Discovering Software Process and Product Quality Criteria in Software as a Service

Maiara Heil Cancian[1], Jean Carlo Rossa Hauck[2],
Christiane Gresse von Wangenheim[3], and Ricardo José Rabelo[1]

[1] Department of Automation and Systems - Federal University of Santa Catarina - Brazil
[2] Graduate Program in Knowledge Engineering and Management,
Federal University of Santa Catarina - Brazil
[3] Graduate Program in Computer Science (PPGCC),
Federal University of Santa Catarina - Brazil
maiara@das.ufsc.br, {jeanhauck,gresse}@gmail.com,
rabelo@das.ufsc.br

Abstract. SaaS (Software as a Service) has become one of the fastest-growing innovative fields of the IT sector. Yet, as any other software intensive organizations, SaaS providers also need to deliver the service quality they offer. And, although, today exist a multitude of software quality models, so far, a specific adaptation does not exist to the SaaS context. Therefore, we present the results of our research on discovering software process and product quality criteria in the SaaS scenario. We adopted a research methodology, including, domain analysis, stakeholders interviews and literature review to elicit quality criteria and a survey to validate and prioritize the identified criteria. Such a set of identified quality criteria may help service clients to select providers as well as serve as a basis for a mapping relevant software process areas and best practices in order to adapt existing capability/maturity models and standards to this specific domain.

Keywords: Software process quality, software product quality, SaaS - Software as a Service.

1 Introduction

The Service Oriented Architecture (SOA) paradigm has introduced a new outlook on system design and integration, where all system features are treated as independent and self-contained software services. There are many technologies to implement SOA solutions and web services are currently the most used [1], [2]. Following this trend, business models that employ this paradigm have been widely adopted, like in the case of Software as a Service (SaaS). SaaS is an availability model for software services offered to clients through the Internet that are accessed on demand and paid per use [3]. Among the advantages we can highlight that software solutions can be more rapidly composed, avoiding high investments on infrastructure and IT administration. So, the provider (organization that delivers a service) offers the service through the Internet and creates a Service License Agreements (SLA) [4]. In this model, clients seek

M. Ali Babar, M. Vierimaa, and M. Oivo (Eds.): PROFES 2010, LNCS 6156, pp. 234–247, 2010.

and select services that, when invoked, will be used by their local applications, no matter, if it is isolated or integrated into a Business Process Management (BPM) environment. As such, clients need to assure that the quality of the evocable services is reliable [5]. On the other hand, in order to be competitive within the market and to provide quality services that meet the SLA established with the clients, SaaS providers must deliver the service quality they offer. And, as any other kind of software intensive organization today, organizations that provide SaaS solutions also need to improve the maturity of their software process as a whole.

Yet, the remaining question is: what are exactly software process and product quality criteria to be considered in the SaaS scenario. In general, quality is defined as the satisfaction of a product or a service to specific customer needs [6]; perception and satisfaction of the market needs, use adequacy and/or process results homogeneity [7]; and/or alignment to the product requirements [8]. Considering software development, in general, there is a large number of models and standards related to software quality, such as, ISO 9126 [9], as well as software process maturity/capability models and standards, including, CMMI [10] or ISO/IEC 15504 [11] and ISO/IEC 12207 [12]. Yet, considering that those models and standards are intended to provide a generic support for a large range of software organizations, they typically need to be customized to a specific domain or sector to adequately support the specific needs of this environment [13]. In fact, we can observe a current trend to the customization of process capability/maturity models and standards for various domains, such as, OOSPICE [14] for component-based development, S4S [15] for space applications and CMCM [16] for the medical device industry. And, although, there are a great variety of such customizations is being performed, so far, there is not one directed specifically on the SaaS scenario. The existing models focus on traditional acquisition business models. Recently new models have attempted to cope with SOA requirements or services, but not with SaaS in particular. Quality and trustworthiness in loosely coupled computational systems – which is the case of SaaS – have received great attention in the last recent years [17]. Actually, there is neither a largely adopted definition for it nor for the best way to manage it [18].

In this regard, we can observe that also the SaaS scenario, which combines clients' needs and providers' support, a number of specific requirements meed to be considered in order to assure the necessary global reliability and trustworthiness. But, so far, there does not yet is a customized model indicating important software process and product quality criteria for SaaS [19]. The existence of such a customized process reference model indicating a set of best practices specifically relevant within the SaaS scenario could optimize software process improvement investments and improve the quality of the provided services.

In this context, we describe in this paper a first step regarding the customization of existing quality and process capability/maturity models by systematically discovering relevant software product and process quality criteria in the SaaS scenario. In the second section of this paper, we describe related work, followed by section 3, where our research methodology is presented. Results of this work are stepwise described in section 4 to 7 and the conclusions are presented in section 8.

2 Related Work

Today there is a large variety of software quality and process maturity/capability models [13], [20]. Among then, there is the standard ISO/IEC 25000 (SQuaRE - Software Product Quality Requirements and Evaluation) [21], which establishes a view of software product quality criteria by the definition of quality requirements and evaluation. On the other hand, there is a multitude of process capability/maturity models, including, the CMMI framework, ISO/IEC 15504, ISO/IEC 12207, which define a set of best practices for software process improvement and assessment. And, although those models provide a generic set of best practices applicable on a large scope of software processes, they are not specifically adapted to the SaaS scenario.

Recently, reference models focused on services have received increased attention, such as, the constellation CMMI for Services (CMMI-SVC) [22] or ITIL [23]. Yet, these models refer to services in a broader sense than considered in the SaaS scenario. For example, the CMMI for services constellation covers the activities required to establish, deliver, and manage services, which are defined as intangible, non-storable products. And early users of CMMI-SVC have used the model for services as varied as training, logistics, lawn care, etc. In contrast, within the SaaS context, the concept of a service is defined as a component of SOA. Therefore, those models also are not specifically adapted to the SaaS business model.

On the other hand, we can observe a general trend to develop domain-specific reference models, adapting and customizing those generic reference models. Based on the results of a systematic literature review [24], we can observe that there exist more than 50 domain-specific reference models today, covering a large range of domains, such as, automotive software [25], space software [26], component based software development [14], among others. Yet, so far there is not a specific reference model for the SaaS scenario.

3 Research Methodology

In this context, our long term research goal is the development of a customized software process capability/maturity model for the SaaS scenario. In this context, we observe that currently there is not a systematic methodology on how to perform such an adaptation for a specific domain [24] and most of the current customizations have been done in an ad hoc manner. Exceptions are the works done by [27], [28], which describe the adaptation process in detail.

Thus, based on the approach presented by [27], we firstly identify software process and product quality criteria, which will then in a future research step be used as a basis to identify relevant process areas and best practices required to satisfy those quality criteria.

In this paper, we describe the first step of our research, the identification of software process and product quality criteria.

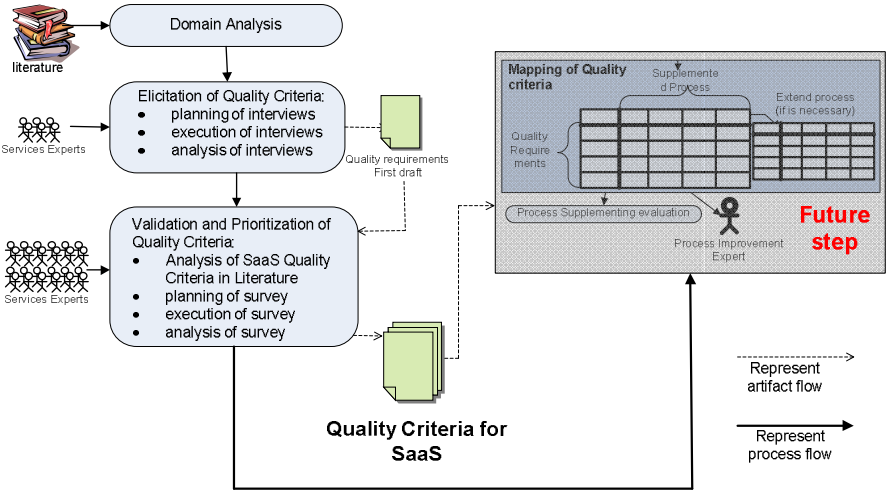

Fig. 1. Research Methodology

In order to identify the software process and product quality criteria we performed the following steps (Figure 1):

Domain Analysis: in order to characterize the specific context of the SaaS scenario and to identify relevant characteristics as well as to identify relevant stakeholders. This step has been done based on literature review. Within this step, we also identified a set of quality criteria for the SaaS scenario described in literature.

Elicitation of Quality Criteria: in order to elicit relevant software process and product quality criteria for services and processes to be considered or demanded from SaaS service providers. Therefore, we conducted interviews with a small group of experienced professionals involved in the use, development and provision of software services. The interviews were planned, executed and analyzed resulting in a list of relevant quality criteria. As result, a set of quality criteria was identified.

Validation and Prioritization of Quality Criteria: in order to review the quality criteria obtained in the previous step and to prioritize them by their importance in the SaaS scenario. Therefore, we mapped and completed the quality criteria obtained in the previous step to quality criteria identified in literature. In a second step, we performed a survey by consulting a greater number of stakeholder representatives and experts, who reviewed (and, if necessary, included new criteria) as well as prioritized the criteria according to their relevance within the SaaS business model.

Based on the results obtained in these steps, we are planning as a future research step the mapping of the identified quality criteria to relevant software process areas and practices by adapting software process capability/maturity models (CMMI and

ISO/IEC 15504) following the methodology presented in [27] in order to specify relevant process profiles within the SaaS scenario.

4 Domain Analysis

SaaS is a software solution offered as a service, which is developed using SOA [29]. The solution is accessed through the internet, saving the implementation and maintenance of a TI infra-structure by the client as the complete structure demanded to develop, process and maintain remains stored at the provider [30]. The client keeps the rights on their data and the software usage. At no point, s/he needs to authorize or buy the software as if it was a product [31]. Those aspects are well known in the Cloud Computing world [32].

Using SaaS, the client assembles their software service portfolio in accordance with their necessity. It can be easily altered, featuring new service or excluding some service hired previously. This is possible, because the release is detached and on-line.

This scenario comes with many benefits. From the client's perspective, they pay only for real usage. Additionally, there is the possibility to use the service for some time, and in case s/he finds the service unsatisfactory, terminate the contract, considering the fact that, in the SaaS scenario, the clients do not own a software license. On the provider's perspective, they can assist the needs of the client more precisely. Figure 2 shows a hypothetical example of a client provider relation in the SaaS model. In this example, the provider provides five services and the client assembles her portfolio with three services.

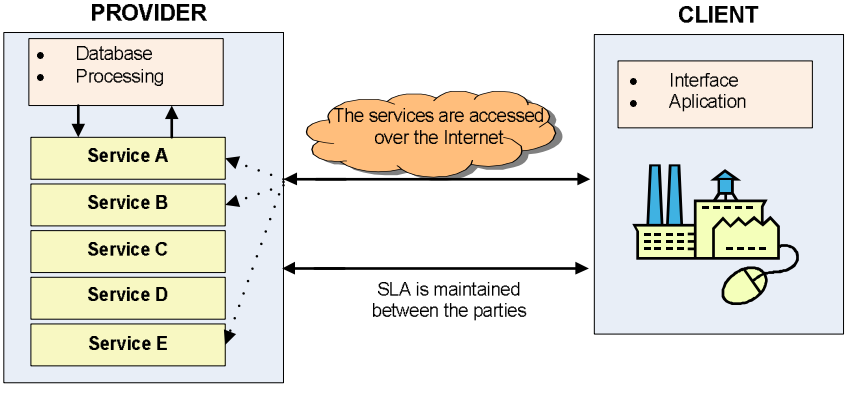

Fig. 2. Overview on the SaaS scenario

What rules a bureaucratic negotiation (referring to usage, execution and service access) between the client and the provider is an Service Level Agreement (SLA) [33].

5 Elicitation of Quality Criteria

Based on the domain analysis, we conducted a series of interviews in order to elicit relevant software process and product quality criteria in the SaaS scenario. As a result of the domain analysis, we identified various relevant stakeholders (Figure 3).

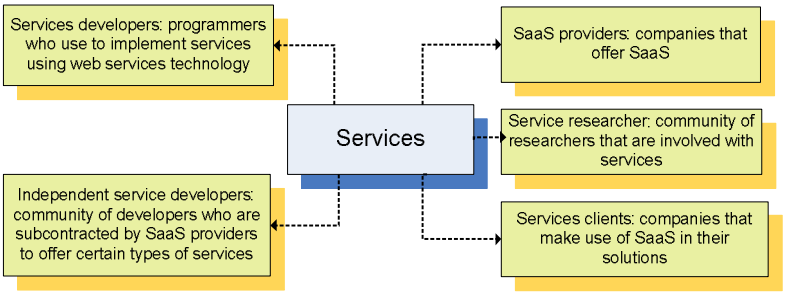

Fig. 3. Stakeholder categories in the SaaS scenario

Based on the domain analysis, structured interviews were planned, including mainly open-ended questions. The goal of these interviews was to identify software product and process quality criteria in the SaaS scenario. To achieve this goal, the following questions were posed:

- General information about the interviewee (experience, work place, etc.);
- Differences between the traditional scenario and SaaS, problems the interviewees observed and their expectations;
- Quality needs when selecting/ providing services in the SaaS scenario.

In total, we performed 6 individual interviews during August/September 2008 with representatives of each of the stakeholder categories. The interviewee's categories are:

- Interviewee 1: service researcher;
- Interviewee 2: independent service developer;
- Interviewee 3: service client;
- Interviewee 4: service developer;
- Interviewee 5 and 6: SaaS provider.

In average, the interviews lasted about 90 minutes.

As a result of the interviews, we analyzed and synthesized the collected information by attributing a formal description to each of the elicited criteria and by unifying items with the same meaning into a unified list of relevant software process and product quality criteria (Table 1).

Table 1. Quality criteria elicited through the interviews

Criteria	Description	Source
Acquisition	Refers to the acquisition process from the point of view of the customer.	Interviewee 1 Interviewee 5
Availability	Refers to the availability of the service for immediate use.	Interviewee 2 Interviewee 4
Change control	Refers to the change control process in order to minimize the impact of changes.	Interviewee 4 Interviewee 5
Help desk	Refers to the support process focusing on the way customers will be assisted when using the service.	Interviewee 2 Interviewee 3 Interviewee 4
Infrastructure capability	Refers to the extent to which the available infrastructure is adequate and sufficient.	Interviewee 4
Interoperability	Refers to the capability of communicating as transparently as possible to other systems.	Interviewee 1
Maintenance	Refers to the maintenance process in order to perform changes in software according to requests.	Interviewee 1 Interviewee 2 Interviewee 4 Interviewee 5
Prevision of continuity of service	Refers to extent to which an organization is able to provide the continuation of technical resources and IT systems / services	Interviewee 4 Interviewee 6
Process improvement	Refers to the process improvement process focusing on the improvement of software process capability.	Interviewee 1 Interviewee 3
Quality control	Refers to the quality control process in order to ensure that the process of the provided service meets the requirements.	Interviewee 4
Requirements development and management	Refers to the requirements development and management process to ensure that the service meets specifications.	Interviewee 1 Interviewee 3 Interviewee 4 Interviewee 5
Security	Refers to the protection of a dataset, in the sense of preserving their value for a person or organization. They are attributes of confidentiality, integrity and availability, security of computational systems, electronic information and data.	Interviewee 1 Interviewee 2 Interviewee 4 Interviewee 5 Interviewee 6
Technically competent employees	Refers to the extent to which employees have sufficient and adequate technical competencies.	Interviewee 1 Interviewee 2 Interviewee 5
Technically competent in business	Refers to the extent to which employees have sufficient and adequate business competencies.	Interviewee 1 Interviewee 3 Interviewee 5
Testing	Refers to the test process in order to verify and validate that the service corresponds to the defined requirements.	Interviewee 1 Interviewee 6
Utilization of standards	Refers to the extent to which services can be accessed and visualized by any person or technology, independently of hardware/software platforms.	Interviewee 4 Interviewee 6
Version control	Refers to the version control process to establish and keep the integrity of versions.	Interviewee 2

This set of quality criteria represents just the first step for the elicitation of SaaS quality criteria, and is being validated and prioritized in the next step.

6 Validation and Prioritization of Quality Criteria

The objective of this step was to validate and prioritize the quality criteria obtained in the previous step. Therefore, we first compared and completed the obtained quality criteria based on quality criteria identified in literature.

Due to the fact, that, currently there do not yet exist quality models specifically for the SaaS scenario, we amplified the scope of related literature, analyzing quality criteria related to the context of web services, which are strongly related to the SaaS scenario. In this regard, a main contribution is the work by Sabata [34] and Mani et al [35]. As a result, the following quality criteria (Table 2) have been identified based on a literature analysis and considered adequate also within the SaaS scenario.

Table 2. Quality criteria based on literature analysis

CRITERIA	DESCRIPTION	SOURCE
Accessibility	Refers to the extent in which a service really provides a service, because a service can be available, but not accessible.	[35], [36], [37]
Integrity	Refers to the behavior of a service executing a transaction. After finishing a transaction, the state of information must be free of inconsistencies.	[36], [38], [39]
Performance	Refers to the throughput (number of requisitions per time unit) and latency (time between sending the requisition and receiving the answer).	[35], [39], [40]
Reliability	Refers to the availability and reliability of IT resources.	[35], [38], [41]
Robustness	Refers to the extent to which a service keeps working even in the presence of inconsistent or incomplete data.	[36], [37]
Scalability	Refers to the capability of processing more requisitions in a time interval without compromising the service.	[36], [41]

Based on this completed set of quality criteria, we performed a survey in order to validate and prioritize the criteria. In order to do so, we prepared a questionnaire listing and describing the elicited quality criteria and requesting the respondents to prioritize each one on a 4-point ordinal scale ranging from essential to unnecessary. Respondents also could include new criteria or make comments. In addition, we also collected demographic information on the background of the respondents. The survey has been made available online using the tool LimeSurvey (http://www.limesurvey.org).

With the objective to involve a larger sample in the survey, we selected stakeholder representatives from various countries by inviting:

- Research groups and individuals, who have been working on software services and are known by the authors through international research projects, mailing lists and special interested groups;
- Authors of related scientific papers of conferences and journals concerning software services;
- A search for relevant professionals based on their resumes posted in the Internet.

We run the survey in 2008 inviting a group of 280 professionals. The survey was available for 60 days and in total we received 84 responses, representing a response rate of 30%. For the analysis of the collected information, we excluded 2 completed questionnaires, as they had been completed by people without sufficient experience regarding the SaaS scenario. The obtained information has been analyzed considering the following objectives:

1. Completeness: in general the respondents confirmed the identified quality criteria. Only two new criteria were indicated (Governance and Reputation) and incorporated into the set of qualities criteria (yet, as these criteria were included only during the survey they were not considered for the prioritization).

	ESSENTIAL	VERY IMPORTANT	IMPORTANT	UNNECESSARY	ESSENTIAL	VERY IMPORTANT	IMPORTANT	UNNECESSARY	TOTAL
Weights:	4	3	2	1	weight x responses number				
	Responses number								
Integrity	57	21	5	0	228	63	10	0	**301**
Reliability	48	30	4	1	192	90	8	1	**291**
Security	45	31	7	0	180	93	14	0	**287**
Accessibility	49	21	13	0	196	63	26	0	**285**
Requirements development and management	41	29	13	0	164	87	26	0	**277**
Infrastructure capability	28	48	6	1	112	144	12	1	**269**
Quality control	32	37	14	0	128	111	28	0	**267**
Acquisition	28	40	15	0	112	120	30	0	**262**
Testing	26	44	12	1	104	132	24	1	**261**
Performance	22	47	13	1	88	141	26	1	**256**
Utilization of standards	29	34	18	2	116	102	36	2	**256**
Change control	26	39	16	2	104	117	32	2	**255**
Interoperability	23	42	17	1	92	126	34	1	**253**
Robustness	24	39	19	1	96	117	38	1	**252**
Availability	28	34	16	5	112	102	32	5	**251**
Maintenance	24	37	20	2	96	111	40	2	**249**
Version control	18	46	18	1	72	138	36	1	**247**
Technically competent in business	19	42	21	1	76	126	42	1	**245**
Technically competent employees	24	31	27	1	96	93	54	1	**244**
Prevision of continuity of service	24	34	21	4	96	102	42	4	**244**
Scalability	17	41	24	1	68	123	48	1	**240**
Help desk	17	40	25	1	68	120	50	1	**239**
Process quality certification	5	30	39	9	20	90	78	9	**197**
Governance	complementation								
Reputation	complementation								

Fig. 4. Priorization results

2. Prioritization: in order to visualize an order of importance based on the classification of each of the criteria by the respondents, we calculated the rating by attributing weights to the classification (4-essential to 1- unnecessary) and then added up the weighted responses of all respondents per criteria. Figure 4 illustrates the results of the prioritization by indicating in the column Total the total weighted sum of each of the criteria.

In general, we can observe that basically all criteria identified in the previous steps were considered relevant within the SaaS context. An exception is the criteria "process quality certification" of the provider, which has been excluded from the set of relevant criteria due to its low prioritization, indicating the unimportance of the item. The requirements which are marked as "Complementation" were suggested by the survey participants and were considered relevant for this scenario.

7 Results: Quality Criteria for SaaS

Summarizing the results from our research, we obtained a set of relevant quality criteria in the SaaS scenario. The resulting criteria have been classified in:

- Product-related quality criteria: criteria related to the provided product and/or service.
- Process-related quality criteria: criteria related to the software process adopted for the development, operation and maintenance of the provided software product/service.
- Organization-related quality criteria: criteria related to the organization providing software products/services within the SaaS business model.

Table 3, 4 and 5 list the identified quality criteria.

Table 3. Product related quality criteria

Criteria	Description
Accessibility	Refers to the extent in which a service really provides a service, as a service can be available, but not accessible.
Reliability	Refers to the extent of the availability and reliability of IT services or resources
Performance	Refers to the throughput (number of requisitions per time unit) and latency (time between sending the requisition and receiving the answer).
Availability	Refers to the availability of the service for immediate use.
Scalability	Refers to the capability of processing more requisitions in a time interval without compromising the service.
Security	Refers to the protection of a dataset, in the sense of preserving their value for a person or organization as an attribute of confidentiality, integrity and availability, security of computational systems, electronic information and data.
Integrity	Refers to the behavior of a service executing a transaction. After finishing a transaction, the state of information must be free of inconsistencies
Interoperability	Refers to the capability of communicating as transparently as possible to other systems.
Robustness	Refers to the extent to which a service keeps working even in the presence of inconsistent or incomplete data.

Table 4. Process related quality criteria

Criteria	Description
Acquisition	Refers to the acquisition process from the point of view of the customer.
Change control	Refers to the change control process in order to minimize the impact of changes.
Quality control	Refers to the quality control process in order to ensure that the process of the provided service meets the specified requirements.
Version control	Refers to the version control process to establish and keep the integrity of versions.
Requirements dev. and management	Refers to the requirements development and management process to ensure that the service meets specifications.
Maintenance	Refers to the maintenance process in order to perform changes in software according to requests.
Process Improvement	Refers to the process improvement process focusing on the improvement of software process capability.
Help desk	Refers to the support process focusing on the way customers will be assisted when using the service.
Testing	Refers to the test process in order to verify and validate that the service corresponds to the defined requirements.

Table 5. Organization related quality criteria

Criteria	Description
Infrastructure capability	Refers to the extent to which the available infrastructure is adequate and sufficient.
Technically competent employees	Refers to the extent to which employees have sufficient and adequate technical competencies.
Prevision of continuity of service	Refers to extent to which an organization is able to provide the continuation of technical resources and IT systems / services
Technically competent in business	Refers to the extent to which employees have sufficient and adequate business competencies.
Utilization of standards	Refers to the extent to which services can be accessed and visualized by any person or technology, independently of hardware/software platforms.
Governance	Refers to factors that show how a company is directed, administered or controlled.
Reputation	Refers to the conceptualization of the provider in the community/market (image).

7.1 Discussion

With this research, we made a first step into the direction of identifying a set of software process and product quality criteria relevant within the SaaS business model. Yet, it is important to consider some limitations of the validity of the results of the research. A question in this context is the representativeness of the obtained results. Within the first step, the elicitation of the quality criteria, we performed only a small number of interviews (6) with representatives of different viewpoints on the SaaS scenario. As a result, this may limit the generaly of the results. Yet, in order to increase the validity, we performed in a second step a validation and prioritization of

the obtained results in form of a survey involving a total of 84 participants. Regarding the survey, on the other hand, the response rate of about 30% of the 280 invited participants may be a limitation. Although, this represents an acceptable response rate, the derived results may not comprehensively represent the knowledge of experts in this domain and therefore may restrict the external validity of the results.

Another question, may be the level of expertise of the participants, as the SaaS scenario is still very recent and the participants experiences may vary largely, affecting the validity of their responses. This is further complicated by the fact, that as SaaS is an emergent area, there is not a solid theoretical foundation yet. This may also have affected the construct validity of the results due to a lack of well-understood terminology in this domain and thus causing misunderstandings by the participants. As much as possible, descriptions and explanations were provided to sustain the questions.

8 Conclusions

In this paper, we identify a set of software process and product quality criteria relevant within the SaaS business model. Within this context, this may contribute to increase its acceptance in a larger scale by helping service clients to evaluate and select service providers. Such a set of quality criteria may be a useful instrument to help services companies to create new sustainable models as well as help to a larger adoption of SaaS models. Especially, when considering the current paradigm shift from local computing systems to several pervasive-based systems running under SaaS model, the demand on service quality tends to become even greater.

In addition, such a set of software process and product quality criteria also creates a basis for the customization of a software process capability/maturity model pointing out relevant best practices specifically within the SaaS domain, guiding software process assessment and improvement. Therefore, one of our next steps is to map the identified quality criteria to software process areas and best practices adapting and evolving existing software process capability/maturity models and standards (such as, ISO/IEC 15504-5/ISO/IEC 12207 and CMMI).

Acknowledgements

This work was supported by the CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico) and CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior), both entities of the Brazilian government focused on scientific and technological development.

References

1. Singh, M.P., Huhns, M.N.: Service-Oriented Computing Semantics, Processes, Agents. John Wiley & Sons, Ltd., Chichester (2004)
2. Tsai, W.T.: Service-oriented system engineering: a new paradigm. In: IEEE International Workshop on Service-Oriented System Engineering, SOSE 2005, Beijing, China, pp. 3–6 (2005)

3. Eliadis, H., Rand, A.: Setting Expectations In Saas: The Importance of the Service Level Agreement to Saas Providers and Consumers. In: SIIA Software as a Service Working Group Software & Information Industry Association (2007)
4. Arenas, A., Wilson, M.: Contracts as Trust Substitutes in Collaborative Business. Computer 41(7), 80–83 (2008)
5. Kourtesis, D., et al.: Discovery and Selection of Certified Web Services Through Registry-Based Testing and Verification. In: Pervasive Collaborative Networks, pp. 473–482 (2008)
6. Juran, J.M.: Juran Planejamento para a qualidade, Pioneira, São Paulo, p. 394 (1990)
7. Ishikawa, K.: TQC, Total Quality Control: Estratégia e Administração da qualidade. IMC Internacional sistemas educativos São Paulo (1986)
8. Crosby, P.B.: Qualidade falada a sério. Mc Graw - Hill do Brasil São Paulo (1990)
9. ISO/IEC, International Organization for Standardization (ISO) / International Electrotechnical Commission (IEC). Standart 9126: Software engineering - Product quality (2004)
10. Team, C.P.: CMMI for Development (CMMI-DEV), in Technical Report CMU/SEI-2006-TR-008, V. 1.2, Editor Carnegie Mellon University / Software Engineering Institute: Pittsburgh (2006)
11. ISO/IEC, International Organization for Standardization and International Electrotechnical Commission, ISO/IEC 15504-5: Information Technology - Process Assessment, in Part 5: An exemplar Process Assessment Model, Genebra (2006)
12. ISO/IEC, International Organization for Standardization and International Electrotechnical Commission, ISO/IEC 12207: Standart for Information Technology. IEEE, New York (1998)
13. Paulk, M.C.: Surviving the Quagmire of Process Models, Integrated Models, and Standards. In: Annual Quality Congress Proceedings, Toronto, Ontario, Canada, pp. 429–438. Carnegie Mellon University, Pittsburgh (2004)
14. Henderson-Sellers, B.B., Rout, J., Patrick, T.: Creating the OOSPICE model architecture, A case of reuse. In: SPICE 2002: the thired international confernce on software process improvement and capability determination (2002)
15. Automation, S.E.a. OOSPICE (2008), http://www.oospice.com/ (10/06/2008)
16. Caffery, F.M., Coleman, G.: Developing a configuration management capability model for the medical device industry. International Journal of Information Systems and Change Management 2(2), 139–154 (2007)
17. Terzis, S.: Trust Management. Guest Editor's Intoduction - The many faces of trust (2009), http://www.computer.org/portal/web/computingnow/archive/april2009
18. Bhargava, B., et al.: The pudding of trust intelligent systems. IEEE Intelligent Systems, 74–88 (2004)
19. Wangenheim, C.G.v., et al.: Systematic Literature Review of Software Process Capability/Maturity Models. In: SPICE 2010 Conference, Pisa/Italy (2010)
20. Sarah, A.S.: Evolution of the Framework's Quagmire. Computer Management, 96–98 (2001)
21. ISO/IEC, International Organization for Standardization and International Electrotechnical Commission, ISO/IEC 25000: Software Engineering - Software Product Quality Requirements and Evaluation (SQuaRE) (2004)
22. Team, C.P.: CMMI for Services (CMMI-SVC) in Technical Report CMU/SEI- PA 15213-3890. Carnegie Mellon University / Software Engineering Institute, Pittsburgh (2008)
23. Taylor, S.: ITIL Service Management Practices - V3 Qualification scheme. Editor, p. 31 (2007), http://www.itil-officialsite.com/Qualifications/Examiners/SharonTaylor.asp

24. Wangenheim, C.G.v.: How domain-specific process reference models and standarts are developed. In: Working Paper, LAPIX/INE/CTC/UFSC, Editor Florianópolis, Brazil (2010)
25. SIG, A.: Automotive SPICE - Process Assessment Model. In: The procurement Forum (2007)
26. Cass, A., et al.: SPICE for SPACE trials, risk analysis, and process improvement. In: Software Process: Improvement and Practice (2004)
27. Richardson, I.: SPI Models: What Characteristics are Required for Small Software Development Companies. Software Quality Journal, 101–114 (2002)
28. Beecham, S., Hall, T., Rainer, A.: Defining a Requirements Process Improvement Model. Software Quality Journal 13(3), 247–279 (2005)
29. Ma, D.: The Business Model of Software-As-A-Service. In: IEEE International Conference on Services Computing, SCC 2007, Salt Lake City, Utah, USA, pp. 701–702 (2007)
30. Lin, G., Dasmalchi, G., Zhu, J.: Cloud Computing and IT as a Service: Opportunities and Challenges. In: IEEE International Conference on Web Services, ICWS 2008, Beijing, China, p. 5 (2008)
31. Kaufman, L.M.: Data Security in the World of Cloud Computing. IEEE Security & Privacy 7(4), 61–64 (2009)
32. Menasce, D.A.: QoS issues in Web services. IEEE Internet Computing 6, 72–75 (2002)
33. Cancian, M.H., Rabelo, R.J., Wangenheim, C.G.v.: Uma proposta para elaboração de Contrato de Nível de Serviço para Software-as-a-Service (SaaS). In: Proceedings of the 8th International Information and Telecommunication Technologies Symposium, I2TS 2009 (2009)
34. Sabata, B., et al.: Taxonomy for QoS Specifications. In: Proceedings of the Third International Workshop on Object-Oriented Real-Time Dependable Systems, Washington. EUA (1997)
35. Mani, A., Nagarajan, A.: Understanding Quality of Service for Web Services (2002), http://www.ibm.com/developerworks/library/wsquality.html (10/07/2008)
36. Hosamani, M., Narayanappa, H., Rajan, H.: How to Trust a Web Service Monitor Deployed in an Untrusted Environment? In: Third International Conference on Next Generation Web Services Practices, NWeSP 2007, pp. 79–84 (2007)
37. Hongqi, L., Zhuang, W.: Research on Distributed Architecture Based on SOA. In: International Conference on Communication Software and Networks, ICCSN 2009, pp. 670–674 (2009)
38. Azuma, M.: Applying ISO/IEC 9126-1 Quality Model to Quality Requirements Engineering on Critical Software. In: Security Standards (2004)
39. Juric, M.B., et al.: Web Services and Java Middleware Functional and Performance Analysis for SOA. In: Digital EcoSystems and Technologies Conference, DEST 2007, Inaugural IEEE-IES, pp. 217–222 (2007)
40. Momm, C., Gebhart, M., Abeck, S.: A Model-Driven Approach for Monitoring Business Performance in Web Service Compositions. In: Fourth International Conference on Internet and Web Applications and Services, ICIW 2009, pp. 343–350 (2009)
41. Lee, K., et al.: QoS for Web Services: Requirements and Possible Approaches. W3C Working Group Note 25 (November 2003)