
The impacts of non-functional requirements in web system projects

Norazlin Yusop*, Didar Zowghi and
David Lowe

University of Technology Sydney
P.O. Box 123, Broadway
New South Wales 2007, Australia
E-mail: Norazlin@it.uts.edu.au
E-mail: Didar@it.uts.edu.au
E-mail: David.Lowe@uts.edu.au
*Corresponding author

Abstract: In web system development, the Non-Functional Requirements (NFRs) are typically considered only briefly during the requirements elicitation stage and not rigorously articulated by either web developers or the client. This paper reports on an investigation into this issue involving interviews with web developers who were engaged in commercial web development projects. The results from this qualitative research highlight that web developers commonly do not pay sufficient attention to NFRs. This arises due to uncertainty, lack of time, lack of knowledge in the importance of NFRs and partly because NFRs are not readily available and documented from previous similar projects. Web developers also do not elicit NFR at the same time and at the same level of details as Functional Requirements (FRs). This study highlights that exploring the domain at an early stage of development will help developers to better understand NFR. A lack of rigor in articulating NFRs may significantly impact on the development effectiveness and the quality of the resulting web system. An evaluation of NFRs may also lead to discovering new FRs.

Keywords: web system projects; Non-Functional Requirements; NFRs; impacts; security; integration.

Reference to this paper should be made as follows: Yusop, N., Zowghi, D. and Lowe, D. (2008) 'The impacts of non-functional requirements in web system projects', *Int. J. Value Chain Management*, Vol. 2, No. 1, pp.18–32.

Biographical notes: Norazlin Yusop is a Research Assistant and PhD candidate in the Faculty of Information Technology at the University of Technology, Sydney (UTS). She has published in various conferences and in the *Journal of Web Engineering*. She has been a reviewer of papers for many conferences such as *International Conference on Web Engineering*, *International Conference on Requirements Engineering*, *Australian Workshop in Web Engineering*, *APSEC*, *CAiSE* and journals such as the *Journal of Research and Practice in Information Technology*. Yusop's research interests are in web development and business information systems, focusing on systems impacts, scoping, issue resolution processes, business process and knowledge transformation.

Dr. Didar Zowghi is the Associate Dean for Research in the Faculty of Information Technology at the University of Technology, Sydney (UTS). She received her PhD in Software Engineering from the Macquarie University in

2000. Her research in the last 12 years has focused on software development processes and their product quality. She is on the programme committee of many conferences including the *International Conference on Requirements Engineering*, *International Conference on Web Engineering*, *REFSQ*, *APSEC*, *IRMA* and *CAiSE*. Zowghi is the Regional Editor of the *International Requirements Engineering Journal (REJ)* and Associate Editor of the *Journal of Research and Practice in Information Technology (JRPIT)*.

Professor David Lowe is the Faculty of Engineering Associate Dean (Teaching and Learning) at the University of Technology, Sydney. He is an internationally recognised researcher in web engineering, focusing on web system specification and scoping. He is widely published and is on numerous conference committees and journal editorial boards (including Managing Editor of the *Journal of Web Engineering*). He is also passionate about teaching innovations and practice-based engineering education. He was the recipient of the AAEE 2001 McGraw Hill New Engineering Educator Award, and also won the Paul Thistlewaite Award for best paper at AusWeb'2000.

1 Introduction

In order to achieve a better understanding of web systems development, we need to understand the unique characteristics of web systems¹ and how these characteristics impact on the development process. The unique characteristics of web systems are both technical and organisational in nature. The more fundamental of these relate to the way in which web systems are inter-dependant with the organisational processes and contexts within which they exist. Whilst not unique to web systems, they are typically heightened in web systems (Burdman, 1999). Three of the key characteristics are:

- 1 uncertainty in the project domain
- 2 high volatility of the client needs
- 3 rapidly changing technology.

Technological changes and frequent communication between clients and developers impact on the stability of the project requirements, thus making the requirements more volatile (Zowghi *et al.*, 2000).

Like any system, in order for a web system to be successful, it is critical that the web system meets both functional and non-functional requirements. Non-functional aspects such as security, privacy, integration and system performance are crucial to the development of high-quality web systems (Kirner and Davis, 1996; Mylopoulos *et al.*, 1992). Web development typically tends to focus primarily on the Functional Requirements (FRs) as web developers are more inclined to scope a project through an understanding (and representation) of the functional aspects of the web system. Non-Functional Requirements (NFRs) have to be handled and expressed very early in the process of development (Cysneiros *et al.*, 2001).

This paper represents a preliminary stage of a long-term research project which overall objective is to understand and improve web development processes. At this stage we report on interviews with web developers. Our data has led us to investigate the importance and impacts of NFRs on web development projects. We will report on key

lessons with regard to how NFRs should be handled. In subsequent work we will use these lessons to propose a methodology for guiding the most appropriate derivation of NFRs. This methodology will help analysts to rigorously derive and validate the NFRs.

2 Related work: requirements for web systems

Although there is a significant body of research on requirements management and requirements engineering techniques, little of this has been related directly to the specific consideration of requirements in web projects (Lowe, 2003). Zowghi and Gervasi (2001) state that unlike conventional development, web-based system development typically does not differentiate between requirements elicitation and analysis, and system design. Web developers use the early stages of design to elicit more detailed feedback from clients about the requirements of the system.

In web systems development, it appears that web developers are more concerned with functional aspects of the systems, such as user interfaces, the navigability of the pages, 'add-on' to database design, the layout and structure of the pages and the various functionalities that the web systems need to provide (Atzeni *et al.*, 1998; Baresi *et al.*, 2000; Bonifati *et al.*, 2000; Rossi *et al.*, 1999). These functional aspects are often articulated as FRs, based on previous experience in handling similar projects within the same domain, and also by learning from the competitors. NFRs, on the other hand, do not express any functionality to be implemented in the future system directly (Cysneiros *et al.*, 2001). To the contrary, they express behavioural conditions and constraints that must exist in systems development and operation. In addition, Xu *et al.* (2006) states that non-functional requirements are difficult to address due to their unique qualities, such as their conflicting nature, crosscutting nature and open-ended nature.

Although, NFRs such as security, data and system integration, and system performance are crucial to the development of web systems, it seems that they are not given the same emphasis as FRs. The difficulty with articulating NFRs for web system projects lies in identifying and predicting possible causes and impacts that NFRs have on the system and its domain. This is partially due to the uncertainty when the web developer does not understand the domain completely before building the web system, leading organisations to make decisions without complete information.

3 NFR impacts in web systems projects

One of the difficulties with articulating NFRs for web systems projects lies in identifying and predicting possible causes and impacts that NFRs have on the web system and its domain. Web systems tend to fundamentally alter the interface and/or relationship with external stakeholders, thus having a more substantial and immediate impact on the problem domain (Yusop *et al.*, 2004). The functional and non-functional aspects that such systems could or should provide tend to impact highly upon the organisational context, deeply affecting the operational processes, organisational structure, and the working habits of the individual employee, at each organisational level (Donzelli and Setola, 2002). Xu *et al.* (2006) ascertain that non-functional requirements have not received as much attention as FRs, despite the fact that they greatly impact the resulting

design of the system. In addition, according to Xu *et al.* (2006), to correctly design a system to satisfy NFRs early in the software life cycle is very critical as correcting them later in the development can be extremely costly.

With new technology such as web systems, new forms of threats are encountered regularly. These threats include security threats such as illegal access to proprietary data, virus attacks, denial of service attacks, and illegal acquisition or distribution of data that is copyrighted. Security has the potential to have a high impact on the organisation and its systems as there is no security technology that can secure the system completely (Medjahed *et al.*, 2003; Yang and Papazoglou, 2000). Security plays a great role as it involves various technologies and since the system is available on the web, it is more exposed to security threats. Another vital NFR for web system is systems integration. Organisations need to ensure that different web applications and tools are able to integrate seamlessly with the organisation's business processes, existing system and the systems of its trading partners. Baghdadi (2005) states that a business process is a dynamic composition process which requires well-specified, dynamic, cost-effective, timely business interactions among business functions and business objects. According to Baghdadi (2005), integration is a complex issue. This is because it requires first unlocking the heterogeneous and independent implementations of the business functions and business objects to make them visible, accessible and then connecting them, which is the responsibility of the connecting technology. Poor integration of the web applications and tools leads to negative impact for both the organisation and the system (Bohner, 2002; Medjahed *et al.*, 2003).

4 Research methodology

This section covers five interviews of web developers in Singapore about the web system projects that they have recently developed. Initially eight web developers were contacted to participate in these interviews but three web developers had to withdraw their participation due to work commitments. The criteria for eligibility in participating in the interviews were that the developers must have had experience within the last two years in developing web systems. The objective of the interviews was directed towards exploring in what way the domain knowledge was elicited and identifying how this process informed the web developer about constructing the domain model. The numbers of web developers may be considered as a small sample size. However, the web systems that they developed are complex in nature and involved NFR issues that are critical and worth reporting. These interviews were an initial step of the research as it gathered some preliminary findings which are pertinent to the study. This is important as it serves as a basic groundwork for a more thorough search of empirical evidence in the next step of the research.

This is qualitative research in which a set of open-ended questions were used. The web developers were to answer the questions in reference to a particular recent web system project. Each interview took approximately two hours to complete. All responses and comments were written on the questionnaire itself and queries via electronic mail were used to further clarify missing or ambiguous responses.

The five web developers were interviewed on separate occasions at their respective workplaces. The questions that were asked related to:

- the kinds of questions the web developers asked their client/user groups
- the types of issues they discovered during development
- the challenges that they face
- the solutions that they attempt
- the level and nature of communications that they have with the client/user groups.

During the interviews, more questions emerged as the web developers gave candid responses, which were not part of the questions listed in the questionnaire. These responses are very valuable as they illuminated web development and project issues that were previously not anticipated.

Content analysis was used as the research method for making valid inferences from the web developers' responses during the informal interviews (Michael and Lewis, 1994). The primary aim was to determine in what way the domain knowledge is elicited, the issues in NFR those are identified during this process, and in what way these issues are significant to the web project. There are two types of content analysis: qualitative and quantitative. Here, we focused on analysing the data qualitatively in that the primary aim was to determine in what way the domain knowledge is elicited, the issues that are identified during this process, and in what way these issues are significant to the web project. In analysing the data qualitatively, the contents of the responses to the informal interviews' questions were coded in terms of predetermined and precisely defined characteristics (Leedy and Omrod, 2001).

5 Results and analysis of interviews

The informal interviews were initially directed towards exploring the ways in which domain knowledge was elicited and identifying how this process informed the web developer about constructing the domain model. During the informal interviews, many issues particularly relating to non-functional aspects were discussed by the developers. This was actually not the main focus of the study; however it became clear during the informal interviews that NFRs play a pivotal role in web projects. Some of the web developers were interviewed for the second time to provide more detailed responses from the first informal interview. The projects that these web developers were involved with contain several issues that highlight the criticality of NFRs in web projects. Interestingly, this is in contrast to most of the existing literature discussing web systems development, which focuses on the FRs and their importance in determining the success of the web systems.

5.1 Overview of web projects

In the following subsections, the web projects are analysed and discussed in terms of the way the web developers elicited the systems requirements, the types of requirements they gathered, and finally the importance of NFRs in these web projects. In Section 5.3, a web project (ALI) involving the development of an intranet and extranet application will be considered in additional details.

Table 1 below illustrates the description of web projects. The company names do not reflect actual names so as to maintain anonymity of the organisations. In the following sub-sections, the web projects are analysed and discussed in terms of the way the web developers elicited the systems requirements, the types of requirements they gathered, and finally the importance of NFRs in these web projects.

Table 1 Description of web projects

<i>Web project name</i>	<i>Description of web projects</i>
Alpha Life Insurance (ALI)	Development of an intranet (E-Staff) and extranet system (E-Cargo). The E-Cargo system needs to be connected to E-Staff. The E-Cargo system is for shipping companies to buy insurance for their cargo. ALI's staff needs to access both systems to process documents and manage various operations.
Beta Real Estate Agency (BREA)	Development of a fully function website for a real estate company to promote the property sales and rental uptake.
Charlie Tour and Travels (CTT)	Development of online travel website, mainly focusing on the front-end providing information to prospective customers
Delta Land Valuers Institute (DLVI)	Reconstruction of current website to automate workflow processes using a web-based application system.
Echo Telecommunications (ET)	Development of an online ordering system for vendors to query and submit telecommunication goods orders

Table 2 lists the roles within each project. It also explains the time frame of the web project, from the stage where the requirements were gathered to the final completion of the project. This table will be useful for reference and understanding the different types of people who are involved in these web projects.

Table 2 Allocation of development and client/user group

<i>Groups and time frame</i>	<i>ALI</i>	<i>BREA</i>	<i>CTT</i>	<i>DLVI</i>	<i>ET</i>
Development group	1 WD, 1 BA, 1 ITM, 1 PM	1 WD, 1 ITD	1 WD	1 WD, 1 ITM, 1 external ITC	1 WD, 2 ITM
Client/Users group	1 BA, 2 PM	2 DM, 1 ITM	1 BO	4 DM, 2 Directors	2 DM
Time frame from project commencement to completion	10 months	3 months	3 months	3 months	2 years

Notes: WD – Web Developer; BA – Business Analyst; ITD – IT Developer; ITM – IT Manager; ITC – IT Consultant; PM – Project Manager; DM – Department Manager; BO – Business Owner.

5.2 *Eliciting of requirements and types of requirements gathered*

In eliciting requirements for web systems projects, the web developers used various methods. Most of these methods were informal and took several rounds of consultations with the clients/users. The common communication method was to interview the relevant personnel within the client organisation and/or system users, which is then followed by follow-up questions via e-mail or telephone. In order to understand the web systems

domain better, the web developers asked specific questions which related to the client's business model, business operations, technology constraints, business processes and others.

The types of questions the web developers posed to the client/users related to the functional and non-functional aspects of the systems. The web developers were initially given a project brief, which contained basic descriptions about the functions that the web system must have. It also contained information such as the budget and schedule with which the web project needed to comply. NFRs were not specified by the client nor elicited in detail by the web developer during the initial stage of the projects. The web developers' tasks involved planning, designing and developing the web projects according to the requirements specified by the clients. The web developers faced challenges in obtaining information from the clients due to the inherent uncertainty and lack of development knowledge and experiences.

For CTT, the web developer asked about the business operations and how the client's organisation could communicate and perform their work processes with the vendors, such as hotels and airlines. The focus during the elicitation sessions were mostly geared towards finding functional aspects for the system.

The web developer for project BREA indicated that the client initially provided him with only a two-page project description, stating the basic functionality that the systems must support. In addition to a project description, the web developer for the CTT project also asked about the business operations and how the client's organisation performed their work processes with the vendors. The focus during the elicitation sessions were mostly geared towards finding functional aspects for the system. For DLVI, the web developer commented that though functional aspects of the system were clarified, the NFR aspects remain unclear. The web developer contacted the client to clarify NFR issues:

“...we asked questions about security and integration issues. The Web system should also provide access control and security for all levels and integrate applications with Accounting system. We had long hours of meetings with clients after the Website is created to discuss about linking the Webpages with the backend systems in detail.”

The above statements focused on the client requirements. These requirements contain NFRs issues such as security and integration which were critical to the system. The web developer was able to gather such requirements after several rounds of discussions with the client. These NFR issues were not clear from the beginning as both the web developer and the client were uncertain about the systems' behavioural conditions and constraints.

5.3 NFRs and their importance to web projects

It was observed that the web developers were not aware that the issues they had were in fact problems relating to NFRs. Only ALI project clearly identified the issues as aspects of NFRs. The clients particularly were more interested in functional aspects of the project. Non-functional issues such as security, system performance, data integration, data confidentiality, and privacy were not discussed in detail. The web developers discovered that such NFRs became issues that impacted on their web systems projects only at the later stages of the web development. As a result, the system design was

changed frequently. These changes complicated the web projects as in most of these projects a proper document was not developed or maintained to record such problems and rectifications.

5.4 Alpha Life Insurance (ALI)

We describe here the interviews conducted with respect to the ALI project. This project is explained in greater detail due to the richness and pertinency of information provided by the web developer as it highlights several NFR issues that are critical to the development of web systems. Due to space limitation in this paper, it allows for only one project to be discussed in detail. This web project involved the development of a local staff intranet site, referred to as E-Staff, and also an electronic cargo (E-Cargo) extranet site where vendors can log in and enquire about and purchase cargo insurance.

5.4.1 Web development team structure

The web development team was lead by a Project Manager who coordinated the development team, which consisted of a web developer, a business analyst and an IT Manager. The web developer initially conducted a general meeting with the Users Group, consisting of a Business Analyst and two Project Managers. The Users Group comprised the various departments who provided the requirements when contacted by their Business Analyst. The Business Analyst for the web development team was in charge of gathering/eliciting requirements for the web developer. The web developer's responsibility involved the design and structure of the website, and the overall completion of the web system development.

The web development team was initially given a Project Brief by the Users Group which stated the basic requirements for the project. The web developer then discussed the detailed development plan with the rest of the web development team. All issues that needed solutions were documented by the Business Analyst who was involved in discussing them with the Users Group's business analyst. Shown below in Figure 1 is the communication structure between the web development team and the Users Group. Figures 1 and 2 are designed based on the verbal descriptions given by the web developer.

Figure 1 Communication structure between the web development team and the Users Group

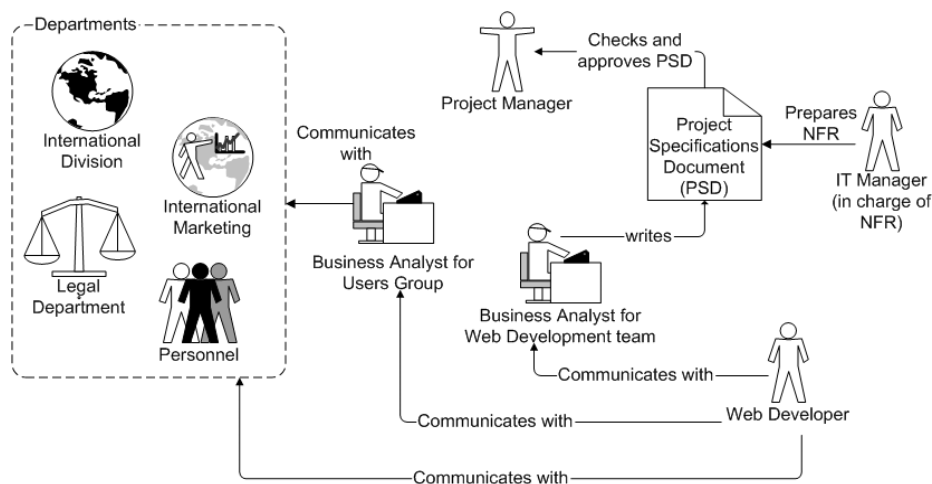
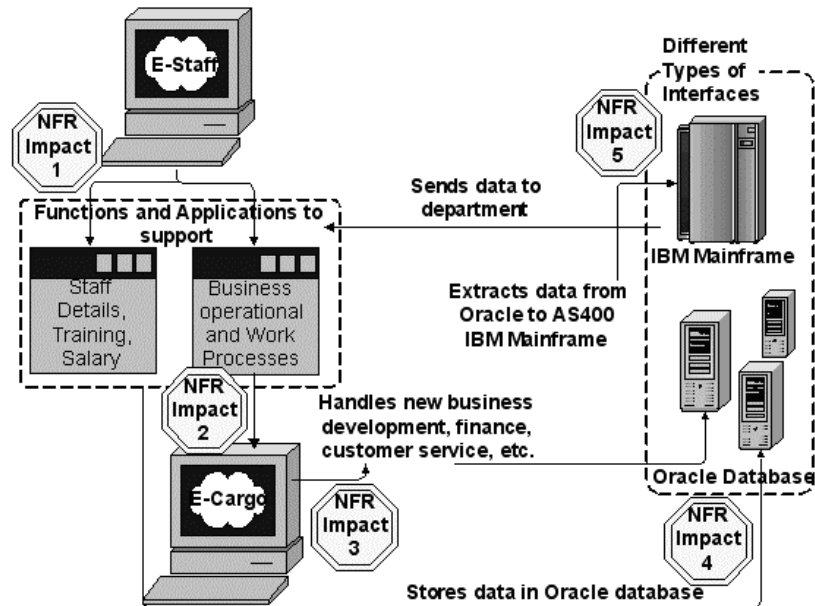


Figure 2 ALI web system structure



Although the development team structure was not directly related to the main message of this paper, it is, however, an important management issue that deserves consideration. As a consequence of a chaotic team structure, the web developer had to take many short cuts to complete his tasks. One of these short cuts resulted in not maintaining the project specifications document in a timely manner. Consequently, NFRs did not get properly documented and maintained.

5.4.2 Shaping of requirements by the web development team

The team started planning for the project using limited guidelines stated in the Project Brief. The web system project was divided into two phases. Phase 1 started with raw requirements extracted from the Project Brief. As the Project Brief did not contain detailed specifications of the FRs and NFRs for the project, the web developer had to get the Business Analyst for the Users Group to clarify certain functions that were unclear or ambiguous. A new document, which is the Project Specification Document (PSD), was developed. However, most of the communications were not documented in a structured document, nor archived appropriately to facilitate future reference. The changes made were also not updated in the PSD. There was also no proper electronic archiving method to archive the communications. The web developer made the following comment about eliciting requirements for the domain:

“I have to contact the Business Analyst of our Web development team to work out the functional specifications. The users, such as the departments’ personnel provide the requirements to their Business Analyst. Our Business Analyst provides the functional specifications. The other Business Analyst will then check the functional specifications and sign off.”

From the comments above, it is clear that the web developer had to communicate the systems requirements with the Development team's Business Analyst. However, communications then became chaotic between the personnel in both groups and follow-up was very difficult to achieve. At the mid-stage of Phase 1 various problems were encountered. The web development team was handling several projects simultaneously. Due to this the Project Manager who led this web project had to leave the web developer to make all the critical decisions. This added to the chaos in this project as the Project Manager understood the departments' structure and business operations better than the other personnel in the web Development team. During the initial stage of the development, problems regarding issues not foreseen before also surfaced. Issues such as security, integration of data between departments, sensitive data that requires access control and systems integration and interoperability issues became prominent. The Business Analysts were unable to resolve the issues in a timely manner because they were unaware of the complexity of the problems.

The web developer therefore assigned an IT Manager to compile more detailed NFRs for the project and to include them in the Project Specification Document (as shown in Figure 1). The NFRs were rewritten and this involved many discussions with the Users Group's Business Analyst and in most cases, the IT Manager had to personally contact the department users to discuss the requirements in detail. The web developer states the following:

“There is one team member, the IT Manager who will specifically deal with all the NFRs and prepare the specification document for this project.”

In order to have a more complete understanding of the business domain, the web developer asked the Business Analyst for information relating to user requirements, functions, the kinds of business processes the departments deal with, and the meaning of each data element. By discussing with other departments, who are direct users of the system, the IT Manager discovered new NFR-related issues not discussed before. The details of these NFRs and their impacts are discussed in the next subsection.

5.4.3 *Non-functional requirement impacts*

There were several NFRs that impacted on ALI's project. These NFRs (relating to the security aspects of the system, data integration issues and the requirements for backing up of the data) only became critical at the end of the first phase of the project. As shown in Figure 2, both systems (E-Staff and E-Cargo) needed to be integrated, where the staff from relevant departments would be able to access the E-Cargo site and perform duties such as providing quotations, promoting insurance packages and managing payment related matters. The kind of functions that the E-Staff system needed to support were for staff to access and update staff details and submission of different forms such as course registration forms, course evaluation forms, leave forms, *etc.* Figure 2 also shows five types of NFR impacts in this project:

NFR Impact 1 The systems security and data safety are impacted. This was a high level impact as information such as staff details; staff training and salary were private. The web developer had to ensure that appropriate access control was being enforced. The work processes also involved private data that required restrictive access.

- NFR Impact 2 The timing and speed aspects of the system were impacted. The web developer was concerned with streamlining the work processes and business operations that were previously done within the information system of the website. The integration of the current system to the web system had to be seamless and should not affect the speed of the system. This was because in certain parts of the day, the traffic to such sites would be higher as it was accessed by many personnel. The web developer also had to ensure that all data was properly backed up so that no data would be lost if the network fail.
- NFR Impact 3 Security and safety related aspects of the system impacted this part of the project. The system stored information such as customer's details and details of policy coverage. This was confidential and each corporate policy holder was given a unique directory where only the corporate policy holder and the person they liaised within ALI could have access. As this system facilitated online payments and digital signatures, proper security tools were used. This requirement was not previously highlighted during the requirements elicitation sessions. The IT Manager in charge of documenting the NFRs conducted a separate meeting with the User Group to discuss details of these NFRs.
- NFR Impact 4 The NFR that impacted on the web system here was interoperability, whereby both systems (Oracle Database and AS400 IBM mainframe systems) needed to interoperate seamlessly and effectively. Information displayed in the E-Staff and E-Cargo system was being stored in an Oracle Database. Whenever needed to meet the work processes and business operations, the data from the Oracle database was being extracted to the AS400 IBM mainframe system. The technical expertise required at this stage was underestimated. The Project Manager, IT Manager and the web developer had to have numerous meetings to structure the technical infrastructure of the ALI's overall system.
- NFR Impact 5 In transferring the data from the AS400 mainframe to the E-Staff web system, the web developer faced system performance issues which required elaboration, particularly as it related to the data back-up issues. The departments required the data to be backed up at certain times of the day and clients needed to receive the data at specific times. These types of constraints impacted on the web system and such additional requirements led to numerous discussions that delayed the project completion.

6 Lessons and implications

From the analysis of the data, we conclude that it is critical for web developers to articulate all aspects of NFRs. We observed that NFRs were usually articulated in certain amounts of detail only after the web project had commenced. The web developers in these projects focused more on the functional aspects of the system. However,

it did not imply that NFRs were less important at the initial stages of web projects. Based on the results from the analysis of the interviews; the following lessons learned are presented.

6.1 Emphasising the importance of NFRs

Paying due attention to NFRs in a timely manner improves the resulting web systems. It also has the potential to reduce the development time while utilising the involved personnel's expertise to the fullest. In the projects reported, the web developers documented the requirements but only to a limited level of detail. However, functional issues such as processing online payments and updating users' personal data posed significant constraints to the project. It is therefore imperative that both the web developers and the client are fully aware of the importance of NFRs at the beginning of the project.

Web developers need to update the documents after each discussion they had with the client/system users. Updating the document regularly, however, was not a common practice with these projects. For example in the ALI project, the chaotic communication between the web development team and the Users Group led to poor documentation and follow up on the outcomes of NFR discussions. The Development team was unable to refer to the project specifications document for updated NFRs. Proper documentation of both NFRs and FRs is vital as this facilitates requirements traceability. Requirements cannot be managed effectively without requirements traceability (Kotonya and Sommerville, 2002). Traceability is an important aspect since it would make future references to the source of the NFRs easier (Cysneiros *et al.*, 2001). These references may be due to a change related to this NFR, conflict resolution, or even the need of a further NFR refinement in cases where doubt has risen.

6.2 Good understanding of the domain

Before eliciting requirements, web developers need to have a good understanding of the domain. Organisations (clients) on the other hand, need to have a good knowledge of their internal and external environment. This refers to understanding the environment where the web system is being developed. In BREA, and DLVI project, the web developers worked for web consultancy companies and were unaware of the client organisation's business processes and operations. As the projects involved reconstruction of the companies' websites and migrating the employees' data to web systems, the web developer needed to understand the client's internal and external environment. This was a difficult task in both projects as only by having good understanding of the domain, can the NFR issues become more obvious. In these projects NFR issues such as privacy, safety, security, systems integration and interoperability were paramount and the web developers were able to articulate these NFRs only after they understood the organisation's structure, business process and business operations. According to Glisson *et al.* (2006), security must address the people and organisational issues, in which they have to be a way to establish trust with employees, and that both employees and developers need to know how security fits into the work environment (*i.e.*, their development process), and the potential impact security has on the web application solution that they are introducing into the organisation.

6.3 *Lack of rigor in articulating NFRs*

Web developers need to elicit and understand the web system's NFRs before the project commences. As discussed in Section 5, in all the projects, the web developers experienced great difficulty in gathering a complete set of NFRs in the initial stage, particularly as the clients/users themselves were uncertain about the potential constraints that the web system might have. The web developers had to rely on previous experience in developing similar systems, albeit only a limited knowledge.

It is pertinent that web developers have a structured way to define the generic NFRs and predict the potential impacts that NFRs will have over the system and its domain. As the web project progresses, more issues involving NFRs emerge. These new issues are important sources of knowledge that need to be documented to facilitate issues traceability when developing web systems. From the interviews, we discovered that NFR-related impacted on the web project, the web system and its domain. The difficulty in documenting and following up on NFRs is due to the complex nature of the impacts.

6.4 *NFRs leading to discovering new FRs*

Further articulation and identification of NFRs leads to the discovery of new FRs. In one of the web projects discussed, the system users were concerned with NFR issues such as usability and application integration. These issues were then resolved and taken into account, leading to more specific descriptions and elaborations of the systems functionality. This subsequently helped in improving the functionality and the design of the web system. This made the documentation of the systems requirements more complete, which allowed the reusability of such knowledge in subsequent web projects.

7 Future work and conclusions

In this paper, we argued that it is a complex task to develop web systems when the FRs and NFRs are not well understood. This is made more difficult when new requirements are continually discovered and implemented. The results from this qualitative research highlight that web developers typically do not pay sufficient attention to NFRs. This is often due to uncertainty, lack of time, lack of knowledge in the importance of NFRs, and that NFRs are not readily available and documented from previous projects for the purpose of reuse. The NFR impacts discussed here are indicators that should be taken as warning signs early in the requirements gathering and system analysis phase to alert practitioners regarding their development process and methodology.

The data further showed that web developers do not elicit NFRs at the same time and at the same level as FRs. Hence, there needs to be a systematic and structured way of documenting the NFRs.

Ongoing work will attempt to address more specific web development project issues relating to the importance of NFRs. Focusing on industrial case studies will provide us with significant data on web projects, specifically on the importance of NFRs. As the next phase of study will include interviewing more web developers, all interviews will be taped and transcribed for more detailed analysis.

Given that NFRs are not easily identified, elicited or documented in web projects, clearly automated tools that could support this process would be beneficial. In particular, tools that facilitate tracking NFR issues and allow impacts to be recorded and articulated could be very useful in web systems development. Such automated tool will enable the tracking and resolution of NFR issues, particularly in guiding developers in resolving issues in a systematic way.

The tool functionalities will facilitate web developers to track NFR when encountering problem of such nature. For example, if the web developer discovers that they have a question about migrating data from a legacy system to a web system, he will be able to track the issues from the issue tracking and issue resolution tool. This tool will provide information such as the person the web developer can contact to resolve the NFR issues, the document and design artefact that he can refer to, other NFR and FR that relates to it, and it also provides information on how this NFR issues have been resolved before in past projects. This tool is an intelligent tool that will provide information to guide web developers in making informed decision about the NFR issue. It will also provide information on how such NFR have been resolved in the pass. This tool therefore will store issues from past projects and will also keep updating the data when newer information become available about resolving NFR issues.

References

- Atzeni, P., Mecca, G. and Merialdo, P. (1998) 'Design and maintenance of data-intensive web site', *Proceedings of the 6th International Conference on Extending Database Technology*, pp.436–450.
- Baghdadi, Y. (2005) 'A web services-based business interactions manager to support electronic commerce applications', *Proceedings of the 7th International Conference on Electronic Commerce (ICEC '05)*, Xi'an, China, 15–17 August, Vol. 113.
- Baresi, L., Garzotto, F. and Paolini, P. (2000) 'From web sites to web applications: new issues for conceptual modeling', *Proceedings of the Workshops on Conceptual Modeling Approaches for E-Business*, pp.89–100.
- Bohner, S.A. (2002) 'Software change impacts-an evolving perspective', *Proceedings International Conference on Software Maintenance (ICSM '02)*, pp.263–272.
- Bonifati, A., Ceri, S., Fraternali, P. and Maurino, A. (2000) 'Building multi-device, content-centric applications using WebML and the W3I3 tool suite', *Proceedings of the Workshops on Conceptual Modeling Approaches for E-Business*, pp.64–75.
- Burdman, J. (1999) *Collaborative Web Development*, Addison-Wesley.
- Cysneiros, L.M., Sampaio, J.C. and Neto, J.M.S.N. (2001) 'A framework for integrating non-functional requirements into conceptual models', *Requirements Engineering*, London: Springer-Verlag, Vol. 6, pp.97–115.
- Donzelli, P. and Setola, R. (2002) 'System applications and experience: handling the knowledge acquired during the requirements engineering process: a case study', *Proceedings of the 14th ICSE*, pp.673–679.
- Glisson, W.B., McDonald, A. and Welland, R. (2006) 'Web engineering security: a practitioner's perspective', *Proceedings of the 6th International Conference on Web Engineering (ICWE '06)*, Palo Alto, California, 11–14 July.
- Kirner, T.G. and Davis, A.M. (1996) 'Nonfunctional requirements of real-time systems', *Adv Computing*, pp.1–37.
- Kotonya, G. and Sommerville, I. (2002) *Requirements Engineering: Processes and Techniques*, USA: John Wiley & Sons.

- Leedy, P.D. and Omrod, J.E. (2001) *Practical Research, Planning and Design*, 7th ed., New Jersey: Merrill Prentice-Hall.
- Lowe, D. (2003) 'Web requirements: an overview', *Requirements Engineering*, London: Springer-Verlag, Vol. 8, pp.102–113.
- Medjahed, B., Benatallah, B., Bouguettaya, A., Ngu, A.H.H. and Elmagarmid, A.K. (2003) 'Business-to-business interactions: issues and enabling technologies', *The VLDB Journal*, Vol. 12, No. 1.
- Michael, S. and Lewis, B. (1994) *Research Practice: International Handbooks of Quantitative Application in Social Sciences*, Sage.
- Mylopoulos, J., Chung, L., Yu, E. and Nixon, B. (1992) 'Representing and using non-functional requirements: a process-oriented approach', *IEEE Trans SE*, Vol. 18, No. 6, pp.483–497.
- Rossi, G., Schwabe, D. and Lyardet, F. (1999) 'Web application models are more than conceptual models', *Proceedings of the Workshops on Evolution and Change in Data Management*, pp.239–253.
- Xu, L., Hendrickson, S.A., Hettwer, E., Ziv, H., van der Hoek, A. and Richardson, D.J. (2006) 'Towards supporting the architecture design process through evaluation of design alternatives', *Proceedings of the ISSTA 2006 Workshop on Role of Software Architecture For Testing and Analysis (ROSATEA '06)*, Portland, Maine, 17–20 July.
- Yang, J. and Papazoglou, M.P. (2000) 'Interoperation support for electronic business', *Communications of the ACM*, Vol. 43, No. 6, pp.39–47.
- Yusop, N., Lowe, D. and Zowghi, D. (2004) 'A domain framework for representation of web system impacts', *Proceedings of the 5th International Conference on Web Information Systems Engineering*, Brisbane, Australia.
- Zowghi, D. and Gervasi, V. (2001) 'Why is RE for web-based software development easier?', *Proceedings of the Seventh International Workshop on Requirements Engineering: Foundation for Software Quality (REFSQ '01)*, Interlaken, Switzerland.
- Zowghi, D., Offen, R. and Nurmiliani, N. (2000) 'The impact of requirements volatility on the software development lifecycle', *Proc. of the Int. Conf. on Software, Theory and Practice (ICS 2000)*, China.

Note

- 1 Whilst we use the term 'web system' throughout this paper, we recognise that this terminology is still somewhat ill-defined within the literature (which includes terms such as web applications, web-based systems, online systems, as well as a host of domain specific terms such as B2B, B2C, e-commerce systems, *etc.*). In this paper, when we use the term 'web system' we are referring to those systems which utilise web technologies as an integral element of a functionally complex system and which typically incorporates interfaces beyond the organisational boundaries.