Specification of Non-Functional Requirements (NFRs) in Request for Proposal (RFP)

Harsimran Kaur, Ashish Sharma
Department of Computer Science and Application
GLA University, Mathura, India.

ABSTRACT
Software Requirement Analysis is one of the most important and critical activity of software development, that affects the development cost and success of end product. This paper proposes to add both formal and informal specifications of NFR to Request for Proposal document. It ought the developers to focus on each NFR right from the beginning of development process. In this paper NFRs are specified informally by extended use case and formally by proposed extension to Formal Reference Model. The paper claims that the proposed add-ons related to NFRs are qualitative one and can be considered as standard add-ons in RFP documents to deal with NFRs. It shall act as important step during early planning system for designing and implementing proposed software. The complete procedure is illustrated using a general case of cafeteria ordering framework.

1. INTRODUCTION
There has been a continuous effort to estimate the important NFRs associated with the particular software, but very few proposals exist that helped developers to understand requirements from software requirements document. It is difficult task to define the user non-functional requirements of complex systems. Thus, before moving on to further phases of software development, it is important to understand the need of customer so that these requirements can be written unambiguously and viably in RFP. This paper considers the issues for specification of the NFRs in RFP. In order to address these issues, the two add-ons have been proposed that can be added to Request for Proposal document as mentioned below:

a. Informal Specification of critical NFRs with the help of use case.
b. Formal specification of critical NFRs by extending reference model beyond its current scope to address the changing needs of many different classes of users.

This research work aims to shift the conventional practices of software engineering to the requirement based software engineering paradigm by focusing NFRs at a much early phase of software development life cycle. The paper is organized as follows. Section 2 discusses a survey of researches carried out so far in various areas related to proposed approaches. Section 3 discusses the proposed add-ons to the current format of RFP with the help of an example. Section 4 evaluates quality of RFP document, by qualitative and quantitative metric. The final section concludes the work followed by references.
2. RELATED WORK

According to IEEE [13]: "non-functional requirement (NFR) – in software system engineering, a software requirement that describes not what the software will do, but how the software will do it, for example, software performance requirements, software external interface requirements, design constraints, and software quality attributes". Boehm [2], McCall [23], Roman [29], Grady [11], Sommerville [31], ISO 25010[14] model, Glinz [10], Mairiza [22], Chung [6] addressed concerns to identify and classify the NFRs. Chung compared different quality models [6]. Chung argued to rethink the notion of NFRs as there is no fact which explains the meaning of NFR. But none of the model has been able to solve various inconsistencies in NFR. Mairiza introduced novel classification of NFRs based on typology, definition, types of systems, and application domains to benefit software engineering community. In Modelling, most of work use informal modeling with UML to deal with NFR as discussed in [19]. Many researchers proposed extensions of Use Case and other diagrams like class diagram, activity diagram to specify NFR as discussed in [4], [7, 8], [16], [18], [20], [25], [29], [31]. There are other approaches like Relational Model, Petrinets, Multimodel, NFR Framework, NFR Framework + that can be used to specify NFRs [33]. But no formal model for NFR specification has been found in literature. Chung [6] discusses the need of an integration of NFRs into other requirements models, such as the reference model [21] and the four variable models [48]. As both are functional models and NFR’s are completely ignored in both the models. Miller in [24] has extended the four-variable model to bridge the system-software gap. Gunter in [12] has compared the two models and concluded that “Reference Model supplies what is missing in four variable model”. But NFRs are completely ignored in this model. Therefore, an attempt has been made in this paper to extend the Reference Model for formal specification of NFRs.

3. PROPOSED WORK

In order to understand the proposal we have considered the example of a simplified version of Cafeteria Ordering System (COS) [32] that will permit Process Impact employees to order meals from the company cafeteria on-line to be delivered to specified campus locations. There are four NFR’s identified in RFP [32]. These are:

1. Availability
2. Performance
3. Robustness
4. Security

Figure 1. Informal Specification of NFRs for COS [34]
The informal specification of these NFRs that can be added to RFP is shown in Figure 1 which is extended version of standard use case diagram. The process model for informal specification is discussed in []. Each Use Case with its NFRs is specified individually in separate boundary. It helps to include levels of detail regarding NFRs necessary for the system engineers for system development.

Once the NFRs are specified informally in RFP, it would be easy for the stakeholder and customer to map the NFRs. But, Formal Specification is also required in RFP to specify and analyze the behavior of system. Reference Model [12] is proposed to use for specification of NFRs which otherwise is used to specify functional requirements. Proposed Model shown in Figure 2 takes one step further to add quality artifact into both environment \( e_\text{q} \) and system \( s_\text{q} \) division as shown in Figure 2. \( e_\text{q} \) phenomenon is the part of environment which contributes to specify the quality of system. Similarly \( s_\text{q} \) is the part of system which contributes to specify quality of system. Quality of System here means how it meets NFRs that support the delivery of the functional requirements. Terms denoting phenomenon in \( e_\text{h}, e_\text{v}, e_\text{q}, s_\text{v} \) are visible to environment and used in W and R. Terms denoting \( s_\text{h}, s_\text{v}, s_\text{q}, e_\text{q} \) are visible to system and used in P and M. Therefore \( e_\text{h}, s_\text{h}, e_\text{q}, s_\text{q} \) are visible to both environment and system. Symbols used has the following meanings:

\[ e_\text{h} : \text{Employee order, Cooking and delivering of food, } e_\text{v} : \text{Ordering, Making Payment and Confirmations, } s_\text{h} : \text{Data Flow, } s_\text{v} : \text{Intimation at Cafeteria, } e_\text{q} : \text{Speeding up the order processing, } s_\text{q} : \text{Responding to orders and accounting the time gap between order and delivery} \]

Figure 2. Extended Reference Model

Specifications of NFRs formally for COS are defined as a set of related requirements, environment properties, system properties and specifications:

1. Security: Requirement allows that there should be consistency of W, P and M together. For this \( e_\text{h} e_\text{q} e_\text{v} s_\text{h} s_\text{v} s_\text{q} \). W should be consistent. To make environment consistent with the system the desired property is \( e_\text{h} e_\text{v} e_\text{q} s_\text{h} s_\text{v} s_\text{q} \). W should be consistent. To make environment consistent with the system the desired property is \( e_\text{h} e_\text{v} e_\text{q} s_\text{h} s_\text{v} s_\text{q} \).

2. Robustness:

3. Availability and Performance of Cafeteria Ordering System can be specified with proposed extension:

\[ e_\text{h} e_\text{q} (t) - s_\text{h} (t) s_\text{q} (t) < t_{\text{max}} \] where value of \( t_{\text{max}} \) can be set on the basis of application. It will put restriction on the response time. These specifications are verified by LoTrec 2.0.

Hence, there is a need to incorporate extended use case model and formal reference model specification for any proposed software in RFP document. This incorporates helps to improve the quality of both RFP and Software as indicated by the result analysis done in next section.

4. RESULT ANALYSIS

The quality of the RFP is validated by using following check list questions on requirement specification as listed below:

Q1. Does each requirement have some origin?
Q2. Is each requirement achievable in customer environment?

---

*Specification of Non-Functional Requirements (NFRs) in Request for Proposal (RFP)... (Harsimran Kaur)*
Q3. Each requirement is testable once implemented?
Q4. Is each requirement bounded and unambiguous?
Q5. Do any requirements conflict with other requirements?
Q6. Is the requirement traceable to goals of the system?
Q7. Is the requirement is bounded in quantitative terms?
Q8. Are requirements stated clearly? Can they be misinterpreted?

Questions Q1–Q8 was taken from the Pressman [27]. In total, we have received 40 complete responses shown in Figure 3 based on three level grading scales. The results indicate that the majority of the stakeholders find the incorporates suggested indeed raise the readability of RFP in terms of NFRs. However, some responded it as inadequate.

The cognitive dimensions study will provide additional insight into the validation of the proposed work as shown in Table 1.

Table 1. The results of the cognitive dimensions questions

<table>
<thead>
<tr>
<th>Cognitive Dimension</th>
<th>Agree (%)</th>
<th>Neutral (%)</th>
<th>Disagree (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Viable</td>
<td>87.5</td>
<td>12.5</td>
<td>0.05</td>
</tr>
<tr>
<td>Achievable</td>
<td>75</td>
<td>12.5</td>
<td>12.5</td>
</tr>
<tr>
<td>Testable</td>
<td>95</td>
<td>2.5</td>
<td>0.25</td>
</tr>
<tr>
<td>Unambiguous</td>
<td>95</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>Interdependency</td>
<td>72.5</td>
<td>10.0</td>
<td>17.5</td>
</tr>
<tr>
<td>Traceable</td>
<td>97.5</td>
<td>2.75</td>
<td>0</td>
</tr>
<tr>
<td>Bounded</td>
<td>50.5</td>
<td>25</td>
<td>25</td>
</tr>
<tr>
<td>Visibility</td>
<td>77.5</td>
<td>10.5</td>
<td>12.5</td>
</tr>
<tr>
<td>Premature Commitment</td>
<td>0</td>
<td>40</td>
<td>60</td>
</tr>
<tr>
<td>Mental Effort</td>
<td>40</td>
<td>40</td>
<td>20</td>
</tr>
</tbody>
</table>

The results of the cognitive dimensions are shown in Table 1 based on a three level grading scale. Most cognitive dimensions scored highly in favor of the proposed technique, which explains the favorable evaluation result. The hard mental effort dimension received lesser scores as the participants have indicated that applying the technique requires some degree of skill on part of its users. Hence proposed work can be considered while specification of NFRs in RFP.

5. CONCLUSION

The work proposed add-ons in RFP documents to deal with non-functional requirement. The entire approach is based on Requirement Engineering document so it is for sure that an RFP must have all the characteristics, content and functionality to make this estimation precise and perfect. The proposed add-ons include extended use case for informal specification of NFR and extended reference model for formal specification. Finally on the basis of result and validation it can be observed that the proposed measure
follows the trend of all the other established measures in a comprehensive manner. The proposed add-ons should incorporate in standard form of RFP and this will aid the developer and practitioner to consider NFR at very early stage of software development.

REFERENCES


