International Journal of Computer Applications Technology and Research
Volume 4– Issue 4, 220 - 225, 2015, ISSN:- 2319–8656

A Case Study upon Non-functional Requirements of Online Banking System

Harmeet Kaur
Punjab Technical University
Jalandhar, Punjab, India

Shahanawaj Ahamad
DCScSWE, College of
Computer Sc. & Engineering
University of Ha'il, K.S.A.

Gurvinder N. Verma
Shri Sukhmani Institute of
Engineering & Technology
Derabassi, Punjab, India

Abstract: The proper working of online banking is essential for development and advancement over the world and influencing organizations, society and individuals. Online banking, a vital activity within the financial markets is experiencing real change fuelled by advances in information systems techniques. Non - functional requirements (NFRs) are important software requirements that have been evaluated and specified right from the beginning of software development cycle while specifying the functional software requirements (FRs). Nfrs could be thought of as additional requirements that must be fulfilled by the functional requirements. The dissatisfaction of Nfrs is one of the indispensable purposes behind the failure of software projects. Non- functional requirements for instance accuracy, usability, performance and security are often crucial to online banking system. Accordingly the nonfunctional requirements or Nfrs should be given attention as early as possible in a software lifecycle and must be shown in the software design before conferring to the detailed design. The inspiration driving this paper is to examine how the assessment of Nfrs of online banking system serves to systematically make choice among different alternatives.

Keywords – Reengineering, Legacy, Non-functional Requirement, Requirement engineering, Usability.

1. INTRODUCTION

The success of a software system can be accessed on the basis of its fulfillment of two interdependent types of requirements. First and foremost, the client focused requirements that focus on the functionalities given to the users of the system. Second, the context focused requirements which can be system, procedure and human necessities. The later type of requirements is suggested as non-functional requirements (Nfrs). Nfrs are systems goals that may impact the operational environment and design decisions an employer may look for the progression of the software. Besides other things Nfrs incorporate operational environment: hardware and software interfaces, accuracy, performance etc. Functional requirements portray the noticeable external input and output interactions with the system under scrutiny, despite the fact that non-functional requirements are those that drive unique conditions and qualities on the system to develop. Thus, system acceptance testing is centered on both functional and non-functional system's requirements.

The NFR has various definitions in the literature and the industry. [19] Characterizes it as "Umbrella term to cover all those requirements which are not explicitly defined as functional". Nfrs can be grouped either using intra model dependency view or intermodal dependency view. In the intra model dependency view, Nfrs are refined into a hierarchy comprising of a root NFR classification and various child refinements, for example, decomposition and operationalization [19].The disintegration is a unique procedure where a NFR is portrayed using child sub-Nfrs [19]. For example, the security NFR can be disintegrated into smaller sub-Nfrs so that it can be addressed in a better way. The operationalization is an extraordinary methodology where a NFR is refined into operations, functions, information representations and architecture design decisions that are important to address the NFR effectively.

2. NON FUNCTIONAL REQUIREMENTS

In competing situations time-to-market, robustness, and quality are some of the components which are critical for measuring the success of system. As the business requirements are changing rapidly there is a need to change and develop. In the system development process requirement specification is the important and significant step to be considered. Subsequently, there is a strong need to formulate the methodologies and standards by which one can capture the requirements. In addition to providing functional requirements or services the software system should posses the nonfunctional requirements as they are important for considering the quality of the software and amazingly impact the design and execution decisions a developer may make. These also have an impact on the acceptability of the software by the intended clients or users.

3. THE NATURE OF NFRS

There are different clarifications why non- functionality is complicated to handle. To appreciate why, it has to research what a NFR is and the distinctions that exist contrasted with functional requirements. A NFR may be seen as how the system act as per specific attributes, for instance performance, modifiability and security. A functional requirement also is seen as what the system must be capable of performing. Functional requirement are particular entities that are not difficult to understand

and administer. In fact they are easy to verify because of their temperament, which infers that it is not difficult to give a boolean answer whether the requirement is satisfied i.e. "Is it conceivable to upload a document?" Concerning NFR it is not that irrelevant to ask essential inquiries and expect a boolean answer that shows if the requirement is fulfilled or not, this is a due to nature of these requirements. NFR could be seen as subjective, since different people decipher and evaluate the Nfrs contrastingly. This is exasperates even increasingly because of how small and uncertain NFR for the most part are depicted in software requirement specification.

From time to time in distinctive organizations the systems are built according to the need, there for the explanation and criticalness of the various Nfrs changes from system to system. Which is the motivation behind why Nfrs can be seen as relative? Not precisely how Nfrs are translated or how important they are, could be seen as relative but how the requirements are attained. May be existing techniques could be created or upgraded to acknowledge particular Nfrs, there for the methodology "one solution fits all" is not suitable. Moreover Nfrs can be communicating, which implies that when understanding one requirement this could help solving or overturn an alternate one. In the light of worldwide effect of the aforementioned requirements local solutions for the NFR is often not sufficient. For example, performance centered security. Assume that the system will be made secure at the cost of performance.

4. WHY NON-FUNCTIONAL REQUIREMENTS

Functional requirements exhibit the behavior of the system that fulfills client's objectives or tasks whereas non- functional requirements consolidate constraints and qualities. Qualities are properties of the system that are examined by its and in this manner will affect their level of satisfaction with the system. Constraints are not subject to negotiation and, not at all like qualities, are (theoretically at any rate) off-limits all through design trade-offs. Contextual constraints are characteristics of the "super-system " or the advancement affiliation that oblige the change somehow. Examples join the target working system or fittings stage by virtue of the customer environment, or the inclination set of open visionaries by virtue of the improvement association. Nfrs are in general called as quality prerequisites [4] [2] the business asks for the more non-viable perspectives to be satisfied in information schemas other than its usefulness. Later works proposed by [12] [21] [7] have demonstrated that complex reasonable models must administer nonfunctional requirements. Mistakes in light of oversight of Nfrs or to not suitably administering them are around the most costly and generally hard to reexamine [21] [13] [9]. Non-functional requirements are always related to a viable need [7].

5. ROLE OF NON-FUNCTIONAL REQUIREMENTS IN A SYSTEM

There is much verbal confrontation about the part non-functional requirements play in the requirement elicitation and system advancement process. Systems and software designers succumb to two camps concerning this inquiry, with the line of division being the "qualitative" nature of non functional requirements A Qualitative requirement is equal to an enormous need, which is something that must be avoided at any cost. Recently, [4],[14],[16] HAVE SHOWN THAT there has been a push to make non- functional requirements more quantitative in nature. For example, one requirement may be: "The system might allow anonymous login for unlimited functions." A second need expresses: "The system may have defensive firewalls to affirm fluctuating levels of access." Implicitly, the stakeholder is getting two non-functional requirements: Ease of use and security. By and large, when seen uninhibitedly, non- functional requirements can be addressed routinely. On the other hand, there are times, (for example the one above) when they show an accident which begins ripple affect felt by distinctive requirements– trying to fulfill one need. The developer is sacrificing another requirement.

6. NON-FUNCTIONAL REQUIREMENTS IGNORED

Because of less consideration paid to a various non-functional requirements: interoperability and testability NASA's Mars Climate Orbiter and Polar Lander were lost in Mars environment late in 1999.As far as the interoperability is concerned, the orbiter was delivered by different organization: Lockheed Martin Aeronautics (main contractor) and the Jet Propulsion Laboratory (JPL). In transforming push for the unit, Lockheed Martin researchers utilized English estimations, while JPL experts utilized the metric system. This oversight was recently perceived after the test was conveyed, a definite explanation behind the disappointment. Moreover, the modules that were made for the Polar Lander were not as brief as first considered. All through system testing, one module was denied (there was a known issue with that module) and tests were run.

At a later time, the banned module was tried autonomously and final conveyance was made. It was not until the system failed in space that scientists recognized that the exclusion of the module all through system testing (they simply ran one full test), influenced the usefulness of the landing gear. These two missteps cost United States citizens $319m.Another example which alludes to the absence of attention to non-functional requirements can impact framework unfavorably. Among distinctive clarifications and as various non functional requirements were ignored all through the system advancement stage the London ambulance system was deactivated after its deployment. Some of the non functional requirements which were disregarded during the deployment of the ambulance are reliability (vehicles location), cost (stress on the best value), usability (poor control of data on the screen), performance (the system did what should do however the performance was unsatisfactory), [15] [3]

7. THEORATICAL NFR FRAMEWORK

The NFR Framework offers a systematic methodology for characterizing NFRs. It offers extraordinary capacity to see for all significant NFRs and their interdependencies. In addition, it likewise helps software designers to fathom the vital activities for guaranteeing quality. It even captures and documents design decisions and rationales in addition to providing traceability for derived specifications and requirements

The NFR Framework offers a precise approach for describing Nfrs. It offers remarkable ability to see for all noteworthy Nfrs and their interdependencies. Likewise, it similarly helps software designers to understand the essential activities for ensuring quality. It even catches and reports design decisions and methods of reasoning in addition to providing traceability for derived specifications and requirements.

7.1 Security

Security is the feature of the system which ensures that system must be protected from the unintentional or malignant harm; unauthorized access to the data is not permissible. For the safety purpose the data must be backed up after certain period of time say 24 hours and the backed up data must be stored in a secure location. In online banking system the application must be able to send or receive the information to or from the server and client in an encrypted way. Security must stick to some standard and plans.

The security is significant subject of online banking as client is more worried about the security of the account, personal data and transactions. The information kept in the system ought to be precise and complete.

7.2 Performance

The term performance alludes to the capacity of the system or software to process as many as transactions per second as submitted to it without failure. Despite the fact that the system is functional and reliable if it fails to make efficient use of resources such as CPU cycles, disk space etc its performance is not good that is the performance of the system is not up to the mark. Performance measure how well the system can perform and whether the software will have the capacity to reach its response time targets. Performance additionally measures that how effectively the software will have the capacity to scale with countless activities for every second, moment, or hour. The online banking system is a multi- user system, which implies distinctive clients can access the system simultaneously and the system will work accurately and proficiently. So the client is more worried about the performance of the online banking.

The term performance suggests the ability of the system or software to process the same number of transactions every second as submitted to it without failure. Performance is measured in term of how well the system makes optimized or maximum use of the resources without failure. It measure how well the system can perform and whether the software will have the ability to achieve its response time targets. Performance moreover measures that how successfully the software will have the ability to scale with numerous activities submitted to it in every second, minute, or hour.

7.3 Usability

As online banking is carried by various types of clients i.e. whether they have knowledge of computers or not so the application designed for online baking must be easy to use and enable the client to manage their accounts or transactions with simplicity. The application must have graphical user interface and it must have the ability to provide informative error messages. The qualities of the ease of use which can be measured are learning time it points out the time required to learn the application, number of errors while working with the normal speed furthermore the likeness of the client to measure the system i.e. the client fulfillment in utilizing the framework. The interfaces of the system ought to be clear, easy and simple to use and understand. To expand the ease of use, on-line help and customer care executive ought to be incorporated into the system to encourage online banking.

7.4 Availability

The online banking should be available round the clock. It means for how long the system is available for its users or clients and for how long the system will be operational. As far as the online banking system is concerned availability of the system is of uttermost importance as the business is round the clients and clients should be able to avail the benefits of online baking without any constraints and round the availability makes it happen and thus it is of importance for the bank and the clients as well. The online banking system must have the availability of 99% if not hundred than the availability of the system must be nearly 100%.

7.5 Confidentiality

Client should be able to access the online banking account after successful authentication. The data entered by the client is not accessible to other clients using online banking. As far as the confidentiality of the is concerned it means to maintain the secrecy as online banking is round the clock to access the account i.e. any where any time so it is important that the software of the online baking must provide the facility to maintain the secrecy of the clients and the clients should have their own passwords and user names and these must be automatically become inaccessible to the person who so ever wants to have the unauthorized access to the account.

7.6 Reliability

Reliability reflects the capacity of the software to maintain its performance over the time. It implies how well the system performs in peak hours. A robust system is one which has the capacity to handle the bugs without failure i.e. how effortlessly it handles the bugs because of data or handling, surprising conditions while working conditions furthermore the software imperfections, and if the system is robust it is reliable also. The application must self contained, consistent and complete with in itself. The failure rate in the online banking system should be least or negligible as the system is supposed to be reliable. Reliability of the system depends on the failure free transactions and how fast the system is able to recover from the failure.

7.7 Operability

It is concerned with the how well the software will work in distinctive environment. The online banking application must have the capacity to operate on any gadget i.e. hand held gadgets or desktops or laptops without failure the change in the environment ought not to hamper the operability of the software. The online banking system ought to be operational on any hand held gadgets and it ought to have the capacity to show the different types of currencies used in distinctive nations. While in operation the system ought to give the consistency of the operation. It is supposed to give ease of operation and controls by the client. The online banking system ought to be operational on any hand held gadgets and it ought to have the capacity to show the different types of currencies used in distinctive nations. While in operation the system ought to give the consistency of the operation. It is supposed to give ease of operation and controls by the client.

7.8 Traceability

Traceability refers to the capability for tracing the status of the transaction and account on account number basis. Traceability is an important aspect in the banking industry, where it makes tracking of transaction possible. It should contribute to the safety of the transaction. Online banking should enable the user to trace the state of his/her transaction at any time.

7.9 Recoverability

Recoverability implies the ability to restore your software to the point when failure occurred. The ability to recoup quickly from a system failure depends not simply on having backup of the data, also on having a predefined plan for recuperating that information. in case of online banking system it is obliged that the software ought to be tested to check it will have the capacity to recuperate from the failures or not to do so the software is subject to failures via doing deeds which prompts its failures e.g. restarting the machine when the online banking application is running, or it is getting information and so on. Software can get back from failures at the very instance when the failures occur. The online banking must have the capacity to recover itself from the failures which may happen due to internet problem or because of some other reason. The online banking ought to have the capacity to recover the influenced information after failure and must have the ability to restore the sought level of execution or performance.

7.10 Visibility

It alludes to condition of having the capacity to see online banking empowers the client to see the login screen and the configuration of the online banking application as per the client desire. As the customer is dealing with his or her transactions or accounts online the application must be composed in such a way, t that every single feature of the application is obvious to the client as the client logins. The different features of the application must be self contained so that even a novice can utilize it without hesitation. All the features of the application must be visible and easy to understand and use.

8. RESEARCH METHODOLOGY

8.1 Selecting Samples

The main objective of this study was concentrated on the role of non functional requirements to assess the quality for reengineering. The study was intended to find the importance of non functional requirements like security, performance, usability availability, operability etc in online banking system. Therefore, the sample for this study was selected from the users of online banking.

8.2 Data Collection Procedures

Data was assembled from the clients using online banking. A set of questionnaire was given to online banking users. The first part of the questionnaire comprises of the general data of the respondent. Non functional requirements were utilized in second part. The last part comprises of online banking user’s recommendations if any. While filling up the questionnaire, each part of the survey was clarified to the respondents.

8.3 Research Design

A sample of 122 online banking customers was drawn, however 65 respondents could manage to provide the requisite information which was further utilized to find out the perception of users how and which out of 12 non functional requirements are of importance for them while dealing with online banking.

9. RESULTS AND DISCUSSION

While measuring the information with the help of statistical tools, to be specific Total Weightage Score system (TWS), it was observed that 67% of target respondents were male and 33% were females, of which most of the respondents were lying in the age group of 20-30 yrs i.e. 41.54%, followed by age of 40-50 yrs i.e. 30.77% and 12.30% were lying between the age of 30-40 yrs while the rate of respondents in the age gathering of 50-60 yrs and 60-70 yrs was discovered to be less i.e. 9.24 and 6.15% respectively. The information gathered was analyzed and total weightage score was computed and from the observation of the analysis it was discovered that in the online banking system client is more worried about the security followed by performance, usability and availability of the online banking services. From the graph indicated in Fig.

reliability, visibility and confidentiality are closely related with total weightage score ranging from 252,240 and 233.The total score for operability and accuracy was discovered to be 198 and 165 respectively though the total weightage score for portability, recoverability and traceability varies between 135 and 126.

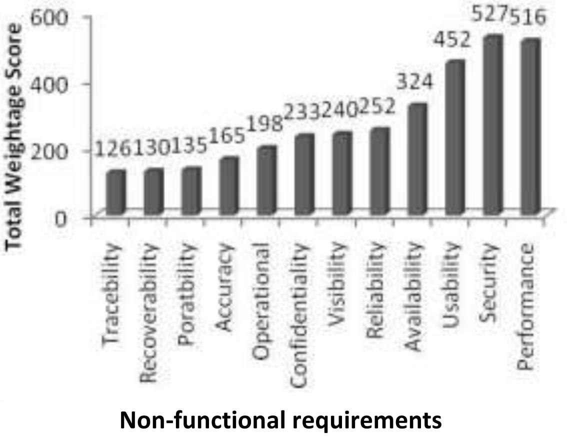

Figure 1: Weightage of NFRs in online banking system

During the study it was observed that security is of utmost importance amongst the other non functional requirements. Security is of concern for the user of the online banking system and also for the success of the online banking system as client or user is more worried about the security issues followed by performance which is affected by the speed and response of the connection over which the user is working. To have hassle free operations user is concerned about the usability and availability of the online banking system. For proper functioning the system must be reliable so another non functional requirement of user concern is reliability where as portability, recoverability, traceability, accuracy, usability and confidentiality makes the online banking system easy to use and user friendly

10. CONCLUSION AND FUTURE SCOPE

In this paper, non functional requirement for online banking system has been exhibited. The non-functional requirement corresponds to the quality of the system but in online banking system Nfrs are considered as requirements of paramount importance for the system. The security is one of the non functional requirements which is thought to be of absolute vitality and is considered as one of the primary prerequisites that determine the success or failure of the system. Performance and usability are also of importance as far as the proper working and running of the system is concerned. It has been proposed that while considering the quality requirements of the system the non functional requirements ought not to be ignored for proper working of the system. This work would help the software engineering community and to the researcher in enhancing the notion of Non functional requirements for online banking system and furthermore inspire them to come up with pre-specified benchmarks of Nfrs. The present work will be further extended as per the interest of the research problem and will be put to statistical investigation for increasing accuracy of the work and to improve the precision later on.

11. REFERENCES

[1] Basili, V. and Rombach, H., “Tailoring the software process to project goals and environments”, Proceedings of the 9th International Conference on Software Engineering (ICSE-9), IEEE Computer Society Press, pp. 345-357 March, 1987.

[2] Boehm, Barry e In, Hoh., “ Identifying Quality-Requirement Conflicts”, IEEE Software, pp. 25-35, March 1996.

[3] Breitman,K. K., Leite J.C.S.P, Finkelstein Anthony, “The World's Stage: A Survey on Requirements Engineering using a Real-Life Case Study”, Journal of the Brazilian Computer Society, No. 1, Vol. 6, pp.13-37, Jul. 1999.

[4] Chung, K. L., “Representing and Using Non-functional Requirements for Information System Development: A Process Oriented Approach”, Ph.D. Thesis, also Tech. Rpt. DKBS-TR-93-1, Department of Computer Science, University of Toronto, June 1993.

[5] Chung L, Nixon B, Yu E. Using non-functional requirements to systematically select among alternatives in architectural design In: Proceedings of the first international workshop on architecture for software systems, Seattle, WA, 1995.

[6] Chung L, Nixon B, Yu E. Dealing with change: an approach using non-functional requirements. Requirements Eng pp.238–260,1996.

[7] Chung, Lawrence et al., Non-Functional Requirements in Software Engineering Boston: Kluwer Academic Press 2000.

[8] Cysneiros, L.N., Leite, J.C.S.P, “Driving NFR to Use Cases and Scenarios”, XV Brazilian Symposium on Software Engineering, 2001.

[9] Cysneiros L.N., Leite J.C.S.P., “Integrating non-functional requirements into data model”, In: Proceedings of the 4th international symposium on requirements engineering – Ireland. IEEE Computer Society Press, Los Alamitos, pp. 162–171, 1999.

[10] Daniel, E., “Provision of electronic banking in the UK and the Republic of Ireland”, International Journal of Bank Marketing, 17(2), pp.72-82, 1999.

[11] D. Arnold, et al., "Scenario-Based validation: Beyond the user requirements notation", in Australian Software Engineering Conference (ASWEC), Los Alamitos, CA, USA, pp. 75- .84, 2010.

[12] Dardenne A., van Lamsweerde A, Fickas, S., “Goal Directed Requirements Acquisition”, Science of Computer Programming, Vol. 20, pp: 3-50, 1993.

[13] Ebert, Christof, “Dealing with Nonfunctional in Large Software Systems”, Annals of Software Engineering, 3, pp. 367-395,1997.

[14] Franch, X., “Systematic Formulation of Non-Functional Characteristics of Software”, Proc. 3rd Int. Conf. on Requirements Engineering, Colorado Springs, Colorado, USA, 1998.

[15] Finkelstein, A. and Dowell J., “A comedy of Errors: The London Ambulance Service Case Study”, Proceedings of the Eighth International Workshop on Software

Specification and Design, IEEE Computer Society Press pp. 2-5, 1996.

[16] Jacobs, Stephan, “Introducing Measurable Quality Requirements: A Case Study”, Fourth IEEE International Symposium on Requirements Engineering, 1999.

[17] J. H. Hill, et al., "Unit Testing Non-functional Concerns of Component-based Distributed Systems", presented at the Proceedings of the 2009 International Conference on Software Testing Verification and Validation, 2009.

[18] Jureta, I., S. Faulkner, P.-Y. Schobbens, “A More Expressive Softgoal Conceptualization for Quality Requirements Analysis”, Proc. of IEEE Int. Conf. on Conceptual Modelling (RE06), pp.281-295.

[19] Kassab , M , “ Formal and Quantitative Approach to Non Functional Requirements Modeling and Assessment in Software Engineering”, Doctoral dissertation Concordia University , Montreal, Canada,2009.

[20] K. Saleh and A. Al-Zarouni, “Capturing non-functional requirements using the user requirement notatio”, Proceedings of the International Research Conference on Innovations in Information Technology (IIT 2004), Dubai, pp. 222-230, 2004.

[21] Mylopoulos J, Chung L, Yu E, Nixon B., “Representing and using non-functional requirements: a process-oriented approach”, IEEE Trans Software Eng, 18(6), pp.483–497, 1992.

[22] Mylopoulos J, Chung L, Yu E., “From object-oriented to goal oriented requirements analysis”, Commun. ACM, 42(1), pp.31–37, 1999.

[23] Paech, B. & Kerkow, D., “Non-functional requirements engineering – quality is essential”, 10th International Workshop on Requirements Engineering – Foundation for Software Quality, pp. 237-250, 2004.

[24] Q. Tran, “A CASE Tool for the Non-Functional Requirements Framework”, M.S. Thesis, Dept. of Computer Science, Univ. of Texas Dallas, 1998.

[25] R. Sprague, “A framework for the development of decision support systems”, MIS Quarterly, Vol.4, No. 4, 1980.