



UNIVERSITY OF SÃO PAULO

School of Arts, Sciences and Humanities

Technical Report PPgSI-003/2018
*Dictionary of Non-functional Requirements of
Business Process and Web Services*

Camila Faria de Castro, Marcelo Fantinato

December - 2018

The contents of this report are the sole responsibility of the authors.

Technical Report Series

PPgSI-EACH-USP

Arlindo Bértio St. 1000 - Ermelino Matarazzo - 03828-000.

São Paulo, SP, Brazil.

TEL: 55 (11) 3091-8197

<http://www.each.usp.br/ppgsi>

Dictionary of Non-functional Requirements of Business Process and Web Services

Camila Faria de Castro, Marcelo Fantinato¹

¹School of Arts, Sciences and Humanities
São Paulo – SP, Brazil

{camila.faria.castro, m.fantinato}@usp.br

***Abstract.** This technical report presents a dictionary of non-functional requirements of business processes and web services. This dictionary is a first step towards simplifying the decomposition of non-functional requirements from process to service level, a crucial task in a context of business process automation through service composition. The elaboration of this dictionary involved an exploratory study on software and web services quality models, being the ISO/IEC 25010 used as the main reference.*

1. Introduction

Business processes can be automated through the composition of distinct web services provided by business partners. Each web service involved in process automation implements a business function, or part of it, and must be performed under certain constraints, with the aim of contributing to the achievement of a business goal. Such web service constraints must be defined in terms of functional and non-functional requirements, the former refers to the activity itself and the latter refers to Quality of Service (QoS) attributes.

In the context of the automation of business processes through the composition of web services, the definition and decomposition of non-functional requirements of business process models into QoS attributes is a crucial task since the behavior of each web service continuously affects compliance of the business goal related to the corresponding business process. However, this decomposition may be challenging seeing that the Information Technology (IT) team involved needs to be able to identify the appropriate QoS attributes necessary to meet the given business non-functional requirements. This implies a strong reliance on the knowledge of an IT specialist who needs to be aware of both the business and technical domains to define QoS attributes relevant to web services aligned with the non-functional requirements of the business process being automated.

Within this motivational context, this technical report presents a dictionary of non-functional requirements of business processes and web services, aiming to support the process of non-functional requirements decomposition in a scenario of business process automation. The ISO/IEC 25010 product quality model was used as the main reference for the elaboration of the dictionary, being supported by additional studies on software engineering, software quality models and QoS attributes.

The remainder of this report is organized as follows. In section 2, basic information about non-functional requirements of business processes and web services is presented. The dictionary structure and its content are presented in section 3. Finally, the section 4 presents some concluding remarks.

2. Non-functional requirements

In software engineering, non-functional requirements are defined as constraints on the services or functions provided by a system, including time constraints, constraints on the development process and constraints imposed by standards and regulatory institutions [Sommerville 2010]. Contrary to functional requirements, which directly address user's necessities in terms of services to be provided by a system, they are more related to the behavior of the system while performing them. Typical examples of non-functional requirements are performance, security and reliability.

According to Sommerville [2010], non-functional requirements are often more critical than individual functional requirements. In the presence of system functionalities that do not necessarily meet their functional needs, system users can often adapt other existing functionalities. However, failing to meet a non-functional requirement may imply inefficiency of the system as a whole.

In Business Process Management (BPM), non-functional requirements correspond to a business perspective of quality constraints involved in one or more business activity for a business process model. For instance, given a business process describing a procedure for loan application in a financial institution, activities such as the analysis of customer's data and the issue of the contract may have time restrictions [Salles 2014]. It follows that process non-functional requirements are defined by business teams; whose specialists describe business restrictions on related processes.

In Service-Oriented Architecture (SOA), non-functional requirements related to web services focus mainly expressed by means of QoS attributes [Abramowicz et al. 2006]. QoS differ from process non-functional requirements as they are defined by technical teams, whose specialists describe technical restrictions on the service being provided. According to Garcia and de Toledo [2008], QoS publication supports the comparison of services with the same functionality and helps service composition. Typical examples of QoS attributes include restrictions on service's cost, execution duration, response time and scalability.

3. Dictionary of non-functional requirements

The dictionary describes non-functional requirements at process and service levels, which could be used by business and IT teams to define quality features of business process activities and web services, respectively.

The elaboration of the dictionary involved an exploratory study of main publications in literature with respect to software engineering and quality models related to software and web services as well as SOA and QoS attributes. The structure of characteristics and sub-characteristics of ISO/IEC 25010 Product Quality Model ISO/IEC [2010] was used as the main reference, due to its extensive use and acceptance by IT and business practitioners [Abramowicz et al. 2009]. This base structure was then expanded through additional research to broaden the scope of the proposed approach and consider more specific quality characteristics related to business and technical aspects of web service provisioning. Table 1 presents the main references used in the elaboration of the dictionary.

Table 1. Main references used to construct the dictionary.

ID	Research subject	References
1	QoS attributes for software and web services	ISO/IEC [2002, 2010], Lee et al. [2003], McCall et al. [1977], Menascé [2002], Miguel et al. [2014], OASIS [2005], O'Brien et al. [2005], Pettersson [2007], Sheoran and Sangwan [2015], Sommerville [2010]
2	Web services discovery using NFRs	Alrifai and Risse [2009], Liu et al. [2004], Ran [2003], Yoon et al. [2004], Zeng et al. [2003]
3	Other subjects	Abramowicz et al. [2006], Garcia and de Toledo [2008]

Considering the extent of publications in literature regarding non-functional requirements and QoS attributes, as well as the plurality of business areas, a small set of characteristics was considered to be part of the dictionary's scope. We believe that the selected non-functional requirements represent the most relevant quality features to be considered in the characterization of business process activities and web services, disregarding the business domain. Thus, the dictionary is generic enough to be considered in all contexts in which business processes are automated through web services.

Furthermore, many of the requirements in the dictionary were found in more than one reference – in some cases with the same name, and in others with a different name but equivalent definition. In the latter case, the requirements were considered as synonyms and were grouped as corresponding characteristics. Given a group of synonyms, the requirement whose name is best known to represent the given quality feature is emphasized in the dictionary, while the others are identified in brackets.

In terms of structure, the dictionary is formed by two sections: the first, for business processes' non-functional requirements; and the second, for web services' non-functional requirements. The structure of both sections includes four attributes:

- **ID:** a numerical identification of the non-functional requirement. Synonyms are identified and grouped using a unique ID.
- **Name:** the name of the non-functional requirement.
- **Definition:** a brief description of the non-functional requirement.
- **Reference:** the references from which the non-functional requirement was extracted from.
- **Measurement Unit:** [specifically for web services' non-functional requirements] identifies the primary unit used to measure a quantitative non-functional requirement. Measurement units are not present for all non-functional requirements found in literature. As a result, they are included only for some cases in the dictionary.

Many of the non-functional requirements in the dictionary were extracted from researches on software quality models and software engineering. Consequently, their definition in literature is generically related to any kind of product, software or system, which also means software for delivering web services. For the purpose of standardization, the definitions found in literature were adapted to the context of business processes and web services in the dictionary; i.e., when applicable, the terms *product*, *system* or *software* found in the definition were replaced by the terms *business process* or *service*.

The requirements' definition texts were also standardized according to the requirement's types. For requirements that can be measured in terms of percentage, the definition begins with "*Proportion of (...)*". For those that can be measured in terms of time

and quantity, the definition begins with “*Mean Time*” or “*Time*”, and “*Measurement of (...)*”, respectively. Finally, the definition of most of the remainder requirements begins with “*Degree to which (...)*” or “*Ease with which (...)*”, with some exceptions of specific requirements.

Moreover, the names of some requirements were also standardized with the use of the suffix *-ity*, which in English denotes the quality of being what an adjective or noun describes. One example of this standardization in the dictionary is the requirement *notifiability*, which is an adaptation of *notification*, found in the referred literature.

With respect to the business processes’ non-functional requirements, the first section of the dictionary describes eight attributes, which are presented in Table 2. These requirements were mainly extracted from ISO/IEC 25010 product quality model to define generic aspects of product quality to be selected by business areas. From the eight characteristics represented in the ISO/IEC’s model, seven were adapted to be considered in the dictionary as business processes’ non-functional requirements: *performance efficiency*, *compatibility*, *usability*, *reliability*, *security*, *maintainability* and *portability*. Only the characteristic *functional suitability* was not included in the scope of this dictionary since it refers to functional requirements.

Table 2. Dictionary of business processes’ non-functional requirements

ID	Name	Definition	Reference
1	Performance efficiency	Degree to which a business process can efficiently use an amount of resources (such as software, products, hardware and generic materials) under stated conditions.	ISO/IEC [2010]
2	Compatibility	Degree to which a business process can exchange information with other processes, and/or perform its activities while sharing the computing environment.	ISO/IEC [2010]
3	Usability	Degree to which a business process can be used by specified users to achieve specific goals with effectiveness, efficiency and satisfaction.	ISO/IEC [2010]
4	Reliability	Degree to which a business process performs specified activities under specified conditions for a period of time.	ISO/IEC [2010]
5	Security	Degree to which a business process can protect information and data from unauthorized access.	ISO/IEC [2010]
6	Maintainability	Degree of effectiveness and efficiency with which the activities of a business process can be modified.	ISO/IEC [2010]
7	Portability	Degree of effectiveness and efficiency with which a business process can be configured in an environment and transferred from one environment to another.	ISO/IEC [2010]
8	Compliance	Degree to which a business process is compliant with internal procedures of an organization and external guidelines.	Sommerville [2010]

ISO/IEC 25010 and other generic software quality models in literature describe quality characteristics exclusively from the product perspective. Thus, aiming at completeness for the business context, the business processes' non-functional requirements dictionary was supplemented with an attribute that addresses regulatory, legislative and operational aspects involved in process enactment. This non-functional requirement is called here *compliance* and was adapted from Sommerville's study on types of non-functional requirements for software systems [Sommerville 2010].

Regarding the web services' non-functional requirements, the sub-characteristics proposed in the ISO/IEC 25010 product quality model [ISO/IEC 2010], as well as the internal and external metrics of product quality measurement proposed by ISO/IEC 9126 [ISO/IEC 2002], were adapted to describe technical aspects of service provisioning to be defined by IT areas. The models were then refined by additional literature studies on software quality models and QoS attributes, aiming to consider technical and specific quality characteristics involved in service provisioning. As a result, the second section of the dictionary presents 93 web services' non-functional requirements extracted from different references, which are presented in Table 3.

The requirement *environment setup* was the only one created by the authors to address concerns related to the configuration of a service in a certain environment. The closest attribute found in literature regarding this concept was *installability* [ISO/IEC 2010]; however, we believe they refer to distinct concepts. Installability comes before as it refers to the ease with which a component of a service can be installed. After the installation, the configuration of the computing environment for the purpose of service provisioning is then characterized in terms of environment setup, which was not found in any of the literature studies analyzed.

4. Conclusion

In this technical report, we presented a dictionary of non-functional requirements of business processes and web services, which was elaborated based on researches in literature regarding software engineering, software product quality models and QoS attributes. We believe that elaborating a dictionary of non-functional requirements in business and service levels is the first step towards simplifying the decomposition of business goals to IT's constraints, supporting the strategic alignment between business and IT in a context of business process automation through web service composition. This decomposition is crucial for business as it turns possible for organizations to define non-functional requirements related to business processes, and systematically link them to QoS attributes and levels related to web services composing such business processes at the implementation level. Thus, future studies should be devoted to elaborating a decomposition framework, which hierarchically maps the interdependence relationships between the requirements defined in the dictionary.

Table 3. Dictionary of web services' non-functional requirements

ID	Name	Definition	Measurement Unit	Reference
1	Access Auditability	Proportion of user accesses to a service recorded in the access history database.	Percentage	ISO/IEC [2002]
2	Access Control	Degree to which a service restricts unauthorized user's access by using Web services security token or similar approach.		OASIS [2005]
3	Access Controllability	Proportion of illegal operations detected by the service, in comparison to the number of illegal operations defined in the specification.	Percentage	ISO/IEC [2002]
4	Accessibility	Degree to which a service can be used by people with distinctive characteristics and capabilities to achieve a specified goal in a context of use.		ISO/IEC [2010]
5	Accountability	Degree to which the actions of a user can be traced uniquely to the user.		ISO/IEC [2010]
6	Accuracy	Degree of precision to which a service provides right outcomes or effects.		Sheoran and Sangwan [2015]
7	Adaptability	Degree to which a service can effectively and efficiently be adapted for different hardware, software or other operational environments.		ISO/IEC [2010]
8	Analyzability	Degree of effectiveness and efficiency with which a modification to one or more parts of a service is assessed.		ISO/IEC [2010]
9	Auditability	Degree to which a service keeps sufficiently adequate records in the database to support financial or legal audits.		O'Brien et al. [2005]
10	Authenticity	Degree to which the identity of a subject or resource can be proved.		ISO/IEC [2010]
11	Availability	Proportion of total time during which a service is operational and accessible when required for use.	Percentage	ISO/IEC [2010]
12	Breakdown Avoidance	Proportion of breakdowns in production environment in comparison to the total number of failures.	Percentage	ISO/IEC [2002]
13	Capacity	Maximum limits of a service (i.e., concurrent users, stored data etc.) for which performance is guaranteed.		ISO/IEC [2010]
14	Change Impact	Proportion of detected adverse impacts after the modification of a service, in comparison to the number of modifications performed.	Percentage	ISO/IEC [2002]
15	Co-existence	Degree to which a service can perform its required functions while sharing environment and resources with other products.		ISO/IEC [2010]

Continues on the next page ...

Table 3. Dictionary of web services' non-functional requirements

ID	Name	Definition	Measurement Unit	Reference
16	Communication Com- monality	Degree to which a service uses standard protocols and interface routines for communication.		McCall et al. [1977]
17	Confidentiality	Degree to which a service ensures that its data is accessible only by authorized users.		ISO/IEC [2010]
	[Data Confidentiality]	Degree to which a service protects data against unauthorized disclosure.		OASIS [2005]
	[Privacy]	Degree to which access to sensitive data by unauthorized people can be controlled.		McCall et al. [1977]
18	Conformability	Degree to which a service uses the standard technology for Web Services (i.e. SOAP, WSDL e UDDI).		O'Brien et al. [2005]
19	Continued Use of Data	Proportion of data that can be used in the same way after service migration or replacement, in comparison to the number of total data items required to be used from old services.	Percentage	ISO/IEC [2002]
20	Customizability	Proportion of operations and procedures of a service that can be customized by the user.	Percentage	ISO/IEC [2002]
21	Data Commonality	Degree to which a service uses standard data representations.		McCall et al. [1977]
22	Data Corruption Pre- vention	Measurement of data corruption events identified and prevented by the service.	Corruption events	ISO/IEC [2002]
23	Data Exchangeability	Proportion of successful data transfers between the target service and other services.	Percentage	ISO/IEC [2002]
24	Data Granularity	Granularity of data a service provides in response to user requests.		O'Brien et al. [2005]
25	Ease of function learning	Mean time a user takes to learn to use a service function correctly.	Time	ISO/IEC [2002]
26	Ease of use	Degree to which a service is easy for users to operate and control.		ISO/IEC [2002]
27	Encryption	Degree to which a service's data is encrypted, making it unreadable without special knowledge.		Pettersson [2007]
28	Environment Setup	Ease with which a service can be configured to be used in a certain environment.		–

Continues on the next page ...

Table 3. Dictionary of web services' non-functional requirements

ID Name	Definition	Measurement Unit	Reference
29 Error correction	Mean time a user takes to correct an error on a task while using the service.	Time	ISO/IEC [2002]
30 Execution time	Time for a service to execute a sequence of activities and process a request.	Time	Lee et al. [2003]
31 Expandability	Degree to which a service can be expanded in terms of data storage requirements or computational functions.		McCall et al. [1977]
32 Extensibility [Extensibility]	Ease with which the service's capabilities can be extended without affecting other services or parts of the system. Degree to which a service can be expanded to new specification changes or other domains.		O'Brien et al. [2005] Sheoran and Sangwan [2015]
33 Failure Analysis Capability	Proportion of times a user was able to identify which operations caused a failure in a service, considering the total number of registered failures.	Percentage	ISO/IEC [2002]
34 Failure Avoidance	Proportion of fault patterns identified by the service to avoid critical and serious failures, considering the number of executed test cases of fault patterns during testing.	Percentage	ISO/IEC [2002]
35 Failure Resolution	Proportion of failure conditions which are resolved and do not reoccur against the number of detected failures.	Percentage	ISO/IEC [2002]
36 Fault Detection	Measurement of failures detected by a service in a certain period of time.	Detected failures	ISO/IEC [2002]
37 Fault tolerance [Error tolerance]	Degree to which a service operates as intended despite the presence of hardware or software faults. Degree to which a service provides continuity of operation under abnormal conditions.		ISO/IEC [2010] McCall et al. [1977]
38 Flexibility	Effort needed to change an operational service.		Sheoran and Sangwan [2015]
39 I/O Utilization	Measurement of estimated I/O utilization to complete a specified task.	Number of buffers	ISO/IEC [2002]

Continues on the next page . . .

Table 3. Dictionary of web services' non-functional requirements

ID Name	Definition	Measurement Unit	Reference
40 Installability	Degree of effectiveness and efficiency with which a component of a service can be successfully installed and/or uninstalled in a specified environment.		ISO/IEC [2010]
41 Integrity [Data Integrity]	Degree to which a service prevents unauthorized access to, or modification of, functions and data.		ISO/IEC [2010]
42 Interoperability [Integrability]	Degree to which a service ensures that data has not been altered or destroyed in an unauthorized manner.		OASIS [2005]
43 Latency time	Degree to which two or more components or services can exchange information and use the information that has been exchanged.		ISO/IEC [2010]
44 Learnability	Degree to which two or more components and services of a system are integrated.		Pettersson [2007]
45 Legislative	Round-Trip Delay (RTD) between the dispatch of a request and receive of a response for a service.	Time	Abramowicz et al. [2006]
46 Machine Independence	Degree to which a service can be used by specified users to achieve specified goals of learning with effectiveness, efficiency and satisfaction in a context of use.		ISO/IEC [2010]
47 Manageability	Legislative requirements that must be followed to ensure that the service operates within the law.		Sommerville [2010]
48 Maturity	Degree to which a service lends itself to efficient administration of its components.		McCall et al. [1977] McCall et al. [1977]
49 Mean Down Time	Degree to which a service meets needs for reliability under normal operation.		ISO/IEC [2010]
50 Mean Recovery Time	Mean time a service stays unavailable when a failure occurs before it gradually starts up.	Time	ISO/IEC [2002]
	Mean time a service takes to complete recovery from initial partial recovery.	Time	ISO/IEC [2002]

Continues on the next page . . .

Table 3. Dictionary of web services' non-functional requirements

ID Name	Definition	Measurement Unit	Reference
51 Mean Time Between Failures	Mean time between failures of a service in operation.	Time	ISO/IEC [2002]
52 Memorability	Degree to which users remember the operations of a service over time.	Time	Pettersson [2007]
53 Memory Utilization	Measurement of the estimated memory size a service will occupy to complete a specified task.	Bytes	ISO/IEC [2002]
54 Message Clarity	Proportion of messages with clear explanations provided by a service, from the total number of messages implemented.	Percentage	ISO/IEC [2002]
[Communicativeness]	Degree to which a service provides useful inputs and outputs which can be assimilated by users.		McCall et al. [1977]
55 Modifiability	Degree to which a service can be effectively and efficiently modified without introducing new defects or degrading the product quality.		ISO/IEC [2010]
[Changeability]	The ease with which a service can be modified.		Miguel et al. [2014] ISO/IEC [2010]
56 Modification Stability	Degree to which a service can avoid unexpected effects from modification of the software.		Miguel et al. [2014] ISO/IEC [2010]
57 Modification Time	Time required to modify a component of a service, from the moment a new requirement or modification is identified until its implementation and validation.	Time	McCall et al. [1977]
58 Modularity	Degree to which a service is composed of discrete components such that a change to one component has minimal impact on other components.		ISO/IEC [2010]
59 Non-repudiation	Degree to which action or events in a service can be proven to have taken place, so that they cannot be repudiated later.		ISO/IEC [2010]
60 Notifiability	Degree to which service providers notify changes on a service's functions and resources to an external quality manager or anyone who wishes to know about it.		OASIS [2005]
61 Operability	Degree to which a service has attributes that make it easy to operate and control, in conformance with user expectations.		ISO/IEC [2010]

Continues on the next page ...

Table 3. Dictionary of web services' non-functional requirements

ID Name	Definition	Measurement Unit	Reference
62 Operational	Operational process requirements that define how a service should be used.		Sommerville [2010]
63 Precision	Measurement of the frequency to which users encounter results with inadequate precision.	Inaccurate results / Time	ISO/IEC [2002]
64 Recoverability	Degree to which, in the event of an interruption or failure, a service can recover the data directly affected and re-establish the desired state of operation.		ISO/IEC [2010]
65 Regulatory	Regulatory requirements that set out what must be done for the service to be approved for use by a regulator.		Sommerville [2010]
66 Replaceability	Degree to which a service can replace another for the same purpose in the same environment.		ISO/IEC [2010]
67 Reputation	Measurement of a service's trustworthiness in terms of service quality, user's satisfaction and reliability on its operation.		Liu et al. [2004]
68 Resource utilization	Degree to which the amount and type of resources used meet requirements in service provisioning.		ISO/IEC [2010]
69 Response time	Time necessary to complete a certain service request, from the moment it is dispatched until a response is received.	Time	Lee et al. [2003]
[Average and maximum response time] [Execution duration]	Mean time needed for the packet of control data to get to the provider's server and return to the requester.	Time	Abramowicz et al. [2006]
70 Restore Effectiveness	Expected delay from the dispatch of a service request until the result is received by the client.	Time	Zeng et al. [2003]
71 Reusability	Proportion of successful restorations meeting the target restore time in comparison to the number of restorations required.	Percentage	ISO/IEC [2002]
72 Robustness	Degree to which an asset can be used in more than one service, or in the development of other services.		ISO/IEC [2010]
	Degree that represents the ability of the service to act properly even if some of the input parameters are missing or incorrect.		Abramowicz et al. [2006]

Continues on the next page ...

Table 3. Dictionary of web services' non-functional requirements

ID Name	Definition	Measurement Unit	Reference
[Exception handling]	Degree that represents how well a service handles exceptions on data inputs.		Ran [2003]
73 Scalability	Degree to which a service operates correctly, without degradation of other quality attributes, when the system is changed in size or in volume in order to meet users' needs.		O'Brien et al. [2005]
74 Security Audit Trail	Degree to which a service records a log of attempted attacks in order to evaluate its vulnerability.		OASIS [2005]
75 Self-Descriptiveness	Ease with which a service's functions and documentation can be understood by humans.		McCall et al. [1977]
76 Simplicity	Degree to which the functions of a service are implemented in the most understandable manner, avoiding practices which increase complexity.		McCall et al. [1977]
77 Software System Independence	Degree of service dependency on the software environment, including operating systems, utilities, input/output routines, etc.		McCall et al. [1977]
78 Stability	Degree to which a web service is able to provide continuous, consistent and recoverable services despite of increased throughput, congestion, system failures, natural disaster and intentional attack from users.		OASIS [2005]
79 Successability	Degree to which web services yield successful results over request messages.	Percentage	OASIS [2005]
80 Supported Standard	Degree to which a service operation complies with standards (e.g. industry specific standards).		Ran [2003]
81 Test Coverage	Degree to which a service is effectively tested in terms of source code statements executed.	Percentage	Sommerville [2010]
82 Test Maturity	Proportion of test cases that have been successfully performed during testing in comparison to the total number of test cases.	Percentage	ISO/IEC [2002]
83 Testability	Degree of effectiveness and efficiency with which test criteria can be established for a service and tests can be performed to determine whether those criteria have been met.		ISO/IEC [2010]

Continues on the next page ...

Table 3. Dictionary of web services' non-functional requirements

ID	Name	Definition	Measurement Unit	Reference
84	Throughput	Measurement of the number of web service requests served in a given time interval.	Processed requests / time	Lee et al. [2003]
	[Maximum Throughput]	Maximum number of services that a platform providing web services can process for a unit time.		OASIS [2005]
85	Time behavior	Degree to which the response and processing times and throughput rate meet requirements in service provisioning.		ISO/IEC [2010]
86	Timeliness	Degree to which a service meets deadlines, i.e., to process a request in a deterministic and acceptable amount of time.		O'Brien et al. [2005]
87	Tolerance	Degree to which a service accepts different forms of the same information as valid or withstand a degree of variation in input without malfunction or rejection.		McCall et al. [1977]
88	Transaction time	Time that passes while the web service is completing one complete transaction. This concept may depend on the definition of a web service transaction.	Time	Lee et al. [2003]
89	Transferability	Ease of moving a computer program from one computing environment to another.		McCall et al. [1977]
90	Transmission Utilization	Estimated amount of transmission resources utilized by a service.	Bits/time	ISO/IEC [2002]
91	Understandability	Degree to which a user understands the logical concept of a service and its applicability.		Pettersson [2007]
	[Appropriateness Recognizability]	Degree to which users can recognize whether a service is appropriate for their needs.		ISO/IEC [2010]
92	User error protection	Degree to which a system protects users against making errors.		ISO/IEC [2010]
93	User interface aesthetics	Degree to which a user interface enables satisfying interaction for the user, such as the use of color and the nature of the graphical design.		ISO/IEC [2010]

References

- Abramowicz, W., Haniewicz, K., Hofman, R., Kaczmarek, M., and Zyskowski, D. (2009). Decomposition of SQuaRE-based requirements for the needs of soa applications. In *Proc. of the Int. Conf. on Advances in Communication Technologies and Engineering Science*, pages 81–94. Springer.
- Abramowicz, W., Kaczmarek, M., and Zyskowski, D. (2006). Duality in web services reliability. In *Proc. of the Advanced Int. Conf. on Telecommunications and Int. Conf. on Internet and Web Applications and Services*, pages 165.1–165.6. IEEE.
- Alrifai, M. and Risse, T. (2009). Combining global optimization with local selection for efficient QoS-aware service composition. In *Proc. of the 18th Int. Conf. on World Wide Web*, pages 881–890. ACM.
- Garcia, D. Z. G. and de Toledo, M. B. F. (2008). Quality of service management for web service compositions. In *Proc. of the 11th IEEE Int. Conf. on Computational Science and Engineering*, pages 189–196. IEEE.
- ISO/IEC (2002). ISO/IEC 9126 software product quality, IEEE Std 9126:2002.
- ISO/IEC (2010). ISO/IEC 25010 system and software quality models.
- Lee, K., Jeon, J., Lee, W., Jeong, S.-H., and Park, S.-W. (2003). QoS for web services: Requirements and possible approaches. Technical Report NOTE-ws-qos-20031125, W3C Korea Office.
- Liu, Y., Ngu, A. H., and Zeng, L. Z. (2004). QoS computation and policing in dynamic web service selection. In *Proc. of the 13th Int. World Wide Web Conf.*, pages 66–73. ACM.
- McCall, J. A., Richards, P. K., and Walters, G. F. (1977). Factors in software quality. volume-iii. preliminary handbook on software quality for an acquisition manager. Technical Report RADC-TR-77-369, Defense Technical Information Center.
- Menascé, D. A. (2002). QoS issues in web services. *IEEE Internet Comp.*, 6(6):72–75.
- Miguel, J. P., Mauricio, D., and Rodriguez, G. D. (2014). A review of software quality models for the evaluation of software products. *Int. J. of Software Engineering & Applications*, 5(6):31–54.
- OASIS (2005). Quality model for web services.
- O’Brien, L., Bass, L., and Merson, P. (2005). Quality attributes and service-oriented architectures. Technical Report CMU/SEI-2005-TN-014, Software Engin. Inst., Carnegie Mellon Univ.
- Pettersson, A. (2007). Service-oriented architecture (SOA) quality attributes – a research model. Master’s thesis, Dept. of Informatics, Lunds Univ., Sweden.
- Ran, S. (2003). A model for web services discovery with QoS. *ACM SIGecom Exchanges*, 4(1):1–10.
- Salles, G. B. M. (2014). Tratamento de acordos em processos de negócio: Em busca de alinhamento estratégico. Master’s thesis, Information Systems Program, School of Arts, Sciences and Humanities, Univ. of São Paulo, Brazil. (in Portuguese).
- Sheoran, K. and Sangwan, O. P. (2015). An insight of software quality models applied in predicting software quality attributes: A comparative analysis. In *Proc. of the 4th Int. Conf. on Reliability, Infocom Technologies and Optimiz.*, pages 1–5. IEEE.
- Sommerville, I. (2010). *Software Engineering*. Pearson, Addison-Wesley, 9 edition .
- Yoon, S., Kim, D., and Han, S. (2004). WS-QDL containing static, dynamic, and statistical factors of web services quality. In *Proc. of the IEEE Int. Conf. on Web Services*, pages 808–809. IEEE.

Zeng, L., Benatallah, B., Dumas, M., Kalagnanam, J., and Sheng, Q. Z. (2003). Quality driven web services composition. In *Proc. of the 12th Int. Conf. on World Wide Web*, pages 411–421. ACM.