does not explain how to execute this. It forbids rework [2]. The later errors are found the more difficult it would be to correct them. Determining a consistent set of specifications before system design and implementation commences is very important [3].

The RE cycle can be divided into various phases such as Requirement Elicitation, Requirement Interpretation, Device Modelling, Requirement Design, Requirement Validation and Requirement Management. The Requirement Elicitation refers to gathering of requirements by interview, brainstorming, questionnaire and workshop with stakeholders to name a few [5]. In Requirement Analysis phase the graphical tool like Unified modelling language has used and analyse the requirements. System Modelling phase designs the blueprint of the system. In Requirement Specification phase SRS has designed .Requirement Validation activity checks the documented requirements and blueprint are consistent and meet the need of the stakeholder [6]. Only if the final draft passes the validation process, the Requirement Specification becomes official. The last activity requirement management deals with the change and extension of requirements even after it is put into use. In the paper we focused on Requirement Elicitation phase .Broadly the Requirements can be classify into Functional Requirements (FRs) and Non Functional Requirements (NFRs). FRs are requirements which specify what the system is supposed to do . FRs defines a particular action or device function when certain requirements are met, e.g. System should send an e- mail when a new client signs up. FRs are business rules, Transaction correction, External interface, Legal requirements to name a few [7]. FRs are the descriptions of services to be rendered by the system, how the system responds to a given input, and how the system will behave in a particular situation. It defines functionalities or facilities to the system. FRs are user required specifications. This provides detailed overview of system services. FRs are a high-level description of what the program will do.eg enables users to search for whole databases or a subset from them. FRs derives system application architectural design whereas NFRs derives system technological architecture. FRs identify specific behaviours where NFRs describe requirements that can be used to evaluate a system's activities rather than the actual behaviours [8]. In system design, the implementation plan for FRs is detailed. System design is a system of describing the design, components, interfaces and relevant information for a framework to meet specific requirements.

Non Functional Requirements

NFRs are standards that define parameters that can be used to measure a system's activities, not actual behaviour. In system architecture the design for implementing NFRs is comprehensive. These are typically important criteria in architecture. NFRs usually define qualitative qualities of the method. It encompasses performance, scalability, flexibility, quality, reliability,

Assistant Professor
DAV College Amritsar
naina.pisces17@yahoo.co.in

usability, interoperability and serviceability [9]. It is very important to take into account NFRs appropriately because they affect customer experience while interacting with the program. NFRs explain how a framework should behave and what restrictions on its features, e.g. changed data in databases should be updated within two seconds for all users who access it. It defines how a system should be, and defines how the system is working as well. NFRs are hard to test, and are usually subjectively evaluated[10]. During development, it is usually informal, sometimes inconsistent, difficult to implement and assess for the customer before delivery. NFRs are restrictions on the resources or functions that the program provides such as timing restrictions, implementation cycle and standards constraints etc. NFRs are tacit specifications for the product and required from it. These are intended features and not explicitly reported specifications are known as quality attributes NFRs. In the system architecture the proposals for enforcing the NFRs are comprehensive. System architecture is a mathematical model specifying the system's structure, behaviour, and vision. Thus NFRs are often referred to as architecture significant requirements [11].

Eminence of NFRs

Users and developers spend the majority of their efforts in the development of new applications in modelling the FRs. NFRs are treated ad second class requirements and snubbed until the last phases of software cycle. Sometimes these are covered, overlooked and often ignored or forgotten. NFRs are hard to model, to build and to test. Misuse of NFRs is a cause of failure in a project. The evidence can be seen in history that not considering the NFRs during the starting of traditional and an Agile software development has result in failure of software. Existing NFRs Classification method can be classified into NFRs Elicitation and NFRs Detection [12]. The elicitation of NFRs assists stakeholders in defining negotiation and modeming NFRs. The Detection method refers to retrieving NFRs manually or automatically from existing documents. Elicitation process is very time consuming and error prone when performed manually by Requirement Engineer. The elicitation of high quality requirement is a critical success factor [13].

Machine Learning

Machine Learning is the domain of data science that uses statistical methods to offer computer science the ability to understand from data without being specifically programmed [14].ML examines the analysis and design of the algorithm from which data can be learned and predicted [15] .

The machine learning algorithms have used the different metrics for evaluating the model like Recall, Precision, F-measure and Accuracy [16]

Accuracy=True Positive + True Negative/Total number of cases

Precision= True Positive / True Positive + False Positive

Recall= True Positive / True Positive + False Negative

F-Measure=2*Precision*Recall/Precision+ Recall

True positive implies the number of instances which are correctly categorized.

False positive is the number of wrongly categorized cases in categories.

True negative is the number of cases that are not correctly identified in category.

False negative is the number of cases which are wrongly not classified [17].

Now the question arises. Are NFRs viewed as essential in Agile and Traditional Software development projects? To address this research question we questioned software developers about their relative value in identifying NFRs .The Google doc has used for designing the online questionnaire. Software developers have given the different perspectives for NFRs .

Survey Design

The study's goal is to define the significance of NFRs in software development programs. To accomplish this aim, we conceived a survey .To collects the thoughts of a huge variety of developers, we sent the online questionnaire to various software developers and questioned about the perceived prominence of the practice: "How relevant the usability for traditional and agile projects to identify NFRs.

a) Survey instrument

In order to collect the views of a wide variety of people, we have sent online questionnaire to software developers and asked about the importance of usability in traditional and agile software development. The respondents can choose the very important, important, less important and not important .The respondents can also go for the other option and able to describe the answer from their perspective or also able to skip the question .The questionnaire which was designed with the help of google docs have distributed over Whatsup and Email .

b) Population

Our target demographic can be described as the software developers in any kind of software development. We have not limited to our attention to any particular domain

or application. The aim of the study is to have general understanding of importance of usability issues of NFRs in software development. We have tried to cover a big range of people. For that we have approached different software development companies and developers. The research has relied on convenience sampling technique.

Related Work

The various research works have done in the field of NFRs. The different fields like Cloud computing, IOT, Big data , Machine Learning to name a few have considered the NFRs [18].

Martino et al. also discussed that the Criteria specification is a very demanding cloud computing task. An automated technique for extracting and classifying which is in Natural Language Type for designing cloud-based applications has been suggested in the paper [19].

Tóth and Vidács provided a comparative analysis of Machine Learning and NLP techniques as regards their use and success in the information engineering field [20]. Bhowmik and Do analyzed 50 open source NFRs i.e. Firefox, Mylyn and Lucene [21] Marinho et al. have described the significance of taking NFRs into consideration during early phases of system development and Requirement Elicitation become the most important[22].

Iqbal et al. showed the use of Machine Learning (ML) to simplify the different activities in the area of Information Engineering. The researchers focused particularly on Requirement Engineering [23].

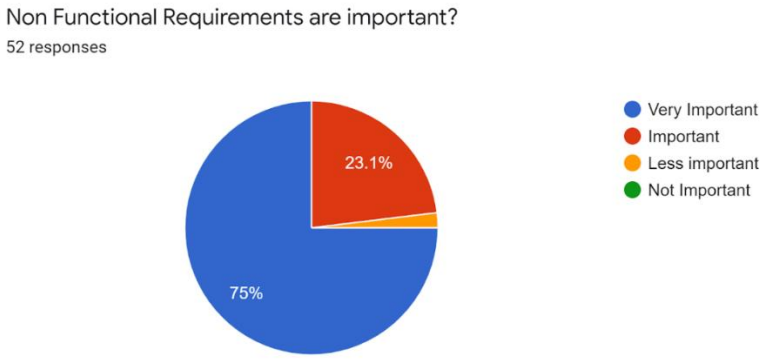
Sachdeva and Chung addressed the need for early device life cycle treatment of NFRs [24]. This shows that NFS stability and efficiency are becoming very critical in cloud-and big data initiatives, where lack of consideration can also seriously harm many important NFRs.

Title, Author, Publication, Year	Description
Title:" Study of the Performance of Various Classifiers in Labeling Non-Functional Requirements"[25] Authors" L Tóth, L Vidács " Publication:" Journal of Information Technology and Control (2019)"	Authors have demonstrated the usability of various machine learning techniques like Multinomial Naive Bayes, Support Vector Machine, Decision Tree and Logistic Regression on small-sized Tera Promise NFR dataset and on the large dataset extracted from Stack Overflow for experiment purpose. The results are compared using the Precision and Recall.
Title:" Semantic Clustering of Functional Requirements Using Agglomerative Hierarchical Clustering"[26] Authors" H Eyal Salman, M Hammad, AD Seriai, A Al-Sbou " Publication:" Information(2018)"	In this paper, a new dynamic clustering framework has designed. The functional requirements are clustered on the basis of textual similarity. This framework focused on an agglomerative clustering approach. The empirical study has done on open source products for showing the performance of algorithm.
Title:" NFRfinder: a knowledge based strategy for mining non-functional requirements "[27] Authors"RLQ Portugal, T Li, L Silva, E Almentero" Publication."International Conference Procedding Series(ICPS) (2018)"	Authors have focused on the challenge of find quality requirements in elicitation phase. In the paper, a novel semi-automatic approach named NFRFinder has designed for mining the NFRs from unstructured text. The result of the approach has measured in Precision and Recall.
Title:" Automatic Multi-class Non-Functional Software Requirements Classification Using Neural Networks "[28] Authors"C Baker, L Deng, S Chakraborty " Publication:" Computer Software and Applications Conference (COMPSAC), IEEE Annual International (2019)"	An Neural Network approach (Convolutional and Artificial Neural Network) has explored in the paper for extracting the NFRs and also classified further into categories like Usability, Performance, Security, Maintainability and Operability. The results of the experiment has measure in F-score and Recall and showed that the CNN effectively classify the NFRs

Title:" An empirical study on catalog of non-functional requirement templates Usefulness and maintenance issues "[29] Authors" S Kopczyńska, J Nawrocki, M Ochodek " Publication:" Information and Software Technology(2019)"	A template has designed for eliciting the NFRs. The NFR Catalog has explored in the paper and used to elicit the NFRs from inexperienced users. In the study 41 industry projects have used and find the 2231 NFRs .In the result it has given that the NFR templates are quite useful and also enhance the elicitation of NFRs.

Result and Discussions

There are approximately 54 respondents have answer the questionnaire. The coverage of participant-developed fields and varieties of software applications was also strong.



This includes an overview of the comments regarding the relative value of implementing NFRs. The pie chart has depicted that the 75% of respondents considered that it is very important to be considered the NFR in the initial phases of software development. More than 25 % respondents have considered the NFRs as important and very less number of respondents have considered it as less important and no one taken it as not important at all.

Conclusions

Our survey aimed to investigate the significance of NFRs in Agile and Traditional software development. The different researchers have used the different approach for eliciting the NFRs. We proposed the following observations based on the responses of 52 participants who are dealing in software development. The online questionnaire has designed for collecting the survey. The results showed that the NFRs are very important to be considered during initial phases of software development. The 75 % respondents considered the NFRs are very important and 25% considered it as important .So it has concluded from the survey that NFRS have significance in software development process.

References

[1] Khan, F., Jan, S. R., Tahir, M., Khan, S., & Ullah, F. (2016). Survey: dealing non-functional requirements at architecture level. *VFAST Transactions on Software Engineering*, *9*(2), 7-13.

[2] Handa, N., & Sharma, A. (2019). An Inclusive Study of Several Machine Learning Techniques. *Journal of the Gujarat Research Society*, *21*(5), 221-225.

[3] Kaur, G., & Singh, B. (2017, June). Improving the quality of software by refactoring. In *2017 International Conference on Intelligent Computing and Control Systems (ICICCS)* (pp. 185-191). IEEE.

[4] Arora, M., Verma, S., & Chopra, S. (2020). A Systematic Literature Review of Machine Learning Estimation Approaches in Scrum Projects. In *Cognitive Informatics and Soft Computing* (pp. 573-586). Springer, Singapore.

[5] Thakur, S., & Kaur, A. (2013). Role of agile methodology in software development. *International Journal of Computer Science and Mobile Computing*, *2*(10), 86-90.

[6] Boehm, B., & In, H. (1996). Identifying quality-requirement conflicts. *IEEE software*, *13*(2), 25-35.

[7] Breitman, K. K., Leite, J. C. S., & Finkelstein, A. (1999). The world sa stage: a survey on requirements engineering using a real-life case study. *Journal of the Brazilian Computer Society*, *6*(1), 13-37.

[8] Lindstrom, D. R. (1993). Five ways to destroy a development project (software development). *IEEE Software*, *10*(5), 55-58.

[9] Nuseibeh, B. (1997). Ariane 5: who dunnit?. *IEEE Software*, (3), 15-16.

[10] Binkhonain, M., & Zhao, L. (2019). A review of machine learning algorithms for identification and classification of non-functional requirements. *Expert Systems with Applications: X*, 100001.

[11] Dhir, S., Kumar, D., & Singh, V. B. (2019). Success and failure factors that impact on project implementation using agile software development methodology. In *Software Engineering* (pp. 647-654). Springer, Singapore.

[12] Ali, S., Bashir, T., & Yousaf, I. (2020). The Quality Requirements Analysis with Machine Learning. *International Journal of Computational and Biological Sciences*, *1*(1), 1-1.

[13] Domah, D. (2013). *The NERV Methodology: Non-Functional Requirements Elicitation, Reasoning and Validation in Agile Processes* (Doctoral dissertation, Nova Southeastern University).

[14] Bhogal, G. S., & Rawat, A. K. (2020). Analysis on Smoke Detection Techniques. In *Smart Intelligent Computing and Applications* (pp. 163-180). Springer, Singapore.

[15] Swain, M., Singh, R., Thakur, A. K., & Gehlot, A. A Machine Learning Approach of Data Mining in Agriculture 4.0.

[16] Godara, J., & Aron, R. (2020). *An Approach to Detect Sarcasm in Tweets* (No. 3035). EasyChair.

[17] Singh, M., & Salaria, D. S. (2013). Software defect prediction tool based on neural network. *International Journal of Computer Applications*, *70*(22).

[18] Handa, N., Sharma, A. & Gupta, A (2019). Non Functional Requirements Analysis using Data Analytics . International Journal of Advanced Science and Technology Vol. 27, No. 1, (2019), pp. 383 - 392

[19] Di Martino, B., Pascarella, J., Nacchia, S., Maisto, S. A., Iannucci, P., & Cerri, F. (2018, May). Cloud Services Categories Identification from Requirements Specifications. In *2018 32nd International Conference on Advanced Information Networking and Applications Workshops (WAINA)* (pp. 436-441). IEEE.

[20] Tóth, L., & Vidács, L. (2018, May). Study of various classifiers for identification and classification of non-functional requirements. In *International Conference on Computational Science and Its Applications* (pp. 492-503). Springer, Cham.

[21] Bhowmik, T., & Do, A. Q. (2019). Refinement and resolution of just-in-time requirements in open source software and a closer look into non-functional requirements. *Journal of Industrial Information Integration*, *14*, 24-33.

[22] Marinho, M., Arruda, D., Wanderley, F., & Lins, A. (2018, September). A Systematic Approach of Dataset Definition for a Supervised Machine Learning Using NFR Framework. In *2018 11th International Conference on the Quality of Information and Communications Technology (QUATIC)* (pp. 110-118). IEEE.

[23] Iqbal, T., Elahidoost, P., & Lúcio, L. (2018, December). A Bird's Eye View on Requirements Engineering and Machine Learning. In *2018 25th Asia-Pacific Software Engineering Conference (APSEC)* (pp. 11-20). IEEE.

[24] Sachdeva, V., & Chung, L. (2017, January). Handling non-functional requirements for big data and IOT projects in Scrum. In *2017 7th International Conference on Cloud Computing, Data Science & Engineering-Confluence* (pp. 216-221). IEEE.

[25] Tóth, L., & Vidács, L. (2019). Comparative Study of The Performance of Various n Labeling Non-Functional Requirements. *Information Technology and Control*, *48*(3), 432-445.

[26] Eyal Salman, H., Hammad, M., Seriai, A. D., & Al-Sbou, A. (2018). Semantic Clustering of Functional Requirements Using Agglomerative Hierarchical Clustering. *Information*, *9*(9), 222.

[27] Portugal, R. L., Li, T., Silva, L., Almentero, E., & do Prado Leite, J. C. S. (2018, September). NFRfinder: a knowledge based strategy for mining non-functional requirements. In *Proceedings of the XXXII Brazilian Symposium on Software Engineering* (pp. 102-111).

[28] Baker, C., Deng, L., Chakraborty, S., & Dehlinger, J. (2019, July). Automatic Multi-class Non-Functional Software Requirements Classification Using Neural Networks. In *2019 IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC)* (Vol. 2, pp. 610-615). IEEE.

[29] Kopczyńska, S., Nawrocki, J., & Ochodek, M. (2018). An empirical study on catalog of non-functional requirement templates: Usefulness and maintenance issues. *Information and Software Technology*, *103*, 75-91.